IBM Security Guardium V10.6

目次

IBM Security Guardium V10.6

IBM Security Guardium 資料のページにようこそ。このページでは、IBM Guardium のインストール方法、保守方法、使用方法に関する情報を参照することができます。

始めに

製品の概要
製品に関する特記事項
新機能
リリース情報
インストール
アップグレード

トラブルシューティングとサポート

Guardium サポート・ホーム
Guardium サポート・リソース
Guardium サポート・ビデオ
Guardium の IBM developerWorks Answers

詳細情報

IBM Security Learning Academy
IBM データ・セキュリティーと保護
IBM developerWorks Guardium コミュニティー
Guardium Tech Talk ビデオ

製品の概要

Guardium® ソリューションの製品およびリリース情報。

- IBM Guardium
  IBM Guardium は、データベース、データウェアハウス、ビッグデータ環境 (Hadoop など) からの情報漏えいを防ぎ、情報の整合性を確保し、異種混合環境全体のコンプライアンス制御を自動化します。
- このリリースの新機能
  新機能、機能、および機能拡張。
- リリース情報
  最新の機能と機能拡張、システム要件、そしてアップグレード、インストール、およびサポート情報について説明します。

IBM Guardium

IBM Guardium は、データベース、データウェアハウス、ビッグデータ環境 (Hadoop など) からの情報漏えいを防ぎ、情報の整合性を確保し、異種混合環境全体のコンプライアンス制御を自動化します。

データベース、ビッグデータ環境、およびファイル・システム内の構造化データおよび非構造化データを脅威から保護し、コンプライアンスを確保します。

構造化データおよび非構造化データのトラフィックを継続してモニターすること、および機密データへのアクセスに関するポリシーを企業規模で実施することができる、拡張が容易なプラットフォームを提供します。

安全な中央監査リポジトリーと統合ワークフロー自動化プラットフォームの組み合わせにより、多種多様な要件に関するコンプライアンス検証アクティビティーが簡素化されます。

IT 管理ソリューションおよびその他のセキュリティー管理ソリューションとの統合を活用して、企業全体に対する包括的なデータ保護を実現します。

これらの目的は、拡張が容易なプラットフォームを使用して、各種のデータベースおよび文書共有インフラストラクチャーを継続してモニターできるようにすること、および機密データへのアクセスに関するポリシーを企業全体に対して実施できるようにすることです。セキュリティーを最大化するよう設計された中央監査リポジトリーと、統合コンプライアンス・ワークフロー自動化アプリケーションの組み合わせにより、製品で多種多様な要件に関するコンプライアンス検証アクティビティーを簡素化できます。

IBM Security Guardium は、重要なデータを保護するように設計されています。 Guardium は、包括的なデータ保護プラットフォームであり、セキュリティー・チームが機密データ環境 (データベース、データウェアハウス、ビッグデータ・プラットフォーム、クラウド環境、ファイル・システムなど) の状況を自動的に分析して、リスクを最低限に抑え、内外の脅威から機密データを保護し、データ・セキュリティーに影響を与える可能性がある IT の変更にシームレスに適合できるようにします。 Guardium は、データ・セ ンターの情報の整合性を確保して、コンプライアンス管理を自動化する上で役立ちます。

IBM Security Guardium ソリューションは、以下に示す 2 つのバージョンで提供されます。

- IBM Security Guardium データベース・アクティビティー・モニター (DAM)
- IBM Security Guardium ファイル・アクティビティー・モニター (FAM): Guardium のファイル・アクティビティーのモニター機能を使用して、ファイル・サーバーに対するモニター機能を拡張します。

IBM Guardium 製品は、データベースやファイルからのデータ漏えいを防止するためのシンプルで堅固なソリューションを提供する製品であり、データ・センターの情報の保全性を確保し、コンプライアンス制御を自動化します。

Guardium 製品の支援により、以下を行うことができます。

- データベースを見つけ出し、そのデータベースに入っている機密情報を検出して分類する処理を自動化できます。
- データベースの脆弱性と構成の問題点を自動的に評価できます。
- 推奨された変更を実装した後に、構成が確実にロックダウンされるようにできます。
- 機密データにかかわるデータベース・トランザクションを詳細なレベルまで可視化できます。
- エンタープライズ・アプリケーションを経由してデータに間接的にアクセスするエンド・ユーザーのアクティビティーを追跡できます。
- 機密データのアクセス、データベースの変更制御、特権ユーザーの操作などに関する多種多様なポリシーを適用し、実施状況をモニターできます。
- 異種混合の多数のシステムやデータベースのための、一元管理型でセキュアな、単一の監査リポジトリーを作成できます。
- レポートの作成と配布、コメントやシグニチャーの取り込みなどに関するコンプライアンス監査プロセス全体を自動化できます。

Guardium ソリューションは、簡単に使用および拡張できるように設計されています。単一のデータベース、または企業各所にある数千の各種データベースに対して構成できます。

このソリューションは、IBM® が提供する事前構成済みアプライアンスとして、またはプラットフォームにインストールされるソフトウェア・アプライアンスとして利用できます。インストール後、オプションの機能をシステムに簡単に追加できます。

Guardium のデータベース・セキュリティー・ソリューションには、以下の主要な機能領域があります。

- 脆弱性評価。データベース製品で既知の脆弱性を検出するだけでなく、複雑なデータベース・インフラストラクチャーの完全な可視性を提供し、構成の誤りを検出するとともに、それらのリスクを評価して緩和します。
- データのディスカバリーと分類。分類だけでは保護は提供されませんが、データの重要性とコンプライアンス要件に基づいて、さまざまなデータに応じた適切なセキュリティー・ポリシーを定義する際の重要な最初のステップとなります。
- データ保護。Guardium は、保存中および転送中のデータ暗号化、静的および動的データ・マスキングなど、データの保全性と機密性を保護するためのテクノロジーに対応します。
- モニターおよび分析。これには、データベースのパフォーマンス特性のモニター、および各インスタンスのすべてのアクセスおよび管理アクションの完全な可視性が含まれます。これに加え、高度なリアルタイム分析、異常検出、および Security Information and Event Management (SIEM) 統合を使用できます。
- 脅威に対する保護。これは、分散型のサービス妨害 (DDoS) や SQL インジェクションなどのサイバー攻撃から保護し、パッチが適用されていない脆弱性を緩和するなど、データベース固有のセキュリティー対策を講じる手法を指します。
- アクセス管理。データベース・インスタンスに対する基本的なアクセス制御を超える機能を提供します。より高度かつ動的なポリシー・ベースのアクセス管理を焦点としたレーティング・プロセスでは、過剰なユーザー特権の識別と削除、共有アカウントとサービス・アカウントの管理、疑わしいユーザー・アクティビティーの検出とブロックに対応できます。
- 監査およびコンプライアンス。これには、ネイティブ機能を超えた高度な監査メカニズム、複数のデータベース環境にわたる監査およびレポート作成の一元化、職務分離の適用、およびフォレンジック分析とコンプライアンス監査のサポート・ツールが含まれます。
- パフォーマンスおよびスケーラビリティー。本質的にはセキュリティー機能ではありませんが、すべてのデータベース・セキュリティー・ソリューションが高負荷に耐え、パフォーマンス・オーバーヘッドを最小限にし、高可用性構成でのデプロイメントをサポートするためには重要な要件です。

Guardium 製品ファミリーについて詳しくは、https://www.ibm.com/security/data-security/guardium を参照してください。

親トピック: 製品の概要

このリリースの新機能

新機能、機能、および機能拡張。

IBM Security Guardium V10.6

機密データのディスカバーによる分類ポリシー・ビルダーの置き換え
: 機密データのディスカバー・ツールは、分類ポリシー・ビルダーを置き換えるものであり、ポリシーと監査プロセスの両方を再使用するためのサポートを追加します。詳しくは、機密データのディスカバーを参照してください。

外部 S-TAP
: Guardium 外部 S-TAP では、データベース・サーバーに検査エージェントをインストールすることなく、クラウドおよびオンプレミスのデータベース・サーバーのトラフィックをインターセプトします。このコンポーネントは、Docker イメージとして利用でき、どのサポート対象環境にでもデプロイできます。詳しくは、外部 S-TAP を参照してください。

GDPR 用の FAM
: GDPR アクセラレーター用の新しい FAM は、ポリシー、ディスカバリーと分類、および GDPR に対する準備状況に関するレポートを提供します。GDPR FAM ユーザー・ロールの有効化について詳しくは、ロールについてを参照してください。

FAM for NAS and SharePoint
: FAM は、SharePoint サーバーおよび Network Attached Storage デバイスをモニターおよび監査するための新しい機能を追加します。詳しくは、NAS および SharePoint のファイル・アクティビティー・モニターを参照してください。

異常値検出
: 異常値検出は、デフォルトのシナリオ・ベースのダッシュボード、複数の中央マネージャーが含まれた環境のサポート、有効化および無効化のための簡素化された API、および強化された異常値マイニングの状況ページを備えています。詳しくは、Outliers Detectionを参照してください。

データのポリシー・ビルダー
: 新しいポリシー・ビルダーは、プロパティーによるソートおよびフィルタリング、ポリシーおよびルール・プロパティーが含まれた CSV のエクスポート、専用コレクション・プロファイル・ルール・タイプなどの機能拡張によって、ポリシー管理を簡素化します。詳しくは、ポリシーを参照してください。

照会 - レポート・ビルダー
: 新しい照会 - レポート・ビルダーは、既存のレポート・ビルダーとクエリー・ビルダーを結合して、ワークフローを簡素化します。詳しくは、照会 - レポート・ビルダーの使用を参照してください。

セッション・レベル・ポリシー
: セッション・レベル・ポリシーのサポートにより、セッションまたはアクセス・レベル情報で起動するルールのポリシー処理が向上します。

マルチスレッド化の脆弱性評価サポート

マルチスレッド化の脆弱性評価サポートにより、複数のセキュリティー評価を並行でスケジュールして実行することで、CPU コアごとに 2 つの並行スキャンを実行できるようになり、スキャン時間を短縮できます。詳しくは、マルチスレッド・アセスメントを参照してください。

GIM の機能拡張
: 「クライアント別の設定」の機能拡張: GIM のグループ・ビルダー、API 生成機能、およびバンドル選択時のフィルタリングの拡張。クライアント別の設定を参照してください。

Disk and Database Health Analyzer
: Guardium は、ディスク (/var) 上の DB サイズとファイルをモニターします。今後 14 日間でディスク (/var) 上のサイズまたはファイルが 50% に達する可能性がある DB を識別すると、アラートを送信して、/var 上の最大の表または最大のファイルを示します。アラートは、中央マネージャーの適用状態ダッシュボードにも表示されます。

Unix S-TAP の機能拡張
: S-TAP 制御イベント・ログには、S-TAP のオンライン状態を維持するためにユーザーが構成を保存したときに S-TAP が変更したユーザー構成エラーが示されます。イベント・ログを使用して値を受け入れるか、変更することができます。**Linux システムおよび UNIX システム: GUI からの S-TAP の構成**を参照してください。

親トピック: 製品の概要

リリース情報

最新の機能と機能拡張、システム要件、そしてアップグレード、インストール、およびサポート情報について説明します。

新機能と機能拡張の説明

Guardium の最新バージョンには、数多くの新機能および既存の機能に対する機能拡張が含まれています。詳細なリリース・ノートについては、以下のリンク先を参照してください。Guardium V10.6 リリース・ノート

発表資料

以下の情報については、IBM Guardium の発表資料を参照してください。

- 詳細な製品説明 (新機能の説明を含む)
- 製品の位置付けに関する説明
- パッケージと発注方法に関する情報
- 多国語間の互換性情報

システム要件

Guardium V10.6 のシステム要件およびサポートされるプラットフォームについては、System Requirements/ Platforms supported for IBM Guardium v10.6 を参照してください。

Guardium のアップグレード

Guardium の最新バージョンへのアップグレードについては、Guardiumシステムのアップグレードを参照してください。

Guardium のインストール

最新バージョンの Guardium のインストールについては、Guardiumシステムのインストールを参照してください。

既知の問題

既知の問題は文書化されており、IBM サポート Web サイトで確認できます。

問題が発見および解決されると、IBM サポート Web サイトが更新されます。ダウンロードや詳細なシステム要件に関する資料などに加えて、IBM サポート Web サイトを検索することにより、問題の回避策や解決策を素早く見つけることができます。

サポートのライフサイクル

Guardium ソフトウェアの古いバージョンを使用している場合、アップグレードすることを早めに計画してください。IBM 製品のサポート終了日に関する情報は、IBM Software Support Lifecycle Web サイトで確認できます。

親トピック: 製品の概要

始めに

- コンポーネントおよびトポロジー
  Guardium アプライアンス、エージェント、およびその他のコンポーネントについて説明します。
- ユーザー・インターフェースの概要
  Guardium ユーザー・インターフェースの基礎 (初回ログイン、バナーおよびナビゲーション・メニュー、ユーザー・インターフェース、データ検索など) について説明します。
- ユーザー・インターフェースのカスタマイズ
  Guardium では、特定のユーザーとロールについて、ナビゲーション・メニューをカスタマイズすることができます。
- モニターおよびコンプライアンスのクイック・スタート
  モニター・エージェントをデータベース・サーバーにデプロイし、データベース・モニターをセキュリティー基準と規制に準拠するように構成する方法について説明します。

- システム・ビュー
  「システム・ビュー」は、多くのユーザーにとってのデフォルトの初期ビューです。 これにより、システム状況の重要な要素を確認できます。
- データ・アクティビティーのモニター
  Guardium データ・アクティビティー・モニターで使用される重要なセキュリティー概念について説明します。
- ファイル・アクティビティー・モニター
  ファイル・アクティビティー・モニターはサーバー上の機密データをディスカバーします。また、事前定義の定義またはユーザー定義の定義を使用してコンテンツを分類します。さらに、データ・アクセスに関するルールおよびポリシーや、ルールが満たされたときに実行されるアクションを構成します。
- Big Data Intelligence
  Guardium Big Data Intelligence (GBDI) は、収集したデータをより長期間にわたって保管し、データ・セキュリティーおよびコンプライアンスに関するレポートや洞察への直接的かつリアルタイムのアクセスを提供します。
- 重要な概念とツール
  Guardium の管理に関連した重要な概念について説明します。

関連情報:
Guardium の概要、アーキテクチャー、およびユーザー・インターフェース (ビデオ)

コンポーネントおよびトポロジー

Guardium アプライアンス、エージェント、およびその他のコンポーネントについて説明します。

Guardium コンポーネント

- アプライアンス
  - コレクター: コレクターは、データベース・アクティビティーのキャプチャーおよび分析をリアルタイムで実行し、それをログに記録してさらに分析したり、アラートで使用したりします。
  - アグリゲーター: Guardium アグリゲーターは、複数の Guardium コレクターからの情報と、オプションで他のアグリゲーターからの情報を収集してマージします。そして、環境全体の包括的なビューを作成します。収集プロセスと統合プロセスにより、Guardium はエンタープライズ・レベルのレポートを簡単に生成できます。例えば、大規模なエンタープライズ環境では、さまざまな地理的位置または業務単位をモニターするために、複数の Guardium システムが使用される場合があります。このシナリオでは、すべての地理的位置または業務単位にわたってデータベース使用量を確認するために、すべての Guardium システムからのデータを 1 箇所に収集することが有用である場合があります。 これは、複数のコレクターから単一のアグリゲーターにデータをエクスポートすることによって実現できます。 このアグリゲーターから実行されるレポート、アセスメント、および監査プロセスは、この環境全体から収集されたデータを反映するようになります。
  - 中央マネージャー: 中央マネージャー (CM) はアグリゲーターで有効にされている特殊な機能です。この構成では、1 つの Guardium システムが、単一コンソールから Guardium 環境全体を制御およびモニターする中央マネージャーとして指定されています。この構成では、コレクターおよびアグリゲーターは管理対象ユニットと呼ばれます。 一部のアプリケーション (監査プロセス、照会、ポートレットなど) は、管理対象ユニットと中央マネージャーのいずれからでも実行できますが、アプリケーション定義は中央マネージャーに格納されます。一元管理によって、Guardium は、複数のアグリゲーターがデータ・リポジトリーを中央マネージャーにマージする階層的な統合をサポートできるようになります。 複数レベルのビューを提供するには、これが役立ちます。例えば、さまざまな Guardium アグリゲーターが、さまざまな地理的位置に割り当てられている場合、一元管理ユニットは、すべての地理的位置にまたがる単一のグローバル・ビューに、すべてのアグリゲーターの内容をマージすることができます。統合および一元管理を参照してください。
- エージェント (必須および最も一般的):
  - ソフトウェア TAP エージェント (S-TAP):S-TAP の管理.
  - Guardium Installation Manager エージェント (GIM): GIM エージェントはデータベースおよびファイル・サーバーにインストールされます。これにより、エージェントのインストールおよびエージェントの更新と構成変更が容易になります。『Guardium Installation Manager』を参照してください。
  - 変更監査システム・エージェント (CAS): CAS エージェントはデータベース・サーバーにインストールされます。これにより、データベース・サーバーでの構成ファイルの変更監査情報などのキャプチャーが行われます。構成監査システム (CAS)を参照してください。
  - インスタンス・ディスカバリー・エージェント: インスタンス・ディスカバリー・エージェントはデータベース・サーバーにインストールされ、データベース、リスナー、およびポートの情報を Guardium に送信します。
- データ・ソース: Guardium データ・ソースは、特定のデータベース・インスタンスを識別します。データ・ソースへのアクセスは、データ・ソースに割り当てられているロールと、そのデータ・ソースを使用するアプリケーションに割り当てられているロールに基づいて制限される場合があります。 例えば、「値変更監査」アプリケーションは、特権レベルが低い他のアプリケーションについては適切ではない、ハイレベルの管理アクセス権限を必要とします。
- 検査エンジン: コンポーネントでもエージェントでもない検査エンジンは、データベース・プラットフォームおよび S-TAP が S-TAP ホスト (データベース・サーバー) 上でモニターするインスタンスを指定する必須の構成です。多くの場合、1 つの S-TAP に多数の検査エンジンがあります。

Guardium のトポロジー

基本的なスタンドアロン・アーキテクチャー

最も基本的なアーキテクチャーは、1 つのデータ・センター内の複数のデータベースをモニターし、1 つのスタンドアロン・コレクター・アプライアンスと、モニター対象データベース・サーバーにインストールされた複数の Guardium S-TAP エージェントで構成されます。S-TAP エージェントは、関連するデータベース・アクティビティーを収集し、分析、構文解析、およびロギングのために 1 つの Guardium コレクターに送信します。

中規模のアーキテクチャー

中規模のアーキテクチャーは、データ・センター間の多数のデータベースをモニターします。これは、複数のコレクター・アプライアンスと、各データ・センター内のモニター対象データベース・サーバーにインストールされている多数の S-TAP エージェントで構成されます。S-TAP エージェントは、関連するデータベース・アクティビティーを収集し、分析、構文解析、およびロギングのために Guardium コレクターに送信します。 コレクターは、モニターされているアクティビティーをアグリゲーター・アプライアンスに集約して、中央レポートを作成します。この例では、アグリゲーター・アプライアンスは、アクセス管理、パッチの適用、およびメタデータ・リポジトリーなどの統合管理機能を有効にするソリューションの中央管理アプライアンスとしても機能しています。

エンタープライズ・アーキテクチャー

エンタープライズ・アーキテクチャーは、複数のデータ・センターおよび大陸にまたがる多数のデータベースをモニターします。このアーキテクチャーの例は、多くのコレクター・アプライアンス、および複数のデータ・センターにまたがるメインフレームや分散データベース・サーバーにインストールされている多数の S-TAP エージェントで構成されます。 S-TAP エージェントは、関連するデータベース・アクティビティーを収集し、分析、構文解析、およびロギングのために Guardium コレクターに送信します。コレクターは、モニターされているアクティビティーを個々のアグリゲーター・アプライアンスに集約して、中央レポートを作成します。 専用の中央マネージャー・アプライアンスは、アクセス管理、パッチの適用、およびメタデータ・リポジトリーなどの統合管理機能を提供します。

親トピック: 始めに

ユーザー・インターフェースの概要

Guardium ユーザー・インターフェースの基礎 (初回ログイン、バナーおよびナビゲーション・メニュー、ユーザー・インターフェース、データ検索など) について説明します。

ナビゲーション

Guardium ユーザー・インターフェースに初めてログインする際には、バナーとナビゲーション・メニューの 2 つのメインメニューがあります。

ナビゲーション・メニューを展開/省略するには、シェブロン・アイコン をクリックします。ナビゲーション・メニューを完全に非表示にするには、表示/非表示アイコン をクリックします。

画面の初期レイアウトは、適用されているライセンス、ロールに基づいて許可されるアクセス、マシン・タイプ、可視性要因によって決まります。ロールの例としては、ユーザー、管理者、アクセス・マネージャー、CLI などがあります。ロールは、ユーザーに特定のアクセス権を付与するためにユーザーとアプリケーションに割り当てられます。

サポート対象の Web ブラウザー

http://www-01.ibm.com/support/docview.wss?uid=swg10719695を参照してください。

バナー・メニュー

バナーには以下の項目が含まれています。

項目		記述
システム時刻クロック		Guardium システム上の世界時。
To-do リスト		ユーザーによってフィルタリング可能な「監査プロセスの To-do リスト」と、「保留中の結果がない処理」が含まれています。
ヘルプ		製品ヘルプを開くには、「ヘルプ」 > 「Guardium ヘルプ」をクリックします。 Guardium システムに関する情報 (バージョン番号など) を表示するには、「ヘルプ」 > 「Guardium バージョン情報」をクリックします。 操作している画面や機能に固有のヘルプ内容を表示するには、画面のペインに組み込まれている小さいヘルプ・アイコンをクリックします。 注: どちらのヘルプ・アイコンをクリックしても、同じインフォメーション・センターに移動します。ここで、すべてのヘルプ内容を検索してアクセスすることができます。
ユーザー・インターフェース/データ/ファイル		「データ」または「ファイル」を選択して、 をクリックし、「調査ダッシュボード」を開きます。 「ユーザー・インターフェース」を選択して、UI を検索します。例えば、「ポリシー・ビルダー」を検索したい場合は、「ユーザー・インターフェース検索」で「`policy builder`」という入力を開始します。任意の結果をクリックすると、ユーザー・インターフェースのその部分に移動します。
アカウント・タイプ		ログインに使用したアカウントのタイプを示します。アカウントの詳細 (パスワードまたは名前) の編集、UI レイアウトのカスタマイズ、Guardium からの確実なサインアウトを行います。
マシン・タイプ		稼働中のマシンのタイプ (スタンドアロン、管理対象ユニット、中央マネージャー、アグリゲーターなど) を示します。

バナー・メニューには、重要な始動メッセージ (RAM メモリー不足、クイック検索メモリーおよび CPU 4 コアの最小要件、証明書の有効期限切れ、一元管理の障害、SSLv3 が有効または無効、ライセンスなし、など) も含まれます。

注: Guardium では SSLv3 を無効にすることを推奨しています。 ただし、最新リリースがインストールされていない旧バージョンの Guardium の利用時、SSLv3 が無効であると、中央マネージャーと管理対象ユニットの間で一元管理機能が低下します。

ナビゲーション・メニュー

ナビゲーション・メニュー内の各アイコンは、Guardium セキュリティー・ライフサイクルの 1 フェーズを表しています。任意のアイコンをクリックすると、そのフェーズが展開され、フェーズに含まれる各コンポーネントが表示されます。ライフサイクルに基づくナビゲーション・メニューは、ユーザー・インターフェースをナビゲートする機能の 1 つであり、すべてのロールにわたって一貫性があります。メニュー項目をカスタマイズし、ロールに基づいてメニュー項目を表示または非表示にできます。

フェーズ		記述
セットアップ		ネットワーク設定の構成、サービス状況の確認、データ・ソース定義、グループ、別名、およびアラートの設定を行います。
管理		現行環境の全体的な正常性、S-TAP、データ、モジュール、メンテナンス、レポートを管理します。
ディスカバー		現行環境に導入された新規データベースを自動的にディスカバーし、機密データの検索と分類を行います。
強化		脆弱性評価で現行環境の現在の弱点を評価し、構成監査システム (CAS) で現行環境に加えられた変更をモニターします。
調査		データベース・アクティビティーをモニターし、現行環境の各部に疑わしいアクティビティーがないか調査します。
保護		疑わしいアクティビティーをブロックし、データへの無許可アクセスを防止するデータ・セキュリティー・ポリシーを使用して、現行環境を保護します。 ポリシーについて詳しくは、『ポリシー』を参照してください。
順守		監査プロセスと細分度の高いレポート作成によって、コンプライアンス・イニシアチブを実現します。
レポート		独自のレポートを作成するか、各種の事前定義レポートのいずれかを使用して、現行環境の任意の部分に関するレポートを作成します。レポートについて詳しくは、レポートを参照してください。

マイ・ダッシュボード		関心の高いレポートを容易にレビューできるように、独自のダッシュボードを作成します。ダッシュボードについて詳しくは、ダッシュボードの作成およびレポートの追加を参照してください。

共通して使用されるアイコン

以下の一連のアイコンは、Guardium の多くのファインダー・アプリケーションとビルダー・アプリケーションで共有されます。

アイコン	記述
新規	新規項目 (グループやデータ・ソース定義など) を作成します。
変更	項目を変更します。 注: 項目を変更する際のベスト・プラクティスは、その項目のコピーを作成し、そのコピーに変更を加えることです。
コピー	項目を複製し、その項目のコピーを作成します。
削除	項目を削除します。

親トピック: 始めに

ユーザー・インターフェースのカスタマイズ

Guardium では、特定のユーザーとロールについて、ナビゲーション・メニューをカスタマイズすることができます。

「ナビゲーション・メニューのカスタマイズ (Customize Navigation Menu)」ツールと「ユーザー/ロールのカスタマイズ」ツールにより、ナビゲーション・メニューの内容と構成を簡単に変更することができます。これらのツールは、複数の場所で使用することができます。

- すべてのユーザーは、Guardium バナーの「ユーザー」メニューを開いて「カスタマイズ」を選択することにより、自分専用のナビゲーション・メニューをカスタマイズすることができます。
- 管理ユーザーは、「ユーザー」メニューを開いて「ユーザー/ロールのカスタマイズ」を選択するか、「設定」 > 「ツールとビュー」 > 「ユーザー/ロールのカスタマイズ」にナビゲートすることにより、他のユーザーやロールのナビゲーション・メニューをカスタマイズすることができます。
- *accessmgr* としてログインしたユーザーは、「アクセス」 > 「アクセス管理」にナビゲートして「ロール・ブラウザー」を選択し、「ナビゲーション・メニューのカスタマイズ (Customize Navigation Menu)」リンクをクリックすることにより、他のユーザーやロールのナビゲーション・メニューをカスタマイズすることができます。

ツールでのカスタマイズ操作は、すべてのユーザーで共通しています。

「ナビゲーション・メニュー」リストには、Guardium ナビゲーション・システムの構成と内容が反映されます。項目を「ナビゲーション・メニュー」リストに追加するには、「使用可能なツールとレポート (Available Tools and Reports)」リストでツールとレポートを選択し、アイコンを使用します。「ナビゲーション・メニュー」リストから項目を削除するには、削除したい項目の横に表示されている アイコンをクリックします。「ナビゲーション・メニュー」リスト内の項目の配置を変更するには、ドラッグ・アンド・ドロップ機能を使用するか、アイコン・コントロールを使用します。

「ナビゲーション・メニュー」リストの項目を選択して アイコンをクリックすることにより、Guardium の新しいホーム・ページ (システムにログインしたときに最初に表示されるページ) を定義することができます。

「OK」ボタンをクリックすると Guardium のナビゲーション・メニューが更新され、「ナビゲーション・メニュー」リストで行ったすべての変更内容が反映されます。

これらのツールを使用する場合は、以下の制約事項が適用されます。

- 「マイ・ダッシュボード」グループは削除できませんが、グループ内のダッシュボードは個別に削除することができます。
- 新しいグループを作成しても、そのグループが空の場合は保存されません。
- 「ナビゲーション・メニュー」リストに表示されているグループのうち、空のグループは Guardium のナビゲーション・メニューには表示されません。

親トピック: 始めに
関連情報:
ロールと権限の管理

モニターおよびコンプライアンスのクイック・スタート

モニター・エージェントをデータベース・サーバーにデプロイし、データベース・モニターをセキュリティー基準と規制に準拠するように構成する方法について説明します。

このタスクについて

モニターおよびコンプライアンスのクイック・スタートは、次の 2 つのツールで構成されます。

モニター・エージェントのデプロイ

モニター・エージェントのデプロイ・ツールを使用して、GIM クライアントを自動的にアクティブ化し、S-TAP をインストールして、データベース・トラフィックのモニターを開始します。

モニター・エージェントのデプロイ・ツールは、Guardium デプロイメントを設定するプロセスを単純化します。既存の Guardium Installation Manager (GIM) のインフラストラクチャーで構築されているモニター・エージェントのデプロイ・ツールは、データベース・サーバーを素早く検索し、モニター・エージェント (S-TAP) をインストールし、検査エンジンをデータベース用に構成するのに役立ちます。 また、このツールは、デプロイメント状況を追跡して検討するための中央ビューを備えています。

コンプライアンス・モニター

モニター・エージェント (S-TAP) をデプロイした後、コンプライアンス・モニター・ツールを使用して、特定のセキュリティー基準および規制のモニターを設定します。

Guardium は、以下のような特定の基準および規制に対応するグループ、セキュリティー・ポリシー、およびレポートなどの、コンプライアンス・モニター・テンプレートをいくつか備えています。

- バーゼル銀行監督委員会 (BASEL II)
- 一般データ保護規則 (GDPR)
- Db2 for z/OS 用の一般データ保護規則 (Db2 for z/OS の GDPR)
- 医療保険の相互運用性と説明責任に関する法律 (HIPAA)
- クレジット・カード業界データ・セキュリティー基準 (PCI)
- 個人情報 (PII)
- サーベンス・オクスリー (SOX) 法への準拠

これらのクイック・スタート・コンプライアンス・モニター・テンプレートは、関連する基準または規制のいずれかに短期間で準拠する必要がある組織に特に役立ちます。セキュリティー・ポリシーをインストールした後、コンプライアンス・モニター・ツールは、初期セットアップやグループへの組織固有の情報 (クライアント IP アドレスや特定の特権ユーザー ID など) の取り込みについて、管理者およびコンプライアンス担当者にガイドを提供します。 さらに、コンプライアンス・モニター・ツールは、Guardium 環境を定期的に検査して、コンプライアンス・モニター・テンプレートを使用してモニターできる新規のデータベースを調べます。

手順

1. ツールについて詳しく理解するために以下の情報を確認します。
   - モニター・エージェントをデプロイするためのクイック・スタート
   - コンプライアンス・モニターのクイック・スタート
2. データベース・サーバー用のモニター・エージェントをデプロイします。
   a. モニター・エージェントのデプロイ・ツールを使用するための前提条件を満たしていることを確認します: モニター・エージェントをデプロイするための前提条件。
   b. モニター・エージェントをデータベース・サーバーにデプロイします: モニター・エージェントのデプロイ。
3. データベース・サーバー用にコンプライアンス・モニターを構成します。
   a. コンプライアンス・モニター・ツールを使用するための前提条件を満たしていることを確認します: コンプライアンス・モニターの前提条件
   b. コンプライアンス・モニターを構成します: コンプライアンス・モニターのセットアップ。
   c. データベースへのアクセスが許可されているユーザーおよびアプリケーションを識別するためにグループにデータを設定します: グループへのデータの設定。
   d. 機密データのディスカバリーおよび分類のためにデータベースへのアクセスを Guardium に許可する資格情報を提供します: 機密データのスキャンの有効化

タスクの結果

モニター・エージェントを正常にデプロイし、データベース・サーバー用にコンプライアンス・モニターを構成すると、Guardium はデータベース・トラフィックのモニターを開始します。

コンプライアンス・モニター・ページに表示される内容の解釈について詳しくは、コンプライアンス・モニター・ビューの概要を参照してください。

親トピック: 始めに

システム・ビュー

「システム・ビュー」は、多くのユーザーにとってのデフォルトの初期ビューです。 これにより、システム状況の重要な要素を確認できます。

「システム・ビュー」の下の 3 つのタブは、さまざまなタイプの状況情報を表示します。

- 「S-TAP 状況モニター」は、環境にデプロイされている S-TAPs の要約データを表示します。アイコンは状況の概要を表し、検査エンジンについての情報を確認するためにドリルダウンできます。
- 「ユニット使用状況」タブは、各 Guardium システムの使用状況についての情報を表示します。
- 「システム・モニター」タブには、着信データ、CPU 使用量、およびその他の情報に関する最新の詳細が表示されます。

親トピック: 始めに

データ・アクティビティーのモニター

Guardium データ・アクティビティー・モニターで使用される重要なセキュリティー概念について説明します。

- ポリシーおよびルール
  セキュリティー・ポリシーには、データベース・クライアントとサーバーとの間の監視対象トラフィックに適用される順序付きルール・セットが含まれています。各ルールは、クライアントからの要求、またはサーバーからの応答に適用できます。1 つの Guardium システムに、同時に複数のポリシーを定義することも、複数のポリシーをインストールすることもできます。
- ワークフロー
  ワークフローは、いくつかのデータベース・アクティビティー・モニター・タスクを統合します。このタスクには、資産のディスカバリー、脆弱性評価と強化策、データベース・アクティビティーのモニターと監査レポートの作成、レポートの配布、主要な利害関係者によるサインオフ、エスカレーションなどがあります。
- 監査
  Guardium には、データベース表内の値の変更をトラッキングする「値変更監査」機能があります。
- 分類
  Guardium は、機密データのディスカバリーと分類をサポートすることにより、効果的なアクセス・ポリシーの作成と適用を可能にします。

親トピック: 始めに

ポリシーおよびルール

セキュリティー・ポリシーには、データベース・クライアントとサーバーとの間の監視対象トラフィックに適用される順序付きルール・セットが含まれています。各ルールは、クライアントからの要求、またはサーバーからの応答に適用できます。1 つの Guardium システムに、同時に複数のポリシーを定義することも、複数のポリシーをインストールすることもできます。

ポリシー内の各ルールは、条件付きアクションを定義します。条件は、許可されたクライアント IP グループ内で見つからないクライアント IP アドレスからのアクセスがないか検査するなどの単純なテストにすることも、複数のメッセージおよびセッション属性 (データベース・ユーザー、ソース・プログラム、コマンド・タイプ、時刻など) を評価する複雑なテストにすることもできます。ルールは、指定の時間フレーム内で条件が満たされる回数を識別するようにすることもできます。

ルールで起動されるアクションは、通知アクション (例えば、1 人以上の受信者に対する E メール通知)、ブロッキング・アクション (クライアント・セッションが切断されるなど) のほか、イベントがポリシー違反としてログに記録されるだけの場合もあります。所定の環境やアプリケーションに固有と見なされる条件に対して必要とされるすべてのタスクを実行するカスタム・アクションを開発することができます。

親トピック: データ・アクティビティーのモニター

ワークフロー

ワークフローは、いくつかのデータベース・アクティビティー・モニター・タスクを統合します。このタスクには、資産のディスカバリー、脆弱性評価と強化策、データベース・アクティビティーのモニターと監査レポートの作成、レポートの配布、主要な利害関係者によるサインオフ、エスカレーションなどがあります。

ワークフローは、データベース・セキュリティーの管理を、時間のかかる定期的な手動アクティビティーから、企業のプライバシー要件とガバナンス要件 (PCI-DSS、SOX、データ・プライバシー、HIPAA など) をサポートする継続的な自動化プロセスに変換します。さらに、ワークフローを使用すると、Syslog、CSV/CEF ファイル、外部フィードを介して、追加のフォレンジック分析用に監査結果を外部リポジトリーにエクスポートすることができます。

例えば、コンプライアンス・ワークフロー自動化プロセスは、「必要なレポート、アセスメント、監査証跡、分類のタイプは?」、「この情報の受信者と、サインオフの処理方法は?」、「配布のスケジュールは?」などの質問を処理することができます。

親トピック: データ・アクティビティーのモニター

監査

Guardium には、データベース表内の値の変更をトラッキングする「値変更監査」機能があります。

変更のトラッキング対象にする各表において、モニター対象にする SQL 値変更コマンド (INSERT、UPDATE、DELETE) を選択できます。 モニター対象の表に対して値変更コマンドが実行されるたびに、before 値および after 値が収集されます。 この変更アクティビティーは定期的に Guardium にアップロードされ、その後、Guardium のすべてのレポート機能およびアラート機能を使用できるようになります。

デフォルトの「変更された値」レポートから値変更データを表示できます。あるいは、「値変更のトラッキング」ドメインを使用してカスタム・レポートを作成することもできます。

親トピック: データ・アクティビティーのモニター

分類

Guardium は、機密データのディスカバリーと分類をサポートすることにより、効果的なアクセス・ポリシーの作成と適用を可能にします。

分類ポリシーは、機密データ・エレメントのディスカバリーとタグ付けを行うために設計された一連のルールです。分類ポリシー内で、ルールごとにアクションを定義することができます (例えば、E メール・アラートの生成や、Guardium グループへのメンバーの追加など)。また、分類ポリシーは、指定のデータ・ソースに対して実行するようにスケジュールすることも、ワークフロー内のタスクとして実行するようにスケジュールすることもできます。

組織の規模が大きくなり、クレジット・カード番号や個人の金融データなどの機密情報が複数のロケーションに存在するようになると (そのデータの現在の管理責任者が分からないという場合がよくあります)、ディスカバリー・ルーチンと分類ルーチンの重要性が増します。 こうした状況は、合併買収の際や、元の所有者がいなくなった後もレガシー・システムを引き続き使用している場合に、多く発生します。 Guardium の分類ポリシーは、このような機密データのディスカバリーとタグ付けを行うことにより、適切なアクセス・ポリシーを適用できるようにします。

親トピック: データ・アクティビティーのモニター

ファイル・アクティビティー・モニター

ファイル・アクティビティー・モニターはサーバー上の機密データをディスカバーします。また、事前定義の定義またはユーザー定義の定義を使用してコンテンツを分類します。さらに、データ・アクセスに関するルールおよびポリシーや、ルールが満たされたときに実行されるアクションを構成します。

ファイル・アクティビティー・モニターは、以下の機能で構成されます。

- ディスカバリーには、ファイルおよびフォルダーのメタデータおよび資格の収集が含まれます。
- 分類では、*判定プラン*を使用して、ファイル内の機密データと見なされるデータ (クレジット・カード情報、個人を特定できる情報など) が識別されます。
- 監査情報のモニターと収集、ポリシー・ルール、および疑わしいユーザーや接続のリアルタイム・アラートまたはブロック。

ファイル・アクティビティー・モニター:

- 制御を自動化および一元化。
- データ量の増加や企業要件の増大に合わせた規模の拡大が可能。
- 一般的なシステムに対応する広範な異種混合サポートを提供。

ユース・ケース 1

アプリケーション・サーバーやデータベース・サーバーに対するバックエンド・アクセスを通じて、重大なアプリケーション・ファイルがアクセスまたは変更される可能性や、破壊される可能性もある。

解決策: ファイル・アクティビティー・モニターにより、構成ファイル、ログ・ファイル、ソース・コード、およびその他多くの重大なアプリケーション・ファイルをディスカバーしてモニターし、無許可ユーザーや無許可プロセスがアクセスを試みた場合に、アラートの発行やブロックを実行できる。

ユース・ケース 2

個人情報 (PII) や専有情報を含むファイルを、日常の業務に影響を与えずに保護する必要がある。

解決策: ファイル・アクティビティー・モニターにより、多くのファイル・システムに保管された機密文書へのアクセスをディスカバーしてモニターできる。ファイル・アクティビティー・モニターは、データを集約して、アクティビティーに対する情報を提供し、疑わしいアクセスが行われた場合にアラートを発行し、特定のファイルおよびフォルダーに対する特定のユーザーからのアクセスをブロックできるようにする。

ユース・ケース 3

アプリケーションが管理する文書へのバックエンド・アクセスをブロックする必要がある。

解決策: ファイル・アクティビティー・モニターにより、通常はアプリケーションのフロントエンド (Web ポータルなど) を通じてアクセスする文書に対するバックエンド・アクセスをディスカバーしてモニターし、これをブロックすることができる。

- ファイル・アクティビティー・モニターの機能
- ファイル・アクティビティー・モニターの前提条件
- ファイル・アクティビティー・モニターの上位ワークフロー
  この一般ワークフローを使用して、ファイル・アクティビティー・モニターを計画および実行します。

親トピック: 始めに
関連情報:
Guardium を使用したファイル・アクティビティーのモニター (ビデオ)

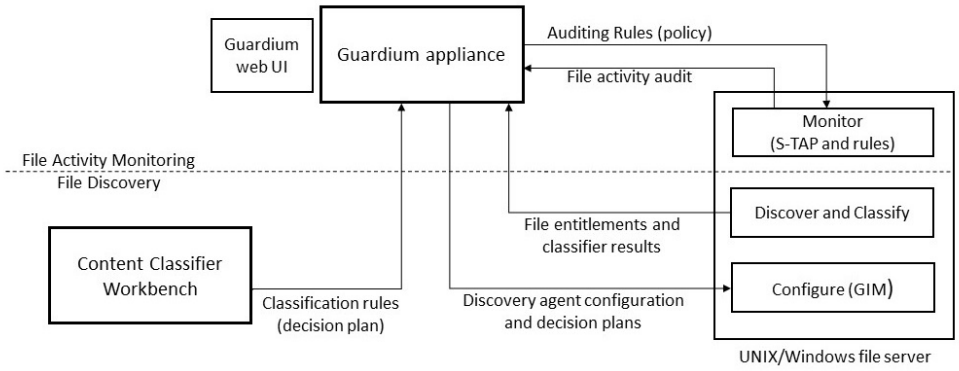
ファイル・アクティビティー・モニターの機能

Windows 環境の NAS デバイスおよび SharePoint 上のファイルとディレクトリーのアクティビティー・モニターについては、NAS および SharePoint のファイル・アクティビティー・モニターを参照してください。

Windows 環境の NAS デバイスおよび SharePoint 上のファイルとディレクトリーのディスカバリー、資格、および分類については、NAS および SharePoint のディスカバリーと分類を参照してください。

ファイル・サーバーのファイル・アクティビティー・モニターは、以下の機能からなります。

- ディスカバリーには、ファイルおよびフォルダーのメタデータおよび資格の収集が含まれます。
- 分類では、*判定プラン* を使用して、ファイル内の機密データと見なされるデータ (クレジット・カード情報、個人を特定できる情報など) が識別されます。
- 監査情報のモニターと収集、ポリシー・ルール、および疑わしいユーザーや接続のリアルタイム・アラートまたはブロック。



ディスカバリーと分類

基本的な**ディスカバリー**・スキャンにより、フォルダーおよびファイルと、それぞれの所有者、アクセス権限、サイズ、および最終更新日時のリストが識別されます。また、ユーザー権限とグループ権限も識別されます。ディスカバリーではすべてのファイル・タイプがサポートされます。**分類**は、*判定プラン* により定義されています。各判定プランには、特定タイプのデータを認識するためのルールが含まれます。(ファイル・アクティビティー・モニターの判定プランは、データ・アクティビティー・モニターの分類ポリシーと類似しています)。分類では、プレーン・テキスト、HTML、Office、PDF を含め、さまざまなタイプのファイルがサポートされます。HIPAA、PCI、SOX、およびソース・コードには、デフォルトの判定プランがあります。 デフォルトの判定プランを使用して、結果のレポート/調査ダッシュボードから分類エンティティーを変更できます。さらに、コンテンツ分類ワークベンチ (コレクター・アプライアンスにアップロードする Windows アプリケーション) を使用して、新しいプランを作成したり、既存のプランを変更したりすることもできます。IBM Content Classification バージョン 8.8 の要件については、IBM 技術情報 (http://www-01.ibm.com/support/docview.wss?uid=swg27020838) を参照してください。Guardium Installation Manager (GIM) を使用してプランをアクティブ化および構成します。

ディスカバリーと分類は、ファイル・クローラーと呼ばれるディスカバリー・エージェントによって処理されます。ファイル・クローラーは、ファイル・メタデータとデータをディスカバリーおよび分類プロセスから Guardium システムに送信します。スキャンのスケジュールは構成可能です。初期ディスカバリーおよび分類が完了すると、以降の (増分) スキャンでは、新規ファイルおよび変更後のファイルのみの増分の変更が識別されます。ファイル・クローラーは、他のバンドルと同様に Guardium Installation Manager (GIM) を使用してインストールし、構成します。

モニター、監査、ブロック

ファイル・アクティビティー・モニターは、ファイル・サーバー上で稼働する S-TAP によって実装されます。NFS ボリュームでは、それらのボリュームにアクセスするすべてのマシンに S-TAP がインストールされ、構成されていることが重要です。S-TAP は、Guardium のポリシー・ルールに従って、ファイル・アクセスの継続モニター、アラート生成、ブロックを管理します。これらのルールは、モニター対象のファイル・サーバーおよびファイルと、ポリシー・ルールに違反している場合に実行するアクション (例えば、違反の記録、アラート生成、アクセスのブロック) を指定します。モニター対象の操作は、読み取り、書き込み、実行、削除、および所有者、許可、プロパティーの変更です。セキュリティー・ポリシー・ルールの基準に一致するアクティビティーは、いずれも Guardium コレクターに送信されて、Guardium リポジトリーに保管されます。(データベース・アクティビティー・モニターでは、S-TAP により、すべてのデータ・アクティビティーが Guardium に送信され、そこでモニターされます)。Guardium リポジトリーに記録されるイベントは、いずれも監査済みイベントです。

また、オペレーティング・システムがファイルへのアクセスを許可している場合でも、そのアクセスをブロックできます。S-TAP でファイル・モニター・ルールがアクティブ化されるため、ブロックはただちに実行されます。ユーザーが要求するデータがディスクから読み取られることは決してありません。そのような操作は、S-TAP がブロックして防ぐためです。

モニター・アクティビティーは、定義済みのレポート (ユーザー特権、ファイル特権、ユーザーあたりのアクティビティー数、クライアントあたりのアクティビティー数、「公開」されているファイル、休止ユーザー、休止ファイルなど)、FAM – アクセス・レポート (すべてのモニター対象のアクティビティーのログ)、および調査ダッシュボードに表示されます。

重要: Windows 管理者と Linux ROOT ユーザーのアクティビティーは、ファイル・アクティビティー・モニターではモニターされず、またブロックもされません。

1 つの S-TAP エージェントが、ファイル・サーバーのアクティビティー・モニターとデータベースのアクティビティー・モニターの両方を管理します。両方の機能のライセンスがある場合は、同じ S-TAP エージェントをファイル・アクティビティー・モニターとデータベース・アクティビティー・モニターの両方に使用できます。S-TAP は、他のバンドルと同様に、Guardium Installation Manager (GIM) を使用してインストールし、構成します。

Windows FAM の UID チェーン

> Windows FAM エージェントは、単一のユーザー名を、プロセスの履歴に属するユーザー名のチェーン (UID チェーン) に変更します。例えば、プロセス 1 (ユーザー janedoe) がプロセス 2 (ユーザー johndoe) を作成した場合、プロセス #2 に関連するファイル・イベントについて、FAM は {janedoe, johndoe} からなる UID チェーンを報告します。

親トピック: ファイル・アクティビティー・モニター

ファイル・アクティビティー・モニターの前提条件

NAS および SharePoint 環境のファイル・アクティビティー・モニター:

- Windows 環境の NAS デバイスおよび SharePoint 上のファイルとディレクトリーのアクティビティー・モニターについては、NAS および SharePoint のファイル・アクティビティー・モニターを参照してください。
- Windows 環境の NAS デバイスおよび SharePoint 上のファイルとディレクトリーのディスカバリー、資格、および分類については、NAS および SharePoint のディスカバリーと分類を参照してください。

Linux、UNIX、または Windows のサーバー上の機密データのディスカバリー、分類、およびモニターを行う場合は、以下のコンポーネントをインストールします。

- GIM クライアント (すべてのファイル・サーバー上)
- S-TAP
- ライセンス・キー
- bash シェル (FAM ディスカバリー・エージェントのインストール前にインストールする必要があります)
- FAM ディスカバリー・エージェント (別名 FAM バンドルまたは FAM エージェント。ディスカバリーおよび分類のために必要)
- オプションの IBM Content Classification (デフォルト以外の判定プランの場合。)

ファイル・アクティビティー・モニターは UID チェーンをサポートしています。つまり、FAM エージェントは単一のユーザー名を、プロセスの履歴に属するユーザー名のチェーン (UID チェーン) に変更します。例えば、プロセス 1 (ユーザー janedoe) がプロセス 2 (ユーザー johndoe) を作成した場合、プロセス #2 に関連するファイル・イベントについて、FAM は {janedoe, johndoe} からなる UID チェーンを報告します。

FAM は、SMB V2.0 以降の CIFS 実装をサポートしています。

サポートされるプラットフォームについては、FAM Support を参照してください。

親トピック: ファイル・アクティビティー・モニター

ファイル・アクティビティー・モニターの上位ワークフロー

この一般ワークフローを使用して、ファイル・アクティビティー・モニターを計画および実行します。

Linux、UNIX、および Windows 環境のファイル・サーバー上のファイル・アクティビティー・モニターの上位ワークフロー:

1. ファイル・サーバー上のファイル・アクティビティー:
   a. すべてのファイル・サーバーでのFAM コンポーネントのインストールおよびアクティブ化。
   b. ファイルのディスカバリーおよび分類 GIM パラメーターを構成します。
   c. オプションのFAM 判定プランのカスタマイズ。デフォルトの判定プランを使用することも、IBM Content Classification を使用して独自のプランを作成することもできます。
2. モニターおよび監査を行います。
   - ファイル・アクティビティーは、事前定義レポートの「ファイル・アクティビティー」、「ファイル・ライセンス」、「ファイル: クライアントあたりのアクティビティー数」、「ファイル: サーバーあたりのアクティビティー数」、「ファイル: ユーザーあたりのアクティビティー数」、「ファイル: 特権」などのレポートに含めることができます。
   - 継続的な調査および分析には、調査ダッシュボードを使用します。調査ダッシュボードには、テキスト検索や異常値の機能、および拡張された可視化が含まれます。以下を参照してください。
     - ファイルの調査ダッシュボード
     - 調査ダッシュボードでのファイル・アクティビティーの異常値の解釈
3. 保護: 進行中のモニターおよびファイル・サーバーの保護のポリシーを作成して適用します。ファイル・アクティビティーのポリシーおよびルールを参照してください。

NAS および SharePoint 環境のファイル・アクティビティー・モニター:

- Windows 環境の NAS デバイスおよび SharePoint 上のファイルとディレクトリーのアクティビティー・モニターについては、NAS および SharePoint のファイル・アクティビティー・モニターを参照してください。
- Windows 環境の NAS デバイスおよび SharePoint 上のファイルとディレクトリーのディスカバリー、資格、および分類については、NAS および SharePoint のディスカバリーと分類を参照してください。

親トピック: ファイル・アクティビティー・モニター

Big Data Intelligence

Guardium Big Data Intelligence (GBDI) は、収集したデータをより長期間にわたって保管し、データ・セキュリティーおよびコンプライアンスに関するレポートや洞察への直接的かつリアルタイムのアクセスを提供します。

Big Data Intelligence では、データマートを使用して中央ストレージへのエクスポートを行います。このストレージは Guardium 内のデータ・ソースとして定義されています。このデータ・ソースに対して、レポートの実行やクイック検索などを行うことができます。

プロファイルとは、まとめてアクティブ化されるデータマートのグループです。変更不可の事前定義プロファイルが 4 つ用意されています。プロファイルのコピーを作成して、そのコピーを変更したり削除したりすることが可能です。

Guardium Big Data Intelligence の使用を開始するには、次の手順を実行します。

1. 中央ストレージを定義します。API を使用するか、データ・ソース定義を作成します。アプリケーション・タイプとして「Big Data Intelligence」を使用します。
2. GuardAPI Big Data Intelligence 関数 **enable_big_data_interface** (GuardAPI Big Data Intelligence 関数を参照) を使用してデータ・エクスポート・プロファイルを定義します。このプロファイルにより、データ・ソース、データマート抽出プロファイル (1 つ以上のデータマートで構成されます)、およびエクスポート・スケジュールが定義されます。変更不可の事前定義プロファイルが 4 つ用意されています。事前定義プロファイルをコピーおよび変更して、データ・エクスポート・プロファイルを作成できます。

   プロファイルは、中央マネージャーのすべての管理対象ユニットと中央マネージャー自体に適用されます。

3. **enable_big_data_interface** を実行すると、照会 - レポート・ビルダーに Big Data Intelligence ドメインが表示されます。ユーザーが照会を定義する際に、Guardium は GBDI に接続して照会の検証と保存を行う必要があります。この処理には最大 1 分かかる場合があります。GUI の左下隅を確認してください。接続するまで、「サーバーを待機中 `(waiting for server)`」というテキストが表示されます。Guardium は GBDI に接続できない場合、「接続を確立できません... `(Unable to establish connection...)`」というメッセージで応答し、照会は保存されません。
4. 標準の Guardium 事前定義レポートと調査ダッシュボード (クイック検索) を使用して、データを分析します。Big Data Intelligence ドメインを使用してレポートを作成することも可能です。
5. CM に管理対象ユニットを追加するときに、次の API コマンドを実行します。grdapi local_enable_big_data_interface profile_name="<profile name>"

データ処理ガイドライン

- Big Data Intelligence サーバーは長期間にわたってデータを保存するため、コレクターでのデータ保持を 1 日に短縮できます。
- データのバックアップは、Big Data Intelligence サーバーで処理できます。
- 構成のバックアップは Guardium システムで処理する必要があります。
- アーカイブは、ユーザーの規則要件に従って処理する必要があります。　Big Data Intelligence サーバーは、コレクターやアグリゲーターよりも長期間データを保持し、また、アーカイブに使用できます。

Guardium 照会 - レポート・ビルダーまたは Enterprise Search を使用して直接 Big Data Intelligence を読み取るための Guardium の機能には、Guardium Big Data Intelligence バージョン 3.3 が必要です。以前の GBDI バージョンへのデータマート抽出が既に存在する場合は、実行中の抽出を無効にしてから、次の API を使用して再度有効にします。enable_big_data_interface

各プロファイル内のデータマートの要約

基本的な要約
: エクスポート:アクセス・ログ、エクスポート:セッション・ログ、エクスポート:終了したセッション・ログ、エクスポート:例外ログ、エクスポート:完全な SQL、エクスポート:異常値リスト - 拡張、エクスポート:時間単位の異常値概要 - 拡張、エクスポート:グループ・メンバー、エクスポート:抽出ログ、エクスポート:ポリシー違反、エクスポート:バッファー使用状況モニター

包括的な要約
: エクスポート:アクセス・ログ、エクスポート:セッション・ログ、エクスポート:終了したセッション・ログ、エクスポート:例外ログ、エクスポート:完全な SQL、エクスポート:異常値リスト - 拡張、エクスポート:時間単位の異常値概要 - 拡張、エクスポート:グループ・メンバー、エクスポート:抽出ログ、エクスポート:ポリシー違反、エクスポート:バッファー使用状況モニター、エクスポート:脆弱性診断結果、エクスポート:STAP 状況、エクスポート:ディスカバーされたインスタンス、エクスポート:ディスカバーされたデータベース、エクスポート:分類結果、エクスポート:インストール済みのパッチ、エクスポート:システム情報

基本的な詳細
: エクスポート:アクセス・ログ - 詳細、エクスポート:セッション・ログ、エクスポート:終了したセッション・ログ、エクスポート: 例外ログ、エクスポート: 完全な SQL、エクスポート:異常値リスト - 拡張、エクスポート:時間単位の異常値概要 - 拡張、エクスポート:グループ・メンバー、エクスポート:抽出ログ、エクスポート: ポリシー違反 - 詳細、エクスポート: バッファー使用状況モニター

包括的な詳細
: エクスポート:アクセス・ログ - 詳細、エクスポート:セッション・ログ、エクスポート:終了したセッション・ログ、エクスポート: 例外ログ、エクスポート: 完全な SQL、エクスポート:異常値リスト - 拡張、エクスポート:時間単位の異常値概要 - 拡張、エクスポート:グループ・メンバー、エクスポート:抽出ログ、エクスポート: ポリシー違反 - 詳細、エクスポート: バッファー使用状況モニター、エクスポート:脆弱性診断結果、エクスポート:STAP 状況、エクスポート:ディスカバーされたインスタンス、エクスポート:ディスカバーされたデータベース、エクスポート:分類結果、エクスポート:インストール済みのパッチ、エクスポート:システム情報

親トピック: 始めに
関連情報:
ドメイン、エンティティー、および属性

重要な概念とツール

Guardium の管理に関連した重要な概念について説明します。

- 照会およびレポート
  Guardium 照会は、収集したデータから取得される情報セットを記述します。 レポートは、Guardium 照会によって識別されるデータの表示方法を定義します。
- アクセス制御
  Guardium には、データベース・クライアントとデータベース・サーバー間のデータ・アクセスを簡単に表示する手段として、アクセス・マップが用意されています。
- ユーザー・ロール
  ロールは、同じアクセス権を共有する Guardium ユーザーのグループを定義します。
- グループ
  Guardium では、エレメントのグループ化がサポートされているため、ポリシーの作成と管理を簡単に行い、分かりやすいレポートを表示することができます。
- データのアーカイブとパージ
  「データ・アーカイブ」は、Guardium システムによってキャプチャーされたデータをバックアップします。 データ・アーカイブを構成する際に、データ・パージの基準を指定することもできます。
- Guardium Installation Manager
  Guardium Installation Manager (GIM) を使用して、管理対象システム上で Guardium コンポーネントのインストールと保守を行います。

親トピック: 始めに

照会およびレポート

Guardium 照会は、収集したデータから取得される情報セットを記述します。 レポートは、Guardium 照会によって識別されるデータの表示方法を定義します。

Guardium 照会は、収集したデータから取得される情報セットを記述します。 照会は、エンティティー、フィールド、および条件の 3 つの要素で構成されます。 エンティティーは照会の範囲を定義し、フィールドは照会によって返されるデータの列をリストし、条件はデータに対して突き合わせるテストを定義します (より大きい、より小さい、含むなど)。

レポートは、照会で収集したデータの表示方法を定義するものです。 デフォルトのレポートは表形式のレポートであり、照会の構造を反映して、各属性が別個の列に表示されるものになります。 すべてのランタイム・パラメーターと表形式レポートの表示構成要素はカスタマイズ可能です。

親トピック: 重要な概念とツール

アクセス制御

Guardium には、データベース・クライアントとデータベース・サーバー間のデータ・アクセスを簡単に表示する手段として、アクセス・マップが用意されています。

アプリケーションとツールによるデータ・アクセスは、さまざまな次元 (アクセスされているデータ、アクセスの方法、SQL 呼び出しの実行回数など) に従って分類することができます。 エンタープライズ環境では、データベース・アクセスを適切に処理することが非常に重要です。 このような要件が生じる要因として、コンプライアンス主導で管理する必要があるため、さらにはデータベース環境を調整および最適化する必要があるために、データベース・アクセスについて理解し保護する必要が高まっているという状況があります。 エンタープライズ環境にはさまざまなデータベースと非常に多くのデータベース・クライアントが存在する場合があるため、データ・アクセス・パスを適切に処理することが困難になる場合があります。

「適用状態トポロジー」ビューおよび「適用状態表」ビューには、環境内のシステム間のデータ・フロー関係が表示されます。これらのビューにより容易に、問題のあるシステムを識別し、根本的な問題を調査できます。トポロジー・ビューにアクセスするには、「管理」 > 「システム・ビュー」 > 「適用状態トポロジー」にナビゲートします。表ビューにアクセスするには、「管理」 > 「システム・ビュー」 > 「適用状態表」にナビゲートします。

親トピック: 重要な概念とツール

ユーザー・ロール

ロールは、同じアクセス権を共有する Guardium ユーザーのグループを定義します。

ロールがアプリケーションまたは項目の定義 (特定の照会など) に割り当てられると、そのロールを割り当てられた Guardium ユーザーのみがそのコンポーネントにアクセスできます。 どのセキュリティー・ロールもコンポーネント (レポートなど) に割り当てられていないと、そのコンポーネントを定義したユーザーおよび admin ユーザーのみがそれにアクセスできます。

インストール時に、Guardium はデフォルトのロールセットおよびデフォルトのユーザー・アカウントのセットを使用して構成されます。 Guardium アクセス・マネージャーでは、新しいロールを作成したり、必要に応じて既存のロールを変更したりすることができます。

親トピック: 重要な概念とツール

グループ

Guardium では、エレメントのグループ化がサポートされているため、ポリシーの作成と管理を簡単に行い、分かりやすいレポートを表示することができます。

グループ化によって、 ポリシー作成および照会定義のプロセスが簡略化されます。 多くの場合、同じタイプのエレメントをグループ化すると便利です。グループ化することにより、レポートの情報をより分かりやすい形式で表示することができます。グループはすべてのサブシステムによって使用され、すべてのユーザーが単一のグループ・セットを共用します。

グループ化の例として、従業員の機密情報が含まれている個別のデータ・オブジェクトが社内に 25 個あり、これらの項目に対するすべてのアクセス権についてレポートする必要があるとします。 25 項目それぞれについてテストする、非常に長い照会を編成することができます。 あるいは、これらの 25 オブジェクトを含んだ、「sensitive employee info」という名前の単一のグループを定義することもできます。その方法では、照会またはポリシー・ルール定義において、オブジェクトがそのグループのメンバーであるかどうかのテストのみが必要とされます。

グループには、その他にも、グループの構成変更時に、保守要件が緩和されるという利点があります。 上記の例では、「sensitive employee info」グループにさらに 2 つのオブジェクトを追加する必要があると社内で決定した場合、更新する必要があるのはグループ定義だけです。そのグループを参照するすべての照会、レポート、ポリシーを更新する必要はありません。

親トピック: 重要な概念とツール

データのアーカイブとパージ

「データ・アーカイブ」は、Guardium システムによってキャプチャーされたデータをバックアップします。 データ・アーカイブを構成する際に、データ・パージの基準を指定することもできます。

「データ・アーカイブ」と「結果アーカイブ」という 2 つのアーカイブ操作があります。これらのアーカイブ操作へのパスは、「管理」>「データ管理」>「データ・アーカイブ」または「結果アーカイブ (監査)」です。

データ・アーカイブ
「データ・アーカイブ」を使用すると、通常、キャプチャーされた日の最後にデータがアーカイブされます。これにより、災害が発生した場合、その日のデータだけが失われることになります。 データのパージは、アプリケーション、ビジネス要件、監査要件に基づいて行われますが、ほとんどの場合、データはマシン上に 6 カ月以上保持することができます。

結果アーカイブ
「結果アーカイブ」は、監査タスクの結果 (レポート、アセスメント・テスト、エンティティー監査証跡、プライバシー・セット、分類プロセス) だけでなく、表示とサインオフの証跡と、ワークフロー・プロセスからの調整コメントもバックアップします。 結果セットは、ワークフロー・プロセスの定義に従ってシステムからパージされます。

統合環境では、データはコレクター、アグリゲーター、またはその両方のロケーションからアーカイブできます。 データは、一度だけアーカイブするのが最も一般的です。 アーカイブ元となるロケーションは、ユーザーの要件に応じて異なります。

親トピック: 重要な概念とツール

Guardium Installation Manager

Guardium Installation Manager (GIM) を使用して、管理対象システム上で Guardium コンポーネントのインストールと保守を行います。

GIM コンポーネントには、Guardium システムの一部としてインストールされる GIM サーバーと、モニターするデータベースとファイル・サーバーをホストするサーバー上にインストールされている必要がある GIM クライアントがあります。インストールされた GIM クライアントは、GIM サーバーと連携して以下のタスクを実行します。

- インストールされたソフトウェアの更新がないか検査する
- 新規ソフトウェアを転送およびインストールする
- ソフトウェアをアンインストールする
- ソフトウェア・パラメーターを更新する

中央マネージャーとして構成されている Guardium システムが現在の環境に存在する場合は、GIM サーバーとして使用する Guardium システムを決定する必要があります。 中央マネージャーなどの単一の Guardium システムからすべての GIM クライアントを管理することも、複数の Guardium システムから GIM クライアントをグループ単位で管理することもできます。 単一の Guardium システムからすべての GIM クライアントを管理する場合は、単一のインターフェースですべての GIM クライアントの状況を表示し、関連するタスクを実行することができます。個別の Guardium システムから GIM クライアントをグループ単位で管理する場合は、各システムを使用して、そのシステムで管理される GIM クライアントを処理できますが、全体的なビューや、環境全体にわたるビューは使用できません。

親トピック: 重要な概念とツール

ディスカバー

ディスカバリーとは、セキュリティーやコンプライアンス上の目的でトラッキングする必要のある、環境内のオブジェクトを見つけて識別するプロセスを指します。

ディスカバリーは、特権ユーザー、機密データ、データ・ソースなどの重要なオブジェクトを検出するプロセスです。分類は、セキュリティーやコンプライアンス上の目的でディスカバーされたものを適切に識別するプロセスです。これらのディスカバリー・プロセスと分類プロセスは、大規模な組織で合併や買収、レガシー・システムによって、新しいオブジェクトが非構造化形式または予測不能な方法で現行環境に導入される場合に重要となります。GuardiumGuardium® は、有効なセキュリティー・ポリシーを施行してコンプライアンスを実現できるようにこれらのオブジェクトを現行環境に取り込むのに役立ちます。

一般的なシナリオには、機密データのディスカバリーがあります。機密データとは、クレジット・カード番号、個人の金融データ、社会保障番号、その他特殊な取り扱いを必要とする情報など、規制が掛けられている情報を指します。Guardium では、2 種類の方法で機密データをディスカバーできます。1 つは「機密データのディスカバー」ワークフロー・ビルダーを使用する方法で、もう 1 つはポリシー・ビルダーを他の Guardium ツールとともに使用する方法です。 「機密データのディスカバー」ワークフロー・ビルダーは、機密データのディスカバリーおよび分類プロセスを設定するための包括的なツールとして設計されています。これを使用して、ディスカバリー・ルールの指定、ディスカバーされたデータに対して実行するアクションの定義、スキャンするデータ・ソースの指定、レポートの配布、自動スケジュールでのワークフローの実行を行います。上級者用に、より細分度の高いディスカバリー・ルールおよび分類ルールがポリシー・ビルダーでサポートされています。これらのルールは、既存のプロセスや Guardium アプリケーションに容易に取り込むことができます。

- データ・ソース
  データ・ソースには、データベースやリポジトリーに関する情報 (データベースのタイプ、リポジトリーの場所、関連付けられる可能性のある資格情報など) が格納されます。Guardium アプリケーションでデータ・ソースを使用するには、データ・ソースを定義する必要があります。
- クラウド・データベース・サービス保護
  クラウド・データベース保護は、クラウド・データベースにおける分類、脆弱性評価、およびオブジェクト監査を提供します。
- データベース・オートディスカバリー
  オートディスカバリー・アプリケーションは、サーバーをスキャンおよびプローブしてオープン・ポートを調べ、ネットワークに対して不明な接続や望ましくない接続が行われるのを防ぎます。オートディスカバリー・プロセスはオンデマンドで実行することも、定期的に実行されるようスケジュールすることもできます。
- 分類
  分類ポリシーと分類プロセスは、Guardium が機密データ (クレジット・カード番号、社会保障番号、個人の金融データなど) をディスカバーして処理する方法を定義します。
- 機密データのディスカバー
  機密データをディスカバーして分類するためのエンドツーエンドのシナリオを作成します。
- 正規表現
  正規表現を使用して、データに含まれる複合パターンを求めてトラフィックを検索することができます。
- ファイル・サーバー内での機密データのディスカバーおよび分類
  ファイル・アクティビティー・モニターは、UNIX ファイル・サーバーおよび Windows ファイル・サーバー上の機密データの保全性と保護を確保します。

- NAS および SharePoint のディスカバリーと分類
  File Discovery, Entitlement and Classification (FDEC) for NAS and SharePoint servers により、規制法 (GDPR、HIPAA など) に関連している可能性がある機密データのファイル・ライセンスおよび分類をスキャンによって調べることができます。
- 資格最適化
  資格最適化は、ジョブを効率的に実行するために必要な資格をユーザーに提供する上でのデータベース管理者のロールと、システムの脆弱性を防ぐために資格をできる限り正確に、かつ可能な限り最小限に抑える上でのセキュリティーのロールの間を仲介するものです。

データ・ソース

データ・ソースには、データベースやリポジトリーに関する情報 (データベースのタイプ、リポジトリーの場所、関連付けられる可能性のある資格情報など) が格納されます。Guardium® アプリケーションでデータ・ソースを使用するには、データ・ソースを定義する必要があります。

- データ・ソース定義の作成
  「データ・ソース・ビルダー」を使用して、Guardium アプリケーションで使用するデータ・ソース定義を作成します。
- 既存のデータ・ソースの操作
  データ・ソース定義を作成したら、そのデータ・ソースのコピー、変更、または削除を行うことができます。
- データ・ソースについてのレポート
  Guardium は、現行環境内のデータ・ソースと、それらに加えられた変更内容についてのレポートを提供します。
- サービス名を使用したデータ・ソースの定義
  カスタム URL を使用することで、ユーザーがサービス名を使用して Oracle データベースに接続できるようにするデータ・ソースを定義できます。
- KDC 定義の管理
  データ・ソースが Kerberos を使用する認証を必要とする場合、接続を確立する前に、Guardium が Kerberos チケットを取得するために必要な情報を指定できます。

親トピック: ディスカバー

データ・ソース定義の作成

「データ・ソース・ビルダー」を使用して、Guardium アプリケーションで使用するデータ・ソース定義を作成します。

このタスクについて

データ・ソース定義を作成するための一般的なプロセスは 2 つあります。1 つは、「データ・ソース・ビルダー」からデータ・ソース定義を追加した後、そのデータ・ソースを使用するアプリケーションを指定する方法です。もう 1 つは、使用したいアプリケーションに移動してから、そのアプリケーション内でデータ・ソースを作成する方法です。特定のアプリケーション内でデータ・ソース定義を追加するためのナビゲーションは、選択したアプリケーションや、選択したデータベースのタイプによって異なります。 例えば、監査データベースを作成する場合は、「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「値変更監査データベースの作成」にナビゲートし、「データ・ソースの追加」をクリックします。

アプリケーション・タイプ Big Data Intelligence のデータ・ソースは Guardium システムごとに 1 つのみ定義できます。

手順

1. 「データ・ソース・ビルダー」を開きます。これを行うには、「設定」 > 「データ・ソース定義」にナビゲートします。
2. をクリックして、「データ・ソースの作成」ダイアログを開きます。「データ・ソースの作成」ダイアログを使用して、今後使用するために保管するデータ・ソースに関する情報を指定します。選択したアプリケーションおよびデータベースと、使用するデータ・ソースのタイプに応じて、このダイアログは若干異なります。
3. 「アプリケーション・タイプ」を選択します。
4. データ・ソースに固有の「名前」を入力します。
5. 「データベース・タイプ」メニューから、データベースまたはファイル・タイプを選択します。アプリケーションによっては、データ・ソースがテキスト・ファイルではなくデータベースでなければならない場合があります。選択したデータベースのタイプによっては、パネル上の一部のフィールドが無効になったり、ラベルが変更されたりすることがあります。 例えば、「資格情報の割り当て」はオプションまたは必須のどちらかの可能性があります。必須の場合、それは無効になり、「ユーザー名」フィールドと「パスワード」フィールドが必須になります。オプションの場合、「ユーザー名」と「パスワード」は、「資格情報の割り当て」を選択するまで無効になっています。
6. 「データ・ソースの共有」を選択し、すべてのアプリケーション間でデータ・ソース定義を共有します。データ・ソースを共有しない場合、作成した定義は選択したアプリケーションのみで使用可能になります。
7. オプションで、追加の資格情報を構成します。
   - SSL の使用: SSL を使用する場合に選択します。次に、オプションで 「サーバーの SSL 証明書をインポートします」を選択し、「証明書の追加」をクリックして証明書を選択します
   - LDAP の使用: LDAP を使用する場合に選択します。次に、「資格情報の割り当て」をクリックして、「ユーザー名」と「パスワード」を入力します
   - Kerberos を使用: 事前定義の Kerberos 構成を使用する場合に選択します。「Kerberos 構成」を選択して、「レルム」および「KDC」を入力します。データ・ソースはこれを自身の KDC およびレルムと比較して、一致することを確認します。
8. 「パスワードの保存」を選択し、Guardium アプライアンスへの認証資格情報の保存と暗号化を行います。(オンデマンドではなく) スケジュールされたタスクとして実行されるアプリケーションでデータ・ソースを定義する場合は、「パスワードの保存」が必須になります。「パスワードの保存」を選択した場合は、ログイン名とパスワードが必須になります。
9. 「ログイン名」と「パスワード」に資格情報を入力します。
10. 「ホスト名/IP」フィールドに、データ・ソースのホスト名または IP アドレスを入力します。
11. 以下の表を使用して、データ・ソース・タイプに応じた「ポート」を入力します。
    データ・ソース・タイプとポート番号の表

データベース・タイプ	ポート番号
Aster データ	2046

データベース・タイプ	ポート番号
Db2	50000 Db2 UDB の場合、Guardium は count_big(*) をサポートします。非常に大容量の表では、標準の count(*) は失敗する可能性があります。
Db2 for i	446
Db2 for z/OS	446
GreenplumDB	5432
Big Data Intelligence	27117
Hadoop	21000-21050
Informix	1526
MS SQL Server (動的ポート) および MS SQL Server (DataDirect - 動的ポート)	グレー表示のポート番号 - このデータ・ソースを使用することで、定義済みのポート値のないクライアント、または動的関数が MS SQL Server データベース・サーバーから有効化されているクライアントは、MS SQL Server データベースに動的に接続できます。 動的ポートを定義するには、MS SQL Server のデータベース・サーバーに移動して、動的ポート・タイプに 0 を定義し、デフォルトでポート 1433 の TCP/IP を削除してください。 動的ポートの値を 0 に設定して、サービスを再始動すると、動的 IP が設定されます。 MS SQL の場合、Guardium は count_big(*) をサポートします。非常に大容量の表では、標準の count(*) は失敗する可能性があります。 MS SQL Server 用の DataDirect ドライバー 以前、NTLM および NTLMv を使用して Windows 認証をサポートするには、jTDS ドライバーをダウンロードする必要がありました。 今後は、Guardium DataDirect ドライバーがこれを許可します。 パラメーター Guardium ユーザーが Windows 認証を使用する場合、次のパラメーターを接続プロパティーに追加します。 domain=domain_name;AuthenticationMethod=ntlmjava Windows 認証に NTLMv2 を使用している場合、次のパラメーターを接続プロパティーに追加します。 domain=domain_name;AuthenticationMethod=ntlm2java AuthenticationMethod 目的 接続の確立時にドライバーが使用する認証方式を決定します。指定された認証方式がデータベース・サーバーによってサポートされていない場合、接続は失敗し、ドライバーは例外をスローします。 有効な値 auto | kerberos | ntlm | ntlmjava | ntlm2java | userIdPassword 注意 LMCompatabilityLevel が NTLMv2 に制限されているときに AuthenticationMethod=ntlmjava を指定すると、エラーが返されます。LMCompatabilityLevel が NTLMv2 に制限されている場合、AuthenticationMethod を ntlm2java に設定する必要があります。 AuthenticationMethod=ntlmjava または AuthenticationMethod=ntlm2java を指定する場合、データベースを管理するドメイン・サーバーの名前を指定する必要があります。ドメイン・サーバーは、ドメイン・プロパティーを使用して指定できます。 ドメイン・プロパティーが指定されていない場合、ドライバーはユーザー・プロパティーからドメイン・サーバーを判別しようとします。ドライバーがドメイン・サーバー名を判別できない場合は、例外がスローされます。 ユーザー・プロパティーはユーザー ID を提供します。パスワード・プロパティーはパスワードを提供します。 値「type4」、「type2」、および「なし」は非推奨ですが、後方互換性のために認識されます。代わりに、kerberos、ntlm、および userIdPassword 値をそれぞれ使用してください。 Guardium ユーザーが Azeri_Cyrillic_100_CI_AS または Chinese_Hong_Kong_Stroke_90_CI_AS などの非標準のデータベース Unicode を使用している場合は、このパラメーターを接続プロパティーに追加します。 CodePageOverride=UTF-8 SSL (Force encryption=Yes) を使用している場合、以下を追加します。 encryptionMethod=SSL;validateServerCertificate=false
MS SQL サーバー (DataDirect)	1433

データベース・タイプ	ポート番号
MongoDB	27017
MySQL	3306
Netezza	5480
Oracle (DataDirect)	1521
PostgreSQL	5432
SAP HANA	39015
Sybase	4100
Sybase IQ	2638
Teradata	1025
テキスト	0
Text:HTTP	8000
Text:FTP	21
Text:SAMBA	445
Text:HTTPS	8443
N_A	0
MS SQL サーバー (オープン・ソース) (「強化」 > 「脆弱性評価」 > 「カスタム・アップロード」を使用して、これらの JDBC ドライバーをアップロードします。『サブスクライブしたグループのアップロード』を参照してください。)	1433
Oracle (オープン・ソース) (「強化」 > 「脆弱性評価」 > 「カスタム・アップロード」を使用して、これらの JDBC ドライバーをアップロードします。『サブスクライブしたグループのアップロード』を参照してください。)	1521
HIVE、HiveServer2	10000
HADOOP、Hive CLI は非推奨	9083
HIVE、Hue からの Impala	21050
HADOOP、Impala シェル	21000
HUE、Oracle Hue バックエンド	1521
HUE、MySQL Hue バックエンド	3306
HUE、PostgreSQL Hue バックエンド	5432
WEBHDFS	50070

注: SSL データ・ソースを使用して初めて接続しようとすると、接続のテスト中にこのエラーが発生することがあります。

エラー
接続に失敗しました

```
Could not connect to: 'jdbc:db2://su11u1x64t-va:55000/VA_DB' for user: '(DELETE ME) db2 10.1 SSL_DB2(Security Assessment)'.
DataSourceConnectException: Could not connect to: 'DB2 (DELETE ME) db2 10.1 SSL 9.70.146.39:55000' for user: 'db2inst1'.
Exception: com.ibm.db2.jcc.am.DisconnectNonTransientConnectionException: [jcc][t4][2030][11211][4.15.134] A communication
error occurred during operations on the connection's underlying socket, socket input stream,
```

これは、メモリーにロードされた証明書の正しい鍵ストア・ファイルが GUI にないためです。これを修正するには、GUI を再起動してください。そうすると、このエラーが解決して、接続が成功するはずです。

12. データ・ソース・タイプに応じて、このダイアログの「ポート」の後のフィールドが若干異なります。
    - Db2 の場合は、データベース名を入力します。
    - Db2 iSeries または Oracle の場合は、サービス名を入力します。
    - Informix の場合は、Informix サーバー名を入力します。
    - テキスト以外のデータベース・タイプの場合は、「データベース」ボックスにデータベース名を入力します (Informix、Sybase、MS SQL サーバー、PostgreSQL、または Teradata の場合のみ)。Sybase または MS SQL サーバーの場合にこれをブランクのままにすると、デフォルトでマスターが使用されます。Sybase データベースの場合、「データベース」テキスト・ボックスにはデータベース名を指定するか、ブランクにした場合はマスターにデフォルト設定されます (これは「資格レポート」および「分類」の場合に機能します。 VA の場合はデータベース・インスタンス名を使用してください。)
    - Db2、Db2 iSeries、または Oracle の場合は、「スキーマ」ボックスに使用する有効なスキーマ名を入力します。
    - テキスト・ファイルのデータベース・タイプの場合は、「ファイル名」ボックスにファイル名を入力します。
13. このデータ・ソースとの JDBC 接続を確立するために、追加の接続プロパティーを JDBC URL に含める必要がある場合にのみ、「接続プロパティー」ボックスを使用します。使用するフォーマットは、「property=value」である必要があります。このとき、プロパティーと値の各ペアはセミコロンで区切ります。
    - Roman8 をデフォルトの文字セットとする Sybase データベースの場合、次のプロパティーを入力します。 charSet=utf8
    - Oracle 暗号化接続の場合は、接続プロパティーを oracle.net.encryption_client=REQUIRED;oracle.net.encryption_types_client=RC4_40 (モニター対象インスタンスで必要とされる暗号化アルゴリズム (タイプを問わない) と置換) のように定義する必要があります。
    - 3DES168 暗号化には問題があることに注意してください。 3DES168 暗号化を使用するよう定義されたデータ・ソースは、SQL エラーの検出時に誤って「ORA-17401 プロトコル・エラー」または「ORA-17002 チェックサム・エラー」をスローします。その後、接続を閉じてから再度開くまで、接続が機能しなくなります。
    - Db2 暗号化接続の場合は、接続プロパティーを securityMechanism=13 のように定義する必要があります。
    - Db2 iSeries 接続の場合は、接続プロパティーを property1=com.ibm.as400.access.AS400JDBCDriver;translate binary=true のように定義します。

- Db2 z/OS データ・ソースの場合、データベース・パフォーマンスを向上させるために、次の接続プロパティーを追加してください。resultSetHoldability=2
- Oracle では、sys は Oracle デフォルト・ユーザーであり、データベース・インスタンスの所有者であり、かつスーパーユーザー特権を持ちます。これは Unix の root に似ています。 SYSDBA はロールであり、データベースの始動と停止やバックアップ/リカバリー操作の実行などの多くのハイレベルな管理操作を行うために必要な管理特権を持ちます。 このロール (SYSDBA) を他のユーザーに与えることもできます。 `sys as SYSDBA` 句は、sys ユーザーとして接続するのに必要な接続方式を参照します。
- Oracle 10 (sys as SYSDBA) のモニター値 (これは Oracle オープン・ソース・ドライバー用です) には、internal_logon=sysdba を入力します。
- DataDirect (Oracle ドライバー) の場合は SysLoginRole=sysdba を入力します。
- さらに、CRYPTO_CHECKSUM_TYPES を sqlnet.ora 内で使用する場合は、以下の例を使用してください。
  - oracle.net.encryption_client=aes256;oracle.net.crypto_checksum_types_client=SHA1
  - oracle.net.encryption_client=rc4_256;oracle.net.crypto_checksum_types_client=MD5
  - oracle.net.encryption_client=aes256;oracle.net.crypto_checksum_types_client=MD5
  - oracle.net.encryption_client=rc4_256;oracle.net.crypto_checksum_types_client=SHA1
- 例: OID と呼ばれる Oracle LDAP に対する認証を使用します。 必要な値は、LDAP サーバー・ホストまたは IP、LDAP サーバー・ポート、Oracle インスタンス名、およびレルムです。カスタム URL は、次のように正確に入力する必要があります。
  jdbc:guardium:oracle:@ldap://wi3ku2x32t4:389/on0maver;cn=OracleContext;dc=vguardium;dc=com

14. 必要に応じて、データ・ソースに対する「カスタム URL」接続文字列を入力します。 「カスタム URL」フィールドがブランクの場合は、他のデータ・ソース定義フィールドに入力したプロパティー (ホスト、ポート、インスタンスなど) を使用して接続が行われます。
    重要:
    - 「カスタム URL」フィールドを Oracle オープン・ソース形式で指定する場合は、`jdbc:guardium:oracle://;SID=<SID>` と指定します。
    - Oracle Advanced Security を有効にして Oracle データベースのデータ・ソースを作成する場合は、データ・ソース定義の「カスタム URL」フィールドに `EncryptionLevel=required` と指定します。
15. 「拡張オプションの表示」をクリックして、ロールおよび CAS オプションを表示します。
16. オプションで「ロール」をクリックして、データ・ソースのロールを割り当てます。データ・ソースにロールを追加すると、ユーザーはデータ・ソース構成を表示できます。所有者および管理者だけがデータ・ソースを変更および削除できます。

    ベンダーはインストールに柔軟性を持たせるよう工夫しているため、データ・ソース定義で必要な 2 つのフィールドをユーザーが決めるようになっています。

    CAS では、UNIX でデータベース・ツールの一部を実行するためのデータベース・インスタンス・アカウントと、モニター対象のファイルを検出するためのデータベース・インスタンス・ディレクトリーの名前の、2 種類の情報が必要です。一般的に、データベース・インスタンス・アカウントおよびディレクトリーがデータ・ソース定義に正しく入力されない場合、CAS でデータを検出できなかったテストで使用できる `CAS データがない` というメッセージが表示されます。

    a. CAS により使用される「データベース・インスタンス・アカウント」(ソフトウェア所有者) および「データベース・インスタンス・ディレクトリー」(データベース・ソフトウェアがインストールされたディレクトリー) を入力します。
    以下に、データ・ソース用の CAS 情報を入力するために必要な情報を見つける方法の推奨事項を示します。この情報は、インストール済み環境によって異なる場合があります。UNIX で使用する方法の 1 つに、特定のデータベース・インストール済み環境で /etc/passwd ファイルをリストして、データベース・インスタンス・アカウントおよびインスタンス・ディレクトリーを指定できるよう使用するといった方法があります。インストール中の任意の時点で、インスタンス・ディレクトリー (ORACLE_HOME など) を指定する環境変数が、データベース・インスタンス・アカウントに定義されます。この場合、データ・ソース定義フォームの「データベース・インスタンス・ディレクトリー」フィールドに「$ORACLE_HOME」と入力します。この変数が展開され、データベース・サーバー上の正しいディレクトリー名が検出されます。
    注: 複数のディレクトリーを検索するには、「データベース・インスタンス・ディレクトリー」に複数のファイル・パスを定義します。この例は、MongoDB の行を参照してください。

表 1. データベース・インスタンス

データベース・タイプ	データベース・インスタンス・アカウント	データベース・インスタンス・ディレクトリー / 追加ヒント
Db2	通常は db2inst1	db2inst1 のホーム・ディレクトリーまたは Windows の C:¥Program Files¥IBM¥SQLLIB。 プログラム db2cmd.exe は、システム・パス上、またはデータベース・インスタンス・ディレクトリーの「bin」サブディレクトリー内にある必要があります。
Informix	通常は informix	UNIX の「/opt/IBM/informix」など、または「C:¥Program Files¥IBM¥Informix」。環境変数 INFORMIXDIR を定義できます。 プログラム <servicename>.cmd はシステム・パス上にある必要があります。ここで、<servicename> はデータ・ソース定義の「Informix サーバー」に入力されている値です。

データベース・タイプ	データベース・インスタンス・アカウント	データベース・インスタンス・ディレクトリー / 追加ヒント
MongoDB	通常は mongodb または mongos	MongoDB では、データベース・インスタンス・ディレクトリーに複数のパスを指定する必要があります。パスを区切るにはパイプ記号 (|) とスペースを使用します。 例えば、/var/lib/mongo | MongoBinary=/usr/bin | dbpath=/var/lib/mongo | logpath=/var/log/mongodb | keytab=/home/keytab | dbdumppath=/opt/backup | sslpath=/etc/ssl | keyfile=/home/mongod/mongo_server.keyfile です。 /var/lib/mongo パスは mongo ユーザーのホーム・パスなので必須です。 MongoBinary=/usr/bin は、mongo バイナリーのパスです。変数 (大/小文字の区別あり) を指定し、その後に等号とパスを指定する必要があります。 dbpath=/var/lib/mongo は、データ・ファイルのパスです。このケースでは、偶然 MongoDB ホーム・ディレクトリーと同じになっています。 logpath=/var/log/mongodb は、MongoDB ログのパスです。 keytab=/home/keytab は、MongoDB キータブ・ファイルのディレクトリーです。 dbdumppath=/opt/backup は、MongoDB バックアップ・ダンプのディレクトリーです。 sslpath=/etc/ssl は、MongoDB SSL ファイルのパスです。 keyfile=/home/mongod/mongo_server.keyfile は、MongoDB 鍵ファイルを指しています。 リストされたパスをすべて定義する必要はありません。定義されていないパスは分析されません。
Oracle	通常は oracle、またはバージョンが特定された oracle9 または oracle10 など	例えば、UNIX の/home/oracle9や、Windows のC:¥oracle¥product¥10.2.0¥db_1です。環境変数 ORACLE_HOME を定義できます。 Windows では、環境変数 PERL5LIB および ORACLE_HOME を定義する必要があります。また、プログラム「opatch.bat」はシステム・パス上にある必要があります。
SQL サーバー	Windows 認証が使用されている場合を除き、必要ありません。Windows 認証が使用されている場合は、Windows 認証で受け入れられる「ドメイン/ユーザー名」という形式である必要があります。	SQL サーバーで CAS を使用するために「データベース・インスタンス・ディレクトリー」にデータを取り込むには、2 つの方法があります。 脆弱性評価のテストにデータ・ソースを使用している場合は、この列にデータベース・インスタンス・ホーム・ディレクトリーを設定する必要があります。 例 MSSQL2008 C:¥Program Files¥Microsoft SQL Server¥MSSQL10.MSSQLSERVER¥MSSQL MSSQL2014、デフォルト・インスタンス C:¥Program Files¥Microsoft SQL Server¥MSSQL12.MSSQLSERVER¥MSSQL MSSQL2016、名前インスタンス C:¥Program Files¥Microsoft SQL Server¥MSSQL13.SQL2016¥MSSQL 脆弱性評価以外のテストで、CAS によるファイルまたはレジストリーのモニター用にデータ・ソースを使用している場合があります。 その場合、この列は「Program Files」の「Microsoft SQL Server」ディレクトリーになります。 例: C:¥Program Files (x86)¥Microsoft SQL Server または C:¥Program Files¥Microsoft SQL Server 注: 脆弱性評価テストおよび CAS によるファイルのモニターを行う場合は、データ・ソースが 2 つ必要です。
Sybase	通常は「sybase」	UNIX の場合は /home/sybase、Windows の場合は C:¥sybase です。環境変数 SYBASE を定義できます。
MySQL		環境変数 MYSQL_HOME を定義できます。 注: データベース名が Unicode の MySQL データ・ソースはサポートされていません。MYSQL のデータ・ソース名は ASCII でなければなりません。
Teradata		必要ありません。インストール済み環境の構造はすべて同じように見えます。

データベース・タイプ	データベース・インスタンス・アカウント	データベース・インスタンス・ディレクトリー / 追加ヒント
Netezza		必要ありません。インストールは、すべてのマシンで同じロケーションにあります。
PostgreSQL		これは、最も柔軟なインストールです。ユーザーは、Postgres データベース・サーバーに 2 つの環境変数を定義する必要があります。PostgreSQL_BIN はインストールのバイナリーのロケーション、PostgreSQL_DATA はデータのロケーションである必要があります。

注: 「データベース・インスタンス・ディレクトリー」フィールド内で環境変数を使用する場合は、データベース・サーバーでその環境変数が定義されている必要があります。

b. データ・ソースの「重大度分類」 (または影響レベル) を選択します。レポートや結果の表示中には、重大度分類を使用してデータ・ソースのソート、フィルタリング、フォーカスを行うことができます。
c. 「保存」をクリックして、データ・ソース定義を保存します (定義が保存されるまで、ロールまたはコメントを追加できません)。
d. オプションで「コメントの追加」をクリックして、定義にコメントを追加します。
e. オプションで「接続のテスト」をクリックして、定義したデータ・ソースの接続をテストします。
f. 定義が完了したら、「閉じる」をクリックします。

親トピック: データ・ソース

既存のデータ・ソースの操作

データ・ソース定義を作成したら、そのデータ・ソースのコピー、変更、または削除を行うことができます。

手順

- 「データ・ソース・ビルダー」を開きます。これを行うには、「設定」 > 「データ・ソース定義」にナビゲートします。
- 「アプリケーション選択」メニューに、データ・ソース定義を使用できるアプリケーションがすべてリストされます。 変更したいデータ・ソースが作成された対象のアプリケーションを選択し、「次へ」をクリックして「データ・ソース・ファインダー」に進みます。

親トピック: データ・ソース

データ・ソースのコピー

手順

- 「データ・ソース・ファインダー」からコピーするデータ・ソースを選択し、「コピー」をクリックします。
- データ・ソース定義の作成時に入力した情報が「データ・ソース定義」ダイアログに表示され、元のデータ・ソース名の前に「copy Of」と表示されます。 必要なフィールドに変更を加えます。
- 「適用」をクリックし、コピーしたデータ・ソースを保存します。

データ・ソースの変更

手順

- 「データ・ソース・ファインダー」から変更するデータ・ソースを選択し、「変更」をクリックします。
- データ・ソース定義の作成時に入力した情報が「データ・ソース定義」ダイアログに表示されます。 必要なフィールドに変更を加えます。
- 「適用」をクリックし、データ・ソースに加えた変更内容を保存します。

データ・ソースの削除

手順

「データ・ソース・ファインダー」から削除するデータ・ソースを選択し、「削除」をクリックします。

データ・ソースについてのレポート

Guardium® は、現行環境内のデータ・ソースと、それらに加えられた変更内容についてのレポートを提供します。

手順

- 「データ・ソース」レポートを開きます。これを行うには、「レポート」 > 「レポート構成ツール」 > 「データ・ソース」にナビゲートします。 表示される表に、すべてのデータ・ソースと、各データ・ソース定義に保管されている情報がリストされます。
- 表内のセルを右クリックすると、「データ・ソース・バージョン履歴」と「呼び出し」の 2 つのオプションが表示されます。
  - データ・ソース定義に加えられた変更内容を表示するには、「データ・ソース・バージョン履歴」をクリックします。
  - データ・ソースに使用可能な API のいずれかを選択して実行するには、「呼び出し」をクリックします。

注: 鉛筆のアイコンをクリックすると、データ・ソース・レポートのランタイム・パラメーターと表示パラメーターをカスタマイズできます。

親トピック: データ・ソース
関連概念:
GuardAPI データ・ソース関数

サービス名を使用したデータ・ソースの定義

カスタム URL を使用することで、ユーザーがサービス名を使用して Oracle データベースに接続できるようにするデータ・ソースを定義できます。

このタスクについて

カスタム URL のほか、ホスト名、ポート、サービス名を入力する必要があります。

手順

1. Oracle サービス名を決定します。 次のようなコマンドを使用できます。

```
SQL> set line size 5000;
SQL> select host_name, instance_name from v$instance;
SQL> select name from v$database;
SQL> show parameter service
```

「値」列に表示される名前を使用します。
2. 適切な Oracle JDBC シン・ドライバーを Guardium システムにロードします。
   a. 次の URL から、Oracle データベース用のドライバーを見つけてダウンロードします。http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-112010-090769.html
   b. 「カスタム・アップロード」ウィンドウを開きます。これを行うには、「強化」 > 「脆弱性評価」 > 「カスタム・アップロード」にナビゲートします。
   c. 「Oracle JDBC ドライバーのアップロード」というタイトルのセクションを見つけます。 「参照」をクリックし、ファイルをダウンロードした場所を参照します。「すべてにオープン・ソース・ドライバーを使用」をクリックします。
   d. アップロードが完了したら、Guardium ユーザー・インターフェースを再始動します。
3. このデータベースのデータ・ソースを定義します。
   a. 「データ・ソース・ビルダー」を開きます。これを行うには、「設定」 > 「データ・ソース定義」にナビゲートします。
   b. 「アプリケーション選択」メニューに、データ・ソース定義を使用できるアプリケーションがすべてリストされます。 変更したいデータ・ソースが作成された対象のアプリケーションを選択し、「次へ」をクリックして「データ・ソース・ファインダー」に進みます。
   c. 「サービス名」フィールドにサービス名を入力します。「カスタム URL」フィールドに、「`jdbc:oracle:thin@//hostname:port/svcname`」と入力します。ここで、hostname と port はデータベースの標準値、svcname はサービス名 (「サービス名」フィールドに入力した値と同じ) です。

親トピック: データ・ソース

KDC 定義の管理

データ・ソースが Kerberos を使用する認証を必要とする場合、接続を確立する前に、Guardium が Kerberos チケットを取得するために必要な情報を指定できます。

このタスクについて

Guardium V.10.1.3 以降、KDC を特定のデータ・ソースまたは管理対象ユニット・グループに割り当てて、Guardium 認証を Mongo データベースおよび Hive データベースに提供できます。アプライアンスは JDBC 接続を介してチケットを取得するため、ユーザーは自分自身でチケットを取得する必要はありません。これは、アプライアンス自体が使用するように設定されているものとは無関係であることに注意してください。

最大 5 つの Kerberos 鍵配布センター (KDC) を中央マネージャーで定義し、1 つをスタンドアロンの Guardium で定義できます。鍵配布センターを Guardium に追加するには、次のように指定します。

- レルム: 大文字のドメイン・ネーム
- KDC: Kerberos サーバーのホスト名
- Kerberos チケットの暗号化タイプ
  - des-cbc-md5
  - des-cbc-crc
  - rc4-hmac
  - des3-cbc-sha1
  - aes128-cts-hmac-sha1-96
  - aes256-cts-hmac-sha1-96

  デフォルトは aes256-cts-hmac-sha1-96 です。これは最も安全な暗号化タイプです。

手順

1. 「設定」 > 「ツールとビュー」 > 「Kerberos 構成」をクリックします。
2.  をクリックして、新しい構成を作成します。
3. 「名前」、「KDC」、および「レルム」を指定します。
4. 「暗号化タイプ」を指定します。デフォルトは aes256-cts-hmac-sha1-96 です。
5. 「保存」をクリックします。

次のタスク

Kerberos KDC を作成したら、データ・ソースのセットアップを構成するときに、それを選択できます。

親トピック: データ・ソース

クラウド・データベース・サービス保護

クラウド・データベース保護は、クラウド・データベースにおける分類、脆弱性評価、およびオブジェクト監査を提供します。

Guardium とクラウドの接続を一度セットアップすると、以下を実行できるようになります。

- データベース・インスタンスを検出し、それらを Guardium にカタログします。
- 分類プロセスにカタログされたデータ・ソースを割り当てるか、新しいプロセスを作成します。分類はクラウド・データベースに対して実行され、定義されたルールに従ってオブジェクトを識別します。
- カタログされたデータ・ソースを脆弱性評価プロセスに割り当てるか、新しいプロセスを作成します (有効な VA ライセンスが必要)。VA はクラウド・データベースに対して実行され、そのデータを Guardium レポートで使用します。
- DB 監査の有効化: Oracle 標準監査データが、インストールされたポリシーに従って、Guardium レポート用にクラウドからプルされます。(https://docs.oracle.com/cd/B28359_01/server.111/b28337/tdpsg_auditing.htm#TDPSG50051 の Oracle 定義を参照してください)。
- オブジェクト監査を有効にします (Oracle の監査証跡)。分類結果を確認し、オブジェクト監査の対象とするオブジェクトを選択します。(DB 監査が有効になっている必要があります)。オブジェクト監査は、オブジェクトに対して実行されるすべてのアクティビティーを追跡します。Guardium はこのデータをレポートや調査ダッシュボードなどに使用します。Guardium は、データ・ソース別にオブジェクトを自動的に追加するように構成できます。さらに、そのすべてのデータ・ソースで継承される、アカウントごとのデフォルトも設定できます。 これは監査を必要としていて、それ以上の評価は必要ないオブジェクトを持つデータベースに特に有用です。 分類プロセスでの検出が予期される妥当な上限を設定します。分類に誤りがあった場合でもオブジェクトがオーバーフローしないようにするために、設定値は高すぎないようにします。(オーバーフローはデータベースのパフォーマンスに影響を与える可能性があります)。

クラウド DB で Guardium 機能を実行するには、AWS 権限が必要です。AWS IAM 定義を参照してください。

オンプレミス・データベースでは、データベースにインストールされている S-TAP は、すべてのデータベース・トラフィックを Guardium システムに送信します。 クラウド環境では、Guardium はクラウド DB からログ・ファイルをプルし、S-TAP データと同じようにデータを処理します。相違点として、S-TAP ではすべてのデータベース・アクティビティーが記録されるのに対し、クラウド環境では選択した表のみが監査されます。クラウドからのデータの取得が多少遅れる可能性があるという相違点もあります。

監査対象のデータベースおよびオブジェクトに対するアクティビティーは、データベース・ログに書き込まれます。ログ・アクティビティーの量は、モニター対象項目の数により増大します。大量のログ・アクティビティーは、データベースのパフォーマンスに影響を与える可能性があります。すべての関連データが収集されており、なおかつシステムが過負荷になっていないことを確認する必要があります。

CM 環境内およびスタンドアロンの Guardium コレクター上で、クラウド・データベース・サービス保護を実行できます。

クラウド DB サービス保護のコンテキストでは、データベースはクラウド上のデータベースのことであり、データ・ソースは Guardium カタログ・データベースのことです。

1 つの Guardium システムのみが、任意の 1 つの DB の DB 監査およびオブジェクト監査を所有できます。他の Guardium システムも同じクラウド・アカウントにアクセスし、DB 詳細を表示できますが、DB 監査を無効にしたり、オブジェクト監査データにアクセスしたりすることはできません。例えば、ある Guardium システムがダウンして復旧を予期できない場合は、そのシステムから別のシステムに所有権を移動することができます。

すべての AWS RDS データベース・エンジンで、ディスカバリー、分類、および VA がサポートされます。

データベース監査の制限事項

- RDS 定義を (例えば、DB インスタンスの削除や資格情報の変更において) 最新の状態に保つ必要があります。
- Guardium v10.5 は、AWS クラウド上の Oracle V.11 データベースおよび Oracle V.12 データベースをサポートします。Oracle 12 監査にはログイン・レコードが含まれていないため、クライアント IP は使用できません。
- 戻りデータのパターンの編集やテストなどの抽出ルールはサポートされません。
- バインド変数値のロギングや影響を受けるレコードなどの戻りデータはサポートされません。
- S-GATE ターミネート、無視、照会再書き込みなど、S-TAP と対話するルール・アクションはサポートされません。
- 失敗したログインは Oracle 監査ではキャプチャーされないため、Guardium に転送されません。
- Oracle 監査でキャプチャーされないステートメント (例えば、構文エラーが含まれているステートメント) は、モニターできません。
- 監査データはバインド変数値を持つがタイプは持たない (例: 123) ため、SQL で置換されたときは、対象を囲む引用符が常に追加されます。
- 変数値に ASCII 制御文字 (「¥001」など) や複数バイト文字が含まれる場合、監査ファイルはダウンロードできません。
- Blob バインド変数値はサポートされません。

- クラウド・データベース・サービス保護のワークフロー
- AWS IAM 定義
  必要な権限に応じて、AWS アカウントの IAM ポリシーを定義します。
- クラウド・アカウントの作成、変更、削除
  DB 資格情報を使用して、クラウド・データベース・サービス・アカウントを作成するか、クラウド・アカウントを変更または削除します。
- クラウド・データベースのディスカバー
  検索するリージョンを選択することで、クラウド・アカウント内のデータベースをディスカバーします。
- データベースのカタログおよび管理
  Guardium でデータ・ソースを作成するためのデータベースをカタログし、ユーザーとパスワードを変更し、データベース構成を更新します。
- 分類および脆弱性評価の管理
  データ・ソースを既存の分類プロセスまたは脆弱性評価プロセスに割り当てます。または新規プロセスを作成します。
- データベース監査の構成
  データベース上での監査を有効にし、オブジェクト監査データを Guardium がプルできるようにします。 分類に自動的に追加されるオブジェクトの制限を変更し、コレクターを変更します。
- オブジェクト監査の管理
  管理しているデータベースの分類プロセスによって識別された潜在的な機密オブジェクトを表示し、これらのオブジェクトに実行されるすべてのアクティビティーをモニターするために、選択したオブジェクトに対するオブジェクト監査を有効にします。

親トピック: ディスカバー

クラウド・データベース・サービス保護のワークフロー

このタスクについて

これは一般的なワークフローです。具体的なワークフローは、クラウド・データベース監査の目的に応じて異なります。

手順

1. クラウド・アカウントを作成します。
2. そのデータベース・インスタンスをディスカバーします。
3. 処理するデータベースをカタログします。カタログにより Guardium 内にデータ・ソースが作成され、特定のデータベース上のクラウド・データベース Guardium 機能を管理できます。
4. 必要に応じて、新規または既存の VA プロセスにデータ・ソースを追加します (脆弱性評価ライセンスが必要です)。
5. 必要に応じて、新規または既存の分類プロセスにデータ・ソースを追加します。
6. オプションで、関連データベース上で DB 監査を有効にして、今すぐ Guardium UI から、または後で DB コンソールから、データベースを再始動します。DB 監査は、有効になると標準 Oracle 監査を実行します。DB 監査を有効にすると、Guardium システムは、その DB 上の DB 監査の固有の所有者になります。他の Guardium システムは、DB 監査およびオブジェクト監査を変更できません。分類結果を表示するには、DB 監査を有効にした後に分類を 1 回実行するか (今すぐ 1 回実行)、またはスケジュールされた次の実行を待機します。(データ・ソースを分類プロセスに割り当てる必要があります。)
7. データ・ソースの分類結果を次のように確認します (分類プロセスと DB 監査が必要です)。
   - オブジェクトごとまたはオブジェクトを識別した分類プロセスごとにオブジェクトをグループ化して表示し、結果をさらに絞り込むためにフィルターを使用します。
   - オブジェクト監査を、個別、または表別に有効または無効にします。
   - オブジェクト・グループからドリルダウンして、分類結果内の選択したオブジェクトを含むすべてのデータベースのリストを開きます。このビューで、オブジェクト監査を有効/無効にすることもできます。
8. 定期的にステップ 2 から 7 を繰り返します。
9. 定期的にデータ・ソースをレビューして、新しいオブジェクトを確認し、オプションでオブジェクト監査のオブジェクトを追加または削除します。例えば、自動的に追加されたオブジェクトに監査を必要としないと決定したオブジェクトが含まれているか、データベースにパフォーマンスの問題がある場合は、オブジェクトを削除できます。または、監査されていない疑わしいオブジェクトを特定し、それをオブジェクト監査に追加することができます。

親トピック: クラウド・データベース・サービス保護

AWS IAM 定義

必要な権限に応じて、AWS アカウントの IAM ポリシーを定義します。

IAM の最小限の権限には、構成の表示とタグの変更が含まれます。DB 監査の有効化や DB の再始動は含まれません。この JSON では最小限の権限を定義します。これがないと、クラウド・データベース・サービス保護を実行することはできません。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "rds:DescribeDBParameters",
                "rds:DescribeDBInstances",
                "rds:DescribeDBParameterGroups",
                "rds:DownloadDBLogFilePortion",
                "rds:DescribeDBLogFiles",
                "rds:ListTagsForResource",
                "rds:RemoveTagsFromResource",
                "rds:AddTagsToResource",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeVpcs"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

全権限はこれらのパラメーターで使用可能になります。

インスタンスに対する DB 監査を有効、無効にする
構成されていない場合、「DB 監査を有効にする」ボタンと「DB 監査を無効にする」ボタンはグレー表示になり、AWS コンソールで DB インスタンスを有効または無効にするよう DBA に要求する必要があります。

```
"rds:CopyDBParameterGroup",
"rds:CreateDBParameterGroup",
"rds:ModifyDBInstance",
"rds:ModifyDBParameterGroup",
```

DB インスタンスを再始動する
構成されていない場合、「再始動」ボタンはグレー表示になり、AWS コンソールで DB インスタンスを再始動するよう DBA に要求する必要があります。

```
"rds:RebootDBInstance",
```

サポートされているプラットフォームが EC2 の場合のセキュリティー・グループの操作
構成されていない場合は、DBA が Guardium IP をセキュリティー・グループに追加する必要があります。構成されている場合は、Guardium がその IP を DB インスタンスのセキュリティー・グループに追加します。ネットワーク構成により、Guardium システムが自身の IP を識別できない場合は、DBA が AWS コンソールで IP を追加する必要があります。

```
 "rds:ModifyDBInstance"
"rds:AuthorizeDBSecurityGroupIngress",
"rds:CreateDBSecurityGroup",
```

サポートされているプラットフォームが VPC の場合のセキュリティー・グループの操作
構成されていない場合は、DBA が Guardium IP をセキュリティー・グループに追加する必要があります。構成されている場合は、Guardium がその IP を DB インスタンスのセキュリティー・グループに追加します。ネットワーク構成により、Guardium システムが自身の IP を識別できない場合は、DBA が AWS コンソールで IP を追加する必要があります。

```
 "rds:ModifyDBInstance"
"ec2:AuthorizeSecurityGroupIngress",
"ec2:CreateSecurityGroup",
```

これらのパラメーターの構成時に、Guardium は、コレクターのパブリック IP CIDR マスクを 24 に指定して、RDS インスタンス・セキュリティー・グループにインバウンド・ルールを作成します。

親トピック: クラウド・データベース・サービス保護

クラウド・アカウントの作成、変更、削除

DB 資格情報を使用して、クラウド・データベース・サービス・アカウントを作成するか、クラウド・アカウントを変更または削除します。

親トピック: クラウド・データベース・サービス保護

クラウド・アカウントの作成

このタスクについて

前提条件: AWS IAM ポリシーを定義します。AWS IAM 定義を参照してください。

ヒント: このアカウントで多数のデータベースを管理している場合は、デフォルトの分類プロセスを定義することを検討してください。それにより、ディスカバーされた各データベースのプロパティーを定義する手間を省けます。

手順

1. 「ディスカバー」 > 「データベース・ディスカバリー」 > 「クラウド DB サービスの保護」をナビゲートします。
2.  をクリックして、「クラウド DB サービス・アカウント定義の作成 (Create Cloud DB Service Account Definition)」ペインを開きます。
3. アカウントを定義します。
   - 固有アカウント名
   - プロバイダー
   - クラウド・サービス・プロバイダーから提供される固有のアクセス・キー ID および秘密アクセス・キー ID。アカウント秘密鍵はパスワードとして機能します。アクセス・キーとタイトルの両方を固有にして、同じ access_id で複数のアカウント名を使用できないようにする必要があります。
   - 自動的に追加されるオブジェクトを制限 (オプション): これは DB 監査が有効である場合に、オブジェクト監査に対して自動的に有効にできる、分類により検出されるオブジェクトの最大数です。これは、ディスカバー後、データベースごとに変更できます。有効にされたオブジェクトは、「管理対象オブジェクト (Managed Objects)」ウィンドウで自動的に「有効」と表示されます。Guardium にオブジェクトを自動的に追加させるには、分類プロセスでの検出が予期される妥当な上限を設定します。分類に誤りがあった場合でもオブジェクトがオーバーフローしないようにするために、設定値は高すぎないようにします。(オーバーフローはデータベースのパフォーマンスに影響を与える可能性があります)。ゼロ (0) は、オブジェクト監査に対して自動的に有効になるオブジェクトがないことを意味します。 監査済みのオブジェクトの数に、新しく分類されたオブジェクトの数を加えると、この制限を超える場合、新しいオブジェクトは、オブジェクト監査が有効になりません。例えば、15 に設定され、分類が最初の実行時に 5 個のオブジェクトを識別する場合、その 5 個のオブジェクトに監査証跡が割り当てられます。設定が 15 であり、既に 5 つのオブジェクトがオブジェクト監査に有効になっており、分類の次の実行で 16 オブジェクトが識別されると、新しいオブジェクトはどれもオブジェクト監査が有効になりません。設定が 15 であり、既に 5 つのオブジェクトがオブジェクト監査に有効になっており、分類の次の実行で 5 オブジェクトが識別されると、新しい 5 オブジェクトはオブジェクト監査で有効になります。
4. オプションで、デフォルトの分類を定義します。このアカウントのすべてのカタログ済みデータベースは、この分類プロセスに割り当てられます。分類プロセスは、カタログ後に、データベースごとに変更できます。
5. クラウドへのアクセスをテストします。
   a. 「アクセスのテスト」をクリックします。Guardium はクラウドへのアクセスを試行します。
   b. Guardium がクラウドへのアクセスに失敗する場合は、Guardium システムが Amazon にアクセスできることを確認します。指定したキーを調べます。
6. 「作成」をクリックします。
   アカウントが作成され、「クラウド DB サービス・アカウント」リストは新しいクラウド・アカウントで更新され、そのアカウントの詳細が右ペインに表示されます。

次のタスク

データベースを検出してカタログし、分類と脆弱性評価、およびオブジェクト監査をセットアップします。

クラウド・アカウントの変更

プロバイダー以外のすべてのパラメーターを変更できます。

手順

1. 「クラウド DB サービス・アカウント」からクラウド・アカウントを選択し、右ペインの  をクリックします。
2. 構成を変更します。
3. 変更された資格情報がある場合は、「アクセスのテスト」をクリックしてクラウドへのアクセスをテストします。
4. 「保存」をクリックします。

クラウド・アカウントの削除

アカウントを削除すると、現在の環境によって所有されているすべてのデータベース上のオブジェクト監査と DB 監査が無効になります。

手順

1. 「クラウド DB サービス・アカウント」ペインでアカウントを選択し、  をクリックして、確認します。
2. DB を DB コンソールから再始動します。DB への Amazon アクセス権限がない場合は、DB 監査を無効にして DB を再起動するように DBA に依頼してください。 監査を停止し、DB を再始動することで、DB が Guardium によって使用されるログ・ファイルへの書き込みを停止することは重要です。

クラウド・データベースのディスカバー

検索するリージョンを選択することで、クラウド・アカウント内のデータベースをディスカバーします。

このタスクについて

ディスカバーを実行すると、「データベース」表にデータが追加されて更新されます。ディスカバーされたデータベースは、そのデータベースがまだクラウドにあるかどうかに関係なく、表に残ります。

「ディスカバリー」 > 「データベース・ディスカバリー」 > 「クラウド DB サービスの保護」にナビゲートするたびに、Guardium はクラウド内の DB 監査ステータスが UI で報告されるステータスと異なる場合に、データベース表の上に次のメッセージを表示して通知します。「一部のデータベースで DB 監査状況が変更されました。「リフレッシュ」をクリックして表を更新してください」。このメッセージが表示されたら、「リフレッシュ」をクリックして表示を最新表示します。

この確認は「状況の取得」をクリックしてオンデマンドで実行することもできます。 この取得には数分かかる場合があります。完了したら、DB 監査状況が変更されている場合にのみメッセージが表示されます。変更がある場合は、「リフレッシュ」をクリックします。

クラウド・データベース定義は CSV ファイルでもアップロードできます。必須パラメーターは『GuardAPI クラウド・データ・ソース関数』にリストされます。API パラメーター cloudTitle をパラメーター environmentTitle (機能は同じですが、名前は異なります) に置き換える必要があります。『カスタム・アップロード』の『CSV のアップロード・メニューにより CSV をアップロードしてデータ・ソースを作成する』のアップロード手順を参照してください。ファイルをアップロードするには、「強化」 > 「脆弱性評価」 > 「カスタム・アップロード」に移動します。

手順

1. 「ディスカバリー」 > 「データベース・ディスカバリー」 > 「クラウド DB サービスの保護」とナビゲートして、サービス・アカウント名をクリックします。 クラウド・アカウントを作成すると、「データベースのディスカバー」表が開き、すべての領域のリストがその RDS エンドポイントとともに表示されます。
2. その後このページにアクセスすると、表は閉じます。「データベースのディスカバー」をクリックします。 表が開き、領域が表示されます。
3. ディスカバーするデータベースがある各領域の行を選択します。関係する場合はフィルターを使用します。
4. 「ディスカバー」をクリックします。 Guardium は領域を検索し、過去にディスカバーされていないデータベースをデータベース表に追加します。

親トピック: クラウド・データベース・サービス保護

データベースのカタログおよび管理

Guardium でデータ・ソースを作成するためのデータベースをカタログし、ユーザーとパスワードを変更し、データベース構成を更新します。

このタスクについて

カタログにより、分類、脆弱性評価、監査、およびレポートに使用される、Guardium 内のデータ・ソースが作成されます。カタログされていないデータベースの場合、DB 表の「Guardium データ・ソース」列に赤色のアイコンが表示されます。

手順

1. 監査するデータベースをカタログします。
   a. 「データベース」表で、1 つ以上のデータベースを選択します。
   b. 「データ・ソース」 > 「データ・ソースのカタログ」をクリックします。
   c. DBA から受け取った、大/小文字を区別する DB ユーザーとパスワードを入力します。複数のデータベースを選択した場合は、それらで必ず同じユーザーとパスワードのペアを使用するようにしてください。
   d. オプションで、デフォルトの分類プロセスを選択、変更、またはクリアします。
   e. 「カタログ」をクリックします。
   Guardium データ・ソース名が「データベース」表に表示されます。
2. ユーザーまたはパスワードを更新します。
   a. 「データベース」表で、1 つ以上のデータ・ソースを選択します。
   b. 「データ・ソース」 > 「ユーザーとパスワードの更新」をクリックし、詳細を変更します。 両方のフィールドを指定する必要があります。
   c. 「カタログ」をクリックします。
3. データ・ソース定義を変更します。
   a. データ・ソースを選択し、「データ・ソース」 > 「データ・ソース定義を開く (Open Datasource Definition)」をクリックします。
   b. 必要に応じて変更します。データ・ソース定義の作成でパラメーターの詳細を参照してください。
   c. オプションで、「接続のテスト」をクリックしてデータベースへの接続をテストします。
   d. 「保存」をクリックします。

親トピック: クラウド・データベース・サービス保護

分類および脆弱性評価の管理

データ・ソースを既存の分類プロセスまたは脆弱性評価プロセスに割り当てます。または新規プロセスを作成します。

このタスクについて

「脆弱性評価」メニューは、有効な VA ライセンスをお持ちの場合にのみ使用可能です。

分類プロセスをデータ・ソースに割り当てると、分類データが収集され、オンプレミス・データベースと同じように処理されます。所有者以外でも分類を割り当てることはできますが、オブジェクト監査を有効にして結果を表示するためには所有権を取得する必要があります。

緑色のアイコンはプロセスが実行中であることを示します。黄色のアイコンは、プロセスに対してスケジュールが定義されていないことを意味します。「分類プロセス」列または「VA」列の赤色のアイコンは、分類および VA が割り当てられていないか、エラーであることを示します。VA エラーは、「強化」 > 「脆弱性評価」 > 「アセスメント・ビルダー」 > 「結果の表示」で表示します。分類エラーは、「ディスカバー」 > 「エンドツーエンド・シナリオ」 > 「機密データのディスカバー」 > 「レポートのレビュー」リボン > 「プロセス・ログ」で表示します。分類エラーは、「ディスカバー」 > 「分類」 > 「機密データのディスカバー」 > 「レポートのレビュー」リボン > 「プロセス・ログ」で表示します。

「BDUMP でファイル bdump-file-listing が見つかりません。結果を取得できません: 'RDSADMIN.TRACEFILE_」という分類エラーが表示された場合は、グループ・ビルダーで定義済みスキーマ・グループ「除外する分類スキーマ - Oracle (Excluded Classification schemas - Oracle)」に RDSADMIN を追加します。

手順

1. 既存の分類プロセスに 1 つ以上のデータ・ソースを割り当てます。
   a. 1 つ以上のデータ・ソースを選択します。
   b. 「分類」 > 「分類への追加」をクリックします。
   c. 分類プロセスを選択して、「保存」をクリックします。
   d. オプションで「編集/表示 (Edit/View)」をクリックして、分類プロセスを変更または実行します。
   e. 分類プロセスで検出されたオブジェクトに対してオブジェクト監査を自動的に有効にする場合は、**「編集/表示 (Edit/View)」**をクリックして、分類プロセスを開きます。その後、「検索場所」リボンで「Cloud DB のオブジェクト監査の有効化」チェック・ボックスを選択します。
   f. あるいは、次のようにして分類を実行します。「ディスカバー」 > 「分類」 > 「機密データのディスカバー」に移動し、「ディスカバリーの実行」リボンで「今すぐ実行する」をクリックします。
2. 新規分類プロセスを作成し、1 つ以上のデータ・ソースをそれに割り当てます。
   a. 1 つ以上のデータ・ソースを選択します。
   b. 「分類」 > 「分類の作成」をクリックします。
   c. 機密データのディスカバーの手順に従います。デフォルトでは「Cloud DB のオブジェクト監査の有効化」が選択されています。これは選択されたままにしておきます。
   d. 次のようにして分類を実行します。「検索場所」の定義後に、「今すぐ実行する」をクリックします。またはプロセスの保存後に、「ディスカバリーの実行」リボンで「今すぐ実行する」をクリックします。
3. 既存の脆弱性評価に 1 つ以上のデータ・ソースを割り当てます。
   a. 1 つ以上のデータ・ソースを選択します。
   b. 「脆弱性評価」 > 「脆弱性評価への追加」をクリックします。
   c. 脆弱性評価プロセスを選択して、「保存」をクリックします。
   d. 次のようにしてプロセスを実行します。「強化」 > 「脆弱性評価」 > 「アセスメント・ビルダー」にナビゲートし、プロセスを選択して、「今すぐ 1 回実行」をクリックします。
4. 新規脆弱性評価を作成し、1 つ以上のデータ・ソースをそれに割り当てます。
   a. 1 つ以上のデータ・ソースを選択します。
   b. 「脆弱性評価」 > 「脆弱性評価の作成」をクリックします。
   c. 脆弱性評価の説明を入力します。定義した監査プロセスの一部として結果を受け取る場合は、1 つ以上の E メール・アドレスを (複数の場合はコンマで区切って) 入力します。
   d. 「保存」をクリックします。
   すべてのテスト、選択したデータ・ソース、およびユーザーが定義したレシーバーを使用して VA プロセスがで作成されます。
   e. 次のようにしてプロセスを実行します。「強化」 > 「脆弱性評価」 > 「アセスメント・ビルダー」にナビゲートし、プロセスを選択して、「今すぐ 1 回実行」をクリックします。

親トピック: クラウド・データベース・サービス保護

データベース監査の構成

データベース上での監査を有効にし、オブジェクト監査データを Guardium がプルできるようにします。 分類に自動的に追加されるオブジェクトの制限を変更し、コレクターを変更します。

このタスクについて

「データベース」表には、ディスカバーされたデータベースのさまざまな詳細が表示されます。表内の色付きの標識を使用すると、素早く一目でデータ・ソースの状況を確認できます。赤は構成がないこと、つまり例えばデータベースがカタログされていない、またはデータ・ソースが分類または VA プロセスに割り当てられていないといったことを示します。色分けされた状況標識には吹き出しヒントがあり、色が赤または黄色の場合に詳細が提供されます。 事前定義フィルター・リストを使用して、色分けされた状況標識がある列をフィルタリングできます。その他の値にはフリー・テキスト・フィルターを使用できます。

データ・ソースに対してコレクターが定義されている場合、ユーザーが所有者である場合は「アクティブなコレクター」列に表示されます。それ以外の場合、列はブランクです。

DB 監査の所有者は、CM 環境内の CM ホスト名です。スタンドアロン・システムでは、この値はコレクターのホスト名です。

「DB 監査」列には以下のいずれかの値が示されます。

- 有効。再始動の保留中が後に続く場合、状況はインスタンスの再始動時に有効になることを示します。
- 無効。再始動の保留中が後に続く場合、状況はインスタンスの再始動時に有効になることを示します。
- 構成が要件と一致していません。(AWS パラメーター監査証跡が、Guardium の要件 XML, EXTENDED に従って構成されていません。この値を変更するように DBA に依頼してください。)再始動の保留中が後に続く場合、状況はインスタンスの再始動時に有効になることを示します。
- この DB エンジンではサポートされていません。アクティビティー・モニターは現在、Guardium でサポートされていません。

インスタンスを所有しており、分類プロセスが割り当てられている状態で、DB 監査が有効な場合は、「オブジェクト」列に結果が表示されます。合計は、このインスタンスに割り当てられている分類プロセスで識別されるオブジェクトの数です。「監査済み」は、オブジェクト監査で有効なオブジェクトの数です。「新規」は、分類プロセスで検出されたが、自動的に有効になっていないオブジェクトの数です。これらのオブジェクトは検討が必要です。オブジェクト監査の管理を参照してください。

データ・ソースが分類プロセスに割り当てられており、そのプロセスが DB 監査を有効にして後に実行されており、ユーザーが所有者である場合には、結果が「オブジェクト」列に表示されます。オブジェクトが表示されない場合は、分類プロセスを確認し、再度実行してください。

自動的に追加されるオブジェクトとコレクターの制限の変更

自動的に追加されるオブジェクトとコレクターの制限を、1 つ以上のデータベースで同時に変更できます。ブランクのままになっているフィールドは変更されません。

手順

1. 1 つ以上のデータベースを選択します。
2. 「DB 監査」 > 「DB 監査構成」をクリックします。
3. 「自動的に追加されるオブジェクトを制限」の数を変更します。 これは DB 監査が有効である場合に、オブジェクト監査に対して自動的に有効にできる、分類により検出されるオブジェクトの最大数です。これは、ディスカバー後、データベースごとに変更できます。有効にされたオブジェクトは、「管理対象オブジェクト (Managed Objects)」ウィンドウで自動的に「有効」と表示されます。Guardium にオブジェクトを自動的に追加させるには、分類プロセスでの検出が予期される妥当な上限を設定します。分類に誤りがあった場合でもオブジェクトがオーバーフローしないようにするために、設定値は高すぎないようにします。(オーバーフローはデータベースのパフォーマンスに影響を与える可能性があります)。ゼロ (0) は、オブジェクト監査に対して自動的に有効になるオブジェクトがないことを意味します。 監査済みのオブジェクトの数に、新しく分類されたオブジェクトの数を加えると、この制限を超える場合、新しいオブジェクトは、オブジェクト監査が有効になりません。例えば、15 に設定され、分類が最初の実行時に 5 個のオブジェクトを識別する場合、その 5 個のオブジェクトに監査証跡が割り当てられます。設定が 15 であり、既に 5 つのオブジェクトがオブジェクト監査に有効になっており、分類の次の実行で 16 オブジェクトが識別されると、新しいオブジェクトはどれもオブジェクト監査が有効になりません。設定が 15 であり、既に 5 つのオブジェクトがオブジェクト監査に有効になっており、分類の次の実行で 5 オブジェクトが識別されると、新しい 5 オブジェクトはオブジェクト監査で有効になります。
4. コレクターが「中央マネージャー」環境に表示されます、コレクターはこの環境に必須です。CM 環境内のすべてのコレクターのドロップダウン・リストからコレクターを選択します。これは監査データ (アクティビティー) を DB からプルするコレクターです。
5. 「適用」をクリックします。

1 つのデータベースの監査の有効化

DB 監査は一度に 1 つのデータベース上で有効にできます。

このタスクについて

「自動的に追加されるオブジェクトを制限」パラメーター、または任意の許可レベルのコレクターを構成できます。その他の変更には DB 許可が必要です。アクセス・キーにはこれらの許可が含まれる場合もあれば、含まれない場合もあります。以下の説明は、すべてのレベルの許可を対象にしています。

DB 監査を有効にすると、Guardium システムは、この DB 上の DB 監査の固有の所有者になります。他の Guardium システムは DB 監査およびオブジェクト監査を変更しません。「DB 監査の所有の開始 (Start owning DB Audit)」をクリックすることで、別のシステムが強制的に所有権を取得することができます。

DB 監査で監査用オブジェクトを表示して管理できるようにしたら、少なくとも 1 回分類を実行します。 オブジェクトが見つからない場合は、ポリシーを確認します。

注意:
データベースの管理を開始すると、Amazon RDS タグ IBM Guardium IP が Guardium ホスト名の値で作成されます。このタグを変更したり、削除したりしないでください。

手順

1. データベースの行を選択します。
2. 「DB 監査」 > 「DB 監査構成」をクリックします。
3. オプションで、オブジェクト監査に自動的に追加されたオブジェクトの値を変更します。
4. CM 環境で、コレクターが定義されていない場合は、ドロップダウン・リストからコレクターを 1 つ選択して、「適用」をクリックします。 ダイアログがリフレッシュされ、ボタンが有効になります。
5. 「DB 監査を有効にする」が使用可能になっていれば、それをクリックします。ダイアログおよび表がリフレッシュされ、ユーザーが DB 監査の現在の所有者であることが表示されます。ダイアログ・ボックスがリフレッシュされます。「再始動」をクリックしてデータベースを今すぐ再始動するか (確認メッセージが表示されます)、または、例えば保守期間を待機するには、「次の手動再始動の待機 (Wait for next manual restart)」をクリックします。 「次の手動再始動の待機 (Wait for next manual restart)」を選択した場合は、クラウド・コンソールに後で直接アクセスする必要があります。 「再始動」をクリックした場合に十分なアクセス権限がない場合には、エラーが表示されます。DBA に、**監査証跡**を XML, EXTENDED として構成し、インスタンスを再始動するように依頼してください。
6. 「DB 監査を有効にする」が使用可能になっていなければ、「DB 監査の所有」をクリックします。ダイアログ・ボックスがリフレッシュされます。 「次の手動再始動の待機 (Wait for next manual restart)」をクリックし、DBA に、**監査証跡**を XML, EXTENDED として構成し、インスタンスを再始動するように依頼してください。
7. DB 監査状況を変更した場合は、「状況の取得」をクリックし、状況が変更されたことを示すメッセージが表示されるまで待ってから「リフレッシュ」をクリックします。「DB 監査の所有者」列に CM のホスト名またはスタンドアロン Guardium のコレクターのホスト名が表示され、「DB 監査」のアイコンが緑になります。

1 つのデータベースの監査の無効化

DB 監査は一度に 1 つのデータベース上で無効にできます。DB 監査を無効にすると、DB 監査の所有権も放棄することになります。

このタスクについて

DB 監査の所有を停止するかまたは DB 監査を無効にすると、オブジェクト監査全体も無効になり、監査可能なオブジェクトのリスト (分類結果からの成果物) は削除されます。

手順

1. データベースの行を選択します。
2. 「DB 監査」 > 「DB 監査構成」をクリックします。
3. 「DB 監査を無効にする」をクリックし、次に、例えば保守期間を待機するには「次の手動再始動の待機 (Wait for next manual restart)」をクリックし、データベースをすぐに再始動するには、「再始動」をクリックします。 「次の手動再始動の待機 (Wait for next manual restart)」を選択した場合は、クラウド・コンソールに

後で直接アクセスする必要があります。 構成を変更する権限がない場合は、「DB 監査の所有の停止」をクリックし、DBA にこのインスタンスの DB 監査を無効にするように要求します。
4. 「状況の取得」をクリックして、クラウドからの最新の状況で表示を最新表示します。

タスクの結果

変更がある場合は、「一部のデータベースで DB 監査状況が変更されました。「リフレッシュ」をクリックして表を更新してください」というメッセージが表示されます。「リフレッシュ」をクリックします。状況は「無効」または「無効、再始動の保留中」に変わり、「DB 監査」のアイコンは赤になり、「DB 監査の所有者」列はブランクになります。

DB 監査所有権の開始および停止

このタスクについて

一度に 1 つのデータベースの DB 所有権状況を変更できます。

DB 監査を所有すると、DB 監査およびオブジェクト監査の定義への排他的権限と、オブジェクト監査データへのアクセス権限が付与されます (オブジェクト監査の管理を参照)。 他の Guardium システムも同じクラウド・アカウントにアクセスできますが、表示できるのは DB 詳細のみです。

全アクセス権限がある場合、DB 監査を有効にすると、DB の所有権も取得します。アクセス・キーが全アクセス権限を提供していない場合は、DB 監査を有効にせずに所有権を取得します。 DB 監査が (DBA により) 有効になると、監査データにアクセスできるようになります。逆に、DB 監査を無効にすると、所有権を放棄することになります。アクセス・キーが全アクセス権限を提供していない場合、DB 監査の所有が停止され、DBA に DB 監査を無効にするように要求することになります。

ある Guardium システムから別のシステムに所有権を移動することができます。

2 つのライブ・システム間で所有権を移動する場合、まず現行所有者での DB 監査の所有を停止してから、2 つ目の Guardium システムで所有権を取得します。一方の Guardium システムが所有権を放棄すると、すべての監査が停止します。新しい Guardium システムで監査プロセスを定義する (分類への DB の割り当て、プロセスの実行、オブジェクト監査へのオブジェクトの追加を行う) 必要があります。
注意:
一方で DB 監査の所有を停止してから、もう一方でそれの所有を開始してください。 そうしないと、データは、新しいコレクターだけでなく、以前のコレクターにも送信されます。異なるポリシー (異なる CM) を持つ 2 つのコレクターが同じアクティビティーを受け取ると、それぞれのコレクターで異なる (あるいは不完全な) 結果が生成されます。

ある Guardium システムがダウンし、リカバリーが期待されないときに、そのシステムから所有権を移動する場合、監査定義を維持しながら、所有権のみを変更して、別の Guardium システムから DBA 監査の所有を開始できます。このシナリオでは、DB コンソールで、元の Guardium による DB 監査の所有を停止します。

手順

1. 「データベース」表で、データベースの行を選択します。
2. DB 監査を停止するには、「DB 監査」 > 「DB 監査構成」 > 「DB 監査の所有の停止」をクリックします。
3. DB 監査の所有を開始するには、「DB 監査」 > 「DB 監査構成」 > 「DB 監査の所有の開始 (Start owning DB audit)」をクリックします。

オブジェクト監査の管理

管理しているデータベースの分類プロセスによって識別された潜在的な機密オブジェクトを表示し、これらのオブジェクトに実行されるすべてのアクティビティーをモニターするために、選択したオブジェクトに対するオブジェクト監査を有効にします。

このタスクについて

前提条件: DB 監査が有効かつユーザーにより所有されており、分類がこのデータ・ソースに対して少なくとも 1 回実行される必要があります。

新規オブジェクトとは、分類プロセスによって検出されたオブジェクトのうち、監査が有効になっていないオブジェクトのことです。すべての新規オブジェクトをフィルタリングして、それらのオブジェクト監査を有効にするか、または「新規」フラグをクリアすることができます。新規オブジェクトがないときは、新規オブジェクトの評価は最新です。Guardium は分類プロセスが実行されるごとに新規データを受け取る可能性があることを覚えておいてください。オブジェクト監査に自動的に追加されなかった新規オブジェクトが検出された場合、「新規オブジェクトが検出されました (New objects were found)」という通知が出されます。

「分類による検出」列には、そのオブジェクトを特定したすべての分類プロセスがリストされます。

「オブジェクト監査の状況」列の「混合」状況は、いくつかのデータ・ソースではオブジェクト監査が有効で、他のデータ・ソースでは無効であることを意味します。

オブジェクト監査の有効化および無効化は負荷が高いプロセスであり、数分かかる場合があります。クラウドが監査変更を処理している間は、待機中アイコンが表示されます。

「データベース」表からレビューするデータ・ソースの行を選択することで、1 つ以上のデータ・ソースで検出されたオブジェクトをレビューできます。 「オブジェクト監査」ウィンドウには、選択したデータベースに対するすべての分類プロセスによって検出された、すべてのオブジェクトが表示されます。

- 1 つのデータベースでのオブジェクト監査の管理
- 複数のデータベースでのオブジェクト監査の管理

親トピック: クラウド・データベース・サービス保護

1 つのデータベースでのオブジェクト監査の管理

このタスクについて

「データ・ソース <name> 内のオブジェクト (Objects in Datasource <name>)」ウィンドウには、このデータ・ソースに対して実行された分類プロセスにより検出されたすべてのオブジェクトがリストされます。オブジェクトは、複数の分類プロセスによって見つかる可能性があります。

オブジェクトが分類プロセスによって特定されているが、オブジェクト監査が自動的に有効にならなかったときは、オブジェクト表の上に「検出された新規オブジェクト」が表示されます。「新規のみ」をクリックしてフィルターに掛けることで、処理が必要な、すべての検出された新規オブジェクトを表示します。 新規オブジェクトは、分類を実行するたびに見つかる可能性があります。新規オブジェクトがないときは、新規オブジェクトの評価に遅延はありません。

定期的にデータ・ソースをレビューして、新しいオブジェクトを確認し、オプションでオブジェクト監査のオブジェクトを追加または削除します。例えば、自動的に追加されたオブジェクトに監査を必要としないと決定したオブジェクトが含まれているか、データベースにパフォーマンスの問題がある場合は、オブジェクトを削除できます。または、監査されていない疑わしいオブジェクトを特定し、それをオブジェクト監査に追加することができます。

監査の必要があることが分かっているオブジェクトには、分類フィルターを使用します。フィルタリング済みのビューですべてのオブジェクトを選択し、オブジェクト監査を有効にします。

手順

1. DB 監査を有効にする**前に**分類プロセスを割り当てた場合には、ここで分類を 1 回実行して、Guardium がオブジェクトを識別するのを数分待機します (またはスケジュールされた次の実行を待機します)。
2. データ・ソースを 1 つ選択します。新規オブジェクトがあるデータ・ソースを識別するために、「検出された新規オブジェクト」フィルターを使用できます。
3. 「DB 監査」 > 「オブジェクト監査の管理」を選択します。 「オブジェクト監査の管理」ウィンドウが開き、このデータ・ソースが割り当てられている、分類プロセスによって検出されたすべてのオブジェクトのリストが表示されます。
4. 新規として分類されているすべてのオブジェクトを識別するために、「新規のみ」フィルターを使用できます。
5. 表から 1 つ以上のオブジェクト (行) を選択します。
6. 監査証跡を有効にするには、「アクション」 > 「監査を有効にする」を選択します。 システムは、操作の成功または失敗で応答します。
7. 「新規」フラグをクリアするには、「アクション」 > 「新しいフラグのクリア」をクリックします。
8. 監査証跡を無効にするには、「アクション」 > 「監査を無効にする」を選択します。 システムは、操作の成功または失敗で応答します。

親トピック: オブジェクト監査の管理

複数のデータベースでのオブジェクト監査の管理

このタスクについて

このビューには、選択したデータ・ソースに対して実行された分類プロセスによって見つかったすべてのオブジェクトがリストされます。オブジェクトは、複数の分類プロセスによって見つかる可能性があります。オブジェクトをオブジェクトごと (デフォルト)、または種別ごとにグループ化して表示します。「分類による検出」列には、オブジェクトを特定したすべての分類プロセスがリストされます。

オブジェクトが分類プロセスによって特定されたが、オブジェクト監査が自動的に有効にならなかったときは、オブジェクト表の上に「検出された新規オブジェクト」が表示されます。「新規のみ」をクリックしてフィルターに掛けることで、処理が必要な、すべての検出された新規オブジェクトを表示します。「新規」オブジェクトを確認して、オブジェクト監査を有効にするか、「新規」フラグをクリアします。

新規オブジェクトは、分類を実行するたびに見つかる可能性があります。新規オブジェクトがないときは、新規オブジェクトの評価に遅延はありません。

定期的にデータ・ソースをレビューして、新しいオブジェクトを確認し、オプションでオブジェクト監査のオブジェクトを追加または削除します。例えば、自動的に追加されたオブジェクトに監査を必要としないと決定したオブジェクトが含まれているか、データベースにパフォーマンスの問題がある場合は、オブジェクトを削除できます。または、監査されていない疑わしいオブジェクトを特定し、それをオブジェクト監査に追加することができます。

オブジェクトごとにグループ化: 新しく見つかったオブジェクトをすべて表示するには、テキスト・フィルターに「新規」と入力します。

選択したすべてのデータ・ソースであるオブジェクトのオブジェクト監査を有効または無効にするには、行を選択し、「アクション」>「有効化/無効化」をクリックします

データ・ソースごとにアクションを実行するには、「オブジェクトが存在しているデータ・ソースの数」をクリックして、選択したオブジェクトを分類プロセスが特定したすべてのデータ・ソースを表示します

分類ごとのグループ化: これは監査を必要としていて、それ以上の評価は必要ないオブジェクト (例えば GDPR など) を持つ、ほとんど同一のデータ・ソース、または分類ポリシーがあるときは特に有用です。

手順

1. DB 監査を有効にする**前に**分類プロセスを割り当てた場合には、ここで分類を 1 回実行して (またはスケジュールされた次の実行を待機して) 、Guardium がオブジェクトを識別するのを数分待機します。
2. オブジェクトごとのグループ化の場合には、次のようにします。
   a. データベース表の「オブジェクト」列に新規オブジェクトがある複数のデータ・ソースを選択します。 それらのデータ・ソースを識別するには、「検出された新規オブジェクト」フィルターを使用します。
   b. 「DB 監査」 > 「オブジェクト監査の管理」をクリックします。「オブジェクト監査の管理」ウィンドウが開きます。
   c. このオブジェクトを常にすべてのデータ・ソース内で監査する必要がある場合には、行を選択し、「アクション」 > 「監査を有効にする」をクリックします。 システムは、操作の成功または失敗で応答します。
   d. 個々のデータベースでオブジェクト監査を有効にする場合は、オブジェクトの行にある「オブジェクトが存在しているデータ・ソースの数」列の番号をクリックし、「<object> を含むデータ・ソース」ウィンドウを開きます。このウィンドウは、分類プロセスが選択したオブジェクトを識別したすべてのデータ・ソースを示します。1 つ以上のデータ・ソース行を選択し、「アクション」 > 「監査を有効にする」をクリックします。
3. 識別されたオブジェクトが常に監査を必要とし、それ以上の評価は必要としない分類プロセスの場合は、「分類」ラジオ・ボタン (表の上にある) をクリックし、分類プロセスの 1 つ以上の行を選択して、「アクション」 > 「監査を有効にする」をクリックします。

親トピック: オブジェクト監査の管理

データベース・オートディスカバリー

オートディスカバリー・アプリケーションは、サーバーをスキャンおよびプローブしてオープン・ポートを調べ、ネットワークに対して不明な接続や望ましくない接続が行われるのを防ぎます。オートディスカバリー・プロセスはオンデマンドで実行することも、定期的に実行されるようスケジュールすることもできます。

データベース・オートディスカバリーの概要

ネットワーク上にデータベースが未検出の状態で存在し、ネットワークが潜在的なリスクにさらされる可能性のあるシナリオには、さまざまなものがあります。例えば、古いデータベースがモニターされずに忘れられている場合や、新規データベースがアプリケーション・パッケージの一部として追加される場合などが考えられます。また、不正なデータベース管理者が、データベースの新規インスタンスを作成し、モニターされているデータベース以外に対して悪質なアクティビティーを実行する可能性もあります。

オートディスカバリーはスキャン・ジョブとプローブ・ジョブを使用して、環境内に未検出のデータベースが存在しないようにします。

- *スキャン・ジョブ*は、指定した各ホスト (または指定したサブネット内のホスト) をスキャンし、そのホストに指定されているオープン・ポートのリストを作成します。
- *プローブ・ジョブ*は、スキャンの結果を使用して、オープン・ポート上で実行中のデータベース・サービスがあるかどうかを判別します。プローブ・ジョブを実行するには、最初にスキャンを実行する必要があります。「ディスカバーされたデータベース」事前定義レポートで、このジョブの結果を確認できます。

始める前に、オートディスカバリー・アプリケーション用のパッチをダウンロードしてインストールしてください。このパッチは、IBM Fix Central で入手できます。

オートディスカバリー・アプリケーションを使用するには、次のステップを実行します。

1. 特定の IP アドレスまたはサブネットでオープン・ポートを検索するオートディスカバリー・プロセスを作成します。
2. オートディスカバリー・プロセスを、オンデマンドで、またはスケジュールに従って実行します。
3. オートディスカバリー・レポートでプロセスの結果を確認するか、カスタム・レポートを作成します。

オートディスカバリーには、監査プロセスに依存しない独自のプロセスがありますが、それらは監査プロセスと全く同じように動作します。

スキャンを行う場合はホスト名ではなく IP のみを入力できますが、Guardium はレポートの一部としてホスト名を検出します。Guardium は、Guardium 製品内でホスト名を切り捨てることはありません。ただし、列の幅がより広くなるようにレポートを構成する必要があるかもしれません。

Guardium オートディスカバリーでは、プローブ中に表示されるデータベースを推測することはありません。Guardium オートディスカバリーがデータベースを検出したことを示した場合、そのデータベースが何であるかは 100% 明確です。

注: ディスカバリーでは、実行中のデータベースのみが検出されます。 インストール時にディスカバリーを使用する予定の場合は、データベースを始動する必要があります。AIX KTAP インターセプトの機能上の理由から、初めて S-TAP を実行した後に、データベースを再始動しなければなりません。データベースを再始動しないと、一部のインターセプトが動作しません。

オートディスカバリー・プロセスの作成

オートディスカバリー・プロセスがスキャンするホストとポートを指定します。

1. 「ディスカバー」 > 「データベース・ディスカバリー」 > 「オートディスカバリーの構成」をクリックし、オートディスカバリーを構成します。
2. 「新規」をクリックして新規プロセスを作成し、「オートディスカバリー・プロセス・ビルダー」を開きます。
3. Guardium® システム上で固有の「プロセス名」を入力します。
4. スキャン・ジョブの完了直後にプローブ・ジョブを実行するには、「スキャン後にプローブを実行」チェック・ボックスにチェック・マークを付けます。
5. スキャンするホストまたはサブネットごとに、ホストとポートを入力して「スキャンの追加」をクリックします。スキャンを追加するたびに、スキャンがタスク・リストに追加されます。
   注:
   - ワイルドカード文字が使用可能です。例えば、192.168.2 で始まるアドレスをすべて選択するには、「192.168.2.*」と指定します。
   - 一定範囲のポートを指定するには、その範囲内の最初のポート番号と最後のポート番号の間にダッシュを入れます。例: 4100-4102。
   - スキャンを追加した後、ホストまたはポートを上書き入力で変更します。 「適用」をクリックして、変更を保存します。
   - デュアル・スタック構成がある場合は、IPV4 アドレスと IPV6 アドレスの両方に対してスキャンを設定する必要があります。
   - スキャンを削除するには、そのスキャンの「このタスクを削除」アイコンをクリックします。タスクに、それに従属するスキャン結果がある場合は、そのスキャンは削除できません。
6. スキャンの追加が完了したら「適用」をクリックし、ジョブを実行するか、ジョブを後で実行するようスケジュールします。

スケジュールの定義のヘルプ情報が必要な場合は、『スケジューリング』を参照してください。

オートディスカバリー・プロセスの実行またはスケジュール

スキャン・ジョブとプローブ・ジョブをオートディスカバリー・プロセスの一部として実行またはスケジュールします。

1. 「ディスカバー」 > 「データベース・ディスカバリー」 > 「オートディスカバリーの構成」をクリックします。
2. 「オートディスカバリー・プロセス・セレクター」リストから実行するプロセスを選択し、以下のいずれかを実行します。
3.
   - ジョブを即時に実行するには、「今すぐ 1 回実行」をクリックします。
   - 今後ジョブをスケジュールするには、「スケジュールの変更」をクリックします (スケジュール定義のヘルプ情報が必要な場合は、『スケジューリング』を参照してください)。
     注: プローブ・ジョブは、スキャン・ジョブの結果がないと実行できません。これら 2 つのジョブを個々に実行するようスケジュールすることも、スキャン・ジョブの後にプローブ・ジョブを実行するよう構成することもできます。後者の場合は、プロセスに変更を加え、「スキャン後にプローブを実行」チェック・ボックスにチェック・マークを付けます。
4. ジョブを開始またはスケジュールした後、「進行状況/サマリー」をクリックすると、このプロセスの状況を表示できます。

オートディスカバリー・レポート

オートディスカバリー・レポートを開くには、「ディスカバー」 > 「レポート」をクリックし、使用可能なレポートから選択します。

「オートディスカバリー・クエリー・ビルダー」で、カスタム・レポートを作成することができます。「オートディスカバリー・クエリー・ビルダー」を開くには、「ディスカバー」 > 「データベース・ディスカバリー」 > 「オートディスカバリー・クエリー・ビルダー」をクリックします。

「ディスカバーされたデータベース」レポート

「ディスカバーされたデータベース」レポートを開くには、「ディスカバー」 > 「レポート」 > 「ディスカバーされたデータベース」をクリックします。

このレポートのメイン・エンティティーは「ディスカバーされたポート」です。ディスカバーされた各ポートは、レポート内でそれぞれの行に表示されます。「プローブ時間」、「サーバーの IP アドレス」、「サーバー・ホスト名」、「データベース・タイプ」、「ポート」、「ポート・タイプ」(通常は TCP)、およびオカレンス数の各列がリストされます。

このレポートには、特殊なランタイム・パラメーターはありませんが、 データベース・タイプが「不明」であるディスカバーされたポートは除外されます。

オートディスカバリー・プロセス定義を変更すると、そのプロセスの統計はリセットされます。

「オートディスカバリーのトラッキング」ドメイン

「オートディスカバリーのトラッキング」ドメインには、オートディスカバリー・プロセスによってレポートされたすべてのデータが含まれます。エンティティー名をクリックして、その属性を表示します。

「オートディスカバリーのトラッキング」ドメインのエンティティー

- 「オートディスカバリー・スキャン」には、各スキャン操作のタイム・スタンプが示されます。
- 「ディスカバーされたホスト」には、ディスカバーされた各ホストの IP アドレスとホスト名が示されます。
- 「ディスカバーされたポート」には、オープン状態がディスカバーされたポートごとに、タイム・スタンプ、ポート、およびデータベース・タイプが示されます。

親トピック: ディスカバー

分類

分類ポリシーと分類プロセスは、Guardium® が機密データ (クレジット・カード番号、社会保障番号、個人の金融データなど) をディスカバーして処理する方法を定義します。

組織の規模が大きくなり、クレジット・カード番号および個人の金融データなどの機密情報が複数のロケーションに存在するようになると (そのデータの現在の管理責任者が分からないという場合がよくあります)、ディスカバリー・プロセスと分類プロセスが重要になります。こうした状況は、合併買収の際や、元の所有者がいなくなった後もレガシー・システムを引き続き使用している場合に、多く発生します。 機密データをディスカバーするためのワークフローを作成すると、現行環境内の機密データを特定し、適切なアクション (アクセス・ポリシーの適用など) を実行できるようになります。

分類プロセス は、1 つ以上のデータ・ソースと関連付けられた分類ポリシーから成ります。分類プロセスは、1 回だけ実行するように処理依頼することも、コンプライアンス・ワークフロー自動化プロセスで定期的に実行するようスケジュールすることもできます (プロセスで使用されるすべてのデータ・ソースに対してログイン資格情報が保管されている場合)。

分類ポリシー は、指定されたデータ・ソース内の機密データを見つけてタグ付けするよう設計された、分類ルールと分類ルール・アクションから成ります。

分類ルール は、正規表現、Luhn アルゴリズム、およびその他の基準を使用して、分類ポリシーの適用時に内容を突き合わせるためのルールを定義します。

分類ルール・アクション は、分類ポリシー内の各ルールに対して実行する一連のアクションを指定します。例えば、アクションによって E メール・アラートを生成したり、Guardium グループにオブジェクトを追加したりすることができます。ルールが満たされるたびにそのイベントがログに記録されるため、それについてレポートすることができます (実行するアクションとして「無視」が指定された場合は例外で、その場合にはそのルールのログは記録されません)。

- 分類プロセスのパフォーマンス
  分類プロセスの処理には、サンプリング・ルーチンとタイムアウト・パラメーターが使用されます。これにより、データベース・サーバーに対するパフォーマンス上の影響が最小限に抑えられます。
- 分類ルールの処理
  分類ルールは、柔軟なマッチングおよびグループ化基準に従って処理されます。

親トピック: ディスカバー

分類プロセスのパフォーマンス

分類プロセスの処理には、サンプリング・ルーチンとタイムアウト・パラメーターが使用されます。これにより、データベース・サーバーに対するパフォーマンス上の影響が最小限に抑えられます。

分類機能の実行時には、レコードのサンプリング方法を指定するオプションがあります。デフォルトの動作では、該当するデータベース・プラットフォームに適したステートメントを使用して、行がランダムにサンプリングされます。例えば SQL データベースの場合、分類機能は rand() ステートメントを使用してサンプリングを行います。代替動作は順次サンプリングです。この場合は、指定されたサンプル・サイズになるまで順番に行が読み取られます。ランダム・サンプリングはデフォルトの動作であり、より典型的な結果が得られるため、一般的にはこのサンプリングをお勧めします。ただし、ランダム・サンプリングは順次サンプリングと比べて、パフォーマンスが若干低下する可能性があります。

ランダム・サンプリングと順次サンプリングのどちらも、デフォルトのサンプル・サイズは、2000 行か使用可能な行の総数のいずれか少ない方になります。これより大きい、または小さいサンプル・サイズを指定することもできます。ランダム・サンプリング・ボックスにチェック・マークを付けると、その表/ビューから無作為に 2000 行が選択され、スキャンされます。表に含まれる行が 2000 行未満の場合、すべての行がスキャンされます。ランダム・サンプリング・ボックスのチェック・マークを外すと、その表/ビューから最初の 2000 行が選択され、スキャンされます。デフォルトの照会タイムアウト値は 3 分 (180 秒) です。プロセスが実行中であっても 30 分間停止した場合は、プロセス全体が一時停止されます。

分類プロセスがデータベース・サーバーに与える影響をさらに最小化するため、長時間実行されている照会はキャンセルされ、ログに記録されて、表の残りはスキップされます。このポイントまでに取得された行はすべて、表のルールを評価する際に使用されます。同様に、分類プロセスを長時間実行しても完了しない場合は、プロセス全

体が一時停止され、プロセス統計とともにログに記録されて、次の分類プロセスが開始されます。これが発生するのはまれで、通常は、既にパフォーマンス上の問題があるサーバーでのみ発生します。

分類機能は定期的に分類をアイドル状態にして、データベース・サーバーに対する要求が過剰になることを防ぎます。 データをサンプリングしている分類ルールが多数あってもデータベース・サーバー上の負荷は一定のはずですが、プロセスの実行時間が増える可能性があります。

分類機能は、除外されたグループを、スキーマ、表、表の列に対して使用することにより、誤検出を処理します。将来的な分類スキャンのために誤検出の結果を無視するように Guardium を設定するのは、これまでは複雑な作業でした。現在では、分類結果をレビューする際に、誤検出の結果を除外グループに簡単に追加し、そのグループを分類ポリシーに追加すれば、それ以降のスキャンでそれらの誤検出の結果が無視されるようになりました。

マルチスレッド分類

Guardium では、CPU のパフォーマンスと使用状況を最適化するために、複数の分類スレッドを並行して実行できます。並行して実行できるスレッドの数は、マシン内の CPU コアの数に 2 を乗算することで得られます。

一例として、4 個の CPU コアがある場合は、定義して同時に実行できる分類プロセスの最大数は 8 です。CPU コアの数に関係なく、上限は 100 です。

並行性の制限を取得または定義するには、GuardAPI 分類関数を使用してください。

親トピック: 分類

分類ルールの処理

分類ルールは、柔軟なマッチングおよびグループ化基準に従って処理されます。

「限定起動」マーカー

「限定起動」マーカーを使用すると、まったく同一の名前によって分類ルール・タイプをグループ化することができます。さらに、1 つのマーカーを使用して返されるルールはすべて、同じ名前の表に基づいてデータを返す必要があります。 同じマーカーを使用して 2 つ以上のルールが定義されている場合、それらのルールは一緒に起動されます (すなわち、両方のルールが同じ表で起動された場合に、それらは両方ともログに記録され、それらのアクションが呼び出されます)。 反対に、ある表でいずれか一方のルールのみが起動された場合、ルールはどちらもログに記録されず、それらのアクションも呼び出されません。 複数のルールを一緒に起動できるようにすることは、同じ表内に複数の機密データが同時に現れる場合が懸念されるときに重要になります。 例えば、1 つの表に社会保障番号とマサチューセッツ州の運転免許証の両方が含まれる場合に、それを知ることができます。

「限定起動」マーカーは定数値であり、任意の値を指定でき、グループ化するルール全体でまったく同じ値でなければなりません。つまり、1 つのルールのマーカー名が ABC であれば、そのルールと一緒にグループ化する他のルールのマーカー名も ABC でなければなりません。 それ以外のマーカー値およびルールは、グループに含まれなくなります。

同じ名前の表内でデータを探索することが基本条件である場合、任意の同じ値を持つルールを少なくとも 2 つ使用する必要があります。

突き合わせを続行

限定起動マーカーは、「突き合わせを続行」にも基づいています。一例として、以下のルールが定義されており、ルール 3 が「突き合わせを続行」と一致しない場合、他の 3 つのマーカー・ルールがすべて正となったかどうかに関係なく、結果は返されません。 これは、ルール 4 の実行まで至らなかったためです。必ずすべての限定起動マーカーが実行され、結果が正になる必要があるので、このグループは起動されません。

ルール 1. 起動マーカー・ルール ABC (突き合わせを続行)

ルール 2. 起動マーカー・ルール ABC (突き合わせを続行)

ルール 3. 起動マーカー以外のルール・タイプ (突き合わせを続行)

ルール 4. 起動マーカー・ルール ABC (突き合わせを続行)

一致しない列のみ

結果データの細分度を下げる場合は、このオプションを使用します。組織によっては、組織内のデータを調査し、機密データが含まれている表や列だけを確認したいという場合があります。こうした場合、必ずしも、その列に含まれているすべてのタイプの機密データを検索する必要はありません。「突き合わせを続行」の新しいオプションである「一致しない列のみ」を選択すると、対象の列で一致が見つかった場合、分類機能はその列を無視して処理を続行します。

表 1. 分類プロセスで使用できるオプションの概要

突き合わせを続行	一致しない列のみ	結果の細分度
いいえ	N/A	表が返されます。表内で最初のヒットが見つかると、ルールの処理が停止します。
はい	はい	表と列が返されます。特定の列内の最初のヒットが記録され、それ以降のルールでは、その列が無視されます。
はい	いいえ	詳細な情報が返されます。すべてのルールで、すべての列についてヒットが記録されます。

Luhn アルゴリズムによる分類

ルール名が guardium://CREDIT_CARD で始まり、「検索式」ボックスに 有効なクレジット・カード番号パターンが指定されている場合、分類ポリシーは、標準の パターン・マッチングに加えて Luhn アルゴリズム (クレジット・カード番号などの ID 番号 検証のために幅広く使用されているアルゴリズム) を使用します。 Luhn アルゴリズムは追加の検査であり、パターン検査の代わりにはなりません。 有効なクレジット・カード番号は、16 桁の数字文字列、または 4 桁の数字 4 セット (各セットの間がブランクで区切られる) です。 このパターン・マッチングに Luhn アルゴリズムを組み込むには、「検索式」ボックスに guardium://CREDIT_CARD ルール名と有効な [0-9]{16} 数値の両方が指定されている必要があります。

親トピック: 分類

機密データのディスカバー

機密データをディスカバーして分類するためのエンドツーエンドのシナリオを作成します。

このタスクについて

組織の規模が大きくなり、クレジット・カード番号や個人の金融データなどの機密情報が複数のロケーションに伝搬するにつれて、ディスカバリー・プロセスと分類プロセスが重要になります。 こうした状況は、合併および買収の際や、元の所有者がいなくなった後もレガシー・システムを引き続き使用している場合に、多く発生します。 その結果、現在、データを所有している人の知らない場所に機密データが存在する可能性があります。 機密データが存在することを知らなければ、それを保護することができないため、これはよくあるが非常に脆弱なシナリオです。

機密データのディスカバリー・シナリオは、以下のような企業セキュリティーの 3 つの重要な側面にわたります。

- ディスカバリー: 現行環境のどこかに存在する機密データの場所を探索する
- 保護: 機密データへのアクセス時にモニターとアラート処理を行う
- コンプライアンス: 機密データのディスカバリー・プロセスの結果をレビューするための監査証跡を作成する

「機密データのディスカバー」エンドツーエンド・シナリオ・ビルダーを使用して、複数の Guardium ツールを使いやすい、単一のインターフェースに統合することにより、ディスカバリー、保護、およびコンプライアンスのプロセスを簡素化します。

表 1. 機密データ・ディスカバー・ツールの一覧

値	シナリオ・タスク	記述	結果
ディスカバー	名前および記述	シナリオと、そのシナリオに関連するプロセスとポリシーについて、名前と記述を指定します。	分類プロセスと分類ポリシーが作成されます。 オプションで、新しいデータ・ソース定義が作成されます。
	ディスカバー対象	データのディスカバーと分類を行うためのルールとルール・アクションを作成します。	
	検索場所	スキャンするデータ・ソースを特定します。	
	ディスカバリーの実行	シナリオを実行して結果を確認し、特別なグループ化アクションとアラート・アクションを定義します。	
保護	レポートのレビュー		アクセス・ポリシーが作成されます。
順守	監査	受信者、配布順序、レビュー・オプションを定義します。	監査プロセスが作成されます。
	スケジュール	指定した間隔で実行されるスケジュールを作成します。	

この一連のタスクでは、新規ディスカバリー・シナリオを作成するプロセスについて説明します。 また、機密データをディスカバーするためのルールとルール・アクションから構成される*分類ポリシー*の作成方法、機密データをスキャンするためのデータ・ソースを特定して*分類プロセス*を作成する方法、グループ化やアラート処理などを行う特別なポリシーの定義方法、およびスケジュールされた間隔でさまざまな利害関係者に監査の結果を配布する*監査プロセス*の作成方法についても説明します。

1. ディスカバリー・シナリオ
   新規ディスカバリー・シナリオを作成するか、既存のディスカバリー・シナリオを選択してコピーまたは編集します。
2. 名前および記述
   ディスカバリー・シナリオの名前と記述を入力します。
3. ディスカバー対象
   機密データをディスカバーして分類するためのルールとルール・アクションから成るポリシーを作成します。
4. 検索場所
   機密データをスキャンするデータ・ソースを特定します。
5. ディスカバリーの実行およびレポートのレビュー
   オプションでディスカバリー・シナリオを実行し、結果をレビューします。
6. 監査
   オプションで、ディスカバリー・レポートおよび分類レポート用の受信者、配布順序、およびレビュー・オプションを定義することによって監査プロセスを作成します。
7. スケジューリング
   オプションで、定義された間隔で実行するように監査プロセスをスケジューリングすることによって、監査プロセスをアクティブにします。

次のタスク

次のセクションに進み、ディスカバリーおよび分類のシナリオの「名前および記述」を指定してください。
親トピック: ディスカバー

ディスカバリー・シナリオ

新規ディスカバリー・シナリオを作成するか、既存のディスカバリー・シナリオを選択してコピーまたは編集します。

手順

1. 「ディスカバー」 > 「分類」 > 「機密データのディスカバー」をナビゲートします。
2. ディスカバリー・シナリオを作成、コピー、または編集します。
   - アイコンをクリックして新規シナリオを作成します。

-  アイコンをクリックして、既存のシナリオまたはテンプレートをコピーします。ディスカバリー・シナリオをコピーする場合、シナリオに関連付けられている分類ポリシーを再使用するには「再使用」ボタンをクリックし、分類ポリシーのコピーを作成して使用するには「コピーの作成」ボタンをクリックします。
- 「ディスカバリー・シナリオ」リストから既存のシナリオ名をクリックし、そのシナリオの編集を開始します。

ディスカバリー・シナリオで使用されている分類ポリシーは、シナリオ名の後に括弧で囲まれています。例えば、分類ポリシー `policy_cadl` を使用している `discover_cadl` という名前のディスカバリー・シナリオは、discover_cadl (policy_cadl) と表示されます。 いくつかのディスカバリー・シナリオおよびテンプレートがデフォルトで提供され、以下が含まれます。

GDPR [テンプレート]
: 「GDPR [テンプレート]」シナリオは、GDPR コンプライアンス戦略のための最新セットのディスカバリー・ルールと言語サポートを提供します。テンプレートはコピーまたは編集して別の名前で保存できます。「GDPR [テンプレート]」は常に最新の GDPR ディスカバリー・ルールと言語サポートを受け取ります。

GDPR
: 「GDPR」シナリオは、GDPR 対応の戦略の一部として使用できるディスカバリー・ルールの基本セットを提供します。「GDPR」シナリオを編集して変更内容を保存できますが、このシナリオは時間の経過に伴い更新されたルールおよび言語サポートを受け取りません。
重要: 「GDPR [テンプレート]」が使用できる場合、古い「GDPR」シナリオは更新を受け取らないため、「GDPR」シナリオを使用することは推奨されません。

事前定義されているディスカバリー・テンプレートを編集して変更を保存する場合、Guardium により、シナリオは「［ディスカバリー・テンプレート名］ の ［タイム・スタンプ］ のコピー」形式の固有の名前で自動的に保存されます。例えば、「GDPR [テンプレート]」シナリオを編集して保存すると、ディスカバリー・シナリオの名前は「GDPR [テンプレート] の [2018-01-01 12:00:00] のコピー」になります。

親トピック: 機密データのディスカバー
次のトピック: 名前および記述

名前および記述

ディスカバリー・シナリオの名前と記述を入力します。

このタスクについて

このステップの間に、ディスカバリー・シナリオにアクセス可能な*セキュリティー・ロール*を指定することもできます。

手順

1. 「名前および記述」セクションを開き、シナリオの名前と記述 (オプション) を指定または編集します。
2. ディスカバリー・シナリオで使用する分類ポリシーを作成または選択します。
   -  アイコンをクリックして新規分類ポリシーの名前を設定します。ディスカバリー・シナリオの保存時に分類ポリシーが作成されます。
   - ドロップダウン・メニューを使用して、既存の分類ポリシーを選択します。分類ポリシーのルールは、機密データのディスカバー・ツールの「ディスカバー対象」セクションに自動的にロードされます。事前定義されている分類ポリシーを選択して編集する場合、Guardium により、ポリシーは「［ポリシー・テンプレート名］ の ［タイム・スタンプ］ のコピー」 形式の名前で自動的に保存されます。例えば、「GDPR [テンプレート]」ポリシーを編集して保存すると、分類ポリシーの名前は「GDPR [テンプレート] の [2018-01-01 12:00:00] のコピー」になります。
3. タグ付け違反用のカテゴリー・ラベルと分類ラベルを指定します。 カテゴリー・ラベルと分類ラベルのデフォルト値は「機密 (Sensitive)」です。
4. オプションで「ロール」ボタンをクリックして、ディスカバリー・シナリオにアクセスできる*セキュリティー・ロール*を指定します。

次のタスク

ディスカバリー・シナリオの次のセクション、「ディスカバー対象」に進みます。
親トピック: 機密データのディスカバー
前のトピック: ディスカバリー・シナリオ
次のトピック: ディスカバー対象

ディスカバー対象

機密データをディスカバーして分類するためのルールとルール・アクションから成るポリシーを作成します。

このタスクについて

*分類ポリシー*には、機密データを特定してアクションを実行する一連のルールとルール・アクションが、順序を付けて格納されます。ポリシー内の各ルールは、そのルールと一致した場合に実行される条件付きアクションを定義します。条件付きテストは単純なものでも (表内のどこでも検出されるワイルドカード文字列など)、複数の条件を考慮する複雑なテストでも構いません。機密データのディスカバリー・シナリオの場合、ルールによってトリガーされるアクションとして、指定されたグループにオブジェクトを追加するグループ化アクションや、ルールが一致すると通知がトリガーされるアラート・アクションが可能です。 複数のグループ化アクションとアラート・アクションを結合して、一致したルールに対する高度な応答を作成するように指示することができます。

このタスクでは、ディスカバリー・シナリオで使用する分類ルールとルール・アクションを作成および編集するプロセスについて説明します。

手順

1. 「ディスカバー対象」セクションを開き、データをディスカバーするためのルールを定義します。
2. 「言語」メニューを使用して、選択した言語、および選択した言語が国語である国によって、ルール・テンプレートをフィルタリングします すべての「言語」メニュー選択項目で、クレジット・カード番号や E メール・アドレスのような汎用パターンのテンプレートが表示されます。
3. 以下のいずれかを行うことにより、ルールをディスカバリー・シナリオに追加するか、既存のルールを編集します。
   -  アイコンをクリックし、新規ルールを作成します。

- 「分類ルール・テンプレート」表からルールを選択し、  アイコンをクリックして事前定義ルールを追加します。
- 既存のルールを編集するには、  アイコンをクリックします。

4. 分類ルールを追加または編集する場合、以下の手順を実行します。
   a. 実行する検索のタイプに基づいて、「ルール・タイプ」を選択します。
      - 「データの検索」は、データ内の特定のパターンまたは値に一致します
      - 「カタログ検索」は、データベース・カタログ内の表名または列名に一致します
      - 「非構造化データの検索」は、非構造化データ・ファイル (例えば、CSV ファイル、TXT ファイル、または CEF ファイル) 内の特定の値またはパターンに一致します
   b. 名前と記述を指定します。その際にオプションで、「名前」フィールドの先頭に*特殊パターン・テスト* を指定できます。 ルール名は、「分類ポリシー・ビルダー」で、分類ポリシーと関連付けられたルールの命名にも使用されます。特殊パターン・テスト が必要な場合は、それに対応するテンプレートで作業をすることをお勧めします (例えば、クレジット・カード番号の場合は、銀行カード - クレジット・カード番号 を使用します)。
   c. 「ルール基準」セクションを開き、このルールの*正規表現* およびその他の検索条件を定義します。ルール・テンプレートで作業をしている場合は、デフォルトで適切な*正規表現* が指定されます。
   重要: 「機密データのディスカバー」シナリオで作成されたルールの場合、デフォルトの「データ・タイプ」には、「数値」と「テキスト」の両方が含まれます。
   d. 「アクション」セクションを開き、ルール基準と一致した場合に実行する*ルール・アクション* を定義します。
   e. 複数の*ルール・アクション* を定義するとき、オプションで  アイコンをクリックし、  アイコンと  アイコンを使用してアクションの実行順序を変更できます。
   f. ルール定義の追加または編集が完了したら、「保存」をクリックし、ディスカバリー・シナリオの「ディスカバー対象」セクションに戻ります。
5. オプションで、  アイコンをクリックして、  アイコンと  アイコンを使用してルールを適用する順序を変更します。 デフォルトの動作では、「ルール基準」で「突き合わせを続行」が選択されていない限り、最初の一致後にルールの実行が停止するため、ルールの順序は重要です。
6. ルールの作業が完了したら、「次へ」をクリックし、ディスカバリー・シナリオの次のセクションの作業を始めます。

次のタスク

ディスカバリー・シナリオの次のセクション、「検索場所」に進みます。
親トピック: 機密データのディスカバー
前のトピック: 名前および記述
次のトピック: 検索場所
関連概念:
正規表現
関連資料:
実際のメンバー内容
ルール基準
特殊パターン・テスト

ルール基準

表 1.

属性	記述
表タイプ	検索対象の表タイプとして、「シノニム」、「表」、または「ビュー」のうち 1 つ以上を選択します。デフォルトでは「表」が選択されています。
データ・タイプ	検索対象のデータ・タイプとして、「数値」、「テキスト」、または「日付」のうち 1 つ以上を選択します。デフォルトでは「数値」と「テキスト」が選択されています。
検索式	オプションで、マッチングする検索パターンを定義するための正規表現を入力します。正規表現をテストするには、「RE」ボタンをクリックし、正規表現エディターを開きます。
表名 LIKE	オプションで、特定の名前またはワイルドカード・パターンを入力します。これを省略すると、すべての表名が選択されます。
列名 LIKE	オプションで、特定の名前またはワイルドカード・パターンを入力します。これを省略すると、すべての列名が選択されます。
突き合わせを続行	このルールが一致した後で、分類ポリシーの次のルールを評価する必要がある場合は、 「突き合わせを続行」チェック・ボックスにマークを付けます。 デフォルトでは、ルールが一致した時点でルールの評価が停止します。
検索ワイルドカード	オプションで、特定の値またはワイルドカード・パターンを入力します。これを省略すると、すべての値が選択されます。
最小長	オプションで、「最小長」を入力します。これを省略すると、長さ制限がなくなります。
最大長	オプションで、「最大長」を入力します。これを省略すると、長さ制限がなくなります。
評価名	オプションで、作成およびアップロード済みの完全修飾 Java™ クラス名を入力します。 この Java クラス名は、文字列の起動および評価に使用されます。 注: 入力されたクラス名がロード済みであるか、およびインターフェースに順応しているかどうかの検証は行われません。

属性	記述
「限定起動」マーカー	「限定起動」マーカーを使用すると、分類ルールをグループ化することができます。同じマーカーを持つルールは同時に起動されます。さらに、1 つのマーカーを使用して返されるルールはすべて、同じ名前の表に基づいてデータを返す必要があります。 同じマーカーを使用して 2 つ以上のルールが定義されている場合、それらのルールは一緒に起動されます。両方のルールが同じ表で起動された場合、それらは両方ともログに記録され、それぞれのアクションが呼び出されます。 これに対して、ある表でいずれか一方のルールのみが起動された場合、ルールはどちらもログに記録されず、それらのアクションも呼び出されません。複数のルールを一緒に起動できるようにすることは、同じ表内に複数の機密データが同時に現れる場合が懸念されるときに重要になります。 例えば、1 つの表に社会保障番号とマサチューセッツ州の運転免許証の両方が含まれる場合に、それを知ることができます。 「限定起動」マーカーは定数値であり、任意の値を指定でき、グループ化するルール全体でまったく同じ値でなければなりません。 つまり、1 つのルールのマーカー名が ABC であれば、そのルールと一緒にグループ化する他のルールのマーカー名も ABC でなければなりません。 「限定起動」マーカーは、「突き合わせを続行」フラグとも連携します。 例えば、以下のルールが定義されており、ルール 3 が「突き合わせを続行」と一致しない場合、他の 3 つのマーカー・ルールがすべて正となったかどうかに関係なく、結果は返されません。これは、ルール 4 の実行まで至らなかったためです。必ずすべての「限定起動」マーカーが実行され、結果が正になる必要があるので、このグループ化は起動されません。 ルール 1. 起動マーカー・ルール "ABC"(突き合わせを続行) ルール 2. 起動マーカー・ルール ABC (突き合わせを続行) ルール 3. 起動マーカー以外のルール・タイプ (突き合わせを続行) ルール 4. 起動マーカー・ルール "ABC"(突き合わせを続行)
ヒット率	オプションで、このルールを起動するために達成しなければならない一致データのパーセンテージを入力します。検査された一致データのパーセンテージが入力したパーセンテージ値以上 (>=) であれば、データが返されます。ただし、項目が空の場合、それは条件ではなく、ルールが起動されるかどうかに影響せず、ビュー画面にデータが返されることを意味します。 パーセンテージ 0 を指定すると、そのルールはこの条件で起動され、データがビュー画面に返されます。パーセンテージ 100 を指定すると、すべてが一致することが求められます。

SQL の値と比較	オプションで、「SQL ステートメント」を入力します。入力した SQL (唯一の列から情報が返されることが基本条件となっている必要があります) は、選択された表および列に対して検索を行うための値グループとして使用されます。
	注: 「SQL の値と比較」を使用する場合には、以下のルールに従う必要があります。
	
	• SQL ステートメントは `SELECT` で始まる必要がある。
	• SQL ステートメントには ; (セミコロン) を使用できない。
	• 入力した SQL で、正確に結果が返されるようにスキーマ値の名前を指定する必要がある。
	
	• 良い例:
	
	`SELECT ename FROM scott.emp`
	`select EMPNUMBER from SYSTEM.EMP where EMPNUMBER in(5555,4444)`
	`select DNAME from SCOTT.DEPT where DNAME like 'A%G'`
	`SELECT ZIP from SCOTT.FOO where ZIP in (SELECT ZIP FROM SCOTT.FOO)`

グループの値と比較	オプションで、グループを選択します。選択されたグループは、選択された表および列に対して検索を行うための値グループとして使用されます。グループ (PUBLIC または分類グループ) 内の値のいずれかが一致していれば、値のルールによってデータが返されます。
固有値の表示	「固有値の表示」チェック・ボックスにマークを付けると、分類ポリシー・ルールと一致した値の詳細が結果レポートのコメント・フィールドに追加されます。
固有値のマスク	固有値を編集する場合は、「固有値のマスク」フィールドで正規表現を使用します。例えば、「固有値の表示」チェック・ボックスにマークを付け、「固有値のマスク」フィールドで「`([0-9]{2}-[0-9]{3})-[0-9]{4}`」を使用して、最後の 4 桁をログに記録し、接頭部の数字を編集します。

親トピック: ディスカバー対象

実際のメンバー内容

「実際のメンバー内容」フィールドを使用して、「オブジェクトのグループに追加」ルール・アクションによるオブジェクトのラベル付け方法を定義します。

表 1.

「実際のメンバー内容」の選択肢	グループの値
オブジェクト名のみ	表名
Like Name%	tableName%
Like %Name	%tableName
Like %Name%	%tableName%
%/%.Name	%%.tableName
完全修飾名	schemaName.tableName
Like Full%	schemaName.tableName%
Like %Full	%schemaName.tableName
Like %Full%	%schemaName.tableName%
%/Full	%%.schemaName.tableName
Read/%.Name	Read/%.tableName

「実際のメンバー内容」の選択肢	グループの値
Change/%.Name	Change/%.tableName
Read/Full	Read/schemaName.tableName
Change/Full	Change/schemaName.tableName

ルールによって表名 `JJ_CREDIT_CARD` がスキーマ `DB2INST1` から返され、なおかつ「オブジェクトのグループに追加」アクションが指定されている場合、「実際のメンバー内容」の各選択項目は以下のように動作します。

- 「完全修飾名」を選択した場合は、`DB2INST1.JJ_CREDIT_CARD` が選択したグループに追加されます。
- 「オブジェクト名のみ」を選択した場合は、`JJ_CREDIT_CARD` が選択したグループに追加されます。
- 「Change/Full」を選択した場合は、`Change/DB2INST1.JJ_CREDIT_CARD` が選択したグループに追加されます。

親トピック: ディスカバー対象

検索場所

機密データをスキャンするデータ・ソースを特定します。

このタスクについて

データ・ソースには、データベースやリポジトリーに関する情報 (データベースのタイプ、リポジトリーの場所、関連付けられる可能性のある認証資格情報など) が格納されます。データ・ソースをディスカバリー・シナリオに追加すると、選択されたデータ・ソースに*分類ポリシー*が適用されている場所に*分類プロセス*が作成されます。

このタスクでは、機密データの検索対象となるデータ・ソースを特定します。

手順

1. 「検索場所」セクションを開き、機密データの検索対象となるデータ・ソースを特定します。
2. 以下のいずれかを行うことにより、データ・ソースをディスカバリー・シナリオに追加します。
   - アイコンをクリックして「データ・ソースの作成」ダイアログを開き、新規のデータ・ソース定義を追加します。
   - 「選択可能なデータ・ソース」表からデータ・ソースを選択し、アイコンをクリックして、既存のデータ・ソースを追加します。
3. 新規のデータ・ソースを定義するか、または既存のデータ・ソースを選択し、アイコンをクリックしてそのデータ・ソースを編集します。 ディスカバリー・シナリオ経由で定義された新規データ・ソースは、「データ・ソース定義」ツールを使用して表示や編集を行うこともできます。
   a. データ・ソースの名前を入力または編集します。
   b. 「データベース・タイプ」メニューから適切なデータベース・タイプを選択し、データ・ソース定義を完了するために必要な情報を入力します。使用可能なフィールドは、選択したデータベース・タイプによって異なります。
   c. データ・ソース定義の編集が完了したら、「保存」をクリックして作業内容を保存します。オプションで「接続のテスト」をクリックし、データ・ソース接続を検査します。
   d. データ・ソース定義の作業が完了したら、「閉じる」をクリックしてダイアログを閉じます。
4. クラウド・データベースにこの分類プロセスを使用する場合は、「Cloud DB のオブジェクト監査の有効化」も選択します。
5. データ・ソースの追加が完了したら、「次へ」をクリックし、ディスカバリー・ワークフローの次のセクションの作業を始めます。

タスクの結果

*分類プロセス*は、データ・ソースをディスカバリー・シナリオに追加して、そのシナリオを保存した後に作成されます。 このプロセスを表示して直接編集するには、「分類プロセス・ビルダー」を使用します。

次のタスク

ディスカバリー・ワークフローの次のセクション、「ディスカバリーの実行」に進みます。
親トピック: 機密データのディスカバー
前のトピック: ディスカバー対象
次のトピック: ディスカバリーの実行およびレポートのレビュー
関連概念:
データ・ソース
関連タスク:
データ・ソース定義の作成

ディスカバリーの実行およびレポートのレビュー

オプションでディスカバリー・シナリオを実行し、結果をレビューします。

このタスクについて

機密データをディスカバーし、検索するデータ・ソースを特定するためのポリシーを定義した後、*分類プロセス*を実行して結果をレビューすることができます。プロセスを実行して結果をレビューすると、ポリシーを改良することができます。例えば、結果の範囲が広すぎる場合は、追加の検索条件を指定します。希望どおりの結果を得るには、ポリシーの改良、プロセスの実行、および結果の評価を数回繰り返すことが必要な場合があります。

手順

1. 「ディスカバリーの実行」セクションを開き、ディスカバリー・シナリオをテストします。
2. 「今すぐ実行する」をクリックして開始します。

重要:
- 指定したポリシーと、検索対象として選択したデータ・ソースの数によっては、機密データの特定プロセスが完了するまで数分以上かかる場合があります。このプロセスの状況は、「今すぐ実行する」ボタンの横に表示されます。また、「Guardium ジョブ・キュー」オプションを使用して、プロセスをモニターすることもできます。

3. ディスカバリー・シナリオの実行が完了したら、「レポートのレビュー」セクションを開いて結果を確認します。
   - 「生成時刻」メニューを使用して、表示するレポート・インスタンスを選択します。
   - アイコンをクリックすると、レポートに含まれているデータ・ソースがリストされます。
   - 別名や階層グループなどのレポート設定を調整するには、アイコンをクリックします。
   - 「プロセス・ログ」を開いて、詳細なログ情報を確認します。
   - 結果を絞り込む場合は、「フィルター」ボックスを使用します (結果の数が 10,000 件を超えている場合、フィルタリング機能は使用できません)。
4. 結果をレビューしながら、結果に基づいて追加のルールやアクションを定義できます。
   a. アクションを定義する対象のデータを含む行 (複数可) を選択します。
   b. 「グループに追加」をクリックしてグループ化アクションを定義するか、「拡張アクション」をクリックして、アラート、ロギング、または無視などの他のアクションを定義します。
   c. アクションを定義するダイアログの入力が完了したら、「OK」をクリックして結果レポートに戻ります。
   重要:
   - 結果表から追加されたアクションは、結果表から呼び出した場合にのみ実行される特別なアクションとして認識されます。これらのアクションは、ディスカバリー・シナリオの「ディスカバー対象」 > 「ルールの編集」 > 「アクション」セクションには表示されません。また、ディスカバリー・シナリオや関連する *分類プロセス* の一部として自動的に実行されることもありません。
   - アラート・アクションおよびアクセス・ルールのレビュー、編集、インストールを行うには、「ポリシー・ビルダー」を使用します。
   - グループ化アクションをレビューおよび編集するには、「グループ・ビルダー」を使用します。
   - プライバシー・セットのアクションをレビューするには、「プライバシー・セット・ビルダー」を使用します。
   - ポリシー・ロギング・アクションをレビューするには、「インシデント管理」ツールを使用します。
5. 結果レポートのレビューが完了したら、「次へ」をクリックし、ディスカバリー・シナリオの次のセクションの作業を始めます。

タスクの結果

機密データの検索を実行したら、「今すぐ実行する」ボタンの横に表示される検索プロセスの状況をモニターします。または、「Guardium ジョブ・キュー」オプションを使用することもできます。「グループ・ビルダー」を使用すると、任意のグループ化アクションをレビューすることができます。「ポリシー・ビルダー」を使用すると、結果表から追加された任意のアラート・アクションのレビューとインストールを行うことができます。

次のタスク

(オプション) ディスカバリー・シナリオの次のセクション、「監査」に進みます。
親トピック: 機密データのディスカバー
前のトピック: 検索場所
次のトピック: 監査

監査

オプションで、ディスカバリー・レポートおよび分類レポート用の受信者、配布順序、およびレビュー・オプションを定義することによって監査プロセスを作成します。

このタスクについて

ディスカバリー・ワークフローの結果に対して、任意の数の受信者を定義できます。また、受信者が結果を受け取る順序を制御することもできます。さらに、結果が次の受信者に送信される前に、受信者が結果に署名する必要があるかどうかなどのプロセス制御オプションも指定できます。

手順

1. 「監査」セクションを開き、ディスカバリー・レポートの受信者を定義します。
2. 「監査プロセス」メニューを使用して、ディスカバリー・シナリオで使用する監査プロセスを選択します。 新規ディスカバリー・シナリオを作成する場合、新規監査プロセス名が `[scenario name] Audit process [timestamp]` の形式で推奨されます。例えば、discover_cadl というシナリオを作成すると、監査プロセスの名前は disover_cadl Audit process [2018-01-01 12:00:00] になります。
3. アイコンをクリックし、レポートの配布方法を指定するオプションを定義して、レポートの受信者をディスカバリー・シナリオに追加します。
   - Guardium ユーザー、ロール、またはグループにレポートを送信する場合は、プロセス制御オプションを定義する必要があります。
   - E メール受信者にレポートを送信する場合は、E メール・アドレスを指定し、その E メール受信者に適した Guardium ユーザー名でレポートをフィルターに掛けます。
4. 「OK」をクリックして、受信者をディスカバリー・ワークフローに追加します。 必要に応じて、追加の受信者をシナリオに追加します。
5. オプションで、アイコンをクリックして、アイコンと アイコンを使用して、レポートが受信者に配布される順序を変更します。 これにより、レポートが次の受信者に送信される前にどの受信者がレポートのレビューまたは署名を行う必要があるかが決まるため、*順次* 配布を使用する場合はこれが重要です。
6. 受信者の追加、編集、順序付けが完了したら、「次へ」をクリックし、ディスカバリー・ワークフローの次のセクションの作業を始めます。

タスクの結果

監査プロセス は、受信者を定義して、ディスカバリー・シナリオを保存した後に作成されます。 このプロセスの表示、編集、実行を直接行うには、「監査プロセス・ビルダー」を使用します。

監査プロセス は、ディスカバリー・シナリオの「スケジュール」セクションを使用してスケジュールされるか、「監査プロセス・ビルダー」を使用してスケジュールされるまで、非アクティブのままです。「監査プロセス・ビルダー」にアクセスして、*監査プロセス* を選択し、「今すぐ 1 回実行」をクリックすることで、*監査プロセス* を実行することもできます。

次のタスク

(オプション) ディスカバリー・ワークフローの次のセクション、「スケジュール」に進みます。
親トピック: 機密データのディスカバー
前のトピック: ディスカバリーの実行およびレポートのレビュー
次のトピック: スケジューリング
関連概念:
監査プロセスの作成

スケジューリング

オプションで、定義された間隔で実行するように監査プロセスをスケジューリングすることによって、監査プロセスをアクティブにします。

このタスクについて

スケジュールは、ディスカバリー・シナリオの「監査」セクションで受信者が指定されていれば、それと合わせて*監査プロセス*の一部となります。 スケジュールを定義することにより、指定した間隔で*監査プロセス*が実行され、関連付けられている*分類プロセス*からの結果が定期的に配布されてレビューされるようになります。

手順

1. 「スケジュール」セクションを開き、データをディスカバーするためのスケジュールを定義します。
2. 「スケジュールの基準」メニューを使用して、*監査プロセス*の間隔 (日次または月次) を設定します。
3. 「スケジュール開始間隔」チェック・ボックスと「繰り返しの間隔」チェック・ボックスを使用して、*監査プロセス*を実行する 1 日あたりの回数と毎時間内の回数を定義します。
4. 「開始日時」コントロールを使用して、スケジュールを開始する明示的な日時を定義します。
5. 「スケジュールのアクティブ化」チェック・ボックスをクリアして、スケジューリング情報を後で使用できるように保持しながら、*監査プロセス*を非アクティブにします。 「スケジュールのアクティブ化」ボックスは、デフォルトではチェック・マークが付いています。これは、スケジュールを保存した後、監査プロセスがアクティブになることを意味します。
6. スケジュールの定義が完了したら、「保存」をクリックして編集を完了し、ワークフロー・エディターを閉じます。

タスクの結果

*監査プロセス*は、スケジュールを定義して、ディスカバリー・シナリオを保存した後に作成されます。 この監査プロセスを表示して直接編集するには、「監査プロセス・ビルダー」を使用します。スケジュールされている監査タスクの状況、開始時刻、次回の起動時間を確認するには、「スケジュール済みジョブ」レポートを表示します。
親トピック: 機密データのディスカバー
前のトピック: 監査
関連概念:
監査プロセスの作成

正規表現

正規表現を使用して、データに含まれる複合パターンを求めてトラフィックを検索することができます。

IBM Guardium の正規表現の実装は POSIX 1003.2 に準拠します。詳しくは、Open Group の Web サイト www.opengroup.org を参照してください。 正規表現を使用して、データに含まれる複合パターンを求めてトラフィックを検索することができます。例については、『ポリシー』を参照してください。

このヘルプ・トピックでは、正規表現の作成ツールの使用法について説明し、一般的に使用される特殊文字や構造の表を示します。 正規表現の構造や使用方法についての包括的な説明は しません。詳しくは、Open Group の Web サイトを参照してください。

正規表現を使用したパターン・マッチングまたは XML マッチングで 注意すべき重要なポイントは、一致の検索は文字列の先頭から開始し、 その表現と一致する最初のシーケンスが検出されると停止する ということです。異なる正規表現または同じ正規表現を、パターン・マッチングおよび XML マッチングに同時に使用できます。

注: IBM Guardium は、英語以外の言語の正規表現はサポートしていません。

正規表現の作成ツールの使用

入力フィールドで正規表現の入力が要求される場合、正規表現の作成ツールを使用して、 正規表現のコード化およびテストを行うことができます。「正規表現の作成」アイコンは、ポリシー・ビルダーの「ルールの追加」の下にあります。

正規表現の作成ツールを開くには、正規表現を入力するフィールドの隣にある RE アイコンをクリックします。フィールドに既に何か入力されている場合は、入力された内容が「正規表現の作成」パネルの「正規表現」ボックスにコピーされます。

1. ドロップダウン・リストから正規表現のカテゴリーを選択します。
2. ドロップダウン・リストからパターンを選択します。
3. 「正規表現 」ボックスで、表現を入力または変更します。
4. 表現をテストするには、「突き合わせるテキスト」ボックスにテキストを入力して、「テスト」ボタンをクリックします。
   - 表現にエラー (右中括弧の欠落など) が含まれている場合は、 「構文エラー」メッセージ で通知されます。
   - 「一致するものが見つかりました」というメッセージは、入力したテキスト内に正規表現との一致が検出されたことを示します。
   - 一致が検出されない場合は、「一致するものが見つかりません」というメッセージが表示されます。
5. ステップを何度も繰り返し、目的に合わせて、正規表現が予想どおりに一致すること、および一致しないことを検査することをお勧めします。
6. 表現の最後に特殊文字を入力する場合は、 「要素の選択 」リストから選択できます。他の場所に特殊文字を入力する場合は、 その文字を入力するか、コピーする必要があります。
7. 変更およびテストが終了したら、 「OK」をクリックし、 「正規表現の作成」パネルを閉じて、正規表現を定義パネルにコピーします。

特殊文字と構造

以下の表に、一般的に使用される特殊文字と構造のサマリーを 示します。

表 1. 特殊文字と構造

文字	使用法	例	一致	不一致
リテラル	以下に示す 特殊文字以外の文字 (大/小文字の区別あり) の正確なシーケンスに一致	can	can	Can cab caN
. (ドット)	復帰 または改行 (¥n) 文字を含む、すべての文字に一致	ca.	can cab	c cb
*	先行する 文字のゼロ個以上のインスタンスに一致	Ca*n	Cn Can Caan	Cb Cabn
^	後続の文字で 始まる文字列に一致	^C.	Ca	ca a
$	先行する文字で 終了する文字列に一致	C.n$	Can Cn	Cab
+	先行する文字の 1 つ以上のインスタンスに一致	^Ca+n	Can Caan	Cn
?	先行する文字の ゼロまたは 1 つのインスタンスに一致	Ca?n	Cn Can	Caan
|	先行するパターン または後続のパターンのいずれかと一致	Can|cab	Can cab	Cab
(x ...)	括弧で 囲まれたシーケンスと一致	(Ca)*n	Can XaCan	Cn CCnn
{n}	先行する文字の正確に n 個のインスタンスと一致	Ca{3}n	Caaan	Caan Caaaan
{n,}	先行する文字の n 個以上のインスタンスに一致	Ca{2,}n	Caan Caaaan	Can Cn
{n,m}	先行する文字の n 個から m 個のインスタンスに一致	Ca{2,3}n	Caan Caaan	Can Caaaan
[a-ce]	セット内の 単一文字に一致。ダッシュは連続するシーケンスを示す。 例えば、[0-9] は任意の数字に一致。	[C-FL]an	Can Dan Lan	Ban
[^a-ce]	指定したセットに ない文字に一致	[^C-FL]an	aan Ban	Can Dan
[[.char.]]	囲まれた文字、または「名前付き文字の表」に含まれる名前付き文字に一致	[[.~.]]an または [[.tilde.]]an	~an	@an
[[:class:]]	「文字クラス表」の指定された文字クラスに含まれる任意の文字に一致	[[:alpha:]]+	abc	ab3

名前付き文字の表 (英語)

以下の表で、正規表現の大括弧ペア内で使用できる標準文字名について説明します ([[.char]])。 文字名はロケーション固有であるため、 英語版以外の Guardium® では異なる文字名のセットを使用している場合があります。

- NUL ¥0
- SOH ¥001
- STX ¥002
- ETX ¥003
- EOT ¥004
- ENQ ¥005
- ACK ¥006
- BEL ¥007
- alert ¥007
- BS ¥010
- backspace ¥b
- HT ¥011
- tab ¥t
- LF ¥012
- newline ¥n
- VT ¥013
- vertical-tab ¥v
- FF ¥014
- form-feed ¥f
- CR ¥015
- carriage-return ¥r
- SO ¥016
- SI ¥017
- DLE ¥020
- DC1 ¥021
- DC2 ¥022
- DC3 ¥023
- DC4 ¥024
- NAK ¥025
- SYN ¥026
- ETB ¥027
- CAN ¥030
- EM ¥031
- SUB ¥032
- ESC ¥033
- IS4 ¥034
- FS ¥034
- IS3 ¥035

- GS ¥035
- IS2 ¥036
- RS ¥036
- IS1 ¥037
- US ¥037
- space ' '
- exclamation-mark !
- quotation-mark "
- number-sign #
- dollar-sign $
- percent-sign %
- ampersand &
- apostrophe ¥'
- left-parenthesis (
- right-parenthesis)
- asterisk *
- plus-sign +
- comma ,
- hyphen -
- period .
- full-stop .
- slash /
- solidus /
- zero 0
- one 1
- two 2
- three 3
- four 4
- five 5
- six 6
- seven 7
- eight 8
- nine 9
- colon :
- semicolon ;
- less-than-sign <
- equals-sign =
- greater-than-sign >
- question-mark ?
- commercial-at @
- left-square-bracket [
- right-square-bracket]
- backslash ¥
- reverse-solidus ¥¥
- circumflex ^
- circumflex-accent ^
- underscore _
- low-line _
- grave-accent `
- left-brace {
- left-curly-bracket {
- right-brace }
- right-curly-bracket
- vertical-line |
- tilde ~
- DEL 177
- NULL 0

名前付き文字クラス表 (英語)

以下の表で、正規表現の大括弧ペア内で参照できる標準文字クラスについて説明します ([[:class:]])。 文字クラスはロケーション固有であるため、英語版以外の Guardium では異なる文字名のセットを使用している場合があることに注意してください。

- alnum - 英数字 (a-z、A-Z、0-9)
- alpha - 英字 (a-z、A-Z)
- blank - 空白文字 (ブランク、改行、復帰)
- cntrl - 制御
- digit - 0-9
- graph - グラフィックス
- lower - 英小文字 (a-z)
- print - 印刷可能文字
- punct - 句読文字
- space - スペース、タブ、改行、および復帰
- upper - 英大文字
- xdigit - 16 進数の数字 (0-9、a-f)

正規表現の例

任意の正規表現をコピーして、 正規表現の入力が要求されるフィールドに貼り付けることができます。これらの例を使用する場合は、正規表現の作成ツールでその例を使用し、一致または不一致のさまざまな値を入力して試すことを強くお勧めします。これによって、その表現と突き合わせられるものを正確に理解することができます。

正規表現の例

社会保障番号 (ハイフンが必要) [0-9]{3}-[0-9]{2}-[0-9]{4}

電話番号 (北アメリカ - 3334445555、333.444.5555、333-444-5555、333 444 5555、(333) 444 5555、およびそのすべての組み合わせと一致) ¥(?[0-9]{3}¥)?[-.]?[0-9]{3}[-.]?[0-9]{4}

郵便番号 - (カナダ) [ABCEGHJKLMNPRSTVXY][0-9][A-Z] [0-9][A-Z][0-9]

郵便番号 - (英国) [A-Z]{1,2}[0-9][A-Z0-9]? [0-9][ABD-HJLNP-UW-Z]{2}

郵便番号 - (米国) (5 桁が必須で、ハイフンと 4 桁が続く場合がある) [0-9]{5}(?:-[0-9]{4})?

クレジット・カード番号 [0-9]{4}[-,]?[0-9]{4}[-,]?[0-9]{4}[-,]?[0-9]{4}

親トピック: ディスカバー

ファイル・サーバー内での機密データのディスカバーおよび分類

ファイル・アクティビティー・モニターは、UNIX ファイル・サーバーおよび Windows ファイル・サーバー上の機密データの保全性と保護を確保します。

- FAM コンポーネントのインストールおよびアクティブ化
  GIM クライアントをファイル・サーバーにインストールしてから、それを使用してファイル・アクティビティー・モニター・ディスカバリー・エージェントをインストールします。
- Windows ファイル・サーバー上のファイル・アクティビティー・モニターの無効化
  ファイル・アクティビティー・モニターは、GIM クライアントを使用して無効に設定されます。
- ファイルのディスカバリーおよび分類 GIM パラメーター
  以下の GIM パラメーターを使用して、コレクターごとにファイルのモニター、ディスカバリー、および分類を構成します。
- FAM 判定プランのカスタマイズ
  判定プランは、ファイル内の機密の内容を識別するために使用されます。Guardium FAM ディスカバリー・エージェントは、GDPR、HIPAA、PCI、SOX、およびソース・コードのデフォルトの判定プランを提供します。デフォルトの判定プランを使用して、結果のレポート/調査ダッシュボードから分類エンティティーを変更できます。また、IBM コンテンツ分類ワークベンチを使用して、新しいプランを作成したり、既存のプランを変更したりできます。
- GDPR ファイル・アクティビティーのルール
  FAM GDPR の判定プランを作成するには、下記のルールを使用してください。
- FAM 判定プラン・ファイルのアップロードおよび削除
  判定プランを作成した後、そのプランをファイル・サーバー上のファイルの分類のために Guardium システムにアップロードします。

親トピック: ディスカバー
関連情報:
Guardium を使用したファイル・アクティビティーのモニター (ビデオ)

FAM コンポーネントのインストールおよびアクティブ化

GIM クライアントをファイル・サーバーにインストールしてから、それを使用してファイル・アクティビティー・モニター・ディスカバリー・エージェントをインストールします。

始める前に

- ライセンス・キーをインストールする必要があります。『ライセンス・キーのインストール』を参照してください。
- S-TAP for FAM がインストールされている必要があります。ファイル・モニターおよびポリシー実施のために必要です。まだインストールされていない場合、GIM を使用して FAM ディスカバリー・エージェントと一緒にインストールできます。
- この手順を開始する前に、bash シェルをインストールしておく必要があります。
- FAM ディスカバリー・エージェント (別名 FAM バンドルまたは FAM エージェント) にアクセスできなければなりません。これはファイルのディスカバリーおよび分類に必須です。Fix Central からダウンロードするか、Guardium 担当員から入手してください。
- FAM バンドルのディスク・スペース所要量は 2 GB です。AIX プラットフォームでは、インストール時にさらに 2 GB が必要です。

ヒント: FAM ディスカバリー・エージェントを AIX で正常にインストールするには、/etc/security/limits file: default: data = -1 の行を変更して、プロセス・データのサイズを無制限に設定することが推奨されます。

手順

1. ファイル・サーバーに GIM クライアントをインストールします。『Guardium Installation Manager』を参照してください。
2. FAM バンドルをダウンロードして、アクセス可能なドライブに保存します。 ファイル・サーバー OS 用の正しいモジュールを選択します。UNIX バンドルの名前は次のようになります。 guard-bundle-FAM_r*****_trunk_*****.gim. Windows バンドルは次のようになります。 guard-FAM-guardium_r*****Windows-Server-x86_x64_ia64.gim.
3. S-TAP もインストールする場合は、FAM バンドルのインストール前にインストールしてください。S-TAP を Fix Central からダウンロードして、次のステップに記載されている指示に従ってください。S-TAP は、FAM バンドルのインストール前にインストールする必要があります。
4. 中央マネージャーが存在する場合は中央マネージャーで、存在しない場合はアプライアンスで、FAM バンドルをアップロードしてインポートします。
   a. 「管理」 > 「モジュール・インストール」 > 「モジュールのアップロード」にナビゲートします。
   b. 「モジュールのアップロード」で「参照」をクリックして、FAM バンドルにナビゲートします。「アップロード」をクリックします。
   c. 「アップロード済みモジュールのインポート」で、FAM バンドルを選択して、「インストール/更新」をクリックします。
5. 「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」を使用して、FAM バンドルをインストールして構成します。 GIM について詳しくは、クライアント別の設定を参照してください。

a. FAM モニターを有効にするには、STAP_FAM_ENABLED (UNIX) または WSTAP_FAM_ENABLED (Windows) を 1 (有効化) に設定します。この手順は、FAM ディスカバリー・エージェントのみを使用する場合でも実行する必要があります。
b. FAM ディスカバリーはデフォルトで有効になっています (FAM_ENABLED)。 必要に応じて追加のパラメーターを構成します。
- スキャンするディレクトリーの SOURCE_DIRECTORIES を構成します。
- デフォルトでは、エージェントはライセンス情報の基本スキャンを実行します。SOX、HIPAA など、判定プランに基づいたスキャンを有効にするには、FAM_IS DEEP_ANALYSIS を true に設定します。 デフォルトでは、すべてのデフォルト判定プランが使用されます。使用される判定プランを指定できます。
- スキャンのデフォルト・スケジュールは 12 時間ごとで、構成直後に開始されます。GIM パラメーター FAM_SCHEDULER_HOUR_TIME_INTERVAL、FAM_SCHEDULER_START、 FAM_SCHEDULER_REPEAT を使用して、これらの設定を変更できます。

ファイルのディスカバリーおよび分類 GIM パラメーターにある完全なパラメーター・リストを参照してください。
注: grdapi コマンド gim_update_client_params を使用して GIM パラメーターを構成することもできます。

6. Guardium レポートの「S-TAP 状況モニター」を表示することで (「マイ・ダッシュボード」からレポートを追加)、FAM ディスカバリー・エージェントが正しくインストールされたか検査します。S-TAP ホストの IP アドレスで FAM_Agent 接尾部を探します。
7. FAM バンドルをアンインストールして再インストールせずに、後でファイルの再ディスカバリーをトリガーするには、次のようにします。
a. 作業ディレクトリーの下のファイルを削除します。Guardium がデフォルト・ディレクトリーにインストールされている場合、削除対象のファイルはファイル・サーバーのディレクトリー /usr/local/IBM/modules/FAM/current/files/work にあります。
b. GIM の任意の FAM パラメーターを変更します。例えば、時間間隔を 5 分から 10 分に変更します。
c. 「選択したものに適用」をクリックしてから「インストール/更新」をクリックします。

タスクの結果

ディスカバリーおよび分類の結果: FAM ディスカバリー・エージェント (ファイル・クローラー) のインストールが完了すると、インストール中に指定した初期パスを使用してファイル・クローラーの基本実行が開始されます。クローラーは、実行を完了するたびに、「ファイル: クローラー構成」レポートに含まれる状況メッセージを送信します。このプロセスでは、フォルダーとファイル、それらの所有者、アクセス許可、サイズ、および最終更新日時のリストが収集されます。
親トピック: ファイル・サーバー内での機密データのディスカバーおよび分類
関連情報:
GuardAPI ファイル・アクティビティー・モニター関数

Windows ファイル・サーバー上のファイル・アクティビティー・モニターの無効化

ファイル・アクティビティー・モニターは、GIM クライアントを使用して無効に設定されます。

このタスクについて

この手順は、FAM ディスカバリー・エージェントのみを使用する場合でもファイル・アクティビティー・モニターを無効にするために実行する必要があります。

手順

1. 「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」にナビゲートします。
2. 「クライアントの選択」セクションで、S-TAP モジュールをアンインストールするデータベース・サーバーを選択します。表内のチェック・ボックスを使用して個々のクライアントを選択するか、「クライアント・グループを選択します」メニューを使用してクライアントのグループを選択します。「次へ」をクリックして先に進みます。
3. 「パラメーターの選択」セクションで、パラメーター WINSTAP_FAM_ENABLED = 0 を設定します。
4. 「OK」をクリックします。

親トピック: ファイル・サーバー内での機密データのディスカバーおよび分類

ファイルのディスカバリーおよび分類 GIM パラメーター

以下の GIM パラメーターを使用して、コレクターごとにファイルのモニター、ディスカバリー、および分類を構成します。

コレクターごとにファイルのモニター、ディスカバリー、および分類を構成します。 これらのパラメーターはインストール時に構成するか、後から GIM (「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」) または GuardAPI コマンド gim_update_client_params を使用して構成できます。GuardAPI を使用する場合、一度に更新できるのは 1 つのコレクターのみです。

GIM パラメーター	デフォルト	記述	GUI
FAM_CLASSIFICATION_LANGUAGES	英語	自動言語検出の場合は、GenericLanguage に設定してください。 Linux の場合は、Linux サーバーに必要な言語サポートがインストールされていることを確認してください。例えば、中国語の文書の分類をサポートするには、中国語のサポートを Linux にインストールする必要があります。 IBM Content Classification でサポートされている言語について詳しくは、http://www-01.ibm.com/support/knowledgecenter/SSBRAM_8.8.0/com.ibm.classify.workbench.doc/c_WBG_available_languages.htm%23wp9000332?lang=enを参照してください。	X
FAM_DEBUG	0	ファイル・サーバーのログが収集され、Guardium アプライアンスに送信されます。 0=OFF 1= ON	X

GIM パラメーター	デフォルト	記述	GUI
FAM_ENABLED	1	FAM ディスカバリーを有効にします。 0 = FAM ディスカバリー・エージェントは無効化されます。 1 = FAM ディスカバリー・エージェントは有効化されます。これはデフォルトです。 2 = FAM ディスカバリー・エージェントを再始動します。 FAM の再始動方法: GIM GUI の FAM_ENABLED パラメーターを 2 に変更し、「インストール/更新」をクリックしてクライアントに適用します。 ファイル・サーバーの FAM サービスは、PID を変更して、再始動したことを示します (ps -ef | grep fam)。また、事前定義の GUI レポートである「ファイル: クローラー構成」に新しい項目があります。GIM GUI で構成を 1 に戻すと、プロセスを繰り返して再始動できます。 ディスカバリー・エージェントが機能するためには、S-TAP パラメーター fam_enable を有効にする必要があります。	X
STAP_FAM_ENABLED (UNIX の場合) WSTAP_FAM_ENABLED (Windows の場合)	0	FAM モニターを有効にします。 0: 無効 1: 有効 v10.5 へのアップグレード時に、v10.1.4 以前のバージョンで guard_tap.ini パラメーター fam_enable が有効になっていた場合、このパラメーターは v10.5 以上へのアップグレード後に有効になります。	
FAM_ICM_CLASS_DECISION_PLANS		判定プランの名前と分類エンティティーを含めることで、その判定プランを有効化します。 DecisionPlanName1{Entity1.1,Entity1.2,..}:DecisionPlanName2{Entity2.1,Entity2.2,..} 判定プランごとに、エンティティーをセミコロンで区切った判定プランのリストを設定します。 形式: エンティティーは、中括弧とコロンで区切ってリストされます。 一部の判定プランで中括弧が空または欠落している場合、すべての分類エンティティーは FAM レポート/調査ダッシュボードの分類結果に示されます。 中括弧が空または欠落している例は次のとおりです: DecisionPlanName1{}:DecisionPlanName2{} DecisionPlanName1:DecisionPlanName2"~	X
FAM_ICM_CLASS_THREAD_COUNT	5	使用する分類のスレッドの数。デフォルトは 5 で、これが推奨値です。	X
FAM_ICM_URL	http://localhost:18087	IBM Content Classification Server の URL。	X
FAM_INSTALLER		インストーラー・パッケージへのパス。Windows のみ。	
FAM_INSTALL_DIR		ファイル・アクティビティー・モニター・ソフトウェアがインストールされている場所。Windows のみ。	
FAM_IS_DEEP_ANALYSIS		分類の制御 False で分類が無効化されます。メタデータおよびアクセス許可の基本スキャンのみです。 True の場合、ファイルの内容に基づいて分類が有効化されます。 判定プランが有効化されていない場合 (FAM_ICM_CLASS_DECISION_PLANS が未定義の場合)、基本スキャンのみ実行されます。	X
FAM_SCAN_EXCLUDE_DIRECTORIES	NULL	ディスカバリーおよび分類から除外するディレクトリー。 フォーマット: ディレクトリーへの絶対パス ワイルドカードはサポートされません。	X
FAM_SCAN_EXCLUDE_REMOTE_DIRECTORIES	true	ディスカバリーおよび分類から除外するリモート・ディレクトリーです。 true: リモート・ディレクトリーをスキャンしません。 false: リモート・ディレクトリーをスキャンします。 ワイルドカードはサポートされません。 Windows では、次のように設定します。 ¥¥¥¥RemoteMachine¥sharefolder¥directory	X
FAM_SCAN_EXCLUDE_EXTENSIONS	NULL	指定されたファイル拡張子、または拡張子が設定されていない文書を FAM スキャンから除外します。 Windows と Linux の両方に適用されます。 フォーマット: セミコロンで区切られたリスト 設定では大/小文字が区別されます。除外する拡張子の例: pdf;txt;doc。拡張子のない文書を除外するには、「NO_EXTENSION」に設定します。	X

GIM パラメーター	デフォルト	記述	GUI
FAM_SCAN_EXCLUDE_FILES	NULL	ディスカバリーおよび分類から除外するファイル。 フォーマット: 有効なファイル名。 ワイルドカードはサポートされません。	X
FAM_SCAN_MAX_DEPTH		指定された開始ディレクトリー (FAM_SOURCE_DIRECTORIES) に対するスキャンの深さを制限します。	X
FAM_SCHEDULER_HOUR_TIME_INTERVAL	12	ディスカバリーおよび分類のスキャンが実行される頻度 (時間単位)。 フォーマット: 整数 デフォルトは 12 時間です。	X
FAM_SCHEDULER_MINUTE_TIME_INTERVAL		これはスキャンの分単位の間隔で、時間単位の間隔と共に使用します。例えば、スキャンを 12 時間 30 分おきに実行する場合、時間として 12 を指定し、分としてここで 30 を指定します。 フォーマット: 整数	X
FAM_SCHEDULER_REPEAT		True = 指定された時間間隔でディスカバリー・プロセスを繰り返します。 False = スキャンを繰り返しません。	X
FAM_SCHEDULER_START_TIME	NULL	ディスカバリー・プロセスおよび分類プロセスの最初のアクティブ化時刻。 フォーマット: MM-DD-YYYY HH:mm 例えば、01-02-2016 18:00 と入力すると、スキャンは 2016 年 1 月 2 日の午後 6 時に開始されます。時間間隔が 12 時間の場合、プロセスは毎日午後 6 時と午前 6 時に実行されます。	X
FAM_SERVER_PORT	16022	Guardium コレクターのポート (16022)。	X
FAM_SOURCE_DIRECTORIES	NULL	スキャンを開始する対象の 1 つ以上のディレクトリー。 ワイルドカードはサポートされていません。例: /home/test。 形式: セミコロンで区切られた FAM ソース・ディレクトリーのリストを設定します。 例: %IBM_FAM_HOME%/test/dir1;%IBM_FAM_HOME%/test/dir2 ~ FILE_SYSTEM_ROOTS を使用して、サーバー内のすべてのファイルをスキャンします。特に大量のファイルがサーバーに含まれている場合は、お勧めできません。	X

親トピック: ファイル・サーバー内での機密データのディスカバーおよび分類
関連情報:
GIM - GUI
GIM - CLI

FAM 判定プランのカスタマイズ

判定プランは、ファイル内の機密の内容を識別するために使用されます。Guardium FAM ディスカバリー・エージェントは、GDPR、HIPAA、PCI、SOX、およびソース・コードのデフォルトの判定プランを提供します。デフォルトの判定プランを使用して、結果のレポート/調査ダッシュボードから分類エンティティーを変更できます。また、IBM コンテンツ分類ワークベンチを使用して、新しいプランを作成したり、既存のプランを変更したりできます。

始める前に

Guardium 環境に接続できる Windows ワークステーションに IBM Content Classification 8.8 をインストールします。

ファイル・アクティビティー・モニター中、GIM インストール・ユーザーは、ファイル・アクティビティー・モニターの GIM 構成ページで ICM 判定プラン設定を構成する必要があります。

ユーザーは、判定プラン (カテゴリー) と各判定プランのエンティティー (NVP フィールド) のリストをコロンで区切って構成する必要があります。

この構成は、ファイル・アクティビティー・モニターによる内容分類で使用されます。

ファイル・アクティビティー・モニターのインストール中に使用できる、各判定プラン・テンプレートの使用可能なすべてのエンティティーを構成してください。

判定プランの分類は、ファイルが機密ファイルであり、分類が空ではない場合のみ表示されます。

以下に、ファイル・アクティビティー・モニターに付属のすぐに使用可能な各判定プランのエンティティーのリストを示します。判定プランは、GIM で構成できます。

GDPR
: 年齢、誕生日、性別、性的嗜好、政治的意見、E メール・アドレス、氏名、宗教、宗教的意見、国際パスポート、位置、遺伝、犯罪歴、生体、写真、住所、都市、郵便番号、国

HIPAA
: SSN、Name、License、GovermentID、PassportContext、BankAccount、Address、IPAddress、EmailAddress、URL、Phone、CreditCard、possibleHealthPlan、Confidential_match、HIPAA_match

PCI
: SSN、Name、License、GovermentID、PassportContext、BankAccount、Address、IPAddress、EmailAddress、URL、Phone、BankAccountContext、CreditCard、CreditContext、containCardIssuer、PCI_match、Confidential

SOX

SSN、Name、License、GovermentID、PassportContext、BankAccount、Address、IPAddress、EmailAddress、URL、Phone、BankAccountContext、CreditCard、CreditContext、containCardIssuer、piiMatch、Confidential、SOXContext、SOX_match

ソース

containDate、hasSSN、hasBirthDate、containCardIssuer、hasCreditCard、PCIViolation、HIPAA_Match、ConfidentialMatch、Source_match

「ソース」判定プランは、「ソース」判定プランが構成されるとデフォルトでロードされる 2 つの知識ベース (CodeKB および DocumentTypeKB) を参照します。

判定プランは、IBM Classification モジュールがコンテンツ項目を分類する方法を決定するためにユーザーが構成する、一連のルールです。ルールはトリガーとアクションで構成されます。トリガーは、アクションを開始するために満たす必要のある条件を決定します。アクションは、文書の分類方法を決定します。判定プランは、ルール (キーワード・ベースの分類と統計を使用したテキスト・ベースの分類) を組み合わせるために、1 つ以上の知識ベースを参照することもできます。

知識ベースは、コンテンツ項目の分析と分類に使用される、一連の収集データです。知識ベースは、システムが処理することを期待されるデータの種類を反映します。知識ベースでテキストを分析できるようにするには、適切にカテゴリー分けされた十分な数のコンテンツ項目例によって、知識ベースが調整されている必要があります。調整された知識ベースは、項目の関連性を示す数値的尺度から各カテゴリーを割り出すことができます。

注: 中国語の名前を持つ判定プランの場合、ICM は機能しません。中国語のコンテンツ・ドキュメントと、中国語の判定プラン・ルールはサポートされますが、中国語の名前が指定された判定プランはサポートされていません。
注: 中央マネージャーから管理対象ユニットへの判定プランの配布は、サポートされていません。
注: 各判定プランの分類結果は、適切に構成されて認識されたエンティティーごとに指定する必要があります。分類は、ファイルが機密ファイルであり、分類が空ではない場合のみ表示されます。デバッグ・レベルでは、ICM エラーおよび判定プランの失敗に関する文書があります。

このタスクについて

このことを説明するために、会社に「ProjectA」という名前の機密プロジェクトがあると仮定します。この文字列を含むすべてのファイルを識別およびモニターする必要があります。

手順

1. Windows の「スタート」メニューを使用して IBM Content Classification 8.8 Classification Workbench を開きます。
2. 「プロジェクトを開く (Open Project)」ダイアログで「新規...」をクリックします。
3. 「新規プロジェクト (New Project)」ダイアログで、プロジェクト・タイプに対応する判定プランを選択します。この判定プランの名前 (`ProjectA_DP` など) を入力します。必要に応じて説明を入力します。
4. 「新規プロジェクトのオプション (New Project Options)」ダイアログで「空のプロジェクトを作成 (Create an empty project)」を選択します。
5. 「プロジェクト・エクスプローラー」で「語および文字列のリスト・ファイル (Word and string list files)」をクリックします。「語および文字列のリスト・ファイル (Word and string list files)」ダイアログで、「新規...」をクリックして新規ファイルを作成します。「新規ファイル (New File)」ダイアログで、ファイル・タイプとして「語のリスト (Word list)」を選択し、ファイルの名前を選択します。この例では、ファイルに `Names` という名前を付けます。 Wordlist_Names.txt がファイルのリストに表示されます。
6. ファイル名をダブルクリックし、ファイルを編集します。`~ProjectA~` という文字列を含む単一の行を挿入し、ファイルを保存します。
7. 「プロジェクト・エクスプローラー」で、「判定プラン」 > 「新しいグループ」 > 「新規ルール (New Rule)」をクリックします。ルールの名前を `ProjectA` に変更します。
8. 「新規ルール (New Rule)」ダイアログで、「トリガー」タブを開きます。「条件」をクリックします。
9. 「フィールドに特定の語または句が含まれている場合にトリガー (Trigger when fields contains specific words or phrases)」を選択します。「語のリスト・ファイル (Word list file)」を選択します。「OK」をクリックします。
10. 「アクション」タブを開きます。「新規ルールの追加」をクリックします。
11. 「アクション・タイプ」リストから「拡張アクション」を選択します。「内容フィールドの設定 (Set content field)」アクションを選択します。 指定のトリガーが起動すると、この内容フィールドが作成されます。この内容フィールドは FAM レポートで確認できます。
12. 「アクションの追加」ダイアログで、内容フィールド名として `ProjectA_match` と入力し、「値」フィールドに `found` と入力します。
13. 内容セットを判定プラン・プロジェクトにインポートします。
    a. 「ProjectA」という文字列を含むテキスト文書を作成します。
    b. 「プロジェクト・エクスプローラー」で、ProjectA_DP プロジェクトを展開します。「内容セット (Content Set)」を右クリックし、「内容セットのインポート (Import Content Set)」を選択します。
    c. 「ファイル・システム・フォルダー内のファイル (Files from a file system folder)」をクリックします。 ステップ a で作成したファイルを参照します。「次へ」をクリックし、その後「次へ」、「次へ」、「完了」を順番にクリックします。
14. 定義が成功したことを検査します。
    a. 「プロジェクト・エクスプローラー」で、「内容セット (Content Set)」タブを開きます。ファイルを右クリックし、「判定プランを通じて項目を実行 (Run Item through Decision Plan)」を選択します。
    b. 「分析済み項目 (Analyzed item)」ダイアログで、判定プランおよびグループを展開します。Rule:ProjectA が [Triggered] とマークされていることを確認します。
    c. 「内容フィールド... (Content Fields...)」をクリックします。「内容フィールドの選択 (Select Content Fields)」ダイアログで、「ProjectA_match」が「変更済みフィールド (Changed fields)」ボックスに表示されていることを確認し、また「found」が「内容」ボックスに表示されていることを確認します。
15. 「プロジェクト・エクスプローラー」で、「プロジェクト」 > 「保存」をクリックして ProjectA_DP プロジェクトを保存します。
16. 「プロジェクト・エクスプローラー」で、「プロジェクト」 > 「エクスポート」をクリックして ProjectA_DP プロジェクトを dpn ファイルにエクスポートします。
17. GIM を使用して、判定プランを使用するファイル・サーバーに dpn ファイルをプッシュします。

親トピック: ファイル・サーバー内での機密データのディスカバーおよび分類

GDPR ファイル・アクティビティーのルール

FAM GDPR の判定プランを作成するには、下記のルールを使用してください。

住所
Age
BankAccount
BelganID
Canada_SIN
CC_Amex

CC_Diners_Club
CC_Discover_Club
CC_InstaPayment
CC_JCB
CC_Laser
CC_Maestro
CC_MasterCard
CC_Switch
CC_Visa
Confidential_match
CreditCard
DateOfBirth
DNI
EmailAddress
EmailAll
EnglishGDPR
EULA
Firearm
FirearmModels
French_INSEE
FrenchAddress
FrenchBanking
FrenchCreditCard
FrenchDOB
FrenchDriverLicense
FrenchGDPR
FrenchID
FrenchIP
FrenchLicensePlate
FrenchMedical
FrenchPassport
FrenchPhone
GDPR_match
GenericMedical
GermanAddress
GermanBanking
GermanCreditCard
GermanDOB
GermanDriverLicense
GermanEmail
GermanGDPR
GermanIP
GermanLicensePlate
GermanMedical
GermanNatID
GermanPassport
GermanPhone
GermanSSN
GermanTaxID
GermanVAT
HumanAttruibutes
InternetEmail
IPAddress
ITARAircraft
ITARAircraftModels
ITARExplosives
ITARExplosivesType
ItTaxCode
LegalDocuments
使用許諾条件
MedicaDiseases
MedicalRecords
名前
nameMatch
NI
NIE
NIN
NINO_UK
PassportApp_Can
PassportApp_USA
PESEL
Phone
Photo
PolandID
PolandNatID
PrescriptionDrugs
PrescriptionDrugsPrimary
PrescriptionDrugsSecondary
Religion

Sex
SexualOrientation
SocialNetwork
SpainAddress
SpainBanking
SpainCreditCard
SpainDOB
SpainDriverLicense
SpainEmail
SpainGDPR_ST
SpainIP
SpainLicensePlate
SpainMedical
SpainNatID
SpainPassport
SpainPhone
SpainSSN
SpainTaxID
SpainVAT
SSN
TaxEIN_SSN_Summary
TaxEIN_Summary
TaxSSN_Summary
UK_NHS
UKLicense
UKPhoneNum
URL
US_SSN
USPhoneNum
VisaApplication

親トピック: ファイル・サーバー内での機密データのディスカバーおよび分類

FAM 判定プラン・ファイルのアップロードおよび削除

判定プランを作成した後、そのプランをファイル・サーバー上のファイルの分類のために Guardium システムにアップロードします。

このタスクについて

手順

1. 「設定」 > 「ツールとビュー」 > 「判定プラン・ファイルのアップロード」にナビゲートします。
2. .dpn または .kb 判定プラン・ファイルを参照して選択し、「アップロード」をクリックします。
3. ファイルを削除するには、そのファイルを選択して、「削除」をクリックします。

親トピック: ファイル・サーバー内での機密データのディスカバーおよび分類

NAS および SharePoint のディスカバリーと分類

File Discovery, Entitlement and Classification (FDEC) for NAS and SharePoint servers により、規制法 (GDPR、HIPAA など) に関連している可能性がある機密データのファイル・ライセンスおよび分類をスキャンによって調べることができます。

NAS (Network Attached Storage) は、複数のストレージ・デバイスが含まれたネットワーク・アプライアンスに基づくファイル・レベルのストレージ・システムです。SharePoint は、Web ベースのコラボレーション・プラットフォームであり、文書管理およびストレージ・システムでもあります。

この Guardium 製品には、NAS または SharePoint のスケジュール・スキャンを実行する Windows ベースのサービスと、スキャンのターゲット、スケジュール、および分類基準を構成するための構成アプリケーションが含まれています。このクライアント・ソフトウェアは「S-TAP 制御」で確認できます。スキャン結果は「ファイル・ライセンス」レポートに表示されます。

- サポートされるプラットフォーム
- スキャンのアクセス許可
  File Discovery, Entitlement and Classification (FDEC) に対する NAS および SharePoint のアクセス許可
- クライアント・ソフトウェアのインストール
  NAS または SharePoint の環境用に File Discovery, Entitlement and Classification (FDEC) をインストールします。
- 構成
  インストール・ディレクトリーの Bin¥ にある **ConfigureNasScan.exe** を使用して、「スキャン構成ユーティリティー (Scan Configuration Utility)」を起動します。この構成ユーティリティーの実行可能ファイルは、NAS と SharePoint の両方の環境で同じです。
- スキャン結果の表示
  File Discovery, Entitlement and Classification (FDEC) のスキャン結果を表示します。
- ユーザー定義の基準の作成
  「基準エディター (Criteria Editor)」は、ユーザーが基準を最初から作成するか、事前定義の基準項目をコピーして編集することで基準を作成するために使用できるユーティリティーです。

親トピック: ディスカバー

サポートされるプラットフォーム

インストールでサポートされるプラットフォーム

FDEC for NAS and SharePoint は、以下の Windows サーバーへのインストールがサポートされています。

→ Windows 2016

→ Windows 2012 R2

→ Windows 2012

→ Windows 2008 R2

サポートされる SharePoint のバージョン

FDEC for NAS and SharePoint は、以下の SharePoint のバージョンをターゲットとしたスキャンに対応しています。

- SharePoint® 2016
- SharePoint® 2013
- SharePoint® 2010

サポートされる Network Attached Storage デバイス

FDEC for NAS and SharePoint は、以下の Network Attached Storage (NAS) デバイスをターゲットとしたスキャンに対応しています。

- Hitachi® 11.2 以上
- NetApp® Data ONTAP®:
  - Cluster-Mode 8.2 以上
  - 7-Mode 7.2 以上
- EMC® VNX®:
  - VNX® 8.1
  - VNX® 7.1
- EMC® Isilon® 7.0 以上
- EMC® Celerra® 6.0 以上
- Dell EMC Unity™

親トピック: NAS および SharePoint のディスカバリーと分類

スキャンのアクセス許可

File Discovery, Entitlement and Classification (FDEC) に対する NAS および SharePoint のアクセス許可

SharePoint スキャンのアクセス許可

SharePoint エージェントは、SharePoint のサーバー上で、アクセス許可と内容の監査、またはアクセスの監査 (SPAA) および機密データ・ディスカバリーの監査を行うことができます。このエージェントは、*サーバーの全体管理*コンポーネントをホストするアプリケーション・サーバー上にインストールされます。

組織でサービス・アカウントの制限付きのプロビジョニングが不要である場合、SharePoint エージェント・ベースのスキャンを正常に実行するには以下のアクセス許可で十分です。

- SharePoint エージェントがインストールされているサーバー上のローカル管理者グループ・メンバーシップ
- スキャン対象のすべてのサイト・コレクション上のサイト・コレクション管理者
- SharePoint のバージョンに応じて、目的の構成データベースおよびすべてのコンテンツ・データベースに対して DB_Owner または SPDataAccess が適用されている必要があります。
  - SharePoint 2013 および 2016 の場合: SharePoint コンテンツ・データベースおよびすべての構成データベースに対する SPDataAccess
  - SharePoint 2010 の場合: SharePoint コンテンツ・データベースおよびすべての構成データベースに対する DB_Owner

SharePoint スキャンのアクセス許可: 低い特権モデル

制限付きのアクセス許可が必要な組織の場合、SharePoint エージェント・ベースのスキャンを正常に実行するには、サービス・アカウントに以下のアクセス許可が必要です。

SharePoint エージェントのインストールの前提条件として、インストール時に指定するサービス・アカウント (このアカウントは後でターゲットの SharePoint 環境に対してアクセスの監査 (SPAA) または機密データ・ディスカバリーの監査 (あるいはその両方) のスキャンを実行するために使用される) には以下のアクセス許可が必要です。

- ローカル・セキュリティー・ポリシーでの「サービスとしてのログオン」
- IIS_IUSRS に対するローカル・グループ・メンバーシップ
- パフォーマンス・ログ・ユーザー (機密データ・ディスカバリーのみ)

SharePoint エージェントのインストール後、アクセスの監査 (SPAA) または機密データ・ディスカバリーの監査 (あるいはその両方) のスキャンを実行するために、このサービス・アカウントには以下のアクセス許可が追加で必要になります。

- スキャン対象のすべてのサイト・コレクション上のサイト・コレクション管理者
- SharePoint エージェントがインストールされているサーバー上のローカルの「Users」グループ・メンバーシップ

スキャンに Web アプリケーション・スコープが含まれる場合は、以下のアクセス許可も必要です (ファーム全体のスキャンを実行する場合は、この手順をスキップできます)。

- バックアップ・オペレーターに対するローカル・グループ・メンバーシップ
- WSS_WPG に対するローカル・グループ・メンバーシップ
- SharePoint 構成データベース上の WSS_CONTENT_APPLICATION_POOLS

FDEC SharePoint エージェントがインストールされた後、サービス・アカウントに以下のアクセス許可があることを確認してください。

- エージェントのインストール・ディレクトリー (例えば、C:¥Program Files¥IBM¥FDECforSP) に対するフル・コントロール

FDEC SharePoint エージェントは、Microsoft API を使用します。Microsoft API では、すべてのデータを収集するために、以下のアクセス許可を持つアカウントが必要です。

- SharePoint コンテンツ・データベース上の WSS_CONTENT_APPLICATION_POOLS
- SharePoint 構成データベース上の WSS_CONTENT_APPLICATION_POOLS

スキャンに Web アプリケーション・スコープが含まれる場合は、上記の最後のアクセス許可は既に付与されています。

低い特権モデルのプロビジョニングについて詳しくは、SharePoint エージェントのインストール・ガイドの許可オプションのセクションを参照してください。

NAS スキャンのアクセス許可

NetApp Data ONTAP Cluster-Mode のアクセス許可

NetApp Data ONTAP Cluster-Mode デバイスからファイル・システム・データを収集するために使用される資格情報は、以下の能力を備えている必要があります。

- 特定の API 呼び出しを実行することにより、共有を列挙する
- NTFS セキュリティーをバイパスして、スキャン対象のフォルダー構造全体を読み取り、ファイル/フォルダーのアクセス許可を収集する

共有の列挙 – API 呼び出し (Cluster-Mode)

NetApp Data ONTAP Cluster-Mode デバイス上の共有を列挙するには、ファイル・システム・スキャンには、少なくとも以下の CLI コマンドで提供される資格情報が必要です。

CLI コマンド	アクセス権限
version	読み取り専用
volume	読み取り専用
vserver	読み取り専用
server fpolicy	読み取り専用
security login role show-ontapi	読み取り専用

重要: NetApp Data ONTAP Cluster-Mode デバイス v8.3 以上に存在する共有を列挙するには、資格情報に、少なくともターゲット・ホストに対する Power Users グループのグループ・メンバーシップのアクセス許可が必要です。

NTFS セキュリティーのバイパス (Cluster-Mode)

ONTAP_Admin$ という特別な共有に対するアクセスを提供することにより、NetApp Data ONTAP Cluster-Mode デバイス上で資格情報が NTFS セキュリティーをバイパスできるようになります。ONTAP_Admin$ 共有にアクセスするには、資格情報がターゲット・デバイス上の FPolicy に関連付けられている必要があります。

FPolicy は、「空の」FPolicy にすることができ、組織のシステムに対して最小限の影響しか与えません。 このポリシー名は「StealthAUDIT」でなければなりません。

NetApp Data ONTAP 7-Mode のアクセス許可

NetApp Data ONTAP 7-Mode デバイスからファイル・システム・データを収集するために使用される資格情報は、以下の能力を備えている必要があります。

- 特定の API 呼び出しを実行することにより、共有を列挙する
- NTFS セキュリティーをバイパスして、スキャン対象のフォルダー構造全体を読み取り、ファイル/フォルダーのアクセス許可を収集する

以下のセクションでは、これらのターゲット・ホストに割り当てられた接続プロファイル内で使用される資格情報に付与される必要があるアクセス許可の概要を説明します。

共有の列挙 – API 呼び出し (7-Mode)

NetApp Data ONTAP 7-Mode デバイス上の共有を列挙するには、ファイル・システム・スキャンでは、少なくとも以下の API 呼び出しに対するアクセスで提供される資格情報が必要です。

- login-http-admin
- api-system-api-list
- api-system-get-version
- api-cifs-share-list-iter-*

NTFS セキュリティーのバイパス (7-Mode)

NTFS をバイパスするには、資格情報に、少なくともターゲット・ホストに対する以下のアクセス許可が必要です。

- 以下の両方のグループのグループ・メンバーシップ:
  - Power Users
  - Backup Operators

注: すべての NetApp グループには RID が割り当てられています。Power Users や Backup Operators などの組み込みの NetApp グループには特定の RID 値が割り当てられています。7-Mode NetApp デバイスでは、グループに対するシステム・アクセス検査は、グループのロールではなく、グループに割り当てられた RID で識別されます。そのため、Power Users および Backup Operators のグループでアクセス検査をバイパスできることは、power のロールとも backup のロールとも関係ありません。いずれのロールも必要ではありません。例えば、組み込みの Power Users グループは、すべてのロールが取り除かれている場合でも、組み込みではない他のグループよりも高いファイル・システム・アクセス機能を持っています。

EMC Celerra、VNX、VNXe、VMAX3、または Unity のアクセス許可

EMC Celerra、VNX、VNXe、VMAX3、または Unity の各デバイスからファイル・システム・データを収集するために使用される資格情報には、少なくともターゲット・ホストに対する以下のアクセス許可が必要です。

- 以下の両方のグループのグループ・メンバーシップ:
  - Power Users
  - Backup Operators

これらのアクセス許可により、資格情報で、共有の列挙、リモート・レジストリーへのアクセス、フォルダーでの NTFS セキュリティーのバイパスが可能になります。

資格情報でのアクセスを拒否されるフォルダーがある場合は、Backup Operators グループに「ファイルとディレクトリのバックアップ」権限がないことが考えられます。その場合は、それらのグループに「ファイルとディレクトリのバックアップ」権限を追加で割り当てるか、Windows サーバーから「コンピューターの管理」を使用して新規ローカル・グループを作成する必要があります。その後、EMC のお客様が使用できる CelerraManagementTool.msc プラグインを使用して権限を割り当てます。

注: Windows サーバー 2012 以降から EMC デバイスのスキャンを正常に実行するには、そのサーバー上で「Require Secure Negotiate」ポリシーをオフにする必要があります。これは、Windows Server 2012 および Windows 8 の SMB 3.0 に追加された「Secure Negotiate」機能に起因する問題があるためです。この機能は、プロトコル・バージョン 2.0 および 2.1 のみをサポートするサーバーも含め、すべての SMBv2 サーバーによるエラー応答の正確な署名に依存しています。一部のサード・パーティー・ファイル・サーバーは署名付きのエラー応答を返さないため、接続が失敗します。

EMC Isilon のアクセス許可

EMC Isilon デバイスからファイル・システム・データを収集するために使用される資格情報には、ターゲット・ホストに対する以下のアクセス許可が必要です。

- ローカル管理者グループのグループ・メンバーシップ (LOCAL:System Provider)
- 実際のファイル・ツリーまたは IFS ルート共有に対する権限
  - 共有のアクセス許可
    - 読み取りアクセス
  - フォルダーのアクセス許可
    - フォルダーの一覧/データの読み取り
    - フォルダーのスキャン/ファイルの実行
    - アクセス許可の読み取り

これらのアクセス許可により、資格情報でフォルダーおよび共有を監査できるようになります。範囲指定した分類スキャンを実行するには、資格情報に対して、スキャン対象の共有が置かれている各アクセス・ゾーンでの LOCAL:System provider が選択されていることも必要です。

Hitachi のアクセス許可

Hitachi デバイスからファイル・システム・データを収集するために使用される資格情報には、ターゲット・ホストに対する以下のアクセス許可が必要です。

- 以下の両方のローカル・グループのグループ・メンバーシップ:
  - Local Administrators
  - Backup Administrators

このアクセス許可により、資格情報に、すべてのターゲット・フォルダーおよびファイルに対する読み取りアクセスが付与されます。

親トピック: NAS および SharePoint のディスカバリーと分類

クライアント・ソフトウェアのインストール

NAS または SharePoint の環境用に File Discovery, Entitlement and Classification (FDEC) をインストールします。

始める前に

- ユーザーには、サーバー・サイドと NAS および SharePoint の環境の両方で一致している管理特権が必要です。先に進む前に、スキャンのアクセス許可でアクセス権のガイドラインを確認してください。
- プラットフォームの前提条件およびサポートされている NAS デバイスの詳細については、サポートされるプラットフォームを参照してください。

手順

1. 環境として NAS を使用するのか SharePoint を使用するのかを決定します。ユーザーが NAS デバイスをスキャンするためにクライアントをインストールする場合は、ネットワーク経由で NAS デバイスにアクセスできる Windows サーバーにクライアントをインストールします。一方、ユーザーが SharePoint をスキャンするためにクライアントをインストールする場合は、SharePoint サーバーまたは SharePoint サーバー・ファームに直接インストールします。
   注: このサーバー上に他の Guardium 製品があってはなりません。
2. FDEC for NAS and SharePoint パッケージを Fix Central からサーバーにダウンロードして、このファイルを unzip します。
3. FDEC パッケージのインストーラー・ディレクトリーにナビゲートして、インストーラー・ディレクトリー内の実行可能ファイル setup.exe を実行します。
4. ウィザードのプロンプトに従って、インストールを完了します。スキャンする共有に対する管理特権を持つサービス・ユーザーを指定します。
   注:

   FDEC for NAS のデフォルトのインストール・ディレクトリーは、C:¥Program Files¥IBM¥FDECforNAS です。

   FDEC for SharePoint のデフォルトのインストール・ディレクトリーは、C:¥Program Files¥IBM¥FDECforSP です。

親トピック: NAS および SharePoint のディスカバリーと分類

構成

インストール・ディレクトリーの Bin¥ にある **ConfigureNasScan.exe** を使用して、「スキャン構成ユーティリティー (Scan Configuration Utility)」を起動します。この構成ユーティリティーの実行可能ファイルは、NAS と SharePoint の両方の環境で同じです。

スキャン名
: スキャンの名前を選択します。

Guardium アプライアンス
: Guardium アプライアンスのホスト名または IP アドレスを入力します。
  注: アプライアンスのアドレスは、すべてのスキャンに影響を与えます。

スキャン・ホスト (Scan Host)
: NAS の場合、これは NAS 環境の IP またはホスト名です。SharePoint の場合、「スキャン・ホスト (Scan Host)」には `localhost` が自動入力されます。

スキャン・パス (Scan Paths)
スキャン・パスの例を以下に示します。

- NAS の場合: NameOfShareDrive
- SharePoint の場合: http://SharePointServer/my/test

スキャンを次の間隔ごとに実行 (Scan Every)
スキャンのスケジュールは、時間単位または日単位で構成できます。デフォルトでは、スキャンの頻度は 12 時間ごとです。
最大スキャン・レベル (Max Scan Level)
これは、ディレクトリー構造内のスキャン対象レベルの数です。デフォルトは 100 です。
スキャン・オプション (Scan Options)

- コンテナーのみ (Containers Only): ディレクトリーのみをスキャンして、オブジェクト自体の分類はトリガーしません。
- すべてのオブジェクト (All Objects): ファイルおよびディレクトリー・ツリーを含むすべてをスキャンし、基準を突き合わせます。
- 一致のみ (Matches Only): このスキャンでは、基準をトリガーするレコードのみが返されます。

スキャン基準の編集 (Edit Scan Criteria)
これを使用すると、さまざまな基準セットのアップロードが可能になり、ファイルの分類に使用する特定の基準を選択できます。一例として、GDPR.update ファイルから GDPR 機密データ・パターンを選択できます。HIPAA などの他のコンプライアンス・ガイドラインに沿った基準の場合は、patterndefs.update を選択します。
注: ユーザー定義の基準を作成する場合は、ユーザー定義の基準の作成を参照してください。
アクティブ
このトグルを使用すると、スキャンをアクティブ・モードまたは無効モードに設定できます。複数のスキャンをアクティブにできますが、同時にスケジュールされているスキャンは連続的に実行されます。
スキャン (Scan)、保存、実行、削除
新規スキャンの実行、保存と実行、スキャンの削除、または保存を行うには、これらのボタンを使用します。
スキャン状況 (Scan Status)
「最終スキャンの開始 (Started Last Scan)」、「最終スキャンの完了 (Finished Last Scan)」、「スキャン済みオブジェクト (Scanned Objects)」、「新規オブジェクトの更新またはオブジェクトの更新 (Updated New or Updated Objects)」、「オブジェクトの削除または名前変更 (Deleted or Renamed Objects)」など、スキャンの状況が表示されます。

注:

- スキャンが完了するまで、サービスを停止しないことをお勧めします。停止すると、スキャン結果が壊れる可能性があるためです。
- ファイル分類のデフォルトの最大サイズは 2 MB です。

親トピック: NAS および SharePoint のディスカバリーと分類

スキャン結果の表示

File Discovery, Entitlement and Classification (FDEC) のスキャン結果を表示します。

このタスクについて

スキャンを実行する前に、エージェントがアプライアンスに接続されているか確認してください。スキャン結果は、「マイ・ダッシュボード」を使用して「ファイル・ライセンス」レポートおよび「ディレクトリー資格」レポートで表示できます。

手順

1. 「マイ・ダッシュボード」 > 「新規ダッシュボードの作成」をクリックして、新規ダッシュボードを開きます。
2. 「レポートの追加」をクリックすると、使用可能なレポートのリストが表示されます。「レポートの追加」ダイアログには、指定された基準を満たすすべてのレポートのリストが表示されます。レポートのリストを参照することも、「フィルター」フィールドに文字列を入力することもできます。文字を入力するにつれて、レポートのリストが更新されます。
3. スキャン結果は、「ファイル・ライセンス」レポートおよび「ディレクトリー資格」レポートで表示できます。レポートをクリックして、ダッシュボードに追加します。

タスクの結果

これで、「ファイル・ライセンス」レポートおよび「ディレクトリー資格」レポートに簡単にアクセスできるダッシュボードが用意できました。
親トピック: NAS および SharePoint のディスカバリーと分類

ユーザー定義の基準の作成

「基準エディター (Criteria Editor)」は、ユーザーが基準を最初から作成するか、事前定義の基準項目をコピーして編集することで基準を作成するために使用できるユーティリティーです。

基準エディターは、インストール・ディレクトリーの Bin¥SensitiveData の下にある DLPCriteriaEditor.exe から起動できます。

新しい基準を最初から作成するには、「+」ボタンをクリックして、基準のタイプを選択します。 選択した基準に必要な情報を指定します。

事前定義の基準を変更するには、目的の「システム基準 (System Criteria)」リストを右クリックして、「コピー」を選択します。選択した基準リストのコピーが、「マイ基準 (My Criteria)」の下に表示され、名前に *コピー* という語が付加されます。このコピーを必要に応じて編集できます。

選択できる基準には、以下の 3 つのタイプがあります。

- 正規表現基準
- キーワード基準

- サマリー基準

3 つの基準タイプのすべてに、その一部として次の 2 つの項目が含まれます。

1. 「タイトル」。「マイ基準 (My Criteria)」の下に表示され、固有かつ記述的にする必要があります。
2. 「サマリーでのみ使用する (Only for use in Summaries)」チェック・ボックス。選択すると、その基準はサマリー基準の一部としてのみ使用されます。

正規表現基準
: 正規表現基準は、テキストのストリングを突き合わせるための簡潔かつ柔軟な手段を提供する一連のパターン・マッチング・ルールです。この基準タイプを使用して、一連の数字が潜在的に有効であることを検証できます (クレジットカード番号など)。

  - 「大/小文字を区別する」チェック・ボックスを選択すると、スキャンで大/小文字が区別されます。
  - 「検証を使用する (Use validation)」チェック・ボックスでは、Luhn または Mod 11 によって検証できます。
  - 「分析する (Analyze)」チェック・ボックスでは、ファイル名、ファイル・メタデータ、およびファイル内容を分析するための基準を作成できます。
  - テキスト・ボックスには式を入力します。
  - 「数値スライダー (Number Slider)」は、ファイルが潜在的に機密性が高いファイルとして分類されるために必要な、リスト内の一致の数を示します。

キーワード基準
: キーワード基準は、コンマ区切りの単語のリストで構成されます。このリスト内のいずれかの単語がファイルで検出されると、その単語はヒットと見なされます。

  - 「大/小文字を区別する」チェック・ボックスにチェック・マークを付けると、スキャンで大/小文字が区別されます。
  - 「分析する (Analyze)」チェック・ボックスでは、ファイル名、ファイル・メタデータ、およびファイル内容を分析するための基準を作成できます。
  - テキスト・ボックスにはキーワードを入力します。
  - 「A..z」ボタンを使用すると、キーワードがアルファベット順でソートされ、重複が自動的に削除されます。「一意 (Distinct)」チェック・ボックスにチェック・マークを付けると、スキャンでは基準の一意のヒットが検索されます。
  - 「数値スライダー (Number Slider)」は、ファイルが潜在的に機密性が高いファイルとして分類されるために必要な、リスト内のキーワードの数を示します。「一意 (Distinct)」にチェック・マークが付けられている場合、これは、必要な一意のキーワードの数を示します。

サマリー基準
: サマリー基準は、レポート作成の目的で正規表現基準とキーワード基準を結合する手段として設計されています。これらの結果は、選択された比較演算子 (「数値スライダー (Number Slider)」の値を使用する「いずれか (Any of)」、「すべて (All of)」、または「少なくとも (At least)」) に基づいて計算されます。

  - 「マイ基準 (My Criteria)」または「システム基準 (System Criteria)」、あるいはその両方から、目的の基準にチェック・マークを付けます。
  - *サマリー照会* ステートメントが適切な比較演算子を使用して下部に作成されます。

  目的の比較演算子のラジオ・ボタンを選択します。「少なくとも (At least)」が選択されている場合、基準の一致の最小数を設定するために「数値スライダー (Number Slider)」が表示されます。

親トピック: NAS および SharePoint のディスカバリーと分類

資格最適化

資格最適化は、ジョブを効率的に実行するために必要な資格をユーザーに提供する上でのデータベース管理者のロールと、システムの脆弱性を防ぐために資格をできる限り正確に、かつ可能な限り最小限に抑える上でのセキュリティーのロールの間を仲介するものです。

システムの日常の管理には、脆弱性をもたらす状況が必然的に発生します。例えば、以下のようなものがあります。

- アクセスが一般化し過ぎている
- ユーザーに与えられた特権は一回限りの使用に必要だったが、その後に除去されなかった
- ユーザーと表の経時的変化により休止ユーザーや休止表が発生する
- あるユーザーから別のユーザーに特権が渡される

資格は、絶えず継続的に注視する必要があります。例えば、Advanced Persistent Threat (APT) 攻撃は、通常、こうした背後の入口のいずれかからシステムに侵入することで発生します。

資格最適化は、ユーザーの特権とアクションを絶えず分析し、ユーザー・アクセスを必要最小限に抑えることを目的とした固有のアクションを特定する推奨事項を作成します。分析はすべてシステムによって実行されます。管理者は結果を検討し、各ケースを調べ、適切なアクション (例えば、データベース・ユーザーからの特権の除去や休止ロールの削除など) を実行します。

また、過去 1 週間にわたる資格の変更、ユーザーとロールの完全なリスト、データ・ソース特権と実際の使用法、および特定のユーザーとロールの組み合わせのシミュレートされた理由を調査することもできます。これらのビューは推奨に関連する情報を提供します。また、他の調査の開始点ともなります。

Guardium レポートに対する資格最適化の利点は、すべてのデータベース・タイプの情報 (複数の Guardium レポートに表示される) を統合し、それ自体の包括的な統合レポートに新しい分析を追加し、資格管理を簡素化することにより、システム・セキュリティーを向上させる点です。

資格最適化は、データベース・タイプとして Microsoft SQL Server および Oracle をサポートします。SQL Contained Database はサポートしません。(Guardium レポートはデータベース・タイプ別になっています。)

資格最適化アクティビティー・モニターは、現在 Guardium によってモニターされているデータに制限されます。推奨、資格の参照、および仮定の分析の正確性は、モニター対象のデータの関連性に依存します。このツールの能力を最大限に高めるには、 userScope パラメーターおよび objectScope パラメーターを構成し、セキュリティー・ポリシーを変更することを検討してください。

資格最適化を使用したモニターの開始時から休止しているユーザーは、資格最適化レポートに含まれません。モニターされているが、推奨がない特定のユーザーを監視するには、資格の参照またはその他のいずれかの Guardium アクティビティー・モニター・ツールを使用して、ユーザーのアクティビティーを手動で確認してください。 ポリシーが正しく定義されている場合、ツールにはすべての情報が含まれます。

資格分析はコレクターごとに実行され、grdapi によって構成されたデータ・ソースでのみ機能します。

Must Gather 機能は資格最適化をサポートします。『IBM サポートのための基本情報』を参照してください。

「ディスカバー」 > 「データベース資格」 > 「資格最適化」から資格最適化にアクセスします。

- 資格最適化の有効化および構成
  資格最適化を有効化および構成するには、次の grdapi コマンドを使用します。
- 資格最適化の新機能
  「新機能」タブは、暦週の日曜日から日曜日までに、システムに対して行われた追加と変更を要約します。
- 資格最適化のユーザーおよびロール
  「ユーザーおよびロール」タブには、このコレクターで資格最適化に対して有効になっているすべてのデータ・ソースの、すべてのユーザーとそのロールがリストされます。
- 資格最適化に関する推奨
  推奨は、ユーザー・アクセスを必要最小限に抑えることを目的とした固有のアクションを特定します。
- 資格最適化の資格の参照
  このウィンドウのビューおよびフィルターを使用して、資格のアクティビティー・レベルおよび資格のリネージュを表示します。
- 資格最適化の仮定
  仮定は、(資格が存在するかどうかに関係なく) 特定のオブジェクトに対して 1 つ以上の特定の動詞を持つ特定のユーザーの資格に関する推定理由を示します。

親トピック: ディスカバー

資格最適化の有効化および構成

資格最適化を有効化および構成するには、次の grdapi コマンドを使用します。

すべてのコマンドはコレクターで実行され、既に定義されている Guardium データ・ソースを使用します。まず、コレクターで機能を有効にし、データ・ソースを指定して、特定の機能を有効にします。

資格最適化に含まれるデータを微調整することによって、最も正確な結果が得られます。

ユーザーおよびロールと資格の参照はデフォルトで有効になっていますが、関連データを抽出するには、extractActivity および extractEntitlement を `true` に設定する必要があります。その他の 3 つの機能 (「新機能」、「推奨」、「仮定」) は個別に有効化されます。例えば、「仮定」を無効にしたまま「推奨」を有効にすることができます。

資格の推奨は、userScope パラメーターおよび objectScope パラメーターによってフィルタリングされたデータのサブセットを使用します。資格の参照は、データのフィルタリングに userScope パラメーターを使用します。どちらのパラメーターも Guardium グループを 1 つ以上指定します。通常、この目的に使用するための特定のグループを作成します。ストレージおよび処理を最小化するために、必要なデータだけを抽出するようにグループを定義してください。 すべてのデータが分析され、最終的な結果が得られるようにするため、グループには完全な監査が必要です。完全な監査を持つグループを使用する場合、資格の参照は、アクティビティーに関係なく、すべてのユーザーのすべての権限を表示します。userScope 定義以外のユーザーがウィンドウに表示されますが、そのアクティビティー・カウントは「不明」です。

ベスト・プラクティスは、まれにしか変更されないデータ収集スキームを慎重に評価して設計することです。これには 2 つの理由があります。1 つ目の理由は、構成を変更するたびに、レポートのデータを生成するために 1 週間かかることです。2 つ目の理由は、データは過去 3 週間のデータと比較されるため、データ定義を変更すると、最初の 3 週間分の比較の意義が薄れることです。

個々の機能を有効にすると、データは最初の日曜日から各タブに表示されます。

『GuardAPI 資格最適化機能』でコマンドの完全な詳細を参照してください。

前提条件

- クイック検索は有効になっています。(仮定、推奨、および資格の参照でのアクティビティーの更新に必要です。)
- 資格最適化を構成するユーザーは、構成されたデータ・ソース内にあるすべてのメタデータおよびスキーマ表に対する許可を持っている必要があります。

コレクターでの entitlement_optimization の有効化

コレクターで資格最適化を有効にします。

構文:

grdapi enable_entitlement_optimization

コレクターでの entitlement_optimization の無効化

コレクターで資格最適化を無効にします。

構文:

grdapi disble_entitlement_optimization

資格最適化へのデータ・ソースの追加

1 つ以上のデータ・ソースを資格最適化に追加して、個々の分析を有効にします。

構文:

grdapi add_datasource_to_entitlement_optimization datasourceName=[datasource] isEnabled=[true/false] userScope=[USER SCOPE] objectScope=[OBJECT SCOPE] extractActivity=[true/false] extractEntitlement=[true/false] generateRoleClusters=[true/false] generateNews=[true/false] generateRecommendations=[true/false]

この表を使用して、機能ごとに必要な抽出を判別します。

表 1. 分析タイプごとに必要な enable_entitlement_optimization パラメーター

	新機能 (generateNews)	ユーザーおよびロール	推奨 (generateRecommendations)	資格の参照	仮定 (generateRoleClusters)

	新機能 (generateNews)	ユーザーおよびロール	推奨 (generateRecommendations)	資格の参照	仮定 (generateRoleClusters)
extractActivity				X	X
extractEntitlement	X	X	X	X	

資格最適化からのデータ・ソースの除去

データが収集されないように、資格最適化から 1 つ以上のデータ・ソースを除去します。

remove_datasource_from_entitlement_optimization datasourceName=[datasource name]

資格データ・ソース・パラメーターの変更

資格最適化に既に有効になっているデータ・ソースのパラメーターを変更します。

構文:

grgapi set_entitlement_datasource_parameter datasourceName=[datasource name] parameterName=[value] parameterName=[value]

ここで、parameterName は次のいずれかです。

isEnabled

userScope

objectScope

extractActivity

extractEntitlement

generateRoleClusters

generateNews

generateRecommendations

filterTempObjects

filterIgnoreVerbs

最適化情報の表示

構文:

grdapi get_entitlement_optimization_info

一般的な出力:

資格最適化は有効です

```
==================================
Datasource: SCALE-DB16
==================================
isEnabled: true
userScope:
objectScope:
extractActivity: true
extractEntitlement: true
generateRoleClusters: true
generateNews: true
generateRecommendations: true
filterTempObjects: true
filterIgnoreVerbs: true
```

親トピック: 資格最適化

資格最適化の新機能

「新機能」タブは、暦週の日曜日から日曜日までに、システムに対して行われた追加と変更を要約します。

この機能を有効にすると、データは最初の日曜日からタブに表示されます。

「新機能」タブには、次のものが表示されます。

- 新規のユーザー、ロール、オブジェクトの数、およびこれらの追加項目に関連付けられているデータベースの数
- 新規の被付与者と付与者の数と、付与の数

何を確認すべきか

現在の傾向を調べます。例えば、ドリルダウンして以下を検索します。

- 資格における異常なタイプまたは変更の数量

- 最もアクティブな付与者/被付与者

いずれかのトピックの「詳細」をクリックして、追加項目の詳細表を開きます。例えば、新規ユーザーの詳細はサーバー名とサービス名です。

親トピック: 資格最適化

資格最適化のユーザーおよびロール

「ユーザーおよびロール」タブには、このコレクターで資格最適化に対して有効になっているすべてのデータ・ソースの、すべてのユーザーとそのロールがリストされます。

この機能を有効にすると、データは最初の日曜日からタブに表示されます。

このタブは、1 つのデータベース・タイプのみでデータを表示する標準の Guardium ユーザーおよびロール・レポートに基づきます。次の項目が表示されます。

- ホスト
- サービス名
- データベース・タイプ
- 被付与者
- 被付与者のタイプ
- ロール

標準の照会 - レポート・ビルダー機能を使用できます。これには表の上にあるアイコンでアクセスします。

親トピック: 資格最適化

資格最適化に関する推奨

推奨は、ユーザー・アクセスを必要最小限に抑えることを目的とした固有のアクションを特定します。

この機能を有効にすると、データは最初の日曜日からタブに表示されます。

システムはユーザーおよび特権を継続的に評価しています。週次資格推奨レポートは過去 3 週間のデータ (デフォルト) に基づいており、それぞれの新規レポートが前のレポートのデータと重複するようになっています。「推奨」タブは「レポート」の推奨レポートに相当します。このレポートは配布レポートとして有効にすることができます。

userScope パラメーターをカスタマイズした場合、推奨には指定されたユーザー・グループのユーザーのみが含まれます。userScope パラメーターと objectScope パラメーターは、推奨の範囲を明示的に定義するために使用されます。ユーザーおよびオブジェクトに関する推奨の正確さを最大限にするには、指定されたグループのユーザーとオブジェクトに完全な監査が必要です。

実装の前に、特定のサーバー、データベース、オブジェクト、および推奨タイプをドリルダウンすることにより、管理者はすべての推奨を徹底的に調査する必要があります。

タブの上部には、推奨をタイプ別に表示する円グラフが含まれています。ウィンドウの下部にある表には、推奨がリストされています。標準のレポート・アイコンを使用して推奨レポートを変更したり、「エクスポート」をクリックしてレポートをエクスポートしたり、「アクション」をクリックして API にマップしたりできます。

推奨タイプは以下のとおりです。

表 1. 推奨タイプ

タイプ	文字列	詳細
ANOMAL USER	ロール {source} の中でユーザー {object} に異常なアクティビティーがあります (User {object} has anomal activity within role {source})	特定のロール内のユーザー・アクティビティー・カウントが異常です。これは、当該ユーザーが他のユーザーよりも極端にアクティブであるか、または極端にアクティブでないことを意味します。
ALERT ACTIVITY (特別ユーザー)	ユーザー {source} が特権 {verb}-{object} を使用しましたが、資格は検出されませんでした	標準的な特別ユーザーは自分自身に許可を付与し、アクションを実行してから、許可を削除します。ユーザーは、資格の変更とそのアクティビティーの時差のため、誤って特別ユーザーとして識別されることがあります。Guardium アクティビティー・モニター・ツールを使用して、特権が正当かどうかを判別します。
DORMANT_USER	非アクティブまたは空のユーザー {object} を削除します	ユーザーに特権が割り当てられていないか、特定の間隔内にアクティビティーがありませんでした。
DORMANT_ROLE	非アクティブまたは空のロール {role} を削除します	ユーザーがないか、ユーザーによるアクティビティーがないか、または特権が空になっています
REVOKE_FROM_USER	ユーザー {source} から {verb}-{object} を取り消します	ユーザーは関連するオブジェクト、動詞に対してどのアクティビティーも実行しませんでした。
REVOKE _FROM_ROLE	ロール {source} から {verb}-{object} を取り消します	特定のロール内のすべてのユーザーは、オブジェクト、動詞に対してどのアクティビティーも実行しませんでした。
REMOVE_FROM_ROLE	ユーザー {object} をロール {source} から削除します	ユーザーは、ロールによって付与されたどの特権も使用しませんでした。
INACTIVE DATABASE	データベースにはアクティビティーがありません (Database has no activity)	未使用のデータベースを正当化できない場合は、除去してください。

親トピック: 資格最適化

資格最適化の資格の参照

このウィンドウのビューおよびフィルターを使用して、資格のアクティビティー・レベルおよび資格のリネージュを表示します。

この機能を有効にすると、データは最初の日曜日からタブに表示されます。最初の日曜日以降、アクティビティーは毎日更新されます。

この情報は一般的な資格の調査や、推奨レポートの推奨事項をさらに評価するのに役立ちます。このウィンドウのデフォルトのビューは、最高レートの未使用の特権を含むデータ・ソースを示した棒グラフです。

資格の参照は、extractEntitlement が使用可能な grdAPI で定義されているデータ・ソースのすべての資格を表示します。これは、アクティビティー収集がオフの場合およびユーザー有効範囲とオブジェクト有効範囲が定義されている場合に当てはまります。いつでも、すべてのユーザーの許可を検索して表示することができます。

アクティビティー・カウント・フィールドの結果は、以下のように userScope パラメーターの影響を受けます。

- userScope に含まれているユーザー:
-
  - アクティブ・ユーザーは緑で表示され、「アクティビティー・カウント」列に数値結果が示されます。
  - 非アクティブ・ユーザーは赤で表示され、アクティビティー・カウントは「非アクティブ」です。
- userScope に含まれないユーザー:
  - アクティブ・ユーザーは緑で表示され、アクティビティー・カウントに数値結果が示されます。
  - 非アクティブ・ユーザーはグレーで表示され、アクティビティー・カウントは「不明」です。

一般的な調査:

- ユーザーが許可を持つオブジェクトと、ユーザーがそれらを使用するかどうかを判別します
- ユーザーが、許可された特定の時刻にオブジェクトに対する許可を使用したかどうかを判別します
- 予想を上回って使用された許可があるかどうか
- 1 回だけ使用された許可があるかどうか
- 親ロールまたはロール階層から継承された、著しく使用されている許可のリネージュ (明示的または暗黙的) は何であるか

完全な SQL を使用して特定の特権がどのように使用されているかを詳しく調べるには、データ・アクティビティーを検索し (「調査」 > 「データ・アクティビティーの検索」)、「結果表」の「データベース・ユーザー」または「ソース・プログラム」を右クリックして、「データベース・ユーザー別の完全な SQL」を選択します。

未使用の資格は、通常、以下のいずれかです。

- アクションはほとんど実行されないが、有効な資格 (例えば、四半期レポートの生成など)
- 未使用のため、認められない (脆弱点)

特定のサーバーの特定のサービスに関する資格の使用状況を表示するには、次のようにします。

1. 左側で、サーバー IP とサービスを選択します。
2. 「名前」、「オブジェクト名」を 1 つ以上指定してフィルタリングします。
3. オプションで、動詞または日付範囲を入力します。

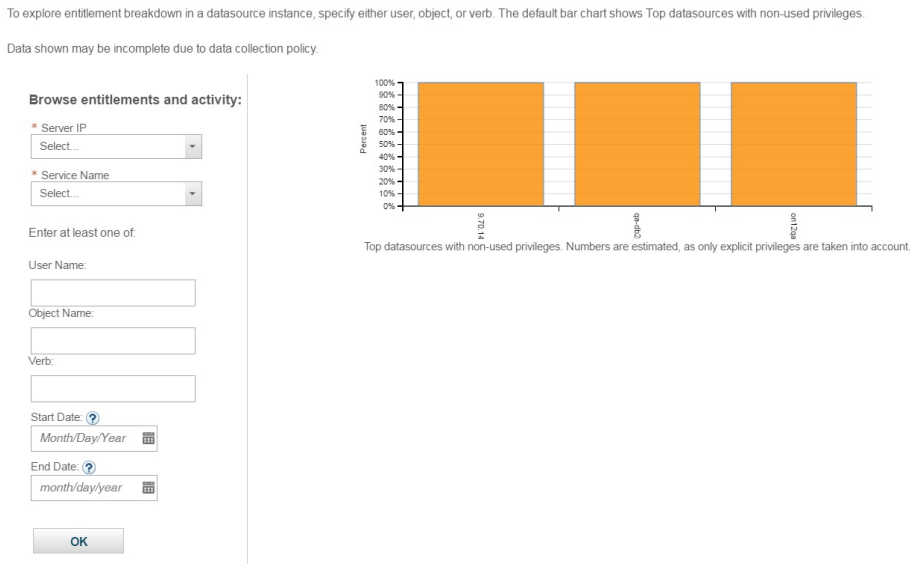


図 1. 資格基準の選択

この表は、「被付与者のタイプ」、「被付与者」、「動詞」、「名前」、「アクティビティー・カウント」、および「リネージュ」を示します。1 人のユーザーは、親ロールまたはロール階層から継承された複数の特権リネージュ (明示的または暗黙的) を持つことができます。

親トピック: 資格最適化

資格最適化の仮定

仮定は、(資格が存在するかどうかに関係なく) 特定のオブジェクトに対して 1 つ以上の特定の動詞を持つ特定のユーザーの資格に関する推定理由を示します。

機能を有効にすると、データは最初の日曜日からこのタブに表示されます。

Guardium は類似したユーザーの動作を分析して、推定理由を作成します。これは非常に関連性の高い情報を提供する場合があります。この分析は、未使用の資格および REVOKE_FROM_USER の推奨を調べる際に役立ちます。これは一般的な指示であり、他の資格最適化機能とともに使用する必要があります。

次の詳細を入力し、「OK」をクリックして、可能性を導出します。

- ユーザー名
- オブジェクト名
- 動詞 (1 つ以上)
- サーバー IP
- サービス名

可能な応答は次のとおりです。

- このデータベース・ユーザーがこの特権を使用する可能性は **n%** です。可能性が 100% の場合、ユーザーがアクティビティーを少なくとも 1 回使用したことを示します。
- サーバー上でデータ・ソースが見つかりませんでした。
- オブジェクトおよび DB ユーザーは範囲内にありません。
- データベース・ユーザーおよび特権の十分な証拠が見つかりません。選択されたデータベースにユーザー/オブジェクト/動詞が存在しない、ユーザーのアクティビティーが見つからない、またはオブジェクト/動詞タプルのアクティビティーが見つからない、のいずれかです。考えられる修正: アクティビティーの収集が実行されるのを待機してください。入力が正確であることを確認してください。

親トピック: 資格最適化

保護

機密データを含むデータベースおよびファイル・システムを識別した後、いくつかのステップを実行することで、そのデータを保護できます。保護オプションには、データのマスキング、データ・アクセスに基づく担当者へのアラート生成、アクセス制限を強制するポリシーの確立などがあります。

- ポリシー
  ポリシーは、Guardium システムによって監視されているデータベース・トラフィックにリアルタイムで適用されるルールとアクションの集合です。ポリシーは、どのトラフィックが無視されるのか、またはログに記録されるのか、どのアクティビティーにより細分度の高いロギングが必要であるか、どのアクティビティーがアラートをトリガーしたりデータベースへのアクセスをブロックしたりするのかを定義します。
- 相関アラート
  アラートは、例外またはポリシー・ルール違反が検出されたことを示すメッセージです。
- 相関アラートを使ってイベントを通知する方法
  アプリケーションのいずれかの個別ユーザーで最近 3 時間に 15 個より多い SQL エラーが存在する場合、相関アラートをトリガーします。
- インシデント管理
  統合インシデント管理 (IIM) アプリケーションには、 データベースのセキュリティー・インシデントをトラッキングして解決するワークフロー自動化機能を備えたビジネス・ユーザー・インターフェースがあります。
- 複数のデータベース・セキュリティー・インシデントのレビュー管理方法
  インシデントの管理を行い、データベースのセキュリティー・インシデントをトラッキングして解決します。
- 照会再書き込み
  照会再書き込み機能を使用してデータベース照会をインターセプトし、そのデータベース照会をセキュリティー・ポリシーで定義された条件に基づいて再書き込みすることにより、データベースに対するアクセス権を詳細に制御することができます。
- ファイル・アクティビティーのポリシーおよびルール
  ファイル・アクティビティー・モニターは、UNIX ファイル・サーバーおよび Windows ファイル・サーバー上の機密データの保全性と保護を確保します。
- FAM MS Office イベントの統合の構成
  FAM モニターの Office イベント統合機能を使用して、MS Word、Excel、および PowerPoint の無関係のファイル・アクティビティーをフィルターで除外します。

ポリシー

ポリシーは、Guardium システムによって監視されているデータベース・トラフィックにリアルタイムで適用されるルールとアクションの集合です。ポリシーは、どのトラフィックが無視されるのか、またはログに記録されるのか、どのアクティビティーにより細分度の高いロギングが必要であるか、どのアクティビティーがアラートをトリガーしたりデータベースへのアクセスをブロックしたりするのかを定義します。

- ポリシーについて
  セキュリティー・ポリシーには、データベース・クライアントとサーバーとの間の監視対象トラフィックに適用される順序付きルール・セットが含まれています。各ルールは、クライアントからの要求、またはサーバーからの応答に適用できます。複数のポリシーを定義でき、Guardium アプライアンスに一度に複数のポリシーをインストールすることができます。
- ポリシー・ルールのアクション
  ポリシー・ルールが一致した場合に実行するブロッキング・アクション、アラート・アクション、またはロギング・アクションを定義します。
- ポリシーおよびポリシー・ルールの作成とインストール
  ポリシーおよびポリシー・ルールを管理するには、「データのポリシー・ビルダー」を使用します。
- 「ポリシー・インストール」ツールの使用
  このトピックを使用して、Guardium コレクターにポリシーをインストールし、スケジュールを変更します。

親トピック: 保護

ポリシーについて

セキュリティー・ポリシーには、データベース・クライアントとサーバーとの間の監視対象トラフィックに適用される順序付きルール・セットが含まれています。各ルールは、クライアントからの要求、またはサーバーからの応答に適用できます。複数のポリシーを定義でき、Guardium アプライアンスに一度に複数のポリシーをインストールすることができます。

ポリシー内の各ルールは、条件付きアクションを定義します。条件は、単純なテスト (例えば、「許可されたクライアント IP」グループに属していないクライアント IP アドレスからのすべてのアクセスを検査する) にすることも、複数のメッセージおよびセッション属性 (データベース・ユーザー、ソース・プログラム、コマンド・タイプ、時刻など) を考慮した複雑なテストにすることもできます。また、指定された時間フレーム内で条件が満たされた回数を識別する場合もあります。

ルールで起動されるアクションは、通知アクション (例えば、1 人以上の受信者に対する E メール通知)、ブロッキング・アクション (クライアント・セッションが切断されるなど) のほか、イベントがポリシー違反としてログに記録されるだけの場合もあります。所定の環境やアプリケーションに固有と見なされる条件に対して必要とされるすべてのタスクを実行するカスタム・アクションを開発することができます。

ポリシー違反は、アラートまたは「ロギングのみ」アクションが起動されるごとにログに記録されます。オプションで、ルールを起動した SQL (データ値も含む) を、ポリシー違反で記録することができます。ポリシー違反は、プロセスによって自動的に、または権限を持つユーザーによって手動でインシデントに割り当てられます (詳しくは、インシデント管理を参照してください)。
重要: 相関アラートをポリシー違反ドメインに書き込むこともできます (詳しくは、『相関アラート』を参照してください) 。
ポリシー・ルールは、違反をロギングするほか、クライアント・トラフィックのロギング (構成体および構成体インスタンスとしてログに記録される) に影響を与える場合があります。

- 構成体は、Guardium® がトラフィック内で検出する要求のプロトタイプです。構成体に含まれるコマンド、オブジェクト、およびフィールドの組み合わせは非常に複雑な場合がありますが、それぞれの構成体は固有性が高いアクセス要求のタイプを表します。新しい構成体の検出およびロギングは、検査エンジンの開始時に始まり、デフォルトではセキュリティー・ポリシー・ルールに関係なく続行します (説明した例外は除きます)。
- トラフィック内で検出された構成体の各インスタンスもログに記録されます。各インスタンスは、特定のクライアント/サーバー・セッションに関連付けられます。構成体インスタンスの SQL は、ポリシー・ルールによりそのインスタンスまたはそのインスタンスの特定のクライアント/サーバー・セッション (値の有無に関わらず) の SQL のロギングが要求された場合を除いて、保管されません。

セキュリティー・ポリシー・ルールは、クライアント構成体インスタンスへの SQL の組み込みを制御するほか、セッションの残りの構成体およびインスタンスのロギングを無効にすることができます。

ボリュームが非常に大きい場合は、情報の解析や構成体およびインスタンスへの統合は、未解析ログ・オプションを使用して据え置くことができます。未解析ログを使用すると、アラートおよびレポートの生成は、ログに記録された情報が統合されるまで遅延されます。

ログに記録されるクライアント・トラフィックを完全に制御するために、ポリシーを選択的な監査証跡ポリシーとして定義することができます。このタイプのポリシーでは、監査のみ ルールおよびオプション・パターンによって、ログに記録されるすべてのクライアント・トラフィックが識別されます。

データ・モニター・ポリシーは、「保護」 > 「セキュリティー・ポリシー」 > 「データのポリシー・ビルダー」または「保護」 > 「セキュリティー・ポリシー」 > 「ポリシー・インストール」からインストールします。
重要: デフォルト・ポリシーが存在するのは、新規インストール (アップグレードではない) の場合のみです。ルールは存在しませんが、「選択的な監査」がチェックされています (これは Guardium システムがデフォルト・ポリシーによって収集するトラフィックはないということを意味します)。64 ビットの Guardium (新規インストール) のデフォルト・ポリシーは、デフォルト - 不明な接続に対するデータ・アクティビティーを無視です。

- ルールのタイプ、カテゴリー、分類
  ポリシー内では、ルールは、トラフィックの各エレメントが分析されるに従って、出現順に評価されます。
- 最小数およびリセット間隔
  一部のアクティビティーは、特定の発生率を下回る場合には正常かつ受け入れ可能ですが、発生率が許容可能なしきい値を超える場合には、注意を要します。
- 次のルールに進む
  デフォルトでは、1 つのトラフィック単位に対するアクセス・ルールおよび例外ルールの評価は、1 つのルールに複数のアクションが含まれている場合を除いて、ルールの起動時に終了します。同じ、または類似した条件に対して複数のアクションを実行する必要がある場合、そのルールの「次のルールに進む」設定を有効にしてください。
- ポリシー違反による値の記録
  「値を記録」設定を使用すると、ルールが満たされる原因となった実際の構成体が SQL 文字列属性に記録され、レポートで使用可能になります。
- ルールでの値および値のグループ
  多くのポリシー・ルール基準では、単一値または値のグループを指定できます。単一値と値のグループの両方を同時に指定することもできます。
- 正規表現とのパターンのマッチング
  正規表現を使用してトラフィックでデータの複合パターンを検索します。
- 特殊パターン・テスト
  これらの特殊パターン・テストを使用することで、データベース・サーバー とクライアントの間で送受信されるトラフィックに含まれる機密データを識別できます。
- 未解析ログ
  「未解析ログ」を使用すると、Guardium は情報を即時に解析することなくログに記録することができます。
- 未解析ログに関するルール
  「未解析ログに関するルール」設定は、Guardium によるポリシー・ルールの処理方法を変更します。
- 選択的な監査証跡
  「選択的な監査証跡」オプションを使用して、Guardium システムでのロギングの量を制限します。
- アナライザー・ルール
  アナライザー・レベルには特定のルールを適用することができます。
- 文字セット
  抽出ルールで文字セット・コードを使用できます。

親トピック: ポリシー

ルールのタイプ、カテゴリー、分類

ポリシー内では、ルールは、トラフィックの各エレメントが分析されるに従って、出現順に評価されます。

ルールには、次の 3 つのタイプがあります。

- クライアント要求に適用されるアクセス・ルール。例えば、特定の IP アドレス・グループから発行された UPDATE コマンドをテストするなど。
- サーバーから返される例外 (応答) を評価する例外ルール。例えば、1 分間に 5 回のログイン失敗があるかどうかの検査など。
- (要求に応じて) サーバーからの戻りデータを評価する抽出ルール。例えば、社会保障番号やクレジット・カード番号など、数値パターンに従って戻りデータをテストするなど。

ルールごとに、オプションで「カテゴリー」および「分類」の一方、または両方を割り当てることができます。これらは、レポートおよびインシデント管理の両方について、ポリシー違反をグループ化するために使用します。

親トピック: ポリシーについて

最小数およびリセット間隔

一部のアクティビティーは、特定の発生率を下回る場合には正常かつ受け入れ可能ですが、発生率が許容可能なしきい値を超える場合には、注意を要します。

例えば、対話式のデータベース・アクセスが許可されている場合、ログインの失敗が一定して、しかし比較的低い比率で発生することは想定内です。ただし、急激に発生率が上昇する場合は、攻撃が進行中であることを示している可能性があります。

しきい値を処理するために、各ポリシー・ルールに最小数とリセット間隔を指定できます。例えばこれは、ログイン失敗数が、1 分 (リセット間隔) 以内に 100 (最小数) を上回ったときに、ルール・アクションを起動させる場合などに使用できます。この指定を省略した場合、デフォルトではルールが満たされるごとにルール・アクションが実行されます。

親トピック: ポリシーについて

次のルールに進む

デフォルトでは、1 つのトラフィック単位に対するアクセス・ルールおよび例外ルールの評価は、1 つのルールに複数のアクションが含まれている場合を除いて、ルールの起動時に終了します。同じ、または類似した条件に対して複数のアクションを実行する必要がある場合、そのルールの「次のルールに進む」設定を有効にしてください。

「次のルールに進む」設定は、アクセス・ルールに続くアクセス・ルールおよび例外ルールに続く例外ルールに適用されます。アクセス・ルールに続く例外ルールまたは例外ルールに続くアクセス・ルールには適用されません。

抽出ルールは、そのルールに先行するアクセス/例外ルールの終了に関係なく処理されます。

親トピック: ポリシーについて

ポリシー違反による値の記録

「値を記録」設定を使用すると、ルールが満たされる原因となった実際の構成体が SQL 文字列属性に記録され、レポートで使用可能になります。

「値を記録」設定が使用されない場合、SQL ステートメントは記録されません。

ポリシー違反にすべての値を含めるには、そのルールの「値を記録」パラメーターを「1 – ポリシー違反の完全な SQL をログに記録する」に設定してください。
重要: 値が指定された全 SQL は、ポリシー違反レコードの、ポリシー違反レポート・ドメイン内でのみ使用可能になります。クライアント・トラフィック・ログや、データ・アクセス・ドメインからのレポートでは使用可能になりません。(データ値の有無に関係なく) 全 SQL をクライアント・トラフィック・ログに含めるには、「全 SQL をロギング」ルール・アクションを使用してください。
親トピック: ポリシーについて

ルールでの値および値のグループ

多くのポリシー・ルール基準では、単一値または値のグループを指定できます。単一値と値のグループの両方を同時に指定することもできます。

グループ・メンバーにはワイルドカード (%) 文字が含まれる場合があり、グループの各メンバーが複数の実値に一致することがあるのでご注意ください。

- ネガティブ・ルール: ネガティブ・ルールを作成するには、!= 演算子または「グループ外」演算子を使用します。例えば、指定された「アプリケーション・ユーザー」ではない、または選択されたグループのメンバーではない、などです。単一値と値のグループの両方を除外することもできます。例えば、「アプリケーション・ユーザー」の 2 つのネガティブ・ルール基準を定義して、1 つのルール基準では特定のユーザーに対して != を使用し、もう 1 つでは「グループ外」を使用して、選択したグループのメンバーを除外します。
- 空の値: トラフィック内の空の値を検査するために、特殊値 guardium://empty を入力します。この値は、フィールド「アプリケーション・イベント・テキスト」、「アプリケーション・ユーザー」、「データベース名」、「データベース・ユーザー」、「イベント・タイプ」、「イベント・ユーザー名」、「オペレーティング・システム・ユーザー」、および「ソース・アプリケーション」にのみ指定可能です。
- テストする新規グループを定義するには:「グループ内」演算子または「グループ外」演算子を選択して、  アイコンをクリックし、新規グループを定義します。
- 任意の値とマッチングするには: = 演算子を選択して、値フィールドをブランクのままにします。
- 特定の値とマッチングするには: = 演算子を選択して、マッチングする値をテキスト・フィールドに入力します。
- グループの任意のメンバーとマッチングするには:「グループ内」演算子を選択して、グループのリストからグループを選択します。最小数が 1 より大きい場合、カウンターは 1 つになり、グループの任意のメンバーが一致するごとに増分されます。
- 個々の値またはグループの任意のメンバーとマッチングするには: 2 つのルール基準を定義して、1 つのルール基準では = 演算子を使用して特定の値をマッチングし、もう 1 つでは「グループ内」を使用して、選択したグループのメンバーをマッチングします。最小数が 1 より大きい場合、カウンターは 1 つになり、個々の値またはグループの任意のメンバーが一致するごとに増分されます。
- 最小数が 1 より大きい場合に、個々の値をそれぞれ個別にカウントする: = 演算子を選択して、値フィールドにピリオド (.) を入力します。「サービス名」または「ネットワーク・プロトコル」の基準では、ピリオドのオプションを使用できないので注意してください。
- 最小数が 1 より大きい場合に、グループのメンバーをそれぞれ個別にカウントする:「グループ内」演算子を選択して、リストからグループを選択します。

親トピック: ポリシーについて

正規表現とのパターンのマッチング

正規表現を使用してトラフィックでデータの複合パターンを検索します。

UNIX の正規表現の実装とは異なり、Guardium の正規表現の実装は POSIX 1003.2 に準拠します。正規表現は、後ろに RE ボタンがあるすべてのフィールドで使用可能です。
ヒント: 値フィールドに特殊値 `guardium://regexp/(正規表現)` を入力することで、「データベース・ユーザー」「アプリケーション・ユーザー」、「ソース・アプリケーション」、「フィールド名」、「オブジェクト」、「アプリケーション・イベント値テキスト」の基準でも正規表現を使用できます。
制約事項: Guardium は、英語以外の言語の正規表現はサポートしていません。
正規表現の使用について詳しくは、正規表現を参照してください。
親トピック: ポリシーについて

特殊パターン・テスト

これらの特殊パターン・テストを使用することで、データベース・サーバー とクライアントの間で送受信されるトラフィックに含まれる機密データを識別できます。

各ポリシー・ルールには、1 つの特殊パターン・テストを含めることができます。これらのテストのいずれかを使用するには、最初にいずれかの特殊パターン・テスト名、次に 1 つのスペースとルール名を固有にするための 1 つ以上の追加の文字を含むルール名を使用します。例えば、従業員の社会保障番号を検索する場合、ルールに `guardium://SSEC_NUMBER employee` という名前を付けます。ルールの他のすべてのコンポーネント (特定のクライアント、サーバーの IP アドレス) も指定できます。

これらのテストは文字パターンに一致しますが、その一致では、対象の項目 (社会保障番号など) が確実に検出されるわけではありません。さまざまな環境下 (特に、データ内で長い数値シーケンスが連結されている場合など) においては、誤検出が生じる可能性があります。

guardium://CREDIT_CARD

クレジット・カード番号パターンを検出します。16 桁の数字文字列のテスト、または各セットの間がブランクで区切られた 4 桁の数字 4 セットのテストを行います。 この特殊パターン・テストは、アメリカン・エキスプレスの 15 桁のクレジット・カード番号パターン (最初の数字が 3、2 番目の数字が 4 または 7) についても機能します。例: `1111222233334444` または `1111 2222 3333 4444`

ルール名が "guardium://CREDIT_CARD" で始まり、「データ・パターン」フィールドに 有効なクレジット・カード番号パターンが指定されている場合、ポリシーは、標準の パターン・マッチングに加えて Luhn アルゴリズム (クレジット・カード番号などの ID 番号 検証のために幅広く使用されているアルゴリズム) を使用します。Luhn アルゴリズムは追加の検査であり、パターン検査の代わりにはなりません。 有効なクレジット・カード番号は、16 桁の数字文字列、または 4 桁の数字 4 セット (各セットの間がブランクで区切られる) です。 このパターン・マッチングに Luhn アルゴリズムを組み込むには、「検索式」ボックスに guardium://CREDIT_CARD ルール名と有効な [0-9]{16} 数値の両方が指定されている必要があります。

guardium://PCI_TRACK_DATA

磁気ストライプ・データの 2 つのパターンを検出します。最初のパターンは、セミコロン (;)、16 桁の数字、および等号 (=)、20 桁の数字、および疑問符 (?) で構成されています。 以下に例を示します。

```
;1111222233334444=11112222333344445555?
```

2 つ目のパターンは、パーセント記号 (%)、文字 B、16 桁の数字、カラット記号 (^)、スラッシュ (/) で終了する可変長の文字文字列、カラット記号 (^) で終了する 2 つ目の可変長文字文字列、31 桁の数字、および疑問符 (?) で構成されています。 以下に例を示します。

```
%B1111222233334444^xxx/xxxx x^11112222333344445555666677778888?
```

guardium://SSEC_NUMBER

社会保障番号形式 (3 桁の数字、ダッシュ記号 (-)、2 桁の数字、ダッシュ記号 (-)、4 桁の数字) の数値を検出します (`123-45-6789` など)。ダッシュ記号はいずれも必須です。

guardium://CPF

Cadastro de Pessoas Físicas (CPF)、ブラジルの個人識別用の ID。`nnn.nnn.nnn-nn` 形式の 11 桁の数字が含まれます。最後の 2 桁はチェック・ディジットです。番号が有効であるか検査できるようにするために、元の 9 桁の数字からチェック・ディジットが計算されます。式内の書式制御文字はオプションです。式に一致する場合、チェック・ディジットが検査されます。

guardium://CNPJ

Cadastro Nacional de Pessoas Jurídicas (CNPJ)、ブラジル企業に使用される ID 番号。`00.000.000/0001-00` 形式の 14 桁の数字が含まれます。

- 最初の 8 つの数字は登録番号を表します。
- 次の 4 つの数字はそのエンティティーの支社を示します。0001 が本社を示すデフォルト値です。
- 最後の 2 つの数字はチェック・ディジットです。

式内の書式制御文字はオプションです。式に一致する場合、チェック・ディジットが検査されます。

親トピック: ポリシーについて

未解析ログ

「未解析ログ」を使用すると、Guardium は情報を即時に解析することなくログに記録することができます。

こうすることで、処理リソースが節約され、より大量のトラフィックを処理できるようになります。そのデータは、コレクターまたは統合サービス単位のいずれかで、後ほど Guardium の内部データベースに対して解析およびマージすることができます。

未解析ログ処理に関連する「ポリシー定義による未解析ログ」と「スロットル・メカニズムによる未解析ログ」という 2 つの Guardium 機能があります。

スロットル・メカニズムによる未解析ログ - これは、CLI コマンド store alp_throttle 1 を実行することによって実装される機能です。GDM_FLAT_LOG 表に記録されたトラフィックを処理するために、リアルタイムの S-TAP トラフィックに適用されるのと同じポリシーが使用されます。

スロットル・メカニズムによる未解析ログでは、ポリシー・ビルダーで「未解析ログ」チェック・ボックスにチェック・マークを付けないでください。

ポリシー定義による未解析ログ - この機能を選択するには、「設定」>「ツールとビュー」の「ポリシー・ビルダー」メニューと、「管理」>「アクティビティー・モニター」の「未解析ログ処理」メニューを操作します。

注: 未解析ログに関するルールは、フィールド、オブジェクト、SQL 動詞 (コマンド)、オブジェクト/コマンド・グループ、およびオブジェクト/フィールド・グループを含んだポリシー・ルールでは機能しません。未解析ログ処理において、「未解析」とは構文ツリーが構築されていないことを意味します。 構文ツリーがない場合、フィールド、オブジェクト、および SQL 動詞は判別できません。

LOG FULL DETAILS、LOG FULL DETAILS PER SESSION、LOG FULL DETAILS VALUES、LOG FULL DETAILS VALUES PER SESSION、LOG MASKED DETAILS の各アクションは、フラット・ポリシーのルールでは機能しません。

「ポリシー・ビルダー」の「ポリシー定義」画面にリストされている「未解析ログ」チェック・ボックス・オプションを選択すると、次のようになります。

- データはリアルタイムでは解析されません。
- 未解析ログは、指定された「未解析ログ・リスト」レポートで確認できます。

親トピック: ポリシーについて

未解析ログに関するルール

「未解析ログに関するルール」設定は、Guardium によるポリシー・ルールの処理方法を変更します。

「未解析ログに関するルール」にチェック・マークを付けると、以下のようになります。

- セッション・レベルのルールがリアルタイムで検査されます。
- オフライン処理が行われない場合、ルールは評価されません。

「未解析ログに関するルール」にチェック・マークを付けないと、以下のようになります。

- ポリシー・ルールは、現行のインストール・ポリシーを使用して処理時に起動します。

注: 未解析ログに関するルールは、フィールド、オブジェクト、SQL 動詞 (コマンド)、オブジェクト/コマンド・グループ、およびオブジェクト/フィールド・グループを含んだポリシー・ルールでは機能しません。未解析ログ処理において、「未解析」とは構文ツリーが構築されていないことを意味します。 構文ツリーがない場合、フィールド、オブジェクト、および SQL 動詞は判別できません。

LOG_FULL_DETAILS、LOG_FULL_DETAILS_PER_SESSION、LOG_FULL_DETAILS_VALUES、LOG_FULL_DETAILS_VALUES_PER_SESSION、LOG_MASKED_DETAILS の各アクションは、フラット・ポリシーのルールでは機能しません。

親トピック: ポリシーについて

選択的な監査証跡

「選択的な監査証跡」オプションを使用して、Guardium システムでのロギングの量を制限します。

これは、検査エンジンで受信されているトラフィックのうち重要なトラフィックの割合が比較的小さい場合や、レポート対象となり得るすべてのトラフィックが完全に識別可能である場合に適しています。

選択的な監査証跡ポリシーを指定しない場合、Guardium アプライアンスは検査エンジンで受信したすべてのトラフィックをログに記録します。アプライアンスまたは S-TAP の検査エンジンは、それぞれ 1 つ以上のポートで特定のデータベース・プロトコル (例えば、Oracle など) をモニターするように構成されています。さらに、クライアント/サーバー接続のサブセットからトラフィックを受信するように検査エンジンを構成することができます。この場合、選択的な監査証跡ポリシーよりも多くの情報が取り込まれる傾向にあります。ただし、ユーザーのセキュリティー要件および規制要件を満たすために必要とされるもの以外にも大幅に多くの情報が Guardium アプライアンスにより処理および保管される可能性があります。

選択的な監査証跡ポリシーをインストールすると、ポリシーが要求したトラフィックのみがログに記録されます。そのトラフィックは、次の 2 とおりの方法で識別できます。

- 重要なトラフィックを識別するために使用可能な文字列を、「ポリシー定義」パネルの「監査パターン」ボックスに指定する方法。これにより、例えば、データベースやデータベース表のグループを識別できます。監査パターンは、ロガーが (正規表現のマッチングを通じて) 一致するかどうかを確認するために処理する各 SQL に適用されるパターンである点に注意してください。このパターン・マッチングは、厳密には文字列マッチングです。ポリシー・ルールの場合のような、セッション変数 (例えば、データベース名など) とのマッチングは行われません。
- あるいは、「ルール定義」パネルの 1 つ以上のポリシー・ルールに、「監査のみ」またはロギング・アクションのいずれか (「ロギングのみ」、「全詳細をロギング」など) を指定する方法。ポリシー・ルールを使用すると、考えられるすべての属性のタイプ (データベース・タイプ、データベース名、ユーザー名など) に対してマッチングする、正確な値、グループ、またはパターンを、高精度で指定することができます。

Guardium セキュリティー・ポリシーで「選択的な監査証跡」を有効にした状態で、オブジェクト・グループについてルールが作成された場合、そのグループ内の各エレメントの文字列が検査されます。一致が検出されると、情報をログに記録するかどうかの決定が下され、処理が続行されます。Guardium セキュリティー・ポリシーで「選択的な監査証跡」を有効にした状態で、「NOT」指定を使用してオブジェクト・グループについてルールが作成された場合、やはりグループ内の各エレメントの文字列を検査する必要があり、いずれのエレメントも一致しない場合にのみ、ログに記録して続行することが決定されます。NOT が指定されたルールは、「選択的な監査証跡」とともに使用すると、通常のルールと同様に振る舞います。

以下のような内容です。

- 複数のオブジェクトまたはコマンドに基づくルールのような、OR 状態。
- 2 つの NOT 条件を持つ (例えば、NOT オブジェクト・グループの一部、および NOT コマンド・グループの一部) 状態。および
- 1 つの NOT 条件および 1 つの YES 条件 (例えば、NOT オブジェクト・グループの一部、および YES コマンド・グループの一部) を持つ状態。

注: (少なくとも選択的な監査モードでは) SELECT /*+ ORDERED USE_MERGE(m) */ SELECT /*+ ORDERED */ SELECT /*+ all_rows */ などの照会ヒントを使用した SELECT ステートメントは、それらのステートメントをスキップするためのルール定義にかかわらず、パーサーをパススルーしてログに記録することができます。これは、選択的な監査ポリシーでは、他の機能 (アプリケーション・ユーザー・トランスレーションなど) に必要になる可能性がある特定の SQL のロギングを防いではならないためです。

選択的な監査証跡およびアプリケーション・イベント API

「選択的な監査証跡」ポリシーを使用する場合で、アプリケーション・ユーザーまたはイベントがアプリケーション・イベント API を使用して設定されているときは、アプリケーション・イベントの設定/クリア、またはアプリケーション・ユーザー・コマンドの設定/クリアが検出されるごとに起動される「監査のみ」ルールをポリシーに含める必要があります。 アプリケーション・イベント API を使用したアプリケーション・ユーザーの設定について詳しくは、API によるユーザーの識別を参照してください。

選択的な監査証跡およびアプリケーション・ユーザー・トランスレーション

「選択的な監査証跡」ポリシーを使用する場合、「アプリケーション・ユーザー・トランスレーション」も使用されます。

- ポリシーは、アプリケーション・ユーザー・トランスレーション・ルールに合致しない (例えば、 アプリケーション・サーバーが発信元ではない) すべてのトラフィックを無視します。
- そのポリシーのパターンに合致する SQL だけが、 特殊アプリケーション・ユーザー・トランスレーション・レポートで有効になります。

選択的な監査証跡および空のグループの指定

ルールに付加された空のタプルにより、ルール・アクションが一致しなくなります。

親トピック: ポリシーについて

アナライザー・ルール

アナライザー・レベルには特定のルールを適用することができます。

アナライザー・ルールの例としては、ユーザー定義文字セット、ソース・プログラムの変更、ファイアウォール・モードへの監視判定の発行などがあります。以前のリリースでは、ポリシーやルールは、ロギング状態での要求処理の最後に適用されていました。一部のケースにおいて、これは、これらのルールに基づいた決定が遅れることを意味していました。アナライザー・レベルでルールを適用するということは、より早い段階で決定を行えることを意味します。

親トピック: ポリシーについて

文字セット

抽出ルールで文字セット・コードを使用できます。

使用可能な文字セット・コードのリスト

ANSI_X3.4-1968 - 1
ANSI_X3.4-1986 - 2
ASCII - 3
CP367 - 4
IBM367 - 5
ISO-IR-6 - 6
ISO646-US - 7
ISO_646.IRV:1991 - 8
US - 9
US-ASCII - 10
CSASCII - 11
UTF-8 - 12
ISO-10646/UCS2 - 13
UCS-2 - 14
CSUNICODE - 15
UCS-2BE - 16
UNICODE - 17
UNICODEBIG - 18
TSCII - 19
UCS-2LE - 20
UNICODELITTLE - 21
ISO-10646/UCS4 - 22
UCS-4 - 23
CSUCS4 - 24
UCS-4BE - 25
UCS-4LE - 26
UTF-16 - 27
UTF-16BE - 28
UTF-16LE - 29
UTF-32 - 30
UTF-32BE - 31
UTF-32LE - 32
UTF7 - 33
UTF-7 - 34
UTF-8 - 35
UCS2 - 36
UCS2 - 37
UCS4 - 38
UCS4 - 39
UTF8 - 40
UTF8 - 41
CP819 - 42

IBM819 - 43
ISO-8859-1 - 44
ISO-IR-100 - 45
ISO8859-1 - 46
ISO_8859-1 - 47
ISO_8859-1:1987 - 48
L1 - 49
LATIN1 - 50
CSISOLATIN1 - 51
ISO-8859-2 - 52
ISO-IR-101 - 53
ISO8859-2 - 54
ISO_8859-2 - 55
ISO_8859-2:1987 - 56
L2 - 57
LATIN2 - 58
CSISOLATIN2 - 59
ISO-8859-3 - 60
ISO-IR-109 - 61
ISO8859-3 - 62
ISO_8859-3 - 63
ISO_8859-3:1988 - 64
L3 - 65
LATIN3 - 66
CSISOLATIN3 - 67
ISO-8859-4 - 68
ISO-IR-110 - 69
ISO8859-4 - 70
ISO_8859-4 - 71
ISO_8859-4:1988 - 72
L4 - 73
LATIN4 - 74
CSISOLATIN4 - 75
CYRILLIC - 76
ISO-8859-5 - 77
ISO-IR-144 - 78
ISO8859-5 - 79
ISO_8859-5 - 80
ISO_8859-5:1988 - 81
CSISOLATINCYRILLIC - 82
ARABIC - 83
ASMO-708 - 84
ECMA-114 - 85
ISO-8859-6 - 86
ISO-IR-127 - 87
ISO8859-6 - 88
ISO_8859-6 - 89
ISO_8859-6:1987 - 90
CSISOLATINARABIC - 91
ECMA-118 - 92
ELOT_928 - 93
GREEK - 94
GREEK8 - 95
ISO-8859-7 - 96
ISO-IR-126 - 97
ISO8859-7 - 98
ISO_8859-7 - 99
ISO_8859-7:1987 - 100
CSISOLATINGREEK - 101
HEBREW - 102
ISO-8859-8 - 103
ISO-IR-138 - 104
ISO8859-8 - 105
ISO_8859-8 - 106
ISO_8859-8:1988 - 107
CSISOLATINHEBREW - 108
ISO-8859-9 - 109
ISO-IR-148 - 110
ISO8859-9 - 111
ISO_8859-9 - 112
ISO_8859-9:1989 - 113
L5 - 114
LATIN5 - 115
CSISOLATIN5 - 116
ISO-8859-10 - 117
ISO-IR-157 - 118
ISO8859-10 - 119
ISO_8859-10 - 120
ISO_8859-10:1992 - 121
L6 - 122

LATIN6 - 123
CSISOLATIN6 - 124
ISO-8859-13 - 125
ISO-8859-13 - 126
ISO-8859-13 - 127
ISO-8859-13 - 128
L7 - 129
LATIN7 - 130
ISO-8859-14 - 131
ISO-CELTIC - 132
ISO-IR-199 - 133
ISO8859-14 - 134
ISO_8859-14 - 135
ISO_8859-14:1998 - 136
L8 - 137
LATIN8 - 138
ISO-8859-15 - 139
ISO-IR-203 - 140
ISO8859-15 - 141
ISO_8859-15 - 142
ISO_8859-15:1998 - 143
ISO-8859-16 - 144
ISO-IR-226 - 145
ISO8859-16 - 146
ISO_8859-16 - 147
ISO_8859-16:2000 - 148
KOI8-R - 149
CSKOI8R? - 150
KOI8U? - 151
KOI8R? - 152
CP1250 - 153
MS-EE - 154
WINDOWS-1250 - 155
CP1251 - 156
MS-CYRL - 157
WINDOWS-1251 - 158
CP1252 - 159
MS-ANSI - 160
WINDOWS-1252 - 161
CP1253 - 162
MS-GREEK - 163
WINDOWS-1253 - 164
CP1254 - 165
MS-TURK - 166
WINDOWS-1254 - 167
CP1255 - 168
MS-HEBR - 169
WINDOWS-1255 - 170
CP1256 - 171
MS-ARAB - 172
WINDOWS-1256 - 173
CP1257 - 174
WINBALTRIM - 175
WINDOWS-1257 - 176
CP1258 - 177
WINDOWS-1258 - 178
850 - 179
CP850 - 180
IBM850 - 181
CSPC850MULTILINGUAL? - 182
862 - 183
CP862 - 184
IBM862 - 185
CSPC862LATINHEBREW? - 186
866 - 187
CP866 - 188
IBM866 - 189
CSIBM866 - 190
MAC - 191
MACINTOSH - 192
MACUK - 193
CSMACINTOSH - 194
MACIS - 195
MAC - 196
MAC - 197
MAC - 198
MAC - 199
MACUKRAINIAN - 200
MAC - 201
MAC - 202

MAC - 203
MAC - 204
MAC - 205
HP-ROMAN8 - 206
R8 - 207
ROMAN8 - 208
HPROMAN8 - 209
ROMAN8 - 210
ARMSCII-8 - 211
GEORGIAN-ACADEMY - 212
GEORGIAN-PS - 213
KOI8-T - 214
KOI8-T - 215
CP1133 - 216
IBM-CP1133 - 217
ISO-IR-166 - 218
TIS-620 - 219
TIS620 - 220
TIS620-0 - 221
TIS620.2529-1 - 222
TIS620.2533-0 - 223
TIS620.2533-1 - 224
CP874 - 225
WINDOWS-874 - 226
VISCII - 227
VISCII - 228
VISCII - 229
TCVN - 230
TCVN-5712 - 231
TCVN5712-1 - 232
TCVN5712-1:1993 - 233
ISO-IR-14 - 234
ISO646-JP - 235
JIS_C6220-1969-RO - 236
JP - 237
CSISO14JISC6220RO? - 238
JISX0201-1976 - 239
JIS_X0201 - 240
X0201 - 241
CSHALFWIDTHKATAKANA - 242
ISO-IR-87 - 243
JIS0208 - 244
JIS_C6226-1983 - 245
JIS_X0208 - 246
JIS_X0208-1983 - 247
JIS_X0208-1990 - 248
X0208 - 249
CSISO87JISX0208? - 250
ISO-IR-159 - 251
JIS_X0212 - 252
JIS_X0212-1990 - 253
JIS_X0212.1990-0 - 254
X0212 - 255
CSISO159JISX02121990? - 256
CN - 257
GB_1988-80 - 258
ISO-IR-57 - 259
ISO646-CN - 260
CSISO57GB1988? - 261
CHINESE - 262
GB_2312-80 - 263
ISO-IR-58 - 264
CSISO58GB231280? - 265
CN-GB-ISOIR165 - 266
ISO-IR-165 - 267
ISO-IR-149 - 268
KOREAN - 269
KSC_5601 - 270
KS_C_5601-1987 - 271
KS_C_5601-1989 - 272
CSKSC56011987 - 273
EUC-JP - 274
EUCJP - 275
EXTENDED_UNIX_CODE_PACKED_FORMAT_FOR_JAPANESE - 276
CSEUCPKDFMTJAPANESE - 277
MS_KANJI - 278
SHIFT-JIS - 279
SHIFT_JIS - 280
SJIS - 281
CSSHIFTJIS - 282

CP932 - 283
ISO-2022-JP - 284
CSISO2022JP? - 285
ISO-2022-JP-1 - 286
ISO-2022-JP-2 - 287
CSISO2022JP2? - 288
CN-GB - 289
EUC-CN - 290
EUCCN - 291
GB2312 - 292
CSGB2312 - 293
CP936 - 294
GBK - 295
GB18030 - 296
ISO-2022-CN - 297
CSISO2022CN? - 298
ISO-2022-CN-EXT - 299
HZ - 300
HZ-GB-2312 - 301
EUC-TW - 302
EUCTW - 303
CSEUCTW - 304
BIG-5 - 305
BIG-FIVE - 306
BIG5 - 307
BIGFIVE - 308
CN-BIG5 - 309
CSBIG5 - 310
CP950 - 311
BIG5-HKSCS - 312
BIG5HKSCS? - 313
EUC-KR - 314
EUCKR - 315
CSEUCKR - 316
CP949 - 317
UHC - 318
CP1361 - 319
JOHAB - 320
ISO-2022-KR - 321
CSISO2022KR? - 322
IBM037 - 323
IBM038 - 324
IBM256 - 325
IBM273 - 326
IBM274 - 327
IBM275 - 328
IBM277 - 329
IBM278 - 330
IBM280 - 331
IBM281 - 332
IBM284 - 333
IBM285 - 334
IBM290 - 335
IBM297 - 336
IBM367 - 337
IBM420 - 338
IBM423 - 339
IBM424 - 340
IBM437 - 341
IBM500 - 342
IBM775 - 343
IBM813 - 344
IBM819 - 345
IBM848 - 346
IBM850 - 347
IBM851 - 348
IBM852 - 349
IBM855 - 350
IBM856 - 351
IBM857 - 352
IBM860 - 353
IBM861 - 354
IBM862 - 355
IBM863 - 356
IBM864 - 357
IBM865 - 358
IBM866 - 359
IBM866NAV? - 360
IBM868 - 361
IBM869 - 362

IBM870 - 363
IBM871 - 364
IBM874 - 365
IBM875 - 366
IBM880 - 367
IBM891 - 368
IBM903 - 369
IBM904 - 370
IBM905 - 371
IBM912 - 372
IBM915 - 373
IBM916 - 374
IBM918 - 375
IBM920 - 376
IBM922 - 377
IBM930 - 378
IBM932 - 379
IBM933 - 380
IBM935 - 381
IBM937 - 382
IBM939 - 383
IBM943 - 384
IBM1004 - 385
IBM1026 - 386
IBM1046 - 387
IBM1047 - 388
IBM1089 - 389
IBM1124 - 390
IBM1129 - 391
IBM1132 - 392
IBM1133 - 393
IBM1160 - 394
IBM1161 - 395
IBM1162 - 396
IBM1163 - 397
IBM1164 - 398
MSCP949 - 399
EUC-JISX0213 - 400
UJIS - 401
CP852 - 402
EUCJP-MS - 403
IBM902 - 404
IBM921 - 405
WINDOWS-31J - 406
IBM1025 - 407
IBM1140 - 408
IBM1137 - 409
IBM1122 - 410
IBM1141 - 411
IBM1142 - 412
IBM1143 - 413
IBM1144 - 414
IBM1145 - 415
IBM1146 - 416
IBM1147 - 417
IBM1148 - 418
IBM1149 - 419
IBM1153 - 420
IBM1155 - 421
IBM1157 - 422
EBCDICUS - 423
IBM1112 - 424
IBM1158 - 425
437 - 426
500g - 427
500V1g - 428
851g - 429
852g - 430
855g - 431
856g - 432
857g - 433
860g - 434
861g - 435
863g - 436
864g - 437
865g - 438
866NAVg - 439
869g - 440
874g - 441
904g - 442

1026g - 443
1046g - 444
1047g - 445
8859_1g - 446
8859_2g - 447
8859_3g - 448
8859_4g - 449
8859_5g - 450
8859_6g - 451
8859_7g - 452
8859_8g - 453
8859_9g - 454
10646-1:1993g - 455
10646-1:1993/UCS4/ - 456
ANSI_X3.4g - 457
ANSI_X3.110-1983g - 458
ANSI_X3.110g - 459
ARABIC7g - 460
ASMO_449g - 461
BALTICg - 462
BIG-5g - 463
BIG-FIVEg - 464
BIG5-HKSCSg - 465
BIG5g - 466
BIG5HKSCSg? - 467
BIGFIVEg - 468
BS_4730g - 469
CAg - 470
CN-BIG5g - 471
CN-GBg - 472
CNg - 473
CP-ARg - 474
CP-GRg - 475
CP-HUg - 476
CP037g - 477
CP038g - 478
CP273g - 479
CP274g - 480
CP275g - 481
CP278g - 482
CP280g - 483
CP281g - 484
CP282g - 485
CP284g - 486
CP285g - 487
CP290g - 488
CP297g - 489
CP420g - 490
CP423g - 491
CP424g - 492
CP437g - 493
CP500g - 494
CP737g - 495
CP775g - 496
CP803g - 497
CP813g - 498
CP851g - 499
CP852g - 500
CP855g - 501
CP856g - 502
CP857g - 503
CP860g - 504
CP861g - 505
CP863g - 506
CP864g - 507
CP865g - 508
CP866NAVg? - 509
CP868g - 510
CP869g - 511
CP870g - 512
CP871g - 513
CP875g - 514
CP880g - 515
CP891g - 516
CP901g - 517
CP902g - 518
CP903g - 519
CP904g - 520
CP905g - 521
CP912g - 522

CP915g - 523
CP916g - 524
CP918g - 525
CP920g - 526
CP921g - 527
CP922g - 528
CP930g - 529
CP932g - 530
CP933g - 531
CP935g - 532
CP936g - 533
CP937g - 534
CP939g - 535
CP949g - 536
CP950g - 537
CP1004g - 538
CP1008g - 539
CP1025g - 540
CP1026g - 541
CP1046g - 542
CP1047g - 543
CP1070g - 544
CP1079g - 545
CP1081g - 546
CP1084g - 547
CP1089g - 548
CP1097g - 549
CP1112g - 550
CP1122g - 551
CP1123g - 552
CP1124g - 553
CP1125g - 554
CP1129g - 555
CP1130g - 556
CP1132g - 557
CP1137g - 558
CP1140g - 559
CP1141g - 560
CP1142g - 561
CP1143g - 562
CP1144g - 563
CP1145g - 564
CP1146g - 565
CP1147g - 566
CP1148g - 567
CP1149g - 568
CP1153g - 569
CP1154g - 570
CP1155g - 571
CP1156g - 572
CP1157g - 573
CP1158g - 574
CP1160g - 575
CP1161g - 576
CP1162g - 577
CP1163g - 578
CP1164g - 579
CP1166g - 580
CP1167g - 581
CP1361g - 582
CP1364g - 583
CP1371g - 584
CP1388g - 585
CP1390g - 586
CP1399g - 587
CP4517g - 588
CP4899g - 589
CP4909g - 590
CP4971g - 591
CP5347g - 592
CP9030g - 593
CP9066g - 594
CP9448g - 595
CP10007g - 596
CP12712g - 597
CP16804g - 598
CPIBM861g - 599
CSA7-1g - 600
CSA7-2g - 601
CSA_T500-1983g - 602

CSA_T500g - 603
CSA_Z243.4-1985-1g - 604
CSA_Z243.4-1985-2g - 605
CSA_Z243.419851g - 606
CSA_Z243.419852g - 607
CSDECMCSg - 608
CSEBCDICATDEg - 609
CSEBCDICATDEAg - 610
CSEBCDICCAFRg - 611
CSEBCDICDKNOg - 612
CSEBCDICDKNOAg - 613
CSEBCDICESg - 614
CSEBCDICESAg - 615
CSEBCDICESSg - 616
CSEBCDICFISEg - 617
CSEBCDICFISEAg - 618
CSEBCDICFRg - 619
CSEBCDICITg - 620
CSEBCDICPTg - 621
CSEBCDICUKg - 622
CSEBCDICUSg - 623
CSEUCKRg - 624
CSEUCPKDFMTJAPANESEg - 625
CSGB2312g - 626
CSIBM037g - 627
CSIBM038g - 628
CSIBM273g - 629
CSIBM274g - 630
CSIBM275g - 631
CSIBM277g - 632
CSIBM278g - 633
CSIBM280g - 634
CSIBM281g - 635
CSIBM284g - 636
CSIBM285g - 637
CSIBM290g - 638
CSIBM297g - 639
CSIBM420g - 640
CSIBM423g - 641
CSIBM424g - 642
CSIBM500g - 643
CSIBM803g - 644
CSIBM851g - 645
CSIBM855g - 646
CSIBM856g - 647
CSIBM857g - 648
CSIBM860g - 649
CSIBM863g - 650
CSIBM864g - 651
CSIBM865g - 652
CSIBM868g - 653
CSIBM869g - 654
CSIBM870g - 655
CSIBM871g - 656
CSIBM880g - 657
CSIBM891g - 658
CSIBM901g - 659
CSIBM902g - 660
CSIBM903g - 661
CSIBM904g - 662
CSIBM905g - 663
CSIBM918g - 664
CSIBM921g - 665
CSIBM922g - 666
CSIBM930g - 667
CSIBM932g - 668
CSIBM933g - 669
CSIBM935g - 670
CSIBM937g - 671
CSIBM939g - 672
CSIBM943g - 673
CSIBM1008g - 674
CSIBM1025g - 675
CSIBM1026g - 676
CSIBM1097g - 677
CSIBM1112g - 678
CSIBM1122g - 679
CSIBM1123g - 680
CSIBM1124g - 681
CSIBM1129g - 682

CSIBM1130g - 683
CSIBM1132g - 684
CSIBM1133g - 685
CSIBM1137g - 686
CSIBM1140g - 687
CSIBM1141g - 688
CSIBM1142g - 689
CSIBM1143g - 690
CSIBM1144g - 691
CSIBM1145g - 692
CSIBM1146g - 693
CSIBM1147g - 694
CSIBM1148g - 695
CSIBM1149g - 696
CSIBM1153g - 697
CSIBM1154g - 698
CSIBM1155g - 699
CSIBM1156g - 700
CSIBM1157g - 701
CSIBM1158g - 702
CSIBM1160g - 703
CSIBM1161g - 704
CSIBM1163g - 705
CSIBM1164g - 706
CSIBM1166g - 707
CSIBM1167g - 708
CSIBM1364g - 709
CSIBM1371g - 710
CSIBM1388g - 711
CSIBM1390g - 712
CSIBM1399g - 713
CSIBM4517g - 714
CSIBM4899g - 715
CSIBM4909g - 716
CSIBM4971g - 717
CSIBM5347g - 718
CSIBM9030g - 719
CSIBM9066g - 720
CSIBM9448g - 721
CSIBM12712g - 722
CSIBM16804g - 723
CSIBM11621162g - 724
CSISO4UNITEDKINGDOMg? - 725
CSISO10SWEDISHg? - 726
CSISO11SWEDISHFORNAMESg? - 727
CSISO15ITALIANg? - 728
CSISO16PORTUGESEg? - 729
CSISO17SPANISHg? - 730
CSISO18GREEK7OLDg? - 731
CSISO19LATINGREEKg? - 732
CSISO21GERMANg? - 733
CSISO25FRENCHg? - 734
CSISO27LATINGREEK1g? - 735
CSISO49INISg? - 736
CSISO50INIS8g? - 737
CSISO51INISCYRILLICg? - 738
CSISO58GB1988g? - 739
CSISO60DANISHNORWEGIANg? - 740
CSISO60NORWEGIAN1g? - 741
CSISO61NORWEGIAN2g? - 742
CSISO69FRENCHg? - 743
CSISO84PORTUGUESE2g? - 744
CSISO85SPANISH2g? - 745
CSISO86HUNGARIANg? - 746
CSISO88GREEK7g? - 747
CSISO89ASMO449g? - 748
CSISO90g - 749
CSISO92JISC62991984Bg? - 750
CSISO99NAPLPSg? - 751
CSISO103T618BITg? - 752
CSISO111ECMACYRILLICg? - 753
CSISO121CANADIAN1g? - 754
CSISO122CANADIAN2g? - 755
CSISO139CSN369103g? - 756
CSISO141JUSIB1002g? - 757
CSISO143IECP271g? - 758
CSISO150g - 759
CSISO150GREEKCCITTg? - 760
CSISO151CUBAg? - 761
CSISO153GOST1976874g? - 762

CSISO646DANISHg? - 763
CSISO2022CNg? - 764
CSISO2022JPg? - 765
CSISO2022JP2g? - 766
CSISO2022KRg? - 767
CSISO2033g - 768
CSISO5427CYRILLICg? - 769
CSISO5427CYRILLIC1981g? - 770
CSISO5428GREEKg? - 771
CSISO10367BOXg? - 772
CSKSC5636g - 773
CSNATSDANOg - 774
CSNATSSEFIg - 775
CSN_369103g - 776
CSPC8CODEPAGE437g? - 777
CSPC775BALTICg? - 778
CSPCP852g - 779
CSSHIFTJISg - 780
CSUCS4g - 781
CSWINDOWS31Jg? - 782
CUBAg - 783
CWI-2g - 784
CWIg - 785
DEg - 786
DEC-MCSg - 787
DECg - 788
DECMCSg - 789
DIN_66003g - 790
DKg - 791
DS2089g - 792
DS_2089g - 793
E13Bg? - 794
EBCDIC-AT-DE-Ag - 795
EBCDIC-AT-DEg - 796
EBCDIC-BEg - 797
EBCDIC-BRg - 798
EBCDIC-CA-FRg - 799
EBCDIC-CP-AR1g - 800
EBCDIC-CP-AR2g - 801
EBCDIC-CP-BEg - 802
EBCDIC-CP-CAg - 803
EBCDIC-CP-CHg - 804
EBCDIC-CP-DKg - 805
EBCDIC-CP-ESg - 806
EBCDIC-CP-FIg - 807
EBCDIC-CP-FRg - 808
EBCDIC-CP-GBg - 809
EBCDIC-CP-GRg - 810
EBCDIC-CP-HEg - 811
EBCDIC-CP-ISg - 812
EBCDIC-CP-ITg - 813
EBCDIC-CP-NLg - 814
EBCDIC-CP-NOg - 815
EBCDIC-CP-ROECEg - 816
EBCDIC-CP-SEg - 817
EBCDIC-CP-TRg - 818
EBCDIC-CP-USg - 819
EBCDIC-CP-WTg - 820
EBCDIC-CP-YUg - 821
EBCDIC-CYRILLICg - 822
EBCDIC-DK-NO-Ag - 823
EBCDIC-DK-NOg - 824
EBCDIC-ES-Ag - 825
EBCDIC-ES-Sg - 826
EBCDIC-ESg - 827
EBCDIC-FI-SE-Ag - 828
EBCDIC-FI-SEg - 829
EBCDIC-FRg - 830
EBCDIC-GREEKg - 831
EBCDIC-INTg - 832
EBCDIC-INT1g - 833
EBCDIC-IS-FRISSg - 834
EBCDIC-ITg - 835
EBCDIC-JP-Eg - 836
EBCDIC-JP-KANAg - 837
EBCDIC-PTg - 838
EBCDIC-UKg - 839
EBCDIC-USg - 840
EBCDICATDEg - 841
EBCDICATDEAg - 842

EBCDICCAFRg - 843
EBCDICDKNOg - 844
EBCDICDKNOAg - 845
EBCDICESg - 846
EBCDICESAg - 847
EBCDICESSg - 848
EBCDICFISEg - 849
EBCDICFISEAg - 850
EBCDICFRg - 851
EBCDICISFRISSg - 852
EBCDICITg - 853
EBCDICPTg - 854
EBCDICUKg - 855
EBCDICUSg - 856
ECMA-128g - 857
ECMA-CYRILLICg - 858
ECMACYRILLICg - 859
ESg - 860
ES2g - 861
EUC-CNg - 862
EUC-JISX0213g - 863
EUC-JP-MSg - 864
EUC-JPg - 865
EUC-KRg - 866
EUC-TWg - 867
EUCCNg - 868
EUCJP-MSg - 869
EUCJP-OPENg - 870
EUCJP-WINg - 871
EUCJPg - 872
EUCKRg - 873
EUCTWg - 874
FIg - 875
FRg - 876
GBg - 877
GB2312g - 878
GB13000g - 879
GB18030g - 880
GBKg - 881
GB_1988-80g - 882
GB_198880g - 883
GOST_19768-74g - 884
GOST_19768g - 885
GOST_1976874g - 886
GREEK-CCITTg - 887
GREEK7-OLDg - 888
GREEK7g - 889
GREEK7OLDg? - 890
GREEKCCITTg - 891
HUg - 892
IBM-803g - 893
IBM-856g - 894
IBM-901g - 895
IBM-902g - 896
IBM-921g - 897
IBM-922g - 898
IBM-930g - 899
IBM-932g - 900
IBM-933g - 901
IBM-935g - 902
IBM-937g - 903
IBM-939g - 904
IBM-943g - 905
IBM-1008g - 906
IBM-1025g - 907
IBM-1046g - 908
IBM-1047g - 909
IBM-1097g - 910
IBM-1112g - 911
IBM-1122g - 912
IBM-1123g - 913
IBM-1124g - 914
IBM-1129g - 915
IBM-1130g - 916
IBM-1132g - 917
IBM-1133g - 918
IBM-1137g - 919
IBM-1140g - 920
IBM-1141g - 921
IBM-1142g - 922

IBM-1143g - 923
IBM-1144g - 924
IBM-1145g - 925
IBM-1146g - 926
IBM-1147g - 927
IBM-1148g - 928
IBM-1149g - 929
IBM-1153g - 930
IBM-1154g - 931
IBM-1155g - 932
IBM-1156g - 933
IBM-1157g - 934
IBM-1158g - 935
IBM-1160g - 936
IBM-1161g - 937
IBM-1162g - 938
IBM-1163g - 939
IBM-1164g - 940
IBM-1166g - 941
IBM-1167g - 942
IBM-1364g - 943
IBM-1371g - 944
IBM-1388g - 945
IBM-1390g - 946
IBM-1399g - 947
IBM-4517g - 948
IBM-4899g - 949
IBM-4909g - 950
IBM-4971g - 951
IBM-5347g - 952
IBM-9030g - 953
IBM-9066g - 954
IBM-9448g - 955
IBM-12712g - 956
IBM-16804g - 957
IBM037g - 958
IBM038g - 959
IBM256g - 960
IBM273g - 961
IBM274g - 962
IBM275g - 963
IBM277g - 964
IBM278g - 965
IBM280g - 966
IBM281g - 967
IBM284g - 968
IBM285g - 969
IBM290g - 970
IBM297g - 971
IBM420g - 972
IBM423g - 973
IBM424g - 974
IBM437g - 975
IBM500g - 976
IBM775g - 977
IBM803g - 978
IBM813g - 979
IBM848g - 980
IBM851g - 981
IBM852g - 982
IBM855g - 983
IBM856g - 984
IBM857g - 985
IBM860g - 986
IBM861g - 987
IBM863g - 988
IBM864g - 989
IBM865g - 990
IBM866NAVg? - 991
IBM868g - 992
IBM869g - 993
IBM870g - 994
IBM871g - 995
IBM874g - 996
IBM875g - 997
IBM880g - 998
IBM891g - 999
IBM901g - 1000
IBM902g - 1001
IBM903g - 1002

IBM904g - 1003
IBM905g - 1004
IBM912g - 1005
IBM915g - 1006
IBM916g - 1007
IBM918g - 1008
IBM920g - 1009
IBM921g - 1010
IBM922g - 1011
IBM930g - 1012
IBM932g - 1013
IBM933g - 1014
IBM935g - 1015
IBM937g - 1016
IBM939g - 1017
IBM943g - 1018
IBM1004g - 1019
IBM1008g - 1020
IBM1025g - 1021
IBM1026g - 1022
IBM1046g - 1023
IBM1047g - 1024
IBM1089g - 1025
IBM1097g - 1026
IBM1112g - 1027
IBM1122g - 1028
IBM1123g - 1029
IBM1124g - 1030
IBM1129g - 1031
IBM1130g - 1032
IBM1132g - 1033
IBM1133g - 1034
IBM1137g - 1035
IBM1140g - 1036
IBM1141g - 1037
IBM1142g - 1038
IBM1143g - 1039
IBM1144g - 1040
IBM1145g - 1041
IBM1146g - 1042
IBM1147g - 1043
IBM1148g - 1044
IBM1149g - 1045
IBM1153g - 1046
IBM1154g - 1047
IBM1155g - 1048
IBM1156g - 1049
IBM1157g - 1050
IBM1158g - 1051
IBM1160g - 1052
IBM1161g - 1053
IBM1162g - 1054
IBM1163g - 1055
IBM1164g - 1056
IBM1166g - 1057
IBM1167g - 1058
IBM1364g - 1059
IBM1371g - 1060
IBM1388g - 1061
IBM1390g - 1062
IBM1399g - 1063
IBM4517g - 1064
IBM4899g - 1065
IBM4909g - 1066
IBM4971g - 1067
IBM5347g - 1068
IBM9030g - 1069
IBM9066g - 1070
IBM9448g - 1071
IBM12712g - 1072
IBM16804g - 1073
IEC_P27-1g - 1074
IEC_P271g - 1075
INIS-8g - 1076
INIS-CYRILLICg - 1077
INISg - 1078
INIS8g - 1079
INISCYRILLICg - 1080
ISIRI-3342g - 1081
ISIRI3342g - 1082

ISO-2022-CN-EXTg - 1083
ISO-2022-CNg - 1084
ISO-2022-JP-2g - 1085
ISO-2022-JP-3g - 1086
ISO-2022-JPg - 1087
ISO-2022-KRg - 1088
ISO-8859-9g - 1089
ISO-8859-10g - 1090
ISO-8859-11g - 1091
ISO-8859-16g - 1092
ISO-10646g - 1093
ISO-10646/UTF-8/ - 1094
ISO-10646/UTF8/ - 1095
ISO-IR-4g - 1096
ISO-IR-8-1g - 1097
ISO-IR-9-1g - 1098
ISO-IR-10g - 1099
ISO-IR-11g - 1100
ISO-IR-15g - 1101
ISO-IR-16g - 1102
ISO-IR-17g - 1103
ISO-IR-18g - 1104
ISO-IR-19g - 1105
ISO-IR-21g - 1106
ISO-IR-25g - 1107
ISO-IR-27g - 1108
ISO-IR-37g - 1109
ISO-IR-49g - 1110
ISO-IR-50g - 1111
ISO-IR-51g - 1112
ISO-IR-54g - 1113
ISO-IR-55g - 1114
ISO-IR-57g - 1115
ISO-IR-60g - 1116
ISO-IR-61g - 1117
ISO-IR-69g - 1118
ISO-IR-84g - 1119
ISO-IR-85g - 1120
ISO-IR-86g - 1121
ISO-IR-88g - 1122
ISO-IR-89g - 1123
ISO-IR-90g - 1124
ISO-IR-92g - 1125
ISO-IR-98g - 1126
ISO-IR-99g - 1127
ISO-IR-103g - 1128
ISO-IR-111g - 1129
ISO-IR-121g - 1130
ISO-IR-122g - 1131
ISO-IR-127g - 1132
ISO-IR-139g - 1133
ISO-IR-141g - 1134
ISO-IR-143g - 1135
ISO-IR-150g - 1136
ISO-IR-151g - 1137
ISO-IR-153g - 1138
ISO-IR-155g - 1139
ISO-IR-156g - 1140
ISO-IR-166g - 1141
ISO-IR-193g - 1142
ISO-IR-197g - 1143
ISO-IR-209g - 1144
ISO/TR_11548-1/ - 1145
ISO646-CAg - 1146
ISO646-CA2g - 1147
ISO646-CNg - 1148
ISO646-CUg - 1149
ISO646-DEg - 1150
ISO646-DKg - 1151
ISO646-ESg - 1152
ISO646-ES2g - 1153
ISO646-FIg - 1154
ISO646-FRg - 1155
ISO646-FR1g - 1156
ISO646-GBg - 1157
ISO646-HUg - 1158
ISO646-ITg - 1159
ISO646-JP-OCR-Bg - 1160
ISO646-KRg - 1161
ISO646-NOg - 1162

ISO646-NO2g - 1163
ISO646-PTg - 1164
ISO646-PT2g - 1165
ISO646-SEg - 1166
ISO646-SE2g - 1167
ISO646-YUg - 1168
ISO2022CNg? - 1169
ISO2022CNEXTg? - 1170
ISO2022JPg? - 1171
ISO2022JP2g? - 1172
ISO2022KRg? - 1173
ISO6937g - 1174
ISO8859-11g - 1175
ISO11548-1g - 1176
ISO88591g - 1177
ISO88592g - 1178
ISO88593g - 1179
ISO88594g - 1180
ISO88595g - 1181
ISO88596g - 1182
ISO88597g - 1183
ISO88598g - 1184
ISO88599g - 1185
ISO885910g - 1186
ISO885911g - 1187
ISO885913g - 1188
ISO885914g - 1189
ISO885915g - 1190
ISO885916g - 1191
ISO_2033-1983g - 1192
ISO_2033g - 1193
ISO_5427-EXTg - 1194
ISO_5427g - 1195
ISO_5427:1981g - 1196
ISO_5427EXTg - 1197
ISO_5428g - 1198
ISO_5428:1980g - 1199
ISO_6937-2g - 1200
ISO_6937-2:1983g - 1201
ISO_6937g - 1202
ISO_6937:1992g - 1203
ISO_8859-7:2003g - 1204
ISO_8859-16:2001g - 1205
ISO_9036g - 1206
ISO_10367-BOXg - 1207
ISO_10367BOXg - 1208
ISO_11548-1g - 1209
ISO_69372g - 1210
ITg - 1211
JIS_C6229-1984-Bg - 1212
JIS_C62201969ROg - 1213
JIS_C62291984Bg - 1214
JOHABg - 1215
JP-OCR-Bg - 1216
JSg - 1217
JUS_I.B1.002g - 1218
KOI-7g - 1219
KOI-8g - 1220
KOI8g - 1221
KSC5636g - 1222
L10g - 1223
LATIN-9g - 1224
LATIN-GREEK-1g - 1225
LATIN-GREEKg - 1226
LATIN10g - 1227
LATINGREEKg - 1228
LATINGREEK1g - 1229
MAC-CYRILLICg - 1230
MAC-ISg - 1231
MAC-SAMIg - 1232
MAC-UKg - 1233
MACCYRILLICg - 1234
MIKg - 1235
MS-MAC-CYRILLICg - 1236
MS932g - 1237
MS936g - 1238
MSCP949g - 1239
MSCP1361g - 1240
MSMACCYRILLICg - 1241
MSZ_7795.3g - 1242

MS_KANJIg - 1243
NAPLPSg - 1244
NATS-DANOg - 1245
NATS-SEFIg - 1246
NATSDANOg - 1247
NATSSEFIg - 1248
NC_NC0010g - 1249
NC_NC00-10g - 1250
NC_NC00-10:81g - 1251
NF_Z_62-010g - 1252
NF_Z_62-010_(1973)g - 1253
NF_Z_62-010_1973g - 1254
NF_Z_62010g - 1255
NF_Z_62010_1973g - 1256
NOg - 1257
NO2g - 1258
NS_4551-1g - 1259
NS_4551-2g - 1260
NS_45511g - 1261
NS_45512g - 1262
OS2LATIN1g? - 1263
OSF00010001g - 1264
OSF00010002g - 1265
OSF00010003g - 1266
OSF00010004g - 1267
OSF00010005g - 1268
OSF00010006g - 1269
OSF00010007g - 1270
OSF00010008g - 1271
OSF00010009g - 1272
OSF0001000Ag? - 1273
OSF00010020g - 1274
OSF00010100g - 1275
OSF00010101g - 1276
OSF00010102g - 1277
OSF00010104g - 1278
OSF00010105g - 1279
OSF00010106g - 1280
OSF00030010g - 1281
OSF0004000Ag? - 1282
OSF0005000Ag? - 1283
OSF05010001g - 1284
OSF100201A4g? - 1285
OSF100201A8g? - 1286
OSF100201B5g? - 1287
OSF100201F4g? - 1288
OSF100203B5g? - 1289
OSF1002011Cg? - 1290
OSF1002011Dg? - 1291
OSF1002035Dg? - 1292
OSF1002035Eg? - 1293
OSF1002035Fg? - 1294
OSF1002036Bg? - 1295
OSF1002037Bg? - 1296
OSF10010001g - 1297
OSF10020025g - 1298
OSF10020111g - 1299
OSF10020115g - 1300
OSF10020116g - 1301
OSF10020118g - 1302
OSF10020122g - 1303
OSF10020129g - 1304
OSF10020352g - 1305
OSF10020354g - 1306
OSF10020357g - 1307
OSF10020359g - 1308
OSF10020360g - 1309
OSF10020364g - 1310
OSF10020365g - 1311
OSF10020366g - 1312
OSF10020367g - 1313
OSF10020370g - 1314
OSF10020387g - 1315
OSF10020388g - 1316
OSF10020396g - 1317
OSF10020402g - 1318
OSF10020417g - 1319
PTg - 1320
PT2g - 1321
PT154g - 1322

RK1048g - 1323
RUSCIIg - 1324
SEg - 1325
SE2g - 1326
SEN_850200_Bg - 1327
SEN_850200_Cg - 1328
SHIFT-JISg - 1329
SHIFT_JISg - 1330
SHIFT_JISX0213g - 1331
SJIS-OPENg - 1332
SJIS-WINg - 1333
SJISg - 1334
SS636127g - 1335
STRK1048-2002g - 1336
ST_SEV_358-88g - 1337
T.61-8BITg - 1338
T.61g - 1339
T.618BITg - 1340
TS-5881g - 1341
UHCg - 1342
UJISg - 1343
UKg - 1344
UTF8g - 1345
UTF16g - 1346
UTF16BEg? - 1347
UTF16LEg? - 1348
UTF32g - 1349
UTF32BEg? - 1350
UTF32LEg? - 1351
WCHAR_Tg - 1352
WIN-SAMI-2g - 1353
WINDOWS-31Jg - 1354
WINDOWS-936g - 1355
WINSAMI2g - 1356
WS2g - 1357
YUg - 1358

親トピック: ポリシーについて

ポリシー・ルールのアクション

ポリシー・ルールが一致した場合に実行するブロッキング・アクション、アラート・アクション、またはロギング・アクションを定義します。

- ブロッキング・ルール・アクション
  このセクションでは、S-TAP ターミネートおよび S-GATE のルール・アクションについて説明します。
- アラート・ルール・アクション
  アラート・アクションは、1 人以上の受信者に通知を送信します。
- ロギングまたは無視のルール・アクション
  ロギング・アクションは、監視対象トラフィックに基づいてロギングのレベルを制御します。
- 無視アクションについて
  ポリシー・ルールで無視アクションを使用した場合のデータの処理方法を詳しく説明します。
- 全詳細をロギング
  「全詳細をロギング」では、Guardium は、個別の要求ごとにマスクの解除された値を持つデータをログに記録します。「全詳細をロギング」では正確なタイム・スタンプも提供されます。
- 文字セットの設定
  代替文字セットをセッションにアタッチするには、ポリシー抽出ルールでアクションを使用できます。
- ルール定義フィールド
  ポリシー・ルールを定義する際に、以下のフィールドを使用することができます。

親トピック: ポリシー

ブロッキング・ルール・アクション

このセクションでは、S-TAP ターミネートおよび S-GATE のルール・アクションについて説明します。

S-TAP ターミネート

S-TAP ターミネート・アクションは、データベース接続 (セッション) を終了し、そのセッションでの追加の要求をブロックします。このアクションは、S-GATE が使用されているかどうかに関わらず、S-TAP で使用可能です。

注: S-TAP ターミネートを使用すると、起動している要求は通常ブロックされませんが、そのセッションからの追加の要求はブロックされます。リクエスト・レートが高い場合、複数の要求がセッション終了前に通過する場合があります。

S-GATE

S-GATE は、ネットワーク接続とローカル接続の両方に関して、S-TAP を通じたデータベース保護を提供します。S-GATE が有効な場合、すべてのデータベース接続 (セッション) は評価され、以下の S-GATE モードのいずれか 1 つでモニターされるようにタグ付けされます。

- 接続 (S-GATE は「オン」): S-TAP は、そのセッションに対してファイアウォール・モードになります。このモードでは、データベース要求は保留され、要求ごとに判定を待機してからその応答をリリースします。このモードでは、待ち時間が想定されます。ただし、問題要求は確実にブロックされます。
- 切断 (S-GATE は「オフ」): S-TAP は、そのセッションに対して通常のモニター・モードになります。このモードでは、遅延なしで要求がデータベース・サーバーに渡されます。このモードでは、待ち時間は想定されません。

S-TAP 自体の S-GATE 構成は、すべてのセッションにデフォルト S-GATE モードを定義するほか、コレクターが応答しない場合の S-GATE 判定に関連するその他のデフォルトを定義します。詳しくは、Linux システムおよび UNIX システム: S-TAP ファイアウォール・パラメーターおよび Windows: S-TAP ファイアウォール・パラメーターを参照してください。
以下の S-GATE ポリシー・ルール・アクションを使用して、デフォルト S-GATE 構成をリアルタイムで変更することができます。

- S-GATE アタッチ: 特定のセッションに対して S-GATE モードを「接続」に設定します。そのセッションのトラフィックを慎重に監視 (そして必要に応じてブロック) する必要が生じる特定の基準が満たされた場合の使用が想定されています。
- S-GATE デタッチ: 特定のセッションに対して S-GATE モードを「切断」に設定します。S-GATE デタッチは、安全と見なされるセッションや、待ち時間が許容されないセッションでの使用が想定されています。
- S-GATE ターミネート: セッションが接続されている場合にのみ適用されます。S-GATE ターミネートは、ファイアウォール保護された要求の応答をドロップし、同じデータベースのセッションを終了させます。S-GATE ターミネート・ポリシー・ルール・アクションは、その前に監視されていたセッションを終了させます。

注意:

- S-TAP および S-GATE ターミネート・アクションは、ワイルドカード文字のあるメンバーを含むクライアント IP グループでは動作しません。S-TAP および S-GATE ターミネートは単一の IP アドレスでのみ動作します。お客様が複数の IP エントリーを使用することを希望する場合は、ワイルドカードをグループで処理する必要があります。お客様は、ポリシーでビジネス・ニーズに対応するために、信頼できるユーザー/クライアントのグループ、または信頼できないユーザー/クライアントのグループを作成できます。
- 古い Linux カーネルで S-GATE を A-TAP と使用する場合には制限事項があります。S-TAP V10.1.2 以上では、2.6.36 より前のカーネルおよび A-TAP を使用する Linux を除き、どのような場合でも S-GATE がサポートされます。
- MySQL データベースの場合、デフォルトのコマンド行接続は `mysql -u <user> -p <pass> <dbname>` です。このモードでは、MySQL は最初にこのデータベース内のすべてのオブジェクトおよびフィールドをマップして Tab キーによるオートコンプリートをサポートします。このマッピングに関与するオブジェクトまたはフィールドに終了ルールが指定されている場合、それにより接続セッションが即時に無効になります。これを防ぐには、`-A` フラグ (オートコンプリート機能を無効にして、終了ルールを起動しない) を指定して MySQL に接続します。もう 1 つのオプションとして、ルールを適切に調整し、これらのオブジェクトまたはフィールドに対するいかなるアクセスにおいても終了せず、代わりにより絞り込まれた基準を定義し、ログイン・シーケンスでルールが起動されないようにする方法があります。

親トピック: ポリシー・ルールのアクション

アラート・ルール・アクション

アラート・アクションは、1 人以上の受信者に通知を送信します。

アラート・アクションごとに複数の通知を送信することが可能です。その通知は、以下の通知タイプを 1 つ以上組み合わせたものであってもかまいません。

- E メール・メッセージ: E メール・メッセージは、Guardium ユーザーにアドレス指定する必要があり、Guardium システム用に構成された SMTP サーバーを使用して送信されます。E メール通知の追加の受信者は、起動者 (ポリシー起動の原因となった実際の SQL コマンドを開始したユーザー) と所有者 (データベースの所有者) です。起動者と所有者は、Guardium API を通じて構成されたユーザー ID (IP ベース) を取得することによって識別されます。これらのマッピングを表示するには、`accessmgr` としてログインし、「データ・セキュリティー」 > 「ユーザー - データベース関連付け」にナビゲートするか、API コマンド list_db_user_mapping を使用します。
- SNMP トラップ: Guardium システム用に構成されたトラップ・コミュニティーにアラートを送信します。
- syslog メッセージ: syslog に書き込まれるメッセージを生成します。
  重要: `%%RecordsAffected` 変数は、`syslog` 通知タイプが指定された「アラートのみ」ルール・アクションのメッセージ・テンプレートで使用される場合は、値を返しません。
- カスタム通知: Java クラスとして実装されるユーザー作成の通知ハンドラー。

重要: アラート定義および通知は、データ・レベル・セキュリティーの影響を受けません。その理由は、アラートがユーザーとの関連で評価されないこと、アラートが複数のユーザーに関連付けられたデータベースに関連している可能性があること、そのアラート通知の受信者が 1 人もいないという状態を回避するためです。

アラート・メッセージ

アラートの内容はメッセージ・テンプレートによって定義されます。 「設定」 > 「グローバル・プロファイル」にナビゲートして、「名前付きテンプレート」フィールドを見つけ、「編集」ボタンをクリックします。「名前付きテンプレート・ファインダー」を使用して、メッセージ・テンプレートの作成、レビュー、および変更を行います。

アラートの動作

アラート・アクションには、以下を含む複数のタイプがあります。

- 毎日アラート: 毎日、ルールの初回マッチング時にのみ通知を送信します。
- セッションごとに 1 回アラート: ルールがマッチングしたセッションごとに 1 回のみ通知を送信します。このアクションは、特定のイベント発生の通知を受けるが、単一セッションの間にそのイベントのインスタンスごとに通知を受ける必要がない場合などに適しています。例えば、特定の機密オブジェクト更新時に通知を送信するとしても、プログラムが単一セッションの間にそのオブジェクトの数千個のインスタンスを更新する場合、アラート受信者に数千もの通知が送信されることは望ましくないはずです。
- アラートのみ: syslog 通知タイプで「アラートのみ」が使用される場合、メッセージは、/var/log/messages に直接送信されます。その他の通知タイプの場合、「アラートのみ」を使用すると、メッセージは MESSAGE 表に送信されます。 「アラートのみ」は、ポリシー違反を通知しません。
  重要: `%%RecordsAffected` 変数は、`syslog` 通知タイプが指定された「アラートのみ」ルール・アクションのメッセージ・テンプレートで使用される場合は、値を返しません。
- 一致ごとにアラート: ルールが満たされるごとに通知を送信します。これは、発生するごとに注意を要する条件に適しています。
- 時間間隔ごとにアラート: ログ細分度期間ごとに 1 回通知を送信します。例えば、ログ細分度が 1 時間に設定された場合、通知は毎時ルールへの最初の一致時にのみ送信されます。
  ヒント: Guardium 管理者は、「管理」 > 「アクティビティー・モニター」 > 「検査エンジン」ツールを使用してログ細分度を設定します。

ロギングまたは無視のルール・アクション

ロギング・アクションは、監視対象トラフィックに基づいてロギングのレベルを制御します。

アクセス・ルール、例外ルール、および抽出ルールは、使用できるアクションの点で異なります。例えば、ロギング・アクションおよび無視アクションは大半のポリシーで使用できますが、監査のみアクションは、「選択的な監査証跡」設定を使用するポリシーでのみ使用できます。

監査のみ
: このアクションは、「選択的な監査証跡」設定を使用するポリシーでのみ使用できます。

  監査のみでは、ルールを起動した構成体をログに記録します。選択的な監査証跡ポリシーでは、デフォルトでは構成体はログに記録されないため、この選択を使用してログに記録される構成体を指定します。例えば、アプリケーション・イベント API を使用している場合、データベース・ユーザー名の情報をレポートで使用できるようにするには、このアクションを使用してデータベース・ユーザー名のロギングを強制する必要があります (そうしないと、この例ではユーザー名はブランクになります)。

許可
: 一致する場合、一致する構成体をログに記録しますが、ポリシー違反はログに記録しません。「許可」アクションを選択した場合、他のアクションをルールに追加することはできません。

ロギングのみ
: ポリシー違反をログに記録します。許可アクションを例外として、ルール・アクションがロギングを抑止している場合を除いて、ポリシー違反はルールが起動されるたびにログに記録されます。

マスクされた詳細をロギング
: このアクションは、アクセス・ルールおよび抽出ルールで使用可能です。

  要求の全 SQL を、値を疑問符 (?) に置き換えてログに記録します。

全詳細をロギング
: 要求の全 SQL 文字列と正確なタイム・スタンプをログに記録します。

値を含む全詳細をロギング
: 「全詳細をロギング」と似ていますが、値が解析されてデータベース内の個別の表に記録されるように、各値が個別のエレメントとして保管されます。このロギング・アクションでは、関連するコマンドの具体的な値もログに記録されるため、より多くのシステム・リソースが使用されます。このロギング・アクションは、これらの値について具体的な条件を持つレポートを生成する必要がある場合にだけ使用してください。
  重要: このロギング・アクションは、技術サービスに確認の上で使用可能になります。

セッションごとに全詳細をロギング
: この要求およびセッションの残りについて、全 SQL 文字列と正確なタイム・スタンプをログに記録します。

セッションごとに値を含む全詳細をロギング
: 「値を含む全詳細をロギング」および「セッションごとに全詳細をロギング」の説明を参照してください。
  重要: このロギング・アクションは、技術サービスに確認の上で使用可能になります。

ロギングをスキップ
: 一致する場合、ポリシー違反はログに記録されず、構成体のロギングが停止します。これは許可アクションに似ていますが、それに加えて、構成体のロギングを停止します。このアクションは、重要性がないことが認識されている要求の構成体のロギングを除外するために使用します。この機能は、データベース・エラー・コードのみに関連する例外ルールにも適用され、アプリケーションが大量のエラーを生成することが不可避の場合に、ユーザーがエラーをログに記録しないことを許可します。
  注: ルールが適用される前に構成体の解析およびロギングが行われても構成体がセッションに組み込まれることはないために、GDM_CONSTRUCT がログに記録される場合があります。

セッションごとに応答を無視
: セッションの残りに対する応答が無視されます。 このアクションはポリシー違反をロギングしませんが、セッションの残りに対する応答の分析を停止します。 この動作は、以後のデータベース応答が重要でないことがわかっている場合に役立ちます。このアクションは、S-TAP からデータをスニッフィングする場合には機能しますが、SPAN ポートからデータをスニッフィングする場合には機能しません。
  注: 「セッションごとに応答を無視」の場合、スニファーは照会に対する応答を 1 件も受信しないか、応答が無視されるため、COUNT_FAILED および SUCCESS の値は表のデフォルトが何であっても、この場合は COUNT_FAILED=0 および SUCCESS=1 です。

セッションを無視
: 現行の要求およびセッションの残りが無視されます。このアクションは、ポリシー違反をログに記録しませんが、構造のロギングを停止し、そのセッションの残りのいずれのタイプのポリシー違反もテストしません。このアクションは、データベースにテスト領域が含まれ、データベースのその領域に対してポリシー・ルールを適用する必要がない場合などに役立ちます。「セッションを無視」ルールを適用すると、個々のセッションからのアクティビティーが S-TAP によって削除されるか、スニファーによって完全に無視されます。
  重要: ただし、セッションが無視されても、接続 (ログイン/ログアウト) の情報は、常にログに記録されます。

S-TAP セッションを無視
: 現行の要求および S-TAP セッションの残りが無視されます。このアクションは、大量のネットワーク・トラフィックを生成する特定のシステム、ユーザー、またはアプリケーションの指定と組み合わせて実行されます。このアクションは、S-TAP セッションからの以後のデータベース応答が重要でないことがわかっている場合に役立ちます。「S-TAP セッションを無視」は次のように使用します。

  - IGNORE_ENTIRE_STAP_SESSION: 「ハード型」の無視で、取り消すことができません。
  - IGNORE_STAP_SESSION (REVOCABLE): 「ソフト型」の無視で、このルール・アクションでは、データベースへの新しい接続を必要とせずにセッション・トラフィックを再送信できます。
  - 無視の取り消し - アクション IGNORE_S-TAP_SESSION (REVOCABLE) によって無視されていたセッションが再開します。つまり、「無視の取り消し」コマンドが S-TAP によって受信された後、トラフィックが Guardium システムに送信されます。このコマンドは、「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」ページから「無視されたセッションをすべて取り消す」ボタンを使用して送信します。
    注: 「無視の取り消し」コマンドは、1 つのスニファー・プロセスで S-TAP ホストに対して保持されます。S-TAP が無視の取り消し状態になった後に開かれた新しいセッションは、ルール IGNORE_STAP_SESSION (REVOCABLE) が起動されたとしても無視されません。

セッションごとに SQL を無視
: セッションの残りの部分に関する SQL はログに記録されません。例外は引き続きログに記録されますが、システムは、その例外に対応する SQL 文字列をキャプチャーしない場合があります。

抽出カウンターのロギング

このアクションは、抽出ルールでのみ使用できます。

カウンターを更新しますが、戻りデータをログに記録しません。このアクションは、カウンター値が戻り値よりも重要である場合に、ディスク・スペースを節約します。

マスクされた抽出カウンターのロギング
このアクションは、抽出ルールでのみ使用できます。

カウンターを更新し、値を疑問符 (?) に置き換えて SQL 要求をログに記録します。このアクションは、戻り値をログに記録しません。

隔離
このアクションは、アクセス・ルール、例外ルール、および抽出ルールで使用できます。

特定の期間中に、同一ユーザーが同一サーバーにログインすることを防止します。隔離ルール・アクションを使用するには、「検疫時間 (分)」設定を使用して隔離の期間を指定する必要があります。セッションが監視されている (S-GATE) 場合、アクションはドロップ判定を送信します。セッションが監視されていない (S-TAP ターミネート) 場合、アクションは S-TAP にセッションを停止させます。

隔離のタイム・スタンプは、現在時刻を取得して、リセット間隔フィールドから得た分数を加算することで計算されます。この新しいタイム・スタンプは、サーバー IP、サーバー・タイプ、データベース・ユーザー名、サービス名、およびセッションが監視されているかどうかを示すフラグとともに (タイム・スタンプでソートされた) 新しい構成で保持されます。

解析なし
SQL ステートメントは解析されません。
クイック解析
このアクションは、アクセス・ルール用です。

セッションの残りの部分では SQL ステートメントの解析は行いません。これにより、構文解析時間が削減されます。このモードでは、アクセスを受けるすべてのオブジェクトを判別可能 (オブジェクトは WHERE 節よりも前に出現するため) ですが、正確なオブジェクト・インスタンスは (WHERE 節によって判別されるため) 不明です。

クイック解析フィールドなし
SQL ステートメント内のフィールドは解析されません。クイック解析ルールは、SQL 文字列が 100 文字を超える場合にのみ適用されます。
クイック解析ネイティブ
このアクションは Guardium S-TAP for Db2 on z/OS でのみ使用されます。

このルール・アクションは、大量のトラフィックのために Guardium スニファーが過負荷になっている環境でパフォーマンスを向上させるために使用します。

編集
このアクションは、抽出ルール用です。
レポート内の特定ユーザーのデータベース照会出力の一部分 (例えば、クレジット・カード番号) にマスクを掛けます。マスク文字は、抽出ルールの「データ・パターン」パラメーターの「置換文字」によって定義されます。抽出ルールによって生成された出力が「データ・パターン」の正規表現に一致する場合は、括弧「(および) 」で囲まれたサブ表現に一致する部分がマスク文字に置き換えられます。 事前定義した正規表現も使用可能です。詳しくは、ルール定義フィールドの『データ・パターン』を参照してください。
制約事項:

- S-TAP ライブ・アップグレード後のオープン・セッションでは、編集は機能しません。
- フィールドと数値型を指定して作成されたテーブルでは、編集は機能しません。
- SQL パターンは、編集ルールではサポートされません。
- 編集は ANSI 文字セットに対してのみサポートされます。
- 編集ルールは、SQL レベル (OBJECT_NAME や VERB などの属性) ではなく、セッション・レベル (IP アドレスやユーザーなどの属性) で設定する必要があります。修正を必要とする SQL で編集ルールを設定すると、修正命令が S-TAP に到達するまでに数ミリ秒の時間がかかってしまい、一部の結果がマスクを掛けられずに通過する可能性があります。

  すべての SQL が確実に編集されるようにするには、guard_tap.ini ファイルを使用して、すべてのセッションの S-TAP のデフォルト S-GATE モードを「接続」に設定します。 こうすることで、通過するすべてのコマンドがルール・エンジンの検査を受けることになり、各要求が保留されて、その要求に関するポリシーの判断を待機するようになります。これをデプロイすると若干の待ち時間が必要になりますが、100% 確実にデータを修正できます。

  Informix データベースの場合、データ型として char が使用されると、各列の終わりが Null で終了されません。sendmsg システム呼び出しでは、4 つのすべての列が 1 つの列としてキャプチャーされ、K-TAP はキャプチャーしたデータが何であろうと、そのデータの編集を試みます。これは、編集と Informix データベースを使用する際の制限事項です。

値を別個に記録する
値を別個に記録しない
このアクションはセッション・ベースのアクセス・ルールです。トランザクション間の区別を行うために、リプレイ機能で使用されます。
自動コミット・オンのマークを付ける
自動コミット・オフのマークを付ける
このアクションはセッション・ベースのアクセス・ルールです。異なるデータベースでは自動コミット・モデルが各種あるため、リプレイ機能で使用されます。
z/OS 監査
このアクションは、データ・セット、Db2、および IMS のコレクション・プロファイル・ポリシー・ルールでのみ使用されます。これらのルールは、z/OS サーバーで収集するトラフィックを指定します。

フィルター条件を満たすトラフィックはコレクターに送信されます。これが、コレクション・プロファイル・ルールで指定できる唯一のアクションです。

HTTP の制約事項

以下のポリシー・アクションは HTTP ではサポートされません。

- S-TAP ターミネート
- ロギングをスキップ

その他のアクションについては、以下のものは HTTP ではサポートされません。

- 「セッションごとに応答を無視」: HTTP が例外と抽出をサポートしていません。
- 「セッションごとに SQL を無視」: HTTP に SQL が含まれていません。
- 「隔離」: このアクションはユーザー隔離に使用されますが、HTTP はデータベース・ユーザーおよびオペレーティング・システム・ユーザーをサポートしていません。
- 「クイック解析」: このアクションはログ SQL 用です。
- S-GATE ターミネート: このアクションは Hadoop ではサポートされません。どのターミネート・アクションも HTTP では機能しません。

以下のポリシー条件は HTTP ではサポートされません。

- クライアント MAC
- データベース名
- データベース・ユーザー
- アプリケーション・ユーザー
- OS ユーザー
- ソース・アプリケーション
- マスキング・パターン
- 置換文字
- 検疫時間 (分)
- 影響を受けるレコードしきい値
- XML パターン
- イベント・タイプ
- イベント・ユーザー名
- アプリケーション・イベント値テキスト
- アプリケーション・イベント値テキスト・グループ
- アプリケーション・イベント値テキストおよびグループ
- 数値
- 日付

親トピック: ポリシー・ルールのアクション

無視アクションについて

ポリシー・ルールで無視アクションを使用した場合のデータの処理方法を詳しく説明します。

セッションを無視
: 現行の要求およびセッションの残りが無視されます。 このアクションは、ポリシー違反をログに記録しませんが、構造のロギングを停止し、そのセッションの残りのいずれのタイプのポリシー違反もテストしません。このアクションは、例えば、データベースにテスト領域が含まれ、データベースのその領域に対してポリシー・ルールを適用する必要がない場合などに役立ちます。

表 1. セッションを無視

クライアントおよび DB サーバー/S-TAP 間でログに記録または無視されるデータ	DB サーバー/S-TAP からコレクターに送られるデータ	スパン・ポート/ネットワーク TAP からコレクターへのデータ
無視 - SQL コマンド、SQL エラー、 結果セット	ログイン/ログアウト スニファーから S-TAP - S-TAP への 1 シグナルによりこのセッションの送信アクティビティーを停止。 追加のアクティビティーが S-TAP により送られる場合、これはスニファー・レベルでのみ無視される。 SQL コマンドを無視 SQL エラーを無視 結果セットを無視	無視 – SQL コマンド、SQL エラー、 結果セット。 スパン・ポート/ネットワーク TAP からの SQL コマンドとエラーは、 スニファーでフィルター処理される。

S-TAP セッションを無視
: 現行の要求および S-TAP セッションの残りが無視されます。 このアクションは、大量のネットワーク・トラフィックを生成する特定のマシン、ユーザー、またはアプリケーションのポリシー・ビルダー・メニュー画面での指定と組み合わせて実行されます。 このアクションは、S-TAP セッションからの以後のデータベース応答が重要でないことがわかっている場合に役立ちます。

表 2. S-TAP セッションを無視

クライアントおよび DB サーバー/S-TAP 間でログに記録または無視されるデータ	DB サーバー/S-TAP からコレクターに送られるデータ	スパン・ポート/ネットワーク TAP からコレクターへのデータ
無視 - SQL コマンド、SQL エラー、 結果セット	スニファーから S-TAP へのログイン/ログアウト - S-TAP への 1 シグナルによりこのセッションの送信アクティビティーを停止。 S-TAP に追加のシグナルが送られると、このセッションへの送信アクティビティーを停止。	適用外 スパン・ポート/ネットワーク TAP からのトラフィックを無視する必要がある場合は、 代わりに「セッションを無視」を使用する。

セッションごとに応答を無視
: セッションの残りに対する応答が無視されます。 このアクションは、ポリシー違反をログに記録しますが、セッションの残りの部分に対する応答の分析を停止します。このアクションは、以後のデータベース応答が重要でないことがわかっている場合に役立ちます。
注: 「セッションごとに応答を無視」の場合、スニファーは照会に対する応答を 1 件も受信しないか、応答が無視されるため、COUNT_FAILED および SUCCESS の値は表のデフォルトが何であっても、この場合は COUNT_FAILED=0 および SUCCESS=1 です。

表 3. セッションごとに応答を無視

クライアントおよび DB サーバー/S-TAP 間でログに記録または無視されるデータ	DB サーバー/S-TAP からコレクターに送られるデータ	スパン・ポート/ネットワーク TAP からコレクターへのデータ
ログ – SQL コマンド、無視 - SQL エラー、結果セット	スニファーから S-TAP へのログイン/ログアウト SQL コマンド - S-TAP への 1 シグナルによりこのセッションの送信アクティビティーを停止。 S-TAP に追加のシグナルが送られると、このセッションへの送信アクティビティーを停止。	適用外 このルール・アクションは S-TAP のみの実装。

セッションごとに SQL を無視

セッションの残りの部分に関する SQL はログに記録されません。例外は引き続きログに記録されますが、システムは、その例外に対応する SQL 文字列をキャプチャーしない場合があります。

表 4. セッションごとに SQL を無視

クライアントおよび DB サーバー/S-TAP 間でログに記録または無視されるデータ	DB サーバー/S-TAP からコレクターに送られるデータ	スパン・ポート/ネットワーク TAP からコレクターへのデータ
無視 - SQL コマンド ログ - SQL エラー、 結果セット	ログイン/ログアウト スニファーから S-TAP - S-TAP への 1 シグナルによりこのセッションの送信アクティビティーを停止。 追加のアクティビティーが S-TAP により送られる場合、これはスニファー・レベルでのみ無視される。 SQL コマンドをログ SQL エラーをログ 結果セットをログ - 抽出ルール使用の場合	無視 - SQL コマンド ログ - SQL エラー、 結果セット SQL コマンドはスニファーでフィルターされる。

選択的な監査証跡

「選択的な監査証跡」ポリシーを使用して、アプライアンスでのロギングの量を制限します。 これは、検査エンジンで受信されているトラフィックのうち重要なトラフィックの割合が比較的小さい場合や、レポート対象となり得るすべてのトラフィックが完全に識別可能である場合に適しています。

選択的な監査証跡を使用している場合であっても、ポリシーに「セッションを無視」ルールを組み込むことが引き続き非常に重要である点に十分注意してください。「セッションを無視」ルールを使用すると、コレクターの負荷を大幅に削減できます。これは、S-TAP レベルで情報をフィルター操作することにより、コレクターがその情報を受信することがなくなり、最終的にログに記録されることのないトラフィックの分析にリソースを消費する必要がなくなるためです。「セッションを無視」ルールが指定されていない「選択的な監査証跡」ポリシーは、すべてのトラフィックがデータベース・サーバーからコレクターに送信され、コレクターがデータベース・サーバーの生成したすべてのコマンドおよび結果セットを分析することになることを意味します。

表 5. 選択的な監査証跡

クライアントおよび DB サーバー/S-TAP 間でログに記録または無視されるデータ	DB サーバー/S-TAP からコレクターに送られるデータ	スパン・ポート/ネットワーク TAP からコレクターへのデータ
無視 - SQL コマンド ログ - SQL エラー、 結果セット	ログイン/ログアウト SQL コマンドを無視 (「監査のみ」ルールまたは「全詳細をロギング」ルールで定義されているものを除く)。 SQL エラーをログ 結果セットをログ - 抽出ルール使用の場合	無視 - SQL コマンド ログ - SQL エラー、 結果セット SQL コマンドはスニファーでフィルターされる。

親トピック: ポリシー・ルールのアクション

全詳細をロギング

「全詳細をロギング」では、Guardium は、個別の要求ごとにマスクの解除された値を持つデータをログに記録します。「全詳細をロギング」では正確なタイム・スタンプも提供されます。

デフォルトでは、Guardium® コレクターは、SQL 文字列をロギングするときに、すべての値にマスクを掛けます。以下に例を示します。

```
insert into tableA (name,ssn,ccn) values ('Bob Jones', '429-29-2921','29249449494949494')
```

この場合、ログには次の値が記録されます: `insert into tableA (name,ssn,ccn) values (?, ?,?)`. これは、次の 2 つの理由から、デフォルトの振る舞いになります。

1. 値は、機密情報を含んでいることがあるため、デフォルトでログに記録するべきではない。
2. 値なしのロギングにより、アプライアンス内部におけるシステム・パフォーマンスとデータ保存に要する時間が増大する可能性がある。データベース・トラフィックには、1 時間に数百、数千、あるいは数億回繰り返される値以外はすべて同一の大量の SQL 要求が含まれていることがよくあります。 Guardium は、値をマスキングすることにより、これらの繰り返される SQL 要求を「構成体」と呼ばれる 1 つの要求に統合することができます。個々の SQL 要求/構成体をそれぞれ個別にログに記録する代わりに、構成体をログに記録すると、構成体が実行された回数のカウンターとともに、毎時 (セッションごとに) 1 回だけログへの記録が行われます。こうすることで、データベース内に数百 (あるいは数億) の行が作成される代わりに、新しい行が 1 つだけ追加されるので、ディスク・スペースの大幅な節約になります。

「全詳細をロギング」では、Guardium はマスクの解除された値および各個別の要求とともにデータをログに記録します。「全詳細をロギング」では正確なタイム・スタンプも提供されます。一方、詳細なしのロギングでは、ログ細分度期間 (通常は 1 時間) 内における構成体の最新のタイム・スタンプが提供されます。

S-TAP® セッションを無視 - 「S-TAP セッションを無視」が起因となり、コレクターは S-TAP に特定のセッションに対するログアウト通知を除くすべてのトラフィックの送信を停止するよう指示するシグナルを送信します。`where DBUserName?=scott, Ignore S-TAP Session` というルールがある場合の例を以下に示します。

- Scott がデータベース・サーバーにログインすると、S-TAP はコレクターに接続情報を送信します。
- コレクターは、接続をログに記録します。セッション情報 (ログイン/ログアウト) は、常にログに記録されます。
- コレクターは、この特定のセッションからのトラフィックの送信を以後停止するように S-TAP にシグナルを送信します。これは、Scott がデータベース・サーバーに対して実行するすべてのコマンド、およびデータベース・サーバーから Scott 宛てに送信されるすべての応答 (結果セット、SQL エラー、その他) が S-TAP によって廃棄され、以後コレクターに到着しなくなることを意味します。
- Scott がデータベース・サーバーからログアウトすると、S-TAP はこの情報をコレクターに送信します (ログイン/ログアウト情報は、セッションが無視される場合であっても常に記録されます)。
- Scott が再度ログインすると、これらのステップが繰り返されます。どのセッションを無視するかのロジックは、S-TAP ではなくコレクターが保守します。

選択的な監査証跡を使用している場合であっても、ポリシーに「セッションを無視」ルールを組み込むことが引き続き非常に重要である点に十分注意してください。「セッションを無視」ルールを使用すると、コレクターの負荷を大幅に削減できます。これは、S-TAP レベルで情報をフィルター操作することにより、コレクターがその情報を受信することがなくなり、最終的にログに記録されることのないトラフィックの分析にリソースを消費する必要がなくなるためです。「セッションを無視」ルールが指定されていない「選択的な監査証跡」ポリシーは、すべてのトラフィックがデータベース・サーバーからコレクターに送信され、コレクターがデータベース・サーバーの生成したすべてのコマンドおよび結果セットを分析することになることを意味します。

MS-SQL または Sybase バッチ・ステートメントの使用

MS-SQL または Sybase バッチ・ステートメント内の SQL コマンドの成功または失敗が正しく表示されない場合があるという制限事項があります。

MS-SQL または Sybase SQL バッチ・ステートメントは、複雑なプロシージャーを作成する場合に主に使用されます。SQL ステートメントを個別に実行すると、各ステートメントの状況は個別に追跡され、正しい成功値または失敗値が示されます。ただし、(MS-SQL または Sybase で使用される) SQL ステートメントのバッチがまとめて実行されると、返される状況は、バッチ内の最後のトランザクションの単一の状況となります。

例:

```
[SQL バッチの開始]
SQL 1 ステートメント - 失敗 (SQL 1 statement - failed)
SQL 2 ステートメント - 失敗 (SQL 2 statement - failed)
SQL 3 ステートメント - 成功 (SQL 3 statement - success)
[SQL バッチの終了]
```

Guardium アプリケーションでは、MS-SQL または Sybase バッチ・ステートメントで、最後の SQL ステートメントの成功または失敗のみがレポートされます。この例では、SQL 1 および SQL 2 は失敗していますが、MS-SQL または Sybase バッチ・ステートメントは成功とレポートされます。

親トピック: ポリシー・ルールのアクション

文字セットの設定

代替文字セットをセッションにアタッチするには、ポリシー抽出ルールでアクションを使用できます。

抽出ルールの例 (hint を使用):

文字セット EUC-JP (コード 274)。

抽出ルール・パターン: `guardium://char_set?hint=274`

結果として、抽出ルールはセッションにアタッチされ、他の文字セットがない場合、アナライザーはセッションで EUC-JP を使用します。

抽出ルールの例 (force を使用):

文字セット EUC-JP (コード 274)。

抽出ルール・パターン: `guardium://char_set?force=274`

結果として、抽出ルールはセッションにアタッチされ、アナライザーはどのような場合でもセッションで EUC-JP 文字セットを使用します。前に使用されていた文字セットの代わりに EUC-JP が使用されます。

抽出ルールは、通常、少し遅れてセッションにアタッチされるので注意してください。したがって、短いセッションやセッションの最初の方では、文字セットの変更が直ちに反映されない場合があります。スキーマは、Oracle、Sybase、MY SQL、および MS SQL に適用されます。

親トピック: ポリシー・ルールのアクション

ルール定義フィールド

ポリシー・ルールを定義する際に、以下のフィールドを使用することができます。

表 1. ルール定義フィールドの参照表

フィールド	記述
アクション	ルールが真の場合に実行されるアクションを示します。すべてのルール・アクションの総合的な説明については、『ルール・アクションの概要』を参照してください。
アプリケーション・イベントの存在	アプリケーション・イベントのみをマッチングします。『アプリケーション・イベントの注』を参照してください。
アプリケーション・イベントの値	指定されたアプリケーション・イベントの「テキスト」、「数値」または「日付」の値がマッチングされます。 オプションで、イベント文字列として「グループ」を選択することもできます。『アプリケーション・イベントの注』を参照してください。

フィールド	記述
(アプリケーション) イベント・タイプ	指定されたアプリケーション・イベントをマッチングします。『アプリケーション・イベントの注』を参照してください。
(アプリケーション) イベント・ユーザー名	指定されたアプリケーション・イベント・ユーザー名のみをマッチングします。『アプリケーション・イベントの注』を参照してください。
アプリケーション・イベントの注	アプリケーション・イベント・フィールドは、「未解析ログ」ボックスにマークが付いている場合は使用できません。
アプリケーション・ユーザー	アプリケーション・ユーザー。「ルールでの値および/またはグループ値の指定」を参照してください。
カテゴリー	レポート目的でポリシー違反をグループ化するために使用可能な、任意のラベル。デフォルト・カテゴリーは、ポリシー定義で指定できますが、このデフォルトは各ルールでオーバーライドされます。
分類	レポート目的でポリシー違反をグループ化するために使用可能な、任意のラベル。デフォルトの分類は、ポリシー定義で指定できますが、このデフォルトは各ルールでオーバーライドされます。
クライアント情報	DB2® クライアント情報: アクセス・ルールのみ。z/OS® のみ。DB_TYPE が Db2、Db2 COLLECTION Profile、VSAM COLLECTION Profile のいずれかの場合は、CLIENT INFO フィールド (および CLIENT_INFO_GROUP_ID) が表示されます。 このフィールドに入力できる情報のタイプは USER=x; WKSTN=y; APPL=z です。
クライアント IP	含める場合は「Not」ボックスをクリアし、除外する場合は「Not」ボックスにマークを付けます。 • 任意のクライアント: すべてのクライアント・フィールドをブランクのままにします。カウントは任意のクライアントがルールを満たすごとに増分されます。(「Not」ボックスにマークが付けられている場合は、すべてのフィールドをブランクのままにすることはできません。) • IP アドレスとマスクによって選択されたすべてのクライアント: 最初のボックスにクライアント IP アドレスを、2 番目のボックスにネットワーク・マスクを入力します。カウントは、指定された任意のクライアントがルールを満たすごとに増分されます。例えば、サブネット 192.168.9.x のすべてのクライアントを選択するには、最初のボックスに 192.168.9.1 を、2 番目のボックスに 255.255.255.0 を入力します。IP アドレスの選択について詳しくは、『マスクを使用した IP アドレスの選択』を参照してください。 • クライアントのグループ: クライアント IP アドレスのグループを「グループ」ドロップダウン・リストから選択するか、「グループ」ボタンをクリックして新規グループを定義し、そのグループを選択します。カウントは、選択したグループの任意のメンバーがルールを満たすごとに増分されます。 • IP アドレスとマスク、およびクライアントのグループによって選択されたすべてのクライアント: 「クライアント IP」および「グループ」フィールドの両方を使用します。カウントは、いずれかの方法を使用して指定された任意のクライアントがルールを満たすごとに増分されます。 IP アドレスでのワイルドカードの使用が可能になります。 クライアント IP グループには、ワイルドカード % をポリシー内で使用できます。
クライアント IP/ソース・プログラム/データベース・ユーザー/サーバー IP/サービス名	7 タプル・グループ - クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名/OS ユーザー/データベース アクセス、例外、および抽出の各ルールでは、5 タプルのグループ・タイプが使用可能です。 タプルでは、複数の属性を組み合わせて 1 つのグループ・メンバーを形成することができます。 タプルには、1 つのスラッシュおよび 1 つのワイルドカード文字 (%) を使用できます。ダブルスラッシュは使用できません。 クライアント IP/ソース・プログラム/DB ユーザー/サーバー IP/サービス名グループには、ワイルドカード % をポリシー内で使用できます。
クライアント MAC	ルールで 1 つのクライアント MAC アドレスを識別するには、アドレスを「nn:nn:nn:nn:nn:nn」フォーマットで入力します (ここで、各 n は 16 進数の数字 (0-F) です)。または「クライアント MAC」ボックスにドット (.) を入力して、クライアント MAC アドレスごとに個別のカウントを保持するよう示します。あるいは「クライアント MAC」ボックスを空のままにして、クライアント MAC アドレスを無視します。
コマンド	コマンド。コマンド・グループを編集できない場合は『ルールでの値および/またはグループ値の指定』を参照してください。また、「および/またはグループ」ラベルが「収集のみ」に切り替わり、選択したグループのコマンドのみが選択されることを示していることを確認してください。 「全て」ボックスにチェック・マークが付けられている場合、SQL ステートメントのすべてのフィールドがグループのメンバーでなければなりません。
次のルールに進む	マークを付けると、ルールのテストは (そのルールが満たされているかどうかを問わず) 次のルールに進みます。これは、1 つの SQL ステートメントまたは例外で、複数のルールが満たされ、(さらに複数のアクションが実行される) 可能性があることを意味します。マークを付けない場合 (デフォルト)、このルールが満たされる際に、現行のトランザクションの追加のルールはテストされません。

フィールド	記述
データ・パターン	すべてのタイプのルール (アクセス、例外、抽出) でデータ・パターンを持つことができますが、抽出ルールでは必須です。 抽出ルールの定義に使用する場合、「データ・パターン」ボックスの正規表現がマッチングされます。「正規表現」ボタンをクリックして「正規表現の作成」ツールを開き、そのツールを使用して正規表現を入力し、テストすることができます。これにより、さらに複雑なパターンのマスキングが可能になります。マスクを掛けるセクションを括弧で囲んでください。この機能は、データベースから取得したデータにマスクを掛けるときに使用します。 例えば、次のようにします。 Windows S-TAP: ([0-9][0-9][0-9][0-9[-,]?[0-9][0-9][0-9][0-9][-,]?[0-9][0-9][0-9][0-9][-,]?)[0-9][0-9][0-9][0-9] Unix S-TAP: ([0-9]{4}[-,]?[0-9]{4}[-,]?[0-9]{4}[-,]?)[0-9]{4}[ ]{0,20} 編集 (修正) のアクションと併用する「データ・パターン」でのみ使用できる追加の正規表現 (Regex): For Windows S-TAP Name: Pattern: Masked to: SCRUB_SSN_ANSI AAA-AA-AAAA ***-***-AAAA SCRUB_SSN_UNICODE UUU-UU-UUUU ***-***-UUUU SCRUB_CC_SPACES_ANSI AAAA AAAA AAAA AAAA **** **** **** AAAA SCRUB_CC_SPACES_UNICODE UUUU UUUU UUUU UUUU **** **** **** UUUU SCRUB_CC_SOLID_ANSI AAAAAAAAAAAAAAAA ************AAAA SCRUB_CC_SOLID_UNICODE UUUUUUUUUUUUUUUU ************UUUU SCRUB_CC_AX_SOLID_ANSI AAAAAAAAAAAAAAA ***********AAAA SCRUB_CC_AX_SOLID_UNICODE UUUUUUUUUUUUUUU ***********UUUU UNIX S-TAP Name: Pattern: Masked to: SCRUB_SSN_ANSI AAA-AA-AAAA ***-***-AAAA SCRUB_SSN_UNICODE UUU-UU-UUUU ***-***-UUUU SCRUB_CC_SPACES_ANSI AAAA AAAA AAAA AAAA A*** **** **** 1234 SCRUB_CC_SPACES_UNICODE UUUU UUUU UUUU UUUU U*** **** **** **** SCRUB_CC_SOLID_ANSI AAAAAAAAAAAAAAAA A*************** SCRUB_CC_SOLID_UNICODE UUUUUUUUUUUUUUUU U*************** SCRUB_AMEX_SOLID_ANSI AAAAAAAAAAAAAAAA A*************** SCRUB_AMEX_SOLID_UNICODE UUUUUUUUUUUUUUUU U*************** 正規表現と編集機能の併用 - IBM Security Guardium ソリューションで正規表現 (Regex) を使用すると (例えば、ポリシーでマスキング機能を使用すると)、その処理がアプライアンスで実行され、正規表現の機能が拡張されます。 ただし、編集機能と一緒に使用できる正規表現ライブラリーは、データベース・サーバーのカーネルで実行され、基本的な正規表現だけに限定されています。したがって、編集機能と一緒に使用できるのは、基本的な正規表現パターンに限られます。 例えば、任意の数の数字を指定するために [0-9]* という正規表現体系を使用することはできません。一連の数字を指定するには、[0-9]-[0-9]-[0-9]... という基本的な正規表現体系を使用する必要があります。 注: S-TAP® では、事前定義の修正パターン名だけが有効であり、それ以外の名前は無視されます。 アクセス・ルール、データ・パターン、置換文字 - [a-z,2]{3}([_][0-9]{1,2}) などのデータ・パターンと置換文字 * を併用すると、データ・パターンの小括弧内の値が *** に変更されます。値にマスクを掛けるために、この機能を使用できます。 ユーザー定義文字セット Oracle、Sybase、MySQL、および MSSQL に対し使用可能、および抽出ルールに対してのみ使用可能です。ユーザーは、特別な抽出ルールを定義することで、文字セットを操作できます。これらの 文字セット ポリシー・ルールは、ユーザーがトラフィックを変換する先の文字セットを設定するためにのみ使用され、アクションの設定は関係しません。そのトラフィックに対してアクションを使用するには、その文字セットルールの後に追加ルールを定義する必要があります。次の例で定義されているように、文字セットルールの設定方法は 2 つあります (hint または force): 抽出ルールの例 (hint を使用) 通常の変換が失敗した場合に限り、インストール済みポリシーの抽出ルールで定義されている文字セットにより、トラフィックが変換されます。 文字セット EUC-JP (コード 274)。 抽出ルール・パターン: guardium://char_set?hint=274 抽出ルールの例 (force を使用) すべてのデータに対し、インストール済みポリシーの抽出ルールで定義されている文字セットにより、トラフィックが変換されます。 文字セット EUC-JP (コード 274)。 抽出ルール・パターン: guardium://char_set?force=274 このトピックの最後にある『使用可能な文字セット・コードのリスト』を参照してください。

フィールド	記述
	注: 抽出ルールは、通常、遅れてセッションにアタッチされるので注意してください。したがって、短いセッションやセッションの最初の方では、文字セットの変更が直ちに反映されない場合があります。
データベース名	データベース名。「ルールでの値および/またはグループ値の指定」を参照してください。
データベース・タイプ	サポートされる DB タイプ アクセス・ルールの場合: Cassandra、CIFS、CouchDB、Db2、Db2 COLLECTION PROFILE* (z/OS で使用する場合のみ)、FTP、GreenPlumDB、Hadoop、HTTP、IBM® INFORMIX (DRDA)、IBM iSeries、IMS、IMS COLLECTION PROFILE (z/OS で使用する場合のみ)、Informix®、MongoDB、MS SQL SERVER、MYSQL、NETEZZA、Oracle、PostgreSQL、Sybase、TERADATA、VSAM、または VSAM COLLECTION PROFILE* (z/OS で使用する場合のみ)。 例外ルールと抽出ルールの場合: Cassandra、CIFS、CounchDB、Db2、FTP、GreenPlumDB、Hadoop、IBM INFORMIX (DRDA)、IBM iSeries、Informix、MongoDB、MS SQL SERVER、MYSQL、NETEZZA、Oracle、PostgreSQL、Sybase、または TERADATA。注: Informix は 2 つのプロトコル、SQLEXEC (ネイティブ Informix プロトコル) または DRDA (IBM プロトコル) をサポートします。これらのプロトコルは、追加の設定を行わなくても、Informix トラフィックでは自動的に識別されます。サーバー・タイプ属性には、INFORMIX (SQLEXEC プロトコルの場合) および IBM INFORMIX (DRDA) (DRDA プロトコルの場合) が表示されます。 注: TERADATA にはサイレント・ログインがあり、クライアントは自動再接続することが可能です。ポリシー内で Teradata ステートメントをブロックするには、デフォルト状態を「オン」にして S-TAP ファイアウォール機能を使用し、セーフ・ユーザーを監視対象から除外します。
データベース・ユーザー	データベース・ユーザー。「ルールでの値および/またはグループ値の指定」を参照してください。
エラー・コード	エラー・コード (例外の)。「ルールでの値および/またはグループ値の指定」を参照してください。
例外タイプ	例外のタイプ (リストから選択)。 注: 例外ルールに基づいて GUI のタイムアウトによってセッションが 閉じた場合、セッション・エラー (Session_Error) は生成されません。
フィールド名	フィールド名。「ルールでの値および/またはグループ値の指定」を参照してください。 「全て」ボックスにチェック・マークが付けられている場合、SQL ステートメントのすべてのフィールドがグループのメンバーでなければなりません。
最小 カウント	(「リセット間隔」に関連して) ルールが満たされるまでに、ルールに含まれる条件が一致しなければならない最小回数。
ネットワーク・プロトコル	ネットワーク・プロトコル。「ルールでの値および/またはグループ値の指定」を参照してください。
オブジェクト	オブジェクト名。「ルールでの値および/またはグループ値の指定」を参照してください。 Sybase および MS SQL Server それぞれについて、ストアード・プロシージャーの名前を含んだ MASKED_SP_EXECUTIONS_SYBASE および MASKED_SP_EXECUTIONS_MS_SQL_SERVER という 2 つのグループがあります。含まれるプロシージャーが実行される場合、すべてにマスクが掛けられます。 「全て」ボックスにチェック・マークが付けられている場合、SQL ステートメントのすべてのフィールドがグループのメンバーでなければなりません。
オブジェクト/コマンド・グループ	選択したオブジェクト/コマンド・グループのメンバーをマッチングします。
オブジェクト/フィールド・グループ	選択したオブジェクト/フィールド・グループのメンバーをマッチングします。
OS ユーザー	オペレーティング・システム・ユーザー。「ルールでの値および/またはグループ値の指定」を参照してください。
パターン	「パターン」ボックスに指定された、マッチングする正規表現。正規表現を手動で入力することも、「正規表現」ボタンをクリックして「正規表現の作成」ツールを開き、そのツールを使用して正規表現を入力およびテストすることもできます。
期間	ルールに 1 つの期間を識別させるには、「期間」リストから事前定義された期間を選択するか、「期間」ボタンをクリックして新しい期間を定義します。
値を記録	マークを付けると、ルールが満たされる原因となった実際の構成体が SQL 文字列属性で記録され、レポートで使用可能になります。ポリシー違反に限り、マークを付けないと SQL ステートメントは記録されません。
影響を受けるレコードしきい値	アクセス・ルールのみ。 一致するレコードに関するしきい値を設定します。 例: 1000 個のインスタンスが発生したらアクションをとるようにします。 このフィールドはルールの定義に影響するのではなく、ルールの出力に影響します (例えば、いつトリガーされるかではなく、トリガーされると何が起きるか)。 影響を受けるレコードしきい値は、ルールとセッションに基づきます。 それは、ルール条件を満たすすべての照会から返される累積行数です。 すべての影響を受けるレコードの累積がしきい値に達すると、ルールがトリガーされ、(全詳細をロギングするアクションの場合) ステートメントの影響を受けるレコードは、影響を受けるレコードの累積値になります。
置換文字	マスク文字を定義します。 抽出ルールによって生成された出力が正規表現に一致する場合、括弧「(」および「)」で囲まれたサブ表現に一致する部分がマスキング文字に置き換えられます。
リセット間隔	「最小数」フィールドがゼロより大きい場合にのみ使用されます。この値は分数で指定し、その経過後に条件一致数カウンターがゼロにリセットされます。

フィールド	記述
取り消し	このチェック・ボックスは、抽出ルールの場合のみ表示されます。これを使用すると、ポリシー内の先行するルールによって既にロギングが選択されている応答を、ロギングから除外することができます。ほとんどの場合、1 つ以上の「NOT」条件を指定した 1 つのルールを定義して、不要な応答を除外し、ルールを満たす残りの応答をロギングすることで、同様の結果がより簡単に得られます。(「取り消し」チェック・ボックスは「NOT」条件よりも古い機能であり、主に既存のポリシーをサポートする後方互換性のために提供されています。)
ルールの記述	ルールの名前。ルールで特殊パターン・テストを使用するには、特殊パターン・テスト名に続けて 1 つのスペースおよびルール名を固有にするための 1 つ以上の追加の文字を入力します (例: guardium://SSEC_NUMBER employee)。(詳しくは、『特殊パターン・テスト』を参照してください。) 表示される際には、名前の前にルール番号と、ルール・タイプを識別する Access Rule、 Exception Rule、または Extrusion Rule というラベルが付けられます。ルールが「DB に基づいた推奨」機能を使用して生成された場合、生成される名前の形式は「Suggested Rule <n>_mm-dd hh:mm」になります。各コンポーネントの意味は以下のとおりです。 n は、生成されるルールのシーケンス番号です。 mm-dd は、ルールが生成された月日です。 hh:mm は、ルールが生成された時刻です。
サーバー IP	含める場合は「Not」ボックスをクリアし、除外する場合は「Not」ボックスにマークを付けます。 • 任意のサーバー: すべてのサーバー・フィールドをブランクのままにします。カウントは任意のサーバーがルールを満たすごとに増分されます。(「Not」ボックスにマークが付けられている場合は、すべてのフィールドをブランクのままにすることはできません。) • IP アドレスとマスクによって選択されたすべてのサーバー: 最初のボックスにサーバー IP アドレスを、2 番目のボックスにネットワーク・マスクを入力します。カウントは、指定された任意のサーバーがルールを満たすごとに増分されます。例えば、サブネット 192.168.3.x のすべてのサーバーを選択するには、最初のボックスに 192.168.3.1 を、2 番目のボックスに 255.255.255.0 を入力します。 • サーバーのグループ: サーバー IP アドレスのグループを「グループ」ドロップダウン・リストから選択するか、「グループ」ボタンをクリックして新規グループを定義し、そのグループを選択します。カウントは、指定されたグループの任意のメンバーがルールを満たすごとに増分されます。 • IP アドレスとマスク、およびサーバーのグループによって選択されたすべてのサーバー: 「サーバー IP」および「グループ」フィールドの両方を使用します。カウントは、いずれかの方法を使用して指定された任意のサーバーがルールを満たすごとに増分されます。 IP アドレスでのワイルドカードの使用が可能になります。 サーバー IP グループには、ワイルドカード % をポリシー内で使用できます。
サービス名	サービス名。「ルールでの値および/またはグループ値の指定」を参照してください。
重大度	リストから重大度コード (「情報」、「低」、「なし」、「中」、または「高」) を選択します。「高」が選択され、このルールで E メール・アラートが送信される場合、その E メールには緊急フラグが付けられます。
SQL パターン	「パターン」ボックスに指定された、マッチングする正規表現。正規表現を手動で入力することも、「正規表現」 をクリックして「正規表現の作成」ツールを開き、そのツールを使用して正規表現を入力およびテストすることもできます。 制約事項: SQL パターンは、編集ルールではサポートされません。
ソース・アプリケーション	アプリケーション・ソース・プログラム。「ルールでの値および/またはグループ値の指定」を参照してください。
セッションごとに 1 回起動	最初の一致の後には、同じルールに対してセッションを分析しません。 特に、「選択的な監査」ポリシーに有効です。
XML パターン	「パターン」ボックスに指定された、マッチングする正規表現。正規表現を手動で入力することも、「正規表現」 をクリックして「正規表現の作成」ツールを開き、そのツールを使用して正規表現を入力およびテストすることもできます。 このボックスでは、マッチングのための正規表現を使用できます。
MSSQL 使用時の FULL_SQL 戻り値	MSSQL では、SELECT データベース照会にストアード・プロシージャー sp_cursoropen および sp_cursorfetch が使用されます。 sp_cursoropen にはオリジナル・ステートメントが保持されます。それに対し、抽出ルールの FULL_SQL 戻り値は、Select * from _____ でなく sp_cursorfetech として表現されます。

親トピック: ポリシー・ルールのアクション

ポリシーおよびポリシー・ルールの作成とインストール

ポリシーおよびポリシー・ルールを管理するには、「データのポリシー・ビルダー」を使用します。

このタスクについて

「データのポリシー・ビルダー」は、ポリシー、ポリシー・ルール、およびポリシー・ルール・アクションを作成して変更するための単一のソリューションを提供します。この手順では、ポリシーを作成してインストールするためのエンドツーエンドのワークフローについて説明します。

手順

1. 「保護」 > 「セキュリティー・ポリシー」 > 「データのポリシー・ビルダー」にナビゲートします。
2. 新規ポリシーを作成するか、既存のポリシーまたはポリシー・テンプレートを複製します。

- 新規ポリシーを作成するには、  アイコンをクリックします。
- ポリシーをコピーするには、「セキュリティー・ポリシー」ウィンドウで既存のポリシーまたはポリシー・テンプレートを選択して、  アイコンをクリックします。
  重要: 事前定義ポリシーの [テンプレート] バージョンが使用可能な場合、古いバージョン ([テンプレート] とマークされていないもの) は、更新を受け取らないため、その使用はお勧めできません。代わりに、[テンプレート] バージョンのコピーを作成し、必要に応じてカスタマイズしてください。

a. 「新規ポリシーの作成」ウィンドウの「名前とプロパティー」パネルで、ポリシーの「タイプ」およびポリシーの「名前」を指定します。
b. データ・セキュリティー・ポリシーの場合は、オプションで追加の設定を指定します。
   - 「カテゴリー」フィールドを使用して、報告の目的でポリシー違反をグループ化するための任意のラベルを指定します。ここで指定したカテゴリーは、各ルールのデフォルト・カテゴリーとして使用され、個々のルール定義でオーバーライドすることができます。
   - 「拡張オプションの表示」をクリックして、以下の設定を行います。
     - 未解析ログ
     - 未解析ログに関するルール
     - 選択的な監査証跡

3. 「ルール」パネルをクリックして、ポリシー・ルールの処理を開始します。
   - 新規ルールを作成するには、  アイコンをクリックします。
   - ルールをコピーするには、既存のルールを選択して、  アイコンをクリックします。
   - ルールを編集するには、既存のルールを選択して、  アイコンをクリックします。

   a. 「新規ルールの作成」ウィンドウの「ルール定義」パネルで、「ルール・タイプ」および「ルール名」を指定します。 アクセス・ルールおよび例外ルールの場合、オプションで、報告を目的とした「カテゴリー」および「分類」の値を指定して、ルールの「重大度」を定義します。
   b. 「ルール基準」パネルをクリックして、ルールのパラメーターおよび値の定義を開始します。 ルール基準の中には、特定のルール・タイプでのみ使用できるものや、他の基準が定義された後でのみ使用できるものがあります。これらの依存関係はポリシー・ビルダーによって管理されます。基準は有効なコンテキストでのみ使用できます。
      - メニューを使用して、個々のパラメーターを選択し、選択演算子を定義してから、マッチングする値またはグループを指定します。
      - ルールで基準を追加または削除するには、  アイコンおよび  アイコンを使用します。

      ルール基準について詳しくは、ルール定義フィールドおよびルールでの値および値のグループを参照してください。
   c. 「ルール基準」パネルで基準を定義した後、オプションで、「次のルールに進む」チェック・ボックスを選択します。 この設定は、同じ、または類似した条件に対して複数のアクションを実行する必要がある場合に使用してください。詳しくは、次のルールに進むを参照してください。
   d. 「ルール・アクション」パネルをクリックして、ルール・アクションの処理を開始します。
      - 新規ルール・アクションを作成するには、  アイコンをクリックし、アクションを選択します。追加の構成が必要な場合は、「新規アクションの追加」ダイアログを使用して、アクションを定義します。
      - ルール・アクションを編集するには、既存のアクションを選択して、  アイコンをクリックし、「アクションの編集」ダイアログを使用して、ルール・アクション構成を更新します。

      使用可能なアクションについて詳しくは、ポリシー・ルールのアクションを参照してください。
   e. ルールの定義が完了したら、「OK」をクリックして、「ルール」パネルに戻ります。 必要に応じて、ルールの作成、コピー、および編集を続行します。
4. ポリシーおよびそのルールの定義が完了したら、「OK」をクリックしてポリシーを保存し、「セキュリティー・ポリシー」表に戻ります。

次のタスク

「セキュリティー・ポリシー」ウィンドウでポリシーを選択して、「インストール」 > 「インストール」をクリックし、ポリシーをインストールします。目的の「インストール・アクション」を選択して、「OK」をクリックし、ポリシーをインストールします。インストールされたポリシーは、「インストール済み」列に  で示されます。

「保護」 > 「セキュリティー・ポリシー」 > 「ポリシー・インストール」を使用してポリシーをインストールすることもできます。詳しくは、「ポリシー・インストール」ツールの使用を参照してください。

親トピック: ポリシー

「ポリシー・インストール」ツールの使用

このトピックを使用して、Guardium コレクターにポリシーをインストールし、スケジュールを変更します。

複数ポリシーのサポート

1. 「設定」 > 「ポリシー・ビルダー」をクリックして「ポリシー・ファインダー」を開くか、「保護」 > 「セキュリティー・ポリシー」 > 「ポリシー・インストール」をクリックして「ポリシー・インストーラー」を開きます。
2. 「ポリシーの記述」ボックスからインストールするポリシーを選択します。
3. 以下のいずれかを実行します。
   - 「インストール」をクリックすると、ポリシーが直ちにインストールされます。
   - 「ポリシー・インストーラー」を使用する場合、「スケジュールの変更」をクリックすると、汎用のスケジューリング・ユーティリティーが開き、ポリシーのインストールをスケジュールできます。

インストールしたポリシーのポリシー・ルールを表示

複数のインストール済みポリシーを同時に使用できます。 インストールされたポリシーはすべて、操作に使用できます。ここで 2 つの制限があります。つまり、選択的な監査ポリシーとして定義されたポリシーは、選択的な監査ポリシーとして定義されていないポリシーと混用できません。さらに、未解析ログとして定義されたポリシーは未解析ログとして定義されていないポリシーと混用できません。 ポリシーを混用しようとすると、これらの混合ポリシーのインストール時にエラー・メッセージが発生します。

表示される順番は、最初、最後、または中間のどこか、というように、ポリシーのインストール中に制御できます。 しかし表示順は後日編集することができません。

以前にインストールされたポリシーを削除するには、「ポリシー・インストール」ページで  アイコンをクリックします。

最初にインストールしたポリシーには特別な意味があります。このポリシーはグローバルなポリシー・パラメーターの値を設定します。 これらのパラメーターには、グローバルなパターン、選択的な監査かどうか、クライアントおよびサーバーのネットマスク、タグ付きクライアントおよびタグ付きサーバーのグループ ID があります。

この複数ポリシーのサポートは GUI (「設定」 > 「ツールとビュー」 > 「ポリシー・インストール」) および GuardAPI から使用できます。

インストールしたポリシーのポリシー・ルールを表示

「現在インストール済みのポリシー」パネルから、すべてのユーザーはインストール済みポリシーのルールを表示でき、さらに、許可されたユーザーはポリシーを開いて編集できます。

1. 「保護」 > 「セキュリティー・ポリシー」 > 「ポリシー・インストール」にナビゲートして「ポリシー・インストーラー」を開きます。
2. 「詳細レポートを表示」ボタンをクリックして、インストールされているポリシーとルールの詳細リストを表示します。
3. ポリシーとそのルールを編集するには、  アイコンをクリックします。

ジョブ依存関係スケジューラー

Guardium コレクターには、「ポリシー・インストール」、「監査プロセス」、「グループ更新 (Group updates)」など、定期的に実行されるようスケジュールする多数のタスクがあります。「ジョブ依存関係」機能では、直接関係があり、スケジュールするタスクが正常に実行されるかどうかに影響するすべてのジョブが検出されます。スケジュールするジョブの前提条件として定義されているジョブを検出しないと、タスクが不適切なデータに基づく可能性があり、これにより誤った結果または不適切な結果が発生します。

主要機能

- ユーザーは、実行時に依存関係を検出および実行するために、スケジュール済みのジョブにマークを付けます。
- スケジューラーによりジョブが実行されると、すべての従属ジョブが自動的に検出され、順番に実行されます。
- 失敗した場合の再試行手順があります。

依存関係の検出

- 依存関係が必要なシナリオを識別します。
- 実行可能なジョブと実行不可のジョブを識別します。
- 事前定義のジョブ依存関係を計算します。

ジョブ	推奨される前提条件ジョブ	理由
ポリシー・インストール	(インストールする) ポリシーで定義されていて、「照会から取り込み」メカニズムでデータを取り込むようにスケジュールされている、またはスケジュールされていないグループ。	グループを使用するポリシー・ルールには、インストール前に、最新のグループ・データを取り込む必要があります。
ポリシー・インストール	分類タスクに「オブジェクトのグループに追加」アクション、「オブジェクト/フィールドのグループに追加」アクション、または「アクセス・ルールに追加」アクションのある、分類監査タスクを含む監査プロセス。	グループを使用するポリシー・ルールには、インストールの前に、最新のグループ・データを含める必要があります。
監査プロセス	カスタム表名がレポート・タイプの監査タスクで (「from」節により) 参照されるカスタム表アップロード・ジョブ。	監査プロセスの実行をスケジュールする前に、レポート・タイプの監査タスクで参照されるカスタム表に最新のデータを取り込む必要があります。
監査プロセス	レポート・タイプの監査タスクの条件で定義されていて、「照会から取り込み」メカニズムでデータを取り込むようにスケジュールされている、またはスケジュールされていないグループ。	レポート・タイプの監査タスクを実行する前に、照会条件で参照されるグループに最新のデータを取り込む必要があります。
照会から取り込み	グループにデータを取り込むために使用される照会のエンティティーを含むカスタム・アップロード表。	
監査プロセス	インポート	アグリゲーターのみに関連します。この前提条件により、監査プロセスの実行前に、すべての統合ユニットから情報が確実にインポートされます。

スケジューラーの機能拡張

- スケジュール済みジョブの実行時にジョブの依存関係を検出します。
- ジョブの依存関係を順番に実行します。

実行可能ジョブはスケジュールでき、実行不可のジョブはスケジュールできません。

グループは実行不可のジョブです。

グループに対する「照会から取り込み」は実行可能です。

直接依存関係は、定義により結び付けられたオブジェクトです。例えば、ポリシーはルールに依存し、ルールはグループに依存します。

間接依存関係は、論理的に結び付けられたオブジェクトです。例えば、ポリシーをインストールする前に監査プロセスを実行します。

GUI サポート

1. 「ポリシー・インストール」から「スケジュールの作成」を選択した後、「従属ジョブの自動実行」チェック・ボックスにチェック・マークを付けます。

2. 「保存」をクリックしてプロセスをスケジュールします。これにより、ユーザーに依存関係の状況が通知されます。

GuardAPI サポート

GuardAPI ジョブ従属関係コマンド:

CLI> grdapi add_job_dependency

関数パラメーター:

dependOnJobExecutedWithin - 文字列

dependOnTrigger - 文字列 - 必須

intervalBetweenRetries - 整数

jobRetries - 整数

jobTrigger - 文字列 - 必須

runIfDependOnJobReturns - 文字列

api_target_host - 文字列

依存関係を自動実行するには、次の GuardAPI コマンドを使用します。

> grdapi auto_execute_suggested_dependencies jobTrigger=*<trigger name of the scheduled job>*

CLI> grdapi auto_execute_suggested_dependencies

関数パラメーター:

jobTrigger - 文字列 - 必須

api_target_host - 文字列

CLI> grdapi delete_job_dependencies

関数パラメーター:

dependOnTrigger - 文字列

jobTrigger - 文字列 - 必須

api_target_host - 文字列

CLI> grdapi disable_auto_execute_suggested_dependencies

関数パラメーター:

jobTrigger - 文字列 - 必須

api_target_host - 文字列

CLI> grdapi list_job_dependencies_tree

関数パラメーター:

jobTrigger - 文字列 - 必須

api_target_host - 文字列

すべてのスケジュール済みジョブ/トリガーのリストを取得するには、次の GuardAPI コマンドを実行します。

> grdapi list_scheduler_jobs

CLI> grdapi list_suggested_job_dependencies

関数パラメーター:

jobTrigger - 文字列 - 必須

api_target_host - 文字列

CLI> grdapi list_existing_job_dependencies

関数パラメーター:

jobTrigger - 文字列 - 必須

api_target_host - 文字列

CLI> grdapi modify_job_dependency

関数パラメーター:

dependOnJobExecutedWithin - 文字列

dependOnTrigger - 文字列 - 必須

intervalBetweenRetries - 整数

jobRetries - 整数

jobTrigger - 文字列 - 必須

runIfDependOnJobReturns - 文字列

api_target_host - 文字列

CLI> grdapi show_job_dependency_execution_profile

関数パラメーター:

dependOnTrigger - 文字列 - 必須

jobTrigger - 文字列 - 必須

api_target_host - 文字列

スケジューラーの実行

スケジューラーは、ジョブの実行時に、ジョブの依存関係をチェックします。

依存関係は逆順に実行されます。

例: 次のような依存関係ツリーがあるとします。

ポリシーのインストール (実行可能)						
	監査プロセス (実行可能/間接依存関係)					
		監査タスク				
			分類プロセス			
				分類ポリシー		
					分類ポリシー・アクション	
						グループ (実行可能/直接 - 照会から取り込み)

実行順は以下のようになります: 照会から取り込み → 監査プロセス → ポリシーのインストール

スケジューラーは依存関係を 1 つずつ実行し、それらが終了するのを待機します。

依存関係ツリーの実行がすべて完了するまで長時間かかる場合がありますが、依存関係はすべて正しい順番で確実に実行されます。

エラーの処理

いずれかの依存関係の実行が失敗した場合、スケジューラーにより現在実行されているジョブが実行されなくなります。

障害が発生した場合、エラー・メッセージが「スケジュール済みジョブ例外」レポートに書き込まれます。

前のジョブに依存するジョブの再試行回数を設定できます。デフォルトは 3 です。有効な値は ≥ 0 です。再試行間の間隔は分単位で設定できます。 デフォルトは 3 です。有効な値は ≥ 0 です。

親トピック: ポリシー

相関アラート

アラートは、例外またはポリシー・ルール違反が検出されたことを示すメッセージです。

アラートは次の 2 つの方法で起動します。

- *相関アラート*は、指定期間をさかのぼってアラートしきい値が満たされたかどうかを判別する照会によって起動されます。 Guardium 異常検出エンジンは、スケジュールに基づいて相関照会を実行します。デフォルトでは、相関アラートはポリシー違反をログに記録しませんが、記録するように構成することもできます。
- *リアルタイム・アラート* は、セキュリティー・ポリシー・ルールにより起動されます。Guardium 検査エンジン・コンポーネントは、データベース・トラフィックをリアルタイムに収集および分析するときに、セキュリティー・ポリシーを実行します。

起動方法に関係なく、Guardium はすべてのアラートを同じ方法 (アラート情報を Guardium 内部データベースに記録する) でログに記録します。ログに記録される情報の量およびタイプは、具体的なアラート・タイプによって異なります。 同じくスケジュールに基づいて実行される Guardium アラート機能コンポーネントでは、それぞれの新しいアラートが処理されます。 このとき、アラートごとにログに記録された情報は、以下の通知メカニズム (任意の組み合わせが可能) に渡されます。

- SMTP - SMTP (E メール発信) サーバー。 アラート機能により、標準 E メール・メッセージが、構成済み SMTP サーバーに渡されます。
- SNMP - SNMP (ネットワーク情報および制御) サーバー。 アラート通知用に SNMP を選択すると、アラート機能により、そのタイプのすべてのアラート・メッセージは、アラート機能が構成されている単一のトラップ・コミュニティーに渡されます。

- Syslog - アラートは Guardium アプライアンスの syslog に 書き込まれます (Guardium 管理者は、syslog メッセージをリモート・システムに書き込むようにこのアプライアンスを構成することもできます)。
  注: SNMP または SYSLOG では、最大メッセージ長は 3000 文字です。 それを超える長さのメッセージは切り捨てられます。
- カスタム - ユーザーがアラート処理のために作成した Java™ クラス。 アラート機能により、アラート・メッセージとタイム・スタンプは、カスタム・アラート・クラスに渡されます。 複数のカスタム・アラート・クラスが存在する場合があり、あるカスタム・アラート・クラスが別のカスタム・アラート・クラスの拡張である場合もあります。

注: アラート定義および通知は、データ・レベル・セキュリティーの影響を受けません。その理由として、アラートがユーザーとの関連で評価されないことや、アラートが複数のユーザーに関連付けられたデータベースに関連している可能性があり、そのアラート通知の受信者が 1 人もいないという状態を回避するためなどが挙げられます。
注: カウンターを含めて 30 以上のフィールドのある照会を使用するアラートがあると、配列境界外の例外 (`Array out of bound exception`) エラー・メッセージが出され、異常検出は失敗に終わります。アラートには、30 以上の列を持つ照会を使用できません。そのような照会は、しきい値アラートに使用可能な照会のリストに表示されません。

管理者用のアラート・タスク

Guardium 管理者は、以下のタスクを実行します。

- 「グローバル・プロファイル」を使用して、アラート・メッセージ・テンプレートをカスタマイズします。
- 「アラート機能」を構成し、開始します。そうすることにより、メッセージが SMTP、SNMP、Syslog、またはカスタム・アラート・クラスに送信されます。
- 定義されたスケジュールに従って相関アラートを実行する異常検出エンジンを開始および停止します。
- カスタム・アラート・クラスを Guardium システムにアップロードします。

ユーザー用のアラート・タスク

Guardium ユーザー (および管理者) は、以下の相関アラート・タスクを実行できます。

- 相関アラートに使用可能な照会を定義します。
- 相関アラートを定義します。
- カスタム・アラート・クラスを作成します。

相関アラート照会について

相関アラートは、いずれかのレポート・ドメインの照会に基づいています。 その照会は、アラートを定義する前に定義しておく必要があります。 照会には、少なくとも 1 つの日付フィールドが含まれていないと、相関アラートで使用できません。

相関アラートの作成

1. 「保護」 > 「データベースの侵入検出」 > 「アラート・ビルダー」をクリックして「アラート・ファインダー」を開きます。
2. 「アラート・ファインダー」パネルで「新規」をクリックして、「アラートの追加」パネルを表示します。
3. 「設定」ペインで以下を実行します。
   a. 「名前」ボックスに、アラートの固有の名前を入力します。 アラート名にはアポストロフィ文字を含めないでください。
   b. 「記述」ボックスに、アラートについて説明する短い文を入力します。
   c. 「カテゴリー」ボックスに、オプションでカテゴリーを入力します。
   d. 「分類」ボックスに、オプションで分類を入力します。
   e. 「推奨アクション」には、特定のアラートに対する推奨アクションとしてのフリー・テキストをユーザーが追加できます。
   f. リアルタイム・アラートの場合と同様に、ユーザーは、しきい値アラートが起動した場合に送信されるメッセージのテンプレートを選択できます。このテンプレートでは、特定のアラート用に適切な値に置換される変数の定義済みリストが使用されます。デフォルト・テンプレートおよび変数のリストについては、『グローバル・プロファイル』ヘルプ・トピックにある『名前付きテンプレート』セクションで詳細を説明しています。
   g. 「重大度」リストから重大度レベルを選択します。 E メール・アラートで「高」を設定すると、E メールに「高」のフラグが付きます。
   h. 照会の実行間隔を「実行頻度」フィールドに分単位で入力します。
   i. 「アクティブ」にチェック・マークを付けるとアラートはアクティブになり、このボックスをクリアすると実行を開始せずにアラート定義が保存されます (後でアクティブにできます)。中央マネージャー環境では、このボックスにマークを付けるとすべての管理対象ユニットでアラートがアクティブになり、クリアするとアラートが停止します。 中央マネージャー環境の特定のアプライアンスでアラートを無効にするには、「管理者コンソール」の「異常検出」パネルを使用します。
   j. このアラートの起動時にポリシー違反をログに記録する場合、「ポリシー違反をロギング」にチェック・マークを付けます。デフォルトでは、相関アラートは「アラートのトラッキング」ドメインにのみ記録されます。 このボックスにマークを付けると、相関アラートおよびリアルタイム・アラート (データ・アクセス・セキュリティー・ポリシーから発行される) を「ポリシー違反」ドメインに一緒に表示できます。
   k. このアラートを適用状態ダッシュボードに含める場合、「適用状態ダッシュボードに表示」にチェック・マークを付けます。
4. 「アラート定義」パネルで以下を実行します。
   a. このアラートに対して実行する照会を選択します。　表示される照会のリストには、次のことが定義された照会がすべて含まれます。
      - 少なくとも 1 つの日付フィールド (タイム・スタンプ) を含んでいる (タイム・スタンプ・フィールドは必須)
      - 1 つのカウント・フィールドを含んでいる (カウント・フィールドは必須)
      - ご使用の Guardium ユーザー・アカウントでアクセスできる
   b. 選択した照会にランタイム・パラメーターが含まれる場合、「アラート定義」ペインに「照会パラメーター」パネルが表示されます。アプリケーションに適したパラメーター値を指定してください。
      トラブルシューティングのヒント
      - カスタム照会をいずれかのを照会 - レポート・ビルダーで作成したのに、照会リストに表示されていない場合は、そのカスタム照会にタイム・スタンプ (日付フィールド) があることを確認してください。
      - 「アラートの追加」画面の「アラート定義」パネルにある照会リストから照会を選択した後、その照会を編集する必要があるが、(「編集」アイコンで) 照会を編集できない場合には、を照会 - レポート・ビルダーに移動して照会を編集してください。
   c. 「集計間隔」ボックスに、時間間隔の長さ (分単位) を入力します。 照会時に、現在時刻からこの間隔をさかのぼって監査リポジトリーが検査されます (例えば、10 と入力すると、過去 10 分間のデータが検査されます)。
   d. 次より早期の移動間隔期間: GBDI データをすぐには使用できません。照会の時間間隔全体にデータが存在するように、時間 (分単位) をさかのぼって集計間隔を移動させるには、このフィールドを使用します。通常は 120 分で十分です。
      注: アグリゲーターで実行されるアラートは、定義されたマージ期間内のデータのみに基づくものです。

e. アラートと共にレポート全体をログに記録するには、「全照会結果をロギング」ボックスにチェック・マークを付けます。

5. 選択した照会に数値データの列が 1 つ以上含まれる場合、それらの列の 1 つを選択してテストに使用します。 デフォルトは、最後にリストされる項目であり、照会の最後の列になります。これは常に、その行に統合されたオカレンス数になります。
6. 「アラートしきい値」ペインで、相関アラートが生成される際のしきい値を以下のように定義します。
   - 「しきい値」フィールドに、パネルの残りのフィールドの名前に従って適用されるしきい値の数値を入力します。
   - アラート条件値リストで、アラート生成のためのレポート値としきい値との関連付けを示す演算子 (より大きい、以上、より小さいなど) を選択します。
   - しきい値の数値をレポートの合計に適用する場合は「レポート当たり」を選択します。しきい値をレポートの 1 行に適用する場合は「行当たり」を選択します (レポートは、選択した、指定した集計時間をさかのぼって実行される照会の出力です)。

     指定した「集計間隔」の間にデータが存在しない場合には、以下のようになります。

     しきい値が「レポート当たり」である場合、間隔の値は 0 (ゼロ) であり、 しきい値条件が満たされる場合 (例えば指定される条件が「値が 1 未満のときにアラート」である場合など) にアラートが生成されます。

     しきい値が「行当たり」である場合、指定した条件に関係なくアラートは生成されません (これは、出力行が存在しないためです)。

   - 「絶対制限として」を選択して入力されたしきい値が絶対数であることを指定するか、 「次の期間内のパーセンテージ変化として」を選択してしきい値が「開始」および「終了」フィールドで指定した期間内における変化のパーセンテージを表すことを指定します。

     「次の期間内のパーセンテージ変化として」 オプションを選択した場合、日付ピッカー・コントロールを使用して「開始」および「終了」の日付を選択します。

     「同じ「統合期間」の次の相対時間におけるパーセンテージ変化として」を選択した場合、1 つの相対日付が入力され、 アラートによって現行期間および相対期間 (同じ間隔を使用する) の照会が実行されて、ベース期間の値のパーセンテージとして値が検査されます。
     注: 相対期間を使用する場合、アラートが検査されるたびに照会が 2 回 (現行期間について 1 回と、相対期間について 1 回) 実行されます。
7. 「通知頻度」ボックスでは、アラート条件が満たされる場合にアラート受信者が通知を受ける頻度 (分単位) を指定します。
8. 「保存」をクリックしてアラート定義を保存します。
   注: 定義が保存されるまでは、受信者やロールを割り当てたり、コメントを入力したりすることはできません。
9. 「アラート受信者」パネルで、このアラート条件が満たされる場合に通知を受けるユーザーまたはグループを、オプションで 1 つ以上指定します。 受信者を追加するには、「受信者の追加」ボタンをクリックして、「アラート受信者の選択」パネルを開きます。
   注: アラートの受信者が管理者ユーザーである場合、その管理者にアラートの送信先 E メールを割り当てる必要があります。
   注: しきい値アラートの追加の受信者は、「所有者」(データベースの所有者) です。 アラートに関連付けられた照会に、サーバー IP およびサービス名が含まれており、アラートが「行当たり」として評価される場合、受信者を「所有者」にすることができます。 その場合のアラート通知は、「アラート通知タイプ: メール、アラート・ユーザー ID: 0、アラート宛先: 所有者」でなければなりません。 リアルタイム・アラートの追加の受信者については、『ポリシー』の『アラート・アクション』を参照してください。
10. オプションで「ロール」をクリックして、アラートのロールを割り当てます。
11. オプションで「コメント」をクリックして、定義にコメントを追加します。
12. 「適用」をクリックし、完了したら「完了」をクリックします。

相関アラートの変更

1. 「保護」 > 「データベースの侵入検出」 > 「アラート・ビルダー」をクリックして「アラート・ファインダー」を開きます。
2. 「アラート・ファインダー」パネルで、変更する相関アラートを選択します。
3. 「変更」をクリックして、「アラートの変更」パネルを開きます。
4. 『相関アラートの作成』トピックを参照して、アラート定義に変更を加えます。
5. 「保存」をクリックします。

相関アラートの削除

1. 「保護」 > 「データベースの侵入検出」 > 「アラート・ビルダー」をクリックして「アラート・ファインダー」を開きます。
2. 「アラート・ファインダー」パネルで、削除する相関アラートを選択します。
3. 「削除」ボタンをクリックします。アクションの確認を求めるプロンプトが出されます。

親トピック: 保護

相関アラートを使ってイベントを通知する方法

アプリケーションのいずれかの個別ユーザーで最近 3 時間に 15 個より多い SQL エラーが存在する場合、相関アラートをトリガーします。

このタスクについて

相関アラートを使用することで、一定時間に累積されたイベントについて通知します。通常、アプリケーションでは SQL エラーが発生しません。あるアプリケーションで SQL エラーが増加している場合、SQL インジェクションが試みられている可能性があり、警戒すべき徴候です。詳しくは、オンライン・ヘルプのトピック『相関アラート』および『照会』を参照してください。

前提条件

- E メール (SMTP) サーバーを構成します (「設定」 > 「ツールとビュー」 > 「アラート機能」)
- 相関アラートを完全に構成した後、それがアクティブ状態で、実行中であることを確認します (「設定」 > 「ツールとビュー」 > 「異常検出」)

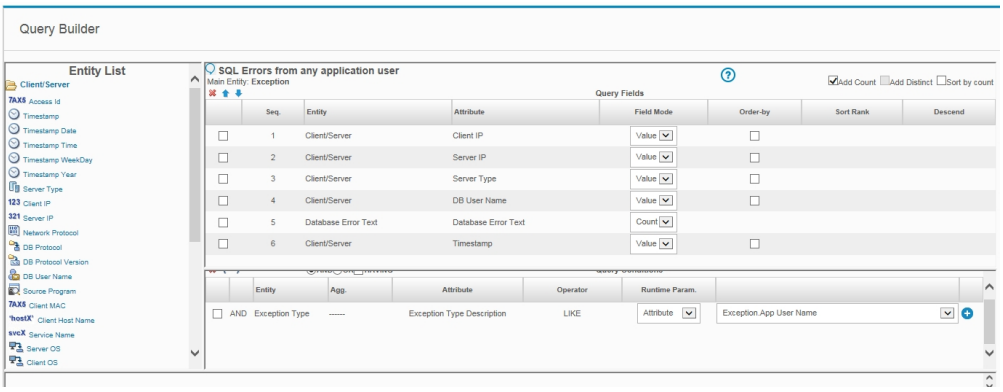
アラートは、例外 (相関アラートの場合) またはポリシー・ルール違反 (リアルタイム・アラートの場合) が検出されたことを示すメッセージです。

相関アラートは、指定された期間をさかのぼって、アラートしきい値が満たされたかどうかを判別する照会によって起動されます。

相関アラートの手順の概要

1. SQL エラーのフィールド (カウント付き) およびアプリケーション・ユーザーの条件を使用して、例外トラッキングからカスタム照会を作成します。アラート・ビルダーの中でこのカスタム照会を使用するには、日付フィールド (タイム・スタンプ) が必要です。

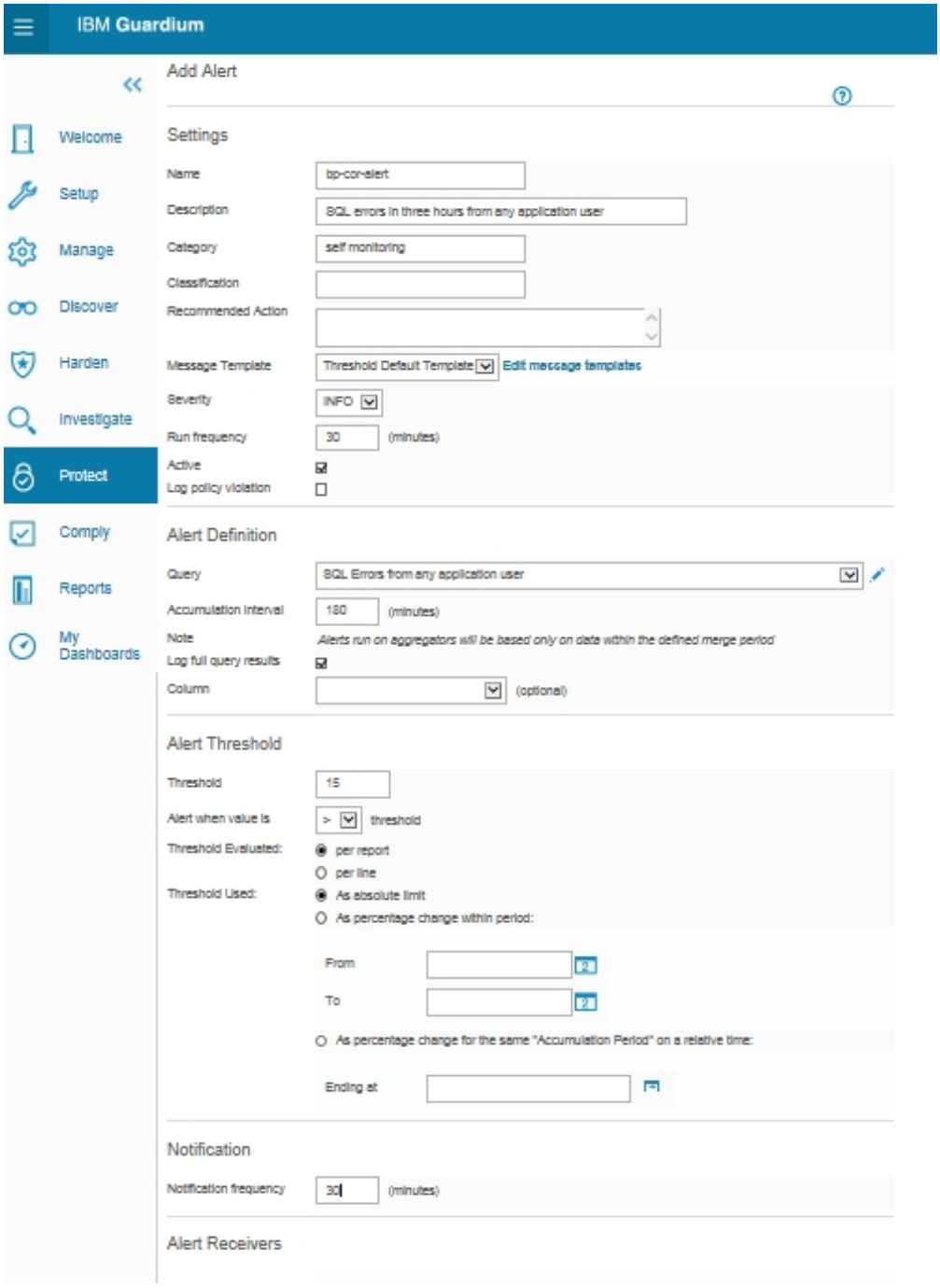
2. 「保護」 > 「データベースの侵入検出」 > 「アラート・ビルダー」をクリックして「アラート・ファインダー」を開きます。
3. 「新規」をクリックします。アラート・ビルダーのメニュー画面が表示されたら、説明に従って各フィールドに入力します。
4. 受信者を追加します。



「例外」ドメイン、SQL エラーの照会

手順

1. 例外のトラッキング - 照会ファインダーを開く
   - ユーザー:「ツール」 > 「レポートのビルド」を選択した後、「例外」ドメインのみを選択します。
2. 照会のドロップダウン選択項目を開きます。「SQL エラー」を選択します。SQL エラーがメイン・タイトルとなった構成画面が開きます。
3. 照会のテキスト・ボックスに固有の名前を入力して、この選択項目のコピーを作成します。照会名にはアポストロフィ文字を含めないでください。
4. カスタム照会では、照会フィールドの下で、クライアント/サーバー・エンティティー・リストから日付フィールド (タイム・スタンプ) を追加して、データベース・エラー・テキスト・フィールドをカウント・フィールド・モードに変更します。「照会条件」の下にある、例外タイプの「ランタイム・パラメーター」を「属性」に変更して、「例外: アプリケーション・ユーザー名」を選択します。
5. 「保存」をクリックします。これで、任意のアプリケーション・ユーザーからの SQL エラーに関するこのカスタム照会をアラート・ビルダーで使用できます。



アラート・ビルダーのメニュー画面

6. アラート・ビルダー - 相関アラートの作成
7. 「保護」 > 「データベースの侵入検出」 > 「アラート・ビルダー」をクリックして「アラート・ファインダー」を開きます。
8. 「アラート・ファインダー」パネルで「新規」ボタンをクリックして、「アラートの追加」パネルを表示します。
9. 「名前」ボックスに、アラートの固有の名前を入力します。 アラート名にはアポストロフィ文字を含めないでください。
10. 「記述」ボックスに、アラートについて説明する短い文を入力します。
11. 「カテゴリー」ボックスに、オプションでカテゴリーを入力します。この場合は自己モニターが使用されました。
12. 「分類」ボックスに、オプションで分類を入力します。
13. 「重大度」リストから重大度レベルを選択します。 E メール・アラートの場合、「高」に設定すると、E メールに緊急フラグが付けられます。
14. 照会の実行間隔を「実行頻度」フィールドに分単位で入力します。
15. 「アクティブ」ボックスにマークを付けると、アラートがアクティブになります。
16. このアラートの起動時にポリシー違反をログに記録する場合、「ポリシー違反をロギング」ボックスにマークを付けます。デフォルトでは、相関アラートは「アラートのトラッキング」ドメインにのみ記録されます。 このボックスにマークを付けると、相関アラートおよびリアルタイム・アラート (データ・アクセス・セキュリティー・ポリシーから発行される) を「ポリシー違反」ドメインに一緒に表示できます。
17. 「アラート定義」パネルの「照会」リストから、このアラートのために実行する照会を選択します。表示される照会のリストには、次のことが定義された照会がすべて含まれます。
    - 少なくとも 1 つの日付フィールド (タイム・スタンプ) を含んでいる (タイム・スタンプ・フィールドは必須)
    - 1 つのカウント・フィールドを含んでいる (カウント・フィールドは必須)
    - ご使用の Guardium® ユーザー・アカウントでアクセスできる

トラブルシューティングのヒント: カスタム照会を照会 - レポート・ビルダーで作成したのに、照会リストに表示されていない場合は、そのカスタム照会にタイム・スタンプ (日付フィールド) があることを確認してください。

トラブルシューティングのヒント: 「アラートの追加」画面の「アラート定義」パネルにある照会リストから照会を選択した後、その照会を編集する必要があるが、(「編集」アイコンで) 照会を編集できない場合には、照会 - レポート・ビルダーに移動して照会を編集してください。

18. 選択した照会にランタイム・パラメーターが含まれる場合、「アラート定義」ペインに「照会パラメーター」パネルが表示されます。アプリケーションに適したパラメーター値を指定してください。
19. 「集計間隔」ボックスに、時間間隔の長さ (分単位) を入力します。 照会時に、現在時刻からこの間隔をさかのぼって監査リポジトリーが検査されます (例えば、10 と入力すると、過去 10 分間のデータが検査されます)。
20. アラートと共にレポート全体をログに記録するには、「全照会結果をロギング」ボックスにマークを付けます。
21. 選択した照会に数値データの列が 1 つ以上含まれる場合、それらの列の 1 つを選択してテストに使用します。 デフォルトは、最後にリストされる項目であり、照会の最後の列になります。これは常に、その行に統合されたオカレンス数になります。
22. 「アラートしきい値」ペインで、相関アラートが生成される際のしきい値を以下のように定義します。
    - 「しきい値」フィールドに、パネルの残りのフィールドの名前に従って適用されるしきい値の数値を入力します。
    - 「値が次のときにアラート」リストから、アラート生成のためのレポート値としきい値との関連付けを示す演算子 (より大きい、以上、より小さいなど) を選択します。
    - しきい値の数値がレポート総計に適用される場合は、「レポートごと」を選択します。

    指定された「集計間隔」においてデータが存在しない場合: しきい値がレポートごとである場合、その間隔における値は 0 (ゼロ) であり、しきい値条件が満たされるならアラートが生成されます (例えば「値が次のときにアラート通知: < 1」という条件が指定されている場合)。
23. 「通知頻度」ボックスでは、アラート条件が満たされる場合にアラート受信者が通知を受ける頻度 (分単位) を指定します。
24. 「適用」ボタンをクリックして、アラート定義を保存します。
    注: 定義が保存されるまでは、受信者やロールを割り当てたり、コメントを入力したりすることはできません。
25. 「アラート受信者」パネルで、このアラート条件が満たされる場合に通知を受けるユーザーまたはグループを、オプションで 1 つ以上指定します。 受信者を追加するには、「受信者の追加」ボタンをクリックして、「アラート受信者の選択」パネルを開きます。 受信者の追加について詳しくは、『通知』を参照してください。
26. オプションで「ロール」ボタンをクリックして、アラートのロールを割り当てます。『セキュリティー・ロール』を参照してください。
27. オプションで「コメント」ボタンをクリックして、定義にコメントを追加します。
28. 完了したら、「適用」ボタン、「完了」ボタンの順にクリックします。

    いずれかのアプリケーション・ユーザーで最近 3 時間に 15 個より多い SQL エラーが存在する場合、指定された受信者にアラートが送信されます。

親トピック: 保護

インシデント管理

統合インシデント管理 (IIM) アプリケーションには、 データベースのセキュリティー・インシデントをトラッキングして解決するワークフロー自動化機能を備えたビジネス・ユーザー・インターフェースがあります。

このインターフェースでは、一連の関連するポリシー違反をグループにして 1 つのインシデントとし、 特定の個人に割り当てることを管理者に可能にすることで、インシデント管理を簡略化します。 これにより、監視チームによるレビューが必要なポリシー違反の数が減らせます。

インシデントの生成プロセスを定義してスケジュールすることで、ポリシー違反ログの読み取りや、新規インシデントの生成が行えます。 インシデントの生成プロセスでは、選択される各インシデントは次のようになります。

- 固有のインシデント番号が割り当てられる。
- ユーザーに割り当てられる。
- 重大度コードが割り当てられる。
- カテゴリーに割り当てられる。

さらに、ポリシー違反は、「ポリシー違反/インシデント管理」レポートから (権限を持つユーザーが) 手動で新規インシデントまたは既存のインシデントに割り当てることができます。

インシデントが生成されると、管理者および他のユーザーは、(admin ポータルとユーザー・ポータルに含まれる) 「インシデント管理」タブからインシデントを操作します。 ここからは、他のすべてのタスク (インシデントの割り当て、通知の送信、状況の割り当て、など) を実行できます。

インシデント管理機能は、「インシデント管理」レポートのドリルダウン・メニューからアクセスできます。 各ユーザーは、ユーザー・アカウントに割り当てられたセキュリティー・ロールに応じて、 レポートや機能のサブセットのみ使用できます。

インシデント管理レポートの独自のコピーを作成できますが、これらのコピーには 「インシデント管理」タブの事前構成されたレポートから使用可能なすべての機能が備わっているわけではありません。 インシデント、重大度コードなどを割り当てるには、「インシデント管理」タブのレポートを使用します。

インシデント生成プロセスの定義

インシデントの生成プロセスでは、ポリシー違反ログへの照会が実行され、 その照会に基づいてインシデントが生成されます。デフォルトでは、インシデント生成プロセスの定義およびスケジューリングは、admin ロールを持つユーザーに制限されます。

1. 「順守」 > 「ツールとビュー」 > 「インシデント生成」をクリックして、「インシデント生成プロセス」を開きます。
2. 「プロセスの追加」をクリックして、「インシデント生成プロセスの編集」パネルを開きます。
3. 「照会」リストから照会を選択します。 インシデント生成プロセスで使用される照会に適用されるいくつかの制約があります。 照会 - レポート・ビルダーで照会を開き、次の条件が満たされていることを確認することを推奨します。
    - 照会はポリシー違反ドメインのものであること。
    - 照会のチェック・ボックスにチェックが付いていること。詳しくは、列表示の選択を参照してください。
    - 照会のメイン・エンティティーがポリシー・ルール違反エンティティーであること。
    - 照会の照会フィールドに SQL 文字列 (SQL エンティティーのもの、またはポリシー・ルール違反エンティティーの「SQL 文字列全体」属性のもの) が含まれないこと。
4. インシデントの重大度を選択します (デフォルトは「情報」)。
5. オプションで、インシデントのカテゴリーを入力します (デフォルトは「なし」)。

6. オプションで、インシデント生成のしきい値を入力します。 デフォルトは 1 で、照会で返されるすべての行がインシデントを生成します。
7. 「ユーザーに割り当て」リストから、インシデントを割り当てるユーザーを選択します。
8. 照会の開始日と終了日を入力します。 スケジュールした照会の場合、相対日付を使用します (例: 「現在 -1 日」や「現在」)。
9. 「保存」をクリックして、プロセスの定義を保存します。プロセスは、保存しないと実行またはスケジュールできません。
10. ここで照会を実行するには、「今すぐ 1 回実行」をクリックします。
11. 照会をスケジュールするには、「スケジュールの変更」をクリックして、汎用のスケジューリング・ユーティリティーを開きます。

インシデントへの割り当て/再割り当て

1. いずれかの「インシデント管理」レポートで、割り当て/再割り当てをするポリシー違反をダブルクリックします。
2. ドリルダウン・メニューから「インシデントに割り当て/再割り当て」を選択します。このメニューを選択すると、オープン・インシデント (例えば、「インシデント #123 に割り当て」) のリストと 1 つの追加オプション (「新規インシデントに割り当て」) を含む新しいメニューが表示されます。
3. この違反を割り当てるインシデントを選択するか、「新規インシデントに割り当て」を選択して次の使用可能なインシデント番号 (順に番号が付けられます) にポリシー違反を割り当てます。

   変更が完了するとメッセージが表示され、「インシデント管理」パネルが更新されます。新しいインシデントが作成されると、「オープン・インシデント」レポートの最初にリストされます。

ユーザーへの割り当て

1. いずれかの「インシデント管理」レポートで、別のユーザーに割り当てるインシデントをダブルクリックします。
2. ドリルダウン・メニューから「ユーザーに割り当て」を選択します。 このメニューを選択すると、ユーザーのリストと 1 つの追加オプション「割り当て解除」を含む新しいメニューが表示されます。
3. ユーザーを選択してインシデントに割り当てます。あるいは、「割り当て解除」を選択して現在割り当てられているユーザーを削除します。 ユーザーが割り当てられている場合「状況の記述」は割り当て済みとなり、割り当てを解除すると「状況の記述」はオープンになります。

   変更が完了するとメッセージが表示され、「インシデント管理」パネルが更新されます。

重大度の変更

1. いずれかの「インシデント管理」レポートで、重大度を変更するインシデントをダブルクリックします。
2. ドリルダウン・メニューから「重大度の変更」を選択します。 このメニューを選択すると、 重大度コード (情報、低、中、高) のリストを含む新しいメニューが表示されます。
3. 新規の重大度コードを選択します。

   変更が完了するとメッセージが表示され、「インシデント管理」パネルが更新されます。

通知

1. いずれかの「インシデント管理」レポートで、ユーザーが通知を受け取るインシデントをダブルクリックします。
2. ドリルダウン・メニューから「通知」を選択します。 このメニューを選択すると、 ユーザーのリストを含む新しいメニューが表示されます。
3. ユーザーを選択します。

   ユーザーが通知を受けるとメッセージが表示されます。

状況の変更

1. いずれかの「インシデント管理」レポートで、状況を変更するインシデントをダブルクリックします。
2. ドリルダウン・メニューから「状況の変更」を選択します。 このメニューを選択すると、 状況コードのリストを含む新しいメニューが表示されます。
   - 割り当て済み - インシデントがこの状況になると、 ポリシー違反をこれ以上追加できません。ポリシー違反を追加するには、 インシデントの状況を「オープン」に戻し、違反を追加してから状況を「割り当て済み」に戻します。
   - クローズ済み - インシデントに「クローズ済み」のマークが付くと、変更ができなくなり、 リストに含まれなくなります。
   - オープン - 新規インシデントの初期の状況です。
3. 新規の状況コードを選択します。

   変更が完了するとメッセージが表示され、「インシデント管理」パネルが更新されます。

コメントの追加

1. いずれかの「インシデント管理」レポートで、コメントを追加するインシデントをダブルクリックします。
2. ドリルダウン・メニューから「コメント」を選択して、 「ユーザー・コメント」ウィンドウを開きます。 コメントの追加方法に関する説明については、『コメント』を参照してください。

親トピック: 保護

複数のデータベース・セキュリティー・インシデントのレビュー管理方法

インシデントの管理を行い、データベースのセキュリティー・インシデントをトラッキングして解決します。

このタスクについて

管理者は、一連の関連するポリシー違反をグループにして 1 つのインシデントとし、特定の個人に割り当てることができます。これにより、監視チームによるレビューが必要なポリシー違反の数が減らせます。

前提条件

- ポリシーを作成します (『ポリシー』を参照)。
- 検査エンジンを起動します (『検査エンジン構成』を参照)。

セキュリティー・ポリシーには、データベース・クライアントとサーバーとの間の監視対象トラフィックに適用される順序付きルール・セットが含まれています。

ポリシー違反は、ルールが起動されるごとにログに記録されます。 ポリシー違反は、プロセスによって自動的に、または権限を持つユーザーによって手動でインシデントに割り当てられます (『インシデント管理』を参照してください)。

手順の要約

1. 「順守」 > 「ツールとビュー」 > 「インシデント生成」をクリックして、「インシデント生成プロセス」を開きます。
2. インシデント生成プロセス (照会、重大度、しきい値、スケジューリング) の編集。
3. 「インシデント管理」タブのレポートを参照します。

インシデント管理

インシデント管理アプリケーションには、データベースのセキュリティー・インシデントをトラッキングして解決するワークフロー自動化機能を持つビジネス・ユーザー・インターフェースがあります。

インシデントの生成プロセスを定義してスケジュールすることで、ポリシー違反ログの読み取りや、新規インシデントの生成が行えます。インシデントの生成プロセスでは、選択される各インシデントは次のようになります。

- 固有のインシデント番号が割り当てられる。
- ユーザーに割り当てられる。
- 重大度コードが割り当てられる。
- カテゴリーに割り当てられる。

さらに、ポリシー違反は、「ポリシー違反/インシデント管理」レポートから (権限を持つユーザーが) 手動で新規インシデントまたは既存のインシデントに割り当てることができます。

インシデントが生成されると、管理者および他のユーザーは、(admin ポータルとユーザー・ポータルに含まれる) 「インシデント管理」タブからインシデントを操作します。 ここからは、他のすべてのタスク (インシデントの割り当て、通知の送信、状況の割り当て、など) を実行できます。

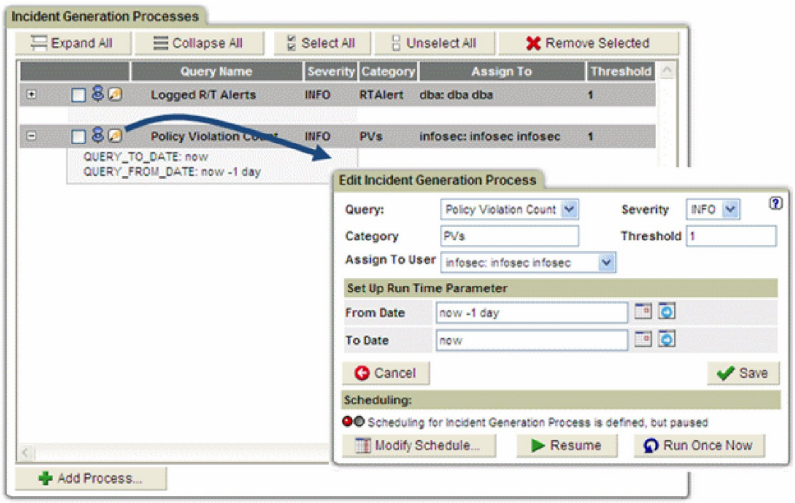
インシデント管理機能は、「インシデント管理」レポートのドリルダウン・メニューからアクセスできます。 各ユーザーは、ユーザー・アカウントに割り当てられたセキュリティー・ロールに応じて、 レポートや機能のサブセットのみ使用できます。

インシデント生成プロセスの定義

インシデントの生成プロセスでは、ポリシー違反ログへの照会が実行され、その照会に基づいてインシデントが生成されます。デフォルトでは、インシデント生成プロセスの定義およびスケジューリングは、admin ロールを持つユーザーに制限されます。

手順

1. 「順守」 > 「ツールとビュー」 > 「インシデント生成」をクリックして、「インシデント生成プロセス」を開きます。
2. 「プロセスの追加」ボタンをクリックして、「インシデント生成プロセスの編集」パネルを開きます。
3. 「照会」リストから照会を選択します。 インシデント生成プロセスで使用される照会に適用されるいくつかの制約があります。 照会 - レポート・ビルダーで照会を開き、次の条件を満たすことを確認します。
   - 照会はポリシー違反ドメインのものであること。
   - 照会のチェック・ボックスにチェックが付いていること。詳しくは、列表示の選択を参照してください。
   - 照会のメイン・エンティティーがポリシー・ルール違反エンティティーであること。
   - 照会の照会フィールドに SQL 文字列 (SQL エンティティーのもの、 またはポリシー・ルール違反エンティティーの「SQL 文字列全体」属性のもの) が含まれないこと。
4. インシデントの重大度を選択します (デフォルトは「情報」)。
5. オプションで、インシデントのカテゴリーを入力します (デフォルトは「なし」)。
6. オプションで、インシデント生成のしきい値を入力します。 デフォルトは 1 で、 照会で返されるすべての「行」がインシデントを生成します。
7. 「ユーザーに割り当て」リストから、インシデントを割り当てるユーザーを選択します。
8. 照会の開始日と終了日を入力します。 スケジュールした照会の場合、相対日付を使用します (例: 「現在 -1 日」や「現在」)。
9. 「保存」をクリックして、プロセスの定義を保存します。 プロセスは、保存しないと実行またはスケジュールできません。
10. ここで照会を実行するには、「今すぐ 1 回実行」をクリックします。
11. 照会をスケジュールするには、「スケジュールの変更」をクリックして、スケジューリング・ユーティリティーを開きます。スケジューラーの使用手順については、共通ツール・ブックの『スケジューリング』を参照してください。

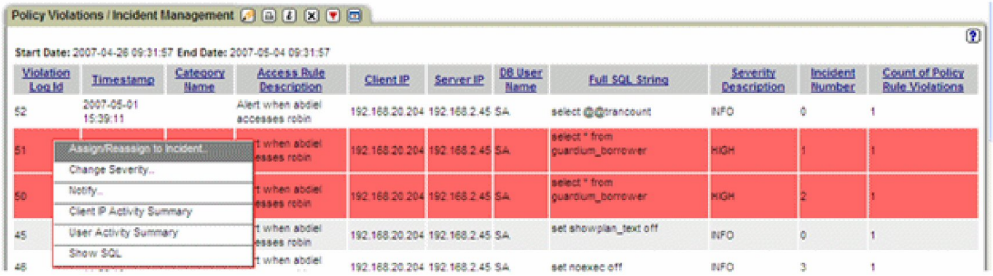


12. インシデントへの割り当て/再割り当て - いずれかの「インシデント管理」レポートで、割り当て /再割り当てを行うポリシー違反をダブルクリックします。
13. ドリルダウン・メニューから「インシデントに割り当て/再割り当て」を選択します。 このメニューを選択すると、オープン・インシデント (例えば、「インシデント #123 に割り当て」) のリストと 1 つの追加オプション (「新規インシデントに割り当て」) を含む新しいメニューが表示されます。
14. この違反を割り当てるインシデントを選択するか、「新規インシデントに割り当て」を選択して次の使用可能なインシデント番号 (順に番号が付けられます) にポリシー違反を割り当てます。

    変更が完了するとメッセージが表示され、「インシデント管理」パネルが更新されます。 新しいインシデントが作成されると、「オープン・インシデント」レポートの最初にリストされます。

    インシデントの「ポリシー違反/インシデント管理」レポートからは、以下が行えます。

    - インシデントへの割り当て/再割り当て (このポリシー違反からインシデントを作成)。
    - インシデントの重大度の変更。
    - 1 名以上のユーザーにインシデントを通知する。
    - インシデントからクライアント IP アクティビティー、ユーザー・アクティビティー、または SQL のレポートを表示する。



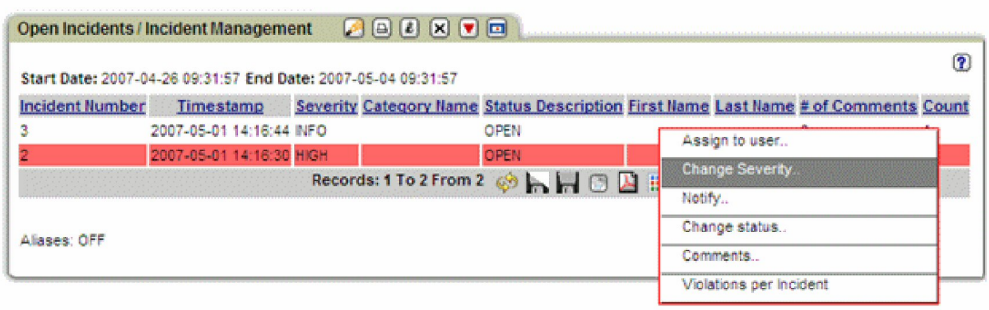
15. ユーザーに割り当てる - いずれかの「インシデント管理」レポートで、別のユーザーに割り当てるインシデントをダブルクリックします。
16. ドリルダウン・メニューから「ユーザーに割り当て」を選択します。 このメニューを選択すると、ユーザーのリストと 1 つの追加オプション「割り当て解除」を含む新しいメニューが表示されます。
17. ユーザーを選択してインシデントに割り当てます。あるいは、「割り当て解除」を選択して現在割り当てられているユーザーを削除します。 ユーザーが割り当てられている場合「状況の記述」は割り当て済みとなり、割り当てを解除すると「状況の記述」はオープンになります。

    変更が完了するとメッセージが表示され、「インシデント管理」パネルが更新されます。

18. 重大度の変更 - いずれかの「インシデント管理」レポートで、重大度を変更するインシデントをダブルクリックします。
19. ドリルダウン・メニューから「重大度の変更」を選択します。 このメニューを選択すると、 重大度コード (情報、低、中、高) のリストを含む新しいメニューが表示されます。
20. 目的の重大度コードを選択します。

    変更が完了するとメッセージが表示され、「インシデント管理」パネルが更新されます。

    ポリシー違反をインシデントに割り当てると、そのインシデントは「オープン・インシデント」レポートに表示されます。 「オープン・インシデント」レポートからは、次のようなアクションを実行できます。



21. 通知 - いずれかの「インシデント管理」レポートで、ユーザーが通知を受け取るインシデントをダブルクリックします。

22. ドリルダウン・メニューから「通知」を選択します。 このメニューを選択すると、 ユーザーのリストを含む新しいメニューが表示されます。
23. ユーザーを選択します。

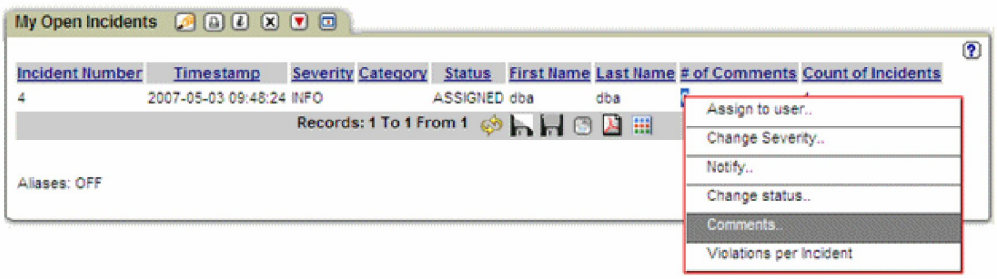
    通知が行われると、メッセージが表示されます。

24. 状況の変更 - いずれかの「インシデント管理」レポートで、状況を変更するインシデントをダブルクリックします。
25. ドリルダウン・メニューから「状況の変更」を選択します。 このメニューを選択すると、 状況コードのリストを含む新しいメニューが表示されます。
    - 割り当て済み - インシデントがこの状況になると、 ポリシー違反をこれ以上追加できません。ポリシー違反を追加するには、インシデントの状況を「オープン」に戻し、違反を追加してから状況を「割り当て済み」に戻します。
    - クローズ済み - インシデントに「クローズ済み」のマークが付くと、変更ができなくなり、 リストに含まれなくなります。
    - オープン - 新規インシデントの初期の状況です。
26. 目的の状況コードを選択します。

    変更が完了するとメッセージが表示され、「インシデント管理」パネルが更新されます。

27. コメントの追加 - いずれかの「インシデント管理」レポートで、コメントを追加するインシデントをダブルクリックします。
28. ドリルダウン・メニューから「コメント」を選択して、「ユーザー・コメント」ウィンドウを開きます。コメントの追加手順については、『コメント』を参照してください。

    ユーザー・ポータルに、それぞれそのユーザーの「マイ・オープン・インシデント」レポートが表示されます。 「マイ・オープン・インシデント」レポートからは、次のようなアクションを実行できます。



親トピック: 保護

照会再書き込み

照会再書き込み機能を使用してデータベース照会をインターセプトし、そのデータベース照会をセキュリティー・ポリシーで定義された条件に基づいて再書き込みすることにより、データベースに対するアクセス権を詳細に制御することができます。

照会の変更は、処理中に透過的に行われるため、照会を発行したユーザーが、再書き込みされた SQL ステートメントに基づく結果をシームレスに受信できます。

照会再書き込み機能は、照会の変更方法や補完方法を示す照会再書き込み定義と、その照会再書き込み定義を適用する特定の状況を示すランタイム・コンテキストを組み合わせることによって実装されます。

データベース照会に対して処理中に再書き込みを行うことで、管理者は、以下の例に示すいくつかのタイプのアクセス制御を実装できるようになります。

表 1. 照会再書き込みによるアクセス制御の例

アクセス制御	元の SQL	再書き込み後の SQL
WHERE 節を追加して行へのアクセスを制限する	`SELECT C from T`	`SELECT C from T WHERE [values]`
SELECT リストを変更して列へのアクセスを制限する	`SELECT C1 from T`	`SELECT C2 from T`
	`SELECT C1,C2 from T`	`SELECT C2 from T`
SQL ステートメントに再書き込みして何もしないようにすることで、データベース・アクティビティーを制限する	`SELECT EMAIL from T`	`SELECT++ EMAIL from T`
照会 verb (SELECT、INSERT、UPDATE など) を変更して、ユーザーが実行できる操作を制限する	`DROP TABLE T`	`UPDATE T SET [values]`
照会オブジェクト (TABLE、VIEW、COLUMN など) を変更して、ユーザーが実行できる操作を制限する	`SELECT C from T1`	`SELECT C from T2`

データベース照会の再書き込みをシームレスに実行できることにより、非常に強力かつ柔軟性の高い方法でアクセス制御を適用できるため、組織はセキュリティーについてのさまざまな懸念に迅速に対応できます。例えば、照会再書き込み定義を作成することで、以下の対策を実施することができます。

- 複数のユーザーおよびアプリケーションが単一のデータベースを共有するが、すべてのユーザーおよびアプリケーションにすべてのデータへのアクセス権限を与えるわけではないマルチテナンシー・シナリオで、セキュリティーを適用する
- データベース全体を公開せずに、テスト目的でデータベースを実稼働環境に公開する
- 脆弱性に対するデータベース・レベルまたはアプリケーション・レベルの永続的な解決策を考案している間に、重大なセキュリティー上の脆弱性を直ちに修正する

以下のセクションを参照して、照会再書き込みのしくみや、Guardium 環境で照会再書き込みを使用できるように構成する方法に関する詳細情報を確認してください。

注: S-TAP が firewall_default_state=1 (照会再書き込みのデフォルト状態) に設定されている場合、qrw_default_state=1 を同時に設定することはできません。

- 照会再書き込みのしくみ
  照会再書き込み機能が Guardium でどのように実装されているかについて説明します。

- 照会再書き込みの使用
  照会再書き込み機能を有効化して使用する方法について説明します。

親トピック: 保護

照会再書き込みのしくみ

照会再書き込み機能が Guardium でどのように実装されているかについて説明します。

概要

S-TAP でサポート対象データベース・サーバーに対して照会再書き込みが有効化されると (照会再書き込みの有効化参照)、以下の 3 つのポリシー・ルール・アクションを通じて照会再書き込み機能が実装されます。

- 照会再書き込み: アタッチ (QUERY REWRITE: ATTACH)
- 照会再書き込み: 定義の適用
- 照会再書き込み: デタッチ (QUERY REWRITE: DETACH)

これらのルール・アクションは、アクセス・ポリシー・ルールとしてインストールされます。これらのアクセス・ポリシー・ルールは、照会の再書き込み方法を示す照会再書き込み定義と、それらの定義がいつ適用されるのかを示すランタイム・コンテキストの両方を指定します。

照会再書き込みルールが指定されると、セッションが以下のように処理されます。

1. SQL 要求が「照会再書き込み: アタッチ (QUERY REWRITE: ATTACH)」ルールをトリガーし、セッションに含まれる以降のすべてのアクティビティーが、照会再書き込みによって監視されます。
2. 照会再書き込みによってセッションの監視が行われている間、トラフィックは S-TAP で保持され、セッション情報がアクセス・ポリシー・ルールと比較して検査されます。
3. 監視対象セッション内の照会が「照会再書き込み: 定義の適用」ルールに一致した場合、その照会は、定義に従って再書き込みが行われ、その後 S-TAP に送信されます。
4. S-TAP が再書き込みされた照会をデータベース・サーバーにリリースします。
5. 「照会再書き込み: デタッチ (QUERY REWRITE: DETACH)」ルールがトリガーされると、照会再書き込みがセッションの残りの照会に対する監視を中止するか、「照会再書き込み: アタッチ (QUERY REWRITE: ATTACH)」ルールが再度トリガーされるまで監視を中止します。

要件と制限事項

照会再書き込みは、以下のデータベース・サーバーと連動するよう意図されています。

- Oracle
- Db2 (Linux および Unix のみ)
- Microsoft SQL

サポートされるデータベース・サーバーおよび関連する制約事項については、『Platforms supported for IBM Guardium 10.1』を参照してください。照会再書き込みに対するデータベース・クライアント・サポートについて詳しくは、IBM Guardium サポートにお問い合わせください。

重要: 照会再書き込みでセッションを監視するときは、セッション内の各 SQL 要求について S-TAP にエンジンの判定を送信するために、スニファーが必要です。このプロセスは非同期で行われ、スニファーと S-TAP の間に待ち時間が発生します。パフォーマンスの影響を受けやすいアプリケーションやトラステッド・アプリケーションの場合は、セッションへのアタッチを防止する照会再書き込みのルール条件を作成してください。
親トピック: 照会再書き込み
関連タスク:
照会再書き込みの有効化

照会再書き込みの使用

照会再書き込み機能を有効化して使用する方法について説明します。

このタスクについて

照会再書き込み機能を有効化して使用を開始するには、以下のタスク・シーケンスを実行します。

1. 照会再書き込みの有効化
   照会再書き込み機能を使用できるように S-TAP を構成する方法について説明します。
2. 照会再書き込み定義の作成
   データ・マスキングやアクセス制御を行う場合に照会再書き込み定義を作成する方法について説明します。
3. 照会再書き込み定義のテスト
   サンプル入力に対して照会再書き込み定義をテストし、再書き込み定義が期待どおりに動作することを確認する方法について説明します。
4. 照会再書き込みをアクティブ化するセキュリティー・ポリシーの定義
   照会再書き込み定義を実際の照会に対して使用するアクセス・ポリシー・ルールを作成する方法について説明します。
5. 照会再書き込み結果を検証するためのカスタム・レポートの作成
   照会再書き込みアクティビティーを監査するための照会再書き込みトラッキング・レポートを作成する方法について説明します。

親トピック: 照会再書き込み

照会再書き込みの有効化

照会再書き込み機能を使用できるように S-TAP を構成する方法について説明します。

始める前に

照会再書き込み機能は、以下の場合にのみ実行されます。

- guard_tap.ini ファイルで照会再書き込みが有効化されている。
- 照会再書き込みポリシー・ルールが存在し、セッション・トラフィックによってトリガーされる。

このタスクについて

照会再書き込みは、S-TAP guard_tap.ini ファイルの [TAP] セクションのパラメーター qrw_installed によって制御されます。以下のいずれかの値を指定できます。

- 0=インストール済みポリシーのルールでトリガーされると、セッションごとに QRW がアクティブ化されます。
- 1=インストール済みポリシーに関係なく、すべてのセッションに対して QRW がアクティブ化されます。
- 2=デフォルトですべてのトラフィックに対して QRW ポリシー違反の監視が行われますが、最初のパケットから PRIORITY_COUNT で指定された数までのパケットでイベントによって監視がトリガーされなければ、セッションの照会再書き込みはオフになります。

  2 に設定される場合、コマンド Watch、Drop、Watch & Drop、および Unwatch によって QRW 操作を変更できます。状態 2 が有効になっているときに Watch コマンドを受け取ると、状態は 2 から 1 に変更され、接続は永続的にファイアウォールや照会再書き込み操作の影響を受けます。Drop または Watch & Drop を受け取ると、接続は即時に強制終了されます。状態 2 が有効になっているときに Unwatch コマンドを受け取ると、状態は 2 から 0 に変更され、接続はファイアウォールおよび照会再書き込み操作の影響を受けなくなります。

手順

1. root アカウントを使用して、データベース・サーバー・システムにログオンします。
2. S-TAP を停止します。
3. guard_tap.ini 構成ファイルのバックアップ・コピーを作成します。デフォルトのファイルの場所は、Windows では ¥Program Files¥IBM¥Windows S-TAP¥Bin¥、Linux では /usr/local/guardium/guard_stap/guard_tap.ini です。
4. guard_tap.ini をテキスト・エディターで開きます。
5. パラメーター qrw_installed = 0 を見つけ、必要に応じて編集します。
6. guard_tap.ini への変更を保存します。
7. 検査エンジンを再始動します。そのためには、「管理」 > 「アクティビティー・モニター」 > 「検査エンジン」にナビゲートして、「検査エンジンの再始動」をクリックします。あるいは、CLI にユーザーとしてログインし、restart_inspection_engines CLI コマンドを使用して検査エンジンを再始動します。

タスクの結果

このタスクが完了すると、照会再書き込み機能が有効化され、照会再書き込みアクションを含むポリシー・ルールに応答するようになります。
親トピック: 照会再書き込みの使用
次のトピック: 照会再書き込み定義の作成

照会再書き込み定義の作成

データ・マスキングやアクセス制御を行う場合に照会再書き込み定義を作成する方法について説明します。

手順

1. 「保護」 > 「セキュリティー・ポリシー」 > 「照会再書き込みビルダー」を開きます。
2. 照会再書き込み定義の分かりやすい固有の名前を「名前」フィールドに入力します。
3. モデル照会を作成して解析します。
   a. 「モデル照会を入力」フィールドにモデル照会を入力します。

      例えば、`SELECT * from` ステートメントの使用を禁止する再書き込み定義を作成するには、モデルとして `SELECT * from EMPLOYEE` と入力します。

   b. 「データベース・タイプ」メニューをクリックし、モデル照会に使用する SQL パーサーを選択します。
   c. 「解析」をクリックしてモデル照会を処理します。

      モデル照会が個別のコンポーネントに分解されます。このとき、アクション可能な各コンポーネントが下線付きテキストで強調表示されます。

4. モデル照会の特定のコンポーネントに再書き込みを行う方法を定義します。
   a. 解析された照会の下線付きコンポーネントのうち、再書き込みを行うコンポーネントをクリックします。照会再書き込み定義を作成できるようにするためのダイアログが開きます。

      オプション:

      - 解析された照会の個別の verb、フィールド、またはオブジェクトを選択して変更します
      - 照会にコンポーネントを追加します (解析された照会の横にグレーの下線付きテキストとして表示されます)
      - 解析された照会の横にあるグレーの下線付きの [R] をクリックして、照会全体を書き換えます

      `SELECT * from EMPLOYEE` の例 (`SELECT * from` ステートメントの使用を禁止する) では、「*」をクリックして再書き込み内容を指定します。

   a. 「変更前:」フィールドは再書き込みの対象を示します。
   b. 「終了」フィールドは再書き込み後のコンポーネントを定義します。

      例えば、`SELECT * from` ステートメントの使用を禁止するには、`*` コンポーネントを特定オブジェクトのリスト (`EMPNO, FIRSTNME, MIDINIT, LASTNAME, WORKDEPT, PHONENO, HIREDATE, JOB, EDLEVEL, SEX`) に置き換えます。

      重要:

再書き込み定義は構文に基づいているため、`SELECT * from [OBJECT]` という形式のすべてのステートメントが例に一致します。例えば、`SELECT * from DEPARTMENT` というステートメントと `SELECT * from EMPLOYEE` というステートメントは、どちらも上記の例に一致します。

照会再書き込み定義を特定のオブジェクトに制限するには、アクセス・ポリシー・ルールを使用します。その方法については、照会再書き込みをアクティブ化するセキュリティー・ポリシーの定義を参照してください。

c. 「保存」をクリックして再書き込み定義を保存し、次に「戻る」をクリックしてダイアログを閉じます。

5. 「リアルタイム・プレビュー」フィールドを使用して照会再書き込み定義の出力を確認し、必要に応じて変更を加えます。

上記の例では、`SELECT * from EMPLOYEE` が `SELECT EMPNO, FIRSTNME, MIDINIT, LASTNAME, WORKDEPT, PHONENO, HIREDATE, JOB, EDLEVEL, SEX from EMPLOYEE` として再書き込みされます。

6. 結果に問題がなければ、「保存」をクリックして照会再書き込み定義を保存します。

照会再書き込み定義が保存され、「照会再書き込みビルダー」の使用可能な照会再書き込み定義のリストに表示されます。

次のタスク

照会再書き込み定義の操作を続けます。

- 追加の定義を作成するには、「新規」をクリックして、このタスクの手順を繰り返します。
- 既存の照会再書き込み定義を編集するには、使用可能な照会再書き込み定義のリスト内の項目をダブルクリックします。
- 既存の照会再書き込み定義をコピーして編集するには、使用可能な照会再書き込み定義のリスト内の項目を選択し、「コピー」をクリックします。
- 既存の照会再書き込み定義を削除するには、使用可能な照会再書き込み定義のリスト内の項目を選択し、「削除」をクリックします。

照会再書き込み定義の操作が完了したら、以下のシーケンスの次の手順に進んで、定義のテストと実装を行います。
親トピック: 照会再書き込みの使用
前のトピック: 照会再書き込みの有効化
次のトピック: 照会再書き込み定義のテスト
関連タスク:
照会再書き込みをアクティブ化するセキュリティー・ポリシーの定義

照会再書き込み定義のテスト

サンプル入力に対して照会再書き込み定義をテストし、再書き込み定義が期待どおりに動作することを確認する方法について説明します。

始める前に

このタスクを実行するには、1 つ以上の照会再書き込み定義が作成されている必要があります。

手順

1. 「保護」 > 「セキュリティー・ポリシー」 > 「照会再書き込みビルダー」を開きます。
2. 「テストのセットアップ」をクリックしてダイアログを開き、テストする照会再書き込み定義を選択します。
   a. 「使用可能な照会再書き込み定義」フィールドから「照会再書き込み定義のテスト」フィールドに項目をドラッグ・アンド・ドロップします。
   b. 「照会再書き込み定義のテスト」フィールドで項目をドラッグ・アンド・ドロップして、複数の定義をアクセス・ポリシーと同様に並べ替えます。
   c. 完了したら、「保存」をクリックしてダイアログを閉じます。
3. 「テスト」フィールドにテスト照会を入力するか貼り付けます。

例えば、`SELECT * from` ステートメントの使用を禁止する再書き込み定義 (照会再書き込み定義の作成を参照) をテストするには、以下のようなサンプル照会を入力します。

```
SELECT * from DEPARTMENT
SELECT * from EMPLOYEE
SELECT FIRSTNME, case
when SALARY > 150000 then 'high'
when SALARY > 100000 then 'medium'
when SALARY > 80000 then 'fair'
else 'poor'
end from EMPLOYEE
DELETE from EMPLOYEE where EMPNO=100
INSERT into TEMP_EMP SELECT * from EMPLOYEE
```

4. 「テストの実行」をクリックしてサンプル照会を処理し、結果を確認します。

例えば、前のステップで示したサンプル照会をテストすると、以下の結果が返されます。

表 1. 照会再書き込みのテスト結果

元の SQL	再書き込み後の SQL	変更
`SELECT * from DEPARTMENT`	`SELECT EMPNO, FIRSTNME, MIDINIT, LASTNAME, WORKDEPT, PHONENO, HIREDATE, JOB, EDLEVEL, SEX from DEPARTMENT`	YES
`SELECT * from EMPLOYEE`	`SELECT EMPNO, FIRSTNME, MIDINIT, LASTNAME, WORKDEPT, PHONENO, HIREDATE, JOB, EDLEVEL, SEX from EMPLOYEE`	YES
`SELECT FIRSTNME, case when SALARY > 150000 then 'high' when SALARY > 100000 then 'medium' when SALARY > 80000 then 'fair' else 'poor' end from EMPLOYEE`	`SELECT FIRSTNME, case when SALARY > 150000 then 'high' when SALARY > 100000 then 'medium' when SALARY > 80000 then 'fair' else 'poor' end from EMPLOYEE`	NO

元の SQL	再書き込み後の SQL	変更
DELETE from EMPLOYEE where EMPNO=100	DELETE from EMPLOYEE where EMPNO=100	NO
INSERT into TEMP_EMP SELECT * from EMPLOYEE	INSERT into TEMP_EMP SELECT * from EMPLOYEE	NO

重要:

再書き込み定義は構文に基づいているため、SELECT * from [OBJECT] という形式のすべてのステートメントが例に一致します。例えば、SELECT * from DEPARTMENT というステートメントと SELECT * from EMPLOYEE というステートメントは、どちらも上記の例に一致します。

照会再書き込み定義を特定のオブジェクトに制限するには、アクセス・ポリシー・ルールを使用します。その方法については、照会再書き込みをアクティブ化するセキュリティー・ポリシーの定義を参照してください。

5. 引き続きサンプル照会を入力して、再書き込み定義をテストします。「テストのセットアップ」をクリックして、テストに使用する再書き込み定義を変更するか並べ替えます。

次のタスク

テスト結果に問題がなければ、セキュリティー・ポリシーを作成して、実際の照会への照会再書き込み定義の使用を開始します。
親トピック: 照会再書き込みの使用
前のトピック: 照会再書き込み定義の作成
次のトピック: 照会再書き込みをアクティブ化するセキュリティー・ポリシーの定義
関連タスク:
照会再書き込みをアクティブ化するセキュリティー・ポリシーの定義
照会再書き込み定義の作成

照会再書き込みをアクティブ化するセキュリティー・ポリシーの定義

照会再書き込み定義を実際の照会に対して使用するアクセス・ポリシー・ルールを作成する方法について説明します。

始める前に

このタスクを実行するには、1 つ以上の照会再書き込み定義が作成およびテストされている必要があります。また、セキュリティー・ポリシーの作成方法を理解しておく必要があります。

手順

1. 「保護」 > 「セキュリティー・ポリシー」 > 「ポリシー・ビルダー」を開きます。
2. 新規ポリシーを作成するか、既存のポリシーを変更して、照会再書き込み定義を使用します。
   ヒント: 新規ポリシーを作成して、照会再書き込み定義をテストすることを検討してください。テスト・ポリシーの動作に問題がなければ、既存のセキュリティー・ポリシーに再書き込みルールを追加します。
3. 選択したポリシーへの再書き込みルールの追加を開始するには、「ルールの編集」をクリックして、「ルールの追加」 > 「アクセス・ルールの追加」を選択します。
   注: 照会再書き込みルールは、常にアクセス・ルールとして分類されます。
4. 「照会再書き込み: アタッチ (QUERY REWRITE: ATTACH)」ルール・アクションを使用してルールを追加します。「次のルールに進む」チェック・ボックスに必ずチェック・マークを付けてください。このルールは、照会再書き込みセッションをトリガーするために突き合わせる必要のある特定のセッション・パラメーターを識別します (特定のデータベース・ユーザー名やクライアント IP アドレスなど)。
5. 1 つ以上の「照会再書き込み: 定義の適用」ルール・アクションを使用してルールを追加し、適用する照会再書き込み定義を選択します。このルールは、再書き込み定義を適用して元の照会を変更するために突き合わせる必要のある特定のオブジェクトまたはコマンドを識別します。

   例えば、SELECT * from EMPLOYEE 照会の発行時にユーザーに返されて表示されるデータを制限できます。そのためには、「オブジェクト」フィールドを EMPLOYEE に設定して、* をユーザーにアクセス権を付与するデータの定義済みの列のリストに置き換えて、照会再書き込み定義を作成します。

6. 「照会再書き込み: デタッチ (QUERY REWRITE: DETACH)」ルール・アクションを使用してルールを追加します。これにより、照会再書き込みセッションが切断され、以降はセッション・トラフィックのモニターが行われなくなります。デタッチ・ルールに設定する条件は、アタッチ・ルールと同じ条件にしないでください。
7. 新規ポリシーをインストールするには、「ポリシー・ファインダー」に戻り、セキュリティー・ポリシーを選択して、「インストール・アクションの選択 (Select an installation action)」 > 「インストールおよびオーバーライド」を選択します。ポリシーのインストールを確認するよう求められたら、「OK」をクリックします。
8. データベース・サーバーにログインし、テスト照会を実行して、アクセス・ポリシーの再書き込みルールが意図したとおりに機能していることを確認します。
   a. データベース・サーバーにログインします。
   b. インストールしたアクセス・ポリシー・ルールをトリガーする (またはトリガーしない) 照会を実行し、照会再書き込み定義の基準と突き合わせます。

      例えば、「オブジェクト」を EMPLOYEE に設定して SELECT * from EMPLOYEE を発行すると、照会再書き込み定義で * に定義した列の結果のみが表示されます。一方、SELECT * from DEPARTMENT を発行すると、DEPARTMENT オブジェクトについて返されるすべての列のデータが表示されます。

   c. 再書き込みされた SQL が結果に反映されていることを確認します。

親トピック: 照会再書き込みの使用
前のトピック: 照会再書き込み定義のテスト
次のトピック: 照会再書き込み結果を検証するためのカスタム・レポートの作成
関連概念:
ポリシー

照会再書き込み結果を検証するためのカスタム・レポートの作成

照会再書き込みアクティビティーを監査するための照会再書き込みトラッキング・レポートを作成する方法について説明します。

始める前に

このタスクを実行するには、照会再書き込み定義を適用するアクセス・ポリシー・ルールが作成およびインストールされている必要があります。また、レポートの作成方法を理解しておく必要があります。

このタスクについて

照会再書き込みトラッキング・レポートは、テスト環境と実稼働環境の両方における照会再書き込みアクションの検証に役立ちます。

手順

1. 照会 - レポート・ビルダーの使用の指示に従って、新規照会を作成します。
2. 以下のメイン・エンティティーのいずれかを選択します。
   - 照会再書き込みログ (Query Rewrite Log)
   - クライアント/サーバー
   - セッション
   - アクセス期間
3. 照会再書き込みレポートの開始点として以下の項目を追加してください。
   - クライアント/サーバー: タイム・スタンプ
   - クライアント/サーバー : DB ユーザー名
   - クライアント/サーバー: サーバー・タイプ
   - 照会再書き込みログ (Query Rewrite Log): 適用される QR 定義名 (Applied QR Definition Names)
   - 照会再書き込みログ (Query Rewrite Log): 入力 SQL (Input SQL)
   - 照会再書き込みログ (Query Rewrite Log): 出力 SQL (Output SQL)
4. 完了したらレポートを保存します。
5. 「マイ・カスタム・レポートに追加」をクリックしてレポートをカスタム・レポートに追加します。
6. 「レポート」 > 「マイ・カスタム・レポート」を開き、作成したレポートを選択して、照会再書き込みアクションのレポートを表示します。

親トピック: 照会再書き込みの使用
前のトピック: 照会再書き込みをアクティブ化するセキュリティー・ポリシーの定義

ファイル・アクティビティーのポリシーおよびルール

ファイル・アクティビティー・モニターは、UNIX ファイル・サーバーおよび Windows ファイル・サーバー上の機密データの保全性と保護を確保します。

- ファイル・アクティビティーのポリシーおよびルールの機能
  ファイル・アクティビティー・モニターのポリシーは、Guardium で各種のファイル・アクティビティー・イベントをどのように処理するかを指定します。各ポリシーは、順序付けられた一連のルールで構成されます。ポリシー内の各ルールは、そのルールと一致した場合に実行される条件付きアクションを定義します。条件付きテストは単純なものにすることも (ある特定のユーザーは特定の場所にアクセスするなど)、複数の条件を考慮する複雑なテストにすることもできます。アクションは、何もしないというものから、イベントをブロックするというアクションまで、多岐に及びます。複数のグループ化アクションとアラート・アクションを結合して、一致したルールに対する高度な応答を作成するように指示することができます。
- FAM ポリシーおよびそのルールを初めから作成する
  ファイル・アクティビティー・モニターをセットアップするには、「ファイルのためのポリシー・ビルダー」ウィンドウでポリシーとルールを定義して管理します。
- 調査ダッシュボードの「資格」タブでの FAM ポリシー・ルールの作成
  調査ダッシュボードの結果表のモニター対象データ (データ・ソース名、ユーザー名、アクション、ファイル・パスなど) を使用して、ポリシー・ルールを作成できます。

親トピック: 保護
関連情報:
Guardium を使用したファイル・アクティビティーのモニター (ビデオ)

ファイル・アクティビティーのポリシーおよびルールの機能

ファイル・アクティビティー・モニターのポリシーは、Guardium で各種のファイル・アクティビティー・イベントをどのように処理するかを指定します。各ポリシーは、順序付けられた一連のルールで構成されます。ポリシー内の各ルールは、そのルールと一致した場合に実行される条件付きアクションを定義します。条件付きテストは単純なものにすることも (ある特定のユーザーは特定の場所にアクセスするなど)、複数の条件を考慮する複雑なテストにすることもできます。アクションは、何もしないというものから、イベントをブロックするというアクションまで、多岐に及びます。複数のグループ化アクションとアラート・アクションを結合して、一致したルールに対する高度な応答を作成するように指示することができます。

例えば、以下のケースに対するポリシーを定義できます。

- John が CONFIDENTIAL フォルダーに書き込んだ場合にポリシー違反をログに記録する。
- 特定のグループのユーザーがファイル SALARIES.XLS を削除するのをブロックする。
- JENNY が、名前が sample* で始まるファイルから読み取りを行った場合に Krishna に E メールを送信する。
- PCI に関連する機密データが含まれているとして分類されたファイルへのすべてのアクセスを監査する。

グループ: Guardium では、ポリシーおよびレポートの作成のためにグループという概念が使用されます。

Guardium グループは、Guardium コレクターまたは中央マネージャーで作成され、保持されます。Guardium グループとファイル・システム・グループを混同しないでください。

グループの命名方法を考慮することをお勧めします。例えば、データ・ソース (ファイル・サーバー) のグループ、ファイルのグループ (機密レベル別、または機密レベルとアプリケーションの組み合わせ別など)、ユーザーのグループ (既知のすべてのユーザー、許可されたユーザー、特権があるユーザーのリスト) などです。

ルールのガイドライン

- 範囲が広すぎるルール (モニターするファイルが多すぎるルール) は、システムが過負荷になり、処理時間と応答時間が長くなる可能性があります。
- 1 つの FAM ルールに複数のパターンを含めることができます。ディレクトリーとその内容の両方を保護するには、「/FAMtest/*」と「/FAMtest」という 2 つのパターンを持つルールを定義します。
- ファイル・パスからなるグループの場合、大/小文字とは関係なく、各パスが固有でなければなりません。例えば、1 つのグループに、C:¥ABC と C:¥abcdef という 2 つのパスを共存させることはできます。一方、1 つのグループに C:¥ABC と C:¥abc という 2 つのパスを共存させることはできません。グループ・ビルダーは大/小文字を区別しません。そのため、すべて大文字またはすべて小文字でメンバーを入力する必要はありません。 ただし、大/小文字を区別する UNIX では、パス /IBM/Guardium はパス /ibm/guardium と異なります。 これら両方のパスをモニターしたい場合、現在のグループ・ビルダーには制限があり、これらのパスは別個の 2 つのパスとして見なされません。
- セキュリティー・ポリシーのルールの順序は非常に重要です。ルールはセットとして S-TAP に送信され、厳密に順番通りに処理されます。所定のユーザー・アクティビティーが、ポリシー内の各ルールと順番に照合されチェックされます。このファイル・アクセスの条件を満たす最初のルールが適用され、後続のルールは無視されます。ほとんどの場合、固有性の最も高いルールを最初に配置し、一般性の最も高いルールを最後に配置します。例えば、以下の 2 つのルールがあるとします。
  - **ルール A:** /data/* へのすべてのアクセスの*監査のみ* 行います。
  - **ルール B:** ユーザー「joe」が /data/salaries にアクセスするのを*ブロックし*、*違反をログに記録し*、*監査します*。

  ルール A を最初に配置した場合、Joe が /data/salaries を読み取ろうとすると、次のルールに進む必要はなく、Joe のアクセスの監査のみ行われます。ルール B を最初に配置すると、Joe の /data/salaries へのアクセスはブロックされ、次のルールに進む必要はありません。

(マルチアクション・サポートが組み込まれた) 10.1.2 以降のスニファーで (マルチアクションをサポートしていない) 10.1.2 より前の S-TAP を使用する場合の FAM の動作

- マルチアクション・ルールを使用する新しい 10.1.2 スニファー/UI で 10.1.2 より前の S-TAP を使用する場合、ブロッキングは正しく実装されます。このアクションは、S-TAP 側で行われるためです。
- スニファー側でのこのアクションは、指定されているすべてのアクションが累積されたものとなります。
- 例えば、READ コマンドに対して「監査のみ」を選択し、DELETE コマンドに対して「ブロック」、「違反のロギング」、および「監査」を選択すると、DELETE コマンドはブロックされますが、READ コマンドはブロックされません。一方、READ コマンドと DELETE コマンドの両方は、READ コマンドが「監査のみ」であっても、監査、違反のロギング、およびアラートの生成をトリガーします。
- ユーザーが 10.1.2 の S-TAP と 10.1.2 より前のスニファー/UI を使用する他の例では、マルチアクション・ルールを定義する手段はないため (したがって、サポートする UI や GuardAPI がないため)、問題なく機能します。

ルールの属性

ルール名

固有の名前

データ・ソース

データ・ソースには、以下を指定できます。

- ドロップダウン・リストから選択したデータ・ソース
- ドロップダウン・リストから選択したグループ
- 「新規ルールの作成」ウィンドウで、選択したグループから作成したグループ
- 手動で入力したパス

ルール・アクション

ルール・アクションは、基準を満たした場合に実行されるアクションです。アクションは、以下のいずれかです。

- ルール基準に一致するすべてのファイル・アクセスに対する単一のアクション
- 複数のアクション (指定されたコマンド・カテゴリーまたは指定されたグループごとに 1 つのアクション) からなるマルチアクション・ルール。マルチアクション・ルールを使用する場合、**「次のルールに進む」**はサポートされないことに注意してください。

以下のルール・アクションがあります。

- アラートおよび監査: 指定された動作によりスニファーから直接生成されたアラートを送信し、イベントをログに記録します。
- 監査のみ: GDM 表にイベントのログを記録します。
- ブロック、違反のロギング、および監査: オブジェクトへのアクセスをブロックし、ポリシー違反とイベントをログに記録します。ブロック・アクションでは、アラート構成も必要です。
- 無視: アクションはとられません。
- 違反として記録、および監査: 対象をポリシー違反としてログに記録し、イベントをログに記録します。

アクセス・コマンド: 数百のファイル・システム・コマンドがあることから、コマンドは以下のカテゴリーに分類されています。

- 読み取り
- 書き込み
- 実行
- 削除
- ファイル操作。これには、ファイル・メタデータに影響する呼び出し (ファイル所有権の変更、ファイル許可の変更、および類似する呼び出し) が含まれます。

これらのカテゴリーはシステムで固定されており、変更できません。ただし、カテゴリーの任意の組み合わせで含めた Guardium グループを作成し、そのグループをセキュリティー・ポリシーで使用できます。例えば、「書き込み」と「実行」をメンバーとして含めた Guardium グループを作成できます。

コマンドを未指定のままにすると、すべてのファイル・システム・コマンドが一致としてカウントされます。一部の呼び出し (システム時刻の取得など) は、ファイルにまったく影響せず、無視されます。

ルール基準

所定のファイル・アクセスに対して、ルール基準を使用して、特定のアクションを実行するかどうかが判断されます。データ・ソース、またはデータ・ソースのグループ (ファイル・サーバー) に対して指定できるルール基準を以下に示します。

ユーザー: ファイルにアクセスする OS ユーザーです。Guardium グループでの定義に従い、ユーザーのグループも使用できます。ブランクのままにすると、root 以外のすべてのユーザーにルールが適用されます。

ファイル・パス: Windows または UNIX のファイル・パス、個々のファイル・パス、または Guardium グループで定義されたファイル・パスのグループを使用できます。これをブランクにすることはできません (取り外し可能メディアが選択されている場合を除く)。ファイル・パスに含まれるサブディレクトリーをモニター対象として選択することもできます。

名前の指定でのワイルドカードは以下のようになります。

- 「*」文字は、任意の数の文字に一致します。
- 「?」文字は、単一の文字に一致します。
- UNIX では、円記号を使用して * および ? をエスケープします。

ヒント: ワイルドカードでは追加の処理が発生します。ワイルドカードを使用し過ぎると、パフォーマンスに影響します。

UNIX

使用法:

ディスク上のすべてのファイルに一致させるには、/* と入力します。

/tmp/My*File.txt に正確に一致させるには、/tmp/My¥*File.txt を使用します。

/tmp 内の .txt 拡張子のファイルと一致させるには、/tmp/*.txt を使用します。

例: 以下の FAM ルール・パターンを使用します。/FAM*

意味

- ディレクトリー: /
- ファイル名: FAM*

所定の位置の FAM ルールには、選択したサブディレクトリーがあります。(Subdirs: Yes)

以下のファイルがアクセスされます。

/guardium/modules/SUPERVISOR/10.0.0/FAM.output

この場合、「FAM.output」というファイル名が「FAM」という名前に一致します。このファイルは、指定されたディレクトリー「/」のサブディレクトリー内に存在しています。

Windows: Windows では、ドライブ (C:¥ など) を指定する必要があります。

使用法:

C ドライブ上のすべてのファイルをモニターするには、C:¥ と入力し、「サブディレクトリーのモニター (Monitor subdirectories)」チェック・ボックスにマークを付けます。

C:¥tmp 内の .txt 拡張子のファイルに一致させるには、C:¥tmp¥*.txt を使用します。

GuardAPI の例: 2 つのルールを使用したポリシーの作成

```
grdapi create_policy ruleSetDesc=policy1 isFam=true
grdapi create_fam_rule policyName=policy1 ruleName=rule1 serverHost="x.x.x.x" filePath="/famtest/*" command="DELETE"
actionName="Alert and Audit" notificationType="SYSLOG"
grdapi create_fam_rule policyName=policy1 ruleName=rule2 serverHost="x.x.x.x" filePath="/famtest/*" command="READ"
actionName="Alert and Audit" notificationType="MAIL"
```

policy1 -> rule1 -> "DELETE" -> "Alert and Audit" -> "SYSLOG"

policy1 -> rule2 -> "READ" -> "Alert and Audit" -> "MAIL"

GuardAPI の例: マルチアクション・ルールを使用したポリシーの作成

FAM のマルチアクション・ルール - マルチアクション・ルールは、複数のアクション (指定されたコマンド・カテゴリーまたは指定されたグループごとに 1 つのアクション) で構成されます。FAM のコンテキストでは、これらのコマンドは読み取り、書き込み、削除、実行、およびファイル操作です。システムがマルチアクション・ルールをサポートしていない場合、システムはルールを無視して次のルールに進みます。

```
grdapi create_policy ruleSetDesc=policy1 isFam=true
grdapi create_fam_rule policyName=policy1 ruleName=rule1 serverHost="x.x.x.x" filePath="/famtest/*"
add_action_to_fam_rule policyName=policy1 ruleName=rule1 command="DELETE, READ" actionName="Alert and Audit"
notificationType="SYSLOG"
add_action_to_fam_rule policyName=policy1 ruleName=rule1 command="WRITE" actionName="Alert and Audit" notificationType="MAIL"
```

policy1 -> rule1 -> "DELETE, READ" -> "Alert and Audit" -> "SYSLOG"

policy1 -> rule1 -> "WRITE" -> "Alert and Audit" -> "MAIL"

commandGroupId=20000 が存在し、"DELETE, WRITE" が含まれるという前提で、commandGroupId を使用して別のアクションを追加します。

```
add_action_to_fam_rule policyName=policy1 ruleName=rule1 command="READ" commandGroupId=20000 actionName="Ignore"
notificationType=""
```

policy1 -> rule1 -> "READ, DELETE, WRITE" -> "Ignore"

V.10.1.2 より前の S-TAP と V.10.1.2 以降のスニファーでの FAM の動作

FAM のマルチアクションは V.10.1.2 で導入されました。10.1.2 より前の S-TAP では FAM のマルチアクションがサポートされませんが、V.10.1.2 以降のスニファーではマルチアクションがサポートされます。スニファー側でのこのアクションは、指定されているすべてのアクションが累積されたものとなります。

例えば、ポリシーで READ コマンドには「監査のみ」が指定されていて、DELETE コマンドには「ブロック」、「違反のロギング」、および「監査」が指定されている場合、DELETE コマンドはブロックされますが、READ コマンドはブロックされません。一方、READ コマンドと DELETE コマンドの両方は、READ コマンドが「監査のみ」であっても、監査、違反のロギング、およびアラートの生成をトリガーします。

親トピック: ファイル・アクティビティーのポリシーおよびルール

FAM ポリシーおよびそのルールを初めから作成する

ファイル・アクティビティー・モニターをセットアップするには、「ファイルのためのポリシー・ビルダー」ウィンドウでポリシーとルールを定義して管理します。

このタスクについて

「ファイルのためのポリシー・ビルダー」を開き、ポリシー・ビルダー内で他のビューを開いた後は、ページの下部にある「ファイルのためのポリシー・ビルダー」、「新規ポリシー」、および「新規ルールの作成」をクリックすることで、各種のビューを切り替えることができます。

GuardAPI を使用してポリシーおよびルールを作成することもできます。

手順

1. スタンドアロンまたは MU で FAM ポリシー・ビルダーにアクセスします。「保護」 > 「セキュリティー・ポリシー」 > 「ファイルのためのポリシー・ビルダー」にナビゲートします。
2. 新規ポリシーの名前を入力します。(ルールの定義後にポリシーを保存できます)。
3. 既存のルールをポリシーに追加するには、以下のようにします。
   a. 「テンプレートの表示」をクリックします。「ルール・テンプレート」表が開きます。
   b. オプションで、フィルター機能を使用してリストをフィルタリングします。
   c. 1 つ以上のルールを選択して、右矢印 をクリックします。
4. 新規ルールを作成するには、以下のようにします。
   a. をクリックして「新規ルールの作成」ウィンドウを開きます。
   b. ルールの名前を入力し、その属性を定義してから、「保存」をクリックします。
5. 既存のルールを変更してからポリシーに追加するには、以下のようにします。
   a. ルールを選択し、 をクリックします。
   b. をクリックして、名前を変更し、必要に応じてその他の属性を変更してから、「保存」をクリックします。
6. ルールの順序を変更するには、 を使用します。
7. ルールを削除するには、ルールを選択してから ルールの削除 をクリックします。
8. 「保存」をクリックしてポリシーを保存するか、「保存およびインストール」をクリックしてポリシーを直ちにインストールします。(「ポリシー・インストール」ツールの使用を参照)。

親トピック: ファイル・アクティビティーのポリシーおよびルール
関連情報:
GuardAPI ファイル・アクティビティー・モニター関数

調査ダッシュボードの「資格」タブでの FAM ポリシー・ルールの作成

調査ダッシュボードの結果表のモニター対象データ (データ・ソース名、ユーザー名、アクション、ファイル・パスなど) を使用して、ポリシー・ルールを作成できます。

始める前に

- FAM バンドルをインストールして構成する必要があります。
- ディスカバリーおよび分類を有効にする必要があります。
- 調査ダッシュボードを有効にする必要があります (調査ダッシュボードの有効化と無効化を参照)。

このタスクについて

手順

1. 製品バナーのドロップダウン・リストから「ファイル」を選択し、検索アイコンをクリックして、ファイル・データの調査ダッシュボードを開きます。
2. 結果表の「資格」タブを開きます。「詳細」をクリックして個々のエントリーを表示します。
3. ルールを取り込むために使用する、結果内の 1 つ以上のエントリーを選択します。「すべて選択」チェック・ボックスを使用すると、現在表示されているすべてのエントリー (データベース内のすべてのエントリーではない) を含めることができます。
4. 右クリックし、「ポリシー・ルールの追加」を選択します。「ルールの作成」ダイアログが開き、選択したエントリーの値が表示されます。複数のエントリーを選択した場合、それらのエントリーの値を含むグループが作成されます。既存のポリシーに追加するルールを作成することや、新規ルールを含む新規ポリシーを作成することができます。
   注: ルールの範囲が広すぎる (モニターするファイルが多すぎる) 場合は、システムが過負荷になり、処理時間と応答時間が長くなります。
   注: 1 つの FAM ルールに複数のパターンを含めることができます。ディレクトリーとその内容の両方を保護するには、「/FAMtest/*」と「/FAMtest」という 2 つのパターンを持つルールを定義します。
   注: FAM ポリシーを使用する場合、モニター対象ファイル・パスを定義するグループを設定する際に大/小文字の区別について考慮する必要があります。そうしないと、グループを正常に作成できません。 回避策は、異なる 2 つの FAM ポリシー・ルールを作成することです。 分類 - グループのメンバーとして定義する文字列が、大/小文字が区別されない場合でも異なるときに、グループは正常に作成できます。例えば、1. C:¥ABC 2. C:¥abcdef です。グループのメンバーとして定義する

文字列が、大/小文字が区別されないと同じである場合にはグループを作成できません。例えば 1. C:¥ABC 2. C:¥abc の場合などです。そのため、すべて大文字またはすべて小文字でメンバーを入力する必要はありません。 グループ・ビルダーは大/小文字を区別しません。 ただし、大/小文字を区別する UNIX では、パス /IBM/Guardium はパス /ibm/guardium と異なります。 これら両方のパスをモニターしたい場合、現在のグループ・ビルダーには制限があり、同じパスとして見なしません。

5. データ・ソース、アクション、および条件を選択します。変更したい値を上書きします。「編集」をクリックして各フィールドを変更します。
6. 新規ポリシーを作成し、それをインストールするには、「作成およびインストール」をクリックします。ポリシーを作成し、それをインストールしない場合は、「OK」をクリックします。

親トピック: ファイル・アクティビティーのポリシーおよびルール

FAM MS Office イベントの統合の構成

FAM モニターの Office イベント統合機能を使用して、MS Word、Excel、および PowerPoint の無関係のファイル・アクティビティーをフィルターで除外します。

FAM は、MS Office 製品の MS Word、Excel、および PowerPoint をモニターする際、システムで実際に何が起こったのかを判別しづらくなる、無関係の紛らわしい多数のファイル・イベントを生成します。Office イベント統合機能を使用すると、無関係のファイル・アクティビティーをフィルターで除外して、有用なイベントの簡潔なストリームのみがコレクターに提示されるようにすることができます。フィルターにより、無関係のイベントが高い割合でデータ・ストリームから除去されますが、無関係のファイル・イベントが報告されることもあります。例えば、Windows および Office は、ファイルを実際にメモリーにロードすることなく、ファイル属性を読み取るためにファイルを複数回開きます。Office の場合、これは、ユーザーがファイルを開くときに起こります。Office は、最初にファイルを開いてから閉じて属性を読み取り (READ イベントが生成されます)、その後でファイルを実際のメモリーに読み込みます (もう 1 つの READ イベントが生成されます)。ファイルを開いて属性を読み取る場合と、Office が実際のファイルを開いてメモリーに読み込む場合を区別するのは不可能です。

FAM モニターの Office フィルター・ソフトウェアは、一時ファイルに対して実行されたすべてのアクティビティー、Office ジャーナル・ファイルに対して実行されたすべてのアクティビティー、およびエンド・ユーザーが実際に実行した処置を表さないその他のイベントの大部分をフィルターで除外します。また、ファイル・イベントの細分性を高めることで、システムで何が発生したかについてのあいまいさを排除します。例えば、FILEOP イベントの代わりに、FILEOP を構成する実際の基礎的なイベント (RENAME FILE、SET FILE PERMISSIONS、および SET FILE PROPERTIES) を報告します。また、フォルダーで発生するアクティビティーに対しては、別個のイベントがあります。これには、CREATE FOLDER、OPEN FOLDER、CLOSE FOLDER、RENAME FOLDER、READ FOLDER、WRITE FOLDER、EXECUTE FOLDER、DELETE FOLDER、SET FOLDER PERMISSION、および SET FOLDER PROPERTIES が含まれます。これらは、ファイルについて生成されるのと同じイベントです。唯一の相違点は、フォルダーに適用されることです。

OPEN FILE、CLOSE FILE、OPEN FOLDER、および CLOSE FOLDER の各イベントは、FAM モニターによってローカル側で処理されますが、コレクターには配信**されません**。コレクターに配信されない理由は、Windows エクスプローラー・プログラムがバックグラウンドで絶えずファイルを開いたり閉じたりしていて、これらのイベントが有用であることは稀であるためです。それらのすべてをコレクターに報告すると、無用な情報をエンド・ユーザーに提供することになり、無益なトラフィックによってコレクターとネットワークのフラッディングが発生します。

Office イベント統合は、guard_tap.ini ファイル内で下記のパラメーターで構成されます。 これらのパラメーターは、FAM モニターにのみ適用され、S-TAP では無視されます。

guard_tap.ini ファイルを変更するには、次のようにします。

1. root アカウントを使用して、データベース・サーバー・システムにログオンします。
2. S-TAP を停止します。
3. 構成ファイル (guard_tap.ini) のバックアップ・コピーを作成します。デフォルトのファイルの場所は、¥Program Files¥IBM¥Windows S-TAP¥Bin¥ です。
4. 構成ファイルをテキスト・エディターで開きます。
5. 必要に応じてファイルを編集します。パラメーターは、ファイルの [Tap] セクション内になければなりません。
6. ファイルを保存します。
7. S-TAP を再始動します。.

パラメーター名	指定可能な値	デフォルト値	記述
ENABLE_OFFICE_FILTERS	0, 1	1	FAM モニター・ソフトウェアの Office フィルター・コンポーネントを有効または無効にします。無効にすると、一時ファイルやジャーナル・ファイルを含め、すべてのファイルのイベントがコレクターに報告されます。有効にすると、エンド・ユーザーが実行した実際の操作に関連する有用なファイル・イベントのみがコレクターに送信されます。
WORD_EXTENSIONS	ファイル拡張子のリスト	.docx、.doc、.docm、.dotm、.dotx、.dot、.odt	ファイルを Microsoft Office Word ソース・ファイルとして識別するファイル拡張子。FAM モニター・ソフトウェアは、それらのファイルに対して生成されたイベント・ストリームで Office フィルター・コンポーネントを使用します。
EXCEL_EXTENSIONS	ファイル拡張子のリスト	.xlsx、.xls、.xlsm、.xlsb、.xltx、.xltm、.xlt、.ods	ファイルを Microsoft Office Excel ソース・ファイルとして識別するファイル拡張子。FAM モニター・ソフトウェアは、それらのファイルに対して生成されたイベント・ストリームで Office フィルター・コンポーネントを使用します。
POWERPOINT_EXTENSIONS	ファイル拡張子のリスト	.pptx、.pptm、.ppt、.potx、.potm、.pot、.odp	ファイルを Microsoft Office PowerPoint ソース・ファイルとして識別するファイル拡張子。FAM モニター・ソフトウェアは、それらのファイルに対して生成されたイベント・ストリームで Office フィルター・コンポーネントを使用します。
FSM_LOG_EVENTS	0, 1	0	FAM モニターのテキスト・ログ・ファイルへのファイル・イベントのロギングを有効または無効にします。有効にすると、アプライアンスに送信された CREATE FILE、READ FILE などのすべてのイベントは、STAP ..¥logs フォルダー内の循環テキスト・ログ・ファイルにも記録されます。

親トピック: 保護

モニターおよび監査

機密データを識別し、それを保護するステップを実行した後、このデータにアクセスするアクティビティーをモニターする必要があります。多くの場合、モニターにより生成されるデータを使用することで、監査要件 (法的または内部) を順守できます。

- 監査プロセスの作成
  資産ディスカバリー、脆弱性評価と強化策、 データベース・アクティビティー・モニターおよび監査のレポート作成、レポートの配布、 主要な利害関係者によるサインオフ、およびエスカレーションなどの データベース・アクティビティー・モニター・タスクを、1 つのスポットに統合することにより、 コンプライアンス・ワークフロー・プロセスを合理化します。
- 監査およびレポート
  Guardium は、収集したデータをドメイン・セットに編成します。各ドメインには、特定の関心領域 (データ・アクセス権、例外、ポリシー違反など) に関連する異なるタイプの情報が格納されます。
- 外部データ相関
  このトピックでは、エンタープライズ情報をインポートして既存の Guardium 内部データと共に使用するためのカスタム表とカスタム・ドメインの作成および管理について説明します。
- プライバシー・セット
  プライバシー・セットとは、特別なモニターを行うために使用できる要素の集合です。
- カスタム・アラート
  アラート・メッセージを配布する方法として、E メール、SNMP、syslog、またはユーザー作成の Java™ クラスが可能です。この最後のオプションを「カスタム・アラート」といいます。
- 未解析ログ処理
  未解析ログ・オプションは、Guardium® アプライアンスが情報を即時に解析することなくログに記録できるようにする処理です。
- データベース・ライセンス・レポート
  ライセンス・レビューは、ユーザーがそれぞれの業務を行うために必要な特権のみを持っていることを検証および確認するプロセスです。
- ユーザー識別
  Guardium には、データベース・トラフィックから実際のデータベース・ユーザーが識別できない場合に、アプリケーション・ユーザーを識別する方法がいくつか用意されています。
- 値変更監査
  値変更監査フィーチャーは、データベース表内の値の変更をトラッキングします。
- 監査データベースの作成
  監査データベースを作成して値変更モニター・アクティビティーを実行します。
- モニター対象表アクセス
  この機能は、Optim™ Designer データ・ライフサイクル製品との相互作用を可能にするために、「最後の評価」フィールドを関連する表に追加します。
- NAS および SharePoint のファイル・アクティビティー・モニター
  Guardium ファイル・アクティビティー・モニター (FAM) は、Windows 環境の NAS デバイスおよび SharePoint サーバー上のファイルとディレクトリーのアクティビティーをモニターします。
- コンプライアンス・モニターのクイック・スタート
  モニター・エージェント (S-TAP) をデプロイした後、コンプライアンス・モニター・ツールを使用して、特定のセキュリティー基準および規制のモニターを設定します。
- PCI/DSS アクセラレーターを使用して PCI コンプライアンスを実装する方法
  PCI/DSS 要件を満たすために、IBM Security Guardium の PCI/DSS アクセラレーターを構成し、一連のポリシーとレポートを作成します。
- ワークフロー・ビルダー
  ワークフロー・ビルダーは、監査プロセスで使用する、カスタマイズされたワークフロー (ステップ、移行、およびアクション) を定義するために使用します。
- 脅威検出分析
  Guardium には、監査済みデータをスキャンおよび分析して、さまざまなタイプのデータベース攻撃を示す可能性のある徴候を検出するための特殊な脅威検出分析が組み込まれています。
- 調査ダッシュボード
  調査ダッシュボードは、Guardium 環境に存在する可能性がある問題を特定して評価するための強力なツールを提供します。これはローカルまたはシステム全体のフィルタリングされていないデータを使用し、Guardium 環境全体で、その環境内のすべての Guardium コレクターを対象としてデータを照会するための多くのフィルタリング・オプションを提供します。
- Outliers Detection
  2 つの簡単なステップで Outliers Detection を有効にして、Outliers Detection の監査を開始できます。これにより、Guardium が異常なサーバーの動作とユーザーの動作を識別し、考えられる攻撃を早期に検出するための処理を行えるようになります。
- データ保護ダッシュボード
  Guardium のデータ保護ダッシュボードは、上級セキュリティー担当者のためにリスクおよびコンプライアンスのデータの要約ビューを提供します。

監査プロセスの作成

資産ディスカバリー、脆弱性評価と強化策、 データベース・アクティビティー・モニターおよび監査のレポート作成、レポートの配布、 主要な利害関係者によるサインオフ、およびエスカレーションなどの データベース・アクティビティー・モニター・タスクを、1 つのスポットに統合することにより、 コンプライアンス・ワークフロー・プロセスを合理化します。

以下の監査アクティビティーを自動化し、コンプライアンス・ワークフローに統合します。

- 複数の監査タスク (レポート、脆弱性評価など) を 1 つのプロセスにグループ化する機能。
- これらのプロセスの定期的な実行をスケジュールする。
- これらのタスクをバックグラウンドで実行する。
- タスクの結果を Comma-Separated Value (CSV) ファイルまたは ArcSight Common Event Format (CEF) ファイルに 書き込むか、Syslog を使用して他のシステムに転送する。あるいは、その両方を行う。
- コメントおよびメモを追加する。
- プロセスをその発信者に割り当て、表示可能にする (結果が準備できると、発信者の To-Do リストに新規項目が追加されます)。
- プロセスを他のユーザー、ユーザー・グループ、またはロールに割り当てる。
- これらの割当先が結果にサインオンするための要件を作成する。
- 結果のエスカレーション (オリジナルの監査証跡の外部にいるユーザーへの割り当て) を許可する。

データベース・セキュリティーの管理を、定期的に実行される、時間を要する手動のアクティビティーから、 継続的な、企業のプライバシーおよびガバナンス要件 (PCI-DSS、SOX、データ・プライバシーおよび HIPAA など) を サポートする自動化プロセスに変換します。

監査結果を、追加のフォレンジック分析のために外部リポジトリー (Syslog、CSV/CEF ファイル、外部フィード) にエクスポートします。

「監査プロセス・ログ」レポートには、すべてのタスクに関する 詳細なアクティビティー・ログが、開始時刻および終了時刻も含めて示されます。 このレポートは、admin ユーザーが「Guardium® モニター」タブを通じて入手可能です。監査タスクには、開始および終了時刻が示されますが、セキュリティー・アセスメントおよび分類 (キューに入れられます) の開始および終了は同じになります。

各ワークフロー・プロセスの結果 (レビュー、サインオフ証跡、およびコメントなど) はアーカイブ可能であり、 後ほど調査センターを通じてリストアし、レビューすることができます。

コンプライアンス・ワークフロー自動化プロセスは、以下の疑問に答えます。

- 必要とするレポート、アセスメント、監査証跡、または分類のタイプは?
- この情報の受信者およびサインオフの処理方法は?
- 配布のスケジュールは?

コンプライアンス・ワークフロー自動化プロセスには、さらに次のようなエレメントも含まれます。

- プロセス定義
- 配布計画。以下を行います。
  - 受信者 (個々のユーザー、ユーザー・グループ、またはロールのいずれであっても可) を定義します。 (『プロセス受信者』を参照してください。)
  - 各受信者に、レビュー/署名の責務を定義します。
  - 「継続」フラグを設定することにより、配布シーケンスを定義します。
- タスク・セット (『プロセス・タスクのタイプ』を参照してください。)
- スケジュール - 監査プロセスはすぐに実行することも、スケジュールを定義して定期的に実行することもできます。

プロセス・タスクのタイプ

ワークフロー・プロセスには、以下の監査タスクを任意の数、組み込むことができます。

- レポート (カスタムまたは事前定義)。 Guardium は、100 を超える調整固有のレポートのほか、数百もの事前定義レポートを提供します。
- セキュリティー・アセスメント・レポート。 セキュリティー・データベース・アセスメントは、データベース・インフラストラクチャーで脆弱性をスキャンし、アセスメントリアルタイムの測定および履歴測定の両方による、データベースおよびデータ・セキュリティーの正常性のアセスメントを提供します。 このアセスメントでは、カスタム・テストを取り込むほか、共通データベース・セキュリティー・ベスト・プラクティス (STIG および CIG1 など) を使用してグループ化された、既知の問題および脆弱性に基づく 事前定義脆弱性テストと、現行の環境を対比します。アプリケーションは、 (ベスト・プラクティスに基づく) 重み付け測定基準を取り入れた セキュリティー・ヘルス・レポート・カードを生成し、 データベース・セキュリティーを強化するためのアクション計画を推奨します。
- エンティティー監査証跡。 特定のエンティティー (例えば、クライアント IP アドレスまたはアドレス・グループなど) に 関連するアクティビティーの詳細なレポートが作成されます。
- プライバシー・セット。 オブジェクト/フィールド・ペア (例えば、社会保障番号と生年月日など) のグループに対するアクセスについて詳述したレポートが、 指定された期間中に作成されます。
- 分類プロセス。 既存のデータベースのメタデータおよびデータがスキャンされ、機密である可能性のある情報 (社会保障番号や クレジット・カード番号など) についてレポートを作成します。
- 外部フィード。 データを外部の特殊なアプリケーションにエクスポートして、さらに詳細なフォレンジック分析を行うことができます。
  注: 「オプション外部データ・フィード」は、プロダクト・キーによって使用可能になるオプション・コンポーネントです。このフィーチャーは、使用可能になっていない場合には「監査タスク」選択項目には表示されず、 「フィード・タイプ」リストは空になります。

ワークフロー・プロセス、一元管理および統合

中央マネージャーにおいて、レポートはリモート・データ・ソース (管理対象ユニット) から データを参照可能です。これらのレポートを使用する監査プロセスは、中央マネージャーからのみアクセス可能であり、 管理対象ユニットからは不可視になります。

アグリゲーター・サーバーのワークフロー自動化 (監査処理) に、各アグリゲーター・タスクの一時データベースを作成し、 そのタスクに関連する日のみを指定するための機能が組み込まれました。

注: 統合サーバーの一時データベースは、必要に応じて Guardium サポート・サービスによる実行後分析を行うため、 (CLI コマンド drop_ad_hoc_audit_db の値に応じて) 最大 14 日 までシステムに保持することができます。

監査プロセスでレポートを定義するとき、(FROM-TO フィールドで定義される) レポートの日数は特定のしきい値を超えることができません (デフォルトでは 1 カ月)。このしきい値を超えると、アグリゲーターで監査タスクの実行を試行中にランタイム・エラーが発生します。

(CLI で設定された) max_audit_reporting 値を 超える FROM-TO 範囲を指定した監査タスクを作成することが許容されています。これは、アグリゲーターに定義された監査プロセスが、 (このアグリゲーターがマネージャーである場合に) 管理対象コレクターで実行される場合があるからです。 コレクター・ユニットで実行される監査タスクには、max_audit_reporting 制限がありません。

したがって、許容範囲を超えるタスクを保存することは有効ですが、アグリゲーターでタスクを実行する際に、 実行時例外が発生する可能性があります。

監査レポートのしきい値は、CLI コマンド show max_audit_reporting または store max_audit_reporting を使用して構成できます。 無効な FROM-TO 範囲を指定してレポートを作成しても、警告メッセージは出されません。 代わりに、「監査プロセス」設定メニュー画面の「タスク・パラメーター」パネルに、固定メッセージが表示されます (「ツール」/「監査プロセス・ビルダー」。「監査タスク」を開いて、「タスク・パラメーター」を表示)。固定メッセージを以下に示します。

アグリゲーターに関しては、許容される時刻範囲 (CLI: max_audit_reporting) を超えないレポートのみが実行されます。

注: パッチ・インストールの実行中は、すべての監査プロセスが停止します。

監査プロセスの停止

監査プロセスの停止は、監査タスクが実行されていない場合、または実行中の場合にのみ実行可能です。 監査プロセスを停止すると、以後、まだ開始されていないタスクは実行されません。 監査プロセスを停止しても、部分的な結果は送信されません。 監査プロセスが停止した結果として、停止したことを示すエラー・メッセージが出されます。 ただし、タスクが完了している場合、監査プロセスを停止しても、結果の送信は停止されません。

「順守」 > 「ツールとビュー」 > 「監査プロセス・ログ」レポートから起動した GuardAPI (カーソルを任意の行に置いてダブルクリックし、ドリルダウン) を使用して、監査プロセスを停止します。

任意のユーザーの場合、監査プロセスを停止すると、そのユーザーに属する行 (タスクのみ、全詳細は含まれない) が表示されます。 admin ユーザーは全詳細を確認可能であり、すべてのユーザーの監査プロセスを停止できます。 その他のユーザーは、自身の監査プロセスのみ停止できます。

注:

リモート・ソースを使用している照会は停止できません。リモート・ソースを使用している オンライン・レポートは停止できません。

監査プロセスの停止は、プライバシー・セット監査タスクや、外部フィード監査タスクには適用されません。 プライバシー・セット・タスクや外部フィード・タスクが開始した場合、 プロセスが停止されてもこれらのタスクは完了します。

結果の配布

監査プロセスの受信者は、E メール、または各自の To-Do リスト (あるいはその両方) によって、処理中の監査プロセスの結果について通知を受けます。任意の受信者をプロセスの署名者に指定できます。 その場合、配布リストのその受信者のポイントで、受信者によって電子署名が付けられるかリリースされるまで、 結果をオプションで保留にすることができます。 受信者は、個々のユーザー、ユーザー・グループ、またはロールのいずれであってもかまいません。

監査プロセス・サマリー

「監査プロセス・ファインダー」画面内に「監査プロセス状況サマリー (Audit Process Status Summary)」があります。このセクションには、スケジュールされた監査プロセス、結果、未処理の受信者およびエラーに関する情報が含まれています。 このサマリーは、複数の監査プロセス・レポートからのデータを統合したものです。

監査プロセスの結果を削除するためのボタンもあります。「監査プロセス・ファインダー」画面を参照してください。 「今すぐ 1 回実行」ボタンの横にある「結果」ボタンを見つけます (選択肢は「表示」または「削除」)。

監査プロセスの結果が削除されますが、レポートの削除者がトラッキングされ、ログに記録されます。 audit-delete ロールは、監査プロセスの結果が削除された場合のトラッキングまたはロギングに使用されます。 audit-delete ロールを持っているユーザーは、レポートを削除することができます。管理ユーザーも、レポートを削除することができます。トラッキングは、ユーザー・アクティビティー監査証跡レポートを使用して実行されます。

注: リモート・ソースの監査プロセスの結果は、100,000 件までに制限されています。 この制限を超える場合は、CLI コマンドの store save_result_fetch_size (show save_result_fetch_size) を使用してください。

プロセス受信者

ワークフロー自動化プロセスには、任意の数の受信者を定義でき、各受信者が結果を受け取る順番を制御することができます。 また、受信者は、「エスカレート」機能を使用して他の受信者に通知することができます。 受信者を定義せずに監査プロセスを実行することも可能です。 例えば、syslog に書き込みを行い、結果のレビュー (署名) を必要としない、 受信者のいない監査プロセスなど。

誰を受信者とするか?

「プロセス定義」パネルの受信者のドロップダウン・リストには、全 Guardium ユーザー、 ユーザー・グループ、およびロール (グループおよびロールには、そのようにラベルが付けられています) が含まれています。 グループまたはロールを選択すると、そのグループに属する、またはそのロールを持つすべてのユーザーが、結果を受信します。

グループ受信者を選択した場合、 いずれかのワークフロー自動化タスクで照会条件に特殊ランタイム・パラメーター ./LoggedUser が使用されていると、 その照会はグループ内のユーザーごとに個別に実行され、各ユーザーは自身の結果だけを受信します。

例えば、会社に 3 つの DBA があり、各 DBA がそれぞれ異なるサーバー・セット の管理下にあるとします。「カスタム・データのアップロード」機能を使用して、各 DBA の責務分野を (サーバー IP とともに) Guardium システムにアップロードし、それをデータベース・アクティビティー・ドメインに相関し、このカスタム・ドメインでレポートを監査タスクとして使用します。3 つの DBA を含んだユーザー・グループが受信者に指定された場合、各 DBA は受信者の サーバーのコレクションに関連するレポートのみを受信します。

グループ受信者を選択し、 サインオフが必要とされる場合、各グループ・メンバーが結果に個別に署名する必要があります (前述の 説明のとおり、グループのメンバーごとに表示される結果セットが異なる可能性があります)。

E メール・アドレスだけを受信者とし、 結果がその E メール・アドレスに送信されるようにすることができます。 E メール・アドレスを入力するユーザーは、データをフィルターに掛けるために 使用されるユーザーを入力する必要があります。このユーザーは、ログイン・ユーザーと同じであるか、 データ階層でログイン・ユーザーの下位にいるユーザーでなければなりません。

ロール受信者を選択した場合、 結果に署名する必要があるのは、そのロールの 1 人のユーザーだけであり、 同じロールの他のユーザーは、結果に署名が付けられた際に通知を受けます。

注:

ワークフロー・イベントが作成された際には、 そのイベントが使用するすべての状況にロールを割り当てることができます (つまり、 その状況にある場合、イベントはこのロールからのみ参照可能になります)。 イベントを監査プロセスに割り当てる際には、このイベントの状況に割り当てられた すべてのロールに、この監査プロセスの受信者がいることが重要です。 そうでないと、監査結果行が、この行を参照したり、状況を変更したりできる受信者が いない状況に置かれる可能性があります。

この場合、admin ユーザー (ロールに関係なくすべてのイベントを参照可能) が、この行を確認し、その状況を変更できます。ただし、データ・レベル・セキュリティーがオンになっていると、admin ユーザーがこの行を参照できない可能性があります。admin ユーザーは、 (「グローバル・プロファイル」から) データ・レベル・セキュリティーをオフにするか、 dataset_exempt ロールを保持することが必要になります。 監査プロセスは、その監査プロセスに関連するイベントに対処する必要があるすべてのロールが受信者となるように構成することが重要です。

E メール通知

オプションで、受信者は、E メールを通じて新規プロセスの結果の通知を受けることができます。結果を E メールで配布するためのオプションとして、次の 2 つがあります。

- リンクのみ - E メール通知には、Guardium システムに格納された結果へのハイパーテキスト・リンクが含まれます。リンクを機能させるには、Guardium システムに対するアクセス権を保持しているシステムからメールにアクセスする必要があります。E メール・リンクについて詳しくは、以下のセクションを参照してくださ

い。
- 全結果 - 結果を含んだ PDF ファイルまたは生成された CSV ファイルが E メールに添付されます。 ただし、オリジナルの配布リストに含まれない受信者を指定する「エスカレーション」については例外であり、 この場合は PDF ファイルや CSV ファイルは添付されません。「全結果」オプションを選択する際には、 PDF ファイルや CSV ファイルに機密データやプライベート・データが含まれる可能性があるため、 注意が必要です。監査プロセスを実行しており、「全結果を CSV で」にチェック・マークが付けられた受信者がいる場合、 タイプが「アセスメント」、「分類」、または「外部フィード」のタスクでは、CSV ファイルは生成されません。 これらのタイプのタスクでは、エクスポート用の CSV/CEF/PDF ファイルも生成できません。 タイプが「レポート」、「プライバシー・セット」、または「エンティティー監査証跡」のタスクで、 「全結果を CSV で」にチェック・マークが付けられている受信者がいる場合にのみ、 CSV ファイルが生成されます。
  注: 監査結果を表示する際に、生成済みの PDF が既存の場合は、「PDF の再作成」ボタンが表示され、そのボタンを使用して PDF を再作成し、ダウンロードすることができます。

プロセス結果へのハイパーテキスト・リンク

E メール・メッセージでは、Guardium システム上のプロセス結果へのリンクが機能しない条件があります。例:

- 通常 Guardium システムにアクセスできないロケーションから E メールにアクセスしている場合、リンクは機能しません。例えば、オフィス外にいるときには、インターネット経由で自身の E メールにアクセスすることはできますが、システムがインストールされている社内のプライベート・ネットワークや LAN にはアクセスできません。
- レポート結果が保持されるよりも長い期間、自身の E メールにアクセスしていない場合、それらの結果はリンクをクリックしても使用可能になりません。例えば、結果が 7 日間保持されるのに対し、2 週間の休暇を取っていたユーザーの E メールには、7 日よりも前の結果に対するリンクが含まれている可能性があり、それらのリンクは機能しません。

凍結される受信者リンクについて

プロセスが一度実行されると、既存の受信者リストは凍結されます。 これは、以下のことを意味します。

- リストから受信者を削除できません。
- 既存の受信者をリスト内で上位または下位に移動できません。
- リストの末尾にはいつでも受信者を追加でき、追加した時点で新規受信者を位置変更することは可能です。
- リスト上の受信者の Guardium ユーザー・アカウントが削除された場合、 その受信者は admin ユーザー・アカウント (削除不可) で 置き換えられます。したがって、削除された受信者に送信される予定であったすべての E メール通知を admin ユーザーが受信し、admin ユーザーはその受信者に対してリリースされたすべての結果に対処する必要があります。
- 既存のプロセスに対して完全に異なる受信者セットを作成する必要がある場合、 オリジナルのプロセスを非アクティブ化し、そのコピーを作成して、 コピー・バージョンの受信者リストに変更を加えた後、保存します。

結果を受信者にリリースする方法

結果は、受信者リストにリストされている Guardium ユーザーにリリースされます。このとき、以下のように「継続」チェック・ボックスに準拠します。

- 「継続」チェック・ボックスにマークが付けられている場合、配布は中断なしでリストの次の受信者に継続されます。
- 「継続」チェック・ボックスがクリアされている場合、次の受信者への配布は、現行の受信者が必要なアクション (レビューまたは署名) を実行するまで保留になります。

例えば、以下のようにワークフロー・プロセスを定義するとします。

- DBA - すべての DBA は、同時に結果を受け取ります。 各 DBA は、それぞれに関連付けられたサーバー IP に基づいて異なる結果セットを受信します。
- すべての DBA が署名付けされたときにのみ、DBA マネージャーが結果を表示できます。
- DBA マネージャーがレポートをリリースしたときにのみ、監査員が結果を表示できます。
- すべての監査員は、同時にレポートを受け取りますが、各結果に署名する必要があるのは、そのうちの 1 人 (任意) の監査員だけです。 その他の監査員は、結果に署名が付けられたときに更新されます。
- 監査員は、結果を監査マネージャーにエスカレートすることができます。

このフローを定義する手順は、次のとおりです。

- DBA グループが最初の受信者として指定されます。
- DBA マネージャーは、リストでその次に位置します。
- 監査員ロール (グループではない) は、リストでその次に位置します。 いずれかの監査員が署名し、他の監査員は通知を受けることができます。 また、監査員は結果セットを監査マネージャーにエスカレートすることができます。
  注: 現行の受信者によって「継続」ボタンにマークが付けられている場合にのみ、次の受信者に結果が配布されます。 これは、レビュー/署名機能とは完全に異なるもので、レビュー/署名機能には全く依存しません。
  注: CSV または CEF ファイルにエクスポートされたプロセス結果は、Guardium アーカイブおよびエクスポート・メカニズムによって、別のネットワーク・ロケーションに送信されます。これらの結果は、受信者リストや署名アクションの影響を受けません。 これらは、Guardium CSV/CEF エクスポート・スケジュール (定義されている場合) の対象となり、 最終的な格納先となるディレクトリーに認可されているアクセス権限の影響を受けます。

監査タスク出力を CSV、CEF、または PDF ファイルにエクスポート

他のアプリケーションが使用可能な情報を含んだレポート、または 大量のデータを含んだレポートを、別のファイル・フォーマットでエクスポートすることができます。レポート、エンティティー監査証跡、およびプライバシー・セット・タスクの出力は CSV (Comma Separated Value) ファイルに、 データベース・アクティビティー・レポートの出力は ArcSight Common Event Format (CEF) ファイルにエクスポートすることができます。

さらに、CEF および CSV ファイルの出力を syslog に書き込むことができます。 リモート syslog 機能を使用する場合、出力 CEF/CSV ファイルがリモート syslog のロケーションに 直ちに送信されます。remote syslog 関数により、 メッセージを各ファシリティーと重大度の組み合わせから、特定のリモート・システムに送信することができます。 詳しくは、remotelog (syslog) CLI コマンドの説明を参照してください。

CSV ファイルや CEF ファイルの各レコードは、レポートの 1 行を表しています。

標準タスク出力に置き換わるのではなく、追加する形で、エクスポート・ファイルが作成されます。 これらのファイルは、以下を行う必要がある場合に便利です。

- インフラストラクチャー (Qradar、ArcSight、Network Intelligence、LogLogic、TSIEM など) 内の 既存の SIEM (Security Incident and Event Manager) との統合。

- 非常に大規模なコンプライアンス・タスク結果セットのレビューおよび分析。 (Web での表示用のタスク結果セットの出力は、5,000 行までに制限されています。一方、CSV または CEF エクスポート・ファイルに書き込まれる行数に制限はありません。)

CSV および CEF エクスポート・ファイルは、次のフォーマットで名前が付けられ、Guardium システムに格納されます。

```
process_task_YYYY_MMM_DD-HHMMSS.<csv | cef>
```

ここで、process は監査プロセス定義で定義したラベル、 task はプロセス内の各タスクに定義可能な第 2 レベルのラベル、 YYYY_MMM_DD-HHMMSS はタスク実行時に作成される日時スタンプです。

Guardium システム上の CSV または CEF エクスポート・ファイルには、直接アクセスすることはできません。Guardium 管理者は、「CSV/CEF エクスポート」機能を使用して、これらのファイルを Guardium システムからネットワーク上の他のロケーションに移動する必要があります。これらのファイルにアクセスするには、Guardium 管理者に確認の上、 これらのファイルがコピーされたロケーションを判別してください。

エクスポート・ファイルが Guardium システムの外部に送信されることには、次の 2 つの重要な意味があります。

- これらのファイルのリリースは、監査プロセスに定義された結果配布計画に関連付けられていません。 これらのファイルは、Guardium 管理者が定義したスケジュールでエクスポートされます。
- 一度「CSV/CEF エクスポート」機能を実行すると、「CSV/CEF エクスポート」操作で 定義された宛先ディレクトリーにアクセス可能なユーザーであれば (Guardium ユーザーであるかどうかを問わず)、 すべてのエクスポート・ファイルを使用できるようになります。このため、Guardium 管理者は、 Guardium CSV/CEF エクスポート宛先ディレクトリーから、適切なアクセス権限を備えた ディレクトリーに、エクスポート・ファイル・セットをコピーする追加の ジョブを (Guardium システムの外側で) スケジュールする場合があります。

CSV/CEF エクスポート・アクティビティーは、統合/アーカイブ・アクティビティー・レポートで使用可能です。

注: 監視データ・レベルでのセキュリティーが有効になっている場合、監査プロセスの出力 (ファイルを含む) がフィルタリングされ、ユーザーは、自身に割り当てられたデータベースの情報だけを見ることができます。E メール受信者に添付ファイルとして送信されるファイル がフィルターに掛けられます。ただし、マシンにローカルにダウンロードされ、「結果エクスポート」機能を使用して他の場所に移動されたファイルは、データ・レベル・セキュリティー・フィルター操作の対象になりません。CSV/CEF エクスポートについて詳しくは、 このトピックで後述する『CSV/CEF エクスポート』を参照してください。

次の表は、監査プロセス・ファイルを CSV/CEF/PDF にエクスポートした場合の動作を要約しています。

表 1. 監査タスク出力を CSV、CEF、または PDF ファイルにエクスポート

機能	レベル	CSV	CEF	PDF
E メールに添付	受信者	「全詳細」ラジオ・ボタン -->「PDF」チェック・ボックス	N/A	「全詳細」ラジオ・ボタン -->「PDF」チェック・ボックス ラジオ・ボタンは、受信者 PDF のみを対象としています。
ファイルのエクスポート	タスク	「CSV ファイルへのエクスポート」チェック・ボックス	「CSV ファイルへのエクスポート」チェック・ボックス	「CSV ファイルへのエクスポート」チェック・ボックス
空の場合に報告、および Empty = yes の場合に承認	受信者	エクスポートに影響なし (空のファイルはエクスポートされます) 添付ファイル、E メールを添付しません	エクスポートに影響なし (空のファイルはエクスポートされます) 添付ファイル、E メールを添付しません	エクスポートに影響なし (空のファイルはエクスポートされます) 添付ファイル、E メールを添付しません
zip 添付ファイル	監査プロセス	ファイルが生成されない場合、何も zip しません すべての CSV を 1 つの zip ファイルにマージ	N/A	ファイルが生成されない場合、何も zip しません PDF は zip されません
圧縮 (エクスポート)	タスク	CSV ファイルごとに別ファイルで圧縮	CSV ファイルごとに別ファイルで圧縮	PDF は圧縮されません

監査タスク出力における「zip して E メール」および「圧縮」の機能

「zip して E メール」は、「監査タスク・エクスポート」の最上位レベルのコントロールです。 「zip して E メール」することで、CSV または CEF ファイル・セットが生成されます。 PDF は zip されることも、圧縮されることもありません。

圧縮は個々のファイルに対して機能します。

注: CSV 添付ファイルでは、「zip して E メール」がクリアされている場合で あっても、「圧縮」を適用できます。「圧縮」は、タスクごとに行えます。 そのため、同じ E メールで、1 つの監査タスクは .csv ファイルを、別の監査タスクは .csv.gz ファイルを送信することができます。

「zip して E メール」および「圧縮」の相互作用は、以下のようになります。

- 「zip して E メール」がチェックされている場合 (「圧縮」がチェックされているかどうかに関係なく) CSV ファイルの 1 つの zip ファイルが添付ファイルになります。
- 「zip して E メール」がチェックされておらず、「圧縮」がチェックされている場合、 csv.gz ファイル・セットが添付ファイルになります。
- 「zip して E メール」と「圧縮」がいずれもチェックされていない場合、 csv ファイル・セットが添付ファイルになります。
- 「圧縮」がチェックされている場合、「すべてダウンロード」は csv.gz になります。
- 「圧縮」がクリアされている場合、「すべてダウンロード」は csv になります。
- 「圧縮」がチェックされている場合も、クリアされている場合も、「表示のダウンロード」は csv のままになります。
- 「圧縮」がチェックされている場合、CSV/CEF ファイルのエクスポートは gzip されます。
- 「圧縮」がクリアされている場合、CSV/CEF ファイルのエクスポートは gzip されません。

SCAP または AXIS へのエクスポート

「監査プロセス定義」の「タスクの新規追加」セクションで、「セキュリティー・アセスメント」の「タスク・タイプ」を 選択すると、いくつかの選択項目 (「AXIS xml のエクスポート」および 「SCAP xml のエクスポート」) が表示されます。監査プロセスの結果を保存するには、および「結果エクスポート」(「管理」 > 「データ管理」 > 「結果エクスポート (ファイル)」) でセットアップした宛先に XML ファイルを転送するには、これらの選択肢のいずれかを選択します。他の選択項目 (「レポート」、「差異」、「レポートと差異」) は PDF 形式を構成するためのものです。

SCAP は、Security Content Automation Protocol です。AXIS は、Apache EXtensible Interaction System であり、 QRadar によって使用されます。

レポートの作成または変更

「レポート・ビルダー」を使用すると、レポートを作成またはカスタマイズできます (行への強調表示色の適用など)。「レポート・ビルダー」を開くには、「レポート」 > 「レポート構成ツール」 > 「レポート・ビルダー」にナビゲートします。

監査ワークフロー・プロセスの作成

1. 「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」にナビゲートして「監査プロセス・ビルダー」を開きます。
2. 「新規」ボタンをクリックして「監査プロセス定義」パネルを開きます。「監査プロセス定義」パネルは、「一般」、 「受信者」、および「タスク」という 3 つのセクションに分割されます。
3. 最初に、「タスク」セクションに進みます。 プロセスを保存する前に、少なくとも 1 つの監査タスクを定義する必要があります。選択項目を設定する各タスクを順に実行します。 監査プロセスに含める各監査タスクに該当する手順を実行してください。このセクションで詳細を示すタスクの選択項目は、 以下のとおりです。
   - レポート・タスクの定義
   - セキュリティー・アセスメント・タスクの定義
   - エンティティー監査証跡タスクの定義
   - プライバシー・セット・タスクの定義
   - 分類プロセス・タスクの定義
   - 外部フィード・タスクの定義
4. 「受信者」セクションに進みます。 ドロップダウン・ボックスを開いて、プロセスの受信者を追加します。 『受信者の追加』を参照してください。必要なアクション、To-do リストへの追加、E メール通知、および継続的な配布を決定するには、チェックを付ける必要があります。 この場合も、『受信者の追加』で、これらの選択項目の設定に関する完全な詳細を参照してください。
5. 「一般」セクションに進みます。「記述」ボックスに名前を入力します。 アポストロフィ文字は含めないでください。
6. 「アクティブ」ボックスにチェック・マークを付け、 このプロセスにスケジュールを関連付けます。
7. 保存期間の期限が切れた後にオフラインで結果を保存する場合は、「結果のアーカイブ」ボックスにマークを付けます。 アーカイブされた結果は、後ほど、システムにリストアして再度表示することができます。
8. 「Archive Result purge before Reviewed」ボックスを使用すると、すべてのレビューアーのレビューが完了し、すべてのサインオフが行われ、すべてのワークフロー・アクティビティーが満たされるまで結果を保持せずに特別プロセスの結果が削除されます。 この機能により、ユーザーは、結果がレビューされたかどうかに関係なく、指定した期間 (1 日など) の結果をオプションで削除することができます。
9. 「最低保持期間」の「(n) 日」または「(n) 実行」ボックスに、結果を保存する期間を入力します。 期間は日数 (デフォルトは 0) または実行数 (デフォルトは 5) のどちらかで入力します。 その後、結果がアーカイブされ (「最低保持期間」ボックスにマークが付けられている場合)、システムからパージされます。
   注: 結果は、結果の受信者がいる場合に限り表示されます。受信者を追加し、結果を再実行すると、その実行がドロップダウン・リストに表示されます。
10. 1 つ以上のタスクで CSV または CEF ファイルが作成される場合は、 オプションとして、すべてのファイルに含めるラベルを「CSV/CEF ファイル・ラベル」ボックスに入力できます。 これらのファイルも (Zip 形式に) 圧縮ができます。圧縮するには、「zip してメール」をクリックしてチェック・マークを付けます。
    注: 10240 MB (10.240 GB) を超えるサイズの CSV/CEF ファイルのエクスポートは制限されます。 推奨されるベスト・プラクティスは、「.zip してメール」ボックスにチェック・マークを付けることです。
11. 監査プロセス定義の「E メールの件名」フィールドは、 その監査プロセスの全受信者の E メールに使用されます。 件名には、実行時にその件名を置き換える以下の変数を 1 つ (以上) 含めることができます。
    - %%ProcessName は、監査プロセスの記述に置き換えられます。
    - %%ExecutionStart は、最初のタスクの開始日時に置き換えられます。
    - %%ExecutionEnd は、最後のタスクの終了日時に置き換えられます。

    件名の入力時には、変数 (%% で始まる) の有無と、それらすべてが有効な変数であるかどうかが確認されます。
12. オプションで、セキュリティー・ロールを割り当てます。
13. オプションでコメントを追加します。
14. 該当するボタンをクリックして、監査ワークフロー・プロセスをスケジュールまたは実行します。
15. 「保存」をクリックします。作業を保存する前に、このメニュー画面を離れて他の構成を実行しないでください。 このセクションを離れて、監査タスクに必要な他のものを作成する作業に移ると、 処理中の作業は保存されず、途中まで作成して中断した内容は保持されません。

    例えば、監査プロセス・ビルダーでアセスメント・タスクを定義する場合、 最初に「セキュリティー・アセスメント・ビルダー」に進んでアセスメント・テストを作成し、 次に「データ・ソース定義」に進んでアセスメント対象のデータベースを指定する必要があります。 監査ワークフローの作成時には、作業内容を保存してから他のタスクに進むか、 それら他のタスクを最初に実行してから監査ワークフロー・プロセスを作成してください。

受信者の追加

1. 「受信者」列で、 Guardium 個人ユーザー、グループ、またはロールのドロップダウン・リストから 受信者を選択します。グループまたはロールを選択すると、そのグループの全メンバーまたはそのロールを持つ全ユーザー が結果を受信します。署名が必須の場合、結果に署名する必要があるのは、1 人のメンバーまたはユーザーだけです。
2. 「必要なアクション」列で、次のいずれか 1 つのオプションを選択します。
   - レビュー (デフォルト) - この受信者が、結果に署名する必要がないことを指定します。
   - レビューと署名 - この受信者が、 (結果をオンラインで表示中に「結果に署名」ボタンをクリックして電子的に) 結果に 署名を付ける必要があることを指定します。
3. 「To-Do リスト」列で、「追加」チェック・ボックスにマークを付けるかクリアして、この受信者が処理中の結果についてそれぞれの「監査プロセスの To-Do リスト」に通知を受けるかどうかを指定します。
   注: 外部サーバーへのファイルの送信を、結果の To-Do リストへの追加や E メールの送信をせずに行う場合は、受信者を指定しない監査プロセスを 定義します。さらに、結果を To-Do リストに追加しないようにするために、「受信者の追加」セクションで「To-Do リスト」チェック・ボックスをクリアし、「受信者」セクションに受信者がある場合はすべて削除し、受信者の追加は行わないでください。
4. 「E メール通知」列で、次のいずれか 1 つのオプションを選択します。
   - なし - この受信者に E メールは送信されません。

- リンクのみ - E メール通知には、(Guardium システム上の) 結果に対するハイパーテキスト・リンクが含まれます。
- 結果 - E メールには、結果のコピーが PDF または CSV フォーマットで含まれます。分類タスクやアセスメント・タスクの結果、機密情報が返されることがあるので注意してください。

5. 「継続」列のチェック・ボックスによって、結果の配布が次の受信者に継続される (デフォルト) か、この受信者が適切なアクションを実行するまで停止するどうかが制御されます。「継続」ボックスがクリアされており、 この受信者がグループまたはロールに属している場合、そのグループまたはロールのメンバーである いずれかのユーザーが選択されたアクションを実行すると、結果がリスト上の次の受信者にリリースされます。
   注: 現行の受信者によって「継続」ボタンにマークが付けられている場合にのみ、次の受信者に結果が配布されます。 これは、レビュー/署名機能とは完全に異なるもので、レビュー/署名機能には全く依存しません。
6. 「追加」をクリックしてリストの末尾に受信者を追加し、受信者ごとにこれらのステップを繰り返します。1 人の受信者は必須です。
7. ユーザーではない受信者も許可されます。「E メール」を選択して E メール・アドレスを 入力すると、結果はその E メール・アドレスに送信されます。 ユーザー以外の E メール・アドレスを入力する際には、 データのフィルター操作に使用されるユーザー名の要件があります。 このユーザーは、ログイン・ユーザーと同じであるか、 階層でログイン・ユーザーの下位にいるユーザーでなければなりません。このユーザーは、画面の「受信者」セクションの 新しい列に保存されます。
8. 空の場合は承認 - このチェック・ボックスをチェックすると、タスクのすべてのレポートが空の場合、 次の動作を行います。自動的に結果に署名 (および/または確認済みのマークを付ける)、 自動的に「続行」をクリック (関連付けられている場合)、 E メール通知を送信しない、タスクをそのユーザーの To-Do リストに追加しない、 PDF/CSV/CEF ファイルを作成しない。このチェック・ボックスを使用すると、空の監査結果に自動的に署名が付けられ、その結果は監査結果ログ内で、他の完了した (確認済み/署名済み) 監査結果と同じように表示されます。このアクションは、空のレポートおよび空のセキュリティー・アセスメント結果に適用されます。 セクション『監査タスク出力を CSV、CEF、または PDF ファイルにエクスポート』で、Empty = YES の場合に承認する際の動作を要約した表を参照してください。

CSV または CEF ファイルのエクスポート

レポート、エンティティー監査証跡、およびプライバシー・セット監査タスクの出力は CSV ファイルに、 レポート監査タスクの出力は CEF ファイルにエクスポートすることができます。 「監査タスク」の下の「レポート」、「エンティティー監査証跡」、または「プライバシー・セット」セクションから、 以下手順を実行します。

1. タイトルを選択します。
2. 「CSV/CEF ファイル・ラベル」ボックスに、オプションでファイル・ラベルを入力します。 デフォルトでは、タスクの「記述」がファイル・ラベルになります。このラベルは、生成されるファイル名の 1 コンポーネントになります (他のコンポーネントは、 ワークフロー自動化プロセスで定義されたラベルです)。
3. 「CSV ファイルへのエクスポート」または 「CEF ファイルのエクスポート」にマークを付けます。
   注: CEF ファイル出力は、データ・アクセス・ドメインのレポート (例えば、 アクセス、例外、またはポリシー違反など) にのみ適しています。Guardium セルフ・モニター・ドメイン (統合/アーカイブ、監査プロセス、Guardium ログインなど) のようなその他のドメインは、 CEF 拡張子にマップされません。
4. 「CEF のエクスポート」ファイルを選択した場合、オプションで「CEF を Syslog に書き込む」ボックスにマークを付けて、CEF レコードを syslog に書き込みます。 リモート syslog ファシリティーが有効になっている場合、CEF ファイル・レコードはリモート syslog に書き込まれます。
5. 「圧縮」ボックスがチェックされている場合、 CSV/CEF エクスポート・ファイルは圧縮されます。
6. 「PDF ファイルのエクスポート」ボックスがチェックされている場合、 この監査タスクの (CSV エクスポート・ファイルと同様の名前が付けられた) PDF ファイルが作成され、 CSV/CEF ファイルと一緒にエクスポートされます。
   注: PDF エクスポート・ファイルは、前のステップで「圧縮」ボックスがチェックされている場合であっても、 圧縮されません。

レポート・タスクの定義

コンプライアンス・ワークフロー自動化プロセスの定義をまだ開始していない場合は、この手順を実行する前に、ワークフロー・プロセスを作成してください。使用するレポートがまだ定義されていない場合は、最初に定義してください。

1. 「新規タスクの追加」ペインが開いていない場合、「監査タスクの追加」をクリックします。
2. 「レポート」ラジオ・ボタンをクリックします。
3. 「CSV/CEF ファイル・ラベル」、「CSV/CEF のエクスポート」、「PDF のエクスポート」、「Syslog に書き込む」、「圧縮」には、いくつかの選択項目があります。 『CSV または CEF ファイルのエクスポート』を参照してください。
4. 「PDF オプション」の選択項目には、「レポート」(現在の結果)、「差異」(1 つ前のレポートと新しいレポートの間の差異)、「レポートと差異」(その両方) があります。
   注: 「PDF オプション」の選択内容は、PDF 添付ファイルと PDF エクスポート・ファイルの両方に適用されます。「差異」の結果は、このタスクの初回実行後にのみ適用されます。 前の結果がない場合に、前の結果との差異は存在しません。一度に比較可能な行の最大数は、5000 です。 結果行の数が最大数を超える場合、差異の結果にメッセージ

   「(最初の 5000 行のみ比較)」

   が表示されます。
5. 「タスク・パラメーター」ペインに、すべてのパラメーター値を入力します。 パラメーターは、選択したレポートに応じて異なります。
6. 「適用」をクリックします。

自動実行のための API

デフォルトでは、Guardium アプリケーションにはレポートへの多くの API 関数にリンクした設定データが添付されています。ユーザーには、GUI を通じてレポート・データの API への作成済み呼び出しが提供されます。「レポート」の「API 割り当て」を使用して、事前定義された Guardium レポートまたはカスタム・レポートへの追加 API 関数にリンクできます。メニュー選択項目「自動実行のための API」が「監査タスクの追加」に表示されます。 レポート内に API パラメーターにリンクされたフィールドがある、 該当する事前定義 Guardium レポートまたはカスタム・レポートを選択する際にレポートします。 自動実行のための API メニュー項目が表示される事前定義レポートの例として、 「アクセス・ポリシー違反」、「ディスカバーされたデータベース」、および「Guardium グループの詳細」があります。

ワークフロー・ビルダー

ワークフロー・ビルダーで作成されるイベント・タイプの 正式な順序の管理は、「監査タスク」ウィンドウの「イベントおよび追加列」ボタンをクリックして行います。このボタンは、監査タスクを作成して保存すると表示されます。この追加のボタンは、監査タスクを保存しないと表示されません。以下の手順で、監査タスクを追加する際にこれらのワークフロー・アクティビティーを構成します。

1. 監査タスクを作成して保存します。保存後、追加の「イベントおよび追加列」ボタンが表示されます。
2. この追加のボタンをクリックします。
3. 次の画面で、「イベントおよびサインオフ」ボックスにチェック・マークを付けます。 ワークフロー・ビルダーで作成したワークフローが「イベントおよびサインオフ」の選択項目として表示されます。

4. この選択項目を強調表示します。選択内容を適用 (保存) します。
5. 追加の情報 (会社コード、ビジネス・ユニット・ラベルなど) が、ワークフロー・レポートの 一部として必要である場合は、この情報を画面の「追加列」セクションに追加して、「適用」(保存) をクリックします。 事前定義グループ列または作成したグループ列を選択するには、「タイプ」列を「グループ」に変更します。 完了したら、このウィンドウを閉じます。
6. 監査タスクを適用 (保存) します。監査プロセス定義全体を適用 (保存) します。

この「イベントおよび追加列」ボタンは、すべての監査タスクで表示されます。 このボタンの上にカーソルを置くと、特定の監査タスクにリンクされた「イベント」または「サインオフ」列が監査タスクにある かどうかをユーザーに通知する情報バルーンが表示されます。

注:

監視データ・レベルでのデータ・レベル・セキュリティーが有効な場合、ユーザーに対してユーザーのデータベース情報のみが表示されるよう、監査プロセスの出力がフィルタリングされます。

「監査タスクの追加」内の「レポート」の選択項目は、 「未処理のイベント」および「イベント状況の移行」の 2 つのプロシージャー型レポートです。これら 2 つのレポートを 2 つの新規監査タスクに追加して、すべてのワークフロー・イベントおよび移行の詳細を表示します。 これら 2 つのレポートは、フィルターに掛けられません (監視データ・レベル・セキュリティー・フィルター操作は適用されません)。 これら 2 つのレポートは、admin ユーザーおよび admin ロールを持つユーザー に対するレポート・リストでのみ、デフォルトで選択可能です。

「追加列」ボタンは分類タスクでは使用不可になっています。

監査タスクのコピーを作成。プロセスのコピーを作成している場合、コピー・プロセスを保存する前にコピー・タスクに変更を加えた場合、オリジナル・タスクに関連付けられたワークフローのコピーは作成されません。

イベント状況の削除は、その状況が何らかのイベントの最初の状況ではなく、アクションに使用されていない場合にのみ許可されます。検証では、状況の削除を防止するイベント/アクションのリストが提供されます。

ワークフロー・イベントの所有者または作成者は、このイベントのすべての状況を、これらの状況に対してどのようなロールが割り当てられているかに関係なく、常に見ることが可能です。

セキュリティー・アセスメント・タスクの定義

コンプライアンス・ワークフロー自動化プロセスの定義をまだ開始していない場合は、この手順を実行する前に、ワークフロー・プロセスを作成してください。使用するアセスメントがまだ定義されていない場合は、最初に定義してください。

1. 「新規タスクの追加」ペインが開いていない場合、「監査タスクの追加」をクリックします。
2. 「セキュリティー・アセスメント」ボタンをクリックします。
3. 「セキュリティー・アセスメント」リストから、セキュリティー・アセスメントを選択します。
4. 「PDF の内容」の選択項目には、「レポート」(現在の結果)、「差異」(1 つ前のレポートと新規レポートの間の差異)、 および「レポートと差異」(その両方) があります。
5. 「適用」をクリックします。

注:

監視データ・レベルでのデータ・レベル・セキュリティーが有効な場合、ユーザーに対してユーザーのデータベース情報のみが表示されるよう、監査プロセスの出力がフィルタリングされます。

セキュリティー・アセスメント・タスクが空である場合 (例えば、ロール・セットなしのセキュリティー・アセスメントなど)、 この空のセキュリティー・アセスメントは、「監査ビルダー」のドロップダウン・リストには表示されません。

エンティティー監査証跡タスクの定義

コンプライアンス・ワークフロー自動化プロセスの定義をまだ開始していない場合は、この手順を実行する前に、ワークフロー・プロセスを作成してください。

1. 「新規タスクの追加」ペインが開いていない場合、「監査タスクの追加」をクリックします。
2. 「エンティティー監査証跡」ボタンをクリックします。
3. 監査するエンティティーのタイプを選択します。選択したタイプに応じて、 以下の情報の指定が必要になります。
   - オブジェクト: オブジェクト名を入力します。
   - オブジェクト・グループ: リストからオブジェクト・グループを選択します。
   - クライアント IP: クライアント IP アドレスを入力します。
   - クライアント IP グループ: クライアント IP グループを選択します。
   - サーバー IP: サーバー IP アドレスを入力します。
   - アプリケーション・ユーザー名: アプリケーション・ユーザー名を入力します。
4. 「CSV/CEF ファイル・ラベル」、「CEF を Syslog に書き込む」、「圧縮」、および「PDF のエクスポート」には、いくつかの選択項目があります。『CSV または CEF ファイルのエクスポート』を参照してください。
5. 「タスク・パラメーター」ペインで、ランタイム・パラメーター値を指定します (「開始」および「終了」期間のみが必須です)。
6. 「適用」をクリックします。

注: 監視データ・レベルでのデータ・レベル・セキュリティーが有効な場合、ユーザーに対してユーザーのデータベース情報のみが表示されるよう、監査プロセスの出力がフィルタリングされます。

プライバシー・セット・タスクの定義

コンプライアンス・ワークフロー自動化プロセスの定義をまだ開始していない場合は、この手順を実行する前に、ワークフロー・プロセスを作成してください。使用するプライバシー・セットがまだ定義されていない場合は、最初に定義してください。

1. 「新規タスクの追加」ペインが開いていない場合、「監査タスクの追加」をクリックします。
2. 「プライバシー・セット」ボタンをクリックします。
3. 「プライバシー・セット」リストから、プライバシー・セットを選択します。

4. 「アクセス詳細別レポート」または「アプリケーション・ユーザー別レポート」のいずれかを選択して、 結果のソートおよび表示方法を指定します。
5. 「CSV/CEF ファイル・ラベル」、「CEF を Syslog に書き込む」、「圧縮」、および「PDF のエクスポート」には、いくつかの選択項目があります。『CSV または CEF ファイルのエクスポート』を参照してください。
6. 「期間の開始」および「期間の終了」ボックスに、レポートの開始および終了の日付を入力します。
7. 「適用」をクリックします。

注: 監視データ・レベルでのデータ・レベル・セキュリティーが有効な場合、ユーザーに対してユーザーのデータベース情報のみが表示されるよう、監査プロセスの出力がフィルタリングされます。

分類プロセス・タスクの定義

コンプライアンス・ワークフロー自動化プロセスの定義をまだ開始していない場合は、この手順を実行する前に、ワークフロー・プロセスを作成してください。使用する分類プロセスがまだ定義されていない場合は、最初に定義してください。

1. 「新規タスクの追加」ペインが開いていない場合、「監査タスクの追加」をクリックします。
2. 「分類プロセス」ボタンをクリックします。
   注: 分類プロセスで機密データが返されることがあり、これらの結果が PDF ファイルや CSV ファイルに追加されることに対する警告があります。
3. 「分類プロセス」リストから、分類プロセスを選択します。「適用」をクリックします。

注: 監視データ・レベルでのデータ・レベル・セキュリティーが有効な場合、ユーザーに対してユーザーのデータベース情報のみが表示されるよう、監査プロセスの出力がフィルタリングされます。

外部フィード・タスクの定義

このタイプのワークフロー自動化タスクは、Guardium が収集したデータを外部アプリケーションにフィードし、 データをそのアプリケーションが認識するフォーマットにマッピングします。このタスク・タイプは、 パッチによって使用可能になる、追加のコストを要するフィーチャーです。

注: このフィーチャーを中央マネージャー環境で使用する場合、 外部フィード・パッチを中央マネージャーおよびこのタスクが実行されるすべての管理対象ユニットにインストールする必要があります。

Guardium から外部アプリケーションへのデータのマップについて詳しくは、 購入したオプションの資料を参照してください。

コンプライアンス・ワークフロー自動化プロセスの定義をまだ開始していない場合は、この手順を実行する前に、ワークフロー・プロセスを作成してください。

1. 「新規タスクの追加」ペインが開いていない場合、「監査タスクの追加」をクリックします。
2. 「外部フィード」をクリックします。
3. 「フィード・タイプ」リストからフィード・タイプを選択します。
4. 次に表示されるコントロールは、選択したフィード・タイプに応じて異なります。 特定の外部フィード・タイプに関する追加情報については、『オプションの外部フィード』を参照してください。
5. 「イベント・タイプ」リストから、イベント・タイプを選択します。
6. 「レポート」リストからレポートを 1 つ選択します。選択したレポートに応じて、「タスク・パラメーター」ペインに表示されるパラメーターの数が異なります。
7. 「抽出ラグ」ボックスにフィードが遅延される時間数を入力するか、「継続」ボックスにマークを付けて監査タスクが実行される直前の時間までのデータを組み込みます。
8. 「データ・ソース」ペインで、外部フィードの 1 つ以上のデータ・ソースを指定します。
9. 「タスク・パラメーター」ペインに、すべてのパラメーター値を入力します。 パラメーターは、選択したレポートに応じて異なります。
10. 「適用」をクリックします。

結果の表示または署名

1. コンプライアンス・ワークフロー自動化の結果を開きます。
2. 署名が必要な場合は、「結果に署名」ボタンをクリックしてください。
3. オプション。2これらの結果を別のユーザーに転送するには、「エスカレート」をクリックしてください (セクションの『プロセス結果のエスカレート』を参照)。
4. 「このウィンドウを閉じる」をクリックします。

注: 未処理のイベントがある場合、監査ビューアーからも To-do リストからも結果に署名できません。 未処理のイベントがある場合に、結果への署名が試みられると、以下のメッセージが表示されます。

監査プロセスに署名できません。保留イベントがあります。

この結果に署名する前に、未解決のイベントをすべて更新してください。

注: 監査プロセスの結果を表示している際に、その結果に関連付けられたイベントがある場合、 すべてのイベントが最終状態になるまでは、その結果についての「結果に署名」ボタンが使用可能にならないか、 このユーザーでは (データ・レベル・セキュリティーにより) 表示できない可能性があります。
注: このレポートには、 日付または最終アクション時刻も含まれます。これらは「受信者」と「状況」の間の列にあります。 このレポートには、結果がユーザー AAA によって署名されたことの他に、そのユーザー AAA がこの結果に署名した時間も示されます。

署名または表示せずに結果をリリース

1. 「To-Do リスト」パネルを開きます。
2. 配布リストの次の受信者にリリースする結果の「続行」ボタンをクリックします。
3. 「このウィンドウを閉じる」をクリックします。

結果配布の表示

1. コンプライアンス・ワークフロー自動化の結果を開きます。
2. 「詳細を表示」ボタンをクリックして、「配布状況」パネルを展開します。
3. 「このウィンドウを閉じる」をクリックします。

結果に追加された受信者コメントの表示

1. コンプライアンス・ワークフロー自動化の結果を開きます。
2. 「詳細を表示」ボタンをクリックして、「コメント」パネルを展開します。
   注: これらは、Guardium システムからレポート・ページが取り出された際に結果に添付されたコメントです。独自にコメントを追加する場合、または他の受信者が同時にコメントを追加している場合、(ブラウザーのリフレッシュ機能を使用して) ページをリフレッシュするまでこれらのコメントは表示されません。
3. 「このウィンドウを閉じる」をクリックします。

プロセス結果のエスカレート

プロセス結果の受信者は、結果通知をレビューまたはサインオフ (あるいはその両方) のために、他の受信者に転送できます。 結果をオリジナルの監査およびサインオフ証跡の外側の受信者にエスカレートする場合、 結果に CSV ファイルが含まれていると、そのファイルは通知には組み込まれません。

「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」メニューで「結果をすべてのユーザーにエスカレート」ボックスがチェックされている場合、監査結果の受信者が誰かに関係なく、エスカレーションにシステムのどのユーザーも含めることができます。このボックスにチェック・マークを付けると、監視データ・レベルでのデータ・レベル・セキュリティー が有効になっている場合であっても、監査プロセスの結果がすべてのユーザーにエスカレートされます。 デフォルトでは有効に設定されています。チェック・ボックスが無効になっている (チェック・ボックスに チェック・マークが付けられていない) 場合、監査プロセスのエスカレーションは ユーザー階層の上位レベルのユーザーに対してのみ許可されます。 チェック・ボックスが無効になっており、ユーザー階層がない場合、エスカレーションは許可されません。

また、イベント権限に応じても異なります。例えば、infosec ユーザーは status1 のイベントのみ 表示可能であり、dba ユーザーは status2 のイベントのみ表示可能です。dba ユーザーが受け取る結果は、infosec ユーザーが「エスカレート」をクリックすると表示される結果とは異なります。 infosec が dba に対してエスカレートすることは可能であり、その場合 dba は行が含まれない監査結果を受け取ることになります。

1. 転送するワークフロー自動化結果が開かれていない場合は、開いてください。
2. 「エスカレート」をクリックします。
3. 「受信者」リストから受信者を選択します。
4. 「必要なアクション」列で、 「レビュー」(デフォルト) または「レビューと署名」を選択します。
5. 「エスカレーション」ボタンをクリックして、操作を完了します。

注:

監査プロセスの結果をユーザー・グループに対してエスカレートすることはできません。ユーザーまたはロールに対してのみエスカレートできます。

To-Do リストに既に結果を得ているユーザーに対してエスカレートする際には、 追加の E メールを送信するかどうかを訊ねるポップアップ・メッセージが表示されます。yes の場合、追加の E メールがそのユーザーに送信されますが、To-Do リストは増分されません。

コンプライアンス・ワークフロー自動化プロセスのスケジュールまたは実行

1. 「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」にナビゲートして「監査プロセス・ビルダー」を開きます。
2. 「プロセス選択リスト」からプロセスを選択します。
3. 「変更」をクリックして、「監査プロセス定義」パネルを開きます。
4. プロセスを 1 回実行する場合は「今すぐ 1 回実行」を、 プロセスのスケジュールを定義する場合は「スケジュールの変更」をクリックします。
   注: プロセスにスケジュールを定義した後、 プロセスは「アクティブ」とマークが付けられている場合のみ、そのスケジュールに従って実行されます。監査プロセスをアクティブまたは非アクティブにする方法については、次のセクションを参照してください。

コンプライアンス・ワークフロー自動化プロセスのアクティブ化または非アクティブ化

監査プロセスにスケジュールを定義した後、プロセスは「アクティブ」とマークが付けられている場合のみ、 そのスケジュールに従って実行されます。

監査プロセスをアクティブまたは非アクティブにする手順は、次のとおりです。

1. 「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」にナビゲートして「監査プロセス・ビルダー」を開きます。
2. 「プロセス選択リスト」から監査プロセスを選択します。
3. 「変更」をクリックします。
4. 「監査プロセス定義」パネルで、「アクティブ」ボックスにマークを付けて、プロセスをスケジュールに従って実行開始します。 または、「アクティブ」ボックスをクリアして、(定義されたスケジュールを無視して) プロセスの実行を停止します。
   注: プロセスをアクティブにしているが、 スケジュールがない場合は、「スケジュールの変更」をクリックしてプロセス実行のスケジュールを定義してください。
5. 「保存」をクリックします。

- 監査ワークフローの作成方法
  あらかじめ設定したスケジュールで事前定義されたレポートを作成する監査プロセス・ワークフローを作成し、レビューと署名のためにレポートをデータベース管理者に割り当て、レビュー済みのレポートがさらに上司のレビューと署名のために送られるようにします。
- ワークフロー・プロセスの結果を開く
  「表示」を使用して、ワークフロー・プロセスの結果を表示します。
- Guardium グループを使用してワークフローを配布する方法
  受信者グループ・オプションを使用することによって、事前定義されたカスタム・マッピングに基づいてそれぞれの結果をそれぞれの Guardium ユーザーに送信する、単一のコンプライアンス・ワークフロー監査プロセスを定義します。
- 監査プロセスの To-do リスト
  このトピックでは、「監査プロセスの To-do リスト」と、これを開いて使用するために必要なステップについて説明します。

親トピック: モニターおよび監査

監査ワークフローの作成方法

あらかじめ設定したスケジュールで事前定義されたレポートを作成する監査プロセス・ワークフローを作成し、レビューと署名のためにレポートをデータベース管理者に割り当て、レビュー済みのレポートがさらに上司のレビューと署名のために送られるようにします。

このタスクについて

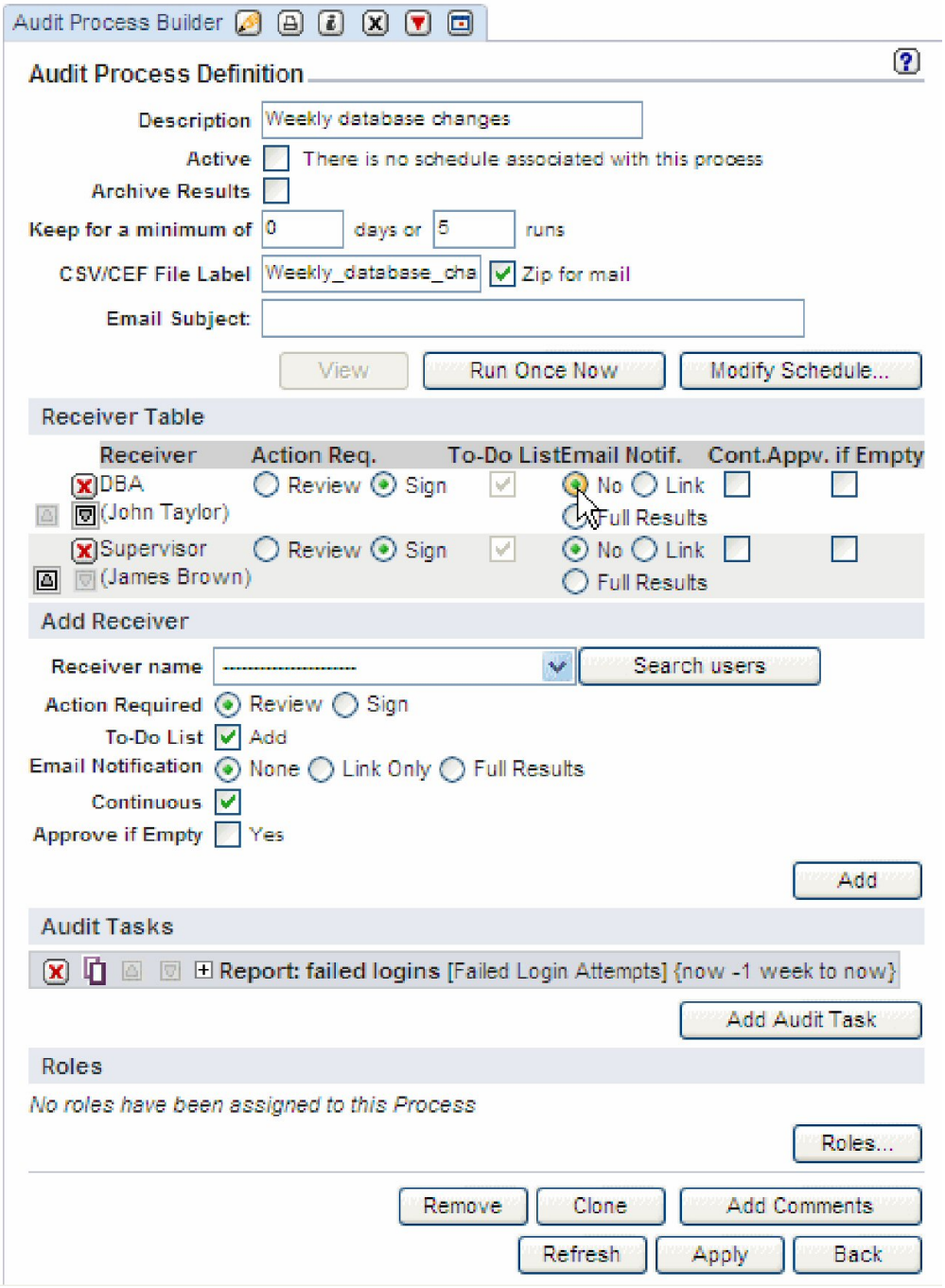
顧客の監査プロセスのワークフロー・ステップを自動化します。

これについての詳細は、『コンプライアンス・ワークフロー自動化』トピックを参照してください。

手順

1. 「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」にナビゲートして「監査プロセス・ファインダー」を開きます。
2. 「新規」ボタンをクリックして、「監査プロセス定義」パネルを開きます。

「監査プロセス定義」パネルは、「一般」、「受信者の表」および「監査タスク」の 3 つのセクションに分かれています。



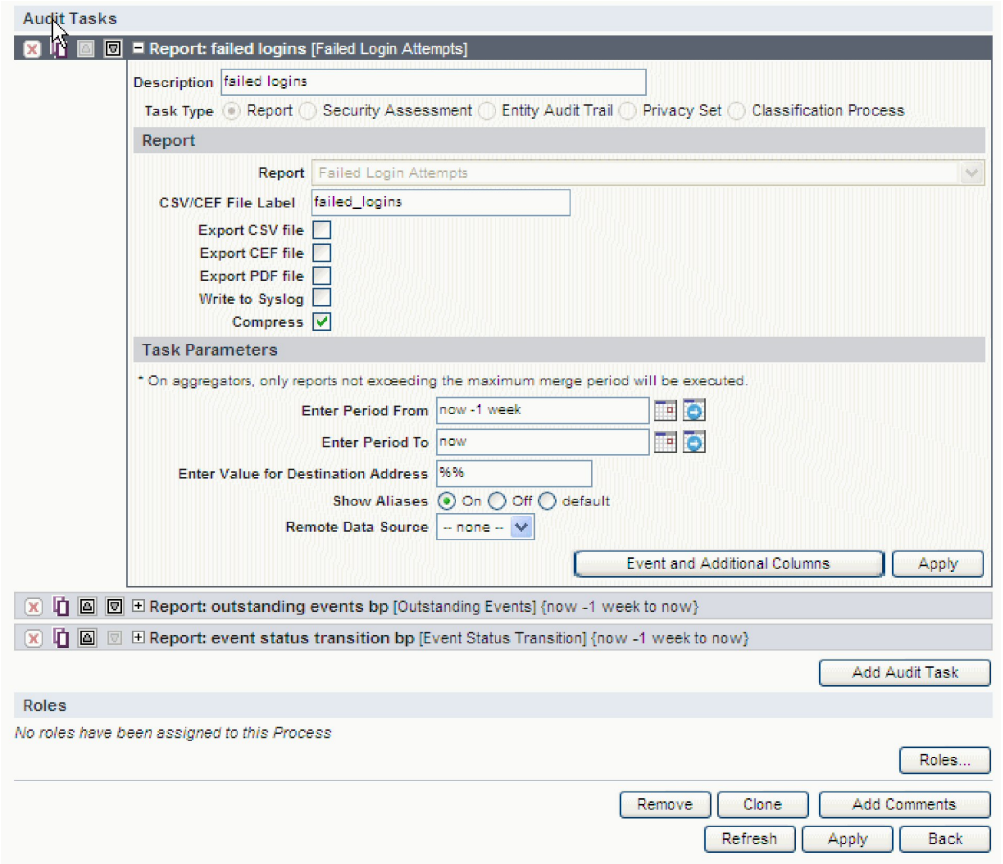
「監査プロセス・ビルダー」メニュー画面

3. 「一般」セクションに進みます。「記述」ボックスに名前を入力します。 アポストロフィ文字は含めないでください。
4. 「アクティブ」ボックスにチェックを付けて、プロセスにスケジュールを関連付けます。 プロセスを保存するには、少なくとも 1 つの監査タスクが定義されている必要があります。
5. 保存期間の期限が切れた後にオフラインで結果を保存する場合は、「結果のアーカイブ」ボックスにマークを付けます。 アーカイブされた結果は、後ほど、アプライアンスにリストアして再度表示することができます。
6. 「最低保持期間」の「(n) 日」または「(n) 実行」ボックスに、結果を保存する期間を入力します。 期間は日数 (デフォルトは 0) または実行数 (デフォルトは 5) のどちらかで入力します。 その後、結果はアーカイブされ (アーカイブ・ボックスにマークが付いている場合)、アプライアンスからパージされます。

7. 1 つ以上のタスクで CSV または CEF ファイルが作成される場合は、 オプションとして、すべてのファイルに含めるラベルを「CSV/CEF ファイル・ラベル」ボックスに入力できます。 これらのファイルも (Zip 形式に) 圧縮できます。圧縮するには、「CSV を zip してメール」ボックスをクリックしてチェック・マークを付けます。
8. 監査プロセス定義の「E メールの件名」フィールドは、その監査プロセスの全受信者の E メールに使用されます。 件名には、実行時にその件名を置き換える以下の変数を 1 つ (以上) 含めることができます。
   - %%ProcessName は、監査プロセスの記述に置き換えられます。
   - %%ExecutionStart は、最初のタスクの開始日時に置き換えられます。
   - %%ExecutionEnd は、最後のタスクの終了日時に置き換えられます。

   件名の入力時には、変数 (%% で始まる) の有無と、それらすべてが有効な変数であるかどうかが確認されます。

9. 「受信者」セクションに進みます。 ドロップダウン・ボックスを開いて、プロセスの受信者を追加します。 詳しい説明は、 『コンプライアンス・ワークフロー自動化』のトピックの「受信者の追加」を参照してください。 必要なアクション、To-do リストへの追加、E メール通知、および継続的な配布を決定するには、チェックを付ける必要があります。 これら項目の設定についても、詳しい説明は『受信者の追加』を参照してください。 この例では、受信者の「継続」ボックスにはチェックを付けません。 「継続」チェック・ボックスにマークが付けられている場合、 配布は中断なしでリストの次の受信者に継続されます。 「継続」チェック・ボックスがクリアされている場合、 次の受信者への配布は、現行の受信者が必要なアクション (レビューまたは署名) を実行するまで保留になります。 この例では、上司に送る前に、DBA がレポートを見て署名する必要があります。
10. 「タスク」セクションに進みます。 プロセスを保存する前に、少なくとも 1 つの監査タスクを定義する必要があります。
11. レポート・タスクを定義します。
    a. 「タスクの新規追加」ペインが開いていない場合は、 「監査タスクの追加」をクリックします (図を参照)。
    b. 「レポート」ボタンをクリックします。
    c. 必要に応じて、CSV または CEF ファイル出力を作成して、syslog に書き込みます。
    d. 「タスク・パラメーター」ペインに、すべてのパラメーター値を入力します。 パラメーターは、選択したレポートに応じて異なります。
    e. 「適用」をクリックします。

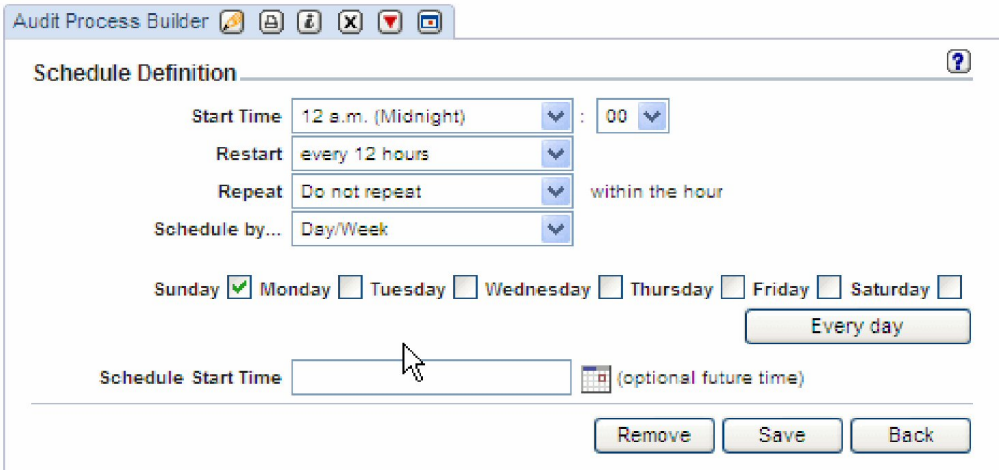


監査タスク - レポート

12. オプションでセキュリティー・ロールを割り当てます。
    a. 1 つ以上のセキュリティー・ロール (レポート定義など) を割り当てる 項目を開くか、選択します。
    b. 「ロール」ボタンを クリックします。
    c. 「セキュリティー・ロールの割り当て」パネルで、割り当てるすべてのロールにマークを付けます (自分のアカウントに 割り当てられたロールのみが表示されます)。
    d. 「適用」をクリックします。
13. オプションでコメントを追加します。
14. 監査ワークフロー・プロセスをスケジュールまたは実行するためのボタンをクリックします (リンクを参照)。
15. 「適用」をクリックします。
16. コンプライアンス・ワークフロー自動化プロセスのスケジュールまたは実行

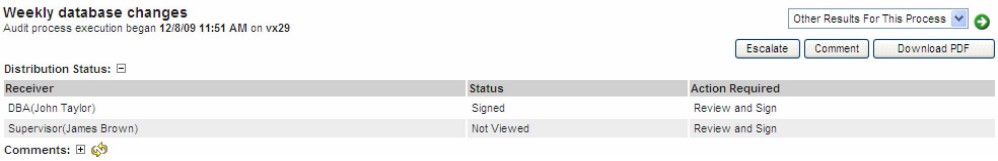
    「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」にナビゲートして「監査プロセス・ファインダー」を開きます。
    a. 「プロセス選択リスト」からプロセスを選択します。
    b. 「変更」をクリックして、「監査プロセス定義」パネルを開きます。
    c. プロセスを 1 回実行する場合は「今すぐ 1 回実行」をクリックし、プロセスのスケジュールを定義する場合は「スケジュールの変更」をクリックします。

    注: プロセスにスケジュールを定義した後、 プロセスは「アクティブ」とマークが付けられている場合のみ、そのスケジュールに従って実行されます。



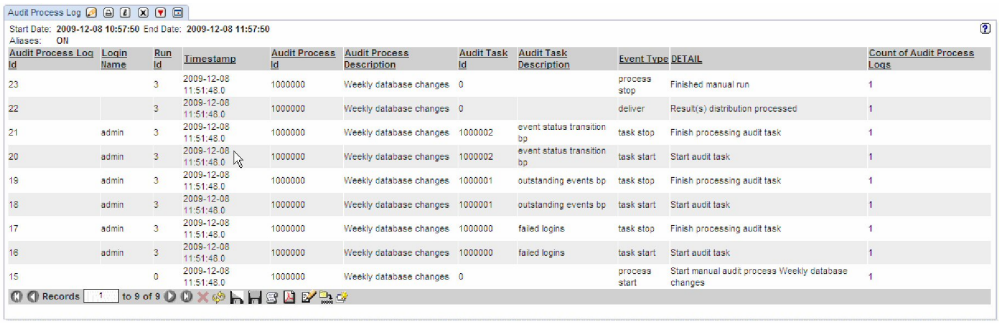
17. レポートの署名とレビュー

    レポートを実行すると、レポートから配布状況を監視できます。 この例では、DBA はレポートを表示、署名していますが、上司はまだです。



    配布状況

    「監査プロセス・ログ」レポートには、すべてのタスクに関する 詳細なアクティビティー・ログが、開始時刻および終了時刻も含めて示されます。 このレポートは、「レポート」 > 「Guardium 運用レポート」 > 「監査プロセス・ログ」にナビゲートすると使用できます。監査タスクには、開始時刻と終了時刻が示されます。



    監査プロセス・ログの例

親トピック: 監査プロセスの作成

ワークフロー・プロセスの結果を開く

「表示」を使用して、ワークフロー・プロセスの結果を表示します。

以下のいずれかを実行します。

- 「ワークフロー自動化 To-Do リスト」(『監査プロセスの To-do リスト』を参照) を開き、表示または署名する結果の「表示」をクリックします。
- To-do リストまたは結果へのハイパーテキスト・リンクが含まれている E メール通知を受信した場合は、それらのリンクの 1 つをクリックして、E メールから直接 To-do リストまたは結果を開きます。E メールにアクセスしているロケーションにおける Guardium システムへのアクセス権限を持っている必要があります (持っていない場合、これらのリンクは機能しません)。Guardium システムにログインしていない場合は、ログインするようプロンプトが表示されます。

注: 新しい管理対象ユニットを中央マネージャーに登録するときに、監査結果が表示できないことがあります。アプリケーションは、管理対象ユニットが中央マネージャーに登録される前のタイム・スタンプが含まれている結果を表示しません。登録のタイム・スタンプには中央マネージャーの時刻が使用され、監査結果のタイム・スタンプには管理対象ノードの時刻が使用されます。このため、中央マネージャーの時刻が管理対象ユニットの時刻より先行している場合、管理対象ユニットの時刻が登録の時刻を過ぎるまでは、管理対象ユニットで生成された結果は表示されません。 これは、2 台のマシンのロケーションによって発生する可能性は低く、24 時間以内で起こります。管理対象ユニットでの監査プロセスの結果は、登録後 24 時間以内に表示できる必要があります。
親トピック: 監査プロセスの作成

Guardium グループを使用してワークフローを配布する方法

受信者グループ・オプションを使用することによって、事前定義されたカスタム・マッピングに基づいてそれぞれの結果をそれぞれの Guardium ユーザーに送信する、単一のコンプライアンス・ワークフロー監査プロセスを定義します。

得られる価値: 単一の監査プロセスを設定し、適切な結果を適切なマネージャーに配布します。これにより、受信者ごとに別個の監査プロセスを作成せずに済みます。

IBM Security Guardium のコンプライアンス・ワークフロー自動化では、スケジュールに基づいて、レポート、分類結果、およびセキュリティー・アセスメントの結果を Guardium ユーザーに自動配信します。結果の受信者は、Guardium ユーザー、Guardium のロール、またはユーザー・グループとして定義できます。

例えば、15 人の DBA マネージャーがいる大規模な組織において、マネージャーは他のマネージャーの DBA の活動を見ることなく、自らが管理する DBA の活動をレビューする必要がある、という場合を考えてみます。1 つの解決方法としては、マネージャーごとに 1 つずつの 15 種類の監査プロセスを設定する方法があります。 この方法は構成にたいへん時間がかかります。また、各監査プロセスのスケジュールを別々に設定する必要があり、15 種類すべての監査プロセスに対してすべての全体的な変更を個別に行う必要があるため、管理が困難です。

一方、ユーザー・グループ配布方式では、単一の監査プロセスの設定が可能で、マネージャー/DBA のマッピングに基づいて、適切な結果が各マネージャーに配布されます。このプロセスでは事前に必要な構成が多くなりますが、保守時間が削減されます。 スケジュールする必要がある監査プロセスは 1 つだけであり、変更を適用する必要があるロケーションも 1 つだけです。

ユーザー・マッピング

プロセスの最初のステップは、レポート配布の基礎となる、Guardium 内のデータ・エレメントに、ユーザーをマップすることです。このドキュメントで使用される例はオブジェクトに基づいていますが、これらの概念は Guardium 内の任意のデータ・エレメントに適用することもできます。

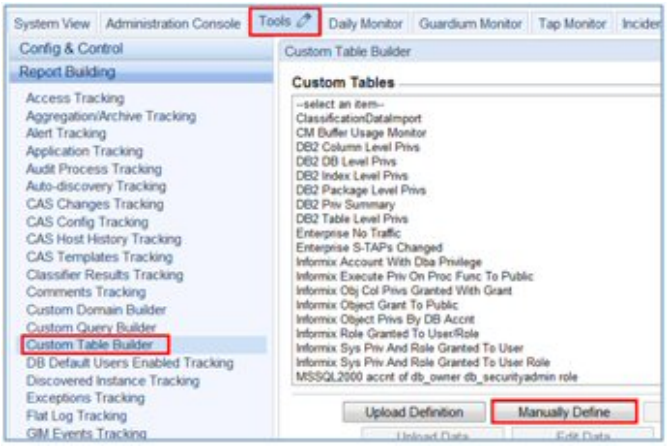
例: 3 人のユーザーは、データベース・サーバー内の監査要件 (PCI、HIPPA、および CCI) に基づいて、3 つの異なる表のセットに対し、以下のような責任を持っています。

表 1. ユーザーと表/オブジェクト

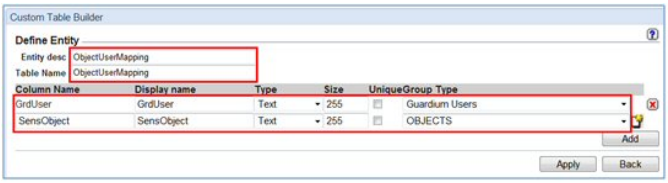
ユーザー	表/オブジェクト
User01	db2inst1.cc_numbers
User01	db2inst1.ccn
User02	db2inst1.ADDRESSES
User02	db2inst1.SSN_NUMBERS
User02	db2inst1.G_CUSTOMERS
User02	db2inst1.G_EMPLOYEES
User02	db2inst1.G_FUNDS
User03	db2inst1.doctor
User03	db2inst1.medicare
User03	db2inst1.med_history

この表を、手動またはデータ・アップロードによって、Guardium 内にカスタム表として追加する必要があります。以下のステップで、カスタム表を手動で作成する方法について説明します。スクリーン・ショットは「管理者」ユーザー・インターフェースのものですが、「ユーザー」ユーザー・インターフェース内からもアクセスできます。

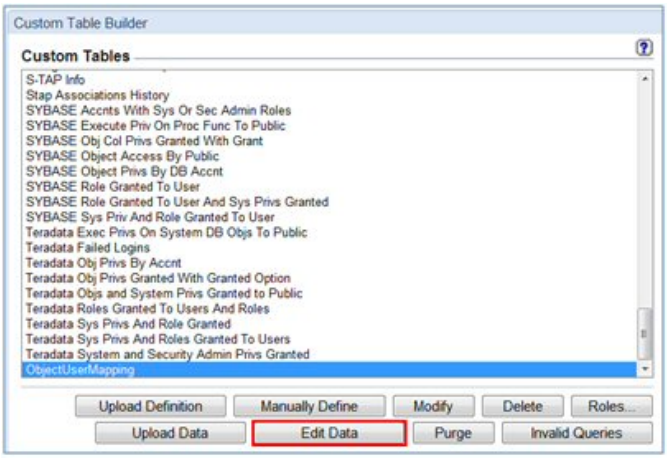
1. 「レポート」 > 「レポート構成ツール」 > 「カスタム表ビルダー」にナビゲートし、**「手動定義」**ボタンを押します。



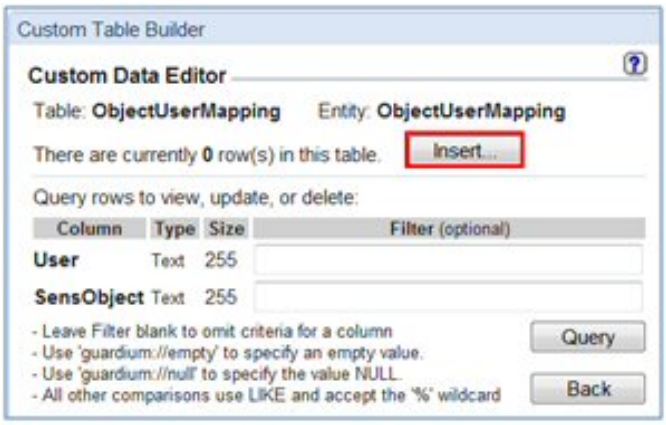
2. **「カスタム表ビルダー」**画面で、表のレイアウトを定義します。**「グループ・タイプ」**が Guardium 内の正しいデータ・エレメントと一致していることを確認します。完了したら**「適用」**および**「戻る」**を押します。



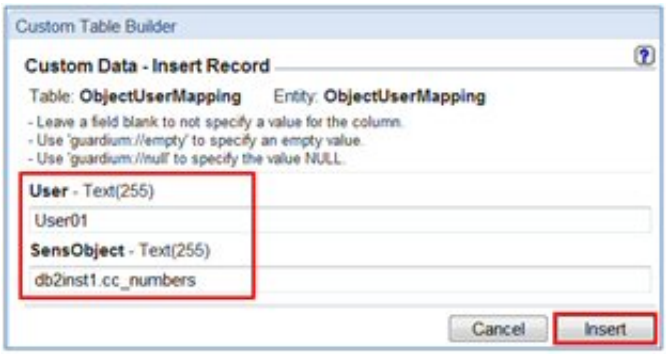
3. **「データの編集」**を押して、レコードを手動で追加します。なお、大量のデータがある場合は、**「データのアップロード」**を選択して外部データ・ソースからインポートします。



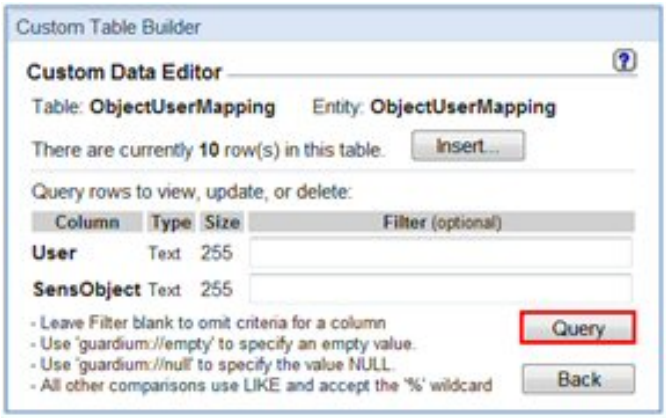
4. 「**挿入**」を押します。



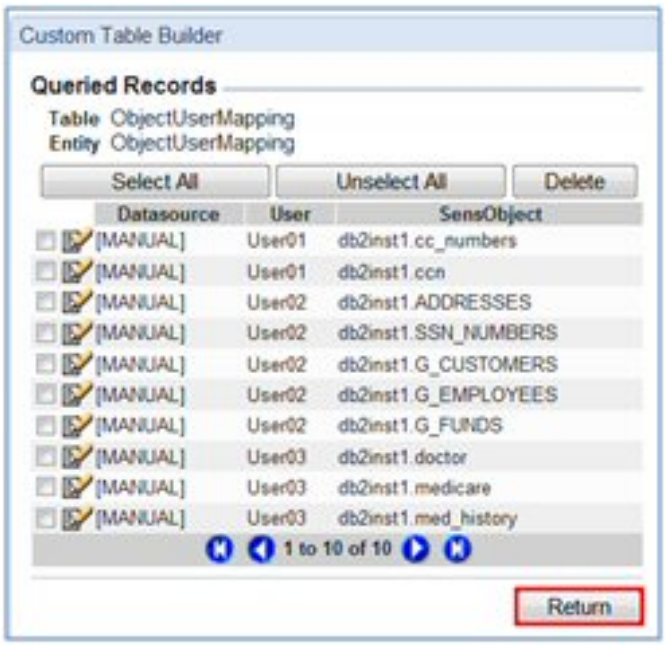
5. 値の組み合わせを入力し、すべての必須レコードを追加したら、「**挿入**」を押します。



6. 完了したら、「**照会**」ボタンを押して、データをレビューします。



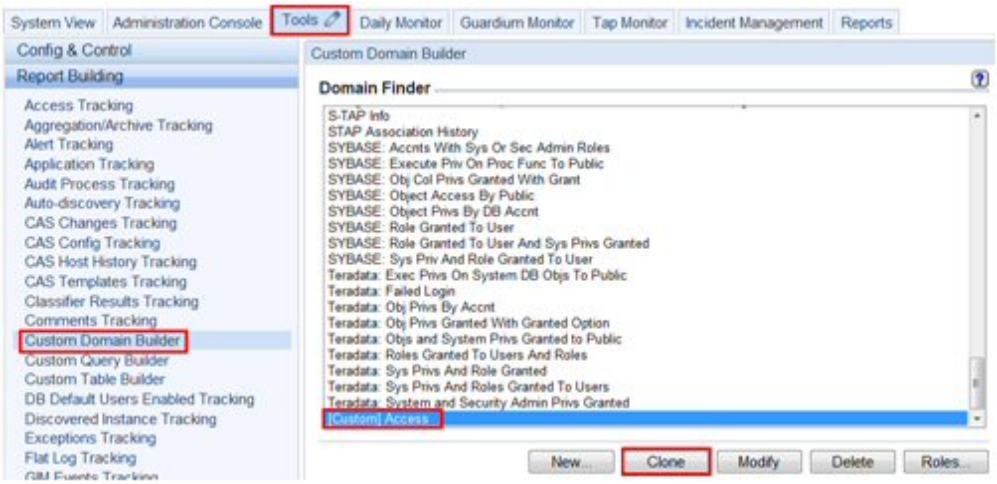
7. 完了したら、「先頭に戻る」を押します。



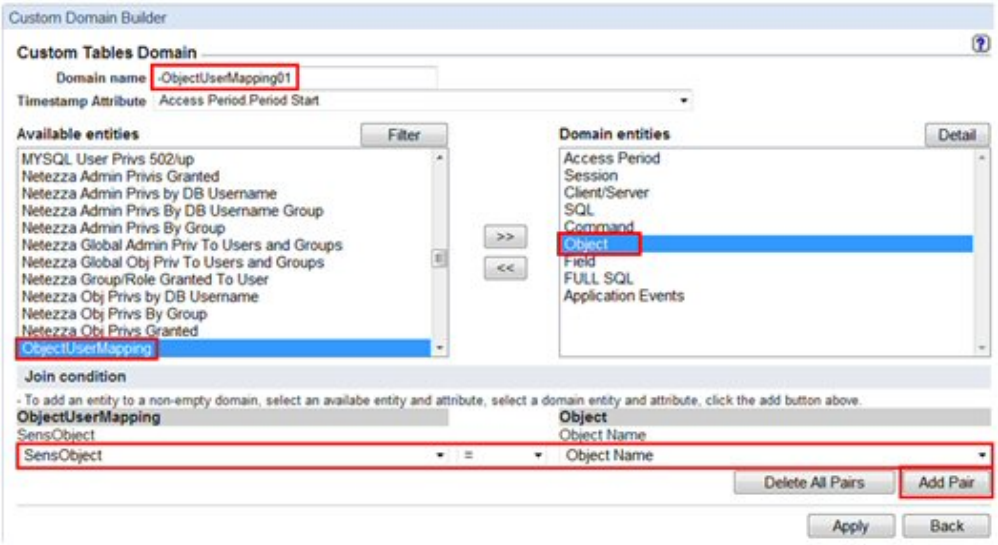
カスタム・ドメイン

次に、カスタム・ドメインを使用して、このカスタム表を Guardium 表構造に結合します。

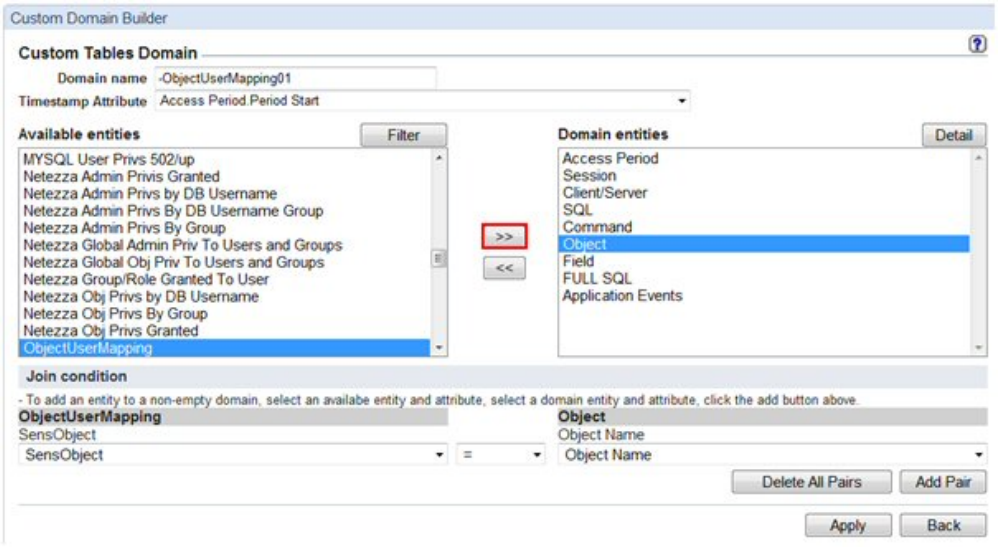
1. 「レポート」 > 「レポート構成ツール」 > 「カスタム・ドメイン・ビルダー」にナビゲートします。 *「[カスタム] アクセス」*を強調表示して、**「コピー」**を押します。



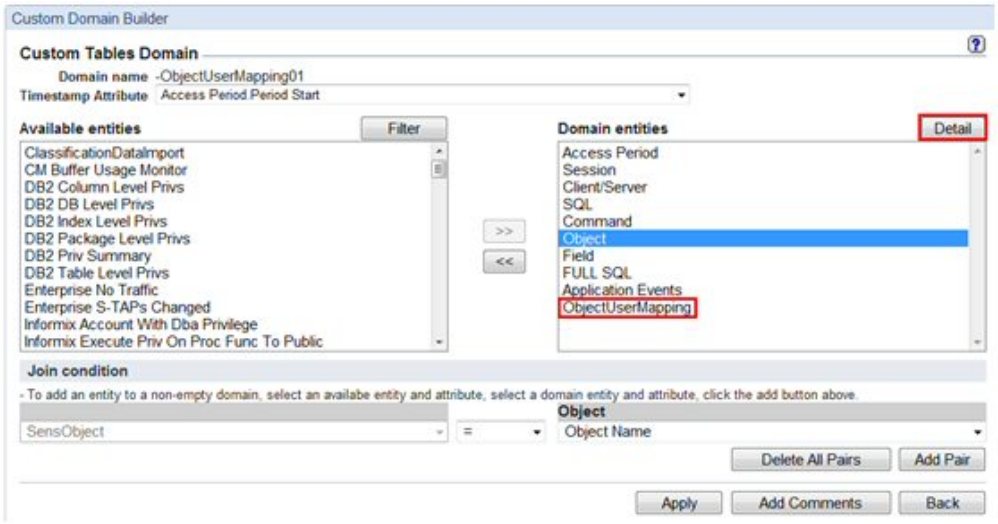
2. 「カスタム・ドメイン・ビルダー」で、次を行います。
   a. **「使用可能エンティティー」**で作成された新しい表を強調表示します
   b. **「ドメイン・エンティティー」**にある、カスタム表を結合する表を強調表示します。
   c. **「結合条件」**で、結合を作成する各表のフィールドを選択し、**「ペアの追加」**を押します



3. 矢印 (>>) ボタンを押して、**「使用可能エンティティー」**から**「ドメイン・エンティティー」**にカスタム表を移動します。

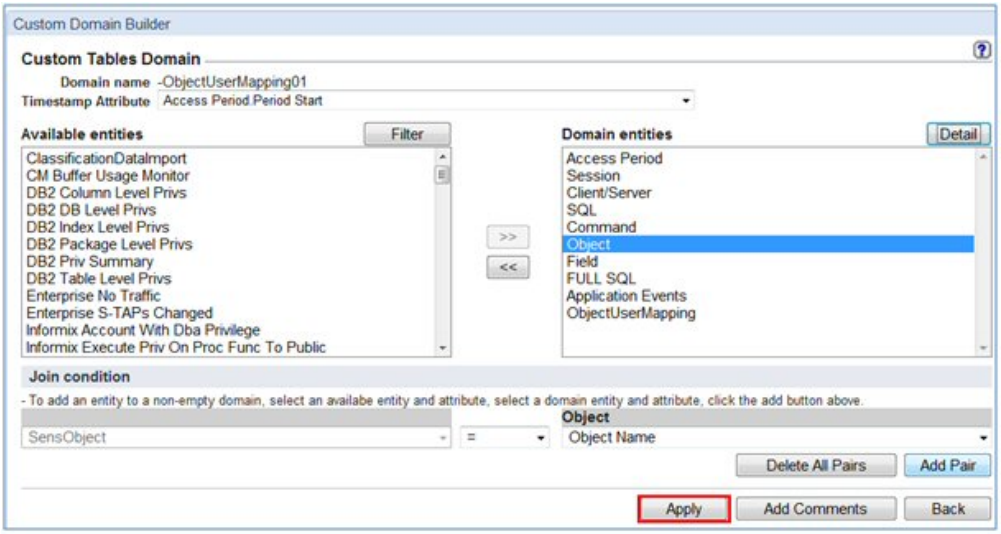


4. **「詳細」**ボタンを押して、結合をレビューします。



5. 結合が正しいことを確認し、**「閉じる」**を押します。

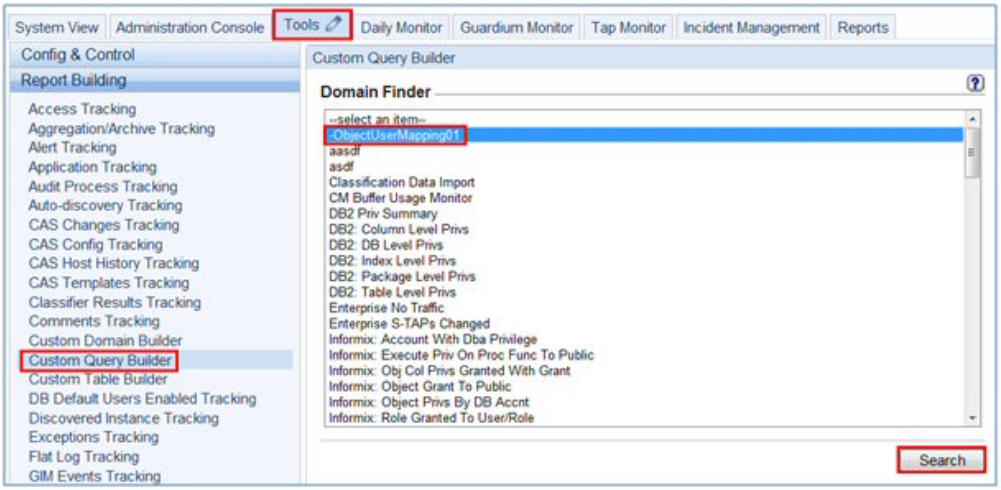
6. 「適用」を押して新しいカスタム・ドメインを保存します。



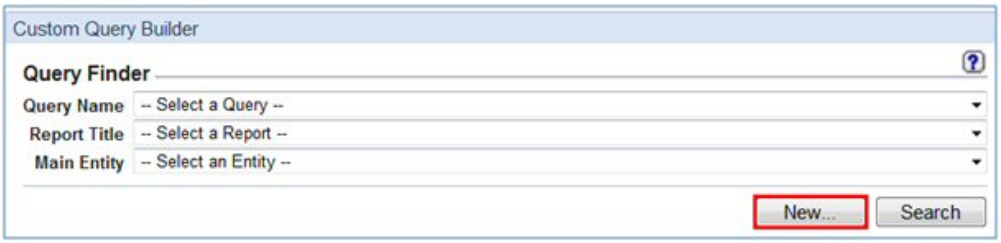
カスタム・レポート

次に、ユーザーに配布するレポートを作成します。

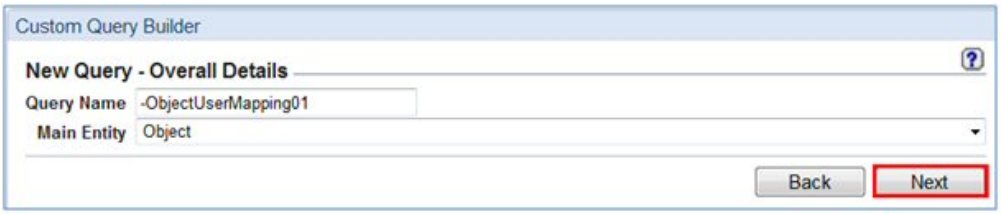
1. 「レポート」 > 「レポート構成ツール」 > 「レポート・ビルダー」にナビゲートし、「ドメイン」ドロップダウン・メニューから新規ドメインを選択します。



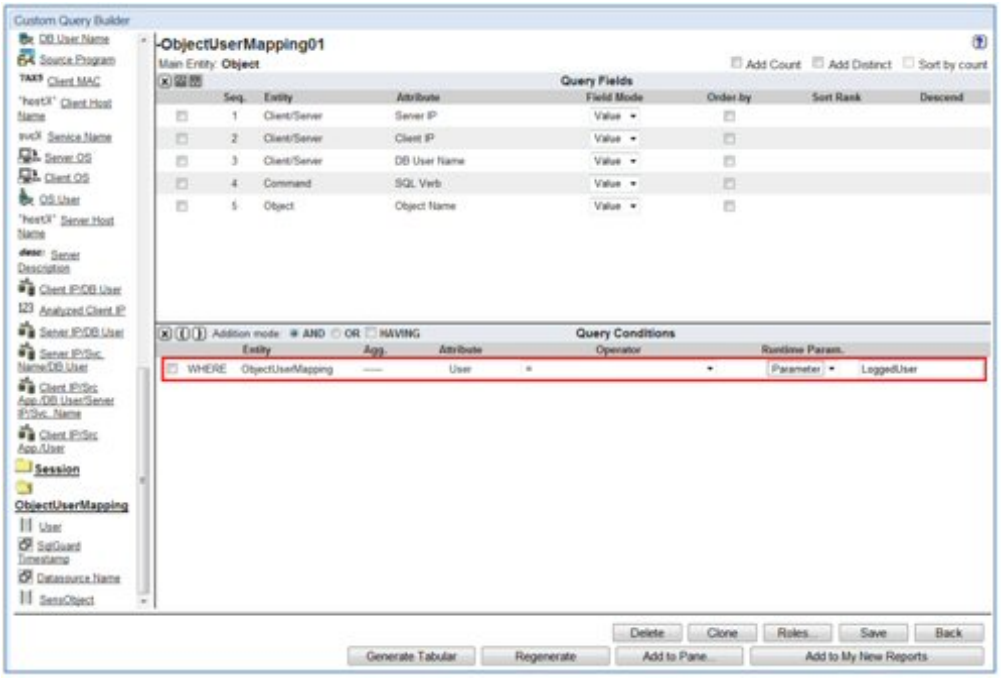
2. 「新規」をクリックします。



3. 「照会名」および「メイン・エンティティー」を入力し、「次へ」を押します。



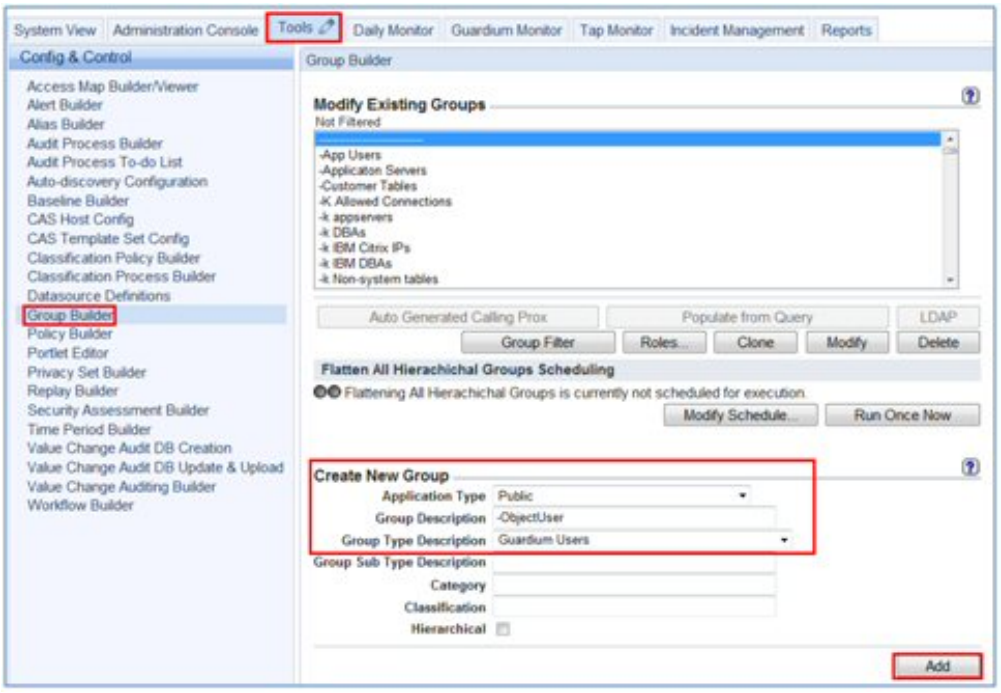
4. カスタム表内に作成されたユーザー・フィールドに対するランタイム・パラメーターを持つ新規レポートを作成します。



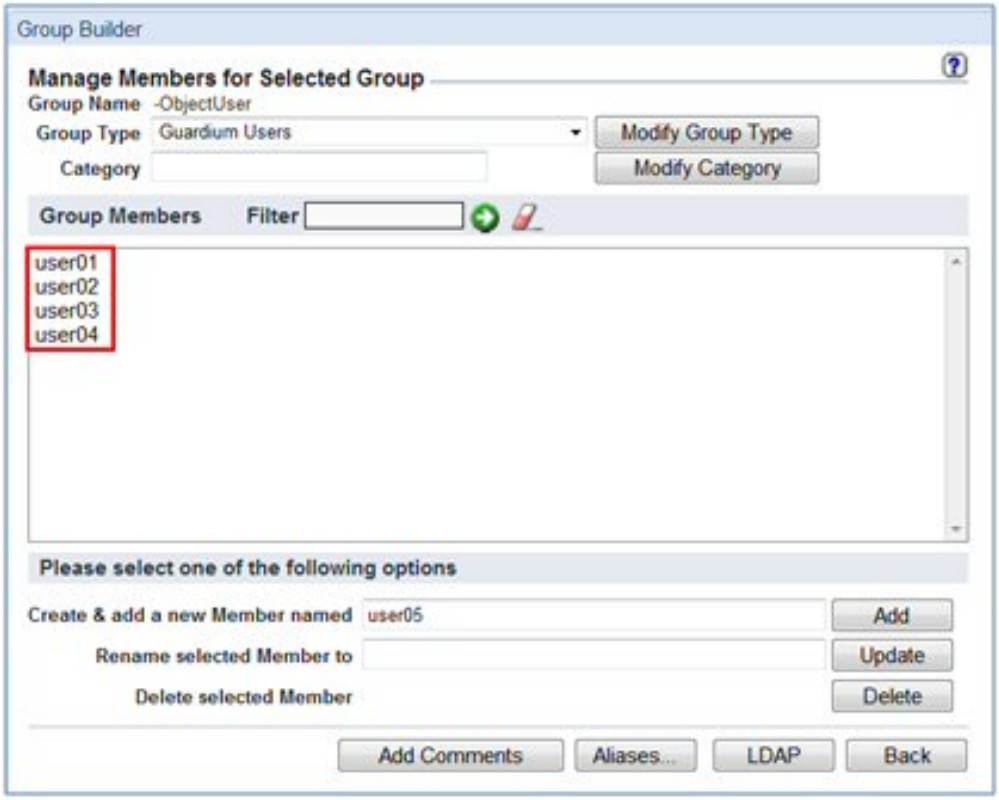
ユーザー・グループ

カスタム表に基づいて、新しいグループ「Guardium Users」を作成します。

1. 「設定」 > 「ツールとビュー」 > 「グループ・ビルダー」にナビゲートし、**「グループ・タイプ」**として**「Guardium ユーザー」**が設定された新規グループを作成します。

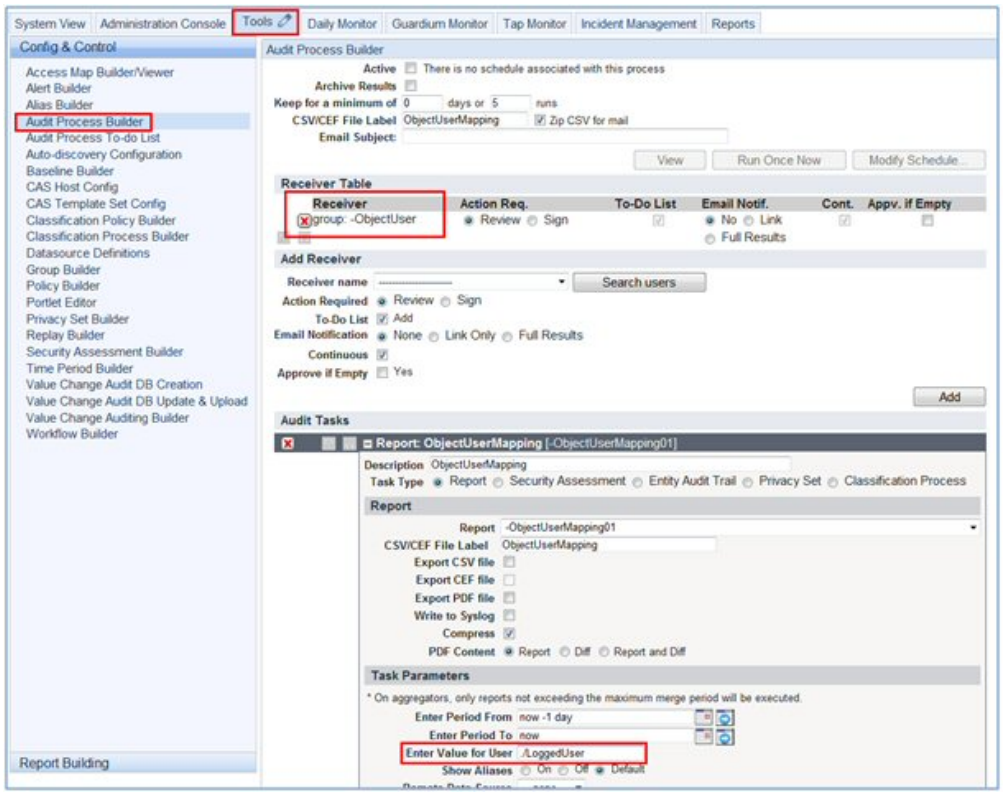


2. カスタム表からすべてのユーザーを追加します。



監査プロセス

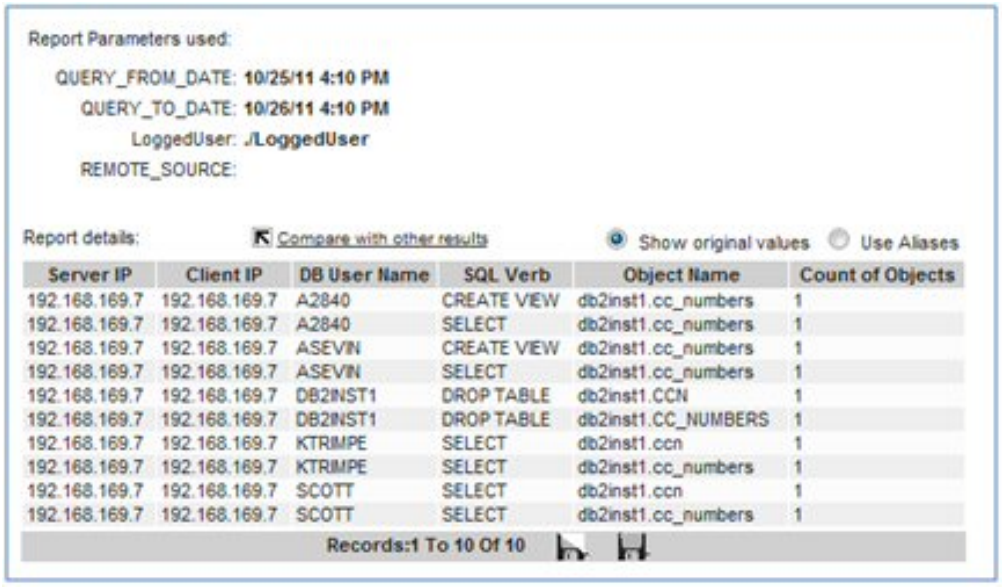
1. 新しい監査プロセスを作成します。
2. 『ユーザー・グループ』で作成されたグループを「**受信者**」として選択します。
3. ステップ 4 で作成されるカスタム・レポートをタスクとして選択します。
4. ランタイム・パラメーターに、特殊なタグ「./LoggedUser」を入力します。 これにより、カスタム・マッピングに基づいて結果が配布されます。
5. 「**今すぐ 1 回実行**」を押して監査プロセスを実行します



監査プロセスが完了すると、各受信者はマッピングに基づいて、それぞれの結果セットを受け取ります。

ユーザー

User01



Report Parameters used:

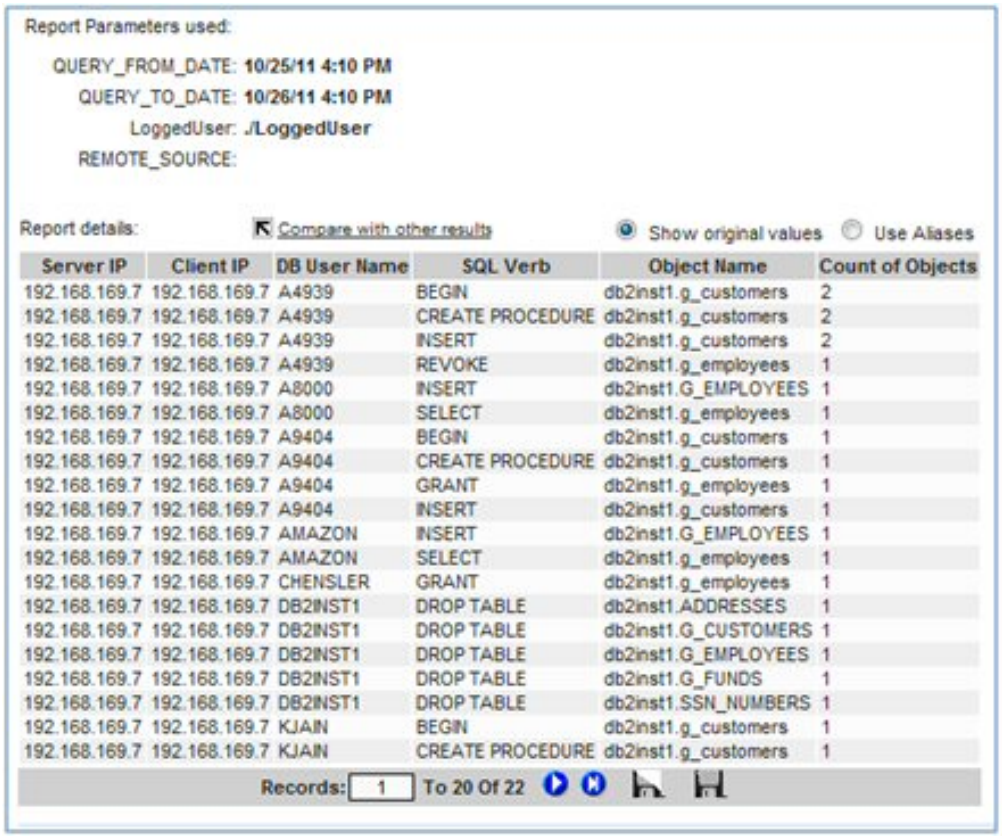
QUERY_FROM_DATE: **10/25/11 4:10 PM**
QUERY_TO_DATE: **10/26/11 4:10 PM**
LoggedUser: **./LoggedUser**
REMOTE_SOURCE:

Report details: Compare with other results — Show original values — Use Aliases

Server IP	Client IP	DB User Name	SQL Verb	Object Name	Count of Objects
192.168.169.7	192.168.169.7	A2840	CREATE VIEW	db2inst1.cc_numbers	1
192.168.169.7	192.168.169.7	A2840	SELECT	db2inst1.cc_numbers	1
192.168.169.7	192.168.169.7	ASEVIN	CREATE VIEW	db2inst1.cc_numbers	1
192.168.169.7	192.168.169.7	ASEVIN	SELECT	db2inst1.cc_numbers	1
192.168.169.7	192.168.169.7	DB2INST1	DROP TABLE	db2inst1.CCN	1
192.168.169.7	192.168.169.7	DB2INST1	DROP TABLE	db2inst1.CC_NUMBERS	1
192.168.169.7	192.168.169.7	KTRIMPE	SELECT	db2inst1.ccn	1
192.168.169.7	192.168.169.7	KTRIMPE	SELECT	db2inst1.cc_numbers	1
192.168.169.7	192.168.169.7	SCOTT	SELECT	db2inst1.ccn	1
192.168.169.7	192.168.169.7	SCOTT	SELECT	db2inst1.cc_numbers	1

Records:1 To 10 Of 10

User02



Report Parameters used:

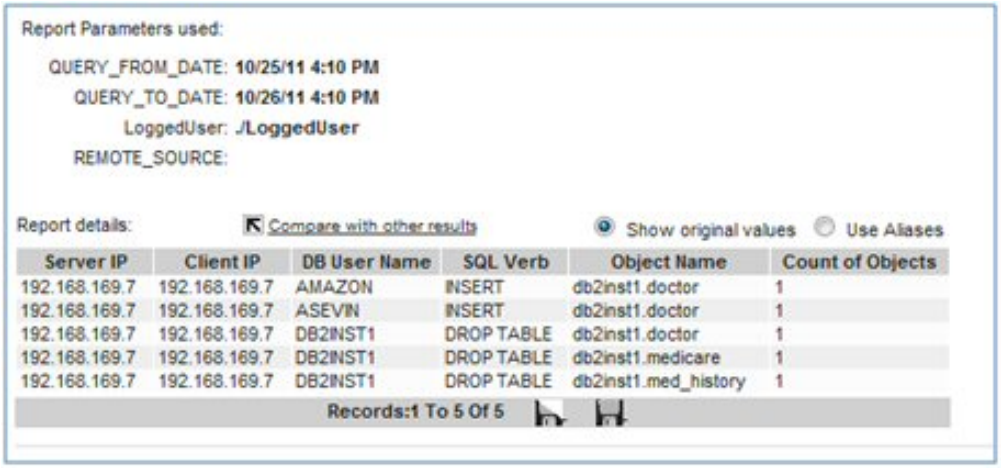
QUERY_FROM_DATE: **10/25/11 4:10 PM**
QUERY_TO_DATE: **10/26/11 4:10 PM**
LoggedUser: **./LoggedUser**
REMOTE_SOURCE:

Report details: Compare with other results — Show original values — Use Aliases

Server IP	Client IP	DB User Name	SQL Verb	Object Name	Count of Objects
192.168.169.7	192.168.169.7	A4939	BEGIN	db2inst1.g_customers	2
192.168.169.7	192.168.169.7	A4939	CREATE PROCEDURE	db2inst1.g_customers	2
192.168.169.7	192.168.169.7	A4939	INSERT	db2inst1.g_customers	2
192.168.169.7	192.168.169.7	A4939	REVOKE	db2inst1.g_employees	1
192.168.169.7	192.168.169.7	A8000	INSERT	db2inst1.G_EMPLOYEES	1
192.168.169.7	192.168.169.7	A8000	SELECT	db2inst1.g_employees	1
192.168.169.7	192.168.169.7	A9404	BEGIN	db2inst1.g_customers	1
192.168.169.7	192.168.169.7	A9404	CREATE PROCEDURE	db2inst1.g_customers	1
192.168.169.7	192.168.169.7	A9404	GRANT	db2inst1.g_employees	1
192.168.169.7	192.168.169.7	A9404	INSERT	db2inst1.g_customers	1
192.168.169.7	192.168.169.7	AMAZON	INSERT	db2inst1.G_EMPLOYEES	1
192.168.169.7	192.168.169.7	AMAZON	SELECT	db2inst1.g_employees	1
192.168.169.7	192.168.169.7	CHENSLER	GRANT	db2inst1.g_employees	1
192.168.169.7	192.168.169.7	DB2INST1	DROP TABLE	db2inst1.ADDRESSES	1
192.168.169.7	192.168.169.7	DB2INST1	DROP TABLE	db2inst1.G_CUSTOMERS	1
192.168.169.7	192.168.169.7	DB2INST1	DROP TABLE	db2inst1.G_EMPLOYEES	1
192.168.169.7	192.168.169.7	DB2INST1	DROP TABLE	db2inst1.G_FUNDS	1
192.168.169.7	192.168.169.7	DB2INST1	DROP TABLE	db2inst1.SSN_NUMBERS	1
192.168.169.7	192.168.169.7	KJAIN	BEGIN	db2inst1.g_customers	1
192.168.169.7	192.168.169.7	KJAIN	CREATE PROCEDURE	db2inst1.g_customers	1

Records: 1 To 20 Of 22

User03



親トピック: 監査プロセスの作成

監査プロセスの To-do リスト

このトピックでは、「監査プロセスの To-do リスト」と、これを開いて使用するために必要なステップについて説明します。

「監査プロセスの To-do リスト」は、以下を含むいくつかの方法で開くことができます。

- ページ・バナーの アイコンをクリックします。
- 「順守」 > 「ツールとビュー」 > 「監査プロセスの To-do リスト」にナビゲートします。
- E メール通知を受け取った場合は、「To-do リスト」リンクをクリックして To-do リストを開きます。または、「レポート」リンクをクリックして結果を開きます。 いずれの場合も、E メールには Guardium® システムにアクセス可能なロケーションからアクセスする必要があります。

以下のステップで、「監査プロセスの To-do リスト」の使用法を説明します。

1. 開く対象の To-Do リストを所有するユーザーを選択します。これは、ドロップダウン・メニューを開くか、「ユーザーの検索」をクリックすることによって行います。 リストが空の場合は通知を受けます。
2. 管理者は、任意の To-Do リスト項目にある任意のアクションを実行できます。 管理者が実行するすべてのアクションはログに記録され、管理者がユーザーの代わりにアクションを実行したことが示されます。
3. To-Do リスト項目ごとに選択可能な項目は、「表示」、「PDF 形式でダウンロード」および「表示された結果に署名」です。

   「PDF の内容」の選択項目には、「レポート」(現在の結果)、「差異」(1 つ前のレポートと新規レポートの間の差異)、 および「レポートと差異」(その両方) があります。

   注: 「PDF の内容」の選択内容は、PDF 添付ファイルと PDF エクスポート・ファイル の両方に適用されます。「差異」の結果は、このタスクの初回実行後にのみ適用されます。 前の結果がない場合に、前の結果との差異は存在しません。一度に比較可能な行の最大数は、5000 です。 結果行の数が最大数を超える場合、差異の結果にメッセージ「最初の 5000 行のみ比較」が表示されます。
4. セットの最新表示をするには、回転矢印のアイコンをクリックします。

注: 外部サーバーへのファイルの送信を、結果の To-Do リストへの追加や E メールの送信をせずに行う場合は、受信者を指定しない監査プロセスを 定義します。さらに、結果を To-Do リストに追加しないようにするために、「受信者の追加」セクションで「To-Do リスト」チェック・ボックスをクリアし、「受信者」セクションに受信者がある場合はすべて削除し、受信者の追加は行わないでください。

To-Do リストとデータ・レベル・セキュリティー

To-Do リストにはプルダウン・メニューがあり、そこから他のユーザーの To-Do リストを確認できます。 admin ロールを持つユーザーのプルダウン・メニューとは異なり、それ以外のユーザーのプルダウン・メニューには、データ・レベル・セキュリティー (DLS) 階層における、現行ユーザーの下位に属するユーザーのみが含まれます。 ユーザーが exempt ロールを持っている場合、プルダウン・メニューにはすべてのユーザーが表示されます。 admin ロールを持つユーザーのプルダウン・メニューには、すべてのユーザーが表示されます。

ユーザーが別のユーザーの結果にアクセスするとき、レポートに示されるデータは、 データ・レベル・セキュリティーおよび選択したユーザーのロールに従ってフィルターに掛けられます (例えば、 カスタム・ワークフローの場合、データは選択したユーザーのロールとそのロールに定義された状況に従ってフィルターに掛けられます)。

admin ロールを持つユーザーが階層内の下位に属するユーザーの結果にアクセスする場合、前の段落で説明したように動作します。 管理者が階層内の下位に属していないユーザーの結果にアクセスする場合、 管理者のデータ・レベル・セキュリティーを使用して結果が表示され、すべてのロールに関して表示されます。

イベントの状況の変更があったために、ユーザーの To-Do リストに今まで存在していなかった結果がそのリストに追加される場合には、ユーザーに対して E メールが送信されます。 E メールに含まれるのは通知とリンクのみであり、PDF は含まれません。

ユーザーが他のユーザーの To-Do リストに移動すると、DLS フィルタリングを決定しているユーザーを示すメッセージが表示されます。

親トピック: 監査プロセスの作成

監査およびレポート

Guardium は、収集したデータをドメイン・セットに編成します。各ドメインには、特定の関心領域 (データ・アクセス権、例外、ポリシー違反など) に関連する異なるタイプの情報が格納されます。

すべてのドメインおよびそれぞれのエンティティーについては、ドメイン、エンティティー、および属性で説明しています。

照会 - レポート・ビルダー内での各ドメインへのアクセスは、セキュリティー・ロールによって制御されます。照会の作成方法について詳しくは、『照会 - レポート・ビルダーの使用』を参照してください。

ユーザーは、標準のドメイン・セットに加えて、Guardium アプライアンスにアップロードされた情報を格納するカスタム・ドメインを定義できます。例えば、企業環境に、総称データベース・ユーザー名 (hr23455、qa4872 など) を実際の人名 (Paula Smith、John Doe など) に関連付ける表があるとします。 その表がアップロードされると、カスタム・ドメインから Guardium レポートに実名を表示できます。カスタム・ドメインの定義方法および使用方法について詳しくは、『外部データ相関』を参照してください。

親トピック: モニターおよび監査

外部データ相関

このトピックでは、エンタープライズ情報をインポートして既存の Guardium 内部データと共に使用するためのカスタム表とカスタム・ドメインの作成および管理について説明します。

貴重な情報が環境のさまざまなデータベースに貴重な情報が置かれていることがあります。データベース内の関連情報を Guardium によって収集されたデータと相互に関連付けると、監査レポートで大いに役に立つ可能性があります。Guardium では、エンタープライズ情報を既存の Guardium 内部データと結合するカスタム表を作成できます。その後、この情報の照会を、それがあたかも事前定義データであるかのように作成できます。

例えば、全社員、社員それぞれのデータベース・ユーザー名、各社員の所属部門 (開発部門、財務部門、営業部門、人事部門など) が含まれる表があるとします。この表とそのすべてのデータをアップロードすると、この表を Guardium の内部表と相互参照し、例えば、営業部門のどの社員が財務データベースにアクセスしているか (疑わしいアクティビティーになる可能性がある) を調べることができます。

データマートについて詳しくは、データマートを参照してください。

カスタム表

カスタム表には、アプライアンスで使用可能にする 1 つ以上の属性が含まれます。その表のデータを既存の表からアップロードすることにより、コード化された名前と実名を関連付けることができるようになります。

カスタム表を定義する前に、まず、既存のデータベース上の必要なデータが、サポートされるデータ・タイプであることを確認してください。 基礎となる JDBC ドライバーが以下の SQL タイプのいずれかとして受け取るデータ・タイプがサポートされます。INTEGER、BIGINT、SMALLINT、TINYINT、BIT、BOOLEAN、DECIMAL、DOUBLE、FLOAT、NUMERIC、REAL、CHAR、VARCHAR、DATE、TIME、TIMESTAMP。次の表は、カスタム表へのアップロードがサポートされるデータ・タイプとサポートされないデータ・タイプのいくつかを要約したものです。

カスタム表でサポートされるデータ・タイプとサポートされないデータ・タイプ

次の表を使用して、特定のデータベースでどのデータ・タイプがサポートされ、どのデータ・タイプがサポートされないかを確認してください。

表 1. カスタム表でサポートされるデータ・タイプとサポートされないデータ・タイプ

データベース	サポートされるデータ・タイプ	サポートされないデータ・タイプ
Oracle	float number char varchar2 date nchar nvarchar2	long clob raw nclob longraw bfile rowid urowid blob
DB2®	char varchar bigint integer smallint real double decimal date time timestamp	blob clob longvarchar datalink
Sybase	char nchar varchar nvarchar int smallint tinyint datetime smalldatetime	text binary varbinary image timestamp
MS SQL	bigint bit char datetime decimal float int money nchar numeric nvarchar real smalldatetime smallint tinyint smallmoney varchar unique identifier	text
Informix®	char nchar integer smallint decimal smallfloat float serial date money varchar nvarchar datetime	text
MY SQL	bigint decimal int mediumint smallint tinyint double float date datetime timestamp time year char binary enum set	longtext tinyblob tinytext blob text mediumblob mediumtext longblob longtext

注: 動的 SQL では blob 値 (値が 1K でも) をキャプチャーできますが、静的 SQL では同じサイズの blob 値をキャプチャーできません。

カスタム表のアーカイブおよびリストア

「カスタム表ビルダー」画面には、「パージ/アーカイブ」というボタンがあります。

「カスタム表データのパージ」画面には、「アーカイブ」用のチェック・ボックスがあります。このボックスにチェック・マークを付けると、カスタム表のデータが通常のデータ・アーカイブに組み込まれます。

このカスタム表データは、カスタム表の SQLGUARD_TIMESTAMP 列の日付に基づいてアーカイブされます。

カスタム表のデータは、コレクターまたはアグリゲーターからアーカイブ可能です。

コレクターからアーカイブされたカスタム表のデータは、ソース・コレクターと同じ中央マネージャーによって管理されているコレクターまたはアグリゲーターにリストアできます (メタデータが存在する必要があります)。

アグリゲーターからアーカイブされたカスタム表のデータは、ソース・アグリゲーターと同じ中央マネージャーによって管理されているアグリゲーターにリストアできます。

Guardium システムにリストアするアーカイブ・ファイルにメタデータがない場合、カスタム表のデータはリストアされません。

カスタム表の構造が、アーカイブ時とリストア時で、SQL エラーの原因となるような方法で変わっている場合 (列が削除されていたり、タイプが変わっていたりする場合)、統合/アーカイブ・アクティビティー・レポートに警告メッセージが表示され、データはリストアされません。

カスタム表がデフォルト・パージによってパージされるように設定されている場合、リストアされたデータは、リストア画面で指定した日数だけ保持されます。

カスタム表がアップロード時にデータを上書きするように設定されている場合、リストアされたデータは、アップロードの実行時に削除されます。

カスタム・ドメイン

カスタム・ドメインにはカスタム表が 1 つ以上含まれます。 表が複数含まれる場合は、カスタム・ドメインを定義する際に表間の関連を定義します。

カスタム照会

カスタム照会は、カスタム・ドメインに含まれるデータにアクセスします。 カスタム・ドメインに対する照会を作成するには、カスタム・クエリー・ビルダーを使用します。 その後、カスタム照会を他の照会のように使用して、レポートや監査タスクを生成したり、グループにデータを設定したり、別名を定義したりできます。

データベース・ライセンス・レポート

DB ライセンス・レポートでは、カスタム・ドメイン機能を使用して、選択したデータベース上の外部データと事前定義ライセンス・レポートの内部データとのリンクを作成します。これについては、トピック『内部データへの外部データのリンク』を参照してください。事前定義データベース・ライセンス・レポートの使用法について詳しくは、『データベース・ライセンス・レポート』を参照してください。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

カスタム表の作成

以下のいずれかにナビゲートして「カスタム表ビルダー」を開きます。

- 「順守」 > 「カスタム・レポート作成」 > 「カスタム表ビルダー」
- 「レポート」 > 「レポート構成ツール」 > 「カスタム表ビルダー」

表定義のアップロード

カスタム表の作成を行うには表定義をアップロードします。このためには、メタデータが定義されているデータベース・サーバーにあるこのメタデータにアクセスします。

注: Guardium® にアップロードされたカスタム表は、プロダクト・キーで使用可能になるオプション・コンポーネントです。これらのコンポーネントが使用可能になっていなければ、カスタム表に関する選択項目は、カスタム表ビルダーの選択項目として表示されません。

1. カスタム表ビルダー を開きます。
2. 「定義のアップロード」をクリックして、「表構造のインポート」パネルを開きます。項目を選択する必要はありません。
3. 「エンティティーの記述」フィールドに表の記述を入力します。これが、カスタム照会の作成時にこの表を参照するために使用する名前になります。
4. 「表名」フィールドに、表のデータベース表名を入力します。これが、ローカル・データベースに表を作成するために使用する名前になります。
5. 「SQL ステートメント」フィールドに、表に対する有効な SQL ステートメントを入力します。この SQL ステートメントによって返される結果セットの構造は、定義したカスタム表と同じでなければなりません。 例えば、my_table という名前の表のすべての列がカスタム表に含まれる場合は、select * from my_table と入力します。
   注:

   SQL ステートメントには、改行文字を使用しないでください。すべての列に明示的に名前を付ける必要があります (必要であれば列の別名を使用)。

6. 「データ・ソースの追加」をクリックして別のウィンドウにデータ・ソース・ファインダーを開きます。 データ・ソース・ファインダーで、外部データベースの場所を定義できるほか、このプロセスで後ほど表の定義と内容を取得する際に必要となる資格情報を定義できます。
7. データ・ソース・ファインダーを使用して、表定義のアップロード元のデータベースを特定します。
8. 「取得」をクリックして表定義をアップロードします。この操作によって SQL ステートメントが実行され、表構造が取得されます。 SQL 要求は Guardium システムから外部データベースに送信されます。アップロードされるのは定義だけであることに注意してください。データは後でアップロードできます。

表定義を手動で定義する

1. カスタム表ビルダー を開きます。
2. 「手動定義」をクリックして「エンティティーの定義」パネルを開きます。
3. 「エンティティーの記述」フィールドに表の記述を入力します。これが、カスタム照会の作成時にこの表を参照するために使用する名前になります。 エンティティーの記述に特殊文字 `¥$|&;'`"` は使用できません。
4. 「表名」フィールドに、表のデータベース表名を入力します。これが、ローカル・データベースに表を作成するために使用する名前になります。
5. 定義する表の列ごとに、以下のようにします。
   - 「列名」ボックスに名前を入力します。これが、データベース表の列の名前になります。
   - 「表示名」 ボックスに名前を入力します。これが、カスタム・ドメイン・ビルダーおよびカスタム・クエリー・ビルダーで属性を参照するために使用する名前になります。
   - データ・タイプ (テキスト、日付、整数、浮動小数点、またはタイム・スタンプ) を選択します。
   - テキスト属性の場合は、「サイズ」ボックスに最大文字数を入力します。(他のデータ・タイプでは「サイズ」ボックスは使用できません。)
   - 列の固有性を適用する場合は、「固有」 ボックスにチェック・マークを付けます。
   - 定義する属性はグループ・タイプと対応する場合は、そのグループ・タイプを「グループ・タイプ」 リストから選択します。
   - 「追加」をクリックして列を追加します。
6. 「エンティティー・キー」ドロップダウン・リストを使用して、エンティティー・キーとして使用する列を指定します。 エンティティー・キーは、カウントを選択するときにクエリー・ビルダーで使用されます。
7. 「追加」ボタンをクリックした後で列の削除や属性の変更などの追加変更を行った場合は、「適用」をクリックして変更を保存します。
8. 表のすべての列を追加したら、「完了」をクリックします。

表定義の変更

カスタム表の定義を変更すると、その表を使用する照会に基づいた既存のレポートが無効になる場合があります。 例えば、削除された属性やデータ・タイプが変更された属性を既存の照会が参照している場合があります。 カスタム表に変更を適用する際に、その表の属性を使用する照会が既に作成されている場合は、「照会リスト」パネルにそれらの照会が表示されます。 注:「変更」を使用して、インポートされている表構造を表示して確認することもできます。

1. カスタム表ビルダーを開きます。
2. 「エンティティー・ラベル」をクリックして強調表示することで、カスタム表を選択します。
3. 「変更」をクリックして、「エンティティーの変更」パネルを開きます。
4. 『表を手動で定義する』を参照してください。
5. カスタム表に変更を適用する際に、その表の属性に対する変更のために無効になる可能性のある照会がある場合は、「照会リスト」パネルにそれらの照会が表示されます。 「照会リスト」パネルを使用して、照会を選択して変更します。 直ちにすべての変更を行う必要はありません。いつでも戻って「無効な照会の検査」オプションを使用できます。

無効な照会

カスタム表の定義を変更すると、その表を使用する照会に基づいた既存のレポートが無効になる場合があります。 例えば、削除された属性やデータ・タイプが変更された属性を既存の照会が参照している場合があります。 表の変更処理後に無効な照会を検査することをお勧めします。

1. カスタム表ビルダーを開きます。
2. 「無効な照会」をクリックします。
3. 「照会リスト」パネルに照会が表示されます。「照会リスト」パネルを使用して、照会を選択して変更します。

カスタム表からデータをパージする

オンデマンドまたはスケジュール・ベースで、Guardium サーバー上のカスタム表からデータをパージできます。

1. カスタム表ビルダーを開きます。
2. カスタム表を、表の名前をクリックして強調表示することで選択します。
3. 「パージ」をクリックして「カスタム表データのパージ」パネルを開きます。
4. 今すぐパージするには、「すべてパージ」をクリックします。
   注:「今すぐ 1 回実行」パージは、保持データがないか RESTORED_DATA 表を調べます。「すべてパージ」は、保持データを検査することなく、削除されたすべてのレコードをパージします。
5. 「構成」パネルで、パージするデータの経過日数を、このパージ操作の日付より前の日数、週数、または月数として入力します。
6. パージのスケジュール設定操作を 1 回実行するには、「今すぐ 1 回実行」 をクリックします。
7. 「スケジュールの変更」をクリックして標準の「スケジュール定義」パネルを開き、パージ操作のスケジュールを設定します。
8. 「完了」をクリックしてパネルを閉じます。

カスタム表へのデータのアップロード

1. カスタム表ビルダーを開きます。
2. カスタム表を、表の名前をクリックして強調表示することで選択します。
3. 「データのアップロード」をクリックして「データのインポート」パネルを開きます。
4. 「SQL ステートメント」ボックスに、表に対する有効な SQL ステートメントを入力します。この SQL ステートメントによって返される結果セットの構造は、定義したカスタム表と同じでなければなりません。 例えば、my_table という名前の表のすべての列がカスタム表に含まれる場合は、`select * from my_table` と入力します。Guardium 内部の以下のフィールドを SQL ステートメントで使用できます。
   - ^FromDate?^ および ^ToDate?^。値はそれぞれ、前回のアップロード日と現在のアップロード日に相当します。
   - ^fromID^ および ^toID^。「ID 列名」と共に使用される場合に、それぞれ、前回のアップロードでの ID 列の最大値、および現在のアップロードの最大値から成ります。

   注: SQL ステートメントには、改行文字を使用しないでください。
5. 「ID 列名」の列名 (データ・ソース内で定義された表の列名) が使用されて ID によるトラッキングが可能であり、内部 Guardium フィールドの ^fromID^ および ^toID^ と共に使用されることを、必要に応じて指定します。
6. 「アップロード後の DML コマンド」ボックスに、データのアップロード後に実行する DML コマンド (update または delete SQL ステートメント) をセミコロンなしで入力します。 注: SQL ステートメントには、改行文字を使用しないでください。
7. 「デフォルト・パージの上書き (Overwrite Default Purge)」を構成するには、アップロード前にカスタム表のデータをパージする場合は、「アップロードごと」ラジオ・ボタンを選択します。そのデータ・ソースのデータをアップロード前にパージする場合は、「データ・ソースごと」ラジオ・ボタンを選択します。
8. 「デフォルトのカスタム表パージ・ジョブ」パージ・オブジェクト (初期デフォルトの経過日数は 60 日) の一部にするには、「デフォルト・パージ」ボタン (「カスタム・データのアップロード」画面内) にチェック・マークを付けます。 この表のパージ・スケジュールを追加するには、最初の「カスタム表ビルダー」ページに移動し、「カスタム表」を選択し、「パージ」をクリックして「カスタム表データのパージ」構成画面を開きます。
9. 以前のバージョンの Guardium から表をアップロードする場合のみ、「デフォルト・スケジュールを使用」ボックスにチェック・マークを付けます。 このチェック・ボックスは、中央マネージャー・ビューにのみ表示され、定義済みのカスタム表である「CM バッファー使用状況モニター」、「エンタープライズの「トラフィックなし」、「S-TAP® 変更」、および「S-TAP 情報」に対してのみ表示されます。
10. 「データ・ソースの追加」をクリックして別のウィンドウにデータ・ソース・ファインダーを開きます。 このウィンドウを使用して、表データのアップロード元のデータベースを 1 つ以上指定します。 複数のソースからアップロードするには、複数のデータ・ソースを追加します。 注:中央マネージャーの場合、「データのインポート」ページに、「デフォルト・ソースを含める」という読み取り専用のチェック・ボックスがあります。このチェック・ボックスにチェック・マークが付いている場合は、オンラインの登録済み管理対象ユニットすべてでデータのアップロードが繰り返されます。 注: データ・ソースを追加する場合、選択したデータ・ソースのユーザー名とパスワードを指定せずに実行されるようにアプリケーションのスケジュールを設定することはできません。
11. 「検査/修復」をクリックすることにより、カスタム表のスキーマをメタデータのスキーマと比較できます。 一元管理環境の場合: 一元管理環境では、中央マネージャー上にカスタム表定義があるので、ローカル (管理対象ユニット)データベース上にカスタム表が存在するとは限りません。 カスタム表がローカルに存在するかどうかを確認し、存在しない場合はそれを作成するには、「検査/修復」ボタンをクリックします。
12. 「データ・ソースの検査」をクリックして外部データベース接続をテストします。 確認画面が表示されます。
13. 「適用」をクリックします。
14. このカスタム表にデータをアップロードするには、以下のいずれかの操作を行います。
    - データを手動でアップロードする場合は、「今すぐ 1 回実行」をクリックします。
    - スケジュールを構成する場合は、「スケジュールの変更」をクリックします。

カスタム表の保守

カスタム表の作成手順 (前述) に従い、事前定義のカスタム表を選択する場合は、「メンテナンス」をクリックして、表エンジン・タイプと表索引を管理してください。Guardium 内部データベースに保管されるデータが MYSQL ベースの場合は、すべての事前定義カスタム・データベースに関する カスタム表/ライセンスの表エンジン・タイプ (InnoDB および MyISAM) が表示されます。MySQL データベースの表ストレージ・エンジンには InnoDB と MyISAM という 2 つの主要なタイプがあります。これらの MYSQL 表エンジン・タイプの主な違いは以下のとおりです。

- InnoDB の方が複雑で、MyISAM の方が単純です。
- データ保全性は、InnoDB の方が厳格で、MyISAM の方が緩やかです。
- 挿入と更新について InnoDB は行レベルのロックを実装し、MyISAM は表レベルのロックを実装します。
- InnoDB にはトランザクションがありますが、MyISAM にはありません。
- InnoDB には外部キーと関係制約がありますが、MyISAM にはありません。

注: 表の行番号が 1M より大きい場合、エンジン・タイプを変更することは許可されていません (選択項目がぼかし表示になります)。

「カスタム表の保守」メニュー内の他の選択項目として、「表索引の管理」があります。「挿入」をクリックして「表の索引の定義」を開きます。このポップアップ画面には、 カスタム・ドメイン上で結合条件として使用される列に基づき、 索引に追加すべき表の列が提示されます。列を選択して保存します。索引が作成 (または再作成) されます。

カスタム・データのアップロードのスケジュール設定

カスタム表定義が整うと、Guardium アプライアンス上のカスタム表にスケジュール・ベースでデータをアップロードできるようになります。

注: 新規インストールでは、エンタープライズ・レポートは自動的に開始されません。アップロード・スケジュールはカスタム表ごとに 1 つあります。Guardium アプライアンス上でカスタム表のために予約されているディスク・スペース総量は 4GB です。

1. カスタム表ビルダー を開きます。
2. 「エンティティー・ラベル」をクリックして強調表示することで、カスタム表を選択します。
3. 「データのアップロード」をクリックして「データのインポート」パネルを開きます。
4. デフォルト・スケジュールを使用してこの表をアップロードするには、「デフォルト・スケジュールを使用」チェック・ボックスにマークを付けます。 それ以外の場合、このカスタム表は独自のデータ・アップロード・スケジュールを使用します。
5. 「スケジュールの変更」 をクリックして標準の「スケジュール定義」パネルを開き、スケジュールを変更します。
6. 完了したら、「完了」をクリックします。

エンタープライズにより、他のジョブと同様にカスタム・アップロードについて報告されます。これらのジョブを有効にするには、次の 2 とおりの方法があります。

- カスタム表アップロード GUI を使用する (カスタム・アップロードに関するライセンスが必要)。
- 以下のように、CLI から GuardAPI を使用する。

```
grdapi add_schedule jobName=CustomTablePurgeJob_CM_SNIFFER_BUFFER_USAGE obGroup=customTableJobGroup Enterprise S-TAPs
Changed: grdapi add_schedule jobName=customTableDataUpload_106 jobGroup=customTableJobGroup CM Buffer Usage Monitor: grdapi
add_schedule jobName=customTableDataUpload_104 jobGroup=customTableJobGroup S-TAP Info: grdapi add_schedule
jobName=customTableDataUpload_80 jobGroup=customTableJobGroup
```

カスタム・ドメインの作成

カスタム表を 1 つ以上定義した後、カスタム・データを使用して照会およびレポート作成タスクを実行できるように、カスタム・ドメインを定義します。 収集された情報はドメインに編成され、ドメインごとに特定の関心領域 (データ・アクセス、例外、ポリシー違反など) に関連する異なるタイプの情報が含まれます。ドメインごとに別々のクエリー・ビルダー・ツールがあります。 カスタム・ドメインではユーザー定義のドメインが可能であり、Guardium アプライアンスにアップロードするデータの任意の表を定義できます。 『カスタム・ドメイン』を参照してください。これらのカスタム・ライセンス (特権) ドメインを使用するということは、ライセンス・レポートを使用するということです。ライセンス・レポートには、ユーザーとしてログインした場合にアクセスできます。 これらのレポートを表示するには、「ユーザー」タブの「データベース特権」に移動します。

注: DB ライセンス・ドメインは、プロダクト・キーにより有効になるオプションのコンポーネントです。これらのコンポーネントが使用可能になっていなければ、『カスタム・ドメイン』ヘルプ・トピックで示される選択項目は、カスタム・ドメイン・ビルダーの選択項目として表示されません。

1. 以下のいずれかにナビゲートして「カスタム・ドメイン・ビルダー」を開きます。
   - 「順守」 > 「カスタム・レポート作成」 > 「カスタム・ドメイン・ビルダー」
   - 「レポート」 > 「レポート構成ツール」 > 「カスタム・ドメイン・ビルダー」
   - 「順守」 > 「カスタム・レポート作成」 > 「カスタム・ドメイン・ビルダー」
2. 「ドメイン」をクリックして「ドメイン・ファインダー」パネルを開きます。
3. 「新規」をクリックして「カスタム表のドメイン」パネルを開きます。
4. ドメイン・ネームを入力します。ドメインに含めるカスタム表は通常 1 つなので、それと同じ名前をドメインに使用すると便利です。
5. 「使用可能エンティティー」ボックスに、定義されている (かつアクセス権限のある) カスタム表がすべてリストされます。 エンティティーを選択します。必要に応じて、(「フィルター」) ツールをクリックして「エンティティー・フィルター」を開き、リストするエンティティーのみを選択するための Like 値を入力し、「OK」をクリックします。フィルター・ウィンドウが閉じて「カスタム表のドメイン」パネルに戻ります。このパネルには、「使用可能エンティティー」ボックスにリストされた Like 値に一致するエンティティーのみが表示されます。 含めるエンティティーを選択します。
6. >> 矢印ボタンをクリックして、「使用可能エンティティー」リストで選択したエンティティーを「ドメイン・エンティティー」リストに移動します。
7. 既に表が 1 つ以上あるドメインにエンティティーを追加するには、以下の手順を行います。結合条件を使用して、エンティティー間の関係を定義する必要があります。追加エンティティーごとに、以下のようにします。
   注: データ・レベル・セキュリティーがオンになっている場合、カスタム・ドメインに追加された内部エンティティーは、フィルター・ポリシーが定義された異なるドメインに属することはできません。
   a. 「ドメイン・エンティティー」ボックスからエンティティーを選択します。そのエンティティーのすべての属性が、「ドメイン・エンティティー」ボックスのフィールド・ドロップダウン・リストで選択可能になります。そのリストから、結合演算で使用する属性を選択します。
   b. 「使用可能エンティティー」リストから追加するエンティティーを選択します。そのエンティティーのすべての属性が、「使用可能エンティティー」ボックスのフィールド・ドロップダウン・リストで選択可能になります。そのリストから、結合演算で使用する属性を選択します。
   c. 結合条件を等価 (例えば domainA.attributeB = domainC.attributeD) にする場合は、「=」(等価演算子) を選択します。 選択した属性を使用して結合条件を外部結合にする場合は、外部結合を選択します。
   d. 「フィールド・ペアの追加」をクリックします。「フィールド・ペアの追加」を使用して、この 2 つのエンティティーの属性のペアを、さらに結合条件に追加できます。
   e. 追加の結合演算があれば、ステップを繰り返します。

8. カスタム・ドメイン・エンティティーのタイム・スタンプ属性を選択します。
   注: タイム・スタンプが設定されたエンティティーを少なくとも 1 つ使用する必要があります。カスタム・ドメインを保存するためには、タイム・スタンプが必要のためです。
9. 「適用」をクリックします。

カスタム・ドメインの変更

この目的は、外部データと内部データのリンケージを作成することです。

1. カスタム・ドメイン・ビルダー を開きます。
2. コピーを作成するカスタム・ドメインを選択します。
3. 「変更」をクリックして「カスタム表のドメイン」パネルを開きます。
4. 『カスタム・ドメイン・ビルダーを開く』 および『内部データへの外部データのリンク』を参照してください。
5. 「適用」をクリックして、変更を保存します。

カスタム・ドメインの削除

1. カスタム・ドメイン・ビルダー を開きます。
2. コピーを作成するカスタム・ドメインを選択します。
3. 「ドメイン」をクリックして「ドメイン・ファインダー」パネルを開きます。
4. 「削除」をクリックしてカスタム・ドメインを削除します。

カスタム・ドメインのコピー作成

1. カスタム・ドメイン・ビルダー を開きます。
2. コピーを作成するドメインにあるカスタム表を選択します。
3. 「ドメイン」をクリックして「ドメイン・ファインダー」パネルを開きます。
4. 「コピー」をクリックして「カスタム表のドメイン」パネルを開きます。
5. ドメイン・ネームを変更して新しいドメインを反映した名前にします。
6. 『カスタム・ドメイン・ビルダーを開く』 および『内部データへの外部データのリンク』を参照してください。
7. 「適用」をクリックして、変更を保存します。

内部データへの外部データのリンク

この目的は、外部データと内部データのリンケージを作成することです。

1. カスタム・ドメイン・ビルダー を開きます。
2. 外部データのあるカスタム表を選択します。
3. 「ドメイン」をクリックして「ドメイン・ファインダー」パネルを開きます。
4. 「変更」をクリックして「カスタム表のドメイン」パネルを開きます。
5. 「使用可能エンティティー」の横にある「フィルター」アイコンをクリックします。
6. フィルターの「カスタム」ボックスのチェック・マークを外します。必要に応じて、エンティティー名をフィルターに掛けるための「LIKE」条件を入力し、「OK」をクリックします。
7. 外部データとリンクさせるエンティティーを、「使用可能エンティティー」 から選択します。
8. データを外部データと結合するために使用するフィールドを選択します。
9. 「ドメイン・エンティティー」で、外部データが含まれる表を強調表示します。
10. データを内部データと結合するために使用するフィールドを選択します。
11. 「フィールド・ペアの追加」をクリックして関係を追加します。
12. 二重矢印 >> をクリックして内部表を「ドメイン・エンティティー」リストに追加します。
13. 「適用」をクリックして、変更を保存します。

カスタム照会の処理

このセクションでは、カスタム・クエリー・レポート・ビルダーの開き方について説明します。照会の定義については、照会 - レポート・ビルダーの使用を参照してください。カスタム表が 1 つ以上含まれるカスタム・ドメインのデータに対する照会を作成するには、カスタム・クエリー・ビルダーを使用します。

1. 「順守」 > 「カスタム・レポート作成」 > 「カスタム・クエリー・ビルダー」にナビゲートして「カスタム・クエリー・ビルダー」を開きます。
2. リストからカスタム・ドメインを選択します。このドメインにある照会/レポートのリストが開きます。
3. 既存の照会の表示、変更、またはコピー作成を行うには、「照会名」リストから選択します。

InfoSphere Discovery との間の双方向インターフェース

IBM Guardium と InfoSphere® Discovery にはどちらにも、社会保障番号、クレジット・カード番号などの機密データを識別し、分類する機能があります。

IBM Guardium 製品のカスタマーは、双方向インターフェースを使用して、識別された機密データ情報を一方の製品から他方の製品に転送できます。一方の InfoSphere 製品に対して既に時間を注ぎ込んでいるカスタマーは、他方の InfoSphere 製品にその機密データ情報を転送できます。
注: IBM Guardium では、分類プロセスは、定期的に実行される継続プロセスです。InfoSphere Discovery では、分類は、通常 1 回実行されるディスカバリー・プロセスの一部です。

このデータは CSV ファイルを介して転送されます。

エクスポート/インポート手順の概要を以下に示します。

- Guardium からのエクスポート - 定義済みレポートを実行し (「Discovery への機密データのエクスポート」)、CSV ファイルとしてエクスポートします。
- Guardium へのインポート - CSV データ・ソースに対してカスタム表をロードします。このデータ・ソースに対してデフォルトのレポートを定義します。

以下の手順を行います。

- Guardium からのエクスポート
- IBM Guardium から InfoSphere Discovery に分類データをエクスポート

1. Guardium アプリケーションで admin ユーザーとして、「ツール」>「レポートのビルド」>「分類結果のトラッキング」>「レポートの選択」>「Discovery への機密データのエクスポート」に移動します。
   注: このレポートを UI ペインに追加します (これはデフォルトでは行われません)。
2. 「レポート結果」画面で「カスタマイズ」アイコンをクリックし、検索条件を指定して、Discovery に転送する分類結果データをフィルターに掛けます。
3. レポートを実行し、「レコードをすべてダウンロード」をクリックします。
4. CSV として保存し、このファイルを InfoSphere Discovery の指示に従い Discovery にインポートします。

Guardium

へのインポート

InfoSphere Discovery から IBM Guardium に分類データをインポート

1. InfoSphere Discovery の指示に基づき、InfoSphere Discovery から分類データを CSV としてエクスポートします。
2. 以下のいずれかにナビゲートして「カスタム表ビルダー」を開きます。
   - 「順守」 > 「カスタム・レポート作成」 > 「カスタム表ビルダー」
   - 「レポート」 > 「レポート構成ツール」 > 「カスタム表ビルダー」
3. 「分類データのインポート」を選択し、「データのアップロード」をクリックします。
4. 「データのアップロード」画面で、「データ・ソースの追加」をクリックし、「新規」をクリックし、新規データ・ソースとして Discovery からインポートする CSV ファイルを定義します (「データベース・タイプ」=「テキスト」)。
   注: または、Discovery データベースおよび分類結果データにアクセスする方法が判明している場合、Discovery データベースからデータを直接ロードできます。
5. データ・ソースとして CSV を定義した後、「データ・ソース・リスト」画面で 「追加」をクリックします。
6. 「データのアップロード」画面で、「データ・ソースの検査」、「適用」の順にクリックします。
7. 「今すぐ 1 回実行」をクリックして CSV からデータをロードします。
8. 「レポート・ビルダー」に移動し、分類データのインポート レポートを選択し、「ペインに追加」をクリックしてそのレポートをポータルに追加し、そのレポートに移動します。
9. レポートにアクセスし、「カスタマイズ」をクリックして開始日付/終了日付を設定し、レポートを実行します。

レポート結果には、InfoSphere Discovery からインポートされた分類データが含まれます。ダブルクリックして、このレポートに割り当てられている API を呼び出します。Discovery からインポートしたデータは以下の目的で使用できます。

- 結果セットに基づき新規データ・ソースを追加する。
- 機密データ・グループを追加/更新する。
- データ・ソースおよび機密データの詳細に基づきポリシー・ルールを追加する。
- プライバシー・セットを追加する。

CSV インターフェース・シグニチャー

以下の表に、IBM Guardium と InfoSphere Discovery の間の双方向転送で使用される CSV インターフェース・シグニチャーの例を示します。

表 2. CSV インターフェース・シグニチャー

インターフェース・シグニチャー	例
タイプ	Db2
ホスト	9.148.99.99
ポート	50001
dbName (Db2 または Oracle のスキーマ名、またはその他のデータベース名)	cis_schema
データ・ソース URL	
表名	MK_SCHED
列名	ID_PIN
分類名	SSN
ルールの記述	InfoSphere Discovery のすぐに使用可能なアルゴリズム
HitRate	70% - Guardium バージョンではエクスポートで使用不可 8.2
使用しきい値	60% - Guardium バージョンではエクスポートで使用不可 8.2

親トピック: モニターおよび監査

プライバシー・セット

プライバシー・セットとは、特別なモニターを行うために使用できる要素の集合です。

プライバシー・セットは、1 つ以上のオブジェクト-フィールド・ペアで構成されています。 例えば、employee 表の salary フィールド、 または salary history 表の全フィールドなど。 所定の時間フレーム内のこれらの要素に対する全アクセスをレポート可能です。

プライバシー・セットの処理については、いずれかのトピックを選択してください。

プライバシー・セット・ビルダーを開く

プライバシー・セット定義にアクセスするには、Guardium® ユーザー・アカウントに、 対象のプライバシー・セット定義にも割り当てられているセキュリティー・ロールを割り当てる必要があります。 ユーザーがアクセスできないプライバシー・セットは、プライバシー・セットのリストには表示されません。

1. 以下のいずれかにナビゲートして「プライバシー・セットの識別」パネルを開きます。
   - 「順守」 > 「ツールとビュー」 > 「プライバシー・セット・ビルダー」
   - 「ディスカバー」 > 「データベース・ディスカバリー」 > 「プライバシー・セット・ビルダー」
2. 以下のいずれかを実行します。
   - 「新規」ボタンをクリックして、 新規プライバシー・セットを定義します (『プライバシー・セットの作成』を参照)。
   - リストからプライバシー・セットを選択し、以下のボタンのいずれか 1 つをクリックします。
     - コピー -『プライバシー・セットのコピー作成』を参照してください。
     - 変更 - このボタンを使用して、定義を変更したり、その定義に基づいてレポートを実行したりします。 『プライバシー・セットの変更』または 『プライバシー・セット・レポートの実行』を参照してください。
     - 削除 -『プライバシー・セットの削除』を参照してください。

プライバシー・セットの作成

1. 以下のいずれかにナビゲートして「プライバシー・セットの識別」パネルを開きます。
   - 「順守」 > 「ツールとビュー」 > 「プライバシー・セット・ビルダー」
   - 「ディスカバー」 > 「データベース・ディスカバリー」 > 「プライバシー・セット・ビルダー」
2. 「新規」をクリックして、「プライバシー・セット定義」パネルを開きます。
3. 「プライバシー・セットの記述」ボックスで、 プライバシー・セットの固有の名前を入力します。 名前にはアポストロフィ文字を含めないでください。この名前が、「プライバシー・セットの識別」パネルに表示されます。
4. 「セキュリティー分類」ドロップダウン・リストで、 このプライバシー・セットのセキュリティー分類をオプションで選択します。
5. 「このプライバシー・セットの要素」 ペインで、組み込む要素ペアごとに以下を行います。
   - 「オブジェクト」ボックスにオブジェクト名を入力します。
   - 「フィールド」ボックスにフィールド名を入力するか、 「このオブジェクトの任意のフィールド」ボックスにマークを付けて、 指定したオブジェクト (上記) に含まれるすべてのフィールドを組み込みます。
   - 「新規オブジェクト – フィールド・ペアの追加」をクリックします。
6. すべての要素の追加後、「保存」をクリックします。
7. オプションで「ロール」ボタンをクリックして、ロールを追加します。
8. オプションで「コメント」ボタンをクリックして、コメントを追加します。

プライバシー・セットの変更

1. プライバシー・セット・ビルダーで、変更するプライバシー・セットを開きます。 『プライバシー・セット・ビルダーを開く』を参照してください。
2. 必要に応じて、プライバシー・セット定義に変更を加えます。 すべてのフィールドの説明は、『プライバシー・セットの作成』で参照してください。
3. 「保存」をクリックします。
4. 完了したら、「完了」をクリックします。

プライバシー・セットのコピー作成

1. プライバシー・セット・ビルダーで、コピーを作成するプライバシー・セットを開きます。 『プライバシー・セット・ビルダーを開く』を参照してください。
2. プライバシー・セットのコピーには COPY OF 選択したプライバシー・セットという名前が付けられます。 この名前をより意味のある名前に変更することを推奨します。 名前にはアポストロフィ文字を含めないでください。
3. 必要に応じて、プライバシー・セット定義に追加の変更を加えます。 すべてのフィールドの説明は、『プライバシー・セットの作成』で参照してください。
4. 「保存」をクリックします。
5. 完了したら、「完了」をクリックします。

プライバシー・セットの削除

監査プロセスを実行中の場合は、プライバシー・セットを削除できません。監査プロセスを停止してから手順を実行し、プライバシー・セットを削除してください。

1. 「プライバシー・セットの識別」パネルで、削除するプライバシー・セットを選択します。『プライバシー・セット・ビルダーを開く』を参照してください。
2. 「削除」をクリックして、アクションを確認します。
3. 「完了」をクリックします。

プライバシー・セットの実行

ここでは、プライバシー・セット・レポートをオンデマンドで実行する手順を説明します。 プライバシー・セット・レポートをスケジュールするには、コンプライアンス・ワークフローに組み込みます (『コンプライアンス・ワークフロー自動化』を参照)。

1. プライバシー・セット・ビルダーで、レポートのプライバシー・セットを開きます。 『プライバシー・セット・ビルダーを開く』を参照してください。
2. 「実行」をクリックします。
3. 「タスク・パラメーター」で、タスクの開始時刻および終了時刻を入力します。
4. 「アクセス詳細別レポート」または「アプリケーション・ユーザー別レポート」を選択して、結果の表示方法を指定します。 最初のオプションがデフォルトであり、その場合クライアント IP、 サーバー IP、サーバー (名)、サーバー・タイプ、データベース・プロトコル、 ソース・プログラム名、およびデータベース・ユーザー名の組み合わせごとにアクセス・カウントが表示されます。 「アプリケーション・ユーザー別レポート」を選択すると、レポートには (「データベース・ユーザー名」に続けて) アプリケーション・ユーザーの名前を持つ 個別の列が組み込まれ、さらに出力がそのアプリケーション・ユーザーによって修飾されます。
5. 「今すぐ 1 回実行」をクリックします。実行後のレポートは、別ウィンドウで表示されます。
6. 「完了」をクリックします。

親トピック: モニターおよび監査

カスタム・アラート

アラート・メッセージを配布する方法として、E メール、SNMP、syslog、またはユーザー作成の Java™ クラスが可能です。この最後のオプションを「カスタム・アラート」といいます。

アラートがトリガーされると、カスタム・アラート・クラスは状況に応じて適切な任意のアクションを実行できます (例えば Web ページの更新、電話番号へのテキスト・メッセージの送信など)。

カスタム・アラート・クラスを作成するには、まず技術サポートに連絡して、必要なインターフェース・ファイルを入手してください。以下のトピックでは、インターフェースの実装方法について説明します。 『カスタム・アラート・インターフェースの使用』、および例を示す『カスタム・アラート・クラスのサンプル』の各トピックを参照してください。

クラスがコンパイルされたら、Guardium® アプライアンスにアップロードする必要があります。『カスタム・クラスの管理』を参照してください。

カスタム・アラート・クラスのテストに関するガイドラインは、このトピック内の後方にある『カスタム・アラート・クラスのテスト』セクションを参照してください。

注: セキュリティー上の脆弱性のリスクを減らすために、信頼できないデータ・ソースのカスタム・コードを利用および実行しないでください。
注: 信頼できないソースのカスタム・コードを利用および実行しないでください
注: 信頼できないソースのデータを取得するカスタム・クラスを作成しないでください。

カスタム・アラート・インターフェースの使用

カスタム・アラート・クラスを com.guardium.custom パッケージの中に含めて、com.guardium.custom.alerts.CustomerDefinedAlertingIfc インターフェースを実装する必要があります (以下を参照)。

```
package com.guardium.custom
public class YourClassNameHere implements CustomerDefinedAlertingIfc {
       }
```

このインターフェースには、以下に説明する 5 つのメソッドが含まれています。

表 1. processAlert メソッド

メソッド 1	
記述	1 つのアラート・メッセージを処理します。
構文	public void processAlert (String message, Date timeStamp)
パラメーター	アラートによって生成されるメッセージを含む String。 アラート・メッセージの作成時間を示す java.util.Date。

表 2. getMessage メソッド

メソッド 2	
記述	アラート・メッセージを戻します。
構文	public String getMessage ()
パラメーター	アラート・メッセージを含む String。

表 3. getTimeStamp メソッド

メソッド 3	
記述	アラート・メッセージに関連付けられたタイム・スタンプを戻します。
構文	public Date getTimeStamp ()
パラメーター	アラート・メッセージの作成時間を示す java.util.Date。

表 4. setMessage メソッド

メソッド 4	
記述	アラート・メッセージを設定します。
構文	public void setMessage (String inMessage)
パラメーター	アラート・メッセージを含む String。

表 5. setTimeStamp メソッド

メソッド 5	
記述	アラート・メッセージに関連付けられるタイム・スタンプを設定します。
構文	public void setTimeStamp (Date inDate)
パラメーター	アラート・メッセージの作成時間を示す java.util.Date。

カスタム・アラート・クラスのサンプル

以下のサンプル・プログラムは、前のセクションで説明した 5 つのメソッドを実装しています。このプログラムの processAlert メソッドは、単にアラート・メッセージとタイム・スタンプをシステム・コンソールに書き込むだけです。

```
/*
 * Sample Custom Alerting Class
 *
 */
```

```
package com.guardium.custom;
import java.text.DateFormat;
import java.util.Date;
public class HandleAlerts implements CustomerDefinedAlertingIfc {
private String message = "";
private Date timeStamp = null;
public void processAlert(String message, Date timeStamp){
setMessage(message);
setTimeStamp(timeStamp);
System.out.println(getMessage() + " on " +
  DateFormat.getDateInstance(). format(getTimeStamp()));
}
    public void setMessage(String inMessage){
        message = inMessage;
    }
    public String getMessage(){
        return message;
    }
    public void setTimeStamp(Date inDate){
        timeStamp = inDate;
    }
    public Date getTimeStamp(){
        return timeStamp;
    }
}
```

カスタム・アラート・クラスのテスト

カスタム・アラート・クラスをコンパイルした後、以下のような手順に従ってテストします。

1. カスタム・クラスをアプライアンスにアップロードします。これは、管理者コンソールから実行する管理機能です。『カスタム・クラスの管理』を参照してください。
2. カスタム・アラート・クラスを使用する相関アラートまたはリアルタイム・アラートを定義します。どのアラート・タイプによってアラートが生成されるかに関わらず、カスタム・アラートの結果の比較対象となる 2 番目の通知タイプ (例えば E メール) を割り当てると、テストが簡単になります。
3. 以下のいずれかを行うことにより、環境を検査します。
    - 相関アラートの場合:
        - 異常検出ポーリング間隔がテストに適した設定になっていること、および異常検出が開始済みであることを確認します。ポーリング間隔が長すぎると (30 分以上)、照会の実行までの待機時間が長くなる可能性があります。
        - アラート機能のポーリング間隔がテストに適した設定になっていること、およびアラート機能が開始済みであることを確認します。
        - テスト対象のアラートにアクティブ状態のマークが付いていることを確認します。
    - リアルタイム・アラートの場合:
        - カスタム・アラート・アクション付きのルールを含むポリシーが、インストール済みポリシーであることを確認します。
        - 更新後のポリシーがインストールされた後、検査エンジンが再始動したことを確認します。
        - アラート機能のポーリング間隔がテストに適した設定になっていること、およびアラート機能が開始済みであることを確認します。
4. アラートをトリガーするために必要なアクションを実行します (例えば、多数のログイン失敗を生成します)。

親トピック: モニターおよび監査

未解析ログ処理

未解析ログ・オプションは、Guardium® アプライアンスが情報を即時に解析することなくログに記録できるようにする処理です。

こうすることで、処理リソースが節約され、より大量のトラフィックを処理できるようになります。後でコレクターまたはアグリゲーター・ユニットにおいて、そのデータを解析して Guardium の内部データベースに組み入れることができます。

未解析ログ処理に関連する「ポリシー定義による未解析ログ」と「スロットル・メカニズムによる未解析ログ」という 2 つの Guardium 機能があります。

スロットル・メカニズムによる未解析ログ - これは、CLI コマンド store alp_throttle 1 を実行することによって実装される機能です。GDM_FLAT_LOG 表に記録されたトラフィックを処理するために、リアルタイムの S-TAP トラフィックに適用されるのと同じポリシーが使用されます。

スロットル・メカニズムによる未解析ログでは、ポリシー・ビルダーで「未解析ログ」チェック・ボックスにチェック・マークを付けないでください。

ポリシー定義による未解析ログ - この機能を選択するには、「設定」>「ツールとビュー」の「ポリシー・ビルダー」メニューと、「管理」>「アクティビティー・モニター」の「未解析ログ処理」メニューを操作します。

注: 未解析ログに関するルールは、フィールド、オブジェクト、SQL 動詞 (コマンド)、オブジェクト/コマンド・グループ、およびオブジェクト/フィールド・グループを含んだポリシー・ルールでは機能しません。未解析ログ処理において、「未解析」とは構文ツリーが構築されていないことを意味します。 構文ツリーがない場合、フィールド、オブジェクト、および SQL 動詞は判別できません。

LOG FULL DETAILS、LOG FULL DETAILS PER SESSION、LOG FULL DETAILS VALUES、LOG FULL DETAILS VALUES PER SESSION、LOG MASKED DETAILS の各アクションは、フラット・ポリシーのルールでは機能しません。

「ポリシー・ビルダー」の「ポリシー定義」画面にリストされている「未解析ログ」チェック・ボックス・オプションを選択すると、次のようになります。

- データはリアルタイムでは解析されません。
- 未解析ログは、指定された「未解析ログ・リスト」レポートで確認できます。

1. 「管理」 > 「アクティビティー・モニター」 > 「未解析ログ処理」にナビゲートします。
2. 実行するアクティビティーを以下から選択します。
    - プロセス - 未解析ログ情報を内部データベースにマージします。
    - アーカイブ/統合/パージ - 未解析ログをアーカイブまたは統合し、さらにオプションでパージします。
    - パージのみ - 未解析ログ・データをパージします。
3. 「適用」をクリックして構成を保存します。

4. プロセス・アクティビティーの場合、オプションで、以下のいずれかを実行できます。
    - 「今すぐ 1 回実行」をクリックすると、直ちに未解析ログ情報を内部データベースにマージします。
    - 「スケジュールの変更」をクリックすると、このアクティビティーのスケジュールを定義できます。開始時刻、再開の頻度、および繰り返しの頻度を選択できます。「スケジュールの基準..」フィールドで、「曜日」または「月」を選択する必要があります。スケジューリングについて詳しくは、『スケジューリング』を参照してください。

親トピック: モニターおよび監査

データベース・ライセンス・レポート

ライセンス・レビューは、ユーザーがそれぞれの業務を行うために必要な特権のみを持っていることを検証および確認するプロセスです。

ユーザーの認証およびデータに対するロールに基づいたアクセス権の制限に加え、最も多くの特権を持つデータベース・ユーザーに対しても、定期的なライセンス・レビューを行う必要があります。このレビューは、ユーザーが自分の業務を行うのに必要な特権のみを持っていることを検証および確認するプロセスです。これは、データベース・ユーザー権限の認証レポート作成とも呼ばれます。

Guardium の事前定義データベース・ライセンス (特権) レポートを使用して、(例えば) システム特権を持つユーザーや、他のユーザーやロールにこれらの特権を付与したユーザーを確認します。データベース・ライセンス・レポートは、データベース・アクセスの変更をトラッキングしたり、使用されないまま残っているアカウントや誤って付与された特権によるセキュリティー・ホールが存在しないことを確認したりする監査員にとって重要なものです。

カスタム・データベース・ライセンス・レポートは、構成にかかる時間を減らし、Oracle、MYSQL、DB2®、SYBASE、SYBASE IQ、Informix®、MS SQL 2000/2005/2008、Netezza®、Teradata、PostgreSQL、および Db2 on z/OS の各データベースからのデータのアップロードおよびレポート作成を容易にするために作成されました。

Microsoft SQL Server データベースおよび Oracle データベースの場合、資格最適化を使用してこの情報にアクセスすることもできます。

以下の手順に従って、データベース・ユーザーおよびアクセス権の最新のスナップショットで作成された Guardium の事前定義データベース・ライセンス (特権) レポートを使用してください。

1. データ・ソース/データベースをアプライアンスに追加します (「順守」 > 「カスタム・レポート作成」 > 「カスタム・ドメイン・ビルダー」にナビゲートします)。
2. データ・ソースをライセンスに割り当てます (「順守」 > 「カスタム・レポート作成」 > 「カスタム表ビルダー」にナビゲートします)。使用するライセンスのカスタム表リストを選択します。「データのアップロード」をクリックします。「データのインポート」メニュー画面で、ライセンス・レポートにデータ・ソースを割り当てます。完了したら、「今すぐ 1 回実行」をクリックします。
3. ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

DB ライセンス・レポートでは、Guardium® のカスタム・ドメイン機能を使用して、選択したデータベース上の外部データと事前定義ライセンス・レポートの内部データとのリンクを作成します。カスタム・ドメイン・ビルダー、カスタム・クエリー・ビルダー、カスタム表ビルダーについての詳細情報は、『外部データ相関』を参照してください。

事前定義の資格レポートは、データベース・ライセンス・レポートにリストされています。

親トピック: モニターおよび監査

ユーザー識別

Guardium® には、データベース・トラフィックから実際のデータベース・ユーザーが識別できない場合に、アプリケーション・ユーザーを識別する方法がいくつか用意されています。

一部のデータベース・アプリケーションは、少数のデータベース・ユーザー・アカウントを使用または共有するように設計されています。 これらのアプリケーションでは、ユーザーがデータベース管理システムとは関係なく管理されます。 つまり、そのアプリケーションの外部からデータベース・トラフィックを監視する場合、任意の時点でデータベース接続を制御しているアプリケーション・ユーザーを判別することが困難になる可能性があります。 しかし、疑わしいデータベース・アクティビティーが発生した場合は、特定のアクションを、個人のグループが共有するアカウントではなく、特定の個人に関連付ける必要があります。 つまり、データベース・ユーザーだけではなく、アプリケーション・ユーザーを認識している必要があります。

Guardium には、データベース・トラフィックから実際のデータベース・ユーザーが識別できない場合に、アプリケーション・ユーザーを識別する方法がいくつか用意されています。

- アプリケーション・ユーザー・トランスレーションによるユーザーの識別 - 広く使用されている一部の商業アプリケーション (Oracle EBS、PeopleSoft、SAP、など) の場合、Guardium は自動的にユーザーを識別できます。
- API によるユーザーの識別 - アプリケーション・イベント API を使用すると、アプリケーション・ユーザーが接続の制御を行ったり解放したりした場合、またはその他関心のあるイベントが発生した場合に、Guardium にシグナルを通知することができます。(これは、ユーザーの識別以外にも使用できます。)
- ストアード・プロシージャーによるユーザーの識別 - 多くのアプリケーションでは、データベースのストアード・プロシージャーがアプリケーション・ユーザーの識別に使用されます。 このような場合、ユーザー情報は、通常、ストアード・プロシージャーのパラメーターから抽出できます。

社内では、使用するアプリケーションに応じて、ユーザーの識別に複数の方法を使用する必要が生じる場合があります。

- アプリケーション・ユーザー・トランスレーションによるユーザーの識別
  一部のアプリケーションは、データベース接続のプールを管理します。 そのような 3 層アーキテクチャーでは、プールされた接続はすべて単一の機能 ID を使用してデータベースにログインし、すべてのアプリケーション・ユーザーを内部で管理します。ユーザー・セッションでそのデータベースにアクセスする必要がある場合、プールから接続を獲得し、その接続を使用した後、リリースしてプールに返します。このような状況が発生した場合、Guardium では、アプリケーションがデータベースと対話する方法は確認できますが、特定のデータベース・アクションを特定のアプリケーション・ユーザーに結びつけることはできません。
- API によるユーザーの識別
  ユーザーを内部的に管理するアプリケーションでは、アプリケーション・ユーザーをトラフィックから識別できないものがあります。 この場合に、Guardium アプリケーション・イベント API を使用できます。
- ストアード・プロシージャーによるユーザーの識別
  既存の多くのアプリケーションでは、アプリケーション・ユーザーの識別に必要なすべての情報が既存のデータベース・トラフィックから (ストアード・プロシー

ジャー呼び出しから) 得られます。どの呼び出しを監視すべきか、どのパラメーターにユーザー名その他の必要情報が含まれるかを Guardium で認識しておくと、ユーザーを自動的に識別できます。

親トピック: モニターおよび監査

アプリケーション・ユーザー・トランスレーションによるユーザーの識別

一部のアプリケーションは、データベース接続のプールを管理します。 そのような 3 層アーキテクチャーでは、プールされた接続はすべて単一の機能 ID を使用してデータベースにログインし、すべてのアプリケーション・ユーザーを内部で管理します。ユーザー・セッションでそのデータベースにアクセスする必要がある場合、プールから接続を獲得し、その接続を使用した後、リリースしてプールに返します。このような状況が発生した場合、Guardium® では、アプリケーションがデータベースと対話する方法は確認できますが、特定のデータベース・アクションを特定のアプリケーション・ユーザーに結びつけることはできません。

Guardium には、一部の幅広く使用されるアプリケーションを対象として、アプリケーションからエンド・ユーザー情報を識別する標準装備サポートがあります。 それにより、データベース・アクティビティーをアプリケーション・エンド・ユーザーに関連付けることができます。

この機能を使用するには、以下の手順に従ってください。

1. アプリケーションのアプリケーション・ユーザー・トランスレーション構成を定義します。 『アプリケーション・ユーザー検出の構成』を参照してください。
2. そのアプリケーションに必要なすべての事前定義グループにデータを設定します。 『事前定義アプリケーション・グループへのデータの設定』を参照してください。
3. そのアプリケーションの特殊レポート用のすべてのポートレットを再生成して、そのポートレットをページに配置します。 『特殊アプリケーション・レポート・ポートレットの再生成』を参照してください。

選択的な監査証跡およびアプリケーション・ユーザー・トランスレーション

インストールしたデータ・アクセス・ポリシーで、選択的監査証跡機能を使用してログに記録されるデータ数を制限している場合、アプリケーション・ユーザー・トランスレーションに適用される 2 つの重要な考慮事項があります。

- ポリシーは、アプリケーション・ユーザー・トランスレーション・ルールに合致しない (例えば、 アプリケーション・サーバーが発信元ではない) すべてのトラフィックを無視します。
- そのセキュリティー・ポリシーのパターンに合致する SQL だけが、特殊アプリケーション・ユーザー・トランスレーション・レポートの対象になります。

アプリケーション・ユーザー検出の構成

1. 「保護」 > 「データベースの侵入検出」 > 「アプリケーション・ユーザー・トランスレーション」にナビゲートします。既存のアプリケーション・ユーザー・トランスレーション構成の詳細がページの上部に表示されます。
2. 「アプリケーション・コード」ボックスに固有のコードを入力して、新規のアプリケーション・ユーザー・トランスレーション構成の作成を開始します。
   注: 一元管理を行っている場合、管理マシンごとに異なるアプリケーション・コードを使用する必要があります。そうすることで、ユーザーごとに生成される別名が、相互に競合することを防ぎます。 (一元管理を行っている場合、すべての管理対象ユニットで共有される一式の別名セットがあります。
3. 「アプリケーション・タイプ」リストから、次のアプリケーション・タイプを選択します。
   - BO-WI - Business Objects / Web Intelligence
   - EBS - Oracle E-Business Suite
   - PeopleSoft
   - SAP Observed
   - SAP DB
   - SIEBEL Observed
   - SIEBEL DB
4. 「アプリケーション・バージョン」ボックスで、アプリケーション・バージョン番号 (例えば、11 など) を入力します。
5. 「データベース・タイプ」リストから、データベース・タイプを選択します。 選択したアプリケーション・タイプおよびバージョンで使用可能なタイプのみが表示されます。
   注: 「アプリケーション・タイプ」が EBS、SIEBEL DB、または SAP DB に設定されている場合、「データ・ソースの追加」ボタンをクリックして既存のデータ・ソースから選択するオプションがあります。データ・ソースは、構成中のアプリケーション・タイプに対してサポートされているデータベース・タイプの 1 つと一致する必要があります。
6. 「サーバー IP」ボックスで、アプリケーションがデータベースに接続するために使用する IP アドレスを入力します。
7. 「ポート」ボックスで、アプリケーションがデータベースに接続するために使用するポート番号を入力します。
8. 「インスタンス名」ボックスで、アプリケーションがデータベースに接続するために使用するインスタンス名を入力します。
9. 「データベース名」ボックスで、アプリケーションのデータベース名を入力します。 (一部のアプリケーションでのみ必須であり、それ以外では使用されません。)
10. 「アクティブ」ボックスにマークを付けて、ユーザー・トランスレーションを有効にします。 ユーザー定義の最初のインポートが完了するまで、変換は行われません。
11. データベースへのアクセス時に使用する Guardium の「ユーザー名」を入力します。データベースへのアクセス時に使用する Guardium の「パスワード」を入力します。
12. 責務 (例えば、管理など) とユーザー名を関連付ける場合は、「責務」ボックスにマークを付けます。 ユーザー名だけを記録する場合、「責務」ボックスはクリアします。 このボックスをクリアすると、ユーザーが実行したすべてのアクティビティーが、アクティビティー発生時の責務に関係なくグループ化されます。
    注: アプリケーション・タイプが EBS (データベース・タイプは Oracle) である場合、「接続先サーバー IP」および「接続先ユーザー名」という 2 つの追加の選択項目が表示されます。これらに値を指定すると、システムはその IP およびユーザー名を使用して接続し、責務およびユーザー名を検索します。
13. 「追加」ボタンをクリックして、アプリケーション・ユーザー・トランスレーション定義を保存します。
14. 続いて、『事前定義アプリケーション・グループへのデータの設定』 および『特殊アプリケーション・レポート・ポートレットの再生成』の手順に進みます。
15. 前のステップが完了したら、「管理」 > 「アクティビティー・モニター」 > 「検査エンジン」にナビゲートし、「検査エンジン構成」パネルで「検査エンジンの再始動」をクリックします。
16. ステップ 16 の 2 つの手順で指定したタスクの実行後、「アプリケーション・ユーザー・トランスレーション」に戻って「今すぐ 1 回実行」をクリックし、このアプリケーション (およびその他定義されたもの) のユーザー定義をインポートします。
17. データ・インポート操作が正常に処理されたことを検証 (ステップ 20 を参照) した後で、このパネルに戻って「スケジュールの変更」ボタンをクリックし、定期的に実行されるインポート操作を定義します。 ユーザー定義データのインポートは、使用環境に適した間隔で行われるようにスケジュールする必要があります。 新規アプリケーション・ユーザー名が使用可能になるまでの最大時間は、インポート操作の実行間隔の時間です。 スケジューラーの使用方法に関する説明については、『スケジューリング』を参照してください。

18. アプリケーション・ユーザー・トランスレーションのデータ・インポートは、事前定義レポート (例えば、SAP アプリケーション・アクセス) を調べることで確認できます。「レポート」 > 「レポート構成ツール」 > 「照会 - レポート・ビルダー」にナビゲートし、「SAP アプリケーション・アクセス」レポートを選択します。このレポートを再生成してペインに追加してから、日付範囲を比較的大きく (例えば、データについて過去 1 年間遡るなど) 設定してください。

注: アプリケーション・ユーザー・トランスレーション設定のインストール後に初めて「今すぐ 1 回実行」をクリックすると、調べている表の最終更新日付が検索されます。 その後は、新規データのみがインポートされます。この操作を行わないと、数十年分に相当するデータが不必要にインポートされて、多くの表やデータベースが満杯になる可能性があります。

事前定義アプリケーション・グループへのデータの設定

アプリケーション・ユーザー・トランスレーションが構成済みの場合、少なくとも 2 つの事前定義グループに、使用環境に固有の情報を設定する必要があります。 以下の表は、アプリケーション・タイプごとにデータを設定する必要があるグループを示しています。 グループへのデータの設定方法に関する説明については、『グループの概要』を参照してください。

アプリケーション	事前定義グループ	グループ・タイプ
EBS	EBS アプリケーション・サーバー	クライアント IP
	EBS データベース・サーバー	サーバー IP
PeopleSoft	PSFT アプリケーション・サーバー	クライアント IP
	PSFT データベース・サーバー	サーバー IP
	PeopleSoft オブジェクト	オブジェクト
Siebel	SIEBEL アプリケーション・サーバー	クライアント IP
	SIEBEL データベース・サーバー	サーバー IP
SAP	SAP アプリケーション・サーバー	クライアント IP
	SAP データベース・サーバー	サーバー IP
	SAP - PCI	オブジェクト

EBS アプリケーションの DB_USER パスワードを指定するのが好ましくない場合

特定の状況では、EBS トラフィックの変換に Oracle EBS の DB_USER を使用したくないユーザーも存在します。 このシナリオでは、Oracle EBS を設定して、アプリケーション・ユーザー・トランスレーションでトラフィックを変換するための以下の 2 つの選択肢があります。

- EBS が Oracle への通信に使用するユーザー名およびパスワード (多くの場合 APPS/$passwd) を指定します。
- DB_USER EBS が Oracle へのアクセスに使用するパスワードをユーザーが指定/入力したくない場合でも、アプリケーション・ユーザー・トランスレーションを行うことは可能ですが、その処理はさらに複雑になります。

1. 別名/ユーザー/責務を収集するためにデータベースにアクセスすることを許可する Oracle 用のログインを作成/選択します。 そのユーザーには、表 [APPLSYS.]FND_USER およびビュー FND_RESPONSIBILITY_VL (2 つの表 APPLSYS.FND_RESPONSIBILITY および APPLSYS.FND_RESPONSIBILITY_TL を組み合わせたもの) に対するアクセス権限が必要です。

```
(  CREATE VIEW FND_RESPONSIBILITY_VL AS SELECT /* $HEADER$ */ B.ROWID ROW_ID , B.WEB_HOST_NAME ,
B.WEB_AGENT_NAME , B.APPLICATION_ID , B.RESPONSIBILITY_ID ,
B.RESPONSIBILITY_KEY , B.LAST_UPDATE_DATE , B.LAST_UPDATED_BY ,
B.CREATION_DATE , B.CREATED_BY , B.LAST_UPDATE_LOGIN ,
B.DATA_GROUP_APPLICATION_ID , B.DATA_GROUP_ID , B.MENU_ID ,
B.START_DATE , B.END_DATE , B.GROUP_APPLICATION_ID ,
B.REQUEST_GROUP_ID , B.VERSION , T.RESPONSIBILITY_NAME ,
T.DESCRIPTION FROM FND_RESPONSIBILITY_TL T, FND_RESPONSIBILITY B
WHERE B.RESPONSIBILITY_ID = T.RESPONSIBILITY_ID
AND B.APPLICATION_ID = T.APPLICATION_ID
AND T.LANGUAGE = USERENV('LANG')  )
```

2. 次の SQL ステートメントを Guardium システムから直接実行します。select RESPONSIBILITY_ID, RESPONSIBILITY_NAME from FND_RESPONSIBILITY_VL order by RESPONSIBILITY_ID; および SELECT USER_ID, USER_NAME from FND_USER ORDER BY USER_ID;

   これら 2 つのステートメントを正常に実行するためのユーザーの設定が完了した後、2 つの異なるアプリケーション・ユーザー・トランスレーション項目が必要になります。 これらの各項目に含まれるサーバー IP、ポート、およびインスタンス名 (そしてアプリケーション・タイプおよびアプリケーション・サーバー・タイプとして選択された EBS および Oracle) は、同じでなければなりません。

   アプリケーション・コードが同じであるかどうかは関係ありません。 一方の項目には、EBS がデータベースへの接続に使用するユーザー名 (通常は APPS) が必要ですが、指定するパスワードは不正確 (ダミー) であってもかまいません。 もう一方の項目には、これらの表にアクセスするために作成されたユーザー名およびパスワードが必要です。

3. 「アクティブ」および「責務」を選択してこれら両方の項目を入力した後、「今すぐ 1 回実行」をクリックして EBS を開始または再始動します (トラフィックを調べる検査エンジン (S-TAP® またはネット) があることを想定しています)。 このようにすると、EBS トラフィックに関するデータの収集とそのデータに対する APPS ユーザー名の割り当てが行われるようになります。

Oracle EBS アプリケーション・ユーザーに必要な Oracle の特権

トランスレーション:

1. カスタム DB ユーザーに対して、以下の表における select を認可します。

   APPLSYS.FND_USER

APPLSYS.FND_RESPONSIBILITY

APPLSYS.FND_RESPONSIBILITY_TL

2. カスタム DB ユーザーに対して、APPLSYS.FND_USER における専用の同義語 FND_USER を作成します。
3. カスタム DB ユーザーに対して、FND_RESPONSIBILITY_VL という名のビューを作成します。 このビューは APPS ユーザーの下にあり、テンプレートとして使用できます。

SAP スタックのアプリケーション・ユーザー・トランスレーション対応の検証方法

IBM Guardium SAP アプリケーション・ユーザー・トランスレーションをサポートする場合、ABAP スタックと Java™ スタックでは方法が異なります。
注:

ABAP スタックと Java スタックではカーネルの仕様が異なります。

ABAP スタックと Java スタックのシステムでは、使用する表が異なります。

ABAP スタック

従来の ECC (Enterprise Core Components) SAP システムは ABAP コードで記述され、 主に SAP GUI を使用してアクセスしますが、Web アクセスも可能です。

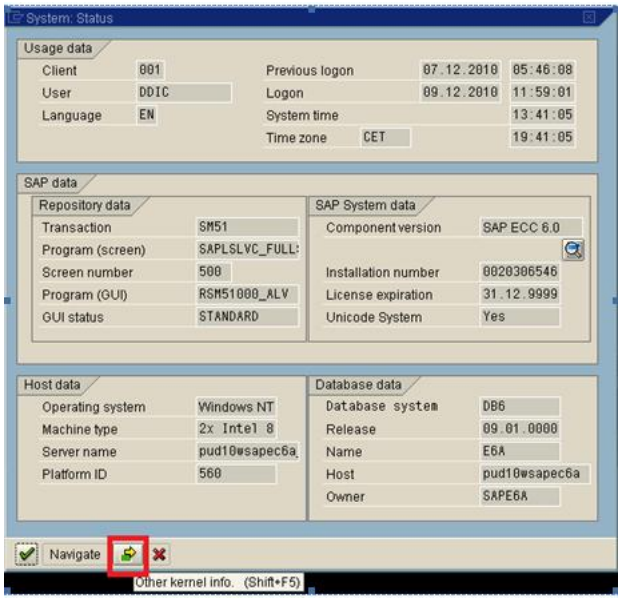
SAP ABAP システムは、従来の SAP データベースに対して直接 (読み取り/書き込み/更新) アクセスを行います。 データベースは非常に大規模で、すべての機密データが含まれます。 このような状況では、IBM Guardium が非常に役に立ちます。

SAP GUI (ABAP スタック) にアクセスすると次の画面が表示されます。

1-SAP GUI (ABAP スタック)

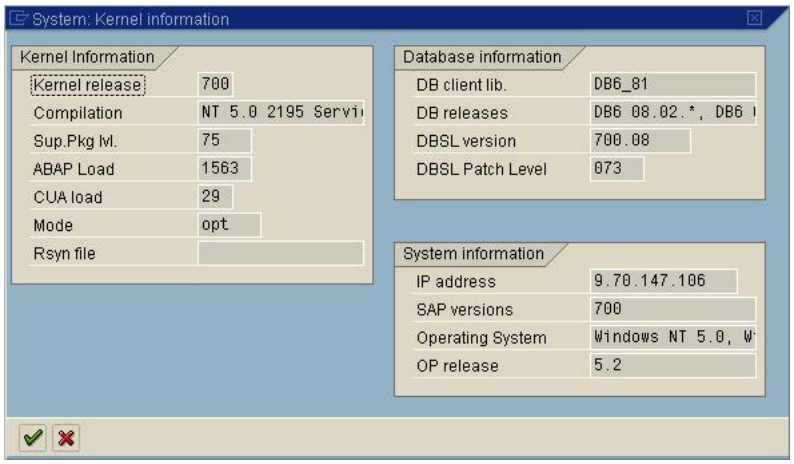
ABAP スタックの SAP カーネル・モジュールのアプリケーション・ユーザー・トランスレーション対応を検証するには、以下の手順を行います。

1. SAP にログインします。
2. 「システム」>「ステータス」に移動します。



2-システム状況 (ABAP スタック)

3. 「システム状況」画面で「その他のカーネル情報 (Other Kernel Info)」をクリックします。



3- システム・カーネル情報 (ABAP スタック)

この例では、カーネルは 700 です。

DB2® をバックエンドに持つ SAP は、SAP カーネル 640 でも使用可能ですが、ユーザーは DB6_DBSL_ACCOUNTING=1 を設定する必要があります (カーネル 700 以降では、この DB6_DBSL_ACCOUNTING 値はデフォルトで 1 です)。Oracle をバックエンドに持つ SAP では、カーネル 710 以降が必要です。

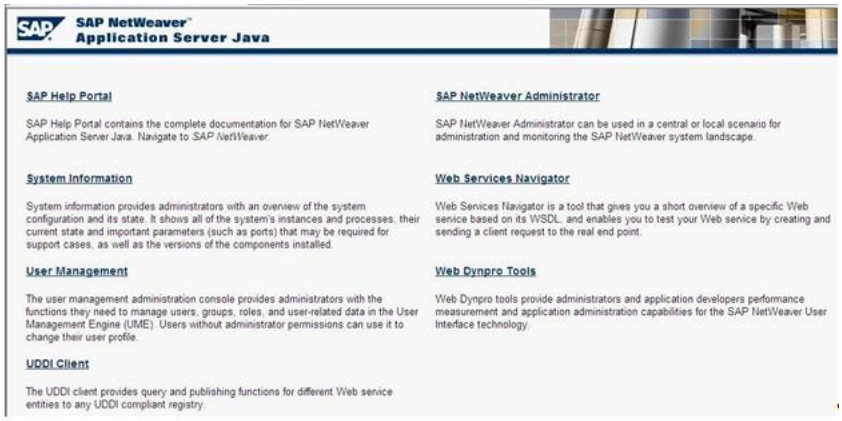
データは、アプリケーション・ユーザー・フィールドおよびアプリケーション・イベント文字列に入力されます。

Java スタック

SAP ポータル・システムは Java コードで記述されたフロントエンド Web アプリケーションで、事前に用意された照会を利用して SAP 関連の Web ページを表示します。

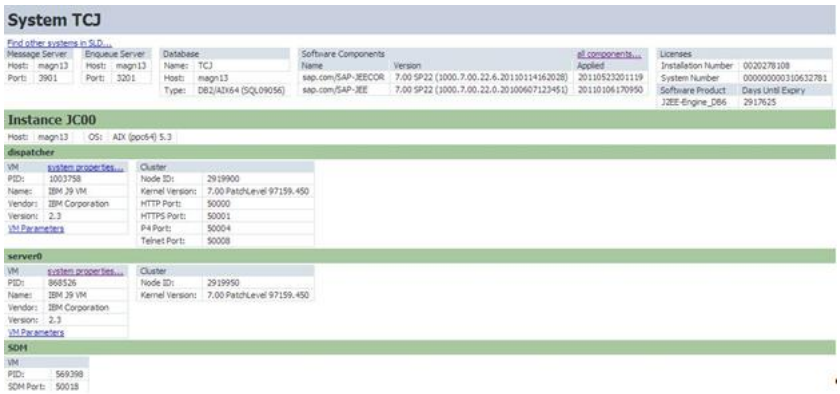
ポータル・システムは、Web ブラウザーからのみアクセス可能です。ポータル・システムのデータベースは非常に小規模であり、表領域はわずかしかありません。

SAP ポータル・システム (Java スタック) にアクセスすると次の画面が表示されます。



4-SAP ポータル・システム (Java スタック)

Java スタックの SAP カーネル・モジュールのアプリケーション・ユーザー・トランスレーション対応を検証するには、以下の手順を行います。 1.「システム情報」をクリックします。



5-システム TCJ (Java スタック)

この例では、SAP カーネル・バージョンは 7.00 です。

Db2 または Oracle 用の SAP では、7.02 以降のカーネルが必要です。

SAP は、ABAP スタックと同様に Java スタックにクライアント・プロパティーを設定します。

親トピック: ユーザー識別

API によるユーザーの識別

ユーザーを内部的に管理するアプリケーションでは、アプリケーション・ユーザーをトラフィックから識別できないものがあります。 この場合に、Guardium® アプリケーション・イベント API を使用できます。

アプリケーション・イベント API は、ユーザーが接続を獲得/リリースしたときや、他の対象とするイベントが発生した場合に、Guardium にシグナル通知するためにアプリケーション内から発行できる単純な呼び出しを用意しています。

注: Guardium セキュリティー・ポリシーで選択的な監査証跡が有効にされている場合、アプリケーション・ユーザー/アプリケーション・イベントの設定とクリアに使用したアプリケーション・イベント API コマンドはデフォルトで無視され、アプリケーション・ユーザー名/アプリケーション・イベントは記録されません。これらの項目を記録してレポートや例外で使用できるようにするには、「監査のみ」のルール・アクションを指定して、適切なコマンドを識別するためのポリシー・ルールを組み込んでください。

GuardAppUser - API によるユーザーの識別

アプリケーション・ユーザー名とアプリケーション・イベント名の両方に 2 つの事前定義トリガーを使用して、 GDM_CONSTRUCT_INSTANCE.APP_USER_NAME と GDM_APP_EVENT.* を設定します。

これらの事前定義トリガーは以下のとおりです。

- GuardAppEvent
- GuardAppUser

これらの各トリガーにより、トリガーが開始および停止されています。イベントには、Type、Username、StrValue、NumValue、および Date を設定するサブトリガーがあります。

Guardium システムは、AppUserName と AppEvent の詳細についての特殊な SELECT ステートメントを読み取ることができます。

形式は次のとおりです。

```
Select "action" [additional parameters] FROM [location].
```

表 1. アクション・オプション

構文	アクション
GuardAppUser:<username>	GDM_CONSTRUCT_INSTANCE.APP_USER_NAME を <username> に設定
GuardAppUserReleased	後続の照会のために APP_USER_NAME をクリア
GuardAppEvent:Start	GuardAppEvent を開始 (そして追加のパラメーターを検索)
GuardAppEvent:Released	GuardAppEvent を終了 (後続の照会のために情報をクリア)

表 2. 追加パラメーター (GDM_APP_EVENT の値を設定)

パラメーター	構文
GuardAppEventType: <event type string>	APP_EVENT_TYPE を <event type string> に設定
GuardAppEventUserName:<evntursname>	GDM_APP_EVENT.APP_USER_NAME を <evntursname> に設定
GuardAppEventStrValue:<strvalue>	EVENT_VALUE_STR を <strvalue> に設定
GuardAppEventNumValue:<num>	EVENT_VALUE_NUM を <num> に設定
GuardAppEventDateValue:<date>	EVENT_DATE を <date> に設定

SELECT ステートメントの例をいくつか示します。

Select guardappuser:tiberius from dual

Select guardappuserreleased from dual

Select GuardAppEvent:Start, GuardAppEventType:Event1, GuardAppEventUserName:Tiberius, GuardAppEventStrValue:abc, GuardAppEventNumValue:123, GuardAppEventDateValue:2016-01-26 15:55:28 from dual

Select GuardAppEvent:Released from dual

ステートメントの FROM 部分は、データベース・タイプにより異なります。

Oracle の場合: from DUAL

Db2 の場合: from SYSIBM.SYSDUMMY1

Informix の場合： from SYSTABLES

MS-SQL の場合： <blank>

Sybase の場合: <blank>

MySQL の場合: <blank> または from DUAL のいずれか

Guardium によるアプリケーション・ユーザー名および名前付きテンプレートの特定

Guardium を使用して「アプリケーション・ユーザー名」を取得するにはいくつかの方法があります。 Guardium には、データが受信された方法に基づいて、APP_USER_NAME フィールド値が保管される 2 つの Turbine 表があります。

GDM_CONSTRUCT_INSTANCE

GDM_APP_EVENT

Guardium 内の名前付きテンプレートの %%AppUserName パラメーター (「グローバル・プロファイル」メニューを参照) は、Turbine 表 (GDM_CONSTRUCT_INSTANCE) にマップされます。Guardium では、名前付きテンプレートでそのパラメーターを使用するには、GDM_CONSTRUCT_INSTANCE 表内の APP_USER_NAME に「アプリケーション・ユーザー」の値を取り込む必要があります。

アプリケーションの SQL コマンドの構文を以下のように変更します。

SELECT 'GuardAppUser:<value>'

これにより値は、正しい表に挿入され、名前付きテンプレートの %%AppUserName パラメーターは正しい値に置き換わります。

例

.....

select 'GuardAppUser:Db2_User' FROM SYSIBM.SYSDUMMY1 ;

select * from AppUser_DB2;

select 'GuardAppUserReleased' FROM SYSIBM.SYSDUMMY1 ;

select * from NoMoreUser_DB2;

.....

/var/log/messages ファイルで結果を検索します。

Jan 24 12:49:41 vx64 guard_sender[28274]: LEEF:1.0|IBM|Guardium|10.0|Alert per match|ruleID=20003|ruleDesc=Alert per match|severity=INFO|devTime=2016-01-24
11:50:39|serverType=DB2|classification=|category=|dbProtocolVersion=3.0|usrName=Db2_User|sourceProgram=DB2JCC_APPLICATION|start=1448383760000|dbUser=DB2INST1|dst=9.70.144.126|dstPort=50000|src=9.70.144.126|srcPort=58781|protocol=TCP|type=SQL_LANG|violationID=20|sql=select * from AppUser_DB2 FOR READ ONLY|error=

GuardAppUser によるアプリケーション・ユーザーの設定

この呼び出しを使用して、新しいアプリケーション・ユーザーが接続の制御を取得したことを示します。 指定されたアプリケーション・ユーザー名は、アクセス期間エンティティーのアプリケーション・ユーザー属性で使用可能になります。 このセッションについてこの時点以降、Guardium は、接続におけるすべてのアクティビティーがこのアプリケーション・ユーザーによるものと見なします。これは、 Guardium が別の GuardAppUser 呼び出しまたは GuardAppUserReleased 呼び出し (アプリケーション・ユーザー名をクリアする) を受け取るまで続きます。

他のイベントの発生をシグナル通知するには (必要に応じてイベント・タイプが定義可能)、後述のセクションの GuardAppEvent 呼び出しを使用します。

構文: SELECT 'GuardAppUser:user_name' FROM location

user_name は、アプリケーション・ユーザー名を含んだ文字列です。 この文字列は、アクセス期間エンティティーのアプリケーション・ユーザー属性値として使用できます。

FROM location は、Oracle、DB2®、または Informix® の場合にのみ使用します。 (他のデータベース・タイプでは省略。) これは、 次のように正確に入力する必要があります。

- Oracle: FROM DUAL
- DB2: FROM SYSIBM.SYSDUMMY1
- Informix: FROM SYSTABLES

GuardAppUserReleased によるアプリケーション・ユーザーのクリア

GuardAppUserReleased 呼び出しを使用して、現行ユーザーが接続の制御を解放したことをシグナル通知します。 Guardium によりアプリケーション・ユーザー名がクリアされます。この名前は、 別の GuardAppUser 呼び出しを受け取るまで、その接続で空の状態にされます。

構文: SELECT 'GuardAppUserReleased' FROM location

FROM location は、Oracle、Db2、または Informix の場合にのみ使用します。 (他のデータベース・タイプでは省略。) これは、 次のように正確に入力する必要があります。

- Oracle: FROM DUAL
- DB2: FROM SYSIBM.SYSDUMMY1
- Informix: FROM SYSTABLES

GuardAppEvent によるアプリケーション・イベントの設定

この呼び出しは、アプリケーション・イベントの発生をシグナル通知するもっと汎用の方法を提供します。 独自のイベント・タイプを定義して、イベント (イベントの開始時と終了時の両方) とともに格納するテキスト、数値、または日付値を指定できます。この呼び出しは、GuardAppUser 呼び出しとともに使用できます。Guardium は、接続におけるすべてのアクティビティーがこのアプリケーション・イベントのものとみなします。これは、別の GuardAppEvent:Start コマンドか GuardAppEvent:Released コマンドを受け取るまで続きます。

構文:

SELECT 'GuardAppEvent:Start|Released',

'GuardAppEventType:type',

'GuardAppEventUserName:name',

'GuardAppEventStrValue:string',

'GuardAppEventNumValue:number',

'GuardAppEventDateValue:date' FROM location

Start | Released - キーワード Start を使用してイベントが接続の制御を取得していることを示します。 または Released を使用してイベントが接続の制御を解放したことを示します。

type はイベント・タイプを示します。 これは、Login、Logout、Credit、Debit などの任意の文字列値にできます。アプリケーション・イベント・エンティティーで、この値は「イベント・タイプ」属性 (Start 呼び出しの場合) か「イベント・リリース・タイプ」属性 (Released 呼び出しの場合) に格納されます。

name は、このイベントに設定するユーザー名の値です。 アプリケーション・イベント・エンティティーで、この値は「イベント・ユーザー名」属性 (Start 呼び出しの場合) か「イベント・リリース・ユーザー名」属性 (Released 呼び出しの場合) に格納されます。

string は、このイベントに設定する任意の文字列値です。 例えば、Login イベントの場合に、アカウント名を指定することができます。 アプリケーション・イベント・エンティティーで、この値は「イベント値 (文字列)」属性 (Start 呼び出しの場合) か「イベント・リリース値 (文字列)」属性 (Released 呼び出しの場合) に格納されます。

number は、このイベントに設定する任意の数値です。 例えば、Credit イベントの場合に、取引金額を指定することができます。 アプリケーション・イベント・エンティティーで、この値は「イベント値 (数値)」属性 (Start 呼び出しの場合) か「イベント・リリース値 (数値)」属性 (Released 呼び出しの場合) に格納されます。

date は、このイベントのユーザー指定の日付とオプションの時刻です。 形式は、yyyy-mm-dd hh:mm:ss とする必要がありますが、時刻の部分 (hh:mm:ss) はオプションです。 これは、現在の日時にすることも、トラックされているトランザクションから取得することもできます。 アプリケーション・イベント・エンティティーで、この値は「イベントの日付」属性 (Start 呼び出しの場合) か「イベント・リリース日付」属性 (Released 呼び出しの場合) に格納されます。

FROM location は、Oracle、Db2、または Informix の場合にのみ使用します。 (他のデータベース・タイプでは省略。) 以下の例を参照してください。ただし、ダミー SQL では、ダミーの表名が許可されます。

- Oracle: FROM DUAL
- DB2: FROM SYSIBM.SYSDUMMY1
- Informix: FROM SYSTABLES

GuardAppEvent 呼び出しは、アプリケーション・イベント・エンティティーにデータを設定します (付録の『エンティティーおよび属性』セクションの『アプリケーション・イベント・エンティティー』を参照)。 Guardium の照会およびレポートを作成する際、アクセスのトラッキング・ドメインまたはポリシー違反ドメインからアプリケーション・イベント・エンティティーにアクセスができます。

GuardAppEvent 呼び出しを使用してアプリケーション・イベント・エンティティーの属性が設定されていなければ、これらの値は空となります。

2 つの日付属性について:

- 「イベントの日付」は、 GuardAppEvent 呼び出しを使用して設定するか、カスタム識別の手順 (以降のセクションで説明) から設定します。
- 「タイム・スタンプ」は、 Guardium がアプリケーション・イベント・エンティティーのインスタンスを格納した時刻です。

親トピック: ユーザー識別

ストアード・プロシージャーによるユーザーの識別

既存の多くのアプリケーションでは、アプリケーション・ユーザーの識別に必要なすべての情報が既存のデータベース・トラフィックから (ストアード・プロシージャー呼び出しから) 得られます。どの呼び出しを監視すべきか、どのパラメーターにユーザー名その他の必要情報が含まれるかを Guardium® で認識しておくと、ユーザーを自動的に識別できます。

最も単純なケースとしては、多数のプロパティー値 (そのうち 1 つはユーザー名) を設定する 1 つのストアード・プロシージャーがアプリケーションに含まれるといったものを想定できます。ユーザー名を設定する呼び出しは、例えば次のようになります。

```
set_application_property('user_name', 'JohnDoe');
```

(下記で説明する) カスタム・プロシージャー・マッピングでは、以下の動作を Guardium に指示できます。

- 最初のパラメーター値が user_name である set_application_property というストアード・プロシージャーを監視する。
- アプリケーション・ユーザーを、呼び出しにおける 2 番目のパラメーターの値 (例では JohnDoe) に設定する。

1 つのアプリケーションに複数のストアード・プロシージャーが含まれる場合もあり得ます。そのうち 1 つはアプリケーション・ユーザー・セッションを開始し、1 つはセッションを終了し、それ以外のストアード・プロシージャーはそのアプリケーションに特有の主要なイベントを通知するような場合です。Guardium のカスタム識別プロシージャー・メカニズムを使用すると、任意のアプリケーション・イベントをモニター対象としてトラッキングできます。

ユーザーを識別する方法はアプリケーションごとに異なる可能性があるため、各アプリケーション用に別個のカスタム識別プロシージャー・マッピングを定義する必要が生じることがあります。これを行うには、以下に要約する手順に従います。

カスタム識別プロシージャー・マッピングの定義

1. 「保護」 > 「データベースの侵入検出」 > 「カスタム ID プロシージャー」にナビゲートします。
2. 既存のマッピングを表示するには、表示対象のマップを含む行の「詳細情報」列アイコンの上にマウス・ポインターを置きます。
3. マッピングを追加するには、「追加」をクリックします。
4. 「カスタム・マップ名」ボックスで、このマッピングに使用する名前を入力します。
5. 「プロシージャー名」ボックスでは、情報を提供するデータベース・プロシージャーの名前を入力します。

6. アクション・リストから「設定」または「クリア」を選択します。これは、プロシージャー呼び出しによってアプリケーション値の設定または消去のどちらを実行するかを指示します。
7. 既存のストアード・プロシージャー呼び出しから、1 つまたは 2 つの条件の下でのみ、アプリケーション情報が得られる場合には、
   - 条件のロケーション・ボックスを使用して、検査対象となるストアード・プロシージャー呼び出しパラメーターを指定します
   - 対応する条件の値ボックスを使用して、他の 1 つ以上のパラメーターからアプリケーション情報を設定するためにマッチさせる必要のある値を指定します。
   - 例えば、多数の値 (その 1 つはユーザー名) を設定するために set_context という名前のストアード・プロシージャーがアプリケーションで使われるとします。3 つのパラメーター (アプリケーション名、プロパティー名、および値) がプロシージャーに渡されます。標準的な 3 つの呼び出しを示します。
     - set_context('publishing_application', 'role_name', 'manager');
     - set_context('publishing_application', 'user_name', 'jsmith');
     - set_context('publishing_application', 'company', 'guardium');
   - 例では、対象となる呼び出しの形式が 2 番目のステートメントによって表されています。検査すべきパラメーターは 2 番目のパラメーター (プロパティー名) であるため、「条件 1: ロケーション」ボックスに 2 を入力して、「条件 1: 値」ボックスに user_name を入力します。
   - さらに、呼び出しの別の形式でもユーザー名を設定する場合には、「条件 2: ロケーション」および「条件 2: 値」ボックスを使用できます。例えば、以下のような形式のプロシージャー呼び出しを使ってユーザー名が設定されることがあるとします。
     - set_context('admin_application', 'admin_name', 'wjones');
   - このプロシージャーを使ってアプリケーション・ユーザー名を設定するには、「条件 2: ロケーション」ボックスに 2 を入力して、「条件 2: 値」ボックスに admin_name を入力します。
     注: 2 つの条件を使用する場合、ユーザー名、または抽出される他の情報は、両方のタイプの呼び出しにおいて同じパラメーター位置でなければなりません。
8. 「クリア」アクションの場合:
   - アプリケーション・ユーザーのみをクリアするには、「アプリケーション・ユーザー名の位置」を 1 に設定して、その他すべての位置をゼロに設定します。
   - その他すべてのクリア・アクションでは、アプリケーション・イベントとアプリケーション・ユーザーがクリアされます。
9. 「設定」アクションの場合、「パラメーター位置」ペインを使用して、ストアード・プロシージャー・パラメーターと Guardium アプリケーション・イベント属性の間のマッピング関係を指定します。 最初のプロシージャー・パラメーターの番号は 1 になります。呼び出しによって設定されないすべての属性に対しては 0 (ゼロ、デフォルト) を使用します。アプリケーション・ユーザー名の位置 - この時点以降 (既に説明したようにリセットの時点まで)、データベース・アクティビティーに関連付けるアプリケーション・ユーザー名のパラメーター位置を入力します。イベント文字列値の位置 - イベントの文字列値のパラメーター位置を入力します (ログインの場合、例えばユーザーまたはアカウントの名前)。イベント数値の位置 - イベントの数値のパラメーター位置を入力します (トランザクションの場合、例えばドル金額)。イベント・タイプの位置 - イベント・タイプの名前のパラメーター位置を入力します (ログイン、ログアウト、クレジット要求など)。イベント日付の位置 - イベントの日付/時刻値のパラメーター位置を入力します。形式は yyyy-mm-dd hh:mm:ss でなければなりません。時間の部分 (hh:mm:ss) はオプションであり、省略した場合は 00:00:00 に設定されます。
   注: 「アプリケーション・ユーザー名の位置」が構成されている唯一のフィールドであり、このセッションに関連付けられている現行のアプリケーション・イベントがない場合は、新規イベントは作成されません。代わりに、アプリケーション・ユーザーは、「アクセス期間」の「アプリケーション・ユーザー」で使用できるようになります。このセッションに関連付けられている現行のアプリケーション・イベントがある場合は、アプリケーション・ユーザーは、「アクセス期間」の「アプリケーション・ユーザー」および新規アプリケーション・イベントで更新されます。
10. 「サーバー情報」ペインで、「サーバー・タイプ」リストから、データベース・サーバーの種類を選択します。「データベース・ユーザー名」ボックスで、データベース・ユーザー名を入力します。オプション:「データベース名」ボックスにデータベース名を入力します。省略した場合、すべてのデータベースがモニターされます。オプション: 1 つまたは複数のサーバーを指定します。サーバーを指定しない場合、すべてのサーバーがモニターされます。特定の 1 つのサーバーだけを選択するには、「サーバー IP」および「サーバー・ネットマスク」ボックスにサーバー IP アドレスとネットワーク・マスクを入力します。あるいは、複数サーバーから成るグループを選択するには、「サーバー IP グループ」リストからサーバー・グループを選択するか、「グループ」ボタンをクリックして新しいサーバー・グループを定義します。
11. 完了したら、「追加」ボタンをクリックしてマッピングをリストに追加します。

親トピック: ユーザー識別

値変更監査

値変更監査フィーチャーは、データベース表内の値の変更をトラッキングします。

値変更監査フィーチャーは、データベース表内の値の変更をトラッキングします。変更のトラッキング対象にする各表において、モニター対象にする SQL 値変更コマンド (INSERT、UPDATE、DELETE) を選択します。 モニター対象の表に対して値変更コマンドが実行されるたびに、before 値および after 値が収集されます。スケジュール・ベースで変更アクティビティーが Guardium® システムにアップロードされます。このシステムでは、すべてのレポート作成機能とアラート機能を使用できます。値変更監査フィーチャーを使用するには、以下の基本的なステップを実行します。

1. データベース・サーバー上に監査データベースを作成します。このデータベースは、値変更データを、Guardium システムにアップロードされるまで保管しておく場所です。監査データベースの作成を参照してください。
2. モニター対象にする表を指定し、それぞれの表に対して、変更を記録する値変更コマンド (INSERT、DELETE、UPDATE) を選択します。変更を記録するため、モニター対象の各表にはトリガーが作成され、そのトリガーが監査データベースに値変更データを書き込みます。監査データベースを (トリガー経由で) 更新できるようにするため、モニター対象の表に対して更新特権を持つすべてのユーザーに、監査データベースへの適切な特権が与えられます。これは、後にその表への更新特権を与えられるユーザーに影響を与えます (ステップ 4 を参照)。モニター・アクティビティーの定義方法に関する詳細な説明については、『モニター・アクティビティーの定義』を参照してください。
3. データベース・サーバーから Guardium システムへの、値変更データ転送のためのアップロードのスケジュールを設定します。『値変更アップロードのスケジュール設定』を参照してください。
4. 監査データベースへのアクセス権を保守します。 トリガーの作成後に新規のユーザーにそのトリガーのベースになっている表に対するアクセス権を与えられる場合があります。 そのユーザーがモニター対象の値変更コマンドを発行した場合、そのコマンドは失敗します。ユーザーが監査データベースを更新するための適切な特権を持っていないためです。 『特権ユーザー・リストの保守』を参照してください。
5. 管理コンソールから変更アクティビティーをモニターするか、または「値変更のトラッキング」照会ドメインを使用して、Guardium アプライアンス上にカスタム・レポートを作成します。『値変更レポートの作成』を参照してください。

モニター・アクティビティーの定義

監査データベースを定義した後、「値変更監査ビルダー」を使用して、モニター対象にする表を特定し、記録する変更のタイプ (INSERT、UPDATE、DELETE) を選択します。

1. 「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「値変更監査ビルダー」にナビゲートして「値変更監査ビルダー」を開きます。
2. 「データ・ソースの追加」をクリックして「データ・ソース・ファインダー」パネルを開きます。
3. 監査データベースを定義したデータ・ソースを選択します。監査データベースをまだ定義していない場合は、監査データベースの作成を参照してください。

4. 「追加」をクリックして、ファインダーを閉じ、選択したデータ・ソースを「値変更監査」パネルに追加します。
5. オプションで、「スキーマ所有者」と「オブジェクト名」のどちらかまたは両方を入力すると、モニター対象にする表を選択するときに表示される表の数を制限できます。「%」(パーセント) をワイルドカード文字として使用できます。 例えば、文字「a」で始まるすべての表を表示するように制限するには、「オブジェクト名」ボックスに「a%」と入力します。
6. 「モニターする表の選択」をクリックして、「データ監査の定義」パネルを開きます。
7. モニター対象にするそれぞれの表の「選択」ボックスにチェック・マークを付けます。
   注: ユーザー定義のデータ型を 1 つ以上含む表にはトリガーを定義できません。

   「定義されたトリガー」列は、この表で既にトリガーが定義されているかどうかを示します。「挿入の監査」、「削除の監査」、および「更新の監査」のチェック・ボックスは、トリガーでそのコマンドによる変更を記録するかどうかを指定します。

   「定義されたトリガー」列にマークが付いていない場合、表の「選択」チェック・ボックスにチェック・マークを付けると、自動的に 3 つすべての監査チェック・ボックス (「挿入の監査」、「削除の監査」、および「更新の監査」) にマークが付きます。 このうちの 1 つか 2 つのコマンドのモニターを行わない場合は、該当するチェック・ボックスをクリアします。

8. 「選択の追加」をクリックして、選択した表で使用するトリガーを定義します。実行されるアクションが通知されます。
9. 「OK」をクリックしてメッセージ・ボックスを閉じ、「データ監査の定義」パネルを再表示します。選択した表は選択されたままの状態になっており、これらの表の「トリガー定義済み」列にマークが付きます。注: 表にトリガーを定義するとすぐに、トリガーはアクティブになり、選択したコマンドによる変更が監査データベースに記録されるようになります。 トリガーの構成はすべて、データベース・サーバー上で実行される点が、他のほとんどの Guardium 構成と異なっています。つまり、ほとんどの構成は Guardium データベース上で定義され、その後別のタスクとしてアクティブ化/非アクティブ化されます。
10. 追加のアクションを定義するには、これらのステップを繰り返します。また、トリガーを削除するには、該当する「選択」チェック・ボックスにチェック・マークを付けて、「選択の削除」をクリックします。
11. すべての変更を完了したら、「完了」をクリックします。
    注: 「キャンセル」ボタンを使用しても、「選択の追加」ボタンまたは「選択の削除」ボタンを使用してトリガーに加えた変更は元に戻りません。

モニター・アクティビティー定義後の作業

データ・ソースに初めて値変更モニター・アクティビティーを追加した場合、このデータ・ソースのアップロードのスケジュールを設定する必要があります。これは、監査データベースは、記録されたデータを Guardium システムにアップロードした後に初めて空にされるからです。次のセクションを参照してください。

値変更アップロードのスケジュール設定

1. 「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「値変更監査ビルダー」にナビゲートして「値変更監査ビルダー」を開きます。
2. アップロードのスケジュールを設定する監査データ・ソースを選択して、「アップロードのスケジュール」pをクリックすると、汎用のタスク・スケジューラーが開きます。スケジュール設定の定義について詳しくは、共通ツール・ブックの『スケジュール』を参照してください。

特権ユーザー・リストの保守

値変更フィーチャーによってデータベース表にトリガーが追加されるときには、その時点でその表に更新権限を持つすべてのユーザーに対して、監査データベース表への更新権限が付与されます。 これが必要なのは、トリガーが監査データベースを更新して新しい値と古い値を書き込むためです。 新規のユーザーにモニター対象の表に対する更新権限を付与しても、そのユーザーが更新を試行したときに更新は許可されません。これは、そのユーザーが監査データベースを更新する権限を持っていないからです。この状況になった場合は、「値変更監査ビルダー」を使用して、監査データベースの特権ユーザー・リストを更新する必要があります。

監査データベースの特権ユーザー・リストを更新するとき、モニター対象データベースへのログインに使用するデータベース・ユーザー ID は、新規ユーザーの追加対象であるロールの作成者でなければなりません。ユーザー ID が異なっていると、そのロールのメンバーは使用できません。

1. 「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「値変更監査ビルダー」にナビゲートして「値変更監査ビルダー」を開きます。
2. 「データ・ソースの追加」をクリックして「データ・ソース・ファインダー」パネルを開き、リストから該当するデータ・ソースを選択して「追加」をクリックします。
3. 「監査表特権ユーザーの更新」をクリックします。 監査データベース表を更新するためにトリガーを実行できるすべてのユーザーのアクセス権が更新され、その操作が完了すると通知されます。
4. 「OK」をクリックしてメッセージ・ボックスを閉じます。

値変更レポートの作成

デフォルトの「変更された値」レポートから値変更データを表示できます。あるいは、「値変更のトラッキング」ドメインを使用してカスタム・レポートを作成することもできます。 デフォルトでは、「値変更のトラッキング」ドメインは、admin ロールを持つユーザーに制限されています。

「変更された値」デフォルト・レポート

「レポート」 > 「リアルタイム Guardium 運用レポート」 > 「変更された値」にナビゲートすることで、デフォルトの「変更された値」レポートを使用できます。

「変更された値」レポートの主要なエンティティーは「変更された列」エンティティーです。 ほとんどの場合、それぞれの監査アクション (INSERT、UPDATE、DELETE) に関して検出されたそれぞれの列変更ごとに、別のレポート行が表示されます。 ただし、MS SQL Server と Sybase では、モニター対象の表に主キーがない場合、1 つの変更に対して 2 つの行が表示されます。つまり、古い値と新しい値が別の行に表示されます。

親トピック: モニターおよび監査

監査データベースの作成

監査データベースを作成して値変更モニター・アクティビティーを実行します。

監査データベースを作成して値変更モニター・アクティビティーを実行するには、以下の操作を行うための適切な権限を持つユーザー・アカウントが必要です。

- サーバー上へのデータベースの作成
- サーバー上へのデータベース・ユーザー・アカウントの作成

モニター対象の各データベースへログインして、モニター対象の各データベースで表とトリガーを作成します。

Informix または Sybase で監査データベースを定義する前に行う作業

Informix® および Sybase (トリガーをサポートしていないため Sybase IQ は除外) では、データベース・サーバーが稼働するオペレーティング・システムに応じて、監査データベースを定義する前に以下の手順のいずれかを行う必要があります。

Informix の設定 - 新規データベース・スペースの配置または作成

このトピックは Informix (9.4 またはそれ以降) に適用されます。 Informix では、デフォルトのルート・データベース・スペースである root_dbs の使用を避けるよう強くお勧めします。 このスペースは、ドロップすることも、サイズを削減することもできません。

定義済みの他のデータベース・スペースを使用するか、以下のいずれかの手順 (オペレーティング・システムによって異なります) を実行してデータベース・スペースを新規作成する必要があります。

Informix - Windows Server 上での Informix データベース・スペースの作成

この手順は Guardium® GUI 外部で実行します。また、Informix バージョン 9.4 以降に適用されます。

1. データベース・サーバーがオンラインで listen 中であることを確認します。
2. guardium_dbs_dat.000 という名前のゼロ・バイトのファイルを C:¥IFMXDATA¥server-name ディレクトリーに作成します (sever-name は Informix サーバー名またはサービス名です)。 これは、空のテキスト・ファイルを保存してからそのファイルを名前変更し、接尾部の txt を 000 に置き換えることによって行えます。
3. 以下のディレクトリーを作業ディレクトリーにします。

   C:¥Program Files¥Informix¥bin

4. 以下のコマンドを実行します。

   ```
   C:¥Program Files¥Informix¥bin>onspaces
   -c -d guardium_dbs -p C:¥IFMXDATA¥server-name¥guardium_dbs_dat.000
   -o 0 -s 150000
   ```

   ファイルの作成が成功すると、以下のメッセージが表示されます。

   ```
   Verifying physical disk space, please wait ...
   Space successfully added.
   ** WARNING **  A level 0 archive of Root DBSpace will need to be done.
   ```

5. Informix サーバーを再始動し、適切なツール (例えば、Aqua Data Studio リモート・クライアント) を使用して、作成した guardium_dbs という名前のスペースへの接続と検査を行います。 最初の接続試行では、サーバーが静止モードで実行されていることに関するメッセージが表示されて失敗する場合があります。 これが発生した場合は、少なくともさらに 2 回再接続を試行します。これにより、動作するようになるはずです。
6. guardium_dbs データベース・スペースが作成されたことを検査するには、Aqua Data Studio を使用して、Storage の下を確認します。

Informix - Unix Server 上での Informix データベース・スペースの作成

この手順は Guardium GUI 外部で実行します。また、Informix バージョン 9.4 以降に適用されます。

1. コマンド行ウィンドウで以下のコマンドを実行します。

   ```
   su - informix
   cd demo/server
   vi guardium_dbs
   ```

2. テキストを追加せずに、空の guardium_dbs ファイルを保存します。
3. 以下のコマンドを入力します。

   ```
   chmod 660 guardium_dbs
   cd ../../bin
   onspaces -c -d guardium_dbs -p /home/informix10/demo/server/guardium_dbs -o 0 -s 100000
   ```

Sybase の設定 - ディスクの初期化

このトピックは Sybase サーバーのみに適用されます (Sybase IQ は該当しません。トリガーをサポートしていないからです)。 データベース・サーバーが稼働するオペレーティング・システムによって異なりますが、ディスクを初期化するために以下の手順のいずれかを行う必要があります。

Sybase - Windows Sybase Server でのディスクの初期化

1. Guardium 監査データベース guardium_audit を作成するサーバーに接続します。
2. C: ドライブに guardium_audit という名前のフォルダーを作成します。
3. データベースに接続します。
4. 以下のコマンドを実行します。

   ```
   use master
   go
   disk init name="guardium_auditdev", size=8192
   go
   disk init name="guardium_auditlog",
   physname="c:/guardium_audit/guardium_auditlog", size=8192
   go
   ```

Sybase - Unix Sybase Server でのディスクの初期化

1. データベースに接続します。
2. 以下のステートメントを実行します。

```
use master
go
disk init name = 'guardium_auditdev', physname
 ='/home/sybase/data/guardium_auditdev' , size = 8192
go
disk init name = 'guardium_auditlog', physname
 ='/home/sybase/data/guardium_auditlog' , size = 8192
go
```

データベースの作成

Informix または Sybase データベースの場合は、この手順を実行する前に、必ず準備タスクを実行しておいてください。

1. 「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「値変更監査データベースの作成」にナビゲートして「値変更データベース・ビルダー」を開きます。
2. 「データ・ソースの追加」をクリックして、「データ・ソース・ファインダー」パネルを開きます。 「値変更監査」アプリケーションから定義したデータ・ソースには、「値のモニター」というラベルが付けられています。 他のアプリケーション用に定義されたデータ・ソースには異なるラベル (例えば、Listener、DBanalyzer など) が付けられています。そのようなデータ・ソースには、「値変更監査」アプリケーションのための適切なデータベース・アクセス権のセットがない可能性があります。「値変更監査」アプリケーションではデータベース管理者権限を持つユーザー・アカウントが必要です。 適切なデータ・ソースが使用できない場合、「新規」ボタンをクリックして、モニター対象のデータベースに新しいデータ・ソースを定義します (データ・ソースの定義に関する詳細については、共通ツール・ブックの『データ・ソース』を参照してください。
   注: このデータベース・サーバーに GUARDIUM_AUDIT データベースを既に作成済みの場合、もう 1 つ作成することはできません。新規作成する前に、GUARDIUM_AUDIT データベース/ユーザーをドロップする必要があります。
3. 管理者アカウントを使用するデータ・ソースを選択して「追加」をクリックすると、それが「値変更監査データベースの作成」パネルの「データ・ソース」ペインに追加されます。
4. 「監査データ・ソース名」に入力します。 これは、後でモニター・タスクの定義とデータのアップロードを行う際にこのデータ・ソースを識別するために使用される名前になります。 この名前と「データ・ソース」パネルにあるデータ・ソースの名前を混同しないようにしてください。
5. オプションで、「データ・ソースの共有」ボックスにマークを付けて、このデータ・ソースを他のアプリケーション (例えば、分類) と共有します。 デフォルトではデータ・ソースの共有はしません。 このタイプのデータ・ソースでは管理者特権が必要です。それで、このデータ・ソースを他のアプリケーションとは共有しないことにするかもしれません。
   注: データ・ソースを他のユーザーと共有するには、そのデータ・ソースにセキュリティー・ロールを割り当てます。
6. DB2® 以外のすべてのタイプのデータベースでは、「監査構成」ペインに追加のフィールドがあります。 すべてのフィールドが必須です。以下の表を参照して、適切な値を入力してください。

表 1. 「監査構成」の追加フィールドの表

データベース・タイプ	フィールド: 説明
Informix	**データベース・スペース**: 使用する既存のデータベース・スペースの名前を入力するか、または監査データベース用に作成したデータベース・スペースの名前 (前述の例では guardium_dbs) を入力します。 これをブランクのままにした場合、デフォルトの root_dbs space が使用されます。これは推奨しません。
MS SQL Server	**監査ユーザー名**: 監査データベースにアクセスするときに使用する新しいデータベース・ユーザー名を入力します。 このユーザーには sysadmin ロールが付与されます。 **監査パスワード**: パスワードを入力します。 データ・ソースが MSSQL サーバーの場合は、「値変更監査データベースの作成」メニュー画面に追加の選択項目が表示されます。この追加の選択項目は、データ・ソースが MSSQL サーバーの場合にしか表示されません。 互換モード: 選択項目は「デフォルト」と「MSSQL 2000」です。表のモニター時に使用する互換モードをプロセッサーに指示します。 MS SQL Server の互換モードを表示するには、GuardAPI コマンド grdapi list_compatibility_modes を使用してください。
Oracle	**監査パスワード**: システム・ユーザーのパスワードを入力します。これは監査データベースへのアクセスに使用するデータベース・アカウントになります。 **デフォルト表スペース**: デフォルトの表スペースの名前を入力します。 **一時表スペース**: 一時表スペースの名前を入力します。
Sybase	**監査ユーザー名**: 監査データベースにアクセスするときに使用する新しいデータベース・ユーザー名を入力します。 このユーザーには sa_role が付与されます。 **監査パスワード**: パスワードを入力します。 **データ・デバイス名**: 監査データベースで使用するディスクを初期化する際に使用したのと同じデータ・デバイス名を入力します (前述したディスク初期化手順の場合、guardium_auditdev)。 **ログ・デバイス名**: 監査データベースで使用するディスクを初期化する際に使用したのと同じログ・デバイス名を入力します (前述したディスク初期化手順の場合、guardium_auditlog)。

7. 「監査データベースの作成」をクリックして、監査データベースを作成します。
8. 「構成と制御」タブにある「値変更監査データベースの更新とアップロード」を使用して、この表にあるアクションを選択します。

アクション	記述
削除	「データ・ソース」ペインからデータ・ソースを削除するときにクリックします。
変更	「データ・ソース定義」パネルでこのデータ・ソース定義を編集するときにクリックします。
アップロードのスケジュール	この監査データ・ソースのアップロードのスケジュールを設定するときにクリックします。

監査データベース定義後の作業

監査データベースをデータベース・サーバー上に作成すると、「値変更監査ビルダー」からそれを使用できるようになります。このビルダーは、トリガーのビルドに使用するツールです。値変更監査を参照してください。

親トピック: モニターおよび監査

モニター対象表アクセス

この機能は、Optim™ Designer データ・ライフサイクル製品との相互作用を可能にするために、「最後の評価」フィールドを関連する表に追加します。

この機能は、「表の最後の参照」とも呼びます。

この機能は、データ (事前定義外部フィード・マップ) とともに事前定義された Guardium の外部フィード、およびそれを実行する監査プロセスを使用します。

以下の手順を行います。

1. ターゲット (Optim) 表を Informix® データベースに作成します。スクリプトを使用してください。
2. 「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」にナビゲートして「監査プロセス・ビルダー」を開き、「表の最後の参照」という名前のプロセスを編集します。外部フィード・タスクにデータ・ソース (表を含む Informix データ・ソース) を追加して、サーバー・グループのランタイム・パラメーターをセットアップします。それ以外はすべて事前定義されているため、変更する必要はありません。
3. 監査プロセスを実行 (または定期的に実行するようにスケジュール) します。

注: 結果の表には、最後の実行のみ表示されます。受信者カウントは受信者の数であり、最後の実行以後の実行結果のみの数ではありません。

IBM Guardium は、データベース・オブジェクト (特に表) への外部参照を検出できます。この機能を Optim Designer とともに使用することで、非アクティブ表の廃止や、特定の保存ポリシーによるアーカイブを管理できます。

Guardium® は、最後の参照日を含む表のリストを収集して保持します。リストは Guardium でポリシーを使用して作成されます。 このポリシーには、リスト内容の更新に使用する、最後の参照の間隔と頻度が示されています。Guardium によってキャプチャーされる情報は「最後の参照」リストと呼ばれ、参照されなくなった表、および廃止の対象となる表のアクセス傾向などの情報を提供します。

アプリケーションの廃止を正確に計画できると、以下の作業に便利です。

- ハードウェアの廃止または再デプロイメントの計画
- アプリケーションをサポートするリソース (例えば、ハードウェア、DBA、アプリケーションの所有者、バックアップなどの IT 運用) の移動または廃止による、所有コストの削減。
- ほとんど、あるいはまったくアクセスされない表の認識

IBM Guardium のこの機能は、Optim Designer ユーザー・インターフェースに直接追加されました。

Guardium によって Optim に提供される情報は、表項目ごとの以下の属性で構成されています。

表 1. モニター対象表アクセスのリスト項目

リスト項目	記述
フィールド	コメント
DataSourceDesc	記述
サーバー IP	
ホスト名	
DB ベンダー	Oracle や DB2® など。
ユーザー名	例えば、Oracle の場合は、主にスキーマを定義します。
データベース名	
スキーマ	
表	
日付	最後のアクセス日

Optim 製品での Informix 表の作成スクリプト

Last_referenced_datasource

```
create table last_referenced_datasource (
    id              serial(1) not null,
    datasource_desc    varchar(100),
    server_ip         char(39),
    host_name          varchar(200),
    db_vendor         char(40),
    primary key (id) constraint last_referenced_datasource_pk
```

```
);
Last_referenced_table
create table last_referenced_table (
    id              serial(1) not null,
    datasource_id   int not null,
    user_name       char(32),
    db_name         char(128) not null,
    schema_name     char(128) not null,
    table_name      char(128) not null,
    last_reference  datetime year to second not null,
    primary key (id) constraint last_referenced_table_pk,
    foreign key (datasource_id) references last_referenced_datasource(id) constraint last_referenced_table_fk
);
```

親トピック: モニターおよび監査

NAS および SharePoint のファイル・アクティビティー・モニター

Guardium ファイル・アクティビティー・モニター (FAM) は、Windows 環境の NAS デバイスおよび SharePoint サーバー上のファイルとディレクトリーのアクティビティーをモニターします。

NAS (Network Attached Storage) は、複数のストレージ・デバイスが含まれたネットワーク・アプライアンスに基づくファイル・レベルのストレージ・システムです。SharePoint は、Web ベースのコラボレーション・プラットフォームであり、文書管理およびストレージ・システムでもあります。

FAM でこれらの環境をモニターできるため、ユーザーは、脅威を識別して操作を簡素化できます。

NAS デバイスまたは SharePoint 環境でファイル・アクティビティー・モニターを有効にするには、以下のワークフローを使用します。

- サポートされるプラットフォーム
  Guardium ファイル・アクティビティー・モニター (FAM) は、以下の Windows プラットフォーム、NAS デバイス、および SharePoint のバージョンにインストールできます。
- モニターのアクセス許可
  NAS 環境または SharePoint 環境でファイル・アクティビティー・モニターを許可するために、以下のアクセス許可を有効にします。
- インストール
  NAS または SharePoint 環境にファイル・アクティビティー・モニター (FAM) をインストールするには、以下の手順を実行します。
- 構成
  インストールした後、NAS または SharePoint 環境のモニターを開始するために、ファイル・アクティビティー・モニター (FAM) を構成します。
- 結果の表示
  NAS デバイス上のファイル・アクティビティーを表示するには、「NAS ファイル・アクティビティー」レポートを使用します。SharePoint の場合は、「SharePoint ファイル・アクティビティー」レポートを使用します。

親トピック: モニターおよび監査

サポートされるプラットフォーム

Guardium ファイル・アクティビティー・モニター (FAM) は、以下の Windows プラットフォーム、NAS デバイス、および SharePoint のバージョンにインストールできます。

サポートされる Windows プラットフォーム

FAM は、以下の Windows サーバーにインストールできます。

- Windows 2016
- Windows 2012 R2
- Windows 2012
- Windows 2008 R2

サポートされる Network Attached Storage デバイス

FAM は、以下の Network Attached Storage (NAS) デバイスと互換性があります。

- Hitachi® 11.2 以上
- NetApp® Data ONTAP®:
  - Cluster-Mode 8.2 以上
  - 7-Mode 7.2 以上
- EMC® VNX®:
  - VNX® 8.1
  - VNX® 7.1
- EMC® Isilon® 7.0 以上

- EMC® Celerra® 6.0 以上
- Dell EMC Unity™

サポートされる SharePoint のバージョン

FAM は、以下の SharePoint のバージョンと互換性があります。

- SharePoint® 2016
- SharePoint® 2013
- SharePoint® 2010

親トピック: NAS および SharePoint のファイル・アクティビティー・モニター

モニターのアクセス許可

NAS 環境または SharePoint 環境でファイル・アクティビティー・モニターを許可するために、以下のアクセス許可を有効にします。

NAS のアクセス許可

NetApp Data ONTAP Cluster-Mode のアクセス許可

NetApp Data ONTAP Cluster-Mode デバイスをターゲットとする場合、ポリシー名と資格情報の大文字小文字が区別されます。ポリシー名は `StealthAUDIT`、エンジン名は `StealthAUDITEngine` でなければなりません。FPolicy をカスタマイズすることをお勧めします。NetApp デバイスへの影響が軽減されるためです。

アクティビティーのモニターに使用される FPolicy に関連付けられる資格情報は、少なくとも以下の CLI コマンドでプロビジョンされる必要があります。

CLI コマンド	アクセス権限
version	読み取り専用
volume	読み取り専用
vserver	読み取り専用

FPolicy を有効にして構成するためのその他のオプションについては、以下の CLI コマンドを使用します。

「FPolicy の有効化および接続 (Enable and connect FPolicy)」オプションの使用

ファイル・アクティビティー・モニターは、FPolicy の定期的な検査によって、すべてがアクティブにモニターされるように構成することができます。「FPolicy の有効化および接続 (Enable and connect FPolicy)」オプションを有効にする場合、FPolicy を有効にして、FPolicy に接続し、イベントを収集するために、資格情報には以下のアクセス許可が必要になります。

CLI コマンド	アクセス権限
version	読み取り専用
volume	読み取り専用
vserver	読み取り専用
vserver fpolicy disable	すべて
vserver fpolicy enable	すべて
vserver fpolicy engine-connect	すべて

「FPolicy の構成 (Configure FPolicy)」オプションの使用

ファイル・アクティビティー・モニターは FPolicy を自動的に構成できます。「FPolicy の構成 (Configure FPolicy)」オプションを有効にする場合、FPolicy を有効にして、FPolicy に接続し、イベントを収集するために、資格情報には以下のアクセス許可が必要になります。

CLI コマンド	アクセス権限
version	読み取り専用
volume	読み取り専用
vserver	読み取り専用
server fpolicy	すべて
security certificate install (FPolicy TLS 接続の場合にのみ必要)	すべて

NetApp Data ONTAP 7-Mode のアクセス許可

FAM のインストール先で「ファイルとプリンターの共有」を有効にする必要があります。

ターゲット・デバイス上でファイル・アクティビティー・モニター用に FPolicy が構成されている必要があります。FPolicy をカスタマイズすることをお勧めします。NetApp デバイスへの影響が軽減されるためです。アクティビティーのモニターに使用される FPolicy に関連付けられる資格情報は、以下の API 呼び出しに対するアクセスでプロビジョンされる必要があります。

- login-http-admin api-system-api-list
- api-system-get-version
- api-cifs-share-list-iter-* api-volume-list-info-iter-*

ファイル・アクティビティー・モニターが FPolicy を自動的に構成する場合は、以下のコマンドも必要です。

- api-fpolicy*

ファイル・アクティビティー・モニターが「FPolicy の有効化および接続 (Enable and connect to the FPolicy)」オプションを使用するように構成される場合、以下のコマンドも必要です。

- cli-fpolicy*

資格情報には、ターゲット・デバイスに対する以下のアクセス許可も必要です。

- 以下の両方のグループのグループ・メンバーシップ:

- ONTAP Power Users
- ONTAP backup Operators

EMC Celeriac または Unity のデバイス

FAM エージェントがデプロイされる Windows プロキシー・サーバーに EMC Common Event Enabler (CEE) をインストールする必要があります。

EMC Isilon デバイス

ファイル・アクティビティー・モニター・エージェントがデプロイされる Windows プロキシー・サーバーに EMC Common Event Enabler (CEE) をインストールする必要があります。

Hitachi

Hitachi デバイスは、複数のエンタープライズ仮想サーバー (EVS) をホストできます。各 EVS には複数のファイル・システムがあります。監査の有効化および構成は、ファイル・システムごとに行われます。HNAS は、監査ログ・ファイルを EVT 形式 (Windows XP/2003 以前の標準のイベント・ログ形式) で生成します。Hitachi は、生成された監査ログを、ファイル・システム上のユーザーが指定した場所に保管します。FAM は、この場所にアクセスして、生成されるログ・ファイルを収集します。アクティビティーのモニターに使用される資格情報は、以下でプロビジョンされる必要があります。

- Hitachi デバイスに対してファイル・システム監査ポリシーを有効にする機能
- Hitachi ログ・ディレクトリーに対する監査権限

ファイアウォール・ルール - Windows プロキシー・サーバー

NetApp Data ONTAP Cluster-Mode ファイアウォール・ルール

FAM と NetApp Data ONTAP Cluster-Mode デバイスの間の通信には、以下のファイアウォール設定が必要です。

通信方向	プロトコル	ポート	記述
FAM から NetApp	HTTP (オプション)	80	ONTAPI
FAM から NetApp	HTTPS (オプション)	443	ONTAPI
NetApp から FAM	TCP	9999	FPolicy イベント

NetApp Data ONTAP 7-Mode ファイアウォール・ルール

FAM と NetApp Data ONTAP 7-Mode デバイスの間の通信には、以下のファイアウォール設定が必要です。

通信方向	プロトコル	ポート	記述
FAM から NetApp*	HTTP (オプション)	80	ONTAPI
FAM から NetApp*	HTTP (オプション)	443	ONTAPI
FAM から NetApp	TCP	135, 139 動的範囲 (49152 から 65535)	RPC
FAM から NetApp	TCP	445	SMB
FAM から NetApp	UDP	137, 138	RPC
NetApp から FAM	TCP	135, 139 動的範囲 (49152 から 65535)	RPC
NetApp から FAM	TCP	445	SMB
NetApp から FAM	UDP	137, 138	RPC

* ファイル・アクティビティー・モニターの中で「FPolicy 構成 (FPolicy Configuration)」オプションおよび「FPolicy の有効化および接続 (FPolicy Enable and Connect)」オプションを使用する場合にのみ必要です。

EMC ファイアウォール・ルール

FAM と EMC Celerra、Dell EMC Unity、または EMC Isilon の各デバイスとの間の通信には、以下のファイアウォール設定が必要です。

通信方向	プロトコル	ポート	記述
EMC Isilon デバイスから CEE サーバー	TCP	TCP 12228	CEE 通信
EMC デバイス (Isilon 以外) から CEE サーバー	TCP	RPC の動的範囲	CEE 通信

Hitachi ファイアウォール・ルール

FAM と Hitachi デバイスの間の通信には、以下のファイアウォール設定が必要です。

通信方向	プロトコル	ポート	記述

通信方向	プロトコル	ポート	記述
単一方向	TCP	445	SMB

SharePoint のアクセス許可

- 指定されるドメイン・ユーザーは、SharePoint アプリケーション・サーバー上のローカル管理者でなければなりません。
- SharePoint で監査設定を有効にする必要があります。

親トピック: NAS および SharePoint のファイル・アクティビティー・モニター

インストール

NAS または SharePoint 環境にファイル・アクティビティー・モニター (FAM) をインストールするには、以下の手順を実行します。

始める前に

- 先に進む前に、モニターのアクセス許可でアクセス権を確認してください。
- プラットフォームの前提条件およびサポートの詳細については、サポートされるプラットフォームを参照してください。
- サード・パーティーの前提条件: EMC デバイスをモニターするには、FAM がインストールされる Windows プロキシー・サーバーに EMC Common Event Enabler (CEE) をインストールする必要があります。

手順

1. 環境として NAS を使用するのか SharePoint を使用するのかを決定します。ユーザーが NAS デバイスをモニターする場合は、ネットワーク経由で NAS デバイスにアクセスできる Windows サーバーに FAM をインストールします。一方、ユーザーが SharePoint をモニターする場合は、SharePoint サーバーまたは SharePoint サーバー・ファームに FAM を直接インストールします。
   注: このサーバー上に他の Guardium 製品があってはなりません。
2. FAM for NAS パッケージまたは FAM for SharePoint パッケージを Fix Central からサーバーにダウンロードして、このファイルを unzip します。
3. FAM パッケージのインストーラー・ディレクトリーにナビゲートして、インストーラー・ディレクトリー内の実行可能ファイル setup.exe を実行します。
4. ウィザードのプロンプトに従って、インストールを完了します。
   注:

   NAS の場合、デフォルトのインストール・ディレクトリーは、C:¥Program Files¥IBM¥FAMforNAS です。

   SharePoint の場合、デフォルトのインストール・ディレクトリーは、C:¥Program Files¥IBM¥FAMforSP です。

親トピック: NAS および SharePoint のファイル・アクティビティー・モニター

構成

インストールした後、NAS または SharePoint 環境のモニターを開始するために、ファイル・アクティビティー・モニター (FAM) を構成します。

NAS デバイスの構成

Guardium ファイル・アクティビティー・モニターで、「モニター対象ホスト (Monitored Hosts)」タブの以下のオプションを使用して、NAS デバイスを構成します。

追加:
リストから、モニターする NAS デバイスを選択します。
編集:
メニューの「編集」ボタンをクリックして、以下のタブを使用し、NAS デバイスを構成します。

(a) 選択された NAS デバイス (Selected NAS Device)
「選択された NAS デバイス (Selected NAS Device)」タブで、テキスト・フィールドを使用して NAS サーバーの名前を指定します。
(b) 操作 (Operations)
モニターするアクティビティー・イベントを選択します。これらの操作をファイルまたはディレクトリーのアクティビティーに関連付けることができます。
(c) パス・フィルタリング (Path Filtering)
ホストのプロパティー・ウィンドウにあるこのタブでは、ユーザーは、ファイル・パスのコレクション・スコープ・フィルターを追加できます。指定されたパスをモニター対象に含めることや、モニター対象から除外することができます。
(d) アカウントの除外 (Account Exclusions)
ここに追加されたアカウントは、ファイル・システム・アクティビティーのモニター対象から除外されます。
(e) UNIX ID (Unix IDs)
このタブでは、UNIX ID (UID) を Windows SID に変換する構成オプションを指定できます。これは、NetApp デバイスおよび EMC デバイスにのみ適用されます。NAS デバイス上のアクティビティーがある場合、そのアクティビティー・イベントの UID が返されます。オペレーティング・システムによっては、Active Directory の uidNumber 属性を使用して、UID を Active Directory アカウントにマップできます。アクティビティー・エージェントは、アクティビティー・イベントからの UID に基づいて Active Directory SID を解決します。

SharePoint の構成

Guardium ファイル・アクティビティー・モニターで、「モニター対象ホスト (Monitored Hosts)」タブの以下のオプションを使用して、SharePoint デバイスを構成します。

追加:
モニターする SharePoint サーバーまたはサーバー・ファームを追加します。
編集:

メニューの「編集」ボタンをクリックして、以下のタブを使用し、SharePoint を構成します。

(a) SharePoint
ローカル・システムと SharePoint の両方に対する管理特権を持つ資格情報を指定します。次に、「接続 (Connect)」をクリックします。
(b) 操作 (Operations)
モニターする SharePoint 操作および許可操作を選択します。
(c) アカウントの除外 (Account Exclusions)
ここに追加されたアカウントは、SharePoint アクティビティーのモニター対象から除外されます。

親トピック: NAS および SharePoint のファイル・アクティビティー・モニター

結果の表示

NAS デバイス上のファイル・アクティビティーを表示するには、「NAS ファイル・アクティビティー」レポートを使用します。SharePoint の場合は、「SharePoint ファイル・アクティビティー」レポートを使用します。

手順

1. 「マイ・ダッシュボード」 > 「新規ダッシュボードの作成」をクリックして、新規ダッシュボードを開きます。
2. 「レポートの追加」をクリックすると、使用可能なレポートのリストが表示されます。「レポートの追加」ダイアログには、指定された基準を満たすすべてのレポートのリストが表示されます。レポートのリストを参照することも、「フィルター」フィールドに文字列を入力することもできます。文字を入力するにつれて、レポートのリストが更新されます。
3. NAS デバイス上のファイル・アクティビティーは「NAS ファイル・アクティビティー」レポートで表示でき、SharePoint 上のファイル・アクティビティーは「SharePoint ファイル・アクティビティー」レポートで表示できます。レポートをクリックして、ダッシュボードに追加します。

タスクの結果

結果を表示するには、カスタム・ポリシーを作成してインストールします。インストールされているデフォルトのポリシーは、すべてのトラフィックを無視する「不明な接続に対するデータ・アクティビティーを無視 [テンプレート]」です。ポリシーの作成方法について詳しくは、ポリシーおよびポリシー・ルールの作成とインストールを参照してください。
親トピック: NAS および SharePoint のファイル・アクティビティー・モニター

コンプライアンス・モニターのクイック・スタート

モニター・エージェント (S-TAP) をデプロイした後、コンプライアンス・モニター・ツールを使用して、特定のセキュリティー基準および規制のモニターを設定します。

Guardium は、以下のような特定の基準および規制に対応するグループ、セキュリティー・ポリシー、およびレポートなどの、コンプライアンス・モニター・テンプレートをいくつか備えています。

- バーゼル銀行監督委員会 (BASEL II)
- 一般データ保護規則 (GDPR)
- Db2 for z/OS 用の一般データ保護規則 (Db2 for z/OS の GDPR)
- 医療保険の相互運用性と説明責任に関する法律 (HIPAA)
- クレジット・カード業界データ・セキュリティー基準 (PCI)
- 個人情報 (PII)
- サーベンス・オクスリー (SOX) 法への準拠

これらのクイック・スタート・コンプライアンス・モニター・テンプレートは、関連する基準または規制のいずれかに短期間で準拠する必要がある組織に特に役立ちます。セキュリティー・ポリシーをインストールした後、コンプライアンス・モニター・ツールは、初期セットアップやグループへの組織固有の情報 (クライアント IP アドレスや特定の特権ユーザー ID など) の取り込みについて、管理者およびコンプライアンス担当者にガイドを提供します。 さらに、コンプライアンス・モニター・ツールは、Guardium 環境を定期的に検査して、コンプライアンス・モニター・テンプレートを使用してモニターできる新規のデータベースを調べます。
コンプライアンス・モニター・テンプレートを選択し、そのコンプライアンス・タイプを適用する必要があるデータベースを示すと、コンプライアンス・モニター・ツールは以下のアクションを実行します。

- 選択されたコンプライアンス・タイプのセキュリティー・ポリシーが作成されて、インストールされます。一元管理された環境では、ポリシーはコレクターにインストールされます。
- ポリシー・インストール・スケジュールは毎日午前 10:30 に定義されます。一元管理された環境では、ポリシー・インストール・スケジュールはコレクターで実行されます。
- 選択したデータベースのサーバー IP アドレスがサーバー IP グループに取り込まれます。
- 現行ユーザーは、選択されたコンプライアンス・タイプのロールに割り当てられます。このロールにより、Guardium のメイン・ナビゲーションから関連するレポートおよびアクセラレーターへのアクセスが可能になります。
- サポートされている場合、機密データのディスカバー・シナリオが作成されます。
- 機密データのディスカバー・シナリオが作成され、選択されたデータベースの少なくとも 1 つにデータ・ソースが定義されている場合、シナリオは 1 週間に 1 回、日曜日の午前 10:30 に実行するようにスケジュールされます。一元管理された環境では、スケジュールは中央マネージャーで実行されます。

次の表に、使用可能な各コンプライアンス・タイプでサポートされる機能を要約します。

表 1. コンプライアンス・モニター・ツールによってサポートされる、コンプライアンス・タイプ別の機能の要約

	バーゼル II	GDPR	HIPAA	PCI	PII	SOX
セキュリティー・ポリシー	✓	✓	✓	✓	✓	✓
レポート	✓	✓		✓	✓	✓
機密データのディスカバー・シナリオ		✓		✓	✓	

- コンプライアンス・モニターの前提条件
  コンプライアンス・モニターを構成する前に、前提条件および制約事項を確認します。
- コンプライアンス・モニターのセットアップ
  コンプライアンス・モニターの初期構成を実行する方法について説明します。
- グループへのデータの設定
  コンプライアンス・モニターのためにグループにデータを設定する方法について説明します。
- 機密データのスキャンの有効化
  データベース資格情報を保管し、機密データのディスカバリーおよび分類を許可する方法について説明します。
- コンプライアンス・モニター・ビューの概要
  コンプライアンス・モニター・ビューの解釈および応答方法について説明します。

親トピック: モニターおよび監査
関連概念:
ポリシー
データ・ソース
関連タスク:
機密データのディスカバー
関連情報:
グループ
Guardium GDPR アクセラレーター (ビデオ)

コンプライアンス・モニターの前提条件

コンプライアンス・モニターを構成する前に、前提条件および制約事項を確認します。

コンプライアンス・モニター・ツールのクイック・スタートでは、テンプレートを使用することにより、コンプライアンス・モニターを環境内の新規のデータベース・サーバーに対して素早く設定します。これらのテンプレートは、新規の Guardium デプロイメントまたは拡張している Guardium デプロイメントでの使用に合わせて最適化されます。 始める前に、以下の前提条件を検証することにより、最も簡単な構成と最も完全な機能を確保します。

- 中央マネージャーまたはスタンドアロン・システムとして構成された Guardium V10.1.3 以降を実行している、管理特権を持つ Guardium ユーザーである。
- S-TAP が新規のデータベース・サーバーにインストールされ、作動可能になっている。
- データベース・サーバーは、コンプライアンス・モニター・テンプレートによってサポートされている。
- *「デフォルト - 不明な接続に対するデータ・アクティビティーを無視」*ポリシー以外のポリシーがインストールされていない。
  警告:

  既存のポリシーに次の「ポリシー定義」設定がある場合にのみ、既存のポリシーとともにクイック・スタート・コンプライアンス・モニター・セキュリティー・ポリシーをインストールできます。

  - 未解析ログ: 無効
  - 未解析ログに関するルール: 無効
  - 選択的な監査証跡: 有効

  既存のポリシーの設定が競合している場合、クイック・スタート・セキュリティー・ポリシーのインストールは失敗します。既存のデプロイメントで作業している場合は、クイック・スタート・ポリシーを使用する前に、既存のポリシーをアンインストールすることを検討してください。この制約事項は、新規の Guardium デプロイメントに影響を及ぼすことはありません。

以下のセクションでは、クイック・スタート・コンプライアンス・モニターの前提条件について詳しく説明します。

モニター・エージェントのデプロイ

コンプライアンス・モニターの構成を開始する前に、Guardium モニター・エージェント (S-TAP) がデータベース・サーバーにインストールされ、Guardium システムと通信するように構成されている必要があります。Guardium モニター・エージェントの迅速なインストールおよび構成については、『モニター・エージェントのデプロイ』を参照してください。

他のインストール方法を含む S-TAP の詳細については、「S-TAP 管理ガイド」を参照してください。

サポートされるデータベース

コンプライアンス・モニター・ツールは、以下の基準に基づいて、Guardium 環境でデータベースを検出します。

- Guardium システム上のアクティブ・トラフィック。
- ディスカバーされたインスタンス・レポート (「ディスカバー」 > 「レポート」 > 「ディスカバーされたインスタンス」)。

次の表に要約されているように、検出方式はサポートされるデータベース・タイプによって異なります。

表 1. サポートされるデータベース・タイプおよび検出方式の要約。

データベース	アクティブ・トラフィック	ディスカバーされたインスタンス
Db2 for Linux, UNIX, and Windows	✓	

データベース	アクティブ・トラフィック	ディスカバーされたインスタンス
Db2 for z/OS	✓ 重要: *「デフォルト - 不明な接続に対するデータ・アクティビティーを無視」*ポリシーでは、Db2 for z/OS データベースのトラフィックがキャプチャーされません。Db2 for z/OS データベースでコンプライアンス・モニター・ツールを使用する前に、このトピックで説明されているポリシー定義およびアクティブ・トラフィック基準を満たすポリシーをまずインストールする必要があります。	
Informix	✓	✓
Microsoft SQL Server	✓	✓
MySQL	✓	✓
Netezza	✓	
Oracle	✓	✓
PostgreSQL	✓	
Sybase	✓	✓
Teradata	✓	✓

アクティブ・トラフィックは、以下の基準を満たします。

- トラフィックでは、以下のいずれかのプロトコルが使用されます。
  - Db2 for z/OS データベース: BATCH、CALL、CICS、CTL、DRDA、PRIV、RRSAF、TRAN、TSO、または UTIL。
  - その他のすべてのデータベース: TCP。
- トラフィックはローカルではない (サーバー IP はクライアント IP と等しくない)。
- 失敗したログインは無視される。
- トラフィックは暗号化されていない。

新しいデータベースについて、アクティブ・トラフィックが毎正時 17 分後に検査されます。例えば、13:00 に設定されたデータベースからのアクティブ・トラフィックは、13:17 に検出されます。
ディスカバーされたインスタンスは、以下の基準を満たします。

- データベースでポート範囲が指定されていない。
- データベースがデータ・ソースの作成にデータベース名を必要としない。

抽出ルールおよび戻りデータの検査

クイック・スタート・コンプライアンス・モニター・ポリシーによっては、抽出ルールを使用するものがあります。抽出ルールは、要求に応答してサーバーから返されたデータを評価します。例えば、抽出ルールは、社会保障番号やクレジット・カード番号などの機密データに関連付けられた数字パターンを検査する場合があります。

抽出ルールでは、ポリシーを使用するすべての検査エンジンについて「戻りデータの検査」設定が有効になっている必要があります。次のコンプライアンス・テンプレートに含まれる抽出ルールを使用するには、返されたデータを検査エンジンが検査できるようにする必要があります。

- GDPR
- HIPAA
- PCI
- PII (データ・プライバシー)

重要: 「戻りデータの検査」を有効にすると、返された結果セットによってネットワーク・トラフィックが増大します。
「戻りデータの検査」は、「管理」 > 「アクティビティー・モニター」 > 「検査エンジン」から有効にします。「戻りデータの検査」の設定について詳しくは、『検査エンジン構成』を参照してください。

ポリシー定義の設定

すべてのコンプライアンス・モニター・セキュリティー・ポリシーは、以下のポリシー定義設定を使用します。

- 未解析ログ: 無効
- 未解析ログに関するルール: 無効
- 選択的な監査証跡: 有効

競合する「未解決ログ」、「未解析ログに関するルール」、または「選択的な監査証跡」の設定が含まれるポリシーは、同じ Guardium 環境にインストールできません。その結果、異なる設定を使用するポリシーがインストールされている場合、クイック・スタート・コンプライアンス・モニター・テンプレートを使用できません。

新規の Guardium デプロイメントまたはユーザー定義のポリシーがないデプロイメントの場合、これらのポリシー設定との競合が発生する可能性は低くなります。既存の Guardium デプロイメントの場合、「コンプライアンス・モニターのセットアップ」ツールの使用中に「*ポリシーが競合しています (conflicting policies)*」というメッセージを受け取った場合、ポリシー定義設定を確認してください。

選択的な監査証跡について詳しくは、ポリシー・ルールのアクションを参照してください。

例外: インストールされているポリシーが 1 つのみの場合、*「デフォルト - 不明な接続に対するデータ・アクティビティーを無視」*ポリシーは、コンプライアンス・モニター・ポリシーのインストールによってオーバーライドされます。
親トピック: コンプライアンス・モニターのクイック・スタート

コンプライアンス・モニターのセットアップ

コンプライアンス・モニターの初期構成を実行する方法について説明します。

始める前に

前提条件および制約事項について詳しくは、コンプライアンス・モニターの前提条件を参照してください。

このタスクについて

コンプライアンス・モニターのセットアップ・ツールを使用して、データベースを 1 つ以上のコンプライアンス・テンプレートに関連付けます。この手順では、セキュリティー・ポリシー、グループ、レポート、および機密データのディスカバー・シナリオ (サポートされている場合) を迅速にインストールします。

手順

1. 「設定」 > 「クイック・スタート」 > 「コンプライアンス・モニター」にナビゲートして、コンプライアンス・モニター・ページを開きます。
2. 「コンプライアンス・モニターのセットアップ」タイルの アイコンをクリックして、コンプライアンス・モニターのセットアップ・ツールを開きます。
3. 「コンプライアンス・タイプ」セクションで、「有効にするコンプライアンス・タイプを選択してください」メニューを使用して、構成するデータベース・モニターのタイプを選択します。 例えば、GDPR モニターを有効にするには、「一般データ保護規則 (GDPR)」を選択します。「次へ」をクリックして先に進みます。
4. 「データベース」セクションで、「使用可能なデータベース」表からデータベースを選択し、 アイコンをクリックして、そのデータベースを「選択されたデータベース」表に追加します。
   ヒント:
   - 「モニター対象データベースの除外」チェック・ボックスを使用すると、コンプライアンス・モニターが既に構成されているデータベースを非表示にできます。
   - 「Db2 for z/OS 用の一般データ保護規則 (Db2 for z/OS の GDPR)」コンプライアンス・タイプを使用する場合、Db2 for z/OS データベースのみが含まれるように使用可能なデータベースのリストがフィルタリングされます。同様に、z/OS 以外のコンプライアンス・タイプを使用する場合、Db2 for z/OS データベースは表示されません。
   - 「選択されたデータベース」表からデータベースを選択し、「資格情報の指定」をクリックして、データベース資格情報を保管します。 資格情報を保管すると、一部のコンプライアンス・タイプの機密データのディスカバリーおよび分類が可能になります。自動構成がサポートされていない場合、資格情報の保管時に作成されるデータ・ソースを独自の機密データのディスカバー・シナリオで使用できます。
   - データベースとコンプライアンス・タイプの関連付けを解除するには、構成を編集して、データベースを「選択されたデータベース」表から削除するか、またはコンプライアンス・タイプ・タイルから「詳細表示」 > 「データベース」にナビゲートして、データベースの横にある アイコンをクリックします。
5. モニターするデータベースの識別が完了したら、「セットアップの実行」をクリックしてポリシーをインストールし、サーバー IP グループにデータを設定して、コンプライアンス・モニター・レポートを実行します。
6. 「ページを最新表示して新しいコンテンツを表示しますか?」ダイアログで、「はい」をクリックしてページを最新表示し、セットアップを完了します。

タスクの結果

コンプライアンス・モニターをセットアップした後、構成したコンプライアンス・テンプレートに対応するコンプライアンス・モニター・ダッシュボードにタイルが表示されます。

次のタスク

コンプライアンス・モニターを構成した後、コンプライアンス・モニター・タイルにいくつかの アイコンが表示される場合があります。これらのアイコンは、追加の構成が必要であることを示しています。「グループにデータを設定する」リンクを使用して追加グループにデータを設定するか、または「データ・ソース資格情報」リンクを使用して機密データのディスカバー・シナリオ用にデータベース資格情報を指定します。

重要: コンプライアンス・モニター・セットアップ・ツールを使用してモニターを構成すると、デフォルトのサーバー IP グループが自動的に作成され、データが設定されます。ただし、いくつかの追加グループにデータを設定することによって、データベースへのアクセスを許可されるユーザーおよびアプリケーションを定義することが重要です。コンプライアンス・モニター・ページからのグループへのデータの設定については、グループへのデータの設定を参照してください。
親トピック: コンプライアンス・モニターのクイック・スタート
関連概念:
コンプライアンス・モニターの前提条件
関連情報:
モニター・エージェントのデプロイ

グループへのデータの設定

コンプライアンス・モニターのためにグループにデータを設定する方法について説明します。

始める前に

コンプライアンス・モニターのセットアップで説明されている手順に従って、コンプライアンス・モニター・テンプレートをインストールします。

このタスクについて

コンプライアンス・モニター・セットアップ・ツールを使用してモニターを構成すると、デフォルトのサーバー IP グループが自動的に作成され、データが設定されます。ただし、いくつかの追加グループにデータを設定することによって、データベースへのアクセスを許可されるユーザーおよびアプリケーションを定義することが重要です。 以下の手順では、グループに素早くデータを設定する方法について説明します。
重要:

- 空のグループはワイルドカードとして扱われず、トラフィックを収集しません。
- 階層グループおよびネストされたグループはサポートされません。

手順

1. 以下のいずれかの方法を使用して、データが設定されていないグループを識別し、「グループの編集」ダイアログを開いて開始します。
   - コンプライアンス・モニター・タイルの「モニター使用可能」セクションで、 アイコンを探して、関連付けられた「グループにデータを設定する」リンクをクリックします。
   - コンプライアンス・モニター・タイルの「詳細表示」リンクをクリックして、詳細パネルを開き、「要約」タブを選択して、グループの横にある アイコンをクリックします。
     ヒント: 詳細パネルで、データが設定されていないグループが小さい アイコンで強調表示されます。

   この時点で、詳細ビューおよび 「グループの編集」 ダイアログがコンプライアンス・モニター・ダッシュボードの上に表示されます。
2. 「グループの編集」ダイアログから、オプションでグループの「カテゴリー」および「分類」を指定します。 「アプリケーション・タイプ」フィールド、「グループ・タイプ」フィールド、および 「説明」(「グループ名」として使用される) フィールドは、前のステップで選択されたグループに基づいてデータが設定されており、編集できません。
3. 「グループの編集」ダイアログから、以下のいずれかの方法を使用して、選択したグループのデータの設定を開始します。
   - アイコンをクリックして、項目を「メンバー」表に追加し、手動でグループ・メンバーを指定します。
   - CSV ファイルからグループ・メンバーをインポートするには、「インポート」 > 「CSV から」をクリックします。
   - 同じグループ・タイプの別の Guardium グループからグループ・メンバーをインポートするには、「インポート」 > 「グループから」をクリックします。例えば、*「許可されたユーザー」*グループは、ユーザーのリストが含まれる別のグループからデータを設定することができますが、IP アドレスのリストが含まれるグループからデータを設定することはできません。
   - 外部データ・ソースからグループ・メンバーをインポートするには、「インポート」 > 「外部データ・ソースから」をクリックします。「データ・ソース」メニューには、*共有* というマークが付いているか、またはタイプが *カスタム・ドメイン* であるすべてのデータ・ソースが含まれます。詳しくは、『外部データ・ソースからのインポート』を参照してください。
4. グループへのメンバーの追加が完了したら、「OK」をクリックして、コンプライアンス・モニター・ダッシュボードに戻ります。

親トピック: コンプライアンス・モニターのクイック・スタート

機密データのスキャンの有効化

データベース資格情報を保管し、機密データのディスカバリーおよび分類を許可する方法について説明します。

始める前に

コンプライアンス・モニターのセットアップで説明されている手順に従って、コンプライアンス・モニター・テンプレートをインストールします。

このタスクについて

以下の手順では、コンプライアンス・モニター・ツールを使用してデータベース資格情報を保管することによってデータ・ソースを作成する方法について説明します。資格情報を保管して、データ・ソースを作成することにより、Guardium は機密データのディスカバリーおよび分類のためにデータベースにアクセスできます。

手順

1. 以下のいずれかの方法を使用して、データベース資格情報が必要な場所を特定します。
   - コンプライアンス・モニター・タイルの「機密データのスキャン中」セクションで、 アイコンを探して、関連付けられた「データ・ソース資格情報」リンクをクリックします。 コンプライアンス・モニター・データベース・ビューが、資格情報を必要とするデータベースのフィルター済みリストに対して開きます。
   - 「データベースの表示」リンクをクリックして、コンプライアンス・モニター・データベース・ビューを開き、「データ・ソース」列に アイコンが表示されていないデータベースを探します。
2. コンプライアンス・モニター・データベース・ビューから、資格情報を必要とするデータベースを選択し、「データ・ソース・アクション」 > 「資格情報の指定」をクリックします。
   ヒント:
   - 複数のデータベースを選択して、「データ・ソース・アクション」 > 「資格情報の指定」をクリックすると、選択したすべてのデータベースの指定された資格情報が保存されます。複数のデータベースの資格情報を指定する際には、選択したデータベースがすべて同じ資格情報を使用していることを確認してください。 同じ資格情報を指定していない場合、異なる資格情報を使用するデータベースは接続テストに失敗します。
   - 資格情報を保管すると、一部のコンプライアンス・タイプの機密データのディスカバリーおよび分類が可能になります。自動構成がサポートされていない場合、資格情報の保管時に作成されるデータ・ソースを独自の機密データのディスカバー・シナリオで使用できます。
3. 「資格情報の指定」ダイアログで、「ユーザー名」フィールドと「パスワード」フィールドを使用して、選択したデータベースの資格情報を指定します。「OK」をクリックして、コンプライアンス・モニター・データベース・ビューに戻ります。
4. コンプライアンス・モニター・データベース・ビューから、資格情報を保管しているデータベースを選択し、「データ・ソース・アクション」 > 「接続のテスト」をクリックします。 「接続のテスト」を使用して、保管された資格情報がデータベースへのアクセスを許可していることを検証します。接続のテストが失敗した場合、機密データのディスカバリーおよび分類は機能しません。

重要:

- 接続のテストは時間がかかる可能性があります。一度に多数の接続をテストすることは推奨されません。
- 接続のテストが失敗した場合は、「設定」 > 「ツールとビュー」 > 「データ・ソース定義」にナビゲートし、データ・ソースを選択して、データ・ソース定義を検証します。例えば、Db2 for z/OS データベースの正しいポートを指定するか、大/小文字混合の PostgreSQL データベース名を修正するか、あるいは使用環境に必要な他の接続プロパティーを設定することが必要な場合があります。
- Microsoft SQL Server の接続のテストが失敗した場合は、「SQL サーバー・ブラウザー」の Windows サービスが開始されていることを確認します。

タスクの結果

機密データのスキャンを有効にした後には、スキャン結果、およびポリシーに対して行った変更 (グループおよびグループ・メンバーシップに対する変更を含む) は、ポリシーがポリシー・インストールのスケジュールに従ってインストールされた後に使用可能になります。デフォルトでは、クイック・スタート・コンプライアンス・モニター・ツールは、毎日午前 10:30 に実行されるポリシー・インストール・スケジュールを定義します。

親トピック: コンプライアンス・モニターのクイック・スタート

コンプライアンス・モニター・ビューの概要

コンプライアンス・モニター・ビューの解釈および応答方法について説明します。

ユーザー・インターフェース

「コンプライアンス・モニター」ツールは、以下のビューから構成されています。

ダッシュボード・ビュー

これはデフォルトのビューであり、コンプライアンス・タイプ別に編成された、コンプライアンス・デプロイメントの現在の状況の概要を示します。個々のタイルには、いくつかのコンプライアンス・モニター・コンポーネントの現在の構成状況が反映されます。これにより、追加構成を必要とするコンプライアンス・タイプを迅速に特定できます。

データベース・ビュー

データベース・ビューには、サポートされるコンプライアンス・モニター・テンプレートを使用して構成されているデータベースを示す表が表示されます。

コンプライアンス・モニターのセットアップ

コンプライアンス・モニターのセットアップ・ツールは、データベースとコンプライアンス・テンプレートを迅速に関連付けるため、および初期セットアップを実行するためのガイド付きインターフェースを提供します。ツールにアクセスするには、ダッシュボード・ビューの「コンプライアンス・モニターのセットアップ」タイルの  アイコンをクリックするか、データベース・ビューでデータベースを選択して、「コンプライアンス・モニターのセットアップ」ボタンをクリックします。

コンプライアンス・モニター・ビューには、コンプライアンス・モニターの設定に関連した構成タスクを完了するための相互に関連する方法がいくつか示されます。次の表に、さまざまなビューでサポートされるタスクの要約を示します。

表 1. コンプライアンス・モニター・ビューによってサポートされるタスクの要約

タスク	コンプライアンス・モニターのセットアップ	ダッシュボード・ビュー	データベース・ビュー
コンプライアンス・タイプとデータベースの関連付け	「データベース」セクションで、「使用可能なデータベース」表からデータベースを選択し、  アイコンをクリックして、そのデータベースを「選択されたデータベース」表に移動します。		
グループへのデータの設定		コンプライアンス・タイプ・タイルから、「グループにデータを設定する」リンクをクリックするか、「詳細表示」 > 「要約」にナビゲートして、グループの横の  アイコンをクリックします。	
機密データをディスカバーするためのデータ・ソースの定義	「データベース」セクションで、「選択されたデータベース」表からデータベースを選択し、「資格情報の指定」ボタンをクリックします。	コンプライアンス・タイプ・タイルから、「データ・ソース資格情報」リンクをクリックし、データベースを選択して、「データ・ソース・アクション」 > 「資格情報の指定」をクリックします。	データベースを選択して、「データ・ソース・アクション」 > 「資格情報の指定」をクリックします。

重要: コンプライアンス・モニター・テンプレートを使用して構成されると、オフラインにされたデータベースは引き続きコンプライアンス・モニター・ツールに表示されます。

ポリシー

クイック・スタート・コンプライアンス・モニター・テンプレートは、効果的で、変更せずに機能するように設計されたセキュリティー・ポリシーを提供します。これらのポリシーを使用して、コンプライアンス・モニターを素早く稼働状態にします。コンプライアンス・モニター・ダッシュボード・ビューから、「詳細表示」 > 「ポリシー」をクリックして、特定のコンプライアンス・タイプに関連付けられたポリシーを表示します。

コンプライアンス・モニターが中央マネージャーから構成されている場合、クイック・スタート・セキュリティー・ポリシーは自動的にすべてのコレクターにプッシュ・ダウンされます。デフォルトのクイック・スタート・セキュリティー・ポリシー以外のポリシーがインストールされている場合は、クイック・スタート・ポリシーが最後にインストールされます。

コンプライアンス・モニター・ポリシーを詳細に検討したい場合、「ポリシー・ファインダー」を介して確認できます。クイック・スタート・コンプライアンス・モニター・ポリシーは、「Quick Start *compliance type*」という命名規則で識別されます。例えば、デフォルトの GDPR ポリシーの名前は Quick Start GDPR です。 また、「データのポリシー・ビルダー」を使用して、コンプライアンス・モニター・セキュリティー・ポリシーを編集することもできます。

コンプライアンス・モニター・ポリシーを変更した場合は、「コンプライアンス・モニター」ダッシュボード・ビューからデフォルト設定に戻します。それには、必要なコンプライアンス・タイプ・タイルで「詳細表示」をクリックし、「ポリシー」タブを選択して、「デフォルトにリセット」をクリックします。デフォルト設定を復元する前に、命名規則 Quick Start *compliance type timestamp* (*timestamp* はデフォルト設定が復元された日時を示す) を使用して、カスタマイズされたすべての設定がポリシーに保持されます。例えば、Quick Start GDPR 2017-05-01 19:17:59 のようになります。

ポリシー・インストールのスケジュール

デフォルトでは、クイック・スタート・コンプライアンス・モニター・ツールは、毎日午前 10:30 に実行されるポリシー・インストール・スケジュールを定義します。

コンプライアンス・モニターがスタンドアロン・マシンから構成されている場合、ポリシー・インストール・スケジュールは、(スケジュールがアクティブか一時停止かどうかに関係なく) 既存のポリシー・インストール・スケジュールが存在しない場合に定義されます。コンプライアンス・モニターが中央マネージャーから構成されている場合、ポリシー・インストール・スケジュールは、(既存のポリシー・インストール・スケジュールが存在するかどうかに関係なく) すべてのコレクターに対して構成されます。

グループ

コンプライアンス・モニター・ツールは、各コンプライアンス・タイプに関連付けられている複数のグループに依存します。有効なコンプライアンス・モニターを設定するには、これらのグループにデータを設定する必要があります。コンプライアンス・モニター・ダッシュボード・ビューから、「詳細表示」 > 「要約」をクリックして、特定のコンプライアンス・タイプに関連付けられたグループを表示します。

制約事項:

- 階層グループおよびネストされたグループはサポートされません。
- 空のグループはワイルドカードとして扱われず、トラフィックを収集しません。
-

データベースの数と、コンプライアンス・タイプの「詳細表示」 > 「要約」タブに示されている「サーバー IP」グループのメンバーの間に不一致がある場合があります。この不一致は、単一のデータベース・サーバー上で実行されている複数のデータベース、またはコンプライアンス・モニター・ツールの外部で更新された「サーバー IP」グループを反映しています。

レポート

クイック・スタート・コンプライアンス・モニター・テンプレートは、コンプライアンス・タイプごとにいくつかの事前定義されたレポートを提供します。コンプライアンス・モニター・ダッシュボード・ビューから、「詳細表示」 > 「レポート」をクリックして、特定のコンプライアンス・タイプに関連付けられたレポートを表示します。 これらのレポートは、Guardium のメイン・ナビゲーションの「アクセラレーター」セクションでも使用できます。このレポートのリストは、コンプライアンス・タイプごとに事前定義されており、ユーザーが定義したカスタム・レポートは反映されません。

制約事項: HIPAA コンプライアンス・モニター・テンプレートには、事前定義レポートは用意されていません。

ユーザーおよびロール

現行ユーザーは、選択されたコンプライアンス・タイプのロールに割り当てられます。このロールにより、Guardium のメイン・ナビゲーションから関連するレポートおよびアクセラレーターへのアクセスが可能になります。複数の異なる Guardium ユーザーが別々のコンプライアンス・タイプを構成する場合、個々のユーザーは、構成されたコンプライアンス・タイプに関連付けられているレポートおよびアクセラレーターにのみアクセスできます。

例えば、*user1* が *GDPR* を構成し、*user2* が *PCI* を構成する場合、*user1* は PCI レポートおよびアクセラレーターにアクセスできません。なぜなら、PCI ロールが *user1* に割り当てられていないからです。ユーザーへの特定のロールの手動割り当てについては、『アクセス管理の概要』を参照してください。

機密データ

コンプライアンス・タイプ・タイルの「一致するものが見つかりました」の値と、「詳細表示」 > 「要約」タブの関連するオブジェクト・グループの間に不一致がある場合があります。「一致するものが見つかりました」は、機密データのディスカバリー・シナリオの基準に一致した固有表名と列名のペアの数を示します。OBJECTS グループのメンバーの数は固有表名の数で、すべてのスキャンからの累積値です。

重要: タイルの「機密データのスキャン中」セクションの  アイコンは、機密データのディスカバー用に 1 つ以上のデータ・ソースが構成されていることを示します。「データベースの表示」をクリックして、機密データのディスカバー用にデータ・ソースが定義されているデータベースを調べます。
親トピック: コンプライアンス・モニターのクイック・スタート

PCI/DSS アクセラレーターを使用して PCI コンプライアンスを実装する方法

PCI/DSS 要件を満たすために、IBM Security Guardium の PCI/DSS アクセラレーターを構成し、一連のポリシーとレポートを作成します。

PCI/DSS (Payment Card Industry/ Data Security Standard) は、カード所有者データを保護するために設計された一連の技術要件と運用要件です。

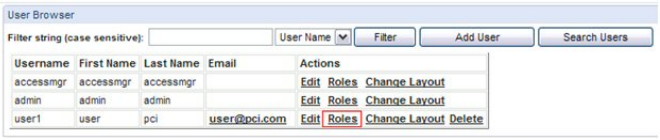
付加価値: PCI/DSS の全体的なビューをユーザーに提供し、構成にかかる時間を短縮するために定義済みのポリシーとレポートを提供します。

以下の手順を行います。

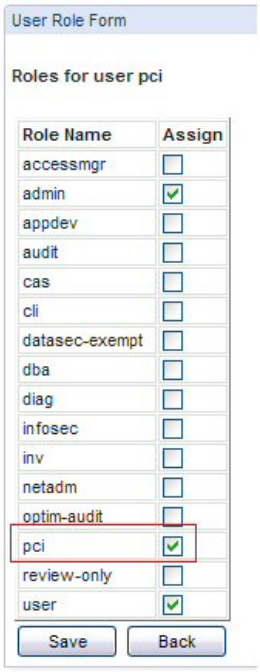
1. PCI ロールを構成します。
2. 要件を満たすレポートとポリシーを構成します。

PCI ロールの構成

1. Guardium GUI ページから「accessmgr」ユーザー・アカウントを使用してログインします。ユーザー (この場合は user1) を選択して、「ロール」をクリックします。

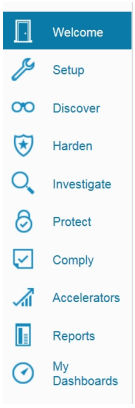


2. 「ユーザー・ロール・フォーム」で PCI を確認し、割り当てを保存します。



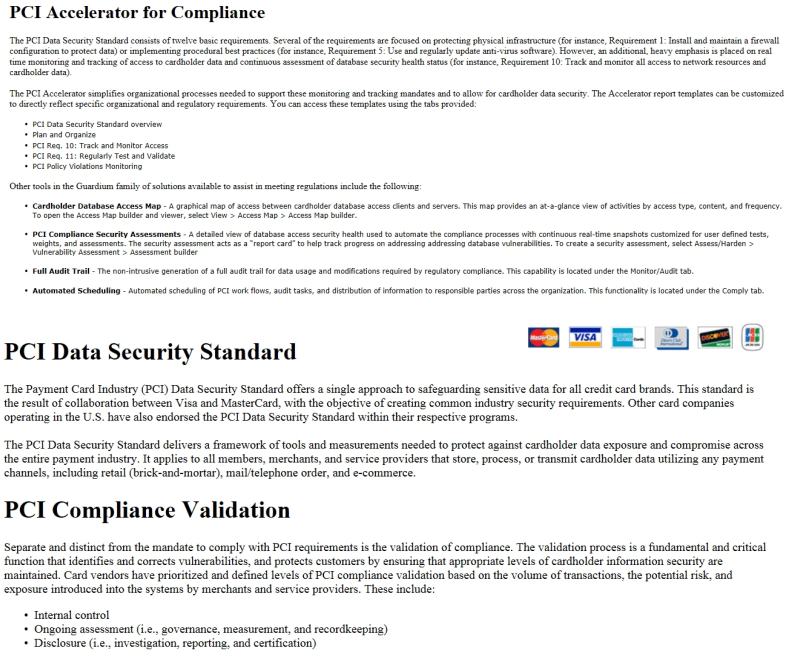
PCI アクセラレーターの実装

「user1」を使用してログオンし、「アクセラレーター」をクリックします。



概要

1. 「コンプライアンスのための PCI アクセラレーター」をクリックします。
2. 「PCI データ・セキュリティー基準」をクリックします。



PCI Accelerator for Compliance

The PCI Data Security Standard consists of twelve basic requirements. Several of the requirements are focused on protecting physical infrastructure (for instance, Requirement 1: Install and maintain a firewall configuration to protect data) or implementing procedural best practices (for instance, Requirement 5: Use and regularly update anti-virus software). However, an additional, heavy emphasis is placed on real time monitoring and tracking of access to cardholder data and continuous assessment of database security health status (for instance, Requirement 10: Track and monitor all access to network resources and cardholder data).

The PCI Accelerator simplifies organizational processes needed to support these monitoring and tracking mandates and to allow for cardholder data security. The Accelerator report templates can be customized to directly reflect specific organizational and regulatory requirements. You can access these templates using the tabs provided:

- PCI Data Security Standard overview
- Plan and Organize
- PCI Req. 10: Track and Monitor Access
- PCI Req. 11: Regularly Test and Validate
- PCI Policy Violations Monitoring

Other tools in the Guardium family of solutions available to assist in meeting regulations include the following:

- **Cardholder Database Access Map** - A graphical map of access between cardholder database access clients and servers. This map provides an at-a-glance view of activities by access type, content, and frequency. To open the Access Map builder and viewer, select View > Access Map > Access Map builder.
- **PCI Compliance Security Assessments** - A detailed view of database access security health used to automate the compliance processes with continuous real-time snapshots customized for user defined tests, weights, and assessments. The security assessment acts as a "report card" to help track progress on addressing addressing database vulnerabilities. To create a security assessment, select Assess/Harden > Vulnerability Assessment > Assessment builder
- **Full Audit Trail** - The non-intrusive generation of a full audit trail for data usage and modifications required by regulatory compliance. This capability is located under the Monitor/Audit tab.
- **Automated Scheduling** - Automated scheduling of PCI work flows, audit tasks, and distribution of information to responsible parties across the organization. This functionality is located under the Comply tab.

PCI Data Security Standard

The Payment Card Industry (PCI) Data Security Standard offers a single approach to safeguarding sensitive data for all credit card brands. This standard is the result of collaboration between Visa and MasterCard, with the objective of creating common industry security requirements. Other card companies operating in the U.S. have also endorsed the PCI Data Security Standard within their respective programs.

The PCI Data Security Standard delivers a framework of tools and measurements needed to protect against cardholder data exposure and compromise across the entire payment industry. It applies to all members, merchants, and service providers that store, process, or transmit cardholder data utilizing any payment channels, including retail (brick-and-mortar), mail/telephone order, and e-commerce.

PCI Compliance Validation

Separate and distinct from the mandate to comply with PCI requirements is the validation of compliance. The validation process is a fundamental and critical function that identifies and corrects vulnerabilities, and protects customers by ensuring that appropriate levels of cardholder information security are maintained. Card vendors have prioritized and defined levels of PCI compliance validation based on the volume of transactions, the potential risk, and exposure introduced into the systems by merchants and service providers. These include:

- Internal control
- Ongoing assessment (i.e., governance, measurement, and recordkeeping)
- Disclosure (i.e., investigation, reporting, and certification)

計画と編成

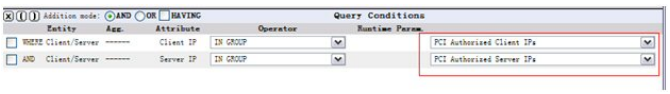
計画と編成

「概要」をクリックして、事前定義レポートがコンプライアンスにどのように準拠しているのかを示す概要を表示します。

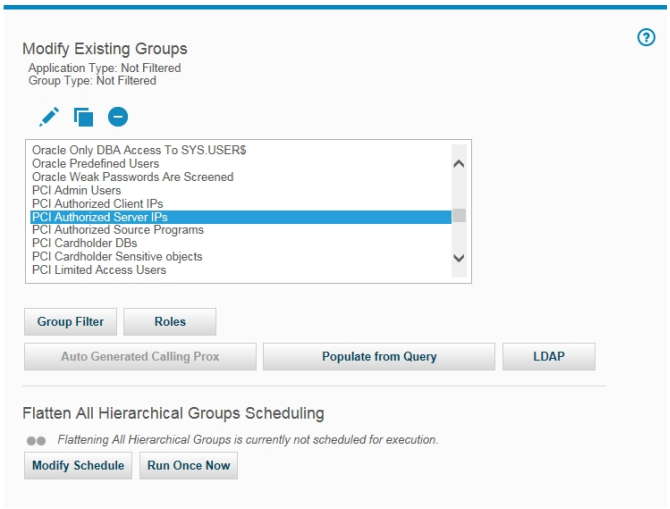
1. カード所有者サーバー IP リスト: カード所有者情報データベース・サーバーのリスト。会社の実際の状態に従って、「PCI 許可されたサーバー IP」グループの情報を設定してください。この情報により、カード所有者の情報を保管するデータベース・サーバーが指定されます。
2. カード所有者データベース: カード所有者情報データベース。「PCI カード所有者 DB: 指定済み」グループの情報を設定してください。この情報は、データベースのカード所有者情報に保管されます。
3. カード所有者オブジェクト: カード所有者情報オブジェクト。これは、PCI カード所有者の機密オブジェクトを設定する必要があります。
4. DB クライアントからサーバーへのマップ: クライアント/サーバー・マッピングの「PCI 許可されたサーバー IP」により、カード所有者情報を保管するデータベース・サーバーを指定するグループ情報が設定されます。照会を使用して、カード所有者データベースへのクライアント・アクセスを検出することができます。
5. アクティブ DB ユーザー: ユーザーのカテゴリーのほかに、カード所有者データベースにアクセスした管理者を示します。「PCI 許可されたサーバー IP」と「PCI 管理ユーザー」を設定してください。
6. カード所有者 DB 管理: カード所有者データベースの管理操作。「許可されたサーバー IP」と「PCI 管理ユーザー」を設定します。
7. 許可されたソース・プログラム: クレジット・プログラム・アクセス。「PCI 許可されたサーバー IP」と「PCI 許可されたソース・プログラム」を設定してください。クレジット・カード所有者データベース・アクセスを記録するためのプロシージャーです。
8. 無許可アプリケーション・アクセス: 非クレジット・プログラム・アクセス。「PCI 許可されたサーバー IP」と「PCI 許可されたソース・プログラム」を設定してください。カード所有者データベース・アクセスに関するクレジット・プログラムのレコードです。
9. 8.5.8 共有アカウント: コンピューターへのアクセス権限を持つ各ユーザーに固有の ID を割り当てるための、PCI の 8 番目の要件。同じデータベース・ユーザー名がカード所有者データベース IP からアクセスしようとしている回数をカウントするために、「PCI 許可されたサーバー IP」を設定します。

ステートメントで、クリックしてレポート書式を表示し、入力する必要がある特定のグループ・コンテンツを判別します。

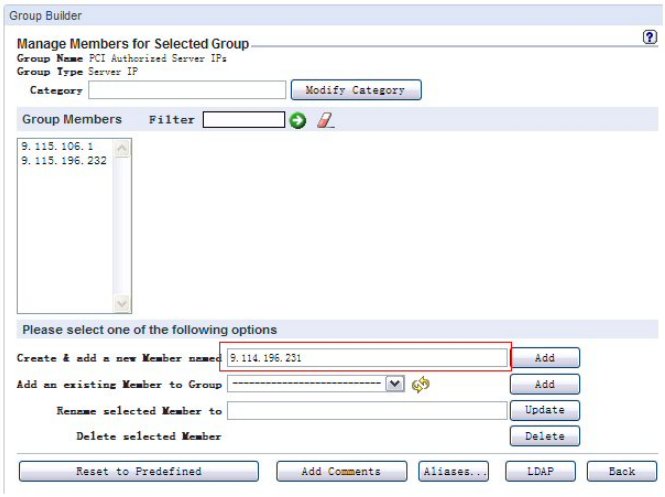
以下に実際のグループの名前を示します。



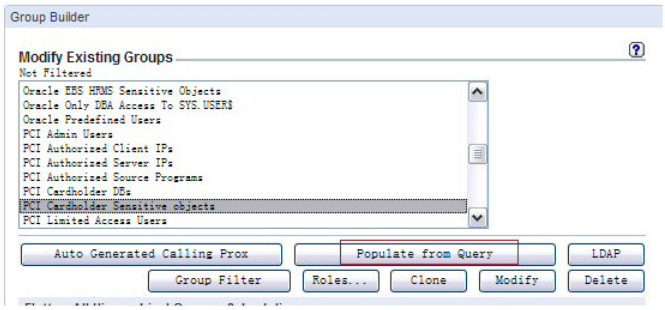
「設定」 > 「ツールとビュー」 > 「グループ・ビルダー」にナビゲートし、「既存グループの変更」選択項目でグループ名を選択します。



「変更」(鉛筆アイコン) をクリックして、「選択したグループのメンバーの管理」ページに移動します。新規メンバーを追加します。



カスタマイズされた照会を使用してグループを入力することもできます。



PCI 要件 10 トラッキングとモニター

「概要」をクリックして、Guardium モニターと事前定義レポートがコンプライアンスにどのように準拠しているのかを示す概要を表示します。

1. 10.2 および 10.3 自動化 - オンライン・ヘルプの保護ヘルプ・ブックと順守ヘルプ・ブックを使用して、このセクションを自動化します。
2. 10.2.1 データ・アクセス - カード所有者データへの PCI アクセス。「PCI 許可されたサーバー IP」と「PCI 管理ユーザー」を設定します。
3. 10.2.2 管理アクティビティー - 管理ユーザーによる PCI アクティビティー。「PCI 許可されたサーバー IP」と「PCI 管理ユーザー」を設定します。
4. 10.2.3 監査証跡アクセス - このセクションの手順を完全に実行するには、少なくとも「SQLGuard へのログイン (Logins to SQLGuard)」、「SQLGuard サーバー上のユーザー・アクティビティーの監査証跡 (User activity audit trails on SQLGuard server)」、「スケジュールされたジョブの例外」、「ユーザー To-do リスト」の 4 種類のレポートを定義する必要があります。「調査」 > 「照会 - レポート・ビルダー」 にナビゲートして、必要なレポートを作成します。
5. 10.2.4 無効なアクセス - PCI - 無効なログイン・アクセス試行: 失敗したログイン試行をデータベースに記録します。PCI - 無許可アプリケーション・アクセス: 「PCI 許可されたソース・プログラム」で定義されていないデータベース・アクセスを記録します。
6. 「10.2.6 初期化ログ」、「10.5 セキュア監査証跡」、および「10.6 アクセスの監査」の 3 つのセクションでは、組み込みオンライン・ヘルプのモニターおよび監査ヘルプ・ブックを使用することもできます。

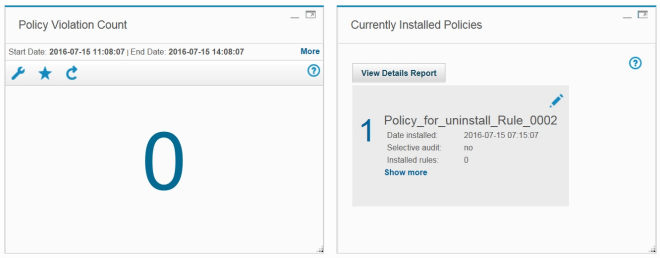
PCI 要件 11 継続的な検証

「概要」をクリックすると、脆弱性アセスメントの重要性に関する説明が表示されます。アセスメント・プロセスを作成するには、「強化」 > 「アセスメント・ビルダー」をクリックします。

PCI ポリシー・モニター

「概要」をクリックすると、ポリシーについての概要が表示されます。

1. 現在のポリシー・インストールを表示するには、「設定」 > 「ツールとビュー」 > 「ポリシー・インストール」にナビゲートし、インストールに適したポリシーを選択します。



2. ポリシー違反 - 違反操作のレコード。

親トピック: モニターおよび監査

ワークフロー・ビルダー

ワークフロー・ビルダーは、監査プロセスで使用する、カスタマイズされたワークフロー (ステップ、移行、およびアクション) を定義するために使用します。

追加情報は、『監査プロセスの作成』を参照してください。以下の手順に従います。

- ワークフロー・ステップを定義します (イベント状況)。
- ステップを追って移行のフローを定義します (アクション)。
- サインオフを要求するアクションを定義します。
- 各状況にロールを割り当てて、 各状況の確認を許可されるユーザーを定義します。

この機能の関連用語

イベント・タイプ - カスタム・ワークフロー

イベント状況 - ワークフローの状態/状況。

イベント・アクション - アクション/移行

注: ワークフロー・ビルダーは、プロダクト・キーによって使用可能になるオプション・コンポーネントです。

ワークフロー・プロセスの作成

1. 管理アカウントを使用して、「順守」 > 「ツールとビュー」 > 「ワークフロー・ビルダー」にナビゲートして、「ワークフロー・ビルダー」を開きます。DataPrivacy 特権を持つユーザー・アカウントを使用して、「アクセラレーター」 > 「データ・プライバシー」 > 「トラッキングとモニター」 > 「監査証跡およびワークフローの自動化」にナビゲートして、「ワークフロー・ビルダー」を開きます。
2. 最初の画面 (「イベント・タイプ」) で、「イベント状況」をクリックして「イベント状況」構成に移動します。
3. 「イベント状況の追加」 をクリックして、新規イベント状況を定義します。複数のイベント状況が予期されています。状況の定義を入力し、 ワークフロー内で最後となるタスクについては、「 最終」チェック・ボックスにチェック・マークを付けます。
4. 「イベント・タイプ」をクリックし、「イベント・タイプ定義の追加」で「追加」をクリックして、新規イベント・タイプを定義します。
5. 定義を入力し、ワークフロー内の最初のタスクを 指定します。
6. 次に、状況項目を強調表示し、「選択可能な状況」リストと 「許可される状況」リストの間にある「>」ボタンをクリックして、「選択可能な状況 」リストから、そのワークフローの 許可される状況 をすべて選択します。
7. 終了したら、「保存」ボタンをクリックします。注: 「保存」ボタン (または「キャンセル」ボタン) は、名前、デフォルトのイベント、 または使用可能なイベントに行った変更にのみ適用されます。
8. 「イベント・タイプ」メニュー画面の「 定義済みイベント・アクション」に進みます。「定義済みイベント・アクション」では、 そのワークフローの個々のイベント・アクションの指定を行います。
9. 「新規」 ボタンをクリックします。
10. 「イベント・アクションの記述」に入力し、「前の状況」、「次の状況」、およびこのイベント・アクションでサインオフが必要かどうかを指定します。「適用」ボタンをクリックします。
11. すべてのイベント・アクションの記述と指定を行うまで、ステップ 9 と 10 を繰り返します。
12. 「イベント・タイプ」 メニュー画面の「ロール」セクションに 進みます。「ロール」では、 イベントが特定のイベント・アクションにある場合に、そのイベントを確認できる人を定義します。例えば、「検討中」のイベントを確認できる人や 「承認済み」のイベントを確認できる人です。
13. イベント・タイプ状況を選択し、 「ロール」ボタンをクリックします。
14. 「セキュリティー・ロールの割り当て」パネルで、割り当てるすべてのロールにマークを付けます (自分のアカウントに 割り当てられたロールのみが表示されます)。「適用」をクリックして、セキュリティー・ロールの 選択を保存します。「戻る」ボタンをクリックします。
15. すべてのイベント・タイプ状況でロールを 定義するまで、ステップ 13 から 14 を繰り返します。
16. ワークフロー・ビルダーからの構成が 終了しました。
17. 「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」にナビゲートして「監査プロセス・ビルダー」を開き、ワークフローをスケジュールし、ワークフロー・レポートを作成および表示します。『レポート・タスクの定義』の下にある監査プロセス・ビルダーのステップを参照してください。

付録に、使用法のシナリオ『ワークフロー・ビルダーの ワークフロー例』があります。

注: 「監査プロセス・ビルダー」のタスク・タイプが「分類プロセス」である場合は、ワークフロー・ビルダーでカスタマイズ・ワークフローを作成することはできません。

警告: ワークフロー・イベントが作成された際には、 そのイベントが使用するすべての状況にロールを割り当てることができます (つまり、 その状況にある場合、イベントはこのロールからのみ参照可能になります)。 イベントを監査プロセスに割り当てる際には、このイベントの状況に割り当てられた すべてのロールに、この監査プロセスの受信者がいることが重要です。 そうでないと、監査結果行が、この行を参照したり、状況を変更したりできる受信者が いない状況に置かれる可能性があります。

監査行がアクセス不能になった場合、admin ユーザー (ロールに関係なくすべてのイベントを表示可能) はこの行を表示し、その状況を変更できます。ただし、データ・レベル・セキュリティーがオンになっていると、 admin ユーザーがこの行を参照できない 可能性があります。admin ユーザーは、 (「グローバル・プロファイル」から) データ・レベル・セキュリティーをオフにするか、 dataset_exempt ロールを保持することが必要になります。 監査プロセスは、その監査プロセスに関連するイベントに対処する必要があるすべてのロールが受信者となるように構成することが重要です。

注: イベント状況の削除は、その状況が何らかのイベントの最初または最後の状況ではなく、アクションに 使用されていない場合にのみ許可されます。検証では、状況の削除を防止するイベント/アクションのリストが提供されます。

限定された数のレコードのみへのデフォルト・イベントの追加

監査プロセス・レポート・タスクの実行中、このプロセス・タスクの結果は、表 REPORT_RESULT_DATA_ROW に保存されます。この表は、レポートのすべての行について、1 行を持ちます。このレポート・タスクにデフォルトのイベントが 割り当てられている場合には、そのレポートのすべての行について、表 TASK_RESULT_ADDITIONAL_INFO に 行が追加されます。デフォルトのイベントが大規模な結果に対して使用される場合は、 これによってディスク・スペースの問題が発生する可能性があります。限られた数のレコードを持つ タスク結果でのみイベントを作成します。そうしないと、多数のレコードを管理することが できません。デフォルトのイベントが意図したとおりの、制限された方法で 使用される場合は、ディスク・スペースの問題もユーザビリティーの問題も発生しません。 数千個のイベントをクローズすることは容易ではありません。

- カスタマイズ・ワークフローの作成方法
  特定のカスタマー・ステップ、移行、およびアクションで構成されるカスタマイズ・ワークフローを、 監査プロセスで使用されるように定義します。
- カスタマイズしたワークフローの使用方法
  カスタマイズした顧客のワークフロー慣行に応じた監査プロセスを定義します。顧客固有の監査プロセスや手法を Guardium® ソリューションに組み入れます。

親トピック: モニターおよび監査

カスタマイズ・ワークフローの作成方法

特定のカスタマー・ステップ、移行、およびアクションで構成されるカスタマイズ・ワークフローを、 監査プロセスで使用されるように定義します。

このタスクについて

ユーザーの特定の業務に基づいてワークフローの定義および 管理を行います。

このコンポーネントの概要については、『ワークフロー・ビルダー』を 参照してください。

前提条件

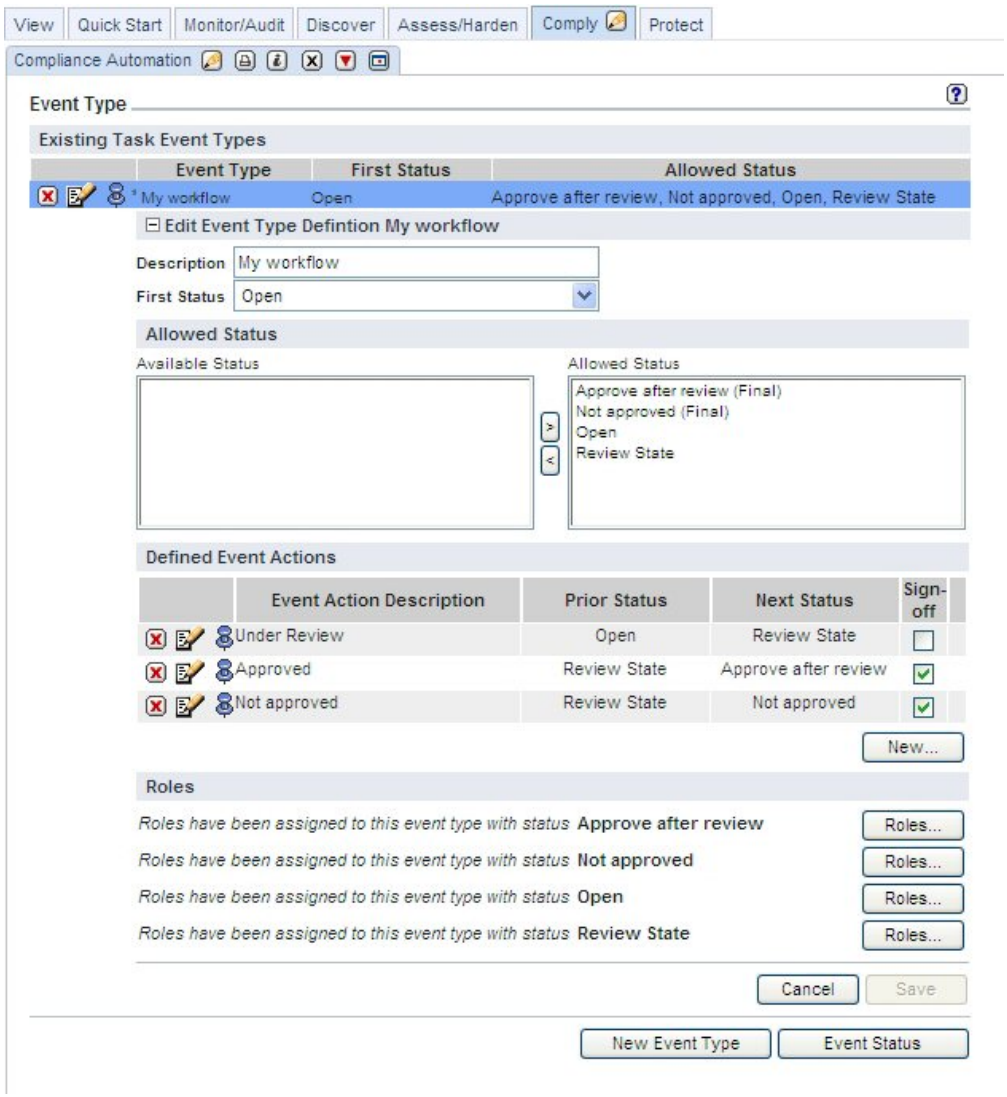
- 『監査ワークフローの作成方法』を参照してください。詳しくは、 『コンプライアンス・ワークフローの自動化』を参照してください。
- このカスタマイズ・ワークフローを作成した後、『 カスタマイズ・ワークフローと監査ワークフローの結合方法』を参照してください。

手順

1. 「順守」 > 「ツールとビュー」 > 「ワークフロー・ビルダー」にナビゲートして「ワークフロー・ビルダー」を開きます。
2. 最初の画面 (「イベント・タイプ」) で、「イベント状況」ボタンをクリックして「イベント状況」構成に移動します。
3. 「イベント状況の追加」をクリックして、新規イベント状況 を定義します。複数のイベント状況が予期されています。状況の定義を入力し、 ワークフロー内で最後となるタスクについては、「 最終」チェック・ボックスにチェック・マークを付けます。終了したら、次のステップに進みます。

   3 ステップの簡単なワークフローの例: オープン、状態の検討、承認または非承認。ワークフローの各ステップは、 個別の定義済みタスク・イベント状況です。

   例のワークフロー・タスク: オープン、 状態の検討、検討後に承認、または非承認。また、タスクがワークフロー内で 最後となる場合は、「最終」列にチェック・マークを付けます。 この例では、最後のタスクの例は「承認済み」または「非承認」です。



4. 「イベント・タイプ」ボタンをクリックし、「イベント・タイプ定義の追加」の「追加」ボタンをクリックして、新規イベント・タイプ を定義します。
5. 定義を入力し、ワークフロー内の最初のタスクを 指定します。
6. 次に、状況項目を強調表示し、「選択可能な状況」リストと 「許可される状況」リストの間にある「>」ボタンをクリックして、 「選択可能な状況」リストから、そのワークフローの許可される 状況をすべて選択します。
7. 終了したら、「保存」ボタンをクリックします。
8. 「イベント・タイプ」メニュー画面の「 定義済みイベント・アクション」に進みます。「定義済みイベント・アクション」では、 そのワークフローの個々のイベント・アクションの指定を行います。
9. 「新規」 ボタンをクリックします。

   3 ステップの簡単なワークフローの例で説明すると、 イベント・アクション「検討中」では、前の状況が「オープン」で、次の状況が「検討の状態」になります。「検討中」に続くイベント・アクションは「承認済み」で、前の状況が「検討の状態」、次の状況が「検討後に承認」になります。そうでなく、「非承認」のイベント・アクションが続く場合は、前の状況が「検討の状態」で、次の状況が「非承認」になります。また、イベント・アクションごとに指定されたレビューアーの、サインオフ機能があります (継続的または順次)。 前のスクリーン・ショットを参照してください。

10. 「イベント・アクションの記述」に入力し、「前の状況」、「次の状況」、およびこのイベント・アクションでサインオフが必要かどうかを指定します。「適用」ボタンをクリックします。
11. すべてのイベント・アクションの記述と指定を行うまで、ステップ 9 と 10 を繰り返します。
12. 「イベント・タイプ」 メニュー画面の「ロール」セクションに 進みます。「ロール」では、 イベントが特定のイベント・アクションにある場合に、そのイベントを確認できる人を定義します。例えば、「検討中」のイベントを確認できる人や 「承認済み」のイベントを確認できる人です。
13. イベント・タイプ状況を選択し、 「ロール」ボタンをクリックします。
14. 「セキュリティー・ロールの割り当て」パネルで、割り当てるすべてのロールにマークを付けます (自分のアカウントに 割り当てられたロールのみが表示されます)。「適用」をクリックして、セキュリティー・ロールの 選択を保存します。「戻る」ボタンをクリックします。
15. すべてのイベント・タイプ状況でロールを 定義するまで、ステップ 13 から 14 を繰り返します。
16. ワークフロー・ビルダーからの構成が 終了しました。
17. 「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」にナビゲートして「監査プロセス・ビルダー」を開き、ワークフローをスケジュールし、ワークフロー・レポートを作成および表示します。『レポート・タスクの定義』の下にある監査プロセス・ビルダーのステップを参照してください。

親トピック: ワークフロー・ビルダー

カスタマイズしたワークフローの使用方法

カスタマイズした顧客のワークフロー慣行に応じた監査プロセスを定義します。顧客固有の監査プロセスや手法を Guardium® ソリューションに組み入れます。

このタスクについて

Guardium 監査ワークフロー・プロセスにおけるカスタマイズ・ワークフロー

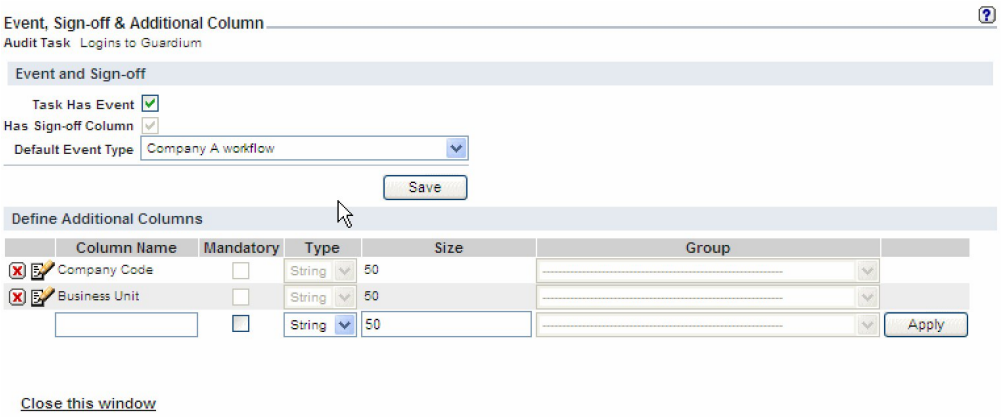
ワークフロー・ビルダーで作成されるイベント・タイプの 正式な順序の管理は、「監査タスク」ウィンドウの「イベントおよび追加列」ボタンをクリックして行います。このボタンは、監査タスクを作成して保存すると表示されます。この追加のボタンは、監査タスクを保存しないと表示されません。

前提条件

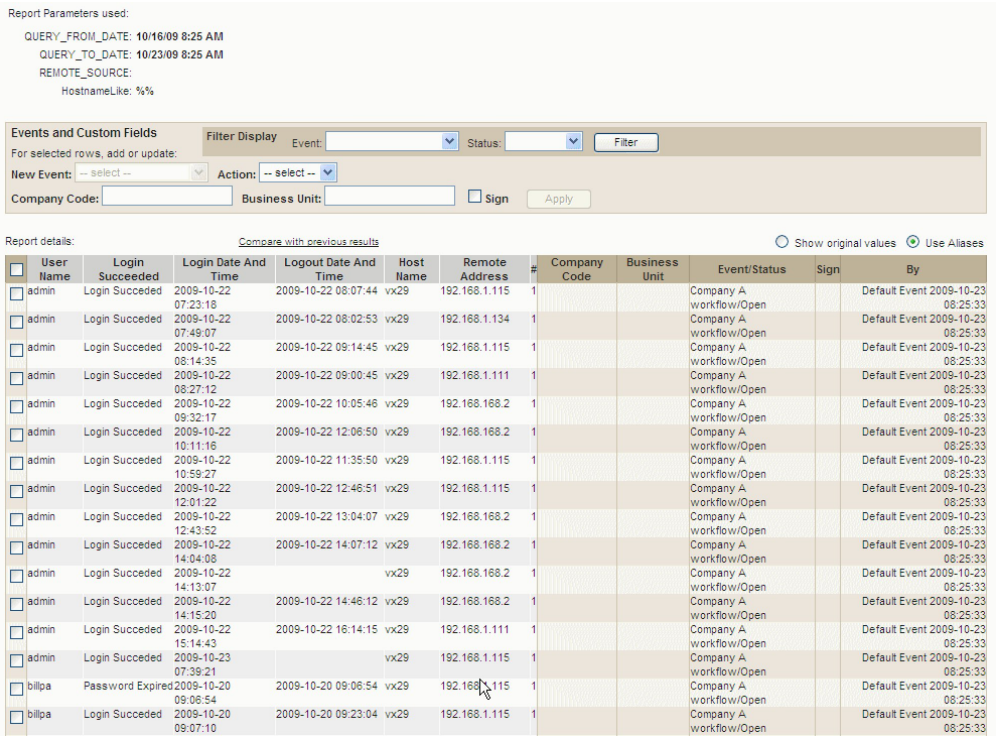
- カスタマイズしたワークフローの作成方法を参照。追加情報はワークフロー・ビルダーを参照してください。
- 『監査ワークフローの作成方法』を参照してください。詳しくは、『コンプライアンス・ワークフローの自動化』を参照してください。
- 以下の追加ステップに従い、カスタマイズした顧客のワークフロー慣行に応じた監査プロセスを定義します。

手順

1. 以下の手順で、監査タスクを追加する際にこれらのワークフロー・アクティビティーを構成します。
2. 監査タスクを作成して保存します。保存すると、追加のボタンの「イベントおよび追加列」が表示されます。
3. この追加のボタンをクリックします。



4. 次の画面で、「イベントおよびサインオフ」ボックスにチェック・マークを付けます。 ワークフロー・ビルダーで作成したワークフローが「イベントおよびサインオフ」の選択項目として表示されます。
5. この選択項目を強調表示します。選択を保存します。
6. 追加の情報 (会社コード、ビジネス・ユニット・ラベルなど) が、ワークフロー・レポートの 一部として必要である場合は、この情報を画面の「追加列」セクションに追加して、「適用」(保存) をクリックします。 完了したら、このウィンドウを閉じます。
7. 監査タスクを適用 (保存) します。監査プロセス定義全体を適用 (保存) します。 「今すぐ 1 回実行」をクリックして、レポートを作成します。「表示」をクリックすると、レポートが表示されます。
8. 「今すぐ 1 回実行」をクリックして、レポートを作成します。「表示」をクリックすると、レポートが表示されます。



この「イベントおよび追加列」ボタンは、すべての監査タスクで表示されます。

注:

監視データ・レベルでのデータ・レベル・セキュリティーが有効な場合 (「グローバル・プロファイル」設定を参照)、ユーザーに対してユーザーのデータベース情報のみが表示されるよう、監査プロセスの出力がフィルタリングされます。

「監査タスクの追加」内の「レポート」の選択項目は、「未処理のイベント」および「イベント状況の移行」の 2 つのプロシージャー型レポートです。これら 2 つのレポートを 2 つの新しい監査タスクに追加することで、すべてのワークフロー・イベントと移行の詳細を表示します。これら 2 つのレポートは、フィルター処理されません (監視データ・レベルのセキュリティー・フィルターは適用されません)。これら 2 つのレポートは、admin ユーザーおよび admin ロールを持つユーザー に対するレポート・リストでのみ、デフォルトで選択可能です。

親トピック: ワークフロー・ビルダー

脅威検出分析

Guardium には、監査済みデータをスキャンおよび分析して、さまざまなタイプのデータベース攻撃を示す可能性のある徴候を検出するための特殊な脅威検出分析が組み込まれています。

脅威検出分析では、監査済みデータをスキャンおよび分析して、SQL インジェクションまたはストアード・プロシージャーによるデータベース攻撃を示す可能性がある徴候を検出します。Guardium は、常に変化するアタック・シグニチャーのディクショナリーとの比較には依存しません。代わりに、Guardium は、長期間にわたって監査データのアクティビティー、例外、および異常値データ (Outliers Detection) を分析して、攻撃を示すパターンを探します。疑わしいイベントを一定期間にわたり追跡し、イベントを相互に関連付けることによって、Guardium は潜在的なリスクを包括的に示します。この手法はより柔軟かつ包括的であり、シグニチャーを継続的に更新する必要がありません。

脅威検出分析は、MySQL、Oracle、および Db2 でサポートされています。

- SQL インジェクション攻撃の特性
- ストアード・プロシージャー攻撃の特性
- 脅威検出分析の有効化と無効化
  このトピックでは、脅威検出分析を有効化する際の前提条件と手順について説明します。
- ケース・レポートの操作
  このトピックでは、ケース・レポートの操作について説明します。
- 脅威分析の監査プロセス・ワークフローのアクティブ化
  この手順では、監査プロセスをスケジュールして、疑わしいストアード・プロシージャー・ケースおよび疑わしい SQL インジェクション・ケースの脅威分析結果を配布する方法を説明します。
- 脅威診断ダッシュボードの操作
  「疑わしい悪意のある STP ケース」(ストアード・プロシージャー) または「疑わしい SQL インジェクション攻撃」レポートの特定の脅威ケースから呼び出されるダッシュボードは、脅威診断ダッシュボードと呼ばれます。
- 脅威検出分析機能

親トピック: モニターおよび監査

SQL インジェクション攻撃の特性

SQL インジェクション攻撃は、ユーザーの入力と SQL 照会を連結することで、Web アプリケーションの脆弱性を悪用しようと試みます。これが成功すると、その攻撃で正当な Web アプリケーションの接続を使用して、悪意のある SQL コマンドを実行できるようになります。SQL インジェクション攻撃は識別が困難である場合があります。これは、攻撃の個々のステップを他のステップから独立して分析した場合、そのステップが正当であると見なされる可能性があるためです。Guardium は脅威検出分析を使用して、個々のステップを取り込み、それらのステップを 1 つの複雑な攻撃の一部として分析することで、潜在的な SQL インジェクション攻撃を識別します。

Guardium が識別する SQL インジェクション攻撃の典型的な徴候には、以下があります。

- 動的 SQL 照会の構造 (照会対象の列の数など) を識別しようとする攻撃者
- 極端に大量の新規照会 (具体的には、独特に構造化された照会、または通常とは異なる方法で構造化された照会)
- データベース構造に関する情報を格納する表へのアクセス

親トピック: 脅威検出分析

ストアード・プロシージャー攻撃の特性

悪意のあるストアード・プロシージャーとは、検出を免れ、一定の期間にわたって複雑な攻撃を実行するように設計されたコード・ブロックのことです。完全に同じ攻撃を繰り返すことも、時間とともにその特性が変化することもあります。ストアード・プロシージャーは長期間にわたって休止する場合があります。その場合、疑わしい対象として識別するのは、より困難になります。前の監査で通常とは異なるアクティビティーに気付いたとしても、次の監査が行われるまでには、以前のアクティビティーは忘れられてしまうためです。悪意のあるストアード・プロシージャーを使用することで、重要なテーブルが除去されたことを隠ぺいしたり、テーブルの内容を抽出したりすることができます。

疑わしいアクティビティーの例としては、機密オブジェクトでの DROP ステートメントを使用したストアード・プロシージャーの作成、DROP 動詞、欠落オブジェクトによる SQL 例外、長期間の休止後に変更されたプロシージャーなどが挙げられます。

Guardium は、個々のストアード・プロシージャーに関連するアクティビティーを追跡するとともに、異常値マイニング・データを使用して、各種の徴候とユーザーとを相互に関連付けます。Guardium では、この悪意のあるストアード・プロシージャー・ユース・ケースのこれらの典型的な徴候を検出できます (典型的な発生順で記載されています)。

1. データベース管理者が、顧客テーブルからデータを削除する、悪意のあるプロシージャー A を作成する。
2. 1 カ月後、データベース管理者が、プロシージャー A を呼び出すために共通して使用されているプロシージャー B を変更する。
3. 別のユーザーが変更後のプロシージャー B を呼び出し、結果としてその罪のないユーザーによって顧客テーブルのデータが削除される。

親トピック: 脅威検出分析

脅威検出分析の有効化と無効化

このトピックでは、脅威検出分析を有効化する際の前提条件と手順について説明します。

脅威検出分析は、Guardium バージョン 10.1.4 以上ではデフォルトで有効になっています。

脅威検出分析を有効化するには、以下を行います。

- 検索に必要な最小限のメモリーおよびストレージ要件 (4 CPU および 24 GB RAM) を満たしていることを確認します。
- システムに、ログに記録されたアプリケーション・データがあることを確認します。具体的には、インジェクションはアプリケーションから開始されるため、SQLI にはアプリケーション・データが必要です。システムがアプリケーションを「信頼」して、Guardium でモニターしていなければ、インジェクションを識別することはできません。
- 異常値検出は、SQL インジェクションの脅威検出には必要になりませんが、疑わしいストアード・プロシージャーの検出を完全にサポートするために必要です。詳しくは、異常値検出の有効化と無効化を参照してください。
- Guardium API コマンド `grdapi enable_advanced_threat_scanning` を使用して、各コレクターまたは中央マネージャーの複数の管理対象ユニットでの脅威検出スキャンを有効にします。enable_advanced_threat_scanning コマンドで使用可能なパラメーターについて詳しくは、『GuardAPI 脅威検出分析機能』を参照してください。
- ケース・レポートを関連する調査ユーザーに送信するための監査プロセスをセットアップします。これはオプションですが、推奨されています。詳しくは、脅威分析の監査プロセス・ワークフローのアクティブ化を参照してください。

重要: 脅威検出は、ログに記録されたデータの分析と相関に依存します。したがって、ログに記録する前にフィルターでトラフィックを除外するルールは、脅威検出では考慮されません。「S-TAP セッションを無視」ルールの使用を慎重に検討して、コレクターの容量を最適化する代わりにこれらのセッションがログに記録されなくなるリスクを判断してください。

脅威検出分析を無効にするには、個々のコレクターまたは中央マネージャーに対してコマンド disable_advanced_threat_scanning を使用します。

悪意のあるストアード・プロシージャーを分析する際の前提条件は以下のとおりです。

- 分析アルゴリズムは、部分的に機密オブジェクトのグループに依存します。 デフォルトでは、アルゴリズムはシステム定義の機密オブジェクト・グループ (グループ ID 5) のメンバーを使用します。異常値検出に別の機密オブジェクト・グループを既に指定している場合、脅威検出でもそれと同じグループが使用されます。異常値検出が有効にされていないとしても、同じ GuardAPI コマンド `set_outliers_detection_parameter parameter_name="sensitiveObjectGroupIds" parameter_value=<group ID>,<group ID>,...` を使用して独自の機密オブジェクト・グループを設定できます。
- 悪意のあるストアード・プロシージャー分析に必要なトラフィックを収集するためのポリシー・ルールをインストールする必要があります。
  推奨: ポリシー内に、以下のルールを推奨されている順で作成してください。重要な点として、これらのすべてのルールについて「次のルールに進む」チェック・ボックスにチェック・マークを付ける必要があります。
  1. アクセス・ルール: 「全詳細をロギング」(コマンド・グループ・フィルターが PROCEDURE DDL の場合)。
  2. アクセス・ルール: 「全詳細をロギング」(コマンド・グループ・フィルターが EXECUTE コマンドの場合)。使用しているデータベースが Oracle の場合は、コマンド BEGIN をルールに含めます。
  3. 例外ルール: 「ロギングのみ」(エラー・タイプ・フィルターが SQL_ERROR の場合)。

親トピック: 脅威検出分析

ケース・レポートの操作

このトピックでは、ケース・レポートの操作について説明します。

Guardium は一定期間にわたって徴候を分析し、それらを相互に関連付け、識別された潜在的な攻撃ごとにスコアを割り当てます。攻撃である可能性が高いことをスコアが示す場合、一連のイベントが 1 つのケースになり、そのケースの ID はコレクターごとに固有になります。これらのケースは、疑わしい攻撃ごとにケース・レポートで外部化されます。 ケース・レポートには、以下のいずれかの方法でアクセスします。

- 中央マネージャーの To Do リストで通知を受け取るように監査プロセスをセットアップし、関連するコレクターで直接レポートを開きます。To Do リストは 1 時間に 1 回更新されることに注意してください。
- 「調査」 > 「例外」にアクセスします。

ケース・レポートのウィンドウには、デフォルトでは、1 つのレポートで 1 行ごとに 1 件のインシデントが最大 3 件表示されます。各ケースには、1 から 3 までのリスク・スコアが含まれます (3 が最も重大です)。以下の操作が可能です。

- 攻撃の要約を表示するには、ケース ID の上にカーソルを移動します (ストアード・プロシージャーのケースのみ)。
- 詳細な徴候レポートにアクセスするには、ケース ID の上にカーソルを移動して、「徴候にリンクします」をクリックします。
- ケース固有の脅威診断ダッシュボードを開くには、ケース ID をクリックします。脅威診断ダッシュボードの操作を参照してください。

制約事項: ケース・レポートには以下の制約事項があります。

- データ・レベル・セキュリティーはありません。
- これらのレポートを複製することはできません。
- ケース・レポートの配布レポートを作成することはできますが、中央マネージャーでは、ケース・レポートから脅威診断ダッシュボードに直接リンクすることはできません。また、追加のホバー・ヘルプや徴候へのリンクもありません。

親トピック: 脅威検出分析

脅威分析の監査プロセス・ワークフローのアクティブ化

この手順では、監査プロセスをスケジュールして、疑わしいストアード・プロシージャー・ケースおよび疑わしい SQL インジェクション・ケースの脅威分析結果を配布する方法を説明します。

このタスクについて

適切なレビューアーへの脅威分析レポートの配布を制御する 2 つの監査プロセスが事前構成されています。

- 疑わしい悪意のある STP ケース (ストアード・プロシージャーのケース)

- 疑わしい SQL インジェクション・ケース

プロセスごとに 1 つの攻撃タイプに関して、疑わしいケースをプルします。これらのプロセスをカスタマイズすることも、プロセスをコピーして独自のプロセスを作成することもできます。

手順

1. 「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」にナビゲートします。オプションで、「非アクティブのみ」ラジオ・ボタンをクリックするか、「フィルター」ボックスに `Suspected` と入力することで、使用可能な監査プロセスをフィルタリングできます。

   このプロセスのデフォルトのタスクは、対応するレポート (「疑わしい悪意のある STP ケース」または「疑わしい SQL インジェクション・ケース」) です。これらのレポートのランタイム・パラメーターは変更しないでください。ただし、この同じ監査プロセスに、他のタスクを追加することはできます。例えば、両方の脅威レポートを 1 つの監査プロセスに追加できます。

   これらの監査プロセスを中央マネージャーから定義している場合、脅威データを確認する対象のコレクターごとにタスクを定義し、「リモート・データ・ソース」オプションを使用します。

2. 「結果の送信」をクリックして、監査プロセスの受信者を定義します。定義した受信者が、疑わしい悪意のあるストアード・プロシージャーに関するレポートを受け取ることになります。
3. デフォルトの受信者 (「ユーザー」) を選択してから アイコンをクリックして、組織に応じて適切な受信者 (複数可) を定義します。完了したら、「OK」をクリックします。
4. 「監査プロセスのスケジュール」」をクリックし、監査プロセスのスケジュールをレビューします。

   このプロセスを毎日、午前 12 時 30 分 (異常値検出と脅威検出の両方の通常の実行後) から 1 時間に 1 回実行することをお勧めします。このタスクには、「従属ジョブの自動実行」チェック・ボックスは適用されないことに注意してください。

   重要: 「スケジュールのアクティブ化」チェック・ボックスが選択されていることを確認します。
5. 「次へ」をクリックし、「保存」をクリックして監査プロセスの作業を完了します。

親トピック: 脅威検出分析

脅威診断ダッシュボードの操作

「疑わしい悪意のある STP ケース」 (ストアード・プロシージャー) または「疑わしい SQL インジェクション攻撃」レポートの特定の脅威ケースから呼び出されるダッシュボードは、脅威診断ダッシュボードと呼ばれます。

脅威診断ダッシュボードが実行する内容は他の調査ダッシュボードとほとんど同じです。ただし、異なる点として、当該ケースのダッシュボードに疑わしいイベント (データベース・ユーザー、サーバー、オブジェクトなど) のデータが取り込まれ、潜在的な攻撃を調査する際に役立つ、それらのイベントおよび周囲のイベントのさまざまなビューを提供する各種のグラフが使用されます。関連する検索および異常値データも、グラフと同じダッシュボードのページで使用できます。

多くの場合、事前定義されている脅威診断ダッシュボードの既存のフィルターは、いずれも変更する必要はありません。ただし、独自の比較分析を行う必要がある場合は、既存のフィルターを変更できます。

ダッシュボードとグラフ・フィルターの操作について詳しくは、調査ダッシュボードを参照してください。

ヒント: 脅威診断ダッシュボードは、関連する脅威レポートでケース番号をクリックすることによってのみ開くことができます。このダッシュボードや、その他すべての事前定義されたダッシュボードに変更を保存することはできません。変更した後のダッシュボードを維持して以降の調査でも使用できるようにするには、ダッシュボードをコピーし、新しい名前を付けて保存する必要があります。さらに、「フィルター」メニューをクリックし、「保存」を選択して、フィルターも保存する必要があります。

参照データとは、脅威検出分析専用に事前定義された、一連のグラフ固有のフィルターのことです。これらのフィルターにより、調査中のケースと類似するが、一般的なダッシュボード・フィルターでは含まれないデータが示されます。参照データをユーザーが変更することはできません。各グラフのフィルター・アイコンの上にカーソルを移動すると、参照データが表示されます。

通常の疑わしい悪意のあるストアード・プロシージャー攻撃では、この攻撃で脅威診断ダッシュボードがフィルタリングされ、以下の一般的なダッシュボード・フィルターが組み込まれます。

- サーバー: 8.34.223.145
- データベース・ユーザー: USER1
- データベース: 8.4.134.213:31.5.12
- データベース・タイプ: MYSQL
- オブジェクト: stp1_name

データベース・ユーザーに関するグラフには、同様のデータベース・ユーザー (USER2、USER3、USER4 など) の参照データが含まれます。これにより、一般的なダッシュボード・フィルターではこれらの追加のユーザーが含まれないとしても、疑わしいユーザーのアクティビティーを同様のユーザーと比較することができます。

関連する参照データはすべてのフィールドに含まれるわけではありません。参照フィルターが事前定義されていないフィールドはすべて、ダッシュボードでの場合と同じようにフィルタリングされます。

一部のグラフでは、ダッシュボード全体に選択されているフィルターにかかわらず、データを比較できるよう、フィルターを非アクティブにすることができます。こうすることにより、アクティビティーの状況をより包括的に捉えることができます。

フィルター・アイコンをクリックして「グラフ・フィルターの設定」を開き、変更を行います。

- SQL インジェクションの脅威の調査
- ストアード・プロシージャーの脅威の調査

親トピック: 脅威検出分析

SQL インジェクションの脅威の調査

このタスクについて

この手順では、脅威診断ダッシュボードを使用して疑わしい SQL インジェクション攻撃を調査する方法を説明します。

手順

1. To Do リストまたは「調査」 > 「例外」から、「疑わしい SQL インジェクション・ケース」ダッシュボードを開きます。各行が 1 つのケースを表し、攻撃の確実性に対する「信頼度 (%)」評価、および攻撃のリスク・レベルが示されます。
2. 誤検出を評価するために、「表示」をクリックします。選択したケース ID の上にカーソルを移動し、「徴候 (Symptoms)」をクリックして「SQL インジェクション・ケースの徴候」ページを開きます。すべての疑われるアクションが記述され、当該 SQL 文字列が表示されます。ユーザーが文字列に加えた変更そのものを確認できます。文字列ごとに確認していくことで、前の照会から返されたエラーを使用して攻撃者がさらに多くのデータを系統的に手に入れる仕組みを観察できます。
3. ID 番号をクリックして、SQL インジェクション攻撃のデフォルト診断ダッシュボードを開きます。このダッシュボードは、インシデントの日付と、疑わしい Web アプリケーション接続の詳細でフィルタリングされています。これにより、攻撃が行われている間に発生したデータベース・トラフィックに、調査の対象を絞り込めるようになっています。フィルターを変更または除去することで、調査の範囲を広げることもできます。グラフのデータに関する詳細を調べるには、下部にあるグリッドを使用します。標準のダッシュボードに移動すると、疑われる SQL インジェクション攻撃に固有のフィルターがすべて取り消されることに注意してください。
4. グラフを調査する際は、以下のガイドラインを使用してください。
   - 時間目盛りを変更して攻撃のピーク時を見つけます。
   - セキュリティー・ポリシーの違反を見つけて、攻撃時に特定の違反が他のアクティビティーと相関しているかどうかを確認します。
5. フィルター、時間フレームなどを変更してドリルダウンし、システム全体で違いがあるかどうかを確認します。
6. ダッシュボード内の以下のグラフを評価します。

   時間およびオブジェクトごとのアクティビティー数 (Activities count per time and object)
   : このグラフには、攻撃時に最も使用されたデータベース・オブジェクトが示されます。ダッシュボードの時間フレームを拡大することで、攻撃の前後でのアクティビティーの違いを比較できます。 特定のオブジェクトをフィルタリングするには、該当するセルをクリックします。色分けによって異なるオブジェクト名が示されます。

   時間およびエラーごとのエラー数 (Error count per time and error)
   : このグラフは、Web アプリケーションで生成された SQL エラーの数を示します。SQL エラー・レートが高い場合、それは、ある種の SQL インジェクション攻撃が行われている可能性があることを意味します。色分けによって異なるエラー・タイプが示されます。

   時間および異常値の理由ごとの異常値の数 (Outlier count per time and outlier reason)
   : SQL インジェクション攻撃には、通常の照会とは構造が異なる、大量の新規照会が伴います。これらの照会により、異常値が生成されます。このグラフを使用して、問題となっている Web アプリケーションで生成された異常値の量と範囲を確認します。

   時間および違反ごとの違反数 (Violations count per time and violation)
   : SQL インジェクション攻撃が行われている間、攻撃者は、無許可のオブジェクトに対するアクセスがログに記録されるセキュリティー・ポリシーの違反を行う可能性があります。攻撃のリスクを理解するには、違反の量とタイプを比較します。

   疑わしいエラー・タイプ
   : このグラフを使用して、SQL インジェクション攻撃で脆弱性を悪用するために使用されている特定の SQL エラーを調べます。特定のセルをクリックして検索をフィルタリングし、該当するエラーを生成した SQL ステートメントを調べます。注入された SQL コードに気付く場合があります。

   疑わしいオブジェクト名 (Suspicious object names)
   : このグラフを使用して、SQL インジェクション攻撃で使用される疑わしいオブジェクトを確認します。検索の時間フレームを拡大して、攻撃が開始される前に、これらのオブジェクトが使用されたかどうかを確認します。これらのオブジェクトの使用量を比較します。

親トピック: 脅威診断ダッシュボードの操作

ストアード・プロシージャーの脅威の調査

このタスクについて

この手順では、脅威診断ダッシュボードを使用して疑わしいストアード・プロシージャー攻撃を調査する方法を説明します。

手順

1. To Do リストまたは「調査」 > 「例外」から、「疑わしい悪意のある STP ケース」ダッシュボードを開きます。各行が 1 つのケースを表し、攻撃の確実性に対する「信頼度」評価、および攻撃のリスク・レベルが示されます。
2. 誤検出を評価するために、「表示」をクリックします。
3. 選択したケース ID の上にカーソルを移動すると、ケースの詳細が表示されます。
4. 徴候をクリックして、「悪意のある STP ケースの徴候」ページを開きます。
5. ID 番号をクリックして、SQL インジェクション攻撃のデフォルト診断ダッシュボードを開きます。このダッシュボードは、インシデントの日付と、疑わしい Web アプリケーション接続の詳細でフィルタリングされています。これにより、攻撃が行われている間に発生したデータベース・トラフィックに、調査の対象を絞り込めるようになっています。フィルターを変更または除去することで、調査の範囲を広げることもできます。グラフのデータに関する詳細を調べるには、下部にあるグリッドを使用します。
6. グラフを調査する際は、以下のガイドラインを使用してください。
   - 時間目盛りを変更して攻撃のピーク時を見つけます。
   - セキュリティー・ポリシーの違反を見つけて、攻撃時に特定の違反が他のアクティビティーと相関しているかどうかを確認します。
7. フィルター、時間フレームなどを変更してドリルダウンし、システム全体で違いがあるかどうかを確認します。
8. ダッシュボード内の以下のグラフを評価します。

   さまざまなサーバーでのエラーの比較
   : このグラフを使用して、このサーバーとデータベース・ユーザーのエラーの数が、他のサーバーとデータベース・ユーザーと比べて異常に多いかどうかを判別します。

   行動が類似する異なるデータベース・ユーザーのエラーの比較 (Compare errors from different database users with similar behavior)

このグラフを使用して、このデータベース・ユーザーのエラー・タイプと量を、類似するデータベース・ユーザーと比較します。類似するデータベース・ユーザーは、ストアード・プロシージャーを作成したすべてのユーザーです。

このデータベース・ユーザーによるストアード・プロシージャーでの類似のアクティビティー (Similar activities on stored procedures by this database user)

このグラフを使用して、特定の期間にユーザーが作成/変更したストアード・プロシージャーを確認します。 このグラフは、動詞でフィルタリングされます。このグラフを使用して、さまざまなストアード・プロシージャーでユーザーが実行したアクティビティーをドリルダウンして確認することもできます。

行動が類似するデータベース・ユーザーの違反の比較 (Compare violations from database users with similar behavior)

ストアード・プロシージャーを作成するデータベース・ユーザーの間で、違反 (ポリシー) の量とタイプを比較します

行動が類似するデータベース・ユーザーの異常値の比較 (Compare outliers from database users with similar behavior)

このグラフを使用して、このデータベース・ユーザーの異常値の量とタイプを、ストアード・プロシージャーを作成する他のデータベース・ユーザーと比較します。

このデータベース・ユーザーのデータごとの異常値 (Outliers by data on this database user)

このグラフを使用して、特定のデータベース・ユーザーの異常値の量と範囲を確認します。

親トピック: 脅威診断ダッシュボードの操作

GuardAPI 脅威検出分析機能

enable_advanced_threat_scanning

特定のデータベース攻撃 (SQL インジェクションや悪意のあるストアード・プロシージャーなど) がないか検査するスキャナー・プロセスを有効にします。

パラメーター	値	記述
すべて		オプション。一元管理構成に限り、すべての管理対象ユニット上のすべての脅威検出スキャナーを有効にします。指定可能な値: `true`、`false`。これは、api_target_host パラメーターに対する「all」オプションと同等です。
schedule_start		オプション。プロセスの実行を開始する日時を指定します。形式は、yyyy-mm-dd hh:mm:ss (24 時間クロック) です。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi enable_advanced_threat_scanning all=true schedule_start="2016-03-24 12:00:05”
```

異常値検出が無効になっているときに脅威分析が有効になっている場合は、以下のメッセージが表示されます。

```
Warning - Enabling advance threat scanning (AKA Eagle Eye) when Analytic anomaly detection is disabled.
Advance threat scanning (AKA Eagle Eye) enabled.
ok
```

disable_advanced_threat_scanning

コレクター上の脅威検出スキャナーを無効にします。

パラメーター	値	記述
すべて		一元管理構成に限り、すべての管理対象ユニット上のすべての脅威検出スキャナーを無効にします。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

get_eagle_eye_info

脅威検出パラメーターの現在の設定を表示します。

パラメーター	値	記述

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi get_eagle_eye_info
Eagle Eye Parameters Values:
EI_CASES_DISPLAY_LIMIT = 3
EI_CONFIDENCE_PCT_CHANGE_TO_REDISPLAY_CASE = 30
EI_EAGLE_EYE_ENABLED = 1
EI_PROCESSOR_TIMEOUT_SEC = 420
EI_SCANNER_PATCH_DEF = 10
EI_SCANNER_TIMEOUT_SEC = 300ok
```

set_eagle_eye_parameter

IBM 担当者の指示に従って使用してください。脅威検出の構成パラメーターを変更します。 これらのパラメーターは、以下のように `parameter_name` および `parameter_value` を使用して明示的に設定する必要があります。

```
set_eagle_eye_scanner_parameter parameter_name=[parameter] parameter_value=[value]
```

パラメーター	値	記述
EI_CASES_DISPLAY LIMIT		To-do リスト・レポートに表示されるケースの数。デフォルトは 3 です。
EI_CONFIDENCE_PCT_CHANGE_TO_REDISPLAY CASE		To-do リスト・レポートにこのケースが既に表示されていても、そこに再表示されるようにする「信頼度」変更のパーセンテージ。Guardium が、このパーセンテージ値によって信頼度を引き上げる別の兆候を検出した場合、これが発生する可能性があります。デフォルトは 30 です。
EI_PROCESSOR_TIMEOUT_SEC		このしきい値より長い時間実行されたプロセッサーはオフになります。デフォルトは 420 秒です。
EI_SCANNER_PATCH_DEF		パッチ・インストールの結果として誤検出が発生するのを防ぐために、単一プロセス実行で作成されたストアード・プロシージャーの数がこのパラメーターを越えた場合、そのプロセスはパッチがインストールされたと想定し、兆候の分析を停止します。デフォルトでは、1 回の実行で検出されるストアード・プロシージャーの作成数は 10 です。
EI_SCANNER_TIMEOUT_SEC		このしきい値より長い時間実行されたスキャナーはオフになります。デフォルトは 300 秒です。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

get_eagle_eye_scanners_info

スキャナー設定情報を返します。

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

返されるデータには、以下の情報が含まれます。

フィールド	記述
ID	スキャナー ID。

フィールド	記述
Name	スキャナー名。
Status	最後の実行以降のスキャナーの状況: I: 進行中 D: 完了 K: 強制終了 E: エラーで終了
Enabled	スキャナーが有効であるかどうかを示します。 True: 有効 False: 無効
Permanent disabled	スキャナーが 24 時間で 3 回無効になった場合、そのスキャナーは永続的に無効になります。 True: 無効 False: 有効

例:

```
grdapi get_eagle_eye_scanners_info
ID=0
ID:1, Name:SQLInjectionExceptionsScanner, Status:D, Enabled:true, Permanent disabled:false
ID:2, Name:NumNewConstructScanner, Status:D, Enabled:true, Permanent disabled:false
ID:3, Name:SQLInjectionSuspiciousObjectScanner, Status:D, Enabled:true, Permanent disabled:false
ID:4, Name:SqliQueryScanner, Status:Unknown, Enabled:false, Permanent disabled:true
ID:5, Name:EagleEyeSTPCreateProcedureScanner, Status:D, Enabled:true, Permanent disabled:false
ID:6, Name:EagleEyeSTPCallProcedureScanner, Status:D, Enabled:true, Permanent disabled:false
ID:7, Name:EagleEyeSTPExceptionProcedureScanner, Status:D, Enabled:true, Permanent disabled:false
ID:8, Name:EagleEyePreviousStpUsageProcedureScanner, Status:D, Enabled:true, Permanent disabled:false
ID:9, Name:EagleEyeSTPViolationProcedureScanner, Status:D, Enabled:true, Permanent disabled:false
ID:10, Name:EagleEyeSTPUserOutlierScanner, Status:D, Enabled:true, Permanent disabled:false
ok
```

set_eagle_eye_scanner_parameter

IBM 担当者の指示に従って使用してください。スキャナーをアクティブ化または非アクティブ化します。これらのパラメーターは、以下のように `parameter_name` および `parameter_value` を使用して明示的に設定する必要があります。

```
set_eagle_eye_scanner_parameter parameter_name=[parameter] parameter_value=[value]
```

パラメーター	値	記述
scanner_id		必須。スキャナーの固有 ID。これは、get_eagle_eye_scanners_info GuardAPI コマンドから取得できます。
is_active		スキャナーを実行するかどうかを定義します。タイムアウトになったために自動的に停止されたスキャナーを開始するために使用されます。 0 : スキャナーは停止される 1: スキャナーはアクティブ化される
is_permanent_inactive		スキャナーが 24 時間で 3 回無効になった後に永続的に無効になった場合、この GuardAPI を使用することでのみ再び有効にすることができます。 1: スキャナーは永続的に停止される 0: スキャナーは有効化される
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

以下の例では、永続的に非アクティブ化されたスキャナーを再アクティブ化します。

```
set_eagle_eye_scanner_parameter scanner_id=2 parameter_name=is_permanent_inactive parameter_value=0
```

get_eagle_eye_symptom_period_hours

徴候期間パラメーターの値を時間単位で示します。徴候期間は、1 つのケースについて収集された徴候を、どのくらい前までプロセスが検索して分析するかを決定します。

パラメーター	値	記述
case_name		必須。ケース・タイプ。以下の値を使用できます。 STP: 悪意のあるストアード・プロシージャーのケース SQL_INJECTION: SQL インジェクションのケース
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi get_eagle_eye_symptom_period_hours case_name=STP
The symptoms period for case type: STP is: 168 in hours
ok
```

set_eagle_eye_symptom_period_hours

徴候期間パラメーターの値を時間単位で設定します。徴候期間は、1 つのケースについて収集された徴候を、どのくらい前までプロセスが検索して分析するかを決定します。

パラメーター	値	記述
case_name		必須。ケース・タイプ。以下の値を使用できます。 STP: 悪意のあるストアード・プロシージャーのケース SQL_INJECTION: SQL インジェクションのケース
symptom_period_hours		必須。整数。1 つのケースの兆候を分析するための過去の時間数。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi set_eagle_eye_symptom_period_hours case_name=STP symptom_period_hours=170
The symptoms period for case type: STP was changed. The old value was: 168. The new value is: 170
ok
```

get_eagle_eye_debug_level

IBM サービス担当員によって使用されます。現在のデバッグ・レベルを表示します。

- 1: オン
- 0: オフ

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi get_eagle_eye_debug_level
ID=0
```

```
component=EAGLE_EYE level=1
ok
```

set_eagle_eye_debug_level

IBM サービス担当員によって使用されます。現在のデバッグ・レベルを表示します。

パラメーター	値	記述
level		整数。必須。指定可能な値: 1: オン 0: オフ
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi set_eagle_eye_debug_level level=0
ID=0
ok
```

親トピック: GuardAPI

調査ダッシュボード

調査ダッシュボードは、Guardium 環境に存在する可能性がある問題を特定して評価するための強力なツールを提供します。これはローカルまたはシステム全体のフィルタリングされていないデータを使用し、Guardium 環境全体で、その環境内のすべての Guardium コレクターを対象としてデータを照会するための多くのフィルタリング・オプションを提供します。

調査ダッシュボードは、データ全体にわたるパターン、異常、および関係を明らかにするのに役立つ相関グラフを提供します。トポロジー、統合、またはロード・バランシング・スキームについての詳細な知識は必要ありません。これには、オリジナルの Quick Search for Enterprise の機能、およびデータを視覚化して分析するための他のツールが含まれています。

注: 調査ダッシュボードはフルスクリーン・モードで表示することをお勧めします。
制約事項: 調査ダッシュボードとデータ・レベル・セキュリティーを同時に使用可能にすることはできません。

動作モード

調査ダッシュボードは、以下の 3 つの操作モードをサポートします。

中央マネージャー専用モード

中央マネージャー上で実行依頼された照会は、検索が有効になっているすべての Guardium コレクターから企業規模の結果を返します。管理対象ユニットで実行依頼された照会は、ローカルな結果を返します。

中央マネージャー専用モードは、デフォルトの動作モードです。

全マシン・モード

企業規模の検索照会は、検索が有効になっている Guardium 環境のすべてのマシンから実行依頼されます。このモードでは、検索結果が返されるまで時間がかかる場合があります。また、環境内のすべての管理対象ユニットが接続されている必要があります。

ローカル専用モード

このモードでは、検索照会が、検索を実行依頼したローカル・コレクターに制限されます。そのため、Guardium 環境内の他のコレクターからはデータが取得されません。ローカル専用モードの CM では、データは表示されません。

検索モードの設定については、『GuardAPI Quick Search for Enterprise の機能』を参照してください。

- 調査ダッシュボードの有効化と無効化
  このトピックでは、調査ダッシュボードを有効化および無効化する方法について説明します。
- 調査ダッシュボードでのファイル・アクティビティーの有効化
- 調査ダッシュボードへのアクセス
- データの調査ダッシュボード
  調査ダッシュボードは、事前設定されたグラフのグループと 1 つの表からなり、特定の時点でシステムに何が起こっているのかを理解するのに役立ちます。また、調査ダッシュボードをベースに、独自にカスタマイズしたダッシュボードを作成することもできます。
- ファイルの調査ダッシュボード
  調査ダッシュボードは、事前設定されたグラフのグループと 1 つの表からなり、特定の時点でシステムに何が起こっているのかを理解するのに役立ちます。また、調査ダッシュボードをベースに、独自にカスタマイズしたダッシュボードを作成することもできます。
- 調査ダッシュボードでのデータのフィルタリングおよびフィルターの保存

- 個々のグラフのフィルタリング
- 調査ダッシュボードの作成、保存、およびエクスポート
- トポロジー・ビューの使用
  トポロジー・ビューは、検索結果の Guardium アプライアンスを可視化したものです。
- ローカル検索および分散検索
- データの洞察の使用
  「データの洞察」可視化により、ユーザーは Guardium システムによって収集されたイベントのシーケンスを詳しく検査できます。特定の時間枠におけるアクティビティーを総合的に描写し、異常な動作を検出するのに役立ちます。

親トピック: モニターおよび監査
関連情報:
GuardAPI 調査ダッシュボード機能
調査ダッシュボードの CLI コマンド

調査ダッシュボードの有効化と無効化

このトピックでは、調査ダッシュボードを有効化および無効化する方法について説明します。

始める前に

調査ダッシュボードには以下の最小ハードウェア要件があります。

- 64 ビット・アーキテクチャー
- 24 GB RAM
- 4 コア CPU

制約事項: 調査ダッシュボードとデータ・レベル・セキュリティーを同時に有効化することはできません。

手順

1. CLI ロールを持つユーザーまたは管理者としてマシンにログインします。
2. 次の GuardAPI コマンドを使用して調査ダッシュボードを有効にします。

   ```
   grdapi enable_quick_search schedule_interval=2 schedule_units=MINUTE
   ```

   デフォルトでは、違反は検索結果に含まれません。違反を含めるには、次のように includeViolations パラメーターを `true` に設定します。

   ```
   grdapi enable_quick_search schedule_interval=2 schedule_units=MINUTE includeViolations=true
   ```

   異常値の検出を有効にするには、Outliers Detectionを参照してください。

   検索索引の更新間隔など、追加のパラメーターを指定できます。パラメーターと説明の完全なリストについては、『GuardAPI 調査ダッシュボード機能』の参照情報を確認してください。

3. 次の GuardAPI コマンドを使用すると、任意のタイミングで調査ダッシュボード機能を無効にすることができます。

   ```
   grdapi disable_quick_search
   ```

タスクの結果

有効になったら、調査ダッシュボードへのアクセスで詳細情報を確認し、調査ダッシュボードの使用を開始します。

重要:

- 調査ダッシュボード機能を使用すると、中央マネージャーとコレクターの両方で、ポート 8983 とポート 9983 が開きます。これらのポートは、調査ダッシュボードを有効にすると開かれ、無効にすると閉じられます。調査ダッシュボードを使用するには、ポート 8983 とポート 9983 での中央マネージャーとコレクターの間の双方向通信がファイアウォールでブロックされないようにしてください。
- 索引付き検索データは 3 日間保存されます。保存期間を変更するには、purge object Guardium CLI コマンドを使用します。例えば、「store purge object age 395」というコマンドを実行すると、データの保存期間が 5 日間に変更されます。`39` という値は、検索索引に関連付けられているデフォルトのオブジェクト識別番号であることに注意してください。詳しくは、『構成および制御 CLI コマンド』資料を参照してください。

親トピック: 調査ダッシュボード
関連情報:
GuardAPI 調査ダッシュボード機能
調査ダッシュボードの CLI コマンド

調査ダッシュボードでのファイル・アクティビティーの有効化

始める前に

- FAM バンドルをインストールして構成する必要があります。ファイルのディスカバリーおよび分類 GIM パラメーターを参照してください。
- 調査ダッシュボードを有効にする必要があります。調査ダッシュボードの有効化と無効化を参照してください。
- V10.0 Guardium システムでは、V10.1 FAM クローラーを使用しないでください。V10.1 Guardium システムでは、V10.0 FAM クローラーを使用しないでください。

このタスクについて

注: FAM は、サーバーにサーバーの IP アドレスを照会して、検出された最初のものを選択します。ホストに複数の IP アドレスがある場合、ホスト名から「適切な」IP アドレスを選択する方法はありません。 レポートでその IP アドレスが表示されるようにするには、IP アドレスを明示的に指定してください。

手順

1. コレクターの CLI プロンプトで、GuardAPI コマンドを実行します。
   `grdapi enable_fam_crawler [extraction_start] [schedule_start] [activity_schedule_interval] [activity_schedule_units] [entitlement_schedule_interval] [entitlement_schedule_units]` 例: 次のコマンドは、ディスカバリーおよび分類の更新された結果を、分類データについては 2 分ごとに、資格情報については毎日、エンタープライズ検索に送信します。
   `grdapi enable_fam_crawler activity_schedule_interval=2 activity_schedule_units=MINUTE entitlement_schedule_interval=1 entitlement_schedule_units=DAY`
   デフォルトでは、コマンドの入力時に抽出が開始され、コマンドを入力したとき (時刻) からデータが抽出されます。
2. 各コレクターで繰り返します。

親トピック: 調査ダッシュボード
関連概念:
ファイルの調査ダッシュボード
関連情報:
GuardAPI 調査ダッシュボード機能

調査ダッシュボードへのアクセス

手順

1. 「調査」 > 「データ・アクティビティーの検索」または「調査」 > 「ファイル・アクティビティーの検索」をクリックします。
2. または、検索をユーザー・インターフェースに切り替えて、調査ダッシュボードを検索します。次に、「データ・アクティビティーの検索」か「ファイル・アクティビティーの検索」のいずれかを選択します。

タスクの結果

データまたはファイルのデフォルトの調査ダッシュボードが開きます。デフォルトでは、ダッシュボード全体に適用されるフィルターのみ、過去 1 時間のデータを表示します。
親トピック: 調査ダッシュボード

データの調査ダッシュボード

調査ダッシュボードは、事前設定されたグラフのグループと 1 つの表からなり、特定の時点でシステムに何が起こっているのかを理解するのに役立ちます。また、調査ダッシュボードをベースに、独自にカスタマイズしたダッシュボードを作成することもできます。

データ・アクティビティー・モニターには 4 つのデフォルトのビューがあり、それぞれ異なるグラフと表があります。ダッシュボード・メニュー からビューを選択します。デフォルトのビューは変更できません。

デフォルトのダッシュボードには、過去 1 時間分のデータが以下の 1 つ以上のグラフと表に表示されます。

- トリメトリック・グラフ (3 軸データ・グラフ)。デフォルトのビューはカラー・マップです。追加のビューは、棒グラフ、バブル・グラフ、折れ線グラフ、円グラフ、階段グラフ、および面グラフです。
- 結果表: 元のクイック検索の検索結果と調査機能を提供します。結果表は常にダッシュボードの下部に表示されます。これは任意のダッシュボードに追加できます。以下のタブがあります。
  - アクティビティー : 「要約」タブと「詳細」タブ。「要約」タブの各行には、サーバーと DB のペアごとに記録されたアクティビティーのインスタンスの数と、データベース・タイプの数が表示されます。「詳細な概要」には、ソース・プログラムの数、データベース・ユーザー、OS ユーザー、クライアントのホスト名、クライアント IP、および日付が追加されます。「詳細」タブの各行に、1 つのアクティビティーに関する完全な詳細が示されます。
  - 異常値: 調査ダッシュボードでのデータ異常値の解釈を参照してください。
  - エラー: 「要約」タブと「詳細」タブ。「要約」タブの各行には、サーバーごとに報告されたエラーのインスタンスの数と、データベース・タイプおよびデータベース・ユーザーの数が表示されます。「詳細な概要」には、クライアント IP の数、エラー・タイプ、および日付が追加されます。「詳細」タブの各行に、1 つのエラーに関する完全な詳細が示されます。
  - 違反: 「要約」タブと「詳細」タブ。「要約」タブの各行には、サーバーと DB のペアごとに記録された違反のインスタンスの数と、データベース・タイプの数が表示されます。「詳細な概要」には、ソース・プログラムの数、データベース・ユーザー、OS ユーザー、クライアントのホスト名、クライアント IP、重大度、違反、および日付が追加されます。「詳細」タブの各行に、1 つの違反に関する完全な詳細が示されます。
  - 完全な SQL: 各行はログに記録された 1 つの完全な SQL です。「検索」ボックスに文字列を入力して、「アクティビティー」タブまたは「エラー」タブにフィルタリングされた結果を表示できます。また、「アクティビティー」タブまたは「エラー」タブで開始して、1 つ以上の行を選択し、「完全な SQL」を右クリックして選択し、「完全な SQL」タブで結果を表示することもできます。

追加またはオープンできる追加ビューは、以下のとおりです。

- トポロジー・ビュー 検索サーバー状況ビュー: トポロジー・ビューの使用を参照してください。
- アニメーション・バブル・チャート: 過去 48 時間にわたるデータ変更のアニメーション表示。このグラフは、24 時間の期間にわたるオブジェクトの動作を表します。 各オブジェクトは円として表され、その面積と位置 (x 軸と y 軸) がユーザー選択の 3 つの変数を表します。アニメーションは、24 時間にわたるオブジェクトの動作を表します。「グラフの追加」ドロップダウンからアクセスします。
- アクティビティー・グラフ: 「結果表」の上にある、アクティビティーおよび異常値のボリュームを表示する折れ線グラフ。「グラフの追加」ドロップダウンからアクセスします。
- データの洞察: データ・アクティビティーの 3D 可視化。データの洞察の使用を参照してください。「グラフの追加」ドロップダウンからアクセスします。

このページのコントロールとオプションは以下のとおりです。

- 検索結果からカテゴリー化されたファセットのリスト (「場所」、「ユーザー」、「対象」、「例外」、および「タイミング」) が、すべてのダッシュボードの左側に表示されます。これは削除できません。リストを展開して、個々のファセットをクリックすることにより、特定のファセットを基準にダッシュボード全体をフィルタリングできます。
- ウィンドウの上部にある「アクティブ・フィルター」行には、現在のフィルターが表示されます。フィルターを削除するには をクリックします。
- Big Data Intelligence のみ: 「Guardium システム (Guardium System)」または「GBDI - Guardium Big Data Intelligence」を選択します。
- 検索フィールド: ファセットに関係なく、すべてのフィールドの結果を同時にフィルタリングするフリー・テキスト検索。
- 分散検索: ローカル検索および分散検索を参照してください。
- 表示するデータの対象期間: 変更するには、右上隅にあるドロップダウンをクリックします。オプションは、最後の 1 時間、最後の 3 時間、最後の 1 日、最後の 3 日、ユーザーが指定する任意の期間です。デフォルトは 1 時間です。「GBDI - Guardium Big Data Intelligence」を選択した場合の期間のオプションは、直近 1 日、直近 3 日間、直近 1 週間、直近 3 週間、ユーザーが指定する任意の期間です。期間にはタイム・ゾーンの設定が含まれ、デフォルトでは現在の Guardium システムのタイム・ゾーンになっています。このタイム・ゾーンに従ってデータが報告されます。
- 「フィルター」ドロップダウン: 調査ダッシュボードでのデータのフィルタリングおよびフィルターの保存を参照してください。
- 新規ダッシュボードの追加 ダッシュボードに変更を保存 ダッシュボードを別名で保存: 調査ダッシュボードの作成、保存、およびエクスポートを参照してください。

親トピック: 調査ダッシュボード
関連概念:
調査ダッシュボードでのデータ異常値の解釈
関連タスク:
トポロジー・ビューの使用
データの洞察の使用

ファイルの調査ダッシュボード

調査ダッシュボードは、事前設定されたグラフのグループと 1 つの表からなり、特定の時点でシステムに何が起こっているのかを理解するのに役立ちます。また、調査ダッシュボードをベースに、独自にカスタマイズしたダッシュボードを作成することもできます。

デフォルトの FAM ビューは 2 つあり、それぞれ異なるグラフと表があります。ダッシュボード・メニュー からビューを選択します。デフォルトのビューは変更できません。
注: Windows でリモート・デスクトップを経由して接続する場合を除き、ダッシュボードでは、サーバー IP とクライアント IP は常に同じです。クライアント IP は、リモート・デスクトップ・セッションを使用して接続している場合のみサポートされます。
注: FAM は、サーバーにサーバーの IP アドレスを照会して、検出された最初のものを選択します。ホストに複数の IP アドレスがある場合、ホスト名から「適切な」IP アドレスを選択する方法はありません。 レポートでその IP アドレスが確実に表示されるようにするには、IP アドレスを明示的に指定してください。
デフォルトのダッシュボードには、過去 1 時間分のデータが以下の 1 つ以上のグラフと表に表示されます。

- トリメトリック・グラフ (3 軸データ・グラフ): デフォルトのビューはカラー・マップです。追加のビューは、棒グラフ、バブル・グラフ、折れ線グラフ、円グラフ、階段グラフ、および面グラフです。
- 結果表: 元のクイック検索の検索結果と調査機能を提供します。結果表は常にダッシュボードの下部に表示されます。これは、任意のダッシュボードに追加できます。以下のタブがあります。
    - アクティビティー: ファイル・サーバーのポリシー・ルールに基づき、「要約」タブと「詳細」タブにモニター・データが表示されます。「要約」タブの各行には、サーバーと OS ユーザーごとに記録されたアクセス・アクティビティーのインスタンス数が示されます。「詳細」タブには、「サーバーのホスト名」、「サーバー」、「クライアントのホスト名」、「クライアント IP」、「OS ユーザー」、「ファイルの絶対パス名」、「コマンド」、「日付と時刻」が追加されます。「詳細」タブの各行に、1 つのアクティビティーに関する完全な詳細が示されます。「アクティビティー」タブに表示されるデータは、コレクターの日時と整合しています。
    - 異常値: 調査ダッシュボードでのファイル・アクティビティーの異常値の解釈を参照してください。
    - エラー: 「要約」タブと「詳細」タブ。「要約」タブの各行には、サーバーとクライアント IP ごとに報告されたエラーのインスタンス数と日付が示されます。「詳細な概要」には、エラーの詳細と時刻が追加されます。「詳細」タブの各行に、1 つのエラーに関する完全な詳細が示されます。
    - 違反: 「要約」タブと「詳細」タブ。「要約」タブの各行には、サーバー、ソース・プログラム、OS ユーザーの組み合わせごとに記録された違反のインスタンス数が示されます。「詳細な概要」には、クライアント IP、重大度、違反と違反の詳細、日付、時刻が追加されます。「詳細」タブの各行に、1 つの違反に関する完全な詳細が示されます。「違反」タブに表示されるデータは、ファイル・サーバーの日時と整合しています。
    - 資格: 「要約」タブと「詳細」タブ。ファイル・サーバーの場合、このタブには現在の FAM 判定プランに基づく機密データが表示されます。「要約」タブの各行には、サーバーと所有者ごとに記録されたアクセス・アクティビティーのインスタンス数が示されます。「詳細」タブには、「サーバーのホスト名」、絶対パス、「タイプ」、「」、「サイズ」、「分類エンティティー」(このファイルを機密として識別する判定プラン)、「所有者」、「クライアントのホスト名」、「クライアント IP」、「OS ユーザー」、「ファイルの絶対パス名」、書き込み/読み取り/実行/削除の権限を持つユーザーとグループ、最終変更、「バージョン」(Sharepoint のみ)、作成時間、「日付」、「時刻」が追加されます。「詳細」タブの各行に、1 つのアクティビティーに関する完全な詳細が示されます。この表のデータを使用して、ファイル・サーバーのポリシー・ルールとグループを作成できます (調査ダッシュボードの「資格」タブでの FAM ポリシー・ルールの作成を参照)。

追加または開くことができる追加のビューは、以下のとおりです。

- トポロジー・ビュー 検索サーバー状況ビュー: トポロジー・ビューの使用を参照してください。
- アニメーション・バブル・チャート: 過去 48 時間にわたるデータ変更のアニメーション表示。このグラフは、24 時間の期間にわたるオブジェクトの動作を表します。 各オブジェクトは円として表され、その面積と位置 (x 軸と y 軸) がユーザー選択の 3 つの変数を表します。アニメーションは、24 時間にわたるオブジェクトの動作を表します。「グラフの追加」ドロップダウンからアクセスします。
- アクティビティー・グラフ: 「結果表」の上にある、アクティビティーおよび異常値のボリュームを表示する折れ線グラフ。「グラフの追加」ドロップダウンからアクセスします。

このページのコントロールとオプションは以下のとおりです。

- 検索結果からカテゴリー化されたファセットのリスト (「場所」、「ユーザー」、「対象」、「例外」、および「タイミング」) が、すべてのダッシュボードの左側に表示されます。これは削除できません。リストを展開して個々のファセットをクリックすることで、特定のファセットを基準にダッシュボード全体をフィルタリングできます。
- ウィンドウの上部にある「アクティブ・フィルター」行には、現在のフィルターが表示されます。フィルターを削除するには、 をクリックします。
- 検索フィールド: ファセットに関係なく、すべてのフィールドの結果を同時にフィルタリングするフリー・テキスト検索。

- 分散検索: ローカル検索および分散検索を参照してください。
- 表示するデータの対象期間: 変更するには、右上隅にあるドロップダウンをクリックします。オプションは、最後の 1 時間、最後の 3 時間、最後の 1 日、最後の 3 日、ユーザーが指定する任意の期間です。デフォルトは 1 時間です。
- 「フィルター」ドロップダウン: 調査ダッシュボードでのデータのフィルタリングおよびフィルターの保存を参照してください。
- 新規ダッシュボードの追加 ダッシュボードに変更を保存 ダッシュボードを別名で保存: 調査ダッシュボードの作成、保存、およびエクスポートを参照してください。

親トピック: 調査ダッシュボード
関連概念:
調査ダッシュボードでのファイル・アクティビティーの異常値の解釈
関連タスク:
トポロジー・ビューの使用

調査ダッシュボードでのデータのフィルタリングおよびフィルターの保存

このタスクについて

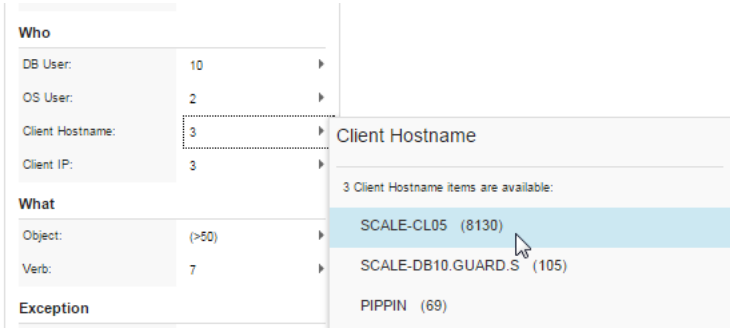
調査ダッシュボード全体および個々のグラフでデータをフィルタリングできます。「結果表」から関連情報にドリルダウンできます。

後で使用するためにフィルターを保存できます。フィルター・セットを保存するときは、フィルター・セットを共有するかどうかを選択し、それを共有するロールを選択します。

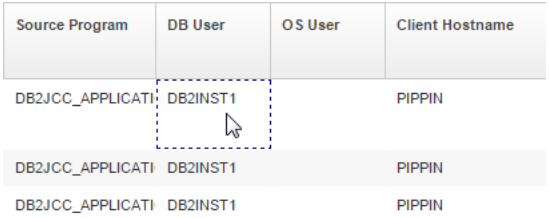
手順

1. 以下のようにルールと構文を使用して、データをフィルタリングします。
   - ある語句と完全に一致する項目を検索するには、検索語を二重引用符で囲みます。例えば、「プロファイル・アラート・リスト」と入力すると、接続プロファイル・アラート・リストの項目は返されますが、プロファイル・リスト・アラートの項目は返されません。
   - 指定したすべての検索語と一致する項目を検索するには、検索語をスペースで区切ります。例えば、Hadoop getlisting と入力すると、語の位置や順序にかかわらず、Hadoop と getlisting の両方が含まれている項目が返されます。
   - 指定したいずれかの検索語と一致する項目を検索するには、検索語を OR または縦棒 (|) で区切ります。例えば、Hadoop OR getlisting と入力すると、語の位置にかかわらず、Hadoop か getlisting のいずれかが含まれている項目が返されます。
   - 指定した検索語が含まれない項目を検索するには、NOT またはピリオド (.) を使用します。例えば、NOT Hadoop と入力すると、語の位置にかかわらず、Hadoop が含まれている項目は返されません。
   - ワイルドカードを使用するには、文字列の先頭または末尾にアスタリスク (*) を付けます。例えば、10.10.70.* と入力すると、文字列 10.10.70. の後にさらに文字が続いている項目が返されます。
   - 検索ルールは組み合わせて使用できます。例えば、2016–5-08 (19.*|20.*) と入力すると、5 月 8 日の 19:00:00 から 20:59:59 までの時刻範囲での結果が返されます。

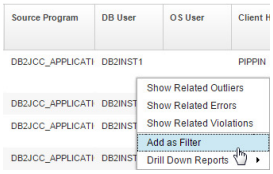
   フィルターを追加すると、ビューに指定された *RefFilter* に基づいて各ビューが変更されます。現在のフィルターがメニュー・バーに表示されます。X をクリックすると、それぞれをクリアできます。
2. 次のいずれかの方法で、検索結果を絞り込みます。
   - 以下のようにファセット・リストに基づいて特定のフィルターを選択します。



   - グラフの X 軸または Y 軸のヘッダーをクリックします。
   - 以下のように結果表の個別の検索結果をクリックします。



   注: 1 つ以上の行を選択し、サーバー/データベース・ユーザー/クライアント IP セルのいずれかを右クリックして既存のグループに追加するか、または新規グループを作成できます。
3. 個々の結果をドリルダウンします。その方法として、特定の検索結果を右クリックして、関連する異常値、エラー、違反を調べたり、いくつかの使用可能なドリルダウン・レポートのうちの 1 つを表示したりします。



4. フィルター・セットを保存するには、「フィルター」 > 「保存」をクリックします。フィルターの名前を指定して、「プライベート」としてマークを付けるか、「共有」をクリックして、フィルターを特定のロールと共有します。 デフォルトのフィルター・セットとして保存するには (ダッシュボードは常にこれらのフィルターで開きます)、「デフォルト・フィルターとして設定」を選択します。作業が終了したら、「OK」をクリックしてフィルターを保存します。

親トピック: 調査ダッシュボード

個々のグラフのフィルタリング

このタスクについて

個々のグラフをフィルタリングできます。 アイコンは、グラフに対して一般的なダッシュボード・フィルターとは異なる特定のフィルターが設定されている場合、赤になります。アイコンの上にマウスを移動すると、そのグラフで使用されているフィルターが表示されます。

グラフではフィルターを非アクティブとして設定できます。これは、グラフ・データがそのフィールドによってフィルタリングされないことを意味します。これにより Guardium は、特定のケースに関連した項目に加え、類似しているかまたは何らかの方法で調査に関する洞察をさらに提供する可能性がある他の項目を表示できます。
例: サーバー上のアクティビティーを調査中に、グラフの 1 つを他のサーバーのデータと比較できます。これは、そのグラフに対してのみ「サーバー」フィルターを非アクティブ化することによって実行できます。 これを行うには、 アイコンをクリックし、その「サーバー」行について「非アクティブ」ラジオ・ボタンを選択します。

手順

1. アイコンをクリックします。 「グラフ・フィルターの設定」が開きます。
2. 必要に応じてラジオ・ボタンをクリックまたはクリアして、「適用」をクリックします。

親トピック: 調査ダッシュボード

調査ダッシュボードの作成、保存、およびエクスポート

このタスクについて

ダッシュボードでデータをフィルタリングするには、多くの方法があります。フィルター・セットは、専用にすることも、共有にすることもできます。例えば、環境に詳しい担当者は関連するフィルターをセットアップできます。この担当者は、特定の調査ユーザー向けのフィルターを作成してから、そのフィルターをそのロールと共有できます。 事前定義のシステム・ダッシュボードを変更して、元の名前で保存することはできません。
重要: すべての調査ダッシュボードはパブリックです。ダッシュボードが保存されると、ダッシュボードにアクセスできるすべてのユーザーは、ダッシュボード・メニューを使用して保存されたダッシュボードにもアクセスできます。さらに、ダッシュボードをデフォルト・ダッシュボードとして保存すると、すべてのユーザーにそのデフォルトが表示されます。
表示するデータに応じて、同じダッシュボードを異なるフィルター・セットで使用できます。
例: ダッシュボードに、データベース・ユーザーのアクティビティーとクライアント IP 別の明細を表示したアクティビティー・グラフが含まれています。同じデータを別のデータベース (HR や Financial など) でフィルタリングして表示することもできます。また、データベースごとに異なるコマンド・タイプを追加することもできます。

- フィルター 1: データベース HR 別、動詞 SELECT 別
- フィルター 2: データベース FINANCIAL 別、動詞 UPDATE 別

同じダッシュボードを開き、「アクティブなフィルター」リストの上の アイコンと アイコンを使用して、そのグラフに関連付けられた異なるフィルター・セットを切り替えることができます。

脅威診断を含む、すべての調査ダッシュボードを暗号化して、共有するためにエクスポートできます。フィルターではなく、ダッシュボード定義のみがエクスポートされます。

特定のインシデント・タイプの調査に適したグラフ・セットを使用して構成されたダッシュボードがある場合は、実際の攻撃データを含めたりフィルターを公開したりせずにこの知識を他の Guardium ユーザーと共有できます。

手順

1. 現在の表示を保存するには、 アイコンをクリックします。
2. 変更およびその後の使用のためにダッシュボードを別の名前で保存するには、 アイコンをクリックし、記述名およびオプションでカテゴリーを指定して保存します。また、ダッシュボードを保存する際にカテゴリーを定義することもできます。名前とカテゴリーにはスペースを含めることができます。 ダッシュボードを後で取得するには、 アイコンをクリックして、ダッシュボード・メニューを開きます。
3. 調査ダッシュボードをエクスポートするには、「管理」 > 「データ管理」 > 「定義のエクスポート」に移動します。「タイプ」メニューから、「調査ダッシュボード」を選択し、エクスポートするダッシュボード定義を選択します。次に、「エクスポート」をクリックします。

親トピック: 調査ダッシュボード

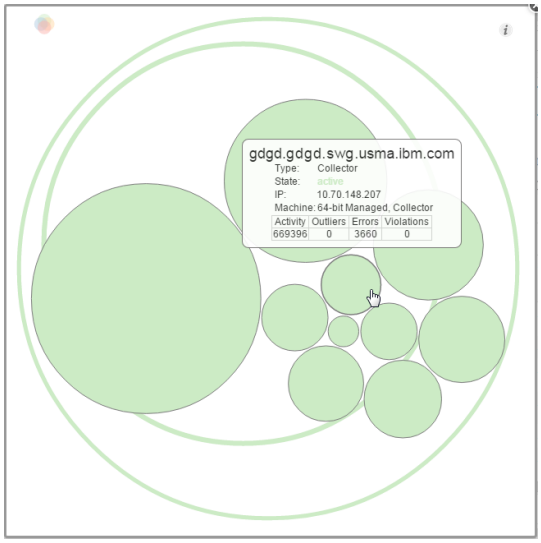
トポロジー・ビューの使用

トポロジー・ビューは、検索結果の Guardium アプライアンスを可視化したものです。

このタスクについて

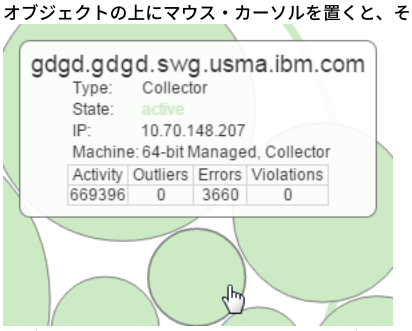
各サーバーの詳細を表示したり、フィルター基準を選択したり、検索結果を Guardium 環境全体の特定のセグメントに絞り込んだりすることができます。塗りつぶされた円はコレクターおよびアグリゲーターを表します。線のみの円は中央マネージャーを表します。円の色は、サーバーの状況を示します。中央マネージャーのアウトライン・カラーは、その状況を示します。円のサイズは、収集されたデータの相対ボリュームを示します。

トポロジー・ビューはスタンドアロン・マシンではサポートされていません。



手順

1. トポロジー・ビューを開くには、調査ダッシュボードのツールバーの「検索サーバー状況ビュー」アイコン 「検索サーバー状況ビュー」 をクリックします。
2. オブジェクトの上にマウス・カーソルを置くと、そのオブジェクトに関する詳細情報が表示されます。



3. オブジェクトを選択し、検索結果をそのオブジェクトとその子 (存在する場合) のみに絞り込みます。トポロジー・ビュー内の複数のオブジェクトを選択または選択解除するには、Ctrl キーを押しながらクリックします。
4. トポロジー・ビューを閉じるには、「閉じる」アイコン をクリックするか、トポロジー・ブラウザーの外側をクリックします。トポロジー・ビューで選択された有効範囲に基づいて、検索結果は使用可能なデータを反映するように自動更新されます。

親トピック: 調査ダッシュボード

ローカル検索および分散検索

このタスクについて

調査ダッシュボードは、ローカル・モードでも分散モードでも実行できます。ローカル・モードでは、検索はローカル・マシン (検索を実行するマシン) で使用可能なデータに制限されます。例えば、個々のコレクターから実行されたローカル検索では、そのコレクターの下にあるデータ・ソースからの結果は返されますが、環境内の他のコレクターの下にあるデータ・ソースからの結果は返されません。分散モードでは、検索を実行すると、Guardium 環境全体からのデータが返されます。結果は、検索が実行された特定のマシンによって制限されることはありません。 検索結果を Guardium 環境全体の中の特定セグメントに便宜的に絞り込むためのトポロジー・ツールが用意されています。

調査ダッシュボードは、ローカル検索モードにデフォルト設定されます。

手順

1. ローカル検索と分散検索を切り替えるには、検索ウィンドウのツールバーにある「すべてのアプライアンスの検索の有効化/無効化 (Enable / Disable search all appliance)」アイコン をクリックします。 ローカル検索または分散検索の選択に基づいて、検索結果は使用可能なデータを反映するように自動更新されます。
2. グローバル検索結果を Guardium 環境の特定セグメントでフィルタリングする方法について詳しくは、トポロジー・ビューの使用を参照してください。

親トピック: 調査ダッシュボード

データの洞察の使用

「データの洞察」可視化により、ユーザーは Guardium システムによって収集されたイベントのシーケンスを詳しく検査できます。特定の時間枠におけるアクティビティーを総合的に描写し、異常な動作を検出するのに役立ちます。

このタスクについて

データの洞察は、データ・トランザクションの全体像を把握し、予期しない動作を識別するために人間の視覚能力を使用するという革新的なパラダイムを導入します。Guardium は、監査を支援し、攻撃を検出するための堅牢な機械学習機能とデータ分析機能を既に提供しています。アルゴリズム、データ分析、およびグラフは、累積された経験と知識に基づいて設計されています。データの洞察は、人間の視覚認知の柔軟性を使用して、これまでは検出できなかった、既知の攻撃のパターンに適合しない生データの関連と移動を特定します。このツールは複雑な視覚シナリオにおけるデータのさまざまな局面を提示し、大量の複雑なデータを直接調査するためのツールを監視者に提供します。

データの洞察は、監査対象データを 3-D で可視化されたデータ・フロー (ソースから宛先まで) に時系列で変換し、発生したとおりに展開されたデータ・トランザクションを表示します。

可視化スペースには 2 つのプレーンがあり、それぞれが特定のタイプの監査ドメインのエンティティーを表します。監査データ内のすべてのエントリーは、上部プレーンのオブジェクト (例えば、クライアント IP) から下部プレーンのオブジェクト (例えば、データベース) に移動する「点滅線」として表されます。 ソースと宛先の間の点滅線は、特定のソースと宛先の間で相互作用があったことを示す証跡 (点線) を残します。それは背景へと徐々に消えていきます。証跡は、選択された期間のソースと宛先の間の相互作用の概要を示します。各ソースおよび宛先のサイズは、アクティビティーのレベルと関連しています。 ソースは、宛先の近くおよび他の類似したソースの近くにあります。この表示はさまざまな方法で変更でき、データに関する追加の情報または局面を提供します。 データの洞察は、VR ヘッドセットを使用して表示できます。

データの洞察は、常に変化するこのパラダイムに対する答えです。これは、人間の視覚認知の柔軟性を追加して、既知の攻撃タイプとは関係なく、これまでは検出できなかった生データの関連と移動を特定します。

データの洞察は、監査対象データを 3-D で可視化されたデータ・ソースおよび宛先に時系列で変換し、発生したとおりに展開されたデータ・トランザクションを表示します。可視化スペースには 2 つのプレーンがあり、それぞれが 1 つのタイプの監査ドメインのエンティティーを表します。監査データ内のすべてのエントリーは、上部プレーンのオブジェクト (クライアント IP、OS ユーザー、またはソース・プログラム) から下部プレーンのオブジェクト (データベース、オブジェクト、またはサーバー) に移動する「点滅線」として表されます。ソースと宛先の間の点滅線は、特定のソースと宛先の間で相互作用があったことを示す証跡 (点線) を残します。それは背景へと徐々に消えていきます。 点滅線には、宛先データベースと同じカラーがあります。証跡は、選択された期間のソースと宛先の間の相互作用の概要を示します。ソースは、宛先の近くおよび他の類似したソースの近くにあります。宛先エンティティーのサイズは、他の宛先エンティティーとの相対的なトランザクションの量に比例します。この表示を変更する方法は多数あります。例えば、上部エンティティーの色分け (データ・ソースの詳細が変更されると色が変わります)、データの洞察グラフのフィルタリング、調査ダッシュボードのファセットなどです。また、VR ヘッドセットを使用してデータの洞察を表示することもできます。

手順

1. 「調査ダッシュボード」ウィンドウで、「グラフの追加」 > 「データの洞察のグラフ」をクリックします。「グラフ設定」ウィンドウが開きます。
2. 「グラフ設定」ペインで、両方のプレーンに表示されるオブジェクト・タイプ、つまり両者の間のデータ・フローのタイプを変更します。オプションで、上部のプレーンのエンティティーを 2 次基準で色分けし、別のレベルの分析を実行できます。例えば、上部のプレーンのオブジェクトがクライアント IP を表す場合、ソース・プログラムの色分けを選択すると、特定の IP クライアントによるさまざまなソース・プログラムの使用法や、異なるクライアント IP による共通のソース・プログラムの使用法を表示できます。カラーが繰り返し変更されるオブジェクトは、単一のクライアント IP 内のソース・プログラムの使用法が頻繁に変更されることを示します。 「適用」をクリックします。

表 1. データの洞察グラフの設定

フィールド	説明および値
データ・フロー・ドメイン	表示されるデータ・フローのタイプ。次のいずれかです: アクティビティー、エラー、違反、異常値。
上部のプレーン・エンティティー	上部のプレーンに表示されるエンティティー。次のいずれかです: クライアント IP、データベース・ユーザー、OS ユーザー、ソース・プログラム。
下部のプレーン・エンティティー	下部のプレーンに表示されるエンティティー。次のいずれかです: データベース、オブジェクト、サーバー。
上部のエンティティーの色のソート基準	上部のエンティティーの追加 (オプション) の色の分類基準: なし、クライアント IP、データベース・ユーザー、OS ユーザー、ソース・プログラム。
上部のプレーン・ラベルの表示	yes、no
下部のプレーン・ラベルの表示	yes、no
上部プレーンのエンティティーの最大数	上部のプレーンに表示されるエンティティーの最大数。
下部プレーンのエンティティーの最大数	下部のプレーンに表示されるエンティティーの最大数。
上部のエンティティーの色	上部のプレーン・エンティティーの色を選択するためにカラー・パレットを開きます。上部のエンティティーの色のソートが設定されている場合は、無効になっています。
背景色	背景の色を選択するためにカラー・パレットを開きます。
プレーンの色	プレーンの色を選択するためにカラー・パレットを開きます (両方のプレーンに 1 色)。

3. 画面の変更基準:
   - フルスクリーン・モードで詳細を表示するには、拡大アイコンをクリックします
   - ビューを回転させるには、左マウス・ボタンを押したままドラッグします
   - 水平移動するには、右マウス・ボタンを押したままドラッグします
   - ズームインおよびズームアウトするには、マウス・ホイールを使用します
4. エンティティーの表示基準:
   - 凡例で詳細を表示するには、エンティティーの上にカーソルを移動します
   - エンティティーのデータ・フローのみを表示するには、そのエンティティーをクリックします (その他のエンティティーはフェードアウトします)。終了するには、背景をクリックします。
   - (ダッシュボード全体に対する) アクティブなフィルターとして使用するには、エンティティーをダブルクリックします
5. 右上隅にある情報ペインには、現在表示されているアクションのタイム・スタンプ、これまで表示されたアクションの数、および 1 秒当たりのイベントのレートの表示が示されます。この表示は、以下のように変更できます。

	データ・フローの一時停止/再開

	
	データ・フローを期間の最初から再開
	データ・フローの速度を上げる
	データ・フローの速度を下げる
	上部から表示 (鳥瞰)
	側面から表示 (デフォルト)

6. 必要に応じて、制御パネルの上の以下のボタンを使用します。

	データの洞察のグラフのフルスクリーン・モードをアクティブにします
	「グラフ設定」を開きます
	データの洞察のグラフを閉じます
	ポップアップ・ヘルプを開きます

親トピック: 調査ダッシュボード

Outliers Detection

2 つの簡単なステップで Outliers Detection を有効にして、Outliers Detection の監査を開始できます。これにより、Guardium が異常なサーバーの動作とユーザーの動作を識別し、考えられる攻撃を早期に検出するための処理を行えるようになります。

異常値とは、特定のデータベースまたはユーザーによるアクティビティーの「通常」の時間フレームまたは範囲から外れた特定の期間または範囲に発生した、特定のソース (DAM では、データベースまたはデータベース上の特定ユーザー。FAM では、サーバーまたは OS ユーザー) による動作のことです。異常値は、アクティビティー自体が既存のセキュリティー・ポリシーに直接的に違反していなくても、セキュリティー違反の発生を示している可能性があります。

疑わしい異常値として識別されるユーザー・アクティビティーには以下があります。

- ユーザーが初めて表にアクセスする
- ユーザーが以前は選択したことのない表内の特定のデータを選択する
- エラーのボリュームが例外的に多い。例えば、アプリケーションがこれまでにない大量の SQL エラーを生成した場合です。この場合、SQL インジェクション攻撃が進行中であることを示している可能性があります。
- アクティビティー自体は珍しいものではないが、アクティビティーのボリュームが異常である
- アクティビティー自体は珍しいものではないが、アクティビティーの発生時刻が異常である。例えば、データベース管理者が特定の表にこれまでになく頻繁にアクセスしている場合です。この場合、データベース管理者がデータを少しずつ時間をかけてダウンロードしていることを示している可能性があります。

疑わしい異常値として識別されるデータベース・アクティビティーには以下があります。

- エラーのボリュームが例外的に多い
- アクティビティー自体は珍しいものではないが、アクティビティーのボリュームが異常である
- アクティビティー自体は珍しいものではないが、アクティビティーの発生時刻が異常である

異常値マイニングによる検出結果は、調査ダッシュボード (クイック検索) およびレポートで使用可能になります。

異常値マイニングは、セキュリティー・ポリシーで監査済みのデータを対象とします。異常値について評価する対象のデータが、セキュリティー・ポリシーで監査済みであることを確認してください。

Outliers Detection は、以下で実行できます。

- 中央マネージャー。アグリゲーターのコレクター (Outliers Detection をローカルで実行しているコレクターを除く) からのデータを使用します。
- コレクター。コレクターが所有するデータのみを使用します。
- 別の CM によって管理されているアグリゲーターからデータを受信する中央マネージャー。これは、複数 CM 環境です。

- 異常値検出のクイック・スタート
  異常値を有効にし、いくつかの簡単なステップでアラートの受信を開始する方法について説明します。
- 異常値検出の有効化と無効化
  集中型環境、複数 CM 環境、またはスタンドアロン・コレクターのユニットからの異常値検出を有効/無効にします。
- 調査ダッシュボードでのデータ異常値の解釈
  Guardium には、アルゴリズムによって検出された異常値を特定し、それに応答するための便利なグラフィカル・インターフェースが用意されています。
- 調査ダッシュボードでのファイル・アクティビティーの異常値の解釈
  調査ダッシュボードのアクティビティー・グラフと結果表でファイル・アクティビティー・モニターの異常値を確認するか (調査ダッシュボードが有効にされている必要があります)、「Analytic 異常値リスト」レポートをレビューします。
- 異常値マイニング状況のモニター
  「異常値マイニングの状況」ページを使用して、プロセスが実行される特定のユニット、および CM またはアグリゲーターのどちらかとの両方での異常値マイニング・プロセスをモニターします。
- 異常値検出で使用するユーザーとオブジェクトのグループ化
  デフォルトの異常値検出アルゴリズムにグループ (ユーザー・グループ、オブジェクト・グループなど) を追加する方法を説明します。
- 異常値検出からのイベントの除外
  特定のイベント (テスト・データからのアクティビティーなど) を、異常値検出から除外することができます。

親トピック: モニターおよび監査

異常値検出のクイック・スタート

異常値を有効にし、いくつかの簡単なステップでアラートの受信を開始する方法について説明します。

始める前に

- 異常検出が有効になっています (「設定」 > 「ツールとビュー」 > 「異常検出」)。

このタスクについて

異常値検出は、任意の数のアグリゲーターで実行可能です。ただし、1 つのアグリゲーターから開始し、構成を詳細化してから、アグリゲーターを追加して拡張することをお勧めします。開始する前に、異常値の調査に使用可能なリソースを判別します。次に、毎日報告される異常値の数を、調査可能な量に制限します。Guardium アルゴリズムでは、調査する必要がある最重要のイベント (例えば、「トップ 10」だけではない) が提供されます。

異常値検出はセキュリティー・ポリシー・ルールおよび適用とは別のプロセスであるため、異常値でリアルタイム・アラートをセットアップすることはできません。ただし、異常値データはレポートに含まれるため、相関アラートを作成することができます。相関アラートは、指定された期間をさかのぼって、アラートしきい値が満たされたかどうかを判別する照会によって起動されます。

手順

1. 異常値を有効にするには、 異常値検出の有効化と無効化を参照してください。
2. オプションとして、異常値定義を微調整します。『異常値検出で使用するユーザーとオブジェクトのグループ化』および『異常値検出からのイベントの除外』を参照してください。
3. 照会を作成します。
    a. 「レポート」 > 「レポート構成ツール」 > 「照会 - レポート・ビルダー」にナビゲートします。
    b. ドメイン =「分析 (analytic)」、照会名 =「Analytic 異常値リスト」または「Analytic 日付別異常値サマリー」と設定します。他のすべての設定はデフォルトのままにしておいてかまいません。
    c. 「レポートの作成」をクリックします。
4. 監査プロセスを作成します。
    a. 「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」にナビゲートします。
    b. プロセスに名前を付け、タスク (直前に作成したレポート) を追加します。
    c. レシーバーを定義します。必要な通知の種類を決定します。アラートをセットアップし、それを To-do リストに追加し、ユーザーが検出結果をレビューおよび正当化するように割り当てます。
    d. プロセスを日次でスケジュールし、「保存」をクリックします。
5. すぐに表示できるように、異常値レポートを「マイ・ダッシュボード」に追加します。

タスクの結果

ラーニング期間が完了すると、レポートにデータが含まれるようになり、アラートが送信されます。
親トピック: Outliers Detection

異常値検出の有効化と無効化

集中型環境、複数 CM 環境、またはスタンドアロン・コレクターのユニットからの異常値検出を有効/無効にします。

始める前に

- 異常値は 24 ギガバイト以上のメモリーを備えた 64 ビット・アグリゲーターでのみ有効化にすることを強くお勧めします。

このタスクについて

制約事項: Outliers Detection とデータ・レベル・セキュリティーを同時に有効にすることはできません。

異常値検出は、デフォルトでは無効になっています。どのトポロジーの、どんな Guardium システムの場合でも、異常値検出を有効および無効にするためには、1 つの API コマンド grdapi enable_outliers_detection が使用されます。

このコマンドが Guardium システムに与える影響は、その Guardium システムのセットアップに応じて異なります。

単一 CM 環境
: 追加のパラメーターを指定せずに API コマンドを実行することにより、CM で異常値検出を有効にして、すべての管理対象ユニット、およびその後に CM に登録されたすべてのユニットに対して異常値検出を有効/無効にします。あるいは、MU グループまたはユニットのリストに制限して有効/無効にすることができます。 同様に、CM に対して異常値検出を無効にすると、CM に登録されているユニットに対しても無効になります。
アグリゲーターにデータを抽出するコレクターに対して異常値検出を有効にします。異常値検出がアグリゲーターに対して有効になり (まだ有効になっていない場合)、コレクターはアグリゲーターへのデータ送信を開始します。コレクターに対して無効にすると、これがアグリゲーターにデータを送信する唯一のコレクターである場合は、コレクターはデータ送信を停止して、異常値検出はアグリゲーターに対して無効になります。

複数 CM 環境
: 追加のパラメーターを指定せずに API コマンドを実行することにより、CM で異常値検出を有効にして、すべての管理対象ユニット、およびその後に CM に登録されたすべてのユニットに対して異常値検出を有効/無効にします。あるいは、MU グループまたはユニットのリストに制限して有効/無効にすることができます。 同様に、CM に対して異常値検出を無効にすると、CM に登録されているユニットに対しても無効になります。
コレクターと同じ CM 環境にないアグリゲーターにデータを抽出するコレクターに対して有効にする場合、コレクターはアグリゲーターへのデータ送信を開始して、システムはアグリゲーターに対して異常値検出を有効にするようにユーザーに指示します。
アグリゲーターに対して有効にする場合、異常値検出が有効になり、同じ CM 環境にあるコレクターがデータ送信を開始します。別の CM 環境にあるコレクターからアグリゲーターがデータを受信する場合、システムは、異常値検出を有効にする必要があるすべてのコレクターのメッセージを出力します。

単一のコレクター
: アグリゲーターにデータを抽出しないコレクターでコマンドを実行して、ローカル側で有効/無効にします。

手順

1. CLI ロールを持つユーザーまたは管理者として Guardium システムにログインします。
2. CM にあるすべてのユニット、およびその後で CM に登録されたすべてのユニットに対して異常値検出を有効にするには、grdapi enable_outliers_detection と入力します。 オプション・パラメーターは以下のとおりです。
   - unitGroup: 既存のグループのグループ ID。CM に関連しており、オプションです。
   - unitlist: ユニットのコンマ区切りリスト。CM に関連しており、オプションです。
   - FAM_DAM は、異常値のタイプを指定するオプション・パラメーターです。デフォルトは DAM です。

   パラメーター schedule_interval および schedule_units は無視されます。
3. 異常値検出機能を無効にするには、次のコマンドを入力します。オプション・パラメーターは指定しても指定しなくてもかまいません。

   ```
   grdapi disable_outliers_detection
   ```

タスクの結果

有効にすると、システムは、異常値データの収集を開始します。ラーニングが完了すると (14 日間)、異常値データが調査ダッシュボード (調査ダッシュボードでのデータ異常値の解釈および調査ダッシュボードでのファイル・アクティビティーの異常値の解釈を参照) および「異常値分析リスト」レポートで使用可能になります。

親トピック: Outliers Detection

調査ダッシュボードでのデータ異常値の解釈

Guardium には、アルゴリズムによって検出された異常値を特定し、それに応答するための便利なグラフィカル・インターフェースが用意されています。

調査ダッシュボードで異常値検出データを表示するには、クイック検索を有効にする必要があります (grdapi enable_quick_search)。

例えば、ユーザー X に対して並外れた数のエラーを示す異常値があるとします。この場合、調査しなければならない点には、以下があります。

- 履歴を調べ、このユーザーに異常値が示されたのは、これが初めてであるかどうかを確認します。また、ユーザーに初めてこのタイプの異常値が示されたのかどうかも確認します。
- このユーザーを他のユーザーと比較して、このエラー・タイプはこのユーザーに固有であるかどうかを確認します。
- エラー・タイプを確認します。
- エラーの数が、このユーザーには標準的な数であるかどうかを調べます。
- ユーザーがアクセスした表の重要度を調べます。
- アクションを他のデータベースと比較します。

異常値の調査フローを辿ります。

1. 「データ」を選択するか、「ユーザー・インターフェース」ドロップダウンを使用し、「入力 (Enter)」をクリックして調査ダッシュボードを開きます、または、検索フィールドにクイック検索を入力して、「データ・アクティビティーの検索」をクリックし、アクティビティー・グラフを追加します (「グラフの追加」 > 「アクティビティー・グラフ」)。(ウィンドウの上部で、グラフの時間間隔を変更できます)。赤色のインディケーターは、直ちに対応する必要のある異常性の高いイベントを反映しています。黄色のインディケーターは、他の調査または関連する調査の一環として注意を払う必要のある、より逸脱度の低い異常を表しています。
2. 異常値アイコンの上にカーソルを移動すると、ポップアップに詳細が表示されます。ここで「詳細を表示」をクリックすることで、結果表をフィルタリングして、同じ期間に発生したアクティビティーまたは異常値を表示できます。
3. 異常値をクリックして「異常値」ビューの「要約」タブを開きます。このタブに、選択されている期間に異常値が検出されたソースの数と、重大な異常値および中程度の異常値が示されます。
   - ファセット・リスト、個々の検索結果、または右クリック・メニューのいずれかを使用して、表内のデータをフィルタリングします。
   - 関連するアクティビティー、関連する例外、関連する違反を表示するには、異常値表の右クリック・メニューを使用します。
4. 特権ユーザーだけをモニターして、データを除外して焦点を絞り込めるよう試みます。
   - 特権ユーザーのアクティビティーのパターンと使用法について深い洞察を得ることができます。例えば、以下のことがわかる場合があります。
     - 特定のデータにアクセスしてはならないユーザー。
     - 異常に見える SQL アクティビティー (特権ユーザーが自分のアクティビティーを SQL 攻撃で隠ぺいしている可能性があります)。SQL インジェクション攻撃の特性 も参照してください。
   - 時刻に関する異常値を確認します。
5. 機密オブジェクトだけをモニターして、データを除外して焦点を絞り込めるよう試みます。
   - それらの機密オブジェクトにアクセスするユーザーのパターンと使用法について深い洞察を得ることができます。それらのオブジェクトに対する異常なアクセス・パターンを確認できる場合があります。
   - 時刻に関する異常値を確認します。
   - それらのオブジェクトに、どのユーティリティー (ソース・プログラム) がアクセスしたのかを調べます。

結果表の「異常値」タブには以下の 2 つのビューがあります。

- 「要約」には、異常値が検出された 1 時間当たりのソースごとの 1 行があり、異常スコアと 1 つ以上の理由が示されます。「要約」タブに表示されるすべての異常値について、「詳細」タブにその詳細が示されるわけではないことに注意してください。
- 「詳細」は、発生したイベントのサンプルであり、イベントごとの 1 行と、その理由 (さまざまである場合を除く。表を参照) およびその他の詳細 (ソース・プログラム、オブジェクト、動詞など) があります。例えば、「大量」の場合、サンプリングはスコアが最も高いイベントを表します。「要約」タブの異常値ごとに「詳細」タブに表示するサンプル (行) の数を構成できます。

この表に、「要約」ビューと「詳細」ビューの両方に含まれる列の説明を記載します。

列名	記述	以降のアクション

列名	記述	以降のアクション
異常スコア	「要約」タブ: 異常値の量、個々のイベントの重大度、特定の時刻における異常値の予測量などの要因に基づいて計算される集約値。例えば、通常は平日の午前 1 時に異常値が 0 個、午後 1 時に異常値が 5 から 10 個特定されるシステムで、異常値が 2 個余分に検出された場合 (午前 1 時に 2 個、または午後 1 時に 12 個) は、1 時間ごとの総数そのものよりも重要な結果 (より高く重み付けされる結果) となります。「詳細」タブ: 異常スコアは、「大量」イベントにのみ関連します。	スコアを右クリックすると、実行可能な他のアクションを選択できるメニューが開きます。「詳細」タブでは、スコアが 0 として示される場合があります。これは、個々のイベント自体は疑わしくないものの、該当する 1 時間にわたって累積されたイベントは疑わしいことを意味します。
大量の異常値	True または False。データベース・ユーザーの何らかのタイプのアクティビティーが (例えばオブジェクトに対して) 大量であることを示します。	
新規異常値	True または False。オブジェクト/動詞のアクティビティーが、以前のアクティビティーと比較して異常に多く発生しています。例えば、管理者がいつになく多数の新しい表を作成していたり、更新が以前に行われたことがないのにユーザーが多数のオブジェクトを選択して更新を実行したりする場合です。	
さまざまな異常値	「要約」ビューのみ。True または False。さまざまなタイプのアクティビティー (例えば、データベース・ユーザーが通常より多様なアクティビティーを行っている、またはそれらのアクティビティーを通常とは異なる時間に行っているなど) が大量であることを示します。「詳細」タブには、さまざまなイベントのサンプルが表示され、データベース・ユーザーによって識別できます。「さまざまな異常値」は「詳細」タブの列ではありませんが、それらには他の理由が割り当てられている場合もあります。 そうでない場合、理由なしで表示されます。	詳しくは、「アクティビティー」表を確認してください。
エラー異常値	True または False。エラーが大量であることを示します。	
現行の異常値	「要約」ビューのみ。True または False。過去数時間に、異常値を生成するまでではないものの、疑いを生じさせるイベントがあることを示します。	表示する必要がある特定のイベントはありません。「アクティビティー」表を表示し、ファセット・リストでデータベース別にフィルタリングして、疑わしい動作の時刻に時間間隔を変更します。
インスタンスの数	「詳細」ビューのみ。該当する 1 時間で、この特定のイベントが確認された回数。	
影響を受けるレコード	特定のイベントによって影響を受けたレコードの数。イベントが本質的にレコードに影響していない場合、負数として表示されます。	

親トピック: Outliers Detection
関連情報:
異常検出

調査ダッシュボードでのファイル・アクティビティーの異常値の解釈

調査ダッシュボードのアクティビティー・グラフと結果表でファイル・アクティビティー・モニターの異常値を確認するか (調査ダッシュボードが有効にされている必要があります)、「Analytic 異常値リスト」レポートをレビューします。

調査ダッシュボードで異常値検出データを表示するには、クイック検索を有効にする必要があります (grdapi enable_quick_search)。

一般的なワークフローのガイドラインを以下に示します。

1. 「ファイル」を選択するか、「ユーザー・インターフェース」ドロップダウンを使用し、「入力 (Enter)」をクリックして調査ダッシュボードを開きます、または、検索フィールドにクイック検索を入力し、「ファイル・アクティビティーの検索」をクリックしてアクティビティー・グラフで異常値を表示します。(ウィンドウの上部で、グラフの時間間隔を変更できます)。赤色のインディケーターは、直ちに対応する必要のある異常性の高いイベントを反映しています。黄色のインディケーターは、他の調査または関連する調査の一環として注意を払う必要のある、より逸脱度の低い異常を表しています。
2. 異常値アイコンの上にカーソルを移動すると、該当する期間に検出された異常値に関する詳細情報が表示されます。
3. 結果表をフィルタリングして、同じ期間に発生したアクティビティーまたは異常値を表示するには、「詳細を表示」をクリックします。
4. 異常値をクリックして「異常値」ビューの「要約」タブを開きます。このタブに、選択されている期間に異常値が検出されたソースの数と、重大な異常値および中程度の異常値が示されます。
   - ファセット・リスト、個々の検索結果、または右クリック・メニューのいずれかを使用して、表内のデータをフィルタリングします。
   - 関連するアクティビティー、関連する例外、関連する違反などを表示するには、異常値表の右クリック・メニューを使用します。
5. 特権ユーザーだけをモニターして、データを除外して焦点を絞り込めるよう試みます。
   - 特権ユーザーのアクティビティーのパターンと使用法について深い洞察を得ることができます。例えば、以下のことがわかる場合があります。
     - 特定のデータにアクセスしてはならないユーザー。
   - 時刻に関する異常値を確認します。
6. 機密オブジェクトだけをモニターして、データを除外して焦点を絞り込めるよう試みます。
   - それらの機密オブジェクトにアクセスするユーザーのパターンと使用法について深い洞察を得ることができます。それには、例えば機密データが含まれるファイル・サーバーのグループを作成します。それらのオブジェクトに対する異常なアクセス・パターンを確認できる場合があります。
   - 時刻に関する異常値を確認します。
   - それらのオブジェクトに、どのユーティリティー (ソース・プログラム) がアクセスしたのかを調べます。
7. モニターを続行します。
   - アラート設定: (「Analytic 異常値サマリー」レポートに基づいて) 異常な時間帯に対するアラートを設定します。
   - 監査: 「異常値のレビュー」を定義 (「Analytic 異常値リスト」レポートで監査プロセスを定義) して、適切なロール/ユーザー・グループに割り当てます。

結果表の「異常値」タブには以下の 2 つのビューがあります。

- 「要約」には、異常値が検出された 1 時間当たりのソースごとの 1 行があり、異常スコアと 1 つ以上の理由が示されます。「要約」タブに表示されるすべての異常値について、「詳細」タブにその詳細が示されるわけではないことに注意してください。
- 「詳細」は、発生したイベントのサンプルであり、イベントごとの 1 行と、その理由およびその他の詳細があります。例えば、「大量」の場合、サンプリングはスコアが最も高いイベントを表します。「要約」タブの異常値ごとに「詳細」タブに表示するサンプル (行) の数を構成できます。

この表に、「要約」ビューと「詳細」ビューの両方に含まれる列の説明を記載します。

列名	記述	以降のアクション
異常スコア	「要約」タブ: 異常値の量、個々のイベントの重大度、特定の時刻における異常値の予測量などの要因に基づいて計算される集約値。例えば、通常は平日の午前 1 時に異常値が 0 個、午後 1 時に異常値が 5 から 10 個特定されるシステムで、異常値が 2 個余分に検出された場合 (午前 1 時に 2 個、または午後 1 時に 12 個) は、1 時間ごとの総数そのものよりも重要な結果 (より高く重み付けされる結果) となります。「詳細」タブ: 異常スコアは、「大量」イベントにのみ関連します。	スコアを右クリックすると、実行可能な他のアクションを選択できるメニューが開きます。「詳細」タブでは、スコアが 0 として示される場合があります。これは、個々のイベント自体は疑わしくないものの、該当する 1 時間にわたって累積されたイベントは疑わしいことを意味します。
大量の異常値	True または False。データベース・ユーザーの何らかのタイプのアクティビティーが (例えばオブジェクトに対して) 大量であることを示します。	
新規異常値	True または False。新しいオブジェクトに対するアクティビティー (例えば、管理者がいつになく多数の新しい表を作成するなど) が大量であることを示します。	
エラー異常値	True または False。エラーが大量であることを示します。	
現行の異常値	「要約」ビューのみ。True または False。過去数時間に、異常値を生成するまでではないものの、疑いを生じさせるイベントがあることを示します。	表示する必要がある特定のイベントはありません。「アクティビティー」表を表示し、ファセット・リストでデータベース別にフィルタリングして、疑わしい動作の時刻に時間間隔を変更します。
インスタンスの数	「詳細」ビューのみ。該当する 1 時間で、この特定のイベントが確認された回数。	
サーバー	イベントが発生したサーバー。	
OS ユーザー	イベントを実行した OS ユーザー。	
特権ユーザー	True または False。ユーザーが特権ユーザーであるかどうかを示します。	
ファイルの絶対パス名	ユーザーがイベントを実行した対象のファイルの名前。	
コマンド	ユーザーがイベントの実行で使用したコマンド。	
日付	yyyy-mm-dd 形式で表記された、イベントが発生した日付。	
時刻	hh:mm:ss 形式で表記された、イベントが発生した時刻。	

親トピック: Outliers Detection
関連概念:
調査ダッシュボード
関連情報:
GuardAPI Outliers Detection 機能
異常検出

異常値マイニング状況のモニター

「異常値マイニングの状況」ページを使用して、プロセスが実行される特定のユニット、および CM またはアグリゲーターのどちらかとの両方での異常値マイニング・プロセスをモニターします。

CM で表示される「異常値マイニングの状況」ページには、すべての管理対象アグリゲーターとそれらのアグリゲーターのコレクターの詳細が表示されます。CM 内のすべてのコレクターは、それぞれのアグリゲーターの下に個別の行で表示されます。別の CM 環境にあるアグリゲーターにデータをエクスポートしているコレクターの状況の詳細は限られています。

アグリゲーターで表示すると、このウィンドウには、その特定のアグリゲーターのコレクターの詳細が表示されます。

コレクターから表示すると、1 つのコレクターだけが表示されます。

このページには、「管理」 > 「保守」 > 「異常値マイニングの状況」からアクセスします。

以下の表に、このページおよび推奨されるユーザー・アクションについての説明を記載します。

列	記述	アクション
>	このアグリゲーターにデータを送信するユニットのリストを開いたり閉じたりします。	クリックしてユニットのリストを表示します。
ユニット	ユニットの名前	NA
ユニット・タイプ	コレクター、アグリゲーター、中央マネージャーのいずれか	
ユニットのオン/オフ	ユニットがオンであるか、オフであるかを示します。	NA
異常値マイニングが有効/無効 (Outlier Mining Enabled/Disabled)	• アグリゲーター: アグリゲーターでの異常値マイニングが有効になっているかどうかを示します。無効になっている場合、この列の下にある残りの行は空になります。 • 単一のコレクターまたはスタンドアロン・ユニットの個別の行: 緑色は、異常値マイニングがローカルで有効にされていることを示します。	NA

列	記述	アクション
異常値マイニングのデータを送信 (Send data for outlier mining)	コレクターのみ。コレクターは異常値マイニング・データをアグリゲーターに送信します。アグリゲーターで異常値マイニングが有効にされており、コレクターがローカルで異常値マイニングを実行していない場合、コレクターからアグリゲーターに異常値マイニングのデータが送信されます。	NA
最後の異常検出日	1 つ以上の異常 (異常値) が検出された最後の異常値マイニング実行の CM でのローカル日時。このデータは、バージョン 10.1.2 以降を実行しているユニットの場合にのみ表示されます。	NA
最後の分析日	最後の異常値マイニング実行の CM のローカル日時 (プロセス終了日時)。	NA
分析状況	最後の異常値マイニング実行の状況。 緑色: プロセスは正常に終了しました。 オレンジ色: プロセスは警告で終了しました。 赤色: プロセスはエラーで終了しました。 このデータは、バージョン 10.1.2 以降を実行しているユニットの場合にのみ表示されます。	エラー/警告が発生したのが 1 回だけである場合、(次の 1 時間に) プロセスをもう一度実行させて、その結果を確認します。エラーが繰り返される場合は、サポートに連絡してください。
詳細	状況は、赤色 (エラー)、黄色 (警告)、または緑色で示されます。	警告 (黄色) で終了したプロセスの場合、クリックするとポップアップが開き、警告が表示されます。 エラー (赤色) で終了したプロセスの場合、クリックするとポップアップが開き、エラーが表示されます。
ラーニング開始日	異常値マイニング・プロセスが有効化された日時。その時点から、プロセスはリソースの動作のラーニングを開始します。	NA
クイック検索のオン/オフ	管理対象ユニットで、クイック検索および Solr が有効にされているかどうかを示します。クイック検索が無効にされている場合、そのマシンのデータは調査ダッシュボードに表示されません。	調査ダッシュボードの有効化と無効化を参照してください。
最後の情報更新	この行の情報が最後に更新された日時。通常は、約 5 分間隔でデータが更新されます。	NA
最後に受信した異常値データ	アグリゲーターにデータを抽出するユニットに関連します。	

親トピック: Outliers Detection

異常値検出で使用するユーザーとオブジェクトのグループ化

デフォルトの異常値検出アルゴリズムにグループ (ユーザー・グループ、オブジェクト・グループなど) を追加する方法を説明します。

このタスクについて

デフォルトでは、Guardium® の機械学習アルゴリズムで、比較的高い重み付けやスコアを付与される 2 つのユーザー・グループとオブジェクト、つまり管理ユーザーと機密オブジェクトがあります。しかし、異常値検出にも使用できる別のグループが既に作成されている場合があります。例えば、疑わしいユーザーのグループや、さまざまなアプリケーションに応じた機密オブジェクトのさまざまなグループが存在する場合があります。

手順

1. このタスクを実行するには、grdapi コマンドで使用する内部グループ ID を知っている必要があります。グループ ID を取得するには、`grdapi list_group_by_desc desc=[group name]` というコマンドを使用します。例えば、BadGuys という名前のグループがある場合は、以下のコマンドを入力することで、その内部グループ ID を取得できます。

   ```
   grdapi list_group_by_desc desc=”BadGuys”
   ```

2. 目的の ID が判明したら、以下のようにランキング調整されたスコアの特権ユーザー・グループとして追加します (デフォルト・グループ 1 のスコアのランキングを調整する場合も、そのグループを含める必要があることに注意してください)。 ID 1234 のグループを追加する場合: grdapi set_outliers_detection_parameter parameter_name="privUsersGroupIds" parameter_value=1,1234
3. ID 333 および ID 156 の機密オブジェクトを追加する場合: set_outliers_detection_parameter parameter_name="sensitiveObjectGroupIds" parameter_value=5,333,156

タスクの結果

指定したグループまたは機密オブジェクトが異常値検出に追加され、アルゴリズムによって追加の重み付けが指定されます。

親トピック: Outliers Detection

異常値検出からのイベントの除外

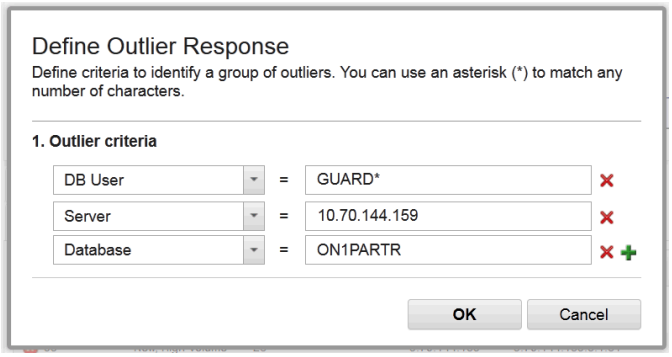
特定のイベント (テスト・データからのアクティビティーなど) を、異常値検出から除外することができます。

異常値応答を使用した、特定の基準に一致するイベントの除外

1. 異常値応答を除外するには、異常値インディケーターを右クリックし、「無視」を選択します。
2. 特定の値を入力するか、ワイルドカード・エントリーを使用して (* 文字を使用)、無視するイベントを定義します。

3. 該当するフィールドの ✖ アイコンをクリックして、不要なフィールドを削除します。
4. 「OK」をクリックして変更をコミットします。
5. 以前に無視したイベントを無視しないようにするには、「Analytic ユーザー・フィードバック」レポートを表示し、以前に無視したイベントをダブルクリックして、「呼び出し」 > 「delete_analytic_user_feedback」を選択します。

例えば、サーバー 10.70.144.159、データベース ON1PARTR、および名前が GUARD で始まるすべてのデータベース・ユーザーからのすべてのアクティビティーを無視する場合、ダイアログは以下のようになります。

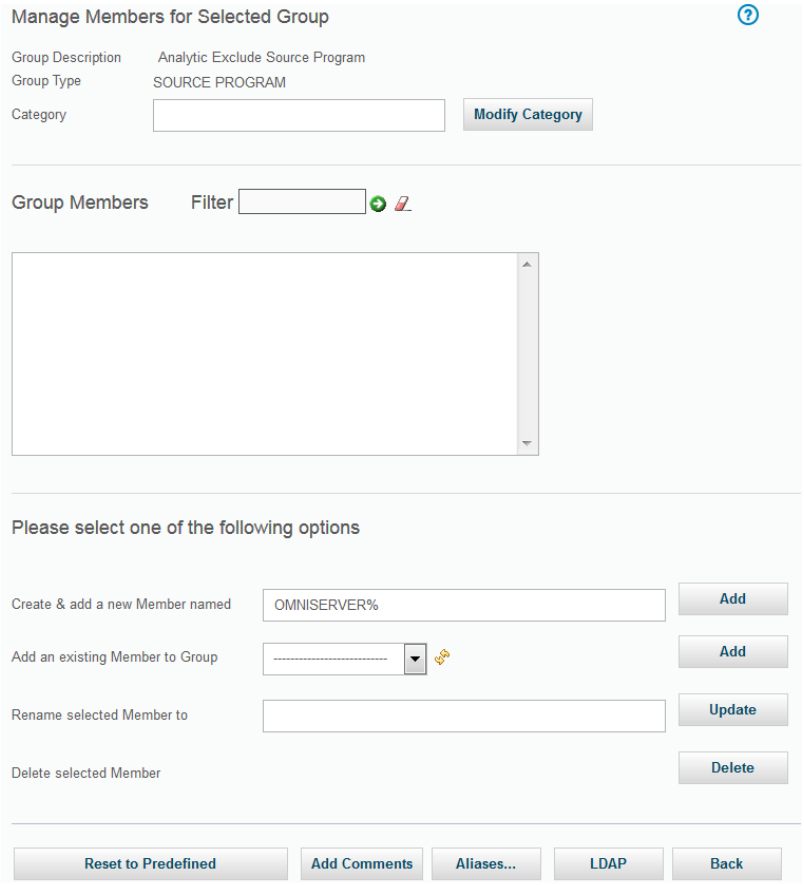


「グループ・ビルダー」を使用したイベントの除外

除外対象の項目が多数ある場合は、「グループ・ビルダー」を使用して、必要に応じて以下のグループのいずれかまたはすべてを設定します。

- Analytic 除外データベース・ユーザー
- Analytic 除外 OS ユーザー
- Analytic 除外サーバー IP
- Analytic 除外サービス名
- Analytic 除外ソース・プログラム

「グループ・ビルダー」には、カスタム表に対する照会から値を読み込む機能など、一括アップロードのためのオプションが用意されています。



別の方法として、GuardAPI コマンドを使用して、Analytic 除外グループを設定することもできます。例えば、「Analytic 除外ソース・プログラム」グループに OMNISERVER を追加するには、以下のコマンドを使用します。

```
grdapi create create_member_to_group_by_desc  desc="Analytic Exclude Source Program" member="OMNISERVER%"
```

親トピック: Outliers Detection

データ保護ダッシュボード

Guardium のデータ保護ダッシュボードは、上級セキュリティー担当者のためにリスクおよびコンプライアンスのデータの要約ビューを提供します。

データ保護ダッシュボードには、コンプライアンスとリスクの統計に加えて複数の図表とグラフが含まれており、大きなモニターに連続的に表示されるように設計されています。このダッシュボードを開くには、「調査」 > Guardium「データ保護ダッシュボード」にナビゲートします。

注意:
データ保護ダッシュボードを表示している間はセッションの有効期限が切れたり、ユーザーが自動的にログアウトすることはありません。 長い時間ダッシュボードを開いたままにする場合は注意してください。
情報:

- ダッシュボードは 20 分ごとに自動的に最新表示されます。
- デフォルトの検索設定は、1 日前から収集されたデータを使用した分散検索です。

図表とグラフ

いくつかの線グラフを使用すると、異なるタイプのデータを素早く比較できます。 例えば、グラフには一定期間におけるアクティビティー、エラー、および違反のボリュームを表示できます。

「異常アクティビティー」グラフには、アクティビティー全体に関する異常値の要約が表示されます。 このグラフでは、異常値の要約を示す点線が異常値の予期しないボリュームを示します。

情報: これらのグラフの Y 軸はログの軸であり、グラフの比率をゆがめる可能性があります。値やカウントはログに記録されていません。

リスクとコンプライアンスの統計

「リスク」統計には、「クリティカル」重大度で失敗したテストの数と、その失敗が発生したデータ・ソースの数が示されます。 各データ・ソースには、失敗したテストが複数含まれる可能性があります。

「モニター対象データ・ソース」には、システムがアクティビティーをログに記録しているデータ・ソースの数が示されます。この統計は、利用可能なアクセス・ドメイン・データを調べることで計算されます。

「コンプライアンス To-do リスト・タスク」には、監査プロセスの以下の要約が表示されます。当日にクローズされたプロセスの数、オープンしていたのが 3 日未満のプロセスの数、4 日以上オープンしているプロセスの数。

情報:

- ファセットおよびテキストの検索フィルターにより影響を受けないが、検索モードに影響を受ける統計。検索モードを変更するには、制御を使用して上部ペインを拡張した後、アイコンをクリックして分散検索とローカル検索を切り替えます。
- 統計コンポーネントは 1 時間に 1 回再計算されます。

親トピック: モニターおよび監査

レポート

データ保護を行うには、データとファイルのアクティビティーをモニターする必要があります。Guardium は、データベース環境および Guardium システムに関する大量のデータを収集します。このデータはレポートで表示できます。

Guardium は、多数の事前定義レポートを含む高度なレポート・ツールを提供します。ただし、独自のカスタム・レポートを作成したり、既存のレポートを変更したりすることができます。

各レポートには、レポートに固有の照会によって定義された情報が示されます。照会は、収集される情報およびレポートでの表示方法を定義します。照会は、その他の目的で情報を収集するためにも使用されます。例えば、照会結果からグループを取り込むことができます。

ダッシュボードでは、レポートを表示するためにグループ化できます。カスタム・レポートおよび事前定義レポートをダッシュボードおよび「マイ・カスタム・レポート」に追加できます。

- 定義済みレポート
  カスタム・レポートを最初から作成する前に、事前定義されている Guardium レポートを利用してください。
- ダッシュボードの作成およびレポートの追加
  1 つ以上のダッシュボードを作成し、それらにレポートを追加し、外観を構成することができます。
- レポートの表示
  レポートの表示には、ダッシュボードや UI 検索など、いくつかの方法があります。
- 照会 - レポート・ビルダーの使用
  事前定義レポートではニーズに対応できない場合は、照会を最初から作成するか、既存の照会をコピーして変更します。
- ドメイン、エンティティー、および属性
  各ドメインでは、特定の目的や機能 (データ・アクセス、例外、ポリシー違反など) に関連するデータ・セットが Guardium に保管されます。データはエンティティーごとにグループ化されます。エンティティーは、関連する属性の集合で、属性は基本的にフィールドの値です。
- カスタム・ドメイン
  カスタム・ドメインではユーザー定義のドメインが可能であり、Guardium システムにアップロードされる任意のデータ表を定義できます。
- データマート
  データを後で使用するために抽出するには、データマートを使用します。データマートは、アクセス頻度の高いデータをさらに効率的に保管するために使用します。データをパージ時間の後に保存したり、Guardium からデータをエクスポートしたり、配布レポートを作成したりする場合に使用します。
- 配布レポート・ビルダー
  この中央マネージャー機能により、特定の中央マネージャーに関連付けられているすべてまたは一部の Guardium 管理対象ユニットから、データを自動的に収集す

ることができます。配布レポートは、概要ビューの提供、データ・ソース間のデータの関連付け、およびデータのビューの要約を行うように設計されています。 コレクター間での行レベル・データ収集については、引き続きアグリゲーターを使用します。

- API 呼び出しおよびレポートの操作
- 外部フィードの操作
  Guardium レポート・データを外部データベースに直接送信するには、外部フィードを使用します。レポート・データを外部データベースに送信することは、いくつかのシナリオで役に立ちます。例えば、Guardium データを非 Guardium データと結合または関連付ける場合や、外部ツールで Guardium データを使用する場合、あるいは、特に大規模なレポートでレコードのマシン構文解析を行う場合です。
- z/OS のレポートの作成
  組み込みレポートとサンプル照会をカスタマイズすることで z/OS データ・ソース用の Guardium レポートを作成する方法について説明します。

定義済みレポート

カスタム・レポートを最初から作成する前に、事前定義されている Guardium レポートを利用してください。

Guardium アプリケーションで使用可能な事前定義レポートにアクセスすることによって、求めている情報を迅速に取得できます。 これらの事前定義レポートは、ユーザーの必要に応じて、複製してカスタマイズすることができます。

Guardium 事前定義レポートを使用することは、推奨されるベスト・プラクティスでです。これにより、組織は、不適切に公開されたオブジェクト、過度の権限を持つユーザー、および無許可の管理アクションなどのセキュリティー・リスクを迅速かつ簡単に識別できます。 多くの事前定義レポートの例として、システム特権を持つアカウント、すべてのシステム特権および管理者特権 (ユーザー別およびロール別)、ユーザー別のオブジェクト特権、および PUBLIC アクセス権限を持つすべてのオブジェクトなどがあります。

すべてのレポートに、すべてのパラメーターと値が表示されます。パラメーターと値の表示は、任意のレポート画面で「カスタマイズ」を使用して編集できます。

ヘルプの検索機能を使用して、特定のレポートに直接進みます。 語句を引用符で囲むと、検索語が正確に定義されます。

事前定義レポートのユース・ケース

データベース管理者

- SQL エラー - SQL エラーの増加は、SQL インジェクション攻撃を示している可能性があります。
- DDL (スキーマ変更の確認) - このレポートは、DDL の要求元となるクライアント IP、メイン SQL 動詞 (特定の DDL コマンド)、およびそのレコード用にアクセスされる合計オブジェクトを表示します。
- 失敗したログイン - このレポートは、期限切れのログイン資格情報を使用したデータベースへのアクセス試行を示します。

機密保護担当者

- 失敗したログイン - データベースにアクセスしようとした、適切な資格情報を持つユーザー。
- 無効なユーザー - データベースにアクセスしようとした無効なユーザー。
- ポリシー違反 - セキュリティー・ポリシーに違反しているユーザーおよび問題。

監査員

- コンプライアンス・レポート - PCI、SOX、データ・プライバシー
- コンプライアンス・ワークフロー - サインオフおよびプロシージャーの証拠を示します。

- 事前定義管理レポート
  このセクションでは、管理ユーザー向けの事前定義レポートについて簡単に説明します。
- 事前定義ユーザー・レポート
  このセクションでは、デフォルトのユーザー・アクセス権限を持つユーザー向けのすべての事前定義レポートについて簡単に説明します。
- 共通の事前定義レポート
  このセクションでは、デフォルトのユーザー・アクセス権限またはデフォルトの admin アクセス権限のいずれかを持つユーザーが使用できるすべての事前定義レポートについて簡単に説明します。

親トピック: レポート

事前定義管理レポート

このセクションでは、管理ユーザー向けの事前定義レポートについて簡単に説明します。

注: 監視データ・レベルでのデータ・レベル・セキュリティーが有効な場合 (「グローバル・プロファイル」設定を参照)、ユーザーに対してユーザーのデータベース情報のみが表示されるよう、監査プロセスの出力がフィルタリングされます。

事前定義管理レポートはアルファベット順にリストされます。

変更されたアクティブ S-TAP

このアラートは、中央マネージャー・システムでのみ実行されます。S-TAPホスト、S-TAP バージョン、変更された S-TAP、タイム・スタンプ、およびカウントが表示されます。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	変更されたアクティブ S-TAP	使用不可

ランタイム・パラメーター	演算子	デフォルト値
期間開始	なし	なし

admin ユーザーのログイン

admin ユーザー・グループで定義したデータベース・ユーザー名によるデータベースへのログインのサマリー。 このレポートには、管理特権を持つユーザーがデータベースへのログインに使用するクライアント IP アドレス、 データベース・ユーザー名、ソース・プログラム、セッション開始日時、そのレコードのセッション総計が表示されます。

ドメイン	ベースとなる照会	メイン・エンティティー
アクセス	admin ユーザーのログイン	セッション
ランタイム・パラメーター	**演算子**	**デフォルト値**
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

統合/アーカイブ・ログ

このレポートでは、Guardium の統合アクティビティーがアクティビティー・タイプ別にリストされます。 レポートの各行には、アクティビティー・タイプ、開始時刻、ファイル名、状況、 コメント、Guardium ホスト名、パージされたレコード、期間の開始、期間の終了、 およびその行のログ・レコードの数が含まれています。 「Guardium ホスト名」ランタイム・パラメーターの設定によって、 出力を制限することができます。このパラメーターは、デフォルトでは % (すべてのサーバーを選択) に設定されています。 「パージされたレコード」列には、アクティビティー・タイプが「パージ」の場合にのみ、パージされたレコードの数が表示されます。

ドメイン	ベースとなる照会	メイン・エンティティー
統合/エクスポート/インポート	統合/アーカイブ・ログ	統合/アーカイブ・ログ
ランタイム・パラメーター	**演算子**	**デフォルト値**
期間開始	>=	NOW -1 WEEK
期間終了	<=	NOW
Guardium ホスト名	LIKE	%

すべての Guardium アプリケーション - ロール

このメニュー・ペインには、2 つのレポート (「全ロール - アプリケーション・アクセス」と 「全ロール - ユーザー」) が表示されます。

全ロール - アプリケーション・アクセス

ロールごとに、このレポートにはそのロールが割り当てられているアプリケーションの数がリストされます。 ロールが割り当てられているアプリケーションをリストするには、 そのロールをクリックして、「レコード詳細」レポートまでドリルダウンします。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	全ロール - アプリケーション・アクセス	使用不可
ランタイム・パラメーター	**演算子**	**デフォルト値**
期間開始	>=	NOW -100 MONTH
期間終了	<=	NOW

全ロール - ユーザー

ロールごとに、このレポートにはそのロールが割り当てられているユーザーの数がリストされます。 ロールが割り当てられているユーザーをリストするには、 そのロールをクリックして、「レコード詳細」レポートまでドリルダウンします。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	ロール - ユーザー	使用不可
ランタイム・パラメーター	**演算子**	**デフォルト値**
期間開始	>=	NOW -100 MONTH
期間終了	<=	NOW

アプライアンス設定

このレポートには、Guardium システムの構成設定が表示されます。 *アプライアンス設定* レポートを使用して、Guardium 設定を迅速にレビューして確認できます。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	変更されたアクティブ S-TAP	使用不可
ランタイム・パラメーター	**演算子**	**デフォルト値**
別名の表示		ラジオ・ボタン (オン、オフ、デフォルト)
リモート・データ・ソース		ドロップダウン・メニュー

アプリケーション・オブジェクト・サマリー

このレポートは、Guardium アプリケーションのすべての定義のサマリーです。 例えば、「アプリケーション・オブジェクト」の「ランタイム・パラメーター」ページの「ObjectNameLike」欄に Oracle と入力すると、Oracle が使われているオブジェクト・タイプとオブジェクトの記述がすべて検出されます。

注: このレポートでは、 メタデータが提示されるため、データ・レベル・セキュリティー・メカニズムではフィルタリングされません。 このメタデータには、Oracle SID などのデータベース関連情報が含まれていることがあります。

ドメイン	ベースとなる照会	メイン・エンティティー
アプリケーション・オブジェクト	アプリケーション・オブジェクト・サマリー	アプリケーション・オブジェクト

ランタイム・パラメーター	演算子	デフォルト値
ObjectNameLike	%	%
ObjectTypeNameLike	%	%

承認された TAP クライアント

特定の S-TAP のみ、Guardium アプリケーションへの接続が許可されます。このレポートは、承認されている S-TAP およびその状況を示します。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	承認された TAP クライアント	使用不可

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

監査プロセス・ログ

監査プロセス・ログ

このレポートには、すべてのタスクに関する詳細なアクティビティー・ログが、開始時刻および終了時刻も含めて示されます。 このレポートは、admin ユーザーが入手可能です。監査タスクには、開始および終了時刻が示されますが、セキュリティー・アセスメントおよび分類 (キューに入れられます) の開始および終了は同じになります。

監査プロセスは、その監査プロセス全体でサインオフするユーザーだけでなく、特定行のサインオフにまで拡張されています。 サインオフされたものと、特定行の状況がリスト表示されます。

監査プロセスを停止するには、この「監査プロセス・ログ」を使用してください。タスクを停止できるのは、 そのタスクが実行されていない場合、または実行中の場合に限られます。 まだ開始されていないタスクは実行されません。 部分的な結果は送信されません。 タスクが完了している場合は、監査プロセスを停止しても、結果の送信は停止されません。 監査プロセスの停止は、「監査プロセス・ログ」レポートから、 GrdAPI コマンド invoke api によって実行されます。ユーザーには、 そのユーザーに属する行だけ (ただし、すべての詳細ではなくタスクのみ) が表示されます。 管理者ユーザーは全詳細を確認でき、任意のユーザーの実行を停止できます。 ユーザーは、自分の実行しか停止できません。

監査プロセスを停止しても、リモート・ソースを使用して実行している照会は取り消されません。リモート・ソースを使用するオンライン・レポートも同様に取り消されません。

プライバシー・セットと外部フィードに対してはサポートされません。つまり、 プライバシー・セット・タスクが開始されていたり、外付けフィードが開始されていたりした場合、 プロセスが停止してもそれは完了します (一方照会は強制終了されます)。

監査プロセス・ログ ID

ログイン名

実行 ID

タイム・スタンプ

監査プロセス ID

監査プロセスの記述

監査タスク ID

監査タスクの記述

イベント・タイプ

詳細

監査プロセス・ログの数

使用可能なパッチ

使用可能なパッチのリストを表示します。ランタイム・パラメーターはありません。 このレポート・ドメインはシステム専用です。

バッファー使用状況モニター

バッファー使用状況の統計の詳細を表示します。このレポートにリストされるフィールドについては、 「スニファーのバッファー使用」エンティティーの説明を参照してください。

ドメイン	ベースとなる照会	メイン・エンティティー

ドメイン	ベースとなる照会	メイン・エンティティー
バッファー使用状況	バッファー使用状況モニター	スニファーのバッファー使用のモニター
ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

CAS デプロイメント

この CAS レポートでは、データベース・タイプ、OS 名、 ホスト名、および OS タイプが詳しく記述されています。

ドメイン	ベースとなる照会	メイン・エンティティー
CAS	CAS デプロイメント	N/A

ランタイム・パラメーター	演算子	デフォルト値
データベース・タイプ	Like	%
OS_Name	Like	%
ホスト名	Like	%
OS_Type	Like	%

変更 (CAS)

CAS 変更詳細

モニター対象の項目ごとに、所有者別に変更がリストされます。

ドメイン	ベースとなる照会	メイン・エンティティー
CAS 変更	CAS 変更詳細	ホスト構成

ランタイム・パラメーター	演算子	デフォルト値
DB_Type	Like	%
Host_Name	Like	%
Instance_Name	Like	%
Monitored_Item	Like	%
OS_Type	Like	%
タイプ	Like	%

CAS 保存データ

このレポートでは、検出された変更ごとに、保存されたデータがリストされます。このレポートは、ホスト名ごとにソートされ、次に最終変更時刻ごとにソートされます。

ドメイン	ベースとなる照会	メイン・エンティティー
CAS 変更	CAS 保存データ	保存データ

ランタイム・パラメーター	演算子	デフォルト値
Host_Name	Like	%
Monitored_Item	Like	%
Saved_Data_Id	Like	%

構成 (CAS)

CAS インスタンス

このレポートは、CAS インスタンス定義 (CAS インスタンスは、特定の CAS ホストにテンプレート・セットを適用します) をリストします。 このレポートのデフォルトのソート順は、標準のソート順とは異なります。ソート・キーは、優先順位の高いものから順に、ホスト名 (昇順)、インスタンス (昇順)、最後の状況変更 (降順) です。

ドメイン	ベースとなる照会	メイン・エンティティー
CAS 構成	CAS インスタンス	モニター項目詳細

ランタイム・パラメーター	演算子	デフォルト値
Host_Name	Like	%
OS_Type	Like	%
DB_Type	Like	%

ランタイム・パラメーター	演算子	デフォルト値
インスタンス	Like	%

CAS インスタンス構成

このレポートは、CAS インスタンスの構成変更をリストします。このレポートのデフォルトのソート順は、標準のソート順とは異なります。ソート・キーは、優先順位の高いものから順に、ホスト名 (昇順)、インスタンス (昇順)、最後の状況変更 (降順) です。以下のランタイム・パラメーターを使用することで、出力を制限できます。これらのパラメーターは、デフォルトではすべての値を選択します。

ドメイン	ベースとなる照会	メイン・エンティティー
CAS 構成	CAS インスタンス構成	モニター項目詳細

ランタイム・パラメーター	演算子	デフォルト値
Host_Name	Like	%
OS_Type	Like	%
Template_Id	Like	%

接続プロファイル・リスト

接続プロファイル・リストは、すべての許可された接続のグループです (接続プロファイル・リストは、すべての接続の詳細を示しています)。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	接続プロファイル・リスト	クライアント・サーバー
ランタイム・パラメーター	**演算子**	**デフォルト値**
照会の開始日付	>=	NOW -1 DAY
照会の終了日付	<=	NOW

隔離された接続

Guardium ポリシーを使用して接続の終了や隔離をリアルタイムで行うことができます。 照会をベースにしたしきい値アラートを使用します。構成の手順については、 トピック『ポリシー』の『隔離』を参照してください。

ドメイン	ベースとなる照会	メイン・エンティティー
接続隔離	隔離された接続	接続隔離
期間開始	>=	NOW -1 DAY

ランタイム・パラメーター	演算子	デフォルト値
サーバー IP	LIKE	%
データベース・ユーザー	LIKE	%
サーバー名	LIKE	%
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

CPU トラッカー

S-TAP ホストと、S-TAP を実行しているマシンの CPU の数をリストします。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	使用不可	使用不可

ランタイム・パラメーター	演算子	デフォルト値
なし	n/a	n/a

CPU 使用量

デフォルトでは、最近 2 時間の CPU 使用量を表示します。このグラフィカル・レポートは、 最近のアクティビティーのみを表示するためのものです。「開始」および「終了」のランタイム・パラメーターを変更して対象となる時間フレームを大きくすると、データが大きすぎるというメッセージが表示される場合があります。 より大きな時間枠を表示する場合は、表形式のレポートを使用してください。

ドメイン	ベースとなる照会	メイン・エンティティー
スニファーのバッファー	CPU 使用量	スニファーのバッファー使用のモニター
ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -2 HOUR

ランタイム・パラメーター	演算子	デフォルト値

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -2 HOUR
期間終了	<=	NOW

タイプ別データベース/タイプ別データベース数

モニター対象の各データベース・タイプのサーバー・タイプとクライアント・ソース。

ドメイン	ベースとなる照会	メイン・エンティティー
アクセス	タイプ別データベース数	クライアント/サーバー

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

ディスカバーされたデータベース

このレポートは、レポート期間について、「データベース・タイプ」属性値が不明でない「ディスカバーされたポート」エンティティーごとに、プローブ・タイム・スタンプ、サーバー IP、サーバーのホスト名、データベース・タイプ、ポート、ポート・タイプ、およびその行の「ディスカバーされたポート」数をリストします。

ドメイン	ベースとなる照会	メイン・エンティティー
オートディスカバリー	ディスカバーされたデータベース	ディスカバーされたポート

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW
PortNotLike	NOT LIKE	デフォルト値なし。

DB ユーザー・マッピング・リスト

データベース・ユーザー (違反の原因となった SQL の起動者) とリアルタイム・アラート用 E メール・アドレス間のマッピング。

ドメイン	ベースとなる照会	メイン・エンティティー
オートディスカバリー	データベース・ユーザー・マッピング・リスト	Guardium ユーザー・ログイン

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

有効になっているデフォルト・データベース・ユーザー

このレポートでは、非資格情報スキャン API に提供されたデフォルト・ユーザーのグループとサーバー・リストに対するデータベース・スキャンの後に、有効であることが検出されたデフォルト・ユーザーの詳細を示します。 有効なユーザーがデータベース内で検出されたとき、このデータベース/ユーザーに関する検索結果は 1 回のみ報告されます。 以降のスキャンでは、データベースのタイム・スタンプおよびデータベース・バージョンが更新されます。 以降のスキャンで以前検出されていたユーザーが検出されなくなった場合、タイム・スタンプはそのまま残ります。これによって、そのユーザーがデータベース上で有効であると最後に検出された時の履歴が保持されます。 スキャンは分類リスナーの下で実行され、実行依頼されるジョブ (non_credential_scan API を使用) は、「Guardium ジョブ・キュー」レポートを使用してトラッキングできます。

ドメイン	ベースとなる照会	メイン・エンティティー
有効になっているデフォルト・データベース・ユーザー	有効になっているデフォルト・データベース・ユーザー	有効になっているデフォルト・データベース・ユーザー

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

データ・ソース

定義されているすべてのデータ・ソースをリストします。データ・ソース・タイプ、データ・ソース名、 データ・ソースの記述、ホスト、ポート、サービス名、ユーザー名、データベース名、 最後の接続、共有、接続プロパティー。

このレポートの出力は、「データ・ソース名」ランタイム・パラメーターで制限できます。 このパラメーターは、デフォルトではすべてのデータ・ソースを選択する「%」に設定されています。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	データ・ソース	使用不可

ランタイム・パラメーター	演算子	デフォルト値

ランタイム・パラメーター	演算子	デフォルト値
データ・ソース名	LIKE	%
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

ディスカバーされたインスタンス

この S-TAP レポートには、以下の情報の詳細が記載されています。

タイム・スタンプ、ホスト、プロトコル、ポート (最小)、ポート (最大)、KTAP データベース・ポート、 インスタンス名、クライアント、除外するクライアント、プロセス名、名前付きパイプ、 データベース・インスタンス・ディレクトリー、DB2® 共有メモリー調整、Db2 共有メモリー・クライアント位置、 Db2 共有メモリー・サイズ。

ドメイン	ベースとなる照会	メイン・エンティティー
例外	ディスカバーされたインスタンス	例外

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

データマート抽出ログ

データマートはデータウェアハウスのサブセットです。データウェアハウスでは、後から分析およびレポートで使用可能なように、データが汎用的な方式で集約および編成されます。データマートはユーザー定義のデータ分析を始めとして、内容、表示、使いやすさの面で、ユーザーの特定の要求に対応していることが特徴です。

データマート抽出プログラムは、指定されたスケジュールに応じてバッチで実行されます。 データは要求された間隔に応じて時間、日、週、月ごとに要約され、要約の結果は Guardium 分析データベース内の新しい表に保存されます。

ユーザーは、標準的なレポートと監査プロセスを使用してこのデータにアクセスできるようになります。データマート抽出データは、DM ドメインで使用可能です。エンティティー名は、データマート・データに対して指定された新しい表の名前に従って設定されます。ユーザーは、標準の照会 - レポート・ビルダーを使用して、デフォルトの照会をコピーし、照会を編集して、ダッシュボードに追加することができます。

抽出ログは、データマート名、コレクター IP、サーバー IP、開始時刻、終了時刻、ID、開始された実行、終了した実行、レコードの数、状況、エラー・コードから構成されます。

定義のエクスポート/インポート・ログ

このレポートでは、Guardium のエクスポート/インポート・アクティビティーがアクティビティー・タイプ別にリストされます。 レポートの各行には、アクティビティー・タイプ、開始時刻、ファイル名、状況、 コメント、およびその行のログ・レコードの数が含まれています。

ドメイン	ベースとなる照会	メイン・エンティティー
統合/アーカイブ	エクスポート/インポート定義ログ	統合/アーカイブ・ログ

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

ドロップされたリクエスト

検査エンジンによってドロップされたリクエストをトラッキングします (例外の記述 = ドロップされたデータベース要求)。極めてまれですが、大量の要求がある状況で、一部の要求が失われることがあります。 その場合、「ドロップされたリクエスト」レポートに、失われた要求があるセッションがリストされます。

ドメイン	ベースとなる照会	メイン・エンティティー
例外	ドロップされたリクエスト	例外
ランタイム・パラメーター	**演算子**	**デフォルト値**
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

例外数

レポート期間中にログに記録された例外の総数。

ドメイン	ベースとなる照会	メイン・エンティティー
例外	例外数	例外

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

エンタープライズ S-TAP (詳細) ビュー

このレポートについては、『S-TAP 情報 (中央マネージャー)』を参照してください。

エンタープライズ S-TAP 関連履歴

エンタープライズ S-TAP 関連履歴は、ロード・バランサー環境で特定の Guardium システムに S-TAP が報告を行った期間について報告します。

エンタープライズ S-TAP ビュー

このレポートについては、『S-TAP 情報 (中央マネージャー)』を参照してください。

Discovery への機密データのエクスポート

Guardium と InfoSphere® Discovery には、機密データを分類するためのメカニズムがあります。

識別された機密データを Guardium から InfoSphere Discovery へ、および InfoSphere Discovery から Guardium へと転送するための双方向インターフェースが提供されています。

このデータは CSV ファイルを介して転送されます。詳しくは、外部データ相関を参照してください。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	Discovery への機密データのエクスポート	分類プロセスの結果

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -3 HOURS
期間終了	<=	NOW
ルールの記述	LIKE	
スキーマ	LIKE	

エンタープライズ・バッファー使用状況モニター

このレポートには、すべての管理対象ユニットからのスニファーのバッファー使用の統合が表示されます。 アップロードのスケジュールを設定する必要があります。このレポートにリストされるフィールドについては、 「スニファーのバッファー使用」エンティティーの説明を参照してください。

ドメイン	ベースとなる照会	メイン・エンティティー
エンタープライズ・バッファー使用状況	エンタープライズ・バッファー使用状況	スニファーのバッファー使用

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

Guardium ジョブ・キュー

Guardium ジョブ・キューを表示します。以前は分類/アセスメントのジョブ・キューと呼ばれていました。ジョブごとに、 プロセス実行 ID、プロセス・タイプ、状況、Guardium ジョブのプロセス ID、レポート結果 ID、 Guardium ジョブの記述、監査タスクの記述、キュー時刻、開始時刻、終了時刻、 およびデータ・ソースがリストされます。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	Guardium ジョブ・キュー	使用不可

ランタイム・パラメーター	演算子	デフォルト値
ジョブの記述	LIKE	%
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

ジョブ・キュー

評価と分類は、それぞれジョブ・キューと呼ばれる個別のプロセスで実行されます。ジョブはキューに入れられ、実行を待機中のジョブを検索するためにリスナーがキューを定期的にポーリングしている間、 その状況を維持します。

停止

実行中のジョブを右クリックしてドリルダウン・メニューを表示すると、 実行中のジョブを停止し、取り消すオプションがあります。この時点ではジョブを再開することはできません。

一時停止

実行中のジョブは、ジョブ・キューが過負荷に陥る原因になるハングしたジョブの数を少なくするために、 モニターされています。30 分間非アクティブの状態が続いているジョブがあれば、リスナーの強制終了と再始動が行われ、 ジョブの操作が事実上停止されます。リスナーが再始動する前にはクリーナーと呼ばれるプロセスが実行され、 状況が RUNNING から HALTED に設定され、その後にリスナーが再始動されます。HALTED は、ジョブの実行が完了できなかったことを示します。

再実行依頼

ときには、リスナーがジョブのハング以外の理由 (例えばマシンのリブート) で再始動されることがあります。クリーナーは、実行中のジョブを一時停止する場合、そのジョブが過去 8 分以内に応答したかどうかを確認します。応答していれば、そのジョブはコピーされ、そのコピーがジョブ・キューに再実行依頼されます。一時停止した元のジョブは引き続きキューに表示され、 そのジョブが処理できた結果も入手可能です。

モニター

ジョブがアクティブ状況を維持するメカニズムは、ジョブ・キュー・レコードのタイム・スタンプにタッチすることによります。ジョブ・キュー・レコードはジョブ全体で使用されていることにご注意ください。個々の分類ルール、あるいは評価テストは、その親プロセスのタイム・スタンプと対話します。 モニターの対象になる個別のタイム・スタンプは持っていません。

分類では、すべてのルールがテストされる前、すべての SQL 操作の後にタイム・スタンプが更新されます。例えば、分類がページングをサポートするデータベース内のデータをスキャンする場合は、 データの各バッチがデータベースから戻された後にタイム・スタンプにタッチします。これは、ターゲット・データベースの状態によっては、分類が、複数の長時間実行する照会 (実行時間は 30 分に制限されます) を呼び出す可能性があるためです。

評価は、評価内の各テストが評価された後にタイム・スタンプにタッチします。ほとんどの評価テストは数秒以内に実行されます。

監視対象テスト

アセスメント・テストは比較的短時間で実行できますが、監視対象のアセスメント・テストは例外です。これらのテストは、Guardium アプライアンスで内部スニッフィング・データを使用する照会やレポートをベースにしていて、 比較的長時間実行できますが、処理中はタイム・スタンプを更新できません。 そのため、監視対象の評価テストでは、開始時にタイム・スタンプが 2 時間先に設定され、 実行が終了するまでに、基本的に 2 時間半が与えられます。このことにより、ユーザーがジョブ・キューを調べたときに、タイム・スタンプが未来の時刻に設定されているのを見て、混乱する場合があります。他の評価テストと同様、監視対象テストも終了時にタイム・スタンプにタッチします。次回のテストが監視対象のテストである場合、タイム・スタンプが再び 2 時間先の時刻に設定されます。次回のテストが監視対象のテストではない場合、タイム・スタンプは現在の時刻に設定されます。

GIM クライアント状況

GIM クライアントのリストを表示します。

ドメイン	ベースとなる照会	メイン・エンティティー
GIM クライアント状況	GIM クライアント状況	GIM クライアント

ランタイム・パラメーター	演算子	デフォルト値
クライアント名	%	N/A
クライアント OS	%	N/A

GIM イベント・リスト

GIM イベントのリストを表示します。

ドメイン	ベースとなる照会	メイン・エンティティー
GIM イベント	GIM イベント	GIM イベント

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

GIM インストール済みモジュール

インストール済み GIM モジュールのリストを表示します。
注: このレポートには、ホストに関連付けられているモジュールが表示されます。モジュールがホストに割り当て済みの場合、そのモジュールがスケジュールまたはインストールされていない場合でも、このレポートには割り当て済みのバージョンが表示されます。現在インストールされているモジュールを確認するには、GIM クライアント状況レポートを確認してください。

ドメイン	ベースとなる照会	メイン・エンティティー
GIM インストール済みベース	GIM インストール済みベース	GIM インストール済み

ランタイム・パラメーター	演算子	デフォルト値
なし	適用外	適用外

グループ使用状況レポート

定義済みグループと、各グループに依存するエンティティーをすべてリスト表示します。

Guardium API 例外

すべての GuardAPI 例外のタイム・スタンプと説明を表示します。これらは、 例外タイプ ID が GUARD_API_EXCEPTION のジョブです。

ドメイン	ベースとなる照会	メイン・エンティティー
例外	Guardium API 例外	例外

ランタイム・パラメーター	演算子	デフォルト値

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

Guardium アプリケーション

Guardium アプリケーションごとに、各行には、割り当てられたセキュリティー・ロール、 または all というワード (すべてのロールが割り当てられていることを示す) がリストされます。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	すべての Guardium アプリケーション	使用不可

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -100 Month DAY
期間終了	<=	NOW

Guardium グループの詳細

レポート期間について、このレポートの各行にはグループ・メンバーがリストされます。 列には以下の情報が入ります。グループの記述、グループ・タイプ、 グループ・サブタイプ、タイム・スタンプ (「グループ・メンバー」エンティティーから)、グループ・メンバー、 およびその行の「グループ・メンバー」エンティティーの数。タイム・スタンプの値は、 レコードの更新時に常に現在時刻に設定されます。

このレポートの出力は、ランタイム・パラメーターで制限できます。パラメーターはいずれも LIKE 演算子を指定して使用され、デフォルト値は % (すべての値を選択) です。

ドメイン	ベースとなる照会	メイン・エンティティー
グループ	Guardium グループの詳細	グループ・メンバー

ランタイム・パラメーター	演算子	デフォルト値
グループの記述	LIKE	%
グループ・タイプ	LIKE	%
期間開始	>=	NOW -100 MONTH
期間終了	<=	NOW

Guardium ユーザー

各ユーザー、最終アクティビティーの日付、割り当てられているロールの数をリストします。 ユーザーごとに、「レコード詳細」レポートまでドリルダウンすると、そのユーザーに割り当てられているロールを確認できます。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	ユーザー・ロール	使用不可

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -100 MONTH
期間終了	<=	NOW

ホスト履歴 (CAS)

このレポートは、CAS ホスト・イベントをリストします。このレポートのデフォルトのソート順は、標準のソート順とは異なります。ソート・キーは、優先順位の高いものから順に、ホスト名 (昇順)、インスタンス、イベント時間 (降順) です。

ドメイン	ベースとなる照会	メイン・エンティティー
CAS ホスト履歴	CAS ホスト履歴	ホスト・イベント

ランタイム・パラメーター	演算子	デフォルト値
Host_Name	Like	%
OS_Type	Like	%
Event_Type	Like	%

ILMT 準備状況

このレポートには、データ・サーバーにインストールされているアクティブ/非アクティブな S-TAP に関する詳細が示されます。ILMT エージェントがインストールされている場合、このレポートには、データ・サーバーのプロセッサー値が示されます。ILMT エージェントがインストールされていない場合、プロセッサー値はブランクです。このレポートは、S-TAP がインストールされていてアクティブになっているサーバーのプロセッサー値を示します。ILMT エージェントは、ILMT エージェントのインストール後にプロセッサー値を提供します。このレポートによって ILMT の要件が置き換わることはありません。(ILMT のコンプライアンスおよび監査の要件に従ってください。)

非アクティブな検査エンジン

非アクティブな検査エンジンすべてをリストします。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	非アクティブな検査エンジン	S-TAP 検査ヘッダー

ランタイム・パラメーター	演算子	デフォルト値
照会の開始日付	>=	NOW -3 HOUR
照会の終了日付	>=	NOW

非アクティブな S-TAP

システムで定義されている非アクティブな S-TAP® をすべてリストします。これには 1 つだけ、「期間開始」というランタイム・パラメーターがあり、デフォルトでは now -1 hour に設定されています。このパラメーターを使用して、非アクティブをどのように定義するのかを制御します。このレポートには、「S-TAP 状況」レポートと同じデータの列が含まれており、レポートの各行のカウントが追加されています。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	非アクティブな S-TAP	使用不可

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 HOUR

インストール済みのパッチ

インストール済みパッチのリストを表示します。ランタイム・パラメーターはありません。 このレポート・ドメインはシステム専用です。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	インストール済みのパッチ	使用不可

ランタイム・パラメーター	演算子	デフォルト値
なし	適用外	適用外

Guardium へのログイン

このレポートの値はすべて、「Guardium ログイン」エンティティーから取得されます。 レポート期間中、このレポートの各行には、ユーザー名、ログイン成功 (1 は成功、0 は失敗)、 ログインの日時、ログアウトの日時 (ユーザーがまだログアウトしていない場合はブランク)、 ホスト名、(ユーザーの) リモート・アドレス、およびその行のログイン数がリストされます。

ドメイン	ベースとなる照会	メイン・エンティティー
Guardium ログイン	Guardium ログイン	Guardium ユーザー・ログイン

ランタイム・パラメーター	演算子	デフォルト値
ホスト名	LIKE	%
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

ログに記録される R/T アラート

レポート期間中にログに記録されたリアルタイム・アラートの総数。ルールの記述ごとにリストされます。

ドメイン	ベースとなる照会	メイン・エンティティー
ポリシー違反	ログに記録される R/T アラート	ポリシー・ルール違反

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

ログに記録されるしきい値アラート

レポート期間中にログに記録されたしきい値アラートの総数。

ドメイン	ベースとなる照会	メイン・エンティティー
アラート	ログに記録されたアラート	しきい値アラート詳細

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

コレクター・ロギング (統合ユニットからの場合のみ有効)

「ロギング・コレクター」レポートは「日次モニター」タブの下に表示され、統合ユニットでのみ有効です。このレポートでは、サーバー IP ごと、コレクターごと、 1 日ごとに、セッション数が表示されます。 例: 5 月 19 日に、アグリゲーター #1 がサーバー 192.168.x.x1 で 100 セッション、 サーバー 192.168.x.x2 で 50 セッションを収集しました。アグリゲーター #2 がサーバー 192.168.x.x3 で 30 セッション、 サーバー 192.168.x.x4 で 90 セッションを収集しました、など。

ドメイン	ベースとなる照会	メイン・エンティティー
例外	ロギング・コレクター	ロギング・コレクター

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

管理対象ユニット (中央マネージャー)

準備完了している管理対象ユニットを示す、中央マネージャーに関するエンタープライズ・レポート。統計アラートでこのレポートを使用して、管理対象ユニットがダウンした場合に管理者に E メールを送信します。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	管理対象ユニット	管理対象ユニット

ランタイム・パラメーター	演算子	デフォルト値
ホスト名	LIKE	%
リモート・データ・ソース		ドロップダウン・メニュー
別名の表示		ラジオ・ボタン (オン、オフ、デフォルト)

アクティブな監査プロセスの数

アクティブな Guardium 監査プロセスの数。一元管理が実施されている場合、 このレポートは、中央マネージャー上でのみデータが入り、すべての管理対象ユニットでは空になります (「要求された照会のデータが見つかりません」という標準メッセージが表示されます)。 このレポートにはランタイム・パラメーターはありません。

ドメイン	ベースとなる照会	メイン・エンティティー
監査プロセス	アクティブ・プロセス数	監査プロセス

未処理監査プロセスのレビュー

未処理の Guardium 監査プロセスの数 (Guardium ユーザー別にリスト表示)。

表 1. 未処理監査プロセスのレビュー

ドメイン	ベースとなる照会	メイン・エンティティー
監査プロセス	未処理監査プロセスのレビュー	タスク結果 To-Do リスト

プライマリー Guardium ホスト変更ログ

S-TAP のプライマリー・ホスト変更のログ。プライマリー・ホストとは、 S-TAP がデータを送信する Guardium ユニットです。このレポートの各行には、 S-TAP ホスト、Guardium ホスト名、期間の開始、期間の終了がリストされます。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	プライマリー SGuard ホスト変更ログ	使用不可

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

照会エンティティーと属性

このレポートでは、Guardium レポート内のすべてのエンティティーと属性がリストされます。 このレポートは、Guardium 属性間のリンケージを単純化して GuardAPI 呼び出しにするために作成されました。

このレポートは、create_constant_attribute、create_api_parameter_mapping、delete_api_parameter_mapping、または list_param_mapping_for_function を呼び出す場合にも使用します。

ドメイン	ベースとなる照会	メイン・エンティティー
任意の Guardium レポート・ドメイン	レポート・ドメインの任意のエンティティー	エンティティー内の任意の属性

ランタイム・パラメーター	演算子	デフォルト値

ランタイム・パラメーター	演算子	デフォルト値
レポート名 LIKE <> '%' の場合は、新規パラメーターと一致するレポートで使用されるドメイン/エンティティーと属性だけが表示されます。 '%' の場合は、すべてのドメイン、照会、および属性が表示されます (どのレポートにも使用されないものも含めて)。	適用外	適用外

リプレイ統計

このレポートは、実行の開始日/実行の終了日のリプレイ統計、構成名、スケジュール・セットアップ名、ジョブ状況、統計記述、セッション ID、正常な照会、失敗した照会、合計照会、タイプ、アクティブ/待機中/完了済みの各タスクを示します。

ドメイン	ベースとなる照会	メイン・エンティティー
リプレイ結果のトラッキング	リプレイ統計	リプレイ結果統計

ランタイム・パラメーター	演算子	デフォルト値
照会の開始日付	>=	NOW -1 DAY
照会の終了日付	<=	NOW
セッション	>=	N/A
セッション	<=	N/A

リプレイ・サマリー

レポート期間内に、どの照会が失敗または成功したかに関する測定です。「リプレイ構成」で「失敗した照会」または「成功した照会」にチェック・マークを付けておくことが必要です。

ドメイン	ベースとなる照会	メイン・エンティティー
リプレイ結果	リプレイ・サマリー	リプレイ結果

ランタイム・パラメーター	演算子	デフォルト値
照会の開始日付	>=	NOW -1 DAY
照会の終了日付	<=	NOW
結果状況	%	N/A
スケジュール・セットアップ名	%	N/A

リストアされたデータ

このレポートには 2 つの列 (RESTORED_DAY と EXPIRATION_DATE) があります。 ユーザーがアーカイブからデータをリストアすると、リストアされたデータと、このデータを保持するために指定した期間に従って、この表にデータが追加されます。 パージ・プロセスはこの表を調べてパージできるデータを判別し、有効期限が切れたレコードをクリーンアップします。 RESTORED_DAY は、リストアされたデータの日付なので、過去の日付です。EXPIRATION_DATE は、このデータがパージされる日付であり、未来の日付です。

ドメイン	ベースとなる照会	メイン・エンティティー
リストアされたデータ	リストアされたデータ	リストアされたデータ

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -10 DAY
期間終了	<=	NOW +10 DAY

リクエスト・レート

デフォルトでは、最近 2 時間のリクエスト・レートを表示します。このグラフィカル・レポートは、 最近のアクティビティーのみを表示するためのものです。ランタイム・パラメーターを変更して対象となる時間フレームを大きくすると、データが大きすぎるというメッセージが表示される場合があります。 より大きな時間枠を表示する場合は、表形式のレポートを使用してください。

ドメイン	ベースとなる照会	メイン・エンティティー
スニファーのバッファー	リクエスト・レート	スニファーのバッファー使用のモニター

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -2 HOUR
期間終了	<=	NOW

スケジュールされたジョブの例外

スケジュールされたジョブ例外 (評価エラーなど) ごとに、そのタイム・スタンプと説明を表示します。. これらは、 例外タイプ ID が SCHED_JOB_EXCEPTION、ASSESSMENT_EXCEPTION、ASMT_ERROR のいずれかであるジョブです。

ドメイン	ベースとなる照会	メイン・エンティティー
スニファーのバッファー	CPU 使用量	スニファーのバッファー使用

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -2 HOUR
期間終了	<=	NOW

スケジュールされたジョブ

現在スケジュールされているジョブのリストを表示します。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	スケジュールされたジョブ	使用不可

セッション数

レポート期間中に開いた各種セッションの総数。

ドメイン	ベースとなる照会	メイン・エンティティー
アクセス	セッション数	セッション

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

SQL 数

レポート期間中に発行されたSQL コマンドの種類の総数。

ドメイン	ベースとなる照会	メイン・エンティティー
アクセス	SQL 数	SQL

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

S-TAP 構成変更履歴

このレポートは、検査エンジンが追加または変更されたときだけ表示されます。 ここには S-TAP 構成変更がリストされます。個々の検査エンジンの変更は、別々の行に表示されます。 各行には、S-TAP ホスト、データベース・サーバー・タイプ、データベース・ポート (始まり)、 データベース・ポート (終わり)、データベース・クライアント IP、データベース・クライアント・マスク、 および変更のタイム・スタンプがリストされます。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	構成変更履歴	使用不可

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

S-TAP 状況

各 S-TAP ホストで定義されている各検査エンジンについて、状況情報を 表示します。このレポートは現在の状況をレポートするため、開始日と終了日の パラメーターはありません。このレポートの各行は、S-TAP ホスト、データベース・サーバー・タイプ、状況、最後の応答、1 次ホスト名、および属性 (KTAP (インストール済み)、共有メモリー・ドライバー (インストール済み)、Db2 共有メモリー・ドライバー (インストール済み)、名前付きパイプ・ドライバー (インストール済み)、およびアプリケーション・サーバー・インストール済み) の Yes/No インディケーターをリストします。さらに、ハンター DBS をリストします。

注: Db2 共有メモリー・ドライバーは Db2 Tap フィーチャーに置き換えられました。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	S-TAP 状況	使用不可

S-TAP 検査

S-TAP 検査のすべての結果をリストします。

ドメイン	ベースとなる照会	メイン・エンティティー

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	S-TAP 検査	S-TAP 検査ヘッダー

ランタイム・パラメーター	演算子	デフォルト値
照会の開始日付	>=	NOW -3 HOUR
照会の終了日付	>=	NOW

S-TAP イベント

S-TAP に関する情報には、このレポートを使用します (内部データベースの SOFTWARE_TAP_EVENT 表から)。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	S-TAP イベント	使用不可

ランタイム・パラメーター	演算子	デフォルト値
イベント・タイプ	LIKE	%
ホスト・タイプ	LIKE	%
期間開始	>=	NOW -3 DAY
期間終了	<=	NOW

S-TAP 情報 (Central Manager)

レポート: 『S-TAP レポート』を参照。 Central Manager では、追加のレポート、「S-TAP 情報」が使用可能です。このレポートは、環境全体の S-TAP をモニターします。このデータのアップロードには、カスタム表ビルダーを使用します。

S-TAP 情報は、「S-TAP 情報」エンティティーが含まれる事前定義カスタム・ドメインであり、ライセンス・ドメインと違って変更できません。

カスタム照会を定義する際は、アップロード・ページに移動して「検査/修復」をクリックし、CUSTOM データベースにカスタム表を作成します。そうしないと、照会を保存するときに照会が検証されません。この表は、すべてのリモート・ソースから自動的にロードします。ユーザーは、使用するリモート・ソースを選択できません。すべてのリモート・ソースから取り込まれます。

このカスタム表とカスタム・ドメインに基づく、次の 2 つのレポートがあります。

エンタープライズ S-TAP ビューは、中央マネージャーから、コレクターまたは管理対象ユニット上のアクティブな S-TAP に関する情報を表示します (同じ S-TAP エンジンに対する重複があり、一方がアクティブで、他方が非アクティブの場合、アクティブな方のみがレポートに使用されます)。

詳細なエンタープライズ S-TAP ビューは、中央マネージャーから、すべてのコレクターまたは管理対象ユニット上のすべてのアクティブおよび非アクティブな S-TAP に関する情報を表示します。

エンタープライズ S-STAP ビューと詳細なエンタープライズ S-TAP ビューが同じに見える場合は、1 つの管理対象ユニット上にあるただ 1 つの S-TAP が表示されているためです。複数の S-TAP および複数の管理対象ユニットがある場合は、詳細なエンタープライズ S-TAP ビューの表示が違ったものになります。

これらの 2 つのレポートは、スタンドアロン・システムの「TAP モニター」タブから選択可能ですが、情報は表示されません。

アラート: 『監査プロセス定義の表示』で、アラート「検査エンジンと S-TAP」(検査エンジンと S-TAP の構成に関連するすべてのアクティビティーについてアラートを出す) を参照。

S-TAP 最後の応答

事前定義の照会およびレポートを使用可能ですが、 パネルには追加されません。

照会/レポートには、すべての S-TAP ホストと、各 ホストから送信された最後の応答 (ハートビート) が表示されます。

この照会の目的は、 ホスト上の S-TAP が特定の期間応答しなかった場合にトリガーするアラートを定義できるようにすることです。

入力パラメーターは、「最後の応答の開始時刻」および 「最後の応答の終了時刻」です。

例えば、「最後の応答の開始時刻」に NOW -5 DAYS と指定し、「最後の応答の終了時刻」に NOW - 3 HOURS と指定して実行した場合、この 5 日間に最後の応答を送信したホストのうち、過去 3 時間以内に応答していないホストに関して、ホスト名および最後の応答時刻が表示されます。

S-TAP 状況モニター

このレポートは、この Guardium アプライアンスへの各 S-TAP レポートについて、 S-TAP ホスト、S-TAP バージョン、データベース・サーバー・タイプ、 状況 (アクティブまたは非アクティブ)、最後に受信した応答 (日時)、1 次ホスト名、 および (KTAP、MS SQL サーバー共有メモリー、Db2 共有メモリー、 ローカル TCP モニター、名前付きパイプの使用、および暗号化の) true/false インジケーターを識別します。

このレポートはランタイム・パラメーターを持たず、変更不能なシステム専用の照会をベースにしています。

STAP/Z ファイル

STAP/Z は、Db2 (z/OS® 上の) から収集した、Db2 イベント、SQL ステートメントなどを含む生データのファイルを提供します。 このレポートでは、インターフェース ID、UA ファイル名 (非正規化された監査イベント)、 UT ファイル名 (非正規化された監査イベントのテキスト)、UH ファイル名 (非正規化された監査イベントのホスト変数)、 ファイル状況、処理されたイベントの総数、失敗したイベントの数、タイム・スタンプがリストされます。 ランタイム・パラメーターは、FileName Like % と FileStatus Like % です。

このレポートでは、2 つのランタイム・パラメーター (FileName Like % と FileStatus Like %) が使用されます。 これは、変更不能なシステム専用の照会をベースにしています。

TCP 例外

レポート期間中、「例外タイプ」エンティティーの「例外の記述」が TCP/IP プロトコルの例外である例外ごとに、このレポートでは、「例外」エンティティーから以下の属性値が 1 行にリストされます。

ドメイン	ベースとなる照会	メイン・エンティティー
例外	TCP 例外	例外

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

テンプレート (CAS)

このレポートは、CAS テンプレートをリストします。デフォルトでは、すべてのテンプレート項目がリストされます。

ドメイン	ベースとなる照会	メイン・エンティティー
CAS テンプレート	CAS テンプレート	テンプレート

ランタイム・パラメーター	演算子	デフォルト値
Access_Name	Like	%
Template_Set_Name	Like	%
Audit_Type	Like	%

テスト例外

一時的に免除されるテストとデータ・ソースのペアを示します。テスト例外の使用について詳しくは、『create_test_exception』を参照してください。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	テスト例外	使用不可

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -12 MONTH
期間終了	<=	NOW

スループット

レポート期間中の「アクセス期間」ごとに、各行に、期間の開始時刻、サーバー IP アドレス数、 アクセスの総数がリストされます (「アクセス期間」エンティティー)。

このレポートの出力は、「サーバー IP」ランタイム・パラメーターで制限できます。このパラメーターは、デフォルトではすべての IP アドレスを選択する % に設定されています。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	データベース・サーバー・スループット	使用不可

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW
サーバー IP	LIKE	%

スループット (グラフィカル)

このレポートは、表形式スループット・レポートの図表バージョン「分散ラベル付き線グラフ」です。 期間の開始時刻ごとに 1 つのデータ・ポイントでレポート期間中のアクセス総数が作図されます。

このレポートの出力は、「サーバー IP」ランタイム・パラメーターで制限できます。このパラメーターは、デフォルトではすべての IP アドレスを選択する % に設定されています。

ドメイン	ベースとなる照会	メイン・エンティティー
アクセス	データベース・サーバー・スループット - グラフ	アクセス期間

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW
サーバー IP	LIKE	%

「ユーザー・アクティビティー監査証跡」レポート

「ユーザー・アクティビティー監査証跡」メニュー選択には 2 つのレポートが表示されます。 さらに、それぞれのレポートから第 3 のレポートが生成されることがあります。 以下を参照してください。

- ユーザー・アクティビティー監査証跡
- システム/セキュリティー・アクティビティー
- Guardium ユーザー・アクティビティーの詳細 (ドリルダウン)

ユーザー・アクティビティー監査証跡

レポート期間中、「Guardium ユーザー・アクティビティー監査」エンティティーに表示される個々のユーザー名で、 各行には Guardium ユーザー名、アクティビティー・タイプの記述 (「Guardium アクティビティー・タイプ」エンティティーから)、変更されたエンティティーの数の値、ホスト名、およびその行の「Guardium アクティビティー監査」エンティティーの総数が表示されます。

このレポートの任意の行から、「Guardium ユーザー・アクティビティーの詳細」レポートをドリルダウン・レポートとして選択できます。

ドメイン	ベースとなる照会	メイン・エンティティー
Guardium アクティビティー	ユーザー・アクティビティー監査証跡	Guardium ユーザー・アクティビティー監査

ランタイム・パラメーター	演算子	デフォルト値
ホスト名	LIKE	%
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

システム/セキュリティー・アクティビティー

レポート期間中、「Guardium ユーザー・アクティビティー監査」エンティティーに表示される個々のユーザー名で、 各行には Guardium ユーザー名、アクティビティー・タイプの記述 (「Guardium アクティビティー・タイプ」エンティティーから)、変更されたエンティティーの数の値、ホスト名、およびその行の「Guardium アクティビティー監査」エンティティーの総数が表示されます。

このレポートの任意の行から、「Guardium ユーザー・アクティビティーの詳細」レポートをドリルダウン・レポートとして選択できます。

ドメイン	ベースとなる照会	メイン・エンティティー
Guardium アクティビティー	ユーザー・アクティビティー監査証跡	Guardium ユーザー・アクティビティー監査

ランタイム・パラメーター	演算子	デフォルト値
ホスト名	LIKE	%
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

Guardium ユーザー・アクティビティーの詳細 (ドリルダウン)

このレポートはメニューからは選択できませんが、「ユーザー・アクティビティー監査証跡」レポートの任意の行、または「システム/セキュリティー・アクティビティー」レポートから開くことができます。このレポートの選択した行 (「ユーザー名」と「アクティビティー・タイプの記述」をベースとしたもの) では、 このレポートで、ユーザー名、タイム・スタンプ、エンティティーの変更、オブジェクトの記述、 すべての値、およびその行の「Guardium ユーザー・アクティビティー監査」エンティティーの数といった属性値がリストされます。 属性値はすべて「Guardium ユーザー・アクティビティー監査」エンティティーから取得されますが、 「アクティビティー・タイプの記述」だけは例外で、「Guardium アクティビティー・タイプ」エンティティーから取得されます。

ドメイン	ベースとなる照会	メイン・エンティティー
Guardium アクティビティー	Guardium ユーザー・アクティビティーの詳細	Guardium ユーザー・アクティビティー監査

ランタイム・パラメーター	演算子	デフォルト値
アクティビティー・タイプの記述		呼び出しレポートからの値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW
ユーザー名		呼び出しレポートからの値

警告: ユーザーは、root ユーザーおよびその他の機密性の高いシステム・アカウントのアクティビティーがログに記録されていることを認識しておく必要があります。これらのユーザーのアクティビティーまでドリルダウンすると、コマンド行で入力された機密のコマンドとパスワードが表示されることがあります。 したがって、ユーザーは、可能な限り、このドリルダウン・レポートに表示させたくない機密のコマンド行情報を入力しないようにしてください。

ユーザー To-do リスト

個々の Guardium 監査プロセスごとに、記述、ログイン名、必要なアクション (レビューまたは承認)、状況、 署名またはレビューしたユーザー、および指定したタスクの実行日を表示します。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	ユーザー To-do リスト	使用不可

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY

ランタイム・パラメーター	演算子	デフォルト値
期間終了	<=	NOW

ユーザー・コメント - 共有可能

共有可能なユーザー・コメントは、検査エンジン、インストール済みポリシー、 監査プロセスの結果の各コメント以外のすべてのコメントです。 共有可能なユーザー・コメントごとに、このレポートでは、作成日、参照されたオブジェクトのタイプ (例: アラート)、オブジェクトの記述、コメントを作成したユーザー、およびコメントの内容がリストされます。

注: 検査エンジン、インストール済みポリシー、または監査プロセスの結果に定義されたコメントは、個々の定義から表示できますが、レポート上には表示できません。

ドメイン	ベースとなる照会	メイン・エンティティー
コメント	定義されたコメント	コメント

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -2 MONTH
期間終了	<=	NOW

ユニット使用状況レベル

以下のデフォルト・レポートには、ユニット使用状況データが表示されます。

- ユニット使用状況: 特定の時間フレームにおける各ユニットのユニット使用状況の最大レベルが表示されます。レポートの時間フレーム内のすべての期間についてのユニットの詳細を表示するドリルダウンがあります。
- ユニット使用状況の分布: このレポートは、ユニットごとに、レポートの時間フレーム内の期間のパーセントを使用状況レベルの低、中、高で示します。
- 使用状況のしきい値: この事前定義レポートは、すべてのユニット使用状況パラメーターの下限しきい値と上限しきい値をすべて表示します。

- ユニット使用状況の日次サマリー - ユニット使用状況データの日次サマリーが表示されます。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	ユニット使用状況の分布	ユニット使用状況レベル

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -24 HOUR
期間終了	<=	NOW

変更された値

レポート期間中、このレポートには、モニター値の変更に関する詳細情報が記録されます。 表示される属性値はすべて、「値のモニター」エンティティーから取得されます。 このレポートのベースとなる照会のソート・シーケンスは、次のように標準のソート・シーケンスとは異なっています。

- サーバー IP
- データベース・タイプ
- 監査タイム・スタンプ
- 監査表の名前
- 監査の所有者

このレポートのベースとなる照会には多数のランタイム・パラメーターがあります。 そのすべてが LIKE 演算子を使用し、デフォルト値は % (すべての値が選択されるという意味) です。

選択されたモニター値ごとに、このレポートでは、タイム・スタンプ、サーバー IP、 データベース・タイプ、サービス名、データベース名、監査ログイン名、監査タイム・スタンプ、 監査表の名前、監査の所有者、監査アクション、古い値の監査、新しい値の監査、 SQL テキスト、トリガーされる ID、およびその行の「列の変更」エンティティーの数が、1 行にリストされます。

ドメイン	ベースとなる照会	メイン・エンティティー
変更された値	変更された値	変更された列

ランタイム・パラメーター	演算子	デフォルト値
監査アクション	LIKE	%
監査ログイン名	LIKE	%
監査の所有者	LIKE	%
監査表の名前	LIKE	%
データベース・タイプ	LIKE	%
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW
サーバー IP	LIKE	%

親トピック: 定義済みレポート

事前定義ユーザー・レポート

このセクションでは、デフォルトのユーザー・アクセス権限を持つユーザー向けのすべての事前定義レポートについて簡単に説明します。

注: 監視データ・レベルでのデータ・レベル・セキュリティーが有効な場合 (「グローバル・プロファイル」設定を参照)、ユーザーに対してユーザーのデータベース情報のみが表示されるよう、監査プロセスの出力がフィルタリングされます。

インストール済みポリシーの表示

「現在インストールされているポリシー」レポートには、インストールされているポリシーの情報が 表示されます。インストール済みポリシーのリンクをクリックすると、別のウィンドウにポリシー・ルールが 表示されます。

タイプ別データベース数

モニターされる各データベース・タイプのサーバーおよびクライアントの数を 表示します (デフォルトの期間は当日です)。

リクエスト・レート

デフォルトでは、最近 2 時間のリクエスト・レートを表示します。このグラフィカル・レポートは、 最近のアクティビティーのみを表示するためのものです。「開始」および「終了」のランタイム・パラメーターを変更して対象となる時間フレームを長くすると、データが大きすぎるというメッセージが表示される場合があります。(より大きな時間枠を表示する場合は、表形式のレポートを使用してください。)

サーバー・タイプ別セッション

サーバー・タイプ (DB2®、Informix® など) ごとに、レポートの 1 行を使用して、 レポート期間 (デフォルトでは過去の 3 時間) にオープンされたセッションの総数が 表示されます。

機密オブジェクトに対する DML 実行

このレポートの 1 行には、「機密オブジェクト」グループのオブジェクト名を参照する「DML コマンド」グループ からの各 SQL 動詞について、アクセス期間、クライアント IP、 ソース・プログラム、およびその行で参照されるオブジェクトの総数が 表示されます。レポート・タイトルには Executions という言葉が含まれますが、 レポートされるすべてのコマンドが実際に実行されたという保証はありません。

機密オブジェクトの使用

「機密オブジェクト」グループの各オブジェクトについて、レポート期間にそのオブジェクトを参照した 各クライアント IP とソース・プログラム、およびオブジェクト参照数が 1 行に表示されます。

「機密オブジェクト」グループはインストール時には空です。企業内の誰かが、該当するメンバーのセットでグループにデータを設定する必要があります。

クライアント IP 別アクティビティー

レポート期間中に参照される各クライアント IP アドレスについて、 SQL 動詞の数、オブジェクト名、およびセッションの総数が 1 行に表示されます。

データベース・サーバー

レポートの 1 行に、レポート期間中にアクセスされる各サーバー IP アドレスについて、 サーバー・タイプ、データベース名、サービス名、 そのサーバーにアクセスするソース・プログラム数、およびその行のセッション総数が 表示されます。

IMS アクセス (z/OS)

IMS へのアクセスをレポートするには、これを使用します (z/OS®)。

ドメイン	ベースとなる照会	メイン・エンティティー
アクセス	IMS アクセス	クライアント・サーバー

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -2 HOUR
期間終了	<=	NOW

IMS オブジェクト (z/OS)

IMS に対するオブジェクトをレポートするには、これを使用します (z/OS)。

ドメイン	ベースとなる照会	メイン・エンティティー
アクセス	IMS オブジェクト	オブジェクト

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -2 HOUR
期間終了	<=	NOW

IMS イベント (z/OS)

IMS に対するイベントをレポートするには、これを使用します (z/OS)。

ドメイン	ベースとなる照会	メイン・エンティティー
アクセス	IMS イベント	SQL

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -2 HOUR
期間終了	<=	NOW

IMS データ・アクセス詳細 (z/OS)

IMS に対するデータ・アクセス詳細をレポートするには、これを使用します (z/OS)。

ドメイン	ベースとなる照会	メイン・エンティティー
アクセス	IMS データ・アクセス詳細	完全な SQL

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -2 HOUR
期間終了	<=	NOW
クライアント IP	LIKE	
DBUserName	LIKE	
IMS 名	LIKE	
ServerIP	LIKE	

ポリシー違反

このレポートは、レポート期間中にログに記録されるすべてのポリシー・ルール違反について、 「ポリシー・ルール違反」エンティティーからのタイム・スタンプ、 アクセス・ルールの記述、クライアント IP、サーバー IP、データベース・ユーザー名、 「ポリシー・ルール違反」エンティティーからの SQL 文字列全体、重大度の記述、およびその行の違反数を提供します。このレポートのベースとなる照会 (重大度付きポリシー違反リスト) にはアクセスできませんが、このレポートのコピーを作成することができます。

例外分布

円グラフの各扇形は、レポート期間中にログに記録された (「例外タイプ」エンティティーからの) 各「例外分布」属性値の例外の比率を 示します。

他のグラフと同様に、円グラフをドリルダウンして、そのグラフのベースとなる 表形式の照会を表示できます。 ここで利用可能な この表形式レポート (または、この表形式レポートのドリルダウン) から アクセス可能な例外レポートがいくつかありますが、これらはメニューには 含まれていません。

例外モニター

レポート期間中に記録される例外の数。ポータルでレポートをリフレッシュするたびに、 1 つのデータ・ポイントが作成されます。

失敗したユーザー・ログイン試行

レポート期間中のログイン試行の各失敗について、 ユーザー名、ソース・アドレス、宛先アドレス、およびユーザーがログインを試行するサーバーの データベース・プロトコル・タイプをリストします。

SQL エラー

レポート期間中の各 SQL エラーについて、クライアント IP アドレス、 サーバー IP アドレス、サーバー・タイプ、データベース・ユーザー名、データベース・エラー・テキスト、 およびそのレコードの合計エラー発生数を表示します。

例外数

レポート期間中にログに記録される例外の総数 (「例外」エンティティー)。

無効なユーザーのログイン

「無効なデータベース・ユーザー」グループのメンバーであるデータベース・ユーザーによる すべてのログインをリストします。各行に、データベース・ユーザー名、クライアント IP、サーバー IP、 サーバー・タイプ、ソース・プログラム、最後のログイン時 (「セッション開始」属性の最大値)、および その行のセッション数がリストされます。

「無効なデータベース・ユーザー」グループはインストール時は空です。 誰かがデータを設定する必要があります。このレポートのベースとなる照会 (無効なユーザーのログイン) には、照会 - レポート・ビルダーからはアクセスできません。

アクティブ・ユーザーの最終ログイン

「アクティブ・ユーザー」グループの各メンバーについて、 レポート期間中に記録された最後のログイン。レポート期間中にログインがない場合でも、 このグループのすべてのメンバーがリストされます。この点は、グループのメンバーに基づく 他のほとんどのレポートとは異なります。「通常」は、メンバーに対して アクティビティーが見つからない場合は、そのメンバーはリストされません。

各行に、データベース・ユーザー名、クライアント IP、サーバー IP、 サーバー・タイプ、ソース・プログラム、最後のログイン時 (「セッション開始」属性の最大値)、および その行のセッション数がリストされます。

「アクティブ・ユーザー」グループはインストール時は空です。 誰かがデータを設定する必要があります。このレポートのベースとなる照会 (アクティブ・ユーザーの最終ログイン) には、照会 - レポート・ビルダーからはアクセスできません。

アクティビティーのないアクティブ・ユーザー

レポート期間中にアクティビティーを持たない、「アクティブ・ユーザー」グループのメンバーの リスト。レポート期間中にすべてのユーザーがアクティビティーを持つ場合、このレポートは 空になります。

「アクティブ・ユーザー」グループは事前定義されていますが、インストール時は空です。 誰かがデータを設定する必要があります。このレポートのベースとなる照会 (アクティビティーのないアクティブ・ユーザー) には、照会 - レポート・ビルダーからはアクセスできません。

無効なユーザーによるログイン試行の失敗

「無効なデータベース・ユーザー」グループのメンバーであるデータベース・ユーザーによる ログイン試行の失敗をリストします。レポート期間中にこのグループの誰かによる ログイン試行の失敗がない場合、このレポートは空になります。

「無効なデータベース・ユーザー」グループは事前定義されていますが、インストール時は 空です。 誰かがデータを設定する必要があります。このレポートの標準装備の照会には アクセスできません。このレポートのベースとなる照会 (無効なユーザーによるログイン試行の失敗) は、クエリー・ビルダーからは アクセスできません。

期間あたりの超過エラー

エラー数/期間を表示します。例えば、同一のクライアント IP アドレス、サーバー IP アドレス、サーバー・タイプ、データベース・ユーザー名で 60 分間のエラー数が N 個より多いなど。

指定日以降に非アクティブなユーザー

アクセス・レコードがあり、最大セッション開始時刻が 90 日より前の全ユーザーについて、ユーザーおよび最後のセッションの開始時刻を示します。(非アクティブ・ユーザーは、一度もログインしたことがない場合や、古いログインがすべてパージされた場合にはなくなります。)

admin ユーザーのログイン

レポート期間中に 1 つ以上のセッションを持っていた 「admin ユーザー」グループに含まれる各データベース・ユーザー名について、各行に クライアント IP、 データベース・ユーザー名、ソース・プログラム、セッション開始の時刻、 およびその行のセッション数がリストされます。

データベース定義済みユーザー・ログイン

レポート期間中に 1 つ以上のセッションを持っていた 「データベース定義済みユーザー」グループに含まれる各データベース・ユーザー名について、 各行に、データベース・ユーザー名、クライアント IP、 サーバー IP、ソース・プログラム、データベース名、 サービス名、およびその行のセッション数がリストされます。

管理コマンドの使用状況

このレポートは、レポート期間中に表示された「管理コマンド」グループに含まれる 各 SQL 動詞について、SQL 動詞、深さ、オブジェクト名、クライアント IP、 および参照されるオブジェクト数をリストします。

管理オブジェクトの使用状況

レポート期間中に表示された「管理オブジェクト」グループに含まれる 各オブジェクト名について、各行にオブジェクト名、クライアント IP、 サーバー IP、サービス名、データベース名、ソース・プログラム、データベース・ユーザー名、 およびその行のオブジェクト数がリストされます。

管理オブジェクトに対する DML 実行

「管理オブジェクト」グループのオブジェクト名を参照する「DML コマンド」グループからの 各 SQL 動詞について、このレポートは 1 行に、データベース・ユーザー名、クライアント IP、サーバー IP、 サーバー・タイプ、サービス名、データベース名、SQL 動詞、オブジェクト名および その行で参照されるオブジェクト数を表示します。

BACKUP コマンド実行

このレポートは、レポート期間中に参照される「BACKUP コマンド」グループからの各 SQL 動詞について、 クライアント IP、サーバー IP、サービス名、データベース・ユーザー名、 ソース・プログラム、データベース名、オブジェクト名、SQL 動詞、およびその行で参照される オブジェクト数を表示します。

RESTORE コマンド実行

このレポートは、レポート期間中に参照される「BACKUP コマンド」グループからの各 SQL 動詞について、 クライアント IP、サーバー IP、サービス名、データベース・ユーザー名、 ソース・プログラム、データベース名、オブジェクト名、SQL 動詞、およびその行で参照される オブジェクト数を表示します。

REVOKE コマンド実行

このレポートは、レポート期間中に参照される「REVOKE コマンド」グループからの各 SQL 動詞について、 クライアント IP、サーバー IP、サービス名、データベース・ユーザー名、 ソース・プログラム、データベース名、オブジェクト名、SQL 動詞、およびこの行で 参照されるオブジェクト数を表示します。

KILL コマンド実行

このレポートは、レポート期間中に参照される「KILL コマンド」グループからの各 SQL 動詞について、 クライアント IP、サーバー IP、サービス名、データベース・ユーザー名、 ソース・プログラム、データベース名、オブジェクト名、SQL 動詞、およびこの行で 参照されるオブジェクト数を表示します。

DBCC コマンド実行

このレポートは、レポート期間中に参照される「DBCC コマンド」グループからの各 SQL 動詞について、 クライアント IP、サーバー IP、サービス名、データベース・ユーザー名、 ソース・プログラム、データベース名、SQL ステートメント、およびその行で 参照されるオブジェクト数を表示します。

GRANT コマンド実行

このレポートは、レポート期間中に参照される「GRANT コマンド」グループからの各 SQL 動詞について、 クライアント IP、サーバー IP、サービス名、データベース・ユーザー名、 ソース・プログラム、データベース名、オブジェクト名、SQL 動詞、およびその行で 参照されるオブジェクト数を表示します。

特権アカウント使用状況

admin ユーザー・グループのユーザーによる VERB の実行に関して、ユーザー、VERB、および期間のカウントを示します。

ビジネス・オブジェクトの特権ユーザー・アクセス

ビジネス・オブジェクトの選択されたグループに存在するオブジェクトによる VERB の実行および admin ユーザーに関して、ユーザー、VERB、およびオブジェクトを示します。

CREATE コマンド実行

このレポートは、レポート期間中に参照される「CREATE コマンド」グループからの各 SQL 動詞について、 クライアント IP、サーバー IP、サービス名、データベース・ユーザー名、 ソース・プログラム、データベース名、オブジェクト名、SQL 動詞、およびその行で 参照されるオブジェクト数を表示します。

DDL コマンド

データベースに送信されるすべての DDL コマンド。このレポートは、 DDL の要求元となるクライアント IP、メイン SQL 動詞 (特定の DDL コマンド)、およびそのレコード用にアクセスされる合計オブジェクトを表示します。

このレポートは、レポート期間中に参照される「DDL コマンド」グループからの各 SQL 動詞について、 クライアント IP、サーバー IP、サーバー・タイプ、SQL 動詞、およびその行で参照されるコマンド数を表示します。

ALTER コマンド実行

発行されるすべての ALTER コマンド。このレポートは、特定の行にリストされている クライアント IP/DDL コマンドの組み合わせごとに、 DDL の要求元となるクライアント IP、サーバー IP アドレス、サービス名、データベース・ユーザー名、 ソース・プログラム、データベース名、オブジェクト名、およびのメイン SQL 動詞 (特定の DDL コマンド) を 表示します。

このレポートは、レポート期間中に参照される「ALTER コマンド」グループからの各 SQL 動詞について、 クライアント IP、サーバー IP、サービス名、データベース・ユーザー名、 ソース・プログラム、データベース名、オブジェクト名、SQL 動詞、およびその行で 参照されるオブジェクト数を表示します。

DDL 分布

この棒グラフは、レポート期間中に「DDL コマンド」グループから参照される コマンドの分布を示します。参照されるコマンドごとに、単一のバーが、影響を受けるオブジェクトの総数を示します。

DROP コマンド実行

このレポートは、レポート期間中に参照される「DROP コマンド」グループからの各 SQL 動詞について、 クライアント IP、サーバー IP、サービス名、データベース・ユーザー名、 ソース・プログラム、データベース名、オブジェクト名、SQL 動詞、およびその行で 参照されるオブジェクト数を表示します。

1 ユーザー 1 IP

このレポートの各行には、レポート期間中にセッション・データが収集されたデータベース・ユーザー名ごとに、 ユーザーのログイン元となるクライアント IP アドレスの数、 およびセッションの総数が表示されます。

クライアント IP アクティビティーのサマリー

このレポートは、ランタイム・パラメーターとして指定されている、単一のクライアント IP アドレスからの レポート期間アクティビティーを表示します。レポートの各行には、 クライアント IP、ソース・プログラム、SQL 動詞、(SQL コマンド内のセンテンスの) 深さ、 オブジェクト名、およびその行のためにオブジェクトが参照された回数が表示されます。

セッション・リスト

このレポートは、レポート期間のすべてのデータベース・セッションをリストします。 各セッションについて、レポートはセッション (エンティティー) タイム・スタンプ、 セッション開始 (タイム・スタンプ)、サーバー・タイプ、クライアント IP、サーバー IP、クライアント・ポート、 サーバー・ポート、ネットワーク・プロトコル、データベース・プロトコル、データベース・プロトコル・バージョン、 データベース・ユーザー名、ソース・プログラム、およびその行のセッション数 (常に 1 でなければ なりません) を表示します。

ほとんどのレポートと同様に、ドリルダウン・レポートが利用できます。 このレポートから アクセス可能なセッション・レポートは多数ありますが、これらはメニューには 含まれていません。以下のレポートと、 選択したレポートの行からの値を使用して設定されたレポートのランタイム・パラメーター が含まれます。

レポート	ランタイム・パラメーター
クライアント IP 別セッション	サーバー IP、サーバー・タイプ
サーバー IP 別セッション	サーバー・タイプ
ソース・プログラム別セッション	サーバー・タイプ、 サーバー IP
ユーザー別セッション	サーバー・タイプ、 サーバー IP
サーバー別セッション詳細	サーバー・タイプ、 サーバー IP

コマンド・リスト

このレポートは、レポート期間中に参照されるすべての SQL 動詞をリストします。最外部レベルでは、 コマンドは正時に「アクセス期間」エンティティーからの「期間の開始」時刻によって グループ化されます (通常は 1 時間)。Guardium 管理者は、 ログ細分度を変更することによって、アクセス期間の長さを変更することが できます (デフォルトでは 1 時間)。レポート期間の各アクセス期間について、 各行にアクセス期間の開始時刻、 SQL 動詞、SQL ステートメントの動詞の深さ、親 (所有動詞へのポインター)、 およびその行のオカレンス数がリストされます。

オブジェクト・リスト

このレポートは、レポート期間中に参照されるすべてのオブジェクトをリストします。最外部レベルでは、 オブジェクトは正時に「アクセス期間」エンティティーからの「期間の開始」時刻によって グループ化されます (通常は 1 時間)。SQL Guard 管理者は、 ログ細分度を変更することによって、アクセス期間の長さを変更することが できます (デフォルトでは 1 時間)。レポート期間中の各アクセス期間について、 各行にアクセス期間の開始時刻、オブジェクト名、および その行のオカレンス数がリストされます。

オブジェクト・アクティビティーのサマリー

このレポートは、ランタイム・パラメーターとして指定されている単一オブジェクト名についての、レポート期間アクティビティーを表示します。レポートの各行には、クライアント IP、ソース・プログラム、SQL 動詞、(SQL コマンド内のセンテンスの) 深さ、 オブジェクト名、およびその行のためにオブジェクトが参照された回数が表示されます。

アーカイブ候補

このレポートは、長期間アクセスされていないオブジェクト (データベース表またはストアード・プロシージャー など) をリストします。このレポートのベースとなる照会にアクセスすることは できません。

時間帯別アクセス詳細

このレポートは、レポート期間 (このレポートではデフォルトは 1 時間) に参照される 各データベース・ユーザー名について、非常に詳細なリストを作成します。 レポートの各行は、データベース・ユーザー名、クライアント IP、サーバー IP、 期間の開始、ソース・プログラム、SQL (SQL エンティティーから)、およびアクセス期間中のオカレンス数をリストします。

データベース・ユーザー名別の完全な SQL

このレポートは、ランタイム・パラメーターに指定されている単一データベース・ユーザー名について、ログに記録されているレポート期間の「完全な SQL」属性値を表示します。レポートの各行には、完全な SQL ID、 (「完全な SQL」エンティティーの) タイム・スタンプ、クライアント IP、データベース・ユーザー名、セッション開始、 ソース・プログラム、完全な SQL、およびその行のオカレンス数が表示されます。

クライアント IP 別の完全な SQL

このレポートは、ランタイム・パラメーターに指定されている単一クライアント IP について、ログに記録されているレポート期間の「完全な SQL」属性値を表示します。レポートの各行には、完全な SQL ID、 (「完全な SQL」エンティティーの) タイム・スタンプ、クライアント IP、データベース・ユーザー名、セッション開始、 ソース・プログラム、完全な SQL、およびその行のオカレンス数が表示されます。

未解析ログ・リスト

未解析ログ処理のタスクをリストします。

分類プロセスの結果

分類プロセスのタスクをリストします。

DW 休止オブジェクト

休止表に焦点を絞って、第 2 グループのメンバーではない、 1 つのグループの全メンバーを表示します。例えば、このレポートは、すべてのオブジェクトグループに含まれているが、選択で使用されていないオブジェクトを表示します。

DW 休止オブジェクト/フィールド

休止表および列に焦点を絞って、第 2 グループのメンバーではない、 1 つのグループの全メンバーを表示します。このインスタンスでは、 グループは 2 タプルのタイプ (値属性のペアの複合であるメンバー) です。 例えば、このレポートは、すべてのオブジェクトおよびフィールドグループに含まれているが、選択で使用されていないオブジェクトを表示します。

DW EXECUTE オブジェクト・アクセス

このレポートを使用して、実行中のストアード・プロシージャー名のセットで「DW EXECUTE オブジェクト」というグループを取り込みます。次に、「グループ・ビルダー」/「自動生成呼び出しプロシージャー」で間接マッピングを使用して、これらのプロシージャー内で使用されるオブジェクトをすべて生成します。

DW SELECT オブジェクト・アクセス

このレポートは、SELECT ステートメントを介してアクセスされる すべてのオブジェクト名を表示します。

DW SELECT オブジェクト/フィールド・アクセス

このレポートは、SELECT ステートメントを介してアクセスされる すべてのオブジェクト名およびフィールド名を表示します。

長時間実行されている照会

このレポートは、レポート期間について、最も長く実行されている照会と、最も長い平均実行時間をリストします。照会ごとに、クライアント IP、サーバー IP、SQL、(「アクセス期間」エンティティーからの) 期間の開始、平均実行時間、およびその行のオカレンス数がリストされます。このレポートのベースとなる照会にアクセスすることは できません。

スループット

このレポートは、レポート期間中に参照されるすべてのサーバー IP の数 および合計アクセス数を示します。最外部レベルでは、アクセスは正時に「アクセス期間」エンティティーからの「期間の開始」時刻によって グループ化されます (通常は 1 時間)。Guardium® 管理者は、ログ細分度を変更することによって、アクセス期間の長さを変更することが できます (デフォルトでは 1 時間)。各行には、期間の開始時刻、参照されるサーバー IP の数、およびその行の合計アクセス数がリストされます。

サーバー IP ランタイム・パラメーター (デフォルトではすべての IP アドレスを選択する “%” に 設定されています) を使用して、このレポートの出力を制限することができます。

スループット (グラフィカル)

このレポートは、表形式スループット・レポートの図表バージョン「分散ラベル付き線グラフ」で、期間の開始時刻ごとに 1 つのデータ・ポイントでレポート期間中のアクセス総数を作図します。

サーバー IP ランタイム・パラメーター (デフォルトではすべての IP アドレスを選択する “%” に 設定されています) を使用して、このレポートの出力を制限することができます。

アクティブなプライバシー・セット・タスクの数

1 つ以上のプライバシー・セット・タスクを含む、アクティブな Guardium 監査プロセスの数。一元管理が使用されている場合、このレポートには中央マネージャーに関するデータのみが含まれ、すべての管理対象ユニットに関しては空になります (「要求された照会のデータが見つかりません」という標準メッセージが表示されます)。このレポートは 標準とは異なるランタイム・パラメーターを持ちます。開始日付と終了日付のパラメーターがないため、1 つ以上のプライバシー・セット・タスクが含まれるすべての監査プロセスが レポートされます。このレポートのベースとなる照会 (アクティブ・プライバシー・セット・プロセスの数) の コピーを作成することはできますが、デフォルト・レポートのコピー作成や再生成は できません。複製された照会には、すべての標準ランタイム・パラメーター (開始日付と終了日付 を含む) が含まれます。

Guardium ジョブ・キュー

Guardium ジョブ・キューを表示します。ジョブごとに、プロセス実行 ID、プロセス・タイプ、状況、分類/評価プロセス ID、レポート結果 ID、分類/評価の記述、監査タスクの記述、キュー時刻、開始時刻、終了時刻、およびデータ・ソースがリストされます。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	Guardium ジョブ・キュー	使用不可

ランタイム・パラメーター	演算子	デフォルト値
ジョブの記述	LIKE	%
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

ディスカバーされたデータベース

このレポートは、レポート期間について、「データベース・タイプ」属性値が不明でない「ディスカバーされたポート」エンティティーごとに、プローブ・タイム・スタンプ、サーバー IP、サーバーのホスト名、データベース・タイプ、ポート、ポート・タイプ、およびその行の「ディスカバーされたポート」数をリストします。

ドメイン	ベースとなる照会	メイン・エンティティー
オートディスカバリー	ディスカバーされたデータベース	ディスカバーされたポート

ランタイム・パラメーター	演算子	デフォルト値
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

データ・ソース

このレポートは、管理者用とユーザー用の両方のデフォルトのレイアウトに表示されます。「事前定義レポート - 共通」ページの『データ・ソース』を参照してください。

データ・ソース・バージョン履歴

このレポートは、管理者用とユーザー用の両方のデフォルトのレイアウトに表示されます。『事前定義レポート - 共通』ページの データ・ソース・バージョン履歴を参照してください。

Guardium ジョブ・キュー

Guardium ジョブ・キューを表示します。ジョブごとに、 プロセス実行 ID、プロセス・タイプ、状況、分類/評価プロセス ID、レポート結果 ID、 分類/評価の記述、監査タスクの記述、キュー時刻、開始時刻、終了時刻、 およびデータ・ソースがリストされます。

未処理監査プロセスのレビュー

このレポートは、各 Guardium ユーザー・ログイン名について、 未処理の Guardium 監査プロセスの数とタイプをリストします。未処理の監査プロセスには、「Reviewed」または「Signed」以外の状況属性値 (「タスク結果 To-Do リスト」エンティティー内) が あります。このレポートは、標準とは異なるランタイム・パラメーターを持ちます。 開始日付と終了日付がないため、未処理のタスクの結果がすべて レポートされます。このレポートのベースとなる照会のコピー (同じ名前を持つ) を 作成することができますが、デフォルト・レポートのコピー作成や再生成は できません。複製された照会には、すべての標準ランタイム・パラメーター (開始日付と終了日付 を含む) が含まれます。

アクティブな監査プロセスの数

アクティブな Guardium 監査プロセスの数。一元管理が使用されている場合、このレポートには中央マネージャーに関するデータのみが含まれ、 すべての管理対象ユニットに関しては空になります (「要求された照会のデータが見つかりません」という標準メッセージが表示されます)。このレポートは標準とは異なるランタイム・ パラメーターを持ちます。開始日付と終了日付のパラメーターがないため、すべてのアクティブな 監査プロセスがレポートされます。このレポートのベースとなる照会 (アクティブ・プロセス数) の コピーを作成することはできますが、デフォルト・レポートのコピー作成や再生成はできません。 複製された照会には、すべての標準ランタイム・パラメーター (開始日付と終了日付 を含む) が含まれます。

インストール済みポリシーの表示

この特別なレポートは、「現在インストールされているポリシー」パネルに、インストールされているポリシーに関する情報 (ポリシーの名前、それに含まれるルールの数、そのポリシー定義の設定など) を表示します。このレポートのベースとなる照会にアクセスすることは できません。

ポリシー違反数

このレポートは、レポート期間について、ログに記録されたポリシー違反の数を 表示します。

ログに記録されるしきい値アラート

このレポートは、「しきい値アラート詳細」エンティティーの「アラートの記述」属性に基づいて、ログに記録されるしきい値アラートの タイプごとに、レポート期間中にログに記録されたアラートの総数を 示すバーを表示します。

ログに記録される R/T アラート

このレポートは、「ポリシー・ルール違反」エンティティーの「アクセス・ルールの記述」属性に基づいて、ログに記録されたリアルタイム・アラートのタイプごとに、レポート期間中にログに記録されたアラートの総数を示すバーを表示します。

違反/インシデント

インシデント管理を参照してください。

親トピック: 定義済みレポート

共通の事前定義レポート

このセクションでは、デフォルトのユーザー・アクセス権限またはデフォルトの admin アクセス権限のいずれかを持つユーザーが使用できるすべての事前定義レポートについて簡単に説明します。

共通のレポートは次のとおりです。

- データ・ソース・バージョン履歴
- データ・ソース

ステータス・モニター

「ステータス・モニター」のグラフィカル・レポートには、Guardium アプライアンスの現在の状態 (1 秒当たりに処理するパケット数と要求数、使用されているディスク・スペースとメモリーの量など) が表示されます。次の表で、各フィールドについて説明します。

ボックスには、Linux VMSTAT コマンドの出力が表示されます。このコマンドの知識があれば、 これらの統計はおなじみのものです。

フィールド	記述
プロセス	プロセス数: **r**: ランタイムを待機しています。 **b**: 割り込み不能なスリープ状態です (ブロックされ、別のイベントを待機中です)。

フィールド	記述
メモリー	メモリー使用量 (KB): **swpd**: 使用されている仮想メモリーの量。 **free**: 使用されていないメモリーの量。 **buff**: バッファーとして使用されている量。 **cache**: キャッシュ用に予約されている量。
スワップ	メモリーの量 (KB): **si**: ディスクからスワップインされています。 **so**: ディスクにスワップアウトされています。
入出力	入出力ブロック (KB/s): **bi**: ブロック・デバイスから受信したブロック **bo**: ブロック・デバイスに送信したブロック
システム	システム: **in**: 1 秒当たりの割り込み (クロックを含む) **cs**: 1 秒当たりのコンテキスト切り替え
CPU	次で使用する合計 CPU 時間のパーセンテージ。 **us**: 非カーネル・コードの実行に費やす時間 **sy**: カーネル・コードの実行に費やす時間 **id**: アイドル時間 (入出力待ちを含まず) **wa**: 入出力待ちに費やす時間 **st**: 仮想マシンから流用する時間
(n)pps / (m)rps	分析エンジンの横の矢印で、最近 5 分間の 2 種類の平均値が算出されます。 **n** はネットワーク・パケットの 1 秒当たりの平均数、**m** はネットワーク・データベース要求の 1 秒当たりの平均数です。
分析エンジン (q-d) ------ (p)	分析エンジンでは、 最初の行に処理のためキューに入れられているメッセージの総数 (**q**) がリストされ、その後に、バッファーがフルになる心配があったためドロップされたメッセージの数 (**d**) が続きます。2 番目の行には、 処理されたメッセージの総数 (**p**) がリストされます。 処理された数は、検査エンジンが再始動されると必ずゼロにリセットされます。
サーバー・タイプ (q) ---- (p)	各サーバー・タイプでは、処理を待つメッセージの数 (**q**) と、 処理済みのメッセージの数 (**p**) がリストされます。
空きディスク・スペース	空いているバイト数。
データベース使用率	データベースのスペース割り振りのうち、 使用されている割合。
ファイル/その他 (Files/Other)	ステータス・モニターの「ファイル/その他 (Files/Other)」部分は、nondb-sql ロガーで累積されたデータを表します。 nondb-sql ロガーは、アナライザーによって内部的にクローズされている (INACTIVE_FLAG=-1) 「無視された」セッションからアナライザーに到達したセッション・クローズ・イベントを記録します。アナライザーは、(セッションが長期間非アクティブである場合に) タイムアウトによって接続を閉じることができます。タイムアウトによってクローズされている「無視された」セッションからアナライザーにセッション・クローズ・データが到達した場合は、それが nondb-sql-logger セクションに記録されます。 アナライザーがデータベースにデータを直接記録することはありません。このセクションは、アナライザーによってロガーに送信された DB 要求 (GDM_SECURE_PARAMS への挿入など) の数や、サポートされるその他のプロトコル (FTP など) も表します。

データ・ソース・バージョン履歴

デフォルト・レイアウトのロケーション

- 管理者:「データ・ソース」レポートからドリルダウンして入手可能
- ユーザー:「ディスカバー」>「データベース・ディスカバリー」

データ・ソース

定義されているすべてのデータ・ソースをリストします。データ・ソース・タイプ、データ・ソース名、 データ・ソースの記述、ホスト、ポート、サービス名、ユーザー名、データベース名、 最後の接続、共有、接続プロパティー。

このレポートの出力は、「データ・ソース名」ランタイム・パラメーターで制限できます。 このパラメーターは、デフォルトではすべてのデータ・ソースを選択する「%」に設定されています。

ドメイン	ベースとなる照会	メイン・エンティティー
内部 - 使用不可	データ・ソース	使用不可

ランタイム・パラメーター	演算子	デフォルト値

ランタイム・パラメーター	演算子	デフォルト値
データ・ソース名	LIKE	%
期間開始	>=	NOW -1 DAY
期間終了	<=	NOW

事前定義監査プロセス

「アプライアンスのモニター」という事前定義監査プロセスがあり、これには、リストされた進行中のレポートが含まれます。この監査プロセスは、デフォルトでは非アクティブになっています。 管理者は、必要に応じてこれを活動化したり、スケジュールに入れたりすることができます。

注: この監査プロセスをスケジューリングする場合、 各レポートの FROM/TO の日付が定義済みのプロセス間隔と整合性があることを確認してください (例えば、 監査プロセスが週に一度しか実行されない場合は、レポート期間を 1 日にしても意味がありません。6 日間はアクティビティーがないことになります)。
「アプライアンスのモニター」監査プロセスには、以下のレポートが含まれます。

- Guardium への失敗したログイン
- アクティブな Guardium ユーザー
- 統合エラーまたはアーカイブ・エラー
- ポリシー関連の変更
- 検査エンジンと S-TAP の変更
- データ・ソース変更
- CAS インスタンス構成変更
- CAS インスタンス
- CAS テンプレート
- スケジュールされたジョブの例外

親トピック: 定義済みレポート

ダッシュボードの作成およびレポートの追加

1 つ以上のダッシュボードを作成し、それらにレポートを追加し、外観を構成することができます。

始める前に

定期的に表示するレポートの編成方法について検討します。レポートは 1 つのダッシュボードで表示しますか、それとも複数のダッシュボードで表示しますか。レポートをグループ化し順序付ける際に、レポートの目的または重要度のいずれを基準としますか、あるいは何か別のアプローチを使用しますか。ダッシュボードの再配置や新規作成は、いつでも行うことができます。

このタスクについて

手順

1. 「マイ・ダッシュボード」 > 「新規ダッシュボードの作成」をクリックして、新規ダッシュボードを開きます。
2. 「名前」フィールドに記述名を入力します。この名前は、メニュー内のダッシュボードのリストで使用されます。
3. 「レポートの追加」 をクリックすると、使用可能なレポートのリストが表示されます。特定のレポートをお気に入りとして指定した場合は、「お気に入り」ボックスにチェック・マークを付けると、それらのレポートのみのリストを表示できます。グラフィカル・レポートのみを表示したい場合は、「グラフのみ」ボックスにチェック・マークを付けます。
4. 「レポートの追加」ダイアログには、指定された基準を満たすすべてのレポートのリストが表示されます。レポートのリストを参照することも、「フィルター」フィールドに文字列を入力することもできます。 文字を入力するにつれて、レポートのリストが更新されます。
5. レポートのタイトルをクリックすると、そのレポートがダッシュボードに追加されます。必要な数だけレポートを追加してください。レポートを追加し終えたら、「閉じる」をクリックします。

タスクの結果

選択されたいくつかのレポートに簡単にアクセスできるダッシュボードが用意できました。

次のタスク

ダッシュボードの外観をレビューします。ダッシュボードが使いやすいか、必要な情報を簡単に検索できるかを確認してください。問題がある場合は、さらに構成することができます。
親トピック: レポート

ダッシュボードの構成

ダッシュボードができるだけ便利になるように、外観のいくつかの側面を構成することができます。

このタスクについて

レポートの使用方法について検討します。どのような配置にすると目標を達成しやすいでしょうか。いろいろ変更してみてください。

手順

1. レポートを再配置します。レポートを移動するには、レポートのタイトル・バーにカーソルを置いて、新しい場所にドラッグします。

2. 新たな列数を選択するには、「列数」エリアで「1」、「2」、または「3」をクリックします。デフォルトでは、レポートは 2 列で表示されます。各レポートにもっとスペースが必要な場合は、「1」をクリックすると、レポートをダッシュボードの最大幅にしたときの表示を確認できます。 より多くのレポートを一度に表示したい場合は、3 列を試してみてください。
3. レポートのサイズを変更します。サイズ変更アイコンをドラッグして、レポートの長さや幅を変更します。レポートの幅を調整すると、その列にあるすべてのレポートで新しい幅が採用されます。列数を変更すると、すべての列はそれぞれのデフォルト幅に戻ります。

ダッシュボードの使用

ダッシュボードにレポートを追加して、外観をカスタマイズするには、以下のステップを実行します。

このタスクについて

ダッシュボードは、「ペインに追加」および「マイ・レポートに追加 (Add to My Reports)」の代わりに使用されます。

手順

1. ナビゲーションでダッシュボード・アイコンをクリックします。
2. 次に「新規ダッシュボードの作成」をクリックします。
3. 「レポートの追加」をクリックして、アクセス可能なすべてのレポート (作成した新規レポートを含む) から、レポートを選択します。
4. フィルタリングを利用すると、関心のあるレポートを素早く見つけることができます。
5. レポート名をクリックして、ダッシュボードに追加します。単に各レポートを選択するだけで、必要な数のレポートをダッシュボードに追加できます。
6. レイアウトを選択して、ダッシュボードをカスタマイズします。デフォルトは 2 列です。1 列 - レポートではダッシュボードの幅が想定されます。2 列 - レポートではダッシュボードの半分の幅が想定されます。3 列 - レポートではダッシュボードの 3 分の 1 の幅が想定されます。
7. 画面内のレポートを移動して、ダッシュボードをカスタマイズします。グラフをカスタマイズするには、 アイコンを使用します。
8. 特定のレポートをお気に入りとして指定するには、 アイコンを選択します。レポートをダッシュボードに追加する場合は、お気に入りに基づいてフィルタリングするか、グラフに基づいてフィルタリングします。
9. 「編集」アイコンをクリックして、ダッシュボードに名前を付けます。
10. ダッシュボードを削除するには、「削除」アイコン をクリックします。

レポートの表示

レポートの表示には、ダッシュボードや UI 検索など、いくつかの方法があります。

レポートは、以下のようにいくつかの方法で表示できます。

- レポートをダッシュボードに保存した場合は、ダッシュボードを開いてレポートを表示します。
- レポートをダッシュボードに追加できます。ダッシュボードを開き、「レポートの追加」をクリックして、リストからレポートを選択します。
- レポートをカスタム・レポートに追加します。
- レポート・ライフサイクルのカテゴリーにいくつかのレポートがリストされます。
- 最も関連性の高いライフサイクルの下にいくつかのレポートがリストされます。
- ユーザー・インターフェース (UI) の検索機能を使用して、ライフサイクルまたはダッシュボードにあるレポートを検索できます。バナーで、検索ボックスの横のドロップダウン・リストから「ユーザー・インターフェース」を選択します。検索ボックスにレポートの名前を入力します。数文字入力すると、結果が表示され始めます。結果のリストからレポートを選択します。

v10.6 より前から v10.6 以上にアップグレードした後、同じ照会に基づくレポートが複数ある場合は、アップグレード前のバージョンのレポートと同じ名前を持つすべてのレポートに対応した照会が 1 つあります。

以下の選択項目 (アイコン) を使用すると、レポートの編集および構成を行うためにアクセスできます。

	このレポートの照会を編集 (照会 - レポート・ビルダーの

	使用)
	今すぐ1回実行する特別プロセス
	新規ウィンドウで開く
	レポート表示の変更
	レポートの列の構成:列を追加または削除します。
	ランタイム・パラメーターの

構成。ランタイム・パラメーターは、照会条件で使用される値を提供します。すべての照会 - レポートにはデフォルトのランタイム・パラメーター・セットがあり、照

	会 - レポートには任意の数のランタイム・パラメーターを定義できます。(ランタイム・パラメーターの変更)
	お気に入りに追加
	レポートの生成を停止 (生成するのに

	長い時間がかかるレポートに関連します)
	レポートの最新表示。レポートのリフレッシュを参照してください。

列を非表示にすることができます。列アイコンをクリックして、非表示にする列のチェック・ボックスをクリアしてください。

任意の列の内容に基づいてレポート・データをソートできます。ソートの基準となる列のタイトルをクリックしてください。順序を逆にするには、タイトルをもう一度クリックします。ソートは常に実際のデータ値に対して実行されるため、定義されている別名は無視されます。

レポートの表示中に、そのレポートを印刷することができます。「エクスポート」 > 「完全印刷用レポート」をクリックすると、レポートの印刷用コピーが新規タブで開きます。 新規タブでプリンター・アイコンをクリックして、レポートを印刷します。レポートを印刷するもう一つの方法として、レポートを PDF ファイルにエクスポートし、その PDF ファイルを印刷することができます。
注: PDF テキストが小さすぎて読めない場合は、ページの幅を考えると、PDF レポートを横方向に拡大するのは物理的限界があります。PDF レポートの各行は 1 行に収まる必要があるため、データに合うように書体サイズが変更されますが、すべてのデータを表示するために非常に小さい書体サイズになる可能性があります。

グラフィカル・レポートをカスタマイズするには、「グラフのカスタマイズ」アイコンをクリックします。選択肢としては、データから線グラフへの変換、X 軸と Y 軸の方向の変更、レポートから円グラフまたは積み上げ縦棒グラフへの変換があります。

Oracle の情報を表示するレポートを表示する場合、疑問符 (?) 文字が使用されることがあります。これは、ログイン情報を使用できないことをビューアーに通知するためのものです。さらに、Oracle の情報を表示するレポートを表示する場合、数字「-1」が現れた箇所は、影響を受けたレコードの数が不明であること意味します。すべての Oracle セッションは記録されます。ログインが失敗したセッションであっても記録されます。

リモート接続を使用する Linux システムまたは Windows システムがログイン・パケットを使用して OS ユーザーを送信しなかった場合、その OS ユーザーはレポートに表示されません。Linux のローカル接続では、UID チェーンを使用してユーザーを識別できます。UID チェーンをサポートするシステムについては、S-TAP サポート・マトリックスを参照してください。

Big Data Intelligence レポートの表示には異なる点がいくつかあります。

- レポートには、データが 1000 項目単位のバッチで表示されます。各ページには右下隅で定義された行数が含まれます。レポートの下にあるページ番号によってそのバッチ内のページが示されます。ページ番号を使用して、バッチ内でページをめくります。次のバッチに進むには、「次のバッチ」を使用します。バッチは順方

向に移動できます。前のバッチを表示するには、照会を再度実行します。
注: いずれかの列で矢印をクリックしてレポートをソートした場合は、次のバッチを使用することはできません。

- 「リフレッシュ」 をクリックすると、照会が再実行されます。

- ランタイム・パラメーターの変更
  ランタイム・パラメーターを変更して、レポートの内容や表示を制御することができます。
- レポートのリフレッシュ
  一部のレポートは、データを自動的にリフレッシュするように構成されています。 その他のレポートでは、UI を使用してデータを手動でリフレッシュできます。
- レポートのエクスポート
  レポートを PDF ファイルまたはコンマ区切り値のファイルにエクスポートできます。
- ドリルダウン・レポートの表示
  多くのレポートが、より細かいデータを提供するドリルダウン・レポートにアクセスできるようになっています。
- 今すぐ 1 回実行する特別プロセス
  このプロセスでは、新しい監査プロセス・レポートを作成します。ユーザーに対してそのようなプロセスが存在している場合、パラメーターが更新され、同じプロセスが使用されます。

親トピック: レポート

ランタイム・パラメーターの変更

ランタイム・パラメーターを変更して、レポートの内容や表示を制御することができます。

次の表に、標準のランタイム・パラメーターをリストしています。レポートには、その他のパラメーターがある場合があります。

ランタイム・パラメーター	デフォルト	記述
期間の開始日を入力	NOW -3 HOUR。	どの場合でも、レポートの開始日は必須です。Big Data Intelligence の場合、デフォルトは NOW -1 DAY です。事前定義レポートのデフォルトは異なる場合があります。
期間の終了日を入力	現在	これはレポートの終了日であり、どの場合でも必須です。事前定義レポートのデフォルトは異なる場合があります。
Guardium アプライアンス	すべて	Big Data Intelligence のみ。すべての Guardium アプライアンス (デフォルト) または選択されたアプライアンスのデータが取得されます。
タイム・ゾーン	Guardium システムのタイム・ゾーン。	Big Data Intelligence のみ。このタイム・ゾーンに従ってデータが取得され、日付フィールドがこのタイム・ゾーンに表示されます。
リモート・データ・ソース	なし	中央マネージャー環境では、リモート・データ・ソースのリストにある Guardium システムを選択してその管理対象ユニット上でレポートを実行できます。
別名の表示	なし (システム共通のデフォルトが使用されることを意味します)。	「オン」を選択すると常に別名が表示され、「オフ」を選択すると別名は表示されなくなります。「オン」または「オフ」を選択してから、「デフォルト」を選択すると、システム共通のデフォルト (管理者によって制御される) に戻ります。
リフレッシュ頻度 (秒)	0	レポートがリフレッシュされる頻度 (秒)。0 は、レポートが自動的にリフレッシュされないことを意味します。

GuardAPI コマンド list_parameter_names_by_report_name を使用します。この関数は、レポート名を入力パラメーターとして取り、そのレポートのランタイム・パラメーター名のリストを返します。

1. レポートにアクセスします。
2. 「ランタイム・パラメーターの構成」 をクリックします。
3. 必要に応じてパラメーターを変更し、「OK」をクリックします。

親トピック: レポートの表示

レポートのリフレッシュ

一部のレポートは、データを自動的にリフレッシュするように構成されています。 その他のレポートでは、UI を使用してデータを手動でリフレッシュできます。

レポート・データを手動でリフレッシュするには、以下のようにいくつかの方法があります。

- レポート・ツールバーの「リフレッシュ」 をクリックします。
- 任意のツールバー・ボタンを使用して、レポートの印刷、レポート・データのダウンロード、またはレポートの PDF ファイルへの書き込みを行います。レポート・データはこれらのアクションを実行する前にリフレッシュされます。
- refreshRate パラメーター値を設定することによって、周期的なリフレッシュのための時間間隔を設定します。このタスクを実行するには次のようにします。
  - レポート・ツールバーの をクリックします。
  - 「構成」ダイアログで、「refreshRate」パラメーターを、次に行うレポート・データの更新までの秒数に設定します。デフォルト値のゼロは、レポート・データをスケジュール・ベースでリフレッシュしないことを表します。
  - 「OK」をクリックします。

親トピック: レポートの表示

レポートのエクスポート

レポートを PDF ファイルまたはコンマ区切り値のファイルにエクスポートできます。

レポートの内容を以下にエクスポートできます。

- PDF. 大きな PDF ファイルを生成すると、処理中に UI がタイムアウトになる可能性があります。大きな PDF ファイルを生成する予定の場合は、この処理を監査プロセスの一部として行うか、UI のタイムアウト値を大きくして、この問題を回避することを検討してください。
- すべてのレコードの CSV
- 表示レコード (現在表示されているデータ) の CSV

1. レポートにアクセスします。
2. ツールバーの「エクスポート」をクリックして、以下のいずれかを選択します。
   - レコードをすべてダウンロード: すべてのレコードの CSV。ファイルを保存したり、表示したりすることができます。
   - 表示レコードのダウンロード: 表示レコード (現在表示されているデータ) の CSV。ファイルを保存したり、表示したりすることができます。
   - 完全印刷用レポート: 新しいブラウザー・ウィンドウが開き、すべてのレポートの詳細が表示されます。
   - PDF 形式でダウンロード: ファイルを保存したり、表示したりすることができます。

親トピック: レポートの表示

ドリルダウン・レポートの表示

多くのレポートが、より細かいデータを提供するドリルダウン・レポートにアクセスできるようになっています。

表形式レポートでドリルダウン・アクションが使用できる場合は、グリッドの行を右クリックすると、使用可能なドリルダウン・アクションを示すコンテキスト・メニューが表示されます。

ドリルダウン・レポートとして使用可能にするには、以下のようにします。

- ドリルダウン・レポートのすべてのランタイム・パラメーターは、表示されているレポートから使用できなければなりません。
- セキュリティー・ロールが割り当てられている場合、ドリルダウン・レポートへのアクセス権が必要です。

親トピック: レポートの表示
関連タスク:
照会のドリルダウン制御の変更

今すぐ 1 回実行する特別プロセス

このプロセスでは、新しい監査プロセス・レポートを作成します。ユーザーに対してそのようなプロセスが存在している場合、パラメーターが更新され、同じプロセスが使用されます。

このタスクについて

このプロセスの動作は次のとおりです。

1. 新規プロセスの場合は、emailContentType パラメーターに示されるコンテンツ・タイプで、リスト (存在する場合) に 1 つまたは複数の E メール受信者を作成できます。また、includeUserReceiver パラメーターが true の場合は、(API を呼び出して) ログイン・ユーザーのためのユーザー・レシーバーも作成します。
2. 既存のプロセスの場合は、すべての E メール受信者が削除され、emailContentType パラメーターに定義されているコンテンツ・タイプで、新規リスト (存在する場合) の E メールに置き換えられます。リストが空の場合は、E メール・アドレス・レシーバーがすべて削除されます。 ユーザーのレシーバーが既に存在する場合は、includeUserreceiver が false でもそれは削除されませんが、このパラメーターが true で、かつそのようなレシーバーが存在しない場合は、追加されます。

監査プロセスが生成されると、これは (「今すぐ 1 回実行」と同じように) 自動的に実行され、ユーザーはその監査プロセスが自分の To-Do リスト上のアイテムとなることを期待します。特別な監査プロセスを作成する GuardAPI では、結果が 1 日ではなく 7 日間保持されます。結果は 7 日後に削除されます。パラメーターについて詳しくは、『GuardAPI 入力生成』ヘルプ・トピックの GuardAPI コマンド create_ad_hoc_audit_and_run_once を参照してください。

手順

1. 「今すぐ 1 回実行する特別プロセス」アイコン  をクリックします。
2. 「随時」ダイアログで必要に応じて入力し、「OK」をクリックします。
   - E メール・アドレス: E メール・アドレスのコンマ区切りリスト
   - E メール受信者のコンテンツ・タイプ: PDF または CSV (ラジオ・ボタン 0 - PDF / 1 - CSV)
   - ユーザーを受信者として追加 (チェック・ボックス)

親トピック: レポートの表示

照会 - レポート・ビルダーの使用

事前定義レポートではニーズに対応できない場合は、照会を最初から作成するか、既存の照会をコピーして変更します。

照会の作成を開始する前に、レポートに何を記述するのかについて慎重に計画してください。一般的なユース・ケースは 2 つあります。

- システム内の特定のオカレンスを確認したい。このタイプの照会には 2 つの部分があります。システムで発生した 1 つ以上のイベント。これは、条件によって定義されます。このイベントが発生したときのシステムの何について知りたいですか? これらは、レポートに表示される詳細 (列) です。
- システムの一部分の状況を知りたい。この場合は、レポートに表示する列を指定するだけでよいことがほとんどです。

最初に、必要なデータを含むドメインを見つけます。ドメインには、特定の機能や目的 (データ・アクセス、例外、ポリシー違反など) に関連するデータ・セットが含まれます。各照会は、1 つのドメインに基づき、そのドメインからデータを返します。すべてのドメインの説明については、『ドメインのエンティティーおよび属性』を参照してください。各ドメインには、属性のグループである 1 つ以上のエンティティーがあります。属性は、レポートで列として使用できるフィールドです。一部のエンティティーは、関連データにアクセスできるように複数のドメインに含まれています。例えば、「セッション」エンティティーは、「アクセス」ドメインと「例外」ドメインの両方に含まれています。

いつ誰が何を実行したかに関するデータがトラッキングされます。詳細には静的なものと動的なものがあります。UI で上部から下部に向かって示されるドメイン内のエンティティーは、静的な詳細から始まり、その後に非静的な詳細が続きます。例えば、「アクセス」ドメインでは、最初のエンティティーはクライアント/サーバーです。クライアント/サーバーの各ペアは 1 回保存されます。次のエンティティーはセッションです。クライアント/サーバーのペアには、非静的な詳細が示される複数のセッション (例えば、セッション開始および非アクティブ・フラグ) があります。これにより、クライアント/サーバーとセッションの間に 1 対多の関係が作成されます。セッション開始の各値を表示するには、複数の行が必要になることがあります。各セッション開始を記述してレポートを無用に長くするのではなく、カウント・オプションを使用してセッション開始の発生回数を表示し、ドリルダウンしてさらに詳細なレポートを表示することができます。経験法則として、GUI でメイン・エンティティーの位置が高いほど、レポートに示される行および値は少なくなります。レポートを管理しやすくなり、いつでもドリルダウンしてさらに詳細を表示することができます。

1 つのドメインに含まれていない 2 つのエンティティーの詳細が必要な場合は、カスタム・ドメインを作成できます (カスタム・ドメインを参照)。

照会では、1 つのドメインからのみデータが返されます。 照会が定義されるときに、そのドメイン内の 1 つのエンティティーが、照会のメイン・エンティティーに指定されます。 照会により返される各データ行には、選択された属性について要求された期間で返された値に一致するメイン・エンティティーの出現数が含まれます。これにより、1 対 1 の関係を持たないエンティティーから 2 次元のレポートの作成が可能になります。

ドメインを特定したら、そのドメイン内の事前定義レポートを調べて、目的のものと近いものがあるかどうかを確認します。ある場合は、それをコピーして変更できます。ない場合は、照会を最初から作成します。

レポート・データ (列) を定義します。ドメイン内のすべてのエンティティーから列を選択できます。

オプションで、条件を定義します。条件は、使用した場合、特定のデータをレポートに組み込むためのトリガーとなります。例えば、状況に関する照会では、トリガーは必要ありません。単に、システム内のエレメントの状況について知りたいだけであるためです。ただし、データベースのグループに対する特定のユーザーによる特定のアクションを確認したい場合は、これらが照会条件になります。条件では、ドメイン内の属性を演算子とともに使用します。条件とレポート列の間には特有の関係はありません。条件に属性を列として追加することも、追加しないこともできます。

オプションで SQL ステートメントで式の作成または Having 節を定義することもできます。

照会の保存時に照会と同じ名前で作成されるレポートには、これらのデータが表示されます。デフォルトのレポートは表形式のレポートであり、照会の構造を反映して、各属性が別個の列に表示されるものになります。すべてのランタイム・パラメーターと表形式レポートの表示構成要素はカスタマイズ可能です。

クエリー・ビルダーには、以下で説明している照会のさまざまな側面を構成するための 6 つのリボンがあります。

「照会 - レポート・ビルダー」ページの下部には以下のボタンがあります。

- ダッシュボードに追加: 表示されているレポートを定義済みのダッシュボードに追加するためにクリックします
- マイ・カスタム・レポートに追加: 「レポート」 > 「マイ・カスタム・レポート」に追加するためにクリックします
- 照会の要約: 照会の文字ベースの要約を開くためにクリックします

照会における「完全な SQL」属性に関する注意事項 照会で「完全な SQL」属性を使用する際には注意が必要です。これを使用すると、 属性の個別の値 (この場合、完全な SQL 照会文字列) がそれぞれ個別の行で返されるため、 過度に大容量のレポートが生成される可能性があります。一方、 「完全な SQL」の文字列を予期している場合に、レポートに全く情報が含まれない、 または多くのブランク列が含まれることがあります。Guardium は、ポリシー・ルールによって 指示された場合にのみ「完全な SQL」をキャプチャーします。そして、レポート期間中には、 ポリシー・ルールが起動されない可能性があります。「完全な SQL」属性と、SQL にドリルダウンする機能を混同しないでください。 データ・アクセス・ドメインの照会は、SQL 要求に関して何も行いません。

- 新規照会の作成、既存の照会の変更
  照会用にドメインを選択すると、その選択されたドメインをメイン・エンティティーとして使用するすべての既存の照会 (レポート) が表示されます。これらの照会をどれでもコピーできるほか、照会を最初から作成することもできます。
- 照会名および属性の定義
  ドメインを選択してから、照会名およびメイン・エンティティーを設定し、ロール、データマート、ドリルダウン制御、および API 割り当てを構成します。
- 列表示の選択
  ドメイン内のすべてのエンティティーの属性から、レポートの列を選択し、ソート階層を定義します。
- ソート順の設定
- 照会条件の定義
- 照会条件での式の作成の追加
  このフィーチャーは、属性の全体的なコンテンツそのものには基づいていないが、その属性の一部、その属性の関数、または複数の属性を組み合わせた関数に基づいた条件 (ユーザー定義文字列および数式を含む) を追加する必要がある場合に使用します。
- HAVING 条件
  1 つ以上の列がカウント、最小、最大、平均、合計のいずれかの機能を使用していて、「グループ化基準」が必要な場合、このリボンを使用できます。
- レポート表示の変更
  レポート・タイプ (表またはグラフ) を選択して、列名を変更し、色表示ルールを構成することができます。

親トピック: レポート
関連概念:
ドメイン、エンティティー、および属性

新規照会の作成、既存の照会の変更

照会用にドメインを選択すると、その選択されたドメインをメイン・エンティティーとして使用するすべての既存の照会 (レポート) が表示されます。これらの照会をどれでもコピーできるほか、照会を最初から作成することもできます。

このタスクについて

各ドメインには、特定の目的や機能 (データ・アクセス、例外、ポリシー違反など) に関連するデータ・セットが含まれます。 名前は自明なので、関連するドメインを簡単に選択できます。すべてのドメインの説明については、『ドメインのエンティティーおよび属性』を参照してください。

手順

1. 「調査」 > 「例外」 > 「照会 - レポート・ビルダー」にナビゲートするか、「レポート」 > 「レポート構成ツール」 > 「照会 - レポート・ビルダー」にナビゲートして、「クエリー・ビルダー」を開きます。
2. 照会するドメインを「ドメインの選択」ドロップダウンから選択します。 そのドメインに基づく照会 (レポート) のリストがドメイン・ネームの下に表示されます。
3. 事前定義の照会をクリックすると、ダイアログ・ボックスが開き、以下を選択できます。
   - 元の照会を開き (「元を開く」をクリック)、いくつかの属性 (例えば、照会のセキュリティー・ロールの管理、データマートへの照会の追加、照会のドリルダウン制御の変更、API 割り当ての変更、ダッシュボードの作成およびレポートの追加) を変更します。照会属性や照会条件を変更することはできません。
   - 「コピーの作成」 をクリックして新規名を指定し、コピーを作成します。照会名および属性の定義に進みます。
4. ユーザー定義の照会をクリックすると、右側のペインにそのプロパティーが開きます。照会を編集するか、 をクリックして照会をコピーすることができます。照会構成タスクを続行します。
5. 新規照会を作成するには、「新規」 をクリックします。「新規照会」ページが開きます。「新規」 をクリックします。「新規照会」ページが開きます。照会名および属性の定義に進みます。

親トピック: 照会 - レポート・ビルダーの使用

照会名および属性の定義

ドメインを選択してから、照会名およびメイン・エンティティーを設定し、ロール、データマート、ドリルダウン制御、および API 割り当てを構成します。

このタスクについて

照会を最初から作成する場合、次のステップとして、(ドメインから) メイン・エンティティーを選択します。 選択したメイン・エンティティーによって、以下のことが決まります。

- レポートの詳細レベル。 レポートに含まれるメイン・エンティティーのオカレンスごとに、1 行のデータがあります。エンティティー階層内でのメイン・エンティティーのロケーションは、 どの値が表示可能になるかという点で重要です。階層の上位にあるエンティティーの属性には、タイプ値のフィールド・モードを設定できます。階層の下位にあるエンティティーの属性には、タイプ値のフィールド・モードを設定できません。照会 - レポート・ビルダーのエンティティー・リストで階層を確認できます。階層は、リストに表示されているとおり (上部から下部) です。
- レポートのデータを選択するために「期間開始」および「期間終了」ランタイム・パラメーターとの対比が行われる時間フィールド。照会 - レポート・ビルダーは、(パラメーターの中でも特に) メイン・エンティティーを使用して、「期間開始」値および「期間終了」値の定義時に使用される時間フィールドを決定します。これは、長期実行セッションの場合 (プールされたセッションがアプリケーション・サーバーによって開かれたままである場合など) に重要になることがあります。適用可能な場合は「アクセス期間」エンティティーの「期間の開始」/「期間の終了」が使用され、それ以外の場合はメイン・エンティティーに従って期間の値が選択されます。
  - セッション - 使用されるタイム・スタンプは、セッション・エンティティーに対する最後の更新になります。
  - セッション開始 - セッション・エンティティーの開始時刻が使用されます。
  - セッション終了 - セッション・エンティティーの終了時刻が使用されます。
  - 完全な SQL -「完全な SQL」ドメインからのタイム・スタンプ。 値にリンクされていない場合であっても、照会には「完全な SQL」ドメインからの行が含まれます (例えば、「全詳細をロギング」が設定されている場合、値はありません)。
  - 完全な SQL 値 -「完全な SQL」ドメインからのタイム・スタンプ。「完全な SQL」ドメインからの値がある場合にのみ、(それらが「フィールド」ドメインにリンクされていなくても) 照会に行が含まれます。
  - フィールド SQL 値 -「完全な SQL」ドメインからのタイム・スタンプ。「完全な SQL」ドメインからの値があり、 それらの値が「フィールド」ドメインにリンクされている場合にのみ、照会に行が含まれます。

このリボンのその他のオプションは、以下のとおりです。

- パーティションの最適化はデフォルトで有効にされており、パーティション化されたデータベース表での照会のパフォーマンスが向上します。「パーティションの最適化」チェック・ボックスを選択解除することで、この機能を無効にできます。Guardium サポートからの指示がない限り、パーティションの最適化を無効にしないでください。
- 「2 つのステージングで実行」を選択できます。タイプが「レポート」の監査タスクを 2 つのステージングで実行する場合はこれを選択してください。これは、特定の表での照会に関するレポートにのみ該当します。この 2 つのステージングによる方式は、列および条件に対する監査プロセスとして、特定のエンティティーで照会を実行する場合にのみ適用されます。そのエンティティーとは、アクセス (クライアント/サーバー)、セッション、アクセス期間、構文 (SQL)、オブジェクト、およびセンテンス (コマンド) です。`Like Group` 演算子または別名に関連したいずれかの演算子 (`In Aliases Group` など) が使用された条件が照会に含まれている場合や、条件で「Having」が使用されている場合、この 2 つのステージングによる方式は使用されません。デフォルトでは、照会は 1 つのステージングで実行されます。システム全体で 2 つのステージングによる照会を無効にするには、ファイル /var/log/guard/DontRunInTwoStages を作成します。 このファイルの存在は、2 つのステージングによる方式が使用されないことを示します。
  注: タプル (結合されたフィールド) を含むフィールドは、2 つのステージングによる実行ではサポートされていません。

注: 「メイン・エンティティー」ドロップダウン・リストに含まれるのは、1 次エンティティーだけです。ただし、2 次エンティティー (例えば、「セッション開始」および「セッション終了」) には、対応する 1 次エンティティー (例えば、「セッション開始」および「セッション終了」の「セッション」) を通じてアクセスすることができます。

手順

1. コピーされたレポートと新規レポートのいずれの場合も、右側のペインで照会名テキスト・ボックスに固有の照会名を入力します。
2. ドロップダウン・リストからメイン・エンティティーを選択します。
3. 「次へ」をクリックして、列を選択し、照会を保存します。照会が保存されると、このリボンで追加の関連ボタンを使用できるようになります。

- 照会のセキュリティー・ロールの管理
  デフォルトでは、照会を定義するユーザーのみがその照会に対するアクセス権限を持ちます。照会に対するアクセス権限を提供するために、その他のロール (またはすべてのロール) を追加および削除することができます。
- データマートへの照会の追加
- 照会のドリルダウン制御の変更
  デフォルトで、レポートのドリルダウン・メニューには、そのレポートの属性で提供できるランタイム・パラメーターを持つすべてのレポートが組み込まれます。ただし、通常のセキュリティー・ロール制約が課されます。

- API 割り当ての変更
  デフォルトで、Guardium アプリケーションには多くの API 関数をレポートにリンクした設定データが添付されています。これによりユーザーには、GUI を通じてレポート・データからの API への作成済み呼び出しが提供されます。「API 割り当て」を使用して、事前定義された Guardium レポートまたはカスタム・レポートへ追加 API 関数をリンクできます。

親トピック: 照会 - レポート・ビルダーの使用

照会のセキュリティー・ロールの管理

デフォルトでは、照会を定義するユーザーのみがその照会に対するアクセス権限を持ちます。照会に対するアクセス権限を提供するために、その他のロール (またはすべてのロール) を追加および削除することができます。

このタスクについて

照会を定義するユーザーは、追加のユーザーにアクセス権限を付与できます。照会は、その照会へのアクセス権限を持たないユーザーの UI から完全に除外されます。

手順

1. 「照会名」リボンで、「ロール」をクリックします。「セキュリティー・ロールの割り当て」ダイアログが開きます。
2. 個々のロールを選択または選択解除するか、すべてのロールにアクセス権限を付与する場合は「すべてのロール」を選択します。
3. 「OK」をクリックします。ロールが更新され、パネルが閉じます。

親トピック: 照会名および属性の定義

データマートへの照会の追加

このタスクについて

手順

1. 「照会名」リボンで「データマート」をクリックします。「データマート構成」ダイアログが開きます。
2. 既存のデータマートを選択するか、「新規」をクリックして新規データマートを作成します。
3. 抽出を構成して、「適用」をクリックします。

親トピック: 照会名および属性の定義

照会のドリルダウン制御の変更

デフォルトで、レポートのドリルダウン・メニューには、そのレポートの属性で提供できるランタイム・パラメーターを持つすべてのレポートが組み込まれます。ただし、通常のセキュリティー・ロール制約が課されます。

このタスクについて

ドリルダウン制御には、「照会名」リボンの「拡張」オプションからアクセスします。

手順

1. 「照会名」リボンの「拡張」オプションの下にある「ドリルダウン制御」をクリックして、レポートの「ドリルダウン制御」パネルを開きます。
2. 無効にするレポートのチェック・ボックスにマークを付け、有効にするレポートのチェック・ボックスをクリアします。
3. 「適用」をクリックします。変更が正常に適用されたことを示すメッセージが表示されます。
4. 完了したら、「完了」をクリックします。

親トピック: 照会名および属性の定義

API 割り当ての変更

デフォルトで、Guardium アプリケーションには多くの API 関数をレポートにリンクした設定データが添付されています。これによりユーザーには、GUI を通じてレポート・データからの API への作成済み呼び出しが提供されます。「API 割り当て」を使用して、事前定義された Guardium レポートまたはカスタム・レポートへ追加 API 関数をリンクできます。

このタスクについて

リンクされた API 関数の使用に関する詳細については、『GuardAPI 入力の生成』にある資料を参照してください。

手順

1. 「照会名」リボンの「拡張」オプションの下の「API 割り当て」をクリックして、「API 割り当て」パネルを開きます。このパネルには、選択したレポートに現在マップされている API 関数が表示されます。API パラメーターにリンクされたフィールドがレポート内にない場合は、API 関数をレポートにリンクすることが不適切である場合があります。API パラメーターとレポート・フィールドのマッピングは、GUI と CLI の両方で行えます。API パラメーターとレポート・フィールドのマッピングの追加情報については、『GuardAPI 入力の生成』セクションの『GuardAPI パラメーターのドメイン・エンティティーおよび属性へのマッピング』を参照してください。

2. 「API 関数」をクリックし、現在の API とレポートのパラメーター・マッピングを示すポップアップ・ウィンドウを表示します。ここでは API パラメーター、API パラメーターが必須かどうか、デフォルト値、そして、現在パラメーターにマップされたレポートのフィールドがあるかどうかが表示されます。
3. 大なり記号「>」をクリックして、選択した API 関数を、このレポートに割り当てられている現在の関数リストに追加します。
4. 「適用」をクリックして、変更を保存します。

親トピック: 照会名および属性の定義

列表示の選択

ドメイン内のすべてのエンティティーの属性から、レポートの列を選択し、ソート階層を定義します。

このタスクについて

列の数は以下を超えることはできません。

- 30 文字列
- 25 数値列
- 6 テキスト列
- 8 日付列

手順

1. 「表示する列 (Columns to Display)」領域で、レポートに表示する列を選択します。ドロップダウン・リストに、ドメイン内のすべてのエンティティーが示されます。
2. エンティティーをクリックして、その属性のリストを開き、すべてのエンティティーから目的の属性をすべて選択して、「追加」をクリックします。
3. オプションで、レポートで値ごとに 1 行で表示するには、「Distinct」を選択します (レポートで列の値の組み合わせごとに 1 行があります)。このオプションにより、圧縮されたレポートが生成されますが、このオプションはパフォーマンスに影響を与える可能性があります。
4. オプションで、「カウント」を選択します。「カウント」オプションを選択すると、レポートにカウント列 (#) が追加され、1 行での値のセットの出現回数が示されます。このカウントはソートに使用できます。「カウント」を使用する場合、カウントを含めて最大 4 列をレポートに表示できます。
5. 上矢印と下矢印を使用して、レポートに表示したい順序で属性を配置します。UI での上部から下部の順序は、レポートでは左から右の順序になります。
6. 属性ごとに、フィールドについて表示する情報 (その値、カウント (個別の値の数)、最小値、最大値、平均値、または合計値) を選択します。

親トピック: 照会 - レポート・ビルダーの使用

ソート順の設定

このタスクについて

「カウントでソート」オプションは、「選択した列」リボンで「カウント」チェック・ボックスを選択している場合に使用できます。このオプションにより、行は発生頻度でソートされます。

手順

1. 「結果のソート」リボンを開きます。
2. カウントでソートするには、「結果をカウントでソート (Sort results by count)」を選択します。
3. 列でソートするには、次のようにします。
   a. 「結果を列でソート」を選択します。
   b. ⊕ をクリックして、1 次ソート基準にする列を選択します。ドロップダウンの昇順/降順のデフォルトは昇順です。必要に応じて変更します。
   c. 追加の列について上記の手順を繰り返し、「次のソート基準」をクリックして行を追加します。
   d. 列を削除するには、行の ⊖ をクリックします。

親トピック: 照会 - レポート・ビルダーの使用

照会条件の定義

このタスクについて

照会条件の形式は <And/Or> <Field> <Operator> <Value/Parameter/Group> <Value> です。

ここで:

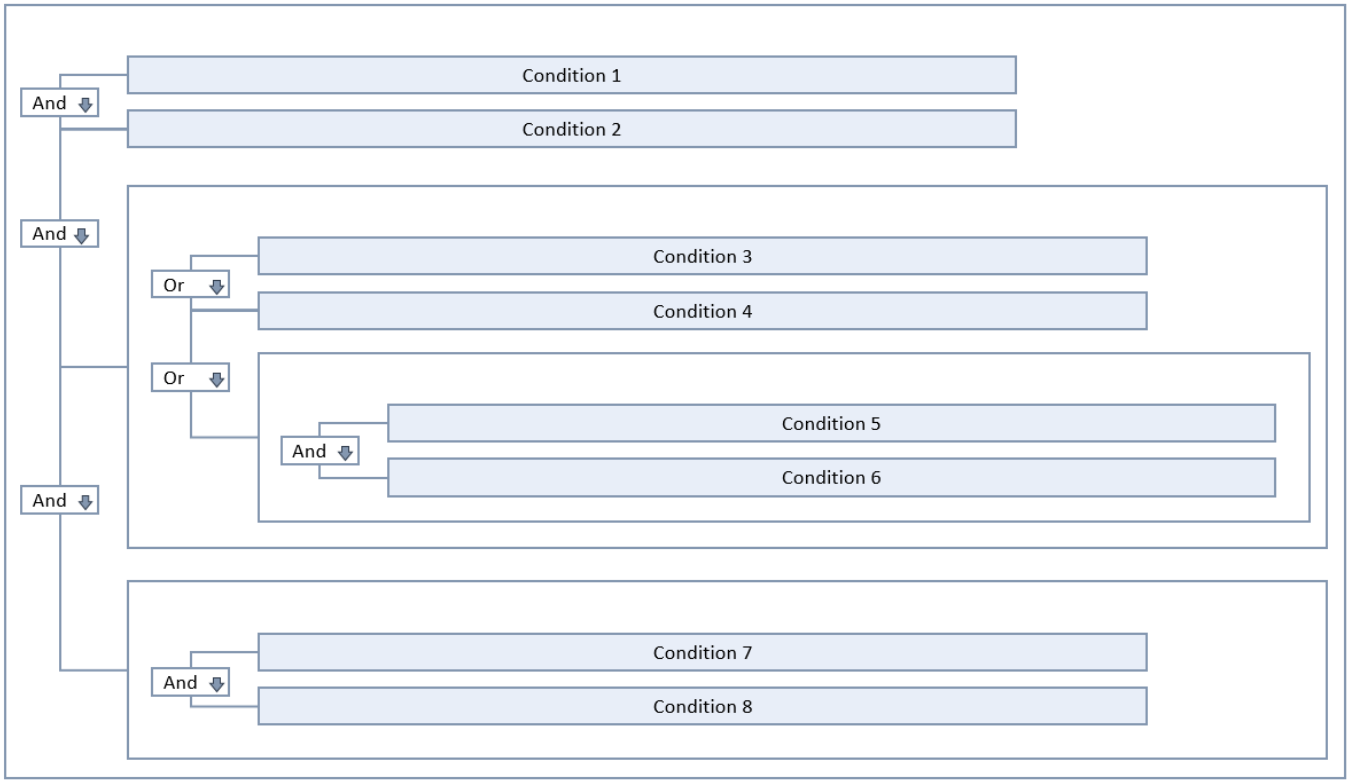
- And/Or: 条件または条件グループの関係を決定します。デフォルトは Add です。
- Field: 照会のドメイン内のいずれかのフィールド。
- Operator: 演算子のタイプは、選択したフィールドに応じて異なります。例えば、グループに関連付けることができない属性では、グループ・オプション (IN GROUP、LIKE GROUP) を指定できません。

演算子	記述
<	より小さい
< =	以下
< >	等しくない
=	等しい

演算子	記述
>	より大きい
> =	以上
CATEGORIZED AS	グループ演算子が選択されている場合に表示される、 ドロップダウン・リストから選択されるカテゴリーに属するグループのメンバー。
CLASSIFIED AS	グループ演算子が選択されている場合に表示される、 ドロップダウン・リストから選択される分類に属するグループのメンバー。
IN ALIASES GROUP	IN GROUP と同じタイプのグループに対して機能するが、 そのグループのメンバーが別名であることを想定する演算子。 IN GROUP 演算子はグループが実際の値を、IN ALIASES GROUP 演算子はグループが別名を含んでいることを予期します。 別名は、特定の属性タイプの保管値に代わる同義語になります。通常は、データ値を意味のある、または分かりやすい名前で表示するために使用されます。 例えば、IP アドレス 192.168.2.18 の別名として、「財務サーバー」を定義することができます。
IN DYNAMIC ALIASES GROUP	IN DYNAMIC GROUP と同じタイプのグループに対して機能するが、 そのグループのメンバーが別名であると想定する演算子。
IN DYNAMIC GROUP	グループ演算子が選択されている場合に表示される、 ランタイム・パラメーター列のドロップダウン・リストから選択されるグループのメンバー。
IN GROUP	グループ演算子が選択されている場合に表示される、 ランタイム・パラメーター列のドロップダウン・リストから選択されるグループのメンバー。IN GROUP または IN ALIASES GROUP は、両方同時に使用することはできません。
IN PERIOD	タイム・スタンプについてのみ、選択された期間内にあります。
IS NOT NULL	属性値は存在しますが、ブランクまたは印刷不能である可能性があります。
IS NULL	空の属性
LIKE	
LIKE GROUP	ボックスに指定された like 値に一致します。like 値は、ワイルドカード文字として % 記号を使用し、値の全部または一部に一致します。 英字には大/小文字の区別がありません。例えば、%tea% は tea、TeA、tEam、steam のいずれにも一致します。 % 記号が含まれていない場合、比較演算は、等価演算 (=) になります。
NOT IN ALIASES GROUP	NOT IN GROUP と同じタイプのグループに対して機能するが、 そのグループのメンバーが別名であることを想定する演算子。
NOT IN DYNAMIC ALIASES GROUP	NOT IN DYNAMIC GROUP と同じタイプのグループに対して機能するが、 そのグループのメンバーが別名であると想定する演算子。
NOT IN DYNAMIC GROUP	グループ演算子が選択されている場合に表示される、 ランタイム・パラメーター列のドロップダウン・リストから選択される、グループのすべてのメンバーと等しくありません。
NOT IN GROUP	グループ演算子が選択されている場合に表示される、 ランタイム・パラメーター列のドロップダウン・リストから選択される、特定のグループのすべてのメンバーと等しくありません。
NOT IN PERIOD	タイム・スタンプ専用。選択された期間内にありません。
NOT LIKE	指定された値と like ではありません (LIKE の説明を参照)。
NOT LIKE GROUP	LIKE GROUP に指定された値と like ではありません。
NOT REGEXP	指定された正規表現に一致しません。
REGEXP	指定された正規表現に一致します。正規表現の使用方法について詳しくは、『正規表現』を参照してください。

- 値/パラメーター/グループ: 演算子に応じて異なります。
  - 値: フィールドが比較される定数。
  - パラメーター: 実行時に値を得るパラメーターの名前。パラメーター名には以下のいずれも使用できません。QUERY_FROM_DATE、QUERY_TO_DATE、REMOTE_SOURCE、SHOW_ALIASES、FETCHSIZE、REFRESHRATE、current_title、action、user、group、role、js_peid、eventsubmit_doupdate、page、_skin、template、media-type。パラメーター名は、文字で始まる必要があり、文字、数字、および下線のみを含めることができます。
  - グループ: フィールドのタイプと一致するグループのドロップダウン・リスト。フィールドと同じタイプのグループが先頭にアルファベット順で表示されます。その後に、すべてのタイプのグループがタイプ別に表示されます。グループ・タイプと各タイプのグループはアルファベット順で表示されます。場合によっては、フィールド・タイプと一致するグループ・タイプがいくつかあることに注意してください。例えば、「クライアント IP」の「クライアント IP/データベース・ユーザー」、「クライアント IP/ソース・アプリケーション/データベース・ユーザー」、「クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名」と一致するグループ・タイプがあります。
- 値: 「演算子」および「値/パラメーター/グループ」に応じて異なります。

照会条件には、個々の条件および条件のグループの両方を含めることができます。条件を追加する場合、または条件をグループに追加する場合、デフォルトの結合は ADD です。これを OR に変更できます。各条件グループは、括弧で囲まれている場合と同様に処理されます。次の図は、2 つの条件および 2 つの条件グループを以下のように指定する照会を示しています。条件 1 **AND** 条件 2 **AND** (条件 3 **OR** 条件 4 **OR** (条件 5 **AND** 条件 6)) **AND** (条件 7 **AND** 条件 8)



手順

1. 「条件」行で「編集」をクリックします。「条件」領域が展開されます。
2. 条件を追加するには、「条件の追加」をクリックして、最初の照会条件の値をドロップダウンで選択します。
3. 条件グループを追加するには、「条件グループの追加 (Add Condition Group)」 をクリックして、グループ内の条件の値をドロップダウンで選択します。
4. 必要に応じて上記の手順を繰り返します。
5. 「保存」をクリックします。

親トピック: 照会 - レポート・ビルダーの使用

照会条件での式の作成の追加

このフィーチャーは、属性の全体的なコンテンツそのものには基づいていないが、その属性の一部、その属性の関数、または複数の属性を組み合わせた関数に基づいた条件 (ユーザー定義文字列および数式を含む) を追加する必要がある場合に使用します。

照会条件が保存されると、照会条件の横に、「式の作成」ペインを開くための「式の追加」アイコン が表示されます。このアイコンは、ユーザー定義文字列および数式を追加するために使用します。

式の作成が定義されている場合は、「式の作成」アイコンの横に赤色のアスタリスク が表示されます。

例:

値 192.150.1.x. から、ストリング「150.1」の位置を返します。ここでストリング「150.1」は、値の 5 番目の文字です。ストリング「150.1」は、リストされている 5 つの文字に一致するクライアント IP のすべてのインスタンスを表します。

「式」フィールドで関数を実行する際、これは値を返し、かつその値は「入力ボックス」に入っていなければなりません。

関数 `INSTR(:attribute, '150.1')` を使用し、「式の追加」アイコンの隣にある入力ボックスに値「5」を入力すると、5 番目の位置に `150.1` があるレコードが返されます。

関数が `INSTR(:attribute, '150.1') = 5` の場合、これはブール句となり、入力ボックスに入力できる値は 0 または 1 のみです。

`INSTR(:attribute, '150.1')` 式を、別の「式の作成」ウィンドウに入力します。

「式の作成」ウィンドウで、式の妥当性をテストします。

もう 1 つの例: `LENGTH(:attribute) >= 40`。これは、40 文字を超えるすべての SQL ステートメントの長さを返します。式には、実際の属性への参照を含める (または含めない) ことが可能です。さらに、他の属性への参照を含めることもできます。

親トピック: 照会 - レポート・ビルダーの使用

HAVING 条件

1 つ以上の列がカウント、最小、最大、平均、合計のいずれかの機能を使用していて、「グループ化基準」が必要な場合、このリボンを使用できます。

HAVING 条件は、追加の集約関数フィールドがある点を除き、他の照会条件と同様に機能します。形式は、<And/Or> <field> <aggregation function > <operator> <value/parameter/group> <value> です。集約関数には以下の値を指定できます。

- カウント
- 最小
- 最大
- 平均

Having 節で使用できる演算子は以下のとおりです。

- より小さい (<)
- 以下 (<=)
- 等しくない (<>)
- 等しい (=)
- より大 (>)
- 以上 (>=)

親トピック: 照会 - レポート・ビルダーの使用

レポート表示の変更

レポート・タイプ (表またはグラフ) を選択して、列名を変更し、色表示ルールを構成することができます。

このタスクについて

図表タイプのレポートは、以下のいずれかが該当する場合にのみ使用できます。

- 「カウント」チェック・ボックスが照会レベルで選択されている。
- 照会のすべての列が数値である。

手順

1. レポート・タイプを選択します。デフォルトは表です。「グラフ」を選択する場合は、タイプのドロップダウン・リストから図表タイプを選択します。
2. 「列見出し」セクションでは、列の名前を属性名から選択する名前に変更できます。
3. 「色表示ルール」セクションで以下を行います。
   a. ルールを作成するには、 をクリックして、色をルールに割り当てます。最初の一致 (上部から下部) が、レポートでの色を決定します。
   b. さらに多くのルールを作成するには、 をクリックして、色をルールに割り当てます。
   c. ルールの順序を変更するには、 および を使用します。

親トピック: 照会 - レポート・ビルダーの使用

ドメイン、エンティティー、および属性

各ドメインでは、特定の目的や機能 (データ・アクセス、例外、ポリシー違反など) に関連するデータ・セットが Guardium に保管されます。データはエンティティーごとにグループ化されます。エンティティーは、関連する属性の集合で、属性は基本的にフィールドの値です。

ドメインへのアクセスは、セキュリティー・ロールによって制御されます。各 Guardium ロールは通常、社内のそのロールの機能に応じて、ドメインのサブセットへのアクセス権限を持ちます。Guardium admin ロール・ユーザーは通常、すべてのレポート作成ドメインへのアクセス権限を持ちます。

ドメインによっては、オプション・コンポーネント (例えば CAS や分類) がインストールされている場合のみ使用可能なものもあります。 その他のドメインは、デフォルトで Guardium admin ロール・ユーザーのみが使用できます。例えば、Guardium アプライアンス (例えばアーカイブ・アクティビティー) に関連する情報をレポートします。

同様に、すべての属性をすべてのデータベース・プロトコルで使用できるわけではありません。 クエリー・ビルダーを使用する際に、この資料に記載されたエンティティーまたは属性が UI に表示されない場合、そのエンティティーまたは属性は、選択したデータベース・タイプには使用できません。

- ドメインのエンティティーおよび属性
  このトピックでは、各ドメインのエンティティーおよび属性について説明します。
- データベース・ライセンス・レポート
  データベース・ライセンス・レポートは、ユーザーが該当するデータのみに対するアクセス権限を持っていることを確認するために使用できます。Guardium システムには、いくつかのデータベース・タイプ用の事前定義のデータベース・ライセンス・レポートが用意されています。

親トピック: レポート

ドメインのエンティティーおよび属性

このトピックでは、各ドメインのエンティティーおよび属性について説明します。

z/OS データ・ソース (Db2、データ・セット、および IMS) には、データ・ソース固有の属性があり、既存の属性の意味が、ここで説明する内容と異なる場合があります。z/OS データ・ソースに固有のエンティティーおよび属性について詳しくは、以下の資料を参照してください。

- データ・セットのレポート・エンティティーおよび属性
- Db2 for z/OS のレポート・エンティティーおよび属性
- IMS のレポート・エンティティーおよび属性

- 「アクセス」ドメイン: エンティティーおよび属性
  このドメインには、モニター対象サーバーに要求が送信されるたびに検査エンジンによって収集されるトラフィック・データが含まれます。クライアント/サーバー、セッション、SQL、およびアクセス期間の関連データのすべてが含まれます。
- 「アクセス・ポリシー」ドメイン: エンティティーおよび属性
  このドメインを使用して、システム上の使用可能なすべてのポリシーをトラッキングします。
- 「統合/アーカイブ」ドメイン: エンティティーおよび属性
  統合およびアーカイブ・アクティビティー。各操作 (アーカイブ、送信、パージなど) の日付、時刻、および状況が含まれます。
- 「アラート」ドメイン: エンティティーおよび属性
  Guardium によって生成および送信されたすべてのアラート。
- 「Analytic 異常値詳細」ドメイン: エンティティーおよび属性
  異常値として識別されたアクティビティーとエラーの詳細な説明
- 「Analytic 異常値の状況」ドメイン: エンティティーおよび属性
  異常値マイニングのプロセスとその結果。
- 「Analytic 異常値サマリー」ドメイン: エンティティーおよび属性
  ソースで直近 1 時間に発生した異常値のサマリー。
- 「アプリケーション・データ」ドメイン: エンティティーおよび属性
  特殊な非 Guardium アプリケーション (例えば Siebel や SAP) について記録された接続、セッション、およびアプリケーション・データ。
- 「監査プロセス」ドメイン: エンティティーおよび属性
  監査プロセスの実行と結果の配布。
- 「オートディスカバリー」ドメイン: エンティティーおよび属性
  データベース・オートディスカバリー・アクティビティー。これには実行されてホストとポートがディスカバーしたすべてのプロセスが含まれます。
- 「BigData Intelligence: バッファー使用状況モニター」ドメイン: エンティティーおよび属性
  すべての「スニファーのバッファー使用」エンティティーの統合を示します。
- 「BigData Intelligence: 分類プロセス・ログ」ドメイン: エンティティーおよび属性
  分類プロセス・ログについてレポートします。
- 「BigData Intelligence: 分類結果」ドメイン: エンティティーおよび属性
  分類プロセスの結果についてレポートします。
- 「BigData Intelligence: ディスカバーされたデータベース」ドメイン: エンティティーおよび属性
  ディスカバーされたデータベースについてレポートします。
- 「BigData Intelligence: ディスカバーされたインスタンス」ドメイン: エンティティーおよび属性
  GIM によってディスカバーされたインスタンスについてレポートします。
- 「BigData Intelligence: 例外」ドメイン: エンティティーおよび属性
  例外と例外関連データのすべて。 Guardium 自体で発生した例外だけでなく、データベース・サーバーから送信されて検査エンジンによって収集された SQL 例外も含まれます。
- 「BigData Intelligence: 完全な SQL」ドメイン: エンティティーおよび属性
  「完全な SQL」エンティティーは、ポリシー・ルール・アクションの「全詳細をロギング」、「値を含む全詳細をロギング」、「セッションごとに全詳細をロギング」、または「値を含む全詳細をセッションごとにロギング」によってのみ作成されます。
- 「BigData Intelligence: インストール済みのパッチ」ドメイン: エンティティーおよび属性
  インストール済みのパッチについてレポートします。
- 「BigData Intelligence: インスタンス」ドメイン: エンティティーおよび属性
  このドメインには、モニター対象サーバーに要求が送信されるたびに検査エンジンによって収集されるトラフィック・データが含まれます。クライアント/サーバー、セッション、SQL、およびアクセス期間の関連データのすべてが含まれます。
- 「BigData Intelligence: 異常値リスト - 拡張」ドメイン: エンティティーおよび属性
  異常値として識別されたアクティビティーとエラーの詳細な説明。
- 「BigData Intelligence: 異常値サマリー - 拡張」ドメイン: エンティティーおよび属性
  1 時間の細分度での異常値のサマリー。
- 「BigData Intelligence: ポリシー違反」ドメイン: エンティティーおよび属性
  Guardium 検査エンジンまたは STAP によって検出されたポリシーのすべての違反に関するすべてのポリシー違反データ。
- 「BigData Intelligence: セッション」ドメイン: エンティティーおよび属性
  クライアント/サーバー・データベース・セッションについてレポートします。
- 「BigData Intelligence: STAP 状況」ドメイン: エンティティーおよび属性
  S-TAP の状況についてレポートします。
- 「BigData Intelligence: システム情報」ドメイン: エンティティーおよび属性
- 「BigData Intelligence: 脆弱性診断結果」ドメイン: エンティティーおよび属性
- 「CAS 変更」ドメイン: エンティティーおよび属性
  モニター項目 (ファイル、レジストリー変数など) に対する変更をトラッキングします。.
- 「CAS 構成」ドメイン: エンティティーおよび属性
  CAS ホスト構成をトラッキングします。 ここで、構成とは、特定のデータベース・サーバー・ホストに 対する、1 つ以上のテンプレート・セットのアプリケーションです。 構成インスタンスから、 テンプレート・セット内で使用可能または使用不可になっている項目や、 ファイル名パターン・テンプレートによって選択されモニターされている (またはされていない) 正確なファイルを 確認することができます。
- 「CAS ホスト履歴」ドメイン: エンティティーおよび属性
  CAS ホスト・イベント (サーバーまたはクライアントのサービス開始やサービス休止など) を追跡します。
- 「CAS テンプレート」ドメイン: エンティティーおよび属性
  CAS テンプレート定義をトラッキングします。 テンプレートは、変更をモニターされる項目を識別します。 モニター項目は、ファイル、環境またはレジストリー変数、OS または SQL スクリプト出力セット、 またはログオン・ユーザー・セットのいずれであってもかまいません。
- 「カタログ」ドメイン: エンティティーおよび属性
- 「分類プロセスの結果」ドメイン: エンティティーおよび属性
  分類プロセスの実行と結果についてレポートします。
- 「CM バッファー使用状況モニター」ドメイン: エンティティーおよび属性
  中央マネージャーにアップロードされたすべての「スニファーのバッファー使用」エンティティーの統合を示します。
- 「コメント」ドメイン: エンティティーおよび属性
  各種 Guardium コンポーネントに関するユーザー定義のコメント。
- 「カスタム・データベース使用状況」ドメイン: エンティティーおよび属性
  カスタム・データベース統計

- 「有効になっているデータベース・デフォルト・ユーザー」ドメイン: エンティティーおよび属性
  デフォルト・ユーザーが有効になっているかどうかの詳細。
- 「ディスカバーされたインスタンス」ドメイン: エンティティーおよび属性
  GIM によってディスカバーされたインスタンス。
- 「分散データマート」ドメイン: エンティティーおよび属性
- 「Eagle Eye」ドメイン: エンティティーおよび属性
- 「例外」ドメイン: エンティティーおよび属性
  このドメインには、トラフィックの詳細 (例外と例外関連データのすべて) が含まれます。 Guardium 自体で発生した例外だけでなく、データベース・サーバーから送信されて検査エンジンによって収集された SQL 例外も含まれます。
- 「FAM」ドメイン: エンティティーおよび属性
  ディスカバーされたファイルに関するメタデータ。
- 「FAM システム」ドメイン: エンティティーおよび属性
- 「ファイル・アクティビティー・モニター」ドメイン: エンティティーおよび属性
- 「未解析ログ」ドメイン: エンティティーおよび属性
  未解析ログ処理アクティビティー。
- 「GIM クライアント」ドメイン: エンティティーおよび属性
- 「GIM イベント」ドメイン: エンティティーおよび属性
- 「グループ」ドメイン: エンティティーおよび属性
  Guardium グループのメンバーシップ。
- 「保護プロセス・ログ」ドメイン: エンティティーおよび属性
  Guardium で実行されているプロセスのログ。
- 「Guardium アクティビティー」ドメイン: エンティティーおよび属性
  Guardium エンティティーに対して Guardium ユーザーが行ったすべての変更 (レポートまたは照会の定義または変更)。
- 「Guardium ジョブ・キュー」ドメイン: エンティティーおよび属性
- 「Guardium ログイン」ドメイン: エンティティーおよび属性
  Guardium ユーザーのログインとログアウトに関する全情報。
- 「IMS イベント」ドメイン: エンティティーおよび属性
- 「インストール済みポリシー」ドメイン: エンティティーおよび属性
  インストール済みポリシーのポリシー・パラメーターとポリシー・ルールの記述。「インストール済みポリシー」ドメインは、複数のポリシーと、ルール 1 つ当たり複数のアクションをサポートします。
- 「パーサー・エラー」ドメイン: エンティティーおよび属性
- 「ポリシー違反」ドメイン: エンティティーおよび属性
  Guardium 検査エンジンまたは STAP によって検出されたポリシーのすべての違反に関するすべてのポリシー違反データ。
- 「ポリシー違反サマリー」ドメイン: エンティティーおよび属性
  Guardium 検査エンジンまたは STAP によって検出されたポリシーのすべての違反のサマリーに関するすべてのポリシー違反データ。
- 「照会再書き込み」ドメイン: エンティティーおよび属性
- 「S-TAP 状況」ドメイン: エンティティーおよび属性
- 「S-TAP 状況履歴」ドメイン: エンティティーおよび属性
- 「S-TAP 検査」ドメイン: エンティティーおよび属性
- 「S-TAP/Z ファイル」ドメイン: エンティティーおよび属性
- 「セキュリティー・アセスメントの結果」ドメイン: エンティティーおよび属性
  脆弱性評価プロセスの結果を記録します。
- 「スニファーのバッファー使用のモニター」ドメイン: エンティティーおよび属性
  検査エンジン統計。
- 「S-TAP の統計」ドメイン: エンティティーおよび属性
- 「ユニット使用状況レベル」ドメイン
- 「ユーザー/ロール/アプリケーション」ドメイン: エンティティーおよび属性
  Guardium ユーザー、ロール、およびアプリケーションを関連付けます (それにより、誰がどの Guardium アプリケーションへのアクセス権限を持っているかをレポートします)。
- 「VA サマリー」ドメイン: エンティティーおよび属性
- 「脆弱性評価テスト」ドメイン: エンティティーおよび属性
  セキュリティー・アセスメントに使用可能なテストについてレポートします。
- 「値の変更」ドメイン: エンティティーおよび属性
  トリガー・ベースの値変更アプリケーションによってトラッキングされたすべての変更。

親トピック: ドメイン、エンティティー、および属性

「アクセス」ドメイン: エンティティーおよび属性

このドメインには、モニター対象サーバーに要求が送信されるたびに検査エンジンによって収集されるトラフィック・データが含まれます。クライアント/サーバー、セッション、SQL、およびアクセス期間の関連データのすべてが含まれます。

使用可能なロール: すべて

「クライアント/サーバー」エンティティー

このエンティティーは、特定のクライアント/サーバー接続について示します。固有の属性セット (タイム・スタンプを除く) が検出されるたびにインスタンスが作成されます。

注: 「アクセスのトラッキング」の場合のみ、「クライアント/サーバー」エンティティー名は、プルダウン・メニューに 2 つの可能なエンティティー (「クライアント/サーバー」および「セッション別クライアント/サーバー」) として表示されます。

「セッション別クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「セッション」から日付状態を取得します。

「クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「クライアント/サーバー」から日付状態も取得します。

ユーザーが「クライアント/サーバー」を選択すると、照会には ATTRIBUTE_ID = 1 が設定されます。ユーザーが「セッション別クライアント/サーバー」を選択すると、照会には MAIN_ATTRIBUTE_ID = 0 が設定されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
分析されたクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
クライアント・ホスト名	クライアント・ホスト名。
クライアント IP	クライアントの IP アドレス。
クライアント IP/データベース・ユーザー	クライアント IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名	タプル・グループ
クライアント IP/ソース・アプリケーション/ユーザー	タプル・グループ
クライアント MAC	クライアントのハードウェア・アドレス。
クライアント OS	クライアントのオペレーティング・システム。 Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
データベース・プロトコル	データベース・サーバー固有のプロトコル。
データベース・プロトコル・バージョン	データベース・プロトコルのプロトコル・バージョン。
データベース・ユーザー名	データベース・ユーザー名: ローカルまたはリモートのデータベースに接続したユーザー。
最終使用日	
ネットワーク・プロトコル	使用されたネットワーク・プロトコル (例えば TCP、UDP など。Oracle 上の K-TAP については IPC または BEQ として表示されることに注意してください)。
OS ユーザー	相互作用の OS ユーザー・アカウント。
サーバーの記述	サーバーの記述 (ある場合)。
サーバー・ホスト名	サーバー・ホスト名。
サーバー IP	サーバーの IP アドレス。
サーバー IP/データベース・ユーザー	サーバー IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
サーバー IP/サービス名/データベース・ユーザー	タプル・グループ

属性	記述
サーバー OS	サーバーのオペレーティング・システム。 Informix の場合、OS が次のように表示される場合があります。 IEEEM (Unix または JDBC を表します) IEEEI (Windows を表します) DEC (DEC Alpha を表します) Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
サーバー・タイプ	DB2、Oracle、Sybase など。
サービス名	相互作用のサービス名。場合によっては (例えば AIX 共有メモリー接続)、別名がサービス名になり、実際のサービスが接続されるまで別名が使用されます。このような場合、実際のサービスが接続された時点で、新しいセッションが開始されます。したがって、ユーザーには単一セッションと認識されるものが 2 つのセッションとしてログに記録されることになります。 Teradata の場合、サービス名にはセッション論理ホスト ID 値が含まれます。
ソース・プログラム	相互作用のソース・プログラム。
セッション別クライアント/サーバー	セッション別クライアント/サーバーは、メイン・エンティティーでもあります。 この「2 次エンティティー」にアクセスするには、「1 次エンティティー」である「クライアント/サーバー」をクリックします。
タイム・スタンプ	このエンティティーのすべての属性が静的情報を持つので、このタイム・スタンプは、定義済みクライアント/サーバー接続上で Guardium が要求を初めて確認したときに、1 回だけ作成されます。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。

「セッション」エンティティー

このエンティティーは、クライアント/サーバー・データベース・セッションごとに作成されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
クライアント・ポート	クライアント・ポート番号。
データベース名	セッションの対象データベースの名前 (MSSQL または Sybase のみ)。 Oracle の場合、アプリケーション固有の追加情報が「データベース名」に含まれることがあります (V$SESSION ビューの MODULE 列に設定された、セッション用に現在実行中のモジュールなど)。
期間 (秒)	セッション開始からセッション終了までの時間の長さを示します (秒単位)。
グローバル ID	セッション・アクセスを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
これ以降を無視	このセッションを無視し始めたときに作成されたタイム・スタンプ。
非アクティブ・フラグ	0 (デフォルト) - オープン (SQL パッケージによって生成されたセッションの場合)。 1 - クローズ (切断/ログアウト受信)。 2 - おそらくクローズ (長時間にわたってパケットがない状態では非クローズ)。 3 - 非 SQL パケットから生成されたセッションの場合。
古いセッション ID	このセッションが作成されたセッションを示します。 これが接続の最初のセッションである場合は、ゼロです。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
プロセス ID	接続を開始したクライアントのプロセス ID (常に使用可能とは限らない)。
サーバー・ポート	サーバー・ポート番号。

属性	記述
セッション終了	セッションが終了した日時。「セッション終了」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション終了の日付	「セッション終了」の中の日付のみ。
セッション終了の時刻	「セッション終了」の中の時刻のみ。
セッション終了の曜日	「セッション終了」の中の曜日のみ。
セッション終了の年	「セッション終了」の中の年のみ。
セッション ID	セッションを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
セッション無視	セッションの一部が (ある時点から) 無視されたかどうかを示します。
セッション開始	セッションが開始された日時。「セッション開始」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション開始の日付	「セッション開始」の中の日付のみ。
セッション開始の時刻	「セッション開始」の中の時刻のみ。
セッション開始の曜日	「セッション開始」の中の曜日のみ。
セッション開始の年	「セッション開始」の中の年のみ。
端末 ID	セッション情報を解決するために内部で使用された、接続の端末 ID。
タイム・スタンプ	初めは、進行中のアクティブ・セッションのないクライアント/サーバー接続上の最初の要求に対して、タイム・スタンプが作成されます。 その後、セッションが閉じられたとき、または長期間にわたってアクティビティーが監視されないために非アクティブのマークがセッションに付けられたときに、更新されます。 セッション情報をトラッキングする場合は、「タイム・スタンプ」属性よりも「セッション開始」属性と「セッション終了」属性を注目する方が適切な場合があります。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
TTL	admin ロール専用に予約済み。
Uid チェーン	Unix S-TAP (K-Tap モードのみ) によってレポートされたセッションが対象。su ユーザーのユーザー名が異なる場合に、OS ユーザーのチェーンを示します。 ここに表示される値は、OS プラットフォームによって異なります。例えば AIX 環境では、接頭部として IBM IBM IBM という文字列が表示される場合があります。 注: Solaris ゾーンの場合は、ユーザー名の代わりにユーザー ID が「UID チェーン」でレポートされる場合があります。
Uid チェーン圧縮	圧縮された値。「Uid」チェーンを参照してください。

「クライアント/サーバー・セッション」エンティティー

属性	記述
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名/OS ユーザー/データベース名	
サーバー IP/サーバー・ポート	

「アクセス期間」エンティティー

アクセス期間はセッションに関連します。デフォルトではアクセス期間の長さは 1 時間ですが、Guardium 管理者は「検査エンジン構成」で変更することができます (「ロギング単位」に該当)。

属性	記述
アプリケーション・イベント ID	アプリケーション・イベント ID (API から設定された場合)。照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。
実行確認応答時間の平均	実行確認応答時間の平均 (ミリ秒単位)。
影響を受けるレコードの平均 (2)	影響を受けたレコードの平均数。照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。
アプリケーション・ユーザー	アプリケーション・ユーザー名。
平均実行時間	期間中の平均コマンド実行時間。SQL ステートメントのみが対象です。FTP トラフィックには適用されません。
構造 ID	コマンド構造 (例えば select a from b) を一意的に識別します。admin ロールを持つユーザーのみが使用できます。
失敗した SQL (2)	失敗した SQL 要求の数。表の最後に記載されている注を参照してください。 照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。

属性	記述
インスタンス ID	構造のインスタンスを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
元のタイム・ゾーン	UTC オフセット。 これは、2 つの異なるタイム・ゾーンに存在する 2 つの異なるコレクターの時間を、正しく統合するために設定する必要がある UTC オフセットを示すものです。 オフセットを設定しなかった場合、物事の発生時刻をユーザーが判別したり、正確な表記で参照したりできないという状況が存在していまいます。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
期間の終了	アクセス期間の終了の日時。
期間の終了日	期間終了属性の中の日付のみ。
期間の終了時刻	期間終了属性の中の時刻のみ。
期間の終了曜日	期間終了属性の中の曜日のみ。
期間の開始日	期間開始属性の中の日付のみ。
期間の開始時刻	期間開始属性の中の時刻のみ。
期間の開始曜日	期間開始属性の中の曜日のみ。
セッション ID	セッションを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
秒を表示	1 秒当たりのアクセス数がトラッキングされている場合、アクセス期間内 (通常は 1 時間) の秒ごとのカウントがここに含まれています。
成功した SQL (2)	成功した SQL 要求の数。表の最後に記載されている注を参照してください。 照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。
タイム・スタンプ	初めは、アクセス期間中にクライアント/サーバー接続上で要求が初めて確認されたときに、タイム・スタンプ値が設定されます。 デフォルトではアクセス期間の長さは 1 時間ですが、Guardium 管理者は「検査エンジン構成」で変更することができます (「Guardium 管理者ガイド」を参照)。その後は、後続の要求ごとに、その期間の平均実行時間とコマンド数が更新されるときに更新されます。
アクセス合計	このアクセス期間における構造インスタンスの総数。admin ロールを持つユーザーのみが使用できます。
影響を受けるレコード合計 (2)	影響を受けたレコードの総数。照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。
影響を受けるレコード合計 (名前) (2)	「影響を受けるレコード合計」属性が数値ではなく文字列の場合、その値はここに表示されます (例えば、「大規模結果セット」や N/A)。照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。 「影響されるレコード」- SQL ステートメントを実行するたびに影響を受けるレコードの数を示す結果セット。 注: 「影響されるレコード」オプションは、スニファーに対して、追加の応答パケットを処理し、影響を受けたデータ (バッファー・サイズを増やし、スニファー全体のパフォーマンスに悪影響を及ぼす可能性があるデータ) のロギングを延期するように要求するスニファー操作です。 大きな影響を受けるのは、非常に大きい応答からです。この操作に関連する大量のオーバーヘッドを避けるために、Guardium はデフォルトのしきい値セットを使用して、しきい値を超えたときに、処理操作をスキップすることをスニファーが決定できるようにします。 CLI コマンド store max_results_set_size、store max_result_set_packet_size、および store max_tds_response_packets を使用して、細分度のレベルを設定することができます。 結果セットの値の例は次のとおりです。 • ケース 1、「影響されるレコード」値: 正数 - これは、結果セットの正しいサイズを表します。 • ケース 2、「影響されるレコード」値: -2 - これは、レコード数が構成可能な限度 (CLI インターフェースによって調整可能) を超えたことを示します。 • ケース 3、「影響されるレコード」値: -1 - これは、Guardium によってサポートされないパケット構成のケースを示します。 • ケース 4、「影響されるレコード」値: -2 - 結果セットがストリーム・モードで送信される場合。 • ケース 5、「影響されるレコード」値: -2 - ユーザーを現在の値について更新するためのレコードのカウント中の中間結果。最終的には、レコードの合計を示す正数になります。

「変更後のデータ値」エンティティー

このエンティティーは、IBM InfoSphere Change Data Capture (InfoSphere CDC) レプリケーション・ソリューションとともに使用されます。このソリューションを使用すると、サポートされるデータベースとの相互レプリケーションを行うことができます。レプリケーションされたデータベースを保守することにより、処理のオーバーヘッドとネットワーク・トラフィックを低減することができます。

Database Activities Monitoring を使用する IBM Guardium ユーザーは、InfoSphere CDC にアクセスできます。

この Guardium 機能は、Java CDC ユーザー出口を使用して、値の変更情報を Guardium コレクターに送信します。

InfoSphere CDC のユーザー出口を使用すると、ユーザーは、指定された表に対してデータベース・イベントが発生する前または後に InfoSphere CDC で実行可能な一連のアクションを定義することができます。

属性	記述
完全な SQL ID	完全な SQL の固有 ID。
表名	データベースの表名。
列名	データベースの列名。
古い値	変更前の値。
新しい値	変更後の値。
タイム・スタンプ	レコードが作成された時刻。

データベース・サーバーにインストールする必要がある 2 つのファイルは、IBM の InfoSphere Change Data Capture (InfoSphere CDC) アプリケーションとのインターフェースを取る Guardium エージェント用のファイルです。これらのファイルは、ビルドの sources/apps/GuardCDC/lib/ ディレクトリー内にあります。これらのファイルは、protobuf-java-2.4.1.jar と GuardCdc.jar です。

インストールの手順

前提条件 - InfoSphere Change Data Capture (InfoSphere CDC) アプリケーションが、データベース・サーバーに既にインストールされている必要があります。

データベース・サーバーに Guardium エージェントをインストールする手順:

1. これら 2 つのファイルを cdchome ディレクトリーの RepEngine/lib/ ディレクトリーにコピーします。絶対パスは、例えば /cdchome/cdc6.5.2/RepEngine/lib/ のようになります。
2. 各ファイルを unzip します。
3. guard_cdc_user_exit_config.mxl ファイルを編集して、Guardium_Host 名を追加します。このファイルが置かれる場所は、例えば /cdchome/cdc6.5.2/RepEngine/lib/com/guardium/cdc/userexit/ のようになります。
4. GuardiumAgent に書き込みを行うように InfoSphere CDC を構成します。CDC アプリケーションのセットアップと構成には、いくつものステップがあります。これらのステップは、IBM の InfoSphere CDC 開発/サポート・チームから入手できます。

「アプリケーション・ユーザー名」エンティティー

このエンティティーは、アプリケーション・イベントが存在する場合に、アプリケーション・イベントから取得したユーザー名を表示します。 存在しない場合、構成体インスタンスから取得したユーザー名が表示されます。

属性	記述
アプリケーション・ユーザー名	この「アプリケーション・ユーザー名」エンティティーの固有 ID。

「完全な SQL 値」エンティティー

これらのエンティティーは、「値を含む全詳細をロギング」および「値を含む全詳細をセッションごとにロギング」ポリシー・ルール・アクションによってのみ作成されます。

属性	記述
値	ログに記録された構造に含まれる 1 つ以上の値。
タイム・スタンプ	「完全な SQL 値」エンティティーが作成された日時。

「完全な SQL」エンティティー

「完全な SQL」エンティティーは、ポリシー・ルール・アクションの「全詳細をロギング」、「値を含む全詳細をロギング」、「セッションごとに全詳細をロギング」、または「値を含む全詳細をセッションごとにロギング」によってのみ作成されます。

属性	記述
アクセス・ルールの記述	使用されたポリシー・ルールの記述。
確認応答時間	確認応答時間 (ミリ秒単位)。
自動コミット	項目が自動的に番号付けされます。
バインド変数値	Db2/zOS の場合は、バインド変数のコンマ区切りリストが含まれます。
退出 KB カウント	応答に含まれるバイト数を記録します。
完全な SQL	値を含む完全な SQL ステートメント。
完全な SQL ID	完全な SQL の固有 ID。admin ロールを持つユーザーのみが使用できます。
進入 KB カウント	要求に含まれるバイト数を記録します。
インスタンス ID	完全な SQL のインスタンスの固有 ID。admin ロールを持つユーザーのみが使用できます。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。

属性	記述
影響を受けるレコード	影響を受けたレコードの数 (セッションごと)。この属性を使用するレポートでは、大規模結果セットや N/A などの特殊ケースが適切に表示されるように、別名をオンにすることをお勧めします。
影響を受けるレコード (名前)	「影響されるレコード」が数値ではなく文字列値である場合、その文字列はここに保管されます。例: 大規模結果セットまたは N/A。
応答時間	要求に対する応答時間 (ミリ秒単位)。ネットワーク・トラフィック内で要求がモニターされる場合、応答時間は要求に応答するのに要した時間を正確に反映しています (Guardium はクライアント要求とサーバー応答の両方のタイム・スタンプを設定します)。
戻りデータ	この要求に対して返されたデータ (ある場合、かつ使用可能な場合)。
戻りデータ・カウント	ポリシー・ルールで使用された SQL ステートメントから返された行数。
ステートメント・タイプ	SQL ステートメントのタイプ。 SQL: 単純な直接 SQL コマンド (例えば、CLI に直接入力されるコマンド) RAW: 後で実行するための SQL ステートメント PREPARE。例えば、conn.prepareStatement (select a from b where c=:value) BIND: バインドされたパラメーター値を含む、準備されたステートメントの実行 ステートメント・タイプは「完全な SQL」エンティティーの一部であり、ポリシー内でこのステートメントに対して「全詳細をロギング」を構成した場合にのみ監査されます。 ポリシー内の特定のステートメント・タイプ (例えば、監査のみの SQL および BIND ステートメント) をフィルタリングすることはできません。ただし、レポートではこれらをフィルタリングできます。
成功	呼び出しが成功したかどうかを示します。admin ロールを持つユーザーのみが使用できます。
タイム・スタンプ	タイム・スタンプは、SQL がデータベース・サーバーで実行されるときに、時刻を記録します。

「アプリケーション・イベント」エンティティー

このエンティティーは、アプリケーション・イベント API 呼び出し (これらの属性値を設定する) またはカスタム識別プロシージャーと識別されたストアード・プロシージャー呼び出し (ストアード・プロシージャー・パラメーターをこれらの属性にマップする) をシステムが確認するたびに、作成されます。

属性	記述
アプリケーション・イベント ID	この「アプリケーション・イベント」エンティティーの固有 ID。admin ロールを持つユーザーのみが使用できます。
イベントの日付	日時値 (GuardAppEvent:Start で設定)。yyyy-mm-dd hh:mm:ss 形式で表示されます。 注: yyyy-mm-dd 以外の形式を使用してイベント日付を設定すると、内容はすべてゼロになります。時刻部分 (hh:mm:ss) はオプションであり、省略した場合は 00:00:00 になります。
イベント・リリース日付	日時値 (GuardAppEvent: Released で設定)。yyyy-mm-dd hh:mm:ss 形式で表示されます。
イベント・リリース・タイプ	イベントのタイプ (GuardAppEvent: Released で設定)。
イベント・リリース・ユーザー名	ユーザー名 (GuardAppEvent: Released で設定)。
イベント・リリース値 (数値)	数値 (GuardAppEvent: Released で設定)。
イベント・リリース値 (文字列)	文字列値 (GuardAppEvent: Released で設定)。
イベント・タイプ	イベントのタイプ (GuardAppEvent:Start で設定)。
イベント・ユーザー名	ユーザー名 (GuardAppEvent:Start で設定)。
イベント値 (文字列)	文字列値 (GuardAppEvent:Start で設定)。
イベント値 (数値)	数値 (GuardAppEvent:Start で設定)。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
タイム・スタンプ	イベントがログに記録されるときに 1 回だけ作成されます。この属性と「イベントの日付」属性を混同しないでください。「イベントの日付」は、API 呼び出しを使用するかストアード・プロシージャー・パラメーターから設定できる属性です。 (アプリケーション・イベント API の説明は、API によるユーザーの識別を参照してください。)

「SQL」エンティティー

このエンティティーは、SQL の固有文字列ごとに作成されます。値は疑問符 (?) に置き換えられ、文字列のフォーマットのみが保管されます。

属性	記述
バインド情報	この SQL 文字列のバインド情報。
構造 ID	SQL が出現する構造を一意的に識別します。
SQL	SQL 文字列。

属性	記述
切り捨てられた SQL	SQL が切り捨てられたかどうかを示します。値は次のとおりです。 0 - false/いいえ (切り捨てられていません) 1 - true/はい (切り捨てられました)

「コマンド」エンティティー

各コマンドの親ノードとコマンド構造内でそのコマンドが現れる位置ごとに、エンティティーが作成されます。

属性	記述
コマンド ID	コマンドを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
構造 ID	構造 (例えば select a from b) を一意的に識別します。admin ロールを持つユーザーのみが使用できます。
深さ	SQL 構文解析ツリーにおけるコマンドの深さ。
親	構文解析ツリーにおける親ノードの ID。
SQL 動詞	SQL コマンド内の主動詞 (例えば select、insert、delete など)。

「オブジェクト・コマンド」エンティティー

オブジェクト・コマンド・エンティティーについて示します。

属性	記述
オブジェクト/コマンド	コマンド値と結合されたオブジェクト値。

「オブジェクト」エンティティー

このエンティティーのインスタンスは、固有スキーマ内のオブジェクトごとに作成されます。

属性	記述
アプリケーション・オブジェクト・モジュール 1	アプリケーション・オブジェクト・モジュールを一意的に識別します。
構造 ID	オブジェクトが参照される構造を一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
オブジェクト ID	オブジェクトを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
オブジェクト名	オブジェクトの名前。
スキーマ	オブジェクトのデータベース・スキーマ。 注: この属性にはデータが取り込まれることが決してないため、推奨されません。

「結合」エンティティー

結合表は、多対多の関係を実装するための 1 つの方法です。結合エンティティーは、SELECT SQL ステートメントで表を結合する場合に使用します。

属性	記述
構造 ID	結合が参照される構造を識別します。
結合 ID	ユニーク ID
Join SQL	結合表
タイム・スタンプ	「結合」エンティティーが作成された日時。
Where SQL	Where 節 (結合条件)

「フィールド SQL 値」エンティティー

これらのエンティティーは、値も一緒にログに記録するポリシー・ルール・アクション (例えば、「値を含む全詳細をロギング」、「値を含む全詳細をセッションごとにロギング」) によってのみ作成されます。ログに記録されたフィールド値は、フィールド名と関連付けられる場合と、そうでない場合があります。 例えば、次のステートメントがログに記録された場合、フィールド名が使用可能です (「フィールド」エンティティー内)。

insert into t1 (foo, bar) (10, 20)

しかし、次のステートメントがログに記録された場合は、使用不可です。

insert into t2 (10, 20)

属性	記述
値	ログに記録された構造に含まれるフィールド値。

「オブジェクト・フィールド」エンティティー

オブジェクト・フィールド・エンティティーについて示します。オブジェクトが指定されていないフィールドは、オブジェクトを含んだレポートには示されないことに注意してください。

属性	記述
オブジェクト/フィールド	フィールド値と結合されたオブジェクト値。

「フィールド」エンティティー

Guardium は、新規フィールドを検出するたびに、フィールド・エンティティーを作成します。

属性	記述
コマンド ID	参照された構造に含まれるメイン・コマンドを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
構造 ID	参照された構造を一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
フィールド ID	フィールドを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
フィールド名	フィールドの名前。
List 節 Where 節 Order by 節 Having 節 Group By 節 On 節	これらの属性は、複合 SQL 照会を順序付けするのに使用します。 SQL 照会の例: Order by SELECT * FROM dept_costs WHERE dept_total > (SELECT avg FROM avg_cost) ORDER BY department Having SELECT column_name1, SUM(column_name2) FROM table_name GROUP BY column_name1 HAVING (数値関数条件) Group By SELECT column_name1, SUM(column_name2) FROM table_name GROUP BY column_name1 Where SELECT FirstName, LastName, City FROM Users WHERE City = Los Angeles
オブジェクト ID	参照された構造に含まれるオブジェクトを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。

「修飾オブジェクト」エンティティー

タプル では、複数の属性を組み合わせて 1 つのグループ・メンバーを形成することができます。この場合、「サーバー IP」、「サービス名」、「データベース名」、「データベース・ユーザー」、および「オブジェクト」の各フィールドがまとめて結合されます。

属性	記述
修飾されたオブジェクト	タプル - サーバー IP、サービス名、データベース名、データベース・ユーザー、オブジェクト

親トピック: ドメインのエンティティーおよび属性

「アクセス・ポリシー」ドメイン: エンティティーおよび属性

このドメインを使用して、システム上の使用可能なすべてのポリシーをトラッキングします。

使用可能なロール: すべて

「アクセス・ポリシー」エンティティー

このエンティティーは、システム上にインストールされているすべてのポリシーに対して使用される「インストール済みポリシー」エンティティーに類似しています。

属性	記述
ポリシー ID	アクセス・ポリシーを一意的に識別します。
ポリシーの記述	アクセス・ポリシーを記述します。
選択的な監査証跡	これが選択的な監査証跡ポリシーであるかどうかを示します (T/F)。
監査パターン	選択的な監査証跡ポリシーに使用されたテスト・パターン。
タイム・スタンプ	レコード作成のタイム・スタンプ。

「ルール」エンティティー

このエンティティーは、「インストール済みポリシー・ルール」エンティティーまたは「アクセス・ポリシー・ルール」エンティティーに対して使用することができます。1 つ以上のインストール済みポリシーまたは 1 つ以上のアクセス・ポリシーのルールごとに、このエンティティーが 1 つ存在します。ID フィールド (内部データベース上のコンポーネントを一意的に識別する) は別として、これらのフィールドについてはすべて、『ポリシー』ヘルプ・トピックに記載されています。

- GDM_INSTALLED_POLICY_RULES_ID - インストール済みポリシー・ルールを識別します。
- ACCESS_RULE_ID - アクセス・ルールを識別します。
- ルールの記述 - ポリシー定義に含まれる。
- ルール位置 - ポリシー内の位置。
- ルール・タイプ - アクセス、例外、または抽出。
- LAST_ACCESSED - 前回のアクセス
- クライアント IP - ルール定義に含まれる。
- クライアント・ネットマスク - ルール定義に含まれる。
- クライアント IP グループ - ルール定義に含まれる。
- サーバー IP - ルール定義に含まれる。
- サーバー IP マスク (Server IP Mask) - ルール定義に含まれる。
- クライアント MAC - ルール定義に含まれる。
- ネット・プロトコル - ルール定義に含まれる。
- ネット・プロトコル・グループ - ルール定義に含まれる。
- フィールド - ルール定義に含まれる。
- フィールド・グループ (Field Group) - ルール定義に含まれる。
- オブジェクト - ルール定義に含まれる。
- オブジェクト・グループ - ルール定義に含まれる。
- コマンド - ルール定義に含まれる。
- コマンド・グループ - ルール定義に含まれる。
- オブジェクト・フィールド・グループ - ルール定義に含まれる。
- データベース・タイプ - ルール定義に含まれる。
- サービス名 - ルール定義に含まれる。
- サービス名グループ (Service Name Group) - ルール定義に含まれる。
- データベース名 - ルール定義に含まれる。
- データベース名グループ - ルール定義に含まれる。
- データベース・ユーザー - ルール定義に含まれる。
- データベース・ユーザー・グループ - ルール定義に含まれる。
- アプリケーション・ユーザー - ルール定義に含まれる。
- アプリケーション・ユーザー・グループ (App User Group) - ルール定義に含まれる。
- OS ユーザー - ルール定義に含まれる。
- OS ユーザー・グループ - ルール定義に含まれる。
- ソース・アプリケーション - ルール定義に含まれる。
- ソース・プログラム・グループ (Source Program Group) - ルール定義に含まれる。
- パターン / XML パターン - ルール定義に含まれる。
- 期間 - ルール定義に含まれる。
- 最小 数 - ルール定義に含まれる。
- リセット間隔 - ルール定義に含まれる。
- 次のルールに進む / 取り消し - ルール定義に含まれる。
- 値を記録 - ルール定義に含まれる。
- アプリケーション・イベントの存在 - ルール定義に含まれる。
- イベント・タイプ - ルール定義に含まれる。
- アプリケーション・イベント・テキスト値 - ルール定義に含まれる。
- アプリケーション・イベント日付値 (App Event Date Value) - ルール定義に含まれる。
- イベント・ユーザー名 - ルール定義に含まれる。
- エラー・コード - ルール定義に含まれる。
- 例外タイプ - ルール定義に含まれる。
- カテゴリー名 - ルール定義に含まれる。
- 分類名 - ルール定義に含まれる。
- 重大度 - ルール定義に含まれる。
- データ・パターン - ルール定義に含まれる。
- SQL パターン - ルール定義に含まれる。
- マスキング・パターン - ルール定義に含まれる。
- クライアント IP / グループ - 単一の属性とその関連情報 (ある場合) を、レポートの単一列に表示できます。
- サーバー IP / グループ - 単一の属性とその関連情報 (ある場合) を、レポートの単一列に表示できます。
- ネットワーク・プロトコル / グループ - 単一の属性とその関連情報 (ある場合) を、レポートの単一列に表示できます。
- フィールド名 / グループ - 単一の属性とその関連情報 (ある場合) を、レポートの単一列に表示できます。
- オブジェクト名 / グループ - 単一の属性とその関連情報 (ある場合) を、レポートの単一列に表示できます。

- コマンド / グループ - 単一の属性とその関連情報 (ある場合) を、レポートの単一列に表示できます。
- サービス名 / グループ - 単一の属性とその関連情報 (ある場合) を、レポートの単一列に表示できます。
- データベース名 / グループ - 単一の属性とその関連情報 (ある場合) を、レポートの単一列に表示できます。
- アプリケーション・ユーザー / グループ - 単一の属性とその関連情報 (ある場合) を、レポートの単一列に表示できます。
- OS ユーザー / グループ - 単一の属性とその関連情報 (ある場合) を、レポートの単一列に表示できます。
- ソース・プログラム / グループ - 単一の属性とその関連情報 (ある場合) を、レポートの単一列に表示できます。
- エラー・コード / グループ - 単一の属性とその関連情報 (ある場合) を、レポートの単一列に表示できます。
- アプリケーション・イベント・テキスト / 数値 / 日付 - アプリケーション・イベントのテキスト属性、数値属性、および日付属性。
- カテゴリー / 分類 - ルールのカテゴリーと分類の結合。
- GDM_Installed_Policy_Header_ID - インストール済みポリシーのヘッダーを識別します。

注: GDM_INSTALLED_POLICY_RULES_ID および ACCESS_RULE_ID は、admin ロールを持つユーザーのみが使用できます。

「ルール・アクション」エンティティー

このエンティティーは、「インストール済みポリシー・ルール・アクション」エンティティーまたは「アクセス・ポリシー・ルール・アクション」エンティティーに対して使用することができます。1 つ以上のインストール済みポリシーまたは 1 つ以上のアクセス・ポリシーのルールごとに、このエンティティーが 1 つ存在します。

- シーケンス - ルール内のアクションのシーケンス。
- アクション
  - 要求のブロック -『ポリシー』に記載された『ブロッキング・アクション』を参照してください。
  - 違反またはトラフィックをロギングまたは無視 -『ポリシー』に記載された『ロギング・アクションまたは無視アクション』を参照してください。
  - アラート -『ポリシー』に記載された『アラート・アクション』を参照してください。

「アラート通知」エンティティー

ポリシー・アラート通知について示します。

属性	記述
ALERT_NOTIFICATION_ID	アラート通知を識別します。admin ロールを持つユーザーのみが使用できます。
ALERT_ID	アラート定義を識別します。admin ロールを持つユーザーのみが使用できます。
アラート通知タイプ	ポリシー・ルール定義に含まれるアラートのタイプ。
アラート・ユーザー	アラートの受信者。
アラート宛先	アラートのタイプ (EMAIL、SNMP、SYSLOG、CUSTM)。
タイム・スタンプ	作成されたタイム・スタンプ・アラート・レコード。

親トピック: ドメインのエンティティーおよび属性

「統合/アーカイブ」ドメイン: エンティティーおよび属性

統合およびアーカイブ・アクティビティー。各操作 (アーカイブ、送信、パージなど) の日付、時刻、および状況が含まれます。

使用可能なロール: admin

「アクティビティー・タイプ」エンティティー

「統合/アーカイブ」ドメインからのみ使用可能。「統合/アーカイブ」ドメインは、デフォルトでは、admin ロールが割り当てられたユーザーのみが使用できます。「アクティビティー・タイプ」エンティティーは、所有する「統合/インポート/エクスポート・ログ」エンティティーからのみアクセス可能です。このエンティティーは、アクションのタイプ (統合の準備、暗号化、送信など) を示します。 .

属性	記述
アクティビティー・タイプ	統合/インポート/エクスポート・アクティビティーの記述。

「統合/アーカイブ・ログ」エンティティー

「統合/アーカイブ」ドメインからのみ使用可能。「統合/アーカイブ」ドメインは、デフォルトでは、admin ロールが割り当てられたユーザーのみが使用できます。 アクティビティーごとに「統合/インポート/エクスポート・ログ」エンティティーが 1 つ以上作成されます。 例えば、アグリゲーター・システムがデータをインポートするときに、通常は少なくとも次の 4 つのアクティビティーが表示されます。

統合の準備

重複インポートの検査 (ファイル当たり 1 つをこのアグリゲーターにエクスポート)

抽出 (ファイル当たり 1 つがマージの対象)

マージ (ファイル当たり 1 つをマージ)

属性	記述
コメント	アクティビティーに関する追加コメント。
終了時刻	アクティビティーの終了時刻。

属性	記述
ファイル名	アクティビティーに使用されたファイルの名前。アーカイブおよびエクスポート操作によって作成されるファイルには、次のような名前が付けられます。 <daysequence>-<scp_host>-w<run_datestamp>-d<data_date>.dbdump.enc 例: 732423-g1.guardium.com-w20050425.040042-d2005-04-22.dbdump.enc ファイルに含まれるデータの日付は、ファイル名の終わり近く (.dbdump.enc の直前) にある data_date (形式は yyyy-mm-dd) です。 この日付と実行日を混同しないように気を付けてください。実行日はファイル名の中の前の方にあり、データがアーカイブまたはエクスポートされた日付を示します。
Guardium ホスト名	Guardium ホストの名前。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
期間の終了	対象データの終了時刻。
期間の開始	対象データの開始時刻。 アーカイブまたは統合の各アクティビティーは、丸 1 日のアクティビティーを対象とします。
パージされたレコード	アクティビティー・タイプが「パージ」の場合は、パージされたレコード数。 それ以外の場合は「N/A」。
状況	統合/インポート/エクスポート・ログ・アクティビティーの状況。
開始時刻	アクティビティーの開始時刻。
タイム・スタンプ	ログに記録されるアクティビティー (アーカイブの準備、暗号化、送信など) の開始時と終了時に更新されます。 .
ユーザー名	アクティビティーの開始に使用されたユーザー名。

「統合/アーカイブ・デバッグ・ログ」エンティティー

属性	記述
統合日	
統合期間	
統合例外	
統合ステージ	
統合状況	
統合表名	
詳細	
ジョブ・グループ	
削除されたレコード	
挿入されたレコード	
更新されたレコード	
実行ログ ID	
タイム・スタンプ	

親トピック: ドメインのエンティティーおよび属性

「アラート」ドメイン: エンティティーおよび属性

Guardium によって生成および送信されたすべてのアラート。

使用可能なロール: すべて

「アクティビティー・タイプ」エンティティー

「統合/アーカイブ」ドメインからのみ使用可能。「統合/アーカイブ」ドメインは、デフォルトでは、admin ロールが割り当てられたユーザーのみが使用できます。「アクティビティー・タイプ」エンティティーは、所有する「統合/インポート/エクスポート・ログ」エンティティーからのみアクセス可能です。このエンティティーは、アクションのタイプ (統合の準備、暗号化、送信など) を示します。 .

属性	記述
アクティビティー・タイプ	統合/インポート/エクスポート・アクティビティーの記述。

「しきい値アラート詳細」エンティティー

このエンティティーは、相関アラートが発生するたびに作成されます。

属性	記述
アラート・ログ ID	アラート詳細エンティティーを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
照会値	照会によって返された値。
基本値	統計アラートに割り当てられた値。
検査日の始まり	アラート条件による検査対象期間の開始日時。
検査日の終わり	アラート条件による検査対象期間の終了日時。
アラートしきい値	アラートに定義されたアラートしきい値。
通知の送信	送信された通知のテキスト。
タイム・スタンプ	統計アラートがログに記録されるときに 1 回だけ作成されます。
アラートの記述	アラート定義に含まれる記述。

「メッセージ・テキスト」エンティティー

しきい値アラートのメッセージのテキスト。

属性	記述
メッセージ・テキスト ID	メッセージ・テキストを一意的に識別します。
メッセージ件名	メッセージの件名 (例えば E メール・メッセージの件名)。
メッセージ・テキスト	メッセージ・テキスト。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。

「送信メッセージ」エンティティー

送信されたしきい値アラート・メッセージごとの、メッセージのメッセージ・タイプ、受信者、状況、および日付。

属性	記述
メッセージ ID	メッセージを一意的に識別します。
メッセージ・タイプ	メッセージのタイプ。
送信先	1 人以上のメッセージ受信者。
メッセージ状況	メッセージの状況。以下の状況があります。 失敗 送信操作は失敗しました。 待機 メッセージはまだ送信されていません。 送信済み メッセージは送信されました。
メッセージの日付	メッセージが送信された日付。
メッセージ・コンテキスト	メッセージ・タイプ。以下のタイプがあります。 情報 情報メッセージ。 警告 エラー状態の可能性あり。 アラート リアルタイム・アラートまたはしきい値アラート。 エラー ソフトウェアまたはハードウェアのエラー状態。 デバッグ デバッグ・メッセージ。
メッセージ発信元	メッセージを作成するモジュール (例えば、モニターまたは GuardiumJetspeedUser)。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。

親トピック: ドメインのエンティティーおよび属性

「Analytic 異常値詳細」ドメイン: エンティティーおよび属性

異常値として識別されたアクティビティーとエラーの詳細な説明

使用可能なロール: admin

「Analytic 異常値詳細」エンティティー

属性	記述
異常スコア	最後の異常スコア
データベース・ユーザー名	アクティビティーを実行したデータベース・ユーザー。
エラー異常値	異常値のタイプがエラーであるかどうか。True/False。
大量の異常値	異常値のタイプが大量であるかどうか。True/False。
新規異常値	異常値のタイプが新規であるかどうか。True/False。
インスタンスの数	アクティビティー・ボリューム。
オブジェクト	ユーザーがアクティビティーを実行した対象のオブジェクト。
オブジェクト/動詞	アクティビティーで使用されたオブジェクト/動詞の組み合わせ。
異常値タイプ	推奨されません
期間の開始	アクティビティーが発生した期間の開始時刻。
サーバー IP	アクティビティーが発生したサーバーの IP。
サーバー・タイプ	アクティビティーが発生したサーバーのタイプ。
サービス名	アクティビティーで使用されたサービスの名前。
ソース・プログラム	アクティビティーが発生したソース・プログラム。
一時異常値	異常値のタイプが一時であるかどうか。True/False。
タイム・スタンプ	アクティビティーのタイム・スタンプ。
動詞	アクティビティーで使用された動詞。

親トピック: ドメインのエンティティーおよび属性

「Analytic 異常値の状況」ドメイン: エンティティーおよび属性

異常値マイニングのプロセスとその結果。

使用可能なロール: admin

「Analytic 状況」エンティティー

属性	記述
詳細	現在の状況に関連する詳細
分析終了時刻	分析が完了した時刻。
分析したデータベースの数	分析が実行されたデータベースの数
異常値の数	期間中に検出された異常値の数
分析したレコードの数	期間中に分析された、統合後の行数。
分析したユーザーの数	分析に含まれていたユーザーの数。
分析期間	期間の開始時刻。
分析開始時刻	コレクターからすべてのデータを受信した後で分析が開始された時刻。
状況	E: エラー (詳細にリストされたエラー)、D: 完了 (正常に完了)、P: 保留 (コレクターからのデータを待機中)、R: 実行中、W: 警告 (ブロッカーではない)
タイム・スタンプ	行が最後に更新された時刻

親トピック: ドメインのエンティティーおよび属性

「Analytic 異常値サマリー」ドメイン: エンティティーおよび属性

ソースで直近 1 時間に発生した異常値のサマリー。

使用可能なロール: admin

「Analytic 異常値サマリー」エンティティー

属性	記述
アラート・フィードバック ID	このフィードバック・アラートの原因となったルールの ID。
異常スコア	このアクティビティーの最後の異常スコア。
データベース・ユーザー名	アクティビティーを実行したデータベース・ユーザー。
さまざまな異常値	異常値のタイプがさまざまであるかどうか。True/False。
エラー異常値	異常値のタイプがエラーであるかどうか。True/False。
大量の異常値	異常値のタイプが大量であるかどうか。True/False。
新規メッセージの平均数	異常値の原因となったエンティティーによる新規メッセージの平均数。
新規メッセージのスコア	新規アクティビティー異常の測定。
新規メッセージの SD	推奨されません
新規異常値	異常値のタイプが新規異常値であるかどうか。True/False。
失敗の数	失敗したアクティビティーの数。
新規メッセージの数	新規アクティビティーの新しいタイプの数。
機密オブジェクトの数	この間隔で接触された機密オブジェクトの数。
一時オブジェクトの数	この間隔で使用された一時オブジェクトの数。
一時ソース・プログラムの数	推奨されません
現行の異常値	異常値のタイプが現行であるかどうか。True/False。
元のホスト名	クライアント・ホスト名。
異常値サマリー ID	固有 ID。
期間の開始	期間開始の日付と時刻。
特権ユーザー	アクティビティーが特権ユーザーによって実行されたかどうか。 True/False。
希少性およびボリュームのスコア	推奨されません
サーバー IP	アクティビティーが発生したサーバーの IP。
サーバー・タイプ	DAM または FAM。
サービス名	アクティビティーで使用されたサービスの名前。
ソース ID	アクティビティーが発生したソースの ID。
一時異常値	異常値のタイプが一時であるかどうか。True/False。
一時オブジェクトの平均数	最近数時間の上記の統計の平均。
一時オブジェクトのスコア	一時オブジェクトの異常な使用の測定。
一時オブジェクトの SD	最近数時間からこの間隔で使用された一時オブジェクトの数の標準偏差。
一時ソース・プログラムのスコア	推奨されません
タイム・スタンプ	アクティビティーのタイム・スタンプ。
一時ソース・プログラムのタイプ	推奨されません
タイプのボリュームの希少性	推奨されません

親トピック: ドメインのエンティティーおよび属性

「アプリケーション・データ」ドメイン: エンティティーおよび属性

特殊な非 Guardium アプリケーション (例えば Siebel や SAP) について記録された接続、セッション、およびアプリケーション・データ。

使用可能なロール: admin

「クライアント/サーバー」エンティティー

このエンティティーは、特定のクライアント/サーバー接続について示します。固有の属性セット (タイム・スタンプを除く) が検出されるたびにインスタンスが作成されます。

注: 「アクセスのトラッキング」の場合のみ、「クライアント/サーバー」エンティティー名は、プルダウン・メニューに 2 つの可能なエンティティー (「クライアント/サーバー」および「セッション別クライアント/サーバー」) として表示されます。

「セッション別クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「セッション」から日付状態を取得します。

「クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「クライアント/サーバー」から日付状態も取得します。

ユーザーが「クライアント/サーバー」を選択すると、照会には ATTRIBUTE_ID = 1 が設定されます。ユーザーが「セッション別クライアント/サーバー」を選択すると、照会には MAIN_ATTRIBUTE_ID = 0 が設定されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。

属性	記述
分析されたクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
クライアント・ホスト名	クライアント・ホスト名。
クライアント IP	クライアントの IP アドレス。
クライアント IP/データベース・ユーザー	クライアント IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名	タプル・グループ
クライアント IP/ソース・アプリケーション/ユーザー	タプル・グループ
クライアント MAC	クライアントのハードウェア・アドレス。
クライアント OS	クライアントのオペレーティング・システム。 Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
データベース・プロトコル	データベース・サーバー固有のプロトコル。
データベース・プロトコル・バージョン	データベース・プロトコルのプロトコル・バージョン。
データベース・ユーザー名	データベース・ユーザー名: ローカルまたはリモートのデータベースに接続したユーザー。
最終使用日	
ネットワーク・プロトコル	使用されたネットワーク・プロトコル (例えば TCP、UDP など。Oracle 上の K-TAP については IPC または BEQ として表示されることに注意してください)。
OS ユーザー	相互作用の OS ユーザー・アカウント。
サーバーの記述	サーバーの記述 (ある場合)。
サーバー・ホスト名	サーバー・ホスト名。
サーバー IP	サーバーの IP アドレス。
サーバー IP/データベース・ユーザー	サーバー IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
サーバー IP/サービス名/データベース・ユーザー	タプル・グループ
サーバー OS	サーバーのオペレーティング・システム。 Informix の場合、OS が次のように表示される場合があります。 IEEEM (Unix または JDBC を表します) IEEEI (Windows を表します) DEC (DEC Alpha を表します) Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
サーバー・タイプ	DB2、Oracle、Sybase など。

属性	記述
サービス名	相互作用のサービス名。場合によっては (例えば AIX 共有メモリー接続)、別名がサービス名になり、実際のサービスが接続されるまで別名が使用されます。このような場合、実際のサービスが接続された時点で、新しいセッションが開始されます。したがって、ユーザーには単一セッションと認識されるものが 2 つのセッションとしてログに記録されることになります。 Teradata の場合、サービス名にはセッション論理ホスト ID 値が含まれます。
ソース・プログラム	相互作用のソース・プログラム。
セッション別クライアント/サーバー	セッション別クライアント/サーバーは、メイン・エンティティーでもあります。 この「2 次エンティティー」にアクセスするには、「1 次エンティティー」である「クライアント/サーバー」をクリックします。
タイム・スタンプ	このエンティティーのすべての属性が静的情報を持つので、このタイム・スタンプは、定義済みクライアント/サーバー接続上で Guardium が要求を初めて確認したときに、1 回だけ作成されます。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。

「セッション」エンティティー

このエンティティーは、クライアント/サーバー・データベース・セッションごとに作成されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
クライアント・ポート	クライアント・ポート番号。
データベース名	セッションの対象データベースの名前 (MSSQL または Sybase のみ)。 Oracle の場合、アプリケーション固有の追加情報が「データベース名」に含まれることがあります (V$SESSION ビューの MODULE 列に設定された、セッション用に現在実行中のモジュールなど)。
期間 (秒)	セッション開始からセッション終了までの時間の長さを示します (秒単位)。
グローバル ID	セッション・アクセスを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
これ以降を無視	このセッションを無視し始めたときに作成されたタイム・スタンプ。
非アクティブ・フラグ	0 (デフォルト) - オープン (SQL パッケージによって生成されたセッションの場合)。 1 - クローズ (切断/ログアウト受信)。 2 - おそらくクローズ (長時間にわたってパケットがない状態では非クローズ)。 3 - 非 SQL パケットから生成されたセッションの場合。
古いセッション ID	このセッションが作成されたセッションを示します。 これが接続の最初のセッションである場合は、ゼロです。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
プロセス ID	接続を開始したクライアントのプロセス ID (常に使用可能とは限らない)。
サーバー・ポート	サーバー・ポート番号。
セッション終了	セッションが終了した日時。「セッション終了」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション終了の日付	「セッション終了」の中の日付のみ。
セッション終了の時刻	「セッション終了」の中の時刻のみ。
セッション終了の曜日	「セッション終了」の中の曜日のみ。
セッション終了の年	「セッション終了」の中の年のみ。
セッション ID	セッションを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
セッション無視	セッションの一部が (ある時点から) 無視されたかどうかを示します。
セッション開始	セッションが開始された日時。「セッション開始」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション開始の日付	「セッション開始」の中の日付のみ。
セッション開始の時刻	「セッション開始」の中の時刻のみ。
セッション開始の曜日	「セッション開始」の中の曜日のみ。
セッション開始の年	「セッション開始」の中の年のみ。

属性	記述
端末 ID	セッション情報を解決するために内部で使用された、接続の端末 ID。
タイム・スタンプ	初めは、進行中のアクティブ・セッションのないクライアント/サーバー接続上の最初の要求に対して、タイム・スタンプが作成されます。 その後、セッションが閉じられたとき、または長期間にわたってアクティビティーが監視されないために非アクティブのマークがセッションに付けられたときに、更新されます。 セッション情報をトラッキングする場合は、「タイム・スタンプ」属性よりも「セッション開始」属性と「セッション終了」属性を注目する方が適切な場合があります。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
TTL	admin ロール専用に予約済み。
Uid チェーン	Unix S-TAP (K-Tap モードのみ) によってレポートされたセッションが対象。su ユーザーのユーザー名が異なる場合に、OS ユーザーのチェーンを示します。 ここに表示される値は、OS プラットフォームによって異なります。例えば AIX 環境では、接頭部として IBM IBM IBM という文字列が表示される場合があります。 注: Solaris ゾーンの場合は、ユーザー名の代わりにユーザー ID が「UID チェーン」でレポートされる場合があります。
Uid チェーン圧縮	圧縮された値。「Uid」チェーンを参照してください。

「アプリケーション・データ」エンティティー

SAP および Siebel レポートに使用されます。

属性	記述
アプリケーション・データ ID	このデータの固有 ID。
アプリケーション・コード	アプリケーション・タイプ・コード。
完全な SQL ID	完全な SQL データを識別します。
アプリケーション・タイプ	アプリケーション・タイプ。
ユーザー	アプリケーション・ユーザー名。
操作タイプ	操作のタイプ。
変更日	変更の日付。
タイム・スタンプ	このレコードのタイム・スタンプ。
項目名	影響を受けた項目の名前。
トランザクション・コード	トランザクション・コード。
システム ID	システムの固有 ID。
レコード詳細 1	項目タイプによって異なります。
レコード詳細 2	項目タイプによって異なります。
レコード詳細 3	項目タイプによって異なります。
レコード詳細 4	項目タイプによって異なります。
VBKey	VBKey 値。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。

親トピック: ドメインのエンティティーおよび属性

「監査プロセス」ドメイン: エンティティーおよび属性

監査プロセスの実行と結果の配布。

使用可能なロール: すべて

「監査プロセス」エンティティー

このエンティティーには、監査プロセスの基本定義パラメーターが含まれます。

属性	記述
アクティブ	プロセスがアクティブかどうか (スケジュール設定できるかどうか) を示します。

属性	記述
結果の保持 (日数)	結果が保持される日数。
結果の保持 (数量)	保持される結果セットの数。
プロセスの記述	監査プロセス定義に含まれる記述。

「監査プロセス・コメント」エンティティー

このエンティティーには、監査プロセス定義に付加されたコメントが含まれます。監査プロセスの結果に付加されたコメントは、「監査プロセスの結果コメント」エンティティーに含まれます。

属性	記述
監査プロセス・コメント	コメントのテキスト。
監査プロセス・コメントの作成者	コメントの作成者。
監査プロセス・コメントのタイム・スタンプ	コメントのタイム・スタンプ。

「監査タスク」エンティティー

このエンティティーは、単一の監査タスク (監査プロセス内) について示します。

属性	記述
タスクの記述	タスク定義に含まれるタスクの名前。
タスク・タイプ	数値は、タスクがレポート、セキュリティー・アセスメント、エンティティー監査証跡、プライバシー・セット、または分類プロセスのいずれであるかを示します。これらのタイプには別名が定義されるので、レポートに別名が使用され、読みやすいレポート出力になります。

「監査プロセスの結果」エンティティー

このエンティティーには、監査プロセスの結果セットの実行日が含まれます。

属性	記述
実行日	監査プロセスが実行された日付。

「タスク受信者」エンティティー

結果の受信者が必要とするアクションを示します。

属性	記述
必要なアクション	署名アクションを必要とするかどうかを示します。

「タスク結果 To Do リスト」エンティティー

結果の現在の状況を示します。

属性	記述
必要なアクション	署名アクションを必要とするかどうかを示します。
(エスカレーション) 必要なアクション	To Do リスト・アクションを必要とするかどうかを示します。
状況	結果の現在の状況を示します。

「ユーザー」エンティティー

監査プロセスの結果の受信者として定義された Guardium ユーザーを識別します。

属性	記述
E メール・アドレス	Guardium ユーザーに定義された E メール・アドレス。
ファーストネーム (名)	Guardium ユーザーのファーストネーム (名)。
最終アクティブ	このユーザーの最終アクティビティーのタイム・スタンプ。
ラストネーム (姓)	Guardium ユーザーのラストネーム (姓)。
ログイン名	Guardium ユーザー名。

「監査プロセスの結果コメント」エンティティー

このエンティティーには、監査プロセスの結果に付加されたコメントが含まれます。監査プロセス定義に付加されたコメントは、「監査プロセス・コメント」エンティティーに含まれます。

属性	記述
監査プロセス・コメント	コメントのテキスト。
監査プロセス・コメントの作成者	コメントの作成者。
監査プロセス・コメントのタイム・スタンプ	コメントのタイム・スタンプ。

親トピック: ドメインのエンティティーおよび属性

「オートディスカバリー」ドメイン: エンティティーおよび属性

データベース・オートディスカバリー・アクティビティー。これには実行されてホストとポートがディスカバーしたすべてのプロセスが含まれます。

使用可能なロール: すべて

「オートディスカバリー・スキャン」エンティティー

このエンティティーは、いつスキャンが実行されたかを識別します。

属性	記述
スキャン・タイム・スタンプ	スキャンが実行された時刻。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。

「ディスカバーされたホスト」エンティティー

このエンティティーは、ディスカバーされたホストを識別します。

属性	記述
サーバー IP	ディスカバーされたホストの IP アドレス。
サーバー・ホスト名	ディスカバーされたホストのホスト名。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。

「ディスカバーされたポート」エンティティー

このエンティティーは、ディスカバーされたポートを識別します。

属性	記述
ポート	ディスカバーされたポート番号。
プローブ試行	サポートされるデータベース・サービスのプローブがこのポートで試行されたかどうかを示します。 T = はい、F = いいえ。
ポート・タイプ	ポート・タイプを示します (通常は TCP)。
データベース・タイプ	サポートされるデータベース・タイプがポートのプローブで検出された場合に、タイプ (DB2®、Informix®、MS SQL Server など) を示します。
プローブ・タイム・スタンプ	この特定のポートでプローブが行われた日時。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: バッファー使用状況モニター」ドメイン: エンティティーおよび属性

すべての「スニファーのバッファー使用」エンティティーの統合を示します。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: バッファー使用状況モニター」エンティティー

属性	記述
Mysql による CPU 使用時間 %	MySQL によって使用された CPU のパーセンテージ。
スニファーの CPU 使用時間 %	スニファーによって使用された CPU のパーセンテージ。
Mysql によるメモリー使用 %	MySQL によって使用されたメモリーのパーセンテージ。
スニファーによるメモリー使用 %	スニファーによって使用されたメモリーのパーセンテージ
ALP	アナライザー逸失パケット
アナライザー・キューの長さ	分析キューのサイズ。
アナライザー・レート	メッセージが分析される速度。
データベース・オープン FD	データベース・オープン・ファイル記述子。
Eth0 受信	ETH 0 で受信されたメッセージ。
Eth0 送信	ETH 0 で送信されたメッセージ。
追加の情報	内部スニッフィング・エンジン・データ。照会では通常は使用されません。
未解析ログ要求	未解析ログ要求
空きバッファー・スペース	空きバッファー・スペース量。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
ハンドラー・データ	内部スニッフィング・エンジン・データ。
モニターされるロガー・データベース	現在モニターされているデータベース・タイプのリスト。
ルールにより無視されるロガー・パケット	ポリシー・ルール・アクションにより無視されたパケット。
ロガー・キューの長さ	ロガー・キューのサイズ。
ロガー・レート	メッセージがログに記録される速度。
ロガー・セッション・カウント	ログに記録されたセッションの数。
メモリー・スニファー	スニファーによって使用されたメモリー量。
Mysql ディスク使用状況	MySQL ディスク使用状況。
Mysql 起動済み	内部データベース再始動を表すブール値インディケーター (1 = 再始動済み、0 = 未再始動)。
オープン FD	オープン・ファイル記述子。
プロミスキャス受信	スニッフィング・ネットワーク・カード (非インターフェース・ポート) を介した受信パケットの率。
セッション直接クローズ	直接閉じられたセッションの数。
セッション推測	推測されたセッション数。
セッション無視	スニファーによって無視されたセッションの数。
セッション・キューの長さ	セッション・キューのサイズ。
セッション・タイムアウト	タイムアウトになったセッションの数。
セッション総計	セッションの総数。
通常のセッション	通常のセッションの数。
終了したスニファー接続	検査エンジンの再始動以降、モニターされて終了した接続の総数。
使用されたスニファー接続	検査エンジンの再始動以降、現在モニターされている接続の総数。
無視されたスニファー・パケット	スニファーによって無視されたパケット。
スロットルされたスニファー・パケット	検査エンジンの再始動以降、スロットルのために無視された接続の総数。
SNO	オープンされていないセッション
SPD	ドロップされたスニファー・パケット
システム CPU 負荷	システム CPU 使用状況。
システム・メモリー使用状況	システム・メモリー使用状況。
システム・ルート・ディスク使用状況	システム・ルート・ディスク使用状況。
システム・アップタイム	最後の始動からの時間。
システム変数ディスク使用状況	/var ディスク使用状況。

属性	記述
TID	スニファー・プロセスの PID
時間スニファー	スニファーによって使用された経過時間。
タイム・スタンプ	アクティビティーのタイム・スタンプ。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: 分類プロセス・ログ」ドメイン: エンティティーおよび属性

分類プロセス・ログについてレポートします。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: 分類プロセス・ログ」エンティティー

属性	記述
データ・ソース	ジョブのデータ・ソース・リスト。
詳細	ログ・イベントに関する情報: エラー・メッセージまたは統計。
終了日	ジョブ終了の日付。
終了日時	ジョブ終了時のタイム・スタンプ。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
Guardium ジョブの記述	グローバル・プロセスの記述のローカル・コピー。
メッセージ	ログ・イベントに関する情報: エラー・メッセージまたは統計。
メッセージ・タイプ	メッセージのタイプ。
プロセス ID	接続を開始したクライアントのプロセス ID (常に使用可能とは限らない)。
プロセス実行 ID	分類プロセス実行 ID。
プロセス・タイプ	分類プロセス・タイプ
キュー日付	ジョブが分類/評価キューに実行依頼されたときのタイム・スタンプの日付。
キュー日時	ジョブが分類/評価キューに実行依頼されたときのタイム・スタンプ。
レポート結果 ID	レポート結果を識別します。
開始日	ジョブ開始のタイム・スタンプの日付。
開始日時	ジョブ開始時のタイム・スタンプ。
状況	ジョブの状況。
タスクの記述	タスク定義に含まれるタスクの名前。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差。

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: 分類結果」ドメイン: エンティティーおよび属性

分類プロセスの結果についてレポートします。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: 分類結果」エンティティー

記述	記述
カタログ	結果セットのカタログのロケーション。
カテゴリー	ルールのカテゴリー。
分類名	ルールの分類。
列名	ルール定義に含まれる列名。
コメント	このルール定義に追加されたコメント。
データ・ソースの記述	ルールのデータ・ソース。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
プロセスの記述	グローバル・プロセスの記述のローカル・コピー

記述	記述
ルールの記述	分類ポリシー・ルールの記述。
スキーマ	スキーマ名 (該当する場合)。
開始日	ジョブ開始のタイム・スタンプの日付。
開始時間	ジョブ開始時のタイム・スタンプ。
表名	ルール定義に含まれる表名。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: ディスカバーされたデータベース」ドメイン: エンティティーおよび属性

ディスカバーされたデータベースについてレポートします。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「ディスカバーされたデータベース」エンティティー

属性	記述
データベース・タイプ	ディスカバーされたデータベースのタイプ。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
ポート	データベースのディスカバー時に使用されていたポート。
ポート・タイプ	ポート・タイプを示します。
プローブ・タイム・スタンプ	この特定のポートでプローブが行われた日時。
プローブのタイム・スタンプの日付	プローブのタイム・スタンプの日付。
サーバー・ホスト名	ディスカバーされたホストのホスト名。
サーバー IP	ディスカバーされたホストの IP アドレス。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: ディスカバーされたインスタンス」ドメイン: エンティティーおよび属性

GIM によってディスカバーされたインスタンスについてレポートします。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData: ディスカバーされたインスタンス」エンティティー

属性	記述
クライアント	クライアントの IP アドレス/マスク。
データベース・インストール・ディレクトリー	データベース・インストール・ディレクトリー。
Db2 共有メモリー調整	パケット・ヘッダー・サイズ
Db2 共有メモリー・クライアント位置	クライアント入出力域オフセット
Db2 共有メモリー・サイズ	Db2 共有メモリー・セグメント・サイズ
除外するクライアント	除外するクライアントの IP アドレス/マスク。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
ホスト	このインスタンスのホスト名。
Informix バージョン	Informix バージョン
インスタンス名	ディスカバーされたインスタンスの名前。
KTAP データベース・ポート	KTAP のデータベース・ポート。
名前付きパイプ	データベースによって使用されたパイプ名。
ポート (最大)	ポート範囲 (検査エンジンの最大ポート番号)。
ポート (最小)	ポート範囲 (検査エンジンの最小ポート番号)。
プロセス名	プロセス名

属性	記述
プロセス名	データベース実行可能プログラムの名前。
プロトコル	このインスタンスに固有のプロトコル。
タイム・スタンプ	Guardium がエンティティーのこのインスタンスを記録したときに作成されたタイム・スタンプ。
タイム・スタンプの日付	タイム・スタンプの日付。
Unix ソケット	Unix ソケット
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: 例外」ドメイン: エンティティーおよび属性

例外と例外関連データのすべて。 Guardium 自体で発生した例外だけでなく、データベース・サーバーから送信されて検査エンジンによって収集された SQL 例外も含まれます。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData: 例外」エンティティー

属性	記述
分析済みのクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
アプリケーション・ユーザー名	アプリケーション・ユーザー名。
クライアント・ホスト名	クライアント・ホスト名
データベース・エラー・テキスト	データベース・エラー・コードの後に、エラーに関する短いテキストの記述が続きます。 エラー・コードは、「例外」エンティティーの「例外の記述」属性から取得されます。 エラー・コードをキーとして使用して、Guardium アプライアンス上の内部表からエラー・テキストが取得されます。この内部表には、最も一般的なエラー・メッセージ (エラー・メッセージのうち約 54,000) が含まれます。 例: ORA-00942: 表またはビューが存在しません
データベース名	データベース名
データベース・プロトコル	例外のデータベース・プロトコル。
データベース・ユーザー名	ローカルまたはリモートのデータベースに接続したデータベース・ユーザー。
記述	データベース例外の場合、これはデータベース管理システムからのエラー・コードです。 最も一般的なメッセージ (メッセージのうち約 54,000) については、「データベース・エラー・テキスト」属性で、より長いテキストの記述を使用できます。 そのテキストは、例外自体からではなく、エラー・メッセージのための内部 Guardium データベース表からのものです。
宛先アドレス	宛先 IP アドレス。
エラー・コード	データベース・エラー・コード。
例外の日付	そのタイム・スタンプの日付のみ。
例外 ID	例外を一意的に識別します。
例外タイム・スタンプ	この「例外」エンティティーがログに記録された日時。
例外タイプ	例外タイプを一意的に識別します。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
例外についての追加情報へのリンク	例外ソースによっては使用可能な場合があるリンク。
OS ユーザー	相互作用の OS ユーザー・アカウント
サーバー・ホスト名	サーバー・ホスト名
サーバー IP	サーバーの IP アドレス
サーバー・ポート	サーバー・ポート番号
サーバー・タイプ	DB2、Oracle、Sybase など。
サービス名	相互作用のサービス名。場合によっては (例えば AIX® 共有メモリー接続)、別名がサービス名になり、実際のサービスが接続されるまで別名が使用されます。このような場合、実際のサービスが接続された時点で、新しいセッションが開始されます。したがって、ユーザーには単一セッションと認識されるものが 2 つのセッションとしてログに記録されることになります。 Teradata の場合、サービス名にはセッション論理ホスト ID 値が含まれます。
セッション ID	セッションを一意的に識別します。
ソース・アドレス	例外のソース IP アドレス。
ソース・プログラム	相互作用のソース・プログラム。

属性	記述
例外の原因となった SQL 文字列	例外を発生させた SQL 文字列。
ユーザー名	データベース・ユーザー名。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: 完全な SQL」ドメイン: エンティティーおよび属性

「完全な SQL」エンティティーは、ポリシー・ルール・アクションの「全詳細をロギング」、「値を含む全詳細をロギング」、「セッションごとに全詳細をロギング」、または「値を含む全詳細をセッションごとにロギング」によってのみ作成されます。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: 完全な SQL」エンティティー

属性	記述
分析済みのクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
クライアント・ホスト名	クライアント・ホスト名
データベース名	データベース名
データベース・ユーザー名	データベース・ユーザー名
完全な SQL	値を含む完全な SQL ステートメント。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
インスタンス ID	完全な SQL のインスタンスの固有 ID。
ネットワーク・プロトコル	ネットワーク・プロトコル
OS ユーザー	OS ユーザー
影響されるレコード	影響を受けたレコードの数 (セッションごと)。この属性を使用するレポートでは、大規模結果セットや N/A などの特殊ケースが適切に表示されるように、別名をオンにすることをお勧めします。
応答時間	要求に対する応答時間 (ミリ秒単位)。ネットワーク・トラフィック内で要求がモニターされる場合、応答時間は要求に応答するのに要した時間を正確に反映しています (Guardium はクライアント要求とサーバー応答の両方のタイム・スタンプを設定します)。
サーバー・ホスト名	サーバー・ホスト名
サーバー IP	サーバー IP
サーバー・ポート	サーバー・ポート
サーバー・タイプ	サーバー・タイプ
サービス名	サービス名
セッション ID	セッション ID
ソース・プログラム	ソース・プログラム
成功	呼び出しが成功したかどうかを示します。
タイム・スタンプ	SQL がデータベース・サーバーで実行された時刻。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: インストール済みのパッチ」ドメイン: エンティティーおよび属性

インストール済みのパッチについてレポートします。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: インストール済みのパッチ」エンティティー

属性	記述
作成日	パッチ作成日。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
Guardium バージョン	Guardium ソフトウェアのバージョン。

属性	記述
インストール実行者	CM、SA、または CLI
パッチの依存性	このパッチより前にインストールする必要がある Guardium のバージョンまたはパッチ。
パッチの記述	パッチの記述
パッチ番号	
要求された日付/時間	パッチをインストールする必要がある日付/時刻
状況	0/1 (成功/失敗)
状況の記述	パッチ
タイム・スタンプ	
アップロードの日付	インストールするパッチがアップロードされた日付
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: インスタンス」ドメイン: エンティティーおよび属性

このドメインには、モニター対象サーバーに要求が送信されるたびに検査エンジンによって収集されるトラフィック・データが含まれます。クライアント/サーバー、セッション、SQL、およびアクセス期間の関連データのすべてが含まれます。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: インスタンス」エンティティー

属性	記述
アクセス・ルールの記述	アクセス・ポリシー・ルール定義に含まれる記述。
分析済みのクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
アプリケーション・ユーザー名	この「アプリケーション・ユーザー名」エンティティーの固有 ID。
アプリケーション・イベント ID	この「アプリケーション・イベント」エンティティーの固有 ID。
平均実行時間	期間中の平均コマンド実行時間。SQL ステートメントのみが対象です。FTP トラフィックには適用されません。
クライアント・ホスト名	クライアント・ホスト名
構造 ID	オブジェクトが参照される構造を一意的に識別します。
データベース名	セッションの対象データベースの名前 (MSSQL または Sybase のみ)。 Oracle の場合、アプリケーション固有の追加情報が「データベース名」に含まれることがあります (V$SESSION ビューの MODULE 列に設定された、セッション用に現在実行中のモジュールなど)。
データベース・ユーザー名	ローカルまたはリモートのデータベースに接続したユーザー。
失敗した SQL	失敗した SQL 要求の数。照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
インスタンス ID	構成体のインスタンスまたは SQL インスタンスの固有 ID。admin ロールを持つユーザーのみが使用できます。
ネットワーク・プロトコル	使用されたネットワーク・プロトコル (例えば TCP、UDP など。Oracle 上の K-TAP については IPC または BEQ として表示されます)。
オブジェクトおよび動詞	セミコロンで区切られたオブジェクトおよび SQL 動詞タプルの名前。
元の SQL	ユーザーによって送信された元の SQL。
OS ユーザー	相互作用の OS ユーザー・アカウント。
期間の開始	期間開始属性。
期間の開始日	期間開始属性の中の日付のみ。
サーバー・ホスト名	サーバー・ホスト名
サーバー IP	サーバーの IP アドレス
サーバー・ポート	サーバー・ポート番号
サーバー・タイプ	例: Db2、Oracle、Sybase など。

属性	記述
サービス名	相互作用のサービス名。場合によっては (例えば AIX 共有メモリー接続)、別名がサービス名になり、実際のサービスが接続されるまで別名が使用されます。このような場合、実際のサービスが接続された時点で、新しいセッションが開始されます。したがって、ユーザーには単一セッションと認識されるものが 2 つのセッションとしてログに記録されることになります。 Teradata の場合、サービス名にはセッション論理ホスト ID 値が含まれます。
セッション ID	セッションを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
ソース・プログラム	相互作用のソース・プログラム。
成功した SQL	成功した SQL 要求の数。照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、これらは使用できません。
タイム・スタンプ	このレコードが作成されたタイム・スタンプ。
タイム・スタンプの日付	タイム・スタンプの日付。
影響を受けるレコード合計	影響を受けたレコードの総数。照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、これらは使用できません。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との時差。

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: 異常値リスト - 拡張」ドメイン: エンティティーおよび属性

異常値として識別されたアクティビティーとエラーの詳細な説明。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: 異常値リスト - 拡張」エンティティー

属性	記述
異常スコア	最後の異常スコア。
データベース名	アクティビティーが発生したデータベースの名前。
データベース・ユーザー名	アクティビティーに関連付けられたデータベース・ユーザー名。
エラー異常値	異常値のタイプがエラーであるかどうか。True/False。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
大量の異常値	異常値のタイプが大量であるかどうか。True/False。
新規異常値	異常値のタイプが新規であるかどうか。True/False。
インスタンスの数	アクティビティー・ボリューム。
オブジェクト	アクティビティーが実行されたオブジェクト。
期間の開始	期間開始の時刻
期間の開始日	期間開始の日付。
影響されるレコード	アクティビティーの影響を受けたレコードの数。
サーバー IP	アクティビティーが発生したサーバーの IP。
サーバー・タイプ	アクティビティーが発生したサーバーのタイプ。
ソース・プログラム	アクティビティーに関連付けられたソース・プログラム。
一時異常値	異常値のタイプが一時であるかどうか。True/False。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差
動詞	アクティビティーで使用された動詞。

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: 異常値サマリー - 拡張」ドメイン: エンティティーおよび属性

1 時間の細分度での異常値のサマリー。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: 異常値サマリー - 拡張」エンティティー

属性	記述

属性	記述
異常スコア	このアラートの最後の異常スコア。
データベース名	アクティビティーが発生したデータベースの名前。
データベース・ユーザー名	アクティビティーに関連付けられたデータベース・ユーザー名。
さまざまな異常値	異常値のタイプがさまざまであるかどうか。True/False。
エラー異常値	異常値のタイプがエラーであるかどうか。True/False。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
大量の異常値	異常値のタイプが大量であるかどうか。True/False。
新規異常値	異常値のタイプが新規であるかどうか。True/False。
現行の異常値	異常値のタイプが現行であるかどうか。True/False。
期間の開始	期間開始の日付と時刻。
期間の開始日	期間開始の日付。
特権ユーザー	アクティビティーが特権ユーザーによって実行されたかどうか。True/False。
サーバー IP	アクティビティーが発生したサーバーの IP。
サーバー・タイプ	アクティビティーが発生したサーバーのタイプ。
一時異常値	異常値のタイプが一時であるかどうか。True/False。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差。

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: ポリシー違反」ドメイン: エンティティーおよび属性

Guardium 検査エンジンまたは STAP によって検出されたポリシーのすべての違反に関するすべてのポリシー違反データ。

このエンティティーは、ポリシー・ルール違反がログに記録されるたびに作成されます。 すべてのポリシー・ルール違反がログに記録されるわけではありません。ポリシー・ルールのアクションに記載のルール・アクションの説明を参照してください。違反を引き起こしているアクセス・ルールは、従属する「アクセス・ルール」エンティティー (前述) で使用可能です。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: ポリシー違反」エンティティー

属性	記述
アクセス・ルールの記述	ルールの定義に含まれるルールの記述。
分析済みのクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
クライアント・ホスト名	クライアント・ホスト名
データベース・ユーザー名	データベース・ユーザー名。ローカルまたはリモートのデータベースに接続したユーザー。
SQL 文字列全体	ポリシー・ルール違反を引き起こしている SQL 文字列。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
オブジェクトおよび動詞	データベース・ユーザー名。データベース・ユーザー名は、ローカルまたはリモートのデータベースに接続したユーザーです。
OS ユーザー	ポリシー・ルール違反を引き起こした OS ユーザー。
サーバー・ホスト名	ポリシー・ルール違反が発生したサーバー。
サーバー IP	ポリシー・ルール違反が発生したサーバーの IP。
サーバー・タイプ	ポリシー・ルール違反が発生したサーバーのタイプ。
サービス名	ポリシー・ルール違反が発生したサービスの名前。
重大度	ポリシー・ルール違反の重大度。
ソース・プログラム	ポリシー・ルール違反が発生したソース・プログラム。
タイム・スタンプ	ポリシー・ルール違反がログに記録されるときに作成されます。すべてのポリシー・ルール違反がログに記録されるわけではありません。
タイム・スタンプの日付	タイム・スタンプの日付。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差
違反ログ ID	違反ログの固有 ID。

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: セッション」ドメイン: エンティティーおよび属性

クライアント/サーバー・データベース・セッションについてレポートします。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: セッション」エンティティー

属性	記述
アクセス ID	アクセス期間を一意的に識別します。
分析済みのクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
クライアント・ホスト名	クライアント・ホスト名
クライアント・ポート	クライアント・ポート番号
データベース名	セッションの対象データベースの名前 (MSSQL または Sybase のみ)。 Oracle の場合、アプリケーション固有の追加情報が「データベース名」に含まれることがあります (V$SESSION ビューの MODULE 列に設定された、セッション用に現在実行中のモジュールなど)。
データベース・ユーザー名	ローカルまたはリモートのデータベースに接続したユーザー。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
無視されたフラグ	
ログイン成功	
ネットワーク・プロトコル	使用されたネットワーク・プロトコル (例えば TCP、UDP など。Oracle 上の K-TAP については IPC または BEQ として表示されます)。
OS ユーザー	相互作用の OS ユーザー・アカウント。
送信者 IP	送信者 IP
サーバー・ホスト名	サーバー・ホスト名
サーバー IP	サーバー IP
サーバー・ポート	サーバー・ポート番号
サーバー・タイプ	例: Db2、Oracle、Sybase など。
サービス名	相互作用のサービス名。場合によっては (例えば AIX® 共有メモリー接続)、別名がサービス名になり、実際のサービスが接続されるまで別名が使用されます。このような場合、実際のサービスが接続された時点で、新しいセッションが開始されます。したがって、ユーザーには単一セッションと認識されるものが 2 つのセッションとしてログに記録されることになります。 Teradata の場合、サービス名にはセッション論理ホスト ID 値が含まれます。
セッション終了	セッションが終了した日時。「セッション終了」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション終了の日付	「セッション終了」の中の日付のみ。
セッション ID	セッションを一意的に識別します。
セッション無視	セッションの一部が (どの時点からでも) 無視されたかどうかを示します。
セッション開始	セッションが開始された日時。「セッション開始」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション開始の日付	「セッション開始」の中の日付のみ。
ソース・プログラム	相互作用のソース・プログラム。
UID チェーン	Unix S-TAP (K-Tap モードのみ) によってレポートされたセッションが対象。su ユーザーのユーザー名が異なる場合に、OS ユーザーのチェーンを示します。ここに表示される値は、OS プラットフォームによって異なります。例えば AIX 環境では、接頭部として IBM IBM IBM という文字列が表示される場合があります。 Solaris ゾーンの場合は、ユーザー名の代わりにユーザー ID が「Uid チェーン」でレポートされる場合があります。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との時差。

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: STAP 状況」ドメイン: エンティティーおよび属性

S-TAP の状況についてレポートします。

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: STAP 状況」エンティティー

属性	記述
アプリケーション・サーバー	アプリケーション・サーバーがインストール済みであるかどうか。はい/いいえ。
CAS サーバー名	CAS サーバーの名前。
DB ポート (最大)	データベースの聴取ポート範囲の終了ポート番号。
DB ポート (最小)	データベースの聴取ポート範囲の開始ポート番号。
データベース・サーバー・タイプ	データベース・サーバーのプロトコル。
DB Tee 実データベース・ポート	S-TAP が Tee の使用時にトラフィックを転送する宛先のポート。
暗号化	S-TAP からの通信が暗号化されているかどうか。未暗号化/暗号化のタイプ。
Guardium アプライアンス	このデータを報告したコレクターの名前。
Guardium ホスト	S-TAP のホストとしての役割を果たす Guardium システムの IP アドレスまたはホスト名。
ハンター DBS	推奨されません
I 名	
KTAP	K-TAP が S-TAP にインストールされているかどうか。はい/いいえ。
KTAP バージョン	インストールされている K-TAP のバージョン。
最後に受信した応答	S-TAP から最後に受信した状況応答。例えば、同期化中やアクティブなどです。
最後の応答時刻	最後の状況応答の時刻。
LHMON	v10.5 で非推奨になっています。LHMON ドライバーがインストールされているかどうか。
MSS 共有メモリー	共有メモリー・ドライバー (インストール済み)。はい/いいえ。
パイプ	名前付きパイプ・ドライバーが S-TAP にインストールされているかどうか。はい/いいえ。
ポリシー	
1 次ホスト名	S-TAP がデータを送信する Guardium アプライアンスの名前。
STAP 変更	
S-TAP ホスト	S-TAP がインストールされているデータベース・サーバー・システムの IP アドレスまたはホスト名。
STAP 検査状況	検査が実行されていない/検査が実行された。
S-TAP バージョン	S-TAP ソフトウェアのバージョン。
状況	S-TAP 状況。アクティブ、非アクティブ、ピンクのいずれか。
TAP ID	検査エンジンの固有 ID。
TAP IP	S-TAP がインストールされているデータベース・サーバー・システムの IP アドレス。
TAP タイプ	インストールされている S-TAP エージェントのタイプ。 stap=UNIX wstap = Windows ztap=Z/OS
TEE	Tee が有効になっているかどうか。はい/いいえ。
時差	
タイム・スタンプ	
ドロップされた合計バイト数 (現時点まで)	
無視された合計バイト数	
合計バイト数 (現時点まで)	
無視された合計応答バイト数	
TLS の使用	TLS によって暗号化されるかどうか。はい/いいえ。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差。

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: システム情報」ドメイン: エンティティーおよび属性

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: システム情報」エンティティー

属性	記述
経過時間によるアーカイブ	
アーカイブ宛先ディレクトリー	
アーカイブ宛先ホスト	
アーカイブ宛先ユーザー	
アーカイブの最大経過時間	
選択されたアーカイブ	
値のアーカイブ	
認証の LDAP タイプ	
認証サーバー・タイプ	
経過時間によるエクスポート	
エクスポート宛先ホスト	
エクスポートの最大経過時間	
選択されたエクスポート	
値のエクスポート	
gid	
グローバル ID	
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
Guardium モデル	
Guardium バージョン	
ホストの MAC アドレス	
選択されたインポート	
マネージャー IP	
経過時間によるパージ	
アーカイブされる場合にパージ	
エクスポートされる場合にパージ	
選択されたパージ	
選択されたリストア	
システム・ドメイン	
タイム・スタンプ	サーバーでこの変更レコードが作成された日時 (Guardium アプライアンス・サーバーのクロック)。
Tomcat ホスト名	
Tomcat IP	
固有の ID 接頭部	
ユニット・タイプ	
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差

親トピック: ドメインのエンティティーおよび属性

「BigData Intelligence: 脆弱性診断結果」ドメイン: エンティティーおよび属性

使用可能なロール: すべて。

このドメインは、タイプ GBDI のデータ・ソースが定義されている Guardium システムで使用できます。

「BigData Intelligence: 脆弱性診断結果」エンティティー

属性	記述
評価の記述	定義に含まれる評価名。
カテゴリー	テスト結果のカテゴリー。
データ・ソース名	データ・ソースの名前。
データ・ソース・タイプ	データベース・タイプ: Oracle、MS-SQL、Db2、Sybase、Informix など。
データベース名	データベース名。
記述	データ・ソースの記述。

属性	記述
実行日	評価が実行された日付。
Guardium アプライアンス	このデータを報告したコレクターのホスト名。
ホスト	データ・ソースのホスト名。
パッチ・レベル	データ・ソースのパッチ・レベル。
ポート	ホスト上のポート番号。
推奨	タスクに対して返された推奨。
結果テキスト	テストによって返されたテキスト。
スコアの記述	スコアの記述。
サービス名	データ・ソースのサービス名。
重大度	インシデントまたはポリシー違反の重大度。情報、低、中、高のいずれか。
テストの記述	テスト定義に含まれる記述。
テスト・スコア	返されたテスト・スコア。
UTC オフセット	UTC 時刻と、そのデータを報告したコレクターの時刻との差。
バージョン・レベル	データベースのバージョン・レベル。

親トピック: ドメインのエンティティーおよび属性

「CAS 変更」ドメイン: エンティティーおよび属性

モニター項目 (ファイル、レジストリー変数など) に対する変更をトラッキングします。.

使用可能なロール: すべて

「モニターされた変更」エンティティー

このエンティティーは、モニター項目が変更されるたびに作成されます。CAS インスタンス内のモニター項目を識別し、変更の保存データを指し示します。

属性	記述
変更 ID	変更の固有 ID。
サンプルの時刻	サンプルが取られたときのタイム・スタンプ (ホスト上の日時)。
監査構成 ID	ホスト構成を識別します。
保存データ ID	この変更に関する「保存データ」エンティティーを識別します。
監査状態ラベル ID	この変更に関する「ホスト構成」エンティティーを識別します。
タイム・スタンプ	サーバーでこの変更レコードが作成された日時 (Guardium アプライアンス・サーバーのクロック)。
MD5	MD5 アルゴリズムを使用してチェックサムを計算し、その値を前回その項目がチェックされたときに計算された値と比較することによって比較を実行するかどうかを示します。デフォルトでは、MD5 は使用しない設定になっています。MD5 を使用しても、生データのサイズが CAS ホスト用に構成された「MD5 サイズ制限」を上回る場合は、MD5 による計算と比較はスキップされます。MD5 を使用するかどうかに関わらず、項目の最終変更タイム・スタンプの現行値と項目のサイズは、前回その項目がチェックされたときに保存された値と比較されます。
所有者	Unix のみ。項目タイプがファイルの場合に、ファイル所有者。
許可	Unix のみ。項目タイプがファイルの場合に、ファイル・アクセス権。
サイズ	ファイル・サイズ。ただし、次のような特殊値があります。 -1 = ファイルは存在するが、バイト数がゼロである。 0 (ゼロ) = ファイルは存在しないが、このファイル名がモニターされている (存在しなかったか、または削除された可能性がある)。
最終変更	最終変更のタイム・スタンプ。サンプルの時刻にファイル・システムから取得されたもの。
最終変更日	最終変更の日付。
最終変更の時刻	最終変更の時刻。
最終変更の曜日	最終変更の曜日。
最終変更の年	最終変更の年。
グループ	Unix のみ。項目タイプがファイルの場合に、グループ所有者。

「ホスト構成」エンティティー

「ホスト構成」エンティティーは、CAS インスタンス内の項目ごとに作成されます。

属性	記述
監査状態ラベル ID	構成項目の固有の数値 ID。

属性	記述
タイム・スタンプ	エンティティー作成のタイム・スタンプ。
ホスト名	データベース・サーバーのホスト名または IP アドレス。
OS タイプ	オペレーティング・システム: Unix または Windows
データベース・タイプ	データベース・タイプ: Oracle、MS-SQL、DB2®、Sybase、Informix®。オペレーティング・システム・インスタンスに対する変更である場合は「N/A」。
インスタンス名	テンプレート・セット・インスタンスの名前。
タイプ	変更されたモニター項目のタイプ。 OS スクリプトまたは SQL スクリプト: モニター項目テンプレート定義に含まれる OS スクリプトによって発生した変更。 環境変数: 環境変数 (Unix のみ) レジストリー変数: レジストリー変数 (Windows のみ) ファイル: 特定のファイル。インスタンスによって使用されるテンプレート・セットで定義されたファイル・パターンを対象としたホスト構成エンティティーはありません。 代わりに、パターンに一致するファイルごとに、別々のホスト構成エンティティーがあります。
モニター項目	変更された項目の名前。記述 (入力されている場合) から取得され、それ以外の場合はタイプに応じたデフォルトの名前 (例えばファイル名) になります。

「保存データ」エンティティー

「保存データ」エンティティーは、モニター対象の項目に対する変更が検出されるたびに作成されます (項目テンプレート定義でその項目の「データを保持」ボックスにマークが付けられた場合)。

属性	記述
保存データ ID	保存データ項目の固有の数値 ID。
保存データ	保存されている実際のデータ。
タイム・スタンプ	保存データ・エンティティーがサーバー・データベースで記録されたときのタイム・スタンプ。
変更 ID	この保存データ・エンティティーに対応するモニターされた変更エンティティーを識別します。

「保存データ ID」は、admin ロールを持つユーザーのみが使用できます。

親トピック: ドメインのエンティティーおよび属性

「CAS 構成」ドメイン: エンティティーおよび属性

CAS ホスト構成をトラッキングします。 ここで、構成とは、特定のデータベース・サーバー・ホストに 対する、1 つ以上のテンプレート・セットのアプリケーションです。 構成インスタンスから、 テンプレート・セット内で使用可能または使用不可になっている項目や、 ファイル名パターン・テンプレートによって選択されモニターされている (またはされていない) 正確なファイルを 確認することができます。

使用可能なロール: すべて

「ホスト」エンティティー

CAS ホスト (データベース・サーバー) および CAS の現行の状況 (オンライン/オフライン) を識別します。データベース・サーバー・ホスト上で CAS が初めて認識されたときに、CAS ホスト・エンティティーが作成されます。 オンライン/オフライン状況が変わるたびに更新されます。 「ホスト」エンティティーは、「CAS ホスト履歴」ドメインおよび「CAS 構成」ドメインで使用可能です。

属性	記述
ホスト ID	
ホスト名	データベース・サーバー・ホスト名 (IP アドレスとして表示される場合があります)。
IP	
オンライン	レコードが書き込まれたときのオンライン状況 (はい/いいえ)。
OS タイプ	オペレーティング・システム: UNIX または WIN

「インスタンス構成」エンティティー

「インスタンス構成」エンティティーは、インスタンス構成が定義されるたびに作成されます。このエンティティーは、CAS インスタンスがデータベースに接続する方法を (必要に応じて) 定義し、インスタンスの使用したテンプレート・セットを識別します。

記述	記述
データベース・タイプ	データベース・タイプ (Oracle、MS-SQL、DB2®、Sybase、Informix® など) またはオペレーティング・システム・インスタンスの場合は「N/A」。

記述	記述
インスタンス	インスタンスの名前。
ユーザー	データベースへのログオンに CAS が使用するユーザー名。オペレーティング・システム・インスタンスの場合は「N/A」。
ポート	データベースへの接続に CAS が使用するポート番号。オペレーティング・システム・インスタンスの場合は空。
データベース・ホーム・ディレクトリー	データベースのホーム・ディレクトリー。オペレーティング・システム・インスタンスの場合は空。
テンプレート・セット ID	このインスタンスによって使用されたテンプレート・セットを識別します。

「データ・ソース」エンティティー

属性	記述
接続プロパティー	このデータ・ソースとの JDBC 接続を確立するために JDBC URL に追加の接続プロパティーが含まれている場合にのみ報告されます。
データ・ソース重大度	重大度分類 (あるいは影響レベル) (データ・ソースに定義されている場合)。
データベース名	データ・ソースに定義されている場合: Db2 または Oracle のデータ・ソースの場合はスキーマ名。 その他の場合はデータベース名。
データベース記述	データ・ソースの詳細説明 (定義されている場合)。
データベース ID	
データベース名	データ・ソースに定義されている場合: Db2 または Oracle のデータ・ソースの場合はスキーマ名。 その他の場合はデータベース名。
データベース・タイプ	データ・ソースのタイプ。
ホスト	ホスト名または IP アドレス
最後の接続	
ポート	ポート (データ・ソースに定義されている場合)。
サービス名	Db2 データ・ソースの場合はデータベース名。Oracle、Informix、および IBM iSeries の場合はサービス名。
共有	True/False。True は、データ・ソースが他のアプリケーションと共有されていることを示します。
ユーザー名	ユーザー名 (このデータ・ソースに定義されている場合)。

親トピック: ドメインのエンティティーおよび属性

「CAS ホスト履歴」ドメイン: エンティティーおよび属性

CAS ホスト・イベント (サーバーまたはクライアントのサービス開始やサービス休止など) を追跡します。

使用可能なロール: すべて

「ホスト」エンティティー

CAS ホスト (データベース・サーバー) および CAS の現行の状況 (オンライン/オフライン) を識別します。データベース・サーバー・ホスト上で CAS が初めて認識されたときに、CAS ホスト・エンティティーが作成されます。 オンライン/オフライン状況が変わるたびに更新されます。 「ホスト」エンティティーは、「CAS ホスト履歴」ドメインおよび「CAS 構成」ドメインで使用可能です。

属性	記述
ホスト ID	
ホスト名	データベース・サーバー・ホスト名 (IP アドレスとして表示される場合があります)。
IP	
オンライン	レコードが書き込まれたときのオンライン状況 (はい/いいえ)。
OS タイプ	オペレーティング・システム: UNIX または WIN

「ホスト・イベント」エンティティー

CAS クライアント/サーバー関係におけるイベントの日時。ホスト・イベント・エンティティーは、CAS によってイベント (イベント・タイプを参照) が検出されるかシグナル通知されるたびに作成されます。

属性	記述
監査ホスト・イベント ID	ホスト・イベント・エンティティーを識別します。
監査ホスト ID	ホストを識別します。
イベント時間	イベントが記録された日時。

属性	記述
イベント・タイプ	記録されているイベントを識別します。以下のタイプがあります。 クライアント稼働 - データベース・サーバー・ホスト上の CAS が始動しました。 クライアント停止 - データベース・サーバー・ホスト上の CAS が停止しました。 フェイルオーバー Off - サーバーが (切断後に) 使用可能になったので、CAS データはサーバーに書き込まれます。 フェイルオーバー On - サーバーが使用不可なので、CAS データはフェイルオーバー・ファイルに書き込まれます。 サーバー停止 - データベース・サーバーが停止しました。 サーバー稼働 - データベース・サーバーが始動しました。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
タイム・スタンプ	エンティティー作成のタイム・スタンプ。

親トピック: ドメインのエンティティーおよび属性

「CAS テンプレート」ドメイン: エンティティーおよび属性

CAS テンプレート定義をトラッキングします。 テンプレートは、変更をモニターされる項目を識別します。 モニター項目は、ファイル、環境またはレジストリー変数、OS または SQL スクリプト出力セット、 またはログオン・ユーザー・セットのいずれであってもかまいません。

使用可能なロール: すべて

「テンプレート・セット」エンティティー

テンプレート・セット定義を記述します。「テンプレート・セット」エンティティーは、テンプレート・セットごとに作成されます。テンプレート・セットとは、特定のオペレーティング・システムまたはデータベース用のテンプレート項目のセットのことです。

属性	記述
テンプレート・セット ID	テンプレート・セットの固有 ID (連番)。
OS タイプ	オペレーティング・システム: Unix または Windows
データベース・タイプ	データベース・タイプ (Oracle、MS-SQL、DB2®、Sybase、Informix® など) またはオペレーティング・システム・テンプレートの場合は「N/A」。
テンプレート・セット名	テンプレート名。
IsDefault	このテンプレートが、指定された OS タイプとデータベース・タイプの組み合わせにとってデフォルトかどうかを示します。
編集可能	このテンプレートを変更できるかどうかを示します。 デフォルト Guardium® テンプレートは変更できません。 さらに、CAS インスタンスで一度使用されたテンプレート・セットは変更できません。ただし、テンプレート・セットのコピーは常に作成可能であり、そのコピー・セットに変更を加えることができます。
タイム・スタンプ	テンプレートが最後に更新された日時。

「テンプレート」エンティティー

テンプレート・セット内のテンプレート項目を記述します。

属性	記述
テンプレート ID	テンプレート・セットの固有 ID (連番)。
アクセス名	監査タイプに応じて、OS スクリプトまたは SQL スクリプト、環境値またはレジストリー値、あるいはファイル名またはファイル名パターンになります。
監査タイプ	モニター項目のタイプ。
監査頻度 (分)	テスト間の最大間隔 (分単位)。
MD5 を使用	MD5 アルゴリズムを使用してチェックサムを計算し、その値を前回その項目がチェックされたときに計算された値と比較することによって比較を実行するかどうかを示します。デフォルトでは、MD5 は使用しない設定になっています。MD5 を使用しても、生データのサイズが CAS ホスト用に構成された「MD5 サイズ制限」を上回る場合は、MD5 による計算と比較はスキップされます。MD5 を使用するかどうかに関わらず、項目の最終変更タイム・スタンプの現行値と項目のサイズは、前回その項目がチェックされたときに保存された値と比較されます。
データ保存	「データを保持」チェック・ボックスにマークが付けられているかどうかを示します。マークが付けられている場合は、前のバージョンの項目を現行バージョンと比較できます。
記述	テンプレートの記述 (オプション)。

属性	記述
タイム・スタンプ	テンプレートが最後に更新された日時。

親トピック: ドメインのエンティティーおよび属性

「カタログ」ドメイン: エンティティーおよび属性

「項目」エンティティー

属性	記述
アグリゲーター	
監査プロセスの記述	
コメント	
ディスク・スペース	
開始日付	
グローバル ID	
Guardium バージョン	
実行番号	
階段	
タイム・スタンプ	
終了日付	
タイプ	

「ロケーション」エンティティー

アーカイブに保存されなかった日を確認する方法

変更可能な照会 (「ツール」タブ>「レポートのビルド」>「レポート・ビルダー」>「ロケーション・ビュー」照会) を使用して、アーカイブに保存されたファイルを示したレポートを作成します。このレポートには、すべてのファイルとアーカイブの日付のリストが表示されます。このレポートに組み込まれなかった日付は、アーカイブに保存されなかった日付です。必要であれば、リストに組み込まれていない日付のアーカイブを実行してください。

属性	記述
ホスト	ホスト名
パス	ユーザーの名前
保持	ファイルへのパス名
ユーザー名	

エンティティーによる使用

属性	記述
マウント・イベント	
マウント時間	
要求カウント	
使用者	

「宛先」エンティティー

属性	記述
クラス ID	
システム・タイプ	

親トピック: ドメインのエンティティーおよび属性

「分類プロセスの結果」ドメイン: エンティティーおよび属性

分類プロセスの実行と結果についてレポートします。

使用可能なロール: すべて

「分類プロセス実行」エンティティー

このエンティティーは、分類プロセスのジョブ実行について示します。

属性	記述
プロセスの記述	プロセス定義から。
状況	ジョブの状況。
キュー日時	ジョブが分類/評価キューに実行依頼されたときのタイム・スタンプ。
開始日時	ジョブ開始時のタイム・スタンプ。
終了日時	ジョブ終了時のタイム・スタンプ。
データ・ソース	ジョブ終了時のタイム・スタンプ。

「分類プロセスの結果」エンティティー

このエンティティーは、実行される分類プロセス・ルールごとに作成されます。

属性	記述
カタログ	結果セットのカタログのロケーション。
スキーマ	スキーマ名 (該当する場合)。
表名	ルール定義に含まれる表名。
列名	ルール定義に含まれる列名。
ルールの記述	分類ポリシー・ルールの記述。
コメント	このルール定義に追加されたコメント。
分類名	ルールの分類。
カテゴリー	ルールのカテゴリー。
データ・ソースの記述	ルールのデータ・ソース。

親トピック: ドメインのエンティティーおよび属性

「CM バッファー使用状況モニター」ドメイン: エンティティーおよび属性

中央マネージャーにアップロードされたすべての「スニファーのバッファー使用」エンティティーの統合を示します。

「CM バッファー使用状況モニター」エンティティー

属性	記述
スニファーのバッファー使用 ID	
タイム・スタンプ	レコードが作成された時刻。
スニファーの CPU PCT	スニファーによって使用された CPU のパーセンテージ。
スニファーのメモリー PCT	スニファーによって使用されたメモリーのパーセンテージ。
MySQL の CPU PCT	MySQL によって使用された CPU のパーセンテージ。
MySQL の MEM PCT	MySQL によって使用されたメモリーのパーセンテージ。
PID	スニファー・プロセス ID。
メモリー	スニファーによって使用されたメモリー量。
時刻	スニファーによって使用された経過時間。
空きバッファー	空きバッファー・スペース量。
アナライザー・レート	メッセージが分析される速度。
アナライザー・キュー	分析キューのサイズ。
アナライザー総計	分析されたメッセージの総数。
ロガー・キュー	ロガー・キューのサイズ。
ロガー総計	ログに記録されたメッセージの総数。
セッション・キュー	セッション・キューのサイズ。
セッション総計	セッションの総数。
ハンドラー・データ	内部スニッフィング・エンジン・データ。
補足 STR	内部スニッフィング・エンジン・データ。
使用されたスニファー接続	検査エンジンの再始動以降、現在モニターされている接続の総数。
ドロップされたスニファー・パケット	スニファーによってドロップされたパケット。
無視されたスニファー・パケット	スニファーによって無視されたパケット。

属性	記述
スロットルされたスニファー・パケット	検査エンジンの再始動以降、スロットルのために無視された接続の総数。
終了したスニファー接続	検査エンジンの再始動以降、モニターされて終了した接続の総数。
ロガー・セッション・カウント	ログに記録されたセッションの数。
ルールにより無視されたロガー・パケット	ポリシー・ルール・アクションにより無視されたパケット。
アナライザー逸失パケット	アナライザーによって失われたパケット。
モニターされるロガー・データベース	現在モニターされているデータベース・タイプのリスト。
Mysql 起動済み	内部データベース再始動を表すブール値インディケーター (1 = 再始動済み、0 = 未再始動)。
システム CPU 負荷	システム CPU 使用状況。
システム・アップタイム	最後の始動からの時間。
Mysql ディスク使用状況	MySQL ディスク使用状況。
システム・メモリー使用状況	システム・メモリー使用状況。
/var ディスク使用状況	/var ディスク使用状況。
システム・ルート・ディスク使用状況	システム・ルート・ディスク使用状況。
Eth0 受信	ETH 0 で受信されたメッセージ。
Eth0 送信	ETH 0 で送信されたメッセージ。
プロミスキャス受信	スニッフィング・ネットワーク・カード (非インターフェース・ポート) を介した受信パケットの率。
オープン FD	オープン・ファイル記述子。
オープン FD MySQL	データベース・オープン・ファイル記述子。
通常のセッション	通常のセッションの数。
オープンされていないセッション	スニファーによって開かれなかったセッションの数。
セッション・タイムアウト	タイムアウトになったセッションの数。
セッション無視	スニファーによって無視されたセッションの数。
セッション直接クローズ	直接閉じられたセッションの数。
セッション推測	推測されたセッション数。
SqlGuard タイム・スタンプ	カスタム表にレコードが挿入された時刻。
データ・ソース名	レコードのアップロードに使用されたデータ・ソースの名前。

親トピック: ドメインのエンティティーおよび属性

「コメント」ドメイン: エンティティーおよび属性

各種 Guardium コンポーネントに関するユーザー定義のコメント。

使用可能なロール: すべて

「コメント」エンティティー

このエンティティーは、ユーザー・コメントについて示します。「コメント」ドメインでのみ使用可能です。これは、admin ユーザーに限定されています。 このドメインには、共有可能コメントのみが含まれます。共有可能コメントとは、ローカルに実行されるもの以外のすべてのコメントのことです (「ローカル・コメント」エンティティーを参照)。

属性	記述
コメント作成者	コメントを作成した Guardium® ユーザー。
コメント参照	コメントが付加された要素 (例えば、照会、監査プロセスの結果、または別のコメント) を示します。
コメントの内容	完全なコメント・テキスト。
タイム・スタンプ	コメントが作成された日時。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。

属性	記述
オブジェクトの記述	定義されたコメントの対象となったオブジェクトの名前。 例えば、ポリシーについて定義されたコメントには、ACCESS_RULE_SET のオブジェクトの記述が含まれています。
レコード・アソシエーション	このコメントが関連付けられたレコードのリスト。

「ローカル・コメント」エンティティー

このエンティティーで、ローカル・コメントを記述します。「コメント」ドメインでのみ使用可能です。これは、admin ユーザーに限定されています。 このエンティティーには、ローカルに実行されたプロセスと結果セットに関するローカル・コメントのみが含まれます。 共有可能なコメントは、「コメント」エンティティーで定義されます。

属性	記述
コメント作成者	コメントを作成した Guardium ユーザー。
コメント参照	コメントが付加された要素 (例えば、照会、監査プロセスの結果、または別のコメント) を示します。
コメントの内容	完全なコメント・テキスト。
タイム・スタンプ	コメントが作成された日時。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
オブジェクトの記述	定義されたコメントの対象となったオブジェクトの名前。 例えば、インシデントについて定義されたコメントには、INCIDENT のオブジェクトの記述が含まれています。
レコード・アソシエーション	このコメントが関連付けられたレコードのリスト。

親トピック: ドメインのエンティティーおよび属性

「カスタム・データベース使用状況」ドメイン: エンティティーおよび属性

カスタム・データベース統計

使用可能なロール: admin

「カスタム・データベース・ディスク使用状況」エンティティー

属性	記述
InnoDB 表の数	InnoDB 表の数
MyISAM 表の数	MyISAM 表の数
表の数	表の数
タイム・スタンプ	この行のデータが作成/更新されたタイム・スタンプ
InnoDB 表の合計サイズ (MB)	InnoDB 表の合計サイズ (MB)
MyISAM 表の合計サイズ (MB)	MyISAM 表の合計サイズ (MB)

「カスタム表ディスク使用状況」エンティティー

属性	記述
カスタム表の名前	カスタム表の名前
表のサイズ (MB)	表のサイズ (MB)
表タイプ	表タイプ

親トピック: ドメインのエンティティーおよび属性

「有効になっているデータベース・デフォルト・ユーザー」ドメイン: エンティティーおよび属性

デフォルト・ユーザーが有効になっているかどうかの詳細。

使用可能なロール: admin

非資格情報スキャン - データベースのリストをスキャンし、デフォルト・ユーザーが使用可能になっているかどうかを検査するプロセス。 デフォルト・ユーザーと、スキャン対象サーバー・リストが、パラメーターとして API に提供されます。 それぞれのデータベース・タイプについて、インストール時にデータベースにより作成されたデフォルト・ユーザーとパスワードを含むデフォルトのグループが提供され、お客様がそのリストに追加したり削除したりできます。 グループのタイプは「データベース・ユーザー/データベース・パスワード」であり、デフォルト・グループの名前は次のとおりです。

ORACLE デフォルト・ユーザー、Db2 デフォルト・ユーザー、SYBASE デフォルト・ユーザー、MS SQL SERVER デフォルト・ユーザー、INFORMIX デフォルト・ユーザー、MYSQL デフォルト・ユーザー、TERADATA デフォルト・ユーザー、IBM ISERIES デフォルト・ユーザー、POSTGRESQL デフォルト・ユーザー、NETEZZA デフォルト・ユーザー

「有効になっているデータベース・デフォルト・ユーザー」エンティティー

属性	記述
データベース・タイプ	データベース・タイプ
データベース・バージョン	データベースのバージョン
ホスト名	このデータベースのホスト名
インスタンス名	データベース・インスタンス名
ポート	このデータベースのポート番号
タイム・スタンプ	ユーザー名
ユーザー名	

親トピック: ドメインのエンティティーおよび属性

「ディスカバーされたインスタンス」ドメイン: エンティティーおよび属性

GIM によってディスカバーされたインスタンス。

使用可能なロール: すべて

「ディスカバーされたインスタンス」エンティティー

このエンティティーは、ディスカバーされたインスタンスを識別します。

属性	記述
タイム・スタンプ	エンティティーのこのインスタンスを Guardium® が記録するときに作成されるタイム・スタンプ値 (インスタンスごとに固有のタイム・スタンプがあります)。
ホスト	このインスタンスのホスト名。
プロトコル	このインスタンスに固有のプロトコル。
ポート (最小)	ポート範囲 (検査エンジンの最小ポート番号)。
ポート (最大)	ポート範囲 (検査エンジンの最大ポート番号)。
クライアント IP	クライアントの IP アドレス/マスク。
除外クライアント IP	除外するクライアントの IP アドレス/マスク。
プロセス名	データベース実行可能プログラムの名前。
名前付きパイプ	データベースによって使用されたパイプ名。
KTAP データベース・ポート	KTAP のデータベース・ポート。
データベース・インストール・ディレクトリー	データベース・インストール・ディレクトリー。
プロセス名	プロセス名
Db2 共有メモリー調整	パケット・ヘッダー・サイズ
DB2®Db2 共有メモリー・クライアント位置	クライアント入出力域オフセット
Db2共有メモリー・サイズ	Db2 共有メモリー・セグメント・サイズ
インスタンス名	ディスカバーされたインスタンスの名前。
Informix バージョン	Informix バージョン

親トピック: ドメインのエンティティーおよび属性

「分散データマート」ドメイン: エンティティーおよび属性

「分散データマート状況」エンティティー

属性	記述
データマート ID	分散データマートの固有の ID。
データマート名	分散データマートの名前。
詳細	分散データマートの実行の詳細。
終了時刻	このユニットで分散データマートがデータの収集を終了した時刻。
ホスト名	分散データマートがデータを収集したユニットのホスト名。
照会 ID	分散データマートの定義のベースとなる照会の ID。
レポート ID	分散データマートの定義のベースとなるレポートの ID。

属性	記述
レポート・タイトル	分散データマートの定義のベースとなるレポートのタイトル。
実行 ID	分散データマートの実行の固有の ID。
2 次ターゲット状況に送信	2 次ターゲット上のデータ統合の状況。
開始時刻	このユニットで分散データマートがデータの収集を開始した時刻。
状況	1 次ターゲット上のデータ収集の状況。
状況 ID	分散データマートの状況の固有の ID。
タイム・スタンプ	分散データマートの状況変更のタイム・スタンプ。
ユーザー ID	分散データマートを所有するユーザー。

親トピック: ドメインのエンティティーおよび属性

「Eagle Eye」ドメイン: エンティティーおよび属性

使用可能なロール: admin。

「ケース」エンティティー

属性	記述
表示の始まり	この疑わしい攻撃について最初に記録された徴候。
ケースの終了	この疑わしい攻撃について最後に記録された徴候。
ケース ID	疑わしい攻撃の固有 ID。
ケース・タイプ ID	攻撃タイプ。例: SP (悪意のある STP)、SQLI (SQL インジェクション)。
信頼度 (%)	これが攻撃であり、正当なアクティビティーではないことの確信度。
作成時刻	この疑わしい攻撃について最初に記録された徴候の時刻。
データベース・タイプ	疑わしい攻撃のターゲットになっているデータベースのタイプ。
データベース・ユーザー名	疑わしいアクティビティーの実行に使用されたデータベース名。
リスク	疑わしい攻撃が機密データを危険にさらす可能性があるリスク: 1 (低)、2 (中)、3 (高)。
STP ID	悪意のある SP のケースのストアード・プロシージャーの固有 ID。
サーバー IP	疑わしい攻撃のターゲットになっている IP アドレス。
サービス名	疑わしい攻撃のターゲットになっているサービスの名前。
ソース・プログラム	疑わしい攻撃のターゲットになっているソース・プログラム。
タイム・スタンプ	このレコードの作成時刻。
追加情報	疑わしい攻撃に関する追加詳細。

「ケース・タイプ」エンティティー

脅威検出ケースのメタデータ表。

属性	記述
ケース・タイプ ID	攻撃 ID。
ケース・タイプ名	攻撃名。
記述	ケース・タイプに関する一般情報。
タイム・スタンプ	このレコードのタイム・スタンプ。

「ケースの徴候のリンク」エンティティー

リンク表。

属性	記述
ケース ID	ケース ID。
徴候 ID	徴候 ID。

「ケースの徴候」エンティティー

属性	記述
構造 ID	関連する SQL 構造 ID。
カウント	この徴候の発生回数。

属性	記述
記述	徴候の説明。
詳細	追加詳細。
エラー	徴候を生成した疑わしい SQL エラー。
元の SQL	徴候を生成した疑わしい SQL ステートメント。
STP ID	悪意のある SP のケースのストアード・プロシージャーの固有 ID。
表示の始まり	この徴候が最初に記録された時刻。
重大度	リスクの計算時に使用される、割り当てられたスコア。
徴候の終了	この徴候が最後に記録された時刻。
徴候 ID	この徴候の固有 ID。
徴候のタイプ ID	この徴候のタイプの固有 ID。
タイム・スタンプ	このレコードのタイム・スタンプ。
追加情報	

「徴候のタイプ」エンティティー

「アクティブ」を除くメタデータ表

属性	記述
記述	テキストの記述。
アクティブ	Guardium がこの徴候をスキャンするかどうか。
徴候の説明の接頭部	徴候の説明の接頭部。
徴候グループ	
徴候名	徴候のテキスト名。
徴候のタイプ ID	徴候のタイプの固有 ID。
タイム・スタンプ	このレコードのタイム・スタンプ。

親トピック: ドメインのエンティティーおよび属性

「例外」ドメイン: エンティティーおよび属性

このドメインには、トラフィックの詳細 (例外と例外関連データのすべて) が含まれます。 Guardium 自体で発生した例外だけでなく、データベース・サーバーから送信されて検査エンジンによって収集された SQL 例外も含まれます。

使用可能なロール: すべて

「クライアント/サーバー」エンティティー

このエンティティーは、特定のクライアント/サーバー接続について示します。固有の属性セット (タイム・スタンプを除く) が検出されるたびにインスタンスが作成されます。

注: 「アクセスのトラッキング」の場合のみ、「クライアント/サーバー」エンティティー名は、プルダウン・メニューに 2 つの可能なエンティティー (「クライアント/サーバー」および「セッション別クライアント/サーバー」) として表示されます。

「セッション別クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「セッション」から日付状態を取得します。

「クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「クライアント/サーバー」から日付状態も取得します。

ユーザーが「クライアント/サーバー」を選択すると、照会には ATTRIBUTE_ID = 1 が設定されます。ユーザーが「セッション別クライアント/サーバー」を選択すると、照会には MAIN_ATTRIBUTE_ID = 0 が設定されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
分析されたクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
クライアント・ホスト名	クライアント・ホスト名。
クライアント IP	クライアントの IP アドレス。
クライアント IP/データベース・ユーザー	クライアント IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。

属性	記述
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名	タプル・グループ
クライアント IP/ソース・アプリケーション/ユーザー	タプル・グループ
クライアント MAC	クライアントのハードウェア・アドレス。
クライアント OS	クライアントのオペレーティング・システム。 Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
データベース・プロトコル	データベース・サーバー固有のプロトコル。
データベース・プロトコル・バージョン	データベース・プロトコルのプロトコル・バージョン。
データベース・ユーザー名	データベース・ユーザー名: ローカルまたはリモートのデータベースに接続したユーザー。
最終使用日	
ネットワーク・プロトコル	使用されたネットワーク・プロトコル (例えば TCP、UDP など。Oracle 上の K-TAP については IPC または BEQ として表示されることに注意してください)。
OS ユーザー	相互作用の OS ユーザー・アカウント。
サーバーの記述	サーバーの記述 (ある場合)。
サーバー・ホスト名	サーバー・ホスト名。
サーバー IP	サーバーの IP アドレス。
サーバー IP/データベース・ユーザー	サーバー IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
サーバー IP/サービス名/データベース・ユーザー	タプル・グループ
サーバー OS	サーバーのオペレーティング・システム。 Informix の場合、OS が次のように表示される場合があります。 IEEEM (Unix または JDBC を表します) IEEEI (Windows を表します) DEC (DEC Alpha を表します) Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
サーバー・タイプ	DB2、Oracle、Sybase など。
サービス名	相互作用のサービス名。場合によっては (例えば AIX 共有メモリー接続)、別名がサービス名になり、実際のサービスが接続されるまで別名が使用されます。このような場合、実際のサービスが接続された時点で、新しいセッションが開始されます。したがって、ユーザーには単一セッションと認識されるものが 2 つのセッションとしてログに記録されることになります。 Teradata の場合、サービス名にはセッション論理ホスト ID 値が含まれます。
ソース・プログラム	相互作用のソース・プログラム。
セッション別クライアント/サーバー	セッション別クライアント/サーバーは、メイン・エンティティーでもあります。 この「2 次エンティティー」にアクセスするには、「1 次エンティティー」である「クライアント/サーバー」をクリックします。

属性	記述
タイム・スタンプ	このエンティティーのすべての属性が静的情報を持つので、このタイム・スタンプは、定義済みクライアント/サーバー接続上で Guardium が要求を初めて確認したときに、1 回だけ作成されます。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。

「セッション」エンティティー

このエンティティーは、クライアント/サーバー・データベース・セッションごとに作成されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
クライアント・ポート	クライアント・ポート番号。
データベース名	セッションの対象データベースの名前 (MSSQL または Sybase のみ)。 Oracle の場合、アプリケーション固有の追加情報が「データベース名」に含まれることがあります (V$SESSION ビューの MODULE 列に設定された、セッション用に現在実行中のモジュールなど)。
期間 (秒)	セッション開始からセッション終了までの時間の長さを示します (秒単位)。
グローバル ID	セッション・アクセスを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
これ以降を無視	このセッションを無視し始めたときに作成されたタイム・スタンプ。
非アクティブ・フラグ	0 (デフォルト) - オープン (SQL パッケージによって生成されたセッションの場合)。 1 - クローズ (切断/ログアウト受信)。 2 - おそらくクローズ (長時間にわたってパケットがない状態では非クローズ)。 3 - 非 SQL パケットから生成されたセッションの場合。
古いセッション ID	このセッションが作成されたセッションを示します。 これが接続の最初のセッションである場合は、ゼロです。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
プロセス ID	接続を開始したクライアントのプロセス ID (常に使用可能とは限らない)。
サーバー・ポート	サーバー・ポート番号。
セッション終了	セッションが終了した日時。「セッション終了」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション終了の日付	「セッション終了」の中の日付のみ。
セッション終了の時刻	「セッション終了」の中の時刻のみ。
セッション終了の曜日	「セッション終了」の中の曜日のみ。
セッション終了の年	「セッション終了」の中の年のみ。
セッション ID	セッションを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
セッション無視	セッションの一部が (ある時点から) 無視されたかどうかを示します。
セッション開始	セッションが開始された日時。「セッション開始」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション開始の日付	「セッション開始」の中の日付のみ。
セッション開始の時刻	「セッション開始」の中の時刻のみ。
セッション開始の曜日	「セッション開始」の中の曜日のみ。
セッション開始の年	「セッション開始」の中の年のみ。
端末 ID	セッション情報を解決するために内部で使用された、接続の端末 ID。
タイム・スタンプ	初めは、進行中のアクティブ・セッションのないクライアント/サーバー接続上の最初の要求に対して、タイム・スタンプが作成されます。 その後、セッションが閉じられたとき、または長期間にわたってアクティビティーが監視されないために非アクティブのマークがセッションに付けられたときに、更新されます。 セッション情報をトラッキングする場合は、「タイム・スタンプ」属性よりも「セッション開始」属性と「セッション終了」属性を注目する方が適切な場合があります。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。

属性	記述
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
TTL	admin ロール専用に予約済み。
Uid チェーン	Unix S-TAP (K-Tap モードのみ) によってレポートされたセッションが対象。su ユーザーのユーザー名が異なる場合に、OS ユーザーのチェーンを示します。 ここに表示される値は、OS プラットフォームによって異なります。例えば AIX 環境では、接頭部として IBM IBM IBM という文字列が表示される場合があります。 注: Solaris ゾーンの場合は、ユーザー名の代わりにユーザー ID が「UID チェーン」でレポートされる場合があります。
Uid チェーン圧縮	圧縮された値。「Uid」チェーンを参照してください。

「クライアント/サーバー・セッション」エンティティー

属性	記述
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名/OS ユーザー/データベース名	
サーバー IP/サーバー・ポート	

「例外タイプ」エンティティー

固定の例外タイプ・セットがあり、そのうちの 1 つが、ログに記録された各例外に関連付けられます。これらは、所有「例外」エンティティーからのレポート作成のみに使用可能です。

属性	記述

属性	記述
例外の記述	例外タイプのテキストの記述。下記リストの中から選択されます。これらの大半は表示されることはありません。 新しい構成が使用されました アラート・プロセスが例外をスローしました カスタム・アラート処理の例外 データベース・サーバーがエラーを返しました このメッセージの場合、データベース・エラー・コードは「例外」エンティティーの「例外の記述」属性に保管され、データベース・エラー・メッセージのテキスト・バージョンが、「データベース・エラー・テキスト」エンティティーの「データベース・エラー・テキスト」属性で使用可能です。 データベース・プロトコルの例外 デバッグが例外メカニズムをプリントスルーします ドロップされたデータベース要求 過剰トラフィックのためにセッション情報がドロップされました。 構成監査システム処理中にエラーが発生しました 分類処理中にエラーが発生しました 無効な照会の呼び出し ログインに失敗しました 低レベルのデータベース・プロトコルの例外 スケジュールに入れられたジョブが例外をスローしました セキュリティー・アセスメントの例外 セキュリティーの例外 このメッセージの場合、違反コードの実行 (ユーザーが Java™ API を使用して独自のアラートまたは評価を定義した場合など) がブロックされたときに、カスタム・クラス例外が発生しています。 セッションが予定より早く終了しました SQL パーサーの例外 S-TAP® 接続の再接続 このメッセージの場合、データベース・サーバーの IP アドレスまたは DNS 名は、「例外」エンティティーの「例外の記述」属性で使用可能です。 S-TAP の接続のタイムアウト このメッセージの場合、データベース・サーバーの IP アドレスまたは DNS 名は、「例外」エンティティーの「例外の記述」属性で使用可能です。 TCP エラー このメッセージの場合、エラーに関する追加情報が「例外」エンティティーの「例外の記述」属性に含めれます。 Turbine クラスが例外をスローしました レポートをパージできません

「例外」エンティティー

このエンティティーは、検出された例外ごとに作成されます。

属性	記述
例外 ID	例外を一意的に識別します。admin ロールを持つユーザーのみが使用できます。
例外タイプ ID	例外タイプを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
例外タイム・スタンプ	この「例外」エンティティーがログに記録されるときに作成された日時。
例外の日付	そのタイム・スタンプの日付のみ。
例外の時刻	そのタイム・スタンプの時刻のみ。
例外の曜日	そのタイム・スタンプの曜日のみ。
例外の年	そのタイム・スタンプの年のみ。
ソース・アドレス	例外のソース IP アドレス。
ソース・ポート	ソース・ポート番号。

属性	記述
宛先アドレス	宛先 IP アドレス。
宛先ポート	宛先ポート番号。
データベース・プロトコル	例外のデータベース・プロトコル。
新規 TTL 値	admin ロール専用に予約済み。
例外の記述	例外の記述。 S-TAP 再接続またはタイムアウト例外の場合、これにはデータベース・サーバーの IP アドレスまたは DNS 名が含まれます。 データベース例外の場合、これはデータベース管理システムからのエラー・コードです。 最も一般的なメッセージ (メッセージのうち約 54,000) については、「データベース・エラー・テキスト」属性で、より長いテキストの記述を使用できます。 そのテキストは、例外自体からではなく、エラー・メッセージのための内部 Guardium® データベース表からのものです。
例外の原因となった SQL 文字列	例外を発生させた SQL 文字列。
ユーザー名	データベース・ユーザー名。相関が必要とされる暗号化トラフィックではこの値を使用できない場合がありますが、「クライアント/サーバー」エンティティーの「データベース・ユーザー名」属性からは常に使用可能です。
アプリケーション・ユーザー名	アプリケーション・ユーザー名。
例外についての詳細情報へのリンク 1	例外ソースによっては使用可能な場合があるオプション・リンク。
グローバル ID1	例外のグローバル ID。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。

「データベース・エラー・テキスト」エンティティー

データベースの各一般的エラー・メッセージのテキストは、Guardium 内部データベースの表に格納されています。これは、所有する「例外」エンティティーから、データベース・エラーである各例外のレポートを作成する場合にのみ使用できます。例外のいくつかのタイプ (例えば、S-TAP 接続または再接続) には、データベース・エラー・テキストはありません。

属性	記述
データベース・エラー・テキスト	データベース・エラー・コードの後に、エラーに関する短いテキストの記述が続きます。 エラー・コードは、「例外」エンティティーの「例外の記述」属性から取得されます。 エラー・コードをキーとして使用して、Guardium アプライアンス上の内部表からエラー・テキストが取得されます。この内部表には、最も一般的なエラー・メッセージ (エラー・メッセージのうち約 54,000) が含まれます。 例: ORA-00942: 表またはビューが存在しません
エラー・コード	データベース・エラー・コードを表示します。

親トピック: ドメインのエンティティーおよび属性

「FAM」ドメイン: エンティティーおよび属性

ディスカバーされたファイルに関するメタデータ。

使用可能なロール: admin、fam。

「FAM ファイル」エンティティー

属性	記述
分類	FAM 分類からの分類。
コンテンツ・タイプ	コンテンツ・タイプ
作成時刻	作成時刻
削除グループ	ファイルに対する削除権限を持つグループ。
削除ユーザー	ファイルに対する削除権限を持つユーザー。
有効な終了時刻	
実行グループ	ファイルに対する実行権限を持つグループ。
実行ユーザー	ファイルに対する実行権限を持つユーザー。
ファイルの絶対パス名	ファイルの絶対パス名
ファイル ID	ファイル ID
ファイル ID	ファイル ID

属性	記述
ファイル名	ファイル名
ファイル・サーバー	ファイル・サーバー
削除済み	ファイルがサーバーから削除されたかどうか。True/False。
シンボリック	ファイルがシンボリック・リンクであるかどうか。True/False。
変更時刻	変更時刻
オペレーティング・システム	オペレーティング・システム
所有者	所有者
親ディレクトリー ID	親ディレクトリー ID
親ディレクトリー・パス	親ディレクトリー・パス
読み取りグループ	ファイルに対する読み取り権限を持つグループ。
読み取りユーザー	ファイルに対する読み取り権限を持つユーザー。
スキャン ID	スキャンの固有 ID。
スキャン時刻	ファイルがスキャンされた時刻。
サイズ	サイズ
ソース・ディレクトリー ID	ソース・ディレクトリーの固有 ID。
ソース・ディレクトリー・パス	ソース・ディレクトリーの固有のパス。
タイム・スタンプ	Guardium でファイルが記録された時刻。
書き込みグループ	ファイルに対する書き込み権限を持つグループ。
書き込みユーザー	ファイルに対する書き込み権限を持つユーザー。
hostName	ファイルが配置されているファイル・サーバー。
SG ID グループ	
SG ID ユーザー	
SU ID グループ	
SU ID ユーザー	

FAM 分類エンティティー

属性	記述
カテゴリー	判定プランの名前。
分類 ID	固有の分類 ID。
エンティティー	機密データのタイプ。
ファイル ID	
スキャン ID	
タイム・スタンプ	Guardium でファイルがディスカバーされて記録された時刻。

親トピック: ドメインのエンティティーおよび属性

「FAM システム」ドメイン: エンティティーおよび属性

「FAM 状況」エンティティー

属性	記述
分類判定プラン	分類のカテゴリーおよびルール。例: HIPAA{HIPAA_match,CreditCard,Name}:PCI{PCI_match}
分類スレッドの数	ICM のシステム・パラメーター。
クローラーの最大深さ	スキャンのディスカバリーおよび分類のディレクトリーの深さ。
除外されたディレクトリー・パス	ディスカバリーおよび分類のプロセスから除外するディレクトリーのリスト。
除外されたファイル・パス	ディスカバリーおよび分類のプロセスから除外するファイルのリスト。
ファイル・クローラーのホスト	ファイル・サーバーのホスト名。
ファイル・クローラーの IP	ファイル・サーバーの IP。
ICM URL	システム・パラメーター。分類は、ローカル・サーバーで実行されます。通常は、http://localhost:18087 です。

属性	記述
コンテンツ・ディスカバリーがアクティブである	True - 内容に基づいてファイルの分類を有効にします。 False – メタデータおよびアクセス許可の抽出のみを行います。
クローラーがアクティブである	0: FAM ディスカバリー・エージェントは無効化されます。 1: FAM ディスカバリー・エージェントは有効化されます。 2: FAM ディスカバリー・エージェントは再始動されます。
スケジューラーの時間間隔	スキャンの間隔 (時間)。
スケジューラーが繰り返しである	True: スキャンを繰り返します False: スキャンを繰り返しません
スケジューラーの分間隔	スキャンの間隔 (分)。
スケジューラー開始時刻	スキャンのアクティブ化の時刻。
サーバー IP	Guardium サーバー IP。
ソース・ディレクトリー・パス	スキャンを実行するディレクトリー・パス。例: /home/ab
状況 ID	内部 ID。
タイム・スタンプ	スキャンが開始された時刻。

親トピック: ドメインのエンティティーおよび属性

「ファイル・アクティビティー・モニター」ドメイン: エンティティーおよび属性

「スキャン」エンティティー

属性	記述
ホスト IP	ファイル・サーバーのホスト IP。
ホスト名	ファイル・サーバーのホスト名。
OS	オペレーティング・システム。
スキャン時刻	スキャンの開始時刻。
ソース・ディレクトリー ID	ソース・ディレクトリー ID。
ソース・ディレクトリー・パス	ソース・ディレクトリー・パス。
タイム・スタンプ	データが Guardium にアップロードされた時刻。

「ディレクトリー」エンティティー

属性	記述
作成時刻	ディレクトリーの作成時刻。
削除グループ	このディレクトリーに対する削除権限を持つグループのリスト。
削除ユーザー	このディレクトリーに対する削除権限を持つユーザーのリスト。
ディレクトリー	使用されません。
ディレクトリー ID	ディレクトリー ID
ディレクトリー名	ディレクトリー名
ディレクトリー・パス	ディレクトリー・パス
表示サイズ	UI でのディレクトリーのサイズ。
有効な終了時刻	内部パラメーター。
実行グループ	このディレクトリーに対する実行権限を持つグループのリスト。
項目の許可	使用されません。
項目タイプ	使用されません。
変更時刻	ディレクトリーの変更時刻。
所有者	ディレクトリー所有者。
親ディレクトリー ID	親ディレクトリー ID。
読み取りグループ	このディレクトリーに対する読み取り権限を持つグループのリスト。
読み取りユーザー	このディレクトリーに対する実行権限を持つグループのリスト。
スキャン時刻	このディレクトリーでスキャンが開始された時刻。
サイズ	ディレクトリーのサイズ。
タイム・スタンプ	ディレクトリーの詳細を Guardium アプライアンスにアップロードした時刻。

属性	記述
書き込みグループの書き込みユーザー	このディレクトリーに対する書き込み権限を持つユーザーおよびグループのリスト。
削除済み	このディレクトリーが最終スキャンから削除されたかどうか。

「ファイル」エンティティー

属性	記述
分類	検出された分類のカテゴリーおよびエンティティー。
コンテンツ・タイプ	ファイルのコンテンツ・タイプ。
作成時刻	ファイルの作成時刻。
削除グループ	このディレクトリーに対する削除権限を持つグループのリスト。
削除ユーザー	このディレクトリーに対する削除権限を持つユーザーのリスト。
ディレクトリー ID	ディレクトリー ID。
ディレクトリー・パス	ファイル・ディレクトリー・パス。
表示サイズ	UI でのディレクトリーのサイズ。
有効な終了時刻	内部パラメーター。
実行グループ	このディレクトリーに対する実行権限を持つグループのリスト。
実行ユーザー	このディレクトリーに対する実行権限を持つユーザーのリスト。
FAM ファイル ID	ファイル ID。
ファイルの絶対パス名	ファイルの絶対パス名 (絶対パスおよびファイル名)。
ファイル ID	ファイル ID。
ファイル名	ファイル名。
削除済み	このファイルが最新スキャンから削除されたかどうか。
シンボリック	これがシンボリック・リンクであるかどうか。
項目の許可	使用されません。
項目タイプ	使用されません。
変更時刻	ファイルの変更時刻。
所有者	ファイル所有者。
読み取りグループ	このディレクトリーに対する読み取り権限を持つグループのリスト。
読み取りユーザー	このディレクトリーに対する読み取り権限を持つユーザーのリスト。
スキャン時刻	ファイルがスキャンされた時刻。
サイズ	ファイルのサイズ。
タイム・スタンプ	ファイルの詳細が Guardium アプライアンスにアップロードされた時刻。
書き込みグループ	このディレクトリーに対する書き込み権限を持つグループのリスト。
書き込みユーザー	このディレクトリーに対する書き込み権限を持つユーザーのリスト。

「分類」エンティティー

属性	記述
カテゴリー	分類のカテゴリー。
エンティティー	分類のカテゴリーのルール。
ファイル ID	ファイル ID。
タイム・スタンプ	分類が Guardium システムに追加された時刻。

親トピック: ドメインのエンティティーおよび属性

「未解析ログ」ドメイン: エンティティーおよび属性

未解析ログ処理アクティビティー。

使用可能なロール: すべて

「クライアント/サーバー」エンティティー

このエンティティーは、特定のクライアント/サーバー接続について示します。固有の属性セット (タイム・スタンプを除く) が検出されるたびにインスタンスが作成されます。

注:「アクセスのトラッキング」の場合のみ、「クライアント/サーバー」エンティティー名は、プルダウン・メニューに 2 つの可能なエンティティー (「クライアント/サーバー」および「セッション別クライアント/サーバー」) として表示されます。

「セッション別クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「セッション」から日付状態を取得します。

「クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「クライアント/サーバー」から日付状態も取得します。

ユーザーが「クライアント/サーバー」を選択すると、照会には ATTRIBUTE_ID = 1 が設定されます。ユーザーが「セッション別クライアント/サーバー」を選択すると、照会には MAIN_ATTRIBUTE_ID = 0 が設定されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
分析されたクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
クライアント・ホスト名	クライアント・ホスト名。
クライアント IP	クライアントの IP アドレス。
クライアント IP/データベース・ユーザー	クライアント IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名	タプル・グループ
クライアント IP/ソース・アプリケーション/ユーザー	タプル・グループ
クライアント MAC	クライアントのハードウェア・アドレス。
クライアント OS	クライアントのオペレーティング・システム。 Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
データベース・プロトコル	データベース・サーバー固有のプロトコル。
データベース・プロトコル・バージョン	データベース・プロトコルのプロトコル・バージョン。
データベース・ユーザー名	データベース・ユーザー名: ローカルまたはリモートのデータベースに接続したユーザー。
最終使用日	
ネットワーク・プロトコル	使用されたネットワーク・プロトコル (例えば TCP、UDP など。Oracle 上の K-TAP については IPC または BEQ として表示されることに注意してください)。
OS ユーザー	相互作用の OS ユーザー・アカウント。
サーバーの記述	サーバーの記述 (ある場合)。
サーバー・ホスト名	サーバー・ホスト名。
サーバー IP	サーバーの IP アドレス。
サーバー IP/データベース・ユーザー	サーバー IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
サーバー IP/サービス名/データベース・ユーザー	タプル・グループ

属性	記述
サーバー OS	サーバーのオペレーティング・システム。 Informix の場合、OS が次のように表示される場合があります。 IEEEM (Unix または JDBC を表します) IEEEI (Windows を表します) DEC (DEC Alpha を表します) Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
サーバー・タイプ	DB2、Oracle、Sybase など。
サービス名	相互作用のサービス名。場合によっては (例えば AIX 共有メモリー接続)、別名がサービス名になり、実際のサービスが接続されるまで別名が使用されます。このような場合、実際のサービスが接続された時点で、新しいセッションが開始されます。したがって、ユーザーには単一セッションと認識されるものが 2 つのセッションとしてログに記録されることになります。 Teradata の場合、サービス名にはセッション論理ホスト ID 値が含まれます。
ソース・プログラム	相互作用のソース・プログラム。
セッション別クライアント/サーバー	セッション別クライアント/サーバーは、メイン・エンティティーでもあります。 この「2 次エンティティー」にアクセスするには、「1 次エンティティー」である「クライアント/サーバー」をクリックします。
タイム・スタンプ	このエンティティーのすべての属性が静的情報を持つので、このタイム・スタンプは、定義済みクライアント/サーバー接続上で Guardium が要求を初めて確認したときに、1 回だけ作成されます。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。

「セッション」エンティティー

このエンティティーは、クライアント/サーバー・データベース・セッションごとに作成されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
クライアント・ポート	クライアント・ポート番号。
データベース名	セッションの対象データベースの名前 (MSSQL または Sybase のみ)。 Oracle の場合、アプリケーション固有の追加情報が「データベース名」に含まれることがあります (V$SESSION ビューの MODULE 列に設定された、セッション用に現在実行中のモジュールなど)。
期間 (秒)	セッション開始からセッション終了までの時間の長さを示します (秒単位)。
グローバル ID	セッション・アクセスを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
これ以降を無視	このセッションを無視し始めたときに作成されたタイム・スタンプ。
非アクティブ・フラグ	0 (デフォルト) - オープン (SQL パッケージによって生成されたセッションの場合)。 1 - クローズ (切断/ログアウト受信)。 2 - おそらくクローズ (長時間にわたってパケットがない状態では非クローズ)。 3 - 非 SQL パケットから生成されたセッションの場合。
古いセッション ID	このセッションが作成されたセッションを示します。 これが接続の最初のセッションである場合は、ゼロです。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
プロセス ID	接続を開始したクライアントのプロセス ID (常に使用可能とは限らない)。
サーバー・ポート	サーバー・ポート番号。

属性	記述
セッション終了	セッションが終了した日時。「セッション終了」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション終了の日付	「セッション終了」の中の日付のみ。
セッション終了の時刻	「セッション終了」の中の時刻のみ。
セッション終了の曜日	「セッション終了」の中の曜日のみ。
セッション終了の年	「セッション終了」の中の年のみ。
セッション ID	セッションを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
セッション無視	セッションの一部が (ある時点から) 無視されたかどうかを示します。
セッション開始	セッションが開始された日時。「セッション開始」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション開始の日付	「セッション開始」の中の日付のみ。
セッション開始の時刻	「セッション開始」の中の時刻のみ。
セッション開始の曜日	「セッション開始」の中の曜日のみ。
セッション開始の年	「セッション開始」の中の年のみ。
端末 ID	セッション情報を解決するために内部で使用された、接続の端末 ID。
タイム・スタンプ	初めは、進行中のアクティブ・セッションのないクライアント/サーバー接続上の最初の要求に対して、タイム・スタンプが作成されます。 その後、セッションが閉じられたとき、または長期間にわたってアクティビティーが監視されないために非アクティブのマークがセッションに付けられたときに、更新されます。 セッション情報をトラッキングする場合は、「タイム・スタンプ」属性よりも「セッション開始」属性と「セッション終了」属性を注目する方が適切な場合があります。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
TTL	admin ロール専用に予約済み。
Uid チェーン	Unix S-TAP (K-Tap モードのみ) によってレポートされたセッションが対象。su ユーザーのユーザー名が異なる場合に、OS ユーザーのチェーンを示します。 ここに表示される値は、OS プラットフォームによって異なります。例えば AIX 環境では、接頭部として IBM IBM IBM という文字列が表示される場合があります。 注: Solaris ゾーンの場合は、ユーザー名の代わりにユーザー ID が「UID チェーン」でレポートされる場合があります。
Uid チェーン圧縮	圧縮された値。「Uid」チェーンを参照してください。

「未解析ログ」エンティティー

このエンティティーは、未解析ログ処理アクティビティーについて示します。

属性	記述
完全な SQL	ログに記録された完全な SQL。
タイム・スタンプ	ログに記録されたときの日時スタンプ。
タイム・スタンプの日付	タイム・スタンプの日付部分。
タイム・スタンプの時刻	タイム・スタンプの時刻部分。
応答時間	要求に対する応答時間 (ミリ秒単位)。
影響を受けるレコード	要求の影響を受けたレコードの数。
成功	要求が成功したかどうかを示します (True/False)。
ステートメント・タイプ	SQL ステートメントのタイプ。 SQL: 単純な直接 SQL コマンド (例えば、CLI に直接入力されるコマンド) RAW: 後で実行するための SQL ステートメント PREPARE。例えば、conn.prepareStatement (select a from b where c=:value) BIND: バインドされたパラメーター値を含む、準備されたステートメントの実行 ステートメント・タイプは「完全な SQL」エンティティーの一部であり、ポリシー内でこのステートメントに対して「全詳細をロギング」を構成した場合にのみ監査されます。 ポリシー内の特定のステートメント・タイプ (例えば、監査のみの SQL および BIND ステートメント) をフィルタリングすることはできません。ただし、レポートではこれらをフィルタリングできます。
戻りデータ	返されたデータ (ある場合)。
バインド情報	要求のバインド情報。
バインド変数値	Db2/zOS の場合は、バインド変数のコンマ区切りリストが含まれます。

属性	記述
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。

親トピック: ドメインのエンティティーおよび属性

「GIM クライアント」ドメイン: エンティティーおよび属性

使用可能なロール: admin。

「GIM クライアント」エンティティー

属性	記述
GIM クライアント IP	クライアントの IP アドレス
GIM クライアント名	クライアント・ホスト名
GIM クライアント OS	クライアント・オペレーティング・システム (例えば、Linux)
GIM クライアント OS ベンダー	オペレーティング・システム・ベンダー (例えば、Redhat)
GIM クライアント OS ベンダー・バージョン	オペレーティング・システム・カーネル・バージョン (例えば、2.6.32-642.13.1.el6.x86_64)
GIM クライアント状態タイム・スタンプ	GIM クライアントが作成された時刻

「GIM クライアント・モジュール状態」エンティティー

属性	記述
接続先	このモジュールが接続されているアプライアンス
GIM モジュール名	モジュール名 (例えば、「GIM」)
GIM モジュールの状態	モジュールの状態 (例えば、「INSTALLED」、「FAILED」)
GIM モジュールのバージョン	モジュールの Guardium のバージョン
スケジュール済み	モジュールのインストール/アップグレード/アンインストールがスケジュールに入っているかどうか。0 – スケジュールに入っていない、 1 - スケジュールに入っている。

「GIM モジュール・ハートビート」エンティティー

属性	記述
GIM クライアント ID	クライアントの主キー (GIM_CLIENTS)
GIM モジュール ID	モジュールの主キー (GIM_MODULES)
GIM モジュール名	モジュール名
最終更新日時	最後のアライブ時刻 (ミリ秒)
状態	モジュールの状態 (UP/DOWN)

親トピック: ドメインのエンティティーおよび属性

「GIM イベント」ドメイン: エンティティーおよび属性

使用可能なロール: すべて

「GIM イベント」エンティティー

このエンティティーは、Guardium Installation Manager (GIM) を使用中に発生したイベントについて示します。

属性	記述
イベント・ジェネレーター	イベントを生成したクライアント (例えば、DB サーバー) の IP アドレス。
イベントの記述	イベントの記述。
イベント時間	イベントが発生した時刻。

親トピック: ドメインのエンティティーおよび属性

「グループ」ドメイン: エンティティーおよび属性

Guardium グループのメンバーシップ。

使用可能なロール: すべて

「グループ・タイプ」エンティティー

このエンティティーは、Guardium グループのタイプ (ユーザー、クライアント IP アドレス、コマンドなど) について示します。

属性	記述
グループ・タイプ	グループ・タイプを識別します。
タイム・スタンプ	グループ・タイプが作成された日時。

「グループ」エンティティー

このエンティティーは、Guardium に対して定義されたグループについて示します。

属性	記述
グループの記述	グループの名前。
グループ・サブタイプ	グループに定義されたサブタイプ (ある場合)。
タイム・スタンプ	グループ・エンティティーが作成された日時。

「グループ・メンバー」エンティティー

このエンティティーは、Guardium に対して定義されたグループのメンバーについて示します。

属性	記述
グループ・メンバー	グループ・メンバーの名前。
タイム・スタンプ	グループ・メンバーが作成または更新された日時。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。

親トピック: ドメインのエンティティーおよび属性

「保護プロセス・ログ」ドメイン: エンティティーおよび属性

Guardium で実行されているプロセスのログ。

使用可能なロール: admin。

「保護プロセス・ログ」エンティティー

属性	記述
コメント	
ログ・メッセージ	
プロセス名	
状況	
TIMESTAMP	

親トピック: ドメインのエンティティーおよび属性

「Guardium アクティビティー」ドメイン: エンティティーおよび属性

Guardium エンティティーに対して Guardium ユーザーが行ったすべての変更 (レポートまたは照会の定義または変更)。

使用可能なロール: admin

Guardium アクティビティー・タイプ

このエンティティーは、さまざまなユーザー・アクティビティーについて示します。

属性	記述
アクティビティー・タイプの記述	アクティビティーの記述。
アクティビティー・タイプ ID	アクティビティー・タイプを一意的に識別します。

「Guardium ユーザー・アクティビティー監査」エンティティー

このエンティティーは、Guardium ユーザー・アクティビティーごとに作成されます。

属性	記述
ログイン ID	ログインに使用された ID。
ユーザー名	アクティビティーに使用された Guardium® ユーザー名。
タイム・スタンプ	アクティビティーがログに記録されるときに作成されたもの。
変更エンティティー	変更された Guardium エンティティー (例えば、グループ定義)。
使用エンティティー・キー	エンティティーにアクセスするために使用されたキー。
キー値	エンティティーの新規の値。
すべての値	変更されたすべての値。
オブジェクトの記述	変更された特定のオブジェクトの名前。
グローバル ID	セッションの固有のグローバル ID。
ホスト名	ユーザーのホスト名。

親トピック: ドメインのエンティティーおよび属性

「Guardium ジョブ・キュー」ドメイン: エンティティーおよび属性

使用可能なロール: admin。

「Guardium ジョブ・キュー」エンティティー

属性	記述
終了時刻	
Guardium ジョブの記述	
プロセス ID	
プロセス実行 ID	
プロセス・タイプ	
キュー時刻	
レポート結果 ID	
開始時刻	
状況	
タスクの記述	
タイム・スタンプ	
CLS_LOG_TYPE_DESC	
詳細	
メッセージ	

Guardium 分類ログ

属性	記述
データ・ソース	
終了時刻	
Guardium ジョブの記述	
プロセス ID	
プロセス実行 ID	
プロセス・タイプ	
キュー時刻	
レポート結果 ID	
開始時刻	
状況	
タスクの記述	
タイム・スタンプ	
CLS_LOG_TYPE_DESC	

属性	記述
詳細	
メッセージ	

親トピック: ドメインのエンティティーおよび属性

「Guardium ログイン」ドメイン: エンティティーおよび属性

Guardium ユーザーのログインとログアウトに関する全情報。

使用可能なロール: admin

「Guardium ユーザー・ログイン」エンティティー

このエンティティーは、Guardium アプライアンスにユーザーがログインするたびに作成されます。

属性	記述
ログイン ID	ログインに使用された ID。
ユーザー名	Guardium® ユーザーがログインまたはログアウトするときに作成されます (Guardium セッション当たり 1 つのエンティティーが存在します)。
ログイン日時	ユーザーがログインした日時。
ログアウト日時	ユーザーがログアウトした日時。
ログイン成功	ログインが成功したかどうかを示します。
グローバル ID	セッションの固有のグローバル ID。
ホスト名	ユーザーのホスト名。
リモート・アドレス	ユーザーのリモート・アドレス。

親トピック: ドメインのエンティティーおよび属性

「IMS イベント」ドメイン: エンティティーおよび属性

使用可能なロール: すべて。

「クライアント/サーバー」エンティティー

このエンティティーは、特定のクライアント/サーバー接続について示します。固有の属性セット (タイム・スタンプを除く) が検出されるたびにインスタンスが作成されます。

注: 「アクセスのトラッキング」の場合のみ、「クライアント/サーバー」エンティティー名は、プルダウン・メニューに 2 つの可能なエンティティー (「クライアント/サーバー」および「セッション別クライアント/サーバー」) として表示されます。

「セッション別クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「セッション」から日付状態を取得します。

「クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「クライアント/サーバー」から日付状態も取得します。

ユーザーが「クライアント/サーバー」を選択すると、照会には ATTRIBUTE_ID = 1 が設定されます。ユーザーが「セッション別クライアント/サーバー」を選択すると、照会には MAIN_ATTRIBUTE_ID = 0 が設定されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
分析されたクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
クライアント・ホスト名	クライアント・ホスト名。
クライアント IP	クライアントの IP アドレス。
クライアント IP/データベース・ユーザー	クライアント IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名	タプル・グループ
クライアント IP/ソース・アプリケーション/ユーザー	タプル・グループ
クライアント MAC	クライアントのハードウェア・アドレス。

属性	記述
クライアント OS	クライアントのオペレーティング・システム。 Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
データベース・プロトコル	データベース・サーバー固有のプロトコル。
データベース・プロトコル・バージョン	データベース・プロトコルのプロトコル・バージョン。
データベース・ユーザー名	データベース・ユーザー名: ローカルまたはリモートのデータベースに接続したユーザー。
最終使用日	
ネットワーク・プロトコル	使用されたネットワーク・プロトコル (例えば TCP、UDP など。Oracle 上の K-TAP については IPC または BEQ として表示されることに注意してください)。
OS ユーザー	相互作用の OS ユーザー・アカウント。
サーバーの記述	サーバーの記述 (ある場合)。
サーバー・ホスト名	サーバー・ホスト名。
サーバー IP	サーバーの IP アドレス。
サーバー IP/データベース・ユーザー	サーバー IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
サーバー IP/サービス名/データベース・ユーザー	タプル・グループ
サーバー OS	サーバーのオペレーティング・システム。 Informix の場合、OS が次のように表示される場合があります。 IEEEM (Unix または JDBC を表します) IEEEI (Windows を表します) DEC (DEC Alpha を表します) Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
サーバー・タイプ	DB2、Oracle、Sybase など。
サービス名	相互作用のサービス名。場合によっては (例えば AIX 共有メモリー接続)、別名がサービス名になり、実際のサービスが接続されるまで別名が使用されます。このような場合、実際のサービスが接続された時点で、新しいセッションが開始されます。したがって、ユーザーには単一セッションと認識されるものが 2 つのセッションとしてログに記録されることになります。 Teradata の場合、サービス名にはセッション論理ホスト ID 値が含まれます。
ソース・プログラム	相互作用のソース・プログラム。
セッション別クライアント/サーバー	セッション別クライアント/サーバーは、メイン・エンティティーでもあります。 この「2 次エンティティー」にアクセスするには、「1 次エンティティー」である「クライアント/サーバー」をクリックします。
タイム・スタンプ	このエンティティーのすべての属性が静的情報を持つので、このタイム・スタンプは、定義済みクライアント/サーバー接続上で Guardium が要求を初めて確認したときに、1 回だけ作成されます。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。

「セッション」エンティティー

このエンティティーは、クライアント/サーバー・データベース・セッションごとに作成されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
クライアント・ポート	クライアント・ポート番号。
データベース名	セッションの対象データベースの名前 (MSSQL または Sybase のみ)。 Oracle の場合、アプリケーション固有の追加情報が「データベース名」に含まれることがあります (V$SESSION ビューの MODULE 列に設定された、セッション用に現在実行中のモジュールなど)。
期間 (秒)	セッション開始からセッション終了までの時間の長さを示します (秒単位)。
グローバル ID	セッション・アクセスを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
これ以降を無視	このセッションを無視し始めたときに作成されたタイム・スタンプ。
非アクティブ・フラグ	0 (デフォルト) - オープン (SQL パッケージによって生成されたセッションの場合)。 1 - クローズ (切断/ログアウト受信)。 2 - おそらくクローズ (長時間にわたってパケットがない状態では非クローズ)。 3 - 非 SQL パケットから生成されたセッションの場合。
古いセッション ID	このセッションが作成されたセッションを示します。 これが接続の最初のセッションである場合は、ゼロです。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
プロセス ID	接続を開始したクライアントのプロセス ID (常に使用可能とは限らない)。
サーバー・ポート	サーバー・ポート番号。
セッション終了	セッションが終了した日時。「セッション終了」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション終了の日付	「セッション終了」の中の日付のみ。
セッション終了の時刻	「セッション終了」の中の時刻のみ。
セッション終了の曜日	「セッション終了」の中の曜日のみ。
セッション終了の年	「セッション終了」の中の年のみ。
セッション ID	セッションを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
セッション無視	セッションの一部が (ある時点から) 無視されたかどうかを示します。
セッション開始	セッションが開始された日時。「セッション開始」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション開始の日付	「セッション開始」の中の日付のみ。
セッション開始の時刻	「セッション開始」の中の時刻のみ。
セッション開始の曜日	「セッション開始」の中の曜日のみ。
セッション開始の年	「セッション開始」の中の年のみ。
端末 ID	セッション情報を解決するために内部で使用された、接続の端末 ID。
タイム・スタンプ	初めは、進行中のアクティブ・セッションのないクライアント/サーバー接続上の最初の要求に対して、タイム・スタンプが作成されます。 その後、セッションが閉じられたとき、または長期間にわたってアクティビティーが監視されないために非アクティブのマークがセッションに付けられたときに、更新されます。 セッション情報をトラッキングする場合は、「タイム・スタンプ」属性よりも「セッション開始」属性と「セッション終了」属性を注目する方が適切な場合があります。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
TTL	admin ロール専用に予約済み。
Uid チェーン	Unix S-TAP (K-Tap モードのみ) によってレポートされたセッションが対象。su ユーザーのユーザー名が異なる場合に、OS ユーザーのチェーンを示します。 ここに表示される値は、OS プラットフォームによって異なります。例えば AIX 環境では、接頭部として IBM IBM IBM という文字列が表示される場合があります。 注: Solaris ゾーンの場合は、ユーザー名の代わりにユーザー ID が「UID チェーン」でレポートされる場合があります。
Uid チェーン圧縮	圧縮された値。「Uid」チェーンを参照してください。

「アクセス期間」エンティティー

アクセス期間はセッションに関連します。デフォルトではアクセス期間の長さは 1 時間ですが、Guardium 管理者は「検査エンジン構成」で変更することができます (「ロギング単位」に該当)。

属性	記述
アプリケーション・イベント ID	アプリケーション・イベント ID (API から設定された場合)。照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。
実行確認応答時間の平均	実行確認応答時間の平均 (ミリ秒単位)。
影響を受けるレコードの平均 (2)	影響を受けたレコードの平均数。照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。
アプリケーション・ユーザー	アプリケーション・ユーザー名。
平均実行時間	期間中の平均コマンド実行時間。SQL ステートメントのみが対象です。FTP トラフィックには適用されません。
構造 ID	コマンド構造 (例えば select a from b) を一意的に識別します。admin ロールを持つユーザーのみが使用できます。
失敗した SQL (2)	失敗した SQL 要求の数。表の最後に記載されている注を参照してください。 照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。
インスタンス ID	構造のインスタンスを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
元のタイム・ゾーン	UTC オフセット。 これは、2 つの異なるタイム・ゾーンに存在する 2 つの異なるコレクターの時間を、正しく統合するために設定する必要がある UTC オフセットを示すものです。 オフセットを設定しなかった場合、物事の発生時刻をユーザーが判別したり、正確な表記で参照したりできないという状況が存在していまいます。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
期間の終了	アクセス期間の終了の日時。
期間の終了日	期間終了属性の中の日付のみ。
期間の終了時刻	期間終了属性の中の時刻のみ。
期間の終了曜日	期間終了属性の中の曜日のみ。
期間の開始日	期間開始属性の中の日付のみ。
期間の開始時刻	期間開始属性の中の時刻のみ。
期間の開始曜日	期間開始属性の中の曜日のみ。
セッション ID	セッションを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
秒を表示	1 秒当たりのアクセス数がトラッキングされている場合、アクセス期間内 (通常は 1 時間) の秒ごとのカウントがここに含まれています。
成功した SQL (2)	成功した SQL 要求の数。表の最後に記載されている注を参照してください。 照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。
タイム・スタンプ	初めは、アクセス期間中にクライアント/サーバー接続上で要求が初めて確認されたときに、タイム・スタンプ値が設定されます。 デフォルトではアクセス期間の長さは 1 時間ですが、Guardium 管理者は「検査エンジン構成」で変更することができます (「Guardium 管理者ガイド」を参照)。その後は、後続の要求ごとに、その期間の平均実行時間とコマンド数が更新されるときに更新されます。
アクセス合計	このアクセス期間における構造インスタンスの総数。admin ロールを持つユーザーのみが使用できます。
影響を受けるレコード合計 (2)	影響を受けたレコードの総数。照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。

属性	記述
影響を受けるレコード合計 (名前) (2)	「影響を受けるレコード合計」属性が数値ではなく文字列の場合、その値はここに表示されます (例えば、「大規模結果セット」や N/A)。照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、使用できません。 「影響されるレコード」- SQL ステートメントを実行するたびに影響を受けるレコードの数を示す結果セット。 注: 「影響されるレコード」オプションは、スニファーに対して、追加の応答パケットを処理し、影響を受けたデータ (バッファー・サイズを増やし、スニファー全体のパフォーマンスに悪影響を及ぼす可能性があるデータ) のロギングを延期するように要求するスニファー操作です。 大きな影響を受けるのは、非常に大きい応答からです。この操作に関連する大量のオーバーヘッドを避けるために、Guardium はデフォルトのしきい値セットを使用して、しきい値を超えたときに、処理操作をスキップすることをスニファーが決定できるようにします。 CLI コマンド store max_results_set_size、store max_result_set_packet_size、および store max_tds_response_packets を使用して、細分度のレベルを設定することができます。 結果セットの値の例は次のとおりです。 • ケース 1、「影響されるレコード」値: 正数 - これは、結果セットの正しいサイズを表します。 • ケース 2、「影響されるレコード」値: -2 - これは、レコード数が構成可能な限度 (CLI インターフェースによって調整可能) を超えたことを示します。 • ケース 3、「影響されるレコード」値: -1 - これは、Guardium によってサポートされないパケット構成のケースを示します。 • ケース 4、「影響されるレコード」値: -2 - 結果セットがストリーム・モードで送信される場合。 • ケース 5、「影響されるレコード」値: -2 - ユーザーを現在の値について更新するためのレコードのカウント中の中間結果。最終的には、レコードの合計を示す正数になります。

「SQL」エンティティー

このエンティティーは、SQL の固有文字列ごとに作成されます。値は疑問符 (?) に置き換えられ、文字列のフォーマットのみが保管されます。

属性	記述
バインド情報	この SQL 文字列のバインド情報。
構造 ID	SQL が出現する構造を一意的に識別します。
SQL	SQL 文字列。
切り捨てられた SQL	SQL が切り捨てられたかどうかを示します。値は次のとおりです。 0 - false/いいえ (切り捨てられていません) 1 - true/はい (切り捨てられました)

「完全な SQL」エンティティー

「完全な SQL」エンティティーは、ポリシー・ルール・アクションの「全詳細をロギング」、「値を含む全詳細をロギング」、「セッションごとに全詳細をロギング」、または「値を含む全詳細をセッションごとにロギング」によってのみ作成されます。

属性	記述
アクセス・ルールの記述	使用されたポリシー・ルールの記述。
確認応答時間	確認応答時間 (ミリ秒単位)。
自動コミット	項目が自動的に番号付けされます。
バインド変数値	Db2/zOS の場合は、バインド変数のコンマ区切りリストが含まれます。
退出 KB カウント	応答に含まれるバイト数を記録します。
完全な SQL	値を含む完全な SQL ステートメント。
完全な SQL ID	完全な SQL の固有 ID。admin ロールを持つユーザーのみが使用できます。
進入 KB カウント	要求に含まれるバイト数を記録します。
インスタンス ID	完全な SQL のインスタンスの固有 ID。admin ロールを持つユーザーのみが使用できます。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
影響を受けるレコード	影響を受けたレコードの数 (セッションごと)。この属性を使用するレポートでは、大規模結果セットや N/A などの特殊ケースが適切に表示されるように、別名をオンにすることをお勧めします。
影響を受けるレコード (名前)	「影響されるレコード」が数値ではなく文字列値である場合、その文字列はここに保管されます。例: 大規模結果セットまたは N/A。
応答時間	要求に対する応答時間 (ミリ秒単位)。ネットワーク・トラフィック内で要求がモニターされる場合、応答時間は要求に応答するのに要した時間を正確に反映しています (Guardium はクライアント要求とサーバー応答の両方のタイム・スタンプを設定します)。

属性	記述
戻りデータ	この要求に対して返されたデータ (ある場合、かつ使用可能な場合)。
戻りデータ・カウント	ポリシー・ルールで使用された SQL ステートメントから返された行数。
ステートメント・タイプ	SQL ステートメントのタイプ。 SQL: 単純な直接 SQL コマンド (例えば、CLI に直接入力されるコマンド) RAW: 後で実行するための SQL ステートメント PREPARE。例えば、conn.prepareStatement (select a from b where c=:value) BIND: バインドされたパラメーター値を含む、準備されたステートメントの実行 ステートメント・タイプは「完全な SQL」エンティティーの一部であり、ポリシー内でこのステートメントに対して「全詳細をロギング」を構成した場合にのみ監査されます。 ポリシー内の特定のステートメント・タイプ (例えば、監査のみの SQL および BIND ステートメント) をフィルタリングすることはできません。ただし、レポートではこれらをフィルタリングできます。
成功	呼び出しが成功したかどうかを示します。admin ロールを持つユーザーのみが使用できます。
タイム・スタンプ	タイム・スタンプは、SQL がデータベース・サーバーで実行されるときに、時刻を記録します。

「完全な SQL 値」エンティティー

これらのエンティティーは、「値を含む全詳細をロギング」および「値を含む全詳細をセッションごとにロギング」ポリシー・ルール・アクションによってのみ作成されます。

属性	記述
値	ログに記録された構造に含まれる 1 つ以上の値。
タイム・スタンプ	「完全な SQL 値」エンティティーが作成された日時。

「IMS アプリケーション・イベント・リンク」エンティティー

属性	記述
IMS アプリケーション・イベント・リンク ID	GDM_APP_EVENT 表のアプリケーション・イベント ID。
IMS 相関 IMSID:相関 ID	S-TAP メッセージの IMS 相関 IMSID フィールド。

「IMS アプリケーション・イベント」エンティティー

属性	記述
アプリケーション・イベント ID	この「アプリケーション・イベント」エンティティーの固有 ID。
イベントの日付	日時値 (GuardAppEvent:Start で設定)。yyyy-mm-dd hh:mm:ss 形式で表示されます。
イベント・リリース日付	日時値 (GuardAppEvent: Released で設定)。yyyy-mm-dd hh:mm:ss 形式で表示されます。
イベント・リリース・タイプ	イベントのタイプ (GuardAppEvent: Released で設定)。
イベント・リリース・ユーザー名	ユーザー名 (GuardAppEvent: Released で設定)。
イベント・リリース値 (数値)	数値 (GuardAppEvent: Released で設定)。
イベント・リリース値 (文字列)	文字列値 (GuardAppEvent: Released で設定)。
イベント・タイプ	イベントのタイプ (GuardAppEvent:Start で設定)。
イベント・ユーザー名	ユーザー名 (GuardAppEvent:Start で設定)。
イベント値 (数値)	数値 (GuardAppEvent:Start で設定)。
イベント値 (文字列)	文字列値 (GuardAppEvent:Start で設定)。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
タイム・スタンプ	イベントがログに記録されるときに 1 回だけ作成されます。この属性と「イベントの日付」属性を混同しないでください。「イベントの日付」は、API 呼び出しを使用するかストアード・プロシージャー・パラメーターから設定できる属性です。 (アプリケーション・イベント API の説明は、API によるユーザーの識別を参照してください。)

「コマンド」エンティティー

各コマンドの親ノードとコマンド構造内でそのコマンドが現れる位置ごとに、エンティティーが作成されます。

属性	記述
コマンド ID	コマンドを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
構造 ID	構造 (例えば select a from b) を一意的に識別します。admin ロールを持つユーザーのみが使用できます。
深さ	SQL 構文解析ツリーにおけるコマンドの深さ。
親	構文解析ツリーにおける親ノードの ID。
SQL 動詞	SQL コマンド内の主動詞 (例えば select、insert、delete など)。

「オブジェクト」エンティティー

このエンティティーのインスタンスは、固有スキーマ内のオブジェクトごとに作成されます。

属性	記述
アプリケーション・オブジェクト・モジュール 1	アプリケーション・オブジェクト・モジュールを一意的に識別します。
構造 ID	オブジェクトが参照される構造を一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
オブジェクト ID	オブジェクトを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
オブジェクト名	オブジェクトの名前。
スキーマ	オブジェクトのデータベース・スキーマ。 注: この属性にはデータが取り込まれることが決してないため、推奨されません。

「フィールド」エンティティー

Guardium は、新規フィールドを検出するたびに、フィールド・エンティティーを作成します。

属性	記述
コマンド ID	参照された構造に含まれるメイン・コマンドを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
構造 ID	参照された構造を一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
フィールド ID	フィールドを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
フィールド名	フィールドの名前。
List 節 Where 節 Order by 節 Having 節 Group By 節 On 節	これらの属性は、複合 SQL 照会を順序付けするのに使用します。 SQL 照会の例: Order by SELECT * FROM dept_costs WHERE dept_total > (SELECT avg FROM avg_cost) ORDER BY department Having SELECT column_name1, SUM(column_name2) FROM table_name GROUP BY column_name1 HAVING (数値関数条件) Group By SELECT column_name1, SUM(column_name2) FROM table_name GROUP BY column_name1 Where SELECT FirstName, LastName, City FROM Users WHERE City = Los Angeles
オブジェクト ID	参照された構造に含まれるオブジェクトを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。

「フィールド SQL 値」エンティティー

これらのエンティティーは、値も一緒にログに記録するポリシー・ルール・アクション (例えば、「値を含む全詳細をロギング」、「値を含む全詳細をセッションごとにロギング」) によってのみ作成されます。ログに記録されたフィールド値は、フィールド名と関連付けられる場合と、そうでない場合があります。 例えば、次のステートメントがログに記録された場合、フィールド名が使用可能です (「フィールド」エンティティー内)。

insert into t1 (foo, bar) (10, 20)

しかし、次のステートメントがログに記録された場合は、使用不可です。

insert into t2 (10, 20)

属性	記述
値	ログに記録された構造に含まれるフィールド値。

親トピック: ドメインのエンティティーおよび属性

「インストール済みポリシー」ドメイン: エンティティーおよび属性

インストール済みポリシーのポリシー・パラメーターとポリシー・ルールの記述。「インストール済みポリシー」ドメインは、複数のポリシーと、ルール 1 つ当たり複数のアクションをサポートします。

使用可能なロール: すべて

「インストール済みポリシー」エンティティー

インストール済みポリシーについて示します。

属性	記述
監査パターン	選択的な監査証跡ポリシーに使用されたテスト・パターン。
ID	ポリシー・インストール・レコードを識別します。
ポリシーの記述	ポリシー定義に含まれる記述。
ルール・セット ID	ルールのセットを識別します。
選択的な監査証跡	これが選択的な監査証跡ポリシーであるかどうかを示します (T/F)。
シーケンス	インストール済みポリシーが複数存在する場合にシーケンスの順序を設定します。
タイム・スタンプ	レコード作成のタイム・スタンプ。

「インストール済みルール」エンティティー

属性	記述
ACCESS_RULE_ID	
アプリケーション・イベント日付値	
アプリケーション・イベントの存在	
アプリケーション・イベント数値	
アプリケーション・イベント・テキスト / 数値 / 日付	
アプリケーション・イベント・テキスト値	
アプリケーション・ユーザー	
アプリケーション・ユーザー / グループ	
アプリケーション・ユーザー・グループ	
カテゴリー / 分類	
カテゴリー名	
分類名	
クライアント IP	
クライアント IP / グループ	
クライアント IP グループ	

属性	記述
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名グループ	
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名/OS ユーザー/データベース名グループ	
クライアント MAC	
クライアント・ネットマスク	
コマンド	
コマンド / グループ	
コマンド・グループ	
次のルール/取り消しに続行	
データベース名	
データベース名 / グループ	
データベース名グループ	
データベース・タイプ	
データベース・ユーザー	
データベース・ユーザー / グループ	
データベース・ユーザー・グループ	
データ・パターン	
エラー・コード	
エラー・コード / グループ	
イベント・タイプ	
イベント・ユーザー名	
例外タイプ	
フィールド	
フィールド・グループ	
フィールド名 / グループ	
GDM_INSTALLED_POLICY_HEADER_ID	
GDM_INSTALLED_POLICY_RULES_ID	
LAST_ACCESSED	
最小数	
ネット・プロトコル	
ネットワーク・プロトコル / グループ	
ネット・プロトコル・グループ	
OS ユーザー	
OS ユーザー / グループ	
OS ユーザー・グループ	
オブジェクト	
オブジェクト・グループ	
オブジェクト名 / グループ	
オブジェクト/コマンド・グループ	
パターン / XML パターン	
期間	
値を記録	

属性	記述
影響を受けるレコードしきい値	
置換文字	
リセット間隔	
戻りデータしきい値	
ルールの記述	
ルール位置	
ルール・タイプ	
SQL パターン	
サーバー・ホスト・グループ	
サーバー・ホスト名	
サーバー IP	
サーバー IP / グループ	
サーバー IP グループ	
サーバー・ネットマスク	
サービス名	
サービス名 / グループ	
サービス名グループ	
重大度	
ソース・プログラム / グループ	
ソース・プログラム・グループ	
ソース・アプリケーション	

「インストール済みルール・アクション」エンティティー

属性	記述
アクセス・ルール ID	
アクション	
シーケンス	
テンプレート名	

「インストール済みアラート通知」エンティティー

属性	記述
ALERT_ID	
ALERT_NOTIFICATION_ID	
ALERT_TYPE	
アラート宛先	
アラート通知タイプ	
アラート・ユーザー	
タイム・スタンプ	

親トピック: ドメインのエンティティーおよび属性

「パーサー・エラー」ドメイン: エンティティーおよび属性

使用可能なロール: admin。

「クライアント/サーバー」エンティティー

このエンティティーは、特定のクライアント/サーバー接続について示します。固有の属性セット (タイム・スタンプを除く) が検出されるたびにインスタンスが作成されます。

注: 「アクセスのトラッキング」の場合のみ、「クライアント/サーバー」エンティティー名は、プルダウン・メニューに 2 つの可能なエンティティー (「クライアント/サーバー」および「セッション別クライアント/サーバー」) として表示されます。

「セッション別クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「セッション」から日付状態を取得します。

「クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「クライアント/サーバー」から日付状態も取得します。

ユーザーが「クライアント/サーバー」を選択すると、照会には ATTRIBUTE_ID = 1 が設定されます。ユーザーが「セッション別クライアント/サーバー」を選択すると、照会には MAIN_ATTRIBUTE_ID = 0 が設定されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
分析されたクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
クライアント・ホスト名	クライアント・ホスト名。
クライアント IP	クライアントの IP アドレス。
クライアント IP/データベース・ユーザー	クライアント IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名	タプル・グループ
クライアント IP/ソース・アプリケーション/ユーザー	タプル・グループ
クライアント MAC	クライアントのハードウェア・アドレス。
クライアント OS	クライアントのオペレーティング・システム。 Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
データベース・プロトコル	データベース・サーバー固有のプロトコル。
データベース・プロトコル・バージョン	データベース・プロトコルのプロトコル・バージョン。
データベース・ユーザー名	データベース・ユーザー名: ローカルまたはリモートのデータベースに接続したユーザー。
最終使用日	
ネットワーク・プロトコル	使用されたネットワーク・プロトコル (例えば TCP、UDP など。Oracle 上の K-TAP については IPC または BEQ として表示されることに注意してください)。
OS ユーザー	相互作用の OS ユーザー・アカウント。
サーバーの記述	サーバーの記述 (ある場合)。
サーバー・ホスト名	サーバー・ホスト名。
サーバー IP	サーバーの IP アドレス。
サーバー IP/データベース・ユーザー	サーバー IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
サーバー IP/サービス名/データベース・ユーザー	タプル・グループ

属性	記述
サーバー OS	サーバーのオペレーティング・システム。 Informix の場合、OS が次のように表示される場合があります。 IEEEM (Unix または JDBC を表します) IEEEI (Windows を表します) DEC (DEC Alpha を表します) Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
サーバー・タイプ	DB2、Oracle、Sybase など。
サービス名	相互作用のサービス名。場合によっては (例えば AIX 共有メモリー接続)、別名がサービス名になり、実際のサービスが接続されるまで別名が使用されます。このような場合、実際のサービスが接続された時点で、新しいセッションが開始されます。したがって、ユーザーには単一セッションと認識されるものが 2 つのセッションとしてログに記録されることになります。 Teradata の場合、サービス名にはセッション論理ホスト ID 値が含まれます。
ソース・プログラム	相互作用のソース・プログラム。
セッション別クライアント/サーバー	セッション別クライアント/サーバーは、メイン・エンティティーでもあります。 この「2 次エンティティー」にアクセスするには、「1 次エンティティー」である「クライアント/サーバー」をクリックします。
タイム・スタンプ	このエンティティーのすべての属性が静的情報を持つので、このタイム・スタンプは、定義済みクライアント/サーバー接続上で Guardium が要求を初めて確認したときに、1 回だけ作成されます。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。

「セッション」エンティティー

このエンティティーは、クライアント/サーバー・データベース・セッションごとに作成されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
クライアント・ポート	クライアント・ポート番号。
データベース名	セッションの対象データベースの名前 (MSSQL または Sybase のみ)。 Oracle の場合、アプリケーション固有の追加情報が「データベース名」に含まれることがあります (V$SESSION ビューの MODULE 列に設定された、セッション用に現在実行中のモジュールなど)。
期間 (秒)	セッション開始からセッション終了までの時間の長さを示します (秒単位)。
グローバル ID	セッション・アクセスを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
これ以降を無視	このセッションを無視し始めたときに作成されたタイム・スタンプ。
非アクティブ・フラグ	0 (デフォルト) - オープン (SQL パッケージによって生成されたセッションの場合)。 1 - クローズ (切断/ログアウト受信)。 2 - おそらくクローズ (長時間にわたってパケットがない状態では非クローズ)。 3 - 非 SQL パケットから生成されたセッションの場合。
古いセッション ID	このセッションが作成されたセッションを示します。 これが接続の最初のセッションである場合は、ゼロです。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
プロセス ID	接続を開始したクライアントのプロセス ID (常に使用可能とは限らない)。
サーバー・ポート	サーバー・ポート番号。

属性	記述
セッション終了	セッションが終了した日時。「セッション終了」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション終了の日付	「セッション終了」の中の日付のみ。
セッション終了の時刻	「セッション終了」の中の時刻のみ。
セッション終了の曜日	「セッション終了」の中の曜日のみ。
セッション終了の年	「セッション終了」の中の年のみ。
セッション ID	セッションを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
セッション無視	セッションの一部が (ある時点から) 無視されたかどうかを示します。
セッション開始	セッションが開始された日時。「セッション開始」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション開始の日付	「セッション開始」の中の日付のみ。
セッション開始の時刻	「セッション開始」の中の時刻のみ。
セッション開始の曜日	「セッション開始」の中の曜日のみ。
セッション開始の年	「セッション開始」の中の年のみ。
端末 ID	セッション情報を解決するために内部で使用された、接続の端末 ID。
タイム・スタンプ	初めは、進行中のアクティブ・セッションのないクライアント/サーバー接続上の最初の要求に対して、タイム・スタンプが作成されます。 その後、セッションが閉じられたとき、または長期間にわたってアクティビティーが監視されないために非アクティブのマークがセッションに付けられたときに、更新されます。 セッション情報をトラッキングする場合は、「タイム・スタンプ」属性よりも「セッション開始」属性と「セッション終了」属性を注目する方が適切な場合があります。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
TTL	admin ロール専用に予約済み。
Uid チェーン	Unix S-TAP (K-Tap モードのみ) によってレポートされたセッションが対象。su ユーザーのユーザー名が異なる場合に、OS ユーザーのチェーンを示します。 ここに表示される値は、OS プラットフォームによって異なります。例えば AIX 環境では、接頭部として IBM IBM IBM という文字列が表示される場合があります。 注: Solaris ゾーンの場合は、ユーザー名の代わりにユーザー ID が「UID チェーン」でレポートされる場合があります。
Uid チェーン圧縮	圧縮された値。「Uid」チェーンを参照してください。

「パーサー・エラー」エンティティー

属性	記述
構造 ID	
カウント	
データベース・プロトコル	
記述	
エラー ID	
エラー・タイプ	
SQL	
セッション ID	
タイム・スタンプ	

親トピック: ドメインのエンティティーおよび属性

「ポリシー違反」ドメイン: エンティティーおよび属性

Guardium 検査エンジンまたは STAP によって検出されたポリシーのすべての違反に関するすべてのポリシー違反データ。

使用可能なロール: すべて

「クライアント/サーバー」エンティティー

このエンティティーは、特定のクライアント/サーバー接続について示します。固有の属性セット (タイム・スタンプを除く) が検出されるたびにインスタンスが作成されます。

注: 「アクセスのトラッキング」の場合のみ、「クライアント/サーバー」エンティティー名は、プルダウン・メニューに 2 つの可能なエンティティー (「クライアント/サーバー」および「セッション別クライアント/サーバー」) として表示されます。

「セッション別クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「セッション」から日付状態を取得します。

「クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「クライアント/サーバー」から日付状態も取得します。

ユーザーが「クライアント/サーバー」を選択すると、照会には ATTRIBUTE_ID = 1 が設定されます。ユーザーが「セッション別クライアント/サーバー」を選択すると、照会には MAIN_ATTRIBUTE_ID = 0 が設定されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
分析されたクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
クライアント・ホスト名	クライアント・ホスト名。
クライアント IP	クライアントの IP アドレス。
クライアント IP/データベース・ユーザー	クライアント IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名	タプル・グループ
クライアント IP/ソース・アプリケーション/ユーザー	タプル・グループ
クライアント MAC	クライアントのハードウェア・アドレス。
クライアント OS	クライアントのオペレーティング・システム。 Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
データベース・プロトコル	データベース・サーバー固有のプロトコル。
データベース・プロトコル・バージョン	データベース・プロトコルのプロトコル・バージョン。
データベース・ユーザー名	データベース・ユーザー名: ローカルまたはリモートのデータベースに接続したユーザー。
最終使用日	
ネットワーク・プロトコル	使用されたネットワーク・プロトコル (例えば TCP、UDP など。Oracle 上の K-TAP については IPC または BEQ として表示されることに注意してください)。
OS ユーザー	相互作用の OS ユーザー・アカウント。
サーバーの記述	サーバーの記述 (ある場合)。
サーバー・ホスト名	サーバー・ホスト名。
サーバー IP	サーバーの IP アドレス。
サーバー IP/データベース・ユーザー	サーバー IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
サーバー IP/サービス名/データベース・ユーザー	タプル・グループ

属性	記述
サーバー OS	サーバーのオペレーティング・システム。 Informix の場合、OS が次のように表示される場合があります。 IEEEM (Unix または JDBC を表します) IEEEI (Windows を表します) DEC (DEC Alpha を表します) Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
サーバー・タイプ	DB2、Oracle、Sybase など。
サービス名	相互作用のサービス名。場合によっては (例えば AIX 共有メモリー接続)、別名がサービス名になり、実際のサービスが接続されるまで別名が使用されます。このような場合、実際のサービスが接続された時点で、新しいセッションが開始されます。したがって、ユーザーには単一セッションと認識されるものが 2 つのセッションとしてログに記録されることになります。 Teradata の場合、サービス名にはセッション論理ホスト ID 値が含まれます。
ソース・プログラム	相互作用のソース・プログラム。
セッション別クライアント/サーバー	セッション別クライアント/サーバーは、メイン・エンティティーでもあります。 この「2 次エンティティー」にアクセスするには、「1 次エンティティー」である「クライアント/サーバー」をクリックします。
タイム・スタンプ	このエンティティーのすべての属性が静的情報を持つので、このタイム・スタンプは、定義済みクライアント/サーバー接続上で Guardium が要求を初めて確認したときに、1 回だけ作成されます。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。

「インシデント状況」エンティティー

「インシデント」エンティティーの状況について示します。

属性	記述
状況の記述	以下のいずれか: オープン - インシデントはまだユーザーに割り当てられていません。 割り当て済み - インシデントは割り当てられています。 クローズ済み - インシデントは閉じられています。

「インシデント重大度」エンティティー

インシデントのインシデント重大度の記述。

属性	記述
インシデント重大度の記述	重大度コード (以下のいずれか): 情報、低、中、高

「ユーザー」エンティティー

監査プロセスの結果の受信者として定義された Guardium ユーザーを識別します。

属性	記述
E メール・アドレス	Guardium ユーザーに定義された E メール・アドレス。
ファーストネーム (名)	Guardium ユーザーのファーストネーム (名)。
最終アクティブ	このユーザーの最終アクティビティーのタイム・スタンプ。
ラストネーム (姓)	Guardium ユーザーのラストネーム (姓)。

属性	記述
ログイン名	Guardium ユーザー名。

「セッション」エンティティー

このエンティティーは、クライアント/サーバー・データベース・セッションごとに作成されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
クライアント・ポート	クライアント・ポート番号。
データベース名	セッションの対象データベースの名前 (MSSQL または Sybase のみ)。 Oracle の場合、アプリケーション固有の追加情報が「データベース名」に含まれることがあります (V$SESSION ビューの MODULE 列に設定された、セッション用に現在実行中のモジュールなど)。
期間 (秒)	セッション開始からセッション終了までの時間の長さを示します (秒単位)。
グローバル ID	セッション・アクセスを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
これ以降を無視	このセッションを無視し始めたときに作成されたタイム・スタンプ。
非アクティブ・フラグ	0 (デフォルト) - オープン (SQL パッケージによって生成されたセッションの場合)。 1 - クローズ (切断/ログアウト受信)。 2 - おそらくクローズ (長時間にわたってパケットがない状態では非クローズ)。 3 - 非 SQL パケットから生成されたセッションの場合。
古いセッション ID	このセッションが作成されたセッションを示します。 これが接続の最初のセッションである場合は、ゼロです。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
プロセス ID	接続を開始したクライアントのプロセス ID (常に使用可能とは限らない)。
サーバー・ポート	サーバー・ポート番号。
セッション終了	セッションが終了した日時。「セッション終了」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション終了の日付	「セッション終了」の中の日付のみ。
セッション終了の時刻	「セッション終了」の中の時刻のみ。
セッション終了の曜日	「セッション終了」の中の曜日のみ。
セッション終了の年	「セッション終了」の中の年のみ。
セッション ID	セッションを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
セッション無視	セッションの一部が (ある時点から) 無視されたかどうかを示します。
セッション開始	セッションが開始された日時。「セッション開始」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション開始の日付	「セッション開始」の中の日付のみ。
セッション開始の時刻	「セッション開始」の中の時刻のみ。
セッション開始の曜日	「セッション開始」の中の曜日のみ。
セッション開始の年	「セッション開始」の中の年のみ。
端末 ID	セッション情報を解決するために内部で使用された、接続の端末 ID。
タイム・スタンプ	初めは、進行中のアクティブ・セッションのないクライアント/サーバー接続上の最初の要求に対して、タイム・スタンプが作成されます。 その後、セッションが閉じられたとき、または長期間にわたってアクティビティーが監視されないために非アクティブのマークがセッションに付けられたときに、更新されます。 セッション情報をトラッキングする場合は、「タイム・スタンプ」属性よりも「セッション開始」属性と「セッション終了」属性を注目する方が適切な場合があります。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
TTL	admin ロール専用に予約済み。

属性	記述
Uid チェーン	Unix S-TAP (K-Tap モードのみ) によってレポートされたセッションが対象。su ユーザーのユーザー名が異なる場合に、OS ユーザーのチェーンを示します。 ここに表示される値は、OS プラットフォームによって異なります。例えば AIX 環境では、接頭部として IBM IBM IBM という文字列が表示される場合があります。 注: Solaris ゾーンの場合は、ユーザー名の代わりにユーザー ID が「UID チェーン」でレポートされる場合があります。
Uid チェーン圧縮	圧縮された値。「Uid」チェーンを参照してください。

「重大度」エンティティー

インシデントまたはポリシー違反のインシデント重大度。

属性	記述
重大度の記述	重大度コードは、以下のいずれかです。 情報、低、中、高

「インシデント」エンティティー

「インシデント」エンティティーは、インシデント生成プロセスによって作成されるか、ポリシー違反をインシデントに割り当てることによって手動で作成されます。

属性	記述
カテゴリー名	インシデントに割り当てられたカテゴリー。
インシデント番号	インシデント番号 (順次割り当て)。
タイム・スタンプ	インシデントが作成された時刻。

「クライアント/サーバー・セッション」エンティティー

属性	記述
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名/OS ユーザー/データベース名	
サーバー IP/サーバー・ポート	

「ポリシー・ルール違反」エンティティー

このエンティティーは、ポリシー・ルール違反がログに記録されるたびに作成されます。 すべてのポリシー・ルール違反がログに記録されるわけではありません。第 11 章『ポリシーのビルド』に記載のルール・アクションの説明を参照してください。 違反を引き起こしているアクセス・ルールは、従属する「アクセス・ルール」エンティティー (前述) で使用可能です。

属性	記述
アプリケーション・イベント ID	アプリケーション・イベント ID (ある場合。これらはアプリケーション・イベント API を使用して設定されます)
アプリケーション・ユーザー名	ポリシー・ルール違反を引き起こしているユーザーの名前。
アクセス・ルールの記述	ルールの定義に含まれるルールの記述。
カテゴリー名	ルールに対して定義されたカテゴリー。
分類名	分類プロセスの名前。
分類プロセス実行 ID	分類プロセスのジョブ実行 ID。
構造 ID	参照された構造を一意的に識別します。
SQL 文字列全体	ポリシー・ルール違反を引き起こしている SQL 文字列。
インシデント番号	インシデントに割り当てられている場合、これがインシデント番号になります。
メッセージの送信	送信されたポリシー・ルール違反メッセージのテキスト。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
重大度	ルールに対して定義された重大度 (これが割り当てられたインシデントの重大度は異なる場合があります)。

属性	記述
タイム・スタンプ	ポリシー・ルール違反がログに記録されるときに作成されます。すべてのポリシー・ルール違反がログに記録されるわけではありません。第 11 章『ポリシーのビルド』に記載のルール・アクションの説明を参照してください。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
オカレンス合計	違反を引き起こしたオカレンス数。
違反ログ ID	違反エンティティーを一意的に識別します。admin ロールを持つユーザーのみが使用できます。

「アプリケーション・イベント」エンティティー

このエンティティーは、アプリケーション・イベント API 呼び出し (これらの属性値を設定する) またはカスタム識別プロシージャーと識別されたストアード・プロシージャー呼び出し (ストアード・プロシージャー・パラメーターをこれらの属性にマップする) をシステムが確認するたびに、作成されます。

属性	記述
アプリケーション・イベント ID	この「アプリケーション・イベント」エンティティーの固有 ID。admin ロールを持つユーザーのみが使用できます。
イベントの日付	日時値 (GuardAppEvent:Start で設定)。yyyy-mm-dd hh:mm:ss 形式で表示されます。 注: yyyy-mm-dd 以外の形式を使用してイベント日付を設定すると、内容はすべてゼロになります。時刻部分 (hh:mm:ss) はオプションであり、省略した場合は 00:00:00 になります。
イベント・リリース日付	日時値 (GuardAppEvent: Released で設定)。yyyy-mm-dd hh:mm:ss 形式で表示されます。
イベント・リリース・タイプ	イベントのタイプ (GuardAppEvent: Released で設定)。
イベント・リリース・ユーザー名	ユーザー名 (GuardAppEvent: Released で設定)。
イベント・リリース値 (数値)	数値 (GuardAppEvent: Released で設定)。
イベント・リリース値 (文字列)	文字列値 (GuardAppEvent: Released で設定)。
イベント・タイプ	イベントのタイプ (GuardAppEvent:Start で設定)。
イベント・ユーザー名	ユーザー名 (GuardAppEvent:Start で設定)。
イベント値 (文字列)	文字列値 (GuardAppEvent:Start で設定)。
イベント値 (数値)	数値 (GuardAppEvent:Start で設定)。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
タイム・スタンプ	イベントがログに記録されるときに 1 回だけ作成されます。この属性と「イベントの日付」属性を混同しないでください。「イベントの日付」は、API 呼び出しを使用するかストアード・プロシージャー・パラメーターから設定できる属性です。 (アプリケーション・イベント API の説明は、API によるユーザーの識別を参照してください。)

「SQL」エンティティー

このエンティティーは、SQL の固有文字列ごとに作成されます。値は疑問符 (?) に置き換えられ、文字列のフォーマットのみが保管されます。

属性	記述
バインド情報	この SQL 文字列のバインド情報。
構造 ID	SQL が出現する構造を一意的に識別します。
SQL	SQL 文字列。
切り捨てられた SQL	SQL が切り捨てられたかどうかを示します。値は次のとおりです。 0 - false/いいえ (切り捨てられていません) 1 - true/はい (切り捨てられました)

「コマンド」エンティティー

各コマンドの親ノードとコマンド構造内でそのコマンドが現れる位置ごとに、エンティティーが作成されます。

属性	記述
コマンド ID	コマンドを一意的に識別します。admin ロールを持つユーザーのみが使用できます。

属性	記述
構造 ID	構造 (例えば select a from b) を一意的に識別します。admin ロールを持つユーザーのみが使用できます。
深さ	SQL 構文解析ツリーにおけるコマンドの深さ。
親	構文解析ツリーにおける親ノードの ID。
SQL 動詞	SQL コマンド内の主動詞 (例えば select、insert、delete など)。

「オブジェクト」エンティティー

このエンティティーのインスタンスは、固有スキーマ内のオブジェクトごとに作成されます。

属性	記述
アプリケーション・オブジェクト・モジュール 1	アプリケーション・オブジェクト・モジュールを一意的に識別します。
構造 ID	オブジェクトが参照される構造を一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
オブジェクト ID	オブジェクトを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
オブジェクト名	オブジェクトの名前。
スキーマ	オブジェクトのデータベース・スキーマ。 注: この属性にはデータが取り込まれることが決してないため、推奨されません。

「フィールド」エンティティー

Guardium は、新規フィールドを検出するたびに、フィールド・エンティティーを作成します。

属性	記述
コマンド ID	参照された構造に含まれるメイン・コマンドを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
構造 ID	参照された構造を一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
フィールド ID	フィールドを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
フィールド名	フィールドの名前。
List 節 Where 節 Order by 節 Having 節 Group By 節 On 節	これらの属性は、複合 SQL 照会を順序付けするのに使用します。 SQL 照会の例: Order by SELECT * FROM dept_costs WHERE dept_total > (SELECT avg FROM avg_cost) ORDER BY department Having SELECT column_name1, SUM(column_name2) FROM table_name GROUP BY column_name1 HAVING (数値関数条件) Group By SELECT column_name1, SUM(column_name2) FROM table_name GROUP BY column_name1 Where SELECT FirstName, LastName, City FROM Users WHERE City = Los Angeles
オブジェクト ID	参照された構造に含まれるオブジェクトを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。

親トピック: ドメインのエンティティーおよび属性

「ポリシー違反サマリー」ドメイン: エンティティーおよび属性

Guardium 検査エンジンまたは STAP によって検出されたポリシーのすべての違反のサマリーに関するすべてのポリシー違反データ。

使用可能なロール: すべて

「クライアント/サーバー」エンティティー

このエンティティーは、特定のクライアント/サーバー接続について示します。固有の属性セット (タイム・スタンプを除く) が検出されるたびにインスタンスが作成されます。

注: 「アクセスのトラッキング」の場合のみ、「クライアント/サーバー」エンティティー名は、プルダウン・メニューに 2 つの可能なエンティティー (「クライアント/サーバー」および「セッション別クライアント/サーバー」) として表示されます。

「セッション別クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「セッション」から日付状態を取得します。

「クライアント/サーバー」は、「クライアント/サーバー」からカウントを取得し、「クライアント/サーバー」から日付状態も取得します。

ユーザーが「クライアント/サーバー」を選択すると、照会には ATTRIBUTE_ID = 1 が設定されます。ユーザーが「セッション別クライアント/サーバー」を選択すると、照会には MAIN_ATTRIBUTE_ID = 0 が設定されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
分析されたクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
クライアント・ホスト名	クライアント・ホスト名。
クライアント IP	クライアントの IP アドレス。
クライアント IP/データベース・ユーザー	クライアント IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名	タプル・グループ
クライアント IP/ソース・アプリケーション/ユーザー	タプル・グループ
クライアント MAC	クライアントのハードウェア・アドレス。
クライアント OS	クライアントのオペレーティング・システム。 Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
データベース・プロトコル	データベース・サーバー固有のプロトコル。
データベース・プロトコル・バージョン	データベース・プロトコルのプロトコル・バージョン。
データベース・ユーザー名	データベース・ユーザー名: ローカルまたはリモートのデータベースに接続したユーザー。
最終使用日	
ネットワーク・プロトコル	使用されたネットワーク・プロトコル (例えば TCP、UDP など。Oracle 上の K-TAP については IPC または BEQ として表示されることに注意してください)。
OS ユーザー	相互作用の OS ユーザー・アカウント。
サーバーの記述	サーバーの記述 (ある場合)。
サーバー・ホスト名	サーバー・ホスト名。
サーバー IP	サーバーの IP アドレス。
サーバー IP/データベース・ユーザー	サーバー IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
サーバー IP/サービス名/データベース・ユーザー	タプル・グループ

属性	記述
サーバー OS	サーバーのオペレーティング・システム。 Informix の場合、OS が次のように表示される場合があります。 IEEEM (Unix または JDBC を表します) IEEEI (Windows を表します) DEC (DEC Alpha を表します) Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
サーバー・タイプ	DB2、Oracle、Sybase など。
サービス名	相互作用のサービス名。場合によっては (例えば AIX 共有メモリー接続)、別名がサービス名になり、実際のサービスが接続されるまで別名が使用されます。このような場合、実際のサービスが接続された時点で、新しいセッションが開始されます。したがって、ユーザーには単一セッションと認識されるものが 2 つのセッションとしてログに記録されることになります。 Teradata の場合、サービス名にはセッション論理ホスト ID 値が含まれます。
ソース・プログラム	相互作用のソース・プログラム。
セッション別クライアント/サーバー	セッション別クライアント/サーバーは、メイン・エンティティーでもあります。 この「2 次エンティティー」にアクセスするには、「1 次エンティティー」である「クライアント/サーバー」をクリックします。
タイム・スタンプ	このエンティティーのすべての属性が静的情報を持つので、このタイム・スタンプは、定義済みクライアント/サーバー接続上で Guardium が要求を初めて確認したときに、1 回だけ作成されます。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。

「クライアント/サーバー・セッション」エンティティー

属性	記述
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名/OS ユーザー/データベース名	
サーバー IP/サーバー・ポート	

「ポリシー・ルール違反サマリー」エンティティー

属性	記述
アクセス・ルールの記述	
クライアント IP	
カウント	
データベース・ユーザー	
ポリシー違反数 ID	
サーバー IP	
サービス名	
セッション ID	
重大度	
ソース・プログラム	
タイム・スタンプ	
判定	
違反日の始まり	

属性	記述
違反日の終わり	

「ポリシー・ルール違反」エンティティー

このエンティティーは、ポリシー・ルール違反がログに記録されるたびに作成されます。 すべてのポリシー・ルール違反がログに記録されるわけではありません。第 11 章『ポリシーのビルド』に記載のルール・アクションの説明を参照してください。 違反を引き起こしているアクセス・ルールは、従属する「アクセス・ルール」エンティティー (前述) で使用可能です。

属性	記述
アプリケーション・イベント ID	アプリケーション・イベント ID (ある場合。これらはアプリケーション・イベント API を使用して設定されます)
アプリケーション・ユーザー名	ポリシー・ルール違反を引き起こしているユーザーの名前。
アクセス・ルールの記述	ルールの定義に含まれるルールの記述。
カテゴリー名	ルールに対して定義されたカテゴリー。
分類名	分類プロセスの名前。
分類プロセス実行 ID	分類プロセスのジョブ実行 ID。
構造 ID	参照された構造を一意的に識別します。
SQL 文字列全体	ポリシー・ルール違反を引き起こしている SQL 文字列。
インシデント番号	インシデントに割り当てられている場合、これがインシデント番号になります。
メッセージの送信	送信されたポリシー・ルール違反メッセージのテキスト。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
重大度	ルールに対して定義された重大度 (これが割り当てられたインシデントの重大度は異なる場合があります)。
タイム・スタンプ	ポリシー・ルール違反がログに記録されるときに作成されます。すべてのポリシー・ルール違反がログに記録されるわけではありません。第 11 章『ポリシーのビルド』に記載のルール・アクションの説明を参照してください。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
オカレンス合計	違反を引き起こしたオカレンス数。
違反ログ ID	違反エンティティーを一意的に識別します。admin ロールを持つユーザーのみが使用できます。

「セッション」エンティティー

このエンティティーは、クライアント/サーバー・データベース・セッションごとに作成されます。

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。 admin ロールを持つユーザーのみが使用できます。
クライアント・ポート	クライアント・ポート番号。
データベース名	セッションの対象データベースの名前 (MSSQL または Sybase のみ)。 Oracle の場合、アプリケーション固有の追加情報が「データベース名」に含まれることがあります (V$SESSION ビューの MODULE 列に設定された、セッション用に現在実行中のモジュールなど)。
期間 (秒)	セッション開始からセッション終了までの時間の長さを示します (秒単位)。
グローバル ID	セッション・アクセスを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
これ以降を無視	このセッションを無視し始めたときに作成されたタイム・スタンプ。
非アクティブ・フラグ	0 (デフォルト) - オープン (SQL パッケージによって生成されたセッションの場合)。 1 - クローズ (切断/ログアウト受信)。 2 - おそらくクローズ (長時間にわたってパケットがない状態では非クローズ)。 3 - 非 SQL パケットから生成されたセッションの場合。
古いセッション ID	このセッションが作成されたセッションを示します。 これが接続の最初のセッションである場合は、ゼロです。

属性	記述
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
プロセス ID	接続を開始したクライアントのプロセス ID (常に使用可能とは限らない)。
サーバー・ポート	サーバー・ポート番号。
セッション終了	セッションが終了した日時。「セッション終了」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション終了の日付	「セッション終了」の中の日付のみ。
セッション終了の時刻	「セッション終了」の中の時刻のみ。
セッション終了の曜日	「セッション終了」の中の曜日のみ。
セッション終了の年	「セッション終了」の中の年のみ。
セッション ID	セッションを一意的に識別します。 admin ロールを持つユーザーのみが使用できます。
セッション無視	セッションの一部が (ある時点から) 無視されたかどうかを示します。
セッション開始	セッションが開始された日時。「セッション開始」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション開始の日付	「セッション開始」の中の日付のみ。
セッション開始の時刻	「セッション開始」の中の時刻のみ。
セッション開始の曜日	「セッション開始」の中の曜日のみ。
セッション開始の年	「セッション開始」の中の年のみ。
端末 ID	セッション情報を解決するために内部で使用された、接続の端末 ID。
タイム・スタンプ	初めは、進行中のアクティブ・セッションのないクライアント/サーバー接続上の最初の要求に対して、タイム・スタンプが作成されます。 その後、セッションが閉じられたとき、または長期間にわたってアクティビティーが監視されないために非アクティブのマークがセッションに付けられたときに、更新されます。 セッション情報をトラッキングする場合は、「タイム・スタンプ」属性よりも「セッション開始」属性と「セッション終了」属性を注目する方が適切な場合があります。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
TTL	admin ロール専用に予約済み。
Uid チェーン	Unix S-TAP (K-Tap モードのみ) によってレポートされたセッションが対象。su ユーザーのユーザー名が異なる場合に、OS ユーザーのチェーンを示します。 ここに表示される値は、OS プラットフォームによって異なります。例えば AIX 環境では、接頭部として IBM IBM IBM という文字列が表示される場合があります。 注: Solaris ゾーンの場合は、ユーザー名の代わりにユーザー ID が「UID チェーン」でレポートされる場合があります。
Uid チェーン圧縮	圧縮された値。「Uid」チェーンを参照してください。

「重大度」エンティティー

インシデントまたはポリシー違反のインシデント重大度。

属性	記述
重大度の記述	重大度コードは、以下のいずれかです。 情報、低、中、高

親トピック: ドメインのエンティティーおよび属性

「照会再書き込み」ドメイン: エンティティーおよび属性

「照会再書き込みログ」エンティティー

属性	記述
適用される QR 定義 ID	適用される照会再書き込み定義 ID
適用される QR 定義名	適用される照会再書き込み定義名。
入力 SQL	入力 SQL
インスタンス ID	インスタンス ID

属性	記述
出力 SQL	照会再書き込み結果 SQL
QR ログの詳細	照会再書き込みの詳細ログ
QR ログ ID	照会再書き込みログ ID
照会再書き込みログ (Query Rewrite Log)	照会再書き込みログ

「クライアント/サーバー」エンティティー

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。
分析されたクライアント IP	暗号化トラフィックにのみ適用されます。設定されると、クライアント IP はゼロに設定されます。 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。
クライアント・ホスト名	クライアント・ホスト名。
クライアント IP	クライアントの IP アドレス
クライアント IP/データベース・ユーザー	クライアント IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名	クライアント IP アドレス/ソース・アプリケーション・プログラム/データベース・ユーザー名/サーバー IP アドレス/サービス名
クライアント IP/ソース・アプリケーション/ユーザー	クライアント IP アドレス/ソース・アプリケーション・プログラム/ユーザー名
クライアント MAC	クライアントのハードウェア・アドレス。
クライアント OS	クライアントのオペレーティング・システム。 Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
データベース・プロトコル	データベース・サーバー固有のプロトコル。
データベース・プロトコル・バージョン	データベース・プロトコルのプロトコル・バージョン。
データベース・ユーザー名	データベース・ユーザー名: ローカルまたはリモートのデータベースに接続したユーザー。
ネットワーク・プロトコル	使用されたネットワーク・プロトコル (例えば TCP、UDP など。Oracle 上の K-TAP については IPC または BEQ として表示されることに注意してください)。
OS ユーザー	相互作用の OS ユーザー・アカウント。
サーバーの記述	サーバーの記述 (ある場合)。
サーバー・ホスト名	サーバー・ホスト名
サーバー IP	サーバーの IP アドレス。
サーバー IP/データベース・ユーザー	サーバー IP アドレスとデータベース・ユーザー名から構成されるペアの属性値。
サーバー IP/サービス名/データベース・ユーザー	サーバー IP アドレス/サービス名/データベース・ユーザー名

属性	記述
サーバー OS	サーバーのオペレーティング・システム。 Informix の場合、OS が次のように表示される場合があります。 IEEEM (Unix または JDBC を表します) IEEEI (Windows を表します) DEC (DEC Alpha を表します) Teradata の場合、クライアント/サーバー OS に関する直接情報がないので、代わりにデータ・フォーマット・タイプが使用されます。これによって、DB セッション中に整数データがどのように保管されるかが示されます。 これは使用するプラットフォームと密接な関連があり、以下のように表示される場合があります。 IBM® MAINFRAME // IBM mainframe data format HONEYWELL MAINFRAME // Honeywell mainframe data format AT&T 3B2 // AT&T 3B2 data format. INTEL 8086 // Intel 8086 data format (IBM PC or compatible) VAX // VAX data format AMDAHL // Amdahl data format
サーバー・タイプ	DB2、Oracle、Sybase など。
サービス名	相互作用のサービス名。場合によっては (例えば AIX® 共有メモリー接続)、別名がサービス名になり、実際のサービスが接続されるまで別名が使用されます。このような場合、実際のサービスが接続された時点で、新しいセッションが開始されます。したがって、ユーザーには単一セッションと認識されるものが 2 つのセッションとしてログに記録されることになります。 Teradata の場合、サービス名にはセッション論理ホスト ID 値が含まれます。
ソース・プログラム	相互作用のソース・プログラム。
タイム・スタンプ	このエンティティーのすべての属性が静的情報を持つので、このタイム・スタンプは、定義済みクライアント/サーバー接続上で Guardium が要求を初めて確認したときに、1 回だけ作成されます。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。

「セッション」エンティティー

属性	記述
アクセス ID	このクライアント/サーバー接続の固有 ID。
クライアント・ポート	クライアント・ポート番号。
コレクター ID	Guardium コレクターの ID。
データベース名	セッションの対象データベースの名前 (MSSQL または Sybase のみ)。 Oracle の場合、アプリケーション固有の追加情報が「データベース名」に含まれることがあります (V$SESSION ビューの MODULE 列に設定された、セッション用に現在実行中のモジュールなど)。
期間 (秒)	セッション開始からセッション終了までの時間の長さ (秒単位)。
暗号化タイプ	ネットワーク・トラフィックの暗号化タイプ。
フェイルオーバー・フラグ	
グローバル ID	セッション・アクセスを一意的に識別します。
これ以降を無視	このセッションを無視し始めたときに作成されたタイム・スタンプ。
非アクティブ・フラグ	0 (デフォルト) - オープン (SQL パッケージによって生成されたセッションの場合)。 1 - クローズ (切断/ログアウト受信)。 2 - おそらくクローズ (長時間にわたってパケットがない状態では非クローズ)。 3 - 非 SQL パケットから生成されたセッションの場合。
検査エンジン ID	検査エンジン ID
検査エンジン名	このデータを報告している検査エンジンの名前
最終使用日	データが最後に使用されたときのタイム・スタンプ。
ログイン成功	ログインが成功したかどうか
MS/ TD SID	Microsoft/Teradata セッション ID。
古いセッション ID	このセッションが作成されたセッションを示します。 これが接続の最初のセッションである場合は、ゼロです。

属性	記述
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。
プロセス ID	接続を開始したクライアントのプロセス ID (常に使用可能とは限らない)。
送信者 IP	送信者の IP アドレス。
サーバー・ポート	サーバー・ポート番号。
セッション終了	セッションが終了した日時。「セッション終了」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション終了の日付	「セッション終了」の中の日付のみ。
セッション終了の時刻	「セッション終了」の中の時刻のみ。
セッション終了の曜日	「セッション終了」の中の曜日のみ。
セッション終了の年	「セッション終了」の中の年のみ。
セッション ID	セッションを一意的に識別します。
セッション無視	セッションの一部が (ある時点から) 無視されたかどうかを示します。
セッション開始	セッションが開始された日時。「セッション開始」は、メイン・エンティティーでもあります。 この 2 次エンティティーにアクセスするには、1 次エンティティーである「セッション」をクリックします。
セッション開始の日付	「セッション開始」の中の日付のみ。
セッション開始の曜日	「セッション開始」の中の曜日のみ。
セッション開始の年	「セッション開始」の中の年のみ。
TTL	admin ロール専用に予約済み。
端末 ID	セッション情報を解決するために内部で使用された、接続の端末 ID。
タイム・スタンプ	初めは、進行中のアクティブ・セッションのないクライアント/サーバー接続上の最初の要求に対して、タイム・スタンプが作成されます。 その後、セッションが閉じられたとき、または長期間にわたってアクティビティーが監視されないために非アクティブのマークがセッションに付けられたときに、更新されます。 セッション情報をトラッキングする場合は、「タイム・スタンプ」属性よりも「セッション開始」属性と「セッション終了」属性を注目する方が適切な場合があります。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
Uid チェーン	Unix S-TAP (K-Tap モードのみ) によってレポートされたセッションが対象。su ユーザーのユーザー名が異なる場合に、OS ユーザーのチェーンを示します。 ここに表示される値は、OS プラットフォームによって異なります。例えば AIX 環境では、接頭部として IBM IBM IBM という文字列が表示される場合があります。 注: Solaris ゾーンの場合は、ユーザー名の代わりにユーザー ID が「Uid チェーン」でレポートされる場合があります。
Uid チェーン圧縮	圧縮された値。「Uid」チェーンを参照してください。

「アクセス期間」エンティティー

属性	記述
アプリケーション・イベント ID	アプリケーション・イベント ID (API から設定された場合)。これらの属性は、照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、これらは使用できません。
アプリケーション・ユーザー	アプリケーション・ユーザー名。
平均実行時間	期間中の平均コマンド実行時間。SQL ステートメントのみが対象です。FTP トラフィックには適用されません。
実行確認応答時間の平均	期間中の平均コマンド実行時間。SQL ステートメントのみが対象です。FTP トラフィックには適用されません。
影響を受けるレコードの平均	影響を受けたレコードの平均数。表の最後に記載されている注を参照してください。 これらの属性は、照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、これらは使用できません。
コレクター ID	Guardium コレクターの ID。
構造 ID	コマンド構造 (例えば select a from b) を一意的に識別します。admin ロールを持つユーザーのみが使用できます。
DB2 i 現行ユーザー	DB2 i 現行ユーザーの名前。
Db2 i/z データベース	DB2 i/z データベースの名前。
Db2 i/z プログラム	DB2 i/z プログラムの名前。

属性	記述
退出 KB カウント	退出データのカウント (キロバイト)。
F5 IP	F5 IP アドレス
F5 ユーザー名	F5 ユーザー名
失敗した SQL	失敗した SQL 要求の数。これらの属性は、照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、これらは使用できません。
IMS PSB 名	IMS システム・ユーティリティー - プログラム仕様ブロック (PSB) 名。
進入 KB カウント	進入データのカウント (キロバイト)。
インスタンス ID	構造のインスタンスを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
オブジェクトおよび動詞	SQL オブジェクトおよびコマンド。
元のタイム・ゾーン	GMT/UTC オフセットとしての Guardium コレクター・マシンの元のタイム・ゾーン。
期間の終了	アクセス期間の終了の日時。
期間の終了日	期間終了属性の中の日付のみ。
期間の終了時刻	期間終了属性の中の時刻のみ。
期間の終了曜日	期間終了属性の中の曜日のみ。
期間の開始日	期間開始属性の中の日付のみ。
期間の開始時刻	期間開始属性の中の時刻のみ。
期間の開始曜日	期間開始属性の中の曜日のみ。
セッション ID	セッションを一意的に識別します。admin ロールを持つユーザーのみが使用できます。
秒を表示	1 秒当たりのアクセス数がトラッキングされている場合、アクセス期間内 (通常は 1 時間) の秒ごとのカウントがここに含まれています。
成功した SQL	成功した SQL 要求の数。これらの属性は、照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、これらは使用できません。
タイム・スタンプ	初めは、アクセス期間中にクライアント/サーバー接続上で要求が初めて確認されたときに、タイム・スタンプ値が設定されます。 デフォルトではアクセス期間の長さは 1 時間ですが、Guardium 管理者は「検査エンジン構成」で変更することができます。その後は、後続の要求ごとに、その期間の平均実行時間とコマンド数が更新されるときに更新されます。
タイム・スタンプの日付	タイム・スタンプの日付。
タイム・スタンプの時刻	タイム・スタンプの時刻。
タイム・スタンプ	タイム・スタンプの曜日。
タイム・スタンプの年	タイム・スタンプの年。
タイム・スタンプ (マイクロ秒)	UNIX エポック時刻 (マイクロ秒)。
影響を受けるレコード合計	影響を受けたレコードの総数。これらの属性は、照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、これらは使用できません。 「影響を受けるレコード合計」属性が数値ではなく文字列の場合、その値はここに表示されます (例えば、「大規模結果セット」や N/A)。 「影響されるレコード」- SQL ステートメントを実行するたびに影響を受けるレコードの数を示す結果セット。 注: 「影響されるレコード」オプションは、スニファーに対して、追加の応答パケットを処理し、影響を受けたデータ (バッファー・サイズを増やし、スニファー全体のパフォーマンスに悪影響を及ぼす可能性があるデータ) のロギングを延期するように要求するスニファー操作です。大きな影響を受けるのは、非常に大きい応答からです。この操作に関連する大量のオーバーヘッドを避けるために、Guardium はデフォルトのしきい値セットを使用して、しきい値を超えたときに、処理操作をスキップすることをスニファーが決定できるようにします。 CLI コマンド store max_results_set_size、store max_result_set_packet_size、および store max_tds_response_packets を使用して、細分度のレベルを設定することができます。 結果セットの値の例は次のとおりです。 • ケース 1、「影響されるレコード」値: 正数 - これは、結果セットの正しいサイズを表します。 • ケース 2、「影響されるレコード」値: -2 - これは、レコード数が構成可能な限度 (CLI インターフェースによって調整可能) を超えたことを示します。 • ケース 3、「影響されるレコード」値: -1 - これは、Guardium によってサポートされないパケット構成のケースを示します。 • ケース 4、「影響されるレコード」値: -2 - 結果セットがストリーム・モードで送信される場合。 • ケース 5、「影響されるレコード」値: -2 - ユーザーを現在の値について更新するためのレコードのカウント中の中間結果。最終的には、レコードの合計を示す正数になります。
影響を受けるレコード合計 (名前)	これらの属性は、照会のメイン・エンティティーでこのレベルの詳細が可能な場合のみ表示されます。「クライアント/サーバー」または「セッション」のどちらかがメイン・エンティティーである場合、これらは使用できません。
アクセス合計	このアクセス期間における構造インスタンスの総数。admin ロールを持つユーザーのみが使用できます。

親トピック: ドメインのエンティティーおよび属性

「S-TAP 状況」ドメイン: エンティティーおよび属性

「STAP プロパティー」エンティティー

属性	記述
# CPU	
S-TAP によって確認済み	
アプリケーション・サーバー・インストール済み	アプリケーション・サーバーがインストール済みであるかどうか。はい/いいえ。
DB2 共有メモリー	DB2 共有メモリー・ドライバーがインストール済みであるかどうか。はい/いいえ。
暗号化	S-TAP からの通信が暗号化されているかどうか。未暗号化/暗号化のタイプ。
ファイアウォール・インストール済み	ファイアウォールが S-TAP にインストールされているかどうか。はい/いいえ。
ハンター DBS	推奨されません
Ktap (インストール済み)	K-TAP が S-TAP にインストールされているかどうか。はい/いいえ。
ローカル TCP	S-TAP が TCP をローカルに実行しているかどうか。はい/いいえ。
MSS 共有メモリー	MSS 共有メモリー・ドライバーがインストール済みであるかどうか。はい/いいえ。
パイプ	名前付きパイプ・ドライバーが S-TAP にインストールされているかどうか。はい/いいえ。
1 次ホスト名	この S-TAP からデータを受信している 1 次 Guardium システムの IP またはホスト名。
送信者 IP	
S-Tap ホスト	この S-TAP をホスティングしているデータベースの IP またはホスト名。
TEE (インストール済み)	推奨されません。
タップ・バージョン	S-TAP ソフトウェアのバージョン。
タイム・スタンプ	S-TAP 状況のタイム・スタンプ
データベース・インストール・ディレクトリー	DB2、Informix、または Oracle の場合: データベース・インストール・ディレクトリーの絶対パス名。例: /home/oracle10。その他すべてのデータベース・タイプの場合: NA。
DB ポート (最大)	データベースの聴取ポート範囲の終了ポート番号。
DB ポート (最小)	データベースに構成されている聴取ポート範囲の開始ポート番号。
データベース・サーバー・タイプ	データベース・サーバーのプロトコル。
検査エンジン名	
インスタンス名	データベース・インスタンスの名前
ポート範囲	
シーケンス	
状況	S-TAP 状況。緑、赤、黄のいずれか。
タップ ID	検査エンジン ID
タイム・スタンプ	レポートのタイム・スタンプ
Unix ドメイン・ソケット・マーカー	Oracle、MySQL、および Postgres の UNIX ドメイン・ソケットのマーカー。

「STAP データベース・サーバー」エンティティー

属性	記述
データベース・インストール・ディレクトリー	DB2、Informix、または Oracle の場合: データベース・インストール・ディレクトリーの絶対パス名。例: /home/oracle10。その他すべてのデータベース・タイプの場合: NA。
DB ポート (最大)	データベースの聴取ポート範囲の終了ポート番号。
DB ポート (最小)	データベースに構成されている聴取ポート範囲の開始ポート番号。
データベース・サーバー・タイプ	データベース・サーバーのプロトコル。
検査エンジン名	検査エンジンの名前。
インスタンス名	データベース・インスタンスの名前
ポート範囲	
シーケンス	
状況	S-TAP 状況。アクティブ、非アクティブ、ピンクのいずれか。

属性	記述
タップ ID	検査エンジン ID
タイム・スタンプ	検査エンジン ID
Unix ドメイン・ソケット・マーカー	

親トピック: ドメインのエンティティーおよび属性

「S-TAP 状況履歴」ドメイン: エンティティーおよび属性

「STAP プロパティー履歴」エンティティー

属性	記述
S-TAP によって確認済み	
アプリケーション・サーバー・インストール済み	アプリケーション・サーバーがインストール済みであるかどうか。はい/いいえ。
変更時刻	
DB2 共有メモリー	DB2 共有メモリー・ドライバーがインストール済みであるかどうか。はい/いいえ。
暗号化	S-TAP からの通信が暗号化されているかどうか。未暗号化/暗号化のタイプ。
ファイアウォール・インストール済み	ファイアウォールが S-TAP にインストールされているかどうか。はい/いいえ。
ハンター DBS	
Ktap (インストール済み)	K-TAP が S-TAP にインストールされているかどうか。はい/いいえ。
ローカル TCP	S-TAP が TCP をローカルに実行しているかどうか。はい/いいえ。
MSS 共有メモリー	MSS 共有メモリー・ドライバーが S-TAP にインストールされているかどうか。はい/いいえ。
パイプ	名前付きパイプ・ドライバーが S-TAP にインストールされているかどうか。はい/いいえ。
1 次ホスト名	この S-TAP からデータを受信している 1 次 Guardium システムの IP またはホスト名。
S-TAP 変更	
S-Tap ホスト	この S-TAP をホスティングしているデータベースの IP またはホスト名。
TEE (インストール済み)	Tee が S-TAP にインストールされているかどうか。はい/いいえ。
タップ・バージョン	S-TAP ソフトウェアのバージョン。
タイム・スタンプ	状況レコードのタイム・スタンプ。
データベース・インストール・ディレクトリー	DB2、Informix、または Oracle の場合: データベース・インストール・ディレクトリーの絶対パス名。例: /home/oracle10。その他すべてのデータベース・タイプの場合: NA。
DB ポート (最大)	データベースの聴取ポート範囲の終了ポート番号。
DB ポート (最小)	データベースに構成されている聴取ポート範囲の開始ポート番号。
データベース・サーバー・タイプ	データベース・サーバーのプロトコル。
インスタンス名	データベース・インスタンスの名前
ポート範囲	
シーケンス	
状況	S-TAP 状況。アクティブ、非アクティブ、ピンクのいずれか。
TAP ID	検査エンジン ID
タイム・スタンプ	レポートのタイム・スタンプ

「STAP データベース・サーバー履歴」エンティティー

属性	記述
データベース・インストール・ディレクトリー	DB2、Informix、または Oracle の場合: データベース・インストール・ディレクトリーの絶対パス名。例: /home/oracle10。その他すべてのデータベース・タイプの場合: NA。
DB ポート (最大)	データベースの聴取ポート範囲の終了ポート番号。
DB ポート (最小)	データベースに構成されている聴取ポート範囲の開始ポート番号。
データベース・サーバー・タイプ	データベース・サーバーのプロトコル。
インスタンス名	データベース・インスタンスの名前
ポート範囲	

属性	記述
シーケンス	
状況	S-TAP 状況
TAP ID	検査エンジン ID
タイム・スタンプ	状況レポートのタイム・スタンプ。

親トピック: ドメインのエンティティーおよび属性

「S-TAP 検査」ドメイン: エンティティーおよび属性

「STAP 検査ヘッダー」エンティティー

属性	記述
データ・ソース ID	MySQL Turbine データベース表からの主キー – データ・ソース
IE データベース・タイプ	検査エンジンのデータベース・タイプ
最終検査時刻	S-TAP 検査が実行された時刻。
次の検査日時	次の S-TAP 検査が実行される時刻。
STAP ホスト	S-TAP がインストールされているサーバーのホスト名。
STAP IP アドレス	S-TAP がインストールされているサーバーの IP アドレス。
STAP インスタンス名	S-TAP インスタンスの名前
STAP ポート	S-TAP がモニターするデータベース・サーバー上のポート。
STAP 検査ヘッダー ID	S-TAP 検査ヘッダー・テーブルの主キー。
STAP 検査結果	検査の結果。
STAP 検査状況	検査の状況。
TAP データベース・サーバー・タイプ	データベース・サーバー・タイプ
タイム・スタンプ	
スケジュールされた検査	検査がスケジュール済みであるかどうか。
検査タイプ	検査のタイプ: 拡張または通常。

「STAP 検査結果」エンティティー

属性	記述
データ・ソースの記述	(データ・ソース表の) データ・ソースの記述。
データ・ソース ID	データ・ソース表の主キー。
データ・ソース・サービス名	データ・ソースで実行されているサービスの名前。
データ・ソース重大度	
データ・ソース・タイプ	検査されている検査エンジンのデータ・ソースのタイプ。
STAP データベース・タイプ	S-TAP がインストールされているデータベースのタイプ。
STAP ホスト	S-TAP がインストールされているサーバーのホスト名。
STAP IP アドレス	S-TAP がインストールされているサーバーの IP アドレス。
STAP インスタンス名	S-TAP インスタンスの名前
STAP ポート	S-TAP がモニターするデータベース・サーバー上のポート。
STAP 検査ヘッダー ID	S-TAP 検査ヘッダー・テーブルの主キー。
STAP 検査結果	検査の結果。
STAP 検査状況	検査の状況。
STAP 検査時刻	検査の時刻。
タイム・スタンプ	
検査タイプ	検査のタイプ: 拡張または通常。

親トピック: ドメインのエンティティーおよび属性

「S-TAP/Z ファイル」ドメイン: エンティティーおよび属性

「STAP/Z 定義のアップロード」エンティティー

属性	記述
アクティブ	
ディレクトリー	
ファイル接尾部	
インターフェース ID	
最終ファイル名	
サーバー IP	
サーバー名	
タイム・スタンプ	
転送方式	
ユーザー	

「STAP/Z ファイル」エンティティー

属性	記述
ファイル状況	
インターフェース ID	
失敗したイベントの数	
タイム・スタンプ	
処理されたイベントの総数	
UA ファイル名	
UH ファイル名	
UT ファイル名	

親トピック: ドメインのエンティティーおよび属性

「セキュリティー・アセスメントの結果」ドメイン: エンティティーおよび属性

脆弱性評価プロセスの結果を記録します。

使用可能なロール: admin

「アセスメント結果ヘッダー」エンティティー

このエンティティーは、アセスメント結果セットのタスクごとに作成されます。

属性	記述
アセスメント結果 ID	アセスメント結果セットを識別します。admin ロールを持つユーザーのみが使用できます。
アセスメント ID	評価を識別します。admin ロールを持つユーザーのみが使用できます。
タスク ID	評価内のタスクを識別します。admin ロールを持つユーザーのみが使用できます。
パラメーター変更フラグ	最後の実行以降にパラメーターが変更されたかどうかを示します。
実行日	評価が実行された日付。
すべてで受信	これらの結果が配布リスト上のすべての受信者で受信されたかどうかを示します。
全体的なスコア	評価の全体スコア。
開始日付	評価の開始日付。
終了日付	評価の終了日付。
評価の記述	定義に含まれる評価名。
クライアント IP フィルター	選択クライアント: 厳密な IP アドレス、ワイルドカード (*) を含んだアドレス、または空 (すべて選択)
サーバー IP フィルター	選択サーバー: 厳密な IP アドレス、ワイルドカード (*) を含んだアドレス、または空 (すべて選択)
推奨	タスクに対して返された推奨。

「テスト結果」エンティティー

このエンティティーは、テスト結果のセットごとに作成されます。

属性	記述
テスト結果 ID	テスト結果を識別します。 admin ロールを持つユーザーのみが使用できます。
アセスメント結果 ID	アセスメント結果セットを識別します。admin ロールを持つユーザーのみが使用できます。

属性	記述
テスト ID1	テストを識別します。
アセスメント・テスト ID (Assessment Test Id)	評価テスト (タスク) を識別します。admin ロールを持つユーザーのみが使用できます。
テスト・スコア	返されたテスト・スコア。
レポート結果 ID	レポート結果を識別します。
パラメーター変更フラグ	最後のテスト以降にパラメーターが変更されたかどうかを示します。
結果テキスト	テストによって返されたテキスト。
テストの記述	テスト定義に含まれる記述。
推奨	テストによって返された推奨。
スコアの記述	スコアの記述。
しきい値文字列	テストのしきい値プロンプト (例えば、1 人のユーザーに許可される異なる IP の最大数)。
重大度	テスト結果に割り当てられた重大度。
カテゴリー	テスト結果のカテゴリー。
アセスメント結果データ・ソース ID1	テスト結果のデータ・ソースを識別します。
結果の詳細	テストの詳細。
例外グループの記述	例外グループの説明。テストが実行される時に、データが設定されます。

「VA サマリー」エンティティー

属性	記述
不合格になってからの累積経過日数	
合格してからの累積経過日数	
現在のスコア	
現在のスコアになった日付	
データ・ソース名	
データベース・ホスト	
データベース・タイプ	
最初の実行日時	
最初の失敗日時	
最初の合格日時	
最後の実行日時	
最後の失敗日時	
最後の合格日時	
ポート	
サービス名	
テストの記述	
テスト ID	
タイム・スタンプ	
VA サマリー ID	

「アセスメント結果 CVSS 情報」エンティティー

属性	記述
CVSS アクセスの複雑性	
CVSS アクセス・ベクトル	
CVSS 認証	
CVSS 可用性への影響	
CVSS 機密性への影響	
CVSS 生成日時	
CVSS 保全性への影響	
CVSS スコア	

属性	記述
CVSS ソース	

「アセスメント結果 CVE 参照」エンティティー

属性	記述
CVE 参照ソース	
参照 HREF	
参照タイプ	

「アセスメント結果データ・ソース」エンティティー

このエンティティーは、評価テストによってアクセスされたデータ・ソースを識別します。

属性	記述
アセスメント結果データ・ソース ID	データ・ソースの結果セットを識別します。
アセスメント結果 ID	結果を識別します。
データベース・タイプ	データベース・タイプ: Oracle、MS-SQL、DB2®、Sybase、Informix® など。
データベース名	データベース名。
バージョン・レベル	データベースのバージョン・レベル。
パッチ・レベル	データベースのパッチ・レベル。
フル・バージョン情報	データ・ソースのフル・バージョン情報。
データ・ソース名	データ・ソースの名前。
記述	データ・ソースの記述。
ホスト	データ・ソースのホスト名。
ポート	ホスト上のポート番号。
サービス名	データ・ソースのサービス名。
ユーザー名	データ・ソース・アクセスに使用されたユーザー名。

「アセスメント結果データ・ソース ID」および「アセスメント結果 ID」は、admin ロールを持つユーザーのみが使用できます。

「重大度」エンティティー

インシデントまたはポリシー違反のインシデント重大度。

属性	記述
重大度の記述	重大度コードは、以下のいずれかです。 情報、低、中、高

「評価ログ」エンティティー

このエンティティーは、評価が実行されるたびに作成されます。

属性	記述
アセスメント・ログ ID (Assessment Log ID)	評価を一意的に識別します。
タイム・スタンプ	評価のタイム・スタンプ。
タイム・スタンプの日付	タイム・スタンプの日付部分。
タイム・スタンプの時刻	タイム・スタンプの時刻部分。
評価ログ・タイプ	定義済みの照会またはカスタム・テスト。
評価ログ重大度	アセスメント・テスト重大度: クリティカル、メジャー、マイナー、注意、情報。これは重大度分類のレベルで順序付けされたリストです。アセスメント・テスト重大度: クリティカル、メジャー、マイナー、注意、情報。最も重大度の高いものは、このリストの最初の分類です。最も重大度の低いものは、このリストの最後の分類です。
アセスメント結果 ID1	アセスメント結果セットを識別します。
メッセージ	評価によって返されたメッセージ。
詳細	この評価の詳細。

「アセスメント・ログ ID (Assessment Log ID)」は、admin ロールを持つユーザーのみが使用できます。

「スニファーのバッファー使用のモニター」ドメイン: エンティティーおよび属性

検査エンジン統計。

使用可能なロール: なし

「スニファーのバッファー使用」エンティティー

store system netfilter-buffer-size CLI コマンドで設定された間隔で (デフォルトでは 60 秒ごとに)、このエンティティーが作成されます。

属性	記述
タイム・スタンプ	レコードが作成された時刻。
スニファーの CPU 使用時間 %	スニファーによって使用された CPU のパーセンテージ。
スニファーによるメモリー使用 %	スニファーによって使用されたメモリーのパーセンテージ。
Mysql による CPU 使用時間 %	MySQL によって使用された CPU のパーセンテージ。
Mysql によるメモリー使用 %	MySQL によって使用されたメモリーのパーセンテージ。
スニファー・プロセス ID	スニファー・プロセス ID。
メモリー・スニファー	スニファーによって使用されたメモリー量。
時間スニファー	スニファーによって使用された経過時間。
空きバッファー・スペース	空きバッファー・スペース量。
アナライザー・レート	メッセージが分析される速度。
ロガー・レート	メッセージがログに記録される速度。
アナライザー・キューの長さ	分析キューのサイズ。
アナライザー総計	分析されたメッセージの総数。
ロガー・キューの長さ	ロガー・キューのサイズ。
ロガー総計	ログに記録されたメッセージの総数。
セッション・キューの長さ	セッション・キューのサイズ。
セッション総計	セッションの総数。
ハンドラー・データ	内部スニッフィング・エンジン・データ。
追加の情報	内部スニッフィング・エンジン・データ。
アナライザー逸失パケット	アナライザーによって失われたパケット。
Eth0 受信	ETH 0 で受信されたメッセージ。
Eth0 送信	ETH 0 で送信されたメッセージ。
モニターされるロガー・データベース	現在モニターされているデータベース・タイプのリスト。
ルールにより無視されたロガー・パケット	ポリシー・ルール・アクションにより無視されたパケット。
ロガー・セッション・カウント	ログに記録されたセッションの数。
Mysql ディスク使用状況	MySQL ディスク使用状況。
Mysql 起動済み	内部データベース再始動を表すブール値インディケーター (1 = 再始動済み、0 = 未再始動)。
プロミスキャス受信	スニッフィング・ネットワーク・カード (非インターフェース・ポート) を介した受信パケットの率。
終了したスニファー接続	検査エンジンの再始動以降、モニターされて終了した接続の総数。
使用されたスニファー接続	検査エンジンの再始動以降、現在モニターされている接続の総数。
ドロップされたスニファー・パケット	スニファーによってドロップされたパケット。
無視されたスニファー・パケット	スニファーによって無視されたパケット。
スロットルされたスニファー・パケット	検査エンジンの再始動以降、スロットルのために無視された接続の総数。
システム CPU 負荷	システム CPU 使用状況。
システム・メモリー使用状況	システム・メモリー使用状況。
システム・ルート・ディスク使用状況	システム・ルート・ディスク使用状況。

属性	記述
システム・アップタイム	最後の始動からの時間。
/var ディスク使用状況	/var ディスク使用状況。
通常のセッション	通常のセッションの数。
オープンされていないセッション	スニファーによって開かれなかったセッションの数。
セッション・タイムアウト	タイムアウトになったセッションの数。
セッション無視	スニファーによって無視されたセッションの数。
セッション直接クローズ	直接閉じられたセッションの数。
セッション推測	推測されたセッション数。
オープン FD	オープン・ファイル記述子。
データベース・オープン FD	データベース・オープン・ファイル記述子。
Di レート	FAM クローラーのトラフィックまたは Di 表に記録するトラフィックに関連します。
Di キュー長	FAM クローラーのトラフィックまたは Di 表に記録するトラフィックに関連します。
Di 合計	FAM クローラーのトラフィックまたは Di 表に記録するトラフィックに関連します。
Di 逸失パケット	FAM クローラーのトラフィックまたは Di 表に記録するトラフィックに関連します。
未解析ログ要求	未解析ログ要求。

親トピック: ドメインのエンティティーおよび属性

「S-TAP の統計」ドメイン: エンティティーおよび属性

使用可能なロール: すべて

「STAP 統計」エンティティー

属性	記述
アクティブ化された ATAP 数	アクティブ化された A-TAP のリスト
リサイクルされたバッファー	S-TAP バッファーがオーバーフローした回数
誤った ATAP 数	正しくないと思われる A-TAP: アクティブ化されているが非アクティブになっているように見えるか、非アクティブ化されているがアクティブ化されているように見える場合
IOCTL 要求数	K-TAP に送信された ioctl の数
非アクティブ化された ATAP 数	アクティブ化されていない A-TAP の数
S-Tap ホスト	データベース・サーバーの IP またはホスト名
Stap CPU (%)	ps から取得された STAP による CPU 使用率
STAP 統計 ID	
システム CPU アイドル率 (%)	ps から取得された CPU アイドル率
システム CPU (%)	ps から取得されたシステムの CPU 使用率
タイム・スタンプ	統計のタイム・スタンプ
合計バッファー初期化数	K-TAP の部分でバッファーが初期化された回数
ドロップされた合計バイト数 (現時点まで)	K-TAP の部分でドロップされたバイト数
無視された合計バイト数	K-TAP の部分で無視されたバイト数
合計バイト数 (現時点まで)	K-TAP の部分でキャプチャーされたバイト数
無視された合計応答バイト数	K-TAP の部分で無視された S2C バイト数

親トピック: ドメインのエンティティーおよび属性

「ユニット使用状況レベル」ドメイン

使用可能なロール:

「ユニット使用状況レベル」エンティティー

「管理」 > 「レポート」 > 「ユニット使用状況」には、デフォルトで以下を含むユニット使用状況レポートがいくつか提供されています。

- ユニット使用状況: 特定の時間フレームにおける各ユニットのユニット使用状況の最大レベルが表示されます。レポートの時間フレーム内のすべての期間についてのユニットの詳細を表示するドリルダウンがあります。
- ユニット使用状況の分布: このレポートは、ユニットごとに、レポートの時間フレーム内の期間のパーセントを使用状況レベルの低、中、高で示します。
- 使用状況のしきい値: この事前定義レポートは、すべてのユニット使用状況パラメーターの下限しきい値と上限しきい値をすべて表示します。

- ユニット使用状況の日次サマリー - ユニット使用状況データの日次サマリーが表示されます。

さらに、「ユニット使用状況レベル」トラッキングを使用すると、ユーザーはカスタムの照会やレポートを作成できます。
ヒント: ユニット使用状況データを使用するすべてのカスタム・レポートおよび事前定義レポートに対して別名を有効にして、ユニット使用状況レベルが数字ではなく、意味のあるストリングとして表示されるようにします。例えば、1、2、3 ではなく「低」、「中」、「高」などです。
属性のリストには、以下が含まれます。

- ホスト名
- 期間の開始
- 再始動の数
- 再始動レベルの数
- スニファー・メモリー
- スニファー・メモリー・レベル
- MySQL メモリーの比率
- MySQL メモリーの比率のレベル
- 空きバッファー・スペース
- 空きバッファー・スペース・レベル
- アナライザー・キュー
- アナライザー・キュー・レベル
- ロガー・キュー
- ロガー・キュー・レベル
- MySQL ディスク使用状況
- MySQL ディスク使用レベル
- システム CPU 負荷
- システム CPU 負荷レベル
- システム変数ディスク使用状況
- システム変数ディスク使用状況レベル
- 全体のユニット使用状況レベル
- 要求の数
- 要求数のレベル
- 完全 SQL の数
- 完全 SQL 数のレベル
- 例外の数
- 例外数のレベル
- ポリシー違反の数
- ポリシー違反数のレベル
- 未解析ログ要求の数
- 未解析ログ要求数のレベル

注: 各パラメーターには、値と、その値およびしきい値に基づいて計算されたレベルが提供されます。
親トピック: ドメインのエンティティーおよび属性

「ユーザー/ロール/アプリケーション」ドメイン: エンティティーおよび属性

Guardium ユーザー、ロール、およびアプリケーションを関連付けます (それにより、誰がどの Guardium アプリケーションへのアクセス権限を持っているかをレポートします)。

使用可能なロール: admin

「Guardium ロール」エンティティー

このエンティティー (「ユーザー」エンティティーの下) は、Guardium ロールを識別します。

属性	記述
ロール ID	識別されたロールの ID。
ロール	Guardium ロールがリストされます。

「Guardium アプリケーション」エンティティー

このエンティティー (「ユーザー」エンティティーの下) は、Guardium アプリケーションを識別します。

属性	記述
アプリケーション ID	識別されたアプリケーションの ID。
アプリケーション	Guardium アプリケーションがリストされます (例えば、照会 - レポート・ビルダー、ポリシー・ビルダーなど)。

「ユーザー」エンティティー

監査プロセスの結果の受信者として定義された Guardium ユーザーを識別します。

属性	記述
E メール・アドレス	Guardium ユーザーに定義された E メール・アドレス。

属性	記述
ファーストネーム (名)	Guardium ユーザーのファーストネーム (名)。
最終アクティブ	このユーザーの最終アクティビティーのタイム・スタンプ。
ラストネーム (姓)	Guardium ユーザーのラストネーム (姓)。
ログイン名	Guardium ユーザー名。

親トピック: ドメインのエンティティーおよび属性

「VA サマリー」ドメイン: エンティティーおよび属性

使用可能なロール: ユーザー

「VA サマリー」エンティティー

属性	記述
不合格になってからの累積経過日数	初回実行後に不合格状況になっている日数
合格してからの累積経過日数	初回実行後に合格状況になっている日数
現在のスコア	最後の実行のスコア
現在のスコアになった日付	現在のスコアが有効になった日付
データ・ソース名	データ・ソースの名前
データベース・ホスト	データベース・ホスト
データベース・タイプ	データベース・タイプ
最初の実行日時	テストが最初に実行された日付と時刻
最初の失敗日時	テストが初めて失敗した日付と時刻
最初の合格日時	テストが初めて合格した日付と時刻
最後の実行日時	テストが最後に実行された日付と時刻
最後の失敗日時	テストが最後に失敗した日付と時刻
最後の合格日時	テストが最後に合格した日付と時刻
ポート	データベース・ポート
サービス名	データベース・サービス名
テストの記述	テストの記述
テスト ID	テストの ID
タイム・スタンプ	この特定のサマリー・レコードが更新された日時
VA サマリー ID	サマリー・レコードの ID

親トピック: ドメインのエンティティーおよび属性

「脆弱性評価テスト」ドメイン: エンティティーおよび属性

セキュリティー・アセスメントに使用可能なテストについてレポートします。

使用可能なロール: admin

「評価テスト」エンティティー

このエンティティーには、使用可能なテストの項目が含まれます。

属性	記述
テストの記述	テストのテキストの記述。
テスト・タイプ	評価テストのタイプ (監視、事前定義、カスタム、照会ベース、CVE)。
データ・ソース・タイプ	データ・ソースのタイプ (DB2®、Informix®、MYSQL、ORACLE、SYBASE など)。
しきい値	ユーザー定義のしきい値。テストの作成時に定義された値をオーバーライドします。
しきい値のデフォルト値	合格/不合格の基準を定義したデフォルトのしきい値。
重大度	評価の重大度 (クリティカル、メジャー、マイナー、注意、情報)。
カテゴリー	評価のカテゴリー (特権、認証、構成、バージョン、その他)。
タイム・スタンプ	テストが作成されたときのタイム・スタンプ。

SQL ベース評価定義

このエンティティーは、SQL ベースの評価定義について示します。

属性	記述
バインド出力変数	オプション。SQL ステートメントに入力されたテキストが、「比較」値との比較で使用される内部 Guardium® 変数にバインドされる値を返すプロシージャー型コード・ブロックであるかどうかを判別します。
比較値	比較演算子を使用して SQL ステートメントからの戻り値に対する比較に使用される比較値。
外部参照	Center for Internet Security (CIS) または Common Vulnerabilities and Exposures (CVE) への参照。
演算子	条件に使用される演算子。
推奨テキスト (不合格)	テストが不合格だったときに表示される不合格用の推奨テキスト。
推奨テキスト (合格)	テストが合格だったときに表示される合格用の推奨テキスト。
結果テキスト (不合格)	テストが不合格だったときに表示される不合格用の結果テキスト。
結果テキスト (合格)	テストが合格だったときに表示される合格用の結果テキスト。
戻りの型	SQL ステートメントから返される戻りの型。
簡略説明	評価テストの簡略説明。
詳細の SQL	詳細な SQL ステートメント。文字列のリストを取得して、Detail 接頭部と文字列リストの詳細文字列を生成する SQL ステートメントです。
SQL	テストで実行される SQL ステートメント。

親トピック: ドメインのエンティティーおよび属性

「値の変更」ドメイン: エンティティーおよび属性

トリガー・ベースの値変更アプリケーションによってトラッキングされたすべての変更。

使用可能なロール: admin

「値のモニター」エンティティー

「値のモニター」エンティティーは、記録された挿入、更新、または削除ごとに作成され、変更の詳細 (表名、アクション、SQL テキストなど) を含んでいます。 .

属性	記述
タイム・スタンプ	Guardium® アプライアンスで変更が記録された日時。 このタイム・スタンプは、データ・アップロード操作時に作成されます。 監査データベースに変更が記録された時刻ではありません。 その時刻を取得するには、「監査タイム・スタンプ」エンティティーを使用します。
タイム・スタンプの日付	そのタイム・スタンプの日付のみ。
タイム・スタンプの時刻	そのタイム・スタンプの時刻のみ。
タイム・スタンプの年	そのタイム・スタンプの年のみ。
タイム・スタンプの曜日	そのタイム・スタンプの曜日のみ。
サーバー IP	データベース・サーバーの IP アドレス。
データベース・タイプ	データベース・タイプ。
サービス名	Oracle のみ。データベース・サービス名。
データベース名	DB2®、Informix®、Sybase、MS SQL Server のみ。データベース名。
監査 PK	Sybase および MS SQL Server のみ。新旧の値 (これらのデータベース・タイプの場合は別々にログに記録する必要がある) を関連付けるのに使用される主キー。
監査ログイン名	データ・ソースで定義されたデータベース・ユーザー名。
監査表の名前	変更された表の名前。
監査の所有者	変更された表の所有者。
監査アクション	挿入、更新、または削除。
古い値の監査	古い値のコンマ区切りリスト。形式は column-name=column_value です。
新しい値の監査	新しい値のコンマ区切りリスト。形式は column-name=column_value です。
SQL テキスト	Oracle 9 の場合のみ使用可能。値を変更する SQL ステートメント全体。
トリガーされる ID	変更に対して生成された固有 ID (この監査データベース上の固有 ID)。
監査タイム・スタンプ	トリガーが実行された日時。
監査タイム・スタンプの日付	監査タイム・スタンプの日付部分。
監査タイム・スタンプの時刻	監査タイム・スタンプの時刻部分。
監査タイム・スタンプの曜日	監査タイム・スタンプの曜日部分。

属性	記述
監査タイム・スタンプの年	監査タイム・スタンプの年部分。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。

「変更された列」エンティティー

このエンティティーは、変更された列について示します。

属性	記述
変更された列名	データベース上の変更された列の名前。
古い値	変更前の値。
新しい値	変更後の値。
元のタイム・ゾーン	UTC オフセット。これは、特に異なるタイム・ゾーンにコレクターを持つアグリゲーターで実行されます。これにより、時間差で発生したアクティビティーが、アグリゲーターにインポートされるときに同じ時刻に発生したように表示されないようにします。 例えば、異なるタイム・ゾーンのデータを統合するアグリゲーター上で、2 つのレコード・セッションが発生したとします。1 つは元のタイム・ゾーン UTC-02:00 に対して 21:00 に開始し、もう 1 つは元のタイム・ゾーン UTC-05:00 に対して 21:00 に開始したとします。これは、これらのセッションは 3 時間違いで発生しましたが、それぞれの現地時間では同じ時刻 (21:00) に発生したことを意味します。

親トピック: ドメインのエンティティーおよび属性

データベース・ライセンス・レポート

データベース・ライセンス・レポートは、ユーザーが該当するデータのみに対するアクセス権限を持っていることを確認するために使用できます。Guardium システムには、いくつかのデータベース・タイプ用の事前定義のデータベース・ライセンス・レポートが用意されています。

注: DB ライセンス・レポートは、プロダクト・キーにより有効になるオプションのコンポーネントです。これらのコンポーネントが有効になっていない場合は、カスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーの選択に、以下に示す選択項目が表示されません。

事前定義ライセンス・レポートを以下にリストします。これらは、カスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーの選択でドメイン名として表示されます。

- Oracle DB ライセンス・ドメイン
- MYSQL DB ライセンス・ドメイン
- DB2® DB ライセンス・ドメイン
- Db2 for i 6.1 および 7.1 DB ライセンス・ドメイン
- SYBASE DB ライセンス・ドメイン
- Informix® DB ライセンス・ドメイン
- Microsoft SQL Server ライセンス・ドメイン
- Netezza® DB ライセンス・ドメイン
- Teradata DB ライセンス・ドメイン
- PostgreSQL DB ライセンス・ドメイン

資格最適化 も参照してください。

Oracle DB ライセンス

以下のドメインは、Oracle DB ライセンスに関するアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

Oracle

- 「ORA ALTER SYSTEM のアカウント」 - ALTER SYSTEM 特権および ALTER SESSION 特権を持つアカウント
- 「ORA BECOME USER 特権を持つアカウント」 - BECOME USER 特権を持つアカウント
- 「ORA 全システム特権および ADMIN オプション」 - ユーザーおよびロールに対するすべてのシステム特権および管理者オプションを示すレポート
- 「ORA オブジェクトおよび列特権」 - 付与されているオブジェクト特権および列特権 (GRANT オプション付きまたはなし)
- 「ORA PUBLIC によるオブジェクト・アクセス」 - PUBLIC によるオブジェクト・アクセス
- 「ORA オブジェクト特権」 - SYS 内になく、DBA ロールではないデータベース・アカウントによるオブジェクト特権
- 「ORA SYS プロシージャーに対する PUBLIC 実行特権」 - PUBLIC に割り当てられている SYS PL/SQL プロシージャーに対する実行特権
- 「ORA 権限付与されたロール」 - ユーザーおよびロールに権限付与されたロール
- 「ORA 権限付与されたシステム特権」 - 再帰的定義 (特権がロールに割り当てられ、そのロールがユーザーに割り当てられた状態) を含む、ユーザーに付与されたシステム特権を示す階層レポート
- 「ORA SYSDBA および SYSOPER アカウント」 - SYSDBA 特権および SYSOPER 特権を持つアカウント

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

/* 以下の表およびビューに対する選択特権は必須です */

grant select on sys.dba_tab_privs to sqlguard;

grant select on sys.dba_roles to sqlguard;

grant select on sys.dba_users to sqlguard;

grant select on sys.dba_role_privs to sqlguard;

grant select on sys.dba_sys_privs to sqlguard;

grant select on sys.obj$ to sqlguard;

grant select on sys.user$ to sqlguard;

grant select on sys.objauth$ to sqlguard;

grant select on sys.table_privilege_map to sqlguard;

grant select on sys.dba_objects to sqlguard;

grant select on sys.v_$pwfile_users to sqlguard;

grant select on sys.dba_col_privs to sqlguard;

MYSQL DB ライセンス

以下のドメインは、MYSQL DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

MYSQL: 末尾が「_40」である照会では、最も基本的なバージョンの mysql スキーマ (MySQL 4.0 以降用) を使用します。information_schema は MySQL 5.0 で導入されてから変更されていないため、末尾が _50 の照会はありますが、末尾が _51 の照会はありません。 末尾が _50 の照会は、MySQL 5.0 および 5.1 で動作します。 また、information_schema は 6.0 でも変更される予定がないため、6.0 がリリースされた際には 6.0 でも動作します。末尾が「_502」の照会 (MYSQL502) では、新しい information_schema を使用します。これにはより多くの情報が含まれ、実際のデータ・ディクショナリーにより一層類似しています。

- MYSQL データベース特権 40
- MYSQL ユーザー特権 40
- MYSQL ホスト特権 40
- MYSQL 表特権 40
- MYSQL データベース特権 500
- MYSQL ユーザー特権 500
- MYSQL ホスト特権 500
- MYSQL 表特権 500
- MYSQL データベース特権 502
- MYSQL ユーザー特権 502
- MYSQL ホスト特権 502
- MYSQL 表特権 502

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリストで、データベース表内 (またはデータベース表のビュー内) でライセンスが機能するために最小限必要な特権について説明します。

注: データをアップロードするには、必要な特権に加え、ユーザーが MYSQL データベースに接続することが必要です。

MySQL の全バージョンについて、MySQL 5.0.1 を使用したライセンス照会では、表集合 mysql.db mysql.host mysql.tables_priv mysql.user を使用します。

MySQL 5.0.2 以降の全バージョンについて、ライセンス照会では表集合 information_schema.SCHEMA_PRIVILEGES mysql.host information_schema.TABLE_PRIVILEGES information_schema.USER_PRIVILEGES を使用します。

データ・ソースに MYSQL データベース・タイプがあるものの、データベース名がない場合 (「データ・ソース定義」を確認し、「ロケーション」の下にあるデータベース名が空である場合) は、データのアップロードが、ユーザーがアクセス権を持つすべての MYSQL データベースでループします。

Db2 DB ライセンス

以下のドメインは、Db2 DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

- Db2 列レベルの特権 (SELECT、UPDATE など)
- Db2 データベース・レベルの特権 (CONNECT、CREATE など)

- Db2 索引レベルの特権 (CONTROL)
- Db2 パッケージ・レベルの特権 (コード・パッケージ対象の BIND、EXECUTE など)
- Db2 表レベルの特権 (SELECT、UPDATE など)Db2 特権サマリー

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

/* 以下の表およびビューに対する選択特権は必須です */

GRANT SELECT ON SYSCAT.COLAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.DBAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.INDEXAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.PACKAGEAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.DBAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.TABAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.SCHEMAAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.PASSTHRUAUTH TO SQLGUARD;

Db2 z/OS ライセンス
: 以下のドメインは、Db2 for z/OS の DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。

Db2 zOS PUBLIC に付与された実行可能オブジェクト特権

DB2 zOS PUBLIC に付与されたオブジェクト特権

Db2 zOS GRANTEE に付与されたシステム特権 - V8

Db2 zOS GRANTEE に付与されたシステム特権 - V9

Db2 zOS GRANTEE に付与されたシステム特権 - V10 以降

DB2 zOS GRANTEE に付与されたデータベース特権

Db2 zOS GRANTEE に付与されたスキーマ特権 - V9 以降

Db2 zOS GRANTEE に付与されたスキーマ特権 - V8 のみ

Db2 zOS GRANTEE に付与されたデータベース・リソース

DB2 zOS GRANTEE に付与されたオブジェクト特権

Db2 zOS GRANT 付きで付与されたシステム特権 - V8

Db2 zOS GRANT 付きで付与されたシステム特権 - V9

Db2 zOS GRANT 付きで付与されたシステム特権 - V10 以降

DB2 zOS PUBLIC に付与されたデータベース・リソース

DB2 zOS PUBLIC に付与されたスキーマ特権

DB2 zOS PUBLIC に付与されたデータベース特権

Db2 zOS PUBLIC に付与されたシステム特権 - V10 以降

Db2 zOS PUBLIC に付与されたシステム特権 - V9

Db2 zOS PUBLIC に付与されたシステム特権 - V8

DB2 zOS GRANT 付きで付与されたオブジェクト特権

DB2 zOS GRANT 付きで付与されたデータベース・リソース

Db2 zOS GRANT 付きで付与されたスキーマ特権 - V8 のみ

Db2 zOS GRANT 付きで付与されたスキーマ特権 - V9 以降

Db2 zOS GRANT 付きで付与されたデータベース特権

Db2 for i 6.1 および 7.1 DB ライセンス

以下のドメインは、Db2 for i DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

スクリプト gdmmonitor-db2-IBMi.sql を使用して、データベース表内 (またはデータベース表のビュー内) でライセンスが機能するために必要な最小限の特権を詳述します。

GRANTEE に付与されたオブジェクト特権 (オブジェクト・タイプ: スキーマ、表、ビュー、パッケージ、ルーチン、シーケンス、列、グローバル変数、および XML スキーマ)

PUBLIC に付与されたオブジェクト特権 (オブジェクト・タイプ: スキーマ、表、ビュー、パッケージ、ルーチン、シーケンス、列、グローバル変数、および XML スキーマ)

PUBLIC に付与された実行可能オブジェクト特権 (オブジェクト・タイプ: パッケージおよびルーチン)

GRANT オプション付きで GRANTEE に付与されたオブジェクト特権 (オブジェクト・タイプ: スキーマ、表、ビュー、パッケージ、ルーチン、シーケンス、列、グローバル変数、および XML スキーマ)

すべてのオブジェクト特権は、事前定義された Guardium グループ「Db2 for i 除外システム・スキーマ - 資格レポート」からデフォルト・システム・スキーマを除外します。除外するスキーマはこのグループに追加してください。

SYBASE DB ライセンス

以下のドメインは、SYBASE DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

- GRANT オプションを含む、SYBASE ユーザーに付与されたシステム特権とロール
- GRANT オプションを含む、SYBASE ユーザーに権限付与されたロールおよびユーザーとロールに付与されたシステム特権
- SYBASE PUBLIC によるオブジェクト・アクセス
- PUBLIC に割り当てられた、プロシージャーおよび関数に対する SYBASE 実行特権
- システムまたはセキュリティー admin ロールを持つ SYBASE アカウント
- GRANT オプション付きで付与された SYBASE オブジェクト特権および列特権
- SYBASE ユーザーに権限付与されたロール

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

/* 以下の表およびビューに対する選択特権は必須です */

/* 以下は MASTER データベースでは必須です */

grant select on master.dbo.sysloginroles to sqlguard

grant select on master.dbo.syslogins to sqlguard

grant select on master.dbo.syssrvroles to sqlguard

/*以下は MASTER を含むすべてのデータベースで必須です */

grant select on sysprotects to sqlguard

grant select on sysusers to sqlguard

grant select on sysobjects to sqlguard

grant select on sysroles to sqlguard

データ・ソースに SYBASE データベース・タイプがあるものの、データベース名がない場合 (「データ・ソース定義」を確認し、「ロケーション」の下にあるデータベース名が空である場合) は、データのアップロードが、ユーザーがアクセス権を持つすべての SYBASE データベースでループします。

SYBASE IQ ライセンス

サポートされるバージョンは sybase IQ 15 以上です。

データをアップロードするために以下のカスタム表定義が作成されます (ID は無視できます)。

- 139 | Sybase IQ オブジェクト特権 (DB ユーザー別)
- 140 | Sybase IQ オブジェクト特権 (グループ別)
- 141 | Sybase IQ ユーザーに付与されたシステム権限とグループ
- 142 | Sybase IQ ユーザーとグループに付与されたシステム権限とグループ
- 143 | Sybase IQ PUBLIC によるオブジェクト・アクセス
- 144 | Sybase IQ PUBLIC に付与されたプロシージャーと関数の実行特権
- 145 | Sybase IQ データベース管理者/アクセス権管理者などの権限を持つユーザー・グループ
- 146 | Sybase IQ GRANT 付きで付与された表、ビューの特権
- 147 | Sybase IQ ユーザーとグループに付与されたグループ
- 148 | Sybase IQ ログインが指定されたユーザー・グループのログイン・ポリシー

対応する照会およびレポートは以下のとおりです (ID は無視できます)。

- 597 | Sybase IQ オブジェクト特権 (DB ユーザー別)
- 598 | Sybase IQ オブジェクト特権 (グループ別)
- 599 | Sybase IQ ユーザーに付与されたシステム権限とグループ
- 600 | Sybase IQ ユーザーとグループの被付与者に付与されたシステム権限とグループ
- 601 | Sybase IQ PUBLIC によるオブジェクト・アクセス
- 602 | Sybase IQ PUBLIC に付与されたプロシージャーと関数に対する実行特権
- 603 | Sybase IQ データベース管理者/アクセス権管理者/ユーザー管理者/リモート・データベース管理者のデータベース権限を持つユーザー・グループ
- 604 | Sybase IQ GRANT 付きで付与された表、ビューの特権
- 605 | Sybase IQ ユーザーとグループに付与されたグループ
- 606 | Sybase IQ ログイン・オプション設定が指定されたユーザーとグループのログイン・ポリシー

これらは、他のライセンスと共に DB ライセンスの下に表示されます。

==========

それぞれについて説明します。一部のものはそれ自体が説明になっています。しかし、さらに説明が必要なものもあります。

1 /*

データベース・ユーザー別のオブジェクト特権。

オブジェクトには、表、ビュー、プロシージャーおよび関数が含まれます。

これらはユーザーにのみ付与される特権であり、グループやグループのメンバーシップは含まれていません。

*/

2. /*

グループ別のオブジェクト特権。

オブジェクトには、表、ビュー、プロシージャーおよび関数が含まれます。

これらはグループにのみ付与される特権です。

*/

3 /* ユーザーに付与されたシステム権限とグループ。

*/

4 /* ユーザーとグループの被付与者に付与されたシステム権限とグループ。

*/

5 /* PUBLIC によるオブジェクト・アクセス。

表、ビュー、関数およびプロシージャーが含まれます。

*/

6 /* PUBLIC に付与されたプロシージャーと関数に対する実行特権。

*/

7 /* データベース管理者、アクセス権管理者、ユーザー管理者、またはリモート・データベース管理者のデータベース権限を持つユーザーとグループ。

*/

8 /* GRANT オプション付きでユーザーとグループに付与された表とビューの特権。

これは Sybase IQ で唯一許可される GRANT オプション・タイプであることに注意してください。 ルーチンに GRANT オプション付きで権限を付与することはできません。

*/

9 /* ユーザーとグループに付与されたグループ。

*/

10 /* ログイン・オプション設定を指定してユーザーとグループに割り当てられたログイン・ポリシー。 */

GuardAPI を使用して Sybase IQ レポートにデータ・ソースを追加する方法

GuardAPI を使用して各 Sybase IQ レポートにデータ・ソースを追加し、それらのレポートを実行する方法です。

新しい各レポートにデータ・ソースを追加して、各レポートを実行する方法については、下記の例を参照してください。

すべての Sybase IQ ライセンス・レポートに関するデータ・ソースを追加

```
grdapi create_datasource type="Sybase IQ" user=ent password=Guardium123 host=9.70.144.152 name="Sybase IQ entitlement6"
shared=true owner=admin application=CustomDomain port=2638 dbName=sn5qpuff
```

すべての Sybase IQ ライセンス・レポートに対してデータ・ソースを追加

```
grdapi create_datasourceRef_by_name application=CustomTables objName="Sybase IQ Exec priv on proc func to
PUBLIC"datasourceName="Sybase IQ entitlement 6"
```

```
grdapi create_datasourceRef_by_name application=CustomTablesobjName="Sybase IQ Group granted to user and group"
datasourceName="Sybase IQ entitlement 6"
grdapi create_datasourceRef_by_name application=CustomTablesobjName="Sybase IQ Login policy for user group with
login"datasourceName="Sybase IQ entitlement 6"
grdapi create_datasourceRef_by_name application=CustomTablesobjName="Sybase IQ Object Access By Public" datasourceName="Sybase IQ
entitlement 6"
grdapi create_datasourceRef_by_name application=CustomTablesobjName="Sybase IQ Object Privileges By DB User"
datasourceName="Sybase IQ entitlement 6"
grdapi create_datasourceRef_by_name application=CustomTablesobjName="Sybase IQ Object Privileges By Group" datasourceName="Sybase
IQ entitlement 6"
grdapi create_datasourceRef_by_name application=CustomTablesobjName="Sybase IQ System Authority And Group Granted To
User"datasourceName="Sybase IQ entitlement 6"
grdapi create_datasourceRef_by_name application=CustomTablesobjName="Sybase IQ System Authority And Group Granted To User And
Group"datasourceName="Sybase IQ entitlement 6"
grdapi create_datasourceRef_by_name application=CustomTablesobjName="Sybase IQ Table View priv granted with
grant"datasourceName="Sybase IQ entitlement 6"
grdapi create_datasourceRef_by_name application=CustomTablesobjName="Sybase IQ User Group With DBA Perms Admin
etc"datasourceName="Sybase IQ entitlement 6"
```

すべての Sybase IQ ライセンス・レポートを実行

```
grdapi upload_custom_data tableName=SYBASEIQ_EXEC_PRIV_ON_PROC_FUNC_TO_PUBLIC
grdapi upload_custom_data tableName=SYBASEIQ_GROUP_GRANTED_TO_USER_AND_GROUP
grdapi upload_custom_data tableName=SYBASE_OBJ_COL_PRIVS_GRANTED_WITH_GRAN
grdapi upload_custom_data tableName=SYBASEIQ_OBJECT_ACCESS_BY_PUBLIC
grdapi upload_custom_data tableName=SYBASEIQ_OBJECT_PRIVS_BY_DB_USER
grdapi upload_custom_data tableName=SYBASEIQ_OBJECT_PRIVILEGES_BY_GROUP
grdapi upload_custom_data tableName=SYBASEIQ_SYSTEM_AUTHORITY_AND_GROUP_GRANTED_TO_USER grdapi upload_custom_data
tableName=SYBASEIQ_SYSTEM_AUTHORITY_AND_GROUP_GRANTED_TO_USER_AND_GROUP grdapi upload_custom_data
tableName=SYBASEIQ_TABLE_VIEWS_PRIV_GRANTED_WITH_GRANT grdapi upload_custom_data
tableName=SYBASEIQ_USER_GROUP_WITH_DBA_PERMS_ADMIN_ETC
```

Informix DB ライセンス

以下のドメインは、Informix DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

- データベース・アカウントによる Informix オブジェクト特権 (システム・アカウントとロールを除く)
- GRANT オプション付きでユーザーに付与された Informix データベース・レベル特権、ロール、および言語
- GRANT オプション付きでユーザーおよびロールに付与された Informix データベース・レベル特権、ロール、および言語
- Informix PUBLIC に付与されたオブジェクト権限
- PUBLIC に付与された Informix プロシージャーおよび関数に対する Informix 実行特権
- DBA 特権付きの Informix アカウント GRANT オプション付きで付与された Informix オブジェクト特権および列特権
- ユーザーおよびロールに権限付与された Informix ロール

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

/* 以下の表およびビューに対する選択特権は必須です */

システム・カタログの SELECT 特権については、すべてのユーザーが十分な特権を持っているため、どのユーザーにも特権を付与する必要はありません。Informix は、ユーザーに対してシステム・カタログを付与しないようです。通常は、以下の権限付与が使用されます。 ただしこの場合は必要ありません。

grant select on systables to sqlguard;

grant select on systabauth to sqlguard;

grant select on sysusers to sqlguard;

grant select on sysroleauth to sqlguard;

grant select on syslangauth to sqlguard;

grant select on sysroutinelangs to sqlguard;

grant select on sysprocauth to sqlguard;

grant select on sysprocedures to sqlguard;

grant select on syscolauth to sqlguard;

データ・ソースに Informix データベース・タイプがあるものの、データベース名がない場合 (「データ・ソース定義」を確認し、「ロケーション」の下にあるデータベース名が空である場合) は、データのアップロードが、ユーザーがアクセス権を持つすべての Informix データベースでループします。

Microsoft SQL Server 2005 以降の DB ライセンス

以下のドメインは、Microsoft SQL Server 2005 以降の DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「**DB ライセンス**」タブを表示します。

注: 動的照会ストリング内のオブジェクトは、xxx_DEPENDENCIES には表示されません。保管されたプログラム単位によって呼び出された EXECUTE IMMEDIATE SQL ストリング内のオブジェクトは、従属関係を表示しません。この照会は、グループ ID 202「Dependencies_exclude_schema-MSSQL」で定義されたスキーマ所有者を除外します。ユーザーは、従属関係照会を行うために、このグループのスキーマ名を追加または除去できます。

- Microsoft SQL Server デフォルトのシステム・ユーザーを除くデータベース・アカウントによるオブジェクト特権
- Microsoft SQL Server ユーザーに付与されたロール/システム特権
- Microsoft SQL Server GRANT オプション付きでユーザーおよびロールに付与されたロールおよびシステム特権
- Microsoft SQL Server PUBLIC によるオブジェクト・アクセス
- Microsoft SQL Server PUBLIC に割り当てられたシステム・プロシージャーおよび関数に対する実行特権
- Microsoft SQL Server db_owner ロールおよび db_securityadmin ロールのデータベース・アカウント
- Microsoft SQL Server sysadmin、serveradmin、および security admin のサーバー・アカウント /* MASTER データベースに対してのみ実行します */
- Microsoft SQL Server GRANT オプション付きで付与されたオブジェクト特権および列特権
- Microsoft SQL Server ユーザーおよびロールに付与されたロール

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

```
/* 以下の表およびビューに対する選択特権は必須です */

/* 以下は MASTER データベースでは必須です */

grant select on sys.server_principals to sqlguard


/*以下は MASTER を含むすべてのデータベースで必須です */

grant select on sys.database_permissions to sqlguard

grant select on sys.database_principals to sqlguard

grant select on sys.all_objects to sqlguard

grant select on sys.database_role_members to sqlguard

grant select on sys.columns to sqlguard
```

データ・ソースに MSSQL データベース・タイプがあるものの、データベース名がない場合 (「データ・ソース定義」を確認し、「ロケーション」の下にあるデータベース名が空である場合) は、データのアップロードが、ユーザーがアクセス権を持つすべての MSSQL データベースでループします。

Netezza DB ライセンス

以下のドメインは、Netezza DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「**DB ライセンス**」タブを表示します。

注: Netezza では、データベース・エラーのテキスト変換は行われません。エラーは例外の記述に表示されます。ユーザーは、必要に応じて Netezza の例外の記述を含むレポートのコピーを作成したり、追加したりできます。

- 「Netezza オブジェクト特権 (データベース・ユーザー名別)」 - ADMIN アカウント以外のデータベース・ユーザー名により、GRANT オプション付きまたはなしで付与されたオブジェクト特権
- 「Netezza 管理者特権 (データベース・ユーザー名別)」 - ADMIN アカウント以外のデータベース・ユーザー名により、GRANT オプション付きまたはなしで付与された管理者特権
- 「Netezza ユーザーに権限付与されたグループ/ロール」 - ユーザーに権限付与されたグループ (ロール)
- 「Netezza オブジェクト特権 (グループ別)」 - PUBLIC を除くグループにより、GRANT オプション付きまたはなしで付与されたオブジェクト特権
- 「Netezza 管理者特権 (グループ別)」 - PUBLIC を除くグループにより、GRANT オプション付きまたはなしで付与された管理者特権
- 「Netezza 管理者特権 (データベース・ユーザー名グループ別)」 - ADMIN アカウントおよび PUBLIC グループを除くデータベース・ユーザー名およびグループにより、GRANT オプション付きまたはなしで付与された管理者特権
- 「Netezza 付与されたオブジェクト特権」 - GRANT オプション付きまたはなしで PUBLIC に対して付与されたオブジェクト特権
- 「Netezza 付与された管理者特権」 - GRANT オプション付きまたはなしで PUBLIC に対して付与された管理特権
- 「Netezza ユーザーおよびグループに対するグローバル管理者特権」 - ADMIN アカウントを除くユーザーおよびグループに付与されたグローバル管理者特権
- 「Netezza ユーザーおよびグループに対するグローバル・オブジェクト特権」 - ADMIN アカウントを除くユーザーおよびグループに付与されたグローバル・オブジェクト特権

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

```
/* 以下の表およびビューに対する選択特権は必須です */

/* このスクリプトは、システム・データベースから実行する必要があります */
```

GRANT SELECT ON SYSTEM VIEW TO sqlguard;

GRANT LIST ON DATABASE TO sqlguard;

GRANT LIST ON USER TO sqlguard;

GRANT LIST ON GROUP TO sqlguard;

GRANT SELECT ON _V_CONNECTION TO sqlguard;

Netezza ライセンス照会では、特にこれらのレポートを実行する予定のユーザーに対して特権を付与する際に、システム・データベースへの接続が推奨されます。特権の付与は、システム・データベースから行う必要があります。それ以外のデータベースから付与された特権は、その特定のデータベースでのみ有効になります。システム・データベースから特権の付与が行われた場合は、特殊機構により、付与された特権がすべてのデータベースで有効になります。

Teradata DB ライセンス

以下のドメインは、Teradata DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、**「DB ライセンス」**タブを表示します。

- デフォルトのシステム・ユーザーを除くデータベース・アカウントにより付与された Teradata オブジェクト特権
- GRANT オプション付きで Teradata ユーザーに付与されたシステム特権とロール
- GRANT オプション付きで Teradata ユーザーおよびロールに権限付与されたロール
- Teradata ユーザーおよびロールに権限付与されたロール。 GRANT オプション付きでユーザーおよびロールに付与されたシステム特権。
- PUBLIC に対して付与された Teradata オブジェクト特権およびシステム特権。Teradata では、PUBLIC に対してロールを権限付与できないことに注意してください。
- PUBLIC に対して付与されたシステム・データベース・オブジェクトに対する Teradata 実行特権
- ユーザーおよびロールに付与された Teradata システム管理者特権およびセキュリティー管理者特権
  注: Teradata には、システム管理者またはセキュリティー管理者というロールはありません。ユーザーは独自のロールを作成する必要があります。次のような重要なシステム特権は、通常、一般的なユーザーに付与されません: ABORT SESSION、CREATE DATABASE、CREATE PROFILE、CREATE ROLE、CREATE USER、DROP DATABASE、DROP PROFILE、DROP ROLE、DROP USER、MONITOR RESOURCE、MONITOR SESSION、REPLICATION OVERRIDE、SET SESSION RATE、SET RESOURCE RATE。
- GRANT オプション付きでユーザーに付与された Teradata オブジェクト特権。DBC および grantee = 'All' は含みません。

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

/* 以下の表およびビューに対する選択特権は必須です */

GRANT SELECT ON DBC.AllRights TO sqlguard;

GRANT SELECT ON DBC.Tables TO sqlguard;

GRANT SELECT ON DBC.AllRoleRights TO sqlguard;

GRANT SELECT ON DBC.RoleMembers TO sqlguard;

PostgreSQL DB ライセンス

以下のドメインは、PostgreSQL DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、**「DB ライセンス」**タブを表示します。

PostgreSQL には、7 つのライセンス・カスタム・ドメイン、照会、レポートがあります。これを以下に示します (それぞれについてレポート名、説明、注記を示します)。

- 「PostgreSQL GRANT オプションあり/なしで PUBLIC ユーザー・ロールに付与されたデータベースに対する特権」。GRANT オプション付きまたはなしで、PUBLIC、ユーザーおよびロールに付与されたデータベースに対する特権です。任意のデータベース (理想的には PostgreSQL) でこれを実行します。
- 「PostgreSQL GRANT オプションあり/なしで PUBLIC ユーザー・ロールに付与された言語に対する特権」。GRANT オプション付きまたはなしで、PUBLIC、ユーザーおよびロールに対して付与された言語に対する特権です。
- 「PostgreSQL GRANT オプションあり/なしで PUBLIC ユーザー・ロールに付与されたスキーマに対する特権」。GRANT オプション付きまたはなしで、PUBLIC、ユーザーおよびロールに付与されたスキーマに対する特権です。
- 「PostgreSQL GRANT オプションあり/なしで PUBLIC ユーザー・ロールに付与された表スペースに対する特権」。GRANT オプション付きまたはなしで、PUBLIC、ユーザーおよびロールに対して付与された表スペースに対する特権です。任意のデータベース (理想的には PostgreSQL) でこれを実行します。
- 「ユーザーまたはロールに付与された PostgreSQL ロールまたはユーザー」。GRANT オプション付きでユーザーまたはロールに権限付与されたロールまたはユーザーです。任意のデータベースでこれを一度実行します。理想的には PostgreSQL で実行します。
- 「PostgreSQL ユーザーまたはロールに権限付与されたスーパーユーザー」。ユーザーまたはロールに権限付与されたスーパーユーザーです。任意のデータベースでこれを一度実行します。理想的には PostgreSQL で実行します。
- 「PostgreSQL ユーザーおよびロールに付与されたシステム特権」。ユーザーおよびロールに付与されたシステム特権です。任意のデータベースでこれを一度実行します。理想的には PostgreSQL で実行します。
- 「PostgreSQL PUBLIC に付与された表、ビュー、シーケンス、および関数特権」。PUBLIC に対して付与された、表、ビュー、シーケンス、および関数特権です。データベースごとにこれを実行します。
- 「PostgreSQL GRANT オプション付きで付与された表、ビュー、シーケンス、および関数特権」。GRANT オプションのみを付加して、ユーザーおよびロールに付与された表、ビュー、シーケンス、および関数特権です。PostgreSQL アカウントを除きます。

- 「PostgreSQL ロールに付与された表、ビュー、シーケンス、関数特権」。ロールに付与された、表、ビュー、シーケンス、および関数特権です。 PUBLIC は除きます。
- 「PostgreSQL ログインに付与された表、ビュー、シーケンス、および関数特権」。ログインに付与された、表、ビュー、シーケンス、および関数特権です。postgres システム・ユーザーを除きます。

注: バージョン 8.3.6 以降、PostgreSQL では PUBLIC に対する管理者オプションの付与をサポートしていません。関数のみで、ストアード・プロシージャーはありません。表の権限付与のみがサポートされ、列の権限付与はサポートされていません。PUBLIC はグループであり、ユーザーではありません。PUBLIC は、pg_roles には表示されません。これらのすべての照会を実行する必要がある特権は、「GRANT CONNECT ON DATABASE PostgreSQL TO username;」のみです。

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

/* 以下の表およびビューに対する選択特権は必須です */

/*これは POSTGRES データベースで必須です。*/

grant connect on database postgres to sqlguard;

/*以下は POSTGRES を含むすべてのデータベースで必須です (デフォルトで既に PUBLIC に付与されています) */

grant select on pg_class to sqlguard;

grant select on pg_namespace to sqlguard;

grant select on pg_roles to sqlguard;

grant select on pg_proc to sqlguard;

grant select on pg_auth_members to sqlguard;

grant select on pg_language to sqlguard;

grant select on pg_tablespace to sqlguard;

grant select on pg_database to sqlguard;

データ・ソースに PostgreSQL データベース・タイプがあるものの、データベース名がない場合 (「データ・ソース定義」を確認し、「ロケーション」の下にあるデータベース名が空である場合) は、データのアップロードが、ユーザーがアクセス権を持つすべての PostgreSQL データベースでループします。

親トピック: ドメイン、エンティティー、および属性

カスタム・ドメイン

カスタム・ドメインではユーザー定義のドメインが可能であり、Guardium システムにアップロードされる任意のデータ表を定義できます。

これらのカスタム・ライセンス (特権) ドメインを使用するということは、ライセンス・レポートを使用するということです。ライセンス・レポートには、ユーザーとしてログインした場合にアクセスできます。 これらのレポートを表示するには、「ユーザー」タブの「データベース特権」に移動します。

いくつかのカスタム・ドメインが事前定義されています。これらを使用してデータをインポートできます。

カスタム・ドメインを作成して、直後にカスタム・クエリー・レポート・ビルダーで表示しない場合は、UI からログアウトして再びログインします。

カスタム表およびカスタム・ドメインの使用について詳しくは、外部データ相関を参照してください。

[カスタム] アクセス

このドメインには、標準のデータ・アクセス・ドメインと同じエンティティーがすべて含まれています。これがカスタム・ドメインとして提供されることにより、このドメインの情報と、ユーザーによって既にアップロードされた任意のカスタム表の情報を含む追加のユーザー定義ドメインを作成できます。 [カスタム] アクセス・ドメインは、コピーする必要があります。このドメインはバージョンごとに更新されるため、このドメイン上でレポートを作成することはお勧めしません。アクセス・ドメインに含まれるエンティティーについての説明は、『ドメイン』のトピックのアクセス・ドメインに関する説明を参照してください。

S-TAP 情報 (中央マネージャー)

レポート:『S-TAP® レポート』を参照。中央マネージャーでは、追加のレポート、「S-TAP 情報」が使用可能です。このレポートは、環境全体の S-TAP をモニターします。このデータのアップロードには、カスタム表ビルダーを使用します。

S-TAP 情報は、「S-TAP 情報」エンティティーが含まれている事前定義されたカスタム・ドメインであり、変更することはできません。

カスタム照会を定義する際は、アップロード・ページに移動して「検査/修復」をクリックし、CUSTOM データベースにカスタム表を作成します。そうしないと、照会を保存するときに照会が検証されません。この表は、すべてのリモート・ソースから自動的にロードします。ユーザーは、使用するリモート・ソースを選択できません。すべてのリモート・ソースから取り込まれます。

このカスタム表とカスタム・ドメインに基づく、次の 2 つのレポートがあります。

エンタープライズ S-TAP ビューは、中央マネージャーから、コレクターまたは管理対象ユニット上のアクティブな S-TAP に関する情報を表示します (同じ S-TAP エンジンに対する重複があり、一方がアクティブで、他方が非アクティブの場合、アクティブな方のみがレポートに使用されます)。

「詳細なエンタープライズ S-TAP ビュー」は、中央マネージャーから、すべてのコレクターおよび/または管理対象ユニット上のすべてのアクティブおよびパッシブな S-TAP に関する情報を表示します。

エンタープライズ S-TAP ビューと詳細なエンタープライズ S-TAP ビューが同じに見える場合は、1 つの管理対象ユニット上にあるただ 1 つの S-TAP が表示されているためです。複数の S-TAP および複数の管理対象ユニットがある場合は、詳細なエンタープライズ S-TAP ビューの表示が違ったものになります。

これらの 2 つのレポートは、スタンドアロン・システムの「TAP モニター」タブから選択可能ですが、情報は表示されません。

DB ライセンス・ドメイン

ユーザーの認証およびデータに対するロールに基づいたアクセス権の制限に加え、最も多くの特権を持つデータベース・ユーザーに対しても、定期的なライセンス・レビューを行う必要があります。このレビューは、ユーザーが自分の業務を行うのに必要な特権のみを持っていることを検証および確認するプロセスです。これは、データベース・ユーザー権限の認証レポート作成とも呼ばれます。

Guardium の事前定義データベース・ライセンス (特権) レポートを使用して、(例えば) システム特権を持つユーザーや、他のユーザーやロールにこれらの特権を付与したユーザーを確認します。データベース・ライセンス・レポートは、データベース・アクセスの変更をトラッキングしたり、使用されないまま残っているアカウントや誤って付与された特権によるセキュリティー・ホールが存在しないことを確認したりする監査員にとって重要なものです。

DB ライセンス・レポートでは、カスタム・ドメイン機能を使用して、選択したデータベース上の外部データと事前定義ライセンス・レポートの内部データとのリンクを作成します。事前定義データベース・ライセンス・レポートの使用法について詳しくは、『データベース・ライセンス・レポート』を参照してください。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

注: DB ライセンス・レポートは、プロダクト・キーにより有効になるオプションのコンポーネントです。これらのコンポーネントが有効になっていない場合は、カスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーの選択に、選択項目が表示されません。

事前定義ライセンス・レポートを以下にリストします。これらは、カスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーの選択でドメイン名として表示されます。

- Oracle DB ライセンス
- MYSQL DB ライセンス
- DB2® DB ライセンス
- SYBASE DB ライセンス
- Informix® DB ライセンス
- Microsoft SQL Server DB ライセンス
- Netezza® DB ライセンス
- Teradata DB ライセンス
- PostgreSQL DB ライセンス

Oracle DB ライセンス

以下のドメインは、Oracle DB ライセンスに関するアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

Oracle

- 「ORA ALTER SYSTEM のアカウント」 - ALTER SYSTEM 特権および ALTER SESSION 特権を持つアカウント
- 「ORA BECOME USER 特権を持つアカウント」 - BECOME USER 特権を持つアカウント
- 「ORA 全システム特権および ADMIN オプション」 - ユーザーおよびロールに対するすべてのシステム特権および管理者オプションを示すレポート
- 「ORA オブジェクトおよび列特権」 - 付与されているオブジェクト特権および列特権 (GRANT オプション付きまたはなし)
- 「ORA PUBLIC によるオブジェクト・アクセス」 - PUBLIC によるオブジェクト・アクセス
- 「ORA オブジェクト特権」 - SYS 内になく、DBA ロールではないデータベース・アカウントによるオブジェクト特権
- 「ORA SYS プロシージャーに対する PUBLIC 実行特権」 - PUBLIC に割り当てられている SYS PL/SQL プロシージャーに対する実行特権
- 「ORA 権限付与されたロール」 - ユーザーおよびロールに権限付与されたロール
- 「ORA 権限付与されたシステム特権」 - 再帰的定義 (特権がロールに割り当てられ、そのロールがユーザーに割り当てられた状態) を含む、ユーザーに付与されたシステム特権を示す階層レポート
- 「ORA SYSDBA および SYSOPER アカウント」 - SYSDBA 特権および SYSOPER 特権を持つアカウント

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

```
/* 以下の表およびビューに対する選択特権は必須です */

grant select on sys.dba_tab_privs to sqlguard;

grant select on sys.dba_roles to sqlguard;

grant select on sys.dba_users to sqlguard;

grant select on sys.dba_role_privs to sqlguard;

grant select on sys.dba_sys_privs to sqlguard;

grant select on sys.obj$ to sqlguard;

grant select on sys.user$ to sqlguard;
```

grant select on sys.objauth$ to sqlguard;

grant select on sys.table_privilege_map to sqlguard;

grant select on sys.dba_objects to sqlguard;

grant select on sys.v_$pwfile_users to sqlguard;

grant select on sys.dba_col_privs to sqlguard;

MYSQL DB ライセンス

以下のドメインは、MYSQL DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

MYSQL: 末尾が_40 である照会では、最も基本的なバージョンの mysql スキーマ (MySQL 4.0 以降用) を使用します。information_schema は MySQL 5.0 で導入されてから変更されていないため、末尾が _50 の照会はありますが、末尾が _51 の照会はありません。 末尾が _50 の照会は、MySQL 5.0 および 5.1 で動作します。 また、information_schema は 6.0 でも変更される予定がないため、6.0 がリリースされた際には 6.0 でも動作します。末尾が _502 の照会 (MYSQL502) では、新しい information_schema を使用します。これにはより多くの情報が含まれ、実際のデータ・ディクショナリーにより一層類似しています。

- MYSQL データベース特権 40
- MYSQL ユーザー特権 40
- MYSQL ホスト特権 40
- MYSQL 表特権 40
- MYSQL データベース特権 500
- MYSQL ユーザー特権 500
- MYSQL ホスト特権 500
- MYSQL 表特権 500
- MYSQL データベース特権 502
- MYSQL ユーザー特権 502
- MYSQL ホスト特権 502
- MYSQL 表特権 502

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリストで、データベース表内 (またはデータベース表のビュー内) でライセンスが機能するために最小限必要な特権について説明します。

注: データをアップロードするには、必要な特権に加え、ユーザーが MYSQL データベースに接続することが必要です。

MySQL の全バージョンについて、MySQL 5.0.1 を使用したライセンス照会では、表集合 mysql.db mysql.host mysql.tables_priv mysql.user を使用します。

MySQL 5.0.2 以降の全バージョンについて、ライセンス照会では表集合 information_schema.SCHEMA_PRIVILEGES mysql.host information_schema.TABLE_PRIVILEGES information_schema.USER_PRIVILEGES を使用します。

データ・ソースに MYSQL データベース・タイプがあるものの、データベース名がない場合 (「データ・ソース定義」を確認し、「ロケーション」の下にあるデータベース名が空である場合) は、データのアップロードが、ユーザーがアクセス権を持つすべての MYSQL データベースでループします。

Db2 DB ライセンス

以下のドメインは、Db2 DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

- Db2 列レベルの特権 (SELECT、UPDATE など)
- Db2 データベース・レベルの特権 (CONNECT、CREATE など)
- Db2 索引レベルの特権 (CONTROL)
- Db2 パッケージ・レベルの特権 (コード・パッケージ対象の BIND、EXECUTE など)
- Db2 表レベルの特権 (SELECT、UPDATE など)Db2 特権サマリー

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

/* 以下の表およびビューに対する選択特権は必須です */

GRANT SELECT ON SYSCAT.COLAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.DBAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.INDEXAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.PACKAGEAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.DBAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.TABAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.SCHEMAAUTH TO SQLGUARD;

GRANT SELECT ON SYSCAT.PASSTHRUAUTH TO SQLGUARD;

SYBASE DB ライセンス

以下のドメインは、SYBASE DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

- GRANT オプションを含む、SYBASE ユーザーに付与されたシステム特権とロール
- GRANT オプションを含む、SYBASE ユーザーに権限付与されたロールおよびユーザーとロールに付与されたシステム特権
- SYBASE PUBLIC によるオブジェクト・アクセス
- PUBLIC に割り当てられた、プロシージャーおよび関数に対する SYBASE 実行特権
- システムまたはセキュリティー admin ロールを持つ SYBASE アカウント
- GRANT オプション付きで付与された SYBASE オブジェクト特権および列特権
- SYBASE ユーザーに権限付与されたロール

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

/* 以下の表およびビューに対する選択特権は必須です */

/* 以下は MASTER データベースでは必須です */

grant select on master.dbo.sysloginroles to sqlguard

grant select on master.dbo.syslogins to sqlguard

grant select on master.dbo.syssrvroles to sqlguard

/*以下は MASTER を含むすべてのデータベースで必須です */

grant select on sysprotects to sqlguard

grant select on sysusers to sqlguard

grant select on sysobjects to sqlguard

grant select on sysroles to sqlguard

データ・ソースに SYBASE データベース・タイプがあるものの、データベース名がない場合 (「データ・ソース定義」を確認し、「ロケーション」の下にあるデータベース名が空である場合) は、データのアップロードが、ユーザーがアクセス権を持つすべての SYBASE データベースでループします。

Informix DB ライセンス

以下のドメインは、Informix DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「DB ライセンス」タブを表示します。

- データベース・アカウントによる Informix オブジェクト特権 (システム・アカウントとロールを除く)
- GRANT オプション付きでユーザーに付与された Informix データベース・レベル特権、ロール、および言語
- GRANT オプション付きでユーザーおよびロールに付与された Informix データベース・レベル特権、ロール、および言語
- Informix PUBLIC に付与されたオブジェクト権限
- PUBLIC に付与された Informix プロシージャーおよび関数に対する Informix 実行特権
- DBA 特権付きの Informix アカウント GRANT オプション付きで付与された Informix オブジェクト特権および列特権
- ユーザーおよびロールに権限付与された Informix ロール

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

/* 以下の表およびビューに対する選択特権は必須です */

システム・カタログの SELECT 特権については、すべてのユーザーが十分な特権を持っているため、どのユーザーにも特権を付与する必要はありません。Informix は、ユーザーに対してシステム・カタログを付与しないようです。通常は、権限付与が使用されます。 ただしこの場合は必要ありません。

grant select on systables to sqlguard;

grant select on systabauth to sqlguard;

grant select on sysusers to sqlguard;

grant select on sysroleauth to sqlguard;

grant select on syslangauth to sqlguard;

grant select on sysroutinelangs to sqlguard;

grant select on sysprocauth to sqlguard;

grant select on sysprocedures to sqlguard;

grant select on syscolauth to sqlguard;

データ・ソースに Informix データベース・タイプがあるものの、データベース名がない場合 (「データ・ソース定義」を確認し、「ロケーション」の下にあるデータベース名が空である場合) は、データのアップロードが、ユーザーがアクセス権を持つすべての Informix データベースでループします。

Microsoft SQL Server 2005 以降の DB ライセンス

以下のドメインは、Microsoft SQL Server 2005 以降の DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・クエリー・レポート・ビルダー、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「**DB ライセンス**」タブを表示します。

注: 動的照会ストリング内のオブジェクトは、xxx_DEPENDENCIES には表示されません。保管されたプログラム単位によって呼び出された EXECUTE IMMEDIATE SQL ストリング内のオブジェクトは、従属関係を表示しません。この照会は、グループ ID 202「Dependencies_exclude_schema-MSSQL」で定義されたスキーマ所有者を除外します。ユーザーは、従属関係照会を行うために、このグループのスキーマ名を追加または除去できます。

- Microsoft SQL Server デフォルトのシステム・ユーザーを除くデータベース・アカウントによるオブジェクト特権
- Microsoft SQL Server ユーザーに付与されたロール/システム特権
- Microsoft SQL Server GRANT オプション付きでユーザーおよびロールに付与されたロールおよびシステム特権
- Microsoft SQL Server PUBLIC によるオブジェクト・アクセス
- Microsoft SQL Server PUBLIC に割り当てられたシステム・プロシージャーおよび関数に対する実行特権
- Microsoft SQL Server db_owner ロールおよび db_securityadmin ロールのデータベース・アカウント
- Microsoft SQL Server sysadmin、serveradmin、および security admin のサーバー・アカウント /* MASTER データベースに対してのみ実行します */
- Microsoft SQL Server GRANT オプション付きで付与されたオブジェクト特権および列特権
- Microsoft SQL Server ユーザーおよびロールに付与されたロール

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

/* 以下の表およびビューに対する選択特権は必須です */

/* 以下は MASTER データベースでは必須です */

grant select on sys.server_principals to sqlguard

/*以下は MASTER を含むすべてのデータベースで必須です */

grant select on sys.database_permissions to sqlguard

grant select on sys.database_principals to sqlguard

grant select on sys.all_objects to sqlguard

grant select on sys.database_role_members to sqlguard

grant select on sys.columns to sqlguard

データ・ソースに MSSQL データベース・タイプがあるものの、データベース名がない場合 (「データ・ソース定義」を確認し、「ロケーション」の下にあるデータベース名が空である場合) は、データのアップロードが、ユーザーがアクセス権を持つすべての MSSQL データベースでループします。

Netezza DB ライセンス

以下のドメインは、Netezza DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・ドメイン照会、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、「**DB ライセンス**」タブを表示します。

注: Netezza では、データベース・エラーのテキスト変換は行われません。エラーは例外の記述に表示されます。ユーザーは、必要に応じて Netezza の例外の記述を含むレポートのコピーを作成したり、追加したりできます。

- 「Netezza オブジェクト特権 (データベース・ユーザー名別)」 - ADMIN アカウント以外のデータベース・ユーザー名により、GRANT オプション付きまたはなしで付与されたオブジェクト特権
- 「Netezza 管理者特権 (データベース・ユーザー名別)」 - ADMIN アカウント以外のデータベース・ユーザー名により、GRANT オプション付きまたはなしで付与された管理者特権

- 「Netezza ユーザーに権限付与されたグループ/ロール」 - ユーザーに権限付与されたグループ (ロール)
- 「Netezza オブジェクト特権 (グループ別)」 - PUBLIC を除くグループにより、GRANT オプション付きまたはなしで付与されたオブジェクト特権
- 「Netezza 管理者特権 (グループ別)」 - PUBLIC を除くグループにより、GRANT オプション付きまたはなしで付与された管理者特権
- 「Netezza 管理者特権 (データベース・ユーザー名グループ別)」 - ADMIN アカウントおよび PUBLIC グループを除くデータベース・ユーザー名およびグループにより、GRANT オプション付きまたはなしで付与された管理者特権
- 「Netezza 付与されたオブジェクト特権」 - GRANT オプション付きまたはなしで PUBLIC に対して付与されたオブジェクト特権
- 「Netezza 付与された管理者特権」 - GRANT オプション付きまたはなしで PUBLIC に対して付与された管理特権
- 「Netezza ユーザーおよびグループに対するグローバル管理者特権」 - ADMIN アカウントを除くユーザーおよびグループに付与されたグローバル管理者特権
- 「Netezza ユーザーおよびグループに対するグローバル・オブジェクト特権」 - ADMIN アカウントを除くユーザーおよびグループに付与されたグローバル・オブジェクト特権

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

```
/* 以下の表およびビューに対する選択特権は必須です */

/* このスクリプトは、システム・データベースから実行する必要があります */

GRANT SELECT ON SYSTEM VIEW TO sqlguard;

GRANT LIST ON DATABASE TO sqlguard;

GRANT LIST ON USER TO sqlguard;

GRANT LIST ON GROUP TO sqlguard;

GRANT SELECT ON _V_CONNECTION TO sqlguard;
```

Netezza ライセンス照会では、特にこれらのレポートを実行する予定のユーザーに対して特権を付与する際に、システム・データベースへの接続が推奨されます。特権の付与は、システム・データベースから行う必要があります。それ以外のデータベースから付与された特権は、その特定のデータベースでのみ有効になります。システム・データベースから特権の付与が行われた場合は、特殊機構により、付与された特権がすべてのデータベースで有効になります。

Teradata DB ライセンス

以下のドメインは、Teradata DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・ドメイン照会、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、**「DB ライセンス」**タブを表示します。

- デフォルトのシステム・ユーザーを除くデータベース・アカウントにより付与された Teradata オブジェクト特権
- GRANT オプション付きで Teradata ユーザーに付与されたシステム特権とロール
- GRANT オプション付きで Teradata ユーザーおよびロールに権限付与されたロール
- Teradata ユーザーおよびロールに権限付与されたロール。 GRANT オプション付きでユーザーおよびロールに付与されたシステム特権。
- PUBLIC に対して付与された Teradata オブジェクト特権およびシステム特権。Teradata では、PUBLIC に対してロールを権限付与できないことに注意してください。
- PUBLIC に対して付与されたシステム・データベース・オブジェクトに対する Teradata 実行特権
- ユーザーおよびロールに付与された Teradata システム管理者特権およびセキュリティー管理者特権
  注: Teradata には、システム管理者またはセキュリティー管理者というロールはありません。ユーザーは独自のロールを作成する必要があります。次のような重要なシステム特権は、通常、一般的なユーザーに付与されません: ABORT SESSION、CREATE DATABASE、CREATE PROFILE、CREATE ROLE、CREATE USER、DROP DATABASE、DROP PROFILE、DROP ROLE、DROP USER、MONITOR RESOURCE、MONITOR SESSION、REPLICATION OVERRIDE、SET SESSION RATE、SET RESOURCE RATE。
- GRANT オプション付きでユーザーに付与された Teradata オブジェクト特権。DBC および grantee = 'All' は含みません。

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

```
/* 以下の表およびビューに対する選択特権は必須です */

GRANT SELECT ON DBC.AllRights TO sqlguard;

GRANT SELECT ON DBC.Tables TO sqlguard;
```

GRANT SELECT ON DBC.AllRoleRights TO sqlguard;

GRANT SELECT ON DBC.RoleMembers TO sqlguard;

PostgreSQL DB ライセンス

以下のドメインは、PostgreSQL DB ライセンスでのアップロードおよびレポート作成を容易にするために提供されています。以下の各ドメインには、同じ名前を持つ単一のエンティティーと事前定義されたレポートが 1 つあります。これらのドメインは、すべてカスタム・ドメイン・ビルダー、カスタム・ドメイン照会、カスタム表ビルダーで使用することができます。これらは、他の事前定義されたエンティティーやレポートと同様、変更はできませんが、コピーを作成してユーザー独自のドメインやレポートにカスタマイズすることができます。ライセンス・レポートを参照するには、ユーザー・ポータルにログオンし、**「DB ライセンス」**タブを表示します。

PostgreSQL には、7 つのライセンス・カスタム・ドメイン、照会、レポートがあります。これを以下に示します (それぞれについてレポート名、説明、注記を示します)。

- 「PostgreSQL GRANT オプションあり/なしで PUBLIC ユーザー・ロールに付与されたデータベースに対する特権」。GRANT オプション付きまたはなしで、PUBLIC、ユーザーおよびロールに付与されたデータベースに対する特権です。任意のデータベース (理想的には PostgreSQL) でこれを実行します。
- 「PostgreSQL GRANT オプションあり/なしで PUBLIC ユーザー・ロールに付与された言語に対する特権」。GRANT オプション付きまたはなしで、PUBLIC、ユーザーおよびロールに対して付与された言語に対する特権です。
- 「PostgreSQL GRANT オプションあり/なしで PUBLIC ユーザー・ロールに付与されたスキーマに対する特権」。GRANT オプション付きまたはなしで、PUBLIC、ユーザーおよびロールに付与されたスキーマに対する特権です。
- 「PostgreSQL GRANT オプションあり/なしで PUBLIC ユーザー・ロールに付与された表スペースに対する特権」。GRANT オプション付きまたはなしで、PUBLIC、ユーザーおよびロールに対して付与された表スペースに対する特権です。任意のデータベース (理想的には PostgreSQL) でこれを実行します。
- 「ユーザーまたはロールに付与された PostgreSQL ロールまたはユーザー」。GRANT オプション付きでユーザーまたはロールに権限付与されたロールまたはユーザーです。任意のデータベースでこれを一度実行します。理想的には PostgreSQL で実行します。
- 「PostgreSQL ユーザーまたはロールに権限付与されたスーパーユーザー」。ユーザーまたはロールに権限付与されたスーパーユーザーです。任意のデータベースでこれを一度実行します。理想的には PostgreSQL で実行します。
- 「PostgreSQL ユーザーおよびロールに付与されたシステム特権」。ユーザーおよびロールに付与されたシステム特権です。任意のデータベースでこれを一度実行します。理想的には PostgreSQL で実行します。
- 「PostgreSQL PUBLIC に付与された表、ビュー、シーケンス、および関数特権」。PUBLIC に対して付与された、表、ビュー、シーケンス、および関数特権です。データベースごとにこれを実行します。
- 「PostgreSQL GRANT オプション付きで付与された表、ビュー、シーケンス、および関数特権」。GRANT オプションのみを付加して、ユーザーおよびロールに付与された表、ビュー、シーケンス、および関数特権です。PostgreSQL アカウントを除きます。
- 「PostgreSQL ロールに付与された表、ビュー、シーケンス、関数特権」。ロールに付与された、表、ビュー、シーケンス、および関数特権です。 PUBLIC は除きます。
- 「PostgreSQL ログインに付与された表、ビュー、シーケンス、および関数特権」。ログインに付与された、表、ビュー、シーケンス、および関数特権です。postgres システム・ユーザーを除きます。

注: バージョン 8.3.6 以降、PostgreSQL では PUBLIC に対する管理者オプションの付与をサポートしていません。関数のみで、ストアード・プロシージャーはありません。表の権限付与のみがサポートされ、列の権限付与はサポートされていません。PUBLIC はグループであり、ユーザーではありません。PUBLIC は、pg_roles には表示されません。これらのすべての照会を実行する必要がある特権は、「GRANT CONNECT ON DATABASE PostgreSQL TO username;」のみです。

さまざまなデータ・ソースからデータをアップロードできるようにするライセンスの場合、通常、データベースのアクセスに使用するログインが、照会で使用される表 (すべてのライセンスに対して非表示) を読み取れる必要があります。

以下のリスト (コメント行の見出し付き) は、ライセンスを機能させるために必要な、データベース表内での (またはデータベース表のビュー内での) 最小限の特権を示します。

/* 以下の表およびビューに対する選択特権は必須です */

/*これは POSTGRES データベースで必須です。*/

grant connect on database postgres to sqlguard;

/*以下は POSTGRES を含むすべてのデータベースで必須です (デフォルトで既に PUBLIC に付与されています) */

grant select on pg_class to sqlguard;

grant select on pg_namespace to sqlguard;

grant select on pg_roles to sqlguard;

grant select on pg_proc to sqlguard;

grant select on pg_auth_members to sqlguard;

grant select on pg_language to sqlguard;

grant select on pg_tablespace to sqlguard;

```
grant select on pg_database  to sqlguard;
```

データ・ソースに PostgreSQL データベース・タイプがあるものの、データベース名がない場合 (「データ・ソース定義」を確認し、「ロケーション」の下にあるデータベース名が空である場合) は、データのアップロードが、ユーザーがアクセス権を持つすべての PostgreSQL データベースでループします。

親トピック: レポート

データマート

データを後で使用するために抽出するには、データマートを使用します。データマートは、アクセス頻度の高いデータをさらに効率的に保管するために使用します。データをパージ時間の後に保存したり、Guardium からデータをエクスポートしたり、配布レポートを作成したりする場合に使用します。

データマートには次の 2 つのタイプがあります。

- データを表に抽出します (Guardium)。
- 外部システムにエクスポートするためにデータをファイルに抽出します。

データマートは、複数のアグリゲーターやコレクターからのデータを結合して配布レポートを作成するために使用できます (配布レポート・ビルダー を参照)。

一元管理およびデータマート
一元管理環境では、構成は、管理対象ユニットに自動的に配布されます。管理対象ユニットでは、抽出スケジュールを指定変更できます。中央マネージャーが複数の場合は、エクスポート/インポート機能を使用してデータマート定義をコピーすることができます。

- 表へのデータマートの抽出
  データマート表は、アクセス頻度の高いデータ、レポートで使用できるようにしたいデータが大量にあるシステム、およびスケジュール済みのパージの後に保存したいデータに使用します。
- ファイルへのデータマートの抽出
  ニーズに対応するために、CSV ファイルへのデータ抽出を作成できます。
- ファイルへの事前定義データ抽出の管理
  Guardium では、ファイルへの事前定義データ抽出はデフォルトで無効になっています。エクスポート抽出を GuardAPI または GUI でスケジュールすることにより、有効にすることができます。

親トピック: レポート

表へのデータマートの抽出

データマート表は、アクセス頻度の高いデータ、レポートで使用できるようにしたいデータが大量にあるシステム、およびスケジュール済みのパージの後に保存したいデータに使用します。

始める前に

前提条件: データマート・ビルダーへのアクセス権 (ユーザー・ロール)。

このタスクについて

表タイプに事前定義されているデータマートはありません。「データマート」アイコン  があるレポートからデータマート表を作成できます。データマートでは、指定された細分度に応じて時間、日、週、または月ごとにデータが要約されます。表データマートを作成すると、Guardium は、割り当てられた名前で照会 - レポートを作成します。この照会 - レポートは、その他の照会 - レポートと同じように変更できます。 照会 - レポートをコピーして、正確なニーズに合わせて変更することができます。 Guardium は、カスタム・ドメインおよび同じ名前のカスタム表も作成します。外部データ相関で、カスタム・ドメインおよびカスタム表の使用について説明しています。

これらのレポートのパラメーターを使用して、関数 (API) を実行し、スクリプトを生成することができます。API 呼び出しおよびレポートの操作を参照してください。

データマートの永続性: オリジナルの照会またはレポートへの変更はデータマートには影響しません。作成時に、元になる分析定義のスナップショットもデータマートと一緒に保存されるためです。

データマートの抽出を初めて実行する場合 (スケジュール実行または「今すぐ 1 回実行」)、時間間隔に基づいて、初始動の日付から現在時刻までのデータが抽出されます。次の期間開始が DM_EXTRACTION_STATE 表に保存されます。次回の実行時に、次の期間開始から開始されるデータが抽出されます。次の期間開始より前にデータマート抽出を実行すると、その期間は抽出によって既に処理されているため、データマート抽出は「空」として表示されます。次の期間開始より前のデータを抽出するには、古いデータをリストアしてから、データマートをもう一度実行してください。

事前に定義されたデータマート抽出ログ・レポートを使用して抽出ログを追跡できます。

アクセス権限があるすべてのデータマートのリストを表示するには、「レポート」 > 「レポート構成ツール」 > 「データマート」にナビゲートします。そのページから、データマートを開いて変更することができます。

変更が作成された後、その「次の経過後にパージ」の日数とスケジュールを変更できます。データマートにアクセスするには、データマートの作成元のレポートを開き、  をクリックして、このレポートに基づくデータマートのリストからデータマートを選択し、この手順の指示に従います。データマートが選択されていて、新規データマートを作成する場合は、「新規」をクリックします。

手順

1. データマートの作成元のレポートにアクセスして、  をクリックします。 「データマート」ダイアログが開きます。
2. データマート名を入力してください。オプションで、説明を入力します。

3. 「結果の抽出先」行で「表」が選択されていることを確認します。
4. オプションで、表名を入力します。指定しない場合は、DM として保存されます。カスタム・ドメインで索引を定義し、GuardAPI を使用して表サイズを確認できるため、直観的な名前を定義すると有用です。
5. 時間の細分度を指定します。これは、結果のデータマート表の細分度です。この細分度は、対応するレポートの実行頻度 (毎時、毎日、毎月) と一致させてください。
6. 「アーカイブ/エクスポート」オプションは、このデータマート表からのデータがデータ・エクスポートおよびデータ・アーカイブに含まれるかどうかを制御します (これらのプロセスがこのユニットで構成されている場合)。このデータマートに対してデータ・エクスポートおよびデータ・アーカイブを有効にするには、「はい」を選択します。
7. パージ日数を設定します。パージ日数は、ビジネス・ケース (例えば、そのデータがレポートで必要になる日数、データのサイズ、使用可能なディスク・スペース) を反映している必要があります。
8. カレンダー・アイコンから「初始動」の時刻を選択します。これは、データマート抽出が初めて実行されるときに抽出されるデータの最初の日付/時刻です。例えば、2018 年 11 月 5 日にデータマートを定義するときに、2018 年 11 月 1 日のデータが必要であるとします。この場合は、「初始動」を 2018 年 11 月 1 日に設定します。
9. 「適用」をクリックし、データマートを保存します。
10. 定期的なデータ抽出を定義するには、「スケジューリング」セクションで「スケジュールの変更」をクリックして、データマート抽出スケジュールを定義します。
    - 開始時刻: 抽出が開始される時刻
    - 再始動: 「1 回だけ実行」のままにします
    - スケジュールの基準: 「曜日」を選択します
    - 「毎日」をクリックします
    - 開始時刻のスケジュール設定: データマートを将来開始する場合を除き、ブランクのままにします。将来開始する場合は、カレンダーを開き、データマートのエクスポートを開始する日付を選択します。
    - 従属ジョブの自動実行。チェック・マークを外したままにします。これはデータマートには関係ありません。
11. このデータマートに対するアクセス権限を持つユーザー・ロールを選択します。
    a. 「ロール」をクリックします。
    b. 「ロール」ダイアログで、「すべてのロール」または個々のロールのいずれかを選択します。
    c. 「適用」をクリックします。ロールが保存され、ダイアログが閉じます。
12. レポート抽出を一時的に停止するには「一時停止」をクリックして、レポート抽出を再開するには「再開」をクリックします。
13. 抽出をリアルタイムで 1 回実行するには、「今すぐ 1 回実行」をクリックします。

親トピック: データマート
関連情報:
GuardAPI データマート関数

ファイルへのデータマートの抽出

ニーズに対応するために、CSV ファイルへのデータ抽出を作成できます。

このタスクについて

抽出された CSV ファイルの名前の形式は、<filename>_<timestamp>.csv です。例えば、SESSION_LIST_20180219060000.csv です。これらのファイルは、抽出の定義時に指定する場所に保管されます。ファイルへの抽出を定義した後、これらの抽出ですべての GRD API コマンドを使用できます。

データマートの全体的な状況を確認するには、データマート抽出ログ・レポートを表示します。

データマート構成コマンドは、CM ごとに 1 回のみ実行します。すべてのコレクターが中央マネージャーからこの情報を受信できるためです。

手順

1. データマートの作成元のレポートにアクセスして、  をクリックします。「データマート」ダイアログが開きます。
2. 「データマート構成」でデータマート名を入力します。オプションで、説明を入力します。
3. 「結果の抽出先」行で「ファイル」を選択します。
4. ファイル名を入力します。
5. オプションで、ファイル・パスを入力します。ブランクのままにする場合、ファイルは、/opt/IBM/Guardium/data/dump/DATAMART/ に保存されます。ファイル・パスは共有ディレクトリーでなければなりません。
6. 時間の細分度を指定します。これは、結果のデータマート表の細分度です。この細分度は、対応するレポートの実行頻度 (毎時、毎日、毎月) と一致させてください。
7. 「アーカイブ/エクスポート」オプションは、このデータマート表からのデータがデータ・エクスポートおよびデータ・アーカイブに含まれるかどうかを制御します (これらのプロセスがこのユニットで構成されている場合)。このデータマートに対してデータ・エクスポートおよびデータ・アーカイブを有効にするには、「はい」を選択します。
8. パージ日数を設定します。パージ日数は、ビジネス・ケース (例えば、そのデータがレポートで必要になる日数、データのサイズ、使用可能なディスク・スペース) を反映している必要があります。
9. カレンダー・アイコンから「初始動」の時刻を選択します。これは、データマート抽出が初めて実行されるときに抽出されるデータの最初の日付/時刻です。例えば、2018 年 11 月 5 日にデータマートを定義するときに、2018 年 11 月 1 日のデータが必要であるとします。この場合は、「初始動」を 2018 年 11 月 1 日に設定します。
10. 「適用」をクリックし、データマートを保存します。
11. 定期的なデータ抽出を定義するには、「スケジューリング」セクションで「スケジュールの変更」をクリックして、データマート抽出スケジュールを定義します。
    - 開始時刻: 抽出が開始される時刻
    - 再始動: 「1 回だけ実行」のままにします
    - スケジュールの基準: 「曜日」を選択します
    - 「毎日」をクリックします
    - 開始時刻のスケジュール設定: データマートを将来開始する場合を除き、ブランクのままにします。将来開始する場合は、カレンダーを開き、データマートのエクスポートを開始する日付を選択します。
    - 従属ジョブの自動実行。チェック・マークを外したままにします。これはデータマートには関係ありません。
12. このデータマートに対するアクセス権限を持つユーザー・ロールを選択します。
    a. 「ロール」をクリックします。

b. 「ロール」ダイアログで、「すべてのロール」または個々のロールのいずれかを選択します。
c. 「適用」をクリックします。ロールが保存され、ダイアログが閉じます。

13. レポート抽出を一時的に停止するには「一時停止」をクリックして、レポート抽出を再開するには「再開」をクリックします。

親トピック: データマート
関連情報:
GuardAPI データマート関数

ファイルへの事前定義データ抽出の管理

Guardium では、ファイルへの事前定義データ抽出はデフォルトで無効になっています。エクスポート抽出を GuardAPI または GUI でスケジュールすることにより、有効にすることができます。

ファイルへの事前定義抽出は、表 1 にリストされています。

デフォルトでは、抽出は毎時実行されます。この設定は変更できます。

外部システムにデータを送信できる一連の事前定義データマートがあります。これらの名前は「エクスポート:」で始まります。

抽出ファイルの接頭部は Global_ID とソース・マシンのホストの短縮名です。

状況: 例えばターゲット・マシンのダウンなど何らかの理由でファイル転送が失敗した場合、次の実行で転送を再試行します。バックログは /var/exportdir ディレクトリーに保持されます。 パージ・プロセスにより、1 日を経過したバックログはクリーンアップされます。

バンドルされたデータマート: 複数の CSV エクスポートのデータマートをまとめてバンドルすることができます。このバンドルには、メインデータマートが設定されます。 バンドルに含まれる各データマートが、それぞれに固有のスケジュールに基づいてデータをプルします。メインデータマートはデータを抽出した後、バンドルに含まれるすべてのデータマートからのデータ・ファイルを同じ tar ファイルに含めて宛先サーバーに送信します。メインデータマートは、最新のスケジュールの抽出時刻を使用して、その他すべてのデータマートが組み込まれるようにする必要があります。

完全な SQL データマートは、「全詳細をロギング」または「マスクされた詳細をロギング」が定義され、インストールされている場合のみ機能します。

異常値データマートは、異常値検出が有効な場合のみ機能します。

データマート・スケジューラーがしばらくの間停止しており、データを遡及的に抽出したくない場合、抽出を再実行するようにスケジュールを変更する前に、「データマート構成」画面で適切な「初始動」を設定してください。

Guardium の内部プロセスに基づき、事前定義抽出に推奨される実行時間があります。これらは、表 2 に示されています。

データ抽出の基本構成 (詳細については、GuardAPI データマート関数を参照)

データマートをアクティブに設定
grdapi datamart_set_active Name=”Export:Exception Log”
データマートを非アクティブに設定
grdapi datamart_set_inactive Name=”Export:Exception Log”
データマート抽出のためのジョブのスケジュール化
grdapi schedule_job jobType=dataMartExtraction cronString=”0 1 0/1 ? * 1,2,3,4,5,6,7” objectName=”Export:Exception Log” startTime=”YYYY-MM-DD HH:MM:SS”
Guardium スケジューラーはアプライアンスに対してローカルであるため、データの抽出元の各アプライアンスで grdapi スケジューリング・コマンドを実行する必要があります。
startTime を使用して、必要に応じて将来の開始時刻を設定します。またデータマートをすぐに開始したい場合は削除できます。
スケジュール済みジョブの削除
grdapi delete_schedule deleteJob="true" jobGroup="DataMartExtractionJobGroup” obname="DataMartExtractionJob_25”
出力 CSV ファイルにヘッダー (列名) を組み込み
grdapi datamart_include_file_header Name=” Export:Exception Log” includeFileHeader=”Yes”
ターゲット・ホストの詳細の設定
grdapi datamart_update_copy_file_info destinationHost=”Machine_Host” destinationPassword=”********” destinationPath=”/where/to/store/” destinationUser=”user” Name=”Export:Exception Log” transferMethod=”SCP” withCOMPLETEfile=false
ターゲット・サーバーに出力を送信

```
grdapi datamart_update_copy_file_info destinationHost="destination server name"
destinationPassword="destination server PW"
destinationPath="destination server" destinationUser="destination server user"
Name="Export:Session Log" transferMethod="SCP"
```

表 1. 事前定義データマート・エクスポート・ジョブ

データマート名/ジョブの記述/オブジェクト名	記述	レポート・タイトル	ユニット・タイプ	データマート ID	ジョブ名
エクスポート: アクセス・ログ	接続情報の詳細と 1 時間ごとのアクティビティーの概要が含まれます。 このログに記録される情報には、OS およびデータベース・ユーザー、成功した SQL と失敗した SQL、クライアント IP とサーバー IP などが含まれます。	エクスポート: アクセス・ログ	コレクター	22	DataMartExtractionJob_22
エクスポート: セッション・ログ	データ・ソースのセッション (ログインからログアウト) に関する詳細が含まれます。このログに記録される情報には、セッション開始とセッション終了のタイム・スタンプ、セッションの OS とデータベース・ユーザー、ソース・プログラムなどが含まれます。	エクスポート: セッション・ログ	コレクター	23	DataMartExtractionJob_23

データマート名/ジョブの記述/オブジェクト名	記述	レポート・タイトル	ユニット・タイプ	データマートID	ジョブ名
エクスポート: 終了したセッション・ログ	セッションは長期間に及ぶ場合があります。抽出は 1 時間ごとに行われます。 このログは、開始時刻 (単位: 時) より後に終了したセッションを送信します。	エクスポート: セッション・ログ	コレクター	24	DataMartExtractionJob_24
エクスポート: 例外ログ	Guardium にキャプチャーされた例外/エラーの詳細を示します。 このログには、例外/エラーの説明、ユーザー名、ソース・アドレス、データベース・プロトコルなどが記録されます。	エクスポート: 例外ログ	すべて	25	DataMartExtractionJob_25
エクスポート: 完全な SQL	実行された SQL の詳細が記録されます。このログに記録される情報には、完全な SQL、影響されるレコード、セッション ID などが含まれます。	エクスポート: 完全な SQL	コレクター	26	DataMartExtractionJob_26
エクスポート: 異常値リスト	異常値が記録されます。このログに記録される情報には、サーバー IP、データベース・ユーザー、異常値タイプ、データベースなどが含まれます。	Analytic 異常値リスト	すべて	27	DataMartExtractionJob_27
エクスポート: 時間単位の異常値概要	1 時間ごとの異常値の概要が含まれます。このログに記録される情報には、サーバー IP、データベース・ユーザー、データベースなどが含まれます。	Analytic 異常値サマリー	任意	28	DataMartExtractionJob_28
エクスポート: グループ・メンバー	すべてのグループ・メンバーのログが含まれます。このログには、グループ・タイプ、グループの記述、グループ・メンバー、およびタプル・フラグが含まれます。	エクスポート: グループ・メンバー	すべて	29	DataMartExtractionJob_29
エクスポート: 抽出ログ	名前が「エクスポート:」で始まるすべてのエクスポート・ファイルまたはコピー・ファイルに関連するデータのログが含まれます。	ユーザー定義抽出ログ	すべて	31	DataMartExtractionJob_31
エクスポート: ポリシー違反	データベース・ユーザー、ソース・プログラム、アクセス・ルールの記述、および SQL 文字列全体などログに記録された違反に関する詳細が含まれます。	エクスポート: ポリシー違反	コレクター	32	DataMartExtractionJob_32
エクスポート: バッファー使用状況モニター	スニファーのバッファー使用状況の統計の詳細を表示します	バッファー使用状況モニター	すべて	33	DataMartExtractionJob_33
エクスポート: 脆弱性診断結果		セキュリティー・アセスメント・エクスポート	すべて	34	DataMartExtractionJob_34
エクスポート: ポリシー違反 - 詳細	「エクスポート抽出ログ」と同じですが、オブジェクト/動詞タプルが含まれます。いずれか一方ログだけを使用することを推奨します。	エクスポート: ポリシー違反	コレクター	38	DataMartExtractionJob_38
エクスポート: アクセス・ログ - 詳細	アクセス・ログと同じですが、アプリケーション・イベント・エンティティーのフィールド (イベント・ユーザー名、イベント・タイプ、イベント値 (文字列)、イベント値 (数値)、イベントの日付) も含まれます。「アクセス・ログ」または「アクセス・ログ – 詳細」のいずれか一方を使用し、両方を同時に使用しないことを推奨します。	エクスポート: アクセス・ログ	コレクター	39	DataMartExtractionJob_39
エクスポート: ディスカバーされたインスタンス	データベース・インスタンスをディスカバーする S-TAP ディスカバリー・アプリケーションの結果を提供します。	ディスカバーされたインスタンス	すべて	40	DataMartExtractionJob_40
エクスポート: ディスカバーされたデータベース		ディスカバーされたデータベース	すべて	41	DataMartExtractionJob_41
エクスポート: 分類結果		分類結果	すべて	42	DataMartExtractionJob_42

データマート名/ジョブの記述/オブジェクト名	記述	レポート・タイトル	ユニット・タイプ	データマートID	ジョブ名
エクスポート:データ・ソース		データ・ソース	中央マネージャー、スタンドアロン	43	DataMartExtractionJob_43
エクスポート:STAP 状況		S-TAP 状況モニター	コレクター	44	DataMartExtractionJob_44
エクスポート:インストール済みのパッチ		インストール済みのパッチ	すべて	45	DataMartExtractionJob_45
エクスポート:システム情報		インストール済みのパッチ	すべて	46	DataMartExtractionJob_46
エクスポート:ユーザー - ロール		ユーザー - ロール	中央マネージャー、スタンドアロン	47	DataMartExtractionJob_47
エクスポート:分類プロセス・ログ		分類プロセス・ログ	すべて	48	DataMartExtractionJob_48
エクスポート:異常値リスト - 拡張		Analytic 異常値リスト - 拡張	すべて	49	DataMartExtractionJob_49
エクスポート:時間単位の異常値概要 - 拡張		Analytic 日付別異常値サマリー - 拡張	すべて	50	DataMartExtractionJob_50

表 2. 事前定義データマート・エクスポート・ジョブのデフォルトの cronString

ジョブの記述	推奨 cronString	毎時:
エクスポート:アクセス・ログ	0 40 0/1 ? * 1,2,3,4,5,6,7	00:40
エクスポート:セッション・ログ	0 45 0/1 ? * 1,2,3,4,5,6,7	00:45
エクスポート:終了したセッション・ログ	0 46 0/1 ? * 1,2,3,4,5,6,7	00:46
エクスポート: 例外ログ	0 25 0/1 ? * 1,2,3,4,5,6,7	00:25
エクスポート: 完全な SQL	0 30 0/1 ? * 1,2,3,4,5,6,7	00:30
エクスポート:異常値リスト	0 10 0/1 ? * 1,2,3,4,5,6,7	00:10
エクスポート:時間単位の異常値概要	0 10 0/1 ? * 1,2,3,4,5,6,7	00:10
エクスポート:抽出ログ	0 50 0/1 ? * 1,2,3,4,5,6,7	00:50
エクスポート: グループ・メンバー	0 15 0/1 ? * 1,2,3,4,5,6,7	00:15
エクスポート:ポリシー違反	0 5 0/1 ? * 1,2,3,4,5,6,7	00:05
エクスポート: バッファー使用状況モニター	0 12 0/1 ? * 1,2,3,4,5,6,7	00:12
エクスポート:脆弱性診断結果	0 0 2 ? * 1,2,3,4,5,6,7	毎日午前 2 時
エクスポート:ポリシー違反 - 詳細	0 5 0/1 ? * 1,2,3,4,5,6,7	00:05
エクスポート:アクセス・ログ - 詳細	0 40 0/1 ? * 1,2,3,4,5,6,7	00:40
エクスポート:ディスカバーされたインスタンス	0 20 0/1 ? * 1,2,3,4,5,6,7	00:20
エクスポート:ディスカバーされたデータベース	0 20 0/1 ? * 1,2,3,4,5,6,7	00:20
エクスポート:分類結果	0 20 0/1 ? * 1,2,3,4,5,6,7	00:20
エクスポート:データ・ソース	0 0 7 ? * 1,2,3,4,5,6,7	毎日午前 7 時

ジョブの記述	推奨 cronString	毎時:
エクスポート:STAP 状況	0 0/5 0/1 ? * 1,2,3,4,5,6,7	5 分ごと
エクスポート:インストール済みのパッチ	0 0 5 ? * 1,2,3,4,5,6,7	毎日午前 5 時
エクスポート:システム情報	0 0 5 ? * 1,2,3,4,5,6,7	毎日午前 5 時
エクスポート:ユーザー - ロール	0 5 0/1 ? * 1,2,3,4,5,6,7	00:05
エクスポート:分類プロセス・ログ	0 25 0/1 ? * 1,2,3,4,5,6,7	00:25
エクスポート:異常値リスト - 拡張	0 10 0/1 ? * 1,2,3,4,5,6,7	00:10
エクスポート:時間単位の異常値概要 - 拡張	0 10 0/1 ? * 1,2,3,4,5,6,7	00:10

親トピック: データマート
関連情報:
GuardAPI データマート関数

配布レポート・ビルダー

この中央マネージャー機能により、特定の中央マネージャーに関連付けられているすべてまたは一部の Guardium 管理対象ユニットから、データを自動的に収集することができます。配布レポートは、概要ビューの提供、データ・ソース間のデータの関連付け、およびデータのビューの要約を行うように設計されています。 コレクター間での行レベル・データ収集については、引き続きアグリゲーターを使用します。

複雑なエンタープライズ環境では、特定のレポートで必要になるデータが存在する管理対象ユニットをユーザーが必ずしも正確に知っているわけではない場合に問題が発生する可能性がありますが、この機能により、こうした問題を軽減することができます。この問題は、ロード・バランシングなどの構成オプションに基づく Guardium コレクターとデータベース間のリンクの時間経過に伴う変化によって発生する場合があります。この問題は、アグリゲーターとコレクターの期間やデータ保存ポリシーなどの考慮事項によってさらに複雑なものになります。

配布レポートは、簡単に作成できます。「配布レポート」画面で配布レポートを定義して任意のペインに追加するだけで、すぐに使用できるようになります。

配布レポートは、中央マネージャー上のデータマートをオプションで使用して、スケジュールされた統合データの収集を経時的に有効にします。要するに、配布レポートのデータはフラット・テーブルとして保管されるため、必要なレポートを作成する際に、複雑な結合処理を行う必要はありません。これにより、これらのエンタープライズ・レポートの応答時間が大幅に短縮されます。

配布レポートのデータは、コレクターおよびアグリゲーター、さらには中央マネージャーからも収集することができます。レポートのデフォルトの配布バージョンには、対象データを管理するユニットのホスト名が含まれます。

配布レポートには 2 つのタイプがあります。

- 即時レポートには、「データ収集元」リストで選択された各ユニットからの限られた量のデータが示されます。これはオンデマンドで行われます。
- スケジュールされたレポートは、スケジュールおよび時間細分度によって定義されているようにバックグラウンドで実行され、1 次と 2 次 (定義されている場合) のターゲット上の表にデータを保存します。

以下に、事前定義されている配布レポートを示します。

- エンタープライズ S-TAP 検査
- 統合/アーカイブ・ログ
- 失敗したユーザー・ログイン試行
- スケジュールされたジョブの例外

前提条件 – 一元管理画面で管理対象ユニットのグループを作成します。

1. 配布レポートを作成します。
2. 収集したデータをレビューします。
3. 収集したデータに関する追加の要約レポートを作成します。

配布レポートの実行: 即時実行またはスケジュールによる実行
配布レポートを定義する際、レポートを即時に実行するか、または、レポートをバックグラウンドで実行するようにスケジュールして、中央マネージャーに結果を収集します。

- 即時: このモードでは、オンデマンド (GUI を使用して実行) でデータが収集され、関連する管理対象ユニットから結果が収集されると共に、結果が表示されます。配布レポートには、データがまだ転送中であるのか、あるいは特定の管理対象ユニットからすべてのデータを受信したのかを示す状況インディケーターが含まれます。このモードの場合、データは中央マネージャーには保存されません。レポートを閉じると同時に、データが消去されます。
- スケジュール済み: このモードでは、すぐに応答を返すことができるよう、事前にデータが収集されます。スケジューラーで指定した時間間隔に従い、指定された管理対象ユニットの関連するすべての統合データが中央マネージャー・マシン上の指定されたデータマート表に送信され、この表に対するデフォルトのレポートが作成されます。この表には、その独自のドメインとエンティティーも含まれており、クエリー・ビルダーを使用して追加のクエリーとレポートを作成することができます。これらのレポートを監査プロセスに追加して、プロセスを定期的に実行することができます。また、プロセスの結果を、レビューまたはサインオフのためにロール、ユーザー、ユーザー・グループに割り当てることができます。

配布レポートを計画する場合の考慮事項

- 32 ビットの中央マネージャーと 64 ビットの管理対象ユニットが存在する混合環境の場合、64 ビット・システムの情報は配布レポートには表示されません。この場合に情報を表示するには、中央マネージャーを 64 ビットにアップグレードする必要があります。
- 中央マネージャーに送信されるデータを調整する必要があるため、すべての管理対象ユニットのクロック時刻を、目的の管理対象ユニットが存在するタイム・ゾーンの現在時刻に設定しておくことが非常に重要です。中央マネージャーと管理対象ユニットの時間が10 分違っているだけでも、配布レポートのパフォーマンスと信頼性に影響します。
- スケジュール済み配布レポートの定義はエクスポートおよびインポートできますが、即時配布レポートの定義はエクスポートすることもインポートすることもできません。エクスポートおよびインポートされた定義には、スケジュール自体は含まれません。バックアップ用の中央マネージャーやテスト用の中央マ

ネージャーなど、他のシステム上で再作成する必要がある場合は、定義とスケジューリングのレコードを保持することをお勧めします。システム・バックアップには、配布レポートの構成情報が含まれます。

- アグリゲーターとコレクターの両方からレポート・データを収集するように指定した場合、デフォルトの配布レポートに重複データが含まれることがあります (ただし、Guardium のホスト名は異なります)。 この場合、配布レポートの構成に対して、コレクターとアグリゲーターのいずれかのみを指定することを特にお勧めします。
- 配布レポートは、配布レポート以外の既存のレポートに基づいています。 スケジュール・モードで配布レポートを定義する際に、元の照会にランタイム・パラメーターが指定されている場合は、それらのパラメーターの値 (または、ワイルドカードの「%」) を指定するための画面が表示されます。
- スケジュール済み配布レポートごとにターゲットの Guardium システムを選択できます。(デフォルトでは、ターゲットは中央マネージャーです。選択可能なターゲット・システムのリストは、GRDAPI コマンド grdapi set_distributed_report_target target_host_name=[unit host name] によって設定されます。) ターゲット・システムでは、配布レポートの前にはなかったデータがデータベース上に配置されます。パージ、アップグレード、バックアップについて、運用方法の変更を前もって計画してください。

編集および更新

配布レポートの場合、ベース・レポートを編集して更新し、更新後のレポート構造に基づいて配布レポートを更新します。

ベース・レポートの列を変更するか、ベース・レポートの Where 節を追加または削除してから、レポートを保存して再生成した場合、この更新後のレポートに基づいて配布レポートを更新するには、既存の配布レポートで「レポート変更の保存」をクリックします。これだけで、変更が適用されます。

既存のレポート・パラメーターを更新する場合は、最初に「レポート変更の適用」をクリックし、パラメーターの値を更新してから「レポート変更の保存」をクリックする必要があります。これで、更新が適用されます。

時刻の詳細

レポートを実行する際に、レポート・カスタマイザーを使用して、照会の絶対時間枠 (from 3-31-2014 8:00am to 3-31-2014 11:00am) または相対時間枠 (NOW -3 HOUR) を指定することができます。

絶対時間を指定した場合、各 Guardium システムは現地時間に基づいて稼働します。例えば、配布レポートが東部標準時 (EST) の Guardium システムと太平洋標準時 (PST) の Guardium システムからデータを収集する場合、各システムは現地時間に基づいて照会を実行します。この例は (午前中のピーク時間帯、深夜の時間帯、特定の絶対時間を確認する場合に便利です)、ニューヨークに存在するシステムは東部標準時の 08:00 から 11:00 までの結果を収集し、カリフォルニアに存在するシステムは、太平洋標準時の 08:00 から 11:00 までの結果を収集します。

相対時間を指定した場合、各システムは、システムの現在時刻に従って「NOW -N」を実行します。これは、リアルタイム・レポートの場合に重要になります。リアルタイム・レポートおよびリアルタイムに近いレポートの場合、絶対時間を使用することはできません。リアルタイムでモニターを行う場合は、「即時」モードを使用してください。

配布レポートの状況の確認

各配布レポートには、結果の取り込みに成功したマシンと失敗したマシンを示す状況レポートが付属しています。GUI で配布レポートにナビゲートすると、状況レポートにアクセスするためのリンクが強調表示されます。

スケジュールされたレポートの場合、状況レポートの任意の行をクリックすると、特定のユニット上でレポートを再実行するための API を実行することができます。

スケジュール・モードでの配布レポートの特定の実行でエラーが発生した場合、以下の手順により、状況レポートからレポートを再実行することができます。

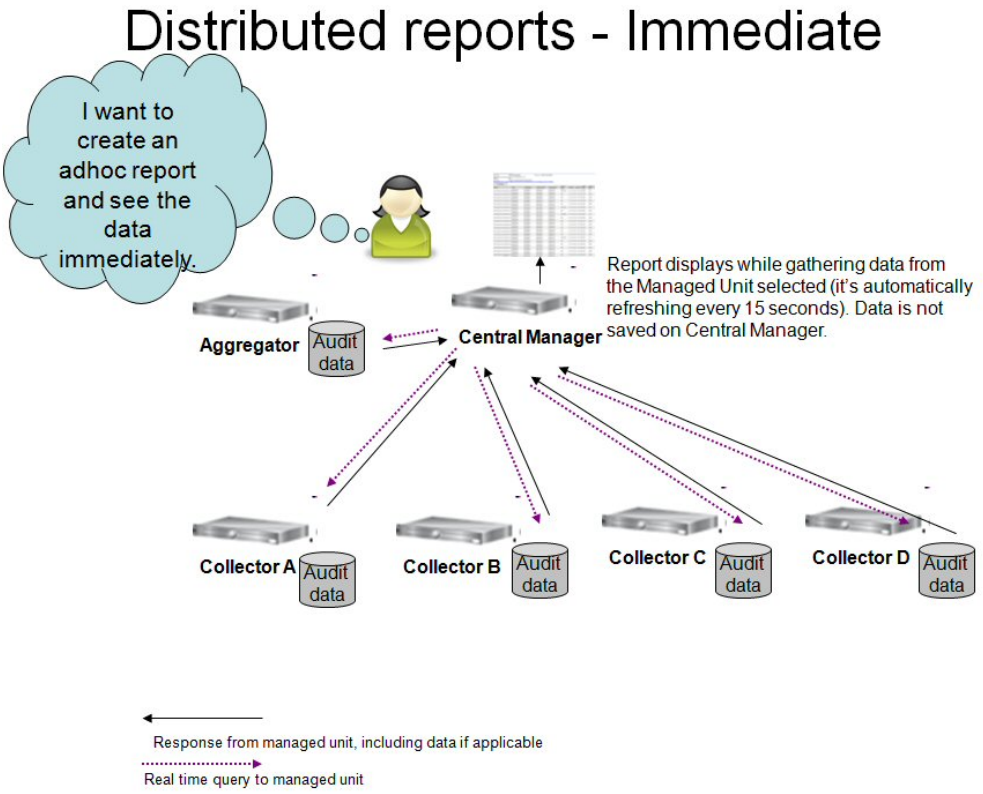
1. 状況レポートのいずれかの行をダブルクリックして「呼び出し」メニューを表示します。次に、「呼び出し」をクリックします。
2. 選択項目の「rerun_distributed_report」をクリックします。
3. これにより、特定の実行を再実行するためのポップアップ画面が開きます。レポート内の任意の行を開くことができますが、再実行できるのは「エラー」状態の行だけです。

配布レポートを再実行するための GuardAPI

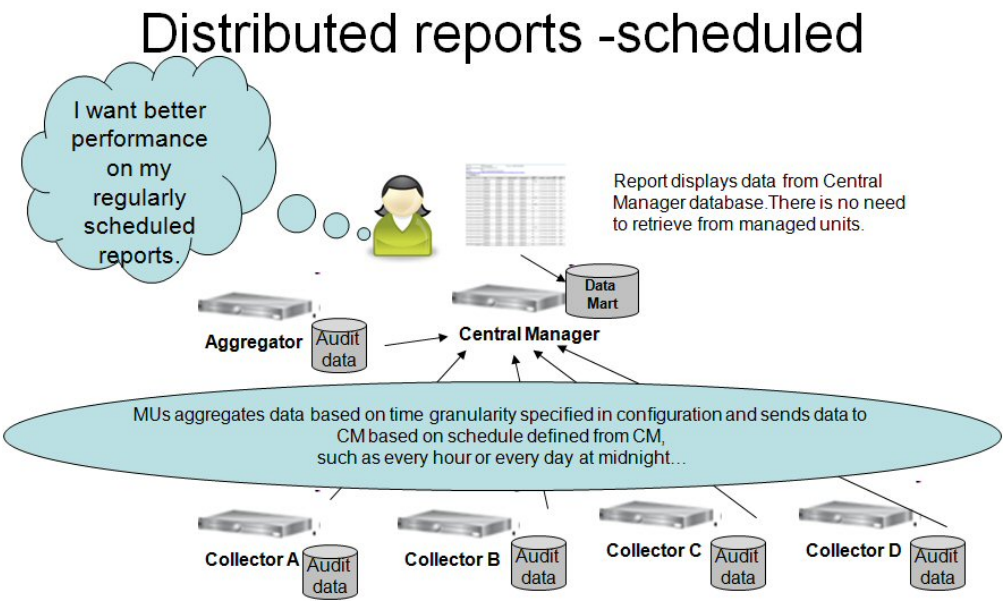
状況レポートを呼び出すための、GUI に記載された再試行コマンドには、GuardAPI コマンドを使用してアクセスすることもできます。

構文

grdapi rerun_distributed_report



この図は、即時配布レポートを実行するためのプロセスを示しています。



この図は、配布レポートをスケジュールするためのプロセスを示しています。

配布レポートの拡張: ターゲット・システムを任意の Guardium システムに設定する

配布レポートは、指定された Guardium システムに照会要求を配布します。システムはターゲット・システム内にデータを収集して結果を統合し、その統合された結果のビューを提供します。 この結果は、追加の照会の定義のためにクエリー・ビルダーで使用することができます。

配布レポート機能を使用して、ターゲット・システムを任意の Guardium システムに設定できるようになりました。以前のバージョンでは、ターゲット・システムの設定は許可されておらず、常に中央マネージャー (CM) に設定されていました。

要件の理由

多くの場合、配布レポートに関係なく CM に対する負荷が高くなります。CM はアグリゲーターとして使用されることもありますが、その場合は CM の負荷がさらに高くなります。

このような場合、ユーザーがターゲット・システムを設定できるようにした方がはるかに効率的です。

ソリューション

- ターゲット・システムは、配布レポートごとに設定することができます。CLI コマンドを使用して、オプションのターゲット・システムを設定することができます。CLI を使用して設定されたリストは、配布レポート・ビルダーの GUI に表示されます。
- 重要: この変更は、配布レポートのスケジュール・モードにのみ影響します。「即時」モードは、この変更には含まれていません。 そのため、随時の配布レポートの結果ビューアーには、CM 経由でのみアクセスすることができます。
- これまでと同様に、CM 経由でのみ配布レポートの定義を編集することができます。

CLI コマンド (CM 経由でのみ使用可能)

1. システムをターゲット・システムとして設定:

grdapi set_distributed_report_target target_host_name=[unit host name]

2. システムのターゲット・システムとしての設定をキャンセル:

grdapi cancel_distributed_report_target target_host_name=[unit host name]

このユニットがターゲット・システムとして設定されている配布レポートがまだ存在する場合は、エラーとともにそれらのレポートのリストが返されます。

3. ターゲット・システムのリストを取得:

grdapi get_distributed_report_target_info

その他の CLI コマンド

スケジュールが設定された配布レポートの場合、ユニットごとの行の最大数の値を保管または表示します。

show scheduled_distributed

store scheduled_distributed

store コマンドには、1 つのパラメーター maximum_rows_per_unit があります。 このパラメーターの値が 15,000 より大きいか、0 と等しい (制限なし) 場合、ユーザーに次の警告メッセージが表示されます。

コレクターの数によっては、ユニット当たりの最大行数を高い値に設定すると、パフォーマンスに悪影響が及ぶ可能性があります。(Depending on number of collectors, setting maximum number of rows per unit to a high value might have negative impact on performance.)

- 配布レポートの作成
  中央マネージャーで配布レポートを作成および変更する方法を説明します。

親トピック: レポート

配布レポートの作成

中央マネージャーで配布レポートを作成および変更する方法を説明します。

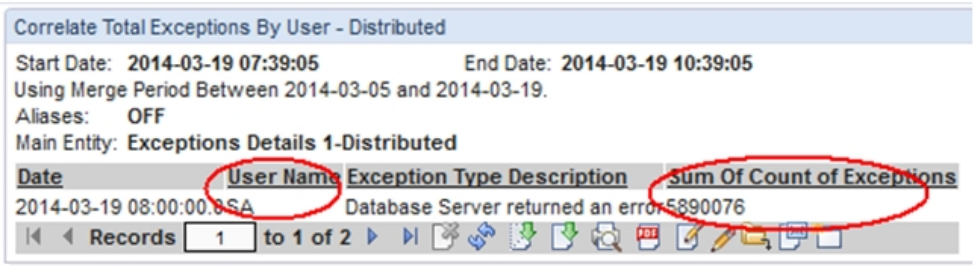
始める前に

管理対象ユニットのグループからデータを収集する場合は、そのグループが定義されていることを確認します。

このタスクについて

この例では、特定のコレクターで記録された例外 (例えば SQL エラー) について、より広い範囲の視野と相関性についての洞察を得るための方法を示します。

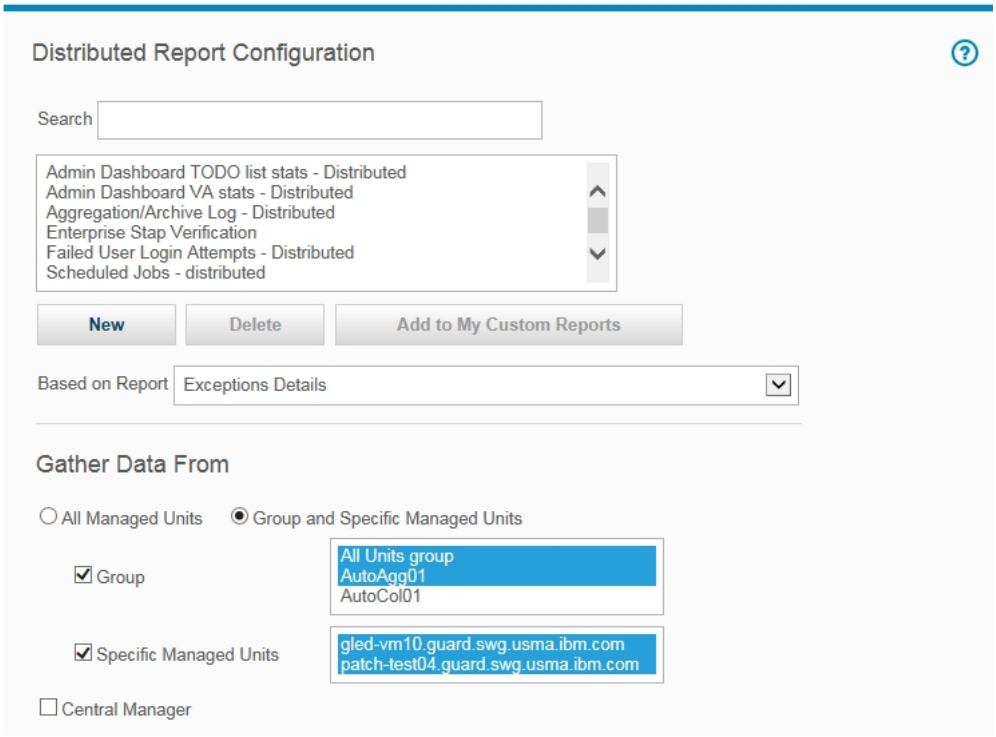
次の画面キャプチャーは、配布レポート「ユーザー別例外の合計の相関 (配布) (Correlate Total Exceptions By User (Distributed))」の例を示しています。 このビューには、この配布レポートで選択した Guardium 管理対象ユニットに関連付けられているすべてのデータベースからのユーザーごとの例外の合計が表示されます。同様に、システム全体における失敗したログイン試行の合計や、ソース・プログラム当たりの例外の合計を表示できます。



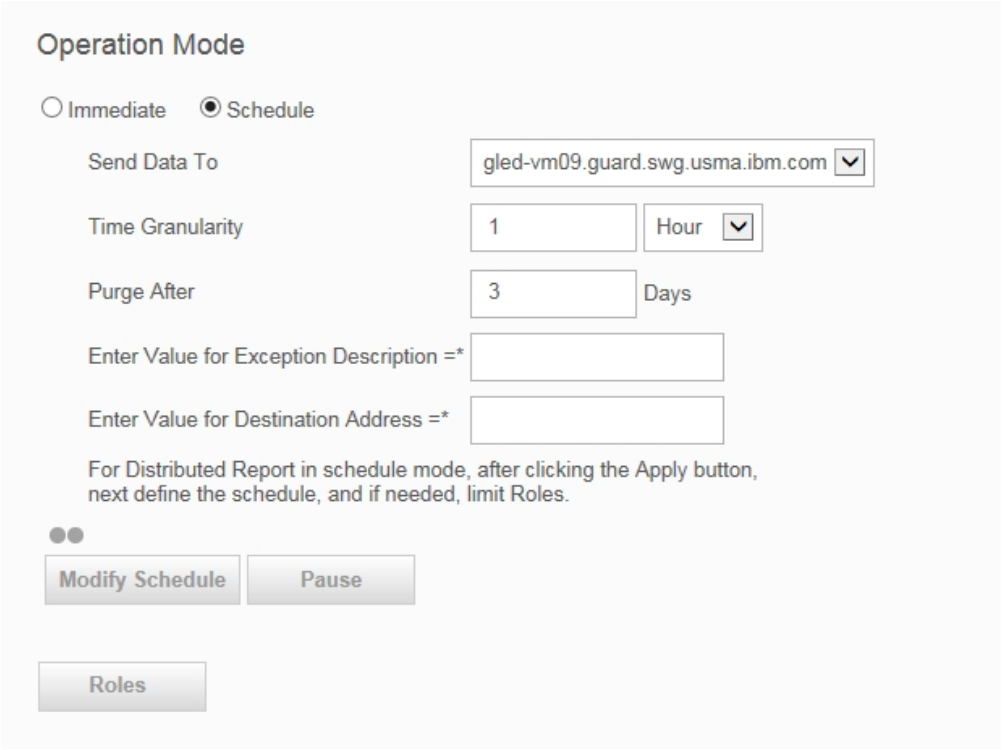
この具体例では、レポートのデータは 1 時間ごとに収集されます。最初の結果を得るために 1 時間以上待つ必要はありません。
注:「配布レポートの状況 - 詳細を表示するには、ここをクリックします。」の行は、データ収集の状況を示します。管理対象ユニットにデータがない場合、この行は赤色で示されます。この行をクリックすると、ユニットごとの 1 時間当たりの状況に関する詳細レポートにナビゲートします。

手順

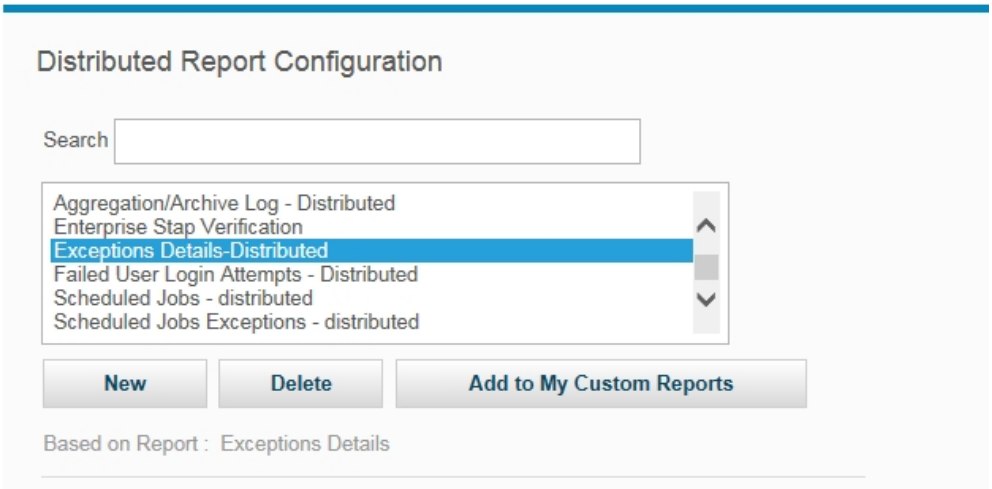
1. 「レポート」 > 「レポート構成ツール」 > 「配布レポート・ビルダー」をクリックします。
2. 「新規」をクリックします。
3. 「レポートに基づく」ドロップダウン・リストから、この配布レポートのベースとするレポートを選択します。 この例では、「例外の詳細」を選択します。



4. ビルダーの「データ収集元」セクションで、「すべての管理対象ユニット」(中央マネージャーで管理されているユニット) を選択するか、特定の「グループと特定の管理対象ユニット」を指定します。これらのデータが配布レポートに組み込まれます。中央マネージャーがアグリゲーターを兼ねている場合は、対象として含める必要が生じることがあります。この例では、「グループ」リストから 2 つのグループを選択し、「管理対象ユニット」リストから数個の管理対象ユニットを選択しますが、「中央マネージャー」のチェック・マークは外したままにします。
5. 「動作モード」で、レポートのタイプとして即時またはスケジュールを選択します。即時モードは、主にオンライン/リアルタイム・モニター用であり、最近の失敗したログイン試行、最近の過度の例外、またはリアルタイム・アラートなどの表示に使用します。スケジュール・モードは、定義されたスケジュールに基づいて定期的に実行される継続的データ収集です。この例では、1 時間ごとに例外を要約します。「例外の記述」および「宛先アドレス」の値を入力するための要件があります。



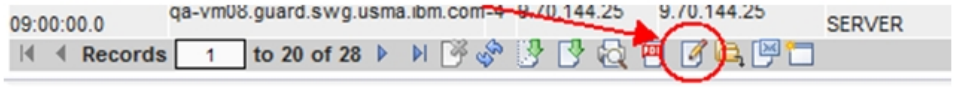
6. 「適用」をクリックして、配布レポートを作成します。
7. 「スケジュールの変更」をクリックして、スケジュールを定義します (これは、プロセスをアクティブ化するためには必須です)。標準の「スケジュール定義」ウィンドウが開きます。必要な詳細をすべて入力します。 スケジューリングについて詳しくは、スケジューリングを参照してください。
8. 適用すると、新しい配布レポートが追加され、リスト・ボックス内で強調表示されます。



9. このレポートに対してロールを制限する場合は、「ロール」をクリックして、「セキュリティー・ロールの割り当て」ウィンドウを開き、ロールを選択して、「適用」をクリックします。

タスクの結果

データは、指定されたすべての管理対象ユニットから収集され、指定された新規エンティティー (表) に保管されます。これで、このエンティティーは、照会 - レポート・ビルダーを通じて、追加の照会の作成に使用できます。追加の照会を作成するためのオプションは、配布レポートの結果画面でも使用できます。「このレポートの照会を編集」をクリックします。



このデフォルト・レポートは変更できません。「コピー」をクリックし、名前を付けて、すべての属性を削除し、「日付」、「ユーザー名」、「例外タイプの記述」、および「例外の合計数 (Sum Of Count Of Exceptions)」をそのままにしてください。

親トピック: 配布レポート・ビルダー

API 呼び出しおよびレポートの操作

- レポートから API 呼び出しを生成する方法
  レポート内の単一行を使用して、またはレポート全体に基づいて、レポートから Guard API 呼び出しを生成します。

- API 呼び出しで定数を使用する方法
  API 関数の呼び出し時に使用する新しいエンティティー属性を作成します。
- カスタム・レポートから API 呼び出しを使用する方法
  API 関数をレポートにリンクし、レポートの各フィールドを API 関数のパラメーターにマップします。

親トピック: レポート

レポートから API 呼び出しを生成する方法

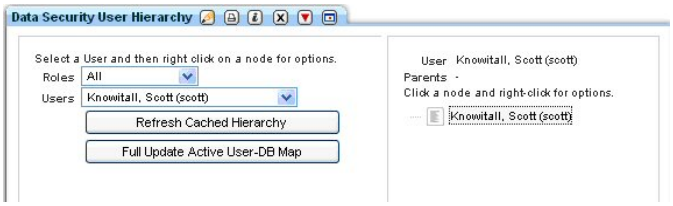
レポート内の単一行を使用して、またはレポート全体に基づいて、レポートから Guard API 呼び出しを生成します。

付加価値: レポートに表示されるシステムの既存データを API 呼び出しのパラメーターとして使用することにより、システム・レベルのコマンドを実行したり長い API 呼び出しを入力したりしなくても、GUI を使用して素早く簡単に API 呼び出しを生成しデータを設定できます。その結果、データ・ソースの作成や検査エンジンの定義、ユーザー階層の保守、あるいは S-TAP などの Guardium フィーチャーの保守などの操作を素早く実行できます。

単一行の API 呼び出し

このシナリオでは、「ユーザー階層によるデータ・セキュリティー」にデータを設定するための API ファンクション・コールを生成します。

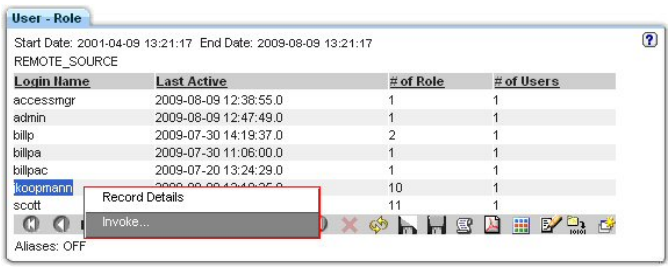
1. 最初に、ユーザー **scott** の現在の**「ユーザー階層によるデータ・セキュリティー」**



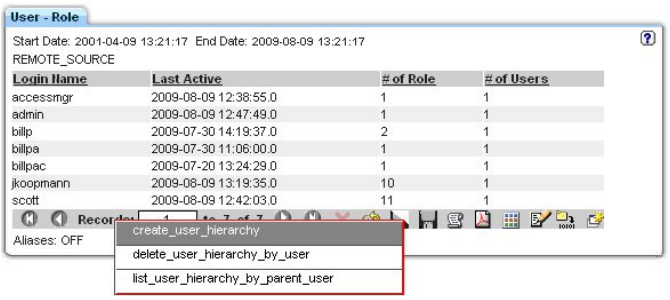
」を表示します。

2. API 関数を呼び出すには、目的の API 関数が現在リンクされているレポートを見つける必要があります。 ユーザー階層の作成はユーザーに関連しているので、ユーザー・レポートを選択することが、よい結果をもたらします。 このシナリオでは、**「ユーザー - ロール」**レポートを選択しました。

3. **行をダブルクリック**してドリルダウンすると、 **「呼び出し...」**オプションが表示されます。

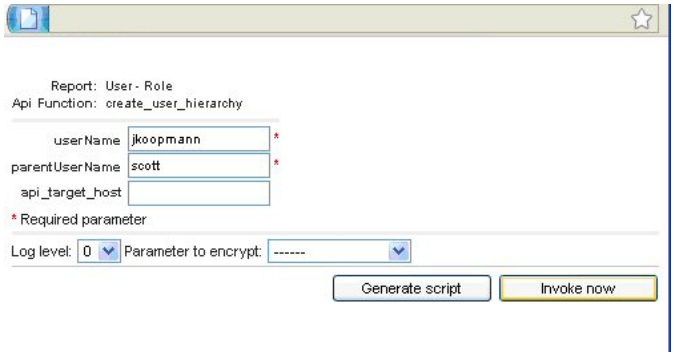


4. **「呼び出し...」**オプションをクリックして、 このレポートにマップされる API 関数のリストを表示します。



5. 呼び出す **API** をクリックします。 レポートと呼び出される API 関数の「API 呼び出しフォーム」が表示されます。

6. 選択した API 呼び出しの必須パラメーターを入力し、必須以外のパラメーターがあれば入力します。パラメーターの多くはレポートに基づいてあらかじめ入力されていますが、変更して固有の API 呼び出しを作成することもできます。必須のパラメーターと必須以外のパラメーターの入力については、 「GuardAPI リファレンス・ガイド」の個々の API 関数呼び出しを参照してください。

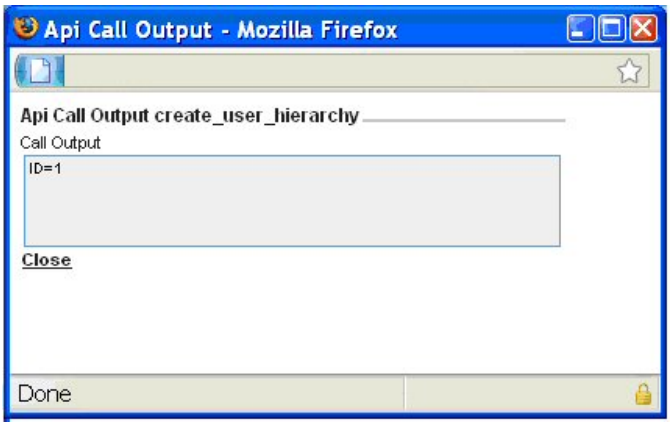


7. ドロップダウン・リストを使用して**ログ・レベル**を選択します。ログ・レベルの意味は、次のとおりです。0:「戻りコード」で定義されている ID=*identifier* と ERR=*error_code* を返します。1: 追加情報を画面に表示します。2: 情報を Guardium アプリケーション・デバッグ・ログに書き込みます。3: 1 と 2 の両方の処理を行います。

8. ドロップダウン・リストを使用して**「暗号化するパラメーター」**を選択します。
注: パラメーター暗号化は共有パスワードを設定することによって有効になり、スクリプト生成を介して API 関数を呼び出す場合のみ該当します。

9. **「今すぐ呼び出し」**または**「スクリプトを生成」**を選択します。

   a. **「今すぐ呼び出し」**を選択すると、ただちに API 呼び出しが実行され、「API 呼び出し出力」画面に API 呼び出しの状況が表示されます。



   b. **「スクリプトを生成」**を選択した場合は、生成されたスクリプトを任意のエディターで開きます。後で編集して実行する場合は、ディスクに保存することもできます。スクリプトに空のパラメーター値 (「< >」で表記) が含まれている場合は、必要な値に置き換えてください。
注: API 呼び出しでは、スクリプトの空のパラメーターは無視されるため、空のままにしておくこともできます。

      スクリプトの例

      # A template script for invoking guardAPI function create_user_hierarchy :

      # Usage: ssh cli@a1.corp.com<create_user_hierarchy_api_call.txt

      # replace any < > with the required value

      #

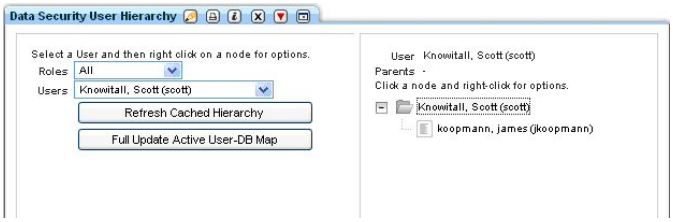
      grdapi create_user_hierarchy userName=jkoopmann parentUserName=scott

   c. CLI 関数呼び出しの実行

      呼び出しの例

      $ ssh cli@a1.corp.com<create_user_hierarchy_api_call.txt

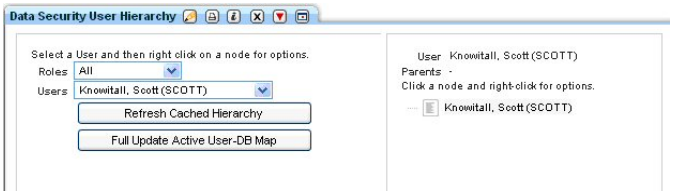
10. 検証します。このシナリオの場合は、「ユーザー階層によるデータ・セキュリティー」の再表示になります。



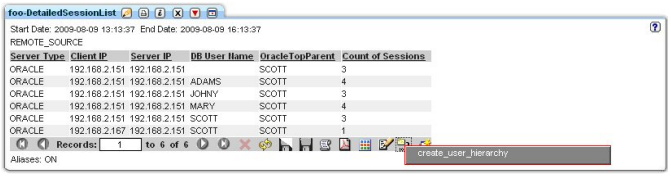
複数行の API 呼び出し

このシナリオでは、レポート・フィールドにパラメーターがマップされたカスタム・レポートを使用します。 追加情報については、このセクションで後述する追加シナリオを参照してください。

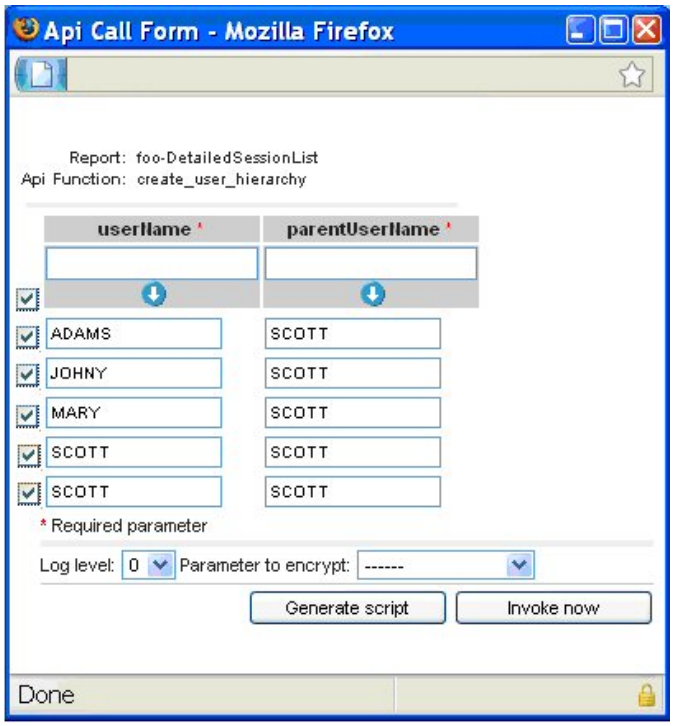
1. 始めに、ユーザー **scott** の現在の**「ユーザー階層によるデータ・セキュリティー」**を見てみましょう。



2. **「呼び出し...」**アイコンをクリックして、このレポートにマップされる API のリストを表示します。



3. 呼び出す **API** をクリックします。 レポートと呼び出される API 関数の「API 呼び出しフォーム」が表示されます。レポートから複数行を対象に API 呼び出しを起動すると「API 呼び出し形式」が生成されて表示され、画面に表示されたすべてのレコードを編集できるようになります。表示されるレコードはフェッチ・サイズによって異なり、最大で 20 です。



4. **チェック・ボックス**を使用して、API 呼び出しのターゲットになる行を選択/選択解除します。

5. 選択した API 呼び出しの**必須パラメーター**を入力し、 **必須以外のパラメーター**があれば入力します。パラメーターの多くはレポートに基づいてあらかじめ入力されていますが、変更して固有の API 呼び出しを作成することもできます。必須のパラメーターと必須以外のパラメーターの入力については、 「GuardAPI リファレンス・ガイド」の個々の API 関数呼び出しを参照してください。さらに、API 用パラメーターのセットを使用して、パラメーターの値を入力します。下矢印をクリックすると、すべてのレコードについてそのパラメーターのデータが設定されます。

6. ドロップダウン・リストを使用して**ログ・レベル**を選択します。ログ・レベルの意味は、次のとおりです。0: 「戻りコード」で定義されている ID=*identifier* と ERR=*error_code* を返します。1: 追加情報を画面に表示します。2: 情報を Guardium アプリケーション・デバッグ・ログに書き込みます。3: 1 と2 の両方の処理を行います。

7. ドロップダウン・リストを使用して**「暗号化するパラメーター」**を選択します。
注: パラメーター暗号化は共有パスワードを設定することによって有効になり、スクリプト生成を介して API 関数を呼び出す場合のみ該当します。

8. **「今すぐ呼び出し」**または**「スクリプトを生成」**を選択します。

   a. **「今すぐ呼び出し」**を選択すると、ただちに API 呼び出しが実行され、「API 呼び出し出力」画面に API 呼び出しの状況が表示されます。階層に循環関係を持たせることができないため、このシナリオの最後の 2 つの API 呼び出しは失敗します。

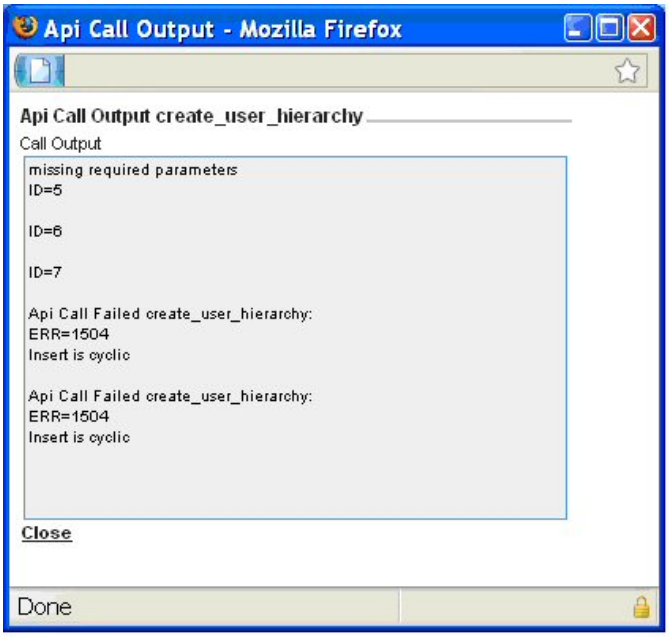
   b. **「スクリプトを生成」**を選択した場合は、生成されたスクリプトを任意のエディターで開きます。後で編集して実行する場合は、ディスクに保存することもできます。スクリプトに空のパラメーター値 (「< >」で表記) が含まれている場合は、必要な値に置き換えることができます。このシナリオの場合、スクリプトの最後の 2 行のために循環エラーが生じることがわかっているので、この 2 行をすぐに削除できます。
   注: 空のパラメーターは、API 呼び出しで無視されるため、スクリプトで有効です。
   スクリプトの例

```
# A template script for invoking guardAPI function create_user_hierarchy :
# Usage: ssh cli@a1.corp.com<create_user_hierarchy_api_call.txt
# replace any < > with the required value
```

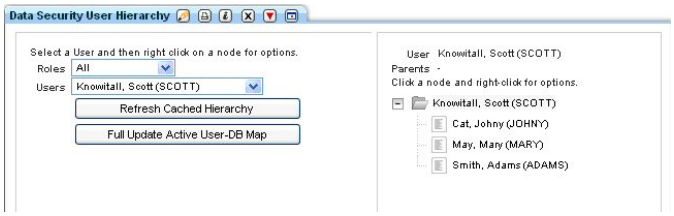
```
#
grdapi create_user_hierarchy userName=ADAMS parentUserName=SCOTT
grdapi create_user_hierarchy userName=JOHNY parentUserName=SCOTT
grdapi create_user_hierarchy userName=MARY  parentUserName=SCOTT
grdapi create_user_hierarchy userName=SCOTT parentUserName=SCOTT
grdapi create_user_hierarchy userName=SCOTT parentUserName=SCOTT
```

次に、以下のような CLI 関数呼び出しを実行します。

```
$ ssh cli@a1.corp.com<create_user_hierarchy_api_call.txt
```



9. 検証します。このシナリオの場合は、「ユーザー階層によるデータ・セキュリティー」の再表示になります。



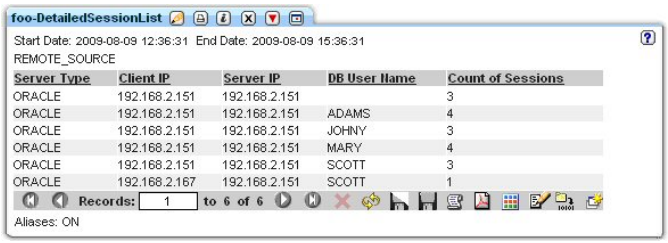
親トピック: API 呼び出しおよびレポートの操作

API 呼び出しで定数を使用する方法

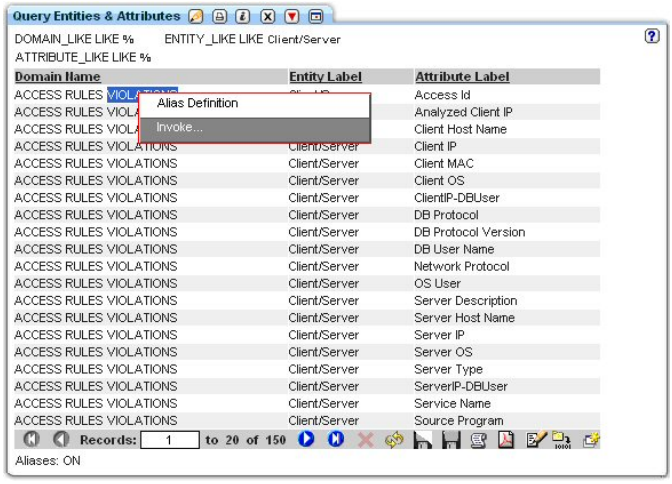
API 関数の呼び出し時に使用する新しいエンティティー属性を作成します。

付加価値: GUI を使用して、API 関数呼び出しでパラメーターの入力に使用できるユーザー定義の定数を作成します。

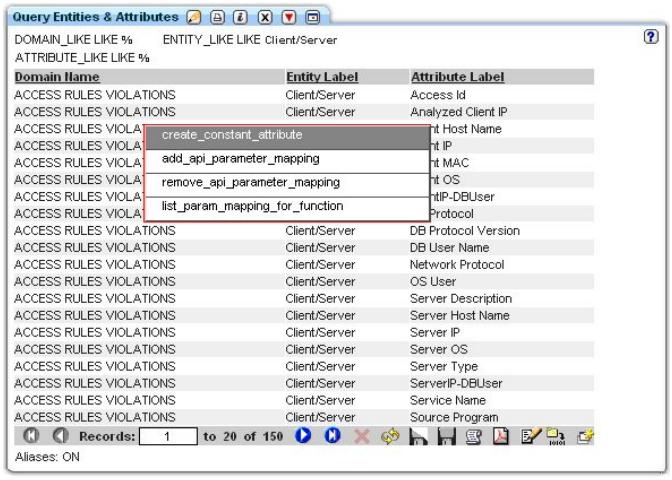
1. このレポートから、パラメーターのマッピングに使用できるフィールドを持つように、これを変更することができます。



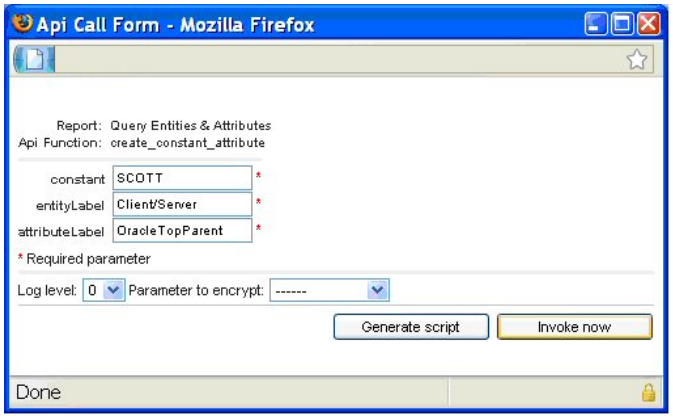
2. ACCESS RULES VIOLATIONS ドメイン内の「クライアント/サーバー」エンティティーの「照会エンティティーと属性」レポートに移動します。 行をダブルクリックして、「呼び出し...」オプションを選択します。



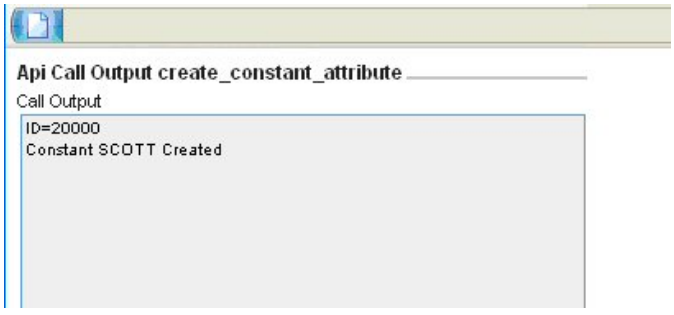
3. API 関数 create_constant_attribute を呼び出します。



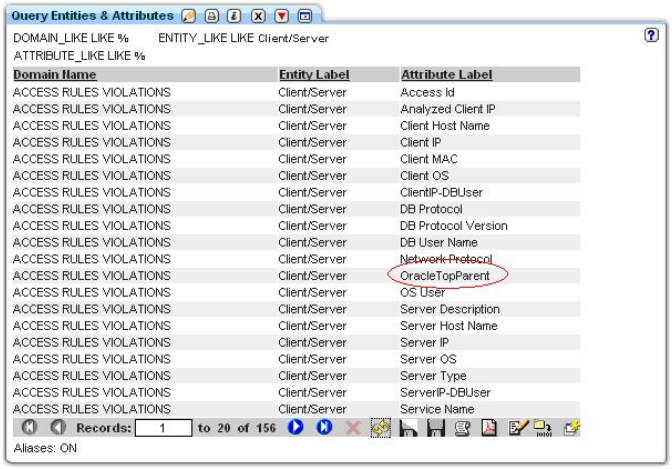
4. 使用する定数値 (「SCOTT」) を入力し、名前を付ける attributeLabel (「OracleTopParent」) を入力してから、「今すぐ呼び出し」ボタンをクリックして定数を作成します。



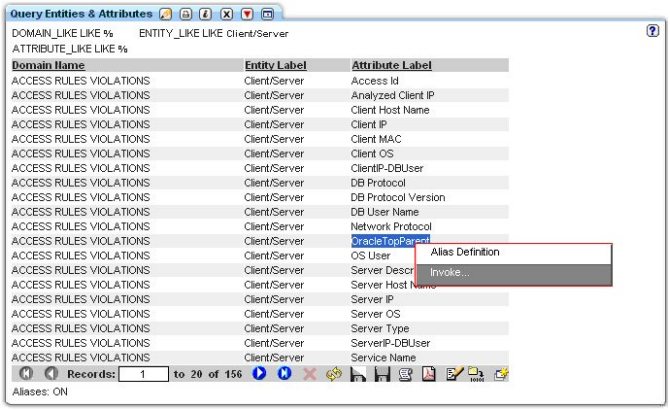
5. 「今すぐ呼び出し」ボタンをクリックすると、定数が作成されたことを示す「API 呼び出し出力」状況が生成されます。



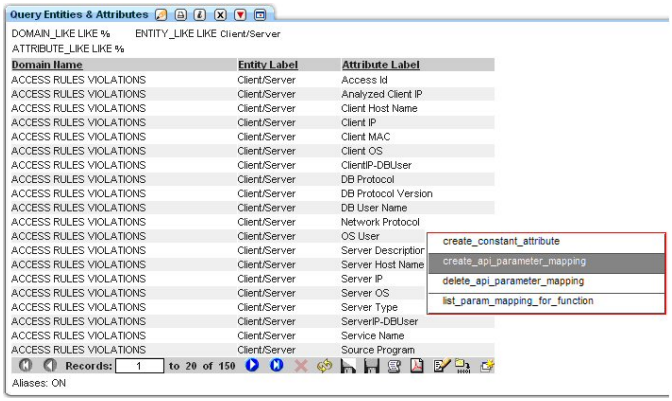
6. 「照会エンティティーと属性」レポートが再表示されて、作成された新規属性が表示されます。



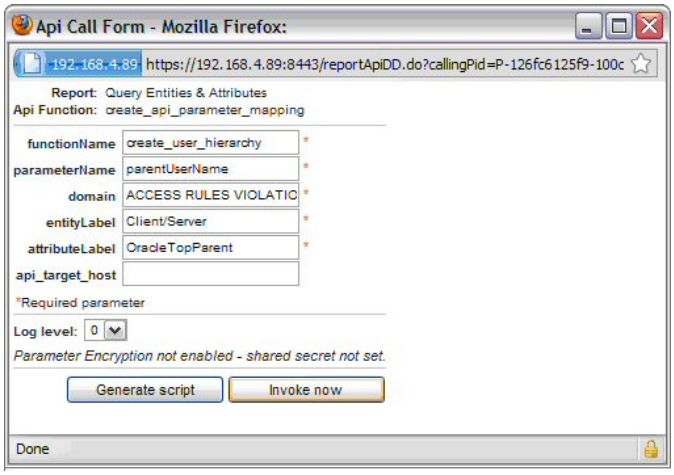
7. 次に、新規に作成された定数をレポート用にマップできます。 新規行をダブルクリックして、「呼び出し...」オプションを選択します。



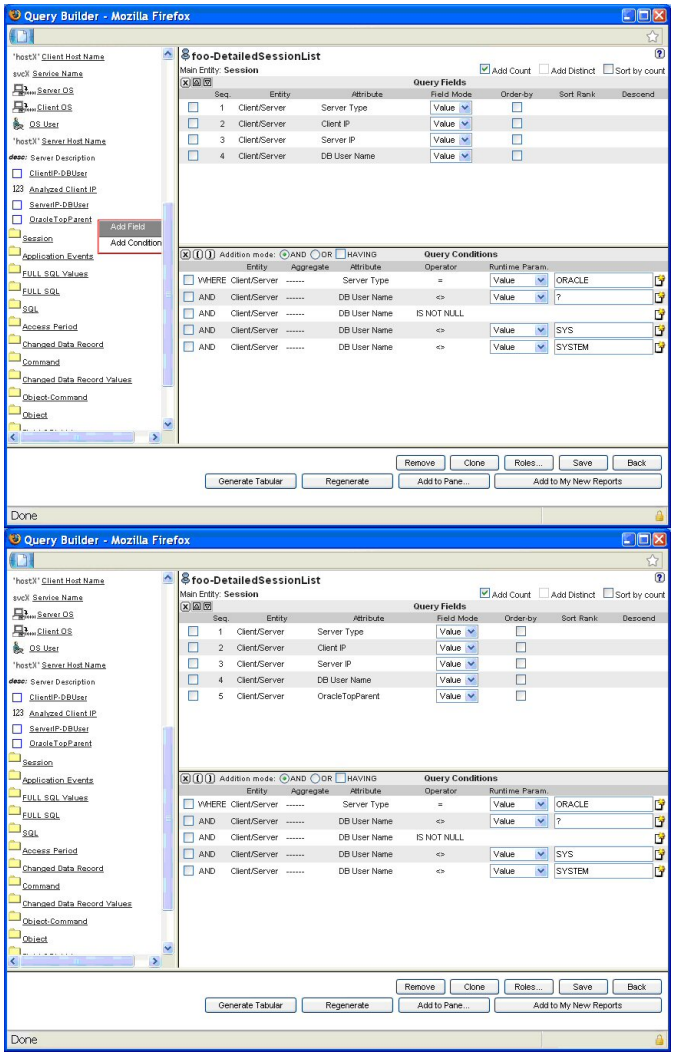
8. create_api_parameter_mapping オプションを選択します。



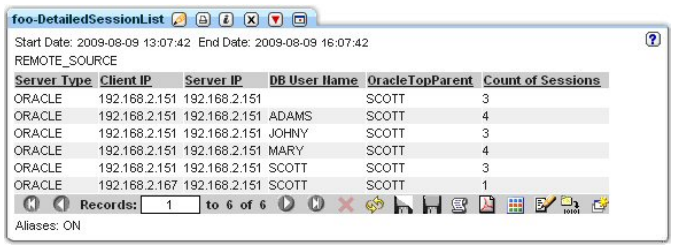
9. functionName と parameterName を入力して、「今すぐ呼び出し」ボタンをクリックします。



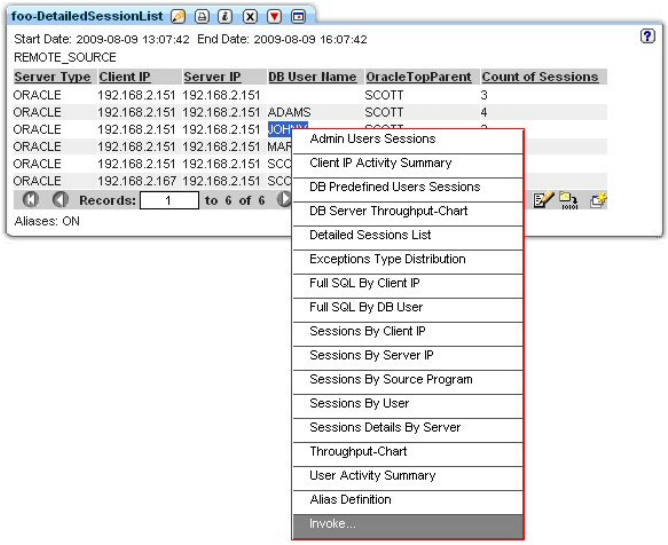
10. 新しく作成した属性はレポートに追加する必要があります。 クエリー・ビルダーを使用して照会を編集し、フィールドを追加します。



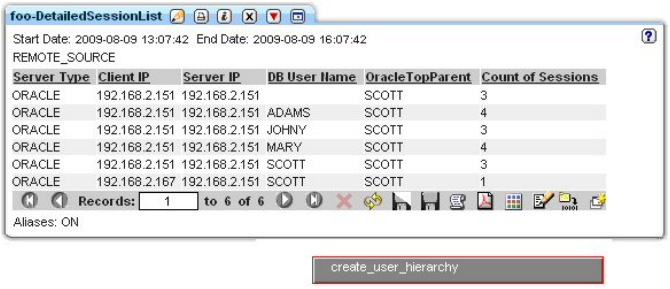
11. 次に、レポートが表示されると、新規属性が表示されます。



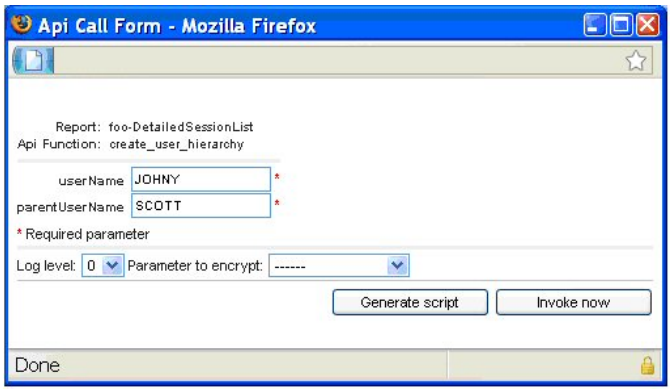
12. 新規定数の使用法を検証するには、行をダブルクリックして「呼び出し...」オプションを選択します。



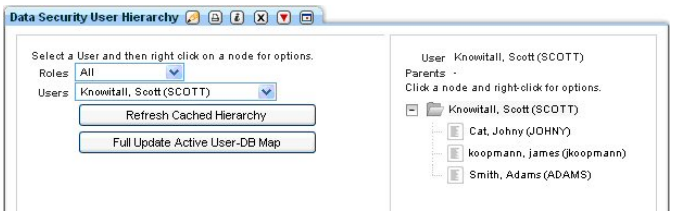
13. API 関数を選択します。



14. 次に、新規に追加した定数から parentUserName が取り込まれます。 「今すぐ呼び出し」ボタンをクリックします。



15. 新規の「データ・セキュリティー・ユーザー階層」を検証します。



親トピック: API 呼び出しおよびレポートの操作

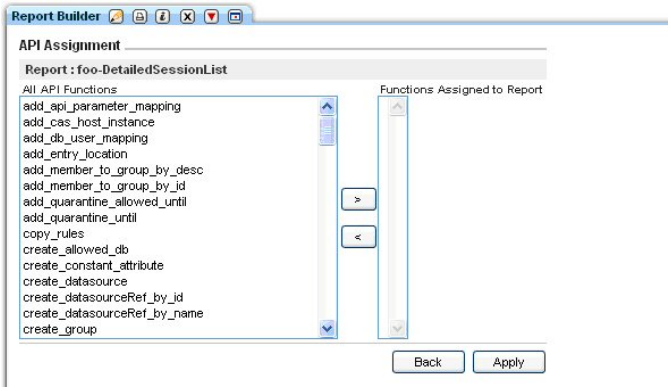
カスタム・レポートから API 呼び出しを使用する方法

API 関数をレポートにリンクし、レポートの各フィールドを API 関数のパラメーターにマップします。

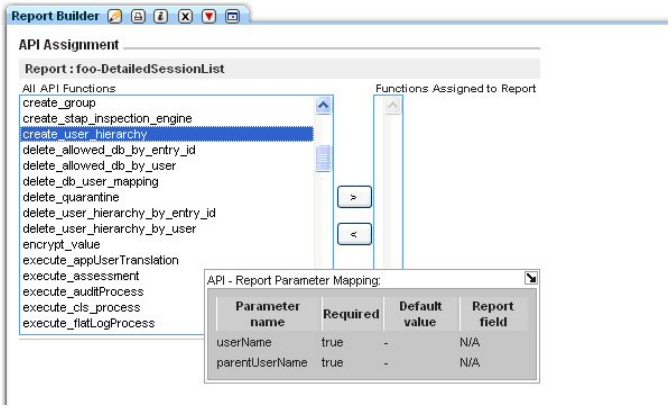
付加価値: GUI を介して、API 関数呼び出しで使用するカスタム・レポート・フィールドに API パラメーターを迅速かつ簡単にマップします

デフォルトでは、新しく作成したカスタム・レポートには API 関数はリンクされていません。レポートへの API 関数のリンクは、Guardium の照会 - レポート・ビルダーから行います。

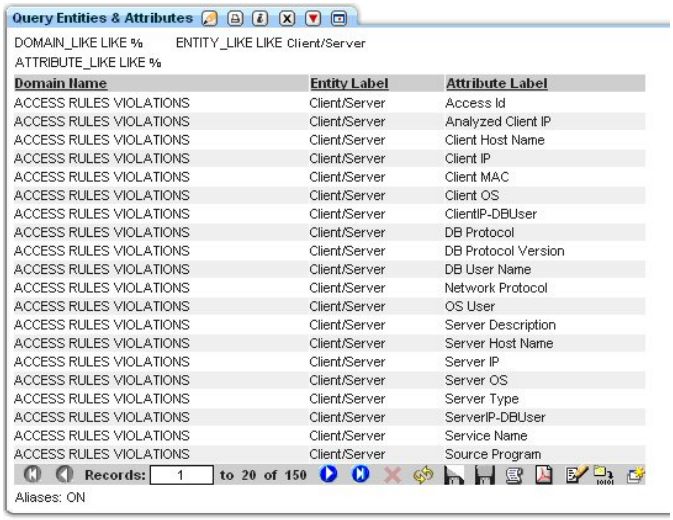
1. 照会 - レポート・ビルダーを開き、ご使用のカスタム・レポートを見つけてから、「API 割り当て」ボタンをクリックします。
2. 「API 割り当て」パネルには、選択したレポートに割り当てられたすべての API 関数が表示されます。このシナリオでは、選択したレポートには API 関数は割り当てられていないことに注意してください。



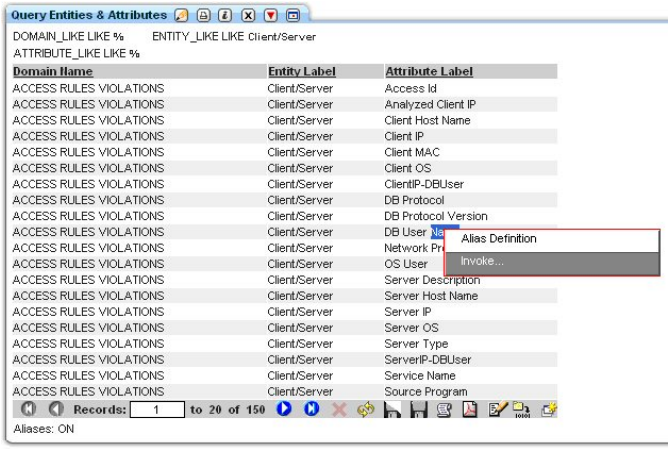
3. API 関数をレポートに割り当てるため、レポートにリンクする API を見つけ、「>」をクリックして「適用」ボタンをクリックします。 このシナリオでは、create_uer_hierarchy を選択しました。 選択すると、パラメーター・マッピング (API 関数呼び出しの際にどのレポート・フィールドが使用されるか) が表示されたポップアップ・ウィンドウが表示されます。 パラメーター名にマップされたレポート・フィールドはないことに注意してください。



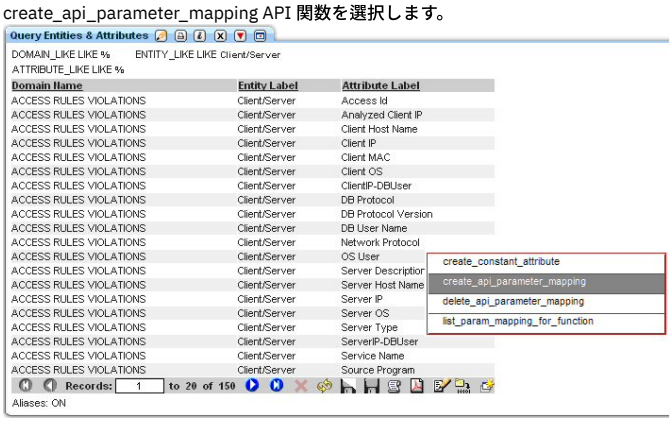
4. この時点では、どのレポート・フィールドも API パラメーターにマップされていません。 ユーザーは、「照会エンティティーと属性」レポートに移動してこれらのマッピングを作成できます。 これを行わない場合、API 呼び出しを行ったときに、どのパラメーターにも値はありません。 API パラメーター・マッピングを追加します。 「照会エンティティーと属性」レポートを開いて、マッピングを作成します。 このシナリオのレポートでは、ACCESS RULES VIOLATIONS ドメイン内では「クライアント/サーバー」エンティティーを使用するので、「カスタマイズ」ボタンを使用してレポートをフィルタリングし、「クライアント/サーバー」エンティティーのみを表示するようにレポートを変更します。



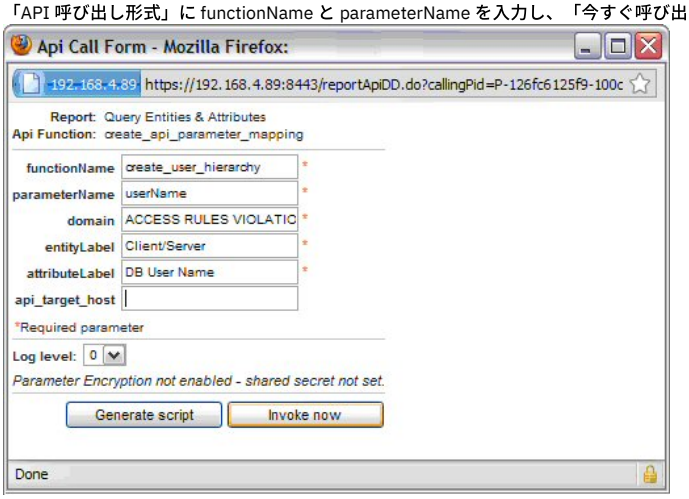
5. パラメーター名に割り当てる属性をダブルクリックして、「呼び出し...」オプションをクリックします。



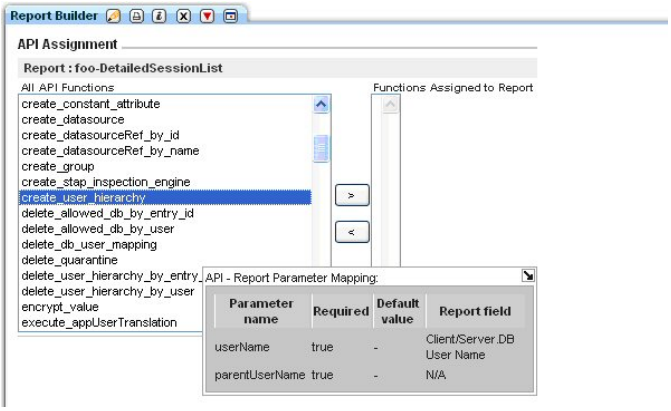
6. create_api_parameter_mapping API 関数を選択します。



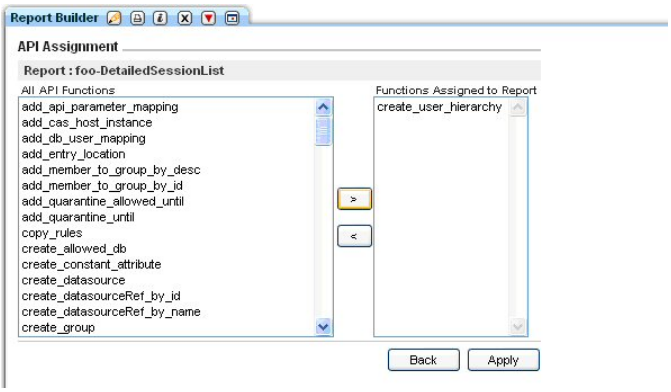
7. 「API 呼び出し形式」に functionName と parameterName を入力し、「今すぐ呼び出し」ボタンをクリックします。



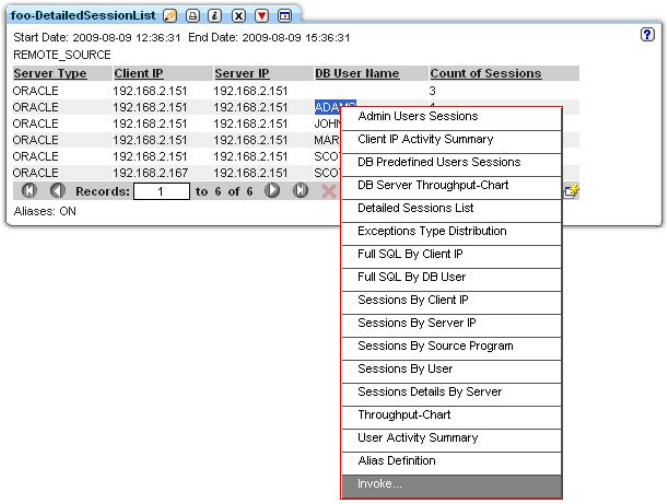
8. レポート・ビルダーの先ほどのレポートに戻って、「API 割り当て」を確認します。「create_user_hierarchy」API 関数をクリックして、レポート・マッピングを表示します。「UserName」が「クライアント/サーバー・データベース名」レポート・フィールドにマップされています。



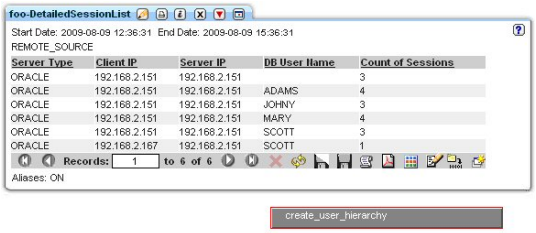
9. 「>」をクリックして「適用」ボタンをクリックします。



10. 次に、このレポートを介して create_user_hierarchy API 関数を呼び出すと、パラメーター userName がレポートから取り込まれます。 これを確認するには、レポートに戻って行をダブルクリックしてから、「呼び出し...」オプションをクリックします。

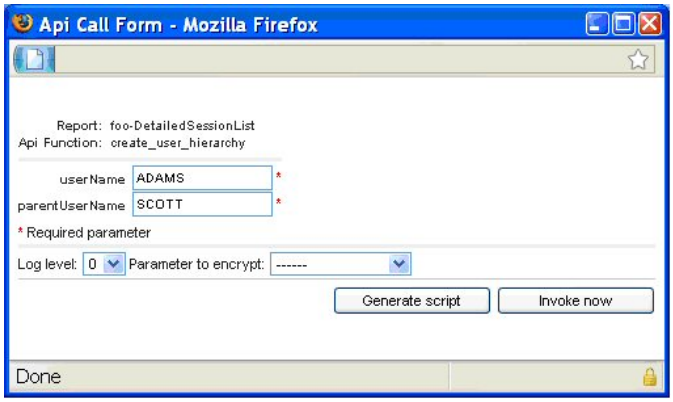


11. API 関数 (この場合は create_user_hierarchy) をクリックします。

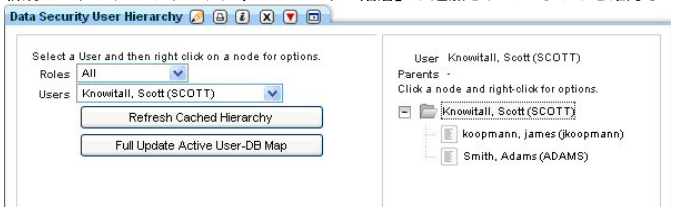


12. userName はすでにレポート・フィールドから取り込まれていることに注意してください。

13. parentUserName を入力して「今すぐ呼び出し」ボタンをクリックします。



14. 新規の「データ・セキュリティー・ユーザー階層」が追加されていることを確認してください。



親トピック: API 呼び出しおよびレポートの操作

外部フィードの操作

Guardium レポート・データを外部データベースに直接送信するには、外部フィードを使用します。レポート・データを外部データベースに送信することは、いくつかのシナリオで役に立ちます。例えば、Guardium データを非 Guardium データと結合または関連付ける場合や、外部ツールで Guardium データを使用する場合、あるいは、特に大規模なレポートでレコードのマシン構文解析を行う場合です。

- 外部フィードのマッピング
  外部フィードをマップして、Guardium レポート・データを外部データベースに直接送信する方法について説明します。
- 外部フィードの作成
  外部フィード・タスクは、タスクおよびターゲットの外部データベースを定義します。

親トピック: レポート

外部フィードのマッピング

外部フィードをマップして、Guardium レポート・データを外部データベースに直接送信する方法について説明します。

始める前に

外部フィードをマップする前に、以下の前提条件を確認してください。

- フィードからデータを受信する外部データベースを特定し、そのデータベースに必要な接続情報 (IP アドレス、ポート番号、ユーザー名、パスワードなど) を収集します。外部フィードは、現在はリレーショナル・データベースをサポートしており、他のデータベース・タイプでは機能しない可能性があります。
- 外部フィードにデータを提供する Guardium レポートを特定します。

このタスクについて

外部フィードを使用すると、Guardium レポート情報を外部データベースに直接送信できます。レポート内に定義できるものはすべて外部フィード経由で送信できます。これらのフィードは、Guardium のレポート・メカニズムから外部データベース上の表フィールドへの DOMAIN_ID および ATTRIBUTE_ID のマッピングに依存しています。各マッピングは、4 つの表 (EF_MAP_TYPE_HDR、EF_MAP_TABLE、EF_MAP_COLUMN、および EF_MAP_GDM_TYPE) 内のレコードから成っています。grdapi_create_ef_mapping 関数を使用すると、これらの表を作成しマッピングを設定することができます。

手順

1. 外部フィードを使用して転送したいデータを含むレポートを生成します。必要なレポート・データにシステムがアクセスできる場合は、中央マネージャー、アグリゲーター、またはスタンドアロン Guardium インスタンスからこの操作を実行できます。
2. CLI から、`grdapi create_ef_mapping reportName="My report"` を実行します。grdapi_create_ef_mapping 関数は、マッピングを設定するだけでなく、後続のステップで使用される create table ステートメントのサンプルも生成します。
3. レポートが定義される Guardium システム上の /var/log/guard で、ef_sample_[my_report].sql などのファイル名を検索します。このファイルには、create table ステートメントの例が含まれています。 外部データベースの要件に合うようにこのファイル内のステートメントを変更する必要があります。ファイルを変更した後、外部データベースに対してステートメントを実行して、ターゲット表を作成します。

タスクの結果

これで外部フィードは、監査プロセス・ビルダーを通して定義されたワークフロー・プロセス内で使用できます。外部フィードの作成に進みます。
親トピック: 外部フィードの操作

外部フィードの作成

外部フィード・タスクは、タスクおよびターゲットの外部データベースを定義します。

外部フィードを使用する前に、以下の前提条件を確認してください。

- 外部フィードのマッピング: Guardium と外部データベースの間のフィードをマップします。外部フィードは、現在はリレーショナル・データベースをサポートしており、他のデータベース・タイプでは機能しない可能性があります。
- 「外部データ・フィード」は、オプション・コンポーネントです。(プロダクト・キーによって) 有効にする必要があります。そうしないと、外部フィードを作成できません。
- 外部フィード経由で送信するデータを定義するレポートを作成します。 事前定義レポートは、外部フィードでは動作しません。事前定義レポートを使用したい場合は、レポートのコピーを作成し、そのコピーを外部フィードに使用します。

オプションの外部フィード・タスクが初めて実行されるときに、必要な監査ソースの内部表現が作成されます。監査ソースの作成日よりも前の日付のタイム・スタンプがあるデータは保管できない、という制限が 1 つあります。 つまり、タスクが初めて実行されるとき、現在日付のデータのみがエクスポートされるという意味です。その日付の後に続いてタスクを実行すると、その日付以降のすべてのデータをエクスポートできます。(言い換えれば、翌日は、その日のデータに加えて前日のデータもエクスポートできます。)

コンプライアンス・ワークフロー自動化プロセスの定義をまだ開始していない場合は、この手順を実行する前に、ワークフロー・プロセスの作成を参照してください。

1. 「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」にナビゲートして、 をクリックします。「新規監査プロセスの作成」ウィンドウが開きます。
2. プロセスの名前を入力します。
3. 「タスクの追加」リボンを開き、 をクリックします。
4. 「タスク・タイプ」ドロップダウン・リストから、「外部フィード」を選択します。「新規タスク」ウィンドウが最新表示され、外部フィードのパラメーターが表示されます。
5. 以下のパラメーターを入力します。
   - 名前: 固有のタスク名
   - フィード・タイプ: (次に表示されるコントロールは、選択したフィード・タイプに応じて異なります。)
   - 外部フィード・イベント:
   - レポート: 選択したレポートに応じて、「タスク・パラメーター」ペインに表示されるパラメーターの数が異なります。
   - 「抽出ラグ」ボックスにフィードが遅延される時間数を入力し、「継続」ボックスにマークを付けて監査タスクが実行される直前の時間までのデータを組み込みます。抽出ラグは、「継続」ボックスにマークが付いている場合のみ動作します。
   - データ・ソース: 外部フィードの 1 つ以上のデータ・ソースを指定するか、 をクリックして新規データ・ソースを作成します。
   - 「タスク・パラメーター」ペインに、すべてのパラメーター値を入力します。 パラメーターは、選択したレポートに応じて異なります。カウント列は、外部フィードではサポートされません。
6. 「OK」をクリックします。

親トピック: 外部フィードの操作
関連概念:
監査プロセスの作成
関連タスク:
外部フィードのマッピング

z/OS のレポートの作成

組み込みレポートとサンプル照会をカスタマイズすることで z/OS データ・ソース用の Guardium レポートを作成する方法について説明します。

z/OS データ・ソースのレポートを作成するプロセスは他のデータベースでも同じですが、メインフレームの概念と Guardium のレポート・エンティティーおよび属性の間には必ずしも直接的な対応はありません。 監査員とメインフレーム担当者がコミュニケーションを取りやすいように、このセクションではメインフレーム・イベント・データと Guardium のエンティティーおよび属性とのマッピングを概説しています。カスタマイズ可能ないくつかの組み込みレポートがあり、この情報では標準的な監査シナリオに役立つ追加の照会について説明しています。

親トピック: レポート
関連概念:
ドメイン、エンティティー、および属性
照会 - レポート・ビルダーの使用
ドメインのエンティティーおよび属性

評価および強化

Guardium® の「脆弱性評価」ソリューションは、IT 環境に対するセキュリティーおよびコンプライアンスのライフサイクル管理における最初のステップです。事前定義アセスメントまたはカスタム・アセスメントとプロセス・ワークフロー監査のセットを使用して、自動化方式でデータベースの脆弱性を特定および処置し、予防的に構成を改善し、インフラストラクチャーを強化することができます。

- Guardium 脆弱性評価の紹介
  Guardium 脆弱性評価では、データベース・インフラストラクチャー内のセキュリティー脆弱性を特定し、修正することができます。
- 脆弱性評価のタイプ
  Guardium には、データベース構成のパラメーター、特権、およびその他の脆弱性などを検査するための、2000 を超える事前定義テストがあります。一部のテストを特定の要件に対応するためにカスタマイズすることもできます。
- 評価
  アセスメントとは、データベース・インフラストラクチャーの脆弱性をスキャンし、リアルタイム測定と履歴測定によるデータベースおよびデータ・セキュリティーの正常性評価を行う一連のテストを指します。
- 必要とされるスキーマ変更
  Guardium V9.1 では、IBM DB2 for z/OS 上での脆弱性評価テストで使用されるスキーマが変更されています。9.1 より前のリリースからアップグレードする場合、

これらのテストを引き続き使用するためには、データベースを更新する必要があります。

- RACF の脆弱性の評価
  IBM DB2 for z/OS を使用する場合は、脆弱性評価テストで RACF の脆弱性を評価することができます。RACF アセスメントを使用するには、少なくともバージョン 9.1 の Guardium がインストールされている必要があります。
- 構成監査システム (CAS)
  構成監査システム (CAS) は、サーバー環境に対する変更をトラッキングして報告します。例えば、構成ファイル、環境変数やレジストリー変数、他のデータベース・コンポーネントやオペレーティング・システムのコンポーネント (データベース管理システムやオペレーティング・システムが使用する実行可能ファイルやスクリプトを含む) の変更などです。このデータは Guardium システム上で使用可能であり、レポートやアラートに使用できます。

Guardium 脆弱性評価の紹介

Guardium 脆弱性評価では、データベース・インフラストラクチャー内のセキュリティー脆弱性を特定し、修正することができます。

データベース脆弱性評価は、データベース・インフラストラクチャーで脆弱性をスキャンし、リアルタイム測定および履歴測定によるデータベースおよびデータ・セキュリティーの正常性を評価するために使用されます。

脆弱性評価では、以下の 3 タイプの成果物を使用します。

テスト
: テストでは、特定の脅威または関心面に対する脆弱性についてデータベース環境が検査されます。

アセスメント
: アセスメントとは、まとめて実行される一連のテストが含まれたジョブを指します。

データ・ソース
: データ自体のソース (データベースや XML ファイルなど)、およびデータへのアクセスに必要な接続情報。

Guardium® 脆弱性評価アプリケーションを使用すると、組織は、一貫性のある自動化された方式で、データベースの脆弱性を識別および処置することができます。Guardium の評価プロセスでは、以下を行うことによって、データベース環境の正常性を評価し、改善方法を推奨します。

- ベスト・プラクティスと対比してシステム構成を評価し、データベース・リソースに対する脆弱性または潜在的な脅威を (構成および動作上のリスクも含めて) 検出します。 例えば、無効化されていないすべてのデフォルト・アカウントを識別する、PUBLIC 特権および選択した認証方式を検査する、などです。
- セキュリティー・パッチの欠落など、IT 環境に内在する脆弱性を検出します。
- 最もクリティカルなリスクおよび脆弱性が発見された領域をベースに、アクション・プランを推奨し、優先順位付けします。 レポートおよび推奨を生成することで、コンプライアンスの変更に対応し、評価されたデータベース環境のセキュリティーを向上させる方法についてのガイドラインが提供されます。

Guardium のデータベース脆弱性評価では、2 つの中心的なテスト方式を組み合わせることにより、全範囲を網羅しています。 この方法では、複数の情報ソースを利用して、データベースとデータ環境のセキュリティー正常性の全体像を作成します。

1. エージェント・ベース - 各エンドポイント (例えば、データベース・サーバー) にインストールされたソフトウェアを使用。 これらにより、データベース・コンソールからの管理者による機密データへの直接アクセスなどの、リモートでは判別できないエンドポイントの側面を判別できます。
2. スキャン - 資格情報によるアクセスを介したネットワーク上でのエンドポイントに対する問い合わせ。

Guardium「脆弱性および脅威の管理」ソリューションには、以下が含まれています。

- データベース・オートディスカバリー - データベース環境のネットワーク・オートディスカバリーを実行し、データベースのクライアントとサーバー間の対話のグラフィカル表現を作成します。
- データベース・コンテンツ分類 - 機密データ (16 桁のクレジット・カード番号および 9 桁の社会保障番号など) を自動的に検出、分類して、問題のあるビジネスや、機密データを保管する IT プロセスを組織が迅速に識別するのに役立ちます。
- データベース脆弱性評価 - データベース・インフラストラクチャーで脆弱性をスキャンし、リアルタイムの測定および履歴測定によるデータベースおよびデータ・セキュリティーの正常性の評価を行います。
- CAS (構成監査システム) - データベース構造、セキュリティーおよびアクセス制御、重要なデータ値、データベース構成ファイルなどの項目に対して加えられたすべての変更をトラッキングします。
- コンプライアンス・ワークフロー自動化 - 評価と強化策、アクティビティー・モニターに始まり、監査レポート、レポート配布、および主要な利害関係者によるサインオフに至るまで、コンプライアンス・プロセス全体を自動化します。

CAS (構成監査システム) は、脆弱性および脅威の識別において重要な役割を果たします。Guardium の事前構成およびユーザー定義された CAS テンプレートを評価テストで使用して、ユーザーのデータベース環境の履歴ビューを起動することができます。 CAS を使用すると、Guardium はデータベースに対する脆弱性を OS レベル (ファイルのアクセス権、所有権、および環境変数など) で識別できます。 これらのテストは名前に *Assessment* というワードが含まれており、「CAS テンプレート・セット定義」パネルから参照可能です。

注: 脆弱性評価 (VA) および構成監査システム (CAS) は、英語のみでサポートされています。

Common Vulnerabilities and Exposures (CVE®) は、公に知られた機密保護の脆弱性についての共通名称 (つまり CVE ID) の辞書です。 CVE の共通 ID を使用することにより、別個のネットワーク・セキュリティー・データベースやツールの間でデータが共有しやすくなり、対象を評価するためのベースラインが提供されます。 例えば、レポートに CVE ID が組み込まれていると、ユーザーは、1 つ以上の別個の CVE 互換データベースにあるフィックス情報に素早く正確にアクセスし、問題を修正することができます。

多くの組織が、CVE ID を組み込むことにより、自社の機密保護製品およびサービスを CVE 互換にしています。Guardium は、MITRE Corporation の提供する Common Vulnerabilities and Exposures (CVE) を常にモニターしています。 関連するデータベース関連の脆弱性を調べるために、これらのテストが追加されます。

ユーザーが特定のデータベースの CVE 名を表示して、個々の脆弱性を検出する助けとして、セキュリティー・アセスメント・ビルダーを使用してテストを構成する際に、必要なデータベースの CVE ラジオ・ボタンを選択し、該当する CVE ID を選択および追加することができます。 追加情報は、MITRE Corporation が保守する CVE リストのマスター・コピーで常に検索できます。

Guardium ソリューションで CVE を最新の状態に保つため、Guardium では最新の CVE データベースをダウンロードして使用し、データベース表にすべての最新 CVE 項目および候補を取り込みます。 Guardium は、ダウンロードした CVE データと、Guardium 脆弱性評価リポジトリー内に既にある CVE データを、プログラムを使用して比較し、レビュー用に新しい CVE のリストを生成します。 次いで、Guardium データベース・セキュリティー・チームは、これらの Guardium 脆弱性知識ベースの候補を手動でレビューし、テストして、関連するものを GA Guardium 脆弱性評価知識ベースに追加します。 これらのテストには該当する CVE 番号のタグが付けられ、一度 GA リポジトリーに入れられると、Guardium 脆弱性評価アプリケーションを使用してこれらのテストを自動的に実行できるようになります。

注:

- 脆弱性評価と資格レポートのどちらについても、資格レポートの特権を付与するためのスクリプトを検索する場合は、gdmmonitor_scripts ディレクトリー内のスクリプトを使用してください。entitlement_monitor_role フォルダーは更新されなくなったため、使用しないでください。
- 有効期限が切れた製品ライセンス・キーを使用した場合、またはデータ・ソースの数が制限されているライセンスを使用した場合は、次のメッセージが表示されることがあります。「データ・ソースを追加できません。データ・ソースのライセンス許容最大数に達しています。「ライセンス有効期限」の日付および「データ・ソースの数」は、「管理者コンソール」の「システム構成」パネルで確認できます。*N* 個のデータ・ソースがある脆弱性プロセスまたは分類プロセスは、実行されるたびに *N* 個のスキャンとしてカウントされます。
- Guardium「脆弱性評価」では、評価対象データベースへのアクセス権が必要です。これを行うために、Guardium には、Guardium で使用するデータベース内にユーザーとロールを作成する一連の SQL スクリプト (データベース・タイプごとに 1 つのスクリプト) が用意されています。

  テンプレート・スクリプトが作成され、ファイル・サーバーを介してパス /log/debug-logs/gdmmonitor_scripts/ で検出およびダウンロードが可能になると、Guardium システムで使用できます。README.txt ファイルには、さらに詳しい情報が含まれています。

Guardium 脆弱性評価テストの例外

Guardium 脆弱性評価テストの例外グループには、データベースのインストール時に作成されたデフォルトのメンバー、スキーマ、オブジェクト、または特権が事前に設定されています。これらのグループを使用して、脆弱性評価の実行時の誤検出を防ぎます。評価が失敗した場合は、適切な例外グループをテストにリンクし、デフォルトのメンバーを除外して、テストを再度実行します。これでテストが違反なしで実行される場合、データベースのインストール時に作成されたデフォルトのメンバー、スキーマ、オブジェクト、または特権が初期違反の原因だったことを示します。

表 1. マッピングをテストするための脆弱性診断のグループ

グループ ID	グループ名	テスト名	テスト ID	データベース・タイプ
82	Sybase 許可された PUBLIC への特権の付与	非免除の特権が PUBLIC に付与されていない	61	SYBASE ASE
83	MS-SQL 許可された PUBLIC への特権の付与	非免除の特権が PUBLIC に付与されていない	270	MSSQL
115	Db2 許可された PUBLIC への特権の付与	オブジェクト特権が PUBLIC に付与されていない	105	Db2 LUW
144	Db2 非制限的に許可された PUBLIC への特権の付与	オブジェクト特権が PUBLIC に付与されていない	105	Db2 LUW
116	Teradata 許可された PUBLIC への特権の付与	PUBLIC に付与されているオブジェクト特権	2029	TERADATA
117	PostgreSQL 許可された PUBLIC への特権の付与	PUBLIC に付与されているオブジェクト特権	315	POSTGRESQL
118	Netezza 許可された PUBLIC への特権の付与	PUBLIC に付与されたオブジェクト特権 (Netezza)	2053	NETEZZA
65	MS-SQL データベース管理者	固定サーバー・ロールに関する権限がデータベース管理者のみに付与されている	159	MSSQL
165	Oracle データベース管理者のみが SYS.USER$ にアクセスできる	データベース管理者のみが SYS.USER$ にアクセスできる	222	ORACLE
166	MS-SQL ユーザーに付与されている DDL	ユーザーに付与されている DDL	321	MSSQL
167	MS-SQL ユーザーに付与されているプロシージャー	ユーザーに付与されているプロシージャー	322	MSSQL
168	MS-SQL 個々のユーザーに特権が付与されていない	個々のユーザーに特権が付与されていない	154	MSSQL
170	PUBLIC に付与されている Sybase IQ プロシージャーおよび関数特権	プロシージャーおよび関数に対する特権が PUBLIC に付与されました。	2230	SYBASE IQ
171	Sybase IQ 個別のプロシージャー特権または関数特権がない	個別のプロシージャー特権または関数特権がありません。	2227	SYBASE IQ
172	MS-SQL レジストリー・アクセス拡張プロシージャーへのアクセス権限がない	レジストリー・アクセス拡張プロシージャーへのアクセス権限がない	215	MSSQL
173	MS-SQL ロールに付与されているロール	ロールに付与されているロール	323	MSSQL
185	サーバー・レベルの MS-SQL アクセス権限が、データベース管理者以外のユーザーに付与されました	サーバー・レベルのアクセス権限が、データベース管理者以外のユーザーに付与されました	2289	MSSQL
186	MS-SQL MSDB データベースのロール・メンバー特権	MSDB データベースのロール・メンバー特権	2296	MSSQL
48	Db2 データベース Version+Patches	バージョン: Db2	16	Db2 LUW
48	Db2 データベース Version+Patches	Db2 パッチ・レベル	54	Db2 LUW

グループ ID	グループ名	テスト名	テスト ID	データベース・タイプ
49	Informix データベース Version+Patches	バージョン: Informix	17	INFORMIX
49	Informix データベース Version+Patches	Informix パッチ・レベル	55	INFORMIX
50	MS SQL Server データベース Version+Patches	バージョン: Microsoft SQL Server	18	MSSQL
50	MS SQL Server データベース Version+Patches	Microsoft SQL Server パッチ・レベル	56	MSSQL
51	MySql データベース Version+Patches	バージョン: MySql	19	MYSQL
51	MySql データベース Version+Patches	MySql パッチ・レベル	57	MYSQL
52	Oracle データベース Version+Patches	Oracle パッチ・レベル	58	ORACLE
52	Oracle データベース Version+Patches	バージョン: Oracle	20	ORACLE
53	Sybase データベース Version+Patches	バージョン: Sybase	21	SYBASE ASE
53	Sybase データベース Version+Patches	Sybase パッチ・レベル	59	SYBASE ASE
109	Teradata PDE Version+Patches	バージョン: Teradata PDE	284	TERADATA
109	Teradata PDE Version+Patches	Teradata PDE パッチ・レベル	286	TERADATA
110	Teradata TDBMS Version+Patches	Teradata TDBMS パッチ・レベル	287	TERADATA
110	Teradata TDBMS Version+Patches	バージョン: Teradata TDBMS	285	TERADATA
111	Teradata TDGSS Version+Patches	バージョン: Teradata TDGSS	290	TERADATA
111	Teradata TDGSS Version+Patches	Teradata TDGSS パッチ・レベル	288	TERADATA
112	Teradata TGTW Version+Patches	バージョン: Teradata TGTW	291	TERADATA
112	Teradata TGTW Version+Patches	Teradata TGTW パッチ・レベル	289	TERADATA
113	Netezza Version+Patches	Netezza バージョン・レベル	306	NETEZZA
113	Netezza Version+Patches	Netezza パッチ・レベル	307	NETEZZA
114	Postgress Version+Patches	PostGreSQL バージョン・レベル	308	POSTGRESQL
114	Postgress Version+Patches	PostGreSQL パッチ・レベル	309	POSTGRESQL
169	SybaseIQ データベース Version+Patches	バージョン: Sybase IQ	377	SYBASE IQ
169	SybaseIQ データベース Version+Patches	Sybase IQ パッチ・レベル	378	SYBASE IQ

MongoDB

2007 年に開発された MongoDB は、NoSQL のドキュメント指向データベースです。MongoDB では、動的スキーマに基づく JSON ドキュメントを使用します (このフォーマットは BSON と呼ばれます)。MongoDB では、コレクションは RDBMS 表に相当し、ドキュメントは RDBMS 表内のレコードに相当します。

MongoDB は、大きく急速に成長している NoSQL データベース・システムです。Web アプリケーションでよく見られる JSON ドキュメントなどの非リレーショナル形式データはプログラミングが容易であるため、運用システムや Web アプリケーションのバックエンドとして使用される傾向にあります。

- Guardium 脆弱性評価 (VA) でサポートされる最初の NoSQL データベースです。
- 最初の非 JDBC データベース接続です。接続では Java ドライバーが使用されます。
- MongoDB データ・ソースは、SSL クライアント証明書を使用した SSL サーバーとクライアント/サーバーの接続をサポートしています。
- MongoDB クラスターの Guardium の VA ソリューションは、複数の Mongo (レプリカ・セットの 1 次ノードとすべての 2 次ノード) 上で実行することができます。
- MongoDB では資格レポートおよび照会ベース・ビルダーはサポートされません。

SSL を使用した MongoDB データ・ソース

自己署名用に別途用意したサーバー証明書をインポートできます。お客様が独自の証明書をインポートすることもできます。 さらに、証明書は中央マネージャー上で機能し、コレクターにプッシュダウンされます。

CAS for MongoDB

Mongo CAS アセスメント・テンプレートを使用すると、データ・ソース内の複数のパスを指定して、ファイル・システムのさまざまなコンポーネントをスキャンすることができます。

Teradata Aster

Aster データ

2011 年に Teradata によって買収されました。通常は、データウェアハウジングおよび分析アプリケーション (OLAP) に使用されます。Aster Data は、構造化照会言語 (SQL) を MapReduce 内で使用できるようにする SQL-MapReduce と呼ばれるフレームワークを構築しました。最もよく連想されるのは、クリック・ストリーム系のアプリケーションです。

クイーン・ノードですべてのテストを実行するには、セキュリティー・アセスメントを作成する必要があります。Aster Data 用のデータベース接続はすべて、クイーン・ノードのみを通過します。

ワーカー・ノードとローダー・ノードでテストが必要となるのは、CAS テスト (ファイル許可とファイル所有権) を実行する場合のみです。

特権テストは、所定の Aster インスタンスに含まれるすべてのデータベースをループします。

SAP HANA

SAP HANA は、SAP SE によって開発され販売されている、メモリー内の列指向型リレーショナル・データベース管理システムです。HANA のアーキテクチャーは、同一プラットフォーム上で、高いトランザクション率と複雑な照会処理の両方に対処するように設計されています。

- 脆弱性評価および分類用のデータベース特権
  Guardium は、脆弱性評価の実行に必要な最小限の特権を備えたグループや役割の作成を簡素化する一連のスクリプトを備えています。
- Db2 for i 用の VA のデプロイ
  ユーザーのグループが脆弱性評価を実行できるようにして、テストを構成して実行します。
- Cloudera での VA の使用
  Apache Hadoop の Cloudera ディストリビューションで Guardium 脆弱性評価を使用する方法を説明します。

親トピック: 評価および強化

脆弱性評価および分類用のデータベース特権

Guardium は、脆弱性評価の実行に必要な最小限の特権を備えたグループや役割の作成を簡素化する一連のスクリプトを備えています。

始める前に

このタスクでは、Guardium システムからスクリプトをダウンロードし、データベース・サーバーでそのスクリプトを実行する必要があります。Guardium システムへのアクセスに使用するマシンの IP アドレスを特定する必要があります。これは、スクリプトをデータベース・サーバーに転送する前にダウンロードする個別のワークステーションの IP アドレスにすることも、データベース・サーバー自体の IP アドレスにすることもできます。

このタスクについて

Guardium 脆弱性評価の実行および Guardium 分類の使用には、データベースに対するアクセス権限および特定のデータベース特権が必要です。Guardium は、脆弱性評価の実行に必要な最小限の特権を備えたグループや役割の作成を簡素化する一連のスクリプトを備えています。作成されたグループまたは役割は、評価を実行する必要がある任意のデータベース・ユーザーに割り当てることができます。そのユーザーを使用して Guardium データ・ソースを作成して、VA スキャンを実行します。

ほとんどのデータベース・タイプをサポートするスクリプトが用意されており、データベース・ツール自体で実行するように設計されています。各スクリプトのスクリプト・ヘッダーに、詳細な説明が含まれています。各データベース・タイプに対して付与される特権は、スクリプトで各権限付与を見ることで確認できます。
重要: スクリプトを実行する前には、データベース管理者は、スクリプト・ヘッダーに含まれている説明を読み、スクリプトで実行されるデータベース・アクションを確認する必要があります。

手順

1. Guardium システムで、fileserver CLI コマンドを使用してファイル・サーバーを有効にします。 例えば、ファイル・サーバーを 1 時間有効にし、IP アドレスが `10.0.0.1` のシステムにスクリプトをダウンロードするには、以下のコマンドを使用します。

   ```
   fileserver 10.0.0.1 3600
   ```

   正常に開始されると、ファイル・サーバーで以下のような出力が表示されます。

   ```
   Starting the file server...
   The file server is ready at https://guardium.host.com:8445
   The timeout has been set to 3600 seconds and it may timeout during the uploading.

   The upload will only be accessible from the IP you are logged in from: 10.0.0.1
   ```

   ファイル・サーバーを停止するには `ENTER` を押してください。

2. スクリプトをダウンロードするマシンで、Web ブラウザーを使用してファイル・サーバーにアクセスします。 例えば、`https://guardium.host.com:8445` で実行されている Guardium システムの場合、以下の URL で脆弱性評価および分類用のスクリプトにアクセスします。

   ```
   https://guardium.host.com:8445/log/debug-logs/gdmmonitor_scripts/
   https://guardium.host.com:8445/log/debug-logs/classification_role/
   ```

   重要: Guardium 分類のディスカバリー・プロセスでは、脆弱性評価テストで求められるよりも高いレベルのデータベース・アクセス権限が必要になります。脆弱性評価では `gdmmonitor_scripts` 内のスクリプト、分類では `classification_role` 内のスクリプトを使用することをお勧めします。スクリプトを実行する前には、データベース管理者は、スクリプト・ヘッダーに含まれている説明を読み、スクリプトで実行されるデータベース・アクションを確認する必要があります。スクリプトを実行する前には、データベース管理者は、スクリプト・ヘッダーに含まれている説明を読み、スクリプトで実行されるデータベース・アクションを確認する必要があります。
3. Web ブラウザーで[右クリック] > 「名前を付けてリンク先を保存...」アクションまたは同様の機能を使用して、必要なスクリプトをダウンロードします。
   README.txt ファイルを確認して、特定のデータベース・タイプで使用するのに適したスクリプトを特定します。
   ヒント: Microsoft SQL Server 用のスクリプトは以下のとおりです。
   - gdmmonitor-mss.sql。Microsoft SQL Server 用です。

- gdmmonitor-mss-SA.sql。Microsoft SQL Server 脆弱性評価テストの 6 つで必要な管理特権を付与します。該当する特権を許可しなかった場合、特権が不十分であることを示すエラーがテストで返されます。該当する 6 つのテストは、使用可能なテストのわずか 5% にすぎません。

次のタスク

データベース・サーバーで必要なスクリプトをダウンロードしたら、スクリプト・ヘッダーに含まれている説明を念入りに確認し、その説明に従ってください。

親トピック: Guardium 脆弱性評価の紹介

Db2 for i 用の VA のデプロイ

ユーザーのグループが脆弱性評価を実行できるようにして、テストを構成して実行します。

このタスクについて

デプロイメントの手順

1. Guardium システムから脆弱性評価機能がデプロイされます。
2. Guardium に付属するスクリプトをターゲット・データベースに対して実行し、適切な特権を持つロールを作成します。次に、データベースに対するデータ・ソース接続を作成します。
3. セキュリティー・アセスメントを作成し、使用するデータ・ソースと実行するテストを選択します。
4. 実行したテストが完了すると、レポートが作成されます。このレポートには、合格したテスト項目と不合格だったテスト項目のほかに、保護を強化する必要がある箇所について、詳細な推奨事項が記録されます。

IBM for i バージョン・サポート:

IBM for i 6.1、7.1、7.2 のパーティション

VA テスト範囲 (合計で 115 件のテスト):

特殊権限を持つプロファイル

データベース関数の使用権限を持つプロファイル

パスワード・ポリシー

PUBLIC に付与されているデータベース・オブジェクト特権

個々のユーザーに付与されているデータベース・オブジェクト特権

GRANT オプションが設定されているデータベース・オブジェクト特権

セキュリティーの APAR

資格レポート:

特殊権限を持つプロファイル

ユーザーに付与されているグループ

PUBLIC に付与されているデータベース・オブジェクト特権

PUBLIC に付与されているデータベース実行可能オブジェクト特権

個々のユーザーに付与されているデータベース・オブジェクト特権

GRANT オプションが設定されているデータベース・オブジェクト特権

手順

1. 「グループ・ビルダー」を使用して、VA を使用するユーザーのグループを作成します。「設定」 > 「ツールとビュー」 > 「グループ・ビルダー」をクリックし、「グループ・ビルダー」を開きます。次のステップでは、gdmmonitor という名前のグループを対象としたスクリプトを使用しています。
2. Db2 for i システムで以下のスクリプトを実行し、VA の実行に必要な特権をこのグループに付与します。この処理は、データベースのネイティブ・クライアントを使用して、Guardium システムの外部で実行されます。

```
grant select on SYSIBMADM.FUNCTION_INFO to gdmmonitor;
grant select on SYSIBMADM.FUNCTION_USAGE to gdmmonitor;
grant select on SYSIBMADM.GROUP_PROFILE_ENTRIES to gdmmonitor;
grant select on SYSIBMADM.SYSTEM_VALUE_INFO to gdmmonitor;
grant select on SYSIBMADM.USER_STORAGE to gdmmonitor;
grant select on Qsys2.Authorizations to gdmmonitor;
grant select on SYSIBMADM.USER_INFO to gdmmonitor;
grant select on QSYS2.SYSSCHEMAAUTH to gdmmonitor;
grant select on QSYS2.SYSTABAUTH to gdmmonitor;
grant select on QSYS2.SYSPACKAGEAUTH to gdmmonitor;
grant select on QSYS2.SYSROUTINEAUTH to gdmmonitor;
grant select on QSYS2.SYSSEQUENCEAUTH to gdmmonitor;
grant select on QSYS2.SYSCOLAUTH to gdmmonitor;
```

IBM Db2 for i v7.1 以降の場合は、以下のスクリプトも含めてください。

```
grant select on QSYS2.SYSVARIABLEAUTH to gdmmonitor;
grant select on QSYS2.SYSXSROBJECTAUTH to gdmmonitor;
```

3. Db2 for i システムへの JDBC 接続を作成します。「データ・ソース・ファインダー」を開きます。これを行うには、「設定」 > 「ツールとビュー」 > 「データ・ソース定義」をクリックし、次に「アプリケーション選択」メニューから「セキュリティー・アセスメント」をクリックします。
    a. 「新規」をクリックして適切な情報を入力します。「接続プロパティー」で、「property1=com.ibm.as400.access.AS400JDBCDriver;translate binary=true」と入力します。
4. 「アセスメント・ビルダー」を使用して、アセスメントを作成します。 「強化」 > 「脆弱性評価」 > 「アセスメント・ビルダー」をクリックし、「アセスメント・ビルダー」を開きます。
    a. アセスメントの説明を入力します。
    b. 前のステップで作成したデータ・ソースを追加します。これを行うには、「データ・ソースの追加」をクリックし、「データ・ソース・ファインダー」からデータ・ソースを選択して「追加」をクリックします。
    注: テストを構成する前に、「適用」をクリックしてアセスメントを保存する必要があります。
5. 「テストの構成」をクリックし、アセスメントにテストを追加します。「IBM for i」タブをクリックし、追加するテストを選択して「選択の追加」をクリックします。
6. 「戻る」をクリックし、「セキュリティー・アセスメント・ファインダー」に戻ります。「今すぐ 1 回実行」をクリックしてテストを実行するか、「監査プロセス・ビルダー」を使用してテストをスケジュールします。「監査プロセス・ビルダー」を開くには、「ディスカバー」 > 「分類」 > 「監査プロセス・ビルダー」をクリックします。
7. 「結果の表示」をクリックすると、実行されたすべてのテストの詳細 (スコアを改善するための推奨事項を含む) が表示されます。

タスクの結果

テストが不合格だった場合、以下の対処が可能です。

- データベースにパッチを適用する (パッチに関係する問題がある場合)
- 推奨されるベスト・プラクティスに従い、データベースのパラメーターを再構成する
- 使用しているアプリケーションでは必要ないオブジェクトやシステム特権を取り消す
- 被付与者に直接付与されているオブジェクトを取り消し、ロールまたはグループに対してオブジェクト特権を付与し、被付与者をそのロールまたはグループに割り当てる
- パスワード・ポリシー設定を変更するか、ユーザーのデフォルト・パスワードを変更する
- 例外グループを作成し、不合格だったテストにそのグループをリンクしてもう一度テストを実行する (使用しているアプリケーションで、特定の権限付与が必要になる場合)

親トピック: Guardium 脆弱性評価の紹介

Cloudera での VA の使用

Apache Hadoop の Cloudera ディストリビューションで Guardium 脆弱性評価を使用する方法を説明します。

Cloudera Manager

データ・ソースのセットアップ

Cloudera Manager データ・ソースは、接続に Cloudera Manager Java API を使用します。JDBC は使用しません。

クラスター名をデータ・ソース GUI で定義する必要があります。クラスター名は、左側の Cloudera Manager GUI でのクラスター表示名です。

Cloudera Manager の脆弱性評価テストを実行するには、ほとんどの脆弱性評価テストで、読み取り専用ロールを持つデータ・ソース・ユーザーを定義する必要があります。しかし脆弱性評価テストの中には、データ・ソース・ユーザーがテストを実行するための最小限の特権としてクラスター管理者ロールを持っている必要があるものが少数あります。

以下の脆弱性評価テストでは、データ・ソース・ユーザーがクラスター管理者ロールを持っている必要があります。

1. 認証バックエンド順序
2. 管理コンソールの HTTP ポート
3. 管理コンソールの HTTPS ポート
4. サーバーに対してエージェントの TLS 認証を使用する
5. 管理コンソールに対して TLS 暗号化を使用する
6. エージェントに対して TLS 暗号化を使用する

この情報は、Cloudera Manager gdmmonitor スクリプト (/log/var-log-guard/gdmmonitor_scripts/gdmmonitor-Cloudera-Manager.sql) でも入手できます。

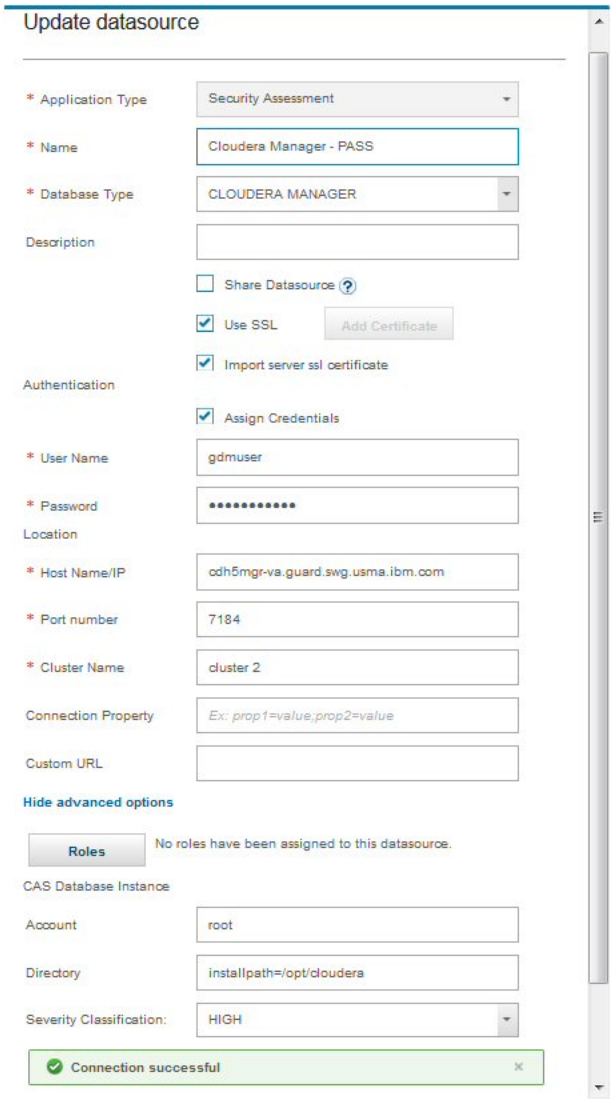
SSL が有効な場合、「SSL の使用」にチェック・マークを付け、「サーバーの SSL 証明書をインポートします」にチェック・マークを付けます。

「CAS データベース・インスタンス」の設定

アカウントは root でなければなりません。

ディレクトリーは、Cloudera Manager のインストール・パスとして定義する必要があります。例: installpath=/opt/cloudera

Cloudera Manager データ・ソースの設定の例。



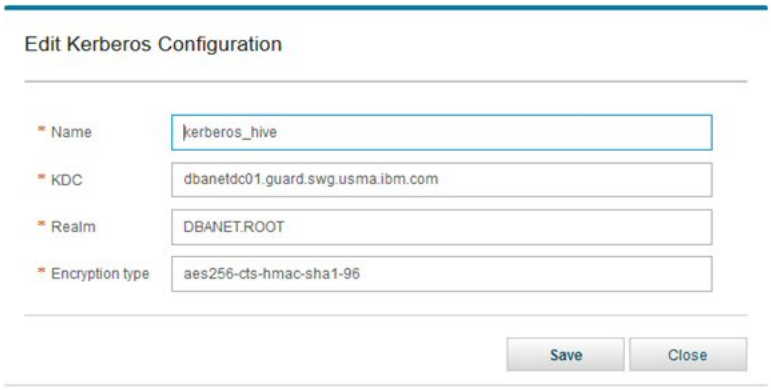
Hive

データ・ソースのセットアップ

Apache Hive JDBC ドライバー 1.1.1 を使用します。

Kerberos - ユーザー名とパスワードは有効な Kerberos ユーザー ID とパスワードでなければなりません。これは CA にも使用されます。ご使用の Kerberos ユーザー ID とパスワードを Hive の beeline コマンド行へのログインに使用できることをテストして確認します。

ご使用のアプライアンスの KDC およびレルムを定義する Kerberos 構成が既に作成されていることを確認してください。Guardium GUI で、「設定」>「ツールとビュー」>「Kerberos 構成」の順に進みます。Kerberos 構成が作成されていない場合は、+ アイコンをクリックして新規の Kerberos 構成を作成します。



Kerberos 構成を作成した後、それを選択して、データ・ソースのセットアップを構成できます。

SSL が有効な場合、「SSL の使用」ボックスにチェック・マークを付け、「サーバーの SSL 証明書をインポートします」ボックスにチェック・マークを付けます。

注: Hive は LDAP/SSL または Kerberos の一方のみサポートでき、両方はサポートできません。

「CAS データベース・インスタンス」の設定

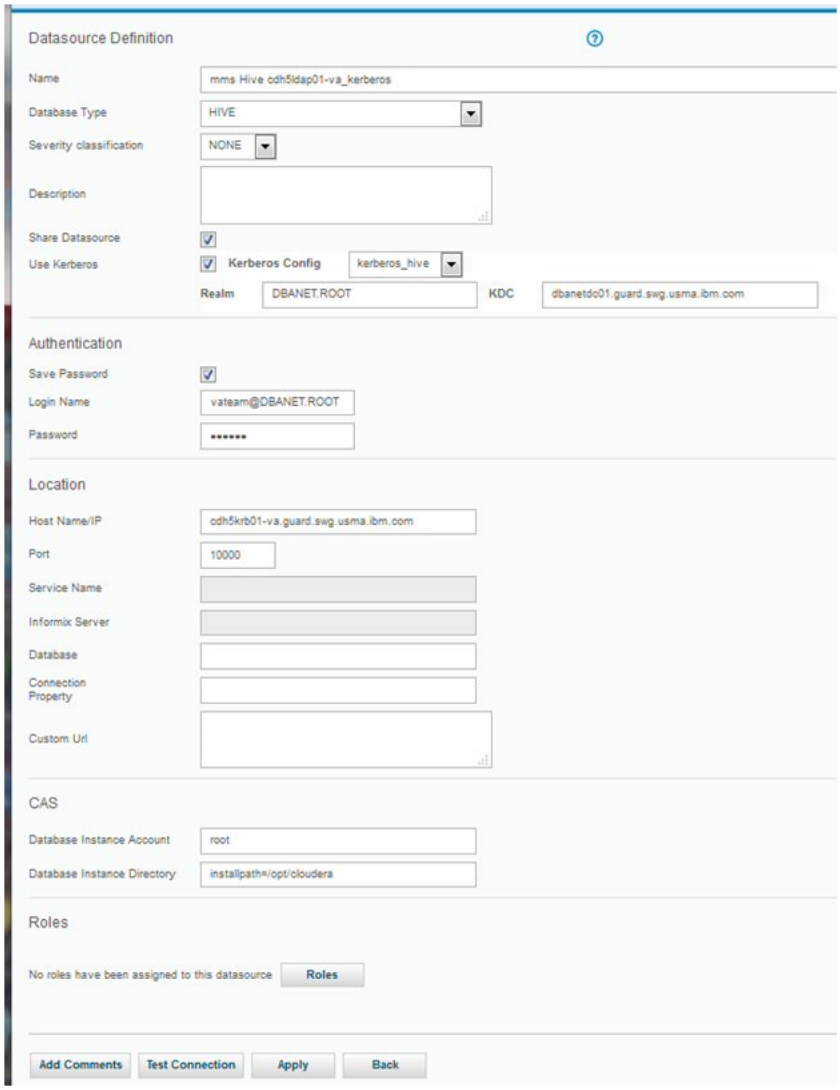
1. ディレクトリーは、Cloudera Manager のインストール・パスとして定義する必要があります。例: installpath=/opt/cloudera
2. HDFS が Kerberos に対して有効になっている場合、データ・ソース・ユーザー名とパスワードは有効な Kerberos ユーザー ID とパスワードでなければなりません。CAS スクリプトは、Kerberos チケットの取得にそれを使用します。
3. アカウントは root でなければなりません。CAS を必要とする特定のパラメーター・テストの場合、Cloudera エージェント・プロセス・ディレクトリー (/var/run/cloudera-scm-agent/process/) でリアルタイム構成にアクセスするために CAS ユーザーが root であることが重要です。

注: Guardium は構成データに何らかの変更や修正を行うことはありません。

Hive の場合

特権テストでは、データ・ソース・アカウントは Sentry Admin グループのメンバーである必要があります。Sentry Admin グループを確認する手順については、Hive gdmmonitor スクリプトを参照してください。

Hive データ・ソースのセットアップ時には、データ・ソースが Hive server2 を指しているときにのみ JDBC テスト接続を実行できます。他のすべての Hive データ・ソースについては、Cloudera サービスがインストールされているノード名を使用して、この特定のデータ・ソースのコピーを作成できます。 Hive server2 データ・ソースと同様に、コピーされたデータ・ソースにも有効なユーザー名およびパスワードがあることを確認してください。これらのデータ・ソースについては、データ・ソース・テスト接続を実行できません。しかし、Guardium は、Kerberos が有効な場合の CAS を使用した Kerberos 接続の実行は、データ・ソースからのユーザー名とパスワードの正確性に基づいて行います。



脆弱性評価のテスト

Hive 特権テストには、Sentry サービスがインストールおよび構成されている必要があります。Sentry がない場合、セキュリティーはありません。だれでも Hive に接続して、データにアクセスできます。

HDFS パラメーターの脆弱性評価 CAS テストは、Cloudera エージェント・プロセス・ディレクトリー (/var/run/cloudera-scm-agent/process/) の構成ファイルから行われます。これらのプロセス・ディレクトリー内のフォルダー名は、Cloudera エージェント・サービスが開始されるたびに変更されます。

一部の HDFS パラメーター CAS テストでは、データ・ソース・システムが特定のノード構成 (例えば、NameNode や DataNode など) である必要があります。また、一部の CAS テストでは、データ・ソース・システムに Yarn、Mapreduce、または Hive Server がインストールされている必要があります。 ご使用のデータ・ソース・システム構成に基づいて、評価のためのテストを慎重に選択してください。テストの要件が満たされない場合は、テストはエラーになり、これらのテストを適切な Cloudera サービスで実行するように推奨されます。要件はテストの説明にも記載されています。

Hive データ・ソースを作成する場合、各 Cloudera サービスに対して 1 つのデータ・ソース (NameNode、DataNode、HiveServer2、Hive メタストア、Yarn NodeManager、および Yarn ResourceManager) があることが推奨されます。

クラスター内のノードの数に関係なく、これらのすべてのサービスに対応する Guardium Hive データ・ソースがある場合は、ご使用の環境を適切にセットアップして脆弱性評価を実行してください。

例

親トピック: Guardium 脆弱性評価の紹介

脆弱性評価のタイプ

Guardium® には、データベース構成のパラメーター、特権、およびその他の脆弱性などを検査するための、2000 を超える事前定義テストがあります。一部のテストを特定の要件に対応するためにカスタマイズすることもできます。

脆弱性評価 - テスト・タイプ

1 つの脆弱性評価に、事前定義またはカスタムの以下のタイプのテストが 1 つ以上含まれる場合があります。

事前定義テストは、データベース環境で発生する可能性のある、共通する脆弱性の問題を示す目的で設計されています。 ただし、データベース・アプリケーションの性質が非常に多様であることが原因で、これらのテストの中には、特定のデータベースに適合しても、別のデータベース (同一組織内にあっても) にはまったく適合しないものがあります。

そのため、Guardium では、事前定義テストの一部を組織に固有の要件に合わせてカスタマイズできるようにしています。さらに、評価に常に業界の最新のベスト・プラクティスを反映させ、新たに発見された脆弱性から保護するため、 Guardium は、データベース保護サブスクリプション (DPS) サービスの一環として新しい評価テストを配布し、四半期ごとに更新します。詳しくは、「Guardium 管理ガイド」を参照してください。

事前定義テストには、「特権」、「認証」、「構成」、「バージョン」、「CVE」、「セキュリティー APAR」、「照会ベース」、および「CAS ベース」のテストが含まれます。照会ベース・テストおよび CAS ベース・テストをカスタマイズすることもできます。

テストのカテゴリー

セキュリティー関連の脆弱性に関するいくつかの高水準テストの現行のカテゴリーには、以下のようなものがあります。

- 特権
  - オブジェクト作成/使用権限
  - DBA および個々のユーザーに付与される特権
  - システム・レベル権限
- 認証
  - ユーザー・アカウントの使用
  - リモート・ログインの使用
  - パスワード規則
- 構成
  - データベース固有のパラメーター設定
  - システム・レベルのパラメーター設定
- バージョン
  - データベース・バージョン
  - データベース・パッチ・レベル
- その他
  - インストールしたサンプル・データベース
  - ファイル所有権
  - ファイルの許可

CVE テスト

Guardium は、MITRE Corporation の提供する Common Vulnerabilities and Exposures (CVE) を常にモニターしています。 関連するデータベース関連の脆弱性を調べるために、これらのテストが追加されます。

セキュリティー APAR テスト

Guardium では、関連するデータベース関連の脆弱性をモニターするために事前定義されたセキュリティー・プログラム診断依頼書 (APAR) テストが追加されています。

照会ベース・テスト

照会ベース・テストは、独自のテスト基準を定義することで迅速かつ容易に作成できる事前定義テストまたはカスタム・テストです。カスタムの照会ベース・テストの作成に関する追加情報については、照会ベース・テストの定義を参照してください。

CAS ベース・テスト

CAS ベース・テストは、OS スクリプト・コマンド・タイプの CAS テンプレート項目に基づく事前定義またはカスタム・テストで、CAS 収集データを使用します。

ユーザーは、テンプレート項目を指定して CAS 結果の内容に対するテストを実行できます。OS スクリプト・タイプの CAS テンプレートの作成については、新しいテンプレート・セットの作成を参照してください。

Guardium は、CAS ベース・テストの作成に使用できる OS スクリプト・タイプの CAS テンプレート項目で事前構成されています。これらのテストは名前に *Assessment* という語が含まれており、「CAS テンプレート・セット定義」パネルからアクセス可能です。例えば、Unix/Oracle 用評価セットの名前は、Guardium Unix/Oracle Assessment になります。さらに、ファイル・アクセス権が関係する追加テンプレートも、アクセス権および所有権の検査に使用されます。 これらのテンプレート・セットを表示し、これらの OS スクリプト・タイプの項目を参照する方法については、『テンプレート・セットの変更』を参照してください。

CAS ベース・テストの作成時または変更時に事前定義テストとカスタム・テストの両方を選択できます。『CAS ベース・テストの定義』で追加情報を参照してください。

- 照会ベース・テストの定義
  SQL ステートメントを実行する照会に基づいてテストを作成します。
- CAS ベース・テストの定義
  脆弱性評価では、CAS メカニズムを使用して、データベース・サーバーに対して OS レベルのテストを実行し、脆弱性を識別します。

親トピック: 評価および強化

照会ベース・テストの定義

SQL ステートメントを実行する照会に基づいてテストを作成します。

このタスクについて

以下のいずれかの方法で、新規照会ベース・テストを作成できます。

新規
: 作成を最初から開始してすべてのフィールドを定義します。

コピー
: 既存の照会ベース・テストをコピーします。

変更
: 既存の照会ベース・テストに変更を加えます。

手順

1. 「強化」 > 「脆弱性評価」 > 「アセスメント・ビルダー」をクリックし、「アセスメント・ビルダー」を開きます。
2. 「ユーザー定義テスト」から、「照会ベース・テスト」をクリックします。
3. 「新規」、「コピー」、「変更」のいずれかをクリックし、「照会ベース・テスト・ビルダー」を開きます。
4. 固有の「テスト名」を入力します。
5. 「データベース・タイプ」を選択します。
6. 「カテゴリー」を選択します。
7. 「重大度」を選択します。
8. オプション: テストの「簡略記述」を入力します。
9. オプション: テストの「外部参照」を入力します。
10. テストに合格したときに表示される「合格の結果テキスト」を入力します。
11. テストに不合格だったときに表示される「不合格の結果テキスト」を入力します。
12. テストで実行される「SQL ステートメント」を入力します。

    SQL ステートメント内でグループ・メンバーを追加および参照するには、以下の規則に従います。

    例:

    グループ MyUsersGroup に定義されたユーザーのグループを参照し、それを実際に使用されるグループ・メンバーで置き換えるには、次のようにします。

    ```
    Select ... from DBA_GRANTS where ... AND USER in (~~G~MyUsersGroup~~) and ...
    ```

    この結果、以下のような SQL ステートメントが得られます。ここで、U1、U2 などは MyUsersGroup グループのメンバーです。

    ```
    Select ... from DBA_GRANTS where ... AND USER in ('U1','U2','U3',...) and ...
    ```

    グループにメンバーが存在しない場合、データベースはエラーを返します。 この場合、参照は、次のような一組の引用符に置き換えられます。

    ```
    Select ... from DBA_GRANTS where ... AND USER in ('') and ...
    ```

    以下の規則に従って、(特定のグループ・タイプの) 特定の別名への参照を実際の別名に置き換えます。

    例:

    Select ... from USER_OBJECTS where ... AND OBJECT_TYPE = '~~A~GroupType~TYPE~~'

    グループ・タイプ GroupType の TYPE に別名がある場合、文字列がそれに置換されて、結果として以下のような SQL が得られます。

    Select ... from USER_OBJECTS where ... AND OBJECT_TYPE = 'TYPE'

    ここで、TYPE は実際の別名です。

13. オプション: 「詳細な SQL ステートメント」を入力します。この SQL ステートメントは、文字列のリストを取得して、「詳細の接頭部」+ 文字列リストから成る詳細文字列を生成するものです。「詳細の接頭部」の例を参照してください。
    注: 生成される詳細は、照会ベース・テストが失敗した時しか表示されません。これにより、ユーザーは、テストの失敗の原因になった情報を取得する SQL ステートメントを入力し、失敗の原因の特定に役立てることができます。
    注: 詳細文字列は、アセスメント・テスト名をクリックすることにより「セキュリティー・アセスメント結果」で確認できます。また、「テスト結果」エンティティーの「結果の詳細」属性により照会することもできます。
14. オプション: 「テスト前検査 SQL ステートメント」を入力します。このステートメントは、テストの実行前に実行されます。 このステートメントで 0 が返されると、テストは実行されません。 このテストで 1 またはエラーが返されると、テストが実行されます。
15. オプション: 「テスト前失敗メッセージ」を入力します。SQL ステートメントで 0 が返されたためにテストが実行されない場合は、このメッセージがアセスメント結果に挿入されます。
16. オプション: 「ループ・データベース」に、テストがループする必要があるデータベースのリストを入力します。 テストでは、指定したすべてのデータベースから返された結果の和集合または合計が返されます。 この関数は、テストで整数値が返され、かつ、データベース・タイプが Informix、SQL Server、Sybase SE、PostgreSQL および MySQL である場合にのみ使用できます。 ループは、「DB ループ・フラグ」ボックスにチェックマークが付いている場合に実行されます。 テストの実行時に、指定された 1 つ以上のデータベースが使用できないことがあります。 このような場合、テストでは、そのデータベースがスキップされて続行されるか、テストが停止して失敗メッセージが発行されます。これは、「エラーの場合はスキップ」ボックスにチェックマークが付いているかどうかによって異なります。
17. オプション: 詳細文字列の先頭に現れる「詳細の接頭部」を入力します。

    「詳細な SQL ステートメント」および「詳細の接頭部」の例:
    特定の権限を付与されたオブジェクトを検査するテスト。
    詳細の接頭部: "Objects found with certain GRANT:"
    詳細な SQL ステートメント: SELECT object FROM....--returning 4 records:

```
      Obj1
      Obj2
      Obj3
      Obj4
==> Details: Objects found with certain GRANT: Obj1, Obj2, Obj3, Obj4
```

18. オプション: SQL ステートメントに入力したテキストが、「比較値」との比較で使用される内部 Guardium® 変数にバインドされる値を返すプロシージャー型コード・ブロックである場合は、「出力変数のバインド」チェック・ボックスにチェック・マークを付けます。

    例 (Oracle):

```
declare
    retval integer := 0;
    strval varchar2(255) := '';
    nver number;
    sver varchar2(255) := '';
begin
    select VERSION
    into sver
    from V$INSTANCE;
    nver := to_number(substr(sver,1,(instr(sver,'.',1,2) - 1)));
    if nver >= 11.1 then
        select VALUE
        into strval
        from V$PARAMETER
        where NAME = 'sec_case_sensitive_logon';
    end if;
    if (nver < 11.1 or strval = 'TRUE') then
        retval := 0;
    else
        retval := 1;
    end if;
    ? := retval;
end;
```

19. SQL ステートメントから返される「戻りの型」を選択します。
20. 条件に使用する「演算子」を選択します。
21. 「比較値」を入力します。この値は、比較演算子を使用して SQL ステートメントからの戻り値と比較するために使用されます。この比較によって、テストの合格/不合格が判定されます。 さらに、「RE」(regex) をクリックして、比較値を正規表現で定義することもできます。
22. 以下のいずれかを実行します。
    - 「戻る」をクリックし、変更をキャンセルして前の画面に戻ります。
    - 「適用」をクリックして、照会ベース・テストを保存します。

タスクの結果

この新規作成された照会ベース・テストをアセスメントに追加できます。

次のタスク

親トピック: 脆弱性評価のタイプ

CAS ベース・テストの定義

脆弱性評価では、CAS メカニズムを使用して、データベース・サーバーに対して OS レベルのテストを実行し、脆弱性を識別します。

始める前に

このタスクについて

既存の CAS ベース・テストに変更を加えるか、作成を最初から開始してすべてのフィールドを定義することにより、新規 CAS ベース・テストを作成することができます。

手順

1. 「強化」 > 「脆弱性評価」 > 「アセスメント・ビルダー」をクリックし、「アセスメント・ビルダー」を開きます。
2. 「ユーザー定義テスト」から、「CAS ベース・テスト」をクリックして「CAS ベース・テスト・ファインダー」パネルを開きます。
3. 「新規」または「変更」をクリックして、新規テストを作成します。
4. 固有の「テスト名」を入力します。
5. 「データベース・タイプ」メニューからデータベースを選択します。
6. 「カテゴリー」メニューからカテゴリーを選択します。
7. 「重大度」メニューからカテゴリーを選択します。
8. オプション: テストの「簡略記述」を入力します。
9. オプション: テストの「外部参照」を入力します。
10. テストに合格したときに表示される「合格の結果テキスト」を入力します。
11. テストに不合格だったときに表示される「不合格の結果テキスト」を入力します。
12. テストに合格したときに表示される「合格の推奨テキスト」を入力します。
13. テストに不合格だったときに表示される「不合格の推奨テキスト」を入力します。不合格の推奨テキスト: クロスサイト・ハッキングを防止するため、「不合格の推奨テキスト」テキスト・ボックス内で expression、function、javascript、script、alert、eval、<img、ContentType のいずれかの名前が使用されている場合、その名前は書き直されます。
14. 「CAS テンプレート」メニューから、使用するテンプレートを選択します。
15. 「演算子」メニューから、使用する演算子を選択します。

16. 「検索文字列」に、CAS テンプレートから返される内容を比較するために演算子とともに使用する検索文字列を入力します。この比較によって、このテストの合格/不合格が判定されます。「RE」アイコンをクリックして、検索文字列に対して正規表現を定義することも可能です。
17. オプション: 検索文字列との一致があればテストを不合格にする場合は、「一致した場合に不合格」チェック・ボックスにチェック・マークを付けます。
18. 「適用」をクリックし、CAS ベース・テストを保存します。

タスクの結果

この新規作成された CAS ベース・テストをアセスメントに追加できます。
親トピック: 脆弱性評価のタイプ

評価

アセスメントとは、データベース・インフラストラクチャーの脆弱性をスキャンし、リアルタイム測定と履歴測定によるデータベースおよびデータ・セキュリティーの正常性評価を行う一連のテストを指します。

- アセスメントの作成
  アセスメントの作成、既存のアセスメントの変更またはコピーを行います。
- セキュリティー・アセスメントの作成方法
  選択したデータ・ソースに対してセキュリティー・アセスメントを実行することで、事前に脆弱性を特定および処置し、構成を改善し、インフラストラクチャーを強化します。
- アセスメントの実行
  アセスメントの結果を得るには、アセスメントを作成後に実行する必要があります。
- アセスメント結果の表示
  アセスメント結果の表示中に、さまざまなアクションを実行できます。
- 脆弱性診断テストの例外の作成
  セキュリティー・アセスメントからグループの特定メンバーを除外するには、テスト例外を使用します。例外グループに対してセキュリティー・アセスメントを実行すると、グループの特定メンバーがアセスメント結果に影響を及ぼしているかどうかを確認できます。これは、グループ設定を変更したくない場合や、変更が許可されていない場合に便利です。
- データベース・バージョンおよびパッチ・レベルの変更
  データベース・バージョンおよびパッチ・レベルを手動で追加して、不合格の脆弱性評価をオーバーライドします。
- VA サマリー
  以下の表に、 VA サマリー表に表示される各テストの情報およびデータベース・キーをリストします。ユニーク ID ごとのテスト結果、不合格になってからの累積経過日数、最初に不合格になった日付/ 最後に不合格になった日付、最後に合格した日付、および最後にスキャンされた日付などが示されます。この情報はトラッキングされ、ユーザーはこの情報に基づいてレポートを作成できます。

親トピック: 評価および強化

アセスメントの作成

アセスメントの作成、既存のアセスメントの変更またはコピーを行います。

始める前に

「強化」 > 「脆弱性評価」 > 「アセスメント・ビルダー」をクリックし、「アセスメント・ビルダー」を開きます。

このタスクについて

手順

1. 完全に新規のアセスメントを作成するには、「セキュリティー・アセスメント・ファインダー」パネルで「新規」をクリックします。 既存のアセスメントで作業をするには、「コピー」または「変更」をクリックします。 これらのいずれのボタンをクリックしても、「セキュリティー・アセスメント・ビルダー」パネルが開きます。 完全に新規のアセスメントを作成する場合は、以下のステップをすべて実行します。既存のアセスメントをコピーまたは変更する場合は、新規の記述を入力した後、変更したいフィールドのみを変更します。
2. そのアセスメントに固有の「記述」を入力します。
3. 「データ・ソースの追加」をクリックし、必要な情報を入力して「追加」をクリックすることにより、データ・ソースを追加します。
4. 「テストの構成」をクリックし、アセスメントにテストを追加します。
   a. 「追加できるテスト」ペインから、以前に追加したデータ・ソースの該当するタブを選択します。
   b. 必要なテストを選択し、「選択の追加」をクリックしてそれらをアセスメントに追加します。追加した選択内容は、「アセスメント・テスト選択」ペインに表示されます。
   c. 「アセスメント・テスト選択」を使用して、アセスメントのテストを管理します。選択したテストを削除したり、任意のテストに対して「このテストのチューニングを調整」をクリックし、そのテストのパラメーターをカスタマイズしたりすることができます。
5. アセスメントにロールを追加します。
   注: アセスメントには、そのベースのデータ・ソースにロールを割り当てるまでは、ロールを割り当てることができません。
6. 「適用」をクリックして、アセスメントを保存します。

「CAS サポート」をクリックし、アセスメントに関する適切なデータを指定します。

任意のアセスメントに対して「コメントの追加」を使用して、アセスメントに対して加えた変更の内容や理由を文書化および記録することもできます。

タスクの結果

これで新規のアセスメントを実行する準備ができました。
親トピック: 評価

セキュリティー・アセスメントの作成方法

選択したデータ・ソースに対してセキュリティー・アセスメントを実行することで、事前に脆弱性を特定および処置し、構成を改善し、インフラストラクチャーを強化します。

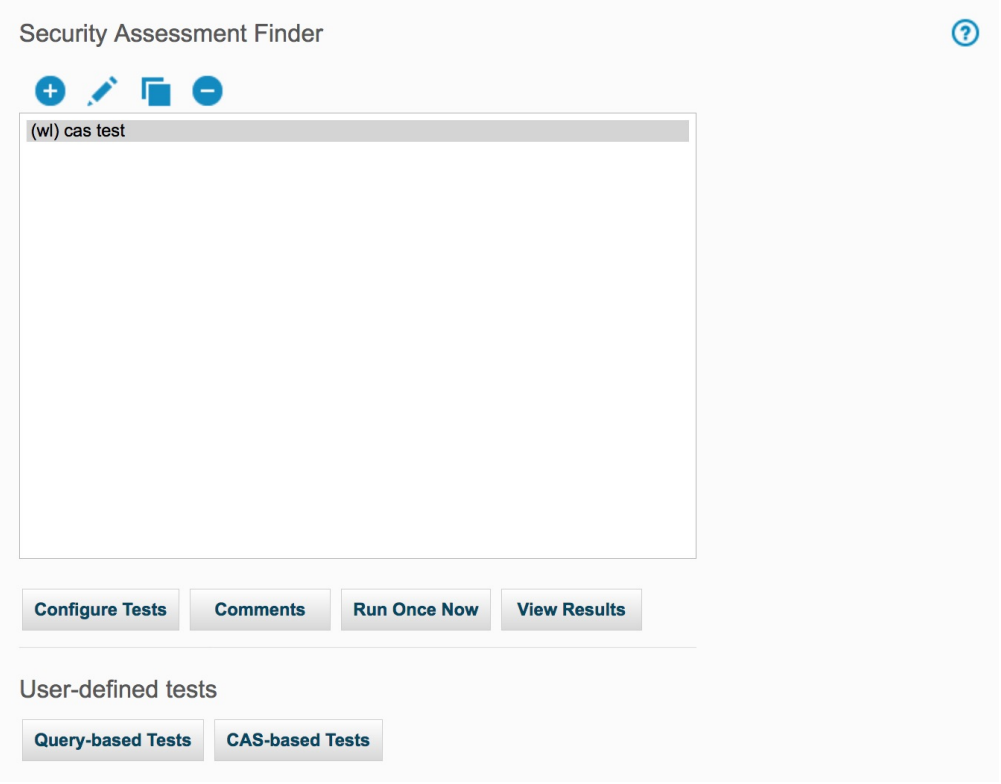
このタスクについて

セキュリティー・アセスメントを作成するための基本的なステップは、以下のとおりです。

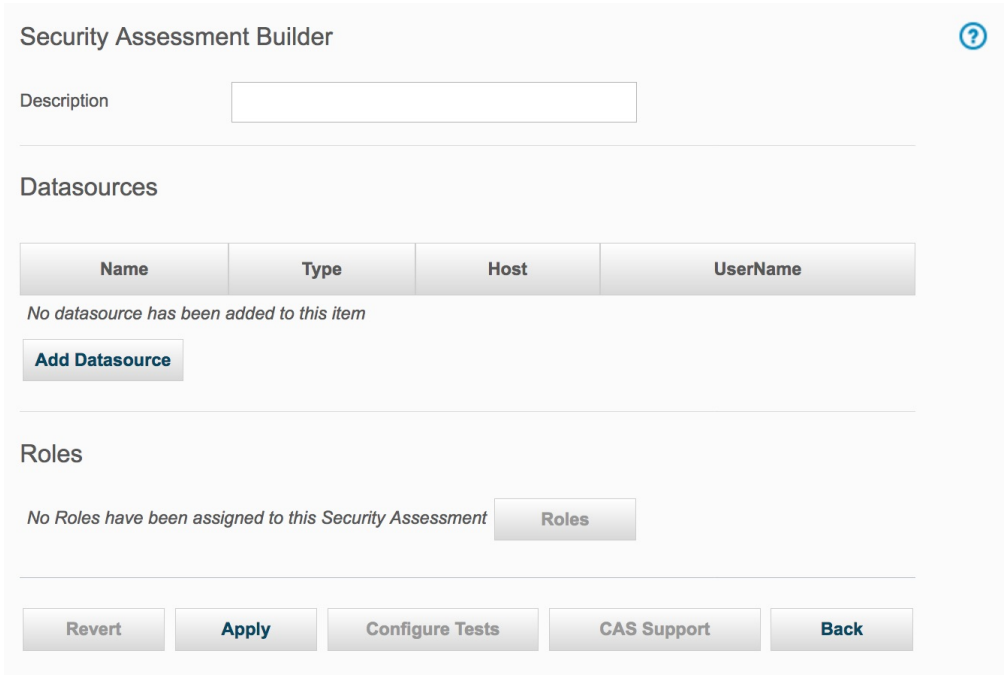
1. アセスメントの作成
2. アセスメントへのデータ・ソースの追加
3. アセスメントへのテストの追加

手順

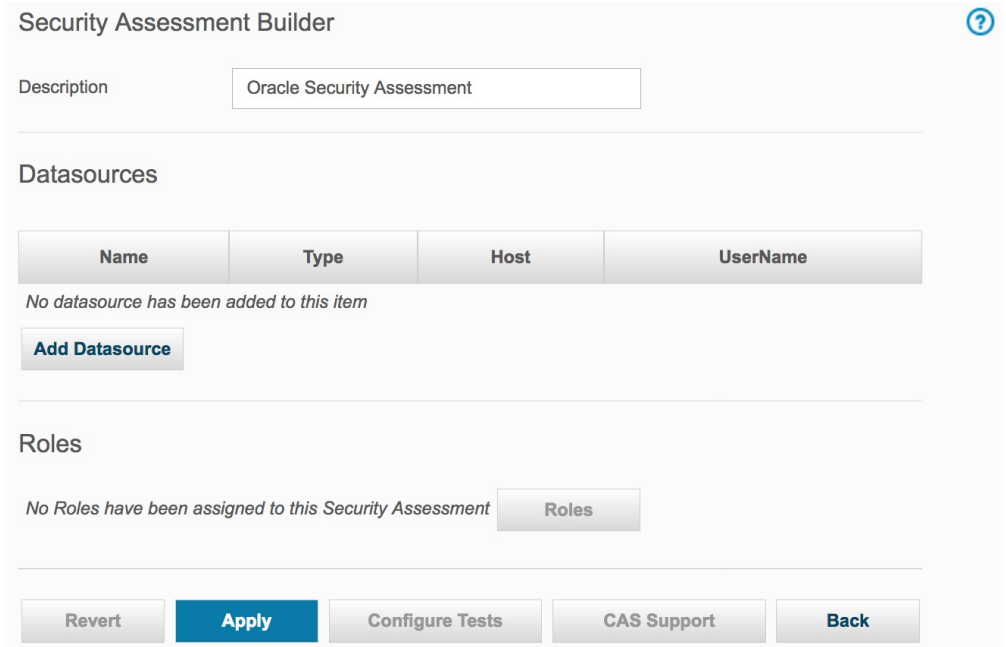
1. 「アセスメント・ビルダー」を開いて、アセスメントを作成または変更します。「強化」 > 「脆弱性評価」 > 「アセスメント・ビルダー」をクリックし、「アセスメント・ビルダー」を開きます。



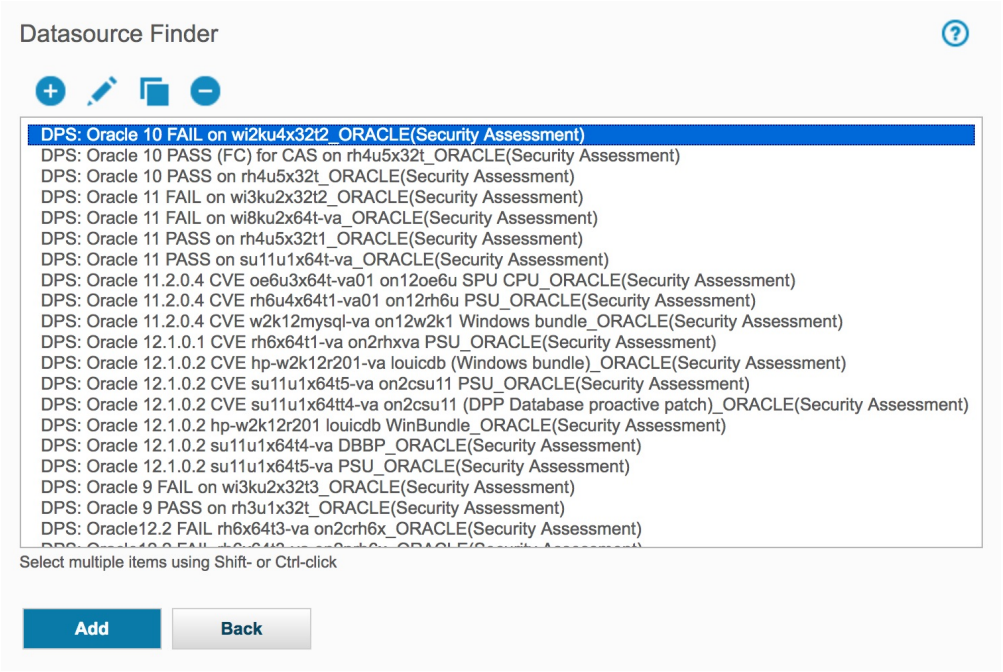
2. 「新規」をクリックして、新規のセキュリティー・アセスメントを作成します。



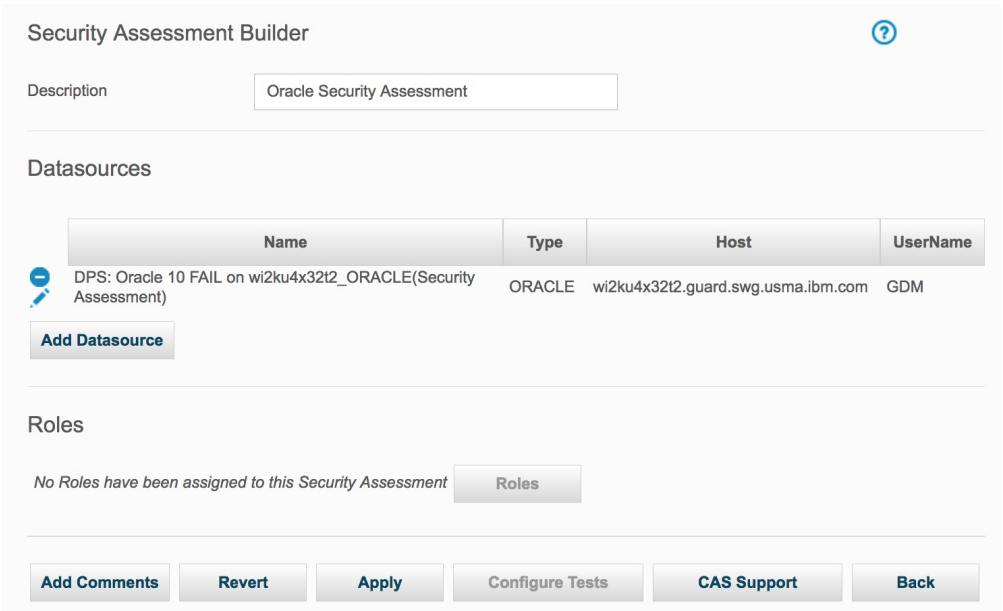
3. 「記述」にアセスメントの固有の名前を入力し、「適用」をクリックしてアセスメントを保存します。



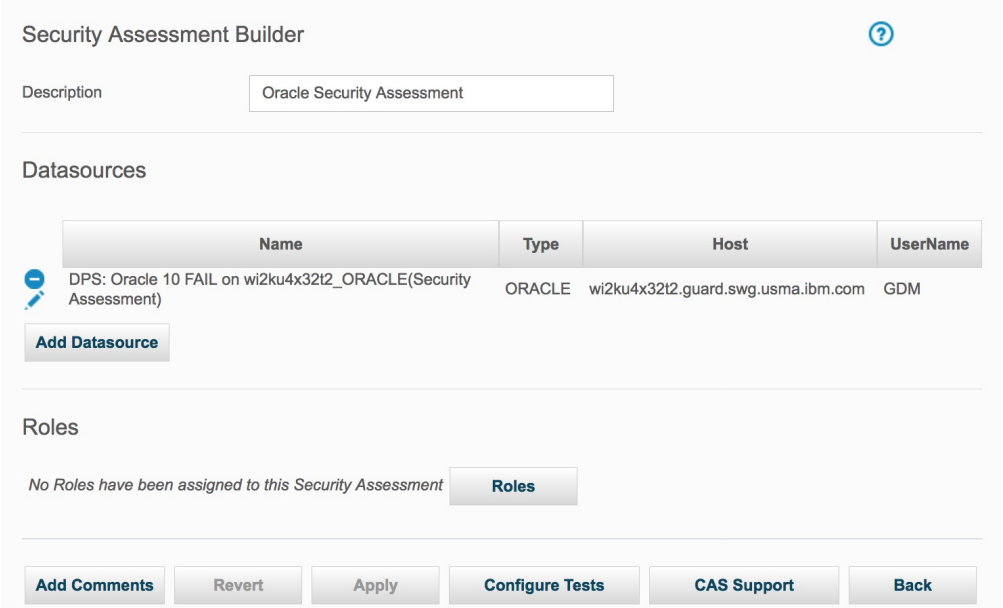
4. 「データ・ソースの追加」をクリックし、アセスメントにデータ・ソースを追加します。「データ・ソース・ファインダー」からデータ・ソースを選択し、「追加」をクリックします。新規データ・ソースを追加するには、 をクリックし、「データベース定義」ウィンドウで情報を入力して、「適用」をクリックします。 詳しくは、『データ・ソース』を参照してください。



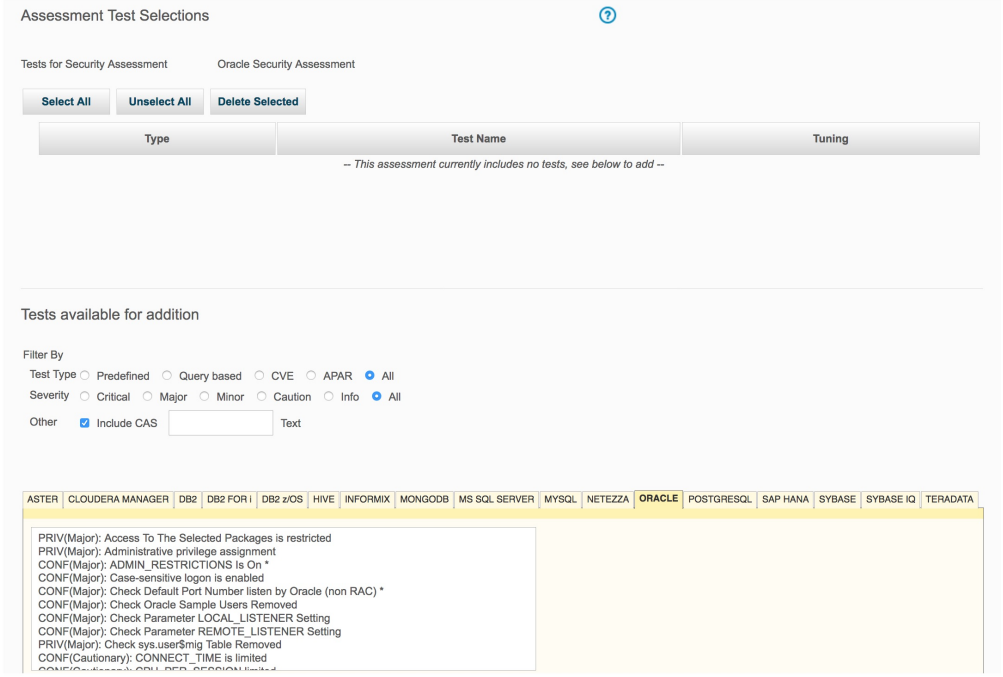
「追加」ボタンをクリックすると、セキュリティー・アセスメント・ビルダーの「データ・ソース」セクションにデータ・ソースが表示されます。

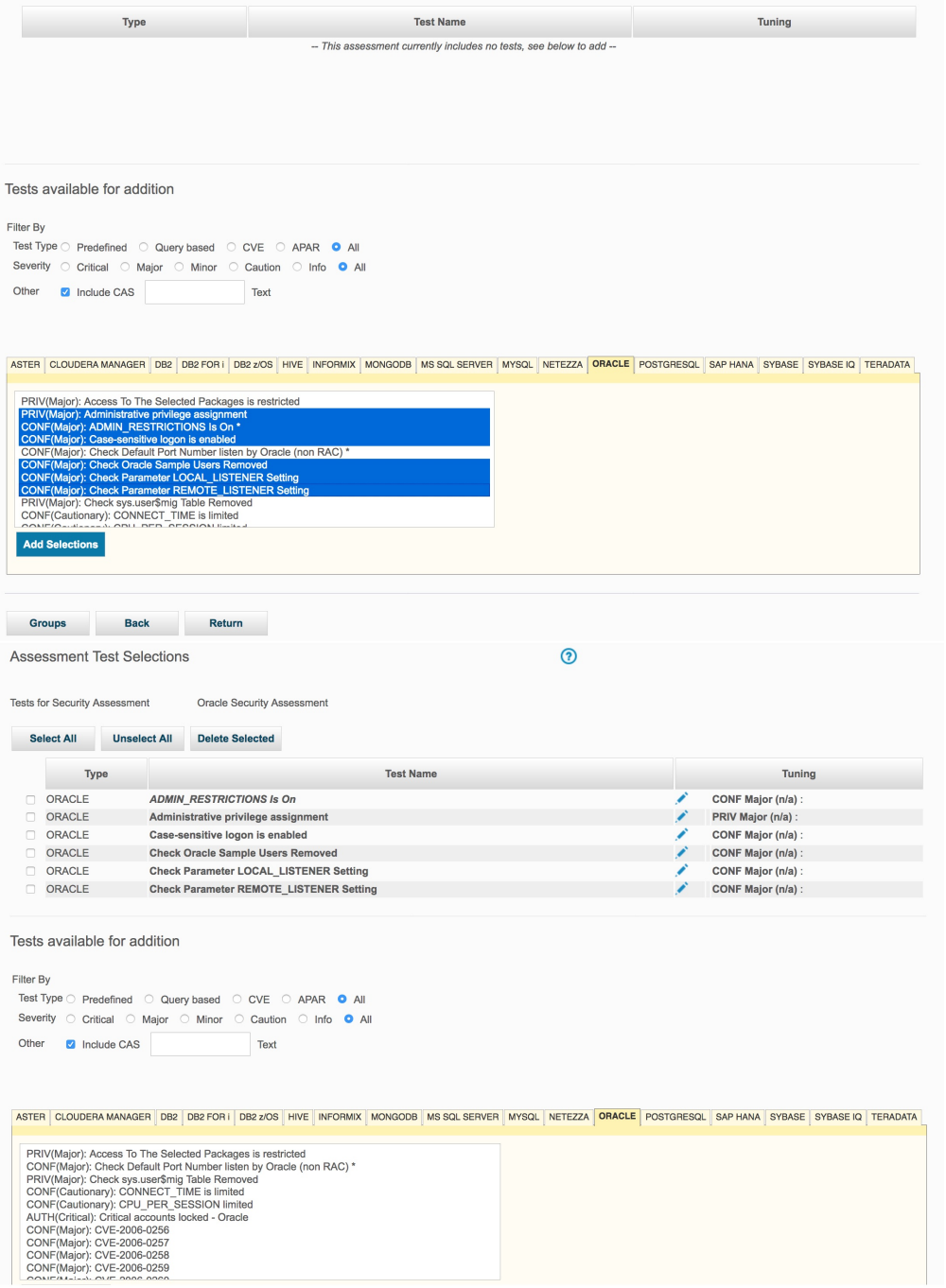


5. 「適用」をクリックして、アセスメントを保存します。



6. 「テストの構成」をクリックして、評価にテストを追加します。「追加できるテスト」パネルで、作成した適切なデータ・ソースを示すタブをクリックし、アセスメントに追加するテストを選択して、「選択の追加」をクリックします。追加するテストをフィルタリングするには、ラジオ・ボタンを使用します。詳しくは、『事前定義テスト』、『照会ベース・テスト』、『CVE テスト』、または『APAR テスト (APAR Tests)』を参照してください。





7. 「戻る」をクリックしてセキュリティー・アセスメント・ビルダーに戻り、「ロール」をクリックして、アセスメントにロールを追加します。
   注: アセスメントには、そのベースのデータ・ソースにロールを割り当てるまでは、ロールを割り当てることができません。
8. 「適用」をクリックしてアセスメントを保存します。これで、選択したデータ・ソースに対して、このアセスメントを実行することができます。

親トピック: 評価

アセスメントの実行

アセスメントの結果を得るには、アセスメントを作成後に実行する必要があります。

アセスメントは、シリアル・モードまたはパラレル・モードで実行できます。複数のアセスメントの実行をスケジュールに入れる場合、Guardium ジョブ・キュー・レポートでキューを表示できます。アセスメントの結果について詳しくは、『アセスメント結果の表示』を参照してください。

オプションで、アセスメント定義の実行の自動化プロセスを定義し、スケジュールに入れることができます。「監査プロセス・ファインダー」パネルから操作を開始して、監査プロセス・スケジュールの作成や変更を行います。 アセスメントを自動的に実行するスケジュールを作成するには、「監査プロセス・ファインダー」パネルに移動します。 監査プロセスの定義について詳しくは、『コンプライアンス・ワークフロー自動化』を参照してください。

マルチスレッド・アセスメント

Guardium では、CPU のパフォーマンスと使用状況を最適化するために、複数の脆弱性評価を並行して実行できます。並行して実行できるスレッドの数は、マシン内の CPU コアの数に 2 を乗算することで得られます。

一例として、4 個の CPU コアがある場合は、定義して同時に実行できるプロセスの最大数は 8 です。CPU コアの数に関係なく、上限は 100 です。

並行性の制限を取得または定義するには、GuardAPI 分類関数を参照してください。

親トピック: 評価

アセスメント結果の表示

アセスメント結果の表示中に、さまざまなアクションを実行できます。

アセスメント結果の表示

「照会 - レポート・ビルダー」でアセスメントの結果を表示します。「調査」 > 「照会 - レポート・ビルダー」をクリックして「照会 - レポート・ビルダー」を開き、フィルターを使用して必要なレポートを見つけます。

アセスメント結果の解釈

アセスメントでは、複数のレポートに基づいて複数のテストを評価します。結果全体は、「セキュリティー・アセスメントの結果」というタイトルが付いた別個のブラウザー・ウィンドウに表示されます。そのウィンドウには以下のセクションがあります。

アセスメント ID

アセスメント結果には、以下の情報が示されます。

- アセスメント名
- アセスメントを実行した日時
- アセスメントの期間
- クライアントおよびサーバーの IP アドレスまたはサブネット

アセスメントの選択

ドロップダウン・メニューを使用して、アセスメントの過去の結果を選択して表示します。デフォルトでは最新の結果が表示されます。

アセスメント結果履歴

「アセスメント結果履歴」にはある期間にわたるテストの合格率が表示されます。テストの合格率をさらに改善するための推奨事項が「アセスメント・テスト結果」セクションの下に表示されます。

ログの表示

これをクリックすると、新しいウィンドウに、アセスメント・テストのランタイム実行を示す「実行ログ」が表示されます。イベントとメッセージのタイム・スタンプは、特定のテストが不合格になった原因と考えられる問題をデバッグするときに役立ちます。

結果のサマリー

表形式のグラフにより、このアセスメントで実行されたすべてのテストの要約が示されます。x 軸はテストの重大度 (クリティカル、メジャー、マイナー、注意、または情報) を表します。y 軸はテストのタイプ (特権、認証、構成、バージョン、またはその他) を表します。 グリッド内に、テストの実行試行時に合格、不合格、あるいはエラーとなったテストの回数がそれぞれ示されます。 これらの数値は、「アセスメント・テスト結果」セクションの下に示されるアセスメント・テストの詳細に直接関連しています。

現在適用されているフィルタリング

フィルタリングを現在の適用内容から変更したい場合は、以下の 2 つのオプションを使用し、必要に応じて結果をフィルターに掛けます。

フィルタリングのリセット -「フィルター/ソート制御」オプションで選択されたフィルタリング・オプションをすべて削除します。

フィルター/ソート制御 - これを使用して、レポートのフィルター/ ソート・オプションを開きます。オプションを使用すると、「重大度」、データ・ソース重大度分類 (「DS 重大度分類」)、「スコア」(合格、不合格、またはエラー)、および「テスト・タイプ」(監視/データベース・タイプ) ごとにフィルター操作ができます。ソート・オプションを使用すると、重大度、スコア、およびデータ・ソースを組み合わせたソートを実行できます。選択したフィルター/ソート・オプションを有効にするには、「適用」をクリックします。

アセスメント・テスト結果

「アセスメント・テスト結果」セクションでは、実行したテストの詳細な説明、ターゲット・データ・ソースとデータ・ソース重大度分類に関する情報、およびテストの合格/不合格状況、重大度、外部参照、および現在状況の理由が示されます。各テスト名はクリックできるようになっていて、その特定のテストについての関連情報以外のすべての情報をレポートからフィルター除去することができます。 「理由」フィールドには吹き出しヘルプ機能があり、不合格またはエラーになったテストに対する改善策に役立つ推奨事項が表示されます。

アセスメント結果には、以下の各カテゴリー内のテスト数と合格したテスト数のカウントが含まれます。

- CIS テスト
- CVE テスト
- STIG テスト

これらの値は、アセスメント結果ビューアーに表示され、VA 結果ドメインの一部としてレポートの作成に使用できます。

データ・ソース詳細

「データ・ソース詳細」セクションを展開すると、このアセスメントで参照されているすべてのデータ・ソースが表示され、それと共にデータ・ソース固有の環境情報が示されます。

CVE および CVSS 情報

アセスメント・テスト結果ビューアーには、CVE レコードおよび CVSS 情報が表示されます。

参照リンクはクリックできます (新しいウィンドウが開きます)。対応するレコードが結果に含まれていない場合は、いずれのセクションも表示されません。

重要な CVSS フィールドは以下のとおりです。

- CVSS スコア
- アクセスの複雑性
- 可用性への影響
- 機密性への影響
- 保全性への影響
- 認証
- アクセス・ベンダー
- ソース
- 生成日時

不合格だったテストの処理

アセスメントで一部のテストに不合格状況が表示される場合、以下のいずれかのアクションを実行できます。

テストの例外を追加する
: このアクションを実行すると、一定期間、テストは必ず合格になります。 例えば、最新の使用可能なサービス更新が適用されていることを確認するテストで不合格となるサーバーのグループがあるとします。 週末のメンテナンス・ウィンドウまで、更新を適用することはできません。 それまでテストで不合格となり続けるのは望ましくありません。 結果パネル内の「不合格」という単語を右クリックすると、「テスト例外の追加」ポップアップ・メニューが表示されます。 例外の終了日時を指定し、オプションでコメントを指定します。 テストがこのアセスメントから実行されるか、あるいは別のアセスメントの一部として実行されるかに関係なく、例外の期限切れ前にテストが実行されるたびに、すべてのデータ・ソースでテストは合格となります。

不合格の要素を例外グループに追加する
: テストで不合格となった場合、テストの名前をクリックすると、さらに情報を表示できます。 新規パネルに、「詳細」というタイトルのエリアが含まれます。このヘッダーの下に、不合格となったテストの要素が表示されます。要素が表示された場合は、それらをこのテストの例外グループに追加できます。 これを行うには、ヘッダー「詳細:」をクリックして新規ダイアログを開きます。 このダイアログに不合格の要素が表示され、各要素に 1 つのチェック・ボックスがあります。 例外グループに追加する要素のチェック・ボックスにチェック・マークを付け、他のチェック・ボックスのチェック・マークを外します。 次に、グループを選択します。このテストにデフォルトの例外グループが定義されている場合は、それがダイアログに表示され、事前選択されています。 ドロップダウン・リストには、定義されている他のすべてのタイプ VA テスト例外のグループが表示されます。 リストからグループを選択するには、リストの横のラジオ・ボタンをクリックし、リストからグループを選択します。 「保存」をクリックして、選択項目を実装します。 残りの要素を別のグループに追加するには、再度「詳細」をクリックします。

PDF へのエクスポート、あるいは SCAP または AXIS XML へのエクスポート

「PDF のダウンロード」をクリックすると、アセスメント結果の PDF バージョンを生成できます。

「XML のダウンロード」ボタンを使用して、2 つのメニュー選択項目 (「SCAP xml としてダウンロード」と「AXIS xml としてダウンロード」) を開きます。 それらの選択項目のいずれかを選んで、表示されているアセスメント結果を表す XML ファイルをワークステーションにダウンロードします。 このファイルは、Security Content Automation Protocol (SCAP) XML または QRadar で使用される Apache EXtensible Interaction System (AXIS) XML 用にフォーマットされます。

親トピック: 評価

脆弱性診断テストの例外の作成

セキュリティー・アセスメントからグループの特定メンバーを除外するには、テスト例外を使用します。例外グループに対してセキュリティー・アセスメントを実行すると、グループの特定メンバーがアセスメント結果に影響を及ぼしているかどうかを確認できます。これは、グループ設定を変更したくない場合や、変更が許可されていない場合に便利です。

手順

1. 「設定」 > 「ツールとビュー」 > 「グループ・ビルダー」をクリックし、「グループ・ビルダー」を開きます。
2. 「グループ・タイプ」メニューから「脆弱性診断テストの例外」を選択し、事前定義された例外グループのリストを表示します。
3. 「既存グループの変更」メニューからグループを選択し、「変更」をクリックします。
4. 脆弱性評価テストから除外するグループ・メンバーを追加します。
5. 「強化」 > 「脆弱性評価」 > 「アセスメント・ビルダー」をクリックし、「アセスメント・ビルダー」を開きます。「セキュリティー・アセスメント・ファインダー」からアセスメントを選択し、「テストの構成」をクリックします。
6. 例外を追加したいテストを見つけ、「チューニング」列からそのテストの「このテストのチューニングを調整」ボタンをクリックします。
7. メニューから例外グループを選択し、「保存」をクリックします。アセスメントを再実行し、例外グループがテスト結果に影響を及ぼしているかどうかを確認します。
   注: デフォルトでは、Guardium には IBM iSeries プロファイル・ユーザー例外という例外グループが含まれています。このグループをコピーし、必要に応じて変更を加えることができます。

   すべてのデータベース・オブジェクト特権テストでは、Guardium グループからデフォルトのシステム・スキーマが除外されます。

親トピック: 評価

データベース・バージョンおよびパッチ・レベルの変更

データベース・バージョンおよびパッチ・レベルを手動で追加して、不合格の脆弱性評価をオーバーライドします。

このタスクについて

データベース・バージョンおよびパッチ・レベルが定義済みのレベルより低い場合は、セキュリティー・アセスメントが不合格となるように設計されています。これをオーバーライドするために、推奨されるパッチ・レベルおよびデータベース・バージョンをグループ・ビルダーに手動で追加できます。
注: Netezza ユーザーの場合、セキュリティー・アセスメントに合格するためには、データベース・バージョンおよびパッチ・レベルが定義済みのレベルと一致している必要があります。

手順

1. 「設定」 > 「ツールとビュー」 > 「グループ・ビルダー」にナビゲートして、「グループ・ビルダー」を開きます。
2. ご使用のデータベース Version+Patches を選択します。 例えば、「Oracle データベース Version+Patches」を選択します。
   ヒント: 表フィルターに `patch` と入力して、「グループ・ビルダー」をフィルタリングします。
3. 編集するために、 をクリックします。 「グループの編集」ウィンドウが開きます。
4. 「メンバー」タブを開き、 をクリックして、メンバーを追加します。
   a. 「DB のバージョン (DB Ver.)」フィールドおよび「パッチ (Patches)」フィールドを使用して、データベース・バージョンおよびパッチ・レベルを入力します。
   b. 「OK」をクリックして、グループ・メンバーを追加します。
   ヒント: 「DB のバージョン (DB Ver.)」フィールドおよび「パッチ (Patches)」フィールドの構文を判別するには、既存のレコードを使用してください。
5. 「保存」をクリックします。

親トピック: 評価

VA サマリー

以下の表に、 VA サマリー表に表示される各テストの情報およびデータベース・キーをリストします。ユニーク ID ごとのテスト結果、不合格になってからの累積経過日数、最初に不合格になった日付/ 最後に不合格になった日付、最後に合格した日付、および最後にスキャンされた日付などが示されます。この情報はトラッキングされ、ユーザーはこの情報に基づいてレポートを作成できます。

VA サマリー

キーには、3 つのオリジナル要素に加えて、データ・ソース名が含まれる場合があります。デフォルトは、ホスト、ポート、およびインスタンス名です。

クエリー・ビルダーで VA サマリー・トラッキングを使用して、 照会およびレポートを定義します。

この表はエクスポート/インポートできます。インポート・データは、 Guardium システムの既存データをオーバーライドします (キーごと)。

表 1. VA サマリー

表の列	タイプ	記述
VA_SUMMARY_ID	Int	自動増加 - 主キー
DATA_SOURCE_HASH	Varchar(40)	キーのハッシュ
DB_TYPE	Varchar	データベース・タイプ
SERVICE_NAME	Varchar	データベース・インスタンス名 (キーの一部である場合、そうでない場合は「N/A」)
DB_PORT	Varchar	データベース・ポート (キーの一部である場合、そうでない場合は「N/A」)
DB_HOST	Varchar	ホスト/IP (キーの一部である場合、そうでない場合は「N/A」)
TEST_ID	Int	テストの ID
FIRST_EXECUTION	DateTime	テストが最初に実行された時
LAST_EXECUTION	DateTime	テストが最後に実行された時
FIRST_FAIL	DateTime	この DB に関してテストが最初に不合格になった時
LAST_FAIL	DateTime	この DB に関してテストが最後に不合格になった時
FIRST_PASS	DateTime	この DB に関してテストが最初に合格した時
LAST_PASS	DateTime	この DB に関してテストが最後に合格した時
CURRENT_SCORE	varchar	合格 / 不合格 / エラー
CURRENT_SCORE_SINCE	Datetime	テストが現行の状況になった日付
CUMULATIVE_FAIL_AGE	Int	不合格になってからの累積経過日数
CUMULATIVE_PASS_AGE	Int	合格してからの累積経過日数

CLI コマンドは、store va_test_show_query および show va_test_show_query です。export va_summary は、この情報をエクスポートするときに使用します。

キーを変更または表示する GuardAPI コマンドは、 grdapi modify_va_summary_key および grdapi reset_va_summary_by_key です。 合格および不合格の両方に関する累積経過日数をリセットする GuardAPI コマンドは、 grdapi reset_va_summary_by_id です。この情報をエクスポートするときには grdapi export_va_summary を使用しま

す。

grdapi reset_va_summary_by_ke および grdapi modify_va_summary_key に、追加のパラメーター datasourceName が追加されました。

VA サマリーのエンティティーは追加の属性 Datasource Name を持ちます。Datasource Name にデータが設定されるのは、データソース名がキーの一部である場合のみです。

注: GrdAPI コマンド modify_va_summary_key の 4 つのパラメーター (useHost、usePort、useServiceName、useDatasourceName) のすべてに false を指定して呼び出すと、キーを空にすることができます。この場合、キーが空になると、VA サマリーの計算は無効になります (サマリー・データの計算、更新、保存は行われません)。
親トピック: 評価

必要とされるスキーマ変更

Guardium V9.1 では、IBM DB2 for z/OS 上での脆弱性評価テストで使用されるスキーマが変更されています。9.1 より前のリリースからアップグレードする場合、これらのテストを引き続き使用するためには、データベースを更新する必要があります。

このタスクについて

ご使用の Guardium システムをバージョン 10.x にアップグレードする場合は、データベース・サーバー上に新しいデータベース表を作成する必要があります。 これらの表により、新しいテスト・セットに対するサポートが追加されますが、新しいテストを使用するかどうかにかかわらず、これらの表を作成する必要があります。
前のリリースでは、次の表は gdmmonitor スキーマで作成され、データが設定されました。

- GDMMONITOR.OS_GROUP
- GDMMONITOR.OS_USER

これらの表は、次の CKADBVA スキーマの表に置き換えられます。

- CKADBVA.CKA_OS_GROUP
- CKADBVA.CKA_OS_USER

手順

1. Install Guardium 10.x
2. ご使用の Guardiumシステムの /var/log/guard/gdmmonitor_scripts ディレクトリーから、create_CKADBVA-schema_tables_zOS.sql を データベース・サーバーにコピーします。 データベース・サーバー上で、fileserver コマンドを実行して、このファイルを取得します。
3. スクリプトには、スクリプトの実行の前後に実行すべきステップを説明した指示が含まれています。これらの指示をよく読んで、スクリプトを実行します。
4. 新しい表に、元の表に格納されていたデータと同様のデータを設定します。

タスクの結果

これで、現行の脆弱性評価テストを使用するようにシステムが構成されました。

次のタスク

親トピック: 評価および強化

RACF の脆弱性の評価

IBM DB2 for z/OS を使用する場合は、脆弱性評価テストで RACF の脆弱性を評価することができます。RACF アセスメントを使用するには、少なくともバージョン 9.1 の Guardium がインストールされている必要があります。

このタスクについて

リソース・アクセス管理機能 (RACF) 特権がデータベース内で付与されたのかデータベース外で付与されたのかを評価します。 RACF 脆弱性評価を構成するこのテストでは、オブジェクト特権、データベース特権、およびシステム特権のアクセス制御が識別されます。

これらのテストを使用するためには、IBM Security zSecure Audit バージョン 2.1 を入手し、インストールする必要があります。 この製品は、これらのテストで RACF との対話に使用されるコマンドを使用可能にします。

資格を検査するテストでは、合格/不合格のグレードは返されません。資格を持つユーザーのリストが返されます。 これらのレポートのサンプルには、被付与者に付与された表特権およびビュー特権と、被付与者に付与されたパッケージ特権が含まれています。非常に多数のユーザーやアプリケーションが含まれている大規模な環境では、これらのレポートによって膨大な量のデータが生成されます。このような大規模な環境でこれらのレポートを実行すると、プロセスが長い時間実行され、大量のリソースが消費される可能性があり、最終的にタイムアウトになることがあります。

手順

1. データベース・サーバー上で脆弱性評価をサポートするために使用されるデータベース・スキーマをアップグレードします。
2. データベース・サーバーに zSecure Audit をインストールします。 zSecure Audit に付属している説明とツールを使用して、新しい zSecure テストをサポートするために、CKADBVA スキーマの およそ 24 個の表にデータを設定する方法を確認します。
3. zSecure チームは、zSecure Audit と Guardium 脆弱性評価との連携を可能にする PTF を発行します。この PTF を入手し、付属している説明に従って PTF を適用します。

タスクの結果

これでシステムが、新しい zSecure テストを活用するように構成されます。

次のタスク

実行する新しいテストを選択して、RACF の脆弱性を評価します。テストを構成し、実行します。

親トピック: 評価および強化

構成監査システム (CAS)

構成監査システム (CAS) は、サーバー環境に対する変更をトラッキングして報告します。例えば、構成ファイル、環境変数やレジストリー変数、他のデータベース・コンポーネントやオペレーティング・システムのコンポーネント (データベース管理システムやオペレーティング・システムが使用する実行可能ファイルやスクリプトを含む) の変更などです。このデータは Guardium システム上で使用可能であり、レポートやアラートに使用できます。

注: 構成監査システムは英語でのみサポートされています。

CAS エージェント

CAS はデータベース・サーバー上にインストールされたエージェントであり、モニター対象のエンティティーの内容、所有権、またはアクセス権に変更が加えられるたびに、Guardium システムに報告します。CAS クライアントは、S-TAP® のインストールで使用するのと同じユーティリティーを使用して、データベース・サーバー・システム上にインストールします。CAS と S-TAP は互いに独立して実行されますが、構成情報はコンポーネント間で共有します。 CAS クライアントをホスト上にインストールした後、実際の変更監査機能を Guardium® ポータルで構成します。

CAS サーバー

CAS サーバーは、Guardium のコンポーネントであり、Guardium システム上で稼働します。これは、Tomcat アプリケーション・サーバーとは関係なく、スタンドアロン・プロセスとして実行されます。 また、inittab ファイルを介して制御されます。

CAS サーバーは、Guardium システム上の少数の使用可能なプロセッサーを使用するように構成されています。CAS で使用されるプロセッサーの数は、*divide_num_of_processors_by* パラメーターを使用して決定されます。 このパラメーターは cas.server.config.properties ファイルに保管されます。そのデフォルト値は 2 です。Guardium システム上で使用可能なプロセッサーの数がこの値で除算されます。これにより、割り振られたプロセッサー上で CAS によって CPU が 100% 使用されている場合でも、残りのプロセッサーを他のアプリケーションで使用できるようになります。

CAS サーバー認証

Guardium は、SSL で提供される基本セキュリティーに加え、データベース・サーバー上で実行される CAS クライアント上での CAS サーバー認証をサポートします。 これは、CAS クライアントが Guardium の CAS サーバーとのみ通信することを保証するものとなります。 非認証接続および共通名 (CN) の不一致は CAS ログ・ファイルで報告されます。

この構成を行うと、CAS サーバー始動時には、署名済み証明書が秘密鍵とともにそこにロードされ、接続を受け入れるサーバー・ソケットにそれらが割り当てられます。データベース・サーバー側にある CAS クライアントは、以下の接続モードをサポートします。

1. 非セキュア接続 (use_tls='0')
2. 認証なしのセキュア接続 (use_tls ='1', guardium_ca_path=NULL)。このモードでは、CAS サーバーとの通信手段として SSL の使用が強制されます (つまり、サーバー認証なしの SSL の使用)。
3. サーバー認証付きセキュア接続 (use_tls ='1', guardium_ca_path=<public key location>)。CAS クライアントは CAS サーバーを認証するために公開鍵を使用します。この公開鍵 (ca.cert.pem) は <install_dir>/etc/pki/certs/trusted の下に配置されます。

   ca.cert.pem はルート認証局証明書 (自己署名されたもの) を格納したファイルです。これは、ブラウザーにおけるトラステッド CA 証明書 (VeriSign の証明書など) に相当します。

   すべての gmachine 証明書はルート認証局によって発行/署名されます。このようにして、証明書は検証され、トラスト・チェーンが確立されます。

   guardium_ca_path には、実際の公開鍵ファイル名を示す絶対パスを設定することもできますし、単にディレクトリー名 (<install_dir>/etc/pki/certs/trusted) を設定することもできます。ディレクトリー名の場合、そのディレクトリーの中にあるすべての公開鍵がサーバーの認証に使用されます。 公開鍵が存在しないファイルまたはディレクトリーを guardium_ca_path に設定すると、接続を試行しても失敗します。

4. サーバー認証付きセキュア接続と共通名検証。 このモードには追加検査があり、サーバーからの証明書 CN はパラメーター sqlguard_cert_cn で設定された CN と比較されます。 sqlguard_cert_cn が NULL または空である場合、この検査は使用不可になります。 それ以外の場合、Guardium の自己署名証明書が持つのと同じ CN ('gmachine') が設定されていることが必要です。

注: ここに記載されているパラメーターはすべて guard_tap.ini ファイルに含まれています。

CAS での SSL の使用

CAS エージェントは、CAS サーバーにデータを送信する際に Secure Sockets Layer (SSL) 接続を使用するように構成できます。 バージョン 10.6 でインストールされた CAS サーバーは、米国連邦情報処理標準 140-2 (FIPS 140-2) の要件に準拠しています。SSL を使用してこの CAS サーバーと通信できるのは、FIPS 準拠の CAS エージェントだけです。 このアプローチを使用する場合は、ご使用の CAS エージェントをこのパッチとともに配布されたバージョンにアップグレードする必要があります。 また、CAS エージェントが実行されているサーバーに IBM Java がインストール済みでなければなりません。さらに、CAS エージェントがそれを使用するように構成されている必要があります。 FIPS 通信を使用するためには、証明書ベースの認証を使用している必要があります。

古い CAS エージェントで、SSL を使用して、更新済みの CAS サーバーと通信しようとすると、CAS エージェント・システム上のログ・ファイルに次のメッセージが表示されます。

```
javax.net.ssl.SSLHandshakeException: Received fatal alert: handshake_failure
```

Guardium システム上の CAS ログ・ファイルに次のメッセージが表示されることもあります。

```
javax.net.ssl.SSLHandshakeException: Client requested protocol SSLv3 not
enabled or not supported
```

CAS エージェントと CAS サーバーの間で非 SSL 接続を使用する場合は、引き続き既存の CAS エージェントを使用できます。

テンプレート・セット

CAS テンプレート・セットには、一緒に組み込まれている項目テンプレートのリストが含まれていて、特定のタイプのデータベース (UNIX 上の Oracle など) のモニターなどの共通の目的を共有しています。このセットは次の 2 つのタイプのいずれかです。

- オペレーティング・システムのみ (UNIX または Windows)
- データベース (UNIX-Oracle、Windows-Oracle、UNIX-Db2、Windows-Db2 など)

データベース・テンプレート・セットは、データベース・タイプとオペレーティング・システム・タイプのどちらの場合も常に固有のものです。

CAS テンプレート項目

単一のモニター対象エンティティーに対するモニター・タスクの定義または属性セット。ユーザーは、新規の CAS テンプレートを作成して新しい CAS テストを定義することも、変更可能な事前定義 CAS テンプレートを使用することもできます。

テンプレート項目は、特定のファイルまたはファイル・パターン、環境変数またはレジストリー変数、OS スクリプトまたは SQL スクリプトの出力、あるいはログインしたユーザーのリストです。 これらの項目すべての状態は、生データ、つまりファイルの内容やレジストリー変数の値などによって影響を受けます。 CAS は、生データのサイズを検査するか、または生データのチェックサムを計算することによって、変更を検出します。 ファイルの場合は、ファイルの所有権、アクセス許可、パスなどの CAS はシステム・レベルの変更も検査します。

すべてのユニット (コレクターとアグリゲーター) が 1 つのマネージャーによって管理されるフェデレーテッド環境では、すべてのテンプレートはコレクターとアグリゲーターの両方で共有され、CAS データはレポート作成や脆弱性評価に使用できます。 コレクターとアグリゲーター (またはアーカイブされるデータがリストアされるホスト) が同一管理クラスターの一部ではない場合、テンプレートは共有されません。それで、CAS データは、たとえデータが存在していたとしても脆弱性評価には使用できなくなります。これを改善するには、定義のエクスポート/インポートを使用してテンプレートをコレクターからアグリゲーター (またはリストア・ターゲット) にコピーします。

注: クライアントあたり、10,000 を超えるファイルをモニターするように CAS に要求しないでください。
注: 処理するモニター対象のファイル数が、1 時間あたり 1,000 以下になるように、 CAS を構成することをお勧めします。

モニター対象エンティティー

モニター対象にできる実際のエンティティーとしては、1 つのファイル (その内容とプロパティー)、環境変数の値または Windows レジストリーの値、さらに OS のコマンドやスクリプトと SQL ステートメントからの出力が挙げられます。

CAS インスタンス

特定のホスト (そのテンプレート・セットのインスタンスの作成と特定のホスト上への適用を行うホスト) への CAS テンプレート・セットの適用

CAS 構成

CAS 構成によって 1 つ以上の CAS インスタンスが定義されます。各インスタンスは、ホスト上のセットになった項目をモニターするために使用されるテンプレート・セットを識別します。

デフォルトのテンプレート・セット

Guardium は、サポートされる各オペレーティング・システム・タイプおよびデータベース・タイプに対して、事前構成されたデフォルトのテンプレート・セットを提供しています。これは UNIX または Windows プラットフォーム上のさまざまなデータベースをモニターするためのものです。 デフォルトのテンプレート・セットから操作を開始して、特定のテンプレート・セット・タイプ用に定義された新規のテンプレート・セットを使用できます。 テンプレート・セット・タイプは、 オペレーティング・システムのみ (Unix または Windows)、またはデータベース管理システム (DB2®、Informix®、Oracle など) のどちらかです。データベース管理システムは常にオペレーティング・システム・タイプで修飾されます。 例えば、UNIX-Oracle、あるいは Windows-Oracle というようになります。多くの事前構成されたデフォルトのテンプレート・セットは、Guardium の脆弱性評価で使用され、例えば、既知のパラメーター、ファイルのロケーション、ファイル・アクセス権などを検査することができます。

Guardium デフォルト・テンプレート・セットを変更することはできませんが、そのコピーを作成してコピーとして作成したバージョンを変更することはできます。各 Guardium デフォルト・テンプレート・セットは、一連のモニター対象項目を定義します。デフォルト・テンプレート・セットでモニターされる項目それぞれの機能と使用法をよく把握して、実際の環境に合ったものを使用してください。独自のテンプレート・セットを定義した場合、そのテンプレート・セットを、そのテンプレート・セットが対象とするタイプにおけるデフォルトとして指定することができます。 そのようにすれば、独自の新しいデフォルト・テンプレート・セットから操作を開始して、特定のオペレーティング・システムとデータベース・タイプ用の新規のテンプレート・セットを定義できます。 そのタイプにおける Guardium デフォルト・テンプレート・セットは削除されません。定義は残りますが、デフォルトとしてのマークは付かなくなります。

テンプレート・セット作成を特定のデータベース構成に合わせるための理論的根拠

Guardium は事前定義 CAS テンプレート・セットを各データベース・タイプ用に提供していますが、データベース構成の種類は多彩であるため、実稼働環境のすべての要件に合わせるには、事前定義テンプレート・セットを微調整するか、またはテンプレート・セットを新規作成しなければならない場合もあります。とりわけ、データベース・ソフトウェアとデータ・ファイルのロケーションに関してはこれが当てはまります。 CAS を使用してデータベース・ファイルの所有権、アクセス許可、そして変更内容をモニターする場合は、追加のテンプレートの作成を計画する必要があります。

例えば、Oracle 用の事前定義 CAS テンプレート・セットにはいくつかのテンプレートが含まれていますが、その中に次のものがあります。

- $ORACLE_HOME/oradata/../.*dbf
- $ORACLE_HOME/oradata/../.*ctl
- $ORACLE_HOME/oradata/../.*log
- $ORACLE_HOME/../init.*.ora

見てわかるとおり、これらのファイル・パターン・テンプレートは、すべて同じルート $ORACLE_HOME で始まります (注: これは、必ずしもご使用のデータベース・サーバーで必ずしも定義されている $ORACLE_HOME 環境変数であるわけではありません。設定によっては、CAS はデータ・ソースのフィールド「データベース・インスタン

ス・ディレクトリー」を $ORACLE_HOME の値として使用します)。

実稼働環境で、Oracle のデータ・ファイルがログ・ファイルと同一のディレクトリー・ツリーに存在しないか、場合によっては同一のデバイス上にさえ存在しない可能性があります。また、Oracle 構成ファイルもさらに別のロケーションにある可能性もあります。

CAS で 以下のような Oracle のファイルすべてを検出してモニターできるような、絶対パスを使用する追加の CAS テンプレートを作成することもできます。

- /u01/oradata/mydb/*.dbf
- /u02/oradata/mydb/*.dbf
- /u03/oradata/mydb/*.dbf
- /u01/oradata/mydb/*.ctl
- /u02/oradata/mydb/*.ctl
- /u03/oradata/mydb/*.ctl
- /home/oracle11/admin/mydb/bdump/*.log
- /home/oracle11/product/11.1/db_1/dbs/init*.ora

さらには、Oracle インスタンス・アカウントで定義した追加の環境変数を使用することもできます。 例えば変数を、$ORA_DATA1、$ORA_DATA2 および $ORA_SOFT として定義した場合、次のように指定できます。

- $ORA_DATA1/mydb/*.dbf
- $ORA_DATA2/mydb/*.dbf
- $ORA_DATA1/mydb/*.ctl
- $ORA_DATA2/mydb/*.ctl
- $ORA_SOFT/admin/mydb/bdump/*.log
- $ORA_SOFT/product/11.1/db_1/dbs/init*.ora

別のロケーションからのファイルのソーシング

CAS テンプレートでは、ユーザー・プロファイルなどの特定のファイルが特定のロケーションにあることを前提としています。 CAS は、正規表現を使用して指定した他のロケーションで、これらのファイルを検索するように構成できます。 この機能を使用するには、user_profile_files パラメーターを config ディレクトリー内の cas.client.config.properties ファイルに追加します。 各項目の形式は、次のとおりです。

```
identifying_string=comma-separated list of files
```

例えば、いずれかの Db2 ユーザーのホーム・ディレクトリーで .profile ファイルを検索するとします。 この例では、これらすべてのホーム・ディレクトリーの名前に「db2」という文字列が含まれていることを前提としています。 プロパティー・ファイルに次の行を追加します。

```
user_profile_files=.*db2.*=.profile
```

複数のパターンを指定する必要がある場合は、縦線記号 (|) を使用してパターンを区切ります。 mysql ユーザーのプロファイルを前の項目に追加する場合は、前述の例を以下のものに置き換えます。

```
user_profile_files=.*db2.*=.profile|.*mysql.*=.profile
```

- Windows サーバーにおける CAS の前提条件、インストール、および実行
  CAS の前提条件と、ご使用のデータベース・サーバーへの CAS エージェントのインストール方法について説明します。
- Linux/UNIX サーバーにおける CAS の前提条件、インストール、および実行
  CAS の前提条件と、ご使用のデータベース・サーバーへの CAS エージェントのインストール方法について説明します。
- Java ホーム・ディレクトリーの場所の探索とバージョンの確認
  CAS をインストールする前に、ホーム・ディレクトリーの場所の探索と Java バージョンの確認を行います。
- CAS の始動とフェイルオーバー
  フェイルオーバーと接続の種々のパラメーターは『S-TAP 制御の変更監査』で変更できます。
- CAS テンプレート
  Guardium には、データ・リポジトリーのタイプごとに CAS テンプレートのセットが 1 つ用意されています。
- CAS テンプレートの処理
  このセクションでは、CAS テンプレートの維持方法について説明します。
- CAS ホスト
  構成監査システム (CAS) のホスト構成では、1 つ以上の CAS インスタンスが定義されます。
- CAS レポート
  このセクションでは、構成監査システム (CAS) のレポート作成について説明します。
- CAS 状況
  「管理」 > 「変更モニター」 > 「CAS 状況」をクリックして、「構成監査システム状況」を開きます。

親トピック: 評価および強化

Windows サーバーにおける CAS の前提条件、インストール、および実行

CAS の前提条件と、ご使用のデータベース・サーバーへの CAS エージェントのインストール方法について説明します。

Windows における前提条件

表 1. Windows サーバーのディスク・スペース要件

ディスク・スペース	記述
CAS プログラム・ファイル (Java™ を含む)	277 MB

表 2. Windows サーバーのポート要件

ポート	プロトコル	Guardium システムの接続先
16017	TCP	クリアな (ポートをオープン) CAS

ポート	プロトコル	Guardium システムの接続先
16019	TLS	暗号化された CAS

CAS のインストール

Windows インストーラーを使用します。これについては特に説明を要しません。

CAS および 64 ビットの Windows レジストリー

Windows では、ソフトウェア構成パラメーターはキー HKEY_LOCAL_MACHINE¥SOFTWARE のレジストリー・ツリーに格納されています。 64 ビット・マシンでは、同じアプリケーションの 64 ビット・バージョンと 32 ビット・バージョンの両方を稼働できるため、64 ビット・アプリケーションと 32 ビット・アプリケーションの構成パラメーターを区別する必要があります。

この問題に対する Microsoft の解決策は、レジストリーをパーティションで区切ることです。WOW6432Node というラベルが付いた特殊キーが、 キー HKEY_LOCAL_MACHINE¥SOFTWARE のレジストリー・ツリーに追加されます。32 ビット・ アプリケーションがキー HKEY_LOCAL_MACHINE¥SOFTWARE にあるパスを介してレジストリーにアクセスしようとすると、 Windows はそのパスに特殊キー WOW6432Node を挿入します。このようにすることで、32 ビット・アプリケーションは 32 ビット・マシンで行うのと同じように Windows レジストリーを扱い、 Windows は正しいパーティションへのリダイレクトを処理します。

CAS は 32 ビットの Java アプリケーションであるため、 通常は 64 ビットのソフトウェア構成パラメーターへのアクセスは持ちません。 CAS は、64 ビットの環境を検出し、パーティション化されたレジストリーを 処理するように拡張されています。CAS がレジストリーに対して関心を持つのは、レジストリー・キーの値を取得して、 変更を検出したり、推奨値と比較したりするためです。

例として、CAS が HKEY_LOCAL_MACHINE¥SOFTWARE¥MyApp¥Parameter1 の値を取得するとします。その値は、 パーティションのどちらかにあるか、両方にあるか、どちらにもないかのいずれかです。値がどちらのパーティションにもない場合、CAS は NULL を取得します。それ以外の場合は、 文字列 WOW6432Node によって区切られた 2 つの値の連結である文字列を戻します。値が 64 ビット・パーティションにはあるが、32 ビット・パーティションにはない場合、 取得される文字列は Value64WOW6432Nodenull のようになります。逆に、値が 32 ビット・パーティションにはあるが、64 ビット・パーティションにはない場合、文字列は nullWOW6432NodeValue32 です。 最後に、値が両方のパーティションにある場合、返される文字列は Value64WOW6432NodeValue32 です。 この新しいレジストリー値パターン検索は、必要に応じて、両方のレジストリー・パーティションを検索します。

JAVA_HOME ロケーションの CAS 再構成

ほとんどの場合、インストール・プログラムは JAVA_HOME 値の検索を処理します。 この値は CAS 構成ファイルにあります。

なんらかの理由 (Guardium® CAS 製品のインストール後に新しい Java バージョンを インストールするなど) で、JAVA_HOME のロケーションを変更する必要がある場合は、 以下の手順に従ってください。

1. CAS 構成ファイルを見つけて、編集用に開きます。絶対パス名は以下のとおりです。<installation directory>/case/conf/wrapper.conf
2. `wrapper.java.command=<value>` の項目を見つけます。
3. 値を JAVA_HOME ディレクトリーに置換します。
4. ファイルを保存します。

親トピック: 構成監査システム (CAS)

Linux/UNIX サーバーにおける CAS の前提条件、インストール、および実行

CAS の前提条件と、ご使用のデータベース・サーバーへの CAS エージェントのインストール方法について説明します。

Linux/UNIX サーバーにおける前提条件

表 1. Linux/UNIX サーバーのディスク・スペース要件

ディスク・スペース	記述
CAS プログラム・ファイル (Java™ を含む)	AIX®: 309 MB、HP-UX: 630 MB、Linux: 405 MB、Solaris: 390 MB CAS には Java が必要です。お客様自身で Java を取得およびインストールしていただく必要があります (ライセンス交付の制約により)。 • HP-UX: Java 1.5 以上 • HP-UX 以外のサーバー: Java 1.4.2 以上

表 2. Linux/UNIX サーバーのポート要件

ポート	プロトコル	Guardium システムの接続先
16017	TCP	クリアな CAS
16019	TLS	暗号化された CAS

コマンド行からの CAS クライアントのインストール

1. root アカウントを使用して、データベース・サーバー・システムにログオンします。
2. コマンド guard-cas-setup を次の構文を使用して実行します。

```
guard-cas-setup -- install --java-home <JAVA_HOME> --install-path <INSTALL_PATH> --stap-conf <FULL_PATH_TO_GUARD_TAP_INI>
usage (variables are shown enclosed in angled brackets: < >):
guard-cas-setup -- uninstall
<guard-cas-setup> is the name of the script file
-- install は、CAS のインストールを示します。
```

```
-- uninstall は、CAS のアンインストールを示します。
--java-home <JAVA_HOME> は、JAVA_HOME ディレクトリーを識別します。
--install-path は、インストール・パスを識別します。
--stap-conf <FULL_PATH_TO_GUARD_TAP_INI>identifies where the guard_tap.ini file is located
```

CAS パラメーターについては、CAS パラメーター を参照してください。

コマンド行からの CAS の始動および停止

インストールまたはアンインストールのシナリオに応じて、コマンド行から CAS を始動および停止する必要がある場合があります。シナリオによっては、guard_tap.ini ファイルへの --stap-conf パスを指定しない場合があります (これはオプション・パラメーターであるため)。この場合、CAS は始動しません。CAS を始動または停止する必要がある場合、次の方法を使用します。

1. root アカウントを使用して、データベース・サーバー・システムにログオンします。
2. Red Hat Enterprise Linux 6 の場合は、stop cas コマンドまたは start cas コマンドを使用して CAS を停止または始動します。
3. その他すべての場合:
   a. /etc/inittab ファイル内の CAS エージェント項目を、コメント化するか (CAS を停止する場合)、コメントを削除します (CAS を始動する場合)。デフォルト・インストールでは、このステートメントは以下のようになります。

      ```
      cas:<nnnn>::respawn:/usr/local/guardium/guard_stap/cas/bin/run_wrapper.sh /usr/local/guardium/guard_stap/cas/bin
      ```

   b. /etc/inittab ファイルを保存します。
   c. init q コマンドを実行します。
4. -fe | grep cas コマンドを実行することで、CAS が実行されているかどうかを検証できます。

JAVA_HOME ロケーションの CAS 再構成

ほとんどの場合、インストール・プログラムは JAVA_HOME 値の検索を処理します。 この値は CAS 構成ファイルにあります。

なんらかの理由 (Guardium® CAS 製品のインストール後に新しい Java バージョンを インストールするなど) で、JAVA_HOME のロケーションを変更する必要がある場合は、 以下の手順に従ってください。

1. CAS 構成ファイルを見つけて、編集用に開きます。絶対パス名は以下のとおりです。<installation directory>/case/conf/wrapper.conf
2. `wrapper.java.command=<value>` の項目を見つけます。
3. 値を JAVA_HOME ディレクトリーに置換します。
4. ファイルを保存します。

親トピック: 構成監査システム (CAS)

Java ホーム・ディレクトリーの場所の探索とバージョンの確認

CAS をインストールする前に、ホーム・ディレクトリーの場所の探索と Java バージョンの確認を行います。

このタスクについて

CAS を UNIX システムにインストールする際には、以下の 2 つの要件があります。

- JAVA_HOME ディレクトリーを特定する。CAS のインストール中に、その場所を求めるプロンプトが出されます。
- サポートされる Java™ のバージョンがインストールされていることを確認する。サポートされるバージョンがインストールされていない場合は、CAS をインストールする前にそれをインストールする必要があります。
  注: FIPS 準拠の環境で SSL による CAS を使用するには、CAS エージェントが実行されているサーバーに IBM Java がインストールされている必要があります。

JAVA_HOME ディレクトリーには、Java コマンドが格納されています。以下に例を示します。

- java コマンドが /usr/local/j2sdk1.4.2_03/bin/java の場合
- JAVA_HOME ディレクトリーは、/usr/local/j2sdk1.4.2_03 です。

手順

1. which java コマンドを入力します。例:

   ```
   [root@yourserver ~]# which java
   /usr/local/j2sdk1.4.2_03/bin/java
   ```

   JAVA_HOME ディレクトリーは /usr/local/j2sdk1.4.2_03 です。
2. which java コマンドによってシンボリック・リンクが戻された場合は、ls -ld <symbolic_link> コマンドを使用して実際の Java ディレクトリー名を判別します。
3. which java コマンドによって「command not found」というメッセージが戻された場合、Java はインストールされていても、PATH 変数には含まれていない可能性があります。この場合は、find コマンドを使用して Java ディレクトリーを見つけます。以下に例を示します。

   ```
   [root@yourserver ~]# find . -name java
   ./usr/bin/
   ```

4. バージョン番号を確認するには、Java ディレクトリーから java -version コマンドを実行します。以下に例を示します。

   ```
   [root@yourserver ~]# /usr/local/j2sdk1.4.2_03/bin/java -version
   java version "1.4.2_03"
   Java(TM) 2 Runtime Environment, Standard Edition (build 1.4.2_03-b02)
   Java HotSpot(TM) Client VM (build 1.4.2_03-b02, mixed mode)
   ```

5. 返された Java バージョンをメモします。この情報を求めるプロンプトは出されませんが、後で問題が発生した場合に、サポートされない Java バージョンの可能性を除外できます。

CAS の始動とフェイルオーバー

フェイルオーバーと接続の種々のパラメーターは『S-TAP® 制御の変更監査』で変更できます。

CAS クライアントは、ホスト上で始動するときに、以前にシステムに書き込んだチェックポイント・ファイルがあるかどうかを検索します。 CAS は、このファイルから前回実行時に行った処理を知ることができます。 それから、CAS は Guardium システムに接続します。チェックポイント・ファイルが検出された場合は、CAS は、そのモニター割り当てのバージョンを、Guardium® データベースに保管されているバージョンと比較して検証するように、Guardium システムに要求します。CAS クライアントと Guardium システムが切断されていた間に、割り当てに変更があった可能性があります。差異があった場合そのすべてが解決されると、CAS はモニターを再開します。チェックポイント・ファイルが検出されなかった場合、CAS は、Guardium システムにどのような処理を行うかを尋ねます。Guardium システムがそのデータベースで CAS ホストを検出した場合、関連するテンプレート・セットが CAS クライアントに送信され、モニター項目に展開され、モニターが開始されます。Guardium システムがデータベースで CAS ホストを検出できなかった場合は、その CAS ホストをデータベースに追加し、CAS ホストのオペレーティング・システム用のデフォルト・テンプレート・セットを送信します。

CAS クライアントと Guardium システムの間の接続が失われると、1 次 Guardium システムとの連絡が失われたことを CAS クライアントと Guardium システムが検出するまでに、最大 5 分 (CAS クライアントが Guardium システムからのメッセージを待機する時間) かかります。通信エラーが検出された場合にはこれより早い場合もあります。

CAS クライアントが Guardium システムとの接続を失った場合、または初期接続ができなかった場合、CAS クライアントはフェイルオーバー・ファイルを開いて、Guardium システムに送信するはずだったメッセージをフェイルオーバー・ファイルに書き込み始めます。このフェイルオーバー・ファイルのパスは、guard_tap.ini の中に、cas_fail_over_file という名前で保管されています。通信が再確立されると、CAS クライアントはシャットダウンと再始動を行い、フェイルオーバー・ファイルに保管したすべてのメッセージを Guardium システムに送信し、そのファイルを削除します。CAS クライアントが初期接続できなかった場合は、チェックポイント・ファイルを使用してモニター対象を判別し、通信障害の前に行っていた作業を続行します。

通信が失われると、クライアントはスレッドも開始します。このスレッドは周期的に 1 次 Guardium システムとの再接続を試行します。CAS が再接続を試行する回数と、再接続試行の平均時間間隔は、構成可能なパラメーターになっています。 クライアントは、guard_tap.ini の cas_server_failover_delay という名前で設定された期間、再接続を試行します。その時間が過ぎると、クライアントは guard_tap.ini で指定される 2 次サーバーへの接続も試行します。2 次サーバーへの試行は、guard_tap.ini の SQL_Guard セクションにリストされた「primary」属性値の順序で行われます。primary が 1 でない場合は 2 次になります。クライアントが 2 次サーバーに接続している間も、1 次サーバーへの再接続の試行を継続します。

再接続の試行限度に達すると、CAS クライアントは再接続の試行を停止しますが、フェイルオーバー・ファイルへのデータの書き込みは継続します。 データベース・サーバー上のディスク・スペース所要量を一定以下にするため、実際には 2 つのフェイルオーバー・ファイルが存在します。 CAS はフェイルオーバー・ファイルの最大サイズ (これは構成可能) に達するまで 1 つのファイルに書き込みます。それからもう一方に切り替え、そのファイルにあった以前のデータに上書きします。 デフォルトのフェイルオーバー・ファイルのサイズは、50MB (ファイルごと) です。

CAS クライアントを構成する際には、1 つ以上の 2 次 Guardium システムを指定できます。フェイルオーバー・モードでは、CAS は guard_tap.ini の cas_server_failover_delay で指定された時間を超えるまでは、1 次サーバーへの再接続のみを試行します。その時間になると、CAS は、1 次サーバーに試行すると同時に 2 次サーバーへの接続試行も開始します (再接続試行中に最初に接続を試行するのは常に 1 次サーバーです)。 CAS は、2 次サーバーに接続している間も、1 次サーバーへの再接続の試行を続行します。

CAS クライアントの構成変更は、1 次サーバーからのみ行うことができ、ホストがオンラインのときにのみ実行できます。 Guardium システムがスタンドアロン構成になっている場合、1 次サーバー上で CAS クライアントの構成が変更されるたびに、ホスト上にエクスポート・ファイルが保存されます。CAS クライアントが 2 次サーバーに接続すると、保存されたエクスポート・ファイルはホストから 2 次サーバーにインポートされます。

1 次サーバーと 2 次サーバーの両方で構成を別個に維持する必要はありません。ただし、1 次サーバー上で個々のモニター対象項目のパラメーターがテンプレート定義から変更された場合、その変更は 2 次サーバーに転送されることはありません。 例えば、特定のファイルに対するテスト間隔がテンプレートのデフォルトである 1 時間から 10 分に変更された場合でも、2 次サーバーでのテスト間隔は元の 1 時間のままです。基本的には、モニター対象項目はインポートされた構成のテンプレートから再生成されます。 2 次サーバーの検索を開始するまでの遅延は時間に直接基づいており、フェイルオーバー・ファイルのサイズは関係ありません。 遅延は guard_tap.ini の cas_server_failover_delay パラメーターで設定され、デフォルトは 60 分です。

フェイルオーバーと接続の種々のパラメーターは『S-TAP 制御の変更監査』で変更できます。

S-TAP の場合と同様、CAS 接続に障害があると Guardiumシステム上に例外が作成されるため、障害を検出すると即座にアラートを出すことができます。

2 次サーバーの設定と保守

データベース・サーバー・システム上の S-TAP/CAS 構成ファイルには、1 つ以上の 2 次 Guardium サーバーを定義できます。1 次 Guardium サーバーが使用不可になった場合、そのデータベース・サーバー・システム上の CAS は 2 次 Guardium システムに接続します (前述のとおりです。『始動とフェイルオーバー』を参照してください)。

フェイルオーバーのルール

ルール #	Guardium システム	フェイルオーバー先	有効
1	スタンドアロン	スタンドアロン	はい
2	管理対象	管理対象 (マネージャーが同一)	はい
3	管理対象	管理対象 (マネージャーが異なる)	いいえ
4	管理対象	スタンドアロン	いいえ
5	スタンドアロン	管理対象	いいえ

CAS フェイルオーバーの制限

1. CAS インスタンスは、ソース Guardium システムが管理対象ユニットであり、ターゲット Guardium システムが以下のいずれかである場合には、フェイルオーバー Guardium システムへ再配置されません。
    - スタンドアロン Guardium システム

- 異なるマネージャーが管理している管理対象ユニット

2. CAS インポート/エクスポート・オプションは、マネージャーとスタンドアロン・マシンのみに制限されます。

CAS ホストのエクスポート

1. 「管理」 > 「統合/アーカイブ」 > 「エクスポート」をクリックし、「定義のエクスポート」パネルを開きます。「タイプ」メニューから「CAS ホスト」を選択し、エクスポートする定義を「エクスポートする定義」メニューから選択し、「エクスポート」をクリックします。
2. exp_<date>_<time>.sql という名前のファイルがシステムに保存されます。このファイルには、選択したすべての CAS ホストの定義と、それらの CAS ホストが使用するすべてのテンプレート・セットの定義が格納されます。

CAS ホストのインポート

1. 「管理」 > 「統合/アーカイブ」 > 「インポート」をクリックし、「定義のインポート」パネルを開きます。
2. 「参照」ボタンと「アップロード」ボタンを使用してファイルを選択してアップロードした後、「アップロード済み定義のインポート」ペインから定義を選択します。
3. 「この定義セットをインポート」をクリックして定義をインポートします。
4. 選択したアクションを (行うか行わないかを) 確認します。
   注: インポート操作では、既存の定義に上書きされません。既存の定義と同じ名前の定義をインポートしようとすると、その項目は置き換えられなかったことが通知されます。 インポートされた定義で既存の定義を上書きする場合は、インポート操作を実行する前に、既存の定義を削除する必要があります。

CAS ホストの 2 次サーバーの保守

CAS 構成は、エクスポートとインポート操作を使用して保守することもできます。 インポート操作では既存の定義は置換されないため、それぞれの 2 次サーバーで、古い CAS ホスト定義を削除してから新しいものをインポートする必要があります。

この手順を実行するのは、必ず、選択した CAS ホストが 1 次サーバーに接続されているときのみにしてください。

1. CAS ホストの定義をエクスポートします (前のセクションを参照)。
2. それぞれの 2 次サーバーで以下の操作を行います。
   - 置換の対象になる古い CAS ホスト定義を削除します。
   - 1 次サーバーからエクスポートした定義をインポートします (前述の『CAS ホストのインポート』を参照)。

CAS クライアントの変更の無視アラート

CAS クライアント・エージェントは、事前定義設定に基づいて、変更通知が CAS サーバーに送信されないようにすることができます。

CAS クライアント・エージェントは、CAS クライアント・エージェントの cas.client.config.properties 構成ファイル内で新しいパラメーター ignore_change_alerts を探します。

このパラメーターが見つからないか設定されていない場合、CAS クライアントは変更なしで動作し、「変更の無視アラート」機能は有効になりません (例えば、CAS クライアントはファイル変更時にアラートを出します)。

新規パラメーターが設定されている場合、CAS クライアント・エージェントは、パラメーター値に指定された変更タイプに基づいて、変更通知の送信を無視します。

使用可能な変更タイプは次のとおりです。

PERMISSION、SIZE、OWNER、GROUP、TIMESTAMP

複数の変更タイプを無視する場合は、指定した複数の変更タイプを「+」で区切って連結して設定できます。

例:

OWNER と GROUP の変更時に変更通知が送信されないようにするには、以下のようにパラメーターを設定します。

ignore_change_alerts=OWNER+GROUP

注: 初期インストール時または新規テンプレートの定義時に、ファイルの最初のスキャンが実行され、これらのファイルは、「変更の無視アラート」の設定に関係なく、「CAS 変更」レポートに表示されます。

無効な非 IP ホスト名の修正

ユーザーが無効な tap_ip (guard_tap.ini パラメーター) または CAS_TAP_IP (GIM パラメーター) を指定して CAS エージェントをインストールすると、そのホストに関して定義された Windows データ・ソースが使用できなくなる可能性があります (リモート・データベースへのアクセスを必要とするアクティビティーについて使用された場合)。

このような状況が起きた場合は、データ・ソースを削除し、tap_ip パラメーターを正しいデータベース・サーバー・ホスト名/IP に変更する必要があります。

親トピック: 構成監査システム (CAS)

CAS テンプレート

Guardium には、データ・リポジトリーのタイプごとに CAS テンプレートのセットが 1 つ用意されています。

CAS テンプレート - Db2

OS スクリプト

実行する OS スクリプトを指定します。CAS ホーム・ディレクトリーの下のスクリプト・ディレクトリーを示す変数 $SCRIPTS で始まり、実行するスクリプト (例えば、`$HOME/ db2_spm_log_path_group_test.sh`") を示している必要があります。スクリプト自体も CAS $SCRIPTS ディレクトリーにある必要があります。 スクリプトからの出力は、Guardium® データベースに格納され、セキュリティー・アセスメントに使用されます。 これは、実行可能なシェル・スクリプトまたはバッチ・スクリプトか、あるいはコマンド行に入力できるコマンドのセットのいずれかです。 Java の構文解析は変更される可能性が高いため、最も単純な種類のコマンド以外については直接実行するのではなく、スクリプトに配置することを推奨します。 UNIX では、スクリプトは指定した OS ユーザーの環境で実行されます。 ユーザーがスクリプトを記述するために使用できる実行環境用に、3 つの環境変数が定義されます。$UCAS はデータベース・ユーザー名、$PCAS はデータベース・パスワード、$ICAS はデータベース・インスタンス名です。 Windows の場合、これらの 3 つの値はバッチ・ファイル実行時に最後の 3 つの引数として付加されます。 例えば、OS スクリプト・テンプレート `%SCRIPTS%¥MyScript.bat my-arg1 my-arg2` を使う場合、%3、%4、および %5 はそれぞれ DB ユーザー名、パスワード、およびインスタンス名になります。

ファイル

セキュリティー・アセスメントでトラッキングおよびモニターの対象にするファイルを指定します。 ファイルのパスは、絶対パスまたは $INSTHOME 変数を基準とする相対パスにすることができます。「データ・ソース定義」パネルの「データベース・インスタンス・ディレクトリー」で、$INSTHOME 変数の値を設定します。 これは単一のファイルの名前と見なされます。OS ユーザー環境の環境変数をファイル名の中で使うことができ、展開して使用されます。例えば、$HOME/START.sh は DB2® ユーザーのホーム・ディレクトリーにある始動スクリプトの名前です。

ファイル・パターン

セキュリティー・アセスメントでトラッキングおよびモニターの対象にするファイルのグループを指定します。 ファイルのパスは、絶対パスまたは $INSTHOME 変数を基準とする相対パスにすることができます。 「データ・ソース定義」パネルの「データベース・インスタンス・ディレクトリー」で、$INSTHOME 変数の値を設定します。 パスの中で指定する「..」は、その前のパス部分とその後のパス部分の間に 1 つ以上のディレクトリーがあることを示します。パスの中で指定する「.+」は、その前のパス部分とその後のパス部分の間に 1 つだけディレクトリーがあることを示します。 例えば、`$INSTHOME/sqllib/../db2.*` という指定は、1 つの単一識別文字列 (ディレクトリー内のすべてのファイルに一致するファイル・パターン) から多くの単一ファイル識別文字列を作成するための簡略表記になります。ファイル・パターンは、「/」で分離した一連の正規表現と見なせます。ファイルの絶対パスの各要素が順番どおりにその正規表現の 1 つと一致すれば、そのファイルが一致したということになります。 パターンの要素が環境変数である場合、突き合わせの前に環境変数が展開されます。 パターンのいずれかの要素が「..」である場合、ゼロ個以上のディレクトリー・レベルと一致することになります。 例えば、/usr/local/../foo は、/usr/local/foo や /usr/local/gunk/junk/bunk/foo と一致します。 1 つのファイル・パターンに複数の「..」要素を使用する必要はありませんし、使用すべきではありません。そのようにすると、パターンを展開するのに非常に時間がかかってしまうからです。混乱を避けるため、正規表現では Windows で使用される可能性のある区切り文字として ¥ を使用することはできません。

さらに、「Guardium Unix/Db2 のアセスメント: UNIX - Db2 for Unix」セットには、以下のテンプレートが含まれています。

Db2govd SETUID ビット未設定

このテストは、DB2GOVD の SETUID ビットが無効になっているかどうかをモニターします。

Db2start SETUID ビット未設定

このテストは、DB2START の SETUID ビットが無効になっているかどうかをモニターします。

Db2stop SETUID ビット未設定

このテストは、DB2STOP の SETUID ビットが無効になっているかどうかをモニターします。

ファイル所有権

このテストは、Db2 ファイルのファイル所有権およびその変更をモニターします。

ファイルの許可

このテストは、Db2 ファイルのファイル・アクセス許可およびその変更をモニターします。

CAS テンプレート - Informix

OS スクリプト

実行する OS スクリプトを指定します。CAS ホーム・ディレクトリーの下のスクリプト・ディレクトリーを示す変数 $SCRIPTS で始まり、実行するスクリプト (例えば、`$HOME/ informix_rootpath_owner.sh`") を示している必要があります。スクリプト自体も CAS $SCRIPTS ディレクトリーにある必要があります。 スクリプトからの出力は、Guardium データベースに格納され、セキュリティー・アセスメントに使用されます。 これは、実行可能なシェル・スクリプトまたはバッチ・スクリプトか、あるいはコマンド行に入力できるコマンドのセットのいずれかです。 Java の構文解析は変更される可能性が高いため、最も単純な種類のコマンド以外については直接実行するのではなく、スクリプトに配置することを推奨します。 UNIX では、スクリプトは指定した OS ユーザーの環境で実行されます。 ユーザーがスクリプトを記述するために使用できる実行環境用に、3 つの環境変数が定義されます。$UCAS はデータベース・ユーザー名、$PCAS はデータベース・パスワード、$ICAS はデータベース・インスタンス名です。 Windows の場合、これらの 3 つの値はバッチ・ファイル実行時に最後の 3 つの引数として付加されます。 例えば、OS スクリプト・テンプレート `%SCRIPTS%¥MyScript.bat my-arg1 my-arg2` を使う場合、%3、%4、および %5 はそれぞれ DB ユーザー名、パスワード、およびインスタンス名になります。

ファイル

セキュリティー・アセスメントでトラッキングとモニターの対象にするファイルを指定します。 ファイルのパスは、絶対パスまたは $ INFORMIXDIR 変数を基準とする相対パスにすることができます。「データ・ソース定義」パネルの「データベース・インスタンス・ディレクトリー」で、$INFORMIXDIR 変数の値を設定します。 これは単一のファイルの名前と見なされます。OS ユーザー環境の環境変数をファイル名の中で使うことができ、展開して使用されます。例えば、$HOME/START.sh は Informix® ユーザーのホーム・ディレクトリーにある始動スクリプトの名前です。

さらに、UNIX 用の「Guardium Unix/Informix のアセスメント」セットには、以下のテンプレートが含まれています。

ログ・ファイルに対するエラーのスキャン

このテストは、online.log ファイルにエラーがあるかどうかをモニターします。

ファイル所有権

このテストは、Informix ファイルのファイル所有権およびその変更をモニターします。

ファイルの許可

このテストは、Informix ファイルのファイル・アクセス許可およびその変更をモニターします。

CAS テンプレート - Oracle

OS スクリプト

実行する OS スクリプトを指定します。CAS ホーム・ディレクトリーの下のスクリプト・ディレクトリーを示す変数 $SCRIPTS で始まり、実行するスクリプト (例えば、$SCRIPTS/oracle_user.sh) を示している必要があります。スクリプト自体も CAS $SCRIPTS ディレクトリーにある必要があります。 スクリプトからの出力は、Guardium データベースに格納され、セキュリティー・アセスメントに使用されます。(これは、実行可能なシェル・スクリプトまたはバッチ・スクリプトか、あるいはコマンド行に入力できるコマンドのセットのいずれかです。 Java の構文解析は変更される可能性が高いため、最も単純な種類のコマンド以外については直接実行するのではなく、スクリプトに配置することを推奨します。 UNIX では、スクリプトは指定した OS ユーザーの環境で実行されます。 ユーザーがスクリプトを記述するために使用できる実行環境用に、3 つの環境変数が定義されます。$UCAS はデータベース・ユーザー名、$PCAS はデータベース・パスワード、$ICAS はデータベース・インスタンス名です。Windows の場合、これらの 3 つの値はバッチ・ファイル実行時に最後の 3 つの引数として付加されます。 例えば、OS スクリプト・テンプレート $SCRIPTS/mysql_mysqld_user.sh を使用する場合、%3、%4、および %5 はそれぞれ DB ユーザー名、パスワード、およびインスタンス名になります。)

ファイル

トラッキングとモニターの対象にするファイルを指定します。 ファイルのパスは、絶対パスまたは $ORACLE_HOME 変数を基準とする相対パスにすることができます。$ORACLE_HOME 変数の値は「データ・ソース定義」パネルの「データベース・インスタンス・ディレクトリー」フィールドで設定した値です。(これは単一のファイルの名前と見なされます。OS ユーザー環境の環境変数をファイル名の中で使うことができ、展開して使用されます。例えば、$HOME/START.sh は Oracle ユーザーのホーム・ディレクトリーにある始動スクリプトの名前です。)

ファイル・パターン

トラッキングとモニターの対象にするファイルのグループを指定します。ファイルのパスは、絶対パスまたは $ORACLE_HOME 変数を基準とする相対パスにすることができます。「データ・ソース定義」パネルの「データベース・インスタンス・ディレクトリー」で、$ORACLE_HOME 変数の値を設定します。 パスの中で指定する「..」は、その前のパス部分とその後のパス部分の間に 1 つ以上のディレクトリーがあることを示します。パスの中で指定する「.+」は、その前のパス部分とその後のパス部分の間に 1 つだけディレクトリーがあることを示します。 例えば、$ORACLE_HOME/oradata/../*.dbf とします。これは、1 つの単一ファイル識別文字列であるファイル・パターンから多くの単一ファイル識別文字列を作成するための簡略表記になります。ファイル・パターンは、「/」で分離した一連の正規表現と見なせます。ファイルの絶対パスの各要素が順番どおりにその正規表現の 1 つと一致すれば、そのファイルが一致したということになります。 パターンの要素が環境変数である場合、突き合わせの前に環境変数が展開されます。 パターンのいずれかの要素が「..」である場合、ゼロ個以上のディレクトリー・レベルと一致することになります。 例えば、/usr/local/../foo は、/usr/local/foo や /usr/local/gunk/junk/bunk/foo と一致します。 1 つのファイル・パターンに複数の「..」要素を使用する必要はありませんし、使用すべきではありません。そのようにすると、パターンを展開するのに非常に時間がかかってしまうからです。混乱を避けるため、正規表現では Windows で使用される可能性のある区切り文字として ¥ を使用することはできません。前述のファイル・パターンは正しくありません。*.dbf は有効な正規表現ではないからです。この場合は .*dbf とする必要があります。

さらに、デフォルトの Guardium Unix/Oracle テンプレート・セットには、以下のテンプレートが含まれています。

ADMIN_RESTRICTIONS がオン

このテストは、listener.ora パラメーター ADMIN_RESTRICTIONS が適切に設定されているかどうかをモニターします。

ファイル所有権

このテストは、Oracle データ・ファイル、ログ、実行可能ファイルなどのファイル所有権およびその変更をモニターします。

ファイルの許可

このテストは、Oracle データ・ファイル、ログ、実行可能ファイルなどのファイル・アクセス許可およびその変更をモニターします。

ログ・ファイルに対するエラーのスキャン

このテストは、Oracle ログ・ファイルにエラー文字列が出現するかどうかをスキャンします。

SPOOLMAIN.LOG が存在しない

このテストは、Oracle SPOOLMAIN.LOG が存在するかどうか検査します。

CAS テンプレート - MongoDB

MongoDB は、非リレーショナル形式のデータ (JSON 文書など) のプログラミングが容易であるため、運用システムや Web アプリケーションのバックエンドとして使用されることが一般的です。

Unix/MongoDB テンプレートを使用すると、データ・ソースに複数のパスや複数のディレクトリーを指定して、MongoDB データ・ソース定義で指定された各種コンポーネントをスキャンすることができます。

ファイル・パターンをスキャンするには、「$」で始まるテンプレート項目を選択します。

$SCRIPTS/mongodb_unmask_value.sh 項目は選択しないでください。これは Guardium の予約項目です。

テンプレート項目が MongoDB データ・ソース定義で「データベース・インスタンス・ディレクトリー」の一部として指定されていない場合は、この項目はスキップオーバーされ、スキャンされません。

注: CAS スクリプトが機能するためには、Mongo DB サーバー上で MongoDB アカウントのログインを有効にする必要があります。ログインを有効にするには、root としてログインし、chsh mongod コマンドを実行し、新しいシェルを求めるプロンプトが出されたら `/bin/bash` と入力します。
注: 複数のファイル・パスを使用して、任意のタイプのデータ・ソースのテンプレートを独自に作成することができます。独自のテンプレートを作成する際には、Unix/MongoDB を参考にすることをお勧めします。MongoDB データ・ソースの新規テンプレートを作成する場合は、Unix/MongoDB テンプレートをコピーし、それに変更を加えることができます。
注: MongoDB データ・ソースは、SSL クライアント証明書を使用した SSL サーバーとクライアント/サーバーの接続をサポートしています。MongoDB 接続では、JDBC データベース接続ではなく Java ドライバーを使用します。
注: MongoDB クラスターの VA ソリューションは、複数の Mongo (レプリカ・セットの 1 次ノードとすべての 2 次ノード) 上で実行することができます。

CAS テンプレート - Netezza®

ファイル所有権

このテストは、CAS テンプレートの定義に従って、ファイルが正しいグループに所有され、所属しているかどうかを検査します。

ファイルの許可

このテストは、CAS テンプレートの定義に従って、ファイル・アクセス権が適切に設定されているかどうかを検査します。

ログ・ファイルに対するエラーのスキャン

このテストは、以下の 2 つのログ・ファイルに対してイベント (FATAL、ERROR、DEBUG、ABORT、および PANIC) があるかどうかを検査します。/nz/kit/log/postgres/pg.log および /nz/kit/log/startupsvr/startupsvr.log

Oracle RAC システムの場合の構成

Oracle RAC システムの場合に必要な構成は以下のとおりです。
S-TAP でインストールされた、各ノード上の guard_tap.ini を以下のように変更します。
unix_domain_socket_marker=<key>
<key> 値は IPC プロトコル定義内の listener.ora に見つかります。
例 1:
listener.ora 内の記述が以下のようになっている場合、
LISTENER=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=ORCL))))
以下のパラメーターをそれに従って変更します。
unix_domain_socket_marker=ORCL
例 2:
listener.ora 内に複数の IPC 行がある場合、すべてのキーの共通項を使用します。LISTENER=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER)))) LISTENER_SCAN1=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER_SCAN1)))) LISTENER_SCAN2=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER_SCAN2)))) LISTENER_SCAN3=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER_SCAN3))))
Guardium はパスで文字列検索を使用するので、「LISTENER」を指定すれば、 上記の 4 つすべてに有効です。この場合、以下のように使用します。
unix_domain_socket_marker=LISTENER

CAS テンプレート - PostgreSQL

注: PostgreSQL_BIN および PostgreSQL_DATA 環境変数が正しく定義されていることは非常に重要です。設定が無効である場合、他の CAS 評価テストが正しく作動しなくなるか、まったく作動しなくなります。

ファイル所有権

このテストは、CAS テンプレートの定義に従って、ファイルが正しいグループに所有され、所属しているかどうかを検査します。

ファイルの許可

このテストは、CAS テンプレートの定義に従って、ファイル・アクセス権が適切に設定されているかどうかを検査します。

PostgreSQL_BIN 環境変数が定義済み

このテストは、データベース・サーバーで $PostgreSQL_BIN 環境変数が定義されているかどうかを検査します。この変数は、Unix/Linux の root アカウントの下に定義する必要があります。あるいは、root ログイン用の .profile に追加することができます。Windows OS の場合は、管理者ログインに対してこれを定義する必要があります。Red Hat Linux の場合、PostgreSQL BIN フォルダーは通常は /usr/bin 内にあります。Solaris の場合、通常これは /data/postgres/postgres/8.3-community/bin/64 などのようになります。この環境変数の設定は非常に重要です。他の評価テストはこのフォルダーのロケーションに依存するためです。

PostgreSQL_DATA 環境変数が定義済み

このテストは、データベース・サーバーで $PostgreSQL_DATA 環境変数が定義されているかどうかを検査します。この変数は、Unix/Linux の root アカウントの下に定義する必要があります。あるいは、root ログイン用の .profile に追加することができます。Windows OS の場合は、管理者ログインに対してこれを定義する必要があります。Red Hat Linux の場合、DATA フォルダーのデフォルトは通常は /var/lib/pgsql/data 内にあります。 Solaris の場合、一定のロケーションはありません。 この環境変数の設定は非常に重要です。他の評価テストは、正しい構成ファイルを検出するためにこのフォルダーのロケーションに依存するためです。

CAS テンプレート - SQL Server

OS スクリプト

実行する OS スクリプトを指定します。 スクリプトからの出力は、Guardium データベースに格納されます。 これは、実行可能なシェル・スクリプトまたはバッチ・スクリプトか、あるいはコマンドに入力できるコマンドのセットのいずれかです。

レジストリー変数

セキュリティー・アセスメント・テストで必要な特定のキー値を Windows レジストリーから検索します。

CAS テンプレート - Sybase

OS スクリプト

実行する OS スクリプトを指定します。CAS ホーム・ディレクトリーの下のスクリプト・ディレクトリーを示す変数 $SCRIPTS で始まり、実行するスクリプト (例えば、$HOME/sybase_sysdevice_type_test.sh) を示している必要があります。スクリプト自体も CAS $SCRIPTS ディレクトリーにある必要があります。 スクリプトからの出力は、Guardium データベースに格納され、セキュリティー・アセスメントに使用されます。 これは、実行可能なシェル・スクリプトまたはバッチ・スクリプトか、あるいはコマンド行に入力できるコマンドのセットのいずれかです。 Java の構文解析は変更される可能性が高いため、最も単純な種類のコマンド以外については直接実行する

のではなく、スクリプトに配置することを推奨します。 UNIX では、スクリプトは指定した OS ユーザーの環境で実行されます。 ユーザーがスクリプトを記述するために使用できる実行環境用に、3 つの環境変数が定義されます。$UCAS はデータベース・ユーザー名、$PCAS はデータベース・パスワード、$ICAS はデータベース・インスタンス名です。 Windows の場合、これらの 3 つの値はバッチ・ファイル実行時に最後の 3 つの引数として付加されます。 例えば、OS スクリプト・テンプレート「%SCRIPTS%¥MyScript.bat my-arg1 my-arg2」を使う場合、%3、%4、および %5 はそれぞれ DB ユーザー名、パスワード、およびインスタンス名になります。

ファイル

セキュリティー・アセスメントでトラッキングとモニターの対象にするファイルを指定します。 ファイルのパスは、絶対パスまたは $SYBASE 変数を基準とする相対パスにすることができます。 $SYBASE 変数の値は「データ・ソース定義」パネルの「データベース・インスタンス・ディレクトリー」フィールドで設定した値です。 これは単一のファイルの名前と見なされます。OS ユーザー環境の環境変数をファイル名の中で使うことができ、展開して使用されます。例えば、$HOME/START.sh は Sybase ユーザーのホーム・ディレクトリーにある始動スクリプトの名前です。

ファイル・パターン

セキュリティー・アセスメントでトラッキングとモニターの対象にするファイルのグループを指定します。 ファイルのパスは、絶対パスまたは $SYBASE 変数を基準とする相対パスにすることができます。 $SYBASE 変数の値は「データ・ソース定義」パネルの「データベース・インスタンス・ディレクトリー」フィールドで設定した値です。パスの中で指定する「..」は、その前のパス部分とその後のパス部分の間に 1 つ以上のディレクトリーがあることを示します。パスの中で指定する「.+」は、その前のパス部分とその後のパス部分の間に 1 つだけディレクトリーがあることを示します。 例えば、「$SYBASE/../.*dat」という指定は、1 つの単一ファイル識別文字列であるファイル・パターンから多くの単一ファイル識別文字列を作成するための簡略表記になります。ファイル・パターンは、「/」で分離した一連の正規表現と見なせます。ファイルの絶対パスの各要素が順番どおりにその正規表現の 1 つと一致すれば、そのファイルが一致したということになります。 パターンの要素が環境変数である場合、突き合わせの前に環境変数が展開されます。 パターンのいずれかの要素が「..」である場合、ゼロ個以上のディレクトリー・レベルと一致することになります。 例えば、/usr/local/../foo は、/usr/local/foo や /usr/local/gunk/junk/bunk/foo と一致します。 1 つのファイル・パターンに複数の「..」要素を使用する必要はありませんし、使用すべきではありません。そのようにすると、パターンを展開するのに非常に時間がかかってしまうからです。混乱を避けるため、正規表現では Windows で使用される可能性のある区切り文字として ¥ を使用することはできません。

さらに、「Guardium Unix/Sybase のアセスメント: UNIX - SYBASE」セットには、以下のテンプレートが含まれています。

ログ・ファイルに対するエラーのスキャン

このテストは、Sybase ログ・ファイルにエラーがあるかどうかをモニターします。

sysdevice 所有者が sysbase

このテストは、sysdevice の所有権をモニターします。

ファイル所有権

このテストは、Sybase ファイルのファイル所有権およびその変更をモニターします。

ファイルの許可

このテストは、Sybase ファイルのファイル・アクセス許可およびその変更をモニターします。

CAS テンプレート - Teradata

ファイル所有権

このテストは、CAS テンプレートの定義に従って、ファイルが正しいグループに所有され、所属しているかどうかを検査します。

ファイルの許可

このテストは、CAS テンプレートの定義に従って、ファイル・アクセス権が適切に設定されているかどうかを検査します。

Aster データ

Aster Data は 2011 年に Teradata によって買収されました。Aster Data は一般的に、データウェアハウジングおよび分析アプリケーション (OLAP) に使用されます。Aster Data は、構造化照会言語 (SQL) を MapReduce 内で使用できるようにする SQL-MapReduce と呼ばれるフレームワークを構築しました。Aster Data で最もよく連想されるのは、クリック・ストリーム系のアプリケーションです。

Aster nCluster には、クイーン・ノード・グループ、ワーカー・ノード・グループ、およびローダー・ノード・グループが含まれています。CAS エージェントは、3 つすべてのノード・グループにインストールされます。

クイーン・ノードですべてのテストを実行するには、セキュリティー・アセスメントを作成する必要があります。Aster Data 用のデータベース接続はすべて、クイーン・ノードのみを通過します。

ワーカー・ノードとローダー・ノードでテストが必要となるのは、CAS テスト (ファイル許可とファイル所有権) を実行する場合のみです。

特権テストは、所定のインスタンスに含まれるすべてのデータベースをループします。

CAS アクセスを必要とする脆弱性評価テストを実行し、CAS データ・ソース構成の選択を入力する際には、データベース・インスタンス・アカウントで Aster のインストールに使用したユーザー名を指定してください。このユーザー名は通常、beehive と呼ばれます。

データベース・インスタンス・ディレクトリーでは、これが beehive ユーザーのホーム・ディレクトリーとなります。デフォルトは通常、/home/beehive です。

CAS を使用しない脆弱性評価テストを実行する場合は、ユーザーがクラスター内のクイーン・ノードを指定して、独自のデータ・ソースを作成する必要があります。

CAS に依存する脆弱性評価テストを実行する際に、テスト対象ノードがワーカーのいずれかである場合は、クイーン・ノードを指すようにデータ・ソース内の「カスタム URL」を設定する必要があります (これは listen 方法を示します)。

例

ホスト名/IP = Worker.guard.xxx.xxx..com または 1xx.1xx.111.111 (ワーカーがこれを listen していない場合でも、これが実際のワーカー・ホストになります。CAS はこのワーカーのノードからデータを送受信できるため、これが必要となります。)

ポート = 2046 または使用される任意のポート

データベース = beehive

カスタム URL = jdbc:ncluster://aster6q:2406/beehive (この JDBC の例は、実際にはポート 2406 および beehive データベース上のクイーン・ノードである aster6q に接続することを示しています。)

データベース・インスタンス・アカウント = beehive

データベース・インスタンス・ディレクトリー = /home/beehive

親トピック: 構成監査システム (CAS)

CAS テンプレートの処理

このセクションでは、CAS テンプレートの維持方法について説明します。

テンプレート/テンプレート・セットの定義

- 新しいテンプレート・セットの作成
- テンプレート・セットの変更
- テンプレート・セットのコピー作成
- テンプレート・セットの削除

新しいテンプレート・セットの作成

1. 「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「CAS テンプレート・セットの構成」をクリックして、「CAS 構成ナビゲーター」を開きます。
2. 「新規」をクリックして、「モニター項目テンプレート定義」パネルを開きます。
3. OS タイプを選択します。
4. データベース・タイプを選択します。特定のデータベース・タイプがテンプレート・セットで必要とされない場合は、「データベース・タイプ」として N_A を選択します。
5. テンプレート・セット名として固有の名前を入力します。
   注: 128 文字を超えるテンプレート・セット名は切り捨てられます。
6. 「適用」をクリックして、CAS テンプレート・セット定義を保存します。
7. 新しいテンプレート・セットに項目を追加するには、「セットに追加」をクリックします。『テンプレート・セット項目の定義』を参照してください。

Guardium® CAS パネルを見つける

デフォルトでは、CAS 構成機能へのアクセス権限は admin ユーザー、および CAS ロールを割り当てられたユーザーに限定されます。

「強化」をクリックします。CAS 機能のリストは「構成変更制御 (CAS アプリケーション)」ヘッダー内に表示されます。

CAS 構成ナビゲーターを開く

「CAS 構成ナビゲーター」パネルから操作を開始して、CAS テンプレート・セットを作成または変更することができます。

「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「CAS テンプレート・セットの構成」をクリックして、「CAS 構成ナビゲーター」パネルを開きます。

リストは OS タイプやデータベース・タイプでフィルタリングすることができます。

テンプレート・セットの変更

既存の CAS テンプレート・セットを変更するには、「CAS 構成ナビゲーター」パネルを使用します。いずれかの CAS ホストでテンプレート・セットが既に使用中の場合、そのテンプレート・セットで変更できる点は限られています。定義のいくつかの要素を少し変更することはできますが、テンプレートの追加/削除はできません。

1. 「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「CAS テンプレート・セットの構成」をクリックして、「CAS 構成ナビゲーター」パネルを開きます。
2. テンプレート・セット・リストを OS タイプまたはデータベース・タイプでフィルターに掛けます。
3. 変更するテンプレート・セットを選択して「変更」をクリックすると、「CAS テンプレート・セット定義」パネルが開きます。
4. 必要な変更を加え、「適用」をクリックして変更内容を保存します。

テンプレート・セットのコピー作成

1. 「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「CAS テンプレート・セットの構成」をクリックして、「CAS 構成ナビゲーター」パネルを開きます。
2. テンプレート・セット・リストを OS タイプまたはデータベース・タイプでフィルターに掛けます。
3. コピーするテンプレート・セットを選択して「コピー」をクリックすると、「CAS テンプレート・セット定義」パネルが開きます。
4. コピーが作成されたら、必要に応じてコピーに変更を加えます。

注: 事前定義テンプレートは編集できません。これには CAS ホストで使用されているものと同じ制約事項があります。お客様が変更を加える場合は、そのコピーを作成し、作成したコピーを編集する必要があります。

テンプレート・セットの削除

1. 「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「CAS テンプレート・セットの構成」をクリックして、「CAS 構成ナビゲーター」パネルを開きます。
2. テンプレート・セット・リストを OS タイプまたはデータベース・タイプでフィルターに掛けます。

3. 削除するテンプレート・セットを選択して、「削除」をクリックします。

テンプレート・セット項目の定義

いずれかの CAS ホストでテンプレート・セットが既に使用中の場合、そのテンプレート・セットで変更できる点は限られています。定義のいくつかの要素を少し変更することはできますが、テンプレートの追加/削除はできません。

- 新しいテンプレート・セット項目の作成
- テンプレート・セット項目の変更
- テンプレート・セット項目の削除

新しいテンプレート・セット項目の作成

1. 「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「CAS テンプレート・セットの構成」をクリックして、「CAS 構成ナビゲーター」パネルを開きます。
2. 「新規」をクリックして、「モニター項目テンプレート定義」パネルを開きます。
3. テンプレート・セット名を入力し、OS タイプとデータベース・タイプを選択して、「適用」をクリックします。
4. 「セットに追加」をクリックし、新規項目を作成します。

テンプレート・セット項目の変更

1. 「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「CAS テンプレート・セットの構成」をクリックして、「CAS 構成ナビゲーター」パネルを開きます。
2. テンプレート・セット・リストを OS タイプまたはデータベース・タイプでフィルターに掛けます。
3. 変更するテンプレート・セットを選択して「変更」をクリックすると、「CAS テンプレート・セット定義」パネルが開きます。
4. 変更する項目を選択し、「選択したものを編集」をクリックします。必要な変更を加え、「適用」をクリックして変更内容を保存します。

テンプレート・セット項目の削除

1. 「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「CAS テンプレート・セットの構成」をクリックして、「CAS 構成ナビゲーター」パネルを開きます。
2. テンプレート・セット・リストを OS タイプまたはデータベース・タイプでフィルターに掛けます。
3. 変更するテンプレート・セットを選択して「変更」をクリックすると、「CAS テンプレート・セット定義」パネルが開きます。
4. 削除する項目を選択し、「選択したものを削除」をクリックします。

CAS アイテム・テンプレート定義パネル

コンポーネント	記述
OS タイプ	オペレーティング・システムの種類 (Windows または Unix)。 テンプレート・セットが空の場合にはこの選択を変更できますが、テンプレート・セットに 1 つ以上の項目が含まれる場合は変更できません。
データベース・タイプ	データベースの種類 (Oracle、MS-Sql、DB2®、Sybase、Informix® など)、またはオペレーティング・システム・テンプレート・セットの場合は N/A。テンプレート・セットが空の場合にはこの選択を変更できますが、テンプレート・セットに 1 つ以上の項目が含まれる場合は変更できません。
記述	レポートで使われる、項目を示すオプションの名前。他の CAS パネル (例えば「CAS テンプレート・セット定義」) の中で項目を識別します。省略した場合、項目名のデフォルトとして、(タイプに応じて) ファイル名またはパターン、変数名、またはスクリプトが使用されます。
タイプ	以下のいずれか 1 つ: SQL 照会、OS スクリプト、環境変数、レジストリー変数、レジストリー変数パターン、ファイル、ファイル・パターン 詳しくは『テンプレートと監査のタイプ』を参照してください。 注: CAS に基づく評価テストと共に使われる場合は、OS スクリプト・タイプでなければなりません。
内容	モニターする特定の項目を定義するタイプ依存テキストや、それを生成する方法。 詳しくは『テンプレートと監査のタイプ』を参照してください。 注: OS スクリプトの場合、CAS はスクリプトが完了するのを待ちます。OS スクリプトの実行時間を制限し、CAS がスクリプトを強制終了できるようにするには、cas_command_wait という guard_tap.ini ファイル中のパラメーターを使用します。デフォルトの待機時間は 300 秒つまり 5 分です。このパラメーターを変更するとき、CAS を再始動する必要はありません。
アクセス権の制限	ファイルおよびファイル・パターン・タイプのみ。 Unix のみで使用 - このファイルに関して、超過してはならないアクセス権 (許可) の数
ファイル所有者	ファイルおよびファイル・パターン・タイプのみ。ファイルの所有者。
ファイル・	ファイルおよびファイル・パターン・タイプのみ。ファイルのグループ所有者。

グループ	
期間	各テスト実行間の最大間隔。分数 (m)、時間数 (h)、または日数 (d) で指定します。最初の期間が始まった後、次の期間が始まるまで、データが使用可能になります。
データを保持	選択した場合、実データのコピーがそれぞれの変更点と共に保存されます。例えばファイル項目の場合、そのファイルのコピーが保存されます。選択した場合でも、項目の生データのサイズが (この CAS ホストについて構成された)「生データ制限」を超えると、データは保存されません。
MD5 を使用	MD5 アルゴリズムを使って生データのチェックサムを計算することにより、追加的な比較を行うかどうかを示します。大きな文字オブジェクトの場合、MD5 チェックサムの計算にかなり時間がかかります。しかし、これは変更の標識として、単なるサイズよりも優れています。デフォルトでは MD5 が使用されません。MD5 を使用しても、生データのサイズが CAS ホスト用に構成された 「MD5 サイズ制限」を上回る場合は、MD5 による計算と比較はスキップされます。
有効	デフォルトで選択済みです。項目の変更を検査するかどうかを示します。

テンプレートと監査のタイプ

タイプ	記述
SQL 照会	内容は有効な SQL ステートメントでなければなりません。ステートメントによって戻される結果は、前回の照会実行で戻された結果と比較されます。使用されるデータ・ソースで指定されたパラメーターを使って照会が実行されます (ユーザー名、パスワード、データベース・ポートなど)。照会の結果を戻せないという障害を防ぐために、データ・ソースでこれらのパラメーターを設定するときには注意が必要です。
OS スクリプト	内容として、有効なコマンド行エントリー、または OS 実行可能スクリプトが入っているファイル名が可能です。スクリプトは、データ・ソース定義の「データベース・インスタンス・アカウント」フィールドで指定されている OS ユーザーの環境で実行されます。
環境変数	内容は、データ・ソース定義の「データベース・インスタンス・アカウント」フィールドで指定されている OS ユーザーのコンテキストで定義された、環境変数を指定する必要があります。
レジストリー変数	内容は、ホストの Windows レジストリー内の変数のパスとして解釈されます。そのパスで検出される値は、前回のパスのトレースで検出された値と比較されます。
レジストリー変数パターン	内容は、Windows レジストリーのパスの構成要素と突き合わせるために使用される一連の正規表現です。パターンは、(前述の説明のように扱われる) レジストリー変数タイプのモニター項目を作成するために使用されます。 複数の正規表現は / によって結合され、1 つのレジストリー・パスに似たパターンとなります。より一般的な ¥ 文字は Java™ 正規表現の構文における特殊文字であるため、使用できません。いずれかの正規表現の中で / を使用する必要がある場合は ¥ を使用してエスケープしなければなりません (例えば U/235 に一致させるには U¥/235 を使用します)。 パターン .. を使用すると、パス内のゼロ個以上の構成要素に一致させることができます。例えば、HKLM/Software/../buzz は HKLM¥Software¥buzz と HKLM¥Software¥one¥two¥three¥buzz のどちらとも一致します。 このタイプのパターンは処理負荷の高いレジストリー検索になる可能性があるため、慎重に使用してください。 これらの例外を除いて、正規表現は Java 正規表現の構文に従います。
ファイル	内容は、ホスト上の絶対ファイル・パスとして解釈されます。このパスで検出されるファイルの特性は、パスが最後にトレースされたときに検出された特性と比較されます。環境変数をパスに含めることができます。環境変数は、データ・ソースで指定されている OS ユーザーのコンテキストで展開されます。また、("$SYBASE_HOME" のような) 置換変数をパスの先頭で使用することもできます。置換変数は、データ・ソース定義の「データベース・インスタンス・ディレクトリー」フィールドで入力された値に置換されます。
ファイル・パターン	内容は一連の正規表現です。これは、ファイル・パスの構成要素と突き合わせて、ファイル・タイプのモニター項目を生成するために使用されます。複数の正規表現は / によって結合され、実際のファイル・パスに似たパターンとなります。正規表現の構文のために、レジストリー・パターンの場合と同様、Windows ファイルには ¥ を使用できません。パターンが *?:* で始まる場合、Windows マシンでは、複数ドライブ・マシンの各ドライブでパターン・マッチングが開始されます。レジストリー・パターンで説明された「..」構造は、ファイル・パターンでも (慎重に) 使用することができます。OS ユーザーのコンテキストからの環境変数をファイル・パターンで使用できます。環境変数は、正規表現の展開の前に展開されます。

GuardAPI コマンド

create_cas_template_set

create_cas_template

create_datasource

create_cas_host_instance

親トピック: 構成監査システム (CAS)

CAS ホスト

構成監査システム (CAS) のホスト構成では、1 つ以上の CAS インスタンスが定義されます。

1 つ以上の CAS テンプレート・セットを定義して CAS をデータベース・サーバーにインストールすると、CAS をそのホスト上で構成する準備ができた状態になります。CAS ホスト構成は 1 つ以上の CAS インスタンスを定義します。各 CAS インスタンスは 1 つの CAS テンプレート・セットを指定し、データベースへの接続に必要なパラメーターをすべて定義します。CAS がインストールされているデータベース・サーバーごとに 1 つの CAS ホスト構成があり、通常は複数の CAS インスタンスがそこに含ま

れます (例えばオペレーティング・システム項目をモニターする 1 つの CAS インスタンスと、個々のデータベース・インスタンスをモニターする追加的な複数の CAS インスタンス)。

- CAS インスタンスの定義
- CAS インスタンスの変更
- CAS インスタンスの削除
- CAS インスタンスの無効化

CAS インスタンスの定義

1. 「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「CAS ホスト構成」をクリックして、「CAS 構成ナビゲーター」を開きます。

   メニューには、CAS がインストールされているすべてのデータベース・サーバーがリストされ、このホストが Guardiumに接続済みです。

2. リストのフィルタリングを使用して、OS タイプまたはデータベース・タイプでフィルターに掛け、処理したいホストを見つけることができます。
3. 変更するホストを強調表示し、「変更」をクリックします。
4. メニューからテンプレート・セットを選択します。
   注: CAS インスタンスは、ホストがオフライン状態である場合、またはホストの 2 次 Guardium システムである場合には、定義することはできません。
5. 「データ・ソースの追加」をクリックして「データ・ソース・ファインダー」パネルを開きます。
   注: このホストで、このテンプレート・セット用の互換データ・ソースが存在しない場合は、「新規」をクリックして「データ・ソース定義」パネルを開き、データ・ソースを追加することができます。
6. テンプレート・セットに追加するデータ・ソースを選択し、「追加」をクリックしてテンプレート・セットに追加します。

Guardium® CAS パネルを見つける

CAS 構成機能へのアクセス権限は admin ユーザー、および CAS ロールを割り当てられたユーザーに限定されます。

「強化」をクリックします。CAS 機能はすべて「構成変更制御 (CAS アプリケーション)」ヘッダー内にリストされます。

CAS 構成ナビゲーターを開く

「CAS 構成ナビゲーター」パネルから操作を開始して、CAS ホストを作成または変更することができます。

「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「CAS ホスト構成」をクリックして、「CAS 構成ナビゲーター」パネルを開きます。

CAS インスタンスの変更

1. CAS 構成ナビゲーターを開きます。
2. リストのフィルタリングを使用して、OS タイプまたはデータベース・タイプでフィルターに掛け、処理したいインスタンスを見つけることができます。
3. 変更するホストを強調表示し、「変更」をクリックします。

   選択したホストに関連付けられた定義済み CAS インスタンスがリストされて、それと共に以下の情報と編集オプションが表示されます。

表 1. CAS インスタンスの変更

コンポーネント	記述
「インスタンスの無効化/有効化」アイコン	「インスタンスの無効化」アイコンをクリックすると、CAS インスタンスが無効/有効になります
「インスタンスの削除」アイコン	「インスタンスの削除」アイコンをクリックすると、CAS インスタンスが削除されます
データ・ソース	インスタンスによって使われるデータ・ソースを識別します。「**データ・ソース**」をクリックすると「データ・ソース定義」パネルが開いて、データ・ソース定義を編集できます
テンプレート・セット	インスタンスによって使われる CAS テンプレート・セットを識別します。このリンクをクリックすると「モニター項目テンプレート定義」パネルが開いて、テンプレート・セット定義を表示または変更できます。 詳しくは、CAS テンプレートの処理を参照してください
モニター項目	インスタンスによって現在モニターされている項目の数。このリンクをクリックすると「モニター項目定義」パネルが開いて、現在モニターされている全項目のリストが表示されます 詳しくは、『モニター項目リストの表示』を参照してください。 注: 定義されたモニター項目の数にかかわらず、デフォルトでは 10,000 個のモニター項目をレポートで表示可能です。モニター項目の数がこの制限に近づいた場合には、複数のインスタンスを定義することをお勧めします。

CAS インスタンスの削除

1. CAS 構成ナビゲーターを開きます。
2. リストのフィルタリングを使用して、OS タイプまたはデータベース・タイプでフィルターに掛け、処理したいインスタンスを見つけることができます。
3. 「インスタンスの削除」をクリックして、CAS インスタンスを削除します。収集されたすべての変更データも削除されます。

CAS インスタンスの無効化

1. CAS 構成ナビゲーターを開きます。
2. リストのフィルタリングを使用して、OS タイプまたはデータベース・タイプでフィルターに掛け、処理したいインスタンスを見つけることができます。
3. 変更するホストを強調表示して「変更」をクリックするか、ダブルクリックすると、「ホスト・インスタンス定義」パネルが開きます。

4. 「インスタンスの無効化」アイコンをクリックすると、CAS インスタンスが無効になります。このアイコンを再びクリックしてインスタンスを有効にするまでは、変更データは収集されません。

モニター項目リストの表示

「ホスト・インスタンス定義」パネルで「モニター項目」リンクをクリックすると、モニターされる項目の詳細リストが「モニター項目定義」パネルに表示されます。以下の表では、このホスト構成に関する「モニター項目定義」パネルに表示されるコンポーネントについて説明します。

モニターされるすべての項目は、生データ、ホスト上の文字オブジェクト、SQL 照会の結果、OS スクリプトの出力、またはファイルの内容を参照します。その文字オブジェクトのサイズが計算されます。項目がファイルである場合、許可、所有者、グループ、最終変更時間もまた検査されます。項目が最後に検査された時点と比べて、これらのいずれかが変更されている場合には、変更が記録されます。

表 2. モニター 項目リストの表示

コンポーネント	記述
選択ボックス	モニター項目を個々に、またはグループとして編集するには、選択ボックスにチェック・マークを付けます。 いずれかのモニター項目をダブルクリックすると、その項目を編集できます。
項目	「CAS アイテム・テンプレート定義」パネルに記述されたモニター項目の名前
タイプ	OS スクリプト、SQL 照会、ファイル、環境変数、またはレジストリー変数のいずれか 1 つ OS スクリプトまたは SQL スクリプト: オペレーティング・システム・スクリプトまたは SQL スクリプトの実際のテキストまたはパス。この出力が、次回の実行時に生成される出力と比較されます。 ファイルまたはファイル・パターン: 1 つの特定のファイル、または複数ファイルのセットを識別するためのパターン 環境変数またはレジストリー変数: 環境変数または (Windows) レジストリー変数
期間	テスト実行の平均間隔。秒数、分数、時間数、または日数で指定します。
データを保持	マークを付けた場合、実データのコピーがそれぞれの変更点と共に保存されます。例えばファイル項目の場合、そのファイルのコピーが保存されます。マークを付けた場合でも、項目の生データのサイズが (この CAS ホストに関して構成された)「生データ制限」を超えると、データは保存されません。
MD5 を使用	MD5 アルゴリズムを使って生データのチェックサムを計算して比較を行うかどうかを示します。大きな文字オブジェクトの場合、MD5 チェックサムの計算にかなり時間がかかります。しかし、これは変更の標識として、単なるサイズよりも優れています。デフォルトでは MD5 が使用されません。MD5 を使用しても、生データのサイズが CAS ホスト用に構成された「MD5 サイズ制限」を上回る場合は、MD5 による計算と比較はスキップされます。

GuardAPI コマンド

delete_cas_host

list_cas_hosts

create_cas_host_instance

delete_cas_host_instance

list_cas_host_instances

update_cas_host_instance

親トピック: 構成監査システム (CAS)

CAS レポート

このセクションでは、構成監査システム (CAS) のレポート作成について説明します。

admin ユーザーは、すべてのクエリー・ビルダーおよびデフォルト・レポートに対するアクセス権限を持ちます。 admin ロールでは、デフォルト CAS レポートにアクセス可能ですが、 CAS クエリー・ビルダーにはアクセスできません。 CAS ロールでは、デフォルト CAS レポートとクエリー・ビルダーの両方にアクセス可能です。

照会 - レポート・ビルダーへのアクセス

照会 - レポート・ビルダーの使用を参照してください。

CAS ドメインについては、ドメイン、エンティティー、および属性で説明しています。

CAS 事前定義レポートについては、事前定義管理レポートで説明しています。

デフォルト CAS レポートへのアクセス

CAS 関連のデフォルト・レポートを表示するには、「強化」 > 「レポート」をクリックします。

親トピック: 構成監査システム (CAS)

CAS 状況

「管理」 > 「変更モニター」 > 「CAS 状況」をクリックして、「構成監査システム状況」を開きます。

CAS がインストールされ、実行されており、かつ、この Guardium システムがアクティブな Guardium® ホストとして構成されているデータベース・サーバーごとに、このパネルに CAS 状況と、そのデータベース・サーバーに構成されている各 CAS インスタンスの状況が表示されます。

状況標識ライトの色を区別できない場合は、状況ライト上にマウスを移動すると、テキスト・ボックスに現在の状況が表示されます。

コンポーネント	記述
CAS システム状況標識ライト	このパネルに表示されるライトは、CAS が Guardium システム上でアクティブに稼働しているかどうかを示します。 赤: CAS はこの Guardium システムで実行されていません。 緑: この Guardium システム上の CAS はアクティブです。
CAS エージェント状況標識ライト	これらの状況ライトは、個々の CAS エージェントが Guardium システムに接続されているかどうかを示します。各 CAS エージェントを特定するには、状況標識ライトの行の前に表示される IP アドレスを参照します。 **赤**: ホストおよび/または CAS エージェントはオフラインか到達不能です。 **緑**: ホストおよび CAS エージェントはオンラインです。 **黄色:** Guardium システムは 2 次 CAS ホストです。
リセット	このモニター対象システム上の CAS エージェントをリセットします。これにより、データベース・サーバー上の CAS エージェントが停止し、再始動します。 注: これによりチェックポイント・ファイルもリセットされるため、最初からやり直すことができ、ファイルは最初から再スキャンされます。
削除 (X)	このモニター対象システムを CAS から削除し、また、CAS クライアントに関連付けられていた Guardium システム上のデータも削除します。 このボタンは、このシステム上で CAS エージェントが実行中の場合は無効になっています。 削除するには、CAS エージェントを停止する必要があります。詳しくは、『CAS エージェントの停止および始動』を参照してください。
赤/黄/緑のライト	各ライト・セットはモニター対象システムの 1 つの CAS インスタンスの状況を示します。 所有するモニター対象システムの状況が赤 (CAS エージェントがオフラインであることを示す) である場合は、この状況ライト・セットは無視してください。 **赤**: インスタンスは使用不可です。 **緑**: インスタンスは使用可能でオンラインです。また、その構成は Guardium システムの構成と同期されています。 **黄色**: インスタンスは使用可能ですが、Guardium システム上のインスタンス構成がモニター対象システム上のインスタンス構成と一致しません (インスタンスは Guardium システム上で更新されたが、その更新がモニター対象システムに適用されていない)。
リフレッシュ	「リフレッシュ」をクリックすると、リスト内のすべてのサーバーの状況が再確認されます。このボタンはデータベース・サーバー上の CAS を停止および/または再始動するためのものではありません。Guardium システム上の CAS と各データベース・サーバー上の CAS との間の接続を検査するだけです。

注: guard_tap.ini ファイルへの TAP_IP 入力が必要です。TAP_IP が欠落していると CAS は始動せず、CAS クライアントのログ・ファイルにエラー・メッセージが記録されます。

CAS エージェントの停止および始動

ある種の状況においては、モニター対象システム上で CAS エージェントを停止または始動しなければならない場合があります。
注: CAS エージェントを停止して再始動する場合は、「管理」 > 「変更モニター」 > 「CAS 状況」をクリックします。

UNIX ホスト上での CAS の停止

1. /etc/inittab ファイルを編集します。
2. CAS の respawn 行を見つけます。

```
cas:2345:respawn:/usr/local/guardium/guard_stap/cas/bin/run_wrapper.sh /usr/local/guardium/guard_stap/cas/bin
```

3. 先頭文字の位置に「#」を挿入して、この行をコメント化します。
4. ファイルを保存します。
5. コマンド init -q を入力します。
6. コマンド ps -er | grep cas を入力します。
7. リストされているそれぞれのプロセスの PID をメモします。
8. リストされているプロセスごとに、コマンド kill -9 <pid> を実行します。
9. Guardium 管理者ポータルの「構成監査システム状況」パネルで、この CAS ホストの状況ライトが赤になっている必要があり、また、「削除」ボタンが有効になっている必要があります。これにより、Guardium システムの内部データベースのこの CAS ホストからデータを削除できます。

UNIX ホスト上での CAS の始動

CAS が、前述のように /etc/inittab ファイルを編集して停止された場合にのみ、次の手順に従って CAS エージェントを再始動します。

1. /etc/inittab ファイルを編集します。
2. 次の行を見つけます。

```
#cas:2345:respawn:/usr/local/guardium/guard_stap/cas/bin/run_wrapper.sh /usr/local/guardium/guard_stap/cas/bin
```

3. 例 (ステップ 2.) で、先頭文字の位置の # を削除して、この行をアンコメントします。オペレーティング・システムによって、コメント文字は異なる場合があります。
4. ファイルを保存します。
5. コマンド init -q を入力して CAS エージェントを再始動します。

Windows ホスト上での CAS の始動および停止

Windows では CAS はシステム・サービスとして実行されます。

1. 「サービス」パネルで、構成監査システムのクライアント項目を強調表示します。
2. 「操作」メニューから「始動」 または「停止」を選択します。

親トピック: 構成監査システム (CAS)

Guardium システムの構成

ビジネス目標を効果的かつ効率的に達成できるようにするために、Guardium システムのいくつかの側面を構成できます。

- システム構成
  「システム構成」パネル上のほとんどの情報は、インストール時に CLI を使用して設定されます。
- 検査エンジン構成
  検査エンジンは 1 つ以上のサーバーからなるサーバー・セットと、1 つ以上のクライアントからなるクライアント・セットとの間の、特定のデータベース・プロトコル (Oracle や Sybase など) を使用したトラフィックをモニターします。
- ポータル構成
  Guardium® アプライアンスの Web サーバーは、デフォルト・ポート (8443) のままにしておくことも、ポータルを再設定することもできます。デフォルト・ポートのご使用を強くお勧めします。
- TLS のバージョンの管理
  すべてのアプライアンス、S-TAP エージェント、CAS クライアント、および GIM クライアントで TLS 1.0/1.1 を無効にして TLS 1.2 を有効にすることができます。
- 新規レイアウトの生成
- 認証の構成
  デフォルトでは、Guardium ユーザー・ログインは他のアプリケーションから独立して Guardium によって認証されます。
- グローバル・プロファイル
  「グローバル・プロファイル」パネルでは、すべてのユーザーに適用されるデフォルトを定義します。
- アラート機能の構成
  アラート機能を構成してアクティブ化するまでは、E メール・メッセージ、SNMP トラップ、アラート関連 Syslog メッセージはまったく送信されません。
- 異常検出
  異常検出プロセスは、アラートの照会に基づいて相関アラート通知を作成して保存する (ただし送信はしない) ためにポーリング間隔ごとに実行されます。
- セッション推論
  セッション推論は、指定された期間にわたって非アクティブ状態が続いている開いたセッションがあるかどうか検査し、それらにクローズ済みのマークを付けます。
- Guardium への S-TAP 接続のブロック (S-TAP 認証)
  この機能を使用して、クライアントが Guardium システムへのアクセスを許可されている特定の S-TAP ホストを制御します。
- IP からホスト名への別名割り当て
  IP からホスト名への別名割り当て機能は、ドメイン・ネーム・システム (DNS) サーバーにアクセスして、クライアントおよびサーバーの IP アドレスのホスト名別名を定義します。
- システム・バックアップ
  システム・バックアップ機能を使用すると、オンデマンドまたはスケジュールに基づいて実行可能なバックアップ操作を定義できます。
- ソケット接続権限の構成
  このトピックは、カスタム・アラート・クラスに適用されます。

システム構成

「システム構成」パネル上のほとんどの情報は、インストール時に CLI を使用して設定されます。

システムを構成する方法、またはその他のシステム構成設定を変更する方法については、『システム構成の変更』を参照してください。

アプライアンス内の各種機能を使用するためには、有効なライセンスがなければなりません。システムが始動してからライセンスを入力した場合、GUI を再始動する必要があります。

システム共有パスワードについて

Guardium® 管理者は、「システム構成」でシステム共有パスワードを定義します。システム共有パスワードには、以下の 2 つの一般的な用途があります。

- アーカイブ/エクスポート・アクティビティーによってアプライアンスからエクスポートされたファイルを暗号化する
- 中央マネージャーと管理対象ユニットの間のセキュア通信を確立する

「一元管理」または「統合」(あるいはその両方) を使用している場合、関連するすべてのシステムの「システム共有パスワード」を同じ値に設定する必要があります。

システム共有パスワードの値は、インストール時は NULL です。 企業のセキュリティー・プラクティスに応じて、システム共有パスワードを定期的に変更する必要が生じることがあります。 各アプライアンスは、そのアプライアンスで定義されているすべての共有パスワードの履歴レコードが入っている共有パスワード・ファイルを保守します。 したがって、同じシステムが後日、そのシステムで暗号化された情報を暗号化解除する場合に問題が発生することはありません。

あるシステムから情報がエクスポートまたはアーカイブされ、別のシステムにインポートまたはリストアされた場合、後者のシステムは、前者のシステムで使用された共有パスワードへのアクセス権限を持っている必要があります。 このような場合のために、ある Guardium システムからシステム共有パスワードをエクスポートし、それらを別のシステムにインポートする目的で使用できる CLI コマンドがあります。

CLI 付録で以下のコマンドを参照してください。

- aggregator backup keys file
- aggregator restore keys file

システム構成の変更

1. 「設定」 > 「ツールとビュー」 > 「システム」をクリックして、「システム構成」を開きます。

2. 変更を加えます。
3. 「適用」をクリックして、更新されたシステム構成を保存します。

注: 適用した変更は、Guardium システムを再始動するまで有効になりません。構成変更を適用した後で、「再始動」をクリックし、システムを停止してから再始動します。

表 1. システム構成パネル・リファレンス

フィールドまたはコントロール	記述
固有グローバル ID	この値は、データの照合と統合に使用します。 デフォルト値は、マシンの MAC アドレスから派生した固有値です。 この値は、システムがモニター操作を開始した後は変更しないでください。
システム共有パスワード	ここで入力した値は表示されません。 入力した文字はマスクされます。 システム共有パスワードはアーカイブ/リストア操作、および一元管理/統合操作で使用されます。これを使用する場合、互いに通信するすべてのユニットの間でその値が同じでなければなりません。インストール時にはこの値は NULL で、時間の経過とともに変化する可能性があります。 システム共有パスワードは次のような場合に使用されます。 • 中央マネージャーと管理対象ユニットの間でセキュア接続が確立されるとき。 • 統合されるユニットが、アグリゲーターにエクスポートされるデータに署名して暗号化するとき。 • いずれかのユニットが、アーカイブ用のデータに署名して暗号化するとき。 • アグリゲーターが、統合されるユニットからデータをインポートするとき。 • いずれかのユニットがアーカイブ・データをリストアするとき。 企業のセキュリティー・プラクティスに応じて、システム共有パスワードを時々変更する必要が生じることがあります。共有パスワードは変更される可能性があるため、各システムは、そのシステムで定義されているすべての共有パスワードの履歴レコードが入っている共有パスワード・ファイルを保守します。これにより、古い共有パスワードを使ってシステムからエクスポート (またはアーカイブ) されたファイルを、新しく置換された同じ共有パスワードを持つシステムでインポート (またはリストア) することができます。 **注意**: 使用する場合は、必ず共有パスワードの値を安全なロケーションに保存するようにしてください。 この値が失われると、アーカイブされたデータにアクセスできなくなります。
パスワードの再入力	システム共有パスワードを入力または変更する場合は、新しい値を再入力します。ここで入力した値は表示されません。入力した文字は、すべてアスタリスクで表示されます。
ライセンス・キー	ライセンス・キーは、インストール時に構成に挿入されます。このフィールドは、技術サポートからの指示がある場合を除き、変更しないでください。 オプションのコンポーネントを追加する場合は、新規プロダクト・キーをここに貼り付けなければならないことがあります。 一元管理ユニットで新規プロダクト・キーをインストールする場合は、「適用」をクリックすると、「警告: 一元管理ユニットのライセンスを変更するには、すべての管理対象ユニットの情報をリフレッシュする必要があります。」という警告メッセージを受け取ります。新規プロダクト・キーをインストールするには、「OK」をクリックしてメッセージ・ウィンドウを閉じた後で、「適用」を再度クリックする必要があります。データは正常に保存されました。というメッセージを受け取ることによって、新規ライセンスがインストールされたことを確認できます。 新規プロダクト・キーを一元管理ユニットにインストールする場合に、CM に適用されたライセンスをその管理対象ユニット上でリフレッシュする必要があるという警告を受け取ることがあります。 この場合は、中央マネージャーからリフレッシュを実行する必要があります。それには、中央マネージャーから、リストされている各コレクターのリフレッシュ・アイコンを押します。 ライセンスは、製品およびそれに対応する機能へのアクセス権をユーザーに付与します。 ライセンスは、追加することもオーバーライドすることもできます。 アクティブ・ライセンスは、ADMINCONSOLE_PARAMETER の LICENSE_KEY に格納されます。 製品タイプ: DAM、FAM、VA 製品タイプのエディション: Express、Standard、Advanced
データ・ソースの数	制限付きライセンスが適用されている場合、データ・ソース・ライセンスごとに許容されるデータ・ソースの最大数が表示されます。
残計量スキャン数	制限付きライセンスが適用されている場合、計量ライセンスごとに許容される脆弱性評価スキャンの数 (データ・ソース計量) が表示されます。脆弱性評価がトリガーされるたびに、このスキャン・カウンターが 1 つずつ減少します。
ライセンス有効期限:	制限付きライセンスが適用されている場合、ライセンスが無効になることが確定している日付が表示されます。
ライセンスの数	この値は、残っているライセンスの数を示します。
注: ネットワーク・アドレス、2 次管理インターフェース、およびルーティングの設定は、CLI を使用して構成します。	これらの設定は、GUI を使用して構成することはできないため、「システム構成」ユーザー・インターフェースではグレー化して表示されます。
システム・ホスト名	Guardium システムの解決可能なホスト名。この名前は、1 次システム IP アドレスの DNS ホスト名に一致している必要があります。
ドメイン	Guardium システムがある DNS ドメインの名前。
システム IP アドレス	ユーザーと S-TAP® または CAS エージェントが Guardium システムへの接続に使用する 1 次 IP アドレス。これは、ETH0 というラベルの付いたネットワーク・インターフェースに割り当てられます。
サブネット・マスク	1 次システム IP アドレスのサブネット・マスク。

フィールドまたはコントロール	記述
ハードウェア (MAC) アドレス	1 次ネットワーク・インターフェースの MAC アドレス。

システム IP アドレス (2 次)	オプション: 高可用性フェイルオーバー IP チーミングを提供するために、1 つのポートを 1 次インターフェースと組み合わせて構成することもできます。
	
	または、デバイス上のポートを 2 次管理インターフェースとして、1 次とは異なる IP アドレス、ネットワーク・マスク、およびゲートウェイを使用して構成することもできます。
	
	これら 2 つのオプションを同時に使用することはできません。
	
	2 次管理接続の種類には、次の 2 つがあります。これらは同時には使用できません。どちらも同じ CLI コマンドに対するオプションによって制御されます。
	
	結合 (チーム化)
	eth0 と、もう 1 つの指定されたネットワーク・インターフェース・カード (NIC) を、スタンバイ・フェイルオーバー機能を備えた結合ペアとします。このオプションを実装するには、store network interface high-availability on <nic> という CLI コマンドを使用します。ここで nic は、使用可能な NIC です。
	2 次インターフェース
	Guardium システム内の別の NIC から、GUI および CLI にアクセスできるようにします。このオプションを実行するには、store network interface secondary on <nic> <ip> <mask> <gateway> という CLI コマンドを使用して、2 次 NIC、その IP アドレス、およびネットワーク・マスクを指定し、必要に応じてゲートウェイを指定します。
	
	物理および VM システムの両方で同じ機能が提供されます。 この機能は、Guardium システムまたは VM に取り付けられている NIC の数によって異なります。
	
	ユニットにインストールされているネットワーク・インターフェースを表示するには、**show network interface inventory** CLI コマンドを使用します。 例:
	
	`show network interface inventory`
	`Current network card configuration:`
	`Device | Mac Address |Member of`
	`----------`
	`eth0 | 00:50:56:3b:c3:73 |`
	`eth1 | 00:50:56:8a:0d:fa |`
	`eth2 | 00:50:56:8a:0d:fb |`
	`eth3 | 00:50:56:8a:00:c1 |`
	
	注: 「Member of」は、結合が存在する場合に、どの NIC が結合ペアであるかを示します。
	
	アプライアンス上の eth コネクターを見つけるには、**show network interface port** CLI コマンドを使用します。 これにより、そのポート上でオレンジ色のライトが 20 回明滅します。 例:
	
	guard14.xyz.com> sho net int port 3
	
	これで、オレンジ色のライトがポート eth5 上で 20 回明滅します。
	
	注: 2 次 IP アドレスとそれに関連付けられたポートは、1 次接続の IP チーム化を介してフェイルオーバー・サポートを提供する高可用性フィーチャーとは無関係です。 高可用性オプションについて詳しくは、CLI 付録の『**store network interface** コマンド』を参照してください。

サブネット・マスク (2 次)	オプション。2 次システム IP アドレスのサブネット・マスク。
デフォルト経路 / 2 次経路	システムのデフォルト・ルーターの IP アドレス / 2 次ルーターの IP アドレス
1 次リゾルバー 2 次リゾルバー 3 次リゾルバー	1 次リゾルバーの IP アドレス (DNS) は必須です。 2 次と 3 次はオプションです。
接続のテスト	対応する DNS (ドメイン・ネーム・システム) サーバーへの接続をテストするには、「接続のテスト」をクリックします。これは単に、指定されたホストのポート 53 (DNS) にアクセスできることを検査するだけです。 機能している DNS サーバーであることの検証はしません。 DNS サーバーが応答したかどうかを示すメッセージ・ボックスを受け取ります。
停止	システムをシャットダウンするには、「停止」をクリックします。
再始動	システムを停止してから再始動するには、「再始動」をクリックします。アクションの確認を求めるプロンプトが出されます。
適用	変更内容を保存するには、「適用」をクリックします。変更内容は、次回にシステムを再始動したときに適用されます。

親トピック: Guardium システムの構成

検査エンジン構成

検査エンジンは 1 つ以上のサーバーからなるサーバー・セットと、1 つ以上のクライアントからなるクライアント・セットとの間の、特定のデータベース・プロトコル (Oracle や Sybase など) を使用したトラフィックをモニターします。

検査エンジンはネットワーク・パケットから SQL を抜き出し、センテンス、要求、コマンド、オブジェクト、およびフィールドを識別する構文解析ツリーをコンパイルして、そのトラフィックについての詳細情報を内部データベースに記録します。

Guardium アプライアンス上で複数の検査エンジンを構成し、開始したり停止したりすることができます。

検査エンジンは中央マネージャー・ユニット上で定義したり実行したりすることはできません。 ただし、管理対象ユニット上の検査エンジンを、中央マネージャー制御パネルから開始および停止することができます。

また、検査エンジンは S-TAP® 上でも定義されます。S-TAP がこの Guardium® アプライアンスにレポートを送る場合は、アプライアンスが S-TAP と同じトラフィックをモニターしないように気をつけてください。 そのような状況が発生すると、分析エンジンが重複するパケットを受け取り、メッセージを再構成できず、そのトラフィックを

無視することになります。

IP アドレスの選択

各検査エンジンは 1 つ以上のクライアントおよびサーバーの IP アドレス間のトラフィックをモニターします。 検査エンジンの定義において、これらは IP アドレスおよびマスクを使用して定義されます。IP アドレスを単一のロケーションと考え、マスクを一定範囲の IP アドレスを定義できるワイルドカードの手段と考えることができます。

IP アドレスの形式は n.n.n.n で、それぞれの n は 0 から 255 の範囲の 8 ビットの数字 (オクテットと呼ぶ) です。

例えば、ご使用の PC の IP アドレスが 192.168.1.3 だとします。このアドレスは例で使用されます。これらは 2 進数なので、最後のオクテット (3) は 00000011 と表記されます。

マスクは IP アドレスと同じ形式 (n.n.n.n) で指定されます。 マスクの任意のビット位置にあるゼロは、ワイルドカードの意味になります。したがって、マスク 255.255.255.240 と IP アドレス 192.168.1.3 を組み合わせると、最後のオクテットが 0 から 15 のすべての値に一致します。これは、値 240 は 2 進数で 11110000 だからです。しかし最初の 3 つのオクテットでは値 192.168.1 にのみ一致します。これは 255 は 2 進数で表すとすべて 1 である (つまり、最初の 3 つのオクテットにはワイルドカードが適用されない) からです。

2 進数によるマスクの指定は多少紛らわしいかもしれません。しかし便宜上、IP アドレスは通常、階層的にグループ化されており、あるカテゴリー (デスクトップ・コンピューターなど) のすべてのアドレスは最後の 2 つのオクテットのいずれかでグループ化されます。したがって、実際にマスクで最もよく目にする数字は 255 (ワイルドカードなし) または 0 (すべて) となります。

このように、マスク 255.255.255.255 (ゼロ・ビットを持たない) は、IP アドレスにより指定される単一のアドレス (例では 192.168.1.3) のみを識別します。

または、マスク 255.255.255.0 を同じ IP アドレスと組み合わせると、192.168.1 で始まるすべての IP アドレスが一致します。

すべてのアドレスの選択

すべての IP アドレスを示すのにしばしば利用される IP アドレスの 0.0.0.0 は、Guardium では許可されません。IP アドレスとマスクの組み合わせですべての IP アドレスを選択するには、任意のゼロ以外の IP アドレスにすべてがゼロのマスクを続けます (例えば 1.1.1.1/0.0.0.0 となります)。ただし、 0.0.0.0/0.0.0.0 は有効な組み合わせです。

すべての検査エンジンに適用される設定の構成

1. 「管理」 > 「アクティビティー・モニター」 > 「検査エンジン」をクリックして、「検査エンジン構成」を開きます。
2. 表を参照して、必要な変更を行います。
3. 変更が終わったら「適用」をクリックして、更新したシステム構成を保存します。
4. 必要に応じて、検査エンジンにコメントを追加します。
5. 「検査エンジンの再始動」をクリックします。

注: 適用された変更は、検査エンジンが再始動するまで有効になりません。検査エンジンの構成変更を適用した後、「再始動」ボタンをクリックしてシステムを停止し、(新しい構成設定を使用して) 再始動します。
注: HTTP サポートについては、検査エンジン構成に制限があります。 次の検査エンジン設定は HTTP ではサポートされません: 「デフォルトで値をキャプチャー」、「デフォルトで自動コミットをマーク」、「順序付けをロギング」、「例外 SQL 文字列をロギング」、「影響を受けるレコードをロギング」、「平均応答時間を計算」、「戻りデータの検査」、「空セッションを記録」。

表 1. すべての検査エンジンに適用される設定

コントロール	記述
デフォルトで値をキャプチャー	デフォルト値は false です。リプレイ関数によって、トランザクションとキャプチャー値を区別するために使用されます。準備済みステートメントがある場合は、割り当てられた値がキャプチャーおよびリプレイされます。キャプチャーされた準備済みステートメントを準備済みステートメントとしてリプレイする場合は、キャプチャーされたデータのチェック・ボックスにチェック・マークが付いている必要があります。
デフォルトで自動コミットをマーク	デフォルト値は true です。さまざまなデータベースには各種の自動コミット・モデルがあるため、この値は、当該トランザクションと各コマンドの後の自動コミットに明示的にマークを付けるためにリプレイ関数により使用されます。 注: チェック・ボックスにチェック・マークが付いている場合、コミットとロールバックは無視されます。現在サポートされているデータベースには、DB2®、Informix®、Oracle などがあります。
順序付けをロギング	マークが付いている場合、直前の SQL ステートメントと現在の SQL ステートメントのレコードが作成されます (ただし、前回の構成が十分短期間内に発生していることが条件となります)。
例外 SQL 文字列をロギング	マークが付いている場合、例外が記録されるときに、SQL ステートメント全体が記録されます。

コントロール	記述

影響を受けるレコードをロギング	「影響されるレコード」- SQL ステートメントを実行するたびに影響を受けるレコードの数を示す結果セット。
	
	マークが付いている場合は、各 SQL ステートメント (該当する場合) で影響を受けるレコードの数が記録されます。「影響を受けるレコードをロギング」のデフォルト値は、FALSE (0) です。
	
	注: JDBC を使用している場合、Oracle バインド変数トラフィックを正しくロギングするためにはこれに必ずマークを付ける必要があります。
	注: 「影響されるレコード」オプションは、スニファーに対して、追加の応答パケットを処理し、影響を受けたデータ (バッファー・サイズを増やし、スニファー全体のパフォーマンスに悪影響を及ぼす可能性があるデータ) のロギングを延期するように要求するスニファー操作です。 大きな影響を受けるのは、非常に大きい応答からです。この操作に関連する大量のオーバーヘッドを避けるために、Guardium はデフォルトのしきい値セットを使用して、しきい値を超えたときに、処理操作をスキップすることをスニファーが決定できるようにします。
	注: 通常、「影響されるレコード」は、ユーザーが「影響を受けるレコードをロギング」をオンにしたときに (「検査エンジン」 > 「影響を受けるレコードをロギング」)、正しく設定されます。ただし、ストアード・プロシージャーを通じて MS-SQL を使用する場合は、「影響されるレコード」が -1 に設定されます。
	
	細分度のレベルを設定するには、構成および制御 CLI コマンドで store max_results_set_size、store max_result_set_packet_size、および store max_tds_response_packets について参照してください。
	
	結果セットの値の例は次のとおりです。
	
	• ケース 1、「影響されるレコード」値: 正数。これは、結果セットの正しいサイズを表します。
	• ケース 2、「影響されるレコード」値: -2。これは、レコード数が構成可能な限度 (CLI コマンドによって調整可能) を超えたことを示します。
	• ケース 3、「影響されるレコード」値: -1。これは、Guardium によってサポートされないパケット構成のケースを示します。
	• ケース 4、「影響されるレコード」値: -2。結果セットがストリーム・モードで送信される場合。
	• ケース 5、「影響されるレコード」値: -2 未満。現在の値についてユーザーを更新するためのレコード・カウント中の中間結果。最終的には、レコードの合計を示す正数になります。例えば、サーバーが 4 つのパケットで 1000 件のレコードを返す場合、
	  ○ パケット #1 250
	  ○ パケット #2 200
	  ○ パケット #3 250
	  ○ パケット #4 200
	
	影響されるレコードは、次のとおり報告されます。
	
	  ○ パケット #1 -250
	  ○ パケット #2 -500
	  ○ パケット #3 -750
	  ○ パケット #4 1000
	
	注: 「影響されるレコード」機能は、ストリーム・モードを使用して結果を送信する場合は、Db2 ではサポートされません。

平均応答時間を計算	これをマークすると、ロギングされる各 SQL 構文について平均応答時間が計算されます。
戻りデータの検査	マークを付けると、SQL 要求から返されるデータが検査され、Ingress 数と Egress 数が更新されます。 セキュリティー・ポリシーでルールが使用される場合は、このチェック・ボックスにマークを付ける必要があります。
空セッションを記録	これをマークすると、SQL ステートメントを含まないセッションが記録されます。マークを付けないと、これらのセッションは無視されます。
XML の構文解析	検査エンジンは通常、XML トラフィックの構文解析を実行しません。このチェック・ボックスにマークを付けると、XML トラフィックの構文解析が実行されるようになります。
ロギング単位	ログ単位の分数 (1、2、5、10、15、30、または 60)。 レポートで要求される場合、Guardium は要求データをこの細分度で要約します。 例えば、ログ細分度が 60 の場合、ある要求が指定した 1 時間に n 回発生したことになります。 チェック・ボックスがマークされていない場合、その時間内でコマンドが起こった正確な時間は記録されません。しかし、ポリシーのルールが要求によってトリガーされると、リアルタイム・アラートにより、正確な時刻を示すことができます。 ポリシーの例外ルールを定義するときに、それらのルールはログ単位にも適用されます。 例えば、1 時間に 5 回のログイン失敗を無視させるが、6 回目のログイン失敗時にアラートを送信させる場合などが考えられます。
戻りデータ当たりの最大ヒット数	戻りデータが検査されるときに、いくつのヒット (ポリシー・ルール違反) が記録されるかを示します。
無視ポート・リスト	無視するポートのリストです。 データベース・サーバーが非データベース・プロトコルを処理しており、Guardium に非データベース・トラフィックの分析でサイクルを無駄にさせたくない場合は、このリストに値を追加します。例えば、データベースのあるホストがポート 80 で HTTP サーバーも実行していることがわかっている場合は、無視ポート・リストに 80 を追加して、Guardium がこれらのストリームを処理しないようにすることができます。 値を複数入れる場合にはコンマで区切り、ポートの範囲をその値も含めて指定する場合は、値をハイフンでつなぎます。 例: 101、105、110-223
バッファー・フリー: n %	表示のみ。 **n** は、検査エンジンの処理に使用できるバッファー・スペースの空きのパーセントです。 この値は、ウィンドウが最新表示されるたびに更新されます。すべての検査エンジンを駆動する単一の検査エンジン・プロセスがあります。 これは、その処理で使用されるバッファーです。
検査エンジンの再始動	「検査エンジンの再始動」をクリックして、すべての検査エンジンを停止して再始動します。
コメントの追加	「コメント」をクリックして、検査エンジン構成にコメントを追加します。

コントロール	記述
適用	「適用」をクリックして、構成を保存します。 注: 行ったすべての一括変更 (かつ「適用」ボタンを使用して保存したもの) は、検査エンジンを再開するまで有効になりません。ただし、個々の検査エンジンの属性 (除外、シーケンスの配列など) は直ちに有効になります。

検査エンジンの作成

1. 「管理」 > 「アクティビティー・モニター」 > 「検査エンジン」をクリックして、「検査エンジン」を開きます。
2. 「検査エンジンの追加」をクリックして、パネルを展開します。
3. 「名前」ボックスに名前を入力します。 この名前はアプライアンスで固有でなければなりません。名前には文字と数字のみを使用することをお勧めします。特殊文字を使うと、CLI を介してこの検査エンジンを操作できなくなるからです。
4. 「プロトコル」ボックスから、モニター対象のプロトコル (Windows: CouchDB, DB2, DB2 Exit, Informix, MongoDB, MS SQL, Mysql, Oracle, PostgreSQL, Sybase; UNIX: Aster, Cassandra, CouchDB, DB2, DB2 Exit, exclude IE, FTP, GreenPlumDB, Hadoop, Hive, HTTP, Hue, IBM iSeries, Impala, Informix, Informix Exit, Kerberos, MariaDB, MongoDB, Mysql, Netezza, Oracle, PostgreSQL, SAP HANAVertics, Sybase, Teradata, Vertica, or WebHDFS) またはキーワード「IE を除外」のいずれかを選択します。指定したクライアントとサーバー間のすべてのトラフィックを無視する場合は、IE を除外を選択します。
   注: 「IE を除外」はポートに対してのみ作動します。IP には関係しません。無視するポートの範囲を入力します。このポートで特定の IP を除外する場合は、作成した検査エンジンで「除外データベース・クライアント IP」を使用できます。特定のポート範囲についてパケットを選出する必要がない場合、タイプ「IE を除外」(IGNORE) の別個の検査エンジンを定義します。このエンジンで定義する必要がある値は、PORT_RANGE_START と PORT_RANGE_END のみです。例えば、ポート範囲 1024-65535 を使用して包括的なすべての Oracle 検査エンジンが定義されている状況で、特定のポートを除外する必要がある場合、この種類の除外処理が必要になります。Oracle for Windows を使用する場合は、ポート範囲を 1000 から 65535 に拡大してください。
   注: GreenPlum データベースから IPC トラフィックを送信した場合、これは Guardium システムで PostgresSQL トラフィックとしてログに記録されます。GreenPlum データベースから TCP トラフィックを送信した場合、これは検査エンジンで GreenPlum データベースとしてログに記録されます。TCP トラフィックの場合、Guardium はポートによって (GreenPlum のポートは、ポート 5432) データベースを判別します。Guardium システムは、IPC トラフィックについては名前付きパイプを使用し、GreenPlum データベースについては、PostgresSQL をデータベースの名前として使用します。PostgresSQL と Greenplum データベースの両方が同じシステム上にあるとき、それぞれの IPC トラフィックは、guard_tap.ini ファイルに設定されている最初の PostgresSQL または Greenplum データベース IE に応じて、DB_PROTOCOL に記録されます。
5. 「データベース・クライアント IP/マスク」ボックスに、モニター対象のクライアントのリスト (データベース接続が開始されたクライアント・ホスト) を入力します (または、「除外データベース・クライアント IP」がマークされている場合は、除外するクライアントのリストを入力します)。各クライアントは IP アドレスおよびサブネット・マスクで識別されます。概要には、これらのフィールドの使用法に関する詳しい説明があります。

   正符号をクリックして、追加の IP アドレスおよびサブネット・マスクを追加します。負符号をクリックすると、最後の IP アドレスとサブネット・マスクが削除されます。

6. 「データベース・サーバー IP/マスク」ボックスに、モニター対象のデータベース・サーバー (データベースがある場所) のリストを入力します。各サーバーは IP アドレスおよびサブネット・マスクで識別されます。 概要には、これらのフィールドの使用法に関する詳しい説明があります。

   正符号をクリックして、追加の IP アドレスおよびサブネット・マスクを追加します。負符号をクリックすると、最後の IP アドレスとサブネット・マスクが削除されます。

7. 「ポート」ボックスに、指定したクライアントとデータベース・サーバー間のトラフィックをモニターするのに使用する単一ポートまたはポートの範囲を入力します。 たいていの場合、これは単一ポートです。

   警告: 広いポート範囲を入力しないでください。適正なポートのみを含めるようにしてください。データベース・トラフィックを送信しないポート上のトラフィックや環境に関係ないトラフィックの分析を試行することにより、検査エンジンの速度が低下することがあります。

8. 開始時にこの検査エンジンを自動的に始動させる場合は、「始動時にアクティブ」ボックスにマークを付けます。
9. 「データベース・クライアント IP/マスク」リストにリストされたクライアントを除くすべてのクライアントのトラフィックを検査エンジンにモニターさせる場合は、「除外データベース・クライアント IP」ボックスにマークを付けます。このオプションと「無視」プロトコルの選択との違いを正しく理解してください。このオプションでは、IP アドレスからのトラフィックを除くすべてのトラフィックが含まれます。その他のすべてのクライアントを含めずに、特定のクライアントのセットを無視するには、それらのクライアント用に別の検査エンジンを定義し、「無視」プロトコルを使用します。
10. 「追加」をクリックして、定義を保存します。
11. 必要に応じて、検査エンジン・リストの検査エンジンの位置を変更します。 検査エンジンで定義したフィルター処理機構が、順番に実行されます。 必要な場合は、定義の枠にある「Up」ボタンまたは「Down」ボタンを使用して、新しい検査エンジン構成の位置や、既存の構成の位置を変更します。
12. 必要に応じて、「開始」をクリックして、ここで構成した検査エンジンを開始します。 検査エンジンが開始されると、「始動」ボタンが「停止」ボタンに置き換わります。
13. 注: TAP_IDENTIFIER の値を指定したときに、その値にスペースが含まれている場合は、Guardium によってそのスペースが自動的にハイフンに置き換えられます。例えば、"Sample description" という値は "Sample-description" になります。

検査エンジンの開始または停止

「管理」 > 「アクティビティー・モニター」 > 「検査エンジン」をクリックして、「検査エンジン」を開きます。検査エンジンを開始するには、「開始」をクリックします。 検査エンジンを停止するには、「停止」をクリックします。

検査エンジンの削除

検査エンジンを使用しなくなった場合は、検査エンジンを誤って再開しないように、定義を削除することをお勧めします。

1. 「管理」 > 「アクティビティー・モニター」 > 「検査エンジン」をクリックして、「検査エンジン」を開きます。
2. 削除する検査エンジンが停止していない場合は「停止」をクリックします。
3. 検査エンジンを削除するには、「削除」をクリックします。

親トピック: Guardium システムの構成

ポータル構成

Guardium® アプライアンスの Web サーバーは、デフォルト・ポート (8443) のままにしておくことも、ポータルを再設定することもできます。デフォルト・ポートのご使用を強くお勧めします。

1. 「設定」 > 「ツールとビュー」 > 「ポータル」をクリックして、「ポータル」を開きます。
2. 「始動時にアクティブ」チェック・ボックスがマークされていない場合はマークを付けます (このチェック・ボックスは無効にしないでください)。
3. 「HTTPS ポート」を 1025 から 65535 までの整数値に設定します。
4. 「適用」をクリックして、値を保存します。(Guardium セキュリティー・ポータルは、再始動するまではこのポートでの listen を開始しません。)あるいは「元に戻す」をクリックして、最後の「適用」操作で保管した値をリストアします。
5. 変更を保存したら、「再始動」をクリックして、Guardium Web サーバーを再始動します。これで、新しく割り当てたポート上のユニットに接続できます。
   注: 新しいポート番号で再始動したユニットに再接続するには、ブラウザーで Guardium ログイン・ページを開く際に使用する URL を変更する必要があります。

Guardium アプライアンスにログインするときにユーザー・パスワードが認証される方法を定義するには、Guardium ポータル構成を使用します。 3 つの選択肢があります。

それらの選択肢は、ローカル (Guardium のデフォルト)、RADIUS、LDAP です。

「設定」 > 「ツールとビュー」 > 「ポータル」の下のポータル構成画面は、以下の用途で使用されます。

1. ユーザー・パスワードを認証するための最善の方法を定義する。
2. 認証タイプをリセットするために GUI を再始動する。

ローカル接続は、特定のユーザーのパスワードがログインから定義される場合に機能します。 ログインは accessmgr ロールを使って定義されます。 デフォルトでは、accessmgr ロールを持つ **accessmgr** アカウントにログインします。 このロールによりユーザーは、ユーザー・アカウントを追加またはアップロードし、パスワードを作成することが可能になります。

accessmgr ロール・タイプを使ってユーザー名とパスワードを定義すると、Guardium アプライアンスにログインする際、ユーザーごとに定義済みのパスワードが使用されます。

RADIUS 接続では、Radius サーバーを通じたログイン認証が可能になります。パスワードと SecurID トークン番号の両方を使って Radius/RSA サーバーを定義できます。SecurID トークンの数値パスワードは、ハードウェア・トークンを介して表示されます。

Radius/RSA サーバーは Windows サーバー上で定義されます。また、セキュリティー RSA SecurID トークンが Radius サーバー上に定義され、保管されます。Radius ポータルを機能させるためにそれをダウンロードする必要はありません。

さらに、Unix プラットフォームを使って Radius サーバー接続を定義することができます。Radius は FreeRadius としても定義されます。 ユーザー・アカウントとパスワードが Radius サーバー上で定義されており、それらのダウンロードは必要ありません。 FreeRadius を使用するために、クライアント (Guardium サーバー)、ユーザー名、およびパスワードが FreeRadius Unix サーバー上で定義され、Radius ポータル接続の定義時に使用されます。

デフォルトのポータルは「ローカル」に設定されます。

LDAP 接続は、特定の LDAP サーバーでパスワードが定義され、保管されている場合に機能します。 ユーザーが LDAP ポータルを使ってログインするためには、まず **LDAP サーバー**からユーザー・アカウント名がインポートされる必要があります。accessmgr アカウントから使用可能な「ユーザー LDAP インポート」機能を使用して、LDAP ロケーションを定義した後、LDAP ユーザーをインポートします。パスワードをアップロードする必要はありません。

親トピック: Guardium システムの構成

TLS のバージョンの管理

すべてのアプライアンス、S-TAP エージェント、CAS クライアント、および GIM クライアントで TLS 1.0/1.1 を無効にして TLS 1.2 を有効にすることができます。

このタスクについて

この機能は、v10.1.4 で導入されました。

Guardium リリース v10.1.4 以降、Guardium システムのセキュリティーを強化するために、通信プロトコル TLS 1.0/1.1 をオプションで無効にすることができます。TLS 1.0/1.1 を無効にすると、結果的に TLS 1.2 プロトコルが有効になります。TLS 1.0/1.1 の使用時には通信の安全性が低くなる場合があります。

中央マネージャーまたはスタンドアロン・ユニット (あるいはその両方) から CLI を使用して TLS 1.0/1.1 を無効にする必要があります。この機能を有効にするには、お客様の Guardium アプライアンス、S-TAP エージェント、CAS クライアント、および GIM クライアントが特定のバージョンでなければなりません。

TLS 1.1 を無効にすると、管理対象ユニットと S-TAP が特定のバージョンであることが自動的に確認されますが、CAS クライアントのバージョンは確認できません。CAS を使用するお客様は、ご使用の CAS クライアントがバージョン 10.1.4 になっていて、それらのデータベース・サーバーで Java 7 が使用可能になっていることを確認する必要があります。この確認を行わないと、データベース・サーバーへの CAS 接続を確認できなくなります。

また、すべての管理対象ユニットにバージョン 10.1.4 がインストールされており、GIM クライアントと S-TAP が最小バージョン 10.1.2 であることも確認する必要があります。すべての要求を満たさないと TLS 1.0/1.1 は無効になりません。

管理対象環境のすべてのユニット (中央マネージャー、アグリゲーター、管理対象ユニット) で TLS 1.0/1.1 に関する情報を取得する場合、および TLS 1.0/1.1 を無効にする場合は、中央マネージャーで以下のコマンドを実行してください。

手順

1. CLI に admin としてアクセスします。
2. 次のコマンドを入力します。

```
grdapi get_secured_protocols_info
```

中央マネージャーからこのコマンドを実行すると、すべての管理対象ユニットに伝搬されます。システムは、有効なプロトコル (TLS 1.0/1.1 および TLS 1.2) を出力し、TLS 1.0/1.1 プロトコルを無効にできるかどうかを示します。エラー・コード 1000+ は、TLS 1.0/1.1 を無効にする前に管理者が対応する必要があるコンポーネントの問

題を示します。表示されるメッセージには、TLS 1.0/1.1 を無効にするための要件を満たしていないコンポーネントが示されます。警告メッセージは、オフラインまたは到達不能の管理対象ユニットに対して生成されます。オフラインのユニットは、オンラインに戻ったときに個別に管理する必要があります。

3. TLS 1.0/1.1 を無効にするには、次のように入力します。

```
grdapi disable_deprecated_protocols
```

中央マネージャーからこのコマンドを実行すると、すべての管理対象ユニットに伝搬されます。このコマンドは、まず上述のバージョン検査を実行します。無効化のための要件が満たされた場合、このコマンドは、中央マネージャーの各サービスおよびすべての管理対象ユニットの構成設定を変更します。無効化のための要件が満たされていない場合、システムは、非推奨のプロトコルが有効であり、すべての管理対象ユニットまたはコンポーネント (あるいはその両方) がアップグレードされるまで有効のままにする必要があることを示します。

4. admin ロールを持つ Guardium ユーザーは、非推奨のプロトコルを無効化する際にオフラインだったすべての管理対象ユニットに対して、それらの管理対象ユニット上で CLI セッションを手動で開始し、local_disable_depreciated_protocols を実行して構成変更を行う必要があります。

   ```
   grdapi local_disable_deprecated_protocols
   ```

5. TLS 1.0/1.1 に戻すには、次のように入力します。

   ```
   grdapi enable_deprecated_protocols all=true
   ```

   この GuardAPI コマンドは、構成設定を元に戻し、中央マネージャーのサービスとすべての管理対象ユニットを再始動して非推奨のプロトコルを有効にするフォールバックです。この GuardAPI コマンドは、中央マネージャーから all=true 引数を指定して実行し、中央マネージャーとすべての管理対象ユニットの非推奨のプロトコルを有効にすることができます。パラメーター all=true を指定しないと、非推奨のプロトコルは GuardAPI を実行するアプライアンスでのみ有効になります。
6. admin ロールを持つ Guardium ユーザーは、中央マネージャーおよび管理対象ユニット間の通信が安定して正常に動作していることを確認する必要があります。

親トピック: Guardium システムの構成

新規レイアウトの生成

ユーザー・レイアウトに基づいてロールの新規レイアウトを生成

Guardium® 管理者またはアクセス・マネージャーは、CLI を使用してロールのデフォルトのレイアウトを生成できます。レイアウトを生成すると、そのロールを割り当てられた新規ユーザーの初回ログイン後に、そのレイアウトが使用されます。

注: ユーザーおよびロールのデフォルトの .psml 構造は、admin ユーザーが GUI で定義できます。 詳しくは、『ポートレット・エディター』を参照してください。

generate-role-layout CLI コマンドを使用することにより、指定したユーザーのレイアウトによって既存のロール用の新規レイアウトを生成できます。 新規のロール用レイアウトが定義されると、そのロールを初めてのログイン前に割り当てられたユーザーは、そのロール用のレイアウトを受け取ります。

generate-role-layout

構文 generate-role-layout <user> <role>

注: ユーザー (ログイン名) とロールには大/小文字の区別がありません。

パラメーター

次のパラメーターのいずれかがスペースを含む場合 (ユーザーの John Doe またはロールの DBA Managers)、スペース文字を下線文字に置き換えてください。

例:

generate-role-layout John_Doe DBA_Managers

user - レイアウトがロール用レイアウトのモデルとして使用されるユーザーの名前。 ユーザーが存在しない場合は、「次のユーザーは存在しません: `<user> (No such user '<user>')`」というメッセージが表示されます。

role - 新規レイアウトの付加先のロール。

親トピック: Guardium システムの構成

認証の構成

デフォルトでは、Guardium® ユーザー・ログインは他のアプリケーションから独立して Guardium によって認証されます。

Guardium admin ユーザー・アカウントのログインは、常に Guardium だけによって認証されます。他のすべての Guardium ユーザー・アカウントの場合、RADIUS または LDAP を使用するよう認証を構成することができます。これら 2 つの場合、認証サーバーと接続するための追加的な構成情報が必要になります。
注: FreeRadius クライアント・ソフトウェアがサポートされます。

代替的な認証方式を使用する場合であっても、すべての Guardium ユーザーを Guardium アプライアンス上でユーザーとして定義する必要があります。他のアプリケーションによって実行されるのは認証だけです。

ユーザー・アカウントとロールは accessmgr ユーザーによって管理されますが、使用される認証方式は admin ユーザーによって管理されます。これは、職掌分散のための標準的なベスト・プラクティスです。

認証を構成する方法については、次のトピックを参照してください。

Guardium 認証の構成

1. 「設定」 > 「ツールとビュー」 > 「ポータル」をクリックして、「認証構成」を開きます。
2. 「認証構成」パネルで「Guardium」ラジオ・ボタンを選択します。

3. 「適用」をクリックします。

RADIUS 認証の構成

1. 「設定」 > 「ツールとビュー」 > 「ポータル」をクリックして、「認証構成」を開きます。
2. 「認証構成」パネルで「RADIUS」ラジオ・ボタンを選択します。追加のフィールドがパネルに表示されます。
3. 「プライマリー・サーバー」ボックスで、1 次 RADIUS サーバーのホスト名または IP アドレスを入力します。
4. オプションで、2 次および 3 次 RADIUS サーバーのホスト名または IP アドレスを入力します。
5. RADIUS によって使用される UDP ポート (1812 または 1645) を入力します。
6. RADIUS サーバーの「共有パスワード」を 2 度入力します。
7. 「タイムアウト秒数」を入力します (デフォルトは 120)。
8. 「認証タイプ」を次のように選択します。
   - PAP - パスワード認証プロトコル
   - CHAP - チャレンジ・ハンドシェーク認証プロトコル
   - MS-CHAPv2 - Microsoft チャレンジ・ハンドシェーク認証プロトコル (バージョン 2)
9. オプションで、「テスト」をクリックして構成を検証します。テストの結果が通知されます。なお、変更内容を保存するために「適用」ボタンをクリックしたときにも、常に構成がテストされます。
10. 「適用」をクリックします。Guardium はテスト・ユーザーの認証を試み、その結果を通知します。

LDAP 認証の構成

1. 「設定」 > 「ツールとビュー」 > 「ポータル」をクリックして、「認証構成」を開きます。
2. 「認証構成」で「LDAP」ラジオ・ボタンを選択します。
3. 「サーバー」ボックスで、LDAP サーバーのホスト名または IP アドレスを入力します。
4. 「ポート」番号を入力します (LDAP over SSL のデフォルトは 636 です)。
5. 「ユーザー RDN タイプ」(相対識別名タイプ) を入力します。デフォルトでは uid です。
   注:

   この属性は LDAP 認証用にユーザーを識別します。Access Manager が LDAP ユーザー・インポート操作を実行するため、ここで使われる属性を Access Manager に認識させる必要があります。LDAP ユーザーのインポートについて、詳しくは『LDAP ユーザー・インポート』ヘルプ・リンクをクリックしてください。

   RDN 値として SamAccountName を使用する場合、フルネームで a =search または =[domain name] のいずれかを使用する必要があります。

   例: SamAccountName=search、SamAccountName=dom

6. 「ユーザー基本 DN」(識別名) を入力します。
7. LDAP サーバーの必要に応じて「SSL を使用」チェック・ボックスにマークを付けるか、クリアします。
8. オプション。1 つ以上のトラステッド証明書を検査するには、「トラステッド証明書」をクリックして、パネルの指示に従います。
9. オプション。トラステッド証明書を追加するには、「トラステッド証明書の追加」をクリックして、パネルの指示に従います。
10. オプション。「テスト」をクリックして、構成を検証します。テストの結果が通知されます。なお、変更内容を保存するために「適用」をクリックしたときにも、常に構成がテストされます。
11. 「適用」をクリックします。Guardium はテスト・ユーザーの認証を試み、その結果を通知します。

親トピック: Guardium システムの構成

グローバル・プロファイル

「グローバル・プロファイル」パネルでは、すべてのユーザーに適用されるデフォルトを定義します。

デフォルト別名設定のオーバーライド

デフォルトでは、どの新規レポートにも、およびデフォルト・レイアウトに含まれるどのレポートにも、別名は使用されません。

別名は、特定の属性タイプの保管値に代わる同義語になります。通常は、データ値を意味のある、または分かりやすい名前で表示するために使用されます。 例えば、IP アドレス 192.168.2.18 の別名として、「財務サーバー」を定義することができます。

デフォルトで別名を表示させるには、次のようにして、すべてのレポートのデフォルト別名設定を変更できます。

- 「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」をクリックして、「グローバル・プロファイル」を開きます。
- 「特に指定されない限りレポートで別名を使用」チェック・ボックスにマークを付けます。
- 「適用」をクリックします。

PDF ページ・フッターのカスタマイズ

さまざまな Guardium® コンポーネント (監査タスクなど) によって作成される PDF ファイルには、標準のページ・フッターがあります。このフッターをカスタマイズするには、次のようにします。

1. 「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」をクリックして、「グローバル・プロファイル」を開きます。
2. 「PDF フッター・テキスト」フィールドで、各ページの下部に出力されるテキストを入力します。
   注: PDF フッター・テキストは、中央マネージャー/アグリゲーターから管理対象ユニットに配布されません。
3. 「適用」をクリックします。

アラート・メッセージ・テンプレートの編集

アラートの生成に使われるメッセージ・テンプレートをカスタマイズするには、次のようにします。

1. 「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」をクリックして、「グローバル・プロファイル」を開きます。

2. 「メッセージ・テンプレート」テキスト・ボックスで、アラート・テンプレート・テキストを編集します。

   「折り返しなし」チェック・ボックスにマークを付けると、メッセージ内の改行の位置を表示できます。

3. 完了したら、「適用」をクリックします。
4. 検査エンジンが再始動するまで、変更内容は有効になりません。これを直ちに有効にするには、「管理」 > 「アクティビティー・モニター」 > 「検査エンジン」をクリックして、「検査エンジン」を開きます。「検査エンジンの再始動」をクリックします。

表 1. アラート・メッセージ・テンプレートの変数

変数	記述
%%AppUserName	アプリケーション・ユーザー名
%%AuthorizationCode	許可コード
%%category	ルール定義でのカテゴリー
%%classification	ルール定義での分類
%%clientHostname	クライアント・ホスト名
%%clientIP	クライアントの IP アドレス
%%clientPort	クライアント・ポート番号
%%ConstructID	アラート・メッセージに関連付けられている SQL 要求の構造 ID
%%DBName	データベース名。
%%DBProtocol	データベース・プロトコル
%%DBProtocolVersion	データベース・プロトコル・バージョン
%%DBUser	データベース・ユーザー名
%%lastError	最後のエラーの記述 (例外ルールを起動する SQL エラー要求に、最後のエラーの記述フィールドが含まれる場合にのみ使用可能)
%%netProtocol	ネットワーク・プロトコル (Oracle 上の K-TAP では IPC または BEQ として表示されます)
%%OSUser	セッション情報 (GDM_ACCESS での OS_USER)
%%receiptTime	アラートの発生時間を表すタイム・スタンプ
%%receiptTimeMills	アラートの発生時間を表す数値 (固定日 1900 年 1 月 1 日からのミリ秒数)
%%RecordsAffected	影響を受けるレコード 重要: `%%RecordsAffected` 変数は、`syslog` 通知タイプが指定された「アラートのみ」ルール・アクションのメッセージ・テンプレートで使用される場合は、値を返しません。
%%requestType	要求タイプ
%%ruleDescription	ポリシー・ルール定義でのルールの記述
%%ruleID	ルール定義でのルール番号
%%serverHostname	サーバー・ホスト名
%%serverIP	サーバーの IP アドレス
%%serverPort	サーバー・ポート番号
%%serverType	データベース・サーバー・タイプ
%%serviceName	サービス名
%%sessionStart	セッション開始時間 (ログイン時間)
%%sessionStartMills	アラートが発生したセッションの開始時間を表す数値 (固定日 1900 年 1 月 1 日からのミリ秒数)
%%severity	ルール定義での重大度
%%SourceProgram	ソース・プログラム名
%%SQLNoValue	マスクされた値を含む SQL 文字列。SYSLOG 内で SQL の値が ? に置き換えられます。
%%SQLString	SQL 文字列 (存在する場合)
%%SQLTimestamp	パケット/要求の時間 (GDM_CONSTRUCT_TEXT での TIMESTAMP)
%%Subject[ ]	この変数がメッセージ・テンプレートで使用されている場合、[ ] の間に表示されるものすべて (例えば、ファイル名、E メール送信者、説明) は、ユーザーに送信される E メールの件名行になります。
%%Verb	深さ 0 の SQL 動詞。同じ深さで複数の SQL 動詞を使用できます。
%%violationID	GDM_POLICY_VIOLATION_LOG でのこのアラートの POLICY_VIOLATION_LOG_ID を表す数値 (これはポリシー違反/インシデント管理レポートの「違反ログ ID」と同じです)

名前付きテンプレート

メッセージ・テンプレートを使用してアラートが生成されます。

この機能は、複数のメッセージ・テンプレートを定義し、異なるルールに対して異なるテンプレートを使用できるようにします。これまでは、すべてのルール、すべての受信者タイプなどに対してただ 1 つのメッセージ・テンプレートしか使用できませんでした。

名前付きメッセージ・テンプレートを追加、変更、および削除するには、「編集」をクリックします。新しい名前付きテンプレートを作成するとき、文字列には最初、グローバル・プロファイルのメッセージ・テンプレートで現在設定されている内容のコピーが入っています。可能な重大度のレベルは「R/T アラート」のみです。

SIEM ソリューション (ArcSight、EnVision および QRadar) 用に事前定義メッセージ・テンプレートが作成されています。Guardium システムには、この 2 つの SIEM ソリューションと統合するための 2 つの認定済み (合意済み) テンプレートがプリロードされています。

名前付きテンプレートのビルダーは、 2 つのテンプレート・タイプ (リアルタイム・アラートおよび 監査プロセス・レポート) から選択できます。

監査プロセス・レポートは、プロセス・タスクを監査するときに使用します。

「名前付きテンプレートの編集」をクリックします。「SIEM」を選択し、「変更」をクリックします。「リアルタイム・アラート」または「監査プロセス・レポート」を選択します。

編集した後、複数のメッセージ・テンプレートを「ポリシー・ビルダー」メニューの中から選択できます。ポリシーを参照してください。

QRadar テンプレートを追加すると、(QRadar のフォーマットである) LEEF フォーマットを使用して、QRadar にリアルタイム・アラートまたは監査プロセス・レポートを送信できます。

ステップに従って、リアルタイム・アラートまたは監査プロセスの結果を QRadar SIEM に送信します。

リアルタイム・アラート (Guardium から QRadar へ)

1. リアルタイム・アラートを作成します。
2. syslog に書き込みます。
3. テンプレート型 (読み取り時間アラート) を選択します。
4. (LEEF マッピング / 事前定義メッセージ・テンプレートを介して) Q1 Labs QRadar SIEM に転送します - グローバル・プロファイルから QRadar 名前付きテンプレートを選択します。
5. CLI から、CLI コマンド「store remotelog」を実行して、syslog メッセージを QRadar に転送します。

監査プロセス・レポート (Guardium から QRadar へ)

「強化」 > 「脆弱性評価」 > 「監査プロセス・ビルダー」をクリックして、「監査プロセス・ビルダー」を開きます。

1. 監査プロセス・レポートを作成します (監査プロセス・ビルダー)。
2. syslog に書き込みます。
3. テンプレート型 (監査プロセス・レポート) を選択します。
4. (LEEF マッピング / 事前定義メッセージ・テンプレートを介して) Q1 Labs QRadar SIEM に転送します - グローバル・プロファイルから QRadar 名前付きテンプレートを選択します。
5. CLI から、CLI コマンド「store remotelog」を実行して、syslog メッセージを QRadar に転送します。

例として、「ディスカバーされたデータベース」レポート用のデフォルト LEEF テンプレートを以下に示します。

```
LEEF:0|IBM|Guardium|9.0|Databases Discovered|Time Probed=${1}|Server IP=${2}|Server Host Name=${3}|DB
Type=${4}|Port=${5}|Port Type=${6}
```

テンプレートにマップされるレポートの列を以下に示します。

プローブ時間　　サーバー IP
サーバー・ホスト名　　データベース・タイプ　　ポート　　ポート・タイプ

1. 「CSV ファイルへのエクスポート」と「Syslog に書き込む」にチェック・マークを付けます。
2. 名前付きテンプレート LEEF Discovered Databases を選択します。
3. store remotelog コマンドを使用して、リモート Syslog を構成します。例:

   ```
   store remotelog add user.info 9.70.145.68 udp
   ```

   これにより、監査プロセスからすべてのレコードが、指定された IP アドレスにプッシュされます。

送信者のエンコード

出力メッセージ (E メールおよび SNMP トラップ) を、UTF8 以外のエンコード・スキーマでエンコードするには、CLI コマンド store sender_encoding を使用します。

1 タイプのテンプレートのフィルター操作

すべてのリアルタイム・アラートまたは監査プロセス・レポートを選択するためのフィルター・メカニズムがあります。各選択項目にチェック・マークを付けるか、外します。

Envision 2 メッセージ・テンプレート

GUARDIUM_ALERT:
rule-id=%%ruleID^^category=%%category^^classification=%%classification^^severity=%%severity^^session-start-time=%%sessionStart^^client-hostname=%%clientHostname^^client-ip=%%clientIP^^server-type=%%serverType^^server-ip=%%serverIP^^src-program=%%SourceProgram^^os-user=%%OSUser^^db-user=%%DBUser^^app-user=%%AppUserName^^service-name=%%serviceName^^req-type=%%requestType^^rule-desc=%%ruleDescription^^sql=%%SQLNoValue

しきい値のデフォルトのテンプレート

リアルタイム・アラートの場合と同様に、しきい値に到達したときに送信されるメッセージのテンプレートを選択できます。このテンプレートでは、特定のアラート用に適切な値に置換される変数の定義済みリストが使用されます。

これらの変数は、以下のとおりです。

%%alertName - アラート名

%%description - アラートの記述

%%alertQueryValue - アラートの原因となった照会値

%%alertThreshold - アラートのしきい値

%%alertQueryFromDate - 照会期間の開始

%%alertQueryToDate - 照会期間の終了

%alertBaseQueryValue - アラートの基本照会値

%%classification - アラートの分類

%%category - アラートのカテゴリー

%%severity - アラートの重大度

%%recommendation - アラートに対する推奨アクション

%%Subject[] - メッセージの件名

しきい値アラートのデフォルトのテンプレートは、以下のとおりです (コピーと編集が可能)。

%%Subject[Guardium アラート。 重大度: (%%severity)、アラート名: %%alertName]

アラート名: %%alertName。 アラートの記述: %%description。

現行値: %%alertQueryValue

基本照会値: %%alertBaseQueryValue

しきい値: %%alertThreshold

照会期間: %%alertQueryFromDate - %%alertQueryToDate

アラートの分類: %%classification

カテゴリー: %%category

重大度: %%severity

推奨アクション: %%recommendation

リアルタイム・アラートと E メールのカスタマイズ
E メールの件名に Guardium アプライアンス名を含む接頭部を表示するかどうかを制御します。
E メール本文に E メールの件名を表示するかどうかを制御します。
Guardium ユーザーが、名前付きテンプレート (件名または本文のいずれか) にアプライアンスのホスト名を追加できるように、命名テンプレート・パラメーター %%applianceHostName を追加します。
これを行うには、ADMINCONSOLE_PARAMETERS 表の以下の 2 つのフィールドを使用します。
APPEND_APPLIANCENAME_SUBJECT
APPEND_SUBJECT_IN_BODY
これら 2 つのフィールドの内容を制御するには、以下の CLI コマンドを使用します。
show alerter email append_name_subject
store alerter email append_name_subject
E メールの件名にアプライアンス名を付加するためのフラグを表示または保管します
show alerter email append_subject_body
store alerter email append_subject_body は、E メール本文の先頭に E メールの件名を付加するためのフラグを表示または保管します
CLI 内の値が変更されるたびに、送信される E メールに直ちに反映されます。

CSV 区切り文字

監査プロセスで使われる区切り文字を定義するには、次のようにします。

1. 「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」をクリックして、「グローバル・プロファイル」を開きます。
2. コンマ、セミコロン、タブのいずれかを選択するか、または、使用される CSV 区切り文字を「その他」ボックスで独自に定義します。
3. 「適用」をクリックします。

Guardium ウィンドウへの他の HTML コンテンツの追加

他の HTML コンテンツを Guardium ウィンドウに追加するには、次のようにします。

1. 「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」をクリックして、「グローバル・プロファイル」を開きます。
2. 「HTML - 左」および「HTML - 右」テキスト・ボックスで、ウィンドウ上に含めるテキストまたは他の項目を表す HTML を入力します。
3. オプションで、プレビュー・ボタンをクリックして、HTML が予期したとおりに表示されるかどうかを確認します。
4. 「適用」をクリックします。

ログイン・メッセージの追加または無効化

ユーザー・ログイン時にメッセージ・ボックスに毎回表示されるメッセージを追加するには、次のようにします。

1. 「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」をクリックして、「グローバル・プロファイル」を開きます。
2. 「ログイン・メッセージ」テキスト・ボックスで、各ユーザーのログイン時に表示するテキストを入力します。
3. 「ログイン・メッセージを表示」ボックスにマークを付けると、ログイン・メッセージの表示が有効になります (ボックスをクリアすると表示が無効になります)。

4. 「適用」をクリックします。

同じユーザーによる複数の同時ログインの有効化/無効化

デフォルトでは、同じ Guardium ユーザーが複数の IP アドレスからアプライアンスにログインできます。同じユーザーからの複数の同時ログインを無効にすることができます。無効化した場合、各 Guardium ユーザーは同時に 1 つの IP アドレスからのみログインできます。ユーザーがログアウトせずにブラウザーを閉じた場合、非アクティブ状態のため接続がタイムアウトになります。したがってユーザー・アカウントが長時間にわたってブロックされることはありません。

この設定を変更するには、次のようにします。

1. 「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」をクリックして、「グローバル・プロファイル」を開きます。
2. フィールド「異なる IP からの同時ログイン」を探します。
3. 現在の状況に応じて、「有効化」または「無効化」をクリックして、設定を変更します。
   注: この機能が無効になっている場合は、「有効化」ボタンの横に「アンロック」ボタンが表示されます。「アンロック」をクリックすると、別のユーザーがこのユーザー・アカウントを使って別の IP アドレスからログインできるようになります。これはサポートを目的として備えられています。

監視データ・レベルにおけるデータ・レベル・セキュリティーの有効化

この機能では、特定の Guardium ユーザーが特定のデータベースを担当することを想定します。そのため、システム全体に渡って結果をフィルタリングするメカニズムが存在し、各ユーザーは自分が担当するデータベースの情報だけを表示できるようになっています。

制約事項: データ・レベル・セキュリティーと調査ダッシュボードは同時に有効化できません。

この設定を変更するには、次のようにします。

1. 「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」をクリックして、「グローバル・プロファイル」を開きます。
2. 「データ・レベル・セキュリティーのフィルタリング」オプションの「有効化」または「無効化」ボタンをクリックします
   注: データ・レベル・セキュリティーが有効で、datasec-exempt ロールがユーザーに割り当てられている場合、datasec-exempt ロールがアクティブになります。
3. 追加の選択項目には、以下のものがあります。
   - すべて表示 - 行がどのユーザーに属するかにかかわらず、ログイン済みのビューアーで結果のすべての行を表示できるようにします。Datasec-exempt ロールと併用すると、データ・レベル・セキュリティーのフィルタリングをオーバーライドできます。
   - 間接レコードを含める - ログインしたビューアーでは、ログイン済みユーザーに属する行を表示できることに加えて、ユーザー階層でログイン済みユーザーの下位にあるユーザーに属するすべての行を表示できます。

注: 監視データ・レベルでのデータ・レベル・セキュリティーが有効になっている場合、監査プロセスのエスカレーションはユーザー階層の上位レベルのユーザーに対してのみ許可されます。

デフォルトのフィルタリング

オンライン・ビューアーおよび監査プロセスの結果配布のデフォルト設定。

「すべて表示」- デフォルト設定は、無効になっています。

結果をすべてのユーザーにエスカレートする

「結果をすべてのユーザーにエスカレート」 - このチェック・ボックスにチェック・マークを付けると、監視データ・レベルでのデータ・レベル・セキュリティーが有効になっている場合であっても、監査プロセスの結果 (および PDF バージョン) がすべてのユーザーにエスカレートされます。 デフォルト設定では有効になっています。このチェック・ボックスが無効になっている (チェック・ボックスにチェック・マークが付けられていない) 場合、監査プロセスのエスカレーションはユーザー階層の上位レベルのユーザーおよび datasec-exempt ロールを持つユーザーに対してのみ許可されます。チェック・ボックスが無効になっており、ユーザー階層がない場合、エスカレーションは許可されません。

カスタム・データベース表の最大サイズ

カスタム・データベース表のサイズを MB 単位で設定します。 デフォルト値は 4000 MB です。

この時点で、「グローバル・プロファイル」メニューに、現在の使用量を確認するためのボタンが表示されます。 「現在の使用量」ボタンをクリックすると、INNODB、MYISAM および合計について値が表示されます。

注: カスタム・サイズ制限は、データのインポートの前にテストされます。 インポートによって最大サイズ制限を超える可能性があります。 制限を超えた場合、その次のインポートが回避されます。

異なるポートを介するファイルの SCP および FTP 送信

SCP および FTP を介するファイル送信に使用できるポートに変更します。

グローバル・プロファイルの場合 - エクスポートおよびパッチ・バックアップを変更できます。 ssh/scp/sftp のデフォルトのポートは 22 です。FTP のデフォルトのポートは 21 です。
注: Guardium GUI にポートとして「0」が表示される場合、デフォルト・ポートが使用されており、変更の必要がないことを示します。

Guardium ウィンドウへのロゴの追加

企業のロゴ・グラフィックを Guardium ウィンドウに追加するか、または他の HTML コンテンツを Guardium ウィンドウに追加するには、次のようにします。

1. 「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」をクリックして、「グローバル・プロファイル」を開きます。
2. ポータル・ウィンドウにロゴ・イメージを含めるには、「ロゴ・イメージのアップロード」でイメージ・ファイル名を入力するか、「参照」をクリックして Guardium アプライアンスにアップロードするファイルを選択した後、「アップロード」をクリックします。
3. ブラウザー・ウィンドウをリフレッシュします。新しいロゴが表示されます。

注: アップロードするロゴ・ファイルの名前に、単一引用符、二重引用符、「より小」記号、または「より大」記号を含めることはできません。

Must Gather の暗号化

「Must Gather の暗号化 (Encrypt Must Gather)」が、「グローバル・プロファイル」に追加されました。デフォルトでは、クリアされています (暗号化しない)。これがクリアされている場合、Must Gather 出力は圧縮されるだけで、暗号化されません。このチェック・ボックスにチェック・マークを付けると、以降のすべての Must Gather 出力は暗号化されます。暗号化は、store encrypt_must_gather on CLI コマンドを使用してオンに設定したり、store encrypt_must_gather off を使用してオフに設定したりすることもできます。

Guardium の更新の確認

チェック・マークを追加すると、お客様がダウンロードできる、関連する随時の Guardium パッチ、GPU/CFP/バンドル、Sniffer パッチ、およびセキュリティー・パッチが表示されます。パッチは、インストールされると、リストに表示されなくなります。

データ・ソース接続タイムアウト

データ・ソース接続タイムアウトを分単位で設定します。デフォルトは 60 秒間です。

この値を更新するための対応する GrdAPI コマンドは、grdapi update_datasource_connection_timeout timeoutInSecond=80 です。

親トピック: Guardium システムの構成

アラート機能の構成

アラート機能を構成してアクティブ化するまでは、E メール・メッセージ、SNMP トラップ、アラート関連 Syslog メッセージはまったく送信されません。

アラート機能用のメッセージは他のコンポーネントによって作成され、キューに入れられます。アラート機能は、構成済みのポーリング間隔に基づいてメッセージを検査し、送信します。

個々の相関アラートを構成、有効化、または無効化するには、相関アラートを参照してください。相関アラートおよびアプライアンス・アラートが生成されるためには、異常検出も開始済みでなければなりません。 リアルタイム・アラートが生成されるためには、セキュリティー・ポリシーがインストール済みでなければなりません。

メール/SNMP/SYSLOG メッセージは、優先度に従って送出されます。

始動時のアラート機能の自動アクティブ化

1. 「設定」 > 「ツールとビュー」 > 「アラート機能」をクリックして「アラート機能」を開くか、「保護」 > 「データベースの侵入検出」 > 「アラート機能」をクリックして「アラート機能」を開きます。
2. 「始動時にアクティブ」チェック・ボックスにマークを付けます。アプライアンスが再始動するたびに、アラート機能が自動的にアクティブ化されます。
3. 「適用」をクリックします。
4. アラート機能が実行中でない場合、これを開始するには「再始動」をクリックします。

アラート機能によるメッセージ検査/送信の頻度の設定

1. 「設定」 > 「ツールとビュー」 > 「アラート機能」をクリックして「アラート機能」を開くか、「保護」 > 「データベースの侵入検出」 > 「アラート機能」をクリックして「アラート機能」を開きます。
2. 「ポーリング間隔」(秒) を入力します。
3. 「適用」をクリックします。

SMTP (E メール) メッセージを送信するようアラート機能を構成する

1. 「設定」 > 「ツールとビュー」 > 「アラート機能」をクリックして「アラート機能」を開くか、「保護」 > 「データベースの侵入検出」 > 「アラート機能」をクリックして「アラート機能」を開きます。
   注: このトピックの残りの項目はすべて、「アラート機能」パネルの「SMTP」セクションに含まれています。
2. 「IP アドレス」ボックスに SMTP ゲートウェイの IP アドレスを入力します。
3. 「ポート」ボックスに SMTP ポート番号を入力します (ほとんどの場合、25 です)。
4. オプションで、「接続のテスト」ハイパーテキスト・リンクをクリックして SMTP アドレスとポートを検証します。これは単に、指定されたホストとポートにアクセスできることを検査するだけです。機能している SMTP サーバーであることを検証するものではありません。この操作の成功または失敗を通知するダイアログ・ボックスが表示されます。
   注: この SMTP サーバーが認証を使用する場合、以下の 2 つのフィールドで、そのメール・サーバーの有効なユーザー名とパスワードを提供する必要があります。そうでない場合は、これらのフィールドをブランクにすることができます。
5. SMTP サーバーが認証を使用する場合、メール・サーバーの有効なユーザー名を「ユーザー名」ボックスに入力します。
6. SMTP サーバーが認証を使用する場合、ユーザーのパスワードを「パスワード」ボックスに入力します。「パスワードの再入力」ボックスにそれを再び入力します。
7. 「送信先 E メールアドレス」ボックスに、システムから送られる E メールの送信先アドレスを入力します。 通常、このアドレスは、頻繁にチェックされる管理アカウントです。
8. SMTP サーバーが認証を使用する場合、「認証方式」で 「認証」 を選択します。そうでない場合は、「なし」を選択します。「認証」を選択した場合、認証で使われるユーザー名とパスワードを指定する必要があります。
9. 「適用」をクリックして、構成を保存します。
   注: アラート機能が再始動するまでは、新しい構成は使用されません。
10. 「再始動」をクリックすると、新しい構成を使ってアラート機能が再始動します。

SNMP トラップを送信するようアラート機能を構成する

1. 「設定」 > 「ツールとビュー」 > 「アラート機能」をクリックして「アラート機能」を開くか、「保護」 > 「データベースの侵入検出」 > 「アラート機能」をクリックして「アラート機能」を開きます。
   注: このトピックの残りの項目はすべて、「アラート機能」パネルの「SMTP」セクションに含まれています。
2. 「IP アドレス」ボックスで、SNMP トラップの送信先となる IP アドレスを入力します。
3. オプションで、「接続のテスト」ハイパーテキスト・リンクをクリックして SNMP アドレスとポート (162) を検証します。これは単に、指定されたホストとポートにアクセスできることを検査するだけです。機能している SNMP サーバーであることを検証するものではありません。この操作の成功または失敗を通知するダイアログ・ボックスが表示されます。
4. 「「トラップ」コミュニティー」ボックスに、トラップのコミュニティー名を入力します。 「コミュニティーの再入力」ボックスにコミュニティーを再入力します。
5. 「適用」をクリックして、構成を保存します。
   注: アラート機能が再始動するまでは、新しい構成は使用されません。
6. 「再始動」をクリックすると、新しい構成を使ってアラート機能が再始動します。

親トピック: Guardium システムの構成

異常検出

異常検出プロセスは、アラートの照会に基づいて相関アラート通知を作成して保存する (ただし送信はしない) ためにポーリング間隔ごとに実行されます。

この通知は、各アラートに対して定義されたスケジュールに従って実行されます。 通知の送信について詳しくは、アラート機能の構成を参照してください。

異常検出プロセスは、指定された期間をさかのぼって調査する相関アラートの照会の結果と、 相関アラートのしきい値を使用して、条件 (例えば過剰なログイン失敗数) が満たされたかどうかを判別します。詳しくは、相関アラートを参照してください。

中央マネージャー環境では、各 Guardium システムの「異常検出」パネルを使用して、その特定の Guardium システムに適さない相関アラートをオフにすることができます。一元管理下では、すべての相関アラートは、どのシステムで作成や更新が行われたかに関係なく、中央マネージャーで定義されます。これらの相関アラートは、すべての Guardium システムに対して同じになり、アクティブ化されるときには、デフォルトですべての Guardium システムに対してアクティブ化されます。

注: 保存したアラート・メッセージを SYSLOG、E メール、または SNMP トラップに送信するには、アラート機能コンポーネントを構成して開始する必要があります。
注: 異常検出は、(セキュリティー・ポリシーによって生成される) リアルタイム・アラートの生成には関与しません。

始動時の異常検出の自動アクティブ化

1. 「設定」 > 「ツールとビュー」 > 「異常検出」をクリックして、「異常検出」を開きます。
2. 「始動時にアクティブ」チェック・ボックスにマークを付けます。Guardium システムが再始動するたびに、異常検出が自動的にアクティブ化されます。
3. 「適用」をクリックします。

異常検出によってアプライアンスの問題を検査する頻度の設定

1. 「設定」 > 「ツールとビュー」 > 「異常検出」をクリックして、「異常検出」を開きます。
2. 「ポーリング間隔」(分) を入力します。
3. 「適用」をクリックします。

アクティブ・アラートの有効化/無効化

中央マネージャー環境でアラートをグローバルに無効化するには、「アラートの変更」パネルの「アクティブ」チェック・ボックスをクリアするのが簡単な方法です。

一元管理環境で 1 つの Guardium システムのアラートを有効または無効にするには、以下の手順に従ってください。

1. 1 つ以上のアラートを無効にする対象の Guardium システムの UI にログインします。
2. 「設定」 > 「ツールとビュー」 > 「異常検出」をクリックして、「異常検出」を開きます。
3. いずれかのアラートを無効にするには、「アクティブ・アラート」ボックスでそれを選択して「無効化」をクリックします。
4. いずれかのアラートを有効にするには、「ローカルでは無効なアラート」ボックスでそれを選択して「有効化」をクリックします。

異常検出の停止/再始動

1. 「設定」 > 「ツールとビュー」 > 「異常検出」をクリックして、「異常検出」を開きます。
2. 異常検出を停止するには「停止」をクリックし、再始動するには「再始動」をクリックします。

親トピック: Guardium システムの構成

セッション推論

セッション推論は、指定された期間にわたって非アクティブ状態が続いている開いたセッションがあるかどうか検査し、それらにクローズ済みのマークを付けます。

セッション推論オプションを構成するには、次のようにします。

1. 「設定」 > 「セッション推論」をクリックして、「セッション推論」を開きます。
2. Guardium® システムの開始時にセッション推論を開始するには、「始動時にアクティブ」ボックスにマークを付けます。
3. 「ポーリング間隔」ボックスに、セッション推論により開いたセッションがないか検査する頻度 (分数) を入力します。デフォルトは 120 (分) です。
4. 「最大非アクティブ期間」ボックスに、セッションにクローズ済みのマークを付けるまでの、非アクティブ状態の分数を入力します。デフォルトは 720 (分) です。
5. 「適用」をクリックすると、構成データベースに値が保管されます。セッション推論が再始動するまでは、新しい構成は使用されません。
6. 「再始動」をクリックすると、新しい構成を使ってセッション推論が再始動します。

セッション推論を停止するには、「セッション推論」パネルを開いて「停止」をクリックします。

親トピック: Guardium システムの構成

Guardium への S-TAP 接続のブロック (S-TAP 認証)

この機能を使用して、クライアントが Guardium システムへのアクセスを許可されている特定の S-TAP ホストを制御します。

このタスクについて

有効にすると、指定された S-TAP クライアントのみが Guardium システムへのアクセスを許可されます。

CLI コマンド store stap approval または GuardAPI コマンド grdapi store_stap_approval を使用してこの機能を制御することもできます。

CLI コマンド store stap approval を使用する場合は、コマンド restart inspection-core を実行した後に新規構成が有効になります。

承認された STAP は、「管理」 > 「レポート」 > 「変更モニター」 > 「承認された Tap クライアント」または「レポート」 > 「リアルタイム Guardium 運用レポート」 > 「承認された Tap クライアント」で確認します。

手順

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 認証」にアクセスします。
2. 「S-TAP 承認が必要」を選択します。
3. 承認された S-TAP クライアント・ホストの IP アドレス (ホスト名ではない) を「承認された S-TAP クライアント」セクションで指定し、「追加」をクリックします。
4. 各 S-TAP クライアントに対してこの手順を繰り返します。

タスクの結果

注: 一元管理された環境内では、承認された S-TAP に IP アドレスを追加した後、同期に関連する待ち時間が発生します。この待ち時間は、最大で 1 時間かかる可能性があります。同期が完了すると、承認済みの S-TAP の状況が「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」に緑色で表示されます。

親トピック: Guardium システムの構成

IP からホスト名への別名割り当て

IP からホスト名への別名割り当て機能は、ドメイン・ネーム・システム (DNS) サーバーにアクセスして、クライアントおよびサーバーの IP アドレスのホスト名別名を定義します。

それぞれクライアント用、およびサーバー用に別個の IP アドレス・セットが 2 つあります。IP からホスト名への別名割り当てが有効になっている場合、適切な場合に Guardium® 内で IP アドレスが別名に置き換えられます。

1. 「保護」 > 「データベースの侵入検出」 > 「IP からホスト名への別名割り当て」をクリックして、「IP からホスト名への別名割り当て」を開きます。
2. 「クライアント IP とサーバー IP のホスト名別名の生成 (使用可能な場合)」チェック・ボックスにマークを付け、ホスト名の別名を有効にします。

   2 番目のチェック・ボックスにアクセスできるようになります。このチェック・ボックスの名前は「既存のホスト名別名の更新 (再発見された場合)」です。

3. このチェック・ボックスにマークを付け、以前に定義された、現在の DNS ホスト名に一致しない別名 (通常、その IP アドレスのホスト名が変更されたことを示す) を更新します。いくつかの別名を手操作で既に割り当てた場合は、このような動作が不適切であることがあります。例えば、ある IP アドレスの DNS ホスト名が dbserver204.guardium.com で、そのサーバーの通称が QA Sybase Server であるとします。その IP アドレスの別名として QA Sybase Server が手動で定義されており、かつ「既存のホスト名別名の更新 (再発見された場合)」のチェック・ボックスにマークが付けられている場合、その別名は DNS ホスト名により上書きされます。
4. 「適用」をクリックして、IP からホスト名への別名割り当て構成を保存します。
5. 以下のいずれかを実行します。
   - 「今すぐ 1 回実行」をクリックすると、別名が直ちに生成されます。
   - 「スケジュールの定義」をクリックすると、このタスク実行のスケジュールを定義できます。詳しくは、スケジューリングを参照してください。

定義した別名を表示するには、『別名』を参照してください。

親トピック: Guardium システムの構成

システム・バックアップ

システム・バックアップ機能を使用すると、オンデマンドまたはスケジュールに基づいて実行可能なバックアップ操作を定義できます。

システム・バックアップ

ハードウェア破損が生じた場合にサーバーを復元する目的で、必要なすべてのデータと構成値をバックアップして保管するために、システム・バックアップを使用します。

すべての構成情報とデータが 1 つの暗号化ファイルに書き込まれ、このアプライアンスのバックアップ用に構成された転送方式を使用して、指定の宛先に送信されます。

バックアップされたシステム情報をリストアするには、restore system CLI コマンドを使用します。また、特定のユーザーのロールとして diag が定義されている場合には、CLI コマンド diag を使用できます。

システム・バックアップは、以下の方式をサポートしています。

- SCP - デフォルトで定義され、CLI および GUI を介してアクセス可能
- FTP - デフォルトで定義され、CLI および GUI を介してアクセス可能
- Centera - CLI にログインし、次のコマンドを実行して GUI に追加可能: store storage centera backup on
- TSM - 追加できます。そのためには CLI にログインし、次のコマンドを実行します: store storage tsm backup on

- AMAZON S3 - デフォルトで定義され、CLI および GUI を介してアクセス可能。CLI からアクセスできるのは、GUI で定義されている場合です。
- Softlayer - Softlayer のクラウド・バックアップ
- Cleversafe - CleverSafe 機能。Amazon S3 に類似する方法でバックアップを保管します。使用可能なバケットのリストが GUI に直接抽出されます。最初にリストされている名前は、データベースに保存したバケットの名前です。注: (Guardium UI/CLI から) 新しいバケットを作成することも、バケットを削除することもできません。

注: システム・リストアは、システム・バックアップと同じパッチ・レベルに対して実行する必要があります。例えばバージョン 7.0 パッチ 7 の時点でアプライアンスをバックアップした後、新しく構築したアプライアンスにこのバックアップをリストアするには、まずバージョン 7.0 のパッチ 1 から 7 までをアプライアンスにインストールした後で、ファイルをリストアする必要があります。

システム情報をバックアップするには、次のようにします。

1. 「管理」 > 「データ管理」 > 「システム・バックアップ」をクリックして、「システム・バックアップ」を開きます。
2. リストから、ストレージ方式のラジオ・ボタンを選択します。Guardium システムの構成方法によっては、これらのボタンの 1 つ以上が選択できない場合があります。アーカイブおよびバックアップのストレージ方式の構成方法については、構成および制御 CLI コマンドで show storage-system コマンドと store storage-system コマンドの説明を参照してください。
   - EMC CENTERA
   - TSM
   - SCP
   - FTP
   - AMAZON S3
   - Softlayer
   - Cleversafe
3. 選択したストレージ方式に応じて、適切な手順を実行します。
   - SCP または FTP アーカイブまたはバックアップの構成
   - EMC Centera アーカイブまたはバックアップの構成
   - TSM アーカイブまたはバックアップの構成
   - AMAZON S3 アーカイブまたはバックアップの構成
   - Softlayer オブジェクト・ストレージ・クラウド・バックアップの構成
   - Cleversafe - 入力> 有効なエンドポイント、有効なバケット名、有効なアクセス・キー、有効な共有パスワード
4. 「バックアップ」の次のチェック・ボックスのいずれかまたは両方にマークを付けます。
   - 「構成」チェック・ボックスにマークを付けると、すべての定義がバックアップされます。
   - 「データ」チェック・ボックスにマークを付けると、すべてのデータがバックアップされます。(データを定期的にアーカイブしている場合は、これは不要です。)
5. 「スケジューリング」セクションを使用して、この操作を定期的に実行するためのスケジュールを定義できます。
6. 「保存」」をクリックすると、構成の変更が検証されて、保存されます。システムはそのロケーションにテスト・データ・ファイルを送信することにより、構成の検証を試みます。
   - 操作が失敗すると、エラー・メッセージが表示され、構成は保存されません。
   - 操作が成功すると、構成が保存されます。
7. 「今すぐ 1 回実行」をクリックして、操作を 1 回実行します。

注: SCP/FTP/TSM/Centera/AMAZON S3/Softlayer ファイル転送中にバックアップ・ファイルの転送が失敗した場合、(システム・バックアップ、構成バックアップ、アーカイブ、CSV アーカイブなど) バックアップ/アーカイブ・ファイルの各セットの最後のファイルが diag/current フォルダーに保存されます。その後、バックアップ・ファイルの宛先が再びオンラインになったときに、diag/current フォルダーから宛先に手操作でバックアップ・ファイルを転送できます。 バックアップ/アーカイブ・ファイルのセットが diag/current フォルダーに保存されるのは、ファイル転送が失敗した場合だけです。 別のバックアップ・ファイル転送中にファイルの転送が失敗した場合、バックアップ/アーカイブ・ファイルのセットが diag/current フォルダーに再び保存されます。ただし、保存されるファイルが多くなりすぎてディスク・スペースが不足するのを防ぐために、それぞれの種類の最新のファイルだけが保存されます。それより前のバックアップ・ファイルは上書きされます。
注: システム・バックアップを実行し、GIM を定義してあるサーバーから別のサーバーにリストアするときは、 リストア・サーバーに対する GIM フェイルオーバーを構成する必要があります。この GIM 構成は、 バックアップ中央マネージャーまたはシステム・バックアップおよびリストアに適用されます。

異なるポートを介するファイルの SCP および FTP 送信

SCP および FTP を介するファイル送信に使用できるポートに変更します。

システム・バックアップの場合 - プロトコル (SCP または FTP) を設定し、ホスト、ディレクトリー、およびポートを指定します。 ssh/scp/sftp のデフォルトのポートは 22 です。FTP のデフォルトのポートは 21 です。

バックアップ/アーカイブのスクリプトによる /var 容量の使い尽くしの防止

バックアップ・プロセスは、実行前に /var の空き容量をチェックして失敗を防止します。 このプロセスは、バックアップ用のスペースが十分にない場合にも、 ユーザーに警告を出します。

アーカイブ・プロセスは、 静的表のサイズをチェックし、アーカイブを作成できる空き容量が /var にあることを 確認します。

バックアップが 50% を超えると、 ログ・ファイルおよび GUI にエラーが記録されるようになっています。例:

```
ERROR: /var backup space is at 60% used. Insufficient disk space for backup.
```

Guardium での Amazon S3 へのアーカイブおよびバックアップ

Guardium から、Amazon S3 へのデータのアーカイブとバックアップを行う場合に、この機能を使用します。

Amazon S3 (Amazon Simple Storage Service) は、いつでも、Web 上のどこからでも容量に関係なく、データを格納/取得できるシンプルな Web サービス・インターフェースを提供します。これによって、Amazon が Web サイトの稼働に使用しているものと同じ、拡張性と信頼性が高く、安全でありながら安価なインフラストラクチャーを、あらゆる開発者が利用することが可能になります。

前提条件

1. Amazon アカウント

2. S3 サービスの登録

3. Amazon S3 にアクセスするためには、Amazon S3 の認証情報が必要です。 必要な認証情報は次のとおりです。
    - Access Key ID (アクセス・キー ID): ユーザーをサービス要求の担当者として識別します。各要求にこの ID が含まれている必要があります。これは機密ではなく、暗号化する必要はありません (20 文字の英数字から成るシーケンス)。
    - Secret Access Key (シークレット・アクセス・キー): Secret Access Key は Access Key ID に関連付けられ、要求に含まれているデジタル署名を計算します。Secret Access Key は機密事項であり、ユーザーと AWS のみが保持する必要があります (40 文字から成るシーケンス)。このキーは、ファイルではなく、単なる長い文字列であり、この文字列を使用して、要求内に含まれている必要があるデジタル署名を計算します。

「管理コンソール」メニューの「データ管理」セクションでは、次の 2 つのアーカイブ操作を選択可能です。

- 「データ・アーカイブ」は、アプライアンスによって所定の期間内にキャプチャーされたデータをバックアップします。
- 「結果アーカイブ」は、監査タスクの結果 (レポート、アセスメント・テスト、エンティティー監査証跡、プライバシー・セット、および分類プロセス) のほか、表示およびサインオフの証跡、ワークフロー・プロセスからの調整コメントをバックアップします。

Guardium データがアーカイブされると、日ごとに別のデータ・ファイルができます。

アーカイブ・データ・ファイルの名前は、次の形式になります。

```
 <time>-<hostname.domain>-w<run_datestamp>-d<data_date>.dbdump.enc
```

アーカイブ機能は、不正に開封できない、署名付きの暗号化ファイルを作成します。 生成されたアーカイブ・ファイルの名前を変更することはできません。 アーカイブ操作は、アーカイブ処理中に作成されるファイル名に依存します。

ハードウェア破損が生じた場合にサーバーを復元する目的で、必要なすべてのデータと構成値をバックアップして保管するために、システム・バックアップを使用します。

すべての構成情報とデータが 1 つの暗号化ファイルに書き込まれ、このアプライアンスのバックアップ用に構成された転送方式を使用して、指定の宛先に送信されます。

バックアップ・システム・ファイルの形式は、次のとおりです。

```
<data_date>-<time>-<hostname.domain>-SQLGUARD_CONFIG-9.0.tgz
<data_date>-<time>-<hostname.domain>-SQLGUARD_DATA-9.0.tgz
```

「統合/アーカイブ・ログ」レポートを使用して、操作が正常に完了したことを確認できます。各アーカイブ操作には、複数のアクティビティーがリストされていなければなりません。 また、各アクティビティーの状況は成功でなければなりません。

Guardium カタログは、アーカイブ・データの宛先に関係なくすべてのアーカイブ・ファイルの送信場所を記録するため、以降のどの時点においても、最小限の労力でシステムでアーカイブ・ファイルを取得およびリストアすることができます。

アプライアンスごとに個別のカタログが保守され、アプライアンスがデータや結果をアーカイブするたびにカタログに新しいレコードが追加されます。

カタログ・エントリーは、以下のいずれかの方法により、アプライアンス間で転送することができます。

- 統合 - カタログ表は統合されます。つまり、アグリゲーターが、そのすべてのコレクターのカタログをマージしたものを保持することになります。
- カタログのエクスポート/インポート - これらの機能は、コレクター間でカタログ・エントリーを転送したり、将来のリストアのためにカタログをバックアップしたりするために使用できます。
- データのリストア - 各データ・リストア操作には、アーカイブした日のデータ (その日のカタログも含む) が含まれます。 したがって、データのリストア時には、カタログも更新されます。

カタログ・エントリーは、別のシステムからインポートされたときには、そのシステムによって暗号化されたファイルをポイントします。 そのようなファイルをリストアまたはインポートする前に、そのファイルを暗号化したシステムのシステム共有パスワードが、インポート側のシステムで使用できなければなりません。

Guardium CLI からの Amazon S3 の有効化

Amazon S3 のアーカイブ/バックアップ・オプションは、デフォルトでは Guardium GUI で有効になっています。Guardium CLI から Amazon S3 を有効にするには、次の CLI コマンドを実行します。

```
store storage-system amazon_s3 archive on
store storage-system amazon_s3 backup on
```

Amazon S3 では、Guardium システムのクロック時刻が正確であること (15 分以内) が求められます。 そうでない場合、Amazon のエラーとなります。要求の時刻と現在の時刻の差が大きすぎると、要求は受け入れられません。

Guardium のシステム時刻が正確でない場合は、次の CLI コマンドを使用して正しい時刻を設定してください。

```
show system ntp server
store system ntp server (An example is ntp server: ntp.swg.usma.ibm.com)
store system ntp state on
```

ユーザー・インターフェース

バックアップを構成するには、「システム・バックアップ」画面 (「管理」 > 「データ管理」 > 「システム・バックアップ」) を使用します。CLI コマンドを使用して Amazon S3 を有効にすると、プロトコルのリストに「Amazon S3」が表示されます。

以下のユーザー入力が必要です。

- S3 Bucket Name (S3 バケット名)。Amazon S3 に保管されるすべてのオブジェクトは、バケット内に格納されます。バケットは、Amazon S3 に保管されるオブジェクトの名前空間をパーティション化します。1 つのバケット内では、保管するオブジェクトに任意の名前を使用できますが、バケット名は Amazon S3 内の全バケットの中で一意である必要があります。
- Access Key ID
- Secret Access Key

バケット名が存在しない場合は、作成されます。

Secret Access Key は、データベースに保存されるときに暗号化されます。

Amazon S3 にファイルがアップロードされたことの確認

1. AWS マネジメント・コンソールに、E メール・アドレスとパスワードを使用してログオンします。

http://aws.amazon.com/console/

1. 「S3」をクリックします。
2. Guardium UI で指定したバケットをクリックします。

Softlayer オブジェクト・ストレージ

SoftLayer オブジェクト・ストレージは、冗長かつハイ・スケーラブルなクラウド・ストレージ・サービスです。 このサービスを使用すると、インターネット上でデータを簡単に保管、検索、取得できます。 これは OpenStack Swift プラットフォームに基づくサービスであり、RESTful API および Web ポータルを使用してアクセスできます。

事前に必要な情報:

- 認証エンドポイント - 認証要求は、ご使用のオブジェクト・ストレージ・アカウントに関連付けられたエンドポイントに送信する必要があります。 https://dal05.objectstorage.softlayer.net/auth/v1.0
- コンテナー - オブジェクト・ストレージ内のすべてのデータの基本ストレージ・ユニットはコンテナーです。ここにはデータ/ファイルが保管され、オブジェクト・ストレージ・アカウントに関連付けられている必要があります。
- X-Auth-User - 認証するユーザー名 (テナント値:ユーザー名)
- X-Auth-Key - 認証する API キー (パスワード)

アカウント資格情報は https://control.softlayer.com/ にログオンすると取得できます。

GUI からの Softlayer によるシステム・バックアップ

1. 「管理」 > 「データ管理」 > 「システム・バックアップ」、「管理」 > 「データ管理」 > 「データ・アーカイブ」、または「管理」 > 「データ管理」 > 「結果アーカイブ」をクリックします。
2. Softlayer プロトコルを選択します。
3. 認証エンドポイント URL を入力します (例: https://dal05.objectstorage.softlayer.net/auth/v1.0)。
4. オブジェクト・ストレージ・コンテナー名を指定します (例: yourname_Container)
5. X-Auth-User (テナント値: ユーザー名) を指定します (例: username)
6. X-Auth Key を入力します (例: password)
7. バックアップ対象 (構成またはデータ) を指定します。
8. 「スケジュールの変更」または「今すぐ 1 回実行」を選択します。

CLI によるシステム・バックアップ (構成)

CLI にアクセスします。

CLI> backup system

1. DATA

2. CONFIGURATION

Please enter the number of your choice: (q to quit) 1

1. SCP

2. CONFIGURED DESTINATION

Please enter the number of your choice: (q to quit) 2

Make sure destination is configured in the GUI under the <System Backup> option

Please wait, this may take some time.

Performing a DEFAULT backup, config=

システム・バックアップおよびシステム・リストア

CLI にアクセスします。

CLI> restore system

1. SCP

2. FTP

3. TSM

4. CENTERA

5. AMAZONS3

7. SOFTLAYER

8. SFTP

Please enter the number of your choice: (q to quit) 7

Enter the SoftLayer Authentication Endpoint URL:

Enter Softlayer Object Storage Container name:

Enter Softlayer X-Auth-User:

Enter X-Auth-Key:

Enter a file name from list:

Authenticate success!

Download file success!

Select your recovery type, for most cases, use the normal option:

1. normal

2. upgrade

システム・バックアップ > Cleversafe

前提条件

Guardium サーバーを正しい現地時間に設定する必要があります。必要に応じて NTPserver サーバーを使用して変更します。

システム・バックアップの選択:

認証エンドポイント URL

(AWS) アクセス・キー

(AWS) 秘密アクセス・キー

バケット名

証明書のすべての質問に対して yes と応答します。

親トピック: Guardium システムの構成

ソケット接続権限の構成

このトピックは、カスタム・アラート・クラスに適用されます。

カスタム・クラスによって使われるすべてのソケット接続の権限を構成するには、この手順に従ってください。

1. 「設定」 > 「評価」 > 「通信の許可」をクリックして、「通信の許可」を開きます。
2. 「ソケット接続権限の追加」 をクリックして、そのペインを拡張します。
3. ホストの IP アドレスまたはホスト名を入力します。
4. ソケット接続のポート番号を入力します。
5. 説明を入力します。
6. 「保存」をクリックします。

親トピック: Guardium システムの構成

アクセス管理の概要

アクセス管理は、アカウントの管理、保守、モニター、および取り消しの 4 つのタスクで構成されています。

アクセス管理は、システム管理の職務とは別個のものです。

Guardium® アプライアンスには、accessmgr および admin という 2 つの事前定義ユーザーがあります。

- *accessmgr* は、アクセス・マネージャーに割り当てられるユーザー名です。デフォルトでは、アクセス・マネージャー が、ユーザー・アカウントおよびセキュリティー・ロールの管理権限を持つ唯一のユーザーになります。
- *admin* は、(1 次) Guardium 管理者に割り当てられるユーザー名です。デフォルトでは、管理者には、ユーザー・アカウントやセキュリティー・ロールを管理する権限がありません。 admin ユーザーは、より広範な一連の特権を持ちます。

注:

admin および accessmgr ロールを、同一ユーザーに割り当てることはできません。既存の状態やアップグレードの結果として、同一ユーザーがこれら両方のロールを持つ場合があります。ただし、現行の使用では、これらの 2 つのロールを同一ユーザーに割り当てることはできません。

以前は、ユニットをアップグレードすると、accessmgr ロールが admin ユーザーに割り当てられ、accessmgr ユーザーが無効にされていました。このアップグレード状態では、 まず admin としてログインして accessmgr ユーザーを有効にした後、 accessmgr としてログインして (初期パスワード「accessmgr」を使用すると、システムから変更を求めるプロンプトがユーザーに出されます)、admin ユーザーから accessmgr ロールを削除する必要がありました。

アクセス管理の選択

- 「ユーザー・ブラウザー」- ユーザーの管理
- 「ロール・ブラウザー」 - アクセス権の管理およびロールのレイアウトのカスタマイズ
- 「ロール権限」- アプリケーションの権限の管理
- 「LDAP ユーザーのインポート」 - LDAP からのユーザーのインポート

データ・セキュリティーの選択

- 関連付けられたデータ・ソース
- 関連付けられていないデータ・ソース
- 関連付けられたサーバー
- 関連付けられていないサーバー
- ユーザー階層
- ユーザー - データベース関連付け

Accessmgr からの事前定義レポート

Accessmgr ユーザーは以下の事前定義レポートを使用できます。

ユーザーとロールのレポート

ユーザーの定義と変更 (『ユーザーの管理』を参照) では、 Guardium システムを使用するユーザーと、そのユーザーに割り当てるロール (『ロールの管理』を参照) の両方を決める必要があります。 ロールとは、そこに属するユーザー全員に同じアクセス権限が割り当てられる、ユーザーのグループです。

ユーザーとロールのレポートには、次のレポートが含まれています。

- ユーザー - ロール -- ユーザーが所属しているロールの数を、ユーザー別に表示するレポート。
- 全ロール - ユーザー -- ロールに属するユーザーの数を、 ロール別に表示するレポート。

注: admin と access manager は既存ですが、その他のロールは access manager によって作成されます。

以下のレポートは、中央マネージャーまたはスタンドアロン・ユニットで使用できます。管理対象マシンで使用しようとすると、エラー・メッセージが表示されます。「関連付けられていないサーバー」では、中央マネージャーのシステム内にあるすべての管理対象ユニットのサーバーが表示されます。

関連付けられたデータ・ソース

このレポートでは、データ・ソース名、ホスト、サービス名、ログイン名、関連付けのタイプが識別されます。 この情報は、「ユーザーとデータベース間の関連」アクティビティーで行った選択から取得されます。ヘルプ・トピック『データ・ユーザー・セキュリティー - 階層および関連付け』を参照してください。

関連付けられていないデータ・ソース

このレポートは、どのユーザーとも関連付けられていないデータ・ソースのリストです。 このレポートでは、データ・ソース名、データ・ソース・タイプ、ホスト、およびサービス名が識別されます。 この情報は、「ユーザーとデータベース間の関連」アクティビティーで行った選択から取得されます。ヘルプ・トピック『データ・ユーザー・セキュリティー - 階層および関連付け』を参照してください。

関連付けられたサーバー

このレポートでは、サーバー IP、サービス名、ログイン名、および関連付けのタイプが識別されます。 この情報は、「ユーザーとデータベース間の関連」アクティビティーで行った選択から取得されます。ヘルプ・トピック『データ・ユーザー・セキュリティー - 階層および関連付け』を参照してください。

関連付けられていないサーバー

このレポートは、どのユーザーとも関連付けられていないサーバーのリストです。 このレポートでは、サーバー IP とサービス名が識別されます。 この情報は、「ユーザーとデータベース間の関連」アクティビティーで行った選択から取得されます。ヘルプ・トピック『データ・ユーザー・セキュリティー - 階層および関連付け』を参照してください。

- ロールについて
  Guardium ユーザーにロールを割り当てて、特定のアクセス権を付与します。ロールの例として、CLI、admin、accessmgr、CAS、および user が挙げられます。
- ロールと権限の管理
  ロールおよびアクセス権により、ユーザーの職務に基づいた各種アクセス・レベルがユーザーに提供されます。
- 最小限のアクセス権しか持たないロールの作成方法
  このトピックでは、最小限のアクセス権しか持たない新規ロール (例えば、監査プロセスの To-do リストへのアクセスおよび特定のレポートの表示のみが可能な監査員ロール) を作成する方法について説明します。
- ユーザーの管理
  ユーザー・アカウントの追加、ユーザー・アカウントの有効化または無効化、LDAP からのメンバーのインポート、またはユーザー権限の編集を行うには、ユーザー名 *accessmgr* が割り当てられたアクセス・マネージャーを使用します。「アクセス」 > 「アクセス管理」 > 「ユーザー・ブラウザー」をクリックして、「ユーザー・ブラウザー」を開き、ユーザー・アカウントを参照します。
- CLI への適切なログイン資格を持つユーザーの作成方法
  このタスクは、CLI を使用して GuardAPI コマンドを実行するための適切なロールとライセンスを持つユーザーを作成するときに使用します。
- LDAP からのユーザーのインポート
  Guardium ユーザー定義を LDAP サーバーからインポートすることができます。これには、該当するユーザーを取得するインポート操作の構成をします。

- 「データ・セキュリティー」- ユーザー階層およびデータベースの関連付け
  データ・セキュリティー機能を使用して、ユーザーの階層を作成し、ユーザーを特定のデータベースおよびサーバーに関連付けることができます。Guardium のデータ・セキュリティー機能は、どのユーザーがどの情報にアクセスしたかを報告し、確実に特定のユーザーのみが自分の担当している情報を表示できるようにします。
- ユーザー階層の定義方法
  アクセス・マネージャー・アカウントから UI を使用すると、容易にユーザー階層を定義できます。
- スマート・カードを使用した Guardium UI へのログイン
  Guardium のスマート・カード・サポートは、すべてのベンダーがユーザー・アクセスで多要素認証をサポートする必要があるという米国政府の義務付けを満たしています。スマート・カード認証がサポートされるのは、Web ベースの Guardium ユーザー・インターフェース (UI) へのアクセスのみです。

ロールについて

Guardium ユーザーにロールを割り当てて、特定のアクセス権を付与します。ロールの例として、CLI、admin、accessmgr、CAS、および user が挙げられます。

アクセス・マネージャーはロールを定義し、それをユーザーおよびアプリケーションに割り当てます。 ロールがアプリケーションまたは項目の定義 (特定の照会など) に割り当てられると、そのロールを割り当てられた Guardium ユーザーのみがそのコンポーネントにアクセスできます。

ユーザー定義が LDAP サーバーからインポートされるときに、その定義が属するグループは、オプションでロールとして定義できます。 詳しくは、LDAP からのユーザーのインポートを参照してください。

注: ユーザーにロールを割り当てるときに、admin ロールとアクセス・マネージャー・ロールを同一ユーザーに割り当てることはできません。
注: カスタム作成のロールを、デフォルトで提供されるロール (例: user、admin、accessmgr、cli、inv、datasec-exempt、review-only) と組み合わせて使用することはできません。
注: admin ロールおよびオブジェクトの所有者は、デフォルトですべてのオブジェクトにアクセスできます。
注: 基本ロールを選択して (ナビゲーション項目を追加して) カスタマイズした後、 そのカスタマイズしたロールをコピーした場合、カスタマイズまたはコピーしたロールをデフォルトにリセットすると、 カスタマイズ内容が失われます。

デフォルトのロール

Guardium システムは、管理者、ユーザー、アクセス・マネージャー、および調査という、大きく 4 つのデフォルトのロールに分類されるユーザーをサポートするために事前構成されています。 Guardium アクセス・マネージャーは新しいロールを作成することもできます。
注: 注: 監視データ・レベルにおいてデータ・レベル・セキュリティーが有効になっている場合 (「グローバル・プロファイル」設定を参照)、監査プロセスのエスカレーションはデータ階層の上位レベルのユーザーに対してのみ許可されます (『アクセス・マネージャー』を参照)。Datasec-exempt ユーザーは誰に対してでも、無制限にエスカレーションできます。

表 1. デフォルトのロール

デフォルトのロール	記述
user	すべての一般ユーザーにデフォルトのレイアウトとアクセス権限を提供します。このロールは削除できません。
admin	Guardium 管理者のデフォルトのレイアウトとアクセス権限を提供します。admin ロールと admin ユーザー を混同しないでください。後者は admin ロールを持つ特別のユーザー・アカウントですが、admin ユーザー・アカウントのみに用意されている追加の権限も持っています。このロールは削除できません。
accessmgr	アクセス・マネージャーのデフォルトのレイアウトおよびアクセス権限を提供します。このロールは削除できません。
cli	CLI へのアクセスを提供します。admin ユーザーは CLI に対してデフォルトのアクセス権限を持っています。その他のユーザーは、アクセス・マネージャーにより作成され、ロールが指定されたときにアクセス権を付与される必要があります。 アクセス・マネージャーはシステム内で同じ数だけのユーザーを定義し、そのユーザーに CLI ロールを付与することができます。 これらのユーザーは CLI に対するアクセス権を持ち、そのユーザーの CLI セッションのすべてのアクティビティーはこのユーザーに関連付けられます。 admin 権限を持っていないユーザーが GrdAPI コマンドまたは CLI コマンドを実行するには、ユーザー・ロール・アクセス権の選択で、「管理コンソール」に対して「CLI」ロールをクリックします。 diag ロールの管理方法については、トピック『diag CLI コマンド』を参照してください。
inv	調査ユーザーのデフォルトのレイアウトとアクセス権限を提供します。 調査ユーザーはそのユーザー定義の姓として、リストア先データベース名である INV_1、INV_2、または INV_3 を持つ必要があります。これは GUI で強制されませんが、調査アプリケーションが正しく機能するために必要です。 名前が割り当てられるときに、ユーザー・ロールも割り当てられる必要があります。 このロールは削除できません。 注:「今すぐ 1 回実行する特別プロセス」ボタンは、調査 (INV) ユーザーを除くすべてのユーザーが、すべてのレポート画面で使用できます。
datasec-exempt	データ・セキュリティー - 免除。このロールは、データ・レベルのセキュリティーが有効にされ (管理コンソールの「グローバル・プロファイル」を参照)、datasec-exempt ロールが割り当てられたときに活動化します。ユーザーがこのロールを持っている場合には、「すべて表示」チェック・ボックスがすべてのレポートに表示されます。チェック・マークを付けると、検出されたすべてのデータ記録が表示されます (フィルター処理は一切行われません)。このロールは ロール・ブラウザーでは削除できません。
review-only	このロールで指定されたユーザーは、各種結果 (監査、評価、分類)、監査結果、および To Do リストのみを表示できます。このロールは ロール・ブラウザーでは削除できません。 このロールを持つユーザーは、監査プロセス・ビューアーでコメントを入力できます (行単位のワークフローやコメント/データではなく、処理/結果レベルのコメント)。 このロールを持つユーザーは、ワークフロー自動化結果 (エスカレート、再割り当てなど) に対して、いかなる変更/アクションを実行することもできません。

サンプル・ロール

デフォルトのロールに加えて、一連のロールのサンプルも定義されています。

表 2. サンプル・ロール

サンプル・ロール	記述
dba	セキュリティーでデータベースを中心とした監視を行うユーザー。データベース関連のレポートにアクセスでき、データベース・オブジェクトのトラッキングを行います。
infosec	機密保護に重点的に関与するユーザー。データベースへのアクセスのトラッキング、ネットワーク要求、監査、およびフォレンジックの処理などを行います。
netadm	ネットワークを中心とした監視を行うユーザー。データベース要求の IP 送信元などを監視します。
appdev	アプリケーション開発者、アーキテクト、および QA 担当者。アプリケーションを中心とした監視を行い、アプリケーションにより生成された SQL ストリームのトラッキングとそれに関するレポートを行います。
audit	監査レポートを表示する必要がある監査員およびその他のユーザー。 注: このロールをコピーしようとすると、このロールのすべての側面をコピーできるわけではないことを示す組み込みメッセージが表示されます。メッセージは以下のとおりです。「"audit" ロールのレイアウトおよびアクセス権を使用して新しいロールを作成します。"audit" ロールに関連付けられた特権および特別なアクションはコピーされません。」
audit-delete	監査プロセスの結果が削除されている場合、このロールを使用して、トラッキングまたはロギングが実行されます。audit-delete ロールを所有するユーザーは、レポートを削除できます。管理ユーザーも、レポートを削除することができます。トラッキングは、ユーザー・アクティビティー監査証跡レポートを使用して実行されます。
admin-console-only	このロールで指定されたユーザーは、管理コンソール・タブにのみアクセスできます。
cas	構成監査システム (CAS)
vulnerability-assess	このロールで指定されたユーザーは、脆弱性結果のみを表示できます。
diag	このロールで指定されたユーザーは、CLI で diag コマンドにアクセスして実行することができます。
workload-replay-admin	このロールで指定されたユーザーは、ワークロード・リプレイ機能を定義および変更することができます。
workload-replay-user	このロールで指定されたユーザーは、ワークロード・リプレイ機能を実行できます。
fam	このロールで指定されたユーザーは、ファイル・アクティビティー・モニター機能を定義および変更することができます。
BaselII	アクセラレーター - Basel II。このロールは削除できません。 バーゼル II の第 2 部のセクション 4 とセクション 5 の要件では、金融機関が財務情報をもとに証券化の枠組みを定義し、それに伴う運営上のリスクを推定しなければならないことが定められています。
DataPrivacy	アクセラレーター - データ・プライバシー。このロールは削除できません。 データ・プライバシー・アクセラレーターは、特に ID の盗用に対して保護を施し、業界のベスト・プラクティスに基づく、事前に定義したポリシー、リアルタイム・アラート、および監査レポートのポートフォリオを提供しています。データ・プライバシー・アクセラレーターを使用すると、セキュリティー管理者、プライバシー保護担当者、およびデータベース管理者は、データ要素 (「プライバシー・セット」という) の組み合わせを定義することから作業を開始できます。これらに対するアクセスは、内部ユーザーによるハッキングまたは不適切なアクティビティーを示している可能性があります。
GDPR	アクセラレーター - GDPR。このロールは削除できません。 Guardium GDPR アクセラレーターは、データに対する GDPR ワークフローと、GDPR のグループおよびポリシーに基づく事前定義レポートを提供します。GDPR アクセラレーターの処理を開始するには、GDPR ロールを Guardium ユーザーに割り当て、そのユーザーのアカウントを使用して「アクセラレーター」 > 「GDPR」にナビゲートします。
GDPR FAM	アクセラレーター - GDPR FAM。このロールは削除できません。 Guardium GDPR FAM アクセラレーターは、ファイル・サーバーに対する GDPR ワークフローと、GDPR のグループおよびポリシーに基づく事前定義レポートを提供します。GDPR FAM アクセラレーターの処理を開始するには、GDPR FAM ロールを Guardium ユーザーに割り当て、そのユーザーのアカウントを使用して「アクセラレーター」 > 「GDPR」にナビゲートします。
pci	アクセラレーター - PCI。このロールは削除できません。 PCI DSS は、カード所有者データを保護するために設計された一連の技術要件と運用要件であり、カード所有者データの保管、処理、使用、または送信を行うすべての組織に適用されます。この要件に準拠できない場合、特権の喪失や厳しい罰金のほか、データ・ブリーチの発生時にはブランドやサービスに関する消費者の信頼感の著しい低下が伴う可能性があります。IBM Guardium アクセラレーターは、事前定義ポリシー、レポート、グループ定義などを使用して、この規格の各部分に準拠するプロセスを実施する上で役立ちます。
sox	アクセラレーター - SOX。このロールは削除できません。 SOX 法の第 404 条では、財務報告について会社ごとに適切な内部統制機構と手続きを確立し、維持していくことが求められています。

中央マネージャー環境のロール

中央マネージャー環境では、すべてのユーザー・アカウント、ロール、およびアクセス権が中央マネージャーによって制御されます。 これらの定義のいずれかを管理するには、中央マネージャーにログインしている必要があります (管理対象ユニットにではなく)。

ロールの作成

1. accessmgr としてログインし、「アクセス」 > 「アクセス管理」 > 「ロール・ブラウザー」をクリックして、「ユーザー・ロール・ブラウザー」を開きます。
2. 「ロールの追加」をクリックして、「ロール・フォーム」パネルを開きます。
3. 「ロール名」に固有の名前を入力して、「ロールの追加」をクリックします。

ロールの削除

1. 「アクセス」 > 「アクセス管理」 > 「ロール・ブラウザー」をクリックして、「ユーザー・ロール・ブラウザー」を開きます。
2. いずれかのロールの「削除」をクリックします (一部のロールは削除できません。そのようなロールには、「削除」オプションはありません)。 すると、そのロールの「ロール・フォーム」が開きます。
3. 「削除の確認」をクリックします。そのロールへのすべての参照が削除されることを通知するメッセージが表示され、操作を確認するよう求められます。
4. 削除を確認するには「OK」を、操作を中止するには「キャンセル」をクリックします。

親トピック: アクセス管理の概要

ロールと権限の管理

ロールおよびアクセス権により、ユーザーの職務に基づいた各種アクセス・レベルがユーザーに提供されます。

ロールには user、admin、audit などがあります。ロールを使用すると、ユーザー・グループ全体に対するアクセス権を容易に定義できます。新規ロールを作成したりそのロールにユーザーを割り当てたりすることができるのは、アクセス・マネージャーのみです。アクセス・マネージャーは、ロール作成の一環として、そのロールのナビゲーション・メニューおよびアクセス権をカスタマイズすることもできます。

カスタマイズしたロールを作成するには、以下のような、いくつかの処理が必要です。

- 新規ロールを作成する
- ロールのアクセス権を管理し、ユーザーがアクセスできる対象を制限する
- オプションで、ロールのナビゲーション・メニューをカスタマイズし、ユーザーが表示可能な内容をさらに制限する
- ユーザーをロールに追加する

特定のアプリケーションへのアクセスを制限するには、以下の 2 つの方法があります。

アプリケーションからのアクセスの制限

> アプリケーションからのアクセスを制限するには、「ロール権限」 > 「アプリケーション・ロール権限の編集」画面で「すべてのロール」チェック・ボックスを選択解除します。次に、アプリケーションにアクセスする必要がある個々のロールを選択します。
>
> この処理は、「すべてのロール」チェック・ボックスが既に選択解除されている場合も同じです。 単に、アプリケーションへのアクセス権を付与するか取り消すロールを個別に選択または選択解除します。
>
> 特定のアプリケーションに対して「すべてのロール」が選択されている場合は、現在定義されているすべてのロールがそのアプリケーションにアクセスできます。

ロールからのアクセスの制限

> ロールからのアクセスを制限するには、 「ロール・ブラウザー」 > 「権限の管理」画面にナビゲートし、 アプリケーションを個別に「アクセス可能なアプリケーション (Accessible applications)」リストから「アクセス不能なアプリケーション (Inaccessible applications)」リストに移動します。
>
> アクセス権を管理したり新規ロールのナビゲーション・メニューをカスタマイズしたりすると、 「アクセス可能なアプリケーション (Accessible applications)」リストに表示されるデフォルトに、 「ロール権限」 > 「アプリケーション・ロール権限の編集」画面で 「すべてのロール」チェック・ボックスを選択したアプリケーションが反映されます。

ロールおよびアクセス権を操作するときにアプリケーションのアクセス権を削除すると、新規ロールのデフォルト・アクセス権も変化します。つまり、アプリケーションのアクセス権を削除すると、その後に作成するすべてのロールでもそのアプリケーションに対するアクセス権が欠落します。デフォルトで「アクセス可能なアプリケーション (Accessible applications)」リストに表示されなくなったアプリケーションのアクセス権を新規ロールに付与する場合は、その新規ロールに対して必要なアプリケーションを「アクセス不能なアプリケーション (Inaccessible applications)」リストから「アクセス可能なアプリケーション (Accessible applications)」リストに移動する必要があります。

「ロール・ブラウザー」 > 「ナビゲーション・メニューのカスタマイズ (Customize Navigation Menu)」ツールを使用してメニュー項目を非表示にすることで、アクセス権を特定のツールに制限することもできます。この方法では、デフォルトのアプリケーション・アクセス権を変更せずにアクセスが制限されますが、アクセス権ベースの方法より安全性が低くなる場合があります。

ベスト・プラクティス:

- ロールのアクセス権を編集した後、 「ロール・ブラウザー」 > 「ナビゲーション・メニューのカスタマイズ (Customize Navigation Menu)」画面に表示される、そのロールのナビゲーション・レイアウトを確認します。必要に応じて「ナビゲーション・メニュー (Navigation Menu)」リストの項目を追加または削除して、ロールに適したレイアウトを作成します。
- 事前定義のロールをコピーしてから編集し、必要なアクセス権およびナビゲーション・メニューを設定します。この方法を行えば、 必要に応じて元のロールに戻すことができます。

親トピック: アクセス管理の概要
関連タスク:
最小限のアクセス権しか持たないロールの作成方法
関連情報:
ユーザー・インターフェースのカスタマイズ (Customizing the user interface)
ユーザー、ロール、および Guardium システムの管理 (ビデオ)

最小限のアクセス権しか持たないロールの作成方法

このトピックでは、最小限のアクセス権しか持たない新規ロール (例えば、監査プロセスの To-do リストへのアクセスおよび特定のレポートの表示のみが可能な監査員ロール) を作成する方法について説明します。

手順

1. 新規ロールを作成します。
   a. *accessmgr* としてログインし、 「アクセス」 > 「アクセス管理」にナビゲートして、「ロール・ブラウザー」を選択します。
   b. 「ロールの追加」ボタンをクリックし、ロールに名前を指定して 「ロールの追加」ボタンをクリックし、新規ロールを作成します。

2. 新規ロールが「監査プロセスの To-do リスト」および「レポート・ビルダー」 (レポートを表示するために必要です) のみにアクセスできるように、アクセス権を管理します。
    a. 「ロール・ブラウザー」で、新規ロールの「権限の管理」リンクをクリックします。
    b. 「アクセス可能な項目」リストのヘッダーにあるチェック・ボックスを選択し、 矢印を使用してすべての項目を「アクセス不能な項目」リストに移動します。 制限の厳しいロールを作成するときは、まずアクセス権を削除するほうが簡単です。
    c. 「アクセス不能な項目」リストで 「監査プロセスの To-do リスト」および「レポート・ビルダー」を選択し、矢印を使用して「アクセス可能な項目」リストに戻します。 これで、新規ロールがこれら 2 つのアプリケーションのみにアクセスできるようになりました。
    d. 「OK」ボタンをクリックして、変更内容を確定します。
3. メニューおよびナビゲーションをカスタマイズするために、新規ロールで使用できるレポートおよびアプリケーションを定義します。
    a. 「ロール・ブラウザー」で、新規ロールの「ナビゲーション・メニューのカスタマイズ (Customize Navigation Menu)」リンクをクリックします。
    b. 「ナビゲーション・メニュー (Navigation Menu)」リストで、「レポート」グループを選択します (強調表示されます)。 選択したグループが、以降のステップで追加するメニュー項目の宛先となります。
    c. 「使用可能なツールとレポート (Available Tools and Reports)」リストで、 「レポート」セクションを展開するか「フィルター」を使用して特定のレポートを探し、 新規ロールで使用可能にする必要がある項目それぞれの横にあるチェック・ボックスを選択した後、 矢印を使用して、その項目を「ナビゲーション・メニュー (Navigation Menu)」リストに追加します。 「ナビゲーション・メニュー (Navigation Menu)」リストに移動した項目が、 このロールに割り当てられたユーザーに表示されるようになります。
    d. 「ナビゲーション・メニュー (Navigation Menu)」リストで、 「レポート」 > 「レポート構成ツール」および「調査」グループの横にある アイコンをクリックして、 「レポート・ビルダー」へのアクセス権を削除します。 これにより、このロールのメニュー構造が簡素化され、 「レポート・ビルダー」ツールへのアクセス権が削除されますが、 レポートにアクセスする必要があるアプリケーション・アクセス権は削除されません。
    e. 「OK」ボタンをクリックして、変更内容を確定します。 これで、ユーザーへの割り当てが可能なごく最小限の特権を持つ新規ロールが作成されました。
4. オプションで、新規ロールのカスタム・ホーム・ページを指定します。
    a. 「ロール・ブラウザー」で、新規ロールの「ナビゲーション・メニューのカスタマイズ (Customize Navigation Menu)」リンクをクリックします。
    b. 「ナビゲーション・メニュー (Navigation Menu)」リストで、 「順守」 > 「ツールとビュー」 > 「監査プロセスの To-do リスト」を選択してから、ツールバーの アイコンをクリックし、 新しいデフォルト・ホーム・ページを指定します。 このロールに割り当てられたユーザーがログインすると、 デフォルト画面として「監査プロセスの To-do リスト」が表示されるようになります。
    c. 「OK」ボタンをクリックして、変更内容を確定します。
5. 新規ユーザーを作成し、そのユーザーをこの新規ロールに追加します。
    a. 「アクセス」 > 「アクセス管理」にナビゲートして 「ユーザー・ブラウザー」を選択します。
    b. 「ユーザーの追加」をクリックし、必要な情報を指定して、「ユーザーの追加」をクリックし、新規ユーザーを作成します。 作成したユーザーが「ユーザー・ブラウザー」にリストされます。

       新規ユーザーを作成したとき、アカウントはデフォルトで無効になっています。ユーザーがアカウントに即時アクセスできるようにする場合は、 「無効」チェック・ボックスを選択解除します。

    c. 「ユーザー・ブラウザー」で新規ユーザーの「ロール」リンクをクリックすると、 使用可能なロールのリストが表示されます。
    d. 以前に作成したカスタム・ロールの横にある「割り当て」チェック・ボックスを選択します。 これにより、新規ロールにユーザーが割り当てられます。
    e. *user* ロールの横にある「割り当て」チェック・ボックスを選択解除します。 *user* ロールを選択解除すると、新規ユーザーがデフォルトの *user* アクセス権および許可を継承できなくなります。
    f. 「保存」をクリックして、変更内容を確定します。

親トピック: アクセス管理の概要
関連概念:
ロールと権限の管理

ユーザーの管理

ユーザー・アカウントの追加、ユーザー・アカウントの有効化または無効化、LDAP からのメンバーのインポート、またはユーザー権限の編集を行うには、ユーザー名 *accessmgr* が割り当てられたアクセス・マネージャーを使用します。「アクセス」 > 「アクセス管理」 > 「ユーザー・ブラウザー」をクリックして、「ユーザー・ブラウザー」を開き、ユーザー・アカウントを参照します。

ユーザーの定義と変更には、Guardium® システムを誰が使用するのか、およびその人たちにどのロールを割り当てるかの両方を決定する作業が含まれます。 ユーザーのグループは、すべて同じロールと同じアクセス権を持つことができます (そのように選択する場合)。 ロールについて詳しくは、ロールについてを参照してください。

注: ロールにデフォルトのレイアウトを定義して、そのロールを割り当てられたすべての新規ユーザーがそのレイアウトを持つようにすることができます。「CLI リファレンス」の 『新規レイアウトの生成』を参照してください。

ユーザー定義は LDAP サーバーからオンデマンドで、またはスケジュールに従ってインポートできます。

ユーザーがどのように Guardium システムに定義されているかに関係なく、Guardium 管理者は Guardium、LDAP、または Radius を介してユーザーを認証するようにシステムを構成できます。

Guardium システムを開始する際の初期の重要なタスクは、そのシステムを使用するのはどのユーザー・グループで、そのグループの機能は何かを確認することです。例えば、情報セキュリティー・グループはアラートおよびトラブルシューティングのために Guardium を使用し、データベース管理者グループはレポートとモニターのために Guardium を使用します。Guardium システムにアクセスするユーザーを決定するときには、会社の機密データがこのシステムによって取り出される可能性があることを念頭においてください。 したがって、そのデータに誰がアクセスできるかには十分に注意を払ってください。

どのユーザー・グループが Guardium システムを使用するか (そして何の目的で使用するか) を決定したら、それぞれのユーザーについて次の情報を収集します。

- ユーザーの姓名
- ユーザーのアカウント名 (ログインに使用する名前)
- ユーザーの E メール・アドレス
- ユーザーの国または場所
- Guardium でのユーザーの機能/ロール

ユーザー・アカウントのセキュリティー

ユーザー・アカウントに対して追加のセキュリティーを提供するために、いくつかの設定を変更できます。 これらの設定は CLI コマンドの show および store password を使用して有効または無効にできます (「CLI リファレンス」の『ユーザー・アカウント、パスワード、および認証 CLI コマンド』を参照)。

- デフォルトで、パスワードの検証は有効になっています。これは、パスワードには最低 8 文字が必要で、次のそれぞれの種類の文字が少なくとも 1 文字含まれていなければならないことを意味します。
  - A から Z の英大文字
  - a から z の英小文字
  - 0 から 9 の数字
  - 特殊文字: @#$%^&.;!-+=_

  注: パスワード検証が無効になっている場合はすべての文字を使用できます。
- デフォルトでは、パスワードの有効期限は有効になっています。指定した日数が経過したら有効期限が切れるようにパスワードを構成できます。
- デフォルトでは、ログイン試行の失敗が指定された回数に達したときのアカウントのロックアウトは有効になっています。一定時間内の試行回数が一定数に達した後、またはアカウントが存続する期間中の試行回数の累計が一定数に達した後でロックアウトが発生するように構成できます。

ロックされたアカウント

1. 「アクセス」 > 「アクセス管理」をクリックして、「ユーザー・ブラウザー」を開き、ユーザーのリストを表示します。
2. 任意のユーザーの「編集」をクリックして、「無効」チェック・ボックスをクリアし、「ユーザーの更新」をクリックして、変更内容を保存します。
   注: admin ユーザー・アカウントがロックされた場合は、unlock admin CLI コマンドを使用してこれをアンロックします (「CLI リファレンス」の『構成および制御 CLI コマンド』を参照してください)。

ユーザー・アカウントの作成

1. 「ユーザー・ブラウザー」を開き、「ユーザーの追加」をクリックして、「ユーザー・フォーム」パネルを開きます。
2. 「ユーザー名」に固有の名前を入力します。名前にはアポストロフィ文字を含めないでください。ユーザー名には大/小文字の区別がありません。
   注: 「ユーザーの追加」パネルまたは「ユーザー LDAP インポート」のいずれかから手動でユーザーを追加するときに、ファーストネームとラストネームのどちらかまたは両方がない場合は、ログイン名が使用されます。
3. パスワードを入力して、「パスワード (確認)」ボックスにもう一度入力して確認します。割り当てたパスワードは一時的なもので、ユーザーは最初のログインの後、これを変更するよう要求されます。
   注: パスワードは大/小文字の区別があります。 パスワード検証が有効 (デフォルト) になっている場合、パスワードは 8 文字以上の長さでなければならず、さらに、英大文字 (A-Z)、英小文字 (a-z)、数字 (0-9)、および特殊文字 (@$%^&.;!-+=_) を、それぞれ 1 つ以上含んでいる必要があります。
   注: ユーザー名での非ラテン文字 (中国語や日本語など) の使用はサポートされていません。
4. ユーザーのファーストネームとラストネームをそれぞれのフィールドに入力します。
   注: 調査センターのロール (inv) を割り当てられているユーザーのラストネームには、制限が適用されます。ユーザーにこの調査ロールを割り当てる場合、そのユーザーのラストネームは INV_1、INV_2、INV_3 でなければなりません。UI では、このフィールドに別の名前を入力することを制限しませんが、前述のラストネームが入力されなければ、アプリケーションは適切に機能しなくなります。また、調査ユーザーにその他のロールを割り当てることはできません。inv のみでなければなりません。これは、user ロールまたは admin ロールが必要とされない唯一の場合です。
5. (オプション) ユーザーの E メール・アドレスを入力します。
6. (オプション) ドロップダウン・メニューからユーザーの国または場所を選択します。
7. (注意) 「無効」チェック・ボックスには、デフォルトでチェック・マークが付けられています。チェック・ボックスをクリアし、アカウントを有効にするのは、一連の正しいロールがユーザーに対して割り当てられた後まで待つようお勧めします。

   ユーザーが初回にログインしたときにそのレイアウトにすべてのコンポーネントが含まれるよう、最初にロールを割り当てておく方がずっと簡単です。 ユーザーが初回にログインするときに、そのレイアウトはその時点で割り当てられているすべてのロールを使用して作成されます。後でロールが追加された場合、ユーザーはそのロールで使用可能なすべての対象にアクセスできますが、そのロールに固有のレポートまたはアプリケーションを手動で追加しなければなりません。
8. 「ユーザーの追加」をクリックして新しいユーザー・アカウント定義を保存し、パネルを閉じます。

これでユーザーの定義は完了です。初回ログインのパスワードをユーザーに通知する前に、ユーザーのために適切なロールを追加することをお勧めします。詳細については、ロールについてを参照してください。

複数のユーザーの有効化/無効化

「ユーザー・ブラウザー」を開いて「ユーザーの検索」をクリックすると、ユーザーをロール別に簡単にフィルタリングできます。ユーザーの選択時には、ユーザーを有効または無効にするオプションがあります。ユーザーはデフォルトで無効になっているため、このメニューは、複数のユーザーの状況を容易に変更するのに非常に便利です。

ユーザー・アカウントの更新

1. 「ユーザー・ブラウザー」を開き、変更を加えるユーザーの「編集」をクリックします。
2. 「ユーザー・フォーム」パネルの任意の値を置き換えます。
3. 「ユーザーの更新」をクリックして、変更内容を保存します。

注: ユーザーのパスワードを変更するには、そのユーザーが次回にログインした後に自身でパスワードを変更する必要があります。

無効なユーザー・アカウントを有効にする

1. 「ユーザー・ブラウザー」を開き、有効にするユーザーの「編集」をクリックします。
2. 「無効」チェック・ボックスをクリアします。
3. ユーザーがパスワードを忘れてしまった場合は、新規パスワードを「パスワード」と「パスワード (確認)」の両方のボックスに入力します。
4. 「ユーザーの更新」をクリックします。

ユーザー・アカウントの削除

1. 「アクセス」 > 「アクセス管理」をクリックして、「ユーザー・ブラウザー」を開きます。

2. 削除するユーザーの「削除」をクリックします。
3. 「削除の確認」をクリックします。

注: 削除されたユーザーに送信されていたアラートは、今度は管理者に送信されるようになります。ただし、これはアクセス・ポリシーが再インストールされるまで有効になりません。

データ・セキュリティー・ユーザー階層を定義する

1. 「データ・セキュリティー」 > 「ユーザー階層」をクリックします。
2. 「ユーザー」メニューからユーザーを選択して画面を最新表示し、選択したユーザーの現在の階層をユーザー・ペインに表示します。
3. ユーザー・ノードを右クリックして、以下のオプションを表示します。
   - ユーザーの追加 - 「ユーザーの追加」をクリックすると、「ユーザーの追加」ダイアログが表示されます。検索するか、ロール別にフィルタリングして、選択したユーザーの子孫としてユーザーを追加します。

     こうして、ある階層の親には特定のサーバーおよびデータベースの表示を許可し、その階層の子には許可しないことで、データ・レベルのセキュリティー軽量のための手段を作成できます。 構成によっては、子のデータ・レベル・セキュリティーを親が継承するという継承を行うこともできます。

     注: ユーザーが複数の親を持つことができ、かつ、親が複数のユーザーを持つことができる、「多対多」の関係が許可されます。
   - 親からユーザーをリンク解除 - 子孫を親から切り離します。
   - すべての子孫を削除 - すべての子孫を親から切り離します。
4. 「キャッシュ階層のリフレッシュ」をクリックして、最近の変更内容をユーザー階層マップに適用します。
5. 「アクティブなユーザー - DB マップの全更新」をクリックして、最近の変更内容を、アクティブなユーザー - DB 関連付けマップにすべて適用します。
   注: ベスト・プラクティスではユーザー階層の変更後に「アクティブな「ユーザー - DB」マップの全更新」を行います。

   階層またはデータベースへの関連付けに (UI または GuardAPI を介して) 変更を加えたとき、その変更内容は自動的には有効になりません。「定期更新」が実行されたのが「初めて」でない限り、「定期更新」はこの変更内容をピックアップしません。 それ以外の場合、変更内容を有効にするには、ユーザーは「全更新」をクリックするか、Full Update GuardAPI コマンドを実行する必要があります。

   ユーザー階層の定期更新は、10 分ごとに自動的に実行されます。 手動では実行できません。これはインクリメンタル更新です。つまり、最後に定期更新が実行されたとき以降に検出された新しいサーバー IP またはサービス名のみを検査しているということです。 既存の階層および関連付けを新しい IP/サービス名と比較し、どのユーザーがこれらの IP/サービス名へのアクセス権限を持つ必要があるかを特定します。

   ユーザー階層の全更新は自動的には実行されません。 UI または GuardAPI 関数を介して行われる場合にのみ、実行されます。これは、すべての IP/サービス名を既存の階層および関連付けと比較し、誰が何に対するアクセス権限を持つかを特定します。

データ・セキュリティー・ユーザーとデータベースの関連付けを定義する

「ユーザーとデータベースの関連付けによるデータ・セキュリティー」を使用して、ユーザーの検索、使用可能なサーバーおよびサービス名 (データベース) へのユーザーの割り当て、およびそれらからのユーザーの削除を行います。

1. 「データ・セキュリティー」 > 「ユーザー - データベース関連付け」をクリックして、「ユーザー - データベース関連付け」パネルを開きます。
2. 「サーバーおよびサービス名の推奨」のチェック・ボックスを選択して、ユーザーに関連付けるデータベースとサービス名を見つけます。次のような選択肢があります。
   - 監視対象アクセス - Guardium 内部データベース表「GDM_Access」からの監視対象トラフィック。
   - データ・ソース定義 - データ・ソースの名前、データベース・タイプ、認証情報、およびロケーションなどの既存のデータ・ソース定義情報。
   - S-TAP® 定義 - データベース・サーバーの IP アドレスや、S-TAP からデータを受け取る Guardium ホストの IP アドレスなどの既存の S-TAP 定義情報。
   - 自動検出されたホスト - Guardium オートディスカバリー・プロセスによって検出され、以前は認識されていなかったホスト。Guardium のオートディスカバリー・アプリケーションは、ネットワークをプローブして、データベースを検索し、 ディスカバーしたすべてのデータベースについてレポートを作成するように構成できます。
   - Guardium Install Manager (GIM) が検出したシステム - GIM によって検出され、以前は認識されていなかったホスト。
3. 「移動」をクリックし、使用可能なサーバー、サービス名、および現在関連付けられているユーザーを見つけて表示します。
   注: ノード・ツリーをトラバースする際に、各サーバーおよびサービス名の横に数値標識が表示され、直接関連付けられたユーザーおよび関連付けられた子孫ユーザーの数が示されます。この標識は直接関連付けの場合は [nn] の形式をとり、子孫の関連付け (例えば、現在のサーバー内のサーバーまたはサービス名にユーザーが関連付けられている) の場合は (mm) の形式をとります。同様に、サーバーまたはサービス名に関連付けられているユーザーを表示するとき、ツリーのより上位レベルのノードに関連付けられているユーザーがいる場合は、そのユーザーが表示されます。
4. サーバーまたはサービス名のノードをクリックして、関連付けられているユーザーを表示します。 任意のノードを選択して、以下のいずれかを実行できます。
   - ユーザーとデータベースの関連付けを新規に追加するには、「ユーザーの追加」をクリックし、追加する任意のユーザー (複数可) をクリックしてから、「追加」をクリックします。
   - グループとデータベースの関連付けを新規に追加するには、「グループの追加」をクリックします。「グループの追加」を選択すると、「グループ・ビルダー」を使用してグループ・タイプ「Guardium ユーザー」として作成されたグループが表示されます。追加するグループを選択して、「追加」をクリックします。
   - サーバーまたはサービス名のノードを右クリックして、以下のいずれかを実行します。
5. サーバーまたはサービス名のノードを右クリックすると、以下のいずれかを実行するためのオプションが表示されます。
   - サーバーを強調表示する。
   - サーバーを展開または縮小する。
   - サーバーを検索する。
   - サーバー、サービス名、または無名サービスを追加する。
   - サーバーを削除する。
6. ツリー構造の前にある「IP」フィールドと「サービス名」フィールドを使用して、IP または IP/サービス名のペアを追加します。
   注: 「検索」ボタンを使用して、IP/サービス名ツリー構造を検索することができます。IP 文字列は部分的に入力したり、ワイルドカード * を含めることができます。例えば、「192.168」と「192.168.*.*」はどちらも有効です。ただし、ワイルドカードの後ろに数値を入れることはできず、数値とワイルドカードでオクテットを形成することはできません。サービス名には任意の場所にワイルドカード「%」を含めることができます。
7. 「アクティブなユーザー - DB マップの全更新」をクリックして、最近の変更内容を、アクティブなユーザー - DB 関連付けマップにすべて適用します。
   注: 「ユーザー - データベース関連付け」を変更してから、「アクティブなユーザー - DB マップの全更新」を行うのがベスト・プラクティスです。

   ユーザー階層の全更新は自動的には実行されません。 「アクティブなユーザー - DB マップの全更新」ボタンまたは GuardAPI 関数を介して行われる場合にのみ、実行されます。これは、すべての IP/サービス名を既存の階層および関連付けと比較し、誰が何に対するアクセス権限を持つかを特定します。

ユーザー階層の定期更新は、10 分ごとに自動的に実行されます (手動では実行できません)。この更新は、最後に定期更新が実行されたとき以降に検出された新しいサーバー IP またはサービス名のみを検査します。既存の階層および関連付けを新しい IP/サービス名と比較し、どのユーザーがこれらの IP/サービス名へのアクセス権限を持つ必要があるかを特定します。

データベースの関連付けに (UI または GuardAPI を介して) 変更を加えたとき、その変更内容は自動的には有効になりません。「定期更新」が実行されたのが「初めて」でない限り、「定期更新」はこの変更内容をピックアップしません。それ以外の場合、変更内容を有効にするには、「アクティブなユーザー - DB マップの全更新」ボタンをクリックするか、full update GuardAPI コマンドを実行する必要があります。

親トピック: アクセス管理の概要

CLI への適切なログイン資格を持つユーザーの作成方法

このタスクは、CLI を使用して GuardAPI コマンドを実行するための適切なロールとライセンスを持つユーザーを作成するときに使用します。

このタスクについて

次の理由から、この how-to トピックは重要です。 (1) GuardAPI コマンドは CLI からしか実行できません。 (2) ほとんどの GuardAPI コマンドは特定のアプリケーションとそのロールに関連付けられているため、適切なロールを持たない 標準の CLI ユーザー (ハードコーディングされた「admin」ロールを持つユーザー) では実行できない GuardAPI コマンドが多数あります。

手順

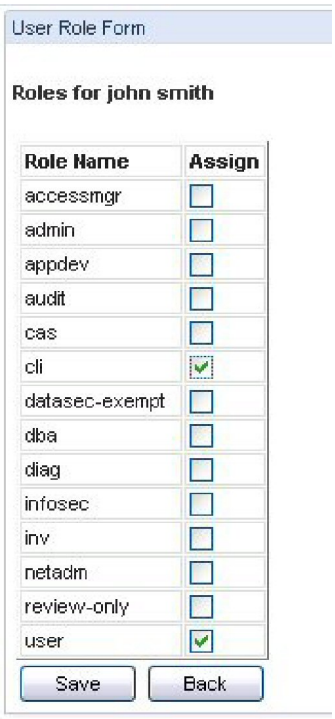
1. accessmgr としてログインし、「アクセス」 > 「アクセス管理」 > 「ユーザー・ブラウザー」をクリックして、「ユーザー・ブラウザー」を開きます。
2. 「ユーザー・ブラウザー」パネルで、「ユーザーの追加」をクリックします。

3. 「ユーザー・フォーム」に入力し、「無効」チェック・ボックスをクリアして作成時にユーザーを有効にするようにしてから、「ユーザーの追加」をクリックします。

ユーザーを作成した初期状態では、CLI にログインする特権がなく、GuardAPI コマンドはいずれも実行できません。例えば、新しく作成したユーザーの CLI アカウント (guardcli1 から guardcli5) を使用すると、 すぐに切断されて、必要なロールが定義されていないことが示されます。

```
$ ssh -l guardcli1 192.168.1.89  guardcli1@192.168.1.89's password:
Last login: Tue Aug 10 18:37:25 2010 from 192.168.1.14
Welcome guardcli1 - your last login was Tue Aug 10 18:37:26 2010
Please enter your GUI login (one with ADMIN or CLI role defined):johnsmith
No such user or user does not have the necessary role defined.
Connection to 192.168.1.89 closed.
```

4. 「ユーザー・ブラウザー」パネルで、任意のユーザーの「ロール」をクリックして、「ユーザー・ロール・フォーム」パネルを表示します。
5. 「CLI」チェック・ボックスにチェックマークを付けて、「保存」をクリックすると、ユーザーに CLI アクセス権が付与されます。



これで、新しく作成したユーザーの CLI アカウント (guardcli1 から guardcli5) の使用を試みると、パスワードが要求されて、CLI へのアクセスが許可されます。

```
$ ssh -l guardcli1 192.168.1.89
guardcli1@192.168.1.89's password:
Last login: Tue Aug 10 18:39:01 2012 from 192.168.1.14
Welcome guardcli1 - your last login was Tue Aug 10 18:39:02 2011
The 'set guiuser' command must be run (successfully) before any other commands will work
set guiuser admin
Enter current password
192.168.1.89>
```

6. 必要に応じて追加のロールを付与し、GuardAPI 関数を実行するためのアクセス権限をユーザーに許可します。

   例えば、ユーザー johnsmith で次の GuardAPI コマンドを実行すると、実行できる API コマンドがないことが分かります。

   ```
   192.168.1.89 >grdapi commands user
   ID=0
   Matching API Function list:
   ok
   ```

   しかし、johnsmith に accessmgr ロールを付与 (上記のステップ 5) した場合は、同じ GuardAPI コマンドを使用すると、次の API コマンドが使用可能であることが分かります。

   ```
   192.168.1.89> grdapi commands user
   ID=0  Matching API Function list :
   create_db_user_mapping
   create_user_hierarchy
   delete_allowed_db_by_user
   delete_db_user_mapping
   delete_user_hierarchy_by_entry_id
   delete_user_hierarchy_by_user
   execute_ldap_user_import
   list_allowed_db_by_user
   list_db_user_mapping
   list_user_hierarchy_by parent_user
   update_user_db
   ok
   ```

親トピック: アクセス管理の概要

LDAP からのユーザーのインポート

Guardium® ユーザー定義を LDAP サーバーからインポートすることができます。これには、該当するユーザーを取得するインポート操作の構成をします。

インポート操作は、オンデマンドで実行するか、または定期的に実行するようにスケジュールすることが可能です。 新規ユーザーのみをインポートするか、または既存のユーザー定義を置換するかを選択することができます。どちらの場合も、LDAP グループは Guardium のロールとしてインポートできます。

LDAP ユーザーをインポートする場合:

- Guardium の admin ユーザー定義は、いかなる形でも変更されません。
- 既存のユーザーは削除されません (つまり、ユーザー・セット全体が LDAP からインポートされるセットで置き換えられるわけではありません)。
- Guardium パスワードは変更されません。
- Guardium に追加される新規ユーザー:
  - デフォルトで非アクティブのマークが付けられる

- ブランク・パスワードを持つ
- ユーザー・ロールを割り当てられる

注:

ユーザー名の特殊文字はサポートされません。

アクセス管理を介して (「ユーザーの追加」または「LDAP ユーザーのインポート」から) 手動でユーザーを追加するときに、姓名のどちらかまたは両方がない場合は、ログイン名が使用されます。

この LDAP 構成メニュー画面では、いくつかのメニュー項目でヒントが表示されます。カーソルをメニュー項目 (「ユーザーのオブジェクト・クラス」など) に移動すると、 短い説明が表示されます。

CLI ユーザーの特権は分離されないため、Guardium CLI ユーザーは LDAP 環境で認証できません。

LDAP ユーザー・インポートの構成

ユーザーの識別に使用する属性は、Guardium 管理者により、「LDAP 認証の構成」パネルの「ユーザー RDN タイプ」ボックスで定義されています。詳しくは、『LDAP 認証の構成』を参照してください。 デフォルトは uid ですが、Guardium 管理者と相談して使用する値を決めてください。RDN 値として SamAccountName を使用する場合、フルネームで a=search または =[domain name] のいずれかを使用する必要があります。例: SamAccountName=search、SamAccountName=dom

注: LDAP ユーザー・インポートを構成する accessmgr ユーザーには、グループ・ビルダーの実行特権が必要です。 特定の状態において、ロール特権に変更が加えられた場合、accessmgr のグループ・ビルダーに対する特権が取り消される可能性があります。 その結果、LDAP ユーザー・インポートを正常に保存することも実行することもできなくなります。 アクセス管理ポータルに移動して、選択項目から「ロール権限」を選択してください。グループ・ビルダー・アプリケーションを選択し、「すべてのロール」ボックスまたは「accessmgr」ボックスにチェック・マークが付けられていることを確認します。

1. 「アクセス」 > 「アクセス管理」 > 「LDAP ユーザーのインポート」をクリックして、「LDAP ユーザーのインポート」パネルを開きます。

   必須情報の入力については、このヘルプ・トピックの最後にある『Tivoli® LDAP 構成の例』を参照してください。

2. 「LDAP ホスト名」に、アクセス先の LDAP サーバーの IP アドレスまたはホスト名を入力します。
3. 「ポート」に、LDAP サーバーへの接続に使用するポート番号を入力します。
4. 「サーバー・タイプ」メニューから、LDAP サーバー・タイプを選択します。
5. Guardium から LDAP サーバーに SSL (Secure Sockets Layer) 接続を使用して接続する場合は、「SSL 接続を使用」チェック・ボックスにチェック・マークを付けます。
6. 「基本 DN」に、検索を開始する、ツリー内のノードを指定します。例えば、企業ツリーは DC=encore,DC=corp,DC=root のように開始されることがあります。
7. 「インポートする属性」に、ユーザーのインポートに使用する属性 (例えば、cn) を入力します。各属性は名前を持ち、objectClass に属します。
8. インポートする前にすべての既存のグループ・メンバーを削除する場合は、「インポートする前に既存のグループ・メンバーをクリアする」チェック・ボックスにチェック・マークを付けます。
9. 「ログイン・ユーザー」および「パスワード」に、LDAP サーバーに接続するユーザー・アカウントの情報を入力します。
10. 「検索フィルターの有効範囲」に、基本レベルにのみ検索を適用する場合は「1 レベル」を、基本レベルの下のレベルに検索を適用する場合は「サブツリー」を選択します。
11. 「制限」に、返される項目の最大数を入力します。過剰な数のメンバーを意図せずロードしてしまうことを防ぐため、このフィールドを 使用して、新規照会や、既存の照会への変更をテストすることをお勧めします。
12. **オプション:** 「検索フィルター」に、基本 DN、有効範囲、および検索フィルターを定義します。通常、インポートは LDAP グループのメンバーシップに基づいているため、memberOF キーワードを使用します。 例えば、「memberOf=CN=syyTestGroup,DC=encore,DC=corp,DC=root」を使用します。
13. 「適用」をクリックして、構成設定を保存します。
    注: 「構成 - 一般」セクションの「状況」標識が「*このグループの LDAP インポートは現在、次のように設定されています*」に変わり、「スケジュールの変更」ボタンと「今すぐ 1 回実行」ボタンが有効になります。これで、LDAP サーバーからインポートすることができます。

LDAP ユーザー・インポートのスケジュール

LDAP インポートがまだ構成されていない場合は、この手順を実行する前に、LDAP ユーザー・インポートの構成を実行する必要があります。

1. 「アクセス」 > 「アクセス管理」 > 「LDAP ユーザーのインポート」をクリックして、「LDAP ユーザーのインポート」パネルを開きます。

LDAP ユーザー・インポートの実行

LDAP ユーザー・インポートをオンデマンドで実行する際には、照会によって返される各ユーザーを受け入れるまたは拒否する機会が与えられます。 これは特に、テストを目的とする場合に便利です。LDAP インポートがまだ構成されていない場合は、この手順を実行する前に、LDAP ユーザー・インポートの構成を実行する必要があります。

1. 「アクセス」 > 「アクセス管理」 > 「LDAP ユーザーのインポート」をクリックして、「LDAP ユーザーのインポート」パネルを開きます。
2. 「今すぐ 1 回実行」をクリックします。タスクの完了後、ユーザーの選択基準を満たすメンバー・セットが「LDAP 照会結果」パネルに表示されます。
3. 「LDAP 照会結果」パネルで、追加する各ユーザーのチェック・ボックスにマークを付けて、「インポート」をクリックします (または「キャンセル」をクリックして、ユーザーをインポートせずに戻ります)。
4. 追加されたユーザーを表示するには、「アクセス」 > 「アクセス管理」 > 「ユーザー・ブラウザー」をクリックして、「ユーザー・ブラウザー」を開きます。 正しいユーザー・アカウントが追加されていることを確認します。

Tivoli LDAP 構成の例

表 1. Tivoli LDAP 構成の例

LDAP ホスト名	**値**
ポート	389
サーバー・タイプ	Tivoli Directory
SSL 接続を使用	

LDAP ホスト名	値
基本 DN	cn=sample realm,o=sample
インポート・モード	「既存の属性をオーバーライド」を選択
インポート・リストにないユーザーは無効にする	
新しくインポートされたユーザーを有効にする	
ログイン・ユーザー	cn=root
パスワード	
検索フィルターの有効範囲	サブツリー
制限	
ユーザー・ログインとしてインポートする属性	cn (ポータルを介して構成可能)
検索フィルター	
ユーザーのオブジェクト・クラス	デフォルト値で埋める - |(objectClass=organizationalPerson)(objectClass=inetOrgPerson)(objectClass=person)
ロールのインポート	チェック・マークを追加
ロールとしてインポートする属性	cn
ロール検索の基本 DB	デフォルト値で埋める - cn=sample realm,0=sample
ロール・フィルター	
ロールのオブジェクト・クラス	デフォルト値で埋める - |(objectClass=groupOfNames)(objectClass=group)(objectClass=groupOfUniqueNames)
ロールを関連付けるユーザーの属性	デフォルト値で埋める - memberOf
ユーザーを関連付けるロールの属性	デフォルト値で埋める - member

親トピック: アクセス管理の概要

「データ・セキュリティー」- ユーザー階層およびデータベースの関連付け

データ・セキュリティー機能を使用して、ユーザーの階層を作成し、ユーザーを特定のデータベースおよびサーバーに関連付けることができます。Guardium® のデータ・セキュリティー機能は、どのユーザーがどの情報にアクセスしたかを報告し、確実に特定のユーザーのみが自分の担当している情報を表示できるようにします。

Guardium のデータ・セキュリティー機能を有効にして使用するには、以下の手順を実行します。

1. データ・セキュリティーの有効化
2. ユーザー階層の作成
3. ユーザーとデータベースの関連付けの作成
4. 結果のフィルタリング

データ・セキュリティー機能を分類機能 (データベースの複数の場所にある機密データをディスカバーおよび分類する機能) と併用すると、データ・レベル・セキュリティーは、指定されたユーザーが指定されたデータ・ソース (データ・ソース定義) からの分類結果を表示できないようにします。また、タスク・タイプが「分類」の場合、データ・レベル・セキュリティーを使用すると、指定されたユーザーが監査タスク結果を表示できないようにすることも可能です。

データ・セキュリティーの有効化

制約事項: データ・レベル・セキュリティーと調査ダッシュボードは同時に有効化できません。

1. admin ユーザーとしてログインし、「設定」 > 「グローバル・プロファイル」をクリックして、「グローバル・プロファイル」を開きます。
2. データ・レベル・セキュリティーのフィルタリングの「有効化」をクリックします。

   注: データ・レベル・セキュリティーのフィルタリングの状況標識アイコンが  として表示されます。

   「サービス状況 (Services Status)」パネル (「設定」 > 「サービス状況 (Services Status)」) を参照することによって、データ・レベル・セキュリティーのフィルタリングが有効になっていることを確認できます。

- データ・レベル・セキュリティーのフィルタリングが有効になっている状態で、accessmgr としてログインし、「ユーザー階層」機能と「ユーザー - データベース関連付け」機能を使用します。

ユーザー階層の作成

「ユーザー階層」には、すべてのユーザー間の親子関係が表示されます。ユーザー階層は、関係の親には特定のサーバーおよびデータベースの表示を許可しますが、子には許可しません。

accessmgr としてログインし、「データ・セキュリティー」 > 「ユーザー階層」をクリックして、「ユーザー階層」を開きます。

以下のいずれかを実行します。

- 「アクティブなユーザー - DB マップの全更新」をクリックして、ユーザーの階層全体を表示します。
- 「ロール」フィルターと「ユーザー」フィルターを使用して、特定のユーザーまたはロールの階層を表示します。ツリーを展開または縮小したり、特定の階層にユーザーを追加したりするには、階層内のノードを右クリックします。
- 「キャッシュ階層のリフレッシュ」をクリックして、階層を更新します。

注: 構成によっては、子のデータ・レベル・セキュリティーを親が継承するという、継承を行うこともできます。

ユーザーとデータベースの関連付けの作成

「ユーザー - データベース関連付け」機能は、ユーザーを特定のデータベースにマップして、ユーザーが、表示を許可されているデータしか表示できないようにします。

accessmgr としてログインし、「データ・セキュリティー」 > 「ユーザー - データベース関連付け」をクリックして、「ユーザー - データベース関連付け」を開きます。

以下のいずれかを実行します。

1. 「アクティブなユーザー - DB マップの全更新」をクリックして、データベースへのユーザーの現在のマッピングを表示します。
2. 「サーバーおよびサービス名の推奨」リストからオプションを選択して、「移動」をクリックし、新規の「ユーザー - DB 関連付け」マップを作成します。
   注: マップが完全に更新された後、すべてのサーバーをリストしたツリーが表示されます。ツリー内の任意のノードをクリックすると、そのノードに現在関連付けられているユーザーが表示されます。

   デュアル・スタック構成を使用している場合は、ルート・ノード、およびアドレスの 2 つのツリー (選択可) が表示されます。1 つのツリーは IPv4 アドレスを示し、長い方のツリーは IPv6 アドレスを示します。

   ユーザーまたはグループをノードに追加するには、ノードを選択して、「ユーザーの追加」または「グループの追加」をクリックします。

一元管理

一元管理アプライアンスでは、「ユーザー - データベース関連付け」画面に、管理対象ノードのデータに基づいてデータベースの関連付けを作成できるようにするためのボックスも表示されます。リモート・ソースの選択は、一元管理アプライアンスに表示されるボックスからのみ行います。また、すべての管理対象ノードからデータを取得するチェック・ボックスも表示されます。

結果のフィルタリング

監視データ・レベルにおけるデータ・レベル・セキュリティーには、特定のユーザーおよびそのユーザーが担当する特定のデータベースに対するデータのフィルタリングが必要です。

システム・レベルでのフィルタリングは「ユーザー階層」および「ユーザー - データベース関連付け」に基づいて行われます。そのため、Guardium システム内のさまざまなレポート、監査プロセス、セキュリティー・アセスメントなどでは、ユーザーに割り当てられているデータベースの情報のみがユーザーに表示されます。

admin ユーザーとしてログインし、「グローバル・プロファイル」を使用して、結果をフィルタリングします。「設定」 > 「グローバル・プロファイル」をクリックして、「グローバル・プロファイル」を開きます。

- デフォルトのフィルタリング
  - すべて表示 - このオプションを使用できるのは、ログインしているユーザーに特殊なロール *datasec-exempt* が定義されている場合だけです。このロールにより、データ・レベル・セキュリティーがない場合と同様に、すべてのデータを表示できるようになります。
  - 間接レコードを含める - このチェック・ボックスは、ログインしたユーザーに属する行だけでなく、その階層におけるその他のユーザーに属する行もすべてビューアーに表示します。
- 監査プロセスのエスカレーション: このタイプのタスクに対するエスカレーションは、datasec-exempt ロールを持つユーザーに対してのみ許可されます。 datasec-exmpt ロールを持たないユーザーは、エスカレーション・リストに表示されません。

  「結果をすべてのユーザーにエスカレート」- このチェック・ボックスにチェック・マークを付けると、監視データ・レベルでのデータ・レベル・セキュリティーが有効になっている場合であっても、監査プロセスの結果 (および PDF バージョン) がすべてのユーザーにエスカレートされます。デフォルト設定では有効になっています。このチェック・ボックスが無効になっている (チェック・ボックスにチェック・マークが付けられていない) 場合、監査プロセスのエスカレーションはユーザー階層の上位レベルのユーザーおよび datasec-exempt ロールを持つユーザーに対してのみ許可されます。チェック・ボックスが無効になっており、ユーザー階層がない場合、 エスカレーションは許可されません。

- 結果の (E メールの添付書類による) 配布用の PDF および CSV の生成では、「管理コンソール」パラメーターで設定されているデフォルトのグローバル・プロファイル値が使用されます。
- ビューアーから生成された PDF および CSV では、画面上で使用されるものと同じフィルタリングが使用されます。

注:

ユーザーとデータベースの関連付けによるデータ・セキュリティーでは、Access、Exception、および Policy Violations の各ドメイン (およびこれらのドメインまたはこれらのドメインの表を使用するカスタム・ドメイン) からのレポートのみをフィルタリングします。その他のすべてのドメイン (レポート) は、ユーザーとデータベースの関連付けによるデータ・セキュリティーによりフィルタリングされません。

admin ロールを持つユーザーは、すべてのロールのイベント・タイプを表示できます (ただし情報は、監視データ・レベルのセキュリティー・パラメーターに基づいてフィルタリングされます)。

データ・レベル・セキュリティーが有効になっている場合、データ・レベル・セキュリティーのフィルタリングが正しく機能するように、カスタム・ドメインに追加された事前定義エンティティーが、その同じドメイン内にある必要があります。

データ・レベル・セキュリティーが有効な場合に、2 つの事前定義エンティティー・サブジェクトが、フィルタリング・ポリシーを使用している (カスタム・ドメイン以外の) 2 つのドメインからデータを送信しようとする場合、2 つの事前定義エンティティー・サブジェクトの送信は許可されません。データ・レベル・セキュリティーは、1 種類のフィルタリング・ポリシーしか実施できません (例えば、server_ip/service_name に応じた 1 つのポリシーおよびデータ・ソースに応じた 1 つのポリシーのみ)。

親トピック: アクセス管理の概要

ユーザー階層の定義方法

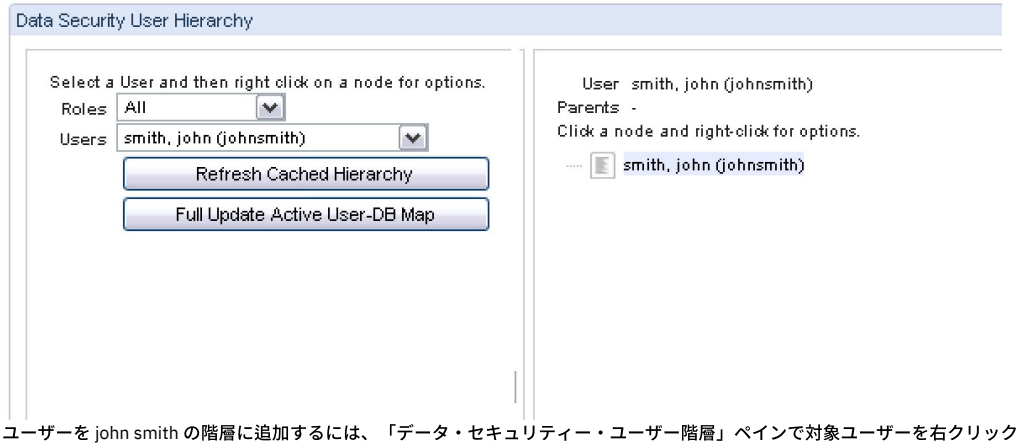
アクセス・マネージャー・アカウントから UI を使用すると、容易にユーザー階層を定義できます。

このタスクについて

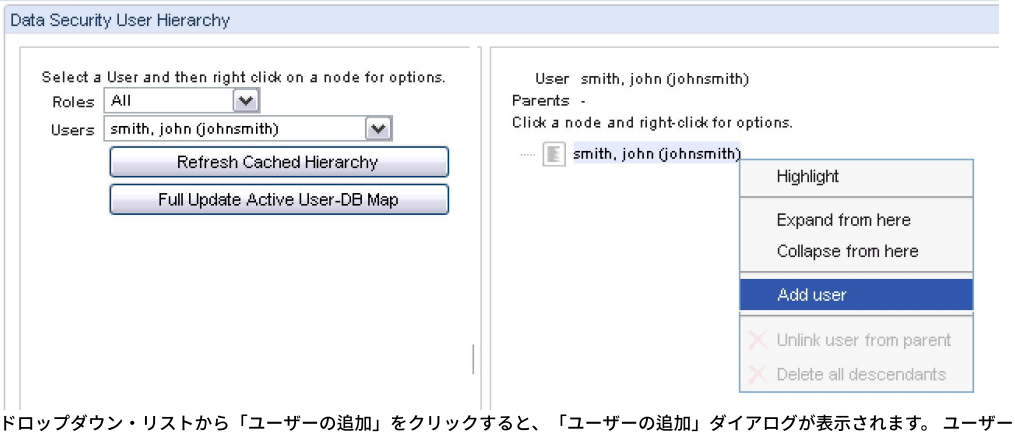
ユーザー階層によるデータ・セキュリティーは、ユーザー間の親子関係を表します。これにより、ある階層の親には特定のサーバーおよびデータベースの表示を許可し、その階層の子には許可しないことで、データ・レベルのセキュリティーを作成および適用できます。構成によっては、子のデータ・レベル・セキュリティーを親が継承するという継承を行うこともできます。

手順

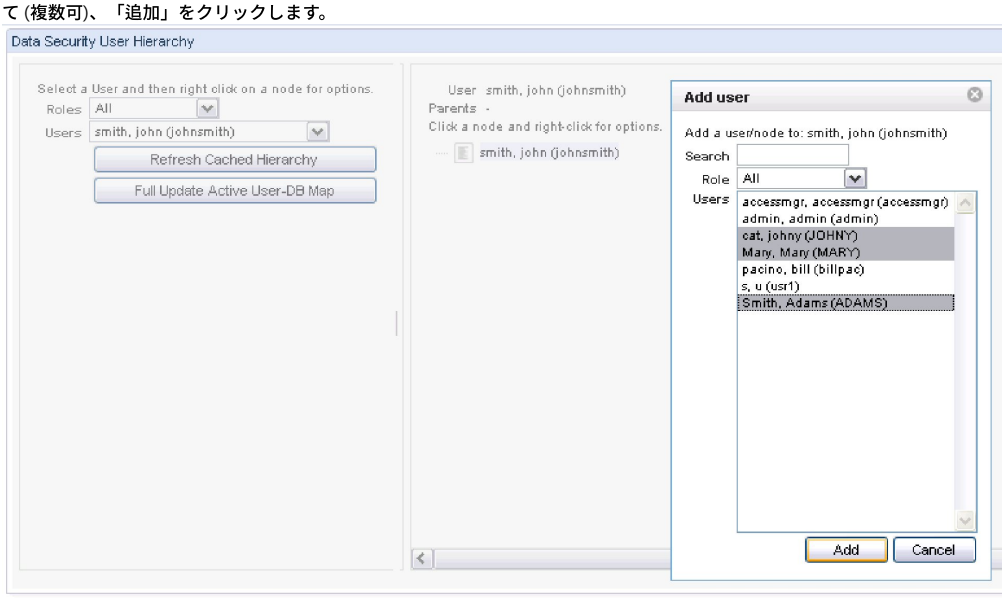
1. accessmgr としてログインし、「データ・セキュリティー」 > 「ユーザー階層」をクリックします。
2. 「ユーザー」ドロップダウン・メニューからユーザーを選択し、そのユーザーを「データ・セキュリティー・ユーザー階層」ペイン内に表示させます。この例では、john smith をユーザーとして使用します。



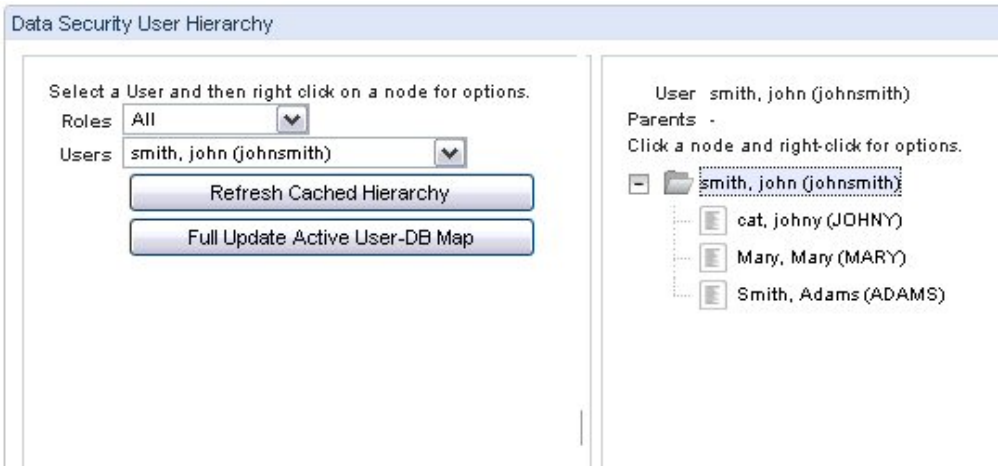
3. ユーザーを john smith の階層に追加するには、「データ・セキュリティー・ユーザー階層」ペインで対象ユーザーを右クリックし、ドロップダウン・メニューから「ユーザーの追加」を選択します。



4. ドロップダウン・リストから「ユーザーの追加」をクリックすると、「ユーザーの追加」ダイアログが表示されます。 ユーザーの階層に追加するユーザーを選択して (複数可)、「追加」をクリックします。



5. ユーザーを階層に追加すると、「データ・セキュリティー・ユーザー階層」パネルがリフレッシュされ、新しい階層をドリルダウンしたり、表示したりできます。



6. この手順を繰り返して、必要なユーザー全員をデータ・セキュリティーのユーザー階層に定義します。

親トピック: アクセス管理の概要

スマート・カードを使用した Guardium UI へのログイン

Guardium のスマート・カード・サポートは、すべてのベンダーがユーザー・アクセスで多要素認証をサポートする必要があるという米国政府の義務付けを満たしています。スマート・カード認証がサポートされるのは、Web ベースの Guardium ユーザー・インターフェース (UI) へのアクセスのみです。

始める前に

多要素認証の要件の詳細は、「Security and Privacy Controls for Federal Information Systems and Organizations」 (NIST Special Publication 800-53) 文書の『Identification and Authentication (Organizational Users) (IA-2)』セクションに記載されています。NIST 800-53 は、NIST の Web サイト (https://www.nist.gov) から入手できます。

政府系アプリケーションでは、Personal Identity Verification (PIV) カードを参照します。民間のアプリケーションでは、Common Access Card (CAC) カードを参照します。PIV カードと CAC カードでは認証局が異なりますが、それ以外の点ではこれらのカードは同じです。

Guardium のスマート・カード・サポートは、「Personal Identity Verification (PIV) of Federal Employees and Contractors」(FIPS Publication 201-2) 文書の『PIV Cardholder Authentication (6)』セクションに記載されている PIV 保証レベルで高信頼度を満たしています。FIPS 201-2 は、NIST の次の Web サイト (https://www.nist.gov) から入手できます。

前提条件

デバイスが必要とするのは次のとおりです。

- Web ブラウザー経由での、スマート・カード証明書にアクセスできる Guardium UI へのアクセス
- スマート・カード・リーダー
- 有効な PIV/CAC カード

このタスクについて

このタスクでは、スマート・カード上の情報を Guardium ユーザーと正しく関連付ける方法について説明します。

スマート・カードと関連付ける Guardium ユーザーを作成します。既存のユーザーをスマート・カードと関連付ける場合、新しいユーザーを作成する必要はありません。ユーザーの作成とアクセス管理について詳しくは、『アクセス管理の概要』を参照してください。

1. 管理ユーザーとして Guardium UI にログインします。
2. 「セットアップ」 > 「ツールとビュー」 > 「ポータル」に移動します。
3. 「認証構成」セクションの下で、「スマート・カード」オプションを選択します。「スマート・カード」オプションがない場合は、スマート・カード・パッチがインストールされていることを確認してください。
4. 「正規表現一致パターン」フィールドで、スマート・カード上のユーザー情報と一致する正規表現 (regex) を指定します。

例

ユーザーの作成

Guardium アプリケーションには、ユーザーを作成するさまざまな方法が用意されています。ユーザーの作成方法に関係なく、認証にスマート・カードを使用するように Web を構成すると、スマート・カードの資格情報のみを使用して SSL/TLS 通信を確立します (Guardium サイトは https を使用します)。

ユーザーを手動で作成する方法の例を次に示します。

1. Accessmgr として CM にログインします。
2. 「アクセス」「ユーザー・ブラウザー」を選択します。
3. 「追加」をクリックします。
4. ユーザー名「Test Cardholder X」を追加します。
5. パスワードを 2 回追加します。
6. ユーザーと同じファーストネーム (名) とラストネーム (姓) を入力します。

7. 「追加」をクリックします。

これで、マッピングを構成できるため、スマート・カードが存在するときは、スマート・カード上の情報がシステム内のユーザーに正しくマップされます。

1. CM から、またはスタンドアロンで Admin としてログインします。
2. ログイン後、「セットアップ」 > 「ツールとビュー」 > 「ポータル」に移動します。

「認証構成」というタイトルのメニュー画面が表示された場合は、スマート・カード・サポート・パッチがインストールされています。

ここで、「正規表現一致パターン」で正規表現を使用して、スマート・カード上のユーザー情報を照合します。正規表現一致パターンの例を次に示します。

CN ?= ?(.*?), ?OU ?= ?Test Agency, ?OU ?= ?Test Department, ?O ?= ?Test Government, ?C ?= ?US

これは、HTTPS を確立するために Web サーバーに送信するように選択したクライアント証明書を持つスマート・カードと連携して動作します。選択したスマート・カード上で、このクライアント証明書は Web サーバーが要求したときにその内容を Web サーバーに提供します。これは、この機能が有効なときに発生する厳密な動作です。例えば、クライアント証明書は、バージョン、シリアル番号、署名アルゴリズム、署名ハッシュ・アルゴリズム、発行者、有効日の始まり、有効日の終わり、所有者などの詳細情報を持ちます。

この例では、次のいずれかのパターンを使用できます。両方ともマッピングと照合されます。 パターン 1 はより厳密です。パターン 2 は、目的に応じて、ニーズに合う独自のパターンを記述できます。効率の良いマッピング・パターンを記述するには、スマート・カード上のデータに詳しいユーザーと協力する必要があります。

パターン 1:

CN ?= ?(.*?), ?OU ?= ?Test Agency, ?OU ?= ?Test Department, ?O ?= ?Test Government, ?C ?= ?US

パターン 2:

CN ?= ?(.*?)

例は両方とも、証明書所有者の CN 属性の値を取得します。これは、ブラウザーで証明書の詳細を調べることで確認できます。この場合は、「Test Cardholder X」です。このパターンを正しく構成することは、スマート・カードの認証を確実に成功させるために最も重要な部分であると考えられます。

他のモジュールで現在使用できる正規表現検証ツールは、この目的では使用できないことに注意してください。(『トラブルシューティングまたはリカバリー・シナリオ』セクションの項目 2 と 3 を参照)。

ここで保存します。まだ完了していないことと、CLI から有効にする必要があることに注意してください。有効化の一部は、GUI が存在しない、サーバーのシャットダウン後に実行する必要があるためです。

CLI での実行部分のために GUI から移動する前に、ルート CA 証明書をトラストストアにアップロードする必要があります。

Web サーバーのトラストストアへのルート CA 証明書のアップロード

この部分では、GUI が使用するトラストストアにルート CA の証明書をアップロードする方法について説明します。「Guardium ポータル」画面と「認証構成」画面の「証明書のインポート」選択肢を使用します。

スマート・カード上の証明書に署名した CA のルート証明書がない場合、CA が署名したユーザー証明書、またはユーザー証明書が含まれているスマート・カードからルート証明書をエクスポートできます。

顧客から授与されたか、certMgr.exe などの認証管理ツールまたはオープン SSL などのツールを使用してスマート・カードからエクスポートすることで、取得した証明書を持っていると想定します。

トラステッド CA のパブリック・ルート証明書。これが、スマート・カード・インフラストラクチャーと、スマート・カードの配布と認証に対する標準的な方法を既に持つ環境では最も一般的なルート証明書のソースです。

スマート・カード認証に使用する証明書を選択します。署名チェーンは、一連の署名認証局をリストします。選択に最適な証明書は、通常、ユーザー証明書の上の中間認証局の証明書です。

CLI からの機能の有効化 (CLI でのみ実行可)

状況を確認するために、次の CLI コマンドを使用します。

```
show system websmartcard
```

この CLI コマンドをオンにするために、次を使用します。

```
store system websmartcard on
```

この CLI コマンドをオフにするために、次を使用します。

```
store system websmartcard off
```

この機能をオフにすると、ローカル認証を使用するシステムとともに GUI が自動的に再開されます。これは、システムの初回デプロイ時に、設定した正規表現が正しくなくてエラーが表示されるときにも役立ちます。

注: 認証にはしが使用されますが、アクセス制御 (ユーザーがアクセス権を持つモジュール、ユーザーがアクセスできるナビゲーションなど) は、引き続きスマート・カード認証がない場合と同様に行われます。

この機能の有効化後

機能を有効にすると、サイトにアクセスできる方法は有効なスマート・カード (PIV、CAC など) 経由のみになります。

これで、GUI サイトにアクセスすると、証明書の選択を求める認証プロンプトが表示されるようになります。

上記の詳細は、管理者がセットアップします。正しく設定されている場合、エンド・ユーザーに必要なのはカードの挿入のみで、直接サイトのコンテンツに移動できます。

有効なスマート・カードを持つユーザーが Web サイトをロードすると、スマート・カードの PIN を求めるプロンプトがブラウザーに表示されます。この PIN により、要求されたときにカード上のクライアント証明書にアクセスできるようになります。

PIN の指定後は、ユーザー・フィールドにスマート・カードから抽出したログイン情報が事前入力された通常の Guardium ログイン・ページが表示されます。ここではパスワードが使用されないことに注意してください。ユーザー・フィールドに表示されるのは、マッピングのために抽出されたユーザーのプレースホルダーのみです。

例えば、証明書が有効で、「Test Cardholder X」に対するスマート・カード発行者のルート CA が Guardium の Web サーバーにロードされた場合 (実行方法についてはセクション『Web サーバーのトラストストアへのルート CA 証明書のアップロード』を参照)、ユーザー・フィールドには「Test Cardholder X」が事前入力され、スマート・カードの PIN を求めるプロンプトが表示されます。これは、スマート・カード上のクライアント証明書にアクセスするためです。クライアント証明書はスマート・カード上に留まり、ファイルにエクスポートすることはできません。プロンプトが 2 回表示される場合がありますが、単に PIN を指定してください。

次のタスク

トラブルシューティングまたはリカバリー・シナリオ

この機能を有効にした後、Guardium の URL をロードしたときに、エラー・ページが表示されます。

診断: 照合する正規表現の構成が正しくないか、カード上に有効な証明書がない可能性が高いと思われます。

照合する正規表現を作成したが、機能しているように見えません。Guardium には正規表現検証ツールがあったことを思い出し、ツールで機能する場合は有効な正規表現であると考えてそのツールを使用しました。 残念ながら、そのツールでテストに成功しても、スマート・カード構成では正規表現パターンが動作しません。

診断: そのツールは、テキスト段落内に正規表現が見つかるかどうかを検出するためのものです。そのため、このケースでは機能しません。この構成は、証明書の詳細に表示される所有者に表示される証明書テキストからテキストの一部を抽出するためのものです。

証明書を選択するためのプロンプトがブラウザーに表示されません。

診断: PC/ラップトップはカード・リーダーとスマート・カードをインストールできます。スマート・カードにある証明書のコピーが Windows OS の certmgr にコピーされます。ただし、サイトにアクセスするときに、ブラウザー (IE、Firefox、または Chrome) が証明書を読み取りません。つまり、この 3 つのブラウザーはすべて、証明書を読み取ることができないため、証明書を選択するためのプロンプトが表示されません。

この現象は、テストしたいくつかのラップトップ上で、すべてのブラウザーについて認められました。この場合、Guardium サイトでのみ発生するわけではありません。スマート・カードの動作を必要とする他のサイトでも、この現象が発生します。これはまれな現象です。

解決策: スマート・カードを管理する部門に連絡してください。

親トピック: アクセス管理の概要

統合および一元管理

統合を使用すると、複数の Guardium システムから取得したデータを 1 つにまとめて表示することができます。一元管理を使用すると、複数の Guardium システム間で整合性を維持することができます。

- 統合
  複数の Guardium® ユニットから情報を収集して 1 つの Guardium 統合アプライアンスにマージすることにより、データベースの使用状況に関するエンタープライズ全般のビューを表示できるようにします。
- 一元管理
  一元管理構成では、1 つの Guardium ユニットが中央マネージャーとして指定されます。 このユニットは、他の Guardium ユニット (管理対象ユニットと呼ばれる) をモニターして制御するために使用できます。 管理対象外のユニットはスタンドアロン・ユニットと呼ばれます。
- 調査センター
  調査センターは統合サーバーの拡張機能です。 調査ユーザーは (一度定義されると) 選択した履歴日付のデータおよび結果をリストアし、フォレンジック調査を実行できます。 日にち (日付) をリストアしたら、調査ユーザーは標準の Guardium UI を使用して、調査対象日付の範囲だけのレポートを定義し、表示できます。

統合

複数の Guardium® ユニットから情報を収集して 1 つの Guardium 統合アプライアンスにマージすることにより、データベースの使用状況に関するエンタープライズ全般のビューを表示できるようにします。

統合プロセス

- ソース・アプライアンスからアグリゲーターに日次ベースでデータをエクスポート (すなわち、日次エクスポート・ファイルをアグリゲーターにコピー) することによって実施されます。
- その後に、アグリゲーターはアップロードされたファイルを調べ、各ファイルを抽出して、アグリゲーター上の内部リポジトリーにマージします。

例えば、エンタープライズ・デプロイメントで Guardium を実行している場合、複数の異なる環境 (例えばさまざまな地理的位置、業務単位など) をモニターする複数の Guardium サーバーが存在することがあります。すべてのデータを中心的なロケーションに集めることができれば、エンタープライズ全体のデータベース使用状況を確認するうえで役立つでしょう。統合アプライアンスとして (初期インストール手順で) 構成された 1 つのサーバーに別の多数のサーバーからデータをエクスポートして、このことを達成できます。このようなデプロイメントでは、通常、すべてのレポート、アセスメント、監査プロセスなどを統合アプライアンスで実行することによって、必ずしもエンタープライズ全般のビューではないにしても、幅広いビューを生成できます。ただし、アグリゲーターでデータを収集するわけではありません。コレクターから取り込んだデータを表示するときに、アグリゲーターを使用します。

事前定義された統合レポートは、「Guardium モニター」タブの「エンタープライズ・バッファー使用状況モニター」、「日次モニター」タブの「ロギング・コレクター」にあります。

アプライアンス・タイプ

コレクター

これを使用して、データベース・アクティビティーを収集し、そのアクティビティーをリアルタイムで分析し、内部リポジトリーに記録して、さらに詳しく分析するか、(アラート送信やブロッキングなどの) 対応をリアルタイムで行うか、あるいはこの両方を行います。 .
このユニットは、データベース・アクティビティーをリアルタイムでキャプチャーおよび分析するために使用します。

アグリゲーター (注 1、2 を参照)
これを使用して、複数のアプライアンス (コレクターおよび他のアグリゲーター) から情報を収集してマージし、環境全体の総括的なビューを作成し、エンタープライズ・レベルのレポートを生成します。 アグリゲーターはデータそのものを収集するのではなく、複数のソースからのデータを統合するだけです。

中央マネージャー (注 1、3、4 を参照)
このアプライアンスを使用して、複数の Guardium アプライアンスを管理および制御します。
中央マネージャー (CM) を使用して、Guardium のデプロイメント全体 (すなわち、すべてのコレクターおよびアグリゲーター) を単一のコンソール (CM コンソール) から管理します。
管理内容として、パッチ・インストール、ソフトウェア更新、および照会、レポート、グループ、ユーザー、ポリシーなどの管理と構成があります。

注:

多くの環境では、中央マネージャーはアグリゲーターでもあります。 中央マネージャーとアグリゲーターは同じアプライアンスにインストールできます。

Guardium アプライアンスは、中央マネージャーにプロモートできるように、インストール時にアグリゲーターとして構成する必要があります。

フェデレーテッド環境ごとの中央マネージャーの数は 1 つです。

中央マネージャー/アグリゲーターの制約
v9.5 (v9.0 パッチ 500) 以降、アプリケーションには、中央マネージャーがアグリゲーター・タイプのアプライアンスでなければならないという制約があります。つまり、v9.5 以降では、アグリゲーター・タイプのアプライアンスのみを中央マネージャー・アプライアンスにプロモートできます。v9.5 より前の既存の CM アプライアンスは、この変更の対象ではありません。

アップグレード後にダウンと表示されるユニットについての解決策

問題: 検索モードが CM_only モードまたは Local_only モードのアグリゲーターをアップグレードすると、アップグレード後に検索でこのユニットがダウンとして表示される。また、アップグレード後にユーザーが検索モードを all_machines に変更することを選択すると、アグリゲーターから検索を使用できなくなる。

解決策: アグリゲーター・ユニットをアップグレードした後、検索のツールチップでそのアグリゲーター・ユニットがダウンと表示されないようにする場合、ユーザーは以下の 2 つのコマンドを実行できます。

1. grdapi enable_quick_search schedule_interval=2 schedule_units=MINUTE
2. restart network

注: 環境が以前 cm_only モードまたは local_only モードであった場合、または今後 cm_only モードまたは local_only モードにする場合、この手順ではアグリゲーターからの検索は有効にならず、アグリゲーターがダウンと表示されなくなるだけです。

用語

表 1.

用語	記述
Guardium アプライアンス	物理的または仮想的な Guardium ボックス。「コレクター」または「アグリゲーター」のいずれかです (一元管理を実施する場合としない場合がある)。
Guardium ユニット	Guardium アプライアンスを参照
マネージャー・ユニット	中央マネージャーとして構成されたアプライアンス
管理対象ユニット	中央マネージャーによって管理されるアプライアンス
スタンドアロン・ユニット	中央マネージャー環境に含まれないアプライアンス
パージ	最適なパフォーマンスを得るために、不要なデータをすべてパージします。パージによって、ディスク・スペースを解放します。
アーカイブ	1 日のデータを圧縮して暗号化されたファイルに入れ、アグリゲーターに送信します。

階層的な統合

Guardium では階層的な統合もサポートされます。これは、複数の統合アプライアンスを上位レベルの中心的な統合アプライアンスにマージすることです。複数レベルのビューを提供するには、これが役立ちます。例えば、複数のユニットを統合する北アメリカ地域用の統合アプライアンスを 1 つ、複数のユニットを統合するアジア地域用の別の統合アプライアンスを 1 つそれぞれ配置し、北米とアジアの統合アプライアンスの内容をマージして単一の企業ビューを提供する中心的なグローバル統合アプライアンスを配置できます。データを集計するために、統合されているすべての Guardium サーバーはスケジュールに基づいてデータを統合アプライアンスにエクスポートします。統合アプライアンスはそのデータを統合アプライアンス上の単一のデータベースの中にインポートします。これにより、統合アプライアンスに対して実行されるレポートは、統合されるすべての Guardium サーバーから集計されたデータに基づくようになります。

システム共有パスワードについて

Guardium 管理者は「システム構成」パネルでシステム共有パスワードを定義します。以下のセクションでは、これについて説明します。システム共有パスワードはアーカイブ/リストア操作、および一元管理/統合操作で使用されます。これを使用する場合、互いに通信するすべてのユニットの間でその値が同じでなければなりません。インストール時にはこの値は NULL で、時間の経過とともに変化する可能性があります。

システム共有パスワードは次のような場合に使用されます。

- 中央マネージャーと管理対象ユニットの間でセキュア接続が確立されるとき。
- 統合される装置が、アグリゲーターにエクスポートされるデータに署名して暗号化するとき。

- いずれかの装置が、アーカイブ用のデータに署名して暗号化するとき。
- アグリゲーターが、統合される装置からデータをインポートするとき。
- いずれかのユニットがアーカイブ・データをリストアするとき。

企業のセキュリティー・プラクティスに応じて、システム共有パスワードを時々変更する必要が生じることがあります。共有パスワードは変更される可能性があるため、各システムは、そのシステムで定義されているすべての共有パスワードの履歴レコードが入っている共有パスワード鍵ファイルを保守します。これにより、古い共有パスワードを使ってシステムからエクスポート (またはアーカイブ) されたファイルを、新しく置換された同じ共有パスワードを持つシステムでインポート (またはリストア) することができます。CLI を使って (現在および過去の) 共有パスワードを 1 つのアプライアンスからエクスポートして、別のアプライアンスにインポートできます。

統合が機能するためには、アグリゲーターおよび統合されるすべてのコレクターで共有パスワードを設定する (それらの間で同じに設定する) 必要があります。

統合、アーカイブ、およびパージ操作

スケジュールされたエクスポート操作により、Guardium コレクター・ユニットから Guardium 統合アプライアンスにデータが送信されます。統合アプライアンスは、独自のスケジュールでインポート操作を実行し、統合処理を完了します。 この一方または両方のユニットにおいて、アーカイブ操作とパージ操作がスケジュールされ、それにより (スペースを解放し、内部データベースのアクセス操作を高速化する目的で) データは定期的にバックアップおよびパージされます。エクスポート、アーカイブ、パージの各機能では、同じデータを処理できますが、同じ日付範囲を処理することはできません。例えば 1 日より古いすべての情報をエクスポートおよびアーカイブし、1 カ月より古いすべての情報をパージすることで、常に 1 カ月分のデータを送信側装置上に残しておくことができます。

注:

アグリゲーターでインポートのスケジュールを設定するときには、すべてのコレクターでエクスポートが完了してからインポートを実行するように計画してください。

CAS データも統合とアーカイブの対象になります。

注: アグリゲーター・サーバーでは No Traffic アラートが非アクティブです。

アグリゲーターでのデータの管理

- データのエクスポート
  - エクスポートの停止
- データのインポート
  - インポートの停止
- アーカイブとパージ
- アーカイブとパージの停止
- アーカイブおよびパージ処理の検証
- 統合およびアーカイブ・アクティビティーに関するレポート
- リストア

データのエクスポート

表 2. データのエクスポート

項目	記述
機能	1 日 (午前零時から次の日の午前零時まで。通常は「昨日」)のデータを圧縮して暗号化されたファイルに入れ、アグリゲーター (またはアーカイブ上の外部リポジトリー) に送信します。
スケジュール	日次ベースで実行されます。 1 日のデータをすべて含めるように、午前零時直後 (00:10)に開始されます。 完了までに要する時間は最大 2 時間 (平均値 - データ量によって異なる) と想定されます。
プロセスの概要	一時データベースを作成します。 関連するデータ (昨日のアクティビティー) を一時データベースにロードします。 一時データベースの自動増分 ID を更新して、その固有性を確保します。 一時データベースの圧縮された暗号化エクスポート・ファイルを作成します。 そのエクスポート・ファイルをアグリゲーター (またはアーカイブ上の外部リポジトリー) にコピーします。

統合アプライアンスにデータをエクスポートするには、手順に従います。各 Guardium ユニットに対して 1 つのエクスポート構成を定義できます。

1. 「管理」 > 「データ管理」 > 「データ・エクスポート」をクリックして、「データ・エクスポート」を開きます。
2. 「エクスポート」ボックスを選択します。これにより、データ・エクスポートの追加オプションが開きます。
3. 「次の期間を経過したデータをエクスポート」に続くボックスで、エクスポート操作の対象となる開始日を、当日 (0 日) よりさかのぼった日数、週数、または月数として指定します。 これらはカレンダーによる計測であり、今日が 4 月 24 日である場合、4 月 23 日にキャプチャーした全データは、操作が実行された時刻に関係なく 1 日古いデータということになります。 昨日のデータからデータ・アーカイブを開始するには、値 1 を入力します。
4. オプションで、「次の期間を経過したデータを無視」に続くボックスを使用して、何日分のデータをアーカイブするかを制御します。 ここで指定する値は「次の期間を経過したデータをエクスポート」の値より大きくする必要があります。つまり、常に少なくとも 2 日分のデータがエクスポートされます。 「次の期間を経過したデータを無視」をブランクのままにした場合、「次の期間を経過したデータをエクスポート」行で指定された値よりも古いすべての日のデータがエクスポートされます。「次の期間を経過したデータを無視」の値を常に設定することをお勧めします。そうしないと、まったく同じ日のデータを何度もエクスポートすることになり、(無視される) 重複データのためにネットワークやアグリゲーターに過負荷が発生します。
5. デフォルトでは「値のエクスポート」ボックスが選択されます。データ・エクスポートが禁止されている国にコレクターが配置され、別の国に統合アプライアンスが配置されているような場合、「値のエクスポート」チェック・ボックスのチェック・マークをクリアすることができます。これにより、データベース値を含んでいるすべてのフィールドがマスクされます。
6. 「ホスト」ボックスに、このシステムの暗号化データ・ファイルの送信先となる統合アプライアンスの IP アドレスまたは DNS ホスト名を入力します。複数のアグリゲーターのエクスポート・データのために 2 次統合を使用可能にするオプションもあります。「ホスト」ボックスは 2 つ使用可能であり、 1 つ目のホストは必須

ですが、2 次ホストはオプションです。 このユニットと、データ送信先の統合アプライアンスとの間では、システム共有パスワードが同じでなければなりません。そうでない場合、エクスポート操作は可能ですが、データを受け取る側の統合アプライアンスはエクスポート・ファイルを暗号化解除できず、インポートが失敗します。詳しくは、システム構成のシステム共有パスワードの説明を参照してください。共有パスワードは、エクスポート・システムと受信システムとで同じでなければなりません。 この理由は、同じ共有パスワードを持っていないと、 エクスポート・システムの構成が設定されず、 受信システムに送信できないというメッセージがテスト・ファイルに関して出力されるからです。

7. 「スケジューリング」セクションを使用して、この操作を定期的に実行するためのスケジュールを定義できます。
8. この装置のエクスポート/パージ構成を保存するには「保存」ボタンをクリックします。「適用」ボタンをクリックすると、システムは、指定されたアグリゲーター・ホストがこのユニットからのデータを受け入れるかどうか確認しようとします。この操作が失敗した場合、「テスト・データ・ファイルをこのホストに送信できませんでした」というメッセージが表示されて、構成は保存されません。ホスト名または IP アドレスが正しく入力されていること、およびホストがオンラインであることを確認してください。
9. 「今すぐ 1 回実行」をクリックして、操作を 1 回実行します。

エクスポートの停止

統合アプライアンスへのデータ・エクスポートを停止するには、次のようにします。

1. 「管理」 > 「データ管理」 > 「データ・エクスポート」をクリックして、「データ・エクスポート」を開きます。
2. 「エクスポート」チェック・ボックスをクリアします。
3. 「保存」をクリックします。

注:「今すぐ 1 回実行」ボタンをクリックした後でエクスポートを停止することはできません。

データのインポート

Guardium コレクター・ユニットは、統合アプライアンスとして構成された別の Guardium アプライアンスに、暗号化されたデータ・ファイルをエクスポートします。統合アプライアンスがインポート操作を実行し、すべてのデータを暗号化解除して独自の内部データベースにマージするまで、暗号化データ・ファイルは統合アプライアンス上の特別なロケーションに保管されます。

注: まだ完全に着信し終わっていないファイルをインポートしてしまうのを防ぐために、統合アプライアンスは最近 2 分間に変更されたファイルをインポートしません。

表 3. データのインポート

項目	記述
機能	データをインポートし、インポートしたデータをアグリゲーターの内部データベースにマージします。
スケジュール	日次ベースで実行されます。1 日に複数回実行しないでください。 02:00 (またはエクスポートの終了後) に開始されます。 完了に要する時間は最大 3 時間と想定されます。
プロセスの概要 (パージする各日ごと)	パージする各表ごとに delete コマンドを構成します (表とパージ条件は AGG_TABLES で定義されます)。 各表ごとに delete コマンドを実行します。

統合アプライアンス上でデータ・インポート操作を定義するには、以下の手順に従います。各ユニットに対して、データ・インポート構成を 1 つだけ定義できます。

1. 「管理」 > 「データ管理」 > 「インポート」をクリックして、「インポート」を開きます。
2. 「インポート」チェック・ボックスを選択します。すると、インポートされるデータ・ファイルのロケーションを示す変更不能な追加のフィールドが表示されます。
3. 「適用」をクリックして、構成を保存します。「適用」ボタンは、「データのインポート元」チェック・ボックスのオン/オフを切り替えたときにのみ、使用可能になります。
4. 「今すぐ 1 回実行」をクリックして、操作を 1 回実行します。
5. 「スケジュールの変更」をクリックすると、汎用のタスク・スケジューラーが開いて、定期的に実行するよう操作をスケジュールできます。この統合アプライアンスと、これにデータをエクスポートするすべてのユニットの間で、システム共有パスワードが同じでなければなりません。そうでない場合、エクスポート操作は正常に行われますが、統合アプライアンスはエクスポートされたデータ・ファイルを暗号化解除できません。

インポートの停止

他の Guardium ユニットから送られるデータのインポートを停止するには、次のようにします。

1. 「管理」 > 「データ管理」 > 「インポート」をクリックして、「インポート」を開きます。
2. 「データのインポート (Import data)」ボックスをクリアします。
3. 「適用」をクリックして、構成を保存します。インポートを停止しても、他の Guardium ユニットからこのシステムへのデータ・エクスポートは停止しません。それを停止するには、送信側の各ユニットにおいてエクスポート操作を停止する必要があります。

注:「今すぐ 1 回実行」ボタンをクリックした後でインポートを停止することはできません。

アーカイブとパージ

Guardium システムを正常に稼働させるには、定期的にデータをアーカイブおよびパージすることが重要です。最適なパフォーマンスを得るために、不要なデータをすべてアーカイブしてパージすることを強くお勧めします。パージを実行して、ディスク・スペースを解放することは重要です。例えば 3 カ月分のデータだけを Guardium アプライアンス上に残しておく必要がある場合、90 日より古いすべてのデータをアーカイブしてパージします。

アーカイブ/パージ処理により、スペースが解放され、将来の使用に備えて情報が保存されます。スタンドアロン・ユニットおよび統合ユニットから定期的にデータをアーカイブ/パージするのが適切です。Guardium のアーカイブ機能は、不正開封できない、署名付きの暗号化ファイルを作成します。アーカイブ・ファイルは、ファイル・サーバーやストレージ・システムなどの外部システムに転送され格納されます。

注:

アーカイブとパージが両方ともスケジュールされている場合、アーカイブの後にパージが実行されます。

コレクターでアーカイブしたデータは、別のコレクターまたはアグリゲーター・サーバーでリストアできます。アグリゲーターでアーカイブしたデータをコレクター・マシンでリストアすることはサポートされていません。

アグリゲーター・システム上のデータのアーカイブ - 月の最初の日にすべての静的表がアーカイブされます。それ以外の日は、 アーカイブ・データに追加されたデータのみがアーカイブされます。 この方法は、コレクターで使用される方法と同じです。静的表を 通常のパージ・プロセスに追加すると、 オーファンの存在が排除され、ディスク・スペースが解放されるため、レポートのパフォーマンスが改善されます。

アグリゲーター上の静的表のアーカイブとエクスポートにおいて、 すべての静的データが対象に含められるのは、アーカイブの場合は月の最初の日に限られます。 エクスポートの場合は、エクスポート構成が変更されたときです。CLI コマンド store archive_table_by_date [enable | disable] または show archive_table_by_date を使用してください。 他に関連する CLI コマンドとしては、store aggregator clean orphans や show aggregator clean orphans があります。

データ管理タスクのスケジューリング - ユニットの作成時には、デフォルトのスケジュール時刻が提供されます。その時刻を状況に応じて修正することができます。データ管理タスクは、夜間などのビジーでないときにスケジュールする必要があります。データ管理タスクはタスク同士が重なることがないように (例えば、1 つのタスクが開始して終了する前に、別のタスクを開始することはできません)、一定の間隔を置いてスケジュールする必要があります。

データ・インポートとデータ・アーカイブを実行するアグリゲーター/中央マネージャーを扱うときのアグリゲーターのデータ・アーカイブ。デフォルトまたは一般的な設定は、データ・アーカイブで、1 日を経過したデータのアーカイブを実行し、2 日を経過したデータを無視することです。他のコレクター/アグリゲーターからのデータ・インポートの前にデータ・アーカイブの実行がスケジュールされていると、アーカイブにその日のアーカイブ用のインポートが含まれません。次のスケジュールがあるとします: 午前 0 時 30 分にデータ・アーカイブを実行し、午前 6 時に 1 日を経過したデータを対象にデータ・インポートを実行する (このとき、2 日を経過したデータは対象としない)。 アーカイブの発生時に、「昨日」に相当するデータはアーカイブされません。その日のデータ用のインポートがまだ行われていないためです。この例では、データ・アーカイブを、データ・インポートが終了した後に発生するように、スケジュール変更する必要があります。このようにすると、アーカイブに正しく昨日のデータが含まれるようになります。

表 4. データのアーカイブとパージ

項目	記述
パージ機能	アプライアンスから古いレコード (通常、60 日を超えたもの) を削除して、スペースを解放し、内部データベースへのアクセス操作を迅速化します。 パージは日付に基づいて行われます (すなわち、終日分のデータが削除されます) が、引き続き「使用中」(オープン・セッションなど) のレコードは削除されません。
スケジュール	デフォルト・パージ・アクティビティーは、毎日午前 5:00 にスケジュールされます。 コレクター (エクスポート/アーカイブ後) アグリゲーター (インポート後) 完了に要する時間は最大 2 時間と想定されます。
プロセスの概要 (パージする各日ごと)	パージ構成は、データ・アーカイブとデータ・エクスポートの両方で使用されます。 「次の期間を経過したデータをパージ」フィールドを使用して、パージ操作の開始日を、当日 (0 日) よりさかのぼった日数、週数、または月数として指定します。
デフォルト・パージ	パージのデフォルト値は 60 日です。 デフォルト・パージ・アクティビティーは、毎日午前 5:00 にスケジュールされます。 新規インストールでは、デフォルトの値とアクティビティーに基づくデフォルト・パージ・スケジュールがインストールされます。 ユニット・タイプをマネージャー管理に変更したり、スタンドアロンに戻したりするときに、デフォルト・パージ・スケジュールが適用されます。パージ・スケジュールは、アップグレード中は影響を受けません。

特定の時点で、このデータに関するレポートまたは調査を実行する必要が生じることがあります。例えば一部の規制環境では、24 時間以内に照会できる形式でこの情報を 3 年、5 年、または 7 年にわたって保持する必要があります。この作業は、アーカイブ・データをユニットに復元する Guardium リストア機能によってサポートされます。

以下のセクションでは、アーカイブを定義してスケジュールする方法、およびアーカイブからリストアする方法について説明します。

注: アーカイブ操作とリストア操作は、アーカイブ処理時に生成されるファイル名に依存します。アーカイブ・ファイルの名前を決して変更しないでください。

アーカイブ・データ・ファイルをネットワーク上の SCP または FTP ホストに送信したり、EMC Centera または TSM ストレージ・システム (構成されている場合) に送信したりすることができます。各ユニットに対して 1 つのアーカイブ構成を定義できます。ネットワーク上の別のホストにデータをアーカイブして、(オプションで) ユニットのデータをパージするには、手順に従います。

1. 「管理」 > 「データ管理」 > 「データ・アーカイブ」をクリックして、「データ・アーカイブ」を開きます。
2. 「アーカイブ」チェック・ボックスを選択すると、アーカイブ処理に関する追加のフィールドが表示されます。
3. 「次の期間を経過したデータをアーカイブ」に続くボックスに、アーカイブ操作の開始日を、当日 (0 日) よりさかのぼった日数、週数、または月数として指定します。 これらはカレンダーによる計測であり、今日が 4 月 24 日である場合、4 月 23 日にキャプチャーした全データは、操作が実行された時刻に関係なく 1 日古いデータということになります。 昨日のデータからデータ・アーカイブを開始するには、値 1 を入力します。
4. オプションで、「次の期間を経過したデータを無視」に続くボックスを使用して、何日分のデータをアーカイブするかを制御します。 ここに指定する値は、「次の期間を経過したデータをアーカイブ」フィールドの値よりも大きくなければなりません。 「次の期間を経過したデータを無視」行をブランクのままにすると、「次の期間を経過したデータをアーカイブ」行で指定した値より古いすべての日のデータがアーカイブされます。つまり、毎日アーカイブを行い、30 日より古いデータをパージする場合には、(31 日目にパージされるまで) 日次データを 30 回アーカイブすることになります。(CLI コマンド store storage-system を使って) システムで構成されたアーカイブ・オプションに応じて、EMC Centera または TSM オプションがパネルに含まれることがあります。このいずれかのアーカイブ宛先を選択する場合は、以下の該当するトピックを参照してください。
    a. EMC Centera のアーカイブとバックアップ
    b. TSM のアーカイブとバックアップ
5. アーカイブ・データを受信するホストの名前を、IP アドレスまたは DNS で「ホスト」に入力します。

6. 「ディレクトリー」ボックスで、データの格納先ディレクトリーを指定します。FTP または SCP のどちらのファイル転送方式を使用するかに応じて、指定方法が異なります。FTP の場合、FTP アカウントのホーム・ディレクトリーを基準にした相対パスでディレクトリーを指定します。SCP の場合、絶対パスとしてディレクトリーを指定します。
7. 「ユーザー名」ボックスで、ホスト・マシンへのログオンに使用するユーザー名を入力します。このユーザーは、「ディレクトリー」ボックスで指定したディレクトリーに対する書き込み/実行権限を保持していなければなりません。
8. ユーザーのパスワードを「パスワード」ボックスに入力した後、「パスワードの再入力」ボックスに再入力します。
9. データ・パージ
10. アーカイブされるかどうかに関わらずデータをパージするには、「パージ」チェック・ボックスを選択します。このボックスにチェック・マークを付けると、「次の期間を経過したデータをパージ」フィールドが表示されます。 重要: パージ構成は、データ・アーカイブとデータ・エクスポートの両方で使用されることに注意してください。ここで行った変更は、すべてのデータ・エクスポートの実行に適用されます。データ・アーカイブの場合も同様です。パージがアクティブになっていて、データ・エクスポートとデータ・アーカイブの両方が同じ日に実行される場合には、最初に実行された操作が古いデータをすべてパージした後に 2 番目の操作が実行されます。そのため、データ・エクスポートとデータ・アーカイブが共に構成されているときは常に、エクスポート基準経過日数とアーカイブ基準経過日数の両方よりもパージ基準経過日数の方が大きくなければなりません。
11. データをパージする場合は、「次の期間を経過したデータをパージ」フィールドを使用して、パージ操作の対象となる開始日を、当日 (0 日) よりさかのぼった日数、週数、または月数として指定します。指定した日、およびそれより古いすべての日のデータは、注に示す例外を除いてすべてパージされます。パージ開始日に指定する値は、「次の期間を経過したデータをアーカイブ」に指定した値よりも大きくなければなりません。さらに、データ・エクスポートがアクティブである場合 (統合アプライアンスへの「データのエクスポート」を参照)、ここに指定するパージ開始日は、「次の期間を経過したデータをエクスポート」の値よりも大きくなければなりません。 それ以前の操作によってまだアーカイブ/エクスポートされていないデータをパージする際には、警告は出されません。パージ操作では、リストア操作で指定される「リストアしたデータをパージしない」時間枠の範囲内に経過日数が入っているリストア・データはパージされません。詳しくは、アーカイブ・データの「リストア」を参照してください。
12. 「スケジューリング」セクションを使用して、この操作を定期的に実行するためのスケジュールを定義できます。
13. 「保存」」をクリックすると、構成の変更が検証されて、保存されます。「保存」ボタンをクリックすると、システムは指定された「ホスト」、「ディレクトリー」、「ユーザー名」、「パスワード」を検証しようとします (テスト・データ・ファイルをそのロケーションに送信することにより)。
14. 「今すぐ 1 回実行」をクリックして、操作を 1 回実行します。

アグリゲーターでのオーファン・クリーンアップ

リストアされたデータがアグリゲーターに含まれる場合、そのリストアされたデータに関連するオーファン・クリーンアップは、データが最初にリストアされたときに設定された有効期限に従って実行されるように設定されます。

GuardAPI コマンドを使用して有効期限に関連する変更を行っても、リストアされたデータがオーファン・クリーンアップの対象となる日付には影響しません。

例えば、ユーザーがデータをリストアし、そのデータを 7 日間保持したいとします。したがって、このデータの有効期限は、本日から、このデータがオーファン・クリーンアップの対象になる 7 日後までの 7 日間です。

有効期限を変更しても、コンピューターに残されているデータは、最初に設計されたとおりにオーファン・クリーンアップの対象になります (データをより短い/より長い期間保持するように設定しても、そのデータがオーファン・クリーンアップの対象となる日付には影響しません。ユーザーは、特に有効期間をより長い期間に変更する場合、データを失わないようにするために、この点に特に注意する必要があります)。

EMC Centera のアーカイブとバックアップ

EMC Centera を使用するには、

1. 「管理」 > 「データ管理」 > 「データ・アーカイブ」をクリックして、「データ・エクスポート」を開きます。
2. 「データ管理」セクションで、「データ・アーカイブ」または「システム・バックアップ」をクリックします。初期状態では、「ネットワーク」ラジオ・ボタンがデフォルトで選択され、ネットワーク・バックアップ・パラメーターが表示されています。
3. 「EMC Centera」ラジオ・ボタンを選択します。EMC Centera パラメーターがパネルに表示されます。
4. 「保持」ボックスに、データを保持する日数を入力します。 最大値は 24855 (68 年) です。それより長く保存する場合には、後ほどデータをリストアして再び保存します。
5. 「Centera プール・アドレス」ボックスに、Centera プール接続文字列を入力します (例えば `10.2.3.4,10.6.7.8/var/centera/profile1_rwe.pea`)
6. 「PEA ファイルのアップロード」をクリックして、接続文字列に使用する Centera PEA ファイルをアップロードします。
7. 「保存」をクリックして構成を保存します。システムは、指定された接続文字列を使用してプールを開くことにより、Centera アドレスの検証を試みます。操作が失敗すると、通知メッセージが表示され、構成は保存されません。

TSM のアーカイブとバックアップ

アーカイブまたはバックアップの宛先として TSM を選択した場合、アーカイブ構成パネルまたはバックアップ構成パネルの TSM 部分が拡張されます。TSM をアーカイブ/バックアップの宛先として設定する前に、Guardium システムをクライアント・ノードとして TSM サーバーに登録しておく必要があります。TSM クライアント・システム・オプション・ファイル (dsm.sys) を (例えばご使用の PC 上に) 作成して、Guardium にアップロードする必要があります。さらに、そのファイルが定義される方法によっては、dsm.opt ファイルもまたアップロードする必要があります。 Guardium で使用するために dsm.sys ファイルを作成する方法については、所属する組織の TSM 管理者に問い合わせてください。TSM 構成ファイルをアップロードするには、CLI コマンド import tsm config を使用します。

TSM (または Spectrum Protect クライアント) ライフサイクルは、Spectrum Protect 製品用語で定義されます。

TSM を使用するには、

1. 「管理」 > 「データ管理」 > 「データ・アーカイブ」をクリックして、「データ・アーカイブ」を開きます。
2. 「TSM」ラジオ・ボタンを選択します。TSM パラメーターがパネルに表示されます。
3. 「パスワード」ボックスで、TSM サービスを要求するためにこの Guardium ユニットが使用する TSM パスワードを入力して、「パスワードの再入力」ボックスに再び入力します。
4. オプションで、dsm.sys ファイルの servername 項目と一致するように、「サーバー名」を入力します。
5. オプションで、「ホスト」に名前を指定します。
6. 「保存」をクリックして構成を保存します。「適用」ボタンをクリックすると、システムは dsmc アーカイブ・コマンドを使用してサーバーにテスト・ファイルを送信することにより、TSM 宛先の検証を試みます。操作が失敗すると、通知メッセージが表示され、構成は保存されません。

アーカイブとパージの停止

1. 「管理」 > 「データ管理」 > 「データ・アーカイブ」をクリックして、「データ・アーカイブ」を開きます。
2. 「アーカイブ」または「パージ」ボックスをクリアします。
3. 「保存」をクリックします。

アーカイブおよびパージ処理の検証

1. 「レポート」 > 「Guardium 運用レポート」 > 「統合/アーカイブ・ログ」 をクリックして、「統合/アーカイブ・ログ」を開きます。
2. それぞれのアーカイブ/パージ操作の状況が「成功」になっていることを確認します。

統合およびアーカイブ・アクティビティーに関するレポート

1. 「管理」 > 「レポート」 > 「データ管理」 > 「統合/アーカイブ・ログ」にナビゲートして、「統合/アーカイブ・ログ」を開きます。
2. 照会を定義してレポートを作成します。

リストア

前述したように、アーカイブは SCP ホストまたは FTP ホストに、あるいは Centera ストレージ・システムまたは TSM ストレージ・システムに書き込まれます。アーカイブをリストアするには、データのリストア先となる Guardium システムに 1 つ以上の適切なファイルをコピーする必要があります。各日のデータに対して 1 つの別個のファイルがあります。 アーカイブ/パージ操作の構成方法によっては、同じ日に対して、アーカイブ・データの複数コピーが存在する場合があります。アーカイブとエクスポートのデータ・ファイル名は同じ形式です (<daysequence>-<hostname.domain>-w<run> datestamp>-d<data_date>.dbdump/TAR ファイル)。バックアップ・システムではなく、アーカイブ済みデータのファイルをリストアするには、「カタログ・アーカイブ」という GUI 画面を使用する必要があります。アーカイブ操作とリストア操作は、アーカイブ処理時に生成されるファイル名に依存します。アーカイブ・ファイルの名前を決して変更しないでください。生成済みのファイル名が変更された場合、リストア操作は正常に機能しません。

例えば 732423-g1.guardium.com-w20050425.040042-d2009-04-22.dbdump/TAR ファイルとなります。

その月に作成された最初のアーカイブからデータをリストアする場合を除いて、複数日のデータをリストアする必要があります。その理由は、Guardium がデータをリストアする際、リストア対象のデータがアーカイブされたときのすべての情報が必要になるためです。アーカイブが作成された後、そのような情報の一部は、使われないため既にパージされた可能性があります。各月にデータが初めてアーカイブされるとき、リストア操作に必要なすべての情報が自動的にアーカイブされます。したがって、データをリストアする場合、月の第 1 日をリストアして、それ以降、目的の日までのすべての日をリストアするか、目的の日をリストアしてから、以降の月の第 1 日をリストアできます。

例えば 6 月 28 日をリストアするには、6 月 1 日から 6 月 28 日までをリストアするか、あるいは 6 月 28 日と 7 月 1 日をリストアします。

バックアップ・システムではなく、アーカイブ済みデータのファイルをリストアするには、「カタログ・アーカイブ」という GUI 画面を使用する必要があります。アーカイブ操作とリストア操作は、アーカイブ処理時に生成されるファイル名に依存します。アーカイブ・ファイルの名前を決して変更しないでください。生成済みのファイル名が変更された場合、リストア操作は正常に機能しません。

1. 「管理」 > 「データ管理」 > 「データのリストア」をクリックして、「データのリストア」を開きます。
2. 「開始」ボックスに日付を入力し、データを必要とする最も古い日付を指定します。
3. 「終了」ボックスに日付を入力し、データを必要とする最も新しい日付を指定します。
4. 「ホスト名」 ボックスに、アーカイブの起点となる Guardium アプライアンスの名前をオプションで入力します。
5. 「検索」をクリックします。
6. 「検索結果」パネルで、リストアする各アーカイブの「選択」ボックスにマークを付けます。
7. 「リストアしたデータを少なくとも次の期間パージしない」ボックスに、リストアしたデータをアプライアンスに保持する日数を入力します。
8. 「復元」をクリックします。
9. 完了したら、「完了」をクリックします。

トラブルシューティング

技術サポートへのエスカレーションを行うときには、問題発生時からの詳細なログを提供してください。「管理」 > 「レポート」 > 「データ管理」 > 「統合/アーカイブ・ログ」にナビゲートし、該当する期間のレポートを定義します。

アグリゲーター当たりのコレクター最大数の計算

Guardium システムが .ISO から作成される場合、アグリゲーター当たりのコレクター最大数にはデフォルト値 10 が設定されます。

お客様が Guardium システムをアップグレードすると、システムは、以下のロジックを使用してコレクターの最大数を計算します。

1. 内部の Guardium 表のデータにしたがって、コレクターの数を取得します。デフォルト値は 10 です。
2. ステップ 1 の結果が 0 (コレクターが見つからない) である場合、システムはこの値を 10 に設定します。
3. 異なる数のコレクターが見つかった場合、システムは、ステップ 2 で判別される数に 20 % を追加します。
4. 例えば、ステップ 1 でコレクターが見つからなかった場合、ステップ 2 で値 10 を設定してから、ステップ 3 で 20% を追加して 12 にします。
5. 別の例では、ステップ 1 でシステムはアグリゲーターにエクスポートする 5 つのコレクターを検出しました。この場合、値は 5 に設定されます。結果が 5 であり、0 でなかったので、ステップ 2 は該当しません。ステップ 3 は、5 に 20% を追加し、この値を 6 に設定します。

親トピック: 統合および一元管理

一元管理

一元管理構成では、1 つの Guardium® ユニットが中央マネージャーとして指定されます。 このユニットは、他の Guardium ユニット (管理対象ユニットと呼ばれる) をモニターして制御するために使用できます。 管理対象外のユニットはスタンドアロン・ユニットと呼ばれます。

ローカル・マシンという概念は一元管理システム内の任意のマシンを指します。アプリケーションによっては (監査プロセス、照会、ポートレットなど)、管理対象ユニットと中央マネージャーの両方で実行できるものがあります。 どちらの場合も、定義は中央マネージャーから、データはローカル・マシンから来ます (ローカル・マシンが中央マネージャーの場合もある)。

一元管理システムが設定されると、お客様は中央マネージャーと管理対象ユニットのどちらかを使用して、大部分の定義の作成または変更を行うことができます。 実際の編集をどのマシンで行っているかには関わらず、定義のほとんどは中央マネージャー上にあることに留意してください。

注:

- リモート・ソース機能を使用すると、マネージャー上のユーザーは (正しいロール特権を保持していれば) 任意のレポートを管理対象ユニット上で実行でき、また、その管理対象ユニットのデータと情報を表示できます。
- CAS テンプレート定義は他のすべての定義 (レポート、ポリシー、アラートなど) と同様に、フェデレーテッド環境内のすべてのユニット間で共有されます。
- ユーザーによる CAS レポートの実行は、マネージャー上で行うことを推奨します。特に、CAS 構成、ホスト、およびテンプレートと関連する CAS レポートについてはそのようにしてください。
- 「カスタム・ドメイン・ビルダー」を使用してリモート表 (中央マネージャー環境内のマネージャー上にある表。「データ・ソース」や「コメント」など) の一部またはすべてを使用するレポートを作成している場合、そのレポートは管理対象ノード上では動作しません。データは戻されません。
- マネージャーの「一元管理」ページでは、特定の時間間隔に基づく自動的なリフレッシュは行わなくなりました。このページは、システムの GUI タイムアウトに基づいてタイムアウトになります。
- 一定期間アクティビティーがない場合、システムは自動的にユーザーをログアウトし、再度サインインするよう求めます。GUI タイムアウトの長さは CLI コマンド show/store session timeout を使用して設定できます (デフォルトは 900 秒)。セッションがアクティブな間は、状況ライトが 5 分ごとにリフレッシュされます。
- ユーザーがデータを中央マネージャーから管理対象ノードに同期またはアップロードしようとしている場合、そのようなタイプのアクティビティーに関係するすべてのノードでは、Guardium のバージョンが同一でなければなりません。
- 予備の一元管理の移行では、その一元管理環境で定義されているユニット数によって、ユニット・タイプの同期化の実行に最大 5 分かかる場合があります。

- Guardium コンポーネント・サービス
  一元管理環境内の Guardium コンポーネントと、その取得元ロケーションを識別します。
- 一元管理の実装
  特定のマシンを中央マネージャーにして、他のマシンを一元管理システムに接続し、管理対象ユニットを登録して中央マネージャーと通信できるようにします。
- 一元管理機能の使用
  一元管理機能を使用すると、ポータル・ユーザー・アカウントの同期化、管理対象ユニットのモニター、および管理対象ユニットへのセキュリティー・ポリシーのインストールを行うことができます。

親トピック: 統合および一元管理

Guardium コンポーネント・サービス

一元管理環境内の Guardium コンポーネントと、その取得元ロケーションを識別します。

この装置は、他の Guardium 装置 (管理対象装置と呼ばれる) をモニターして制御するために使用できます。 管理対象外のユニットはスタンドアロン・ユニットと呼ばれます。

表 1. Guardium コンポーネント・サービス

コンポーネント	記述
ユーザー、ロール、およびアクセス権	中央マネージャーは、すべての管理対象システムのユーザー、ロール、グループ、およびデータマート表の各定義を制御します。中央マネージャーは、すべてのユーザー、セキュリティー・ロール、グループ、およびデータマート表の定義を含むセットを、スケジュールに基づいて、またはオンデマンドでエクスポートします。管理対象ユニットは、内部データベースを 1 時間ごとに更新します。その結果、中央マネージャー上でユーザー、ロール、アクセス権、またはデータマート表が追加または変更された時刻と、管理対象ユニットでそれらの更新が適用される時刻との間に、最大 1 時間の遅延が生じる可能性があります。 注: Guardium® ユーザーまたはセキュリティー・ロールを、一元管理に登録する予定の既存のスタンドアロン・ユニット上で定義した場合、そのユーザーとセキュリティー・ロールを中央マネージャー上でも定義しない限り、それらの定義はそのシステムが登録された後に使用可能になりません。 管理対象ユニット上でユーザーやセキュリティー・ロールの管理を行うことはできません。そのような定義は中央マネージャーにログオンしたときにのみ管理できます。あるユニットが一元管理に未登録の場合、追加されたすべてのユーザーとセキュリティー・ロールは、デフォルト・ユーザー (admin、accessmgr) 以外はすべて削除されます。アクセラレーター・アドイン製品 (PCI、SOX など) を中央マネージャー環境にインストールする場合は、最初に中央マネージャーにインストールしてから、管理対象ユニット上にインストールします。アクセラレーターで必要なすべてのロールとユーザーは、中央マネージャー上で追加します (そこから管理対象ユニットに同期されます)。アクセラレーターの資料は、アクセラレーター・モジュールに含まれています。このコンポーネント・サービス表の最後に記載されている PCI アクセラレーターの概要を参照してください。
別名とグループ	自動的に別名またはグループを生成するすべてのプロセス (LDAP からのユーザー・グループのインポート、照会によるグループ生成、照会による別名生成、分類など) において、(同一マネージャーが管理する) 複数の管理対象マシンに同一のグループまたは別名が自動生成される場合、既存のグループまたは別名との間に競合が発生する可能性があります。この場合、既存のグループが置換されることはありません。
監査プロセス	監査プロセス自体の定義とそれに対応するタスクすべての定義は、中央マネージャーに保存され、すべての管理対象ユニットでそれらを使用できます。 しかし、スケジュール、結果、および To-Do リストはローカル・マシンに保存されます。 このことは、同一の監査プロセス・タスクがすべての管理対象ユニット上、および中央マネージャー上で実行できることを意味します。しかし監査プロセス・タスクは、異なる時間に異なるマシン上で実行することもできます。これは、管理対象ユニットの負荷期間のピークが異なる場合に役立ちます。 各マシンは、そのマシンが収集したデータに基づき、それぞれ独自の結果セットを保持します。また、各マシンはすべてのユーザーの To-Do リストの独自のセットを保持します。監査プロセス定義は中央マネージャーから管理対象ユニットにユーザー同期プロセスの一部としてエクスポートされます (『ポータル・ユーザー・アカウントの同期』を参照)。 監査プロセスの結果が作成されると、ユーザーはその結果を使用できるようになります。しかし、管理対象ユニット上では、「未処理監査プロセスのレビュー」などのレポートまたはモニターが更新されるまでに最大 1 時間の遅延が生じる場合があります。
照会	各照会では、単一のマシンからデータベース情報のみを取得することができます。中央マネージャーの定義と管理対象ユニットのデータの両方を含む、アクセス情報を必要とする照会では、データが表示されないか、データが欠落します。

コンポーネント	記述
ポリシー	ポリシー定義は中央マネージャーに保存されます。ただし、管理対象ユニットにポリシーをインストールすると、ローカル・コピーが作成されて管理対象ユニットに保存されます。これは、なんらかの理由で中央マネージャーが使用できない場合でも、管理対象ユニットでデータベース・アクティビティーをモニターしてポリシーを使用し続ける必要があるためです。 注: ポリシーを管理対象ノード上にインストールしても、中央マネージャー上の「リフレッシュ」をクリックするまでは、このポリシーは中央マネージャーにアップロードされません。ポリシーをインストールする際には、中央マネージャーと管理対象ユニットのバージョンが同じでなければなりません。異なる場合は、ポリシーはインストールされずエラーが生成されます。
レポート	レポート定義は中央マネージャーに保存されます。 中央マネージャー上でポートレットの再生成が呼び出されると、すべての管理対象ユニットに対してもポートレットの再生成 (レポート ID 付き) の管理要求 (HTTPS) が送信されます。再生成が管理対象ユニット上で呼び出されるとき、それが画面から (管理要求ではなく) 呼び出された場合は、ポートレットのリフレッシュを行うようにマネージャーに対して管理要求が送信されます (これはすべてのユニットにも送信されます)。 あるユニットが停止した場合に備え、管理要求には持続性メカニズムがあります。このトピック内の登録とポリシーのインストールに関するセクションを参照してください。 中央マネージャーでは、レポートと監査プロセスで管理対象ユニットからのデータを使用できますが、管理対象アグリゲーターからのものは使用できません。 管理対象ユニットは、ランタイム・パラメーターとして選択され、リモート・データ・ソースとして参照され、さらに管理対象ユニットのみを含むフィルタリングされたドロップダウン選択リストとして表示されます。 監査プロセスがリモート・データ・ソースを参照するとき、監査プロセスは中央マネージャーからのみ実行できます。そのため、これは管理対象ユニット上に表示される監査プロセスのリストには出現しません。 注: ドメイン「スニファーのバッファー使用」(例えば、リクエスト・レート、CPU 使用量、バッファー使用状況モニターなど) の、中央マネージャー上にある特定のレポートでは、データがまったく表示されません。レポートは空になります。
セキュリティー・アセスメント	監査プロセスと同様、セキュリティー・アセスメントの定義自体は中央マネージャーに保存されます。しかし、結果はローカル・マシンに保存されます。 このことは、同一のセキュリティー・アセスメントを、すべての管理対象ユニット上および中央マネージャー上で実行できることを意味します。
コメント	コメントは、そのコメントが関連付けられている対象に応じて、ローカル・マシンと中央マネージャーのいずれかに保存できます。コメントが中央マネージャーにある定義に関連付けられている場合、コメントも Central Manager に保存されます。コメントがローカル・マシンにある結果、または管理対象ユニット固有のもの (検査エンジンなど) に関連付けられている場合、コメントもローカル・マシンに保存されます。
スケジュール	スケジュールは常にローカル・マシンに保存されます。これは、定義が中央マネージャーに保存されている場合でもそうです。
非中央マネージャー・タスク	サーバーが中央マネージャーとして構成されると、あるタスクはそのユニット上で実行できず、他の (非 Central Manager) ユニット上で実行する必要があることに注意してください。 検査エンジンは中央マネージャー上では定義できず、管理対象ユニット上でのみ作成可能です。 しかし、検査エンジンを中央マネージャーで表示することはできます。
アップグレードの考慮事項	中央マネージャーと管理対象ユニットは同じバージョンにすることを推奨します。中央マネージャーを最初にアップグレードし、管理対象ユニットはその後にアップグレードする必要があります。 マネージャーのバージョンが管理対象ユニットのバージョンと異なる状況は、一時的なものにしてください。すべての管理対象ユニットは、マネージャーと同じバージョンにアップグレードすることを強く推奨します。アップグレード後には、同期 (リフレッシュ) をすべての管理対象ノード上で実行することにより、これらの管理対象ノードが適切なソフトウェア・バージョンを認識するようにします。
コンプライアンスのための PCI アクセラレーター	PCI データ・セキュリティー基準は、12 の基本的な要件から構成されています。このうちの多くの要件は、物理的なインフラストラクチャーの保護 (例えば、要件 1: データを保護するためのファイアウォール構成のインストールと保守) や、手続き上のベスト・プラクティスの実施 (例えば、要件 5: アンチウィルス・ソフトウェアの使用と定期的な更新) に焦点を当てています。ただし、そのほかに、カード所有者データへのアクセスのリアルタイムでのモニターとトラッキング、およびデータベース・セキュリティーの正常性状況の連続的なアセスメント (例えば、要件 10: ネットワーク・リソースおよびカード所有者データへのすべてのアクセスのトラッキングおよびモニター) についても、非常に強調されています。 Guardium のデータベース・コンプライアンス用の PCI アクセラレーターは、これらのモニターとトラッキングの要件をサポートし、カード所有者データのセキュリティーを確保するために必要な組織内のプロセスを簡素化するように設計されています。アクセラレーター・レポートのテンプレートは、特定の組織要件と法的要件を直接反映するようカスタマイズすることができます。これらのテンプレートにアクセスするには、用意されている以下のタブを使用します。 • PCI データ・セキュリティー基準の概要 • 計画と編成 • PCI 要件 10: アクセスのトラッキングとモニター • PCI 要件 11: 定期的なテストと検証 • PCI ポリシー違反のモニター 法規制を満たすのに役立つ Guardium ソリューション・ファミリーのその他のツールには、以下のようなものがあります。 • PCI コンプライアンス・レポート・カード - カード所有者データベース・アクセスのセキュリティーの正常性を示す詳細なビュー。ユーザー定義のテスト、重み付け、アセスメントに合わせてカスタマイズされた、連続するリアルタイムのスナップショットをこのビューで使用することにより、コンプライアンス・プロセスが自動化されます。レポート・カードは、セキュリティー・アセスメントを使用して生成することができます。 • 完全な監査証跡 - 法規制へのコンプライアンスのために必要なデータの使用状況と変更内容に関する完全な監査証跡を、ユーザーに負担がかからない方法で生成します。 • 自動化されたスケジューリング - 組織全体における PCI ワークフロー、監査タスク、担当者への情報の通知について、自動的にスケジューリングを行います。

以下の表は、一元管理環境でどのコンポーネントがどのロケーションから取られるかを識別するのに役立ちます。

表 2. 中央マネージャー環境内のコンポーネントおよびロケーション

中央マネージャー	管理対象ユニット
ユーザー	システム構成
セキュリティー・ロール	検査エンジン
アプリケーション・ロールの権限	アラート機能 (構成)
照会	異常検出
レポート	セッション推論
期間	IP からホスト名への別名割り当て
アラート	システム・バックアップ
セキュリティー・アセスメント	統合/アーカイブ
監査プロセス定義	カスタム・アラート
プライバシー・セット	カスタム識別プロシージャー
	csv エクスポート出力
ポリシー	スケジュール
グループ	DB オートディスカバリー構成
別名	監査プロセスの結果

ユーザー、セキュリティー・ロール、監査プロセスの各定義、およびグループは、後で説明するように、中央マネージャーからすべての管理対象ユニットへ、スケジュールに基づいてエクスポートされます。

中央マネージャーから、管理者は以下のことを実行できます。

- 管理対象として Guardium ユニットを登録する
- 管理対象ユニットをモニターする (ユニットの使用可能性、検査エンジンの状況など)
- 管理対象ユニットのシステム・ログ・ファイル (syslogs) を表示する
- 管理対象ユニット上のデータを使用してレポートを表示する
- 管理対象ユニットの主な統計を表示する
- Guardium セキュリティー・ポリシーを管理対象ユニットにインストールする
- 管理対象ユニットを再始動する
- 管理対象ユニット上で Guardium 検査エンジンを管理する
- すべての管理対象システム上で使用されるユーザー、セキュリティー・ロール、グループ、およびアプリケーション・ロール権限の完全なセットを保守する
- パッチ配布
- アップロードした JAR ファイルの配布
- パッチ・バックアップ設定の配布
- 認証構成の配布
- 構成の配布

注: 管理者は、アプリケーション・ロール権限を任意の管理対象ユニットから変更できます。これを行うと、権限はすべての管理対象ユニットで変更されます。
親トピック: 一元管理

一元管理の実装

特定のマシンを中央マネージャーにして、他のマシンを一元管理システムに接続し、管理対象ユニットを登録して中央マネージャーと通信できるようにします。

- 新規インストールでの一元管理の実装
- 既存インストールでの一元管理の実装
- 一元管理ユニットが使用不可の場合

- 新規インストールでの一元管理の実装
  1 台のマシンを中央マネージャーにして、同じ共有パスワードを使用し、ユニットを登録して、管理対象ユニットをグループ化します。
- 既存インストールでの一元管理の実装
  既存の Guardium 環境に一元管理を実装し、アクティブ・インスタンスを持つ CAS コレクターを管理対象にマイグレーションします。

親トピック: 一元管理

新規インストールでの一元管理の実装

1 台のマシンを中央マネージャーにして、同じ共有パスワードを使用し、ユニットを登録して、管理対象ユニットをグループ化します。

1 台のマシンを中央マネージャーにする作業

まず、1 台のマシンを中央マネージャーにします。マシンを選択します。次に、以下の手順を実行します。

1. 中央マネージャーにするマシンの CLI にログインします。
2. store unit type manager と入力します。このステップによって、マシンは中央マネージャーになります。ただし、まだ何も管理していません。

同じ共有パスワードの使用

中央マネージャーを用意したら、他のマシンを一元管理システムに接続する必要があります。セキュリティー上の理由で、マシン間の通信は、同じ共有パスワードを使用して暗号化することが要件になっています。このステップを実行するには、以下のアクション項目を実行します。

1. 「設定」 > 「ツールとビュー」 > 「システム」をクリックして、「システム」を開きます。
2. すべてのシステムで、共有パスワードを同じ文字列に設定します。

- ユニットの登録
  管理対象ユニットを登録して、中央マネージャーと通信できるようにします。
- 管理対象ユニットの登録抹消
  ユニットが登録抹消されたときは、必ず中央マネージャーからそのユニットを登録抹消してください。このメソッドは、中央マネージャーで管理対象ユニットの数を減らす唯一の方法です。
- ポータル・ユーザー・アカウントの同期
  中央マネージャーを使用して、ポータル・ユーザーの同期を管理します。

親トピック: 一元管理の実装

ユニットの登録

管理対象ユニットを登録して、中央マネージャーと通信できるようにします。

一元管理への Guardium 装置の登録は、Central Manager からでも、その装置自体からでも行えます。登録をどのように行った場合でも、中央マネージャーとすべての管理対象ユニットは同じシステム共有パスワードを持つ必要があります。管理対象にするユニットが別のマネージャーの一元管理に既に登録されている場合、その中央マネージャーからユニットを登録抹消してから、新規マネージャーに登録します。一元管理に登録および登録抹消したときにそのユニットに対して実行される処理を必ず正確に把握するようにしてください。

注: 管理対象ユニットにログインしているユーザーが中央マネージャーに存在しない場合は、セッションが無効になります。そのユニットが中央マネージャーに登録されるまで、セッションは無効な状態のままです。

登録時に実行される処理

登録時には、以下のアクションが実行されます。

- ユニット・タイプが管理対象に設定され、マネージャー IP が保管されます。
- マネージャーのプロダクト・キーが適用されます。(ライセンス・キーは、ping またはユーザー同期では伝搬しません。登録時、またはシステムのリフレッシュ時に送信されます。)
- すべてのジョブ・スケジューリングがデフォルトにリセットされます。
- すべての psml ファイル (ポータル GUI カスタマイズ) が削除されます。
- すべてのローカル・ユーザーとロールが削除されます。
- 評価されない、しきい値アラートのリストがリセットされます。
- ユーザー・ロール、マネージャーからのアクセス権がロードされます。
- カスタム・クラス、ユーザーがアップロードした JAR、マネージャーからの LDAP トラストストアがアップロードされます。
- 管理対象からマネージャーへのデータベース接続が有効になります。
- マネージャーから管理対象へのデータベース接続が有効になります。
- 必要に応じて、CAS リスナーが始動します。

登録の後、レポート、照会、グループ、ポリシー、監査などのすべての定義が中央マネージャーから取得されます。

登録したユニットの状況がオフラインのままである場合

登録したユニットがオンラインになっており、中央マネージャーからアクセス可能であるにもかかわらず、状況がオフラインのままである場合は、以下の手順を実行します。

- 管理対象にする装置がオンラインで、アクセス可能なおかつ作動可能であるかどうか検査します。これは、ブラウザー・ウィンドウを使用して、その装置の Guardium システムにログインして行います。
- そのユニットの「リフレッシュ」をクリックします。
- ユニットの IP アドレスの入力が正しいことを確認します。
- ユニットが中央マネージャーと同じ共有パスワードを持っていることを確認します。

注: ユニットの登録がオフラインの場合は、登録要求が保持されます。この要求は、ユニットが登録されるまで、指定された IP/ポートに、設定された間隔で再送されます。正常に実行されない登録要求は、7 日後に有効期限切れとなります。

管理対象ユニットからの登録

管理対象ユニット上で GUI を使用して、ユニットを中央マネージャーに登録できます。あるいは、CLI の register コマンドを使用することもできます。これについては、『CLI を使用した管理対象ユニットの登録』で説明します。

1. 「設定」 > 「一元管理」 > 「登録およびロード・バランシング」をクリックして「一元管理登録」を開きます。
2. 「ホスト IP」に、中央マネージャーの IP アドレスを入力します。
3. 「ポート」に、中央マネージャーの https ポート (通常は 8443) を入力します。
4. 「登録」をクリックします。

管理対象ユニット上で登録を行うとすぐに中央マネージャーとの通信が開始され、これ以上の操作は必要ありません。

注: 一元管理で登録を行うとき、一元管理ユニットはオンラインで、このユニットからアクセス可能でなければなりません。これとは対照的に、一元管理ユニットからユニットを管理対象として登録するときは、現在アクセス可能ではないユニットの登録ができます。

CLI を使用した管理対象ユニットの登録

1. 管理対象ユニットで、CLI にログインします。
2. `register management <Manager IP> <Manager Port>` と入力します。

管理対象ユニット上で登録を行うとすぐに中央マネージャーとの通信が開始され、これ以上の操作は必要ありません。

中央マネージャーからのユニットの登録

現在アクセスできないユニットを登録できます。

1. 「管理」 > 「一元管理」 > 「一元管理」にナビゲートして「一元管理」を開きます。
2. 「新規登録」をクリックします。「ユニット登録」ページが開きます。
3. ユニットの IP およびポートを入力し、「保存」をクリックします。「一元管理」ページがリフレッシュされ、新しいユニットが表示されます。

親トピック: 新規インストールでの一元管理の実装

管理対象ユニットの登録抹消

ユニットが登録抹消されたときは、必ず中央マネージャーからそのユニットを登録抹消してください。このメソッドは、中央マネージャーで管理対象ユニットの数を減らす唯一の方法です。

管理対象ユニットからユニットを登録抹消しても、そのユニットが中央マネージャー上で登録抹消されることはありません。中央マネージャーは、ライセンス交付の目的で引き続きそのユニットを管理対象ユニットとしてカウントし、そのユニットを管理対象として扱います。これにより、別のユニットを中央マネージャーに登録できなくなる可能性があります。管理対象ユニット上の登録抹消機能は、緊急時の使用のみを目的として組み込まれています。Manager がサービスを提供しなくなった場合、ユニットは、登録抹消してからでなければ別のマネージャーに登録できません。

管理対象ユニットからユニットを登録抹消しても、そのユニットは引き続き中央マネージャー画面に表示されます。そのユニットの「リフレッシュ」ボタンを押すと、そのユニットが再登録されます。そのユニットの他の操作ボタンを押すと、そのユニットが管理対象ではなくなったことを示すメッセージが表示され、マネージャーからユニットが削除されます。

管理対象ユニット上で GUI を使用して、ユニットを中央マネージャーから登録抹消できます。また、CLI の unregister コマンドを使用することもできます。これについては、『CLI を使用した管理対象ユニットの登録抹消』で説明します。

1. 管理対象ユニットの Guardium UI に admin としてログインします。
2. 「設定」 > 「一元管理」 > 「登録およびロード・バランシング」をクリックして「一元管理登録」を開きます。
3. 「登録抹消」をクリックします。

登録抹消時に実行される処理

登録抹消時には、以下のアクションが実行されます。

- ユニット・タイプがスタンドアロンに設定されます。
- マネージャーの IP がクリアされます。
- プロダクト・キーがクリアされます (新規マネージャーに登録するかライセンスを手動でロードするまではライセンスは NULL です)。
- 評価されない、しきい値アラートのリストがリセットされます。
- すべてのジョブ・スケジューリングがデフォルトにリセットされます。
- Psml ファイルが削除されます。
- デフォルト・ユーザー (admin、accessmgr) 以外のすべてのユーザーが削除されます。
- 管理対象からマネージャーへのデータベース接続が無効になります。
- GUI が再始動されます。

登録抹消の後、レポート、照会、グループ、ポリシー、監査などのすべての定義はローカル・データベースから取得されます。中央マネージャーに保管された定義にはアクセスできなくなります。

確認方法が分からない場合は、ユニットを登録抹消する前に Guardium サポートに連絡してください。

中央マネージャーからのユニットの登録抹消

1. 中央マネージャーに admin としてログインします。
2. 「管理」 > 「一元管理」 > 「一元管理」をクリックして、「登録」を開きます。
3. 登録抹消を行う管理対象ユニットに対応するチェック・ボックスにマークを付けます。
4. 「登録抹消」をクリックします。

管理対象ユニットを中央マネージャー画面から登録抹消すると、管理対象ユニットのリストからそのユニットが削除され、スタンドアロン・ユニットに設定されます。

注: そのユニットのプロダクト・キーは削除され、ユニットを別のマネージャーに登録しない限り、そのプロダクト・キーは手動で設定されることになります。

管理対象ユニットからの登録抹消

管理対象ユニットで UI を使用して、ユニットを中央マネージャーから登録抹消できます。また、CLI の unregister コマンドを使用することもできます。これについては、『CLI を使用した管理対象ユニットの登録抹消』で説明します。

1. 管理対象ユニットに admin としてログインします。
2. 「設定」 > 「一元管理」 > 「登録およびロード・バランシング」をクリックして「登録」を開きます。
3. 「登録抹消」をクリックします。

CLI を使用して管理対象ユニットを登録抹消するには、以下の手順を実行します。

1. 管理対象ユニットで、CLI にログインします。

2. 「unregister management」と入力します。

管理対象ユニットから登録抹消した後、中央マネージャーとの通信が切断され、これ以上の操作は必要ありません。

親トピック: 新規インストールでの一元管理の実装

ポータル・ユーザー・アカウントの同期

中央マネージャーを使用して、ポータル・ユーザーの同期を管理します。

このタスクについて

前述したように、中央マネージャーはすべての管理対象ユニットのユーザー、セキュリティー・ロール、グループ、およびデータマート表の各定義を制御します。中央マネージャーは、すべてのユーザー・ロールとセキュリティー・ロールのセットの、暗号化された署名付きのコピーを作成します。また、中央マネージャーは、その情報をすべての管理対象ユニットに送信します。さらに、ローカル処理で必要な他の一部の定義 (グループとグループ・メンバー、監査プロセス、別名など) もコピーされます。管理対象ユニットは、その後、内部データベースを 1 時間ごとに更新します。このプロセスは、ロールまたはデータマート表の使用において最大 1 時間の遅延が生じる可能性があることを意味します。

ユーザー同期の全サイクルが実行されるのは、登録時、または一元管理画面で「リフレッシュ」をクリックしたときです。どちらの場合も、同期情報がマネージャーから送信され、管理対象ユニットにただちにロードされます。

注: スケジュールを設定するときは、他のスケジュール済みジョブ (インポートなど) を妨げないように注意してください。そのジョブを開始できなくなる可能性があります。

手順

ポータル・ユーザーの同期を管理するには、「管理」 > 「一元管理」 > 「ポータル・ユーザー同期」をクリックします。

a. 「スケジュールの変更」 をクリックして、標準のタスク・スケジューラーを使用してユーザー同期タスクのスケジュールを変更します。
b. タスクがアクティブにスケジュールされている場合、「一時停止」をクリックすると、それ以後のスケジュールされた実行を停止します。
c. タスクが一時停止している場合、「再開」をクリックすると、タスクの実行を (定義済みのスケジュールに従って) 再度開始します。
d. 「今すぐ 1 回実行」をクリックすると、同期タスクがただちに実行されます。
   注: スケジュールされているタスクまたは「今すぐ 1 回実行」されるタスクとは、データの収集と管理対象ユニットへのそのデータの送信のみを指します。管理対象ユニットは、データを受信してから最大 1 時間後まで、そのデータを使用したユーザー表の更新をしない場合があります。

親トピック: 新規インストールでの一元管理の実装

既存インストールでの一元管理の実装

既存の Guardium 環境に一元管理を実装し、アクティブ・インスタンスを持つ CAS コレクターを管理対象にマイグレーションします。

既存の Guardium 環境では、概略を示す手順を参照して、一元管理の実装計画を作成します。既存の Guardium ユニットを中央マネージャーに変換する場合は、中央マネージャーがネットワーク・トラフィックをモニターできないことに留意してください。例えば、検査エンジンを中央マネージャー上で定義することはできません。

1. 中央マネージャーおよびすべての管理対象ユニットで使用するシステム共有パスワードを選択します。詳しくは、『システム構成』のシステム共有パスワードに関するトピックを参照してください。
2. 中央マネージャー・ユニットをインストールするか、既存システムの 1 つを Central Manager として指定します。どちらの場合でも、store unit type コマンドを使用して、中央マネージャーにマネージャー属性を設定します。
3. スタンドアロン・ユニットからの定義で一元管理環境で使用可能にするものはすべて、そのスタンドアロン・ユニットを管理対象として登録する前にエクスポートしておく必要があります。後で、それらの定義は中央マネージャーにインポートされます。定義をエクスポートまたはインポートする前に、管理対象ユニットとなるスタンドアロン・ユニットごとに、ここで示す手順を実行してください。『定義のエクスポート/インポート』の概要情報を参照してください。
   - システムが管理対象ユニットになった後に使用できるようにするスタンドアロン・システムの定義を決定します。スタンドアロン・システム上のコンポーネントのうち、使用可能にしないものについては無視してください。
   - スタンドアロン・ユニット上で定義されていたセキュリティー・ロールとグループを中央マネージャー上で定義されているものとを比較します。一元管理下では、これらの定義の単一のバージョンをすべてのユニットに適用します。同じ名前のセキュリティー・ロールが両方のシステムに存在し、異なる目的で使用されている場合、中央マネージャーにロールを新規追加し、定義のインポート後に該当する定義にその新規ロールを割り当てます。
   - 同じグループ名がスタンドアロン・ユニットと中央マネージャーに存在し、そのメンバーが異なっている場合、スタンドアロン・システム上に複写したグループを新規作成します。このとき、中央マネージャー上に存在しないグループ名を選択するよう注意してください。エクスポートする対象の定義すべてにおいて、古いグループ名への参照を新規グループ名への参照に変更します。
   - すべての定義に割り当てられたセキュリティー・ロールは、すべてスタンドアロン・システムからエクスポートされます。定義をインポートするときは、ロールなしでインポートされます。そのため、ロールは手動で追加する必要があります。
   - 各システムのアプリケーション・ロール権限を確認します。スタンドアロン・ユニット上のアプリケーションに割り当てられたセキュリティー・ロールのいずれかが中央マネージャーで欠落している場合は、それを中央マネージャーに追加します。
   - システムが管理対象ユニットになった後に使用できるようにするすべてのスタンドアロン・システムの定義をエクスポートします。(『定義のエクスポート/インポート』参照)。ユーザーとセキュリティー・ロールはエクスポートしないでください。ある定義についてインポートするか不明である場合は、それを別個のエクスポート操作でエクスポートしておけば、その定義を中央マネージャーにインポートするかどうかを後で決定できます。 一元管理に登録すると、スタンドアロン・ユニットにあった古い定義はいずれも使用できなくなります。
   - スタンドアロン・ユニット上で、監査プロセスの結果の PDF バージョンを作成し、適切なロケーションに保管しておきます。 一元管理下では、一元管理下で作成された監査結果のみが使用可能です。
   - すべてのユーザーに対して、スタンドアロン・ユニット上で、カスタム・レポートを含むすべてのポートレットを削除するように、そして一元管理への変換が完了するまで新規のレポートは作成しないように指示します。
   - 中央マネージャー上で、スタンドアロン・ユニットにあったすべてのユーザーを手動で追加します。
   - スタンドアロン・ユニット上で、admin ユーザーを除くすべてのユーザー定義を削除します (admin は削除できません)。
   - スタンドアロン・ユニットを一元管理に登録します。『一元管理へのユニットの登録』を参照してください。
   - 中央マネージャー上で、スタンドアロン・システムからエクスポートしたすべての定義をインポートします。組み込んだ項目への参照 (例えば、アラート通知の受信者など) が正しいかどうか確認します。セキュリティー・ロールを、必要に応じてすべてのインポート定義に再度割り当てます。

- レイアウトに表示するカスタム・レポート用にポートレットを再生成するには、「レポート・ビルダー」アプリケーションを使用する必要があることを管理対象ユニットのユーザーに知らせます。

スタンドアロン CAS コレクターの管理対象へのマイグレーション

アクティブ・インスタンスを持つ CAS コレクターを管理対象にマイグレーションするときには、以下の手順を使用します。

1. CAS ホスト定義をスタンドアロン・コレクターからエクスポートします。
2. スタンドアロン・コレクターを管理対象にします。
3. 管理対象になったコレクターの GUI から CAS ホストを再始動します。
4. CAS ホスト定義をマネージャーにインポートします。
5. 管理対象コレクターの GUI から CAS ホストを再度再始動します。

これらのステップを実行すると、CAS コレクターはスタンドアロンだったときと同じインスタンスを保持し、同じファイルをモニターします。

注: スタンドアロンだったときに収集された CAS データは削除されます。ファイルに変更がなければ、収集される CAS データはありません。
親トピック: 一元管理の実装

一元管理機能の使用

一元管理機能を使用すると、ポータル・ユーザー・アカウントの同期化、管理対象ユニットのモニター、および管理対象ユニットへのセキュリティー・ポリシーのインストールを行うことができます。

- 「適用状態」ビュー
  「適用状態」のビューでは、Guardium 環境全体に関する情報が収集され、強力で簡単に取り込まれるグラフィカル・ビューに表示されます。
- エンタープライズ・ロード・バランシング
  エンタープライズ・ロード・バランサーは、管理対象ユニットをシステムの負荷と利用可能性に基づいて動的に S-TAP エージェントに割り当てます。
- デプロイメント・インベントリー
  「インベントリー (inventory)」ビューには、すべてのデータベース・サーバーとインストール済みの S-TAP クライアントまたは GIM クライアントの一元管理ビューが表示されます。
- 「リソース・デプロイメント」ビュー
  「リソース・デプロイメント」ビューには、すべてのデータベース・サーバーと、関連するコレクター、アグリゲーター、および中央マネージャーの一元管理ビューが表示されます。
- 管理対象ユニット・グループの作成
  管理対象ユニットをグループに編成してから、それらのグループにアクションを実行します。
- 管理対象ユニットのモニター
  一元管理を使用して管理対象ユニットをモニターします。
- 管理対象ユニットへのセキュリティー・ポリシーのインストール
  管理対象ユニットにセキュリティー・ポリシーをインストールします。
- 一元化パッチ管理
  パッチのインストール、状況、および履歴を表示可能にし、制御します。
- 構成プロファイルの処理
  構成プロファイルにより、中央マネージャーから構成設定およびスケジューリング設定を定義して、中央マネージャー自体の構成を変更することなく、それらの設定を管理対象ユニット・グループに配布することができます。
- 構成の配布
  構成、ならびにそのスケジュールは、全体またはその一部を、中央マネージャーと管理対象ユニットの間で配布することができます。
- 認証構成の配布
  各アプライアンスで別個に認証を構成する代わりに、中央マネージャー上で一元管理認証 (認証の構成) を 1 回構成し、それからすべての管理対象ユニットに配布することができます。このようにすると、情報の入力を 1 回行うことで、その情報を一部またはすべてのユニットに適用することができます。一部のユニットで異なるタイプの認証を使用することもできます。
- 予備の中央マネージャー
  予備の中央マネージャーまたはバックアップ中央マネージャー (CM) を使用して、プライマリー CM が使用不可になった場合に備えてセカンダリー CM またはバックアップ CM を構成します。

親トピック: 一元管理

「適用状態」ビュー

「適用状態」のビューでは、Guardium 環境全体に関する情報が収集され、強力で簡単に取り込まれるグラフィカル・ビューに表示されます。

「適用状態」のビューを使用すると、システムの使用傾向を調査すること、および不安定なシステムやダウンしているシステムを素早く特定することができます。これらのビューにより、対応時間が削減され、Guardium デプロイメント内の問題によるリスクが軽減されます。「適用状態」のビューは、異なる複数の情報ソースを固有の関連ビューに統合することで連携して動作するように設計されています。

「適用状態トポロジー」ビューおよび「適用状態表」ビュー

> 「適用状態トポロジー」ビューおよび「適用状態表」ビューには、環境内のシステム間のデータ・フロー関係が表示されます。これらのビューにより容易に、問題のあるシステムを識別し、根本的な問題を調査できます。
>
> トポロジー・ビューにアクセスするには、「管理」 > 「システム・ビュー」 > 「適用状態トポロジー」にナビゲートします。表ビューにアクセスするには、「管理」 > 「システム・ビュー」 > 「適用状態表」にナビゲートします。

適用状態ダッシュボード

> 「適用状態ダッシュボード」では、Guardium デプロイメントで検出された問題の概要を一目で確認できます。このダッシュボードは、問題が特定された個々のシステムを調査する前に、正常性データでパターンと傾向を特定するのに特に便利です。

このダッシュボードにアクセスするには、「管理」 > 「システム・ビュー」 > 「適用状態ダッシュボード」にナビゲートします。

以下の表に、「適用状態」の各ビューで使用できるデータのタイプをまとめます。

表 1. 「適用状態」のビューの概要

	ダッシュボード	トポロジー	表
ユニット使用状況	✔	✔	✔
相関アラート	✔		
自己モニター	✔		
システム要件	✔		
統合		✔	✔
検査エンジン (S-TAP 検査データ)		✔	
接続		✔	✔
S-TAP 接続		✔	

重要: 「適用状態」のビューには、Guardium 環境全体から収集されたデータが表示されます。これらのビューは、中央マネージャーからのみ使用できます。

- 「適用状態」のビューのための中央マネージャーの構成
  「適用状態」のビューを使用するには、ユニット使用状況データの収集を有効にし、相関アラートを構成し、環境のデータ・インポートおよびエクスポートを構成します。
- 「適用状態トポロジー」ビューおよび「適用状態表」ビュー
  「適用状態トポロジー」ビューおよび「適用状態表」ビューで、Guardium 環境の構成とそのデータがどのように表示されるかについて詳しく説明します。
- 適用状態ダッシュボード
  Guardium デプロイメント全体からのデータが適用状態ダッシュボードにどのように表示されるかについて詳しく説明します。
- シナリオ:「適用状態トポロジー」ビューを使用した過負荷システムのトラブルシューティング
  このトピックでは、「適用状態トポロジー」ビューを使用して、環境内の過負荷システムを特定し、修正する方法について説明します。

親トピック: 一元管理機能の使用

「適用状態」のビューのための中央マネージャーの構成

「適用状態」のビューを使用するには、ユニット使用状況データの収集を有効にし、相関アラートを構成し、環境のデータ・インポートおよびエクスポートを構成します。

このタスクについて

中央マネージャーの「適用状態」のビューには、Guardium 環境全体からのデータが表示されます。デプロイメント全体に関するデータを表示できるようにするために、ユニット使用状況データの収集、相関アラートの構成が必要であり、またデータのインポート、エクスポート、および S-TAP 検査を正しく構成する必要があります。「適用状態」のビューに表示されるデータの概要については、「適用状態」ビューを参照してください。

デプロイメントは、「適用状態」のビューをサポートするように既に構成されている場合が多いです。いずれかの「適用状態」のビューで以下のいずれかの問題を見つけた場合、この手順で説明されている構成ステップを確認してください。

- CM バッファー使用状況レポートはスケジュールに入っていません
- ユニット使用状況レポートはスケジュールに入っていません
- エクスポートはスケジュールに入っていません
- インポートはスケジュールに入っていません
- 問題は見つかりませんでした
- 状況不明

手順

1. 中央マネージャーからのユニット使用状況データの収集と処理を構成します。 詳しくは、『ユニット使用状況データ処理の構成』を参照してください。
2. 「適用状態ダッシュボード」への相関アラートの組み込みを有効にします。
   a. 「保護」 > 「データベース侵入保護 (Database Intrusion Protection)」 > 「アラート・ビルダー」を開きます。
   b. 既存のアラートを選択し、  アイコンをクリックします。または、  アイコンをクリックして新規アラートを作成します。
   c. アラートの「カテゴリー」を指定します。 カテゴリーが指定されていないアラートは「カテゴリーなし」として表示されます。
   d. ダッシュボードにアラートを組み込むために、「適用状態ダッシュボードに表示」チェック・ボックスを選択します。
   重要: 適用状態ダッシュボードにアラートを組み込むために、「重大度」を「低」、「中」、または「高」に設定する必要があります。
   アラートの定義について詳しくは、アラートの作成を参照してください。
3. 中央マネージャーからのデータのインポートとエクスポートを構成します。 詳しくは、統合を参照してください。
   ヒント: 「構成プロファイルの配布」ツールを使用して、Guardium デプロイメントのデータ・インポートおよびエクスポートを構成するプロセスを簡素化します。詳しくは、構成プロファイルの処理を参照してください。
4. サポートされるすべての S-TAP のために S-TAP 検査を構成します。 詳しくは、Windows での検査エンジンの検査および UNIX での検査エンジンの検査を参照してください。

タスクの結果

構成手順を完了し、データを更新できるようにした後、「適用状態トポロジー」および「適用状態表」ビューに、  状況が主に表示されます (ただし、システムに既存の正常性の問題がある場合は除きます)。「適用状態ダッシュボード」には、既存のユニット使用状況の問題が含まれ、新しい相関アラートの状態の表示が開始されます。

ユニット使用状況、またはデータのインポートおよびエクスポートのスケジュールについてアラートがある場合、最大 1 時間待機し、「適用状態」ビューを新しい情報で更新できるようにします。新しい相関アラート・データを使用できるかどうかは、アラートに対して指定されている通知頻度によって決まります。

親トピック: 「適用状態」ビュー
関連概念:
統合
関連タスク:
ユニット使用状況データ処理の構成
構成プロファイルの処理
関連情報:
相関アラート

「適用状態トポロジー」ビューおよび「適用状態表」ビュー

「適用状態トポロジー」ビューおよび「適用状態表」ビューで、Guardium 環境の構成とそのデータがどのように表示されるかについて詳しく説明します。

「適用状態トポロジー」ビューは、中央マネージャーからアクセス可能であり、その中央マネージャーに接続されている Guardium 環境全体の概要を可視化します。「適用状態トポロジー」ビューには、環境内のノード間の関係の表示に加えて、接続されているすべてのアグリゲーター、コレクター、および S-TAPs に関する適用状態の情報も示されます。環境で検出された正常性の問題に素早く対処できるようにするために、「適用状態トポロジー」ビューから複数の調査および解決アクションを直接使用できます。

デフォルトの「適用状態トポロジー」ビューは、アグリゲーターと管理対象ユニット間のデータのインポートとエクスポートの関係を示すデータ・フロー・ビューです。「適用状態トポロジー」ビューは、「管理」 > 「システム・ビュー」 > 「適用状態トポロジー」で開きます。

また、「管理」 > 「システム・ビュー」 > 「適用状態表」で、適用状態データのソート可能な表ビューも使用できます。

データ可用性

いくつかの要因が、システム・データの可用性、およびそのデータの「適用状態トポロジー」ビューと「適用状態表」ビューでの表示に影響します。「適用状態」ビューを使用するようにシステムを構成する方法については、「適用状態」のビューのための中央マネージャーの構成を参照してください。

バックアップ CM は、その接続状況のみ表示します。

データのタイプ

正しく構成されている場合、「適用状態トポロジー」ビューおよび「適用状態表」ビューには、複数の異なるソースから収集されたデータが表示されます。表示されるデータのタイプはユニット・タイプによって決まります。これについて、以下のセクションで概説します。

接続

接続カテゴリーは、Guardium 環境内のシステムが通信できるかどうかを示します。

- 適用対象: 中央マネージャー、アグリゲーター、コレクター、および S-TAPs
- 例:「ユニットは応答しません」、「S-TAP は応答しません」、「不正確な S-TAP構成」など

ユニット使用状況

ユニット使用状況カテゴリーは、Guardium システムがどの程度ロードされているかに関する情報を示します。

- 適用対象: 中央マネージャー、アグリゲーター、およびコレクター
- 例:「CPU ロード」、「空きバッファー・スペース」、「MySQL ディスク使用状況」など
- 詳しくは、ユニット使用状況レベルを参照してください。

統合

統合カテゴリーは、Guardium システム間のデータのインポートおよびエクスポートのフローに関する情報を示します。

- 適用対象: 中央マネージャー (アグリゲーターとして構成されている場合)、アグリゲーター、およびコレクター
- 例:「インポートに失敗しました」、「エクスポートに失敗しました」、「エクスポートはスケジュールに入っていません」など
- 詳細については、事前定義管理レポートおよび 統合 を参照してください。

検査エンジン

検査エンジン・カテゴリーは、S-TAP 検査情報を提供します。

- 適用対象: S-TAPs
- 例:「S-TAP 検査が失敗しました」など
- 詳細については、Configuring the S-TAP verification schedule および Viewing S-TAP verification results を参照してください。

アイコンをクリックすると、「設定のカスタマイズ」ダイアログが開きます。ここで、「適用状態トポロジー」ビューおよび「適用状態表」ビューに表示するデータのタイプを定義できます。

データ待ち時間

事前設定およびユーザー定義の複数のスケジュールによって、「適用状態トポロジー」ビューに表示されるデータの待ち時間が決まります。これらのスケジュールについて、以下の表にまとめます。

表 1. 「適用状態トポロジー」ビューのデータ待ち時間

正常性カテゴリー	ノード・タイプ	待ち時間

正常性カテゴリー	ノード・タイプ	待ち時間
接続	アグリゲーターまたはコレクター	15 分未満
接続	S-TAP	エンタープライズ・ロード・バランシングが使用可能の場合、15 分未満 エンタープライズ・ロード・バランシングが使用不可の場合、1 時間未満
統合	中央マネージャー、アグリゲーター、またはコレクター	1 時間未満
検査	S-TAP	1 時間未満
ユニット使用状況	中央マネージャー、アグリゲーター、またはコレクター	推奨構成に基づき 1 から 2 時間。詳しくは、ユニット使用状況データ処理の構成を参照してください。

特定の環境変更および構成変更について、以下の待ち時間を監視します。

- 新しく登録されたアグリゲーターまたはコレクターは、15 分以内に「適用状態」ビューで使用できるようになります。
- コレクターからデータ・エクスポート・スケジュールまたはデータ・エクスポート構成を削除した場合、2 時間以内に「適用状態」ビューに反映されます。

データ表示

正常性状況

「適用状態トポロジー」ビューには、Guardium システムに関する 3 つのカテゴリーの正常性情報 (「接続」、「ユニット使用状況」、および「統合」) が表示されます。これらのカテゴリーのメトリックには、次のいずれかの正常性状況が割り当てられます:「状況不明」(重大度最小)、「正常性の問題はありません」、「重大度低 (low severity)」、「重大度中」、「重大度高」(重大度最大)。全体状況は、表示される正常性カテゴリーに含まれる個々のメトリックの最も重大な状況によって決まります。「設定のカスタマイズ」ダイアログを使用して除外したデータは、システムの全体状況を決定するのに使用されません。

例えば、「ユニット使用状況」カテゴリーの「再始動」メトリックに「重大度高」状況が割り当てられているが、別のカテゴリーに正常性の問題が存在しない場合、そのシステムの「全体状況」は「重大度高」になります。この動作により、最も重大な状態をシステムの全体状況としていつでも一目で確認できます。

「管理」 > 「システム・ビュー」 > 「適用状態トポロジー」ビューには、重大度が低、中、または高の問題が少なくとも 1 つ検出された場合に限り、使用可能な正常性カテゴリーの詳細な状況が表示されます。

「管理」 > 「システム・ビュー」 > 「適用状態表」ビューには、使用可能な正常性カテゴリーの詳細な状況が常に表示されます。

正常性状況のロールアップ

「適用状態トポロジー」ビューには、Guardium 環境全体の正常性情報を効率よく表示するために、正常性状況のロールアップ方針が実装されています。この方針を使用すると、子ノードは親ノードの下に縮小され、子の正常性状況は親にロールアップされます。ロールアップされた状況は、親ノードに付加された小さいアイコンとして表されます。

重要: 状況を親コレクターにロールアップする S-TAP ノードに対してのみ、正常性状況のロールアップがサポートされます。

例えば、  は、正常性の問題がないコレクターを示していますが、小さい赤色の円は、そのコレクターに関連する 1 つ以上の S-TAPs に、重大度が高の問題があることを示します。そのコレクターをクリックすると、ノードが展開され、関連した S-TAPs とそれらの正常性状況が表示されます。例えば、

は、コレクターに関連する 4 つの S-TAPs のうち、2 つの S-TAPs に重大度が高の正常性の問題があり、別の 2 つの S-TAPs に重大度が低の正常性の問題があることを示します。

子ノードが縮小されているときに、子ノードから親ノードにロールアップされるのは、最も重大な状況のみです。上記の例では、親ノードは小さい赤色の円を示しています。これは、その 1 つ以上の子に重大度が高の問題があるためです。ただし、1 つ以上の子ノードに重大度が低の問題がある一方で、他のすべての子ノードには正常性の問題がない場合、親ノードには小さい黄色い円が表示されます。

デプロイメントの表示

「適用状態トポロジー」ビューには、予期しないデプロイメント構成が表示されることがあります。これらの構成シナリオのいくつかについて、以下のセクションで説明します。

Guardium V10.1.3 より前の管理対象ユニット

Guardium V10.1.3 より前の管理対象ユニットがバージョン V10.1.3 以後の中央マネージャーに接続している場合、これらの管理対象ユニットでは、不正確または不整合なユニット使用状況データが表示されることがあります。この問題を解決するには、中央マネージャーの CLI にログインし、管理対象ユニットごとに次のコマンドを実行します。

```
grdapi change_tracker_reset host=[managed unit host name or IP address]
```

ベスト・プラクティス: 管理対象環境では、すべてのユニットが同じ Guardium バージョン・レベルで作動するようにお勧めします。

Guardium V10.1 より前の管理対象ユニット

Guardium V10.1 より前の管理対象ユニットを「適用状態トポロジー」ページまたは「適用状態表」から表示した場合、これらの管理対象ユニットでは、「統合」正常性セクションで「状況不明」が表示されます。

ベスト・プラクティス: 管理対象環境では、すべてのユニットが同じ Guardium バージョン・レベルで作動するようにお勧めします。

サポート対象外の S-TAP

「適用状態トポロジー」ビューには、S-TAP 検査用に構成されている S-TAP、およびエンタープライズ・ロード・バランシングに参加している S-TAP が表示されます。S-TAP を、S-TAP 検査用に構成できない場合、またはエンタープライズ・ロード・バランシングに参加するように構成できない場合、S-TAP は表示されません。

S-TAP ロード・バランシング

S-TAP ロード・バランシングが participate_in_load_balancing パラメーターを使用して構成され、S-TAP が、複数のコレクター間でトラフィックのバランスを取るように構成されている場合、「適用状態トポロジー」ビューには、その S-TAP が各コレクターの子ノードとして表示されます。例えば、*S-TAP 1* が *コレクター A* と *コレクター B* でロード・バランシングされる場合、*コレクター A* と *コレクター B* の両方が、「適用状態トポロジー」ビューで *S-TAP 1* を子として表示します。

非管理対象ユニット

コレクターが中央マネージャーまたは中央マネージャーとして構成されているアグリゲーターにデータをエクスポートする一方で、そのコレクターがその一元管理クラスターの管理対象ユニットとして指定されていない場合、「適用状態トポロジー」ビューでは、コレクターの「全体状況」が「正常性状況が利用できません」として表示されます。コレクターが中央マネージャーの管理対象ユニットとして指定されている場合を除いて、コレクターに関する追加情報は「適用状態トポロジー」ビューでは利用できません。

1 次ホストおよび 2 次ホストにデータをエクスポートするコレクター

1 次ホストと 2 次ホストの両方にデータをエクスポートするようにコレクターが構成されている場合、「適用状態トポロジー」ビューでは 1 次ホストのみ使用されます。

親トピック: 「適用状態」ビュー
関連タスク:
「適用状態」のビューのための中央マネージャーの構成

適用状態ダッシュボード

Guardium デプロイメント全体からのデータが適用状態ダッシュボードにどのように表示されるかについて詳しく説明します。

データ可用性

複数の要因が正常性データの可用性および待ち時間に影響し、またそのデータが適用状態ダッシュボードにどのように表示されるかにも影響します。以下の表に、ダッシュボードに含まれるデータ、トリガー基準、データ待ち時間、およびパージについての情報をまとめます。

表 1. 適用状態ダッシュボードのデータの概要

データ・ソース	情報タイプ	トリガー基準	データ待ち時間	データ・パージ間隔
システム・リソース	システム構成 (CPU コア、システム・メモリー、/var ディスク容量など)	システムが最小要件を満たしていない	ユーザー・インターフェース・サーバーが始動または再始動すると常に更新される	適用外
ユニット使用状況	ユニット使用状況データ (スニファー再始動、MySQL ディスク使用状況、CPU 負荷など)	値がユニット使用状況のしきい値を超える	推奨構成に基づき 1 から 2 時間以内に更新される。詳しくは、ユニット使用状況データ処理の構成を参照してください。	ユニット使用状況データは 60 日後にパージされる スニファーのバッファー使用状況データは 14 日後にパージされる
システム自己モニター	MySQL ディスク使用状況およびシステム・ディスク使用状況	使用量がデフォルトのしきい値以上になる (重大度が「高」では 75%、重大度が「クリティカル」では 90%)	5 から 10 分ごとに更新される 重大度が「高」では、15 分の期間内に同じイベントが複数回発生した場合、最新の発生を反映するようにタイム・スタンプが更新されます。15 分の間隔後に同じイベントが発生した場合、最新のタイム・スタンプを使用して新しいエントリーが作成されます。 「クリティカル」な問題では、イベントが発生するたびに固有のタイム・スタンプを使用してエントリーが作成されます。	重大度が「高」の問題は 7 日後にパージされる 「クリティカル」な問題はパージされない
相関アラート	トリガーされた相関アラート	アラートしきい値に到達する	アラート通知の頻度に基づき更新される。詳しくは、相関アラートを参照してください。	データは 7 日後にパージされる

重要:

- Guardium V10.1.2 以降を実行するシステムからのデータのみが適用状態ダッシュボードに表示されます。
- システムのホスト名を変更すると、元のホスト名に関連付けられている既存のデータは適用状態ダッシュボードに表示されなくなります。
- フェイルオーバー・シナリオ中にプライマリー中央マネージャーがバックアップ中央マネージャーにデータを転送しているとき、最大 30 分、適用状態ダッシュボードでデータが使用不可になります。

データ表示

適用状態ダッシュボードでは、各種タイルまたは小さなウィンドウに類似するコンテナーを通じてデータがフォーマット設定され、表示されます。以下の表に、各ダッシュボード・タイルに表示されるデータをまとめます。

表 2. 適用状態ダッシュボードのタイルの概要

	タイル名

データ・ソース	リソース要件	ユニット使用状況の問題	ユニット使用状況のタイム・チャート	アラート (カテゴリー、名前、重大度、またはシステムによる)	イベント	重大度高	クリティカル
システム・リソース	✔					✔	
ユニット使用状況		✔	✔		✔	✔	
システム自己モニター					✔	✔ (使用量がしきい値である 75% 以上になった場合)	✔ (使用量がしきい値である 90% 以上になった場合)
相関アラート				✔	✔	✔	
次のタイルはデフォルトで表示されます: 「*名前別のアラート*」、「*クリティカルな問題*」、「*イベント・タイムライン*」、「*重大度の高い問題*」、および「*ユニット使用状況の問題*」。							

ダッシュボード・フィルター

ダッシュボード・フィルターを使用すると、Guardium システム、問題の重大度、および期間に基づきデータを素早くフィルタリングできます。フィルター設定は、特に注記がない限り、ダッシュボードのどの部分に表示されるデータにも影響します。

Guardium システム・フィルターを使用すると、ユニット・タイプにより、または「管理」 > 「一元管理」 > 「管理対象ユニット・グループ」で定義されているグループにより、ダッシュボードをフィルタリングできます。

デフォルトでは、ダッシュボードには、発生したすべての問題 (「低」、「中」、「高」、および「クリティカル」) が表示されます。「重大度」メニューを使用すると、ダッシュボードで重大度によりデータをフィルタリングできます。「高」を選択すると、ダッシュボード全体がフィルタリングされ、重大度が高の問題のみが表示されます。「クリティカル」を選択すると、ダッシュボード全体がフィルタリングされ、クリティカルな問題のみが表示されます。「高」と「クリティカル」の両方の問題を選択することもできます。この場合、これらよりも重大度が低いすべてのデータがフィルターにより除外されます。
注意:

- 未処理または未解決のクリティカルな問題は、「重大度」フィルターの設定に関係なく、ダッシュボードに表示されます。
- 「*ユニット使用状況の問題*」タイルでは、ダッシュボードの「重大度」フィルターは、ユニット使用状況の重大度全体に基づきます。ユニット使用状況の重大度が割り当てられる方法について詳しくは、ユニット使用状況の問題を参照してください。

時間フィルターにより、ダッシュボードに表示されるデータの範囲が決まります。デフォルトの設定では、1 時間から 3 週間までの期間を使用できますが、カスタム期間もサポートされています。時間フィルターは、クリティカルな問題には適用されません。クリティカルな問題は、時間フィルターの設定に関係なく常に表示されます。

「グラフの追加」メニューを使用すると、ダッシュボードにタイルを追加することや、以前削除したデフォルトのタイルを元の場所に戻すことができます。

ダッシュボード・サマリー (Dashboard summary)

「ダッシュボード・サマリー (dashboard summary)」には、Guardium デプロイメントで検出された正常性の問題の全体数が表示されます。「問題のあるコレクター」および「問題のあるアグリゲーター」の数は、正常性の問題が検出されたシステム (コレクターおよびアグリゲーター) の数を示します。「クリティカル」および「高」の数は、ダッシュボードに含まれるすべてのシステムで検出された問題の数を示します。
注:

- ダッシュボードに対してタイルを追加または削除しても、「クリティカル」および「高」の数は影響を受けません。
- 「ダッシュボード・サマリー (dashboard summary)」バー上の数は、ダッシュボード・フィルターの設定を反映します。

カテゴリー、名前、重大度、またはシステムによるアラート

適用状態ダッシュボードは、Guardium 相関アラート (「*カテゴリー別のアラート*」、「*名前別のアラート*」、「*重大度別のアラート*」、および「*システム別のアラート*」) に基づく複数のタイルをサポートしています。ダッシュボードに相関アラート・タイルを追加するには、「グラフの追加」メニューを使用します。

相関アラートは、適用状態ダッシュボードに含めるかどうか明示的に構成する必要があります。ダッシュボードのアラートの構成については、「適用状態」のビューのための中央マネージャーの構成を参照してください。

リソース要件

「*リソース要件*」タイルは、Guardium デプロイメント内のシステムが、CPU、メモリー、および /var ディスク容量に関する最小ハードウェア要件を満たしているかどうかを示します。最小要件を満たしていないシステム・リソースは、重大度が高の問題として示され、「*リソース要件*」タイルと「*重大度の高い問題*」タイルの両方に表示されます。

タイルの詳細ビューの「正常なシステムを含める」チェック・ボックスを使用すると、ダッシュボードのフィルター・バーで示されているシステムおよび時間フレームに対して使用できるすべてのデータを含めることができます。「正常なシステムを含める」チェック・ボックスは、ダッシュボード・フィルター全体の「重大度」設定よりも優先され、使用可能なすべてのデータが組み込まれます。正常性の問題が検出されなかったシステムは、デフォルトでは除外されます。

Guardium デプロイメントのリソース要件のうち満たされているものと満たされていないものをすべて表示する表は、「管理」 > 「一元管理」 > 「システム・リソース」でも表示できます。
注:

- システム・リソースの問題は、特定のタイム・スタンプに関連付けられていないため、「*イベント*」タイムラインには表示されません。

ユニット使用状況の問題

「*ユニット使用状況の問題*」タイルには、ユニット使用状況のしきい値に基づいて問題が表示されます。このタイルに表示される問題は、各しきい値を超えた個々のメトリックを表します。指定の期間内に個々のシステムに対して使用可能なすべてのメトリックで検出された問題のうち、最も重大度の高い問題に基づき、全体的な重大度が割り当てられます。ユニット使用状況のしきい値について詳しくは、ユニット使用状況レベルを参照してください。

「*ユニット使用状況の問題*」タイルの詳細ビューには、「期間の開始」時刻と「タイム・スタンプ」の両方が含まれます。

- 「期間の開始」時刻は、「*CM バッファー使用状況モニター*」のデータが時間単位の期間 (例えば、13:00、12:00、および 11:00 に開始される期間) にロールアップされたことを示します。
- 「タイム・スタンプ」は、ユニット使用状況レベルのデータが適用状態ダッシュボードにいつ追加されたかを示します。この追加は、ユニット使用状況レベルのスケジュールに基づき、または「*今すぐ 1 回実行*」を使用することで行われます。

詳しくは、ユニット使用状況データ処理の構成を参照してください。
ユニット使用状況データが適用状態ダッシュボードに初めて追加された時点では、すべてのユニット使用状況データで、「*タイム・スタンプ*」が同じになりますが、「*期間の開始*」時刻は異なります。時間が経過すると、タイム・スタンプはユニット使用状況レベルのスケジュールに基づいた間隔を置いて表示されます。例えば、ユニット使用状況レベルのデータが、正時の 1 時間 40 分後に収集される場合、「*期間の開始*」時刻と「*タイム・スタンプ*」の値は以下のようになります。

表 3. ユニット使用状況の「期間の開始」時刻と「タイム・スタンプ」値の例

期間の開始	タイム・スタンプ
13:00	14:40
12:00	13:40
11:00	12:40

タイルの詳細ビューの「正常なシステムを含める」チェック・ボックスを使用すると、ダッシュボードのフィルター・バーで示されているシステムおよび時間フレームに対して使用できるすべてのデータを含めることができます。「正常なシステムを含める」チェック・ボックスは、ダッシュボード・フィルター全体の「重大度」設定よりも優先され、使用可能なすべてのデータが組み込まれます。正常性の問題が検出されなかったシステムは、デフォルトでは除外されます。

ユニット使用状況のタイム・チャート

「*ユニット使用状況のタイム・チャート*」を使用すると、時間の経過に伴うユニット使用状況データの傾向を監視できます。「*ユニット使用状況のタイム・チャート*」は、単一の Guardium システムに対して複数のユニット使用状況メトリックを表示するように構成したり、複数の Guardium システムに対して単一のユニット使用状況メトリックを表示するように構成したりできます。

「*ユニット使用状況のタイム・チャート*」は、以下の基準に基づいて構造化されます。

- X 軸は、「*期間の開始*」時刻を表します。
- 複数のメトリックがグラフ化され、メトリックの値が同じ範囲内にある場合、Y 軸が 1 つ描画されます。例えば、「*MySQL ディスク使用状況*」と「*/var ディスク使用状況*」はどちらもパーセンテージで表され、同じ Y 軸を使用して描画されます。
- 複数のメトリックがグラフ化され、メトリックの値が類似していない場合、Y 軸が 2 つ描画されます。例えば、「*MySQL ディスク使用状況*」はパーセンテージで表され、「*未解析ログ要求*」は整数で表されるため、2 つの Y 軸 (一方はパーセンテージを表示し、もう一方は整数を表示する) が描画されます。
- メトリックの値が Y 軸の範囲外になった場合、その値はグラフの下部に表示されます。この動作により、類似する単位を使用して異なるメトリックを表す一方で、値が大幅に異なるシナリオ (例えば、千単位の範囲の整数と 100 万単位の範囲の整数) にも対応できます。
  ヒント: 値が大幅に異なる範囲にある場合は、複数のタイム・チャートを作成してください。

注: ダッシュボードのフィルター・バーで指定されている時間フレーム内に、システムに関するユニット使用状況データが存在しない場合、そのシステムは「タイム・チャートの設定」 > 「ホスト名」メニューには含まれません。
親トピック: 「適用状態」ビュー
関連タスク:
「適用状態」のビューのための中央マネージャーの構成
ユニット使用状況データ処理の構成

シナリオ:「適用状態トポロジー」ビューを使用した過負荷システムのトラブルシューティング

このトピックでは、「適用状態トポロジー」ビューを使用して、環境内の過負荷システムを特定し、修正する方法について説明します。

このタスクについて

このシナリオでは、「適用状態トポロジー」ビューから正常性の問題を特定し、根本原因を評価し、さらにその評価と追加データを関連付けた後で、問題を解決し、修正を検証します。ここで説明する例では、過負荷のコレクターを取り上げますが、このプロセスは他のケースにも適用できます。

手順

1. 中央マネージャーで、「管理」 > 「システム・ビュー」 > 「適用状態トポロジー」にナビゲートします。
2. 適用状態トポロジーを確認し、環境内のシステムの全体的な正常性を評価します。 大まかには、アイコンにより、正常なシステムが示され、一方 アイコンおよび アイコンにより、正常性に問題があるシステムが示されます。
3. 状況アイコンまたは 状況アイコンがあるシステムを見つけた場合、そのノードをクリックすると、追加の正常性情報を含むオーバーレイが表示されます。
4. ノードのオーバーレイに表示される情報を使用して、正常性の問題の診断を開始します。例えば、「変数ディスク使用状況」、「再始動」、「アナライザー・キュー」、および「ロガー・キュー」の重大度状況が高または中のコレクターは過負荷となっています。
5. 「適用状態トポロジー」ビューから正常性の問題を最初に評価した後、見つけた内容と追加データの関連付けを試行します。例えば、システムが過負荷となっている疑いがある場合、そのシステムのトラフィックのモニターを開始します。

6. 正常性の根本問題の診断が確定した場合、修正アクションを実行します。この過負荷システムの例では、エンタープライズ・ロード・バランシングを確立することや、S-TAPs を別のコレクターに再割り当てすることができます。 エンタープライズ・ロード・バランシングが既に構成され、使用されている場合、この一連の症状は通常発生しません。
7. 修正アクションを実行した後、ユニット使用状況および中央マネージャーのバッファー使用モニター・データの次回更新に続いて、「適用状態トポロジー」ビューでノードの状況が更新されます。この更新間隔は、ユニット使用状況データの処理のスケジュールによって決まります。

親トピック: 「適用状態」ビュー
関連情報:
S-TAP ユーザーズ・ガイド

エンタープライズ・ロード・バランシング

エンタープライズ・ロード・バランサーは、管理対象ユニットをシステムの負荷と利用可能性に基づいて動的に S-TAP エージェントに割り当てます。

概要

ロード・バランシングを行うと、新しい S-TAPs がインストールされた場合や、管理対象ユニットが使用不可である場合のフェイルオーバー時に、管理対象ユニットが S-TAP エージェントに対して自動的に割り当てられます。 ロード・バランシング・アプリケーションは、負荷が低い管理対象ユニットに S-TAP エージェントを再配置することにより、負荷の高い管理対象ユニットやビジー状態の管理対象ユニットの負荷を動的に分散します。

エンタープライズ・ロード・バランシング・アプリケーションにより、いくつかのタスクが自動化されます。

- 管理対象ユニットを S-TAP エージェントに割り当てる前に、それらの管理対象ユニットの負荷を手動で評価する必要がなくなります。
- インストール後の S-TAP の構成作業の一部として、フェイルオーバー用の管理対象ユニットを定義する必要がなくなります。これは、ロード・バランサーにより、フェイルオーバーのシナリオが動的に管理されるためです。
- 負荷の高い管理対象ユニットから負荷の低い管理対象ユニットに S-TAP エージェントを手動で再配置する必要がなくなります。

重要: エンタープライズ・ロード・バランシング・アプリケーションを使用すると、Guardium システムにより、S-TAP エージェントに対する管理対象ユニットの割り当てが制御されます。 これは、動的な自動プロセスです。使用可能な管理対象ユニットの相対的な負荷に基づいて S-TAPs の関連付けが変化します。ロード・バランシングのすべてのアクティビティーを確認するには、「ロード・バランサー・イベント」レポートを使用してください。
注: エンタープライズ・ロード・バランシングを使用するように S-TAP を構成する場合、F5 ベースのロード・バランシングは使用できません。

前提条件

エンタープライズ・ロード・バランサーは中央マネージャーまたは管理対象ユニット上で稼働し、ポート 8443 を listen し、トランスポート層セキュリティー (TLS) を使用します。新しいファイアウォールや追加のシステム設定は必要ありません。S-TAP は V10.1 以上でなければなりません。

ロード・バランシングは、Guardium システム上でデフォルトで無効になっています。S-TAPs を有効にしてロード・バランシングを行う方法については、Windows の一般パラメーターおよび UNIX の一般パラメーターを参照してください。

ロード・バランシングの仕組み

エンタープライズ・ロード・バランシング・アプリケーションは、すべての管理対象ユニットから最新の負荷情報を収集して保守することによって機能します。

このアプリケーションは、管理対象ユニットの負荷情報を使用して、ロード・マップを作成します。このロード・マップにより、ロード・バランシングを指示するデータと、管理対象ユニットの割り当てアクティビティーが指定されます。 GuardAPI コマンドの grdapi get_load_balancer_load_map を使用すると、現在のロード・マップをいつでも表示することができます。

負荷情報は、LOAD_BALANCER_ENABLED=1 パラメーターが構成されているオンライン状態の管理対象ユニットからのみ収集されます。LOAD_BALANCER_ENABLED=0 を設定すると、ロード・バランシングが無効になり、ロード・バランシング・アクティビティーの実行中に管理対象ユニットが S-TAP エージェントに動的に割り当てられることがなくなります。

特定の管理対象ユニットから負荷情報を収集できなかった場合は、「ロード・バランサー・イベント」レポートにエラーとして記録されますが、全体的な負荷情報収集プロセスとロード・バランシング・プロセスには影響しません。ただし、負荷情報を収集できなかった管理対象ユニットは、ロード・バランシング・プロセスの対象から除外されます。

- エンタープライズ・ロード・バランシング用に S-TAP を管理対象ユニットに関連付ける
  ここでは、S-TAP グループを作成して管理対象ユニットのグループに関連付けることにより、エンタープライズ・ロード・バランシング機能を使用する方法について説明します。
- エンタープライズ・ロード・バランシングのロード・マップの表示
  ここでは、現在のエンタープライズ・ロード・バランサーのロード・マップを表示する方法について説明します。
- エンタープライズ・ロード・バランシング・アクティビティー・レポートの表示
  ここでは、エンタープライズ・ロード・バランシングのイベントとアクティビティーのレポートを表示する方法について説明します。
- エンタープライズ・ロード・バランシングの Guardium 構成パラメーター
  この参照情報では、ロード・バランサーの構成パラメーターについて詳しく説明します。CM では、「管理」 > 「一元管理」 > 「エンタープライズ・ロード・バランシング」 > 「エンタープライズ・ロード・バランシング・プロパティー」からアクセスします。MU では、「セットアップ」 > 「一元管理」 > 「登録およびロード・バランシング」からアクセスします。

親トピック: 一元管理機能の使用

エンタープライズ・ロード・バランシング用に S-TAP を管理対象ユニットに関連付ける

ここでは、S-TAP グループを作成して管理対象ユニットのグループに関連付けることにより、エンタープライズ・ロード・バランシング機能を使用する方法について説明します。

このタスクについて

ロード・バランシングを行うと、S-TAP グループと管理対象ユニット・グループとの間に関連付けが作成され、グループ内の S-TAPs を、グループ内で最も可用性が高い管理対象ユニットに再割り当てできるようになります。このタスクでは、エンタープライズ・ロード・バランシング機能を使用できるようにするため、S-TAP グループと管理対象ユニット・グループとの間に関連付けを作成する必要があります。

手順

1. 中央マネージャーで、「管理」 > 「一元管理」 > 「エンタープライズ・ロード・バランサー (Enterprise Load Balancer)」 > 「S-TAP と管理対象ユニットの関連付け」にナビゲートします。
2. まだ S-TAP グループが作成されていないか、新しい S-TAP グループが必要な場合は、新たに S-TAP グループを作成します。
   a. アイコンをクリックして、「新規 S-TAP グループの作成」ダイアログを開きます。
   b. 「グループ名」フィールドに名前を入力します。 例えば、`North_American_S-TAPs` などです。
      推奨: 他の Guardium コンポーネントとの互換性を維持するために、グループ名でスペースや特殊文字は使用しないでください。
   c. 既存のホスト名から選択するか、「グループ・メンバー」フィールドを使用して新規メンバーを追加することにより、グループ・メンバーを追加します。 アイコンが表示されている S-TAPs は、新しい S-TAP グループに含まれています。
   d. 「新規グループの作成」をクリックして、S-TAP グループを作成します。
3. S-TAP グループを管理対象ユニット・グループに関連付けます。
   a. 関連付ける S-TAP グループを選択します。 例えば、`North_American_S-TAPS` などです。
   b. 「管理対象ユニットの関連付け」をクリックして、「管理対象ユニット・グループの関連付け」ダイアログを開きます。
   c. 必要な場合は、新しい管理対象ユニット・グループを作成します。
      i. 「管理」 > 「一元管理」 > 「管理対象ユニット・グループ」にナビゲートします。
      ii. アイコンをクリックして、「新規管理対象ユニット・グループの作成」ダイアログを開きます。
      iii. 「グループ名」フィールドに名前を入力します。例えば、`North_American_MUs` などです。
      推奨: 他の Guardium コンポーネントとの互換性を維持するために、グループ名でスペースや特殊文字は使用しないでください。
      iv. 既存の「管理対象ユニット IP アドレス (Managed Unit IP addresses)」から選択してグループ・メンバーを追加します。
      v. 「新規グループの作成」をクリックして、管理対象ユニットの新しいグループを作成します。
   d. S-TAP グループに関連付ける管理対象ユニット・グループを選択します。 例えば、`North_American_MUs` などです。
   e. 「適用」をクリックします。
4. 「保存」をクリックして、S-TAP グループと管理対象ユニット・グループ間の関連付けを完了します。
5. (オプション) S-TAP グループを管理対象ユニットのフェイルオーバー・グループに関連付けます。
   a. 関連付けたい S-TAP グループを選択します。これには、既に管理対象ユニット・グループに関連付けられているものを選択します。例えば、North_American_S-TAPS などです。
   b. 「フェイルオーバー・グループの関連付け」をクリックして「フェイルオーバー・グループの関連付け」ダイアログを開きます。
   c. 必要な場合は、上記と同じようにして新しい管理対象ユニット・グループを作成します。通常の管理対象ユニット・グループとフェイルオーバー・グループの両方は、S-TAP グループとの関連付けの際に指定されるまで同じです。
   d. S-TAP グループに関連付ける管理対象ユニット・グループを選択します。例えば、North_American_MUs_failover などです。
   e. 「適用」をクリックします。
6. 「保存」をクリックして、S-TAP グループと管理対象ユニット・グループ間の関連付けを完了します。

親トピック: エンタープライズ・ロード・バランシング

エンタープライズ・ロード・バランシングのロード・マップの表示

ここでは、現在のエンタープライズ・ロード・バランサーのロード・マップを表示する方法について説明します。

このタスクについて

エンタープライズ・ロード・バランシング・アプリケーションは、管理対象ユニットから収集した負荷情報を使用して、ロード・マップを作成します。このロード・マップにより、ロード・バランシングを指示するデータと、管理対象ユニットの割り当てアクティビティーが指定されます。

手順

1. 現在のロード・マップを Guardium UI のレポートとして表示するには、「管理」 > 「レポート」 > 「ユニット使用状況」 > 「ロード・バランサー」にナビゲートします。
2. また、現在のロード・マップは Guardium API を使用して表示することもできます。GuardAPI コマンドの grdapi get_load_balancer_load_map を実行します。

   ロード・マップは以下の例のようになります。

```
ID=0

************ LOAD MAP *************

***************  LOADED MU LIST ***********************

***************  VACANT MU LIST ***********************
  {
   MU=myguard_01.domain.com
   MU_QUEUE_SIZE(MB)=25.0
   MU_TIMES_REBALANED=0
   MU_EFFECTIVE_MAX_USED_QUEUE(%)=0.0
   MU_MAX_LOAD_CONTIB_BY_STAP(MB)=0.0
   MU_ADJUSTED_STAP_CONTRIB_IN_MB=0.0
```

```
MU_BASE_MAX_USED_QUEUE_IN_MB=0.0
IS_REBALANCABLE=true
INSTALLED_POLICIES=log full details|
APPLIANCE_RESOURCE_INFO={NUM_PROCESSORS=4,CPU_SPEED=2800,CPU_CACHE=25600,CPU_CORES=4,
   CACHE_READ_RATE=7870,HARD_DRIVE_READ_RATE=186,MEMORY_SIZE=24607}
STAP_LIST=
   {
   STAP_IP=01_gct1.domain.com, STAP_HOST=01_gct1.domain.com, CONNECTED_TO_MU=gct1.domain.com,
      PARTICIPATES_IN_LOAD_BALANCING=false, MAX_STAP_CONTIBUTION_TO_LOAD_IN_MB=0.0,
      AVG_STAP_CONTIBUTION_TO_LOAD_IN_MB=0.0
   }
   {
   STAP_IP=02_gct1.domain.com, STAP_HOST=02_gct1.domain.com, CONNECTED_TO_MU=gct1.domain.com,
      PARTICIPATES_IN_LOAD_BALANCING=false, MAX_STAP_CONTIBUTION_TO_LOAD_IN_MB=0.0,
      AVG_STAP_CONTIBUTION_TO_LOAD_IN_MB=0.0
   }
   {
   STAP_IP=03_gct1.domain.com, STAP_HOST=03_gct1.domain.com, CONNECTED_TO_MU=gct1.domain.com,
      PARTICIPATES_IN_LOAD_BALANCING=false, MAX_STAP_CONTIBUTION_TO_LOAD_IN_MB=0.0,
      AVG_STAP_CONTIBUTION_TO_LOAD_IN_MB=0.0
   }
}

***************  STAP -> MUS ALLOCATION TABLE ***********************
03_gct1.domain.com ----> gct1.domain.com
02_gct1.domain.com ----> gct1.domain.com
01_gct1.domain.com ----> gct1.domain.com
ok
```

親トピック: エンタープライズ・ロード・バランシング

エンタープライズ・ロード・バランシング・アクティビティー・レポートの表示

ここでは、エンタープライズ・ロード・バランシングのイベントとアクティビティーのレポートを表示する方法について説明します。

このタスクについて

「エンタープライズ・ロード・バランサー・イベント」レポートには、S-TAP エージェントと管理対象ユニットとの間の正常な関連付け、管理対象ユニットの負荷の変化、失敗した関連付けなど、ロード・バランシングに関するすべてのイベントとアクティビティーが表示されます。

手順

このレポートを表示するには、「管理」 > 「レポート」 > 「アクティビティー・モニター」 > 「エンタープライズ・ロード・バランサー・イベント」にナビゲートします。

親トピック: エンタープライズ・ロード・バランシング

エンタープライズ・ロード・バランシングの Guardium 構成パラメーター

この参照情報では、ロード・バランサーの構成パラメーターについて詳しく説明します。CM では、「管理」 > 「一元管理」 > 「エンタープライズ・ロード・バランシング」 > 「エンタープライズ・ロード・バランシング・プロパティー」からアクセスします。MU では、「セットアップ」 > 「一元管理」 > 「登録およびロード・バランシング」からアクセスします。

パラメーター	デフォルト値 (有効な値)	記述
STATIC_LOAD_COLLECTION_INTERVAL	720 (≥10)	静的な管理対象ユニット・ロード収集間隔 (分)。 ENABLE_DYNAMIC_LOAD_COLLECTION を 0 に設定すると、ロード・バランサーは、STATIC_LOAD_COLLECTION_INTERVAL で指定された間隔で、すべての管理対象ユニットから負荷情報を収集します。
LOAD_BALANCER_ENABLED	1 (0 または 1)	ロード・バランサー機能を制御します。 • 0 を指定すると、ロード・バランサー機能が無効になります。 • 1 を指定すると、ロード・バランサー機能が有効になります。 特定の管理対象ユニット上でロード・バランサー機能を無効にすると、中央マネージャー上で稼働しているロード・バランサーは、その管理対象ユニットの負荷情報を収集しなくなります。また、その管理対象ユニットに接続されているすべての S-TAPs が、ロード・バランシングの対象から除外されます。 CM 上でこのパラメーターを無効にしてから有効にすると、ロード・バランシングが有効になっているすべての管理対象ユニットを対象とする完全なロード収集が即時にトリガーされます。
ENABLE_DYNAMIC_LOAD_COLLECTION	1 (0 または 1)	ロード収集方法を制御します。 • 0 を指定すると、動的なロード収集間隔が無効になります (収集間隔として STATIC_LOAD_COLLECTION_INTERVAL が使用されます)。 • 1 を指定すると、動的なロード収集間隔が有効になります。 このパラメーターを有効 (1 に設定) すると、収集間隔が管理対象ユニットの数に比例します (接続されている 10 台の管理対象ユニットにつき 1 時間)。このパラメーターを変更すると、次回の完全なロード収集時刻が即時に再計算がトリガーされます。
USE_APPLIANCE_HW_PROFILE_FACT	1 (0 または 1)	ロード・バランサーは、S-TAPs の再配置を行うための空き管理対象ユニットを評価する際に、管理対象ユ

OR		ニットのハードウェア・プロファイル・インディケーター (APPLIANCE_HW_PROFILE_INDICATORS パラメーターで指定) を使用することができます。 • 0 を指定すると、ハードウェア・プロファイル・インディケーターが無視されます。 • 1 を指定すると、管理対象ユニットのハードウェア・プロファイル・インディケーターが使用されます。
MAX_RELOCATIONS_BETWEEN_FULL_LOAD_COLLECTIONS	3 (≥-1)	完全なロード収集の後に許可される、管理対象ユニット間での S-TAP の再配置の最大回数を定義します。 負の値を指定すると、許可される再配置の回数が無制限になります。
ALLOW_POLICY_MISMATCH_BETWEEN_APPLIANCES	1 (0 または 1)	ロード・バランサーは、管理対象ユニットのインストール済みポリシーを考慮することができます。 • 0: 異なるポリシーを持つ MU への S-TAP の再配置を許可しません。 • 1: 異なるポリシーを持つ MU への S-TAP の再配置を許可します。
TIME_TO_IGNORE_STAP_CONNECTION_RELATED_LOAD	10 (≥5)	各管理対象ユニットの S-TAPs の負荷統計情報を収集する際に、その管理対象ユニットに対する初期の S-TAP の接続を表すデータを除外したい場合があります。このデータは、ロード・バランサーの誤検出を作成するトラフィック・スパイクを示している場合があります。 TIME_TO_IGNORE_STAP_CONNECTION_RELATED_LOAD パラメーターにより、S-TAP から管理対象ユニットへの接続後に、指定した時間 (分) だけ S-TAP の負荷をロード・バランサーが無視するように設定することができます。
ENABLE_RELOCATION	1 (0 または 1)	リソースの再配置 (再バランシング) は、完全なロード収集の後にロード・バランサーが実行するプロセスです。ここでの再配置とは、負荷の高い管理対象ユニットから空き管理対象ユニットに S-TAPs を転送するという意味です。 • 0 を指定すると、空き管理対象ユニットに対する S-TAPs の再配置が禁止されます。 • 1 を指定すると、空き管理対象ユニットに対する S-TAPs の再配置が許可されます。
LOADED_SNIFFER_QUEUE_USAGE_THRESHOLD	0.6 (0.1 から 1 の範囲で、0.1 単位で増加)	管理対象ユニットのスニファーにサイズが LOADED_SNIFFER_QUEUE_ USAGE_THRESHOLD に達するキューが少なくとも 1 つある場合、その管理対象ユニットは負荷が高いと見なされます。 通常は、このパラメーターを変更することはありません。
DEFAULT_STAP_MAX_QUEUE_USAGE	0.15 (0.10 から 1 の範囲で、0.10 単位で増加)	S-TAP を初めて管理対象ユニットに割り当てた場合、ロード・バランサーはその管理対象ユニットに関する負荷情報を持っていません。このパラメーターの値により、スニファーで使用されるキューの一時的な最大値を定義します。この値は、管理対象ユニットから実際の負荷情報を収集するまでの間 (TIME_TO_IGNORE_STAP_CONNECTION_RELATED_LOAD パラメーターで定義された間隔が経過するまでの間) 使用されます。 通常は、このパラメーターを変更することはありません。
DEFAULT_STAP_MAX_CONTRIBUTION_TO_MAX_QUEUE_USAGE	0.1 (0.1 から 1 の範囲で、0.1 単位で増加)	S-TAP を初めて管理対象ユニットに割り当てた場合、ロード・バランサーはその管理対象ユニットに関する負荷情報を持っていません。このパラメーターの値により、キューの一時的な最大使用量に対する S-TAP の一時的な最大負荷の値を定義します。この値は、管理対象ユニットから実際の負荷情報を収集するまでの間 (TIME_TO_IGNORE_STAP_CONNECTION_RELATED_LOAD パラメーターで定義された間隔が経過するまでの間) 使用されます。 通常は、このパラメーターを変更することはありません。
REBALANCE_IF_MU_CLASSIFIED_AS_LOADED_N_TIMES_IN_M_HOURS	1:168 (≥0 : ≥0)	負荷の高い管理対象ユニットのリバランスを行うには、その管理対象ユニットについて、指定した時間内に指定したインスタンスの回数だけ負荷の高い管理対象ユニットとして分類する必要があります。例えば 1:168 という値の場合、168 時間以内に 1 回以上、その管理対象ユニットを負荷の高い管理対象ユニットとして分類する必要がある、という意味になります。
APPLIANCE_HW_PROFILE_INDICATORS	NUM_PROCESSORS: CPU_SPEED: CPU_CACHE: CPU_CORES: MEMORY_SIZE (*APPLIANCE_RESOURCE_INFO* テーブルの列名)	ロード・バランサーは、管理対象ユニットのハードウェア・プロファイル・インディケーターを考慮することができます。ロード・バランサーは、コロンで区切られたインディケーター (*APPLIANCE_RESOURCE_INFO* テーブルの列名) のリストを使用して、ハードウェア・プロファイルを評価します。 通常は、このパラメーターを変更することはありません。
MAX_CONCURRENT_LOAD_COLLECTIONS	10 (≥1)	ロード・バランサーが任意の時点で実行する同時ロード収集プロセスの最大数。つまり、中央マネージャーから管理対象ユニットに対する、非永続的な同時リモート SQL 接続の数です。
MAX_RELOCATIONS_PER_MU_BETWEEN_FULL_LOAD_COLLECTIONS	3 (≥-1)	いずれかの完全なロード期間中に許可される特定の管理対象ユニットからの S-TAP の再配置の最大回数。 このパラメーターは、再配置される STAP の MU ごとの最大数です。ロード中の S-TAP が 2 つある場合にこの値が 1 に設定されていると、特定の MU のこれらの S-TAP のうちどちらか 1 つのみを移動できます。この値を 0 に設定すると、STAP は再配置されません。 負の値を指定すると、許可される再配置の回数が無制限になります。
ENABLE_FAILOVER_GROUPS_REBALANCE	0 (0 または 1)	メイン MU グループで MU が再び使用可能になった時点で S-TAP をフェイルオーバー・グループからメイン MU グループに戻すための自動再配置を制御します。 0: S-TAP をメイン MU グループに戻すための自動再配置を許可しません。 1: S-TAP をメイン MU グループに戻すための自動再配置を許可します。

親トピック: エンタープライズ・ロード・バランシング
関連情報:
GuardAPI エンタープライズ・ロード・バランシング関数

デプロイメント・インベントリー

「インベントリー (inventory)」ビューには、すべてのデータベース・サーバーとインストール済みの S-TAP クライアントまたは GIM クライアントの一元管理ビューが表示されます。

親トピック: 一元管理機能の使用

「リソース・デプロイメント」ビュー

「リソース・デプロイメント」ビューには、すべてのデータベース・サーバーと、関連するコレクター、アグリゲーター、および中央マネージャーの一元管理ビューが表示されます。

親トピック: 一元管理機能の使用

管理対象ユニット・グループの作成

管理対象ユニットをグループに編成してから、それらのグループにアクションを実行します。

このタスクについて

管理対象ユニット・グループを使用すると、管理対象ユニットを分かりやすいグループに編成してから、それらのグループにアクションを実行することができます。例えば、特定のユニット・タイプ、地理的位置、または業務別の管理対象ユニット・グループを作成できます。実行するアクションには、管理対象ユニットのグループへのポリシーのインストール、またはパッチもしくは構成の配布などがあります。

手順

1. 「管理」 > 「一元管理」 > 「管理対象ユニット・グループ」にナビゲートします。
2. 「管理対象ユニット・グループ」ページで、  をクリックして新規管理対象ユニット・グループを作成するか、  をクリックして既存のグループを編集します。
3. 「新規管理対象ユニット・グループの作成」ダイアログで、「グループ名」フィールドにグループの名前を入力します。
   推奨: 他の Guardium コンポーネントとの互換性を維持するために、グループ名でスペースや特殊文字は使用しないでください。
4.  アイコンを使用して、グループに組み込む管理対象ユニットを選択します。
5. グループに組み込む管理対象ユニットの選択が完了したら、「保存」ボタンをクリックします。 新しい管理対象ユニット・グループが保存され、「管理対象ユニット・グループ」ページに表示されます。
6. オプションとして、「管理対象ユニット・グループ」ページで、アイコンをクリックして、グループを展開し、その管理対象ユニットを表示します。

タスクの結果

定義後、管理対象ユニット・グループは、「管理」 > 「一元管理」 > 「一元管理」ページから、「管理」 > 「一元管理」 > 「構成プロファイルの配布」ページから、「管理」 > 「一元管理」 > 「エンタープライズ・ロード・バランシング」 > 「S-TAP と管理対象ユニットの関連付け」ツール内の管理対象ユニット・グループとして、および管理対象ユニット・グループが使用されるその他の場所で利用できます。
親トピック: 一元管理機能の使用

管理対象ユニットのモニター

一元管理を使用して管理対象ユニットをモニターします。

管理対象装置をモニターするには次のようにします。

1. admin ユーザーとして、管理対象ユニットの「Guardium® GUI」にログインします。
2. 「レポート」 > 「Guardium 運用レポート」 > 「管理対象ユニット」をクリックして、「管理対象ユニット」を開きます。

「一元管理」ペインの各コンポーネントの説明を、以下の表に示します。

表 1. 管理対象ユニットのモニター

コントロール	記述
「すべて選択」チェック・ボックス	列 1 の陰影付きのエリアにあるこのボックスにマークを付けると、すべての管理対象ユニットが選択されます。
選択をすべて解除	すべての管理対象ユニットをクリアします。
チェック・ボックス	このボックスにマークを付けると、操作対象にするユニットが選択されます。
ユニット情報のリフレッシュ	ユニットの展開ビューに表示されているすべての情報をリフレッシュし、そのユニットに新規要求を発行します。このアクションによって、フル・ユーザー同期サイクルも発生します。
ユニットのリブート	ユニットをオペレーティング・システム・レベルでリブートします。デフォルトでは、Guardium ポータルは始動時に開始します。
ユニット・ポータルの再始動	管理対象ユニット上で Guardium アプリケーション・ポータルを再始動します。その後、そのユニットにログインして、Guardium タスク (検査エンジンの定義または削除など) を実行できます。

コントロール	記述
ユニット SNMP 属性の表示	「SNMP ビューアー 」ペインを別のウィンドウで開きます。「SNMP ビューアー」ペインのリフレッシュ・アイコンをクリックすると、ウィンドウのデータがリフレッシュされます。
ユニット syslog の表示	「syslog ビューアー」を別のウィンドウで開きます。syslog メッセージの最新 64 KB が表示されます。「syslog ビューアー」ペインの「リフレッシュ」アイコンをクリックすると、ウィンドウのデータがリフレッシュされます。
ユニット・ポータルへのショートカット	管理対象ユニット用の Guardium ログイン・ページを、別のブラウザー・ウィンドウで開きます。
ユニット名	管理対象ユニットのホスト名。マウス・ポインターをユニット名の上に置くと、ツールチップに IP アドレスが表示されます。 ユニット上でホスト名が変更された場合、オンライン状況の自動リフレッシュが行われると、中央マネージャーがそのユニットを認識しなくなります。ホスト名が変更された可能性がある場合は、ツールバーの「リフレッシュ」を使用してください。変更されたホスト名を取得し、そのユニットについて表示される現在のオンライン状況とその他の情報を更新します。
オンライン	ユニットがオンラインかどうかを示します。緑のインディケーターが点灯している場合、ユニットはオンラインです。赤いインディケーターが点灯している場合、ユニットはオフラインです。 中央マネージャーは、一元管理構成で指定されたリフレッシュ間隔 (デフォルトは 1 分) でこの状況をリフレッシュします。ユニットへの接続でエラーが発生した場合、ツールチップにエラーの記述が表示されます。管理表のそのユニットのレコードの上にマウス・ポインターを移動してください。
検査エンジン	⊞ ・アイコンをクリックすると、検査エンジンのリストが展開され、⊟ ・アイコンをクリックすると、検査エンジンのリストが非表示になります。 ここから、状況に応じて検査エンジンを停止または開始できます。 各検査エンジンについて表示される情報は次のようになります (この情報は、「リフレッシュ」が押されるときに管理対象ユニットから取り出されます。ping のたびに取り出されるのではありません)。 名前 - 検査エンジンの名前 プロトコル- 検査エンジンがモニター対象にするプロトコル: Oracle、MSSQL、Sybase、Informix®、または DB2® 始動時にアクティブ - システムの始動時に検査エンジンが開始するかどうかを示します。 送信元 IP の除外 - 送信元 IP アドレスのリストを除外する (調査しない) かどうかを示します。 送信元 IP/マスク - クライアントの IP アドレスとサブネット・マスクのリスト。検査エンジンがモニターするのは、このクライアントにおける「送信先 IP/マスク」アドレスへのデータベース・トラフィックです。 ポート - データベース・クライアントとサーバーが通信に使用するポート。単一ポート、ポートのリスト、またはポートの範囲の場合があります。 送信先 IP/マスク - サーバーの IP アドレスとサブネット・マスクのリスト。モニター対象となるのは、対応するクライアント・マシン (「送信元 IP/マスク」) からこのサーバーへのトラフィックです。
インストール済みセキュリティー・ポリシー	管理対象ユニット上にインストールされたセキュリティー・ポリシーの名前。このフィールドは ping のたびに更新されます。
モデル	管理対象ユニットの Guardium モデル番号。
バージョン	管理対象ユニットの Guardium バージョン番号。
最終パッチ	最後にインストールされたパッチ。
最終 ping 時刻	中央マネージャーが管理対象ユニットのオンライン/オフライン状況を判断するために、最後にこのユニットが ping された時刻。
選択済みユニット	
グループ・セットアップ	「グループ・セットアップ」を使用すると、グループの保守 (新規グループの作成、グループの削除、および管理対象ユニットのグループへの関連付け) をユーザーが行えるようにする新規ウィンドウが開きます。
登録抹消	選択されたすべてのユニットを登録抹消します。
再始動	
リブート	選択されたユニットをリブートします。
ポータルの再始動	選択されたポータルを再始動します。
検査エンジンの再始動	選択されたユニットの検査エンジンを再始動します。
配布	
リフレッシュ	選択されたユニットをリフレッシュします。
ポリシーのインストール	ポリシー名は、ポリシーの詳細を示す新規ウィンドウを開くリンクになっています。
パッチ配布	「パッチ配布」を押すと新しい画面が開き、使用可能なパッチのリストが従属関係とともに表示されます。さらにそこからパッチを選択し、選択したすべてのユニットにインストールできます。最大 1 年先までのパッチのスケジュールを設定します。

コントロール	記述
アップロードした JAR ファイルの配布	「強化」 > 「脆弱性評価」 > 「カスタム・アップロード」をクリックします。次に、アップロードするファイルの名前を入力します。または、「参照」クリックしてそのファイルを探し、選択します。ドライバーは、1 つずつアップロードしてください。 「アップロード」をクリックします。操作完了時には通知があり、アップロードしたファイルが表示されます。 このアクションによって、アップロードされたファイルが中央マネージャーに移動します。 これらの JAR ファイルの配布対象となる管理対象ユニットのチェック・ボックスを選択します。「アップロードした JAR ファイルの配布」をクリックします。
パッチ・バックアップ設定の配布	この設定により、選択したユニットに以下が配布されます。 PATCH_BACKUP_FLAG; PATCH_AUTOMATIC_RECOVERY_FLAG; PATCH_BACKUP_DEST_HOST; PATCH_BACKUP_DEST_DIR; PATCH_BACKUP_DEST_USER; PATCH_BACKUP_DEST_PASS
認証構成の配布	一元管理認証の配布を受信する管理対象ユニットを選択します。 「認証構成の配布」 をクリックすると、選択したすべての管理対象ユニットに認証構成が配布されます。
構成の配布	以下の構成が配布され、中央マネージャーと管理対象ユニット間でパラメーターが同期します。 • 異常検出 - 始動時にアクティブ、ポーリング間隔 • アラート機能 - すべてのフィールド • データ・アーカイブ - すべてのフィールド • グローバル・プロファイル - 同時ログイン、データ・レベル・セキュリティー、名前付きテンプレート (既に同期済み) 以外のすべてのフィールド、PDF フッター・テキスト、およびロゴ・イメージ • IP からホスト名への別名割り当て - 両方のチェック・ボックス • 結果アーカイブ - すべてのフィールド • 結果エクスポート - すべてのフィールド • セッション推論 - すべてのフィールド • システム・バックアップ - すべてのフィールド • データ・エクスポート - すべてのフィールド 上記の構成のうち一部 (「異常検出」、「セッション推論」) は、ポータルが再始動されるまで有効になりません。「アラート機能」などの他のプロセスは、管理対象ユニットの「管理」ポータルを介して直接再始動するか、または関係のあるすべての管理対象ユニットをマネージャーからリブートすることによって再始動する必要があります。 「構成の配布」 では、管理対象ユニットは再始動しません。再始動する管理対象ユニットごとに別個のアイコンがあります。 「ポータルの再始動」では、選択したすべてのユニットが再始動されます。 配布後に、管理対象ユニットですべての構成を有効にするため、管理対象ユニットを再始動する必要があることを示すメッセージが表示されます。 スケジュール設定が付いている各パラメーターには、第 2 のチェック・ボックスがあります。この第 2 のボックスにチェック・マークを付けると、このパラメーターのスケジュール設定が配布されます。 構成を選択して配布する方法については、『構成の配布』を参照してください。 ポータルのリブートと再始動の比較 アラート機能 「始動時にアクティブ」チェック・ボックス。アプライアンスが再始動するたびに、アラート機能が自動的にアクティブ化されます。 GUI 再始動では「始動時にアクティブ」値は有効になりません。 中央マネージャーから管理対象ユニットへの構成の配布を完全に有効にするため、管理対象ユニットのリブートが必要です。 アラート機能については、admin ポータルを使用して管理対象ユニット上で手動で再始動します (「管理コンソール」/「アラート機能」)。この再始動は Central Manager からはできないので、同じ効果を得るため、「管理コンソール」から管理対象装置を再始動します。 異常検出 「始動時にアクティブ」チェック・ボックス。アプライアンスが再始動するたびに、異常検出が自動的にアクティブ化されます。 GUI 再始動で「始動時にアクティブ」値が有効になります。 中央マネージャーから管理対象ユニットへの構成の配布を完全に有効にするため、管理対象ユニットでポータルの再始動が必要です。 セッション推論

コントロール	記述
	「始動時にアクティブ」チェック・ボックスを使用して、Guardium アプライアンスの開始時にセッション推論を開始します。 GUI 再始動で「始動時にアクティブ」値が有効になります。 中央マネージャーから管理対象ユニットへの構成の配布を完全に有効にするため、管理対象ユニットでポータルの再始動が必要です。 結果エクスポート/システム・バックアップ/データ・アーカイブ/結果アーカイブ/データ・エクスポート 中央マネージャーから管理対象ユニットへの構成の配布を有効にするために、管理対象ユニットでポータルの再始動をする必要ありません。 グローバル・プロファイル 中央マネージャーから管理対象ユニットへの構成の配布を有効にするために、管理対象ユニットでポータルの再始動をする必要はありません (ただし、異なる名前付きテンプレートの使用は、ポリシーがインストールされたときにのみ適用されます)。
新規登録	「ユニット登録」ペインを開いて、管理対象にするユニットを新規登録します。
パッチ・インストール状況	「パッチ・インストール状況」画面には、各ユニットごとの、失敗したインストールと矛盾が表示されます。例えば、あるパッチが、他のユニットでインストールに失敗したか、インストールされなかったかにかかわらず、一部のユニットにのみインストールされている状況です。

中央マネージャーを使用した、個々の管理対象ユニットまたは管理対象ユニット・グループへの相関アラートの割り当て

この新機能は、管理対象環境に対するものです。

中央マネージャーが、個々の管理対象ユニットまたは管理対象ユニット・グループに相関アラートを割り当てることができるようにします。ユニットまたはグループに割り当てるか、ユニットまたはグループから除外することができます。また、中央マネージャー自体で実行するかどうかも指定する必要があります。使用されるグループは管理対象ユニット・グループであり、中央マネージャーのページで使用されるのと同じタイプのグループです。

管理対象環境の中央マネージャーで、アラート・ビルダーには、「管理対象ユニット」用の新しいセクションがあります。このセクションで、アラートに組み込むか、アラートから除外する単一のユニットまたは管理対象ユニットのグループのどちらかを指定します。また、その中央マネージャー自体が組み込まれるか、除外されるかも、チェック・ボックスで指定します。デフォルトの動作は既存の動作と一致します。すなわち、アラートはどこでも実行されます。アラートがどこでも実行されないことを指定する場合、指定する場所でアラートが実行されることを確認してください。UI には、単一ユニットまたはグループの組み込み/除外のための 4 つのオプションが含まれています。また、管理グループのリストから選択し、必要に応じて新規管理グループを作成するか、または既存の管理対象ユニット・グループを編集するためのダイアログも含まれています。

個々の管理対象ユニットで、アラート・ビルダーは管理対象ユニットのセクションを表示しません。中央マネージャーだけがユニットとグループにアラートを割り当てることができます。

所定の管理対象ユニットで「アラート」表にエントリーがある場合、そのユニットの除外元のアラートごとにそのユニットを除外するために、システム生成グループが自動的に作成されます。これが行われるのは、アラートがその管理対象ユニットで開始する場合です。

アラートをローカルで有効/無効にするために、管理コンソールで異常検出ページのアラート・ペインが使用されました。この機能の場合、アラート・ペインは中央マネージャーにのみ表示されます。

管理対象ユニットには、アクティブ・アラートおよびそれらのアクティブ・アラートが有効であるかどうかを示す表示が表示されます。

親トピック: 一元管理機能の使用

管理対象ユニットへのセキュリティー・ポリシーのインストール

管理対象ユニットにセキュリティー・ポリシーをインストールします。

このタスクについて

管理対象ユニットへのセキュリティー・ポリシーのインストール

手順

1. 「設定」 > 「ツールとビュー」 > 「ポリシー・インストール」をクリックして、「現在インストールされているポリシー」と「ポリシー・インストーラー」を開きます。
2. 「ポリシー」リストから、インストールするポリシーを選択します。
3. リストからインストール・アクションの選択をします。インストール・アクションを選択した後、各ポリシーのインストールの成功 (または失敗) が通知されます。選択したユニットを使用できない場合 (オフラインの場合やリンクが停止している場合)、中央マネージャーからそのことが通知されます。最長で 7 日間 (そのユニットが一元管理に登録されている間)、引き続き新しいポリシーのインストールが試みられます。
4. 「ポリシー」リストから、インストールするポリシーを選択します。
5. 使用可能なインストール・アクションには、以下の項目があります。
    a. 「インストールおよびオーバーライド」 - インストール済みのポリシーすべてを削除し、代わりに選択したものをインストールします。
    b. 「最後のインストール」 - 選択したポリシーをシーケンスの最後のものとしてインストールします。すなわち、現在インストールされているすべてのポリシーの後にこのポリシーをインストールし、優先度は最も低いです。

c. 「最初のインストール」- 選択したポリシーをシーケンスの最初のものとしてインストールします。現在インストールされているすべてのポリシーより前に、このポリシーをインストールします。

注: 中央マネージャーからポリシーをインストールする場合、「今すぐ 1 回実行」 (およびスケジューラー) を選択すると、インストール済みポリシー内の既存のグループが更新されます。

ルールの変更 (グループの追加と削除を含む) をロードするには、次のいずれかを行う必要があります。

a. コレクターからのポリシーの初期インストール
b. コレクターまたは中央マネージャーからのポリシーの再インストール

親トピック: 一元管理機能の使用

一元化パッチ管理

パッチのインストール、状況、および履歴を表示可能にし、制御します。

このタスクについて

パッチのインストール、状況、および履歴を表示可能にし、制御します。一元管理クラスターは、Central Manager から管理対象装置にパッチをインストールする機能を備えています。

パッチをインストールするときには、パッチをインストールする時期を示す日時要求を指定できます。日時を入力しない場合、または「now」が入力された場合、インストール要求時間は「今すぐ」です。

注: 正常にインストールされているパッチを再インストールできます。これは、バッチ処理の対象となるパッチに対して重要です。パッチが既にインストールされている場合は、警告で通知されます。

管理者ユーザーとして、管理対象ユニットの「Guardium® GUI」にログインします。

手順

1. 「管理」 > 「一元管理」 > 「一元管理」をクリックします。
2. パッチが必要なユニットを選択し、「パッチ配布」をクリックします。
3. 「パッチ配布」画面から、配布するパッチを選択し、「今すぐパッチをインストール」または「パッチのスケジュールを設定」をクリックします。
4. インストールの状況を確認するには、「管理」 > 「一元管理」 > 「一元管理」をクリックし、ユニットを選択し、「パッチ・インストール状況」をクリックします。「パッチ・インストール状況」画面には、各ユニットごとの、失敗したインストールと矛盾が表示されます。例えば、あるパッチが、他のユニットでインストールに失敗したか、インストールされなかったかにかかわらず、一部のユニットにのみインストールされている状況です。「パッチ配布」画面からパッチを削除するには、パッチの横の削除アイコン (赤色の x) をクリックします。これにより、アプライアンスのパッチ配布ディレクトリーからはパッチは削除されませんが、表示からは削除されます。

親トピック: 一元管理機能の使用

構成プロファイルの処理

構成プロファイルにより、中央マネージャーから構成設定およびスケジューリング設定を定義して、中央マネージャー自体の構成を変更することなく、それらの設定を管理対象ユニット・グループに配布することができます。

始める前に

構成プロファイルを作成し、配布する前に、以下の前提条件を確認してください。

- 中央マネージャーとその管理対象ユニット間のポート 8447 を介した通信を許可します
- 中央マネージャーと、構成を受け取る管理対象ユニットは、Guardium V10.1 以上でなければなりません

このタスクについて

構成プロファイルには、構成タイプ (構成設定とスケジューリング設定の 1 つ以上のセット) と、構成設定とスケジューリング設定で更新される管理対象ユニット・グループのリストという 2 つのタイプの情報が含まれています。構成プロファイルは、定義した後、保管および変更して、構成設定とスケジューリング設定の特定のセットを特定の管理対象ユニット・グループに配布するために再使用することができます。

構成プロファイルに追加できる構成タイプは次のとおりです。

- アラート機能
- データ・アーカイブ
- データ・エクスポート
- データ・インポートのスケジュール
- 未解析ログ処理
- IP からホスト名への別名割り当て
- Kerberos
- PIM データ相関
- ポリシー・インストールのスケジュール
- 結果アーカイブ (監査)
- 結果エクスポート (ファイル)
- セッション推論
- システム・バックアップ
- ユニット使用状況のスケジュール

- ユニット使用状況のしきい値

構成プロファイルは、中央マネージャーのローカル設定とは無関係に定義されます。 そのため、中央マネージャーの構成を中断したり、管理対象ユニットを個別に構成する必要なしに、構成設定を素早く定義し、その設定を管理対象ユニット・グループにデプロイすることができます。

この作業では、構成プロファイルの作成、配布、および保存の方法を説明します。

手順

1. 「管理」 > 「一元管理」 > 「構成プロファイルの配布」にナビゲートします。
2.  をクリックするか、既存のプロファイルを選択して、構成プロファイルの処理を開始します。
3. 「名前および記述」パネルで、プロファイルの名前、およびオプションで記述を指定します。「次へ」をクリックして先に進みます。
4. オプションで、「ロール」をクリックして、構成プロファイルを使用できるセキュリティー・ロールを指定します。
5. 「配布対象」パネルで、  をクリックして新規構成を定義するか、既存の構成を選択し、  をクリックして編集します。
   a. 「構成タイプ」メニューで、構成タイプを選択してプロファイルを追加します。
   b. 選択した構成タイプの構成とスケジューリングの詳細を指定します。
   制約事項: データ・エクスポート構成設定をアグリゲーターに配布しても、パージ設定は配布されません。アグリゲーター上の既存のパージ設定が保持されます。コレクター上で、保存期間を含むパージ設定が配布され、既存のパージ設定が置き換えられます。
   c. 「保存」をクリックして、構成プロファイルの編集を終了します。
   d. オプションで、  をクリックしてから保存することによって、その他の構成タイプやスケジュールを追加します。
   e. 「次へ」をクリックして先に進みます。
6. 「配布場所」パネルで、「管理対象ユニット・グループ」表からグループを選択して、  アイコンを使用して、グループを「選択されたグループ」表に追加します。「次へ」をクリックして先に進みます。
   注:  をクリックして新規管理対象ユニット・グループを作成するか、  をクリックして既存のグループを編集します。「管理」 > 「一元管理」 > 「管理対象ユニット・グループ」を選択して、管理対象ユニット・グループを定義および編集することもできます。
7. オプションで、「構成の配布」パネルで、「今すぐ実行する」をクリックして、選択したグループに構成プロファイルを配布します。配布が完了したことを状況が示している場合、「次へ」をクリックして続行します。
8. 「結果のレビュー」パネルで、配布プロセスとその結果の概要を確認します。
   オプション: 配布プロセスの詳しいログを参照する場合は、「実行ログ」をクリックします。
9. 再使用するために構成プロファイルを保存する場合は、「保存」をクリックします。

次のタスク

構成プロファイルを中央マネージャー間で移動する必要がある場合は、「管理」 > 「データ管理」 > 「定義のエクスポート」および「管理」 > 「データ管理」 > 「定義のインポート」を使用して、「タイプ」メニューから「構成プロファイル」を選択します。

親トピック: 一元管理機能の使用

関連概念:

統合

アラート機能の構成

定義のエクスポート/インポート

IP からホスト名への別名割り当て

スケジューリング

構成の配布

構成、ならびにそのスケジュールは、全体またはその一部を、中央マネージャーと管理対象ユニットの間で配布することができます。

手順

1. 構成を受け取る管理対象ユニットを「選択」します。
2. 「構成の配布」をクリックして、「構成の配布」ウィンドウを表示します。
3. 配布する「構成」に対応するボックスにチェック・マークを付けます。 ヘッダーにあるチェック・ボックスを使用すれば、すべての構成が選択されます。
4. 配布する「スケジュール」に対応するボックスにチェック・マークを付けます。 ヘッダーにあるチェック・ボックスを使用すれば、すべてのスケジュールが選択されます。構成がスケジュールされていない場合は、チェック・ボックスが表示されず、代わりに「適用外」が表示されます。
5. 「配布」をクリックし、構成とスケジュールを配布します。
6. オプション: 配布を中止するには、「キャンセル」をクリックします。

タスクの結果

コマンドを使用する場合は、「一元管理」 > 「構成の配布」 > 「グローバル・プロファイル」をクリックすると、以下の値が配布されます。

- ACTIVATE_ALIASES
- CUSTOM_DB_MAX_SIZE
- CHECK_CONCURRENT_LOGIN
- HTML_BOTTOM_RIGHT
- HTML_BOTTOM_LEFT
- DISPLAY_LOGIN_MESSAGE
- LOGIN_MESSAGE
- CSV_DELIMETER
- FILTERING_ENABLED
- INCLUDE_CHILDREN_ON_FILTER
- SHOW_ALL_RECORDS
- ACCORDION_DISABLED

- SCHEDULER_RESTART_INTERVAL
- SCHEDULER_RESTART_WAIT_SHUTDOWN
- ESCALATE_TO_ALL
- MESSAGE_TEMPLATE

親トピック: 一元管理機能の使用

認証構成の配布

各アプライアンスで別個に認証を構成する代わりに、中央マネージャー上で一元管理認証 (認証の構成) を 1 回構成し、それからすべての管理対象ユニットに配布することができます。このようにすると、情報の入力を 1 回行うことで、その情報を一部またはすべてのユニットに適用することができます。一部のユニットで異なるタイプの認証を使用することもできます。

手順

1. 認証 (認証の構成) を中央マネージャーと管理対象ユニットの両方で確実に実行します。LDAP 認証を使用している場合、LDAP が中央マネージャー上と管理対象ユニットの両方で構成されていることを確認します。
2. 一元管理認証の配布を受信する管理対象ユニットを選択します。
3. 「認証構成の配布」 をクリックすると、選択したすべての管理対象ユニットに認証構成が配布されます。

親トピック: 一元管理機能の使用

予備の中央マネージャー

予備の中央マネージャーまたはバックアップ中央マネージャー (CM) を使用して、プライマリー CM が使用不可になった場合に備えてセカンダリー CM またはバックアップ CM を構成します。

予備の中央マネージャーでは、以下の機能がサポートされています。

1. バックアップ中央マネージャー - プライマリー中央マネージャーの接続が切断されると、「プライマリー CM に設定」リンクが使用可能になります。
2. ユーザー・レイアウトが保持されます。
3. ユーザーおよびロールは同期バックアップに含まれており、ポータル・ユーザー同期に依存しません。
4. ユーザー・グループのロール・データが保持されます。
5. GuardAPI 関数 make_primary_cm が追加され、中央マネージャーへの切り替えを CLI から実行できるようになっています。
6. プライマリー中央マネージャーからバックアップ中央マネージャーへの切り替え後に、監査プロセス・ビルダーのプロセスからのデータが保持されます。
7. 一元管理バックアップには、以前と同様、すべての定義 (レポート、照会、アラート、ポリシー、監査プロセスなど)、ユーザー、およびロールが含まれます。
8. バックアップ対象に、エンタープライズ・レポート、配布レポート、および LDAP のスケジュールが含まれます。
9. バックアップ対象に、すべての監査プロセスのスケジュール、およびデータ管理プロセス (アーカイブ、エクスポート、バックアップ、インポートなど) のスケジュールと設定が含まれます。
10. バックアップ対象に、アラート機能および送信者の設定が含まれます。
11. ユーザーの GUI カスタマイズ、カスタム・クラス、およびアップロードされた JDBC ドライバーが含まれます。

注: データ (収集されたデータ、監査結果のデータ、およびカスタム表のデータなど) は含まれません。
注:

バックアップ CM 上の cm_sync_file の状況をリストするには、CLI コマンド show local_cm_sync_file を使用します。各管理対象ユニットのバックアップ CM IP の値をリストするには、GuardAPI コマンド grdapi show_backup_cm_ip を使用します (この API コマンドを実行できるのは、中央マネージャー上のみです)。

注: 中央マネージャーのロード・バランシングを伴うフェイルオーバー - フェイルオーバー後に新しい管理対象ユニットが接続してすぐに切断すると、フェイルオーバー・メッセージが受信されるまで、正しい DB_USER が送信されません。

開発サーバーまたはセカンダリー・サーバーで以下の手順を実行して、テストを行ってください。正常に動作した場合は、プライマリー (稼働中の) Guardium サーバーで以下の手順を実行してください。

中央マネージャーへのパッチのインストール

1. 現在のプライマリー CM から、CLI としてログインします。
2. CLI コマンド store system patch install scp を使用して、パッチをインストールします。
3. この CLI コマンドにより、Guardium サーバーにファイルがコピーされ、これらのファイルをインストールできるようになります。
4. CLI コマンド show system patch install を使用して、これらのパッチのインストール状況を監視します。
5. 両方のパッチのパッチ状況が「DONE: Patch installation Succeeded.」と表示されるまで待機します。

バックアップ CM へのパッチのインストール

1. 現在のプライマリー CM GUI に admin としてログインします。
2. 「設定」>「ツールとビュー」を選択し、次に「中央マネージャー」を選択します。

3. 中央マネージャー上にあるバックアップ CM 管理対象ユニットのチェック・ボックスをクリックします。
4. 「パッチ配布」をクリックし、前の手順でプライマリー CM にインストールしたすべてのパッチをインストールします。

パッチのインストールの例

1. 「パッチ配布」をクリックします。
2. 「今すぐパッチをインストール」をクリックします。
3. パッチがすべての管理対象サーバーに確実にインストールされるまで、約 15 分間、待機します。
4. 確認のために、バックアップ CM に CLI としてログインし、バックアップ CM サーバーから CLI コマンド show system patch install を実行します。

その他すべての管理対象サーバーへのパッチのインストール (オプションの手順)

1. 上記の手順を繰り返して、すべての管理対象サーバーにパッチをインストールします。
2. 次の手順に進む前に、すべてのパッチがインストールされていることを確認します。

すべてのパッチが CM および管理対象サーバーにインストールされた後の手順

1. 現在のプライマリー CM に admin としてログインします。
2. 「設定」>「ツールとビュー」を選択し、次に「中央マネージャー」を選択します。「バックアップ CM の指定」をクリックします。
3. 返された適格なバックアップ CM 候補のリストから、バックアップ CM サーバーを選択します。
4. 「適用」をクリックします。
5. バックアップ CM が同期化されて、新規バックアップ CM ファイルが作成され、バックアップ CM にコピーされるまで、約 2 分間、待機します。
6. 2 つのバックアップ CM 同期ファイルについて、バックアップのすべての処理が 2 回完了するまで (約 1 時間) 待機します。これらのファイルは、バックアップ CM にコピーされ、「Guardium モニター」タブの「統合/アーカイブ・ログ」レポートに表示されます。
7. バックアップ CM 同期ファイルの作成の進行状況を表示するには、「Guardium モニター」を選択し、「統合/アーカイブ・ログ」レポートを選択します。
8. アクティビティー・バックアップが開始され、「統合/アーカイブ・ログ」レポートから cm_sync_file.tgz ファイルが作成されたことを確認します。
    a. GUI から管理者としてログインします。
    b. 「Guardium モニター」タブを選択します。
    c. 「統合/アーカイブ」レポートを選択します。
    d. バックアップ・タイプを探します。
9. 完了すると、以下の状態になります。
    a. パッチが CM にインストールされています。
    b. パッチがバックアップ CM にインストールされています。
    c. オプション: パッチが他のすべての管理対象ユニットにインストールされています。
    d. 2 つのバックアップ CM 同期ファイルが作成されています (「Guardium モニター」タブの「統合/アーカイブ・ログ」ファイルを参照)。
    e. 以下の手順では、現在のプライマリー CM とその管理対象ノードをバックアップ CM に変換するプロセスの概要を示します。

注:

- 重要: バックアップ CM をサポートする 2 つのバックアップ CM 同期ファイルの処理が確実に完了するまで、約 1 時間、待機してください。
- バックアップ CM 同期ファイルのバックアップ・スケジュールは、約 30 分ごとです。
- バックアップ CM ファイルを作成し、そのファイルをバックアップ CM 上のディレクトリーにコピーするためのプロセスが、CM で実行されます。

2 つの同期ファイルの処理が完了した後のバックアップ CM プロセスの開始

プライマリー CM Guardium サーバーをシャットダウンします。

プライマリー CM をシャットダウンするためのアクセス権限がない場合は、バックアップ CM に直接移動して、admin としてログインし (「設定」>「ツールとビュー」を選択し、次に「一元管理」を選択)、「プライマリー CM に設定」をクリックします。本資料のセクション『バックアップ CM をプライマリー CM にするための構成を開始するための手順』にスキップします。

1. 約 5 分間待機し、バックアップ CM の GUI に admin として再度ログインします。
2. プライマリー CM が完全にシャットダウンされたら、次のステップに進むことができます。

注:

プライマリー CM にログインしているときに、そのプライマリー CM がシャットダウンされた場合は、接続がタイムアウトになったことを示すメッセージが表示されます。

バックアップ CM をプライマリー CM にするための構成を開始するための手順

セカンダリー CM は、約 5 分間、応答できない状態になります。5 分後にログインすると、「プライマリー CM に設定」リンクが使用可能になります。このリンクは、admin としてログインし、「設定」>「ツールとビュー」>「一元管理」を選択すると使用可能になります。

1. プライマリー・サーバーがシャットダウンされると、バックアップ CM に「リモート・マネージャーに接続できません。(バックアップ CM の名前) への切り替えを検討してください」というメッセージが表示されます。
2. 切り替えを行う場合は、以下の手順を実行します。
    a. 管理者としてログインします
    b. 「設定」>「ツールとビュー」を選択します。
    c. 「プライマリー CM に設定」をクリックします (「プライマリー CM に設定」リンクは複数回クリックしないでください。また、このプロセスの実行中は、この画面から移動せず、他のオプションを選択しないでください。このプロセスの進行状況と完了を確認できるログ・ファイルが作成されます)。このプロセスの完了にはしばらく時間がかかります。安全防護策として、このボタンを複数回クリックしてしまっても、現在のプロセスには変更が加えられないようになっています。
    d. 数秒後に、「このユニットをプライマリー CM にしてよろしいですか?」というメッセージが表示されます。「OK」をクリックします。
    e. さらに数秒後に、「数分の時間がかかる場合があります」というメッセージが表示されます。バックアップ CM がプライマリー CM になるために要する時間は、バックアップ CM 同期ファイルからバックアップされるデータの量、およびプライマリー CM になるバックアップ CM への切り替えを行う管理対象ノードの総数によって異なります。「OK」をクリックします。

        「OK」をクリックすると、load_secondary_cm_sync_file.log というログ・ファイルが直ちに作成されます。このファイルを使用すると、切り替えの進行状況から、バックアップ CM への切り替えプロセスの完了に至るまでを確認できます。このファイルは GUI から表示できます。以降の手順では、このログ・ファイルを表示するための方法を示します。
    f. 最後のメッセージが画面に表示されるまでには、しばらく時間がかかります。これが、バックアップ CM への切り替えが完了する前の最後のメッセージになります。そのメッセージは、「GUI は今すぐ再始動されます。数分後に再度ログインすると、バックアップ CM がプライマリー CM になります」です。「OK」をクリックします。

        バックアップ CM がプライマリーになり、すべての管理対象ノードが新規プライマリー CM への切り替えを完了するまで、数分間待機します。

CM バックアップ・プロセスの実行中 – 進行状況ログ・ファイルの表示

「プライマリー CM に設定」プロセスの実行中に、バックアップ CM から以下の手順を実行して、バックアップ CM がプライマリー CM になるまでの進行状況を確認することができます。

前提条件: ログ・ファイルを表示するには、接続先サーバーの IP が必要です。

1. バックアップ CM サーバーで Putty.exe セッションから CLI としてログインします。
2. CLI から、「Fileserver <IP> (IP 番号を入力) 3600」を実行します (例: fileserver 9.70.32.122 3600)。
3. GUI から、値 http://yourserver.x.x.x.com を入力します (コマンドの入力後に CLI 画面に表示されます)。例: http://joe.server.guardium.com (サーバー名はバックアップ CM サーバー)

    UI で、ファイルを選択するためのファイル・サーバー・ウィンドウが開きます。Sqlguard ログを選択します。
4. ファイル load_secondary_cm_sync_file.log を選択します (このファイルは、ステップ 3 のファイル・リストに表示されます)。これにより、バックアップ CM がプライマリー CM になるまでの進行状況を確認できるようになります。

    表示するログ・ファイルを探します。

    load_secondary_cm_sync_file.log に以下の行が表示されたら、CM バックアップ・プロセスは完了です。

    Import CM sync info - DONE
5. すべての管理対象ユニットが新規プライマリー CM から使用可能になるまで、約 10 分間待機します。

バックアップ CM がプライマリーになり、すべての管理対象ノードがバックアップ CM サーバーによって管理されるようになった後の手順

これで、以前の CM サーバーを起動できるようになります。以前のサーバーを稼働させたら、以下の手順を実行して、このサーバーをバックアップ CM サーバーとして追加します。

1. 以前のプライマリー CM をリブートします。
2. サーバーが起動したら、CLI としてログインします。
3. マネージャー・ユニット・タイプを削除します。delete unit type manager と入力します。
4. 完了すると、CLI から OK メッセージが表示されます。
5. 非常に重要: deleted unit type で成功メッセージと GUI 再始動メッセージが表示された後も、GUI が完全に再始動するまで約 5 分間待機してください。
6. 5 分経過したら、新規プライマリー CM にログインして、以前の CM を管理対象ユニットとして登録します。
7. 新規プライマリー CM に admin としてログインします。
8. 「設定」>「ツールとビュー」>「一元管理」を選択します。
9. 「新規登録」をクリックします。
10. 前のステップでリブートした以前のプライマリー CM の IP を入力します。
11. ポートとして 8443 と入力します。
12. 「保存」をクリックします。(重要: このボタンを 2 回クリックしないように注意してください。)

13. 以前のプライマリー CM が登録されるまで、しばらく待機します。
14. 以前のプライマリー CM を新規バックアップ CM に設定します。
15. 「バックアップ CM の指定」をクリックします。
16. 以前のプライマリー CM サーバーをクリックします。
17. 「適用」をクリックします。
18. これで、以前のプライマリー CM サーバーが新規バックアップ CM サーバーに設定されました。
19. 「一元管理」画面をリフレッシュすると、新規ユニット・タイプの「バックアップ CM」が定義されていることを確認できます。
20. これで、このタスクは完了です。

バックアップ CM プロセスの完了後のレポート・データ

バックアップ CM プロセスの完了後に以下のデータが欠落します。これは、プライマリー CM からセカンダリー CM への「初回」の切り替えにのみ関連します。

欠落データ:

1. 監査プロセスの結果
2. カスタム表データ
3. カスタム・レポート・データ
4. VA 結果
5. 分類結果
6. DSD 結果
7. CAS 結果
8. データマート・データ
9. 収集済みデータ
10. 資格データ

新規プライマリー CM でこれらのレポートを再実行した後、レポートにデータが再追加されます。旧プライマリー CM に切り替えると、これらのレポートのデータが表示されます。

親トピック: 一元管理機能の使用

調査センター

調査センターは統合サーバーの拡張機能です。 調査ユーザーは (一度定義されると) 選択した履歴日付のデータおよび結果をリストアし、フォレンジック調査を実行できます。 日にち (日付) をリストアしたら、調査ユーザーは標準の Guardium® UI を使用して、調査対象日付の範囲だけのレポートを定義し、表示できます。

各 Guardium アプライアンスは、アーカイブされたすべてのデータおよび結果のカタログを保持します。 このカタログにはアーカイブ、アーカイブのロケーション、およびそれらにアクセスするための資格情報についての情報が含まれています。 統合プロセスの一環として、カタログはコレクターからエクスポートされ、統合サーバーの完全なカタログにマージされます。 カタログが整っていれば、調査ユーザーはリストアのために希望する日付を選択することができ、これらの日付は自動的に調査センターにアップロードされてその調査ユーザーのビューにマージされます。 統合サーバーを通じてコレクターのカタログをマージするほかに、「設定」>「ツールとビュー」からカタログをエクスポートおよびインポートすることもできます。

ユーザーおよびロール

Guardium 統合サーバーには特別の調査ロール (inv) があります。inv ロールを持つユーザーは、履歴データに対してフォレンジック調査を実行することができます。

調査ユーザーは、大部分において、他のユーザーが使用するのと同じ照会定義およびレポート定義を利用します。 最大の違いは、調査ユーザーは自分の調査データベース用に選択したデータのみを表示することです (1 つの INV データベースを共用するように複数の調査機能を構成できます)。選択したデータはアーカイブからリストアできます。または、まだパージされていないデータの場合は現在のデータベースから表示できます。 調査ユーザーは、アーカイブされた監査プロセスの結果もリストア、および表示できます。

注意: ロール inv は、ユーザーを別の調査専用内部データベースに接続できる特別のロールです。 これはロール user と組み合わせる必要があり、一般に他のすべてのロールとは両立しません。

注: 調査ユーザーを正しく構成するには、ユーザーの姓を 3 つの調査データベース (「INV_1」、「INV_2」、または「INV_3」) のいずれかの名前に設定する必要があります (大/小文字の区別あり)。

調査ユーザーを作成するとき、使用する調査データベースにユーザー名が対応するよう、または使用する調査データベースを表す表記がユーザー名に含まれるようにすることをお勧めします。例えば、ユーザーが INV_1 データベースを使用する場合、ユーザー名を「john1」や「inv1」のようにします。

注: 「随時監査プロセスを実行」ボタンは、調査 (INV) ユーザーを除くすべてのユーザーが、すべてのレポート画面で使用できます。

監査プロセスおよび INV ロール

ユーザーが INV の場合は、監査プロセス・ファインダーがロールと所有権に従って監査プロセスを表示しますが、INV に所有されていないすべての監査プロセスについては、「コピー」または「新規」のみが許可されます。

ユーザーが INV の場合、監査プロセス定義メニュー画面では次のことが許可されます。

- 調査ユーザーおよび/または特定の E メール・アドレスのみが受信者として許可されます (INV 以外の通常ユーザー、グループ、ロールは受信者として許可されません)。
- 保存されたレポート監査タスク内の「イベントおよび追加列」ボタンは常に使用不可に設定されます。 いずれの API 自動化も指定できません。
- スケジュールは指定できません。INV データに対する監査プロセスは、「今すぐ実行する」ボタンを使用して手動でのみ実行できます。
- タイプが「レポート」の監査タスクのみ許可されます。
- 「アクティブ」は使用できません。「日数の保持」および「実行の保持」フィールドも使用できません。

ユーザーが INV でない場合、監査プロセス・ファインダーは (割り当てられたロールにかかわらず) 調査ユーザーが所有するどの監査プロセスも表示しません。

監査プロセスが INV のデータに対して実行されるときには、結果のタイトルの後ろに「Executed on Investigation center by」という言葉と INV ユーザーの名前が付きます。

結果には、実行時に調査データベースにデータがマウントされた日付とマウントされたデータのソース・ホストを指定したコメントが付けられます。

この結果は監査プロセス・ビルダーから、または結果のナビゲーション・リストで表示できます。

調査センターに対して実行された監査の結果はアーカイブできず、結果は調査データが破棄されるときに破棄されます。

調査コンテキスト

Guardium の調査センターは、並行して 1 から 3 つの調査期間 (INV_1、INV_2、および INV_3 と呼ばれる) をサポートします。これらはそれぞれ別々の履歴データを保持することができ、その期間のフォレンジック調査の手段を提供します。調査ユーザーを作成する際に、そのユーザーを調査データベースの 1 つと関連付けるためには、ユーザーの姓を INV_1、INV_2、または INV_3 のいずれかにする必要があります。 調査ユーザーの 1 人を使用して調査センターにログインすると、選択した調査期間がラベルに示されます。

GUI

調査ロールを持つユーザーには、調査センター固有の 2 つの追加のタブが表示されます。

- 「監査」タブでは、リストアされた監査プロセスの結果にアクセスできます。
- 「ボリューム管理」タブでは、ユーザーが調査期間を設定または変更したり、リストアする監査プロセスの結果を選択したり、調査の終了時にデータを破棄することができます。

調査センターの処理

- 監査結果のリストア
- リストア・ログの表示
- リストアされた監査結果の表示

監査結果のリストア

「監査プロセス・ビルダー」チェック・ボックスで、プロセスの結果をアーカイブするかどうかを指定できます。 すべての署名者によって署名されており、アーカイブのチェック・マークが付けられたプロセスの結果のみがアーカイブされます。 特定の実行の結果は圧縮され、zip されて保管され、ロケーションはカタログに記録され、監査結果のリストアで選択とリストアの際に使用されます。 Guardium 監査プロセスからアーカイブされた結果は、調査センターへリストアされますが、これには結果、ビュー、サインオフ証跡とならんで、これらの結果に関連付けられたコメントが含まれます。

inv ロールを持つユーザーとして Guardium インターフェースにログインした後で、以下を行います。

1. 「ボリューム管理」タブをクリックします。
2. 「監査結果のリストア」をクリックして、「リストアされた結果」パネルを開きます。前にリストアを実行していた場合は、現在リストアされている、使用中の結果がこのパネルに表示されます。 この時点で、「データの破棄」をクリックすると、以前にマウントされたすべての結果をアンマウントできます。
3. 「監査結果のリストア」をクリックして、「結果リストアの検索条件 (Results Restore Search Criteria)」パネルを開きます。
4. 検索する開始期間について、開始日を「開始日:」ボックスに入力します。
5. 検索する終了期間について、終了日を「終了日:」ボックスに入力します。
6. 結果セットをフィルター処理できるように、オプションで「ホスト名」、「監査プロセス」、または「実行番号」を入力します。
7. 「検索」をクリックし、結果セットを表示します。
8. 作成された結果セットから、リストアする結果の「選択」ボックスにチェック・マークを付けます。 選択処理を速めるために、「すべて選択」または「選択をすべて解除」をクリックすることもできます。
9. 選択した結果を復元するには、「復元」をクリックします。リストアする結果の数、およびデータ・セットがシステムに対してローカルであるかどうかによって、リストア・プロセスに時間がかかることがあります。
10. リストア・プロセスの進行状況は「リストア・ログの表示」でチェックできます。

リストア・ログの表示

リストア・ログは、過去のアーカイブ/リストアと、現在ログインしているユーザーの現在のリストアの試行およびフィルター処理を表示します。 このログにより、ユーザーはデータと監査結果の両方について、正常にリストアが行われたかどうかを確認できます。

inv ロールを持つユーザーとして Guardium インターフェースにログインした後で、「リストアのログ」をクリックして「マイ・リストア・ログ」を開きます。このパネルから、すべてのリストアの試行の状態を見ることができます。

リストアされた監査結果の表示

inv ロールを持つユーザーとして Guardium インターフェースにログインした後で、以下を行います。

1. 「監査」タブをクリックします。
2. 「結果のナビゲーション」リンクをクリックして、「監査プロセス・ファインダー」パネルを開きます。

3. 監査プロセスがある場合は、ドロップダウン・リストからプロセスを選択します。
4. 「表示」をクリックして、別のウィンドウを開き、使用可能なレポートを表示して監査結果を確認します。

親トピック: 統合および一元管理

Guardium システムの管理

管理タスクには、システムの正常性のモニターや、グループ、ドメイン、通知などの成果物の管理が含まれます。

- Guardium の管理
  Guardium® 管理者は、各種の管理およびメンテナンス・タスクを行います。
- 証明書
  機能が失われないように、証明書を定期的に確認してください。 新しい証明書を入手してインストールするには、CLI コマンドを使用します。
- ユニット使用状況レベル
  Guardium 環境内の頻繁に使用されているシステムとあまり使用されていないシステムを特定するには、ユニット使用状況レポートを使用します。
- カスタム・アップロード
  データベース・アクティビティー・モニター・コンテンツ・サブスクリプション (旧称は、データベース保護サブスクリプション・サービス)は、事前定義アセスメント・テスト、SQL ベースのテスト、CVE、APAR、およびグループ (データベース・バージョンやパッチなど) の保守をサポートしています。
- 「サービス状況 (Services Status)」パネル
  「サービス状況 (Services Status)」パネルは、CAS やアラート機能などのサービスの状況を確認して、必要な場合には各サービスをさらに調査するための、一元管理された場所です。「設定」 > 「ツールとビュー」 > 「サービス状況 (Services Status)」をクリックして、「サービス状況 (Services Status)」パネルを開きます。「サービス状況 (Services Status)」パネルが開かれるたびに、各サービスの状況が最新表示されます。
- アーカイブ、パージおよびリストア
  アーカイブおよびパージ操作は、スケジュールに基づいて実行する必要があります。 キャプチャーされた情報を監査のために保管するには、「データ・アーカイブ」と「結果アーカイブ」を使用します。このトピックの終わりで、Guardium での Amazon S3 へのアーカイブおよびバックアップについても説明しています。
- Guardium カタログ
  Guardium システムからデータをアーカイブすると、Guardium カタログは、すべてのアーカイブ・ファイルの送信先を追跡して、そのファイルを取得およびリストアできるようにします。
- バックアップとアーカイブの管理方法
  データを保持する方法を確立し、アクティビティー・ボリュームの制御を行い、データのアーカイブとパージのスケジューリングと、毎月のバックアップのスケジューリングを管理します。
- 結果のエクスポート (CSV、CEF、PDF)
  CSV、CEF、および PDF ファイルをワークフロー・プロセスで作成できます。この機能で、Guardium システム上に存在するこれらのファイルをすべてエクスポートします。
- 定義のエクスポート/インポート
  要件が同じまたは同じようなシステムが複数あり、一元管理を使用していない場合は、これらのシステムのソフトウェア・リリース・レベルが同じであれば、必要なコンポーネントを 1 つのシステムで定義し、その定義を他のシステムにエクスポートできます。
- 分散インターフェース
  この構成画面は、分散インターフェースを定義し、プロトコル・バッファー (.proto) ファイルを DIST_INT データベースにアップロードするために使用します。
- カスタム・クラスの管理
  アラートまたは評価で使用されるカスタム・クラスをアップロードして保守します。 カスタム・クラスを管理するには、「設定」 > 「カスタム・クラス」をクリックします。
- ブラウザーの SSL 証明書チャレンジを回避するためのアプライアンス証明書のインストール方法
  IBM Security Guardium CLI コマンドを使用して、証明書署名要求 (CSR) の作成、および Guardium システム上へのサーバー証明書、認証局 (CA) 証明書またはトラステッド・パス証明書のインストールを行います。
- GDPR 対応のための適用
  特定の設定が変更されると、Guardium によって何らかの個人識別情報 (PII) が収集されることがあります。
- 自己モニター
  Guardium ソリューションは、自己モニターを行って、中断を最小限に抑え、可能な場合は常に問題を自動的に修正します。
- グループ
  グループを使用すると、分類、ポリシー、照会の各定義を簡単に作成および管理できるほか、更新を S-TAP クライアントおよび GIM クライアントに展開することができます。アクセス・ポリシーのデータ・オブジェクトのグループを繰り返し定義するのではなく、オブジェクトをグループに入れると、簡単に管理できます。
- セキュリティー・ロール
  セキュリティー・ロールは、データ (グループ、照会、レポートなど) へのアクセスを許可したり、アプリケーション (「グループ・ビルダー」、「クエリー - レポート・ビルダー」、「ポリシー・ビルダー」、「CAS」、「セキュリティー・アセスメント」など) へのアクセスを許可するために使用します。
- 通知
  通知を作成するには、「アラート機能」および「アラート・ビルダー」を使用します。アラート・アクションに E メールまたはその他の通知が必要な場合は、以下の手順に従って、各タイプの通知について定義してください。
- リアルタイム・アラートの作成方法
  同じユーザーによるログインの失敗が 5 分以内で 3 回を超えた場合に、 データベース管理者にリアルタイム・アラートを送ります。
- カスタム・アラート・クラスの管理
  カスタムの受信者にアラートを送信するには、カスタム・アラート・クラスを使用します。 カスタム・クラスをアップロードしてから、「アラート・ビルダー」を使用して、アラート通知受信者としてカスタム・クラスを指定します。
- 事前定義アラート
  表で、「アラート・ビルダー」にある事前定義アラートについて説明します。
- スケジューリング
  汎用スケジューラーは、さまざまなタイプのタスク (アーカイブ、統合、 ワークフロー・オートメーションなど) をスケジュールに入れるために使用します。
- 別名
  レポートまたは照会で使用されるデータ値またはデータ・オブジェクトのシノニムを作成します。
- 日付とタイム・スタンプ
  カレンダー・ツールを使用して絶対日付を選択し、相対日付ピッカーを使用して現在時刻からの相対的な日付を選択します。
- ビルド期間
  ポリシー・ルールおよび照会条件によって、ユーザー定義の期間内にイベントが発生したかどうかをテストできます。
- コメント
  コメントは、定義とワークフロー・プロセス結果に適用されます。

- パッチのインストール方法
  1 つのパッチ、または複数のパッチをバックグラウンド・プロセスとしてインストールします。
- サポート・メンテナンス
  サポート・メンテナンス・フィーチャーは、パスワードで保護されており、技術サポートから指示があった場合にのみ使用できます。詳しくは、技術サポートにお問い合わせください。

Guardium の管理

Guardium® 管理者は、各種の管理およびメンテナンス・タスクを行います。

admin ロールが割り当てられたユーザーは、Guardium 管理者と呼ばれます。これは、admin ユーザー・アカウントとは明らかに異なります。

admin ロール特権

Guardium admin ロールは、そのロールに明示的に割り当てられていない特権があります。例えば、admin ロールを持つユーザーがプライバシー設定定義のリストを表示すると、Guardium システムに定義されているプライバシー・セットがすべて表示され、これらの定義をどれでも表示、変更、または削除できます。admin ロールのないユーザーがプライバシー・セットのリストにアクセスすると、自身の (すなわち自身が作成した) プライバシー・セットのみ表示されます。他にそのユーザーにも割り当てられている、セキュリティー・ロールが割り当てられたすべてのプライバシー・セットも表示されます。

CLI diag コマンド・アクセス

diag CLI コマンドを使用するには、追加のパスワードが必要です。これには admin ロールを持つどのユーザーのパスワードも使用できます。

自動アカウント・ロックアウトが有効な場合 (指定されている回数ユーザー・アカウントへのログインに失敗すると、そのアカウントをロックする機能)、admin ユーザー・アカウントで何度かログインに失敗すると、ロックされる場合があります。これが発生した場合は、unlock admin CLI コマンドを使用してこれをアンロックしてください。

注: アクセス・マネージャー (accessmgr) は、「ユーザー・ブラウザー」からアカウントをアンロックできます。「アクセス」 > 「アクセス管理」 > 「ユーザー・ブラウザー」をクリックして、「ユーザー・ブラウザー」を開きます。

admin ユーザー特権

admin ユーザーは、次のように admin ロールに付与されない追加の特権を持ちます。

- すべてのユーザーの To-Do リストへのアクセス
- インポートされた定義の所有者
- アクセス管理機能

admin ユーザーの To-Do リストに対する権限

To-do リストとは、監査プロセスの結果のユーザーへの配布を制御するワークフロー自動化機能です。admin ユーザーには、この領域で特殊な特権および責務があります。ユーザー・アカウントが無効化されると、そのユーザーに対するすべての監査プロセスの結果が admin ユーザーに自動的に再割り当てされます。ユーザーがその他の何らかの理由で使用不可の場合、監査プロセスの結果をそのユーザーの To-Do リストにインストールする (すなわち、結果を次の受信者にリリースする前にサインオフを待つ) ことができます。admin ユーザーは、あらゆるユーザーの To-Do リストを開き、そのユーザーが使用できるすべてのアクションを実行できます。 admin ユーザーが別のユーザーの To-Do リスト上のアクションを実行すると、その事実が監査プロセスのアクティビティー・ログに記録されます。例えば、「`admin` ユーザーがユーザー `x` の代わりに結果に署名した」。

インポートされる定義所有権

定義がエクスポートされると、すべてのロールが削除され、所有者が admin ユーザーに変更されます。これは、インポート側システムで定義の使用方法を制御する唯一の方法です。

アクセス管理と管理者

セキュリティー上の目的で、アクセス・マネージャーと管理者の職務は分離されています。管理ユーザーはアクセス・マネージャーの特権を持つことができず、逆もまた同様です。

次に admin ユーザーがログインすると、アクセス・マネージャーの機能が使用可能になります。これは、admin ユーザーのみに可能です (admin ロールを持つ他のユーザーには使用可能になりません)。

注:

既存の状態やアップグレードの結果として、同一ユーザーがこれら両方のロールを持つ場合があります。ただし、現行の使用では、これらの 2 つのロールを同一ユーザーに割り当てることはできません。

以前は、ユニットをアップグレードすると、accessmgr ロールが admin ユーザーに割り当てられ、accessmgr ユーザーが無効にされていました。

この状況では、accessmgr および admin を構成するには、admin としてログインして、accessmgr ユーザーを有効にしてから、accessmgr としてログインして (デフォルトの初期パスワードは `guardium`)、admin ユーザーから accessmgr ロールを削除します。

親トピック: Guardium システムの管理

証明書

機能が失われないように、証明書を定期的に確認してください。 新しい証明書を入手してインストールするには、CLI コマンドを使用します。

証明書の有効期限

証明書の有効期限が切れると、機能が失われます。show certificate warn_expire コマンドを定期的に実行して、期限が切れた証明書がないか確認してください。このコマンドは、6 カ月以内に有効期限が切れる証明書と、既に有効期限が切れた証明書を表示します。ユーザー・インターフェースでも、有効期限が切れる証明について通知されます。すべての証明書の概要を表示するには、コマンド show certificate summary を実行します。

証明書関連の CLI コマンドについて詳しくは、『証明書 CLI コマンド』を参照してください。
親トピック: Guardium システムの管理

ユニット使用状況レベル

Guardium 環境内の頻繁に使用されているシステムとあまり使用されていないシステムを特定するには、ユニット使用状況レポートを使用します。

「管理」 > 「レポート」 > 「ユニット使用状況」をクリックし、いずれかのレポートを選択して、ユニット使用状況レポートを開きます。

デフォルトのユニット使用状況レポートには、以下が含まれます。

- バッファー使用状況モニター
- CPU トラッカー
- エンタープライズ・バッファー使用状況モニター
- ユニット使用状況

使用状況パラメーター

ほとんどのパラメーターは、特定のユニットの、特定の時刻範囲における平均値です。「再始動の数」は、さまざまな PID に基づく、特定の時刻範囲におけるスニファーの再始動回数です。

サポートされるパラメーターは次のとおりです。

- 再始動の数
- スニファー・メモリー
- MySQL メモリーの比率
- 空きバッファー・スペース
- アナライザー・キュー
- ロガー・キュー
  制約事項: ロガー・キューの SQL 数は 500 に制限されています。500 を超える SQL を同時にこのキューに入れようとすると、キュー制限を超えた余分な SQL により `RA=-1` がログに記録されます。
- MySQL ディスク使用状況
- システム CPU 負荷
- システム変数ディスク使用状況
- 要求の数
- 完全な SQL の数
- 例外の数
- ポリシー違反の数
- ディスク使用量のクイック検索
- ドキュメントの数のクイック検索
- 未解析ログ要求

しきい値

パラメーターごとに 2 つのしきい値が定義されています。これらのしきい値が、3 つの使用状況レベル (低、中、および高) の区切りとなります。

使用状況レベルは以下のとおりです。

- 低: 値がしきい値 1 を下回っている場合
- 中: 値がしきい値 1 を上回り、しきい値 2 を下回っている場合
- 高: 値がしきい値 2 を上回っている場合

各ユニットの全体的な使用状況レベルもあります。各期間について、このレベルが、その期間におけるすべてのレベルの最高レベルとなります。

レポート作成

使用可能なユニット使用状況レポートを表示するには、「管理」 > 「レポート」 > 「ユニット使用状況」をクリックします。

「ユニット使用状況レベル」トラッキング・オプションを使用すると、カスタムの照会やレポートを作成できます。

カスタム・レポートおよび事前定義レポートでユニット使用状況データを使用するときは、別名を使用することをお勧めします。そうしなければ、使用状況レベルが「低」、「中」、「高」ではなく、数字 (1、2、3) で表示されます。

属性のリストには、以下が含まれます。

- ホスト名
- 期間の開始
- 再始動の数
- 再始動レベルの数
- スニファー・メモリー
- スニファー・メモリー・レベル
- MySQL メモリーの比率

- MySQL メモリーの比率のレベル
- 空きバッファー・スペース
- 空きバッファー・スペース・レベル
- アナライザー・キュー
- アナライザー・キュー・レベル
- ロガー・キュー
- ロガー・キュー・レベル
- MySQL ディスク使用状況
- MySQL ディスク使用レベル
- システム CPU 負荷
- システム CPU 負荷レベル
- システム変数ディスク使用状況
- システム変数ディスク使用状況レベル
- 全体のユニット使用状況レベル
- 要求の数
- 要求数のレベル
- 完全 SQL の数
- 完全 SQL 数のレベル
- 例外の数
- 例外数のレベル
- ポリシー違反の数
- ポリシー違反数のレベル
- 未解析ログ要求数
- 未解析ログ要求数のレベル

注: 各パラメーターには、値があり、さらに値およびしきい値に基づいて計算されるレベルがあります。

ユニット使用状況で使用可能なスループット情報

スループット・データは、コレクター・ユニットごとに収集されます。CM は、すべてのスループット・データを統合して、企業のカスタム表を作成します。この表は、事前定義の使用状況レポートに追加されます。

収集されるスループット情報は、以下のとおりです。

- (その期間における) 要求の数 (構造インスタンスから)
- (その期間における) 完全 SQL の数 (構造テキストから)
- 例外の数
- ポリシー違反の数

デフォルトでは、スループット情報は 1 時間ごとに収集されます。

ユニット使用状況を使用するための GuardAPI コマンドと CLI コマンド

GuardAPI:

- list_Utilization_Thresholds
- update_Utilization_Thresholds
  - *アナライザー・キュー*
  - *未解析ログ要求*
  - *空きバッファー・スペース*
  - *ロガー・キュー*
  - *Mysql ディスク使用状況*
  - *例外の数*
  - *完全な SQL の数*
  - *ポリシー違反の数*
  - *要求の数*
  - *再始動の数*
  - *Mysql メモリーの比率*
  - *ディスク使用量のクイック検索*
  - *ドキュメントの数のクイック検索*
  - *スニファー・メモリー*
  - *システム CPU 負荷*
  - *システム変数ディスク使用状況*
- reset_unit_utilization

CLI コマンド:

- store monitor gdm_statistics: ユニット使用状況に関する情報を取得するスクリプトを実行するためのスケジュールを設定します
- show monitor gdm_statistics: スケジュールを表示します

- ユニット使用状況データ処理の構成
  この手順では、ユニット使用状況データを処理して表示するための Guardium システムの構成方法を説明します。

親トピック: Guardium システムの管理

ユニット使用状況データ処理の構成

この手順では、ユニット使用状況データを処理して表示するための Guardium システムの構成方法を説明します。

このタスクについて

一元管理される環境の場合、ユニット使用状況情報の表示には、中央マネージャーでの 2 つのプロセスのスケジュールが必要です。すなわち、中央マネージャー・バッファー使用状況モニター用のデータのアップロード、およびユニット使用状況データの処理です。

スタンドアロン・システムの場合、ユニット使用状況情報の表示には、ユニット使用状況データの処理のスケジュールのみが必要です。スタンドアロン・システムを使用する場合、中央マネージャー・バッファー使用状況モニター用のデータ・アップロードをスケジュールする必要はありません。

手順

1. 一元管理される環境の場合、中央マネージャー・バッファー使用状況モニター・データをアップロードするためのスケジュールを中央マネージャーで定義します。
   a. 「レポート」 > 「レポート構成ツール」 > 「カスタム表ビルダー」にナビゲートします。
   b. 「カスタム表」画面で、「CM バッファー使用状況モニター」を選択し、「データのアップロード」をクリックして次に進みます。
   c. 「データのアップロード」画面で、「スケジュールの変更」をクリックして、中央マネージャー・バッファー使用状況モニター・データをアップロードするためのスケジュールを定義します。スケジュールの定義後、「保存」をクリックしてから、「先頭に戻る」をクリックして「データのアップロード」画面に戻ります。 1 時間に 1 回プロセスを実行するようにスケジュールするのが、最初は多くのデプロイメントで妥当な方法ですが、使用可能なリソースまたはデータの現行性のニーズに合わせて間隔を調整できます。
   重要: ユニット使用状況レポートで最新のデータを使用できるようにするには、ユニット使用状況データを処理する前に、バッファー使用状況モニター・データを処理するスケジュールを定義します。また、バッファー使用状況モニター・データは、特定時間に正確に実行されるようにスケジュールしないでください。
   ベスト・プラクティス: 正時の 10 分後にバッファー使用状況モニター・データを処理し、正時の 40 分後にユニット使用状況データを処理するスケジュールを定義します。
   d. 「データのアップロード」画面で、オプションとして「今すぐ 1 回実行」をクリックして即時にデータをアップロードします。
2. 一元管理される環境またはスタンドアロン・システムの場合、ユニット使用状況データを処理するためのスケジュールを定義します。 一元管理される環境では、中央マネージャーでユニット使用状況スケジュールを定義するだけで済みます。
   a. 「管理」 > 「ユニット使用状況」 > 「ユニット使用状況レベル」にナビゲートします。
   b. 「ユニット使用状況レベル」画面で、「スケジュールの変更」をクリックして、ユニット使用状況データを処理するためのスケジュールを定義します。スケジュールの定義後、「保存」をクリックしてから、「先頭に戻る」をクリックして「ユニット使用状況レベル」画面に戻ります。 1 時間に 1 回プロセスを実行するようにスケジュールするのが、最初は多くのデプロイメントで妥当な方法ですが、使用可能なリソースまたはデータの現行性のニーズに合わせて間隔を調整できます。
   重要: ユニット使用状況レポートで最新のデータを使用できるようにするには、バッファー使用状況モニター・データを処理した後にユニット使用状況データを処理するスケジュールを定義します。
   ベスト・プラクティス: 正時の 10 分後にバッファー使用状況モニター・データを処理し、正時の 40 分後にユニット使用状況データを処理するスケジュールを定義します。
   c. 「ユニット使用状況レベル」画面で、オプションとして「今すぐ 1 回実行」をクリックして即時にデータを処理します。

タスクの結果

上記のステップの完了後、「管理」 > 「レポート」 > 「ユニット使用状況」にナビゲートしてユニット使用状況レポートを表示します。集中管理される環境では、中央マネージャーとその管理対象ユニットのデータが利用可能です。スタンドアロン・システムの場合、その個別システムのデータのみが使用可能です。スケジュールを定義する際に「今すぐ 1 回実行」オプションを使用しなかった場合、ユニット使用状況レポートが最新データで更新される前に、それらのプロセスが実行されるまで待つ必要があります。

親トピック: ユニット使用状況レベル

カスタム・アップロード

データベース・アクティビティー・モニター・コンテンツ・サブスクリプション (旧称は、データベース保護サブスクリプション・サービス)は、事前定義アセスメント・テスト、SQL ベースのテスト、CVE、APAR、およびグループ (データベース・バージョンやパッチなど) の保守をサポートしています。

アップロードは、情報を常に最新の状態に維持し、業界のベスト・プラクティスにおいて、新たに発見された脆弱性から保護するために使用されます。 更新の配布は、四半期ごとに行われます。

カスタム・アップロードを使用して、以下のものをアップロードします。 DPS 更新ファイル、Oracle JDBC ドライバー、MS SQL Server JDBC ドライバー、および Db2 for z/OS ライセンス jar。

注: 事前定義された Guardium® グループと同じ名前のカスタム・グループが存在する場合、アップロード・プロセスによって、事前定義グループの名前の前に「Guardium」が追加されます。

1. 「強化」 > 「脆弱性評価」 > 「カスタム・アップロード」をクリックして「カスタム・アップロード」を開きます。
2. 「DPS のアップロード」で、「参照」をクリックして、アップロードするファイルを見つけ、選択します。
   注: どのファイルがアップロードされたかを確認するには、「DPS のインポート」ペインを参照してください。
3. 「Db2 z/OS ライセンス jar のアップロード」で、「参照」をクリックして、ファイルを見つけ、選択します。
4. 「Oracle JDBC ドライバーのアップロード」または「MS SQL Server JDBC ドライバーのアップロード」を使用して、オープン・ソース・ドライバーをアップロードします。 アップロードの後、「データ・ソース・ファインダー」にデータベースが追加されています。ドライバーは、1 つずつアップロードしてください。
   注: Oracle データ・ダイレクト・ドライバーまたは MS SQL データ・ダイレクト・ドライバーよりもオープン・ソース・ドライバーを使用することが推奨される場合が 2 種類あります。
   a. Windows Authentication for MS SQL Server をサポートする場合。その他のすべての用法では、Guardium アプライアンスにプリロードされたデータ・ダイレクト・ドライバーを使用することで十分です。
   b. Oracle バージョン 10 以上で「値変更のトラッキング」アプリケーションを使用する場合。トリガーの代わりにストリームの使用をサポートするには、オープン・ソース・ドライバーが推奨されます。

   キーワードを使用して、オープン・ソース JDBC ドライバーを検索してダウンロードします (例: *open source JDBC driver for MS SQL*)。

5. 中央マネージャーを使用して、.jar ファイルを管理対象ユニットに配布します。ファイルのアップロードが成功したら、中央マネージャーと管理対象ユニットで GUI を再始動する必要があります。

注:

ユニット間で相互に定義をエクスポートしたり、インポートしたりする場合は、サブスクライブしたグループがエクスポートされないように注意してください。 サブスクライブしたグループを参照する定義をエクスポートする場合、ユーザーは参照されるサブスクライブしたグループをすべて、インポート側ユニット (フェデレーテッド環境では中央マネージャー) にインストールしておく必要があります。

DB2® z/OS® ライセンス JAR ファイルをアップロードする場合、ライセンスは GUI を再始動した後に有効になります。

注: 何らかの理由 (例えば、サーバーの再始動や GUI の再始動など) で DPS が停止した場合、30 分間待ってから DPS アップロード・プロセスを再開することを推奨しています。

最新の Oracle DataDirect ドライバーを使用して、Oracle サーバーで ASO を有効にする

最新の Oracle DataDirect ドライバーを使用して Oracle サーバーで ASO を有効にする場合、以下を参照してください。

SQLNET.CRYPTO_CHECKSUM_SERVER = 必須

SQLNET.ENCRYPTION_SERVER = 必須

SQLNET.ENCRYPTION_TYPES_SERVER = (AES256, AES192, AES128)

#SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER = (SHA256)

SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER = (SHA1)

Oracle JDBC ドライバーは動作し、接続プロパティーの指定は不要です。

ただし、最新の Oracle JDBC ドライバーを Oracle からダウンロードする必要があります。ファイル名は ojdbc7.jar です。キーワードを使用して、オープン・ソース JDBC ドライバーを検索してダウンロードします (例: open source JDBC driver for Oracle)。次に、Guardium カスタム・アップロード機能を使用して、そのドライバーをアプライアンスにアップロードします。

Oracle DataDirect ドライバーを引き続き使用する場合、データ・ソースに対する接続プロパティーを指定する必要があります。

Oracle DataDirect ドライバーの接続プロパティーを定義する場合、以下を使用します。

DataIntegrityLevel=required;EncryptionLevel=required;DataIntegrityTypes=(MD5、SHA1)

注: 現在の Oracle DataDirect ドライバーは SHA-256 をサポートしていません。このため、SHA-1 を使用する必要があります。これが、sqlnet.ora 参照 (#SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER = (SHA256)) をコメント化する必要があった理由です。ただし、Guardium カスタマーが SHA-256 を使用して接続する必要がある場合、代わりに Oracle JDBC ドライバーを使用する必要があります。

データ・ダイレクトのリファレンス:

https://www.progress.com/documentation/datadirect-connectors

一連のコマンドのリファレンスとして、Oracle データベースの JDBC のユーザー・ガイド (PDF) をダウンロードしてください。

「カスタム・アップロード」機能を使用してデータ・ソース・アップロード・ファイルを作成および保存するときのタブ区切りファイル (.TXT) の使用

コンマ区切りファイル構造 (.CSV) を使用する場合、いずれかの列値にコンマが含まれていると、意図したとおりに動作しません。

以下の手順を行います。

1. EXCEL を使用している場合、タブ区切り (.TXT) ファイルとしてファイルを保存します。
2. OpenOffice または Libre Office を使用している場合、タブ区切り文字付きの (.CSV) ファイルとして保存します。
3. admin としてログインし、「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「カスタム・アップロード」をクリックして、「カスタム・アップロード」を開きます。
4. 「CSV をアップロードしてデータ・ソースを作成/更新」で、「参照...」をクリックして、タブ区切りファイルを選択します。

CSV のアップロード・メニューにより CSV をアップロードしてデータ・ソースを作成する

データ・ソース情報を格納するタブ区切り .TXT 形式ファイルを作成するには、以下の手順を行います。これにより、このタブ区切り .TXT ファイルは、Guardium アプリケーション内の「カスタム・アップロード」機能によって多くのデータ・ソース・タイプに使用できます。

データ・ソースをインポートする機能の使用は、必ずしもすべての Guardium ソフトウェア・リリースで互換性があるわけではありませんでした。この手順によって、あらゆるデータ・ソースのアップロードが可能になります。

以下のリストは、.TXT 形式のタブ区切りデータ・ソース・アップロード・ファイルを作成するときに Excel スプレッドシートに追加する必要があるヘッダー列です。

列の値 (.CSV データ・ソース・アップロード・ファイルで受け入れられる)

表 1. create_datasource

パラメーター	記述

パラメーター	記述
application	必須。 データ・ソースの定義対象となるアプリケーションを指定します。 次のいずれかでなければなりません。 ChangeAuditSystem Access_policy MonitorValues DatabaseAnalyzer AuditDatabase CustomDomain Classifier AuditTask SecurityAssessment Replay Stap_Verification
compatibilityMode	互換モード: 選択項目は「デフォルト」と「MSSQL 2000」です。表のモニター時に使用する互換モードをプロセッサーに指示します。
conProperty	オプション。 このデータ・ソースとの JDBC 接続を確立するために JDBC URL に追加の接続プロパティーを含める必要がある場合にのみ使用します。 使用するフォーマットは、「property=value」である必要があります。このとき、プロパティーと値の各ペアはコンマで区切ります。 Roman8 をデフォルトの文字セットとする Sybase データベースの場合、次のプロパティーを入力します。charSet=utf8
customURL	オプション。 データ・ソースに対する接続文字列。これを入力しない場合は、前回入力したフィールドのホスト、ポート、インスタンス、プロパティーなどを使用して接続が行われます。 これは、例えば Oracle Internet Directory (OID) の接続を作成するときに便利です。
dbInstanceAccount	オプション。 CAS によって使用されるデータベース・アカウント・ログイン名 (ソフトウェア所有者)
dbInstanceDirectory	オプション。 CAS によって使用される、データベース・ソフトウェアがインストールされたディレクトリー
dbName	オプション。 Db2 または Oracle データ・ソースの場合、スキーマ名を入力します。他の場合は、データベース名を入力します。
description	オプション。データ・ソースの詳細説明。
host	必須。 ホスト名または IP アドレスを入力できます。
name	必須。 システム上のデータ・ソースに固有の名前を付けます。
owner	必須。 データ・ソースを所有する Guardium ユーザー・アカウントを指定します。
password	オプション。所有者のパスワード。使用する場合、ユーザーも使用する必要があります。
port	オプション (整数)。ポート番号。
serviceName	Oracle、Informix®、Db2、および IBM® ISeries の場合は必須。Db2 データ・ソースではデータベース名を入力します。それ以外ではサービス名を入力します。
severity	オプション。データ・ソースの重大度分類 (あるいは影響レベル)。
shared	オプション (ブール値)。他のアプリケーションと共有する場合は **true** に設定します。データ・ソースを他のユーザーと共有する場合は、GUI からロールを割り当てる必要があります。

パラメーター	記述
type	必須。 データ・ソース・タイプを指定します。次のいずれかでなければなりません。 Db2 Db2 for i Db2 for z/OS Informix MS SQL Server MS SQL サーバー (DataDirect) MySQL NA Netezza Oracle (DataDirect) Oracle (サービス名) Oracle (SID) PostgreSQL Sybase Sybase IQ Teradata アプリケーションが CustomDomain または Classifier である場合、以下も使用できます。 TEXT TEXT:FTP TEXT:HTTP TEXT:HTTPS TEXT:SAMBA
user	オプション。データ・ソースのユーザー。使用する場合、パスワードも使用する必要があります。
environmentTitle	クラウド・データベース・サービスの保護に必要。アカウント名。
region	クラウド・データベース・サービスの保護に必要。AWS 領域。
objectLimit	クラウド・データベース・サービスの保護に必要。分類プロセスで検出され、監査対象オブジェクトのリストに自動的に追加されるオブジェクトの最大数。クラウド・データベース・サービス保護を参照してください。
primaryCollector	クラウド・データベース・サービスの保護に関連。クラウド・データベースから監査データを抽出するコレクター。

注:

1. 各列名は、タブ区切り形式 (.TXT) ファイルとして保存される Excel スプレッドシートに含める必要があります。
2. 作成されるデータ・ソース名 (データ・ソースを検索するときに表示されるもの) は、名前列とタイプ列の両方で構成されます。
3. アップロード・ファイルは、列タブ区切り形式のファイル・タイプとして保存される必要があります。

txt ファイルを CSV テキスト形式で作成およびアップロードしてデータ・ソースのデータを追加するステップ

1. データ・ソース・インポート機能をサポートする以下のヘッダーおよびデータ・ソースのデータを使用して、Excel スプレッドシート・ファイルを作成してタブ区切り .TXT ファイルとして保存します。
2. Guardium アプリケーションへのアップロード用の PC または UNIX/Linux デバイスに対して .txt ファイルを作成して保存します。
3. admin としてログインし、「強化」 > 「構成変更制御 (CAS アプリケーション)」 > 「カスタム・アップロード」をクリックして、「カスタム・アップロード」を開きます。
4. 「CSV をアップロードしてデータ・ソースを作成/更新」で、「参照」 をクリックして、タブで区切られたデータ・ソース情報を格納している .txt ファイルを選択します。
5. 「アップロード」をクリックします。

.txt ファイルからアップロードされた値を示すメッセージが表示されます。

1. **「新規」**: ファイルをアップロードするたびに (ファイルを保存して新規データ・ソース・メンバーを追加した場合)、これらのメンバーは「新規」状況になります。
2. **「更新」**: 同じデータ・ソースに変更を加えてアップロードすると、「更新」の状況が付与されます。
3. **「不合格」**: 失敗したデータ・ソースまたはエラーが表示されます

親トピック: Guardium システムの管理

「サービス状況 (Services Status)」パネル

「サービス状況 (Services Status)」パネルは、CAS やアラート機能などのサービスの状況を確認して、必要な場合には各サービスをさらに調査するための、一元管理された場所です。「設定」 > 「ツールとビュー」 > 「サービス状況 (Services Status)」をクリックして、「サービス状況 (Services Status)」パネルを開きます。 「サービス状況 (Services Status)」パネルが開かれるたびに、各サービスの状況が最新表示されます。

例えば、5 分間でログインの失敗が 3 回を超えた場合に常にリアルタイム・アラートを送信するポリシーを設定するとします。この潜在的な侵入から保護するために、ポリシーがインストールされていて、アラート機能がオンになっていることを確認する必要があります。

「サービス状況 (Services Status)」 パネルを使用して、これらのサービスの両方が適切に構成されていることを確認します。

いずれかのサービスをクリックすると、その構成ページに移動します。そのページで、サービスのオンとオフの切り替え、サービスの再始動、サービスの構成など関連する操作を実行できます。

何らかの理由でポリシーが正しくインストールされていなかった場合は、「設定」 > 「ツールとビュー」 > 「ポリシー・インストール」をクリックして、「ポリシー・インストーラー」に進み、現在インストールされているポリシーを表示して、必要な変更を加えます。

それぞれのサービスに、以下のいずれかのアイコンが表示されます。

- サービスが実行中/スケジュール済みの場合:
- サービスが一時停止している場合:
- サービスがオフの場合:

親トピック: Guardium システムの管理

アーカイブ、パージおよびリストア

アーカイブおよびパージ操作は、スケジュールに基づいて実行する必要があります。 キャプチャーされた情報を監査のために保管するには、「データ・アーカイブ」と「結果アーカイブ」を使用します。このトピックの終わりで、Guardium での Amazon S3 へのアーカイブおよびバックアップについても説明しています。

「データ・アーカイブ」および「結果アーカイブ」は、「管理」 > 「データ管理」をクリックすると見つかります。

- 「データ・アーカイブ」は、Guardium システムによって一定期間内にキャプチャーされたデータをバックアップします。データ・アーカイブを構成する際には、パージ操作も構成できます。 通常、データは毎日、1 日の終わりにアーカイブされます。こうすることで、災害発生時に失われるのはその当日のデータだけになります。データのパージは、アプリケーションごとに異なり、ビジネス要件や監査要件によってかなり大きな違いがあります。 ほとんどの場合、データは Guardium システム上に 6 カ月以上保持することができます。
- 「結果アーカイブ」は、監査タスクの結果 (レポート、アセスメント・テスト、エンティティー監査証跡、プライバシー・セット、および分類プロセス) のほか、表示およびサインオフの証跡、ワークフロー・プロセスからの調整コメントをバックアップします。 結果セットは、ワークフロー・プロセスの定義に従ってシステムからパージされます。

統合環境では、データはコレクター、アグリゲーター、またはその両方のロケーションからアーカイブできます。 データは、一度だけアーカイブするのが最も一般的です。 アーカイブ元となるロケーションは、ユーザーの要件に応じて異なります。

スケジュールされたエクスポート操作により、Guardium® コレクター・ユニットから Guardium 統合サーバーにデータが送信されます。統合サーバーは、独自のスケジュールでインポート操作を実行し、統合プロセスを完了します。 この一方または両方のユニットにおいて、アーカイブ操作とパージ操作がスケジュールされます。これらの操作は (スペースを解放し、内部データベースのアクセス操作を高速化する目的で) 定期的にデータをバックアップしてパージします。

アーカイブ・ファイルは、SCP または FTP プロトコルを使用して送信することもできますし、EMC Centera や TSM ストレージ・システムが構成されている場合には、そこに送信することもできます。 各 Guardium システムには、単一のアーカイブ構成を定義できます。

Guardium のアーカイブ機能では、改ざんできない、署名付きの暗号化ファイルが作成されます。 生成されたアーカイブ・ファイルの名前を変更しないでください。 アーカイブ操作およびリストア操作は、アーカイブ処理中に作成されるファイル名に依存します。

アーカイブおよびエクスポートの各アクティビティーでは、システム共有パスワードを使用して暗号化データ・ファイルが作成されます。 あるシステムで暗号化した情報を、別のシステムでリストア可能にするには、アーカイブ側の システムでそのファイルを作成したときに使用した共有パスワードが、リストア側のシステムに必要です。

データをアーカイブする際には、必ず操作が正常に完了したことを確認してください。 これを行うには、「管理」 > 「レポート」 > 「データ管理」 > 「統合/アーカイブ・ログ」をクリックして、「統合/アーカイブ・ログ」を開きます。アーカイブ操作ごとに複数のアクティビティーがリストされていて、各アクティビティーの状況は完了になっているはずです。

「管理」 > 「データ管理」 > 「システム・バックアップ」をクリックして、システム・バックアップ・タスクを実行します。CLI からバックアップ・タスクを実行することもできます。詳しくは、ファイル処理 CLI コマンドを参照してください。

デフォルト・パージ

- パージのデフォルト値は 60 日です。
- デフォルト・パージ・アクティビティーは、毎日午前 5:00 にスケジュールされます。
- 新規インストールでは、デフォルトの値とアクティビティーに基づくデフォルト・パージ・スケジュールがインストールされます。
- ユニット・タイプを「管理対象ユニット」に変更するか、「スタンドアロン・ユニット」に戻すと、デフォルト・パージ・スケジュールが適用されます
- パージ・スケジュールは、アップグレード中は影響を受けません
- 多数のレコード (1000 万件以上) をパージするときは、バッチ・サイズを大きく (500k から 1M) 設定するのが最も有効な方法です。 小さめのバッチ・サイズまたは NULL を使用すると、パージに要する時間が何時間も余計にかかります。 小規模のパージは短時間で終了するため、バッチ・サイズを大きく設定する方法はパージが大規模である場合にのみ該当します。

注: バッチ・サイズの設定は、UI では実行できません。GuardAPI コマンド grdapi set_purge_batch_size batchSize を使用して、バッチ・サイズを設定します。

アーカイブに保存されなかった日を確認する方法

「レポート・ビルダー」を使用して、アーカイブの日付を示す全ファイルのリストを表示します。「管理」 > 「レポート」 > 「レポート・ビルダー」をクリックして、「レポート・ビルダー」を開きます。「照会」メニューから、「ロケーション・ビュー」を選択します。このレポートに組み込まれなかった日付は、アーカイブに保存されなかった日付です。必要であれば、リストに組み込まれていない日付のアーカイブを実行してください。

データ・アーカイブおよびパージの構成

1. 「管理」 > 「データ管理」 > 「データ・アーカイブ」をクリックして、「データ・アーカイブ」を開きます。
2. アーカイブするには、「アーカイブ」チェック・ボックスにチェック・マークを付けます。 「構成」パネルに、追加のフィールドが表示されます。
3. 「次の期間を経過したデータをアーカイブ」で、値を入力して、メニューから時間の単位を選択します。前日のデータからデータのアーカイブを開始するには、値 1 を入力し、メニューから「日」を選択します。
4. 何日分のデータをアーカイブするかを制御するには、「次の期間を経過したデータを無視」を使用します。ここに指定する値は、「次の期間を経過したデータをアーカイブ」」の値よりも大きくなければなりません。
   注: このフィールドをブランクのままにすると、「次の期間を経過したデータをアーカイブ」で指定した値より古いすべての日のデータがアーカイブされます。 つまり、毎日アーカイブを行い、30 日より古いデータをパージする場合には、(31 日目にパージされるまで) 日次データを 30 回アーカイブすることになります。
5. SQL 文字列からの値をアーカイブ・データに含めるには、「値のアーカイブ」チェック・ボックスにチェック・マークを付けます。このボックスをクリアすると、値はアーカイブでは疑問符文字に置き換えられます (したがって、リストア操作以後、これらの値を使用できなくなります)。
6. 「プロトコル」オプションを選択して、適切な情報を入力します。Guardium システムの構成方法によっては、これらのボタンの 1 つ以上が選択できない場合があります。 アーカイブおよびバックアップのストレージ方式の構成方法については、show storage-system および store storage-system コマンドの説明を参照してください。
7. 選択したストレージ方式に応じて、適切な手順を実行します。
   - SCP または FTP アーカイブまたはバックアップの構成
   - EMC Centera アーカイブまたはバックアップの構成
   - TSM アーカイブまたはバックアップの構成
8. 「パージ」チェック・ボックスにチェック・マークを付けて、パージ操作を定義します。

   重要: このパージ構成は、データ・アーカイブとデータ・エクスポートの両方で使用されます。 ここで行った変更は、すべてのデータ・エクスポートの実行に適用されます。逆もまた同様です。パージがアクティブになっていて、データ・エクスポートとデータ・アーカイブの両方が同じ日に実行される場合には、最初に実行された操作が古いデータをすべてパージした後に 2 番目の操作が実行されます。

   そのため、データ・エクスポートとデータ・アーカイブが共に構成されているときは常に、エクスポート基準経過日数とアーカイブ基準経過日数の両方よりもパージ基準経過日数の方が大きくなければなりません。

9. データをパージする場合は、「次の期間を経過したデータをパージ」フィールドを使用して、パージ操作の対象となる開始日を、当日 (0 日) よりさかのぼった日数、週数、または月数として指定します。指定した日、およびそれより古いすべての日のデータは、注に示す例外を除いてすべてパージされます。パージ開始日に指定する値は、「次の期間を経過したデータをアーカイブ」に指定した値よりも大きくなければなりません。 また、データ・エクスポートがアクティブになっている場合、ここに指定するパージ開始日は、「次の期間を経過したデータをエクスポート」の値よりも大きくなければなりません。『重要』の注を参照してください。
   注:

   それ以前の操作によってまだアーカイブ/エクスポートされていないデータをパージする際には、警告は出されません。

   パージ操作では、リストア操作で指定される「リストアしたデータをパージしない」時間枠の範囲内に経過日数が入っているリストア・データはパージされません。

10. 「スケジューリング」セクションを使用して、この操作を定期的に実行するためのスケジュールを定義できます。
11. 「保存」をクリックして、構成の変更を保存します。システムはそのロケーションにテスト・データ・ファイルを送信することにより、構成の検証を試みます。
    - 操作が失敗すると、エラー・メッセージが表示され、構成は保存されません。
    - 操作が成功すると、構成が保存されます。
12. 「今すぐ 1 回実行」をクリックして、操作を 1 回実行します。

SCP または FTP アーカイブまたはバックアップの構成

アーカイブまたはバックアップの構成パネルで SCP または FTP を選択した後、以下の情報を指定する必要があります。

1. 「ホスト」に、アーカイブ・データを受信するホストの IP アドレスまたはホスト名を入力します。
2. 「ディレクトリー」に、データの格納先ディレクトリーを指定します。FTP または SCP のどちらのファイル転送方式を使用するかに応じて、指定方法が異なります。
   - FTP の場合: FTP アカウントのホーム・ディレクトリーに対する相対パスでディレクトリーを指定します。
   - SCP の場合: 絶対パスとしてディレクトリーを指定します。
3. 「ポート」に、SCP および FTP を介したファイル送信に使用できるポートを指定します。ssh/scp/sftp のデフォルトのポートは 22 です。FTP のデフォルトのポートは 21 です。
   注: ポートにゼロ (0) が表示される場合、デフォルト・ポートが使用されていて、変更の必要がないことを示します。
4. 「ユーザー名」および「パスワード」に、SCP サーバーまたは FTP サーバーにログオンするユーザーの資格情報を入力します。 このユーザーは、「ディレクトリー」で指定したディレクトリーに対する書き込み権限/実行権限を保持していなければなりません。

   Windows の場合、ドメイン・ユーザーは domain¥user の形式にしてください。

5. 「保存」をクリックして構成を保存します。

EMC Centera アーカイブまたはバックアップの構成

このバックアップまたはアーカイブ・タスクでは、ファイルがオフサイトの EMC Centera ストレージ・システムにコピーされます。 EMC からのライセンスのほか、ユーザー名およびパスワードが必要です。 このタスクでは、次の 4 つのメイン・アクションが必要です。

1. ネットワーク上で EMC Centera でのアカウントを確立する (IP アドレスおよび ClipID が必要)。

2. Guardium システムからデータまたは構成ファイルのいずれかまたは両方を構成する。
3. ライブラリーを定義およびエクスポートする。
4. ファイルが EMC Centera ストレージ・システムに格納されていることを確認する。

CLI アクション

CLI から以下のコマンドを実行します。

```
store storage-system centera backup ON
show storage-system
```

Centera アーカイブまたはバックアップの構成

「管理」 > 「データ管理」 > 「システム・バックアップ」をクリックして、「システム・バックアップ」を開きます。「EMC Centera」を選択します。以下の情報を指定する必要があります。

1. 「保持」に、データを保持する日数を入力します。最大値は 24855 (68 年) です。それより長くデータを保存する場合は、後ほどデータをリストアして再度保存します。
2. 「Centera プール・アドレス」に、Centera プール接続文字列を入力します (例: 10.2.3.4,10.6.7.8?/var/centera/us1_profile1_rwe.pea txt)。
   注: この IP アドレスおよび .PEA ファイルは、EMC Centera から取得します。 パスを構成する際には、疑問符が必須です。 「.../var/centera/...」を含むパス名でなければバックアップが失敗するため、このパス名は重要です。 .PEA ファイルは、Centera バックアップ要求ごとにアクセス権、ユーザー名、およびパスワード認証を提供します。
3. 「PEA ファイルのアップロード」をクリックして、接続文字列に使用される Centera PEA ファイルをアップロードします。この場合にも「Centera プール・アドレス」が必要です。
   注: 「このアドレスのプールを開くことができません」というメッセージが表示される場合、Guardium システムのホスト名のサイズを確認してください。Centera では、長さが 4 文字未満のホスト名の使用時に、タイムアウト問題が報告されています。
4. 「保存」をクリックして構成を保存します。システムは、指定された接続文字列を使用してプールを開くことにより、Centera アドレスの検証を試みます。操作が失敗すると、通知メッセージが表示され、構成は保存されません。
5. 「今すぐ 1 回実行」をクリックし、ダウンロードした .PEA ファイルを使用してバックアップを実行します。

ファイルが EMC Centera にコピーされていることを確認します。このタスクでは、対象のファイルの名前と ClipID が必須です。

TSM アーカイブまたはバックアップの構成

TSM サーバーへアーカイブする前には、CLI を使用して Guardium システムに dsm.sys 構成ファイルをアップロードする必要があります。 import tsm config CLI コマンドを使用します。 アーカイブまたはバックアップの構成パネルで TSM を選択した後、以下の情報を指定します。

1. 「パスワード」に、TSM サービスを要求するためにこの Guardium システムが使用する TSM パスワードを入力し、「パスワードの再入力」ボックスに再び入力します。
2. オプションで、dsm.sys ファイルの servername 項目と一致するように、「サーバー名」を入力します。
3. オプションで、「ホスト」名を指定します。
4. 「保存」をクリックして構成を保存します。「保存」ボタンをクリックすると、システムは dsmc アーカイブ・コマンドを使用してサーバーにテスト・ファイルを送信することにより、TSM 宛先の検証を試みます。操作が失敗すると、通知メッセージが表示され、構成は保存されません。
5. アーカイブまたはバックアップ手順に戻って、構成を完了します。

結果アーカイブの構成

1. 「管理」 > 「データ管理」 > 「結果アーカイブ (監査)」をクリックして、「結果アーカイブ」を開きます。
2. 「次の期間を経過した結果をアーカイブ」に続くファイルに、アーカイブ操作の開始日を、当日 (0 日) よりさかのぼった日数、週数、または月数として指定します。前日のデータから結果のアーカイブを開始するには、値 1 を入力し、リストから「日」を選択します。
3. オプションで、「次の期間を経過した結果を無視」に続くフィールドを使用して、何日分の結果をアーカイブするかを制御します。 ここに指定する値は、「次の期間を経過した結果をアーカイブ」の値よりも大きくなければなりません。
4. ラジオ・ボタンでストレージ方式を選択します。Guardium システムの構成方法によっては、これらのボタンの 1 つ以上が選択できない場合があります。アーカイブおよびバックアップのストレージ方式の構成方法については、構成および制御 CLI コマンドで show storage-system コマンドと store storage-system コマンドの説明を参照してください。
   - EMC CENTERA
   - TSM
   - SCP
   - FTP
5. 選択したストレージ方式に応じて、適切な手順を実行します。
   - SCP または FTP アーカイブまたはバックアップの構成
   - EMC Centera アーカイブまたはバックアップの構成
   - TSM アーカイブまたはバックアップの構成
   - Guardium での Amazon S3 へのアーカイブおよびバックアップ
6. 「スケジューリング」セクションを使用して、この操作を定期的に実行するためのスケジュールを定義できます。
7. 「保存」」をクリックすると、構成の変更が検証されて、保存されます。システムはそのロケーションにテスト・データ・ファイルを送信することにより、構成の検証を試みます。
   - 操作が失敗すると、エラー・メッセージが表示され、構成は保存されません。
   - 操作が成功すると、構成が保存されます。
8. 「今すぐ 1 回実行」をクリックして、操作を 1 回実行します。

データのリストア

このシステムが、リストアするアーカイブを生成したシステムではない場合、カタログ・アーカイブを使用してカタログでロケーション・エントリーを作成し、「追加」(参照:「Guardium カタログ」) または「GuardAPI」(参照:「CLI および API」 > 「GuardAPI リファレンス」 > 「GuardAPI カタログ・エントリー関数」) をクリックする必要があります。データのリストアが開始されると、この情報を使用してファイルがシステムに転送され、その後データが処理されます。

データをリストアする前に

- TSM からリストアする前には、CLI を使用して Guardium システムに dsm.sys 構成ファイルをアップロードする必要があります。import tsm config CLI コマンドを使用します。
- EMC Centera からリストアする前には、「データ・アーカイブ」パネルを使用して Guardium システムに PEA ファイルをアップロードする必要があります。
- 別の Guardium システムが暗号化したファイルをリストアまたはインポートする前に、そのシステムがファイルの暗号化に使用したシステム共有パスワードが、こちらのシステムでも使用可能であることを確認してください (使用可能でない場合、ファイルを暗号化解除できません)。システム構成の『システム共有パスワードについて』を参照してください。
- Guardium コレクターでリストアする前には、CLI コマンド stop inspection-core を実行して、inspection-core プロセスを停止してください。
  注: そのデータは、リストア処理の間はキャプチャーできません。

データのリストア手順は、以下のとおりです。

1. 「管理」 > 「データ管理」 > 「データのリストア」をクリックして、「データのリストア」を開きます。
2. 「開始」に日付を入力し、データを必要とする最も古い日付を指定します。
3. 「終了」に日付を入力し、データを必要とする最も新しい日付を指定します。
4. 「ホスト名」に、アーカイブが行われた Guardium システムの名前をオプションで入力します。
5. 「検索」をクリックします。
6. 「検索結果」パネルで、リストアする各アーカイブの「選択」チェック・ボックスにチェック・マークを付けます。
7. 「リストアしたデータを少なくとも次の期間パージしない」フィールドに、リストアしたデータをシステムに保持する日数を入力します。
8. 「復元」をクリックします。
9. 完了したら、「完了」をクリックします。

注: コレクターからアーカイブしたデータのリストアは、 同じコレクターへのリストア、アグリゲーターへのリストア、 または統合クラスターに属さない調査専用の別のコレクターへのリストアのみにしてください。コレクターが異常終了した場合は、 システム・バックアップを新規のクリーンなコレクターにリストアできます。

Guardium での Amazon S3 へのアーカイブおよびバックアップ

Guardium から、Amazon S3 へのデータのアーカイブとバックアップを行う場合に、この機能を使用します。

Amazon S3 (Amazon Simple Storage Service) は、いつでも、Web 上のどこからでも容量に関係なく、データを格納/取得できるシンプルな Web サービス・インターフェースを提供します。これによって、Amazon が Web サイトの稼働に使用しているものと同じ、拡張性と信頼性が高く、安全でありながら安価なインフラストラクチャーを、あらゆる開発者が利用することが可能になります。

前提条件

1. Amazon アカウント
2. S3 サービスの登録
3. Amazon S3 にアクセスするためには、Amazon S3 の認証情報が必要です。 必要な認証情報は次のとおりです。
   - Access Key ID (アクセス・キー ID): ユーザーをサービス要求の担当者として識別します。各要求にこの ID が含まれている必要があります。これは機密ではなく、暗号化する必要はありません (20 文字の英数字から成るシーケンス)。
   - Secret Access Key (シークレット・アクセス・キー): Secret Access Key は Access Key ID に関連付けられ、要求に含まれているデジタル署名を計算します。Secret Access Key は機密事項であり、ユーザーと AWS のみが保持する必要があります (40 文字から成るシーケンス)。このキーは、ファイルではなく、単なる長い文字列であり、この文字列を使用して、要求内に含まれている必要があるデジタル署名を計算します。

- 「データ・アーカイブ」は、システムによって所定の期間内にキャプチャーされたデータをバックアップします。
- 「結果アーカイブ」は、監査タスクの結果 (レポート、アセスメント・テスト、エンティティー監査証跡、プライバシー・セット、および分類プロセス) のほか、表示およびサインオフの証跡、ワークフロー・プロセスからの調整コメントをバックアップします。

Guardium データがアーカイブされると、日ごとに別のデータ・ファイルができます。

アーカイブ・データ・ファイルの名前は、次の形式になります。

```
<time>-<hostname.domain>-w<run_datestamp>-d<data_date>.dbdump.enc
```

Guardium のアーカイブ機能では、改ざんできない、署名付きの暗号化ファイルが作成されます。生成されたアーカイブ・ファイルの名前を変更することはできません。アーカイブ操作は、アーカイブ処理中に作成されるファイル名に依存します。

ハードウェア破損が生じた場合にサーバーを復元する目的で、必要なすべてのデータと構成値をバックアップして保管するために、システム・バックアップを使用します。

すべての構成情報とデータが 1 つの暗号化ファイルに書き込まれ、このシステムのバックアップ用に構成された転送方式を使用して、指定の宛先に送信されます。

バックアップ・システム・ファイルの形式は、次のとおりです。

```
<data_date>-<time>-<hostname.domain>-SQLGUARD_CONFIG-9.0.tgz
<data_date>-<time>-<hostname.domain>-SQLGUARD_DATA-9.0.tgz
```

Guardium の「統合/アーカイブ・ログ」レポートを使用して、操作が正常に完了したことを確認してください。「管理」 > 「レポート」 > 「データ管理」 > 「統合/アーカイブ・ログ」をクリックして、「統合/アーカイブ・ログ」を開きます。各アーカイブ操作には、複数のアクティビティーがリストされていなければなりません。 また、各アクティビティーの状況は「*成功*」でなければなりません。

Guardium カタログは、アーカイブ・データの宛先に関係なくすべてのアーカイブ・ファイルの送信場所を記録するため、以降のどの時点においても、最小限の労力でシステムでアーカイブ・ファイルを取得およびリストアすることができます。

システムごとに個別のカタログが保守され、システムがデータや結果をアーカイブするたびにカタログに新しいレコードが追加されます。

カタログ・エントリーは、以下のいずれかの方法により、アプライアンス間で転送することができます。

- 統合 - カタログ表は統合されます。つまり、アグリゲーターが、そのすべてのコレクターのカタログをマージしたものを保持することになります。

- カタログのエクスポート/インポート - これらの機能は、コレクター間でカタログ・エントリーを転送したり、将来のリストアのためにカタログをバックアップしたりするために使用できます。
- データのリストア - 各データ・リストア操作には、アーカイブした日のデータ (その日のカタログも含む) が含まれます。 したがって、データのリストア時には、カタログも更新されます。

カタログ・エントリーは、別のシステムからインポートされたときには、そのシステムによって暗号化されたファイルをポイントします。 そのようなファイルをリストアまたはインポートする前に、そのファイルを暗号化したシステムのシステム共有パスワードが、インポート側のシステムで使用できなければなりません。

Guardium CLI からの Amazon S3 の有効化

Amazon S3 のアーカイブ/バックアップ・オプションは、デフォルトでは Guardium GUI で有効になっていません。Guardium CLI から Amazon S3 を有効にするには、次の CLI コマンドを実行します。

```
store storage-system amazon_s3 archive on
store storage-system amazon_s3 backup on
```

Amazon S3 では、Guardium システムのクロック時刻が正確であること (15 分以内) が求められます。 そうでない場合、Amazon のエラーとなります。要求の時刻と現在の時刻の差が大きすぎると、要求は受け入れられません。

Guardium のシステム時刻が正確でない場合は、次の CLI コマンドを使用して正しい時刻を設定してください。

```
show system ntp server
store system ntp server (An example is ntp server: ntp.swg.usma.ibm.com)
store system ntp state on
```

ユーザー・インターフェース

バックアップを構成するには、「システム・バックアップ」を使用します。「管理」 > 「データ管理」 > 「システム・バックアップ」をクリックして、「システム・バックアップ」を開きます。

以下のユーザー入力が必要です。

- S3 Bucket Name (S3 バケット名)。Amazon S3 に保管されるすべてのオブジェクトは、バケット内に格納されます。バケットは、Amazon S3 に保管されるオブジェクトの名前空間をパーティション化します。1 つのバケット内では、保管するオブジェクトに任意の名前を使用できますが、バケット名は Amazon S3 内の全バケットの中で一意である必要があります。
- Access Key ID
- Secret Access Key

バケット名が存在しない場合は、作成されます。

Secret Access Key は、データベースに保存されるときに暗号化されます。

Amazon S3 にファイルがアップロードされたことの確認

1. AWS マネジメント・コンソールに、E メール・アドレスとパスワードを使用してログオンします。

http://aws.amazon.com/console/

1. 「S3」をクリックします。
2. Guardium UI で指定したバケットをクリックします。

Guardium アプライアンスからデータをパージする方法

Guardium アプライアンスでは、次の 2 つの領域が満杯になる可能性があり、それが原因で GUI が停止する場合があります。

- 内部データベース
- ファイル・システム自体 (通常 /var パーティション)

ユーザー CLI として、次の CLI コマンドを使用してデータベースが満杯かどうか確認します。

support show db-status free %

これで 10% 以下という結果が返ってきた場合、データベースは 90% 以上が埋まっています。

/var パーティション (ファイル・システム) が 90% 以上埋まっているかどうか確認するために、CLI から must gather コマンドを実行します。

support must_gather system_db_info

ファイル・サーバーで表示できる system_output.txt ファイル内の df -k 出力を、ファイル・サーバーを使用して確認できます。

must_gather/system_logs/system_output.txt

または、このファイルは、system.<datetime>.tgz ファイルをダウンロードした後にそこから取り出します。

system_output.txt ファイル内で、詳細を確認できます。

以下では、/var パーティションが 65% 埋まっています。

==========2016-11-30 08:36:09 ... Output of df command:==========

Filesystem 1024-blocks Used Available Capacity Mounted on

/dev/sda3 10154020 2272668 7357232 24% /

/dev/sda2 28571320 17384504 9712052 65% /var

/dev/sda1 505604 33476 446024 7% /boot

tmpfs 6169768 0 6169768 0% /dev/shm

より新しい Guardium バージョンには、安全なキャッチ/機能が備わっており、これにより、データベースまたはファイル・システムが特定のレベルに到達すると、メイン・プロセスはそれ以上のデータを収集するのを停止します。

デフォルトでは、データベース、ファイル・システム、またはこれらの両方が 90% 埋まったらプロセスは停止します (この例は v10.1 の資料に基づきます)。CLI を使用して、安全なキャッチの現行値を確認できます。

CLI> show auto_stop_services_when_full

注: auto_stop_services_when_full がオフの場合、アプライアンスでは、システムが 100% 満杯になる可能性があり、それによりシステムにまったくアクセスできなります。

以下の回答で説明されている特定の状況で一時的に使用される場合を除き、auto_stop_services_when_full をオフにする必要はありません。以下の特定の状況では、説明に従ってオフにし、スペースの問題を解決した後にオンに戻す必要があります。

注: auto_stop をオフに切り替える前に inspection-core を停止する必要があります。これにより、システムはそれ以上満杯にならなくなります。

この場合、ファイル・システムまたはデータベースが 90% 満杯のときには、システムにより inspection-core およびその他のプロセスが自動的に停止されます。これには、GUI インターフェースが含まれます。このため、その時点で GUI に接続できなくなります。

次のコマンドを使用して、停止したサービスを再始動しようとすると、システム (および GUI インターフェース) は 5 分後に同じ理由で再び停止する可能性があります。
restart stopped_services

注: このコマンドは、スペースがリカバリーされたことを確認した後でのみ使用する必要があります。

データベースまたはファイル・システムが満杯で「自動停止」レベルになる前に、システム・ログ (メッセージ・ファイル) で警告を受け取るはずです。

自動停止がトリガーされる前に、スペースの問題に関する E メールが送信されるようにアラートを設定できます。Guardium のデータベースが満杯になったときのアラートを参照してください。

must_gather コマンドを実行し、作成された圧縮ファイル内を調べて、最新のメッセージ・ファイルが含まれているか確認できます。

support must_gather system_db_info

>>>GUI がダウンした場合、内部データベースからデータをパージする

自動停止がトリガーされた場合、GUI などのサービスが停止します。これにより、「今すぐ 1 回実行」パージ・オプションを使用したデータの緊急パージも実行できなくなります。

この緊急パージを実行するには、以下を行います。

- アプライアンスにデータがこれ以上フラッディングしないようにするために、コレクターで inspection-core をオフにする必要があります。

stop inspection-core

- show processlist を除くデータベース・コマンドが実行されていないことを確認します (必要な場合は、次のステップの前に、実行中のコマンドを終了してかまいません)。

support show db-processlist running

『What can I do if I see my Guardium Appliance getting full?』で説明されているように、GUI にアクセスしてパージを実行するために、単純に restart gui を実行できるようにする必要があります。

GUI が 5 分ごとにダウンするという問題が発生した場合、GUI を再始動し、一部のデータをパージできるようにするために、auto_stop_services_when_full を「一時的」にオフに切り替えることを検討してください。この方法で GUI を再始動した場合、5 分間だけ GUI が実行されます。十分なデータがパージされる前、またはパージの継続を設定する前に、メインの Nanny プロセスによってサービスが再び停止される場合があります。

注: auto_stop_services_when_full がオフの場合、アプライアンスでは、システムが 100% 満杯になる可能性があり、それによりシステムにまったくアクセスできなります。

ここで説明されている特定の状況で一時的に使用される場合を除き、auto_stop_services_when_full をオフにする必要はありません。特定の状況では、説明に従ってオフにし、スペースの問題を解決した後にオンに戻す必要があります。

auto_stop をオフに切り替える前に inspection-core を停止する必要があります。これにより、システムはそれ以上満杯にならなくなります。

CLI> store auto_stop_services_when_full off

CLI> show auto_stop_services_when_full [off | restart | gui]

これで、GUI にアクセスし、その後「データ管理アーカイブ (Data Management Archive)」にアクセスし、一部のデータを消去するためにパージの実行を設定できます。

データベースが満杯かどうか引き続き確認します。パージ・プロセスが終了すると統合アーカイブ・ログが表示されます。

プロセスが終了し、システムにスペースができた後、auto_stop を再びオンに設定し、停止しているサービスを再始動する必要があります。

store auto_stop_services_when_full on

restart stopped services

必要に応じて、inspection-core を開始します。

これで、データの収集が再び開始されるはずです。

システムが満杯になったとき、多くの場合、非常に多くのアクティビティーが記録されています。

親トピック: Guardium システムの管理
関連情報:
高度な Guardium システム管理および構成 (ビデオ)
Guardium データベースが満杯になる問題の予防および対処 (ビデオ)

Guardium カタログ

Guardium システムからデータをアーカイブすると、Guardium カタログは、すべてのアーカイブ・ファイルの送信先を追跡して、そのファイルを取得およびリストアできるようにします。

このタスクについて

Guardium システムごとに個別のカタログが保守され、データまたは結果をアーカイブするたびにカタログに新しいレコードが追加されます。カタログ・エントリーは、以下のいずれかの方法により、アプライアンス間で転送することができます。

- 統合: カタログ表は統合されます。つまり、アグリゲーターが、そのすべてのコレクターのカタログをマージしています。
- カタログのエクスポート/インポート: これらの機能は、コレクター間でカタログ・エントリーを転送したり、将来のリストアのためにカタログをバックアップしたりするために使用できます。
- データのリストア - 各データ・リストア操作には、アーカイブした日のデータ (その日のカタログも含む) が含まれます。データのリストア時には、カタログも更新されます。

カタログをアーカイブしたり、カタログを外部ストレージにエクスポートしたり、保管されているカタログをインポートしたりすることができます。

カタログ・エントリーは、別のシステムからインポートされると、そのシステムによって暗号化されたファイルをポイントします。そのようなファイルをリストアまたはインポートする前に、そのファイルを暗号化したシステムのシステム共有パスワードが、インポート側のシステムで使用できなければなりません。aggregator backup keys file CLI コマンドおよび aggregator restore keys file CLI コマンドを使用して、ある Guardium システムから別のシステムに共有パスワードをコピーすることができます。

親トピック: Guardium システムの管理

カタログのアーカイブ

手順

1. 「管理」 > 「データ管理」 > 「カタログ・アーカイブ」をクリックします。
2. 日付範囲を指定して選択可能なカタログ・エントリーを表示するか、カタログ・エントリーを追加することができます。 カタログ・エントリーを表示する場合:
   a. 「開始」に日付を入力し、データを必要とする最も古い日付を指定します。
   b. 「終了」に日付を入力し、データを必要とする最も新しい日付を指定します。
   c. オプション: 「ホスト名」に、アーカイブが行われた Guardium® システムの名前を入力します。
   d. 「検索」をクリックします。
   カタログ・エントリーを追加する場合:
   a. 「追加」をクリックします。
   b. 「ファイル名」を入力します。
   c. 「ホスト名」を入力します。
   d. ファイルの「パス」を入力します。
   注:

   FTP の場合、FTP アカウントのホーム・ディレクトリーを基準にした相対パスでディレクトリーを指定します。

   SCP の場合: 絶対パスとしてディレクトリーを指定します。

   TSM の場合: 元のロケーションの絶対パスとしてディレクトリーを指定します。

   e. このロケーションにアクセスするための「ユーザー名」および「パスワード」を入力します。
   f. 「保持」フィールドに、この項目をカタログ内に保持する日数 (デフォルト値は 365) を入力します。
   g. 「ストレージ・システム」メニューから、ファイルを格納するためのオプションを選択します。
   h. 「保存」をクリックします。
3. カタログ・エントリーを削除するには、カタログを開き、エントリーを選択して、「選択したものを削除 (Remove Selected)」をクリックします。
4. 完了したら、「完了」をクリックします。

カタログのエクスポート

手順

1. 「管理」 > 「データ管理」 > 「カタログ・エクスポート」をクリックします。
2. 「タイプ」ドロップダウン・リストから定義タイプを選択します。 「エクスポートする定義」リストに、選択したタイプの定義が設定されます。
3. このタイプの定義でエクスポートするものをすべて選択して、「エクスポート」をクリックします。 ご使用のブラウザーのセキュリティー設定に応じて、ファイルを保存するか開くかを訪ねるメッセージが表示される場合があります。
4. エクスポート・ファイルを保存するためのロケーションを選択します。

カタログのインポート

手順

1. 「管理」 > 「データ管理」 > 「カタログ・インポート」をクリックします。
2. 「参照」をクリックしてファイルを見つけ、選択します。

3. 「アップロード」をクリックします。操作完了時には通知が出され、ファイルに含まれる定義が表示されます。さらにファイルをアップロードするには、手順を繰り返します。
4. アップロード・ファイルをインポートする場合は「インポート」をクリックします。または、内容をインポートせずにアップロード・ファイルを削除する場合は「インポートなしで削除 (Remove without Importing)」をクリックします。

バックアップとアーカイブの管理方法

データを保持する方法を確立し、アクティビティー・ボリュームの制御を行い、データのアーカイブとパージのスケジューリングと、毎月のバックアップのスケジューリングを管理します。

付加価値: ベスト・プラクティス。データが損失しないように、データを保護してください。監査のために、いつでもデータにアクセスできるようにしてください。

システム・バックアップ機能を使用すると、オンデマンドまたはスケジュールに基づいて実行可能なバックアップ操作を定義できます。

ハードウェア破損が生じた場合にサーバーを復元する目的で、必要なすべてのデータと構成値をバックアップして保管するために、システム・バックアップを使用します。

使用可能なアーカイブ操作は 2 つあります。「管理」 > 「データ管理」に移動して、「データ・アーカイブ」機能または「結果アーカイブ」機能を選択します。

- 「*データ・アーカイブ*」は、Guardium システムによって所定の期間内にキャプチャーされたデータをバックアップします。データ・アーカイブを構成する際には、パージ操作も構成できます。 通常、データは、キャプチャーされた日の終わりにアーカイブされます。 こうすることで、災害発生時に失われるのはその当日のデータだけになります。 データのパージは、アプリケーションごとに異なり、ビジネス要件や監査要件によってかなり大きな違いがあります。 ほとんどの場合、データはマシン上に 6 カ月以上保持することができます。
- 「*結果アーカイブ*」は、監査タスクの結果 (レポート、アセスメント・テスト、エンティティー監査証跡、プライバシー・セット、および分類プロセス) のほか、表示およびサインオフの証跡、ワークフロー・プロセスからの調整コメントをバックアップします。結果セットは、ワークフロー・プロセスの定義に従ってシステムからパージされます。

統合環境では、データはコレクター、アグリゲーター、またはその両方のロケーションからアーカイブできます。 データは、一度だけアーカイブするのが最も一般的です。 アーカイブ元となるロケーションは、ユーザーの要件に応じて異なります。

データをアーカイブする際には、必ず操作が正常に完了したことを確認してください。 これを確認するには、管理者ユーザーとしてログインし、「管理」 > 「レポート」 > 「データ管理」 > 「統合/アーカイブ・ログ」をクリックして、「統合/アーカイブ・ログ」を開きます。各アーカイブ操作には、複数のアクティビティーがリストされていなければなりません。また、各アクティビティーの状況は「*成功*」でなければなりません。

データ・バックアップ

推奨されるデータ・バックアップのタイプは、以下の 3 つです。

1. フルバックアップ/システム・バックアップ
    a. 中央マネージャー・ユニットの週次または日次のフルバックアップ (スタンドアロンの中央マネージャーの場合)。
    b. アグリゲーターまたはコレクターについてオフピーク時に実行する月次のバックアップ。
2. アグリゲーターまたはコレクター用の日次アーカイブ。これらのアーカイブは、増分バックアップと考えることができます。アグリゲーターのアーカイブ・ファイルは、コレクターのアーカイブ・ファイルよりも大幅に大きくなります。例えば、1 つのアグリゲーターに対して 10 個のコレクターがデータを送信する場合、アーカイブ・ファイルの最初のサイズは、これら 10 個のコレクターのアーカイブ・ファイル全体と同じサイズになります。ただし、アグリゲーターのアーカイブ・ファイルには、毎日コレクターから送信されるのとは別のデータが含まれるため、このファイルのサイズは、コレクターのアーカイブ・ファイルをすべて組み合わせたサイズよりも大幅に大きくなります。
3. アグリゲーター用の結果アーカイブ (これは、日次バックアップとフルバックアップのデータの特殊なサブセットです)。結果アーカイブを使用する代わりに、すべてのユーザーがレビュー・プロセスを完了した後で、「監査プロセス」から PDF ファイルの保存操作を実行することもできます。

データの保持

データのバックアップとアーカイブ・ファイルは、災害からの復旧と、履歴調査または監査の目的で役立ちます。

以下の推奨事項は、社内のデータ保持ポリシーに基づいて変更することができます。例えば、組織によっては、すべてのバックアップを 18 カ月間保持しなければならない場合があります。

災害からの復旧目的でのデータ保持

- 各ユニットのフルバックアップを 3 カ月分ごとに循環して保持します。
- 管理対象コレクターの日次アーカイブを 2 週間ごとに循環して保持します。

注: スタンドアロンのコレクターを使用している場合、日次アーカイブは、データ保持ポリシーに従って保存してください。

履歴調査または監査目的でのデータ保持

- アグリゲーターのすべての日次アーカイブを、監査ポリシーまたは会社のデータ保持ポリシーで定義されている期間だけ保持します。

記憶容量

以下に示すサイズは、補助ストレージ容量を計画するためのバックアップとアーカイブ・ファイルのサイズの見積もりまたは範囲にすぎません。

実際のサイズは、(1) Guardium コレクターに記録されるデータベース・アクティビティーのボリュームおよび細分度と、(2) バックアップ・ファイルの保存期間に応じて異なります。

日次アーカイブ

コレクター: 約 40 MB (特権ユーザー・モニターの場合) から 1 GB (すべてのトラフィックについてすべての詳細を記録する広範なモニターの場合) まで。

アグリゲーター: コレクターの数のおおよその倍数。例えば、コレクターの数に 40 MB を掛けた値。

月次システム・バックアップ: Dell R610 または IBM xSeries 3550 M4 (600 GB ディスク) で、データベースが 50% 使用されていることを想定

注: このバックアップでは、バックアップ・ファイルの圧縮比率は約 1:8 になります。

コレクター: 7 GB から 10 GB まで

アグリゲーター: 16 GB から 20 GB まで

中央マネージャー (集約なし): << 1 GB

結果アーカイブ

実装された監査プロセスの数と頻度によって異なります。

アクティビティー・ボリュームの制御

データベース・サーバーでモニターされるアクティビティーと、コレクターに記録されるアクティビティーのボリュームを制御すると、ネットワーク使用率の削減、Guardium システムのデータベース・ディスク使用量の削減、IBM Security Guardium インフラストラクチャーの全体的な性能とパフォーマンスの改善に役立ちます。

この制御は、主に、検査エンジン構成を介してポリシー・ルールで行われます。

一般的なガイドラインを以下に示します。

- 検査エンジンで、ポート範囲は使用しないでください。
- 信頼できるアプリケーションとバッチ・プログラムをすべて特定してください。通常、これらのプログラムによって大量のデータベース・アクティビティーが生成されるため、可能であれば、「S-TAP セッションを無視」アクションまたは「ロギングをスキップ」アクションを使用して、これらのアクティビティーを無視またはスキップしてください。
- 必要である場合を除き、「全詳細をロギング」アクションは使用しないでください。
- 可能であれば、選択的な監査ポリシーを「S-TAP セッションを無視」ルールとともに使用して、ネットワーク・トラフィックを最小化してください。
- 例えば、抽出ルールを使用しない場合、結果セットは検査されません。結果セットが Guardium システムに送信されないように、「セッションごとに応答を無視」アクションを使用することを検討してください。
- 新しいデータベースとアプリケーションに対応するために、ポリシー・ルール (グループを含む) のレビューと更新を定期的に実行するプロセスを確立してください。
- SQL エラーを定期的にモニターし、DBA とアプリケーションの開発チームが修復を行うためのプロセスを確立してください。

スケジューリング

以下の表に、Guardium システムで構成する必要がある主なスケジュールの要約を示します。この表の下に、各プロセスの簡単な説明があります。

「統合/アーカイブ・ログ」を使用すると、各プロセスの時刻と状況が記録されるため、スケジューリング時刻の調整に役立ちます。

以下の表に、コレクターとしてデプロイされる、Guardium システムのタスクのスケジュールをリストします。

機能	スケジュール
データ・エクスポート (アグリゲーターへのエクスポート)	日次*: 12:30 AM
データのアーカイブとパージ	日次: 01:30 AM、15 日分をパージ
監査/ワークフローのジョブ	日次: 03:00 AM (スタンドアロンの場合)
SCP/FTP サーバーへの CSV/CEF のエクスポート	日次: 05:00 AM、監査ジョブ内で構成されていて、かつその監査ジョブが完了している場合。
ホスト名の別名割り当て	日次: 10:00 PM
ポリシーの再インストール	日次: 11:00 PM
システム・バックアップ	月次: 毎月第 1 日曜日の 6:00 AM

以下の表に、アグリゲーターとしてデプロイされる、Guardium システムのタスクのスケジュールをリストします。

機能	スケジュール
データのアーカイブとパージ	日次: 4:00 AM、30 日分をパージ
データ・インポート (コレクターからのインポート)	日次 1:15 AM
監査/ワークフローのジョブ	日次: 03:30 AM
SCP/FTP サーバーへの CSV/CEF のエクスポート	日次: 05:15 AM、監査ジョブ内で構成されていて、かつその監査ジョブが完了している場合。
ホスト名の別名割り当て	日次: 10:00 PM
システム・バックアップ	月次: 毎月第 1 日曜日の 7:00 AM

注: 各 Guardium システムにおける内部の始業時処理と競合することを避けるため、12:15 AM よりも前の時刻にはスケジューリングしないでください。

日次データ・アーカイブを設定する場合は、1 日以上経過したデータをアーカイブし、2 日以上経過したデータについては無視するように設定してください。 初回の実行では、すべてのデータがデータベースにアーカイブされ、それ以降の処理では、前日のデータだけがアーカイブされます。

オンラインで保持されるデータの量は各 Guardium システムのデータベースのサイズによって制限されるため、パージ処理は、オンラインで保持されるデータ量の管理に役立ちます。パージ処理は、日次アーカイブと連携して機能します。データベースがいっぱいになるのを避け、データベースのパフォーマンスを改善するために、必要最小限の量のデータだけを保持することをお勧めします。

コレクターの場合、コレクター用に 15 日分、アグリゲーター用に 30 日分のデータを保持することをお勧めします。ただし、実際の期間は、記録されるデータの量 (S-TAP の数、ポリシー・ルールの数、コレクターの数など) によって異なります。

データのエクスポートとインポート

前日に記録されたアクティビティーは、コレクターから、そのコレクターに割り当てられたアグリゲーターへ、統合レポート用に毎日エクスポートされます (プッシュ処理)。 このアクティビティーは、アグリゲーターでの「データ・インポート」に対応しています。

注: 利便性を考慮して、「アーカイブ」設定画面でも「エクスポート」設定画面でもパージを構成できるようになっています。

データ・インポート処理は、アグリゲーター上でのみスケジュールされます。この処理では、コレクターからエクスポートされた前日のデータがインポートされて処理されます。

月次バックアップ

既に説明したように、システム・バックアップはフルバックアップであり、災害からの復旧目的で使用されます。以下に、毎月第 1 日曜日の 6:00 AM に開始される月次スケジュールの例を示します。

親トピック: Guardium システムの管理

結果のエクスポート (CSV、CEF、PDF)

CSV、CEF、および PDF ファイルをワークフロー・プロセスで作成できます。この機能で、Guardium システム上に存在するこれらのファイルをすべてエクスポートします。

ワークフロー・プロセスによって作成される CEF/CSV ファイルを syslog に書き込むこともできます。この処理が行われた場合、それらのファイルはここで説明する方法でエクスポートすることはできません。 それらのファイルは、他の方法で syslog からアクセスする必要があります。

CSV、CEF、および PDF ファイルをエクスポートするには、以下のようにします。

1. 「管理」 > 「データ管理」 > 「結果エクスポート (ファイル)」をクリックして、「結果エクスポート (ファイル)」を開きます。
2. 「プロトコル」ラジオ・ボタンから、オプション (SCP、FTP、Amazon S3、または Softlayer) を選択します。
3. 「ホスト」に、ファイルを受信するホストの IP アドレスまたは DNS ホスト名を入力します。
4. 「ディレクトリー」に、データの格納先ディレクトリーを指定します。このディレクトリーの指定方法は、選択したプロトコルによって異なります。
   - FTP の場合: FTP アカウントのホーム・ディレクトリーに対する相対パスでディレクトリーを指定します。
   - SCP の場合: 絶対パスとしてディレクトリーを指定します。
5. SCP および FTP を介するファイル送信に使用できるポートに変更します。SSH、FTP、および SFTP のデフォルトのポートは 22 です。FTP のデフォルトのポートは 21 です。
6. 「ユーザー名」および「パスワード」に、ホスト・マシンにログインするユーザーの資格情報を入力します。 このユーザーは、「ディレクトリー」フィールドで指定したディレクトリーに対する書き込み権限/実行権限を保持していなければなりません。
7. 「スケジューリング」セクションを使用して、この操作を定期的に実行するためのスケジュールを定義できます。
8. 「保存」をクリックして構成を保存します。システムはそのロケーションにテスト・データ・ファイルを送信することにより、構成の検証を試みます。この操作が失敗した場合は、エラー・メッセージが表示されます。
9. 「今すぐ 1 回実行」をクリックして、操作を 1 回実行します。
10. ファイルがエクスポートされたことを検証するには、「統合/アーカイブ・ログ」にチェック・マークを付けます。エクスポートした CSV または CEF ファイルのそれぞれに、「送信」アクティビティーがなければなりません。

デフォルトの区切り文字を定義するには、「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」をクリックして、「グローバル・プロファイル」を開きます。

すべてのファイル名に含めるラベルを入力するには、「ツール」 > 「監査プロセス・ビルダー」に移動します。
注:

Syslog メッセージの最大サイズは 4000 です。CSV 結果は、この制限を超えると切り捨てられます。

.CSV ファイルを読み取るために使用するアプリケーションが何であっても、エンコードは UTF-8 に設定します。Excel では、異なる文字セットがデフォルトではされるため、.CSV ファイルが破損する可能性があります。Excel を使用する場合、単にファイルを開くのではなく、またはファイルの関連付けに基づいて Excel を起動するのではなく、.CSV ファイルをインポートし、UTF-8 エンコードを選択します。

親トピック: Guardium システムの管理

定義のエクスポート/インポート

要件が同じまたは同じようなシステムが複数あり、一元管理を使用していない場合は、これらのシステムのソフトウェア・リリース・レベルが同じであれば、必要なコンポーネントを 1 つのシステムで定義し、その定義を他のシステムにエクスポートできます。

一度にエクスポートできる定義のタイプ (例えばレポート) は 1 つです。 要素のエクスポートごとに、参照される他の定義もエクスポートされます。 例えば、レポートは常に照会に基づいており、IP アドレス・グループや期間などの他の項目も参照することがあります。 参照される定義はすべて (セキュリティー・ロールを除く)、レポート定義と共にエクスポートされます。 ただし、その定義が複数のエクスポート項目で参照される場合は、1 つの定義のコピーのみがエクスポートされます。 ポリシーまたは照会のエクスポートでは、エクスポートされるポリシーまたは照会が参照するグループのみがエクスポートされます。 これまでは、ポリシーまたは照会をエクスポートすると、すべてのグループがエクスポートされていました。

定義のエクスポート/インポート
: 定義のエクスポートおよびインポートを使用して、特定の Guardium システムから機能データを保存した後、復元します。 例えば、この機能により、1 つの Guardium システム上でレポートを作成し、次に同じ Guardium のバージョンがインストールされている別のサーバー上にその同じレポートをインポートすることができます。

注: この機能は、サーバーのフルバックアップと同じではありません。この機能を使用していても、定期的に、または手動により、バックアップを定義して実行する必要があります。
定義のエクスポート - これは、レポート/照会、CAS データ、分類データなど、定義されている機能値を保存して共有するために使用されます。エクスポート・タイプは PC 上に .sql ファイル・タイプとして保存されます。
定義のインポート - この機能は、同じ Guardium ソフトウェア・バージョンを使用しているサーバー上に、エクスポート済み定義をインポートするために使用されます。例えば、Guardium V10 システムから定義をエクスポートする場合は、別の V10 システムにのみ、それらの定義をインポートできます。

注:

- グラフィカル・レポートをエクスポートする場合、表示パラメーター設定 (色、フォント、タイトルなど) はエクスポートされません。これらのレポートをインポートすると、インポート側システムのデフォルトの表示パラメーター設定が使用されます。
- サブスクライブしたグループはエクスポートされません。 サブスクライブしたグループを参照する定義をエクスポートする場合は、参照されるサブスクライブしたグループをすべて、インポート側アプライアンス (フェデレーテッド環境では中央マネージャー) にインストールしておく必要があります。
- エクスポート/インポート定義のログの保存期間は、モニターされるデータベースのアクティビティー・ログと同じです。
- コメントはエクスポートに含まれません。
- 実行がスケジュール設定 (スケジュール時刻を含む) された監査プロセス定義を別のシステムにエクスポートした場合、「監査プロセス・ビルダー」の「アクティブ」チェック・ボックスにチェック・マークは付けられません (非アクティブ)。
- 1 つのアプライアンスで定義されて別の (無関係の) アプライアンスにエクスポートされた監査プロセスの「開始時刻のスケジュール設定」- 元の「開始時刻のスケジュール設定」が定義されている場合は、保持されます。 元の「開始時刻のスケジュール設定」が定義されていない場合 (空の場合)、インポートされた「開始時刻のスケジュール設定」は、それがインポートされた時刻に設定されます。
- オープン・ソース・ドライバーを使用するデータ・ソースをエクスポートする場合、オープン・ソース・ドライバーはエクスポートに含まれません。オープン・ソース・ドライバーを使用して作成されたデータ・ソース定義をインポートする前に、まずそのオープン・ソース・ドライバーを新しいシステムにアップロードしておく必要があります。そうしないと、データ・ダイレクト・ドライバーがインポートされると、それがオープン・ソース・ドライバーの代わりに使用されるようになります。
- 大規模で複雑なインポートでは、かなり時間がかかることがあり、ユーザーのセッションの長さを超える場合があります。 この状況になってセッションがタイムアウトになっても、インポートは完了するまでバックグラウンドで実行され続けます。
- 分類ポリシーの定義をエクスポートする場合、ポリシーに関連付けられているカスタム評価クラスは定義と共にエクスポートされません。インポート済みポリシーを動作させるには、カスタム評価クラスを個別にアップロードする必要があります。
- 異なる言語間での定義のエクスポート/インポートは機能しません。例えば、中国語 (簡体字) の Guardium® システムからファイルをエクスポートし、英語の Guardium システムにそのファイルをインポートしようとしても成功しません。

XACML プロトコルへのエクスポート

Guardium は、XACML ファイルへのポリシー・ルールのエクスポート、および別の Guardium システムへの XACML ファイルのインポートをサポートしています。

XACML (eXtensible Access Control Markup Language) は、XML 内に実装される宣言アクセス制御ポリシー言語で、ポリシーの解釈方法を記述する処理モデルでもあります。

標準的な XACML のエクスポート/インポートが双方向のインターフェースとして使用され、Optim Designer と Guardium の間でポリシー・ルールが転送されます。

Optim Designer は、さまざまな目的で、さまざまな手段によってデータ値を変換できます。コア Optim ランタイム (z/OS および Distributed) において、列マップ内で宣言されたデータ・プライバシー関数を呼び出すことにより、この操作が行われます。 Optim Privacy では、これは属性に対するデータ・プライバシー・ポリシーの適用としてユーザーにより指定され、データ・アクセス・プラン内のエンティティーによって参照されます。

Optim Privacy と Guardium の両方を購入したお客様は、一方の製品からポリシーとプライバシーの情報を XACML にエクスポートし、他方の製品にインポートすることができます。

注: 以前のバージョンの Guardium からの XACML インポートはサポートされていません。
Guardium のポリシーを XACML にエクスポートするには、以下の手順を実行します。

1. 「管理」 > 「データ管理」 > 「エクスポート」をクリックします。
2. 「タイプ」メニューから「ポリシー」を選択します。
3. 「XACML ファイルへのエクスポート」チェック・ボックスにチェック・マークを付けます。
4. 「エクスポートする定義」メニューから定義を選択します。
5. 「エクスポート」をクリックします。

別の Guardium システムまたは Optim Privacy から XACML ファイルをインポートするには、「管理」 > 「データ管理」 > 「インポート」をクリックして、「定義のインポート」を開きます。

グループのインポート

既に存在するグループをインポートする場合、メンバーが追加される場合がありますが、メンバーが削除されることはありません。

別名のインポート

別名をインポートする場合、新しい別名が追加される場合がありますが、別名が削除されることはありません。

インポートされた定義の所有権

定義が作成されると、それを作成したユーザーが、その定義の所有者として保存されます。 この意味は、その定義にセキュリティー・ロールが割り当てられていなければ、それへのアクセス権限があるのは、所有者と admin ユーザーのみであるということです。

定義がインポートされると、所有者は常に admin ユーザーに変更されます。

インポートされた定義のロール

セキュリティー・ロールの参照は、エクスポートされた定義から削除されます。 したがって、インポートされた定義には、ロールが割り当てられていません。

インポートされた定義のユーザー

エクスポートされる定義内にユーザーの参照があると、そのユーザー定義がエクスポートされます。 定義がインポートされるとき、参照されるユーザー定義がインポートされるのは、インポート側システムにそのユーザー定義が存在しない場合のみです。つまり、既存のユーザー定義に上書きされることはありません。『ロールおよびユーザーの重複に関する考慮点』に記載されているように、これにはいくつかの考慮点があります。

なお、インポートされたユーザー定義は使用不可に設定されます。つまり、インポートされたユーザーは、インポート側システムから送信された E メール通知を受信できますが、管理者がそのアカウントを使用可能に設定するまで、そのシステムにログインできないことになります。

グループおよびユーザーの重複に関する考慮点

エクスポートされた定義によって参照されるグループがインポート側システムに存在する場合、エクスポート側システムからのそのグループの定義はインポートされません。そのために、両方のシステムでグループが同じ目的に使用されていない場合は、何らかの混乱が生じる可能性があります。

インポート側システムにユーザー定義が存在する場合、エクスポート側システムで定義されたその同じ個人のユーザー定義ではない可能性があります。例えば、エクスポート側システムでは、E メール・アドレスが john_doe@aaa.com のユーザー jdoe は、エクスポートされたアラートからの出力の受信者であるとします。 一方、インポート側システムでは、E メール・アドレスが jane_doe@zzz.com のユーザー jdoe が既に存在するとします。 エクスポートされたユーザー定義はインポートされません。インポートされたアラートが起動すると、E メールは jane_doe@zzz.com アドレスに送信されます。いずれにしても、セキュリティー・ロールまたはユーザー定義がインポートされないときは、両方のシステムでの定義を確認して、違いがないか調べてください。 違いがある場合は、それらの定義に対して適切な調整を行ってください。

エクスポートの定義タイプ

表 1. エクスポートの定義タイプ

エクスポート可能	エクスポート不可
アラート	カスタム・アラート・クラス 「定義のエクスポート」画面には、グループ・メンバーの除外を行うチェック・ボックスがあります。 グループ明細項目の説明を参照してください。
別名	カスタム評価テスト
監査プロセス	カスタム識別プロシージャー
オートディスカバリー・プロセス	
CAS ホスト	
CAS テンプレート・セット	
分類プロセス	アクセス・ルール
分類ポリシー	
カスタム・クラス接続の許可	
カスタム・ドメイン	
カスタム表	
データ・ソース	
イベント・タイプ	
グループ	「定義のエクスポート」画面には、グループ・メンバーの除外を行うチェック・ボックスがあります。 このチェック・ボックスは、エクスポート階層のいずれかの場所にグループが存在するデータ・セットの場合にのみ表示されます (例えば、アラートのエクスポートにはアラートの照会も含まれ、照会に照会条件内のグループが含まれることがあります)。データ・ソースのエクスポートにグループが含まれていない場合は、このチェック・ボックスは表示されません。 このチェック・ボックスが設定されている場合、エクスポート・ファイルにはグループが含まれています (グループがエクスポートされた定義にリンクされている場合) が、グループのメンバーはエクスポートされません。 このチェック・ボックスはデフォルトでは設定されず、その状態は永続的ではありません。また、現在のエクスポートにのみ適用されます。
名前付きテンプレート	
期間	
ポリシー (組み込まれたベースラインではない)	
プライバシー・セット	
照会	
Replay	
レポート	「定義のエクスポート」画面には、グループ・メンバーの除外を行うチェック・ボックスがあります。 グループ明細項目の説明を参照してください。
ロール	
セキュリティー・アセスメント	
ユーザー	
ユーザー・データベース・マッピング	
ユーザー・データベース・アクセス権	

エクスポート可能	エクスポート不可
ユーザー階層	

エクスポート定義

1. 「管理」 > 「データ管理」 > 「エクスポート」をクリックして、「定義のエクスポート」ペインを開きます。
2. 「タイプ」メニューからオプションを選択します。「エクスポートする定義」ボックスに、選択したタイプの定義が設定されます。
3. このタイプのすべての定義をエクスポートすることを選択します。
   注: 名前に引用文字が 1 つ以上含まれるポリシー定義はエクスポートしないでください。こうした定義はエクスポートできますが、インポートできません。 このような定義をエクスポートするには、そのコピーを作成し、引用文字を使用しない名前を付けた後にエクスポートします。
4. 「エクスポート」をクリックします。ご使用のブラウザーのセキュリティー設定に応じて、ファイルを保存するか、それともエディターを使用して開くかを尋ねる警告メッセージが表示される場合があります。
5. エクスポート・ファイルを適切な場所に保存します。

インポート定義

1. 「管理」 > 「データ管理」 > 「インポート」をクリックして、「定義のインポート」ペインを開きます。
2. 「参照」をクリックしてファイルを見つけ、選択します。
3. 「アップロード」をクリックします。操作完了時には通知が出され、ファイルに含まれる定義が表示されます。追加のファイルをアップロードするには、手順を繰り返します。
4. 直接インポートされた、または照会やポリシーなどの他のデータ・セットを通じてインポートされた新規グループ・メンバーの追加方法の動作を設定するには、「グループ・メンバーを完全に同期」チェック・ボックスを使用します。 このチェック・ボックスをオフにすると、インポートされた新規メンバーは追加され、一方でインポートされなかったメンバーは削除されません。このチェック・ボックスをオンにすると、インポートされなかったグループ・メンバーは削除されます。チェック・ボックスの設定を保存するには、チェック・ボックスの横にある「デフォルトとして設定」ボタンを使用します。
5. 「この定義セットをインポート」をクリックすると、定義セットがインポートされます。または「この定義セットをインポートなしで削除」をクリックすると、定義をインポートせずに、アップロード・ファイルが削除されます。
6. いずれのアクションの場合も、確認を求めるプロンプトが出されます。
   注: インポート操作では、既存の定義に上書きされません。既存の定義と同じ名前の定義をインポートしようとすると、その項目は置き換えられなかったことが通知されます。 インポートされた定義で既存の定義を上書きする場合は、インポート操作を実行する前に、既存の定義を削除する必要があります。

親トピック: Guardium システムの管理

分散インターフェース

この構成画面は、分散インターフェースを定義し、プロトコル・バッファー (.proto) ファイルを DIST_INT データベースにアップロードするために使用します。

このデータベースから、照会ドメイン・メタデータが自動的に作成されます。 メタデータの作成後、ユーザーは「カスタム・ドメイン・ビルダー」に移動して、データを変更したり、コピー作成したりして、カスタム・レポートを作成することができます。分散インターフェース・データは、プロトコル・バッファーを使用します。 プロトコル・バッファーは、構造化データをシリアライズするための、柔軟、効率的、かつ自動化されたメカニズムです。

Universal Feed タイプ 3 の場合、「管理」 > 「データ管理」 > 「分散インターフェース」をクリックして、DIST_INT データベースの構成用のプロトコル定義ファイルをアップロードします。

注: 表エンジン・タイプと表索引を管理する場合は、「メンテナンス」をクリックします。Guardium 内部データベースに保管されるデータが MYSQL ベースのため、Universal Feed 表の表エンジン・タイプ (InnoDB および MyISAM) が、すべての Universal Feed 表に対して表示されます。 InnoDB および MyISAM の保守について詳しくは、外部データ相関を参照してください。

分散インターフェースの構成

1. 「管理」 > 「データ管理」 > 「分散インターフェース」をクリックして、「分散インターフェース・ファインダー」を開きます。
2. 「新規」をクリックして、新しい分散インターフェースを作成するか、「分散インターフェース・ファインダー」から既存の分散インターフェースを選択して、「変更」または「削除」をクリックします。
3. 「ベンダー ID」に、ベンダーの ID (例えば、20000) を入力します。
4. 「ドメイン・ネーム」に、カスタム・ドメイン・ビルダーから選択可能になるドメインの名前を入力します。
5. 「統合に包含」にチェック・マークを付けます。
6. 「ファイル名」で、「参照」をクリックしてファイルを選択します。
7. 「適用」をクリックして、この構成を保存します。
8. 「カスタム・ドメイン・ビルダー」でカスタム・レポートを作成します。 「設定」 > 「ツールとビュー」 > 「カスタム・ドメイン・ビルダー」をクリックして、「カスタム・ドメイン・ビルダー」を開きます。

.proto ファイルの例

```
package bim;
option java_package = "com.ibm.infosphere.bim.proto";
option java_outer_classname = "BimEvent";
// NOTE: AssetID and Property_type (== Property name!) are strings.
// For AssetID , it is safest to use a UUID since it provides world-wide unique ID.
// This will be the key to the table of current metrics and property values.
// per each asset, per each property , there will be one value (recent, or min, or max,etc)
message EventTypeID {
    required string eventType                   = 1; //e.g. Schema change
}
message AssetID {
    required string assetId                     = 1;
}
message InfoPropertyID {
    required string assetId                     = 1;
```

```
    required string propertyName                = 2;
}
message MetricPropertyID {
    required string assetId                     = 1;
    required string propertyName                = 2;
}
message AssetRelationID {
    // These are asset "native" ids
    required string sourceAssetId               = 1;
    required string targetAssetId               = 2;
}
message RelationPropertyID {
    required string assetRelationId             = 1;
    required string propertyName                = 2;
}
message Event {
    optional InnerEvent innerEvent              = 1;
}
message InnerEvent {
    // Common for all events
    optional EventTypeID eventTypeId                = 1;
    optional string description                     = 2;
    optional string time                            = 3;
    optional string agentId                         = 4;
    // Event can be for asset info, or metric property
    optional AssetInfoEvent assetInfoEvent          = 5;
    optional MetricPropertyEvent metricPropertyEvent  = 6;
    optional AssetRelationEvent  relationEvent      = 7;
    optional RuleEvent ruleEvent                    = 8;
}
message AssetInfoEvent {
    optional AssetID unique_key__               = 1;
    optional string assetType                   = 2;
    optional string assetName                   = 3;
    optional string gdm_server_ip               = 4;
    optional string gdm_service_name            = 5;
    repeated InfoProperty property              = 6;
}
message InfoProperty {
    optional InfoPropertyID unique_key__        = 1;
    optional string value                       = 2;
}
message MetricPropertyEvent {
    optional AssetID assetId                    = 1;
    repeated MetricProperty property            = 2;
}
message MetricProperty {
    optional MetricPropertyID unique_key__      = 1;
    optional AssetID assetId                    = 2;
    optional string stringValue                 = 3;
    optional double doubleValue                 = 4;

    enum Data_type {
        DOUBLE                                      = 1;
        LONG                                        = 2;
        INT                                         = 3;
        FLOAT                                       = 4;
        DATE                                        = 5;
        BOOLEAN                                     = 6; // convention is to store it
as 0 and 1 in the double_value
        STRING                                      = 7; // stored in string_value
    }
        optional Data_type dataType                 = 5;
    optional string unit                        = 6; // unit for the value
}
message AssetRelationEvent {
    optional AssetRelationID unique_key__       = 1;
    required string relationshipType            = 2;
    repeated RelationshipProperty property      = 3;
    optional bool deleted                       = 4;
}
message RelationshipProperty {
    optional RelationPropertyID unique_key__    = 1;
    optional string value                       = 2;
}
message RuleEvent {
    optional string ruleName                    = 1;
    optional bool enabled                       = 2;
}
// --- Metadata --- All unique identifier must be defined here
message Identifier {
    optional InfoPropertyID infoPropertyId              = 1;
    optional MetricPropertyID metricPropertyId          = 2;
    optional AssetID assetId                            = 3;
    optional AssetRelationID assetRelationId            = 4;
    optional RelationPropertyID relationshipPropertyId  = 5;
}
```

親トピック: Guardium システムの管理

カスタム・クラスの管理

アラートまたは評価で使用されるカスタム・クラスをアップロードして保守します。 カスタム・クラスを管理するには、「設定」 > 「カスタム・クラス」をクリックします。

クラスをコンパイルした後、これを Guardium® システムにアップロードする必要があります。

カスタム・クラスのアップロード

1. アラートまたは評価のためのカスタム・クラスをアップロードできます。「設定」 > 「カスタム・クラス」をクリックしてから、「アラート」 > 「アップロード」または「評価」 > 「アップロード」のいずれかをクリックして、カスタム・クラスをアップロードします。
2. カスタム・クラスの説明を入力します。
3. 「参照」をクリックし、アップロードするクラス・ファイルを見つけ、選択します。
4. 「適用」をクリックします。

カスタム・クラスのアップデート

1. 「設定」 > 「カスタム・クラス」を選択してから、「アラート」 > 「更新」または「評価」 > 「更新」のいずれかを選択します。
2. 更新するクラスの「説明」を選択します。
3. 「参照」をクリックして、更新に使用するクラス・ファイルを見つけ、選択します。
4. 「適用」をクリックします。

カスタム・クラスの削除

1. 「設定」を選択してから、「アラート」 > 「削除」または「評価」 > 「削除」のいずれかを選択します。
2. 削除するクラスの「説明」を選択します。
   注: 他のコンポーネントで使用中のクラスは削除できません (インストールされたポリシーなど)。
3. 「削除」をクリックします。

親トピック: Guardium システムの管理

ブラウザーの SSL 証明書チャレンジを回避するためのアプライアンス証明書のインストール方法

IBM Security Guardium CLI コマンドを使用して、証明書署名要求 (CSR) の作成、および Guardium® システム上へのサーバー証明書、認証局 (CA) 証明書またはトラステッド・パス証明書のインストールを行います。

このタスクについて

以下のような証明書エラーの警告画面が表示されなくなります。

```
There is a problem with this website's security certificate.  The security certificate presented by this website was issued  for a different website's address. Security certificate problems may indicate an attempt to fool  you or intercept any data you send to the server.
```

すべての証明書コマンドの詳細については、『証明書 CLI コマンド』を参照してください。

注: 1 つの前提条件は、証明書の署名のために使用する CA (Verisign、Thwate、Geotrust、GoDaddy、Comodo、within-your-company など) からの公開証明書を用意しておくことです。
注: Guardium は、CA サービスは提供しません。また、デフォルトでインストールされているもの以外の証明書をシステムに添付することもありません。独自の証明書を希望するお客様は、サード・パーティー CA にお問い合わせください。
注: 証明書に自己署名がない場合、最低レベルのパブリック証明書 (例えば、自己署名がある証明書) を署名者ごとに取得する必要もあります。openssl x509 -in t.pem -text -noout コマンドを使用すると、x509 証明書の内容を表示できます。

手順

1. 証明書の署名のために使用する CA (認証局) (Verisign、Thwate、Geotrust、GoDaddy、Comodo、自社内など) から取得した公開証明書を使用できる状態にします。
2. 署名付きの証明書を配置する個々の Guardium システムで CLI にログインします。

   コマンドを実行する前に、該当する証明書 (バイナリー形式ではなく PEM 形式) を CA から取得し、その証明書 (Begin 行と End 行を含む) をクリップボードにコピーします。

3. コマンド store certificate keystore を入力します。次のプロンプトが表示されます。

   ```
   What is a one-word alias we can use to uniquely identify this certificate?
   ```

   証明書に付ける 1 単語の名前を入力し、Enter を押します。

   次の指示が表示されます。

   ```
   Please paste your CA certificate, in PEM format. Include the BEGIN and END lines, and then press CTRL-D.
   ```

   PEM 形式の証明書をコマンド行に貼り付けた後、CRTL-D を押します。保管操作の成功または失敗が通知されます。

   署名のために使用する CA が Guardium システムで信頼できる CA として設定されます。

4. 次に、CLI コマンド・プロンプトで `create csr gui` と入力します。

   要求された情報を入力します。証明書の CN (共通名) がボックスの hostname.domain に設定されていないと、ブラウザーから証明書エラーが生成されます。

   パラメーターはありませんが、部門 (OU)、国別コード (C)、などを提供するようプロンプトが出されます。この情報は必ず正確に入力してください。最後のプロンプトは次のようなものです。

```
What encryption algorithm should be used (1=DSA or 2=RSA)?
```

DSA (デジタル署名アルゴリズム) は、デジタル署名に関する連邦情報処理標準 (FIPS) です。RSA は、鍵の生成、暗号化、および暗号化解除を行う公開鍵暗号方式です。デフォルトの暗号化アルゴリズムは RSA です。

最後のプロンプトに応えた後、要求の説明とそれに続いて要求そのもの、さらに続いて追加の説明がシステムに表示されます。 例:

```
This is the generated CSR: Certificate Request: Data: Version: 0 (0x0) Subject: C=US, ST=MA, L=Littleton, O=XYZCorp,
OU=Accounting, CN=g2.xyz.com -----BEGIN NEW CERTIFICATE REQUEST-----
MIICWjCCAhcCAQAwVDELMAkGA1UEBhMCVVMxEDAOBgNVBAgTB1dhbHRoYW0xETAPBgNVBAoTCEd1
YXJkaXVtMRUwEwYDVQQLEwxndWFyZGl1bS5jb20xCTAHBgNVBAMTADCCAbgwggEsBgcqhkjOOAQB
MIIBHwKBgQD9f1OBHXUSKVLfSpwu7OTn9hG3UjzvRADDHj+AtlEmaUVdQCJR+1k9jVj6v8X1ujD2
y5tVbNeBO4AdNG/yZmC3a5lQpaSfn+gEexAiwk+7qdf+t8Yb+DtX58aophUPBPuD9tPFHsMCNVQT
WhaRMvZ1864rYdcq7/IiAxmd0UgBxwIVAJdgUI8VIwvMspK5gqLrhAvwWBz1AoGBAPfhoIXWmz3e
y7yrXDa4V7l5lK+7+jrqgvlXTAs9B4JnUVlXjrrUWU/mcQcQgYC0SRZxI+hMKBYTt88JMozIpuE8
FnqLVHyNKOCjrh4rs6Z1kW6jfwv6ITVi8ftiegEkO8yk8b6oUZCJqIPf4VrlnwaSi2ZegHtVJWQB
TDv+z0kqA4GFAAKBgQCONsEB4g4/limbHkuZ5YnLn9CGM3a2evEnqjXZts4itxeTYwPQvdkjdSmQ
kaQlBxmNUsZOJZrq5nC5Cg3X9spa+BzFr+PgR/5zka17nHcxKXCjVjLk451L67KllXv61TUfv/bU
PKmiaGKDttsP2ktG4dBFXQdICJEGo0aNFCYn6qAAMAsGByqGSM44BAMFAAMwADAtAhUAhHTY5z9X NiBAuyAC9PS4GzleYakCFF2kcfxfjX1BFy5I228XWMAU0N95
-----END NEW CERTIFICATE REQUEST-----
```

注: 共通名では、FQDN (完全修飾ドメイン名) 形式のホスト名を使用します。ただし、FDQN (system1.us.ibm.com など) の代わりに短縮ホスト名 (system1 など) を使用して GUI に通常接続すると、「アドレスが一致しません (Address Mismatch)」という証明書エラーを受け取るため、証明書を使用するために CN=system1 を変更するか、https://system1.us.ibm.com:8443/sqlguard を使用して接続する必要があります。
注: 国別コードは 2 文字でなければなりません。
注: 鍵サイズとして 1024 または 2048 を使用できます。

5. 生成されたハッシュを ---Begin CSR---- から ---End CSR--- までコピーして、テキスト文書に貼り付けます。この文書を CA に送信して、署名付きの鍵を送り返してもらいます。

   続行する前に「件名」行を調べて、会社の情報を正しく入力していることを確認します。 ここから先は、サーバー証明書を CA から取得する際に通常使用する手順に従います。

   注: 要求を CA に送信する場合、証明書を PKCS#7 PEM 形式にするように要求する必要があります。
6. CA が CSR に署名して、署名付きの鍵を送り返します。
7. Guardium システムの CLI プロンプトに戻って、CA から送られてきた署名付きの鍵を配置します。以下を入力します: store certificate gui。

   表示どおり正確にコマンドを入力します。下記の情報が表示され、プロンプトが出されます。

   Please paste your new server certificate, in PEM format.

   Include the BEGIN and END lines, and then press CTRL-D.

   PEM 形式の証明書をコマンド行に貼り付けた後、CRTL-D を押します。保管操作の成功または失敗が通知されます。

```
-----BEGIN CERTIFICATE----- MIIDvTCCAqegAwIBAgIBATALBgkqhkiG9w0BAQUwcjELMAkGA1UEBhMCVVMxEzAR
BgNVBAgTCldhc2hpbmd0b24xDzANBgNVBAcTBllha2ltYTEMMAoGA1UEChMDSUJN
MRUwEwYDVQQLEwxHdWFyZGl1bURlbW8xGDAWBgNVBAMTD0d1YXJkaXVtRGVtb19D
QTAeFw0xMTAzMjUxNTM1MTRaFw02OTEyMzEyMzU5NTlaMHIxCzAJBgNVBAYTAlVT
MRMwEQYDVQQIEwpXYXNoaW5ndG9uMQ8wDQYDVQQHEwZZYWtpbWExDDAKBgNVBAoT
A0lCTTEVMBMGA1UECxMMR3VhcmRpdW1EZW1vMRgwFgYDVQQDEw9HdWFyZGl1bURl
bW9fQ0EwggEgMAsGCSqGSIb3DQEBAQOCAQ8AMIIBCgKCAQEAw08aZVJNdnC69LR6
YtvHO+KbsqA89vCezLw7xmEa7F6+io0NofIFX7b7FvSkxzx1SO4eStaQSTDBxOGk
mqK2vk3VeJk9+lItofUuQXl1CZ1R4wQPMRfaWgELt+t94XB3Y1zmI68vwfr1fB32
u3Yjpt4aq27sTMrjEqZIyDqQ7hQ1tpMtoBUqNi54wN+OJjhtpNYDAkCHs+3NPqXE
6HeL7W5X6PJ+YCyyZiXeqQ+T8qdpH0KDVJGJLGX1YC+0WnQz/S2kaaRfxe6Nhe6q
YeYaD09tlWkVrZQm8a76SDULjzjrQ4wNoTJu17JQk7Uc835RE/bF5WMsaN5HGs3s
9zP3uwIDAQABo2QwYjAPBgNVHRMBAf8EBTADAQH/MA8GA1UdDwEB/wQFAwMHBgAw
HQYDVR0OBBYEFINmKThm8tA+Z8cyFC7MOZ7v398SMB8GA1UdIwQYMBaAFINmKThm
8tA+Z8cyFC7MOZ7v398SMAsGCSqGSIb3DQEBBQOCAQEAcXttcJwy1aowq9wtJksK
q6n6laEFR38i+pLJ6kArjoJGP5WxFdaYcDQr5cAw2Q6YFZvQGaYAqiSS6ezF20PT
3BrrP+Mg/SK8jgPvM0ekodmpr385iQqSDneTTwPPrIaQBrrtb2510wHSEyiVcRRI
4vn3ktVahjiSMD92bfmZilPYQ51pD0jFgGFFRvekulPWGWv7iuCT+alCM99/76xR
uWrRc7cxypfxK1yymptizZVrxLHS47VVoXzmZ7yO3kfhhdZbBmoXg1MDM82rVdnp
WVQdlSasn8deHaVG//RsCrWx4PxN8TVIDGbfh0nWRYU4zPORvWst3fa+h9B2W55z /A== -----END CERTIFICATE-----
```

8. 最後のステップとして、コマンド restart gui を使用して UI を再始動します。

   1 つの Guardium ユニットに 1 つの証明書を正常にインストールできました。オンサイトのすべての Guardium システムで手順を繰り返してください。

親トピック: Guardium システムの管理

GDPR 対応のための適用

特定の設定が変更されると、Guardium によって何らかの個人識別情報 (PII) が収集されることがあります。

1) ポリシー・ビルダー
   ポリシー・ビルダーでポリシー・ルール・アクションを設定しているときに「全詳細をロギング」にチェック・マークを付けると、Guardium コレクターにデータが返されます。調査対象のトラフィックのタイプによっては、PII が含まれていることがあります。詳しくは、『ルール・アクション』を参照してください。

2) 検査エンジン
   検査エンジンを構成しているときに「戻りデータの検査」にチェック・マークを付けると、結果セットを含むトラフィックのデータが Guardium コレクターに返されます。調査対象のトラフィックのタイプによっては、PII が含まれていることがあります。詳しくは、『検査エンジン構成』を参照してください。

GDPR 対応のための適用に関する以下のガイドラインに従ってください。

1) 暗号化

ポリシー・ルール・アクションを「*全詳細をロギング*」に構成するか、検査エンジンを「*戻りデータの検査*」に構成する必要があるお客様は、アプライアンスのディスク暗号化を検討することをお勧めします。詳しくは、『暗号化された LVM によるパーティション化の方法』を参照してください。

2) パージ間隔
例外をトリガーしたデータベース・トラフィックに PII が含まれていた場合、PII が含まれている可能性があるデバッグ情報が Guardium によって収集されることがあります。Guardium の管理者は、GUI パージ・パネルまたは CLI コマンド store purge objects age を使用してパージ間隔を設定することによって、データをパージできます。詳しくは、『調査ダッシュボードの有効化と無効化』を参照してください。
これらのいくつかの項目のデフォルトは、CLI 補完コマンド show purge objects age を使用して表示できます。間隔は、*日数* で定義されます。

3) SQL マスキング
PII が含まれている SQL 照会が失敗すると、Guardium によって PII が収集されることがあります。SQL 例外が発生したら、CLI コマンド snif_mask_sql_value を使用して、値をマスクしてください。このコマンドについて詳しくは、『snif_mask_sql_value』を参照してください。

親トピック: Guardium システムの管理

自己モニター

Guardium ソリューションは、自己モニターを行って、中断を最小限に抑え、可能な場合は常に問題を自動的に修正します。

Guardium ソリューションが使用可能で、正常に機能しており、改ざんされておらず、問題がある場合はユーザーに警告することを保証するために、以下の三方面からのアプローチが使用されます。

- レポート: 文字ベースの場合も、グラフィック・ベースの場合もありますが、レポートは Guardium® ソリューションの中核です。ユーザーは、Guardium のクエリー - レポート・ビルダーを使用することにより、関連付けられたドメインおよびエンティティーを介して収集された任意の自己モニター・データに関するレポートを効率的に作成できます。定義済みレポートの多くは、より詳細に指定することにより、より高い細分度を提供できるように改善できます。ドメインの脆弱性診断テストを使用して、セキュリティーの評価に使用できるテストに関するレポートを作成します。
- アラート: ユーザーは、レポートの作成に加え、定義したしきい値を使用して、それらのレポートに基づくアラートを定義できます。アラートは、例外やポリシー・ルール違反を示します。アラートの生成はリアルタイムか、または履歴分析により決定されます。次に、これらのアラートにより、SMTP、SNMP、syslog、カスタム Java™ クラスを使用したユーザーへの通知をトリガーすることができます。
- 自己モニター・ユーティリティー: Guardium では、内蔵の自己モニター・デーモン (常時稼働) サービス・ユーティリティーが、コレクターとアグリゲーターに実装されます。このユーティリティーは 5 分ごとに起動され、システム・スキャンを実行し、コンポーネントが最適に構成されているか、あるいは効率的に作動しているかをチェックして、必要に応じて修復します。例えば、Web サーバーがダウンしていることをユーティリティーが検出すると、まずサービスが完全にシャットダウンしているかどうかが検証され、サービスが再開されてから、管理ユーザーにアラートが送られます。

モニター対象のコンポーネント

表 1. モニター対象のコンポーネント

コンポーネント	アクセス方法
システム ディスク・スペース (使用量%)	「管理」>「システム・ビュー」>「システム・モニター」 アラート:「スニファーのバッファー」ドメイン および「スニファーのバッファー使用」エンティティーを利用して、照会および相関アラートを使用することにより、アラートを作成できます。
ディスク (/var) 上の DB サイズとファイル	ディスク (/var) 上の DB サイズとファイルが、今後 14 日間に 50% に到達する可能性があるとシステムが識別したとき、アラートが送信されます。アラートは、中央マネージャーの適用状態ダッシュボードに表示されます。 アラートは、/var 上の最大の表または最大のファイルを識別します。
CPU 負荷 アップタイムおよびリブート メモリー使用状況 モニター・エンジン (スニファー) - 状況: 作動中/ダウン/スタック/過負荷 CPU 使用量 メモリー使用状況 過負荷および遅延 (キュー)	「レポート」>「Guardium 運用レポート」>「バッファー使用状況モニター」 アラート:「スニファーのバッファー」ドメイン および「スニファーのバッファー使用」エンティティーを利用して、照会および相関アラートを使用することにより、アラートを作成できます。
失敗したログイン	「管理」>「システム・ビュー」>「システム・モニター」。 アラート:「Guardium ログイン」ドメインおよび「Guardium ユーザー・ログイン」エンティティーを利用して、照会および相関アラートを使用することにより、アラートを作成できます。
失われた要求	「管理」>「レポート」>「アクティビティー・モニター」>「ドロップされたリクエスト」 アラート:「例外」ドメインおよび「例外」エンティティーを利用して、照会および相関アラートを使用することにより、アラートを作成できます。
データ・パターンの変更	「レポート」>「リアルタイム運用レポート」>「変更された値」 アラート:『監査プロセス定義の表示』で、アラート「データ・ソースの変更」(すべてのデータ・ソースの変更にアラートを出す) を参照。

コンポーネント	アクセス方法
パケット・レート リクエスト・レート 無視されたデータ	「レポート」>「Guardium 運用レポート」>「バッファー使用状況モニター」 アラート:「スニファーのバッファー」ドメイン および「スニファーのバッファー使用」エンティティーを利用して、照会および相関アラートを使用することにより、アラートを作成できます。
スケジュールされたジョブの例外	「レポート」>「Guardium 運用レポート」>「スケジュールされたジョブの例外」、または『事前定義管理レポート』を参照。 アラート:「例外」ドメインおよび「例外タイプ」エンティティーを利用して、照会および相関アラートを使用することにより、アラートを作成できます。
監査プロセスの状況	「レポート」>「Guardium 運用レポート」>「アクティブな監査プロセスの数」、または『事前定義管理レポート』を参照。 アラート:「監査プロセス」ドメインおよび「監査プロセス」エンティティーを利用して、照会および相関アラートを使用することにより、アラートを作成できます。
検査エンジンの変更	「レポート」>「アクティビティー・モニター」>「S-TAP 構成変更履歴」 アラート:『監査プロセス定義の表示』で、アラート「検査エンジンと S-TAP」(検査エンジンと S-TAP の構成に関連するすべてのアクティビティーについてアラートを出す) を参照。
Guardium ユーザー・アクティビティー - ログイン/ログアウト	「レポート」>「Guardium 運用レポート」>「Guardium へのログイン」、または『事前定義管理レポート』を参照。 アラート:「Guardium ログイン」ドメインおよび「SQL Guard ログイン」エンティティーを利用して、照会および相関アラートを使用することにより、アラートを作成できます。
失敗したログイン	「レポート」>「Guardium 運用レポート」>「Guardium へのログイン」、または『事前定義管理レポート』を参照。 アラート:『監査プロセス定義の表示』で、アラート「Guardium への失敗したログイン」(最近の 11 分間でログインの失敗が 5 回を超えるとアラートを出す) を参照。または「ツール」>「レポートのビルド」>「レポート・タイトル」ドロップダウンで、「Guardium ログイン」を選択。詳細について、『レポート』を参照。
ユーザー・アクティビティー監査証跡	「レポート」>「Guardium 運用レポート」>「ユーザー・アクティビティー監査証跡」、または『事前定義管理レポート』を参照。 アラート:「Guardium アクティビティー」ドメインおよび「SQL Guard ユーザー・アクティビティー監査」エンティティーを利用して、照会および相関アラートを使用することにより、アラートを作成できます。 注: ユーザー・アクティビティーには、ユーザーがルート・シェルに変更する場合も含まれるため、ルートのアクティビティーのログが提供されます。
ユーザー/ロールの作成/削除	「レポート」>「Guardium 運用レポート」>「ユーザー・アクティビティー監査証跡」、または『事前定義管理レポート』を参照。 アラート:『監査プロセス定義の表示』で、アラート「Guardium - ユーザーの追加/削除」(Guardium ユーザーのすべての追加または削除にアラートを出す) を参照。
許可のモニター	「レポート」>「Guardium 運用レポート」>「Guardium ユーザー」、「Guardium のロール」、または「Guardium アプリケーション」 アラート:「アプリケーション」ドメインおよび「アプリケーション・データ」エンティティーを利用して、照会および相関アラートを使用することにより、アラートを作成できます。
S-TAP® 情報 (中央マネージャー)	レポート:『S-TAP レポート』を参照。中央マネージャーでは、追加のレポート、「S-TAP 情報」が使用可能です。このレポートは、環境全体の S-TAP をモニターします。このデータのアップロードには、カスタム表ビルダーを使用します。このレポートは、リモート・ソースを使用してデータを中央マネージャーにアップロードし、そのデータを使用して S-TAP の統合ビューを表示した結果です。 S-TAP 情報は、「S-TAP 情報」エンティティーが含まれる事前定義カスタム・ドメインであり、ライセンス・ドメインと違って変更できません。

Guardium Nanny プロセス

Guardium Nanny は、システムのクリティカル・リソースをモニターし、潜在的な問題が発生する際にアラートを出す内部プロセスです。Nanny のアラートは syslog に送られ、そこから転送されたり、E メールとして管理者に送信したりできます。場合により、修正処置がとられます。

Nanny は Guardium システム内のキー・コンポーネントおよびクリティカル・リソースを監視し、それらの可用性と信頼性を保証します。リソースおよびコンポーネントには以下が含まれます。

- Web サービスのモニター - サービス・ポート (デフォルトで 8443) が応答していない、または tomcat サービスが起動していない。
  - syslog メッセージ
  - 管理者にメール送信
  - Web サービス再始動の実行
- 検査エンジン・アクティビティー - スニフの過負荷、応答なし、または失敗。
  - syslog メッセージ
  - 管理者にメール送信

    - Guardium サポートにメール送信 (オプション)
    - 条件により、スニファを再始動して修正を試行
    - プロセスが停止した場合にスニファの respawn を試行
- ディスク・スペース使用状況 - 重要なパーティションで 75% 以上になった場合にアラートを出す。
    - syslog メッセージ
    - 管理者にアラート送信
    - 95% を超える場合に一時ファイルをクリーニングし、予防処置を実行
- アプライアンスへのログイン (ssh) の失敗 - 失敗した ssh ログイン試行に関する ssh デーモンのメッセージおよびアラートを確認する。
    - 管理者にメール送信 (既に syslog 内にある)
- 内部データベース (TURBINE) のモニター - サービスが開始されていること、状況、および容量使用状況モニターの検証。
    - syslog メッセージ
    - 管理者にメール送信
    - サービスの再始動
- ファイル・システムの使用状況 - Nanny.pl が 5 分ごとに /var のファイル・システムを検査して、/var ディレクトリーの使用率が 75% を超えるとアラートで警告し、/var ディレクトリーの使用率が 90% を超えるとクリティカル・アラートで警告してサービスを停止する。
    - syslog メッセージ
    - 管理者にアラート送信
    - 管理者がクリーンアップする必要がある (使用する CLI コマンドは show filesystem usage、clear filesystem dir、および restart stopped_services)

- アラートを介して Guardium システムをモニターする方法
  組み込み相関アラートとカスタム相関アラートを組み合わせて使用して、IBM Security Guardium システムのキャパシティー、パフォーマンス、可用性をモニターします。
- SNMP によるモニター
  Guardium システムには SNMP エージェントがインストールされており、guardiumsnmp という名前の SNMP コミュニティーを使用して読み取り専用アクセス権限が提供されています。
- 実行照会モニター
  「実行照会モニター」にはアクティブ・ユーザー照会の状況が表示され、これによってすべてのレポート/モニター照会のタイムアウト値を設定することができます。

親トピック: Guardium システムの管理

アラートを介して Guardium システムをモニターする方法

組み込み相関アラートとカスタム相関アラートを組み合わせて使用して、IBM Security Guardium システムのキャパシティー、パフォーマンス、可用性をモニターします。

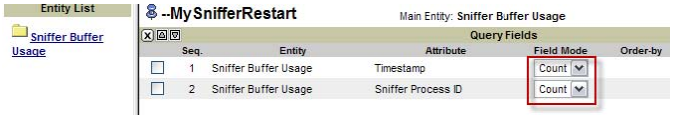
CPU 使用率、データベースのディスク・スペース、非アクティブ STAP、および「トラフィックなし」の各状態など、システムのパフォーマンスに影響を与える可能性のある問題についてユーザーに警告します。

「スニファーのバッファー使用」ドメインは、以下に示すアラートの大部分の基礎になっています。

スニファー再始動アラート

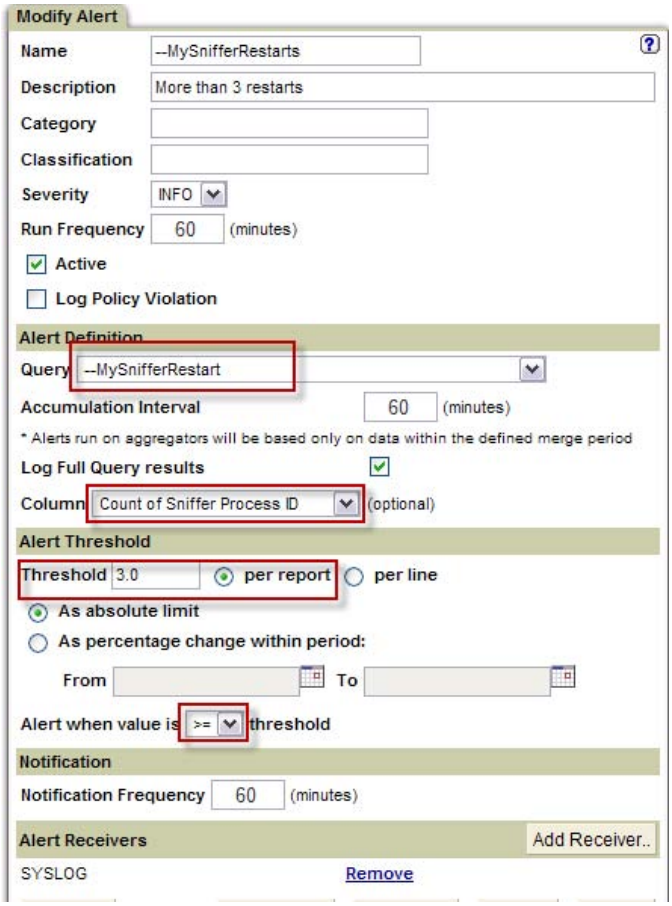
コレクター上のスニファーが 1 時間に 3 回以上再始動した場合にアラートが送信されます。

「スニファーのバッファー使用」ドメインを使用して、以下の列とフィールドを持つ照会を作成します。条件はありません。



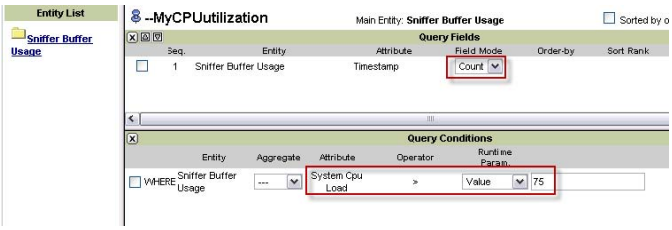
この照会の出力例を以下に示します。

次に、アラートを定義します。



高 CPU 使用率

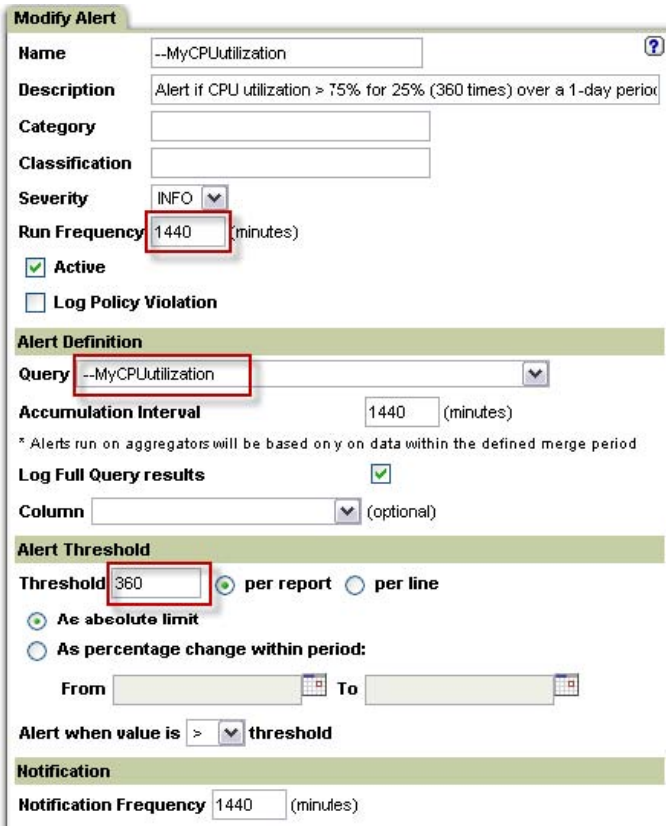
「エンタープライズ・バッファー使用状況」ドメインを使用して、システムの CPU 使用率をモニターするアラートを作成します。CPU 使用率が 75% を超える照会例を以下に示します。



このアラートは、75% の使用率を 24 時間で 360 回 (例えば、1 日の 25%) 超えた場合にのみ起動するように設定されます。

注:「スニファーのバッファー使用」ドメインには 1 分ごとにデータが取り込まれるため、24 時間で 1440 項目が生成されます。

アラートを定義するには、「保護」 > 「データベースの侵入検出」 > 「アラート・ビルダー」をクリックします。



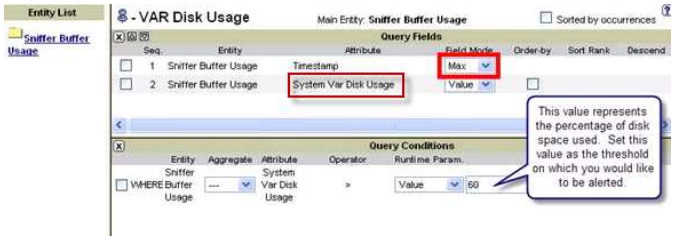
データベース・ディスク・スペースのアラート

クエリー・ビルダーを使用して 2 つのレポート (類似のもの) と 2 つのアラートを作成します。アラートの 1 つはコレクター用で、もう 1 つはアグリゲーター用です。これは、データベース・サイズはコレクターでは固定されていますが、アグリゲーターでは動的であるためです (最大で VAR パーティションのサイズまで)。

ディスク (/var) 上の DB サイズとファイルが、今後 14 日間に 50% に到達する可能性があるとシステムが識別したときに、アラートが自動的に送信されます。『ディスク (/var) 上の DB サイズとファイル』を参照してください。

アグリゲーター・ディスク・スペースのアラート

1. メイン・エンティティーとして「スニファーのバッファー使用」を持つ新しい照会を作成します。
2. フィールドと条件を構成します。

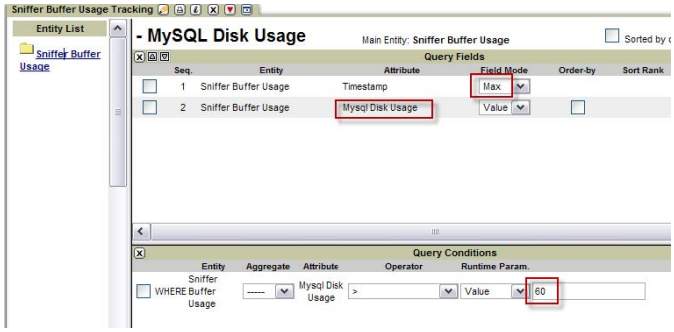


3. 「アラート・ビルダー」で新規アラートを設定します。「保護」 > 「データベースの侵入検出」 > 「アラート・ビルダー」をクリックして、「アラート・ビルダー」を開きます。

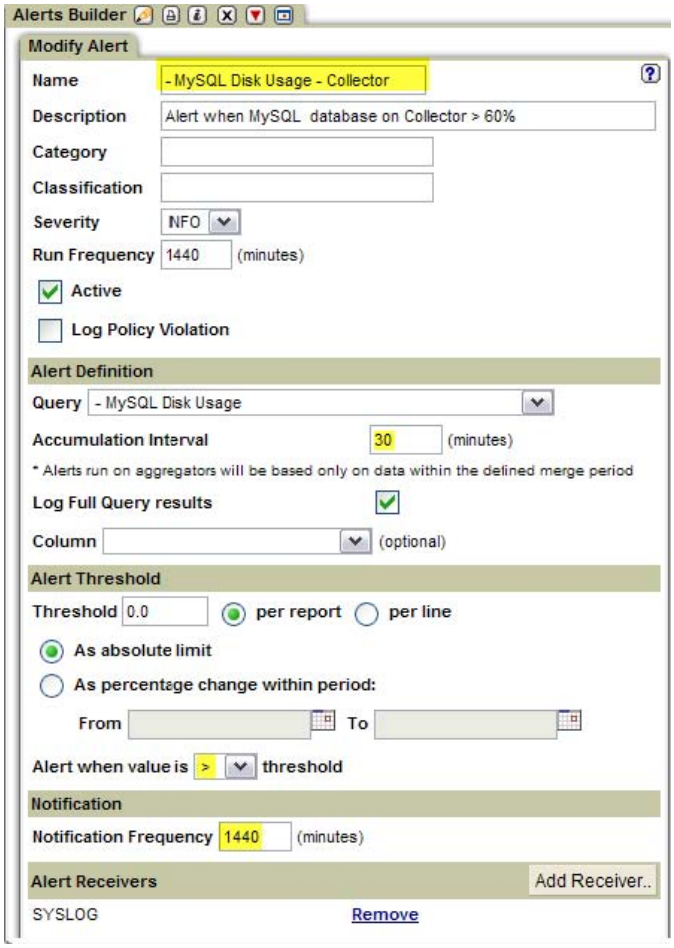
コレクター・ディスク・スペースのアラート

前のステップを繰り返して、コレクターのディスク・スペースをモニターするためのアラートを作成します。

1. 照会を作成します。



2. 「アラート・ビルダー」を使用して、新しいアラートを設定します。



データ・インポート、マージ (統合)、アーカイブ、または障害バックアップのアラート

これは、アクティブ化してスケジューリングする必要がある組み込みアラートです。

非アクティブ S-TAP アラート

これは組み込みアラートであり、アクティブ化してスケジューリングする必要があります。

1 次コレクターと 2 次コレクターで構成された S-TAP では、ネットワークの問題などが原因で S-TAP が 1 次コレクターと通信できない場合、2 次コレクターにフェイルオーバーします。元の 1 次コレクターが S-TAP を ping できない場合、非アクティブな S-TAP アラートが生成されます。

注: 構成が正しくない場合、クラスター構成内の S-TAP によって誤ったアラートが生成されることがあります。

「トラフィックなし」アラート

これは組み込みアラートであり、アクティブ化してスケジューリングする必要があります。

このアラートは、以前にコレクターがトラフィックを受信していたアクティブな検査エンジンからのトラフィックがあるかどうかを検査し、さらに、ポリシーによって処理されるトラフィックがあるかどうかを検査します。両方の条件が 48 時間以内に満たされない場合、アラートが生成されます。

随時レポートを介したアプリケーション・モニター

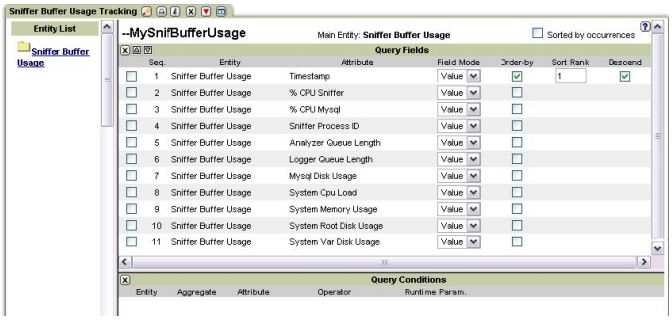
一般的なルールとして、1 時間を超える随時照会または随時レポートをコレクターで呼び出すことは避けてください。大規模な照会や、実行に長時間を要する照会は、アグリゲーターで呼び出す必要があります。監査プロセスを使用すると、最適なスケジュールを設定することができます。

以下の 2 つのレポートについて、各コレクターで毎週実行されるように、中央マネージャーからスケジューリングする必要があります。

注: また、これらのレポートは、それぞれのアグリゲーターで個別にスケジューリングする必要があります。

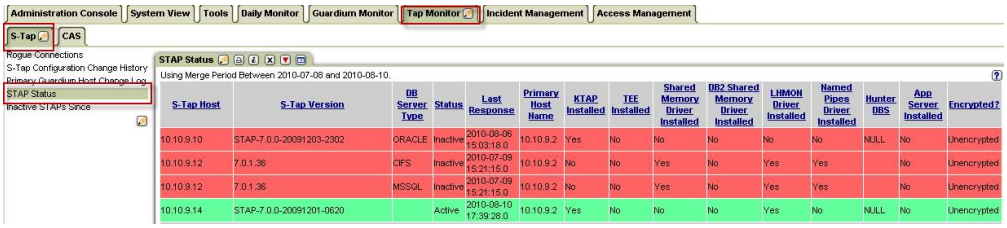
カスタムの「スニファーのバッファー使用」レポート

「スニファーのバッファー使用」ドメインを使用して、以下のフィールドを持つレポートを作成します。



S-TAP 状況レポート

このレポートには、特定のコレクターに関するすべての S-TAP と検査エンジンの主要なパラメーターが表示されます。 このレポートを変更することはできませんが、各コレクターで実行したり、各コレクターを指す中央マネージャーから実行したり、各コレクターの監査プロセスを介してスケジューリングしたりすることができます。



親トピック: 自己モニター

SNMP によるモニター

Guardium® システムには SNMP エージェントがインストールされており、guardiumsnmp という名前の SNMP コミュニティーを使用して読み取り専用アクセス権限が提供されています。

照会を行う際に、値 -1 (マイナス 1) は、データベースで NULL を示します。 このセクションの最後にある表に、使用可能な SNMP OID がリストされています。

SNMP の例

UNIX セッションから、snmpget または snmpwalk コマンドを使用して SQL Guard SNMP 情報を表示できます。 (コマンド構文を表示するには、snmpget -h または snmpwalk -h を使用します。) SNMP 情報を表示するために、さまざまな UI ベースのソフトウェア・パッケージを使用できます。これらの代替手段についてはここでは説明しません。

表 1. SNMP の例

SNMP の例
使用されている、および使用可能なディスク・スペース
> snmpget -v 2c -c guardiumsnmp a1.corp.com UCD-SNMP-MIB::dskAvail.1
UCD-SNMP-MIB::dskAvail.1 = INTEGER: 1043856
> snmpget -v 2c -c guardiumsnmp a1.corp.com UCD-SNMP-MIB::dskUsed.1
UCD-SNMP-MIB::dskUsed.1 = INTEGER: 914856
合計メモリーおよび使用されているメモリーをリストする場合
> snmpget -v 2c -c guardiumsnmp a1.corp.com
HOST-RESOURCES-MIB::hrStorageSize.101
HOST-RESOURCES-MIB::hrStorageSize.101 = INTEGER: 2067352
> snmpget -v 2c -c guardiumsnmp a1.corp.com HOST-RESOURCES-MIB::hrStorageUsed.101
HOST-RESOURCES-MIB::hrStorageUsed.101 = INTEGER: 1017548
使用可能メモリーをリストする場合
> snmpwalk -v 2c -c guardiumsnmp a1.corp.com memAvailReal
UCD-SNMP-MIB::memAvailReal.0 = INTEGER: 1049564
CPU 使用量に関連した値をリストする場合
> snmpwalk -v 2c -c guardiumsnmp a1.corp.com ssCpuRawUser
UCD-SNMP-MIB::ssCpuRawUser.0 = Counter32: 89240
> snmpwalk -v 2c -c guardiumsnmp a1.corp.com ssCpuRawSystem

SNMP の例
UCD-SNMP-MIB::ssCpuRawSystem.0 = Counter32: 195310
> snmpwalk -v 2c -c guardiumsnmp a1.corp.com ssCpuRawNice
UCD-SNMP-MIB::ssCpuRawNice.0 = Counter32: 11
注: RawUser、RawSystem、および RawNice 番号を追加すると、CPU 総使用量に近い概算を得ることができます。
> snmpwalk -v 2c -c guardiumsnmp a1.corp.com ssCpuRawIdle
UCD-SNMP-MIB::ssCpuRawIdle.0 = Counter32: 26734332

Guardium SNMP OID

表 2. Guardium SNMP OID

SNMP OID	記述
.1.3.6.1.4.1.2021.9.1.7.1 UCD-SNMP-MIB::dskAvail.1	/ ディレクトリーで使用可能なディスク・スペース
.1.3.6.1.4.1.2021.9.1.7.2 UCD-SNMP-MIB::dskAvail.2	/var ディレクトリーで使用可能なディスク・スペース
.1.3.6.1.4.1.2021.9.1.8.1 UCD-SNMP-MIB::dskUsed.1	/ ディレクトリーで使用されているディスク・スペース
.1.3.6.1.4.1.2021.9.1.8.2 UCD-SNMP-MIB::dskUsed.2	/var ディレクトリーで使用されているディスク・スペース
.1.3.6.1.2.1.25.2.3.1.5.1 HOST-RESOURCES-MIB::hrStorageSize.1	使用可能な合計メモリー
.1.3.6.1.2.1.25.2.3.1.6.1 HOST-RESOURCES-MIB::hrStorageUsed.1	使用されているメモリー
.1.3.6.1.4.1.2021.8.1.101.1 UCD-SNMP-MIB::extOutput.1	オープンしているモニター対象セッション数
.1.3.6.1.4.1.2021.8.1.101.2 UCD-SNMP-MIB::extOutput.2	現在のスニファー・プロセスにより記録された要求 (再始動ごとにゼロに設定する)
.1.3.6.1.4.1.2021.8.1.101.3 UCD-SNMP-MIB::extOutput.3	最終セッションのタイム・スタンプ
.1.3.6.1.4.1.2021.8.1.101.4 UCD-SNMP-MIB::extOutput.4	最終構成のタイム・スタンプ
.1.3.6.1.4.1.2021.8.1.101.5 UCD-SNMP-MIB::extOutput.5	スニファー・プロセスにより使用されているメモリー
.1.3.6.1.4.1.2021.8.1.101.7 UCD-SNMP-MIB::extOutput.7	ETH1 への到着パケット数/ETH2 からの送信パケット数。通常、SPAN ポートまたは TAP ポートが使用されるときには 1 つの数値だけ (到着) です。
.1.3.6.1.4.1.2021.8.1.101.8 UCD-SNMP-MIB::extOutput.8	ETH3 への到着パケット数/ETH4 からの送信パケット数。通常、SPAN ポートまたは TAP ポートが使用されるときには 1 つの数値だけ (到着) です。
.1.3.6.1.4.1.2021.8.1.101.9 UCD-SNMP-MIB::extOutput.9	ETH5 への到着パケット数/ETH6 からの送信パケット数。通常、SPAN ポートまたは TAP ポートが使用されるときには 1 つの数値だけ (到着) です。

マシンでアクセス可能な他の MIB は SNMPv2-MIB、IF-MIB、RFC1213-MIB、および HOST-RESOURCES-MIB です。

親トピック: 自己モニター

実行照会モニター

「実行照会モニター」にはアクティブ・ユーザー照会の状況が表示され、これによってすべてのレポート/モニター照会のタイムアウト値を設定することができます。

「管理」 > 「アクティビティー・モニター」 > 「実行照会モニター」をクリックして、「実行照会モニター」を開きます。

「実行照会モニター」から以下のことを実行できます。

- ポートレットで稼働中のすべてのレポートおよびモニターの照会タイムアウトを設定します。ポリシー・シミュレーション、監査プロセス、内部処理などの、他の照会処理は、このタイムアウト値の影響を受けません。デフォルトは 180 秒 (3 分) です。
- 現在実行中のユーザー照会を強制終了します。監査プロセスなど、このパネルにリストされている一部の照会は、指定した照会タイムアウトを超えている可能性があります。このようなことが予想されるのは、レポート/モニター照会タイムアウトがポートレットで実行中のレポートおよびモニターにのみ適用されるためです。

照会タイムアウトをデフォルト設定 (180 秒) を超えて長時間にわたって設定することはお勧めしません。この制限を超えて設定すると、特別なレポート・アクティビティーによってシステムに過負荷がかかる可能性が高くなります。

タイムアウト設定を変更するには、「レポート/モニター照会タイムアウト (秒)」に秒数を入力して、「更新」をクリックします。 更新が完了したことが通知されます。

親トピック: 自己モニター

グループ

グループを使用すると、分類、ポリシー、照会の各定義を簡単に作成および管理できるほか、更新を S-TAP クライアントおよび GIM クライアントに展開することができます。アクセス・ポリシーのデータ・オブジェクトのグループを繰り返し定義するのではなく、オブジェクトをグループに入れると、簡単に管理できます。

- グループの概要
  類似のデータ・オブジェクトをグループ化して、照会、ポリシー、および分類の各定義の作成に使用します。事前定義された多数のグループのうちの 1 つを使用するか、「グループ・ビルダー」を使用して独自のグループを作成します。
- グループ・ビルダーの使用
  グループ・ビルダーはグループ・メンバーシップとグループの使用について一目で確認できる情報を提供します。また、グループにデータを取り込むための便利な方法がいくつか用意されています。
- 照会およびポリシーでのグループの使用
  照会の条件演算子、およびポリシーでグループを使用する場所についての簡単な概要
- 例: グループを使用したルールとポリシーの作成
  グループを使用して、ポリシーのルール条件を素早く指定します。
- 事前定義グループ
  このセクションでは、Guardium® の事前定義グループについて詳しく説明します。

親トピック: Guardium システムの管理

グループの概要

類似のデータ・オブジェクトをグループ化して、照会、ポリシー、および分類の各定義の作成に使用します。事前定義された多数のグループのうちの 1 つを使用するか、「グループ・ビルダー」を使用して独自のグループを作成します。

グループの使用が役立つ状況は多くあります。類似のデータ・オブジェクトをグループ化することにより、複数のデータ・オブジェクトを個別に選択する必要なしに、ポリシー、分類、照会、およびレポートでオブジェクトのセット全体を使用できます。

照会またはポリシーに変更を加える必要がある場合、変更内容を各オブジェクトに個別に適用するのではなく、グループに適用できます。

S-TAP および GIM も、複数の管理対象サーバーにまたがって更新を容易に展開できるようにするために、グループを使用します。

グループ・ビルダー

「グループ・ビルダー」では、ユーザー・インターフェースから、新規グループを作成したり、既存のグループに変更を加えたりすることができます。

「設定」 > 「グループ・ビルダー」をクリックして、「グループ・ビルダー」を開きます。

「グループ・フィルター」画面では、アプリケーション・タイプ、グループ・タイプ、説明、またはカテゴリーに基づいてグループを簡単にソートできます。

グループのタイプ

「グループ・タイプ」フィールドは、一緒にグループ化されるデータのタイプを指します。 例えば、「*サーバー IP*」では、IP アドレスとして配列されたデータが予期され、「ユーザー」では、アプリケーションのユーザーの名前が示されることが予期されます。

タプル・グループ

タプル・グループでは、複数の属性を組み合わせて 1 つの複合グループ・メンバーを形成することができます。3 つの順序付き値セットは、3 タプルと呼ばれます。 n 個の値属性セットがあるものを n タプルと呼びます。 これにより、レポートおよびポリシー・ルールの条件の指定が簡略化されます。

以下にタプル・グループの例を示します。

- タプル・グループ - オブジェクト/コマンド、オブジェクト/フィールド、クライアント IP/データベース・ユーザー、サーバー IP/データベース・ユーザー
- 3 タプル・グループ - クライアント IP/ソース・プログラム/データベース・ユーザー、データベース・ユーザー/オブジェクト/特権
- 5 タプル・グループ - クライアント IP/ソース・プログラム/データベース・ユーザー/サーバー IP/サービス・インスタンス
- 7 タプル・グループ - クライアント IP/ソース・アプリケーション/データベース・ユーザー/サーバー IP/サービス名/OS ユーザー/データベース名

タプルには、1 つのスラッシュおよび 1 つのワイルドカード文字 (%) を使用できます。ダブルスラッシュ (//) の使用はサポートされません。
注: タプル・クエリー - ユーザーが LIKE GROUP 条件を使用しようとし、データ内に「¥」がある場合、結果は正しくない場合があります。データ内に「¥」がある場合、ユーザーは代わりに IN GROUP を使用する必要があります。

事前定義グループ

Guardium には、いくつもの事前定義されたグループが含まれています。「グループ・フィルター」および「グループ・タイプ」メニューを使用して、グループのリストを表示し、ニーズに最も適したグループを見つけます。

グループ・タイプ「*データベース・ユーザー/データベース・パスワード*」は、デフォルトで admin ユーザーのみが使用可能です。このデフォルト設定を変更する場合は、グループのロールを変更してください。

重複するグループ・メンバーシップ

グループ・メンバーを複数のグループに入れることができます。

例えば、2 つの事前定義グループ「*Create コマンド*」および「*DDL コマンド*」がいずれも「CREATE TABLE」という名前のメンバーを持つとします。いずれかの一方のグループを照会する場合、レポート期間のすべての CREATE TABLE メンバーがそのグループでカウントされます。

各メンバーが 1 つのグループにのみ属するようにグループ・セットを定義する場合があります。例えば、レポート目的で、データベース・ユーザーを「従業員」または「コンサルタント」の 2 つのグループのいずれかにグループ化する必要があるとします。 これらの各グループを同じサブグループ・タイプ (「雇用者の身分」など) で定義します。 サブグループが使用される場合、メンバーが同じサブグループ・タイプの別のグループに既に追加されていると、システムは、そのメンバーをサブグループに追加することを許可しません。

メンバー内のワイルドカード

グループが照会条件やポリシー・ルールで使用される場合に、グループ・メンバーにワイルドカード (%) 文字を含めることができます。

表 1. メンバー内のワイルドカード

メンバー	一致	不一致
aaa%	aaa aaazzz	zzzaaa aaz
%bbb	bbb,zzbbb	bb bbbzzz
%ccc%	ccc ccczz zzzccczzz	cc zzzcczzz

管理対象ユニット・グループ

ポリシーの作成および管理を簡素化するために、およびレポート表示を明確にするために要素をグループ化しますが、この要素のグループ化に使用される管理対象ユニット・グループとグループ・ビルダーを介して作成されたグループには明確な違いがあります。管理対象ユニット・グループについて詳しくは、管理対象ユニット・グループの作成を参照してください。

親トピック: グループ

グループ・ビルダーの使用

グループ・ビルダーはグループ・メンバーシップとグループの使用について一目で確認できる情報を提供します。また、グループにデータを取り込むための便利な方法がいくつか用意されています。

グループ・ビルダーを使用して、グループを作成し、CSV ファイル、外部データ・ソース、および既存のグループを含むさまざまなソースからグループにデータを取り込みます。さらに、グループ・ビルダーは、グループ・メンバーシップについて、またグループがセキュリティー・ポリシー、分類ポリシー、照会、およびレポートのどこで使用されているかについて、一目で確認できる情報を提供します。

グループ・ビルダーには、「設定」 > 「ツールとビュー」 > 「グループ・ビルダー」でアクセスできます。

- グループの作成および編集
  グループの作成方法と編集方法について説明します。
- グループ・メンバーシップおよびグループの使用場所の表示
  グループ・メンバーシップの表示方法およびグループが使用されているポリシー、レポート、照会の識別方法について説明します。
- グループへの取り込み
  グループ・ビルダーでは、メンバーをグループへ追加するための複数の方法がサポートされています。

親トピック: グループ

グループの作成および編集

グループの作成方法と編集方法について説明します。

親トピック: グループ・ビルダーの使用

グループの作成

手順

1. 「設定」 > 「ツールとビュー」 > 「グループ・ビルダー」にナビゲートして、グループ・ビルダーを開きます。

2. 「グループ・ビルダー」表で  アイコンをクリックします。
3. 「新規グループの作成」ダイアログを使用して、新規グループを定義します。 グループの説明を入力し、「アプリケーション・タイプ」メニューと「グループ・タイプ」メニューを使用してグループを定義します。
4. 新規グループを定義したら、「メンバー」タブを使用してグループにデータを取り込みます。 グループへのデータの取り込みについては、グループへの取り込みを参照してください。
5. 「保存」をクリックして、新規グループの定義を終了します。

グループの編集

手順

1. 「設定」 > 「ツールとビュー」 > 「グループ・ビルダー」にナビゲートして、グループ・ビルダーを開きます。
2. 「グループ・ビルダー」表からグループを選択し、  アイコンをクリックします。
3. 「グループの編集」ダイアログを使用してグループの設定を変更します。 グループへのメンバーの追加またはグループ・メンバーシップの変更を行うには、「メンバー」タブを使用します。グループへのデータの取り込みについては、グループへの取り込みを参照してください。
4. 「保存」をクリックしてグループの編集を終了します。

グループ・メンバーシップおよびグループの使用場所の表示

グループ・メンバーシップの表示方法およびグループが使用されているポリシー、レポート、照会の識別方法について説明します。

親トピック: グループ・ビルダーの使用

グループ・メンバーシップの表示

このタスクについて

「グループ・ビルダー」表の「メンバー」列と「データ設定元」列は、グループ内のメンバー数およびグループへのデータの取り込み方法を示します。以下の手順では、グループ・メンバーシップと、グループへのデータの取り込み方法についての詳細情報を取得する方法について説明します。

手順

1. 「設定」 > 「ツールとビュー」 > 「グループ・ビルダー」にナビゲートして、グループ・ビルダーを開きます。
2. 「グループ・ビルダー」表からグループを選択し、  アイコンをクリックすることで、「グループの編集」ダイアログを開きます。
3. 「グループの編集」ダイアログで「メンバー」タブをクリックして、グループ・メンバーシップを表示します。

グループの使用場所の識別

このタスクについて

「グループ・ビルダー」表の「分類で使用」、「ポリシーで使用」、および「照会で使用」の各列は、グループが Guardium のどこで使用されているかについての概要を示します。以下の手順では、グループが使用されているポリシー、照会、およびレポートの詳細情報を取得する方法を説明します。

手順

1. 「設定」 > 「ツールとビュー」 > 「グループ・ビルダー」にナビゲートして、グループ・ビルダーを開きます。
2. 「グループ・ビルダー」表からグループを選択して「アクション」 > 「詳細表示」をクリックすることで、詳細パネルを開きます。
   重要: 「詳細表示」アクションは、選択されたグループがポリシーや照会などによって使用されている場合にのみ有効になっています。
3. 詳細パネルの「ポリシー」タブと「照会」タブを使用して、選択したグループがセキュリティー・ポリシー、分類ポリシー、照会、およびレポートのどこで使用されているかを表示します。

グループへの取り込み

グループ・ビルダーでは、メンバーをグループへ追加するための複数の方法がサポートされています。

手順

1.  アイコンをクリックして新規グループを作成するか、「グループ・ビルダー」表からグループを選択し、  アイコンをクリックして既存のグループを編集します。
2. 「新規グループの作成」ダイアログまたは「グループの編集」ダイアログの「メンバー」タブを選択します。
3. 以下のいずれかの方法を使用して、グループにデータを取り込みます。
   -  アイコンを使用して、グループ・メンバーを手動で定義します。
   - 「インポート」メニューを使用して、以下のいずれかの方法を使用してグループ・メンバーを追加します。
     - CSV から
     - グループから
     - 外部データ・ソースから
     - 照会から
     - LDAP から

     ヒント: 構成されると、スケジュール可能なインポート・アクションが「新規グループの作成」ダイアログまたは「グループの編集」ダイアログにタブとして表示されます。「CSV からインポート」などの一回限りのアクションはスケジュールできないため、ダイアログ上に新しいタブは表示されません。
   - 一部のグループ・タイプでは、グループへデータを取り込むための方法として、以下の拡張された方法もサポートされています。

- データ・ソースに対するストアード・プロシージャー分析の使用
- データベース従属関係の使用
- 逆従属関係の使用
- 監視対象プロシージャーの使用
- 選択したオブジェクトの生成

重要: Guradium V10.1.4 で導入されたグループ・ビルダーを使用すると、拡張インポート・アクションはターゲット・グループに対して呼び出されます。ターゲット・グループには、ユーザー選択の入力グループに対して実行される分析の結果に基づいてデータが取り込まれます。これは、レガシー・グループ・ビルダーからの動作の変更を意味します。レガシー・グループ・ビルダーでは、拡張アクションは分析対象の入力を含むソース・グループに対して呼び出され、ユーザー選択のグループに分析結果がインポートされました。

- 外部データ・ソースからのインポート
  Guardium グループに独自データベースのデータを素早く追加し、それらのグループとデータの同期を保つ方法について説明します。

親トピック: グループ・ビルダーの使用

外部データ・ソースからのインポート

Guardium グループに独自データベースのデータを素早く追加し、それらのグループとデータの同期を保つ方法について説明します。

このタスクについて

「インポート」 > 「外部データ・ソースから」を使用すると、独自データ・ソースのデータを Guardium グループに追加するためのカスタム表、ドメイン、および照会の作成が自動化されます。これらの成果物は、作成後、Guardium とデータの間の永続的な接続を表します。データを更新すると、関連する Guardium グループに反映されます。

手順

1. 「インポート」 > 「外部データ・ソースから」を選択して「外部データ・ソースからインポート」ダイアログにアクセスします。
2. 「データ・ソース」メニューを使用して、データ・ソースからデータをインポートします。 アイコンをクリックして新規データ・ソースを定義するか、 アイコンをクリックして既存のデータ・ソースを編集します。
3. 「表名」フィールドおよび「列名」フィールドを使用して、データ・ソースからインポートするデータの場所を指定します。
4. 「OK」をクリックして先に進みます。

タスクの結果

「外部データ・ソースからインポート」ダイアログでの入力を完了すると、以下の Guardium 成果物が自動的に作成または更新されます。

- カスタム表
- カスタム・データ・ソース
- カスタム・ドメイン
- カスタム・クエリー
- グループ

これらの成果物は、次の表で説明されている命名規則を使用して、標準 Guardium ツールを通じて使用できます。この表の *[table name]* と *[column name]* は、「外部データ・ソースからインポート」ダイアログの「表名」フィールドと「列名」フィールドから取得されます。

表 1. 外部データ・ソースからインポート: 作成される成果物の概要

成果物	Guardium ツール	命名規則	例	スケジュール済み
カスタム表	「カスタム表ビルダー」 > 「データの編集」	*[table name]_[column name]_[datasource ID]*	`USERS_ADMIN_12345`	
カスタム・データ・ソース	「カスタム表ビルダー」 > 「データのアップロード」	*[datasource name]_[datasource type](カスタム・ドメイン)*	`user_repository`(カスタム・ドメイン)	✓
カスタム・ドメイン	「カスタム・ドメイン・ビルダー」	*[group type]_[table name]_[column name]_[datasource ID]*	`USERS_USERS_ADMIN_12345`	
カスタム・クエリー	「カスタム・クエリー・ビルダー」	*[group type]_[table name]_[column name]_[datasource ID]*	`USERS_USERS_ADMIN_12345`	
グループ	「グループ・ビルダー」 > 「照会から取り込み」		PCI 管理ユーザー	✓

重要: インポートされた名前は、64 文字より後が切り捨てられます。
親トピック: グループへの取り込み

照会およびポリシーでのグループの使用

照会の条件演算子、およびポリシーでグループを使用する場所についての簡単な概要

照会

照会では、条件演算子とグループが使用されます。以下に、各条件演算子の例を示します。

- IN GROUP - 値が、選択したグループの任意のメンバーと一致する場合、 条件は真になります。IN ALIASES GROUP 演算子は、IN GROUP と同じタイプのグループに対して機能しますが、 そのグループのメンバーが別名であると想定します。 IN GROUP 演算子はグループが実際の値を、IN ALIASES GROUP 演算子はグループが別名を含んでいることを予期します。 クエリー・ビルダー・ビルダーは、グループ内の別名値に一致するデータベース値のレコードを探します。
- NOT IN GROUP - 値が、選択したグループのどのメンバーとも一致しない場合、 条件は真になります。NOT IN ALIASES GROUP は、NOT IN GROUP と同じタイプのグループに対して機能しますが、そのグループのメンバーが別名であると想定します。
- IN DYNAMIC GROUP - 値が、ランタイム・パラメーターとして指定された任意のグループのメンバーと一致する場合、条件は真になります。 IN DYNAMIC ALIASES GROUP は、IN DYNAMIC GROUP と同じタイプのグループに対して機能しますが、 そのグループのメンバーが別名であると想定します。
- NOT IN DYNAMIC GROUP - 値が、ランタイム・パラメーターとして指定されるグループの任意のメンバーに一致しない場合、 条件は真になります。NOT IN DYNAMIC ALIASES GROUP は、NOT IN DYNAMIC GROUP と同じタイプのグループに対して機能しますが、 そのグループのメンバーが別名であると想定します。
  注: グループには、使用する演算子 (IN GROUP または IN ALIASES GROUP) に応じて、別名が含まれる場合も実値が含まれる場合もありますが、これらの演算子を同時に使用することはできません。
- LIKE GROUP - 値が、選択したグループのいずれかのメンバーと類似している場合、条件は真になります。この条件では、グループ・メンバー名にワイルドカード (%) 文字を使用できます。
  注: LIKE メンバー値では、値の全部または一部に一致する 1 つ以上のワイルドカード (%) 文字が使用されます。 LIKE 比較では、英字に大/小文字の区別はありません。 例えば、%tea% は tea、 TeA、tEam、または steam と一致します。

ポリシーおよびルール

ポリシーの一部としてルールを作成する場合は、グループを使用することで、目的のパラメーターを指定するプロセスを簡略化できます。

「ルール定義」ペインの「グループ」ドロップダウン・メニューがある場所で、グループを選択できます。

さらに、グループを急いで作成または変更する必要が生じた場合は、「グループ」アイコン をクリックして「グループ定義」ウィンドウを開き、必要な変更を加えます。

例: 実動サーバー上で発生しているアクティビティーをキャプチャーする場合は、完全 IP アドレスを毎回入力する代わりに、*「実動サーバー」* というグループを作成して、これを使用できます。

親トピック: グループ

例: グループを使用したルールとポリシーの作成

グループを使用して、ポリシーのルール条件を素早く指定します。

このタスクについて

各ポリシーは、1 つ以上のルールで構成されています。ルールを規定する条件を指定し、そのルールがトリガーされたときに実行するアクションを 1 つ以上選択します。この例では、グループを使用して無許可ユーザーを識別し、機密オブジェクトのグループへのそれらのユーザーのアクセスの詳細をログに記録し、アクセスが発生したことを示すアラートを送信する方法を説明します。

手順

1. Guardium システムにログインし、「設定」 > 「ツールとビュー」 > 「データのポリシー・ビルダー」をクリックして、「ポリシー・ビルダー」を開きます。
2. アイコンをクリックして、「ポリシー定義」ウィンドウを開き、新規ポリシーを作成します。
3. ポリシー定義を定義し、「適用」をクリックしてポリシーを保存します。
4. 「ルールの編集」をクリックして「ポリシー・ルール」ウィンドウを開き、ポリシーへのルールの追加を開始します。
5. 「ルールの追加」 > 「アクセス・ルールの追加」をクリックして、ポリシーに新しいルールを追加します。
6. 最初にルールの「記述」を指定します。オプションで、「カテゴリー」ラベルおよび「分類」ラベルを指定します。
7. データの検索場所を指定します。「サーバー IP」行で、「(パブリック) PCI 許可されたサーバー IP」グループを選択します。 ルールは、すべての PCI サーバーからのすべてのアクティビティーに適用されます。
   注: 「グループ・ビルダー」に移動して、任意のグループのメンバーを表示したり、任意のグループを変更したりすることができます。
8. 無許可ユーザーを指定します。「データベース・ユーザー」行で「Not」チェック・ボックスにマークを付け、「(パブリック) 許可されたユーザー」グループを選択します。 「*(パブリック) 許可されたユーザー*」グループに所属しないすべてのユーザーにルールが適用されます。
9. 機密オブジェクトを指定します。「オブジェクト」行で、「(パブリック) PCI カード所有者の機密オブジェクト」を選択します。 これで、ルールは、PCI 機密オブジェクトにアクセスしようとしている PCI サーバー上のすべての無許可ユーザーに適用されます。
10. 「アクションの追加」をクリックし、メニューから「アクション」 > 「全詳細をロギング」を選択して、ルールにアクションを追加します。「適用」をクリックしてルールを保存します。 このアクションにより、アクセスの正確なタイム・スタンプなど、アクセスの詳細がログに記録されます。
11. 「アクションの追加」をクリックし、メニューから「アクション」 > 「セッションごとに 1 回アラート」を選択して、ルールに別のアクションを追加します。アラートの宛先を指定し、「適用」をクリックしてルールを保存します。 このアクションにより、ルールがトリガーされたことを示すアラートが送信されるか、ログに記録されます。
12. 「保存」をクリックして、ルールを保存します。
13. ポリシーをインストールします。
    a. 作成したポリシーを検索します。「戻る」を 2 回クリックするか、「ポリシー・ビルダー」をクリックして「ポリシー・ファインダー」に移動し、ポリシーのリストを参照します。
    b. ポリシーを選択した状態で、インストール・アクション・メニューから「インストールおよびオーバーライド」を選択します。
    c. 「OK」をクリックして、ポリシーのインストールを確認してから、「最近のログと違反」にチェック・マークを付けて、ポリシーがインストールされたことを確認します。

    これで、ポリシーはインストールされてアクティブになっています。「*(パブリック) 許可されたユーザー*」グループに属していないいずれかのユーザーが「*(パブリック) PCI カード所有者の機密オブジェクト*」グループ内のオブジェクトにアクセスしようとすると、そのセッションがログに記録され、アクセスを示すアラートがトリガーされます。

親トピック: グループ

事前定義グループ

このセクションでは、Guardium® の事前定義グループについて詳しく説明します。

次の表では、Guardium システムに含まれている事前定義グループについて説明しています。すべてのグループのリストを表示するには、「設定」 > 「グループ・ビルダー」をクリックして、「グループ・ビルダー」を開きます。「アプリケーション」メニューから「*SQL_APP_NAME*」を選択して、「次へ」をクリックします。次の画面の「選択されたグループ」から、メンバーの管理を行います。*グループ・タイプ*という用語は、ラベルによって示されるデータ・タイプの予期を指します。 例えば、グループ・タイプ「*サーバー IP*」では、IP アドレスとして配列されたデータ (192.168.1.0) が予期され、グループ・タイプ「ユーザー」では、アプリケーションのユーザーの名前が示されることが予期されます。

事前定義グループは定期的に追加され、この追加の事前定義グループについてはここに説明されていない場合があります。 「グループ・ビルダー」を開き、すべての既存のグループを表示します。

グループ・タイプ「データベース・ユーザー/データベース・パスワード」の事前定義グループは、admin のロールを持つユーザーにしか許可されていません。 ユーザーは好みに応じて他のロールを追加できます。さらには、グループに対してすべてのロールを許可することもできます。

表 1. 事前定義グループ

SQL_APP_NAME	GROUP_DESCRIPTION	MEMBERS
DB2® zOS グループ	zOS 監査動的 SQL	Db2 コマンドのグループ・タイプ
Db2 zOS グループ	zOS 監査照会	Db2 コマンドのグループ・タイプ
Db2 zOS グループ	zOS 監査のアップデート	Db2 コマンドのグループ・タイプ
Db2 zOS グループ	zOS 監査の削除	Db2 コマンドのグループ・タイプ
Db2 zOS グループ	zOS 監査の挿入	Db2 コマンドのグループ・タイプ
Db2 zOS グループ	zOS 監査ユーティリティー	Db2 コマンドのグループ・タイプ
Db2 zOS グループ	zOS 監査オブジェクト・メンテナンス	Db2 コマンドのグループ・タイプ
Db2 zOS グループ	zOS 監査ユーザー・メンテナンス	Db2 コマンドのグループ・タイプ
Db2 zOS グループ	zOS 監査のユーザー許可の変更	Db2 コマンドのグループ・タイプ
Db2 zOS グループ	zOS 監査 Db2 コマンド	Db2 コマンドのグループ・タイプ
Db2 zOS グループ	zOS 監査計画/パッケージ・メンテナンス	Db2 コマンドのグループ・タイプ
IMS zOS グループ	zOS IMS 監査照会	IMS コマンドのグループ・タイプ
IMS zOS グループ	zOS IMS 監査のアップデート	IMS コマンドのグループ・タイプ
IMS zOS グループ	zOS IMS 監査の削除	IMS コマンドのグループ・タイプ
IMS zOS グループ	zOS IMS 監査の挿入	IMS コマンドのグループ・タイプ
IMS zOS グループ	zOS IMS 監査データベース・コマンド	IMS コマンドのグループ・タイプ
ポリシー・ビルダー	カード所有者オブジェクト	グループ・タイプ、オブジェクト
ポリシー・ビルダー	財務オブジェクト	グループ・タイプ、オブジェクト
ポリシー・ビルダー	PHI オブジェクト	グループ・タイプ、オブジェクト
ポリシー・ビルダー	許可されたクライアント IP	グループ・タイプ、クライアント IP
ポリシー・ビルダー	実動ユーザー	グループ・タイプ、ユーザー
ポリシー・ビルダー	PII オブジェクト	グループ・タイプ、オブジェクト
ポリシー・ビルダー	実動サーバー	グループ・タイプ、サーバー IP
ポリシー・ビルダー	財務サーバー	グループ・タイプ、サーバー IP
ポリシー・ビルダー	機能ユーザー	グループ・タイプ、ユーザー
ポリシー・ビルダー	Sharepoint サーバー	グループ・タイプ、サーバー IP
セキュリティー・アセスメント・ビルダー	Db2 データベース Version+Patches Informix® データベース Version+Patches MS SQL Server データベース Version+Patches MySql データベース Version+Patches Netezza® Version+Patches Oracle データベース Version+Patches Postgress Version+Patches Sybase データベース Version+Patches Teradata PDE Version+Patches Teradata TDBMS Version+Patches Teradata TDGSS Version+Patches Teradata TGTW Version+Patches	(特定の) データベース・バージョンおよびパッチ・レベルのテストに使用

SQL_APP_NAME	GROUP_DESCRIPTION	MEMBERS
セキュリティー・アセスメント・ビルダー	Db2 許可された PUBLIC への特権の付与 Informix 許可された PUBLIC への特権の付与 MS-SQL 許可された PUBLIC への特権の付与 MYSQL 許可された PUBLIC への特権の付与 Netezza 許可された PUBLIC への特権の付与 Oracle 許可された PUBLIC への特権の付与 Postgres 許可された PUBLIC への特権の付与 Teradata 許可された PUBLIC への特権の付与	TUPLE、 オブジェクト/コマンド・アプリケーション 8 (セキュリティー・アセスメント) パブリックへの特権の付与が許可されているオブジェクト/コマンドのリスト これらのオブジェクトは、パブリックへの特権付与をチェックする MS-SQL および Sybase のテストではスキップされます。 注: 例外グループには、正規表現を含めることも、メンバーだけを含めることもできます。 正規表現の場合、グループ・メンバーは (R) (大/小文字の区別あり) で始める必要があり、(R) の後ろに正規表現としてチェックされる詳細なレコードが続きます。 例えば、次のようなグループ・メンバーがあるとします。 (R)SYSTEM.[a-z]+ この場合、各レコードの詳細はパターン SYSTEM.[a-z]+ を使用してチェックされます。 メンバーが (R) で始まらない場合、レコード詳細はそのグループ・メンバーと等しい場合のみ例外と見なされます。 グループには正規表現と特定の例外を混用できることに注意してください。
セキュリティー・アセスメント・ビルダー	MS-SQL 許可された拡張プロシージャー	グループ・タイプはオブジェクト
セキュリティー・アセスメント・ビルダー	MS-SQL データベース管理者	グループ・タイプはユーザー
セキュリティー・アセスメント・ビルダー	Teradata プロファイル	グループ・タイプはオブジェクト
パブリック	アカウント管理コマンド	アカウント (ユーザー、ロール、アクセス権) の保守に使用されるコマンド。例: REVOKE、GRANT、ALTER/CREATE/DROP USER
パブリック	アカウント管理プロシージャー	アカウント (ユーザー、ロール、アクセス権) の保守に使用されるアカウント管理オブジェクト、ストアード・プロシージャー
パブリック	アクティブ・ユーザー	グループ・タイプはユーザー
パブリック	管理者ユーザー	デフォルトの管理ユーザー (DBA および SysAdmin)
パブリック	管理オブジェクト	特権オブジェクト。DBA アカウントまたは Sys アカウントのみがアクセスできるオブジェクト。これらのアカウントは、デフォルトでは「パブリック」に対してロックされています。
パブリック	管理コマンド	特権コマンド。 特権コマンドは、DBA だけが実行できるコマンドです。例: GRANT、BACKUP、DDL の各コマンド
パブリック	管理プログラム	データベースに同梱されているデータベース・ユーティリティー (クライアント) で、通常はデータベース・サーバーに置かれ、サーバー自体で使用できます。
パブリック	ALTER コマンド	例: alter database、alter procedure、alter profile、alter session、 alter user
パブリック	アプリケーション特権コマンド	「パブリック」から取り消す必要があるが、アプリケーションによって使用されているために取り消しできないパブリック特権コマンド。
パブリック	アプリケーション特権プロシージャー	アプリケーション特権オブジェクト。 「パブリック」から取り消す必要があるが、アプリケーションが使用しているために取り消しできないパブリック特権プロシージャー。
パブリック	アプリケーション・スキーマ・ユーザー	アプリケーション・ユーザー。 アプリケーションがアプリケーション表の保守/使用に使用するデータベース・ユーザー。
パブリック	アーカイブ候補	グループ・タイプはオブジェクト
パブリック	許可されたソース・プログラム	グループ・タイプはソース・プログラム
パブリック	許可されたユーザー	グループ・タイプはユーザー
パブリック	接続プロファイル・リスト	グループ・タイプは、クライアント IP/ソース・アプリケーション/DB ユーザー/サーバー IP/SVC です。 名前 許可される接続のリスト
パブリック	CREATE コマンド	例: create context、create database link、create function、create statistics、create type、create user
パブリック	資格情報関連エンティティー	Guardium 監査タイプ、 自己モニター。例: allowed_role、LDAP_config、Turbine_user_group_role

SQL_APP_NAME	GROUP_DESCRIPTION	MEMBERS
パブリック	データ転送コマンド	バックアップ・コマンド。 データベース・データのバックアップ/リストアを処理するコマンド
パブリック	データ転送プロシージャー	データ転送オブジェクト。 データベース・データ (主に MSS および SYB 上にある) のバックアップ/リストアを処理するプロシージャー
パブリック	データベース定義済みユーザー	非 admin の定義済みユーザー、または管理ユーザーを含むすべての定義済みユーザー
パブリック	DBCC コマンド	グループ・タイプはコマンド
パブリック	DDL コマンド	データ定義言語、 スキーマ特権コマンド。例: ALTER、CREATE、DROP
パブリック	DML コマンド	DML コマンド。 例: insert、truncate、update
パブリック	DROP コマンド	例: drop_context、drop_event_monitor、drop_procedure、drop_role
パブリック	DW すべてのオブジェクト・フィールド DW すべてのオブジェクト DW EXECUTE がアクセスしたオブジェクト DW SELECT がアクセスしたオブジェクト DW SELECT がアクセスしたオブジェクト/フィールド	モニター対象データを使用してオブジェクト名を表示する事前定義レポートが 5 つ あります。これらのレポートはすべて接頭部 DW (Data Warehouse) で始まります。これらの 事前定義レポートの使用法について詳しくは、ヘルプ・トピック『休止表/列のレポート方法』を参照してください。
パブリック	EBS アプリケーション・サーバー	グループ・タイプはクライアント IP
パブリック	EBS データベース・サーバー	グループ・タイプはサーバー IP
パブリック	EXECUTE コマンド	例: call、execute、execute function
パブリック	GRANT コマンド	例: grant、grant objectives、grant system privileges
パブリック	Guardium 詳細報告書用監査カテゴリー	Guardium パッチ。 TURBINE_USER_GROUP_ROLE
パブリック	ICM アプリケーション・サーバー	グループ・タイプはクライアント IP
パブリック	ICM データベース・サーバー	グループ・タイプはサーバー IP
パブリック	ImportLDAPUser	グループ・タイプはオブジェクト
パブリック	ImportLDAPUser_bindValues	グループ・タイプはオブジェクト
パブリック	検査エンジン・エンティティー	例: adminconsole_sniffer、software_tap_db_client、software_tap_db_server
パブリック	Java™ コマンド	例: alter java、create java、drop java
パブリック	KILL コマンド	例: kill
パブリック	Masked_SP_Executions_MS_SQL_SERVER	MS SQL Server では、ストアード・プロシージャー (SP) 名のコレクションを含むグループ。 含まれるプロシージャーが実行される場合、それが引用符で囲まれていても、すべてにマスクが掛けられます。 これは、空として事前定義されています。
パブリック	Masked_SP_Executions_Sybase	Sybase では、ストアード・プロシージャー (SP) 名のコレクションを含むグループ。 含まれるプロシージャーが実行される場合、それが引用符で囲まれていても、すべてにマスクが掛けられます。 これは、空として事前定義されています。
パブリック	MongoDB スキップ・コマンド	グループ・タイプはコマンド
パブリック	MS-SQL レプリケーション・プロシージャー	グループ・タイプはオブジェクト
パブリック	MS-SQL セキュリティー・システム・プロシージャー	グループ・タイプはオブジェクト
パブリック	MS-SQL システム・プロシージャー	グループ・タイプはオブジェクト
パブリック	Oracle EBS HRMS 機密オブジェクト	グループ・タイプはオブジェクト
パブリック	Oracle EBS-PCI	グループ・タイプはオブジェクト
パブリック	Oracle EBS-SOX	グループ・タイプはオブジェクト
パブリック	Oracle 定義済みユーザー	グループ・タイプはユーザー
パブリック	ピア関連コマンド	データのリンク/レプリケーション、例、リンク、配送記録、レプリケーション、スナップショットを処理するコマンド
パブリック	ピア関連プロシージャー	ピア関連オブジェクト、 データのリンク/レプリケーションを処理するプロシージャー 例: リンク、配送記録、レプリケーション、スナップショット
パブリック	PeopleSoft オブジェクト	グループ・タイプはオブジェクト
パブリック	PeopleSoft 機密オブジェクト	グループ・タイプはオブジェクト
パブリック	パフォーマンス・コマンド	例: analyze、create statistics、update all statistics
パブリック	Policy 関連エンティティー	例: access_rule、gdm_install_policy_header

SQL_APP_NAME	GROUP_DESCRIPTION	MEMBERS
パブリック	潜在的なオーバーフロー・オブジェクト	グループ・タイプはオブジェクト
パブリック	プロシージャー・コマンド	例: begin、call、execute、exit、repeat、set
パブリック	PROCEDURE DDL	例: alter procedure、create procedure、drop procedure
パブリック	PSFT アプリケーション・サーバー	グループ・タイプはクライアント IP
パブリック	PSFT データベース・サーバー	グループ・タイプはサーバー IP
パブリック	パブリック実行可能プロシージャー	実行専用オブジェクト。 デフォルトでパブリックへのアクセス権を付与されているプロシージャー/関数/パッケージ
パブリック	パブリック選択可能オブジェクト	選択専用オブジェクト。デフォルトでパブリックへのアクセス権を付与されている表
パブリック	RESTORE コマンド	例: restore database、restore log
パブリック	REVOKE コマンド	例: revoke object privileges、revoke system privileges
パブリック	リスク表示エラー・メッセージ	セキュリティーに関連した SQL エラー
パブリック	Sharepoint サーバー	
パブリック	SAP-PCI	グループ・タイプはオブジェクト
パブリック	SAP アプリケーション・サーバー	グループ・タイプはクライアント IP
パブリック	SAP データベース・サーバー	グループ・タイプはサーバー IP
パブリック	SAP HR 機密オブジェクト	グループ・タイプはオブジェクト
パブリック	SELECT コマンド	例: select、select list
パブリック	機密オブジェクト	例: activity、sales
パブリック	SIEBEL アプリケーション・サーバー	グループ・タイプはクライアント IP
パブリック	SIEBEL データベース・サーバー	グループ・タイプはサーバー IP
パブリック	Siebel SIA 機密オブジェクト	グループ・タイプはオブジェクト
パブリック	SPECIAL CASE ソース・プログラム	グループ・タイプはソース・プログラム
パブリック	疑わしいオブジェクト	グループ・タイプはオブジェクト
パブリック	疑わしいユーザー	グループ・タイプはユーザー
パブリック	システム構成コマンド	データベース構成コマンド (管理コマンドのサブセット) 例: ALTER DATABASE、ALTER SYSTEM
パブリック	システム構成プロシージャー	システム構成オブジェクト (「管理オブジェクト」のサブセット)
パブリック	無効なデータベース・ユーザー	グループ・タイプはユーザー
パブリック	脆弱なオブジェクト (ワイルドカード使用)	脆弱性が報告されているデータベース・オブジェクト
パブリック	Db2 デフォルト・ユーザー IBM iSeries デフォルト・ユーザー Informix デフォルト・ユーザー MS-SQL Server デフォルト・ユーザー MYSQL デフォルト・ユーザー Netezza デフォルト・ユーザー Oracle デフォルト・ユーザー PostgreSQL デフォルト・ユーザー Sybase デフォルト・ユーザー Teradata デフォルト・ユーザー	グループ・タイプはデータベース・ユーザー/データベース・パスワード
パブリック	Hadoop スキップ・コマンド Hadoop スキップ・オブジェクト Hadoop 以外のサーバー	グループ・タイプはコマンド グループ・タイプはオブジェクト グループ・タイプはサーバー IP
パブリック	リプレイ - 比較対象から除外する リプレイ - 比較対象に含める	グループ・タイプはオブジェクト
監査プロセス・ビルダー		これは、空として事前定義されています。
Classifier		これは、空として事前定義されています。
エクスプレス・セキュリティー		これは、空として事前定義されています。

親トピック: グループ

セキュリティー・ロール

セキュリティー・ロールは、データ (グループ、照会、レポートなど) へのアクセスを許可したり、アプリケーション (「グループ・ビルダー」、「クエリー - レポート・ビルダー」、「ポリシー・ビルダー」、「CAS」、「セキュリティー・アセスメント」など) へのアクセスを許可するために使用します。

デフォルトでは、 コンポーネントが最初に定義されるときに、owner (定義を行った人) と admin ユーザー (特別な特権を持つ) がそのコンポーネントへのアクセスおよび変更を 許可されます。

セキュリティー・ロールを割り当てることによって、定義したコンポーネントに他のユーザーが アクセスできるようにすることができます。例えば、DBA という名前のセキュリティー・ロールを 監査プロセスに割り当てると、DBA ロールに割り当てられたすべてのユーザーが、その監査プロセスにアクセスできます。

注: LDAP ユーザー・インポートを構成する accessmgr ユーザーには、グループ・ビルダーの実行特権が必要です。 特定の状態において、ロール特権に変更が加えられた場合、accessmgr のグループ・ビルダーに対する特権が取り消される可能性があります。 その結果、LDAP ユーザー・インポートを正常に保存することも実行することもできなくなります。 アクセス管理ポータルに移動して、「ロール権限」を選択してください。グループ・ビルダー・アプリケーションを選択し、「すべてのロール」ボックスまたは「accessmgr」ボックスにチェック・マークが付けられていることを確認します。

セキュリティー・ロールの割り当て

1. 1 つ以上のセキュリティー・ロール (ポリシー定義またはレポート定義など) を割り当てる項目を開くか、選択します。
2. 「ロール」をクリックします。
3. 「セキュリティー・ロールの割り当て」リストから、割り当てるすべてのロールにチェック・マークを付けます。自分のアカウントに割り当てられているロールのみを割り当てることができます。
4. 「適用」をクリックします。

新規セキュリティー・ロールの定義

デフォルトでは、特別な accessmgr ユーザー のみが、セキュリティー・ロールの作成または削除を許可されています。

1. accessmgr としてログインし、「アクセス」 > 「アクセス管理」 > 「ユーザー・ロール・ブラウザー」をクリックして、「ユーザー・ロール・ブラウザー」を開きます。
2. ロール・ブラウザーの最後にある「ロールの追加」をクリックします。
3. 「ロール・フォーム」パネルで、新しい「ロール名」を入力して、「ロールの追加」をクリックします。

セキュリティー・ロールの削除

デフォルトでは、特別な accessmgr ユーザー のみが、セキュリティー・ロールの作成または削除を許可されています。コンポーネントに割り当てられたロールを削除するには、コンポーネントへのセキュリティー・ロールの割り当てを参照してください。

1. accessmgr としてログインし、「アクセス」 > 「アクセス管理」 > 「ユーザー・ロール・ブラウザー」をクリックして、「ユーザー・ロール・ブラウザー」を開きます。
2. ロールの「削除」をクリックしてから、「削除の確認」をクリックします。

親トピック: Guardium システムの管理

通知

通知を作成するには、「アラート機能」および「アラート・ビルダー」を使用します。アラート・アクションに E メールまたはその他の通知が必要な場合は、以下の手順に従って、各タイプの通知について定義してください。

アラート機能の構成

1. アラート・アクションを選択する前に、「アラート機能」で E メールの SMTP 設定を構成する必要があります。
2. 「保護」 > 「データベースの侵入検出」 > 「アラート機能」をクリックして、「アラート機能」を開きます。
3. SMTP または SNMP (あるいは両方) の情報を入力します。
4. 各セクションに入力した後、「接続のテスト」をクリックして、接続が機能していることを確認します。接続が機能していない場合は、接続が到達不能であるというメッセージを受け取ります。
5. 「適用」をクリックして、構成を保存します。
6. 最低でも IP アドレス/ホスト名、ポート、および送信先 E メールアドレスを指定しなければなりません。
7. 「通知タイプ」メニューから「*メール*」を選択します。メッセージの重大度が「*高*」の場合、緊急フラグが設定されます。
8. 「アラート受信者」リストからユーザー (個人またはグループ) を選択します。リアルタイム E メール通知の追加の受信者は、「起動者」(ポリシー起動の原因となった実際の SQL コマンドを開始したユーザー) と「所有者」(データベースの所有者) です。起動者と所有者は、Guardium® API を使用して構成されたユーザー ID (IP ベース) を取得することによって識別されます。
9. 「追加」をクリックします。

アラートの作成

1. 「アラート機能」を構成した後、「保護」 > 「データベースの侵入検出」 > 「アラート・ビルダー」をクリックして、「アラート・ビルダー」を開きます。
2. 「設定」、「アラート定義」、「アラートしきい値」、および「通知」の各セクションに情報を入力して、「適用」をクリックします。
3. 「受信者の追加..」をクリックしてユーザーを選択することで、通知を受け取るユーザーを選択します。

親トピック: Guardium システムの管理

リアルタイム・アラートの作成方法

同じユーザーによるログインの失敗が 5 分以内で 3 回を超えた場合に、 データベース管理者にリアルタイム・アラートを送ります。

このタスクについて

疑わしいアクティビティーが検出された場合や、アクセス・ポリシーに違反があった場合に、リアルタイムのセキュリティー・アラートを生成します。

以下の手順を行います。

1. ポリシーの作成
2. ポリシーへのルールの追加
3. ポリシーのインストール
4. ポリシー起動時のリアルタイム・アラートのセットアップ

前提条件

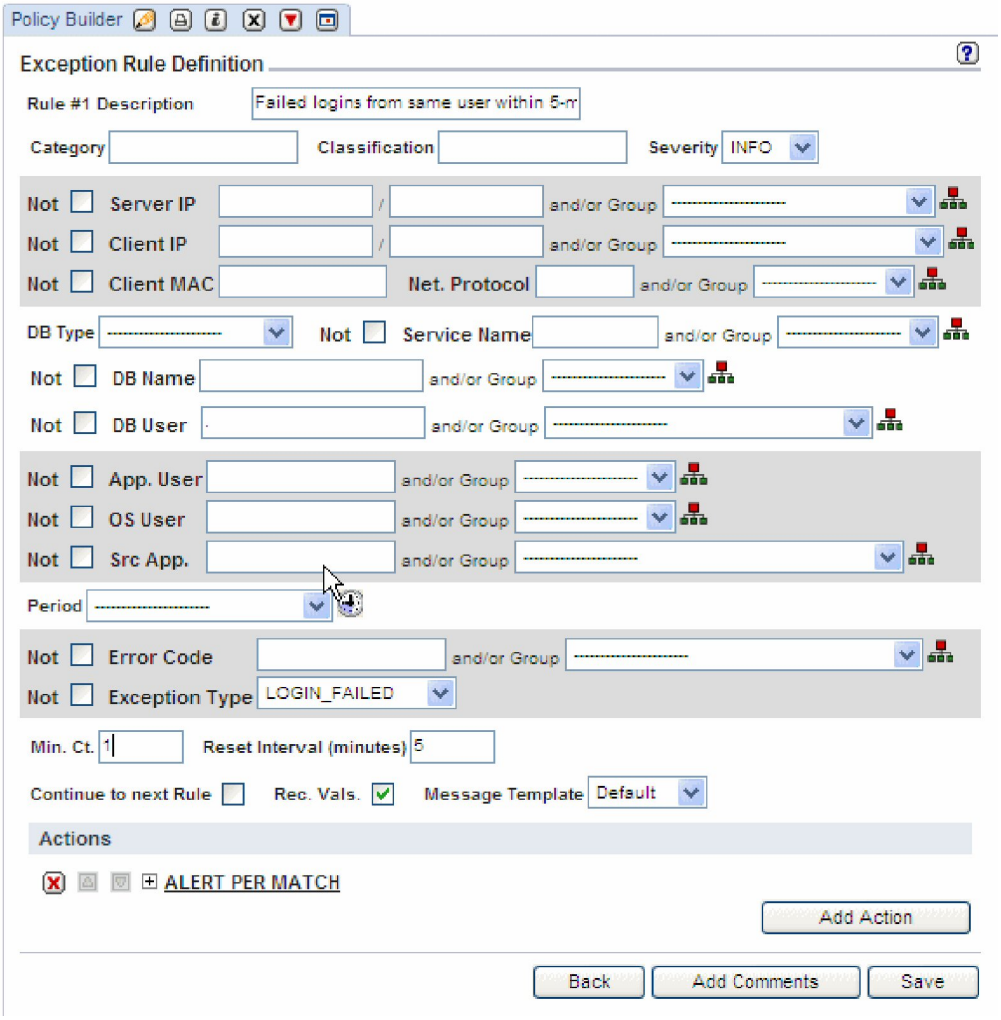
「アラート機能」での SMTP の構成。 「保護」 > 「データベースの侵入検出」 > 「アラート機能」をクリックして、「アラート機能」を開いて、SMTP 情報を入力します。

注: ポリシー違反は、インシデント管理でレポートとして表示することも可能です。詳しくは、『ポリシー』を参照してください。

手順

1. ポリシーを作成します。
   a. 「設定」 > 「ツールとビュー」 > 「データまたはアプリケーションのポリシー・ビルダー」をクリックして「ポリシー・ビルダー」を開きます。
   b. 「新規」をクリックするか、「ポリシー・ファインダー」でポリシーを選択して「変更」をクリックすることで、既存のポリシーを変更します。
   c. 必須情報を入力して、「適用」をクリックし、ポリシーを保存します。
2. ポリシーにルールを追加します。
   a. ポリシーを保存したら、「ルールの編集」をクリックして既存のポリシー・ルールを表示します。
   b. 「ルールの追加...」をクリックすると、5 つのルール・オプションが表示されます。
   c. 「例外ルールの追加」を選択して必須情報を入力します。

   「例外ルール定義」画面には、最初に以下の項目が表示されます:



- 記述 - ルールの簡潔な記述名を入力します。
- カテゴリー - カテゴリーは違反とともにログに記録され、グループ化およびレポート目的で使用されます。何も入力しないと、ポリシーのデフォルトが使用されます。
- 分類 - (オプション) 「分類」ボックスに分類を入力します。カテゴリー同様、これらは例外とともにログに記録され、グループ化およびレポート目的で使用されます。
- 重大度 - メニューから重大度コード (「情報」、「低」、「なし」、「中」、または「高」) を選択します (デフォルトは「情報」です)。

d. 他のフィールドを使用してルールの突き合わせ方法 (検索する場所、検索対象、検索対象ユーザー、検索するタイミング) を指定します。
e. 個々の値を別々にカウントするには、「データベース・ユーザー」フィールドにピリオド「.」を入力します。
f. 「例外タイプ (Excpt. Type)」 (例外タイプ) メニューから、「LOGIN_FAILED」を選択します。
g. 「最小数」を使用して、ルールが何回一致したらアクションを起動するかの最小回数を設定します。この例では 1 を選択します。 ルールが成立した回数は、アクションが起動されるごとに、またはリセット間隔が満了になるとリセットされます。
h. 「リセット間隔」を使用して、ルール・カウンターをゼロにリセットするまでの時間を分数で設定します。カウンターはまた、ルール・アクションが起動するごとにゼロにリセットされます。この例では、「5」を選択してください。
i. 「次のルールに進む」チェック・ボックスにチェック・マークを付け、このルールが成立してルールのアクションが起動された後にルールのテストを続行するようにします。これが選択されていない場合は、このルールが成立した際に、追加のルールはテストされません。
j. 「値を記録」チェック・ボックスにチェック・マークを付けると、ルールのアクションが起動されたときに、そのイベントの原因となる SQL ステートメント全体がログに記録され、ポリシー違反レポートで参照できるようになることを意味します。マークを付けない場合、SQL 文字列属性は空になります。

3. ルールが起動したときのアクションを追加します。
   a. 「例外ルール定義」画面のアクション・セクションから、「アクションの追加」をクリックします。
   b. 「アクション」メニューからオプションを選択して、「適用」をクリックします。この例では、*ALERT PER MATCH* が選択され、ルールが起動するたびに通知を受け取るようにしています。
   c. 「通知タイプ」メニューからオプションを選択します。 メール通知タイプまたは SNMP 通知タイプの「アラート機能」を構成する必要があります。
   d. アラート受信者を追加し、「適用」をクリックしてアクションを保存します。
4. ポリシーをインストールします。
   a. 「設定」 > 「ツールとビュー」 > 「ポリシー・インストール」をクリックします。
   b. 「ポリシー・インストーラー」メニューからポリシーを探し、インストール・アクションを選択して、「スケジュールの変更」または「今すぐ 1 回実行」をクリックします。 これで、ポリシーがインストールされました。アラート受信者は、ポリシー・ルールの起動時にリアルタイム通知を受信します。

親トピック: Guardium システムの管理

カスタム・アラート・クラスの管理

カスタムの受信者にアラートを送信するには、カスタム・アラート・クラスを使用します。 カスタム・クラスをアップロードしてから、「アラート・ビルダー」を使用して、アラート通知受信者としてカスタム・クラスを指定します。

- カスタム・クラスを使用するには、事前に Guardium システムにアップロードしておく必要があります。「セットアップ」 > 「カスタム・クラス」 > 「アラート」 > 「アラート・クラスのアップロード」をクリックして、カスタム・アラート・クラスをアップロードします。「参照」をクリックしてファイルを選択し、「適用」をクリックして保存します。
- カスタム・クラスをアップロードした後、「アラート・ビルダー」を使用し、そのカスタム・クラスをアラートで使用します。「管理」 > 「データベースの侵入検出」 > 「アラート・ビルダー」をクリックして、「アラート・ビルダー」を開きます。必須情報を入力して、「通知タイプ」メニューから「*CUSTM*」を選択し、「保存」をクリックします。

親トピック: Guardium システムの管理

事前定義アラート

表で、「アラート・ビルダー」にある事前定義アラートについて説明します。

Guardium は、「アラート・ビルダー」で見つかる一連の事前定義アラートを備えています。「保護」 > 「データベースの侵入検出」 > 「アラート・ビルダー」をクリックして、「アラート・ビルダー」を開きます。「アラート・ビルダー」を開くと、「アラート・ファインダー」にすべての既存のアラートのリストが表示されます。ファインダーからアラートを選択して、「変更」をクリックし、編集します。

「アラートの変更」画面で、受信者やしきい値など、アラートの任意の部分を変更します。

アラートの基準となっているデフォルトの照会は変更できません。照会を変更する場合、照会の「この照会を編集」アイコン をクリックして、「クエリー・ビルダー」を開きます。ビルダーで照会のコピーを作成した後、ニーズに合わせてこれを変更します。

アラートに変更を加えた後、「適用」をクリックして保存します。

次の表に、すべての事前定義アラートを示します。

表 1. 事前定義アラート

アラート	記述
変更されたアクティブ S-TAP	最後の集計間隔中にアクティブ S-TAP® 検査エンジンに加えられた変更をチェックします。 この期間中に変更された検査エンジンが少なくとも 1 つあれば、このアラートが起動します。 デフォルトでは、このアラートは 30 分ごとに最後の 1 時間をチェックします。
統合エラーまたはアーカイブ・エラー	正常に完了しなかったすべての統合タスクまたはアーカイブ・タスクについて、1 日に一度アラートを出します。
接続プロファイル・アラート	アラートは 60 分間隔で実行され、通知が事前定義グループ (許可された接続の接続プロファイル・リスト - 名前リスト) に送信されます。
CAS インスタンス構成変更	CAS インスタンス構成変更で 1 日に一度アラートを出します。
CAS テンプレート変更	CAS テンプレート構成変更で 1 日に一度アラートを出します。
データ・ソース変更	データ・ソース定義の変更で 1 日に一度アラートを出します。

アラート	記述
データベースのディスク・スペース	内部データベースの満杯率が 80% を超えた場合に、10 分ごとにアラートを出します。ディスク・スペース (満杯率) と Guardium® Nanny プロセスの詳細については、『自己モニター』ヘルプ・トピックを参照してください。
エンタープライズの「トラフィックなし」	エンタープライズの「トラフィックなし」アラートは、 中央マネージャー・システムでのみ実行されます。これは、「トラフィックなし」アラートでの照会と同様の照会を基にして、タイム・スタンプが X と Y の間にあるレコードを検索します。この場合、X は照会パラメーター、Y は集計間隔を基にしたアラート・メカニズムが生成する日付からの照会です (既存の「トラフィックなし」アラートの場合と同じ方法)。
変更されたエンタープライズ S-TAP	このアラートは、中央マネージャー・システムでのみ実行されます。
Guardium への失敗したログイン	Guardium アプライアンスへのログイン試行の失敗が 5 回を超えた場合、10 分ごとにアラートを出します。
Guardium - ユーザーの追加/削除	Guardium ユーザーが追加または削除された場合、 1 日に一度アラートを出します。
Guardium - 資格情報アクティビティー	Guardium 資格情報が変更された場合 (LDAP 構成の変更など)、1 日に一度アラートを出します。
非アクティブ管理対象ユニット	アラートは、30 分間隔で実行され、通知は 1 日に 1 回、「管理対象ユニット・アラート」という事前定義グループに送信されます。
非アクティブな S-TAP	非アクティブなすべての S-TAP について、 1 時間に一度アラートを出します。
検査エンジンと S-TAP	検査エンジンと S-TAP 構成に関連したアクティビティーについて、1 日に一度アラートを出します。
トラフィックなし	特定のデータベース・サーバーからのトラフィックがないかどうかを示すアラート。 このアラートは、Guardium システムのトラフィック収集元であるサーバーから、過去 48 時間のいずれかの時点で収集されるトラフィックがない場合にアラートを出します。 このアラートは、集計間隔に定義されている期間内にトラフィックがない場合に起動します。 例えば、集計間隔が 60 分だとすると、特定のデータベース・サーバーから過去 1 時間内にはトラフィックがなかったが、48 時間以内には何らかのトラフィックがあったという場合に、このアラートは E メールを送信します。 このアラートは、(デフォルトでは) 24 時間ごとにしか E メールを送信しません。集計間隔、通知インターバル、実行頻度などのパラメーターは、カスタマイズできます。 しきい値、「行当たり」、演算子、照会などのパラメーターは変更しないでください。これらのパラメーターを変更すると、アラートが正常に機能しなくなります。「トラフィックなし」照会のコピーを作成しないようにご注意ください。
サーバー/プロトコルによるトラフィックなし	通常の「トラフィックなし」アラートと似ていますが、以下の点が異なります。 このアラートはサービス名/ネット・プロトコルごとに出され、行ごとに報告されます。 新しい追加パラメーター「アクティブ・トラフィック・インターバル」により、各サーバーからの最後の要求をいつ受信したかがわかります。 このアラートは、以下の条件で起動します。各サーバー/ネット・プロトコルからのアラート間隔中にはトラフィックがなかったが、その組み合わせのアクティブ・トラフィック・インターバル以降にトラフィックがあった場合。 アラート間隔中にはトラフィックがなかったが、その直前の 48 時間以内にサーバー IP 当たりのトラフィックがあった場合に起動する通常の「トラフィックなし」アラートとは違います。
ポリシー変更アラート	セキュリティー・ポリシーの変更があった場合に、 1 日に一度アラートを出します。
長時間実行の照会	照会の実行が 900 秒を超える場合に通知します。
スケジュールされたジョブの例外	スケジュールされたジョブの例外 (アセスメント・ジョブを含む) で、10 分ごとにアラートを出します。

親トピック: Guardium システムの管理

スケジューリング

汎用スケジューラーは、さまざまなタイプのタスク (アーカイブ、統合、 ワークフロー・オートメーションなど) をスケジュールに入れるために使用します。

実行中の タスクのタイプにより、ここで説明するすべての機能が使用可能であるとは限りません。 例えば、タスク・タイプのスケジュールは一時停止できるものと、 できないものがあります (停止または開始のみ可能)。

注: 夏時間の期間中にタスクをスケジュールに入れると、スケジュール異常が発生する可能性がある ことに注意してください。

スケジュールの定義または変更

1. タスク (「監査プロセス・ビルダー」など) で、「スケジュールの定義」または「スケジュールの変更」をクリックして、「スケジュール定義」パネルを開きます。
2. 「開始時刻 」に入力します。デフォルトは 12 a.m. (午前零時) です。
3. オプションで、タスクを 1 日に複数回実行するには、以下のようにします。
   - 「再始動」リストから値を 選択します (毎時から最大 12 時間ごと)。デフォルトは「1 回だけ実行 」で、タスクが同じ日に再始動されないことを意味します。
   - 「繰り返し」リストから値を 選択します (毎分から最大 59 分ごと)。デフォルトは「繰り返しなし」です。
4. 「スケジュールの基準 」リストから、以下のいずれかを選択します。
   - 「曜日」 1 つ以上の曜日 (月曜日、火曜日、水曜日など) に基づくスケジュールを 定義します。
   - 「月」 毎月または特定の月で、その月の 1 日以上の日に基づいてスケジュールを 定義します。

   「スケジュールの基準」リストから 「曜日」を選択した場合、タスクを実行する各曜日に マークを付けるか、「毎日」をクリックしてすべての日を選択します (既にすべての日が選択されている場合は、 すべてクリアされます)。

   または

   「スケジュールの基準」リストから 「月」を選択した場合、以下のいずれかを実行します。

   - 日付 (例えば 15 日) を選択するには、次のようにします。
     - 「日」ボタンを選択します。
     - 選択した月に応じて、日付を 1 から 31 から 選択します。
     - 「毎月 」、または 1 つ以上の特定の月を選択します。

- その月内の曜日オカレンス (例えば、第 1 月曜日) を選択するには、 次のようにします。
  - ボタンを選択します。
  - 月初めからの相対的な週 (第 1、第 2、第 3 など) を 選択します。
  - 曜日 (日曜日、 月曜日、火曜日など) を 選択します。
  - 「毎月」または 1 つ以上の特定の月を選択します。

5. 「開始時刻のスケジュール設定 」リストから、タスクを実行する時分を選択します。 NOW より前の時刻が選択されている場合、スケジューラーの開始時刻は NOW に戻ります。
6. 「適用」をクリックします。

スケジュールの一時停止

注: スケジュールに入れられたすべてのタイプのタスクが一時停止オプションを提供するわけではない、ということに注意してください。

1. 「一時停止」をクリックします。
2. アクションを確認します。

スケジュールの削除

スケジュールの定義が完了した後、「スケジュール定義」パネルに 「削除」ボタンが表示されます。

1. 「スケジュールの定義」または「スケジュールの変更」をクリックして、「スケジュール定義」パネルを開きます。
2. 「削除」ボタンをクリックします。

親トピック: Guardium システムの管理

別名

レポートまたは照会で使用されるデータ値またはデータ・オブジェクトのシノニムを作成します。

別名の概要

別名は、データ値を意味のある、または分かりやすい名前で表示するために使用されます。

例えば、IP アドレス 192.168.2.18 の別名として、「*財務サーバー*」を定義することができます。別名を定義すると、ユーザーはデータ値の代わりに別名を使用してレポート結果を表示し、 照会を形成し、パラメーター値を入力することができるようになります。

別名を定義する方法は、何とおりかあります。

- 「IP からホスト名への別名割り当て」ツール - 検出されたクライアント IP とサーバー IP に対して別名を生成するには、このツールを使用します。

  「保護」 > 「データベースの侵入検出」 > 「IP からホスト名への別名割り当て」をクリックして、「IP からホスト名への別名割り当て」ツールを開きます。

- 「別名ビルダー」 – 別名を手動で定義する場合、この方法を使用します。

  「順守」 > 「ツールとビュー」 > 「別名ビルダー」をクリックして、「別名ビルダー」を開きます。

- 照会
- 「別名クイック定義」 (「グループ・ビルダー」使用時)
-

注: 中央マネージャーまたは管理対象ユニットでの別名変更を他のシステム・ユニットで有効にするには、 GUI を再起動するか、そのシステム・ユニットの GUI を使用して別名変更を行う必要があります。

IP からホスト名への別名割り当て

別名の一般的な応用方法の 1 つは、IP アドレスのシノニムとして使用することです。 このツールを使用して、クライアント IP とサーバー IP のディスカバリーのスケジュールを設定し、それらの別名を生成します。

1. 「保護」 > 「データベースの侵入検出」 > 「IP からホスト名への別名割り当て」をクリックして、「IP からホスト名への別名割り当て」ツールを開きます。
2. 「クライアント IP とサーバー IP のホスト名別名の生成 (使用可能な場合)」チェック・ボックスにチェック・マークを付けます。
3. ツールがホスト名別名を継続的に探して更新するようにする場合は、「既存のホスト名別名の更新 (再発見された場合)」チェック・ボックスにチェック・マークを付けます。
4.
5. 「適用」をクリックして構成を保存した後、操作のスケジュールを設定します。
   - 「今すぐ 1 回実行」をクリックすると、ツールが直ちに開始されます。
   - 今後のツールのスケジュールを設定するには、「スケジュールの定義...」をクリックします。
   - クライアント IP とサーバー IP の別名の生成を一時停止するには、「一時停止」をクリックします。

別名ビルダー

別名を手動で作成するには、この方法を使用します。

1. 「設定」 > 「ツールとビュー」 > 「別名ビルダー」をクリックして、「別名ビルダー」を開きます。
2. 別名を定義する属性タイプを選択します。
3. 「値」フィールドと「別名」フィールドを使用して、その属性タイプで検索をフィルタリングし、「検索」をクリックします。
4. いずれかの結果が検索と一致すると、値と別名の表に表示されます。検索結果の「適用」をクリックするか、「値」および「別名」の名前を指定してから「追加」をクリックして新規別名を追加します。

5. 「項目のコメント」アイコン  をクリックして、別名にコメントを追加します。これは、将来、別名の参照先を素早く見つける上で役立ちます。

照会を使用した別名定義

このメソッドは、照会から別名を作成する場合に使用します。カスタム表が Guardium® にアップロードされると、その表を使用して別名を特定の値にマップすることができます。

1. 「設定」 > 「ツールとビュー」 > 「別名ビルダー」をクリックして、「別名ビルダー」を開きます。
2. 別名を定義する属性タイプを「別名ファインダー」から選択して、「照会から取り込み」をクリックして、「照会からの別名ビルダーの設定」パネルを開きます。
3. 必須情報を入力して、「保存」をクリックし、別名を保存します。
   - 「照会」メニューから、実行する照会を選択します。
   - 「値の列の選択」と「別名の列の選択」の両方の値を選択します。
   - 列の値を選択した後、その他のフィールド (「開始日付」、「終了日付」、「リモート・ソース」、および選択した照会に関する追加のパラメーター) が表示され、これらのフィールドに入力する必要があります。
   - 照会から取り込む前にグループの既存の内容を削除するには、「インポートする前に既存のグループ・メンバーをクリアする」チェック・ボックスにチェック・マークを付けます。
   - 「保存」をクリックして、保存します。
   - 照会が保存されると、「スケジューリング」ボタンがアクティブになります。「スケジュールの変更」をクリックして照会を将来に実行することも、「今すぐ 1 回実行」をクリックして照会をすぐに実行することもできます。

グループ・ビルダーからの別名クイック定義

グループの作成時または取り込み時にすぐにグループの別名を作成するには、この方法を使用します。

1. 「設定」 > 「グループ・ビルダー」をクリックして、「グループ・ビルダー」を開きます。リストから任意のグループを選択して、「変更」をクリックします。
2. 「別名...」をクリックして、「別名クイック定義」ウィンドウを開きます。グループ (複数可) の別名を入力して、「適用」をクリックし、別名を保存します。

別名用の GuardAPI

これらの GuardAPI コマンドは、別名機能の作成、更新および削除に使用します。

- grdapi create_alias
- grdapi update_alias
- grdapi delete_alias

親トピック: Guardium システムの管理
関連情報:
高度な Guardium システム管理および構成 (ビデオ)

日付とタイム・スタンプ

カレンダー・ツールを使用して絶対日付を選択し、相対日付ピッカーを使用して現在時刻からの相対的な日付を選択します。

日付フィールドにデータを追加するために使用する 2 つのツールがあります。1 つは絶対日付を選択するためのカレンダー・ツール、もう 1 つは現在時刻からの相対的な日付 (例えば、now -1 day) を選択する相対日付ピッカーです。さらに、絶対日付または相対日付を手動で入力することもできます。

日付を選択または入力する際には、ブラウザーを実行しているシステム上の日付が、接続先の Guardium® アプライアンス上の日付とは異なる場合があることに注意してください。

照会内のタイム・スタンプ

タイム・スタンプを照会に含める場合は、注意が必要です。

まず、timestamp (小文字の「t」) および Timestamp (大文字の「T」) の区別に注意してください。

- timestamp (小文字の t) は、結合された日付と時刻の値を含んでいるデータ・タイプであり、印刷時には yyyy-mm-dd hh:mm:ss のフォーマットで示されます (例: 2005-07-17 15:40:25)。照会の作成または編集時に、timestamp データ・タイプのほとんどの属性は、時計アイコン付きで「エンティティー・リスト」パネルに表示されます。
- Timestamp (大文字の T) は多くのエンティティー・タイプに定義される属性です。これには通常、そのエンティティーの最終更新時刻が含まれます。

Timestamp 属性値を照会に含めると、Timestamp 値ごとに行が作成されます。これにより、過剰な出力が行われる場合があります。この問題を回避するには、照会に Timestamp を含める際に count アグリゲーターを使用し、レポート行にドリルダウンを行い、その行のみに含まれる項目の個別の Timestamp 値をドリルダウン・レポートに表示します。「照会」の『フィールドの統合』を参照してください。

複数のエンティティーの Timestamp 属性を含む照会で Timestamp 値を表示する場合は、そのレポートに適切なエンティティー・タイプの Timestamp 属性を選択するよう注意してください。例えば、「セッション」をメイン・エンティティーに選択して、照会で「クライアント/サーバー」エンティティーおよび「セッション」エンティティーの両方の情報を表示する場合は、1 つまたは両方のエンティティーの Timestamp 属性を表示できます。「クライアント/サーバー」の Timestamp を含める場合は、特定のクライアント/サーバー接続のすべてのセッションで同じ値が出力されます。この値は常に、特定のクライアント/サーバーが最後に更新された時刻です。セッションの Timestamp 属性を含めると、リストされている各セッションが最後に更新された時刻が表示されます。

ヒント: レポートに異なる時刻が表示されると予期される場合に、すべて同じ時刻が表示されるときには、エンティティー階層内での位置が、レポートに必要な詳細レベルに対して高すぎるエンティティーの Timestamp 属性が含まれている可能性があります。

カレンダーから絶対日付を選択する

カレンダー・ウィンドウを使用して絶対日付を選択するには、次のようにします。

1. 日付を挿入するフィールドの「カレンダー」ボタンをクリックします。これにより、別のウィンドウにカレンダーが開きます。
    - 矢印ボタンをクリックすると、カレンダー・ウィンドウに前月または翌月が表示されます。
2. 日付をクリックして選択します。カレンダー・ウィンドウが閉じ、クリックしたカレンダー・ツールの隣にある「日付」フィールドに選択した日付が挿入されます。
   注: カレンダーを使用して選択した日付のデフォルトの時刻は常に 00:00:00 (その日の始め) です。他の時刻を指定するには、24 時間フォーマット (hh:mm:ss) で希望の時刻を入力し、この値を上書きします。ここで hh は時間 (0-23)、mm および ss はそれぞれ分および秒 (いずれも 0-59) です。

絶対日付を手動で入力する

1. 日付を入力するフィールドをクリックし、yyyy-mm-dd フォーマットで日付を入力します。
    - yyyy はオプションであり、任意の正整数値を使用できます。省略した場合、yyyy にはデフォルトで現在の年が使用されます。1 桁または 2 桁の年が入力された場合、日付の世紀の部分にはデフォルトで 19 が使用されます。
    - mm は、月 (1-12) です。
    - dd は、その月の日 (月に応じて 1 から 28、29、30、または 31) です。
2. 時刻が入力されない場合、時刻にはデフォルトの 00:00:00 (その日の始め) が使用されます。他の時刻を指定するには、24 時間フォーマット (hh:mm:ss) で希望の時刻を入力し、この値を上書きします。ここで hh は時間 (0-23)、mm および ss はそれぞれ分および秒 (いずれも 0-59) です。

日付ピッカーから相対日付を選択する

絶対日付を指定するよりも、現在の日付 (now) やその他の日付 (例えば、first Monday) のいずれかに対する相対日付を指定するほうが便利であることがよくあります。例えば、常に過去 7 日間の情報を照会に含める場合は、相対日付 (例えば、start = now minus seven days および end = now) を定義する方が便利です。「相対日付ピッカー」ツールを使用して、多くのタイプのタスクで相対日付を選択できます。

1. 相対日付が使用可能なフィールドの横にある「相対日付ピッカー」ボタンをクリックします。これにより、「相対日付ピッカー」ウィンドウが開きます。
2. リストから「Now」、「Start」、または「End」を選択します。選択内容にかかわらず、表示が変更され、さらに選択項目が表示されます。
3. 中央のリストから、「this」、「last」、または「previous」を選択します。これは、特定の単位 (次のリストで選択される「日」、「週」、「月」、または「曜日」) に対する相対的なものです。
    - 「This」は、現在の単位です。
    - 「Last」は、現在の単位から 1 を引いたものです。
    - 「Previous」は、現在の単位から 2 を引いたものです。
4. 「日」、「週」、「月」、または特定の曜日 (月曜日から金曜日) を選択します。
5. 完了したら、「OK」ボタンをクリックします。クリックした「相対日付ピッカー」ボタンの隣にあるフィールドに、相対日付が挿入されます。
6.

相対日付を手動で入力する

相対日付を手動で入力するには、いずれかのステップに従います。キーワードには大/小文字の区別はありませんが、各コンポーネントを 1 つ以上のスペースで区切る必要があります。

相対日付を入力する際に使用できる一般的なフォーマットには、以下の 3 つがあります。

NOW に続けて指定した負の数と、minute、hour、day、week、month

または

Start of または End of に続けて、this、last、または previous と day、week、または month

または

Last または Previous に続けて曜日 (Sunday、Monday、Tuesday など)

NOW に対する相対日付

1. 相対日付を入力するフィールド内をクリックします。
2. キーワード「NOW」を入力します。
3. 相対的な時間数、日数、週数、または月数を指定する負の整数を入力します (負符号 (-) と整数の間にスペースを入れることはできません)。
4. 使用する単位のキーワード (HOUR、DAY、WEEK、または MONTH) を入力します。複数形 (hours、days など) は使用できないことに注意してください。例: now -14 day

日、週、または月に対する相対日付

1. 相対日付を入力するフィールド内をクリックします。
2. キーワード「START OF」または「END OF」を入力します。
3. 「THIS」、「LAST」または「PREVIOUS」に続けて、「DAY」、「WEEK」、または「MONTH」を入力します。例: end of last week

曜日に対する相対日付

1. 相対日付を入力するフィールド内をクリックします。
2. キーワード「START OF」または「END OF」を入力します。
3. 「LAST」または「PREVIOUS」に続けて、「SUNDAY」、「MONDAY」、「TUESDAY」、「WEDNESDAY」、「THURSDAY」、「FRIDAY」、または「SATURDAY」を入力します。例: start of previous Tuesday

親トピック: Guardium システムの管理

ビルド期間

ポリシー・ルールおよび照会条件によって、ユーザー定義の期間内にイベントが発生したかどうかをテストできます。

「期間ビルダー」を開くと、事前定義された一連の期間が使用可能な状態になっています。要件を満たすようにこれらの使用可能期間を編集したり、独自の期間を定義したりすることができます。

期間の追加

1. 「設定」 > 「ツールとビュー」 > 「期間ビルダー」をクリックして、「期間ビルダー」にナビゲートします。
2.  をクリックして、「期間の追加」ペインを開きます。
3. 連続期間か繰り返しブロック期間のいずれかを作成できます。例えば、以下の 2 つのカスタム期間は、両方とも月曜日の 09:00 (午前 9 時) に始まり、金曜日の 17:00 (午後 5 時) に終了します。
   - 「出勤週 (Workweek)」は連続しており、月曜日の午前 9 時に始まり金曜日の午後 5 時に終わる単一の 164 時間の期間を定義します。
   - 「出勤日」は繰り返しブロック (つまり、不連続) であり、連続した 5 つの日 (月曜日から金曜日まで) に対して 8 時間の期間 (午前 9 時から午後 5 時まで) を個別に 5 つ定義します。
4. 「名前」ボックスに、当該期間の名前を入力します。 この例では、「出勤週 (Workweek)」または「出勤日」のいずれかを入力します。
   注: 記述にはアポストロフィ (') を使用しないでください。
   - 「出勤週 (Workweek)」では、「連続」を選択します。
   - 「出勤日」では、「繰り返しブロック」を選択します。
5. 「開始時刻」ボックスに、開始時刻の時 (00 から 24) および分 (00 から 59) を入力します。この例では、「09:00」と入力します。
6. 「終了時刻」ボックスに、終了時刻の時 (00 から 24) および分 (00 から 59) を入力します。この例では、「17:00」と入力します。
7. 「開始曜日」ボックスで、開始する曜日を選択します。
8. 「終了曜日」ボックスで、終了する曜日を選択します。
9. 「OK」をクリックして新規期間を保存します。

期間の編集

1. 「期間ビルダー」から、既存の期間を選択し、  をクリックして、「期間の編集 (Edit time period)」ペインを開きます。
2. 選択した期間の「期間の編集 (Edit time period)」ペインで、必要に応じて各フィールドを変更します。
3. 「OK」をクリックして変更を保存します。

期間の削除

1. 「設定」 > 「ツールとビュー」 > 「期間ビルダー」をクリックして、「期間ビルダー」にナビゲートします。
2. 削除する期間を選択し、  をクリックして、選択した期間を削除します。
   注: 既存のポリシー・ルールで使用されている期間は削除できません。

親トピック: Guardium システムの管理

コメント

コメントは、定義とワークフロー・プロセス結果に適用されます。

コメントは、UI 全体のいくつかの場所で追加または表示できます。コメントは、参照の目的でグループまたは別名に追加したり、監査要件を軽減するためにレポートに追加したりすることができます。例えば、特定の日付に構成変更が行われた理由を、監査員が確認する場合です。変更を加えた理由を簡単に参照できるように、コメントを使用します。

コメントは定義 (グループ、別名、レポート、ポリシー) およびワークフロー・プロセス結果に適用されます。1 つのコンポーネントに複数のコメントを追加でき、コメントにもコメントを追加できます。ただし、既存のコメントの変更や削除はできません。

以下のとおり、2 つの異なる種類のコメントがあります。

- **「コメント」エンティティー** - 中央マネージャー上に保管されます。その一元管理環境内で使用可能であり、ロールとアクセス権について通常の制約が課されます。
- **「ローカル・コメント」エンティティー** - 単一のユニットに対して定義され、そのユニットのローカルでの使用にとどまります。スタンドアロン・ユニットまたは管理対象ユニットのローカル・コメントは、中央マネージャーには保管されません。

コメントの追加または表示

1. コメントを表示するには、「順守」 > 「レポート」 > 「ユーザー・コメント」をクリックして、「ユーザー・コメント」ウィンドウを開きます。
2. UI 全体で、コメントをエンティティーまたはレポートに追加するための各種の方法があります。
   - グループにコメントを追加するには、グループを変更して、「選択したグループのメンバーの管理」画面で「コメントの追加」をクリックします。
   - 「別名ビルダー」を開き、「コメント」アイコン  をクリックして、別名にコメントを追加します。「設定」 > 「ツールとビュー」 > 「別名ビルダー」をクリックして、「別名ビルダー」を開きます。

レポート・コメント

すべてのユーザー・コメントのレポートを表示するには、「順守」 > 「レポート」 > 「ユーザー・コメント」をクリックします。

- 「ローカル・コメント」エンティティーは、中央マネージャー環境でのみ使用されます。ローカル・コメントは、そのコメントが定義されたシステムのローカルでの使用にとどまり、中央マネージャーには保管されません。
- 「コメント」エンティティーには、中央マネージャーに保管されているコメントが入っています。

親トピック: Guardium システムの管理

パッチのインストール方法

1 つのパッチ、または複数のパッチをバックグラウンド・プロセスとしてインストールします。

このタスクについて

このトピックでは、パッチのインストール、状況、および履歴を表示可能にし、制御する方法について説明します。

詳しくは、『一元管理』を参照してください。

この How-to トピックでは、最新の Guardium パッチのインストールに役立つ CLI のコマンドと GUI の選択項目を組み合わせて使用しています。Guardium システムは、パッチのインストール後にリブートする必要があります。

重要: ZIP 形式でダウンロードしたパッチは、アップロードおよびインストールの前に Guardium システム外部で unzip しておく必要があります。データベース構造の変更を伴うパッチについて以下の制約事項に注意してください。

- 負荷の高いレポート、監査プロセス、バックアップ、インポートなどの長期実行プロセスとの競合を避けるために、パッチのインストールは Guardium システムの負荷が低い時間に実行またはスケジュールしてください。
- パッチ・インストールの正確な所要時間は、データベース使用状況、データ分布などの考慮事項によって異なります。
- パッチのインストールはトップダウン方式、つまり最初に中央マネージャーにパッチを適用してから、アグリゲーターに適用し、最後にコレクターに適用してください。

以下の手順では、中央マネージャーとして指定および構成されている Guardium システムからステップを実行します。

1. CLI コマンド store system patch install を入力して、ネットワーク・ロケーションから 1 つのパッチ、または複数のパッチを中央マネージャーにインストールします。
2. 「設定」 > 「ツールとビュー」 > 「パッチ配布」をクリックして、パッチを CM から管理対象ユニットに移動します。

手順

中央マネージャーへのパッチのインストール

注: 圧縮された 1 つのパッチ・ファイルに複数のパッチが含まれることがありますが、一度にインストールできるパッチは 1 つのみです。複数のパッチをインストールするには、インストールする必要があるすべてのパッチをコンマで区切って選択します。CLI は内部的に、リストの各パッチに関する要求を (ユーザーによって指定された順序で) 実行依頼しますが、その際、最初のパッチはユーザーによって指定された要求時間に行われ、後続の各パッチは前のパッチの 3 分後になります。さらに CLI は、指定された (1 つまたは複数の) パッチが既に要求されているかどうか確認し、重複要求を許可しません。

1. 次のコマンドを入力します。

   ```
   store system patch install <type> <date> <time>
   ```

   ここで、<type> は sys、ftp、scp、または cd で、<date> と <time> は、YYYY-mm-dd および hh:mm:ss という形式のパッチ・インストール要求の日付と時刻です。日付と時刻を入力しない場合、または「now」を入力した場合、インストール要求時刻は「今すぐ」です。

表 1. パッチ・インストール・タイプの説明およびパラメーター

名前	記述
sys	sys オプションは、このコマンドを以前使用して Guardium システムにコピーされた圧縮ファイルに含まれる 2 番目 (またはそれ以降) のパッチをインストールする場合に使用します。以前の store system patch 実行により IBM® Guardium® システムに既にコピーされたパッチ・ファイルに含まれる、2 番目 (またはそれ以降) のパッチを適用するには、このオプションを使用します。 /var/log/guard/patches からインストールします。
ftp または scp	ftp および scp オプションは、圧縮されたパッチ・ファイルをネットワーク上のロケーションから Guardium システムにコピーします。ネットワーク上のいずれかの場所にある圧縮パッチ・ファイルからパッチをインストールするには、ftp または scp オプションを使用して、以下に示すプロンプトに応答します。 重要: ZIP 形式でダウンロードしたパッチは、アップロードおよびインストールの前に Guardium システム外部で unzip しておく必要があります。データベース構造の変更を伴うパッチについて以下の制約事項に注意してください。 ○ 負荷の高いレポート、監査プロセス、バックアップ、インポートなどの長期実行プロセスとの競合を避けるために、パッチのインストールは Guardium システムの負荷が低い時間に実行またはスケジュールしてください。 ○ パッチ・インストールの正確な所要時間は、データベース使用状況、データ分布などの考慮事項によって異なります。 ○ パッチのインストールはトップダウン方式、つまり最初に中央マネージャーにパッチを適用してから、アグリゲーターに適用し、最後にコレクターに適用してください。 Please enter the following information for file transfer: Host to import patch from: User on (host name): Full path to the patch, including name (file name may use wildcard *): (LDAP password)Password: Enter the scp/ftp port if you need to use a special port, else just press Enter key to continue: The file transfer process can take a while to complete. Leave the terminal open and do not answer any questions until the transfer is complete. Starting transfer, please wait. The file transfer is complete. Do you want to continue (yes or no)? はい List the files in the patches directory: 1. (name of file) Please choose patches to install (1-1, or multiple numbers separated by ",", or q to quit): 1 Install item 1 Patch has been submitted, and will be installed according to the request time, please check installed patches report or CLI (show system patch installed). Please don't forget to remove your media if necessary.

名前	記述
cd	cd オプションは、DVD ディスクからパッチをインストールするときに使用します。適用済みのパッチの全リストを表示するには、管理者ポータルの Guardium モニター・タブの「インストール済みのパッチ」レポートを参照してください。この同じ「Guardium モニター」タブには、「使用可能なパッチ」レポートもあります。パッチを DVD からインストールするには、このコマンドを実行する前に IBM Guardium DVD-ROM ドライブに DVD を挿入してください。DVD に含まれるパッチのリストが表示されます。

- パッチ・インストール要求を削除するには、CLI コマンド `delete scheduled-patch` を使用します。
- パッチは、インストール後、中央マネージャーにのみ残ります。スタンドアロンまたは管理対象ユニットのパッチ・ファイルは、インストール後に削除されます。
- 使用可能なパッチを表示するには、以下を使用します: `show system patch available`
- 既にインストールされているパッチ、およびインストールされるようスケジュールされているパッチを表示するには、以下を使用します。日時とインストール状況が示されます: `show system patch installed`
- Guardium アプライアンスで稼働する HTTPS ベースのファイル・サーバーを始動するには、`fileserver` コマンドを使用します。このファシリティーは、ユニットへのパッチのアップロード、またはユニットからのデバッグ情報のダウンロードを容易に実行できるようにすることを目的としています。 このファシリティーは開始のたびに、パッチのアップロード先のディレクトリーに含まれるすべてのファイルを削除します。
  注: ファイル・サーバーがアクセスすることになるファイルを生成する操作は、ファイル・サーバーの開始前に完了する必要があります (ファイルをファイル・サーバーが使用できるようにするため)。
  a. ファイル・サーバーを開始するには、fileserver コマンド `fileserver` を入力します。
  b. ファイル・サーバーを開始しています。これは https://(ユニットの名前) にあります。
  c. ファイル・サーバーを停止するには ENTER を押してください。
  d. ブラウザー・ウィンドウでファイル・サーバーを開き、以下のいずれかを実行します。
     - パッチをアップロードするには、「Upload a patch」をクリックし、指示に従います。
     - ログ・データをダウンロードするには、まず「Sqlguard logs」をクリックします。次に、目的のファイルに移動してそれを右クリックし、他のファイルの場合と同様にダウンロードします。
  e. 完了した後で CLI セッションに戻り、**Enter** を押してセッションを終了します。

UI を使用した中央マネージャーから管理対象ユニットへのパッチの移動

2. 「設定」 > 「ツールとビュー」 > 「パッチ配布」をクリックします。

   「パッチ配布」ボタンを押すと新しい画面が開き、使用可能なパッチのリストが従属関係とともに表示されます。さらにそこからパッチを選択し、選択したすべてのユニットにインストールできます。使用可能なパッチのリストは、使用可能なパッチのうち、選択したユニットごとに現在インストール済みのパッチを使用可能なパッチの従属関係リストとともに評価されたもので構成されます。 使用可能でもインストール可能ではないパッチ (依存するパッチが欠落している) は、リスト中でグレー化して表示され、選択できなくなっています。 インストールするパッチの選択は単一選択です。一度に 1 つのパッチしかをインストールできません。いったんパッチを選択して「インストール」ボタンを押すと、選択したユニットすべてにパッチをインストールするコマンドが送信されます。このパッチ・インストール処理はバックグラウンドで行われます。

3. 「一元管理」 > 「一元管理」 > 「パッチ配布」にナビゲートします。
4. 「パッチ・インストール状況」をクリックします。「パッチ・インストール状況」画面には、各ユニットに対して、失敗したインストールと不一致が表示されます。不一致とは、1 つのパッチが一部のユニットのみにインストールされているような状況であり、他のユニットでのインストールが失敗したのか、インストールしなかったのかには関係ありません。

タスクの結果

これで、パッチを適用したシステムを使用できるようになりました。ただし、パッチをインストールした後、Guardium システムをリブートする必要があります。

親トピック: Guardium システムの管理
関連情報:
Guardium パッチをダウンロードし、インストールする方法 (ビデオ)

サポート・メンテナンス

サポート・メンテナンス・フィーチャーは、パスワードで保護されており、技術サポートから指示があった場合にのみ使用できます。詳しくは、技術サポートにお問い合わせください。

親トピック: Guardium システムの管理

製品の統合

IBM Guardium を他の製品と統合できます。

- Guardium システムと通信するように BIG-IP アプリケーション・セキュリティー・マネージャー (ASM) を構成する
  (F5 Networks が提供する) Big-IP ASM を Guardium のリアルタイム・データベース・アクティビティー・モニターと併用して、Web アプリケーション層とデータベース・アプリケーション・サーバー層との間の ID 伝搬の問題を解決します。
- Hadoop 統合
  このトピックでは、Guardium で Hadoop データをモニターするための基本概念およびプロセスについて説明します。
- Guardium DAM と PIM の統合
  Privileged Information Management (PIM) の支援により、組織は、共有特権 ID の使用を自動化および追跡でき、さらにそれらの共有特権 ID の使用をモニターできます。
- QRadar と Guardium の統合
  QRadar と Guardium は両方向の情報フローで連係して動作して、Guardium データ保護ポリシーを自動的に更新し、また QRadar からのセキュリティー・インテリジェンス・イベントにほぼリアルタイムで応答することができます。
- OPTIM から Guardium へのインターフェース
  OPTIM から Guardium へのインターフェースは、Protobuf (汎用フィード・エージェント) を使用して Optim アクティビティー・ログを Guardium に送信します。
- リアルタイム・アラートおよび相関分析と SIEM 製品との統合
  データベースのアクティビティー・パターン、構造、およびプロトコルのコンテキスト・ナレッジを、SIEM システムのサード・パーティー・データベースに直接

配布します。

- InfoSphere Discovery に機密データを転送する方法
  IBM Security Guardium で識別および分類された機密データ情報を取得し、その情報を InfoSphere® Discovery に転送します。
- CEF マッピング
  ArcSight の CEF 標準は、一連の必須フィールドと、一連のオプション・フィールドを定義しています。
- LEEF マッピング
  QRadar からの Log Event Extended Format (LEEF)

Guardium システムと通信するように BIG-IP アプリケーション・セキュリティー・マネージャー (ASM) を構成する

(F5 Networks が提供する) Big-IP ASM を Guardium のリアルタイム・データベース・アクティビティー・モニターと併用して、Web アプリケーション層とデータベース・アプリケーション・サーバー層との間の ID 伝搬の問題を解決します。

このソリューションでは、BIG-IP ASM と Guardium® システムとの間のワイヤー・フォーマットとして、Google のプロトコル・バッファー (.protobuf) を使用します。

Big-IP ASM と Guardium リアルタイム・データベース・アクティビティー・モニターの統合の構成に関する情報は、F5 の Web サイトで提供されています。

親トピック: 製品の統合

Hadoop 統合

このトピックでは、Guardium で Hadoop データをモニターするための基本概念およびプロセスについて説明します。

キャパシティー・プランニング

以下のサイズ決定ガイドラインは、監査対象トラフィックが平均的な量であることを前提としています。監査対象トラフィックの量が多い場合、追加のリソースが必要になることがあります。

- コレクターごとに 10 個の管理ノードまたはサーバー・ノード
- データ・ノードに対して S-TAP が必要な場合 (すべてのコンポーネントに対しては S-TAP が必要ない場合)、コレクターごとに 20 個以上のデータ・ノード
- 物理アプライアンスを使用するとき、場合によりコレクターごとに追加ノード

ノードのプロセッサー・バリュー・ユニット (PVU) によってサイズ決定することもできますが、少ない量のトラフィックを監査する場合、この方法ではサイズが大きくなりすぎることがあります。キャパシティー・サイズ決定ガイドラインは、コレクターごとに 4000 PVU です。

統合のシナリオ

Cloudera で SSL 暗号化を使用する場合、Cloudera Navigator を使用した Hadoop 統合を参照してください。

Hortonworks Hadoop クラスターで SSL 暗号化を使用する場合、Hortonworks および Apache Ranger を使用した Hadoop 統合を参照してください。
注: Hive を使用した戻りデータの編集はサポートされていません。Hive を使用したデータ編集が必要な場合、標準 Guardium S-TAP を使用した Hadoop 統合を参照してください。

Hadoop クラスターで SSL 暗号化が必要ない場合、標準 Guardium S-TAP を使用した Hadoop 統合を参照してください。

- 標準 Guardium S-TAP を使用した Hadoop 統合
  HDFS および MapReduce モニターのために標準 Guardium S-TAP を使用して Hadoop を統合する方法について説明します。
- Cloudera Navigator を使用した Hadoop 統合
  Cloudera のネイティブ・データ・ガバナンス・ソリューションである Cloudera Navigator を使用して Hadoop を統合する方法について説明します。
- Hortonworks および Apache Ranger を使用した Hadoop 統合
  Hortonworks Data Platform に含まれる Apache Ranger を使用すると、ポリシーにより、Hive、HBASE、HDFS などの Hadoop コンポーネントに対して詳細なアクセス制御および監査を実行できます。

親トピック: 製品の統合
関連情報:
Hadoop 用の S-TAP 構成パラメーター

標準 Guardium S-TAP を使用した Hadoop 統合

HDFS および MapReduce モニターのために標準 Guardium S-TAP を使用して Hadoop を統合する方法について説明します。

Hadoop デプロイメントには、以下の 2 つの基本コンポーネントが含まれます。

- Hadoop 分散ファイル・システム (HDFS)。これにはデータが保管されます。
- MapReduce または MapReduce 2。これらは、データにアクセスし、分析するためのフレームワークを提供します。

管理コンソール・トラフィックを除くすべてのデータが HDFS を経由するため、これらの 2 つのコンポーネントでのキャプチャー・アクティビティーでは、基本監査要件が対象になります。

HDFS アクティビティーは、監査では扱いにくいので注意してください。これは、リレーショナル・データベースでのファイル・アクセスのモニターにやや類似しているためです。Hive、Big SQL、Impala など、環境で使用されている他のコンポーネントからのアクティビティーのモニターを検討してください。これらのコンポーネントは、データベース・アクセスとかなり類似するモニターをサポートしています。

編集ポリシーおよびブロック・ポリシー

Guardium は、Hive および Impala について抽出ルールを使用した編集、および S-GATE ターミネートを使用したブロックをサポートしています。V9.x では、S-TAP 使用時に BigSQL 用のブロックがサポートされていました。

Hadoop での編集およびブロック・ポリシーの使用について詳しくは、「IBM Security Guardium Deployment Guide for Hadoop Systems」を参照してください。

Kerberos

Guardium は、Kerberos セキュア・クラスターの使用をいくつかの制限付きでサポートしています。Kerberos ユーザー ID を暗号化解除するために、Guardium では、キータブ・ファイルを生成し、特定の場所に配置する必要があります。詳しくは、「IBM Security Guardium Deployment Guide for Hadoop Systems」を参照してください。

重要: HBase または Hive を使用する場合に限り、Kerberos 構成が必要になることがあります。

- 推奨事項と制限事項
  Guardium と Hadoop を統合する場合、以下の推奨事項が参考になります。
- Hadoop での S-TAP および検査エンジン
  Guardium S-TAP をデプロイし、Hadoop で使用する検査エンジンを構成します。
- Hadoop に関する Guardium ポリシーおよびルール
  Hadoop アクティビティーをモニターするための Guardium ポリシーおよびルールの作成を開始します。
- Hadoop を使用した Guardium レポート
  Hadoop 用の組み込み Guardium レポートを使用することや、Hadoop オブジェクトおよびコマンドを使用してカスタム・レポートを定義することができます。

親トピック: Hadoop 統合

推奨事項と制限事項

Guardium と Hadoop を統合する場合、以下の推奨事項が参考になります。

デプロイメントの推奨事項

コレクターのフラッディングを回避するために、および問題の診断を簡素化するために、Guardium コレクターが処理するトラフィックの量およびタイプを削減するための以下の方針を検討します。

- ネットワークを介してアプライアンスにフローする必要があるデータを制限するために、構成する検査エンジンの数を制限します。
- コレクターでログに記録されるデータの量を制限するために、ポリシーで条件を設定します。

1 つの戦略として、検査エンジンを追加し、HDFS などの追加の高ボリューム・トラフィックに対してポリシーを開く前に、Hive コマンド行照会を構成し、テストを行う場合があります。

新しい検査エンジンを構成するたびに、S-TAP を再始動する必要があります。

多くのサービスでトラフィックが生成されるため、Guardium システムは必ずモニターしてください。Guardium デプロイメントのレッドブックに、システムをモニターする方法、およびコレクターに対するトラフィックが多すぎないか確認する方法についての詳細が含まれます。

制限

標準 Guardium S-TAP を使用して Hadoop をモニターする場合、以下の制限が適用されます。

- SSL 暗号化はサポートされません (ただし、Ranger で Hortonworks を使用した場合、または Cloudera Manager で Cloudera を使用した場合は除きます)。Ranger と Cloudera Manager の統合は、この情報の個別セクションで扱います。
- UID チェーンはサポートされません。
- ブロックと編集は、Big SQL、Hive、および Impala に対してのみサポートされます。
- 構成監査システムおよび機密データ・ディスカバリーは現時点ではサポートされていません。
- Guardium は、現時点では、例えば、サービスの開始や停止を監査する管理コマンドはサポートしていません。
- Kerberos を使用した場合、Gardium のロード・バランシングおよびフェイルオーバー・オプションはサポートされません。ただし、仮想 IP アドレスが使用される F5 またはその他のロード・バランシングはオプションとして使用できます。

IBM InfoSphere BigInsights および Big SQL の考慮事項

他のほとんどの Hadoop ディストリビューションと異なり、GPFS および Big SQL での Hadoop には以下の制限が適用されます。

PGFS での Hadoop (IBM Spectrum Scale)
: BigInsights の GPFS デプロイメントには、HDFS Transparency Connector が必要です。

Big SQL
: Big SQL エンジンがインストールされているすべてのノードに S-TAP をインストールする必要があります。Big SQL のサポートは包括的であり、Guardium が Db2 に対して既にサポートしている内容と類似しています。

  Kerberos または GPFS を使用する場合、Big SQL ノードごとに特別な通信出口を構成する必要があります。Guardium は、Big SQL と対話する、動的にロードされる共有ライブラリーを提供しています。実行時に Big SQL により、SQL 要求およびユーティリティー要求が実行されると、そのライブラリー内の関数が呼び出されます。
  制約事項: モニターおよび監査は、Big SQL の出口手法を使用した場合にのみサポートされます。編集およびブロックは、S-TAP を使用した場合にのみサポートされる拡張機能です。

親トピック: 標準 Guardium S-TAP を使用した Hadoop 統合

Hadoop での S-TAP および検査エンジン

Guardium S-TAP をデプロイし、Hadoop で使用する検査エンジンを構成します。

S-TAP および GIM クライアントのデプロイ

Guardium は、Hadoop 用の CAS およびデータベース・ディスカバリーをまだサポートしていないため、S-TAP および GIM クライアントのみ必要になります。S-TAP デプロイメントでは、ご使用のオペレーティング・システムおよびカーネル・レベルに対応する正しい S-TAP をダウンロードしてください。

重要: エッジ・ノードに対しては S-TAP をお勧めします (特に、データのランディング・ゾーンとしてエッジ・ノードを使用する場合)。

検査エンジンの構成

S-TAP のデプロイ後、Guardium アプライアンスから適切な検査エンジンを定義する必要があります。検査エンジンで、特定の S-TAP ホストからモニターするトラフィックを指定します。例えば、特定の S-TAP ホストにおいて、ポート 8032 および 60000 からのトラフィックを Guardium でモニターするように検査エンジンで示す場合があります。検査エンジンでは、Hadoop、HTTP など、モニターするプロトコルも指定します。

検査エンジンを構成する前に、Hadoop 管理者と協力して、モニターする各 Hadoop ノードに関する以下の情報を収集します。

- モニターする Hadoop ノードおよびサービス
- サービスのポート番号
- サーバーの IP アドレス (例えば、S-TAP ホストの IP アドレス)

検査エンジン・プロトコルは、次の表に示されているように Hadoop ノード・タイプおよびサービスに基づいて決定します。

表 1. Hadoop ノードおよびサービスのための検査エンジン・プロトコル

Hadoop ノード	**Hadoop サービス**	**検査エンジン・プロトコル**
ネームノード	HDFS ノード名	Hadoop
ネームノード	WebHDFS の HTTP ポート	WEBHDFS
ネームノード	YARN のリソース・マネージャー	Hadoop
ジョブ・トラッカー 注: このノードは、MapReduce1 に対してのみ必要です。	MapReduce ジョブ・トラッカー	Hadoop
HBase マスター	HBase マスター	Hadoop
HBase リージョン	HBase リージョン	Hadoop
hiveserver2	Thrift プロトコル・メッセージ	HIVE
Hive メタストア	Hue から Impala および Hive データベース・ユーザーを取得するために使用される Thrift プロトコル・メッセージ。 注: 計算された属性を使用する必要があります。	HADOOP
Impala デーモン	Impala	IMPALA
Impala	Hue からの Impala	HIVE 注: Hue からの Impala では hiveserver2 が使用されます。
管理ノード	BigSQL サーバー	Db2
計算ノード	BiGSQL サーバー	Db2
Hue ノード	Oracle、MySQL、または PGSQL バックエンドでの Hue ユーザー・インターフェース	HUE
Solr 検索ノード	Solr 検索	HTTP

例えば、HDFS ネーム・ノードは、ポート `8020`、`Hadoop` プロトコルを使用し、ホスト・アドレスとして `10.0.0.21` を使用する場合があります。
Guardium ユーザー・インターフェースで「管理」 > 「アクティビティー」 > 「モニター」 > 「S-TAP 制御」にナビゲートして、または Guardium API コマンドを使用して、この情報を指定することで、検査エンジンを構成できます。Guardium API コマンドは、例えば、以下のようになります。

```
grdapi create_stap_inspection_engine client=0.0.0.0/0.0.0.0 protocol=HADOOP
    ktapDbPort=8020 portMax=8020 portMin=8020 connectToIp=127.0.0.0 stapHost=10.0.0.21
```

制限:

- Hive CLI は、Hadoop ディストリビューションでは非推奨で、Guardium ではサポートされていません。
- Impala では、Impala デーモンを実行するすべてのノードに対して検査エンジンを構成する必要があります。
- HBase では、マスター・ノードを含むすべてのデータ・ノードで S-TAP が必要です。
- Kerberos または GPFS を備えた Big SQL を使用する場合、DB2_Exit のある S-TAP を構成する必要があります。これは、Big SQL/Db2 暗号化トラフィック、GPFS、またはこの両方をキャプチャーするための安全で効率的な方法です。ただし、このシナリオではブロックおよび編集はサポートされません。Big SQL サポートについての追加情報は、Guardium に関する IBM developerWorks で入手できます。

その他の例および詳しい説明については、「IBM Security Guardium Deployment Guide for Hadoop Systems」を参照してください。
親トピック: 標準 Guardium S-TAP を使用した Hadoop 統合

Hadoop に関する Guardium ポリシーおよびルール

Hadoop アクティビティーをモニターするための Guardium ポリシーおよびルールの作成を開始します。

モニターを目的にする場合、ユーザー、モニター対象のデータ・オブジェクト、および実行するアクションまたはコマンドの観点から考えることは有用です。Guardium の用語では、これらはそれぞれ、*DB ユーザー*、*オブジェクト*、および*動詞* または*コマンド* になります。これらのエンティティーは、リアルタイム・アラートなど、特定のアクションをトリガーするためのポリシー・ルールで使用できます。

Guardium ポリシー・ルール・アクションを使用すると、ポリシー違反のログ記録やアラートに加えて、パフォーマンスのためにトラフィックをフィルタリングできるようになります。Hadoop トラフィックでは、「*S-TAP セッションを無視*」など、セッション・レベルのフィルタリング・アクションは使用できません。これは、Hadoop がリレーショナル・データベースと同じ方法でセッション管理を行わないためです。リレーショナル・データベースでは、データベースにログインするとセッションが確立され、ログアウトするまでそのセッション内で SQL トラフィックが生成されます。Hadoop では、各コマンドがそれ独自のセッションであり、クラスター全体に処理が分散されると、各コマンドにより多数のセッションが作成されることがあります。

Guardium は、コマンド行コンポーネントの失敗したログインを Hue および IBM BigSQL から表示できます。しかし、通常はこれらの失敗したログインをキャッチすることはできません。

ファイル・システム・レベルでアクセス権の例外を受け取るため、例外ドメインを使用してそれらをレポートします。

トラフィックがキャプチャーされるようにするために、標準装備の Hadoop ポリシーからポリシーの作成を開始します。トラフィックが少ないテスト環境でデフォルト・ポリシーをテストすることをお勧めします。また、表示されるノイズの量を削減するために、Hive など単一のサーバー・タイプにトラフィックを制限するアクセス・ルールをもう 1 つ追加できます。コレクターへのトラフィックのフローに問題がなければ、デフォルト・ポリシーを複製して、セキュリティーおよびコンプライアンスの要件に合致したポリシーを作成できます。

実動 Hadoop 環境のためのポリシーの詳しい説明および例については、「IBM Security Guardium Deployment Guide for Hadoop Systems」を参照してください。

親トピック: 標準 Guardium S-TAP を使用した Hadoop 統合

Hadoop を使用した Guardium レポート

Hadoop 用の組み込み Guardium レポートを使用することや、Hadoop オブジェクトおよびコマンドを使用してカスタム・レポートを定義することができます。

Guardium には、Hadoop 用の組み込みレポートがいくつか含まれます。使用可能なレポートのリストを表示するには、「マイ・ダッシュボード」 > 「新規ダッシュボードの作成 (Create a new dashboard)」にナビゲートし、「レポートの追加」をクリックします。「レポートの追加」ウィンドウで、検索フィールドに `hadoop` と入力し、使用可能な Hadoop レポートのリストを表示します。

一部の組み込みレポートでは、コンポーネント・ベースのレポートが提供されます。このレポートは、構成を検証する場合や、コンポーネントからトラフィックを正常にキャッチしているか検証する場合に便利です。「*Hadoop - 許可レポート*」、「*Hadoop - 機密オブジェクトにアクセスする特権ユーザー*」、「*Hadoop - 例外レポート*」、「*Hadoop - ユーザー・ログイン*」など、その他のレポートは、セキュリティーおよびコンプライアンスに焦点を当てています。

このセクションには、Hadoop で使用されるオブジェクトとコマンドまたは動詞のリストが含まれています。グループ・ビルダー・ツールを使用して、Guardium のグループにコマンドをカット・アンド・ペーストできます。また、ご使用の環境に基づいてユーザーおよびオブジェクトのグループを作成する必要があります。

Hadoop オブジェクト

- HDFS ファイル/ディレクトリー
- MapReduce 2 ジョブ名

  MapReduce 2 より前、MapReduce ジョブ名は個別オブジェクトとしてログに記録されませんでした。ただし、組み込み MapReduce レポートとその計算済み属性を使用して完全メッセージからジョブ名を取得することで MapReduce ジョブ名を取得できました。
- IBM Big SQL、Impala、Hive、HBase の表およびビューの名前

HDFS コマンド

HDFS の読み取りコマンド:

- getFileInfo
- getBlockLocations
- getFileLocation
- getListing

HDFS の書き込みコマンド:

- addBlock
- complete
- create
- delete
- mkdirs
- rename

HBase コマンド

HBase の読み取りコマンド:

- list
- scan

HBase の書き込みコマンド:

- createTable
- disableTable
- deleteTable

- multi

  通常、これは insert/update コマンドです。Ranger 統合デプロイメント・オプションでは、これは put コマンドです。

- drop

Big SQL、Hive、および Impala のオブジェクトおよびコマンド
: Big SQL、Hive、および Impala の照会言語は、SQL に類似しており、Guardium の他のほとんどのリレーショナル・データベースで使用される通常の解析およびロギング・ルールをサポートしています。ALTER コマンド、CREATE コマンド、管理コマンドなど、これらのコマンドの多くは Guardium コマンド・グループに既に含まれています。SQL 構文のサポート範囲は、これらのディストリビューション間で大きく異なり、Big SQL が最も幅広くサポートしています。

親トピック: 標準 Guardium S-TAP を使用した Hadoop 統合

Cloudera Navigator を使用した Hadoop 統合

Cloudera のネイティブ・データ・ガバナンス・ソリューションである Cloudera Navigator を使用して Hadoop を統合する方法について説明します。

Guardium は、標準 S-TAP を使用した Cloudera Hadoop の監査をサポートしています。詳しくは、標準 Guardium S-TAP を使用した Hadoop 統合を参照してください。

Cloudera Navigator でロギングの代替宛先として Kafka が構成されている場合、Guardium では監査イベントにサブスクライブする機能も使用できます。監査対象アクティビティーは Kafka クラスターに送信され、そこで Guardium S-TAP はイベントを取り込み、それらを解析およびログ記録のために Guardium コレクター・アプライアンスに送信します。データが Guardium に取り込まれたときから、そのデータは十分に保護され、リアルタイム・アラート、SIEM との統合、レポートとワークフロー、分析など、通常の Guardium 機能をすべて使用できます。

標準の Guardium S-TAP を使用した統合と比較すると、Cloudera Navigator 統合では、Hadoop データにアクセスするクライアントのために SSL 暗号化がサポートされます。Cloudera Navigator 統合を使用した場合、データは、Guardium アプライアンスが受信する前に暗号化解除されます。

制約事項: Cloudera Navigator 統合を使用した場合、Hadoop コンポーネントに対して Guardium ベースのブロックはサポートされません。

前提条件

Cloudera Navigator と Guardium の統合では、以下の最小ソフトウェア・リリース・レベルが必要です。

- V10.1.2 以降の IBM Security Guardium および S-TAP
- CDH 5.7、Cloudera Manager 5.8、およびこれらのリリースに含まれるバージョンの Kafka

アーキテクチャーとデータ・フロー

S-TAP が Hadoop サーバーに常駐するのではなく、Cloudera Manager エージェントが監査イベントを Hadoop コンポーネント・ログから Cloudera Navigator 監査サーバーに送信します。この時点で、Cloudera Navigator は監査イベントをその監査データベースに書き込みます。Guardium と統合するには、追加ロガーとして Kafka を設定します。これにより、Guardium は Kafka からイベント・レコードを収集します。

Hadoop クラスター内のノードまたは Hadoop クラスターの外部の個別サーバーに S-TAP をインストールできるという点で、構成は非常に柔軟です。ただし、そのサーバーが Kafka クラスターおよび Guardium アプライアンスとネットワーク接続できる必要があります。Kafka クラスターごとに 1 つの S-TAP しか指定できませんが、その S-TAP は、標準的な高可用性またはロード・バランシング技法を使用して複数の Guardium システムにトラフィックを送信できます。

この構成では、Cloudera Navigator が Hadoop コンポーネントごとにログ・イベントを生成し、S-TAP がそれらのイベントを取り込みます。Guardium ユーザー・インターフェースを使用して、Cloudera Navigator が使用するメッセージ・トピック ID を指定します。これにより、Guardium S-TAP は、ピックアップする予定のイベントを認識します。

推奨: 監査イベントを確実に保護するために、セキュア Kafka クラスターを使用してください。

- Cloudera Navigator との統合の計画
  統合を構成する前に、このトピックのタスクを実行し、確認します。

親トピック: Hadoop 統合
関連情報:
Hadoop 用の S-TAP 構成パラメーター

Cloudera Navigator との統合の計画

統合を構成する前に、このトピックのタスクを実行し、確認します。

Cloudera Navigator と統合するには、Guardium を担当するデータ・セキュリティー・チームから情報を入手するだけでなく、Cloudera および Kafka を担当する管理者から情報を入手する必要があります。開始する前に、次の情報を収集します。

- Kafka ブートストラップ・サーバーのホストおよびポート。
- Kafka クラスターで TLS および Kerberos が使用されているかどうか。
- S-TAP がインストールされているサーバーのホストおよびポート。このサーバーと Kafka クラスターのネットワーク接続、およびこのサーバーと Guardium システムのネットワーク接続があることを確認します。
- S-TAP ホストで使用されているオペレーティング・システムとそのバージョン。これは、正しい S-TAP をダウンロードし、インストールできるようにするためです。
- Guardium システムのホスト。これは、S-TAP をインストールおよび構成するために必要です。

1. モニター用のソリューションの構成
   このセクションでは、モニター用のソリューションを構成する方法について説明します。
2. Guardium と Cloudera Navigator の通信の構成
   Kafka クラスターを使用して Guardium システムと Cloudera Navigator の間の通信を確立する方法について説明します。

親トピック: Cloudera Navigator を使用した Hadoop 統合

モニター用のソリューションの構成

このセクションでは、モニター用のソリューションを構成する方法について説明します。

手順

1. Cloudera Navigator 監査コンポーネントを構成します。

   詳しくは、Cloudera の資料およびIBM Security Guardium Activity Monitoring for Cloudera Hadoop Using Navigator Integration を参照してください。

2. Kafka 用に TLS/SSL が正しく構成されていることを確認します。

   Cloudera 監査イベントを生成するために使用する Kafka クラスターを、SSL クライアント認証を要求するように構成しないでください。詳しくは、IBM Security Guardium Activity Monitoring for Cloudera Hadoop Using Navigator Integration を参照してください。

3. Guardium S-TAP をサーバーにインストールします。

   使用可能な任意の方法を使用して、Hadoop クラスターの内部または外部にある指定サーバーに S-TAP をインストールします。Guardium で、「管理」 > 「システム・ビュー」 > 「S-TAP 状況モニター」にナビゲートして、S-TAP と Guardium システムの間の接続を確認します。

   Hadoop 関連の S-TAP 構成パラメーターのリファレンスについては、Hadoop 用の S-TAP 構成パラメーターを参照してください。

4. Kafka への Cloudera Navigator 監査イベントの発行を構成します。

   Navigator 管理者またはフル管理者は、Cloudera Manager からこのタスクを実行する必要があります。詳しくは、IBM Security Guardium Activity Monitoring for Cloudera Hadoop Using Navigator Integration を参照してください。

5. Guardium と Cloudera Navigator の通信の構成
6. 構成を検証します。

   ソリューションを構成した後、「管理」 > 「システム・ビュー」 > 「S-TAP 状況モニター」に戻り、S-TAP 状況がまだ緑色であることを確認します。検査エンジンの検査は、Hadoop ソースに対してはサポートされておらず、「*未検査*」状況が常に示されます。

7. Guardium および Cloudera Navigator のポリシーをインストールします。

   モニターおよび監査の場合、Hadoop 用の標準 S-TAP モニターを使用するのではなく Cloudera Navigator 統合を使用する場合もポリシー・ルールに実質的に違いはありません。最初に Guardium ポリシーをインストールするか、デフォルト・ポリシーを使用し、Cloudera クラスターで HDFS または Hive コマンドを実行し、Guardium レポートでトラフィックを確認できるか検査します。詳しくは、IBM Security Guardium Activity Monitoring for Cloudera Hadoop Using Navigator Integration を参照してください。

親トピック: Cloudera Navigator との統合の計画
次のトピック: Guardium と Cloudera Navigator の通信の構成

Guardium と Cloudera Navigator の通信の構成

Kafka クラスターを使用して Guardium システムと Cloudera Navigator の間の通信を確立する方法について説明します。

このタスクについて

「セットアップ」 > 「ツールとビュー」 > 「Hadoop モニター」に移動し、**「クラスター情報の追加」**タイルでプラス・アイコンを選択します。

手順

1. 「セットアップ」 > 「ツールとビュー」 > 「Hadoop モニター」にナビゲートし、「クラスター情報の追加」タイルでプラス・アイコンをクリックします。
2. 「S-TAP ホスト名」メニューを使用して、Guardium システムに接続する S-TAP を選択します。
3. Kafka クラスターの「トピック名」を指定します。

   Kafka クラスターの構成設定でこれが変更されていない限り、`NavigatorAuditEvents` (デフォルト値) を使用してください。

4. 「ブートストラップ・サーバー」セクションを使用して、Guardium S-TAP からの初回接続を取得する Kafka ノードを 1 つ以上指定します。

   トピックのパーティションのリーダーであるノードは、コンシューマー要求を処理します。初回接続では、いずれかのブートストラップ・サーバーがダウンした場合にフェイルオーバーを実行できるようにするために、複数のサーバーを指定するのが最も適切です。

5. Kafka クラスターを TLS で構成する場合、「TLS の有効化」チェック・ボックスにチェック・マークを付けます。
   制約事項: Guardium は、SSL クライアント認証を要求するように構成された Kafka クラスターはサポートしません。
6. Kafka クラスターで Kerberos 認証を要求する場合、「Kerberos を使用」チェック・ボックスにチェック・マークを付けます。
   a. 「プリンシパル」フィールドを使用して、S-TAP の Kerberos プリンシパル名を指定します。

      例えば、`guardium/FullyQualifiedDomainName@kerberosDomain` などです。

   b. 「キータブ・ファイルへのパス」フィールドに、S-TAP サーバー上の Kerberos キータブ・ファイルの絶対パスを入力します。

      例えば、`/etc/krb.keytab` などです。キータブを S-TAP ユーザーおよびグループが所有していることを確認し、さらにユーザーがキータブの読み取りのみ可能であることを確認します。

7. 「保存」をクリックします。

結果のタイルでは、Hadoop モニターが構成済みであることが示され、S-TAP 状況が緑色になります。

親トピック: Cloudera Navigator との統合の計画
前のトピック: モニター用のソリューションの構成

Hortonworks および Apache Ranger を使用した Hadoop 統合

Hortonworks Data Platform に含まれる Apache Ranger を使用すると、ポリシーにより、Hive、HBASE、HDFS などの Hadoop コンポーネントに対して詳細なアクセス制御および監査を実行できます。

監査データは HDFS と Solr の両方に書き込まれます。以下の 2 つの方法で Ranger と Guardium を統合できます。

- 監査では、Guardium は、Ranger Auditing のもう 1 つのロガー・ソースとして動作します。監査対象のアクティビティーは Guardium コレクターに送信され、そこで解析され、ログに記録されます。データが Guardium に取り込まれた後、そのデータは強固なアプライアンスで十分に保護され、リアルタイム・アラート、SIEM との統合、レポートとワークフロー、分析など、通常の Guardium 機能をすべて使用できます。
- ブロックの場合、Guardium は、Ranger では動的ポリシーとして知られるものを使用して、Ranger のアクセス制御ポリシーを拡張します。

モニターおよびブロックに関して標準 Guardium S-TAP に依存する Hadoop 統合と異なり、Ranger との統合では、クライアントと Hadoop データの間での SSL 暗号化がサポートされます。Ranger 統合では、データは暗号化解除された後に、監査のために Guardium システムに送信されます。動的ポリシーを使用する Ranger 統合では、SSL のサポートに加えて、標準 S-TAP の使用でサポートされるコンポーネントよりも多くのコンポーネントに対してブロックをサポートできます。

同じクラスター内で検査エンジンと Ranger 統合の両方を使用できますが、両方の方法を同時に使用することはほとんどありません。統合パスの選択について詳しくは、Hadoop 統合を参照してください。

前提条件

Ranger との統合では、以下が必要です。

- IBM Security Guardium 10.1 (S-TAP およびアプライアンス)
- Ranger を含む Hortonworks 2.3 以降
- Solr コンポーネントが構成されている (Guardium でユーザー情報を表示できるようにする必須の Ranger コンポーネント)

アーキテクチャーとデータ・フロー

このアーキテクチャーでの重要な相違点は、S-TAP が Hadoop コンポーネントから監査データを直接収集しないことです。このアーキテクチャーでは、Ranger プラグインが監査メッセージを log4j に書き込み、log4j から S-TAP にそのメッセージが転送されます。続いて、S-TAP は、ロギング、アラート、レポート、および分析の目的で、そのメッセージを Guardium コレクターに送信します。

Ranger 統合をオンにするには、log4j_reader_enabled=1 を指定して S-TAP を構成する必要があります。

S-TAP を複数のノードにインストールできるという点で、構成は非常に柔軟です。例えば、すべてのコンポーネント・トラフィックが 1 つの S-TAP に送信されるように Ranger を構成することや、すべての HBase トラフィックが 1 つの S-TAP に送信され、Hive および HDFS が別の S-TAP に送信されるように指定することができます。

ブロックは、Guardium アプライアンスで指定されているブロック・ポリシー・ルールに対応するように Ranger アクセス制御ポリシーを拡張することで実装します。ブロックの実際の実装は、Ranger からのアクセス否認として実行されます。ブロックをアーキテクチャーおよびデータ・フローに適合させる方法について、およびブロックを実装するためのガイダンスについて詳しくは、IBM Security Monitoring and Blocking for Hortonworks Hadoop Using Apache Ranger Integration を参照してください。

- Hortonworks および Apache Ranger との統合の計画
  統合を構成する前に、このトピックのタスクを実行し、確認します。
- モニター用のソリューションの構成
  このセクションでは、モニター用のソリューションを構成する方法について説明します。

親トピック: Hadoop 統合
関連情報:
Hadoop 用の S-TAP 構成パラメーター

Hortonworks および Apache Ranger との統合の計画

統合を構成する前に、このトピックのタスクを実行し、確認します。

S-TAP およびコレクターのトポロジー

必要なトポロジーを決定します。

- 必要なコレクターの数
- 各 S-TAP でモニターするコンポーネント

一部のカスタマーは、コンポーネントごとに 1 つの S-TAP を使用しています。最低でも、HBase のために 1 つの S-TAP、他のすべてのために 1 つの S-TAP を使用することをお勧めします。
ヒント: S-TAP は、特定のコンポーネントと同じノードに配置する必要はありません。S-TAP 専用の Linux ボックスを設定できますが、Hadoop HA をサポートする場合はこれをお勧めします。
1 つの S-TAP に対する接続の数を構成する場合、次の経験則を使用してください。

- HBase: リージョン・サーバーの数 + 1
- その他のすべて: モニター対象のコンポーネントごとに 1 つ + 1

重要:

- ブロックの場合、すべての HBase リージョン・サーバーへのアクセスを確認します。これは、Guardium プラグインの JAR ファイルをこれらのリージョン・サーバーそれぞれにコピーする必要があるためです。

  高可用性フェイルオーバー・シナリオを構成する場合、フェイルオーバー・ノードの IP アドレスまたはホスト名を記録します。

高可用性およびフェイルオーバー

Hadoop では、1 次ノードで障害が発生した場合、高可用性対応の 2 次ノードを使用してデータ要求が処理されます。フェイルオーバー・シナリオで監査データを継続して収集できるようにするために、S-TAP デプロイメントには複数のオプションがあります。

S-TAP をインストールし、Hadoop クラスターに属さないシステムでそれをセットアップする
: これにより、コンポーネントのフェイルオーバー時に、新しいノードがリモート・ロガーとして S-TAP を自動的に使用する簡素な構成が実現します。S-TAP の構成に対する変更は不要です。

HDFS および Hive S-TAP に対して `localhost` を使用し、HBase に対して別個のシステムを使用する
: S-TAP ホスト・フィールドで `localhost` を使用して HDFS および Hive 用の S-TAP をインストールし、その後、HBase 用のエッジ・ノードとして別個のシステムを使用します。S-TAP をすべてのノードおよびリージョン・サーバーにインストールする代わりに、この方法を使用できます。この方法を使用することをお勧めします。

クラスター内のノードに S-TAP をインストールする
: このモデルでは、各コンポーネント用の 1 次ノードおよびスタンバイ・ノードに S-TAP をインストールします。

  S-TAP ホスト・フィールドで `localhost` を使用して、クラスター内のすべてのノード、および HBASE のすべてのリージョン・サーバーに S-TAP をインストールします。この方法は推奨されていません。

Guardium ロード・バランシング

Ranger 統合を有効にすると、Guardium S-TAP およびエンタープライズ・ロード・バランシング・オプションがサポートされます。

Ambari および Ranger 情報の収集

セットアップの重要な部分は、Hadoop 管理インターフェースである Ambari を使用して行います。構成を実行するには、以下の情報が必要です。

Ambari

- サービス管理者アカウントなど、*log4j* 構成を更新および保存する特権を持つユーザーの ID およびパスワード。簡潔にするために、これを管理アカウントおよび管理パスワードと呼びます。
- ポートと IP アドレスまたはホスト名。
- クラスター名。

Ranger
: ブロックを構成する場合に限り、以下の情報が必要です。ブロックの構成について詳しくは、IBM Security Monitoring and Blocking for Hortonworks Hadoop Using Apache Ranger Integration を参照してください。

  - *log4j* 構成を更新および保存できるサービス管理者アカウント。
  - ポートと IP アドレスまたはホスト名。

必要なポートを開く

以下のポートが開いているか確認します (デフォルト・ポートの使用を想定しています)。

- モニターのために、S-TAP があるノードと Ranger サーバーとの間のポート 5555 を開きます。
- ブロックのために、ポート 5556 を開き、S-TAP と Guardium プラグインがあるクラスター内のすべてのノードとの間で通信を実行できるようにします。

親トピック: Hortonworks および Apache Ranger を使用した Hadoop 統合

モニター用のソリューションの構成

このセクションでは、モニター用のソリューションを構成する方法について説明します。

始める前に

Ranger と Guardium の通信の構成を開始する前に、Ambari を使用して Ranger プラグインを構成します。詳しくは、「IBM Security Monitoring and Blocking for Hortonworks Hadoop Using Apache Ranger Integration」ガイドまたは Hortonworks の資料を参照してください。

資料で、2 つの Hadoop 監査構成設定が欠落しています。インストール・マニュアルに以下のステップを追加してください。

監査ログを log4j に書き込むように Ranger プラグインを構成します。

HDFS

セクション「Custom ranger-hdfs-audit」に以下を追加します。

xasecure.audit.destination.log4j=true

xasecure.audit.destination.log4j.logger=xaaudit

Hive

セクション「Advanced ranger-hive-audit.xml」に以下を追加します。

xasecure.audit.destination.log4j=true

xasecure.audit.destination.log4j.logger=xaaudit

GUI から Ranger を構成する方法よりも、Python スクリプトを使用して Ranger を構成する方法をお勧めします。

1. Guardium と Ranger の通信の構成
   Guardium システムと Ranger の間で通信を確立する方法について説明します。
2. S-TAP のインストールおよび構成
   Ranger 統合のために S-TAP をインストールし、構成します。
3. Hadoop サービスのモニターの有効化
   特定の Hadoop コンポーネントに対してモニターを有効にします。

次のタスク

これらのセットアップ・ステップを完了した後、Guardium および Ranger のポリシーをインストールします。モニターおよび監査の場合、Hadoop 用の標準 S-TAP モニターを使用するのではなく Ranger を使用する場合も、ポリシー・ルールに実質的に違いはありません。詳しくは、「IBM Security Monitoring and Blocking for Hortonworks Hadoop Using Apache Ranger Integration」を参照してください。
親トピック: Hortonworks および Apache Ranger を使用した Hadoop 統合

Guardium と Ranger の通信の構成

Guardium システムと Ranger の間で通信を確立する方法について説明します。

このタスクについて

このタスクでは、Guardium システムと Ranger の間で通信を確立する方法について説明します。

手順

1. 「設定」 > 「ツールとビュー」 > 「Hadoop モニター」にアクセスします。
2. 「クラスター情報の追加」セクションで  をクリックして、新しい構成の定義を開始します。
3. 「名前」フィールドに、構成の名前を入力します。
4. 「Hadoop ディストリビューション」メニューから `Hortonworks` を選択します。
5. 「ホスト名/IP」フィールドに、Ambari サーバーのホスト名または IP アドレスを入力します。
6. 「ポート番号」フィールドに、Ambari サーバーのポート番号を入力します。 このフィールドを空白のままにすると、構成ではデフォルト・ポートの 8080 が使用されます。
7. 「クラスター名」フィールドに、Hadoop クラスターの名前を入力します。
8. 「ユーザー名」フィールドに、Ambari 管理者のユーザー名を入力します。
9. 「パスワード」フィールドに、Ambari 管理者アカウントのパスワードを入力します。
10. 「接続のテスト」ボタンをクリックして、構成を検証します。
11. 「保存」をクリックして構成を保存します。

タスクの結果

「Hadoop モニター」ページから新しい構成を使用できます。
親トピック: モニター用のソリューションの構成
次のトピック: S-TAP のインストールおよび構成

S-TAP のインストールおよび構成

Ranger 統合のために S-TAP をインストールし、構成します。

始める前に

S-TAP の要件およびデプロイメント・オプションについて詳しくは、Hortonworks および Apache Ranger との統合の計画を参照してください。

手順

1. S-TAP をインストールし、Ranger 統合のために有効にします。 トラフィックを処理するために複数の S-TAP が必要になる場合があります。例えば、HDFS、Hive、および Kafka トラフィックのためにネーム・ノードで S-TAP を 1 つ構成し、すべての HBase トラフィックのために HBASE マスター・ノードで S-TAP を 1 つ構成します。
2. 監査のために guard_tap.ini を構成します。
   a. guard_tap.ini をテキスト・エディターで開きます。 これらの設定では UI および GIM はサポートされていないため、ファイルを直接編集する必要があります。
   b. 以下にリストされているパラメーターを追加します。 ご使用の環境に合わせて値を更新します。

```
; Settings for log4j
logging log4j_reader_enabled=1
log4j_port=5555
log4j_listen_address=0.0.0.0
; Maximum number of connections to support from the log4j service
log4j_num_connections=50
```

   c. 設定を更新した後、S-TAP を再始動します。

親トピック: モニター用のソリューションの構成

前のトピック: Guardium と Ranger の通信の構成
次のトピック: Hadoop サービスのモニターの有効化
関連情報:
Hadoop 用の S-TAP 構成パラメーター

Hadoop サービスのモニターの有効化

特定の Hadoop コンポーネントに対してモニターを有効にします。

このタスクについて

このタスクでは、Guardium によるモニターをどの Hadoop コンポーネントに対して有効にするか定義する方法について説明します。

手順

1. 「設定」 > 「ツールとビュー」 > 「Hadoop モニター」にアクセスします。
2. サービスの構成を開始するために、Hadoop クラスターの をクリックします。
3. 「サービス」メニューを使用して、モニターを有効にする Hadoop コンポーネントを選択します。
4. 「S-TAP ホスト名/IP」メニューを使用して、Ranger から監査イベントを収集する S-TAP を選択します。
5. 「ポート番号」フィールドに、リスナーのポート番号を入力します。 このフィールドを空白のままにすると、サービスはデフォルト・ポートの 5555 を使用します。
6. 「モニターをただちにアクティブ化 (Activate monitoring immediately)」を選択して、選択したサービスのモニターを有効にします。
7. 「保存」ボタンをクリックして、サービスの構成を保存します。
   重要: Hadoop 管理者は、サービス構成に行った変更を有効にするために、Hadoop サービスを再始動する必要があります。サービスを再始動する前に、管理者に以下の log4j 構成の確認を依頼してください。

```
# Configuration for Guardium integration with Ranger log4j logging.
log4j.appender.guardlistener=org.apache.log4j.net.SocketAppender
log4j.appender.guardlistener.Port=5555
log4j.appender.guardlistener.RemoteHost=hw-cl5-01.guard.swg.usma.ibm.com
log4j.logger.xaaudit=ALL,guardlistener
```

   また、Hadoop 管理者に、`custom ranger-<service>-audit` の以下の設定を確認するよう依頼してください。

```
xasecure.audit.destination.log4j=true
xasecure.audit.destination.log4j.logger=xaaudit
```

タスクの結果

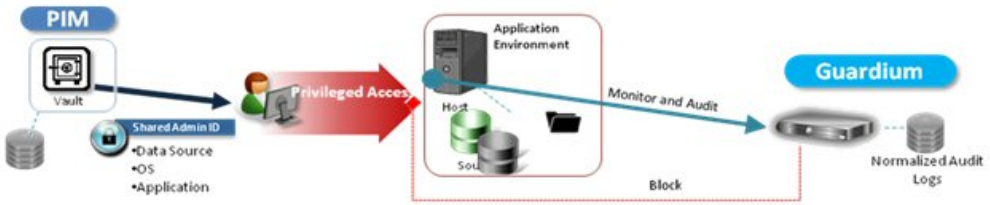
「Hadoop モニター」ページから、有効化したサービスに緑色のチェック・マーク・アイコンが付いていることを確認します。サービスがポート情報を表示せず、S-TAP 状況が「S-TAP がインストールされていません (S-TAP not installed)」 である場合は、構成を編集して有効な S-TAP を指定してください。
親トピック: モニター用のソリューションの構成
前のトピック: S-TAP のインストールおよび構成

Guardium DAM と PIM の統合

Privileged Information Management (PIM) の支援により、組織は、共有特権 ID の使用を自動化および追跡でき、さらにそれらの共有特権 ID の使用をモニターできます。

ここでは、データベースにログインしている実際のユーザー (人) を可視化できるようにするために、PIM アクティビティー・データを Guardium DAM データと統合します。

次の図は統合を表しています。



この統合の主な目的は、以下のとおりです。

- PIM データ (PIM によって管理されるリース履歴 (誰が共有アカウントを使用したか)、資格情報、データベースなど) を、Guardium アプライアンスで可視化する。
- PIM 情報と相関関係のある DAM 情報を提供する。例えば、Guardium で、本日のデータベース・ユーザー、および特定ユーザーが発行した実際の要求を表示できるようになります。この統合により、データベース・ユーザーおよび共有 ID をリースした実際の PIM ユーザーの両方を使用できるようになります。

インストール

Guardium パッチ (v10.1p103) を使用すると、PIM 統合機能をインストールできます。PIM 統合は、スタンドアロンの Guardium システムおよびフェデレーテッド環境で使用できます。

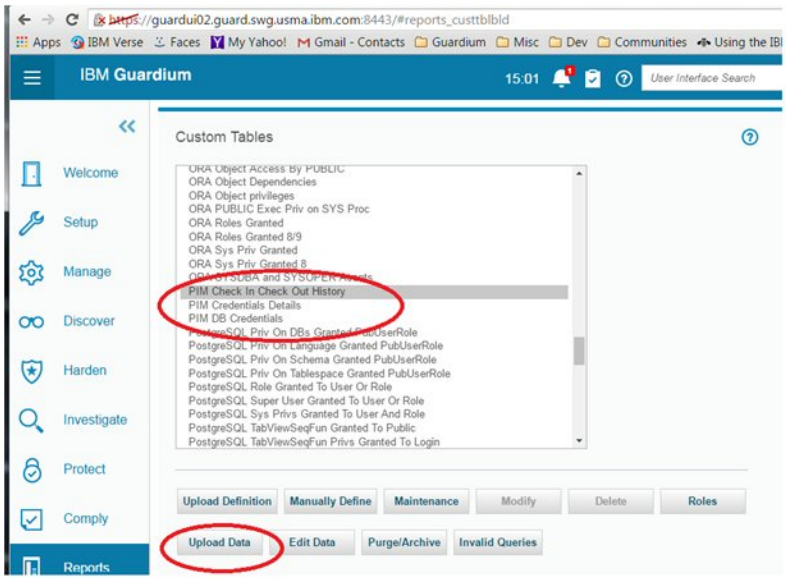
注: PIM アクティビティー・データが既に実装済みであることを前提とします。

以下の手順を行います。

1. Guardium システムにデータを取り込みます。

   データ・ソースを選択し、Guardium UI で「レポート」 > 「レポート構成ツール」 > 「カスタム表ビルダー」を選択します。

   3 つの PIM 事前定義表を見つけ、選択し、表ごとに自動データ・アップロードをスケジュールします。



   Guardium システムへの PIM 表のアップロード

   Guardium 中央マネージャーを使用する場合、Guardium UI で「管理」 > 「中央マネージャー」 > 「PIM データ配布」を選択します。これを行うことで、中央マネージャーからすべての管理対象ユニットへのデータ配布をスケジュールします。

2. データが管理対象ユニットに取り込まれた後、CLI コマンドの store pim_correlation_mode を使用して、PIM データと Guardium セッション・データの相関関係を有効にします。

   CLI コマンド

   store pim_correlation_mode

   使用法: store pim_correlation <state>

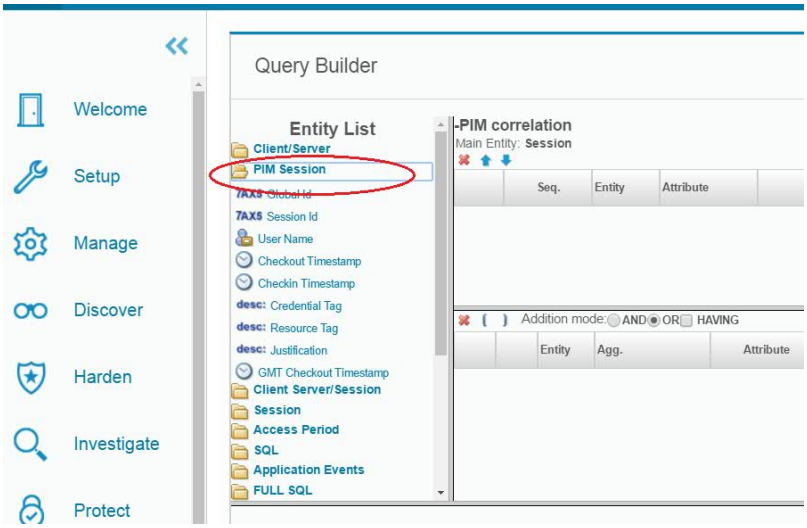
   where state is on/off. on で有効になり、off で無効になります。

   表示コマンド

   show pim_correlation_mode

3. 相関を実行するには、Guardium GUI で「順守」 > 「カスタム・レポート作成」 > 「PIM データ相関」を選択します。

   アクセス・ドメインのレポートを通じて、相関関係のあるデータを確認できます。



   アクセス・ドメインの PIM セッション

親トピック: 製品の統合

QRadar と Guardium の統合

QRadar と Guardium は両方向の情報フローで連係して動作して、Guardium データ保護ポリシーを自動的に更新し、また QRadar からのセキュリティー・インテリジェンス・イベントにほぼリアルタイムで応答することができます。

IBM QRadar は、セキュリティー・インテリジェンス・ツールであり、セキュリティー情報とイベントのモニター、異常を検出するためのカスタマイズ可能ルールの使用、およびインシデント・フォレンジックと脆弱性管理のためのツールの提供によって脅威からの保護を実現します。

IBM Guardium は、サーバーに保管されるデータの保全性の実現を支援する、データ・セキュリティーおよびデータ・プライバシーのためのソリューションです。Guardium では、ポリシーおよび包含/除外リスト (Guardium グループと呼ばれます) を使用してデータへのアクセスが制御されます。

QRadar および Guardium ソリューションでは、QRadar セキュリティー・イベントに応じてアクションをトリガーするための QRTrigger フレームワークが活用されます。構成設定に応じて、QRadar イベントにより、そのイベント自体がもたらす情報に基づき、Guardium グループに新しいメンバーが追加されます。さらに、メンバーシップの変更をただちに有効にするために、グループに関連する Guardium ポリシーが自動的に再インストールされます。

QRadar および Guardium ソリューションを使用すると、単一の Guardium コレクター、または Guardium 中央マネージャー (CM) によって制御される Guardium コレクターのグループを更新できます。

QRadar と Guardium の連携

従来の QRadar と Guardium の統合は、片方向の情報フローで、Guardium がアラートと脆弱性評価 (VA) レポートを QRadar に送信していました。

データベースの一般的なアラート・ユース・ケース:

- 失敗したログイン
- 無許可アクセス
- SQL エラー・コード (例えば、SQL インジェクション攻撃)
- ユーザーによる特権のエスカレート試行
- ユーザーによる機密データに間接アクセスするためのトリガーおよびビューの作成

現在、QRadar と Guardium は、両方向の情報フローで連係して動作できます。

その他のユース・ケース:

- 暗号漏えいしたマシンからのアクセスをブロックする
- 疑わしい対象になったユーザー ID によるアクセスに対する監査レベルを上げる
- Privileged Identity Management (PIM) システムに登録された特権共有ユーザー ID によるアクセスに対する監査レベルを上げる

QRadar イベントに基づく Guardium ポリシーの更新

QRadar および Guardium ソリューションのデプロイ手順を以下に示します。

1. ソリューション・ファイルをインストールします。
2. Guardium でクライアント ID およびパスワードを設定します。
3. QRadar で転送先を構成します。
4. QRadar イベントをソリューションに送信するルールを構成します。
5. 必要に応じて、統合のために Guardium のグループおよびポリシーを定義します。

Guardium バージョン 10.1 以降には、この統合をサポートすることを目的とした、以下の 3 つの事前定義グループがあります。

- QRadarBlockingConnection
- QRadarAlertingConnection
- QRadarLogConnection

これらの各グループには、次のタプル構造があります。

<クライアント IP>、<ソース・アプリケーション>、<DB ユーザー>、<サーバー IP>、<サービス名>、<OS ユーザー>、<DB 名>

3 つのルール (ブロック・ルール、アラート・ルール、およびロギング・ルール) を含む「QRadarPolicy」という名前の事前定義 Guardium ポリシーがあります。各ルールは、上記のリストの各グループに関連付けられています。

QRadar および Guardium ソリューションのインストール方法

QRadar および Guardium ソリューションのインストールに関する詳細な指示については、次の IBM Developerworks の記事を参照してください。

https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/W746177d414b9_4c5f_9095_5b8657ff8e9d/page/QRGuardium

Guardium のセットアップ

QRadar および Guardium ソリューションを Guardium REST API に対して認証できるようにするには、クライアント ID を Guardium に登録し、関連するクライアント・パスワードを取得する必要があります。

クライアント ID の登録は、Guardium の grdapi コマンド行ユーティリティーを使用して行います。この操作は、1 回のみ実行します。クライアント ID を登録すると、クライアント・パスワードなど、新しいクライアントの詳細を含む JSON エントリーが作成されます。

```
> grdapi register_oauth_client client_id=qrguardium
ID=0
{"client_id":"qrguardium","client_secret":"3ac89782-ce55-
4f24-b795-b6c76ecc4045",
"grant_types":"password","scope":"read,write","redirect_uri"
:"https://joeApp"}
ok
```

トラブルシューティング・ログ

QRadar および Guardium ソリューションは、操作の管理およびトラブルシューティングを支援する多数のログ・ファイルを提供します。これらのログ・ファイルには以下が含まれます。

表 1. ログ・ファイル

パラメーター名	記述
guardiumEvents_audit.log	これは、QRadar イベントに基づいて Guardium に対して行われるすべての変更の監査ログです。各行は JSON オブジェクトで、処理されたイベントの ID、タイム・スタンプ、および詳細が含まれます。
QRListener.log	QRadar から転送されたイベント・データを受信するリスナー・プロセスのログ出力。
HANDLER_<イベント名>.log	特定イベントの専用ハンドラー AL のログ出力。
RESPONSE_<イベント名>.log	カスタム応答 AL のログ出力 (この AL がその AssemblyLine 名に基づくロギングを実装している場合)。例えば、これは、次の Javascript を使用して Log Appender File Path パラメーターが計算されるように設定することで実行できます。 return “logs/” + task.getShortName() + “.log”;

親トピック: 製品の統合
関連情報:
ディレクトリー・インテグレーター統合 (ビデオ)

OPTIM から Guardium へのインターフェース

OPTIM から Guardium へのインターフェースは、Protobuf (汎用フィード・エージェント) を使用して Optim アクティビティー・ログを Guardium に送信します。

このインターフェースの目的は、OPTIM アクティビティーに対して Guardium 監査機能を使用することです。この監査機能には、 レポート・ツール (ユーザー定義の照会とレポート)、監査プロセス (ロール/ユーザー/グループ、ユーザー定義の状況フロー・プロセス、 エスカレーション、エクスポートなどに 1 つのタスクを割り当てることを可能にするワークフローの自動化機能)、およびしきい値アラートが含まれます。

Optim 監査アクティビティー情報には、アクセスの詳細、セッション番号、アクティビティー・タイプ (verb)、表 (オブジェクト)、詳細 (フィールド)、実行時間 (応答時間)、エラーの数 (影響を受けたレコードの数) が含まれます。

データは Guardium 標準オブジェクト・モデルにマップされます。

OPTIM の監査を有効にするには、OPTIM による有効化処理と、Guardium での次のステップが必要です。 (1) ユーザーを Optim 監査ロールにリンクする (2) 事前定義レポートを該当するペインに追加する (3) スニファーを有効にする (4) ポリシー・アクションを「値を含むデータをログに記録する (Log Data With Values)」に設定する。

このインターフェースには、optim-audit ロール、optim-audit ロールのデフォルトのレイアウト (psml ファイル)、7 つの事前定義レポートが含まれています。

これらのレポートは以下のとおりです。

- Optim - Optim サーバー当たりの失敗した要求の要約
- Optim - ユーザー当たりの要求の実行
- Optim - Optim サーバー - 表の使用状況の詳細
- Optim - 要求ログ
- Optim - 表の使用状況の要約
- Optim - 要求の要約

注: 「optim-audit」ロールおよびユーザーを作成すると、OPTIM 監査という 1 つのタブのみ表示されます。ユーザーが生成できるカスタム・レイアウトを持つロールと同様に、このロールのレイアウトは単独で使用するためのものです (optim-audit ユーザーには他のユーザー・ロール・タブは不要です)。ただし、ユーザー・ロールは必須であるため、ユーザーが optim-audit ロールを持った時点でレイアウトのマージがオフになり、optim の対象項目のみ取得できるようになります。同じように動作する他のロールに「review-only」と「inv」があります。
注: optim-audit ロールを作成して保存した後に、「ユーザー・ブラウザー」メニュー内の「レイアウトの生成」選択項目をクリックし、 「リセット」をクリックして、そのロールに関連付けられているレイアウトを取得してください。「ユーザー・ブラウザー」内でロールを変更した場合は、この作業を再度行ってください。

親トピック: 製品の統合

リアルタイム・アラートおよび相関分析と SIEM 製品との統合

データベースのアクティビティー・パターン、構造、およびプロトコルのコンテキスト・ナレッジを、SIEM システムのサード・パーティー・データベースに直接配布します。

このタスクについて

Guardium® は、大量のデータベース・トラフィックを前処理し、重要な情報を抽出します。抽出した後は、圧縮した要約を外部の SIEM (Security Incident Event Manager) システム (ArcSight、Envision、QRadar など) に送信します。 したがって、SIEM 製品が大量のトラフィック・ストリームを処理する必要がなくなります。むしろ、すべてのアクティビティーの相関付け、無許可の動作や疑わしい動作に対するアラート、イベント・ログに関する規制コンプライアンス要件への対応に集中することが可能になります。

この Guardium SIEM (Security Incident Event Manager) 統合は、以下のいずれかの方法で実施できます。

- Syslog 転送 (アラートおよびイベントの最も一般的な方法)
- CLI コマンド store remotelog を使用して、機能/優先度およびホスト (宛先) への Syslog 転送を指定する。
- ArcSight、Envision、および QRadar 用の Guardium テンプレートの使用
- SCP/FTP (CSV または CEF ファイルは外部リポジトリーに送られ、SIEM システムはこの外部リポジトリーからアップロードして解析する必要があります。)

Guardium は、データベースのアクティビティー・パターン、構造、およびプロトコルのコンテキスト・ナレッジを、SIEM システムのサード・パーティー・データベースに直接配布します (Guardium は SIEM システムの資格情報を持ち、SIEM データベースへの SIEM スキーマによる直接書き込みが可能です)。Guardium のエンティティーをサード・パーティー・スキーマにマップする必要があるため、Guardium サポートにお問い合わせください。

注: SIEM システムもリモート・ロギングを有効にして、syslog 内に定義された適切な機能/優先度を listen できるようにする必要があります。

Guardium のリアルタイム・セキュリティー・アラートと相関解析を、SIEM およびログ管理製品と組み合わせることによって、企業は以下の能力を高めることができます。

- 外部からの攻撃、信頼された内部関係者、コンプライアンス違反によるリスクを事前に識別して緩和する。
- Sarbanes-Oxley (SOX)、PCI-DSS (クレジット・カード業界のデータ・セキュリティー基準)、 データ・プライバシー規制に応じた自動制御を実施する。
- 企業データベースやアプリケーションといったデータ・センターのコアにあるクリティカル・ログ/イベントと合わせて、システムおよびネットワークのイベントを管理し、会社全体の相関付け、法務、インシデントの優先付け、レポート作成を行う。

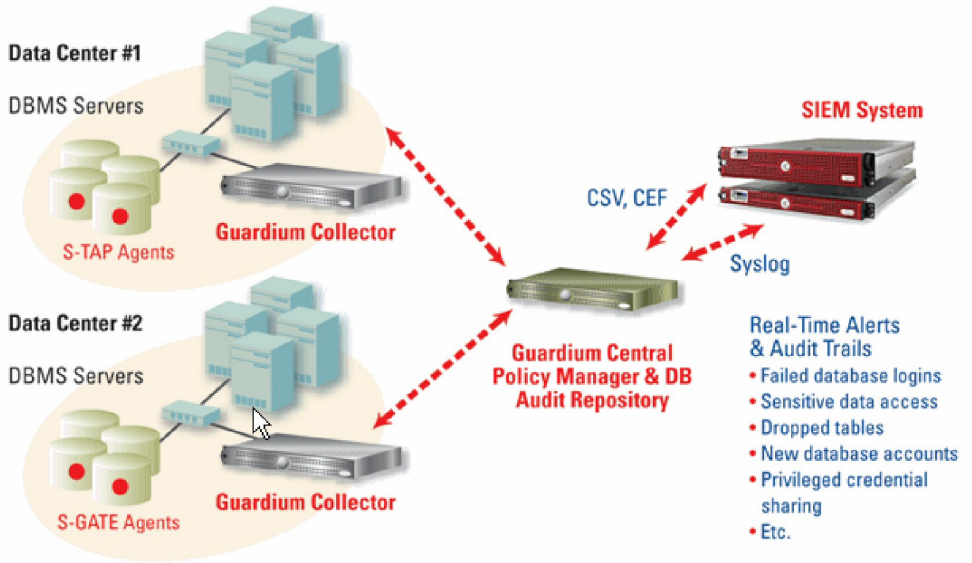
Security Information and Event Management (SIEM) ソリューション (Security Event Management (SEM) ソリューションとも呼ばれる) は、QRadar、ArcSight、CA、Cisco MARS、LogLogic、RSA enVision および SenSage の各社より提供されています。SIEM 製品は、Guardium のデータベース・アクティビティー・モニター・ソリューションを補完する製品です。これらの製品は、Guardium によるデータベース・イベントのフィルター処理および前処理機能を使用して、SOX、PCI-DSS、およびデータ・プライバシーに応じた 100% の可視性とデータベース分析も行うことができます。

SIEM テクノロジーは、ネットワーク・ハードウェアおよびアプリケーションで生成されたセキュリティー・アラートをリアルタイムで分析します。これにより、企業はネットワーク攻撃に対してより迅速に対処したり、毎日生成される大量のログ・データを整理したりすることができるようになります。SIEM ソリューションは、ログ・ベースの相関エンジンです。

SIEM ソリューションは、監査ではなく、主に検出とセキュリティーを重点的に扱います。 他のログのデータを組み合わせることで、ハイレベルの分析を行います。 さらに多くのデータ (IP アドレスやルーターなど) の相関付けをしますが、データベースの可視性はあまり得られません。法務基準、デジタル署名、監査モニター機能には対応していないため、 即座に情報を得るのには使用できますが、履歴を証拠として扱うためには使用できません。

Security Information and Event Management (SIEM) のユーザーは、内部の DBMS ユーティリティーで生成された未加工のログをインポートする必要があります。DBMS ロギング・ユーティリティーのパフォーマンス、このユーティリティーによって生成される未フィルター情報、および必要な細分化された情報がないことにより、不都合が生じます。

Guardium のユーザー・インターフェースを使用することにより、各種の SIEM ツールと統合するための Guardium の構成が簡単に行えます。



注: SIEM との統合において、Guardium システムではレポートおよびポリシーに変更はありません。既存のポリシーおよびレポートの継続使用、アラートのトリガー、および SIEM システムへのレポートの送信を行うことができます。

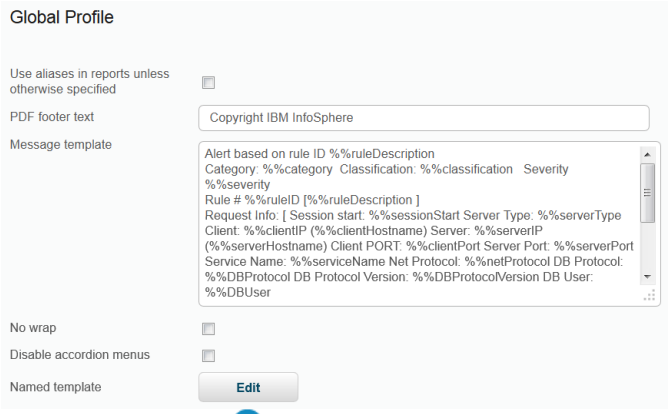
SIEM と Guardium の統合では、QRadar、Envision、および ArcSight 用の事前定義テンプレートがあるため、それらを定義する必要がありません。ルール・アクションにおける適切なメッセージ・テンプレートを選択できます。

デフォルトのメッセージ・テンプレートの変更、syslog 転送に関するパラメーターの指定、およびエクスポートする CSV ファイルまたは CEF ファイルの作成を行うことができます。

注: CEF を使用できるのは、ArcSight の場合のみです。その他の SIEM 製品は別の形式を使用し、CEF を使用しません。

SIEM 製品が送信された情報を認識できるようにするため、「グローバル・プロファイル」を介してメッセージ・テンプレートを変更する必要があります。これは、SIEM ソリューションと Guardium の間で合意された形式であり、SIEM 製品は着信メッセージを解析して、そのデータベースを新しいイベント/データで更新することが可能になります。

1. 「グローバル・プロファイル」を開くには、「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」をクリックします。
2. 「名前付きテンプレート」の「編集」をクリックします。



3. テンプレートを選択するか、 アイコンで新規のテンプレートを作成します。

Guardium アプライアンスは、syslog メッセージをリモート・システムに送信するよう構成できます。特定のタイプの syslog メッセージを特定のホストに送信できます。syslog メッセージのタイプは、メッセージの機能-優先度から判別されます。

機能の例として、all、auth、authpriv、cron、daemon、ftp、kern、local0、local1、local2、local3、local4、local5、local6、local7、lpr、mail、mark、news、security、syslog、user、uucp があります。優先度の例として、alert、all、crit、debug、emerg、err、info、notice、warning があります。

他のアプリケーションが使用可能な情報を含んだレポート、または大量のデータを含んだレポートを、CSV ファイル形式でエクスポートすることができます。 レポート、エンティティー監査証跡、およびプライバシー・セット・タスクの出力は、CSV (区切り文字区切り値) ファイルにエクスポート可能です。 また、CSV ファイル出力を syslog に書き込むことができます。 リモート syslog 機能を使用する場合、出力 CSV ファイルがリモート syslog ロケーションに送信されます。

CSV ファイルや CEF ファイルの各レコードは、レポートの 1 行を表しています。エクスポートする CSV ファイルの再フォーマット可能なツールについては、Guardium サポートまでお問い合わせください。

Guardium アプライアンスは、store remotelog CLI コマンドを使用して syslog メッセージをリモート・システムに送信するよう構成できます。特定のタイプの syslog メッセージを特定のホストに送信できます。 syslog メッセージのタイプは、メッセージの機能-優先度から判別されます。

機能の例として、all、auth、authpriv、cron、daemon、ftp、kern、local0、local1、local2、local3、local4、local5、local6、local7、lpr、mail、mark、news、security、syslog、user、uucp があります。優先度の例として、alert、all、crit、debug、emerg、err、info、notice、warning があります。

他のアプリケーションが使用可能な情報を含んだレポート、または 大量のデータを含んだレポートを、CSV ファイル形式でエクスポートすることができます。 レポート、エンティティー監査証跡、およびプライバシー・セット・タスクの出力は、CSV (区切り文字区切り値) ファイルにエクスポート可能です。 また、CSV ファイル出力を syslog に書き込むことができます。 リモート syslog 機能を使用する場合、出力 CSV ファイルがリモート syslog ロケーションに直ちに送信されます。

CSV ファイルや CEF ファイルの各レコードは、レポートの 1 行を表しています。

CSV ファイルへの syslog メッセージの送信およびレポートのエクスポートを行うには、以下の手順を実行します。
注: 監査プロセス定義内のファイルは、SIEM ベンダーが正しく解析できるようにするために、zip しないでください。

1. 監査プロセス・ファインダーを開くには、「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」をクリックします。
2. アイコンをクリックしてプロセスを追加するか、ドロップダウン・リストから既存のプロセスを選択します。
3. 「監査タスク」にある「新規監査タスク (New Audit Task)」をクリックします。
4. 説明を入力し、「レポート」を選択します。
5. ドロップダウン・リストからレポートを選択して、「CSV/CEF ファイル・ラベル」に入力します。
6. 「CSV ファイルへのエクスポート」と「Syslog に書き込む」を選択します。名前付きテンプレートをドロップダウンリストから選択します。
7. タスク・パラメーターの下で、カレンダー・アイコンを使用して「期間の開始日を入力>=」と「期間の終了日を入力<=」を選択します。
8. 「適用」をクリックします。

CSV/CEF ファイルもスケジュールに基づいて SIEM ホストにエクスポートできます。監査タスクを変更するか追加します。

1. 「順守」 > 「ツールとビュー」 > 「監査プロセス・ビルダー」をクリックして監査プロセス・ファインダーを開き、監査タスクを変更するか追加します。
2. 「CSV ファイルへのエクスポート」または「CEF ファイルのエクスポート」を選択します。
   注: アクセス・レポートは CEF 形式または LEEF 形式で保存して送信できますが、その他のレポート (Guardium ログイン、統合アクティビティー・ログ、CAS イベントなど) は CEF または LEEF にマップできません。
3. 「Syslog に書き込む」のチェック・マークを外します。そうしないと、ファイルでなく syslog メッセージが生成されます。
4. 「管理」 > 「データ管理」 > 「結果エクスポート (ファイル)」をクリックして、CSV/CEF のエクスポート・メニューを開きます。
5. 「SCP」プロトコルまたは「FTP」プロトコルを選択します。 次に、「ホスト」、「ディレクトリー」、「ユーザー名」、「ポート」、および「SCP/FTP パスワード」に入力します。
6. 「スケジューリング」セクションで、「開始時刻」、「再始動」頻度、「繰り返し」頻度、「スケジュールの基準」 (日、週、または月単位)、「開始時刻のスケジュール設定」を定義します。ボックスにチェック・マークを付けて、従属ジョブを自動実行します。
7. 「保存」をクリックして変更をコミットするか、「リセット」をクリックしてフィールドをクリアします。

ポリシーのアラートが syslog に送られるようにするため、syslog への通知の送信をトリガーするように、例外ルール、アクセス・ルール、および抽出ルールを変更しなければなりません。このアクションは、「ポリシー・ビルダー」に移動することで行うことができます。ポリシー・ルールは、E メールで送信、または syslog に送信して転送することが可能です。

1. 「設定」 > 「ツールとビュー」 > 「ポリシー・ビルダー」をクリックして、ポリシー・ビルダーを開きます。
2. 目的のポリシーを選択して、「ルールの編集」をクリックします。
3. 「ルールの追加...」 > 「例外ルールの追加」をクリックします。
4. 「記述」、「カテゴリー」、「分類」に入力し、ドロップダウン・リストから「重大度」レベルを選択します。

「ポリシー違反」レポートは、レポート期間中にログに記録されるすべてのポリシー・ルール違反について、 「ポリシー・ルール違反」エンティティーからのタイム・スタンプ、 アクセス・ルールの記述、クライアント IP、サーバー IP、データベース・ユーザー名、 「ポリシー・ルール違反」エンティティーからの SQL 文字列全体、重大度の記述、 およびその行の違反数を提供します。このレポートを使用することで、違反をグループにしてインシデントを作成し、各違反の重大度を設定して、インシデントをユーザーに割り当てることができます。

親トピック: 製品の統合

InfoSphere Discovery に機密データを転送する方法

IBM Security Guardium で識別および分類された機密データ情報を取得し、その情報を InfoSphere® Discovery に転送します。

IBM Guardium と InfoSphere Discovery にはどちらにも、社会保障番号、クレジット・カード番号などの機密データを識別し、分類する機能があります。

IBM Guardium 製品のカスタマーは、双方向インターフェースを使用して、識別された機密データ情報を一方の製品から他方の製品に転送できます。

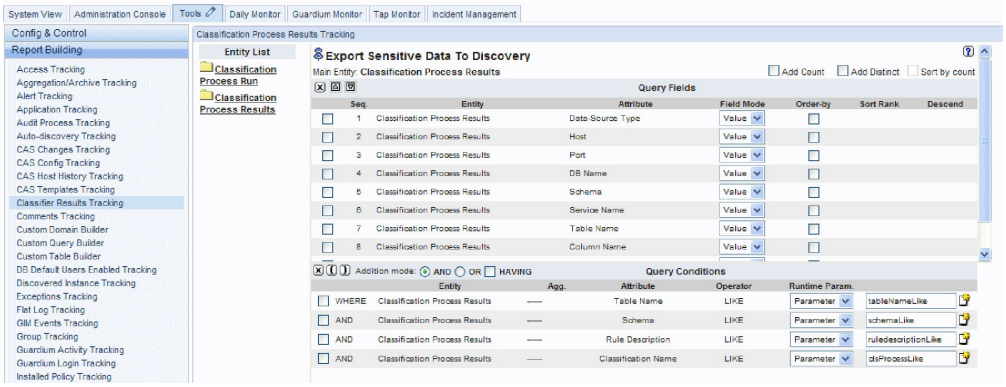
注: IBM Guardium では、分類プロセスは、定期的に実行される継続プロセスです。InfoSphere Discovery では、分類は、通常 1 回実行されるディスカバリー・プロセスの一部です。
注: このデータは CSV ファイルを介して転送されます。

エクスポート/インポート手順の概要を以下に示します。

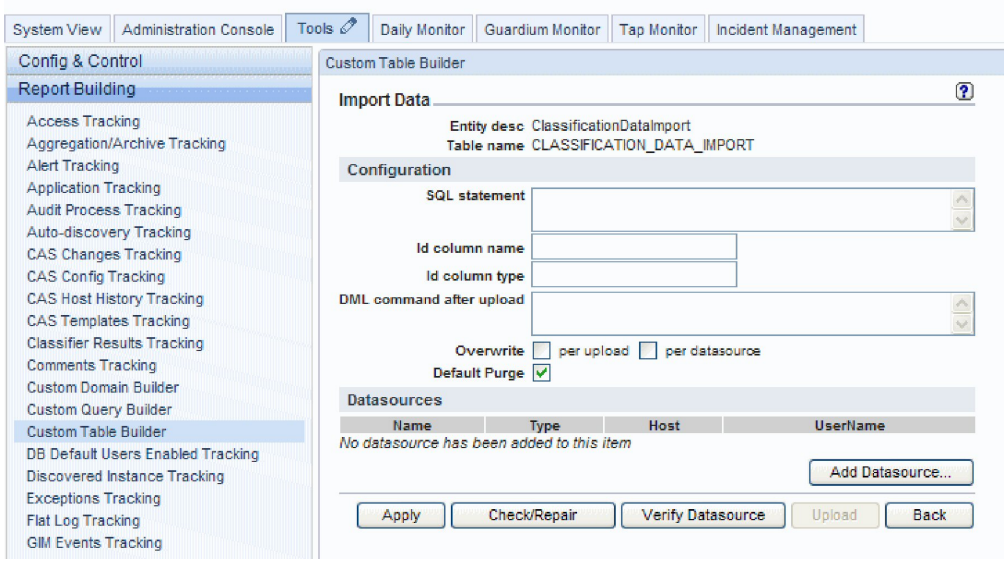
- Guardium からのエクスポート - 定義済みレポートを実行し (「Discovery への機密データのエクスポート」)、CSV ファイルとしてエクスポートします。
- Guardium へのインポート - CSV データ・ソースに対してカスタム表をロードします。このデータ・ソースに対してデフォルト・レポートを定義します。

以下の手順を行います。

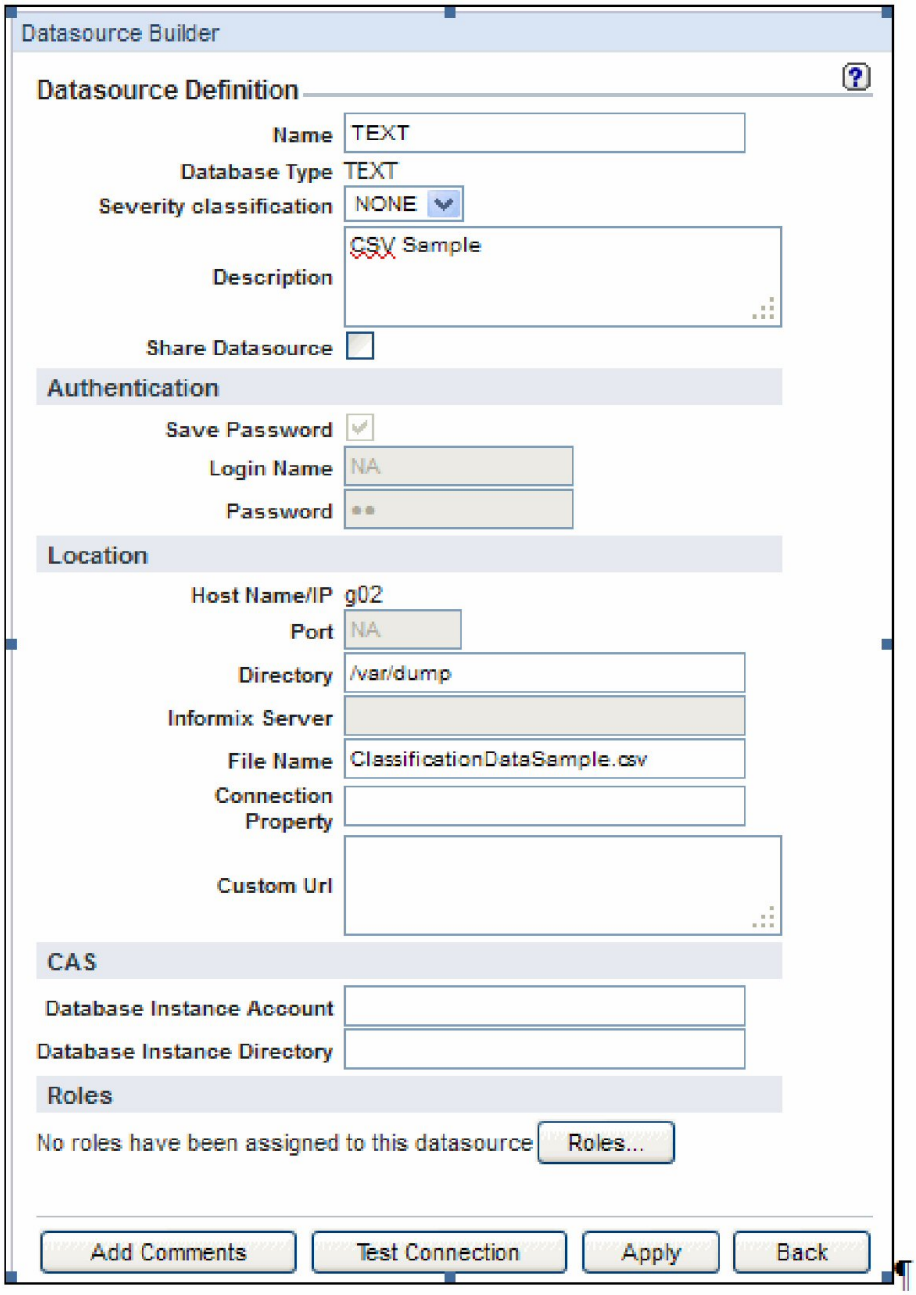
1. Guardium からのエクスポート - IBM Guardium から InfoSphere Discovery に分類データをエクスポートします。
2. Guardium® アプリケーションで admin ユーザーとして、「ツール」 > 「レポートのビルド」 > 「分類結果のトラッキング」 > 「レポートの選択」 > 「Discovery への機密データのエクスポート」に移動します (スクリーン・ショットを参照)。
   注: このレポートを UI ペインに追加します (これはデフォルトでは行われません)。



3. 「レポート結果」画面で「カスタマイズ」アイコンをクリックし、検索条件を指定して、Discovery に転送する分類結果データをフィルターに掛けます。
4. レポートを実行し、「レコードをすべてダウンロード」アイコンをクリックします。
5. CSV として保存し、このファイルを InfoSphere Discovery の指示に従い Discovery にインポートします。
6. Guardium にインポート- InfoSphere Discovery から IBM Guardium に分類データをインポートします。
7. InfoSphere Discovery の指示に基づき、InfoSphere Discovery から分類データを CSV としてエクスポートします。
8. Guardium アプリケーションで admin ユーザーとして、「ツール」 > 「レポートのビルド」 > 「カスタム表」画面に移動し、「分類データのインポート」を選択し、「データのアップロード」ボタンをクリックします。(スクリーン・ショットを参照)。



9. 「データのアップロード」画面で、「データ・ソースの追加」をクリックし、「新規作成」ボタンをクリックし、新規データ・ソースとして Discovery からインポートする CSV ファイルを定義します (「データベース・タイプ」=「テキスト」)。CSV データ・ソース定義の次のスクリーン・ショットを参照してください。



注: または、Discovery データベースおよび分類結果データにアクセスする方法が判明している場合、Discovery データベースからデータを直接ロードできます。

10. データ・ソースとして CSV を定義した後、「データ・ソース・リスト」画面で「追加」ボタンをクリックします。
11. 「データのアップロード」画面で、「データ・ソースの検査」、「適用」の順にクリックします。
12. 「今すぐ 1 回実行」ボタンをクリックして CSV からデータをロードします。
13. 「レポート・ビルダー」に移動し、「分類データのインポート」レポートを選択し、「ペインに追加」をクリックしてそのレポートをポータルに追加し、そのレポートに移動します。
14. レポートにアクセスし、「カスタマイズ」をクリックして開始日付/終了日付を設定し、レポートを実行します。

レポート結果には、InfoSphere Discovery からインポートされた分類データが含まれます。ダブルクリックして、このレポートに割り当てられている API を呼び出します。Discovery からインポートしたデータは以下の目的で使用できます。

- 結果セットに基づき新規データ・ソースを追加する。
- 機密データ・グループを追加/更新する。
- データ・ソースおよび機密データの詳細に基づきポリシー・ルールを追加する。
- プライバシー・セットを追加する。

表 1. CSV インターフェース・シグニチャー

インターフェース・シグニチャー	例
タイプ	DB2®
ホスト	9.148.99.99
ポート	50001

インターフェース・シグニチャー	例
dbName (Db2 または Oracle のスキーマ名、またはその他のデータベース名)	cis_schema
データ・ソースの URL	
表名	MK_SCHED
列名	ID_PIN
分類名	SSN
ルールの記述	InfoSphere Discovery のすぐに使用可能なアルゴリズム
HitRate	70% - Guardium バージョン 8.2 ではエクスポートで使用不可
使用しきい値	60% - Guardium バージョン 8.2 ではエクスポートで使用不可

親トピック: 製品の統合

CEF マッピング

ArcSight の CEF 標準は、一連の必須フィールドと、一連のオプション・フィールドを定義しています。

後者は CEF 標準では、拡張と呼ばれます。 データは、Guardium® 構成情報およびレポートからこれらのフィールドにマップされます。すべての Guardium フィールドが CEF フィールドにマップされるわけではないため、印刷レポートの行とそのレポートから作成した CEF ファイルの間では 1 対 1 の関係にならない可能性があることに注意してください。 またこの機能の意図としては、データ・アクセス・ドメイン (例えば、データ・アクセス、例外、ポリシー違反など) のデータをマップすることであって、Guardium 自己モニター・ドメイン (統合/アーカイブ、監査プロセス、Guardium ログインなど) のデータのマップではないことにも注意してください。

注: 分析済みのクライアント IP には CEF ソース用のマップがあります。CEF で使用される照会内にクライアント IP が指定されておらず、分析済みのクライアント IP が指定されている場合、分析済みのクライアント IP がソースに対して使用されます。クライアント IP と分析済みのクライアント IP の両方が照会内に指定されている場合、クライアント IP が優先されます。

下記の表に示す CEF フィールドは常に存在します。

表 1. 必須の CEF フィールドのマッピング

CEF フィールド	Guardium マッピング
バージョン	0 (ゼロ)。現在 CEF フォーマットの唯一のバージョン
Device Vendor	Guardium
Device Product	Guardium
Device Version	Guardium ソフトウェアのバージョン番号
Signature ID	ReportID
名前	レポート・タイトル
重大度	0 から 10 までの範囲の数値の重大度コード。10 が最重要なイベントです。 レポートで再設定されていなければ 0 (このゼロは、Guardium では情報に変換されます)。

CEF 拡張フィールドはオプションであり、マッピングが適用される場合にのみ存在します。 例えば、レポートにアクセス・ルールの記述が含まれていない場合、act フィールド (最初の拡張フィールド) は存在しません。 Guardium のエンティティーと属性について詳しくは、該当するエンティティー・リファレンスのトピックを参照してください。

表 2. CEF マッピング、Guardium バージョン 8.2

CEF フィールド	エンティティー	属性
severity	ポリシー・ルール違反	重大度
act	ポリシー・ルール違反	アクセス・ルールの記述
app	クライアント/サーバー	データベース・プロトコル
app	例外	データベース・プロトコル
dst	クライアント/サーバー	サーバー IP
dst	例外	宛先アドレス
dhost	クライアント/サーバー	サーバー・ホスト名
dpt	セッション	サーバー・ポート
dpt	例外	宛先ポート
dproc	クライアント/サーバー	ソース・プログラム
duid	クライアント/サーバー	OS ユーザー
duser	クライアント/サーバー	データベース・ユーザー名
duser	例外	ユーザー名
end	例外	例外タイム・スタンプ
end	ポリシー・ルール違反	タイム・スタンプ
end	アクセス期間	期間の終了

CEF フィールド	エンティティー	属性
end	セッション	セッション終了
msg	例外	例外の記述
msg	メッセージ・テキスト	メッセージ・テキスト
msg	メッセージ・テキスト	メッセージ件名
src	クライアント/サーバー	クライアント IP
src	クライアント/サーバー	分析されたクライアント IP
src	例外	ソース・アドレス
shost	クライアント/サーバー	クライアント・ホスト名
smac	クライアント/サーバー	クライアント MAC
spt	セッション	クライアント・ポート
spt	例外	ソース・ポート
start	例外	例外タイム・スタンプ
start	ポリシー・ルール違反	タイム・スタンプ
start	アクセス期間	期間の開始
start	セッション	セッション開始
proto	クライアント/サーバー	ネットワーク・プロトコル
request	完全な SQL	完全な SQL
request	SQL	SQL
cs1	セッション	Uid チェーン
cs2	セッション	Uid チェーン圧縮

表 3. CEF マッピング、Guardium バージョン 9.0

CEF フィールド	エンティティー	属性
severity	ポリシー・ルール違反	重大度
act	ポリシー・ルール違反	アクセス・ルールの記述
app	クライアント/サーバー	データベース・プロトコル
app	例外	データベース・プロトコル
dst	クライアント/サーバー	サーバー IP
dst	例外	宛先アドレス
dhost	クライアント/サーバー	サーバー・ホスト名
dpt	セッション	サーバー・ポート
dpt	例外	宛先ポート
dproc	クライアント/サーバー	ソース・プログラム
duid	クライアント/サーバー	OS ユーザー
duser	クライアント/サーバー	データベース・ユーザー名
duser	例外	ユーザー名
end	例外	例外タイム・スタンプ
end	ポリシー・ルール違反	タイム・スタンプ
end	アクセス期間	期間の終了
end	セッション	セッション終了
msg	例外	例外の記述
msg	メッセージ・テキスト	メッセージ・テキスト
msg	メッセージ・テキスト	メッセージ件名
src	クライアント/サーバー	クライアント IP
src	クライアント/サーバー	分析されたクライアント IP
src	例外	ソース・アドレス
shost	クライアント/サーバー	クライアント・ホスト名
smac	クライアント/サーバー	クライアント MAC
spt	セッション	クライアント・ポート
spt	例外	ソース・ポート
start	例外	例外タイム・スタンプ

CEF フィールド	エンティティー	属性
start	ポリシー・ルール違反	タイム・スタンプ
start	アクセス期間	期間の開始
start	セッション	セッション開始
proto	クライアント/サーバー	ネットワーク・プロトコル
request	完全な SQL	完全な SQL
request	SQL	SQL
cs1	セッション	Uid チェーン
cs2	セッション	Uid チェーン圧縮

CEF に関する詳細については、Web で Common Event Format: Event Interoperability Standard を検索するか、ArcSight の Web サイト www.arcsight.com にアクセスしてください。

親トピック: 製品の統合

LEEF マッピング

QRadar からの Log Event Extended Format (LEEF)

LEEF フォーマットは、オプションの syslog ヘッダー、LEEF ヘッダー、およびそのイベントについて記述した属性のコレクションで構成されます。

Syslog_Header (オプション) LEEF_Header|Event_Attributes

LEEF ヘッダーは、パイプ (「|」) で区切られ、属性はタブで区切られます。

例

Jan 18 11:07:53 host LEEF:Version|Vendor|Product|Version|EventID|Key1=Value1<tab>Key2=Value2<tab>Key3=Value3<tab>...<tab>KeyN=ValueN

表 1. LEEF パラメーター

パラメーター	記述
LEEF: バージョン	そのログ・メッセージに使用された LEEF のバージョンを識別する、バージョンの整数。
ベンダー	イベント・ログを送信したデバイスまたはアプリケーションのベンダーを識別するストリング。
製品	そのイベント・ログを送信した製品を識別する製品ストリング。注: ベンダーと製品の組み合わせは固有のものでなければなりません。
バージョン	イベント・ログを送信したデバイスまたはアプリケーションのバージョンを識別するストリング。
イベント ID	イベントを一意的に識別する ID。
属性 1..N	タブ文字で区切られた、イベントのキー値ペア属性のセット。順序は強制されません。 事前定義のキーのセットを定義し、使用できるときに使用する必要があります。 LEEF フォーマットは拡張可能です。また、イベント・ログに追加のキー値ペアを追加することができます。 キーにスペースまたは等号を含めることはできません。 値にタブを含めることはできません。

例:

```
Jan 18 11:07:53 192.168.1.1 LEEF:1.0|QRadar|QRM|1.0|NEW_PORT_DISCOVERD|src=172.5.6.67 dst=172.50.123.1 sev=5 cat=anomaly msg=there are spaces in this message
```

文字エンコード

UTF8

定義済みの属性

表 2. 定義済みの属性

キー名	データ・タイプ	最大長	記述
Cat	ストリング		イベント・カテゴリー
devTime	日付		デバイスまたはアプリケーションがイベントを発行した時間
devTimeFormat	ストリング		Java SimpleDateFormat によって定義されます。これは、カスタマイズした日付形式を使用している場合のみ必須です。詳しくは、日付形式のセクションを参照してください。
proto	整数		トランスポート・プロトコル
sev	整数 (1 から 10)		このイベントの重大度
src	IPv4 または IPv6 アドレス		ソース・アドレス

キー名	データ・タイプ	最大長	記述
dst	IPv4 または IPv6 アドレス		宛先アドレス
VSrc	IPv4 または IPv6 アドレス		バーチャル・ソース・アドレス
srcPort	整数		ソース・ポート。有効なポート番号は 0 から 65535 までです。
dstPort	整数		宛先ポート。有効なポート番号は 0 から 65535 までです。
srcPreNat	IPv4 または IPv6 アドレス		ネットワーク・アドレス変換 (NAT) が発生する前のメッセージのソース・アドレス。
dstPreNat	IPv4 または IPv6 アドレス		ネットワーク・アドレス変換 (NAT) が発生する前のメッセージの宛先アドレス。
srcPostNat	IPv4 または IPv6 アドレス		ネットワーク・アドレス変換 (NAT) が発生した後のメッセージのソース・アドレス。
dstPostNat	IPv4 または IPv6 アドレス		ネットワーク・アドレス変換 (NAT) が発生した後のメッセージの宛先アドレス。
usrName	ストリング	255	イベントに関連付けられたユーザー名。
srcMAC	MAC アドレス		コロンで区切られた 6 つの 16 進数。 例: `1:2D:67:BF:1A:71`
dstMAC	MAC アドレス		コロンで区切られた 6 つの 16 進数。 例: `11:2D:67:BF:1A:71`
srcPreNATPort	整数		ソース・ポート。有効なポート番号は 0 から 65535 までです。
dstPreNATPort	整数		宛先ポート。有効なポート番号は 0 から 65535 までです。
srcPostNATPort	整数		ソース・ポート。有効なポート番号は 0 から 65535 までです。
dstPostNATPort	整数		宛先ポート。有効なポート番号は 0 から 65535 までです。
identSRC	IPv4 または IPv6 アドレス		
identHostName	ストリング	255	イベントに関連したホスト名。通常、このパラメーターは、ID イベントにのみ関連します。
identNetBios	ストリング	255	イベントに関連した NetBIOS 名。通常、このパラメーターは、ID イベントにのみ関連します。
identGrpName	ストリング	255	レコードに関連したイベント名。通常、このパラメーターは、ID イベントにのみ関連します。

カスタム属性

一部のケースでは、生成中のイベントに関する詳細を識別するために、カスタム属性が必要になる可能性があります。これらのケースでは、ベンダーが独自のカスタム属性を定義し、それらのカスタム属性をイベント・ログに組み込む場合があります。カスタム属性フィールドは、定義済みのフィールドへの受け入れ可能なマッピングが存在しない場合にのみ使用してください。

カスタム属性キーは以下のようになっている必要があります。

- スペースのない単一ワード
- 英数字
- 明快かつ簡潔
- 定義済みの属性キーと同じ名前を付けることはできない

カスタム属性は、カスタム・プロパティーを作成することによって、QRadar イベント・ビューアーでの表示に使用される可能性があります。

カスタム属性は、顧客プロパティーを作成することで、QRadar レポート・エンジンによって使用される可能性があります。

カスタム属性をイベント相関に使用することはできません。

注: MS-SQL データベース名を取り込むには、databaseName=%%DBname を LEEF テンプレートに追加します。既存の LEEF テンプレートを更新するか、クローン作成によって新規テンプレートを作成します。

日付形式

以下の事前定義形式のいずれかを使用できます。

1. 1970 年 1 月 1 日からのミリ秒 (整数)
2. MMM dd yyyy HH:mm:ss (例えば Jun 06 2012 16:07:36)
3. MMM dd yyyy HH:mm:ss.SSS (例えば Jun 06 2012 16:07:36.300)
4. MMM dd yyyy HH:mm:ss.SSS zzz (例えば Jun 06 2012 02:07:36.300 GMT)

これらの形式が適さない場合は、dTimeFormat キーを使用して日付形式を指定することで、dTime フィールドでカスタム日付形式を定義することができます。

日付形式の指定について詳しくは、SimpleDateFormat のページ (http://java.sun.com/javase/6/docs/api/java/text/SimpleDateFormat.html) を参照してください。

親トピック: 製品の統合

Guardium アプリケーション

Guardium アプリケーションを使用して、新しいデータとすぐに使用できるユース・ケースにより、現在の Guardium デプロイメントを拡張および強化します。

Guardium エコシステムは、IBM Security App Exchange、アプリケーション、SDK、および Guardium UI の「アプリケーション・ライフサイクル」で構成されます。 アプリケーションは、エコシステムの中核です。

Guardium アプリケーションは、現在の Guardium システムを、新しいデータおよび機能により拡張および強化します。IBM、そのビジネス・パートナー、およびその他の Guardium ユーザーが作成した他の共有アプリケーションを IBM Security App Exchange ポータルからダウンロードし、インストールできます。また、Guardium SDK を使用して、独自のアプリケーションを作成することもできます。この SDK は、IBM X-Force Exchange で入手できます。アプリケーションをパッケージして、他の Guardium デプロイメントで再使用できます。また、独自のアプリケーションを IBM Security App Exchange ポータルで共有できます。

Guardium アプリケーションは、Python Flask を Web サーバーとして使用し、Guardium システムで自己完結型コンテナーとして実行されます。

公開アプリケーションのダウンロード

すべてのアプリケーションおよびセキュリティー製品の機能拡張は、IBM X-Force Exchange ポータルでホストされます。

IBM Security App Exchange で入手可能なアプリケーションのリストを表示できます。「アプリケーション」チェック・ボックスを選択することで、アプリケーションをフィルターに掛けます。

Guardium アプリケーションは、アーカイブ (.zip) ファイルとしてパッケージ化されており、Guardium にデプロイできます。Guardium UI でのアプリケーションの操作を参照してください。

- Guardium アプリケーション開発の概要
  Guardium Application Framework を使用して、Guardium Web UI と機能を統合する新規アプリケーション・モジュールを開発します。
- SDK の処理
- Guardium UI でのアプリケーションの操作
  アプリケーションのアップロード方法、管理方法、および出力の表示方法について説明します。
- アプリケーションに関する FAQ
- リソース
  IBM GuardiumGUI Application Framework でのアプリケーション作成を支援する各種リソースを使用します。

Guardium アプリケーション開発の概要

Guardium Application Framework を使用して、Guardium Web UI と機能を統合する新規アプリケーション・モジュールを開発します。

アプリケーション (アプリ) は、Application Framework への小さいプラグイン・モジュールです。アプリケーションは、セキュア・コンテナーの内部からエンドポイントを提供して、Guardium Web インターフェース内にコンテンツを直接注入します。

各アプリケーションには、独自の専用メモリーおよび定義された量の CPU リソースが割り振られます。

アプリケーションを作成するために使用される主要な Web 言語は Python です。Flask フレームワークが統合され、アプリケーションはこのフレームワークを使用できます。

アプリケーションの実行および Guardium との対話の方法

Guardium アプリケーションは、Guardium ユーザー・インターフェースから独立している、分離された Python Flask 環境の内部で実行されます。

アプリケーションは、静的イメージ、スクリプト、および HTML ページも使用できます。

アプリケーションとのすべての対話は、Guardium ユーザー・インターフェースを通じてプロキシー処理されます。ネットワーク・ポートおよび Web サービスへの直接アクセスは、通常、許可されません。

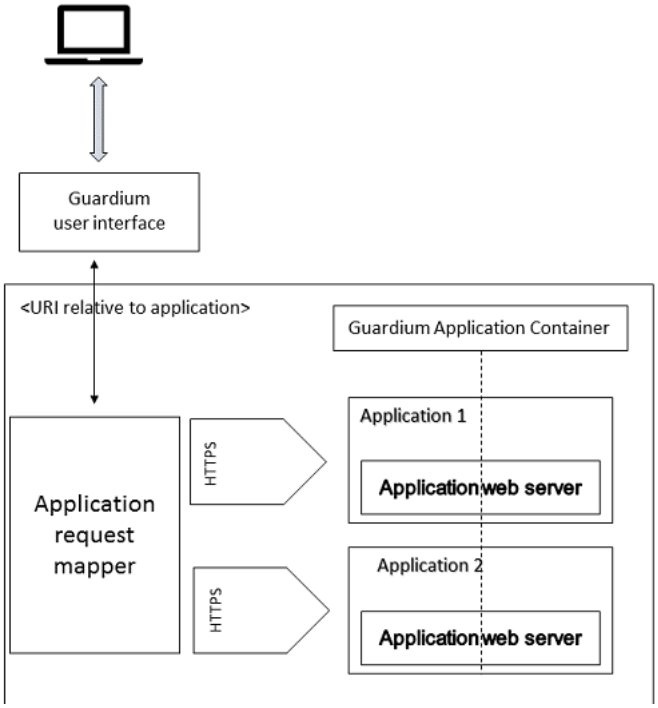


図 1. Application Framework

親トピック: Guardium アプリケーション

関連概念:

Python ヘルパー・ライブラリー関数

GrdAPI クラスと GdrConnection クラス

SDK の処理

- SDK の前提条件および制限事項
  SDK をインストールする前に、これらの前提条件を確認して満たしてください。制限事項について理解しておいてください。
- SDK のインストール
  Guardium Application Framework SDK は、Windows、Linux、または macOS オペレーティング・システム上にインストールできます。
- SDK のアンインストール
  開発者システムから SDK をアンインストールするには、以下の手順を使用します。
- SDK 環境およびワークスペースのアップグレード
  SDK 環境およびワークスペース (既存のアプリケーション) をアップグレードする方法を説明します。
- アプリケーションの作成、実行、パッケージ化、およびデプロイ
  Guardium Application Framework には、独自の Software Development Kit (SDK) が付属しています。
- アプリケーション・ファイル構造
  作成した Guardium アプリケーションは、圧縮ファイル内で配布されます。
- GUI Application Framework の基礎
  Guardium GUI Application Framework アプリケーションは、Flask マイクロフレームワークで実行され、Flask Web サーバーから提供されるスタンドアロンの Web アプリケーションです。
- サンプル・アプリケーション
  SDK には、いくつかのサンプル・アプリケーションが含まれています。
- サポート関数
  Guardium GUI Application Framework には、いくつかの組み込み経路、カスタム Jinja2 Flask 関数、およびアプリケーション開発をサポートするその他のヘルパー・ユーティリティーが付属しています。
- Python ヘルパー・ライブラリー関数
  Guardium Python ヘルパー・ライブラリー (gpylib) には、ロギングを追加し、REST API 呼び出しを行い、JSON オブジェクトを Python ディクショナリーに変換するために使用できるいくつかの役立つ関数が含まれています。
- Jinja2 テンプレート
  Jinja2 は Python ライブラリーです。これを使用すると、コア・テンプレート・テキスト・ファイルから各種出力フォーマットのテンプレートを作成できます。Jinja2 は、Guardium アプリケーション用の HTML テンプレートを作成するために使用できます。
- アプリケーションのログ
  アプリケーションのログは、アプリケーションのコンテナーの /store/log ディレクトリーに保管されます。

親トピック: Guardium アプリケーション

SDK の前提条件および制限事項

SDK をインストールする前に、これらの前提条件を確認して満たしてください。制限事項について理解しておいてください。

SDK のインストールと動作をサポートするオペレーティング・システムのバージョンは、以下のとおりです。

- Windows 7
- Ubuntu 17.10
- CentOS 7.4

- RHEL 7
- macOS 10.11、Sierra 10.12.6

ハードウェア要件:

- アプリケーションをアップロードして実行するためのアプライアンス: 最小 16 GB のメモリー
- 開発者用マシン
  - 調査ダッシュボードが無効である場合、最小 24 GB
  - 調査ダッシュボードが有効である場合、最小 34 GB

SDK をインストールする前に以下を行ってください。

- 最新の Python 2 (バージョン 2.7.9 以上) 64 ビットをインストールします。
  注: Python 3 および Python 32 ビットはサポートされません。

  ご使用のオペレーティング・システムにインストールされている Python のバージョンを確認するには、python `--version` コマンドを使用します。Windows では、インストール時にデフォルトのインストール・パス (c:¥Python27¥) を受け入れてください。他のパスの場合、Python パッケージ・マネージャー (pip) の適切なインストールを妨げるバグが Python に存在する可能性があるためです。

  Python Downloads から Python をダウンロードできます。

- アプリケーションを Guardium システムに送信する前に、ローカルでテストする場合は、Docker CE をインストールします。
  重要: デプロイメント前に Docker をインストールしてローカルでアプリケーションをテストすることを強くお勧めします。

制限事項

- デプロイされたアプリケーションは、Guardium のバックアップとリストアの一部ではありません。アプリケーションをバックアップするには、ハイパーバイザー・バックアップ (VM スナップショット) を使用してください。

- Linux への Python 2.7.9 以降のインストール
  Linux オペレーティング・システムで Guardium Application Framework SDK を実行するには、Python 2.7.9 以降 (ただし、Python 3 より前) をインストールします。
- macOS への Python 2.x のインストール
  macOS オペレーティング・システムで Guardium Application Framework SDK を実行するには、Python 2.7.9 以降 (ただし、Python 3 より前) をインストールする必要があります。
- Windows への Python 2.7.9 のインストール

親トピック: SDK の処理
関連タスク:
SDK のインストール

Linux への Python 2.7.9 以降のインストール

Linux オペレーティング・システムで Guardium Application Framework SDK を実行するには、Python 2.7.9 以降 (ただし、Python 3 より前) をインストールします。

このタスクについて

手順

1. `root` としてログインするか、`sudo` を使用します。
2. `yum update` を実行して、システムが最新の状態であることを確認します。
3. `yum groupinstall -y "development tools"` を実行して、開発ツールをインストールします。
4. 以下を実行して、開発ツールおよび追加のライブラリーをインストールします (厳密には Python のコンパイルには必要ありませんが、Python インタープリターには必須です)。
   - Ubuntu: `yum install -y zlib-devel bzip2-devel openssl-devel ncurses-devel sqlite-devel readline-devel tk-devel gdbm-devel db4-devel libpcap-devel xz-devel expat-devel`
   - CentOS: `yum install -y zlib-devel, openssl-devel, openssl, libffi-devel`
5. CentOS の「最小限」のクリーン・インストール環境の場合、`yum install -y wget` を実行して wget ツールをインストールします。
6. Python をダウンロードし、コンパイルしてインストールします。

   ```
   # Python 2.7.0:
   wget http://python.org/ftp/python/2.7.0/Python-2.7.0.tar.xz
   tar xf Python-2.7.0.tar.xz
   cd Python-2.7.0
   ./configure --prefix=/usr/local --enable-unicode=ucs4 --enable-shared LDFLAGS="-Wl,-rpath /usr/local/lib"
   make && make altinstall
   ```

   新しくインストールされた Python インタープリターは /usr/local/bin/python2.7 として使用可能です。Python 2.6.6 のシステム・バージョンは、引き続き、/usr/bin/python、/usr/bin/python2、および /usr/bin/python2.6 として使用可能です。
7. 共有ライブラリーからシンボルを削除してメモリー占有スペースを削減するには、`strip /usr/local/lib/libpython2.7.so.1.0` と入力します。
8. Python コマンドが、インストールした最新の Python を指していることを確認してください。次に、最新の Python インストールをデフォルトの Python エンジンにするために、`source ~/.bash_profile` を実行します。

親トピック: SDK の前提条件および制限事項

macOS への Python 2.x のインストール

macOS オペレーティング・システムで Guardium Application Framework SDK を実行するには、Python 2.7.9 以降 (ただし、Python 3 より前) をインストールする必要があります。

始める前に

Mac にログインし、コマンド `$ xcode-select --install` を実行します。

このタスクについて

macOS には、通常、Python 2.7.x が付属しています。Python の macOS バージョンは使用しないでください。代わりに、例えば Homebrew パッケージ・マネージャーを使用して、最新の Python 2 バージョンをインストールします。これにより、既存の Python に加えて、代替の Python がインストールされます。

Python コマンドが、インストールした最新の Python を指していることを確認してください。

手順

1. Homebrew パッケージ・マネージャーを使用して Python 2 をインストールするには、以下のようにします。
   a. 次のコマンドを使用して、Homebrew をインストールします。

      ```
      ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
      ```

   b. `brew install python@2` と入力して、Python 2 をインストールします。
   c. Python コマンドが、Homebrew を使用してインストールした最新の Python を指していることを確認してください。

      ```
      ls -l `which python`
      ```
      以下のような出力が表示されます。
      ```
      /usr/local/bin/python -> ../Cellar/python@2/2.7.14_3/bin/python
      ```

   d. `python --version` を入力して、インストールした Python バージョンが実行されていることを確認します。

      システムは、バージョン番号 2.7.9 以降で応答します。

2. あるいは、pyenv を使用して Python をインストールすることもできます。
   a. `pip install pyenv` を入力して、pyenv をインストールします。
   b. `pyenv install <version>` を入力して、Python をインストールします (例: `pyenv install 2.7.9`)。
   c. `pyenv versions` を入力して、該当の Python バージョンがインストールされたことを確認します。
   d. `pyenv global 2.7.9` を入力して、Python をローカルで使用します。
   e. `python --version` を入力して、インストールした Python バージョンが実行されていることを確認します。

親トピック: SDK の前提条件および制限事項

Windows への Python 2.7.9 のインストール

このタスクについて

Windows オペレーティング・システムで Guardium Application Framework SDK を実行するには、Python 2.7.9 以降 (ただし、Python 3 より前) をインストールする必要があります。

手順

1. Python ダウンロード (Windows 版) から最新の Python 2 (64 ビット) インストーラー (通常は「Windows x86-64 MSI installer」という名前) をダウンロードします。SDK では、32 ビットの Python インタープリターはサポートされていません。
2. 次のようにして、Python 2 をインストールします。
   a. インストール・パス: スペースなしのパスにインストールします。スペースがあると (例: C:¥Program Files¥Python27)、Python インストーラーは Scripts フォルダーをインストールしません。
   b. インストール対象: スクロールダウンして、環境変数をインストールするようにチェック・マークを付けます。これにより、Python コマンドが確実に認識されます。
3. コマンド・プロンプトを開いて `python` を入力することで、Python が正常にインストールされたことを確認します。 このコマンドの出力には、Python バージョンが 2.7.9 以上であり、そのバージョンが 64 ビット対応であることが示されます。例:

   ```
   Python 2.7.9 (v2.7.9:84471935ed, Sep 16 2017, 20:25:58) [MSC v.1500 64 bit (AMD64)] on win32
   Type "help", "copyright", "credits" or "license" for more information.
   >>
   ```

4. Ctrl+ Z を押して、Python インタープリターを終了します。

親トピック: SDK の前提条件および制限事項

SDK のインストール

Guardium Application Framework SDK は、Windows、Linux、または macOS オペレーティング・システム上にインストールできます。

始める前に

SDK の前提条件および制限事項をすべて確認します。Docker コンテナーでアプリケーションを実行するには、SDK をインストールする前に、Docker をインストールして、そのサービスを開始する必要があります。

手順

1. Security App Exchange からアプリケーションをダウンロードして、Guardium Application Framework SDK アーカイブ (.zip ファイル) を解凍します。

2. 次のようにして、SDK をインストールします。
   - Docker がインストールされていない Windows の場合:
     - 管理者として `install.bat` を右クリックしてインストール・プロセスを実行します。SDK は、c:¥GuardiumAppSDK フォルダーにインストールされます。端末ウィンドウを閉じてから再オープンすると、`grd_sdk` コマンドが使用可能になります。
   - Docker をインストール済みの Windows の場合:
     a. ここで、Docker Toolbox を開発マシンにインストールします。Install Docker Toolbox on Windows のプロセスを参照してください。
     b. Docker を開始します。
     c. SDK zip アーカイブのコンテンツを解凍します。
     d. Docker Quickstart Terminal を開き、*./install.bat* を入力して、install.bat スクリプトを管理者として実行します。SDK のインストール中に、新しい Docker がローカルの Docker リポジトリーに追加されます。
   - Linux、Unix、macOS の場合:
     a. SDK zip アーカイブのコンテンツを解凍します。
     b. SDK フォルダー内から、次のように、install.sh スクリプトを root として実行します。 `./install.sh.`
     権限の問題がある場合は、`sudo ./install.sh` を使用してインストールします。

次のタスク

Guardium アプリケーションの作成
親トピック: SDK の処理

SDK のアンインストール

開発者システムから SDK をアンインストールするには、以下の手順を使用します。

手順

1. Linux または Mac サーバーの場合は、フォルダー /usr/local/etc/GuardiumAppSDK を削除します。
2. Windows サーバーの場合は、フォルダー c:¥GuardiumAppSDK を削除します。

親トピック: SDK の処理

SDK 環境およびワークスペースのアップグレード

SDK 環境およびワークスペース (既存のアプリケーション) をアップグレードする方法を説明します。

このタスクについて

注: /store 内のデータはアプリケーション・アップグレード中に保護されるため、ベスト・プラクティスとして、このディレクトリー内にアプリケーション構成およびデータを保管してください。
重要: 多くのメモリーを使用するインストール済みアプリケーションをアップグレードする場合、メモリー・リソースの問題を回避するために、アップグレード実行前にアプリケーションを停止し、アップグレードが完了した後、そのアプリケーションを再始動しなければならない場合があります。
SDK のアップグレードでは、以下が行われます。

- virtualenv (grd_appfw_venv) が上書きされます。
- grd_appfw_venv を除き、SDK フォルダーに追加されたファイルは削除されません。
- アップグレード前のバージョンで変更されたすべての Guardium SDK ファイルが上書きされます。
- ファイル run.py、index.html、views.py、APP_CUSTOM_REQUIREMENTS.txt は変更されません。
- SDK_version を除き、マニフェストで追加/変更されたすべての値が保存されます。

手順

1. 開発マシンにログインして、grd_sdk upgrade -w <workspace> を実行します。 システム応答は次のようになります。このコマンドでは、ファイルおよびディレクトリーが削除または変更される可能性があります。続行しますか? `(This command might remove or change files and directories. Are you sure you want to continue?) [y/N]`
2. `y` を入力します。
   アップグレードの終了時に、システムから、「`upgrade_app` 呼び出しが完了しました `(upgrade_app call completed)`」という応答が返されます。
   アップグレードが失敗した場合、システムは「ファイルまたはフォルダーの更新に失敗しました... `(Failed to update file or folder...)`」で始まるメッセージで応答し、ファイルおよびフォルダーはどれも変更されません。
3. 開発マシンでアプリケーションを再始動します。
4. Guardium システムでアプリケーションを更新する方法については、アプリケーションのアップロードおよび管理を参照してください。

親トピック: SDK の処理

アプリケーションの作成、実行、パッケージ化、およびデプロイ

Guardium Application Framework には、独自の Software Development Kit (SDK) が付属しています。

Guardium Application Framework Software Development Kit (SDK) を使用して、以下のアプリケーション開発タスクを実行します。

開発ワークスペースの作成

Guardium Application Framework SDK は、独自のアプリケーションを作成するためのテンプレートとして使用するサンプル・アプリケーションを含む開発ワークスペースをインストールします。

テスト目的でのアプリケーションのローカル実行

アプリケーションをテストするためにライブ Guardium システムにアプリケーション・コードをアップロードする必要はありません。Guardium Application Framework SDK には、アプリケーションをローカルに実行するために使用できる仮想開発環境が含まれています。

アプリケーションが Guardium API エンドポイントを使用する場合は、Guardiumシステム上の API に接続し、ローカルにテストするように、仮想環境を構成できます。Guardiumにアップロードする必要はありません。

アプリケーションのパッケージ化

Guardium Application Framework SDK には、アプリケーション・ファイルを含むアーカイブ (.zip ファイル) を作成するために使用するパッケージ化ユーティリティーが含まれています。

Guardiumへのアプリケーションのデプロイ

Guardium Application Framework SDK には、パッケージされたアプリケーションをライブ Guardium システムに直接アップロードするために使用できるデプロイメント・ユーティリティーが含まれています。

- Guardium アプリケーションの作成
  Guardium GUI Application Framework SDK を使用して、アプリケーションの基本開発環境を作成します。SDK に付属のいずれかのサンプル・テンプレート・アプリケーションで構築し、独自のアプリケーションを作成することもできます。
- アプリケーションのローカル実行
  アプリケーションをパッケージ化してデプロイする前に、ブラウザー・ウィンドウでアプリケーションをローカルに実行してテストします。
- アプリケーションをコンテナーでローカルに実行する (Windows)
  Windows サーバー上でローカル・テスト用のコンテナーを使用して、実稼働環境に類似した環境でアプリケーションを実行します。SDK は、Guardium 実稼働環境で稼働するベース・イメージと同一のベース・イメージを作成します。
- アプリケーションをコンテナーでローカルに実行する (Linux)
  Linux サーバー上でローカル・テスト用のコンテナーを使用して、実稼働環境に類似した環境でアプリケーションを実行します。SDK は、Guardium 実稼働環境で稼働するベース・イメージと同一のベース・イメージを作成します。
- アプリケーションのパッケージ化、デプロイ、および実行
  Guardium GUI Application Framework を使用して、アプリケーションをアーカイブ (.zip ファイル) としてパッケージし、それを Guardium テスト環境にデプロイして、アプリケーションおよびそのログを表示します。
- アプリケーションを作成するためのチュートリアル
  このチュートリアルでは、Guardium SDK を使用してアプリケーションを作成し、そのアプリケーションをワークステーションで実行してテストし、最後に必要に応じて Guardium システムにデプロイする方法を説明します。
- Eclipse でのアプリケーションの開発
  開発環境をセットアップした後で、その開発環境を Eclipse 内にインポートします。その Eclipse 統合開発環境 (IDE) 機能を使用して、アプリケーションを開発します。

親トピック: SDK の処理

Guardium アプリケーションの作成

Guardium GUI Application Framework SDK を使用して、アプリケーションの基本開発環境を作成します。SDK に付属のいずれかのサンプル・テンプレート・アプリケーションで構築し、独自のアプリケーションを作成することもできます。

手順

コマンド `grd_sdk create -w <path to myapp>` を入力することで、コンピューター上に Guardium サンプル・アプリケーション用のフォルダーを作成します。ここで、<path to myapp> はアプリケーション用に作成するローカル・フォルダーのパスです (例: `grd_sdk create -w ./myApp`)。
注意:
SDK ディレクトリーの下にワークスペースを作成しないでください。SDK の下にあるディレクトリーは、SDK のアップグレード時に削除されます。
注: パス内のフォルダー名およびファイル名にスペースを使用することはできません。
注: Linux では、アプリケーションへの絶対パスを短くしてください。BINPRM_BUF_SIZE カーネル定数 (カーネル・バージョンに応じて 79 文字または 127 文字) が原因でファイル・パスが切り捨てられることがあり、その結果、このコマンドが失敗します。

タスクの結果

開発環境スクリプトを実行すると、以下のフォルダーおよびファイルがアプリケーション開発フォルダーに追加されます。

表 1. 開発フォルダー内の Guardium アプリケーション・ファイルおよびフォルダー

ファイル/フォルダー	記述
app	アプリケーション・ファイルのルート・ディレクトリー。以下のファイルが含まれています。 views.py ファイル。Web アプリケーションへのメイン・エントリー・ポイント。すべてのアプリケーションで必要になるファイルは、このファイルおよび manifest.json ファイルのみです。このファイルには、「Hello World」アプリケーションのサンプル・コードが含まれています。 gpylib フォルダー。アプリケーションが Guardium 操作を実行するために使用する Python ライブラリー・ファイルが含まれています。gpylib ライブラリーを使用すると、API エンドポイントへの接続やストレージ・パスの取得などの操作を行うことができます。
grd_appf_venv	アプリケーションをローカルで実行するための Python 仮想環境が含まれています。
manifest.json	サンプル「Hello World」アプリケーションの実行内容を記述します。
src_deps	src_deps フォルダー。アプリケーションによって使用される Python の従属関係のリストが含まれています。

ファイル/フォルダー	記述
grdlib	grdlib フォルダー。アプリケーションによって使用される Python SDK grd api ライブラリーが含まれています。例えば、Guardium レポートを作成するために、views.py の grdapi ライブラリーから関数を呼び出すことができます。
store	store フォルダー。アプリケーション・ストレージ・ファイルおよびアプリケーション・ログ・ファイルが含まれています。
run.py	アプリケーションをローカルで実行するためのデフォルト Python スクリプト。

次のタスク

これで、アプリケーションのコーディングを開始する準備ができました。アプリケーションおよびマニフェスト・ファイルの構造におけるソースの依存関係を十分に理解してください。
親トピック: アプリケーションの作成、実行、パッケージ化、およびデプロイ
関連概念:
アプリケーション・ファイル構造
サンプル・アプリケーション
関連タスク:
アプリケーションのローカル実行

アプリケーションのローカル実行

アプリケーションをパッケージ化してデプロイする前に、ブラウザー・ウィンドウでアプリケーションをローカルに実行してテストします。

このタスクについて

Guardium GUI Application Framework には、テスト目的でアプリケーションをローカルに実行するために使用できる仮想環境が含まれています。

手順

1. コマンド `grd_sdk run -w <path to myapp>` を入力します。 アプリケーションの初回実行時に、Guardium アプライアンスのホスト名または IP、CLI ユーザー・パスワード、および Guardium ユーザーの資格情報を入力するよう求めるプロンプトが出されます。これらの情報を使用して、そのコンピューター専用の認証トークンが作成されます。資格情報は、アプリケーション・フォルダー内の暗号化されたファイル (.guard_config) に保存されます。アプリケーションが通信する Guardium マシンを変更する必要がある場合、または使用する資格情報を更新する場合は、grd_sdk regenerate_tokens を実行して新しい認証トークンを生成します。
2. ブラウザーを開き、アドレス・バーに `http://localhost:5000` と入力します。アプリケーションに REST エンドポイントがある場合は、この URL でそれらのエンドポイントを呼び出せます。 アプリケーションがブラウザー・ウィンドウ内に表示され、アプリケーション出力がコマンド・ラインまたは端末に送信されます。
3. アプリケーションの更新を実行している場合、または別の Guardium システムからアプリケーションをコピーした後で開始しない場合は、以下の手順を試してください。
4. アプリケーションを停止するには、macOS の場合は Ctrl+C、Windows の場合は Ctrl+C Ctrl+C (2 回) を押します。

親トピック: アプリケーションの作成、実行、パッケージ化、およびデプロイ

アプリケーションをコンテナーでローカルに実行する (Windows)

Windows サーバー上でローカル・テスト用のコンテナーを使用して、実稼働環境に類似した環境でアプリケーションを実行します。SDK は、Guardium 実稼働環境で稼働するベース・イメージと同一のベース・イメージを作成します。

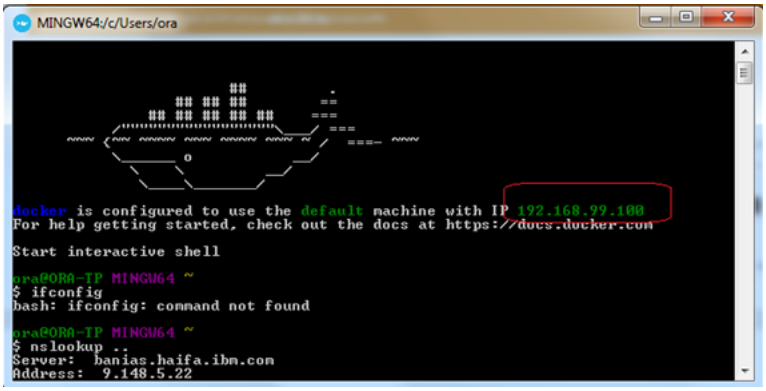
始める前に

Docker コンテナーでアプリケーションを実行するには、SDK をインストールする前に、Docker をインストールして、そのサービスを開始する必要があります。SDK のインストールを参照してください。

Docker が実行されていることを確認します。

手順

1. アプリケーションをローカルに作成して実行した後 (アプリケーションのローカル実行)、Docker Quickstart Terminal を開きます。
2. コマンド `grd_sdk.bat run -d -w <path to app>` を Docker Quickstart Terminal に入力することで、Docker CentOS コンテナーを使用してアプリケーションをローカルに実行します。 ワークスペースのパスを Linux の bash 形式で (`../My Documents/app`)、または Windows 形式で (ただし、`"C:¥My Documents¥app"` のように引用符で囲んで) 記述します。 認証の詳細情報を求めるプロンプトがシステムから出されます。
3. 管理 GUI ユーザーの CLI パスワードおよび資格情報を使用して、Guardium アプライアンス IP を入力します。 ターミナルは、`Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)` で応答します。
4. ブラウザーでアプリケーション URL を開き、サンプル・アプリケーションの出力「Hello World!」を確認します。URL では、コンテナーの Quick Start Terminal の先頭行に (ターミナルの開始時に) 表示される、コンテナー・インスタンスの内部 IP アドレスを使用します。以下の例では、内部 IP は 192.168.99.100 です。アプ



リケーション URL は http://192.168.99.100:5000 になります。

5. アプリケーションを停止するには、Ctrl+C Ctrl+C (2 回) を押します。

親トピック: アプリケーションの作成、実行、パッケージ化、およびデプロイ

アプリケーションをコンテナーでローカルに実行する (Linux)

Linux サーバー上でローカル・テスト用のコンテナーを使用して、実稼働環境に類似した環境でアプリケーションを実行します。SDK は、Guardium 実稼働環境で稼働するベース・イメージと同一のベース・イメージを作成します。

始める前に

Docker コンテナーでアプリケーションを実行するには、SDK をインストールする前に、Docker をインストールして、そのサービスを開始する必要があります。Docker を使用してアプリケーションをローカルで実行するには、sudo を使用するか、root を使用して実行する必要があります。これは、Docker デーモンが使用するソケットは、デフォルトでは root によって所有されているためです。これは、Post-installation steps for Linux の説明を参照して解決できます。

Docker Toolbox をインストールしなかった場合は、ここで、ご使用の開発マシンにインストールしてください。Install Docker Toolbox on Windows のプロセスを参照して、SDK を再インストールします。SDK のインストールを参照してください。

Docker が実行されていることを確認します。

手順

1. アプリケーションをローカルに作成して実行した後 (アプリケーションのローカル実行)、コマンド `grd_sdk run -d -w <path to app>` を入力することで、Docker CentOS コンテナーを使用してアプリケーションをローカルに実行します (例: `grd_sdk run -d -w ./testapp`)。 権限エラーが発生した場合は、`sudo grd_sdk run -d -w <path to app>` を実行してみてください。
2. ブラウザーを開いて、`https://127.0.0.1:5000/` をテストします。
3. アプリケーションを停止するには、Ctrl+C を押します。

親トピック: アプリケーションの作成、実行、パッケージ化、およびデプロイ

アプリケーションのパッケージ化、デプロイ、および実行

Guardium GUI Application Framework を使用して、アプリケーションをアーカイブ (.zip ファイル) としてパッケージし、それを Guardium テスト環境にデプロイして、アプリケーションおよびそのログを表示します。

始める前に

Guardium v10.5 以降が実行されています。

CLI コマンド store system ecosystem on を実行して、Guardium UI (「設定」 > 「ツールとビュー」 > 「アプリケーション・ライフサイクル」) でのアプリケーション・ライフサイクル・ページを含むエコシステム・プロセスおよびコンポーネントを有効にします。

手順

1. シェル・プロンプトを開き、cd コマンドを使用して、SDK インストール・フォルダーの bin サブフォルダーに移動します。
2. コマンド構文 `grd_sdk package -w <path to app folder> -p <Application Root Folder>.zip` を入力することで、アプリケーションをパッケージ化します。
3. `grd_sdk deploy -p <Application Root Folder>.zip -g <Guardium Console IP>` を入力することで、アプリケーション (拡張) を Guardium システムにデプロイします。
   注意:
   コマンド grd_sdk deploy を使用して、実稼働環境にアプリケーションをデプロイしないでください。
4. Guardium システムで、「設定」 > 「ツールとビュー」 > 「アプリケーション・ライフサイクル」にナビゲートします。アプリケーション状況がインストール済みになるまで待ちます
5. アプリケーションをアップロードした Guardium システムの CLI で、次のように入力してファイル・サーバーを開始します。

```
su cli
fileserver <ip of workstation> <time in seconds to run the filserver, for example 3600>
```

6. 「設定」 > 「ツールとビュー」 > 「アプリケーション・ライフサイクル」にナビゲートして、「開始」をクリックし、アプリケーションを開始します。Guardium マシンは run.py のみを呼び出すため、コンテナーの開始時にカスタム・サービス (ノードなど) を開始するようにアプリケーションをプログラミングする必要があります。(開発環境では、init スクリプトを使用してサービスを開始することができます。アプリケーションへの Python ライブラリーの追加を参照してください。)
7. <GuardiumIP>:8445 を参照して、ファイル・サーバーのログを表示します。このファイルは、logs/apps/app.log です。

次のタスク

1. 「Guardium アプリケーション」にナビゲートして、アプリケーション出力を表示します。
2. オプションで、「設定」 > 「ツールとビュー」 > 「アプリケーション・ライフサイクル」にナビゲートして、アプリケーションの行の「アプリケーション・ログ」列で  をクリックし、アプリケーションのアクティビティーのログを表示します。

親トピック: アプリケーションの作成、実行、パッケージ化、およびデプロイ
関連情報:
システム CLI コマンド

アプリケーションを作成するためのチュートリアル

このチュートリアルでは、Guardium SDK を使用してアプリケーションを作成し、そのアプリケーションをワークステーションで実行してテストし、最後に必要に応じて Guardium システムにデプロイする方法を説明します。

前提条件

- Python 2.7.9 以上 (SDK の前提条件および制限事項を参照)
- Guardium SDK (SDK のインストールを参照)
- 作業および認証を行うための Guardium システム (V10.5 以上)
- オプション: Docker (または Windows 上の Docker Toolbox) (SDK の前提条件および制限事項を参照)

アプリケーションの作成

開発用のワークステーションで、新規アプリケーション・ワークスペースを作成します。例:
$ grd_sdk create -w ./myApp

Guardium SDK により、そのフォルダー内に、Guardium システムに接続し、GuardAPI を使用して最近のログインを取り出すサンプル・アプリケーションが作成されます。アプリケーション・ワークスペースでは、アプリケーション用の Python 仮想環境が作成され、アプリケーションに必要な複数の Python パッケージがインストールされます。

注意:
SDK インストール・フォルダーの下にワークスペースを作成しないでください。SDK の下にあるフォルダーは、SDK のアップグレード時に削除されます。

grd_sdk run -w ./myApp と入力してアプリケーションを実行し、すべてが正常であることを確認します。

すべてのアプリケーションは、Guardium マシン (v10.5 以上) に対してセキュア接続を確立するために認証を行う必要があるため、Guardium ホストおよび cli ユーザー・パスワードを求められます。

```
Guardium システムの IP またはホスト名を次のように入力してください。
guardium.mycompany.com

Guardium cli ユーザー・パスワードを次のように入力してください。
<pw>
```

次に、アプリケーションには、実行するための Guardium UI ユーザーが必要になります。 このユーザーには、アプリケーションで使用されるアクションを実行するのに十分な特権が必要であることに注意してください。

```
OAuth アクセス・トークンを生成するために、GUI ユーザーを次のように入力します。
<user>

GUI ユーザー・パスワードを次のように入力します。
<pw>
```

すべてが正常に行われると、アプリケーションにより、トークンおよび GUI ユーザー資格情報が myApp/.guard_config に保存されます。アプリケーションの開始時に、アプリケーションの Python 仮想環境を通して run.py が次のように呼び出されます。

```
Running ['/usr/local/etc/GuardiumAppSDK/bin/run_app.sh',
'~/Documents/myApp', '~/Documents/test/run.py']
```

サンプル・アプリケーションを表示するには、ブラウザーで http://localhost:5000 を開きます。以下のような画面が表示されます。

コードの追加

コードを少し変更してみます。日付の形式を設定する必要があり、それを行うために Python パッケージが必要であると仮定します。パッケージを使用する前に、Python パッケージをダウンロードして、src_deps/pip フォルダーに追加し、そのパッケージが必要であることをアプリケーションに認識させる必要があります。

例えば、サンプルの Hello World アプリケーションで日時の形式を改良するには、Flask-Moment パッケージ (WHL ファイル) を Python サイトからダウンロードして、新規ワークスペースの src_deps/pip に保存します。次に、
src_deps/pip/APP_CUSTOM_REQUIREMENTS.txt に以下を追加することで、このパッケージを Python 仮想環境にインストールするようにアプリケーションに指示します。

Flask_moment==0.6.0

アプリケーションを再度実行して、*Flask moment* が正常にインストールされたことをテストします。アプリケーション・コンソールに以下が表示されます。

```
… Installing collected packages: Flask-moment
… Successfully installed Flask-moment-0.6.0
…
```

パッケージを使用するには、パッケージを app/views.py にインポートしてインスタンス化します。

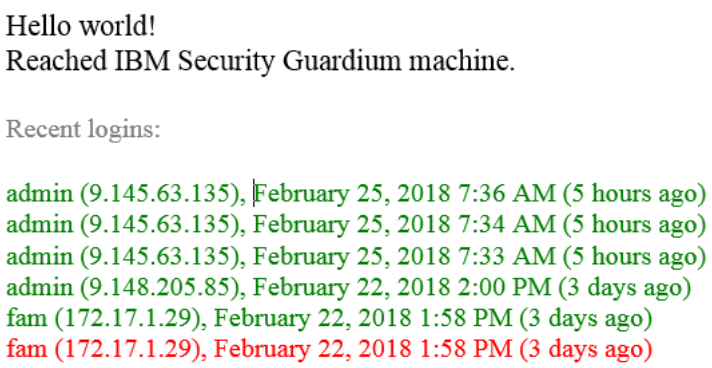
```
…
from grdlib.GRDApi import GRDApi
from flask_moment import Moment
moment = Moment(app)

@app.route('/')
…
```

これで、パッケージをアプリケーション全体で使用できます。例えば、次のように app/templates/index.html を編集して日付の形式を改良します。

```
…
<head>
  …
  {{ moment.include_jquery() }}
  {{ moment.include_moment() }}
</head>
…
{{ moment(row.login_datetime).format('LLL') }}
({{ moment(row.login_datetime).fromNow(refresh=True) }})
```

ブラウザー内でアプリケーション・ページを最新表示して、更新された日付形式を表示します。



Guardium システムへのアプリケーションの送信

アプリケーションを Guardium システムに送信する場合、事前に Docker コンテナーを使用してローカル側でテストすることをお勧めします。Guardium システムにデプロイされたときの実行方法と似ているためです。そのためには、Ctrl+C を使用してアプリケーションを停止し、次のように *-d* フラグ (または --use-docker) を指定して再び実行します。

$ grd_sdk run -w ./myApp -d

ブラウザー・ページを最新表示して、すべてが正常であることを確認します。

アプリケーションを Guardium システムにパブリッシュする前に、Guardium システムでアプリケーション (エコシステム) のサポートを有効にする必要があります。このサポートはデフォルトで無効になっているためです。

cli ユーザーとしてマシンに接続し、以下の 2 つのコマンドを入力します。

```
$ ssh cli@guardium.company.com
$...cli@...$ store system ecosystem on
cli@...$ store system signature off
```

最初のコマンドが完了すると、Guardium UI にログインして、「アプリケーション・ライフサイクル」ビュー (「設定」 > 「ツールとビュー」 > 「アプリケーション・ライフサイクル」 >) を開くことができます。2 番目のコマンドにより、署名されていないアプリケーションを Guardium システムにパブリッシュできます。IBM はまだユーザーのアプリケーションを検証していないためです。

アプリケーションのパブリッシュ

これで、アプリケーションをパブリッシュする準備ができました。内容をパッケージして、Guardium システムに送信し、そのシステムで開始する必要があります。

grd_sdk package -w ./test -p test.zip を実行して、アプリケーションのパッケージを準備し、Guardium システムに送信できるようにします。

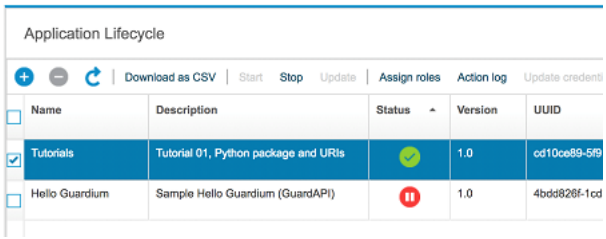
オプションで、manifest.json を事前に編集して、アプリケーションをより詳しく記述します。

次に、

grd_sdk deploy -p test.zip -g guardium.company.com を実行して、アプリケーションを Guardium システムに送信します。

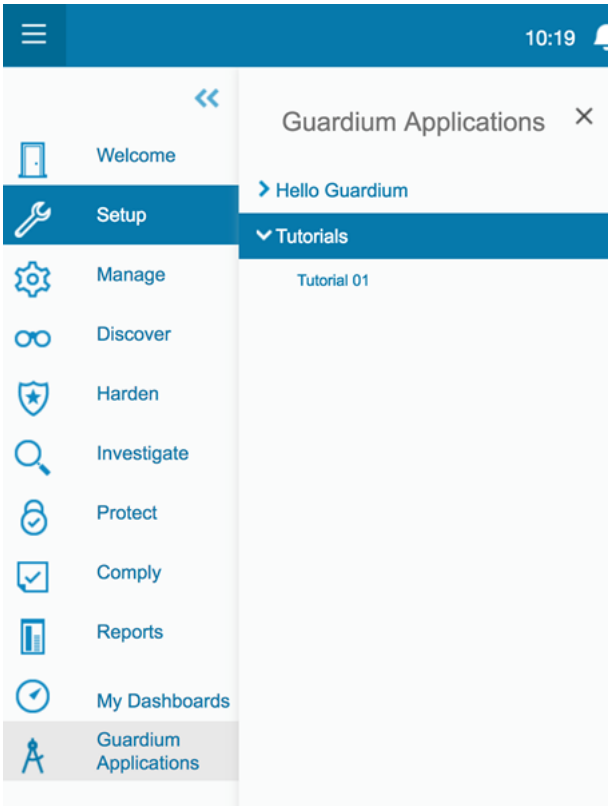
アプリケーションを開発したマシンと異なる場合は特に、この Guardium システムに対して再び認証を行うよう求めるプロンプトが表示されることがあります。

Guardium の「アプリケーション・ライフサイクル」ページで、 をクリックして最新表示します。アプリケーションのリストにアプリケーションが表示されます。そのアプリケーションを選択して、「開始」をクリックします。開始したことが「状況」列に示されたら、「Guardium アプリケーション」の下にあるメインのナビゲーション・メニューからアプリケーションにアクセスできます。



アプリケーションが既に存在している場合、デプロイ・コマンドによってそのアプリケーションは上書きされないことに注意してください。 デプロイする前に、再びコマンド・ラインからアプリケーションを停止して削除する必要があります。アプリケーションを更新するには、「アプリケーション・ライフサイクル」ビューで、そのアプリケーションを停止して、「更新」を選択し、アプリケーションの新規パッケージを選択します。

アプリケーションを表示するには、「Guardium アプリケーション」 > <アプリケーション名> にナビゲートします。



注: 現時点では、Guardium UI では、外部のスクリプト、iFrame、イメージは、それが IBM ソースからのものではない場合にアプリケーションでの組み込みがブロックされます。この状況に対応するには、スクリプトを外部ソースから参照するのではなく、アプリケーションに組み込むことが必要になる場合があります。次の演習で試してください。

上記のアプリケーションの zip パッケージは、GuardiumAppSDK/sample_apps/pythonPackage.zip にあります。

リンクの追加

この例では、デフォルトの 25 件の Guardium システムへの最新ログインではなく、さらに多くのログインを取り出すためのリンクを追加します。これを行うには、GET 引数を指定して、リンクを *index* に追加します。そのためには、views.py を次のように変更します。

```
from flask import send_from_directory
from flask import render_template
from flask import request
from app import app
from grdlib.GrdConnection import GrdConnection
from grdlib.GRDApi import GRDApi
@app.route('/')
@app.route('/index')
def index(methods=['GET']):
```

```
    """Define logic and view for app root."""
    grd_connection = GrdConnection()
    grd_api = GRDApi(grd_connection)

    if request.args.get('fetchSize', '') is None:
        fetchSize = 25
    else:
        fetchSize = request.args.get('fetchSize', '')
    result = json.loads(json.dumps(
        grd_api.create_online_report(
            'Guardium Logins', 'NOW -7 DAY', 'NOW',
            fetchSize=fetchSize,
            sortColumn='Login Date And Time',
            sortType='DESC')))
```

index.html テンプレートを編集し、fetchSize パラメーターを指定してリンクを組み込みます。

```
…
</ul>
        <a href="#_dcs_markdown_workspace_Transform_htmlout_0_nl_ja_com.ibm.guardium.doc.admin_ecosystem_{{ url_for('index',
fetchSize=100) }}">More...</a>
</div>
  <a href="#_dcs_markdown_workspace_Transform_htmlout_0_nl_ja_com.ibm.guardium.doc.admin_ecosystem_{{ url_for('debug_view')
}}">Show app.log</a><br>
  …
```

href 属性で `index?fetchSize=100` にリンクするのではなく、`url_for` 関数が使用されていることに注意してください。どちらの方法も機能しますが、アプリケーションを Guardium システムから表示する予定の場合は、リンク先のすべてのリソースに CSRF トークンを追加する必要があります。Guardium システムは、外部または任意の URI からのコードをブロックすることでセキュリティーを強化しているためです。 あるいは、`gpylib.get_CSRF_token` を使用して CSRF トークンを取得し、任意の場所に手動で追加することができます。

ブラウザー・ビューを最新表示します。アプリケーションがローカル・マシンで実行されている限り、CSRF トークンは空になります。表示するには、アプリケーションを再びデプロイします。

演習として、アプリケーションをパッケージし、Guardium システムに再びデプロイしてください。欠落している外部スクリプトを見つけて、アプリケーションに組み込むこともできます。コマンド・ラインを使用する場合は、事前にアプリケーションを停止して削除する必要があることに注意してください。 あるいは、「アプリケーション・ライフサイクル」ページから「停止」をクリックして、「更新」をクリックします。これで終了です。

上記のアプリケーションの zip パッケージは、

GuardiumAppSDK/sample_apps/tutorial01-python-packages-and-URIs.zip にあります。

親トピック: アプリケーションの作成、実行、パッケージ化、およびデプロイ

Eclipse でのアプリケーションの開発

開発環境をセットアップした後で、その開発環境を Eclipse 内にインポートします。その Eclipse 統合開発環境 (IDE) 機能を使用して、アプリケーションを開発します。

このタスクについて

Python 開発のために、Python 開発で使用される Eclipse IDE である PyDev をインストールします。

Eclipse の最新バージョンは、Eclipse Web サイトにあります。

手順

1. Eclipse 内に PyDev をインストールします。
   - Eclipse Marketplace からインストールするには、メイン Eclipse「ヘルプ」パネル上で「ヘルプ」 > 「Eclipse Marketplace」をクリックします。
   - (http://pydev.org/updates) から PyDev リポジトリーをインストールするには、メイン Eclipse「ヘルプ」パネル上で「ヘルプ」 > 「新規ソフトウェアのインストール (Install New Software)」をクリックします。
2. PyDev をインストールした後で、パースペクティブを PyDev に切り替えます。
3. PyDev パースペクティブで、「ファイル」 > 「新規」 > 「PyDev プロジェクト (PyDev project)」をクリックし、「PyDev プロジェクト (PyDev Project)」ダイアログで以下のタスクを実行します。
   a. プロジェクト名を入力します。
   b. 「文法バージョン (Grammar Version)」リストから「2.6」を選択します。
   c. 「既存のソースへのリンクの作成 (Create links to existing sources)」を選択します。
   d. SDK 内で仮想環境を使用するためのインタープリターを構成するには、「リストされていないインタープリターを構成するには、ここをクリックしてください (Click here to configure an interpreter not listed)」をクリックします。
   e. 「Python インタープリター (Python Interpreters)」ダイアログ内で「新規」をクリックし、システム上の Python 実行可能ファイルの名前およびパスを入力します。
   f. 「OK」をクリックします。

      システム python パスに追加されるフォルダーを選択するときに、「site-packages」フォルダー・パス (.../<*app_name*>/grd_appfw_venw/lib/python2.7/site-packages) チェック・ボックスが選択されていることを確認します。

   g. 「適用」および「OK」をクリックして、「PyDev プロジェクト (PyDev Project)」ダイアログに戻ります。
   h. 「既存のソースへのリンクの作成 (Create links to existing sources)」をクリックし、「次へ」をクリックします。
   i. 「外部ソース・フォルダーの追加 (Add external source folder)」をクリックし、開発環境のルート・ディレクトリーに移動し、「OK」をクリックし、「終了」をクリックします。

      開発環境を含む新規の Eclipse PyDev プロジェクトが「パッケージ・エクスプローラー (Package Explorer)」に表示されます。

親トピック: アプリケーションの作成、実行、パッケージ化、およびデプロイ

アプリケーション・ファイル構造

作成した Guardium アプリケーションは、圧縮ファイル内で配布されます。

開発環境をセットアップするときに作成される Hello World サンプル・アプリケーションは、アプリケーション用に使用できる基本テンプレートです。ただし、より複雑なファイル構造を持つアプリケーションも作成できます。

以下のリストは、アプリケーションのルート・ディレクトリーに追加できるファイルおよびサブディレクトリーのレイアウトの概要を示しています。アプリケーション・ファイルおよびサブディレクトリーで必要な命名方法の概要も示しています。

<App Root Folder>	アプリケーション・ファイルのルート・ディレクトリー。
grd_appfw_venv	Python 仮想環境。アプリケーション・ランタイム環境を分離します。
app/views.py	Web アプリケーションへのメイン・エントリー・ポイント。
app/templates	アプリケーションが必要とする Jinja テンプレートのすべての Python Flask を格納するオプションのサブディレクトリー。
app/static	オプションのサブディレクトリー。
css	CSS ファイル
js	JavaScript ファイル
resources	リソース・ファイル
application_<LANG>.properties	application_<LANG>.properties ファイルは、指定された言語コードのグローバリゼーション・リソース・バンドルです。グローバリゼーション用のテキスト・ストリングは、キー/値のペアとして、Java フォーマット・プロパティー・ファイルに保管されています。グローバリゼーション用のテキスト・ストリングを構成した場合は、該当するロケールがユーザー設定で選択されると、それらのストリングが Guardium に表示されます。
manifest.json	アプリケーション・マニフェスト記述ファイル。
src_deps	ソース依存関係を格納するオプションのディレクトリー。
src_deps/pip	アプリケーションが必要とする追加の Python ライブラリーを格納するオプションのサブディレクトリー。
src_deps/rpms	アプリケーションが必要とするすべての RPM 依存関係を格納するオプションのサブディレクトリー。RPM は CentOS 6.7 x86_64 互換である必要があります。
src_deps/init	アプリケーションが必要とする RPM および Python ライブラリー以外のすべての依存関係を格納するオプションのサブディレクトリー。

- アプリケーションへの Python ライブラリーの追加
  アプリケーションで依存関係 (RPM や Python ライブラリーなど) が必要な場合、app フォルダーの src_deps サブフォルダーにその依存関係を追加できます。
- アプリケーション・マニフェストの構造
  マニフェストは、アプリケーションが提供する機能を Guardium に対して示す JSON ファイルです。
- ソース依存関係としての Node.js のインストール
  Node.js を Web アプリケーション・フレームワークとしてインストールして、IBM Guardium GUI Application Framework SDK に含まれている Flask フレームワークを置き換えることができます。
- アプリケーションのメモリー使用量の最適化
  Guardium アプリケーション・フレームワークを調整して、アプリケーションのメモリー使用量を最適化します。

親トピック: SDK の処理

アプリケーションへの Python ライブラリーの追加

アプリケーションで依存関係 (RPM や Python ライブラリーなど) が必要な場合、app フォルダーの src_deps サブフォルダーにその依存関係を追加できます。

src_deps フォルダーには、以下のサブフォルダーが含まれています。

pip

pip フォルダーは、Guardium アプリケーションに必要な追加の Python ライブラリーを指定して、それらをインストールするために使用します。このフォルダーは、grd_sdk create を使用してアプリケーションを作成するときに (Guardium アプリケーションの作成を参照) 空の python requirements ファイルと一緒に作成されます (python 標準)。このファイルに、アプリケーションによってインストールする必要がある追加の Python パッケージのほか、それらの従属パッケージの名前を追加します。

Python パッケージは、アプリケーションが実行されるオペレーティング・システムと互換性がなければなりません。これは、Docker を使用するかどうかに関係なく、Windows (Windows 7 でテスト済み) または Linux (Ubuntu 16.04 でテスト済み) でローカルに実行する場合に適用されます。ただし、その他のオペレーティング・システム用の Python パッケージもこのフォルダーに追加することをお勧めします。これにより、異なるプラットフォームで作業する開発者の間でアプリケーションを共有できるためです (すべてのプラットフォームで requirements ファイルは同じです)。

ファイル名は、インストールする順番に並べて、改行 (UNIX の行の終わり) で区切る必要があります。

例えば、アプリケーションで observable-0.01.00 Python ライブラリーが必要な場合、observable-0.01.00.tar.gz ファイルを pip フォルダーに追加し、同じフォルダー内にある APP_CUSTOM_REQUIREMENTS.txt 内で observable==0.01.00 を付加してパッケージの名前とバージョンを指定します。

ライブラリーの追加に加えて、以下のテキスト・ファイルにパッケージをリストします (例: observable == 0.01.00)。
<app_workspace_path>/src_deps/pip/APP_CUSTOM_REQUIREMENTS.txt

Python Wheel パッケージがある場合は、必ず使用します。tar.gz パッケージには、C 言語のサード・パーティー・コードが含まれている場合があることに注意してください。これにより、他のファイル・システムでコンパイルの問題が発生する可能性があります。

Python wheel ファイルは、それらがコンパイルされたのと同じシステム・アーキテクチャーにインストールする必要があります。Guardium Application Framework で動作させるために、wheel ファイルは Centos 7.4.1708 x86_64 でコンパイルする必要があります。互換アーキテクチャーが使用されている場合、Python bdist_wheel コマンドを使用して、ユーザーのシステムでライブラリーのソース・コードから wheel ファイルを作成できます。Python ライブラリーのソース・フォルダーのルート・ディレクトリー内から `python setup.py sdist bdist_wheel` コマンドを実行すると、wheel ファイルが作成されます。

アプリケーションの Python パッケージを手動でダウンロードする代わりに、pip2pi Python パッケージを使用すると便利です。これには pip が必要です。これは `pip install pip2pi` コマンドを使用して開発コンピューターにインストールできます。このパッケージをインストールした後、以下のコマンドを実行します。

```
pip2tgz <target-directory> <Python package>
```

例えば、次のコマンドでは、パッケージの wheel がその依存関係とともに指定のフォルダーにダウンロードされます。

```
pip2tgz src_deps/pip/ pytest==2.8.2
pip2tgz src_deps/pip/ pytest
```

Python のバージョン番号のパラメーターはオプションです。このパラメーターを使用すると、特定バージョンのパッケージをダウンロードできます。

別のオペレーティング・システムでアプリケーションを実行している場合は、以下を実行することにより、TXT ファイルで指定したすべての必要なパッケージをインストールできます。

cd ./src_deps/pip
pip2tgz ./ -r APP_CUSTOM_REQUIREMENTS.txt

rpms

追加の Red Hat Enterprise Linus (RHEL) RPM をインストールするには、rpms フォルダーを使用します。RPM は CentOs 7.4.1708 x86_64 互換でなければなりません。

アプリケーションがインストールされるか、Docker コンテナーを介して実行された後、rpms フォルダー内のすべての RPM ファイルがインストールされます。インストールの順序を制御する必要がある場合 (通常は、依存関係がある RPM の場合)、rpms フォルダー内に ordering.txt ファイルを追加して、RPM ファイル名をそれぞれ別の行にリストします。このテキスト・ファイルには、rpms フォルダーにあるファイルの名前を含める必要があります。ファイル名は、インストールする順番に並べて、改行 (UNIX の行の終わり) で区切る必要があります。

アプリケーションを Docker コンテナーでローカルに実行すると、RPM がインストールされます。(Docker コンテナーを使用せずに) アプリケーションをローカルに実行すると、必要となる可能性があるシステムの依存関係についてユーザーに通知するために、このディレクトリー内の依存関係が画面にエコー出力されるのみとなります。

init

pip フォルダーや rpms フォルダーに配置するのが適切ではない依存関係は、init フォルダーに追加します。この ordering.txt ファイルの行 (UNIX の行の終わり) は、アプリケーションのインストール中にシェル・コマンドとして実行されます。

Guardium アプリケーションを Docker コンテナーを介して実行する場合にのみ、ordering.txt は単一行コマンドとして実行されます。rpms と同様に、Guardium アプリケーションをローカルで実行すると、このコマンドは実行されず、画面にエコー出力され、アプリケーションの console/log に書き込まれるだけです。これは、ローカルの開発マシンに特別なパッケージを自動的に再インストールすることはユーザーにとって必要がないとインストーラーが想定するためです。例えば、明示されていない複雑な依存関係チェーンのある RPM のコレクションをインストールする必要があるとします。このユース・ケースでは、dependant_rpms.tar.gz という名前の .tar ファイルを init フォルダーに追加します。以下のコマンドを ordering.txt ファイルに追加します。

```
mkdir /src_deps/init/dependant_rpms
cd /src_deps/init
tar -xzvf dependant_rpms.tar.gz
yum -y localinstall --disablerepo=*dependant_rpms/*rpm
rm -rf dependant_rpms
```

注: この例の `--disablerepo=*` スイッチは、インターネット・アクセスのない Guardium コンソールのリモート・リポジトリーへの接続を yum が試行しないようにするために使用されています。
この例では、yum による RPM 依存関係の自動解決が使用されています。これにより、指定した一連の RPM が必要な順序でインストールされます。RPM が rpms フォルダーに含まれている場合は (rpmsを参照)、ユーザーがインストールの順序を指定する必要があります。
親トピック: アプリケーション・ファイル構造

アプリケーション・マニフェストの構造

マニフェストは、アプリケーションが提供する機能を Guardium に対して示す JSON ファイルです。

次の表で、`manifest.json` ファイルに含めることができるフィールドについて説明します。

表 1. アプリケーション・マニフェストのフィールド

フィールド	必須	タイプ	説明
description	はい	ストリング	ユーザーが判読可能なアプリケーションの説明。アプリケーションがグローバル化されている場合、このフィールドでは、オプションでリソース・バンドルを示すことができます。
name	はい	ストリング	ユーザーが判読可能なアプリケーションの名前。アプリケーションがグローバル化されている場合、このフィールドでは、オプションでリソース・バンドル・キーを示すことができます。
navigation_areas	いいえ	領域タイプの配列	Guardium UI ナビゲーション・レイアウトに追加されるアプリケーションの完全なナビゲーション・ページを示す、1 つ以上のオブジェクト。領域オブジェクトは、タブとして表されます。

フィールド	必須	タイプ	説明
SDK_version	はい	文字列	(Guardium バージョンに基づく) SDK 構造のバージョン・ストリング: <Guardium バージョン>.<SDK バージョン>。 このフィールドは変更しないでください。
sqlguard_ip	はい	ストリング	アプリケーションが通信する Guardium システムの IP アドレス。未指定の場合、このフィールドの値は、127.0.0.1 にデフォルト設定されます。
supported_unit_types	いいえ	ストリングの配列	有効な値: collector、aggregator、cm。指定されたユニット・タイプでこのアプリケーションをインストールする機能を制限するには、このフィールドに 1 つ以上の値を指定します。
uuid	はい	文字列	アプリケーションの RFC 4122 準拠の汎用固有 ID。 create コマンドは Python UUID パッケージを使用して、uuid 値に対してランダムの 128 ビットの数値を生成します。 アプリケーション・マニフェスト・ファイルの作成に SDK を使用しない場合、uuid フィールドに固有値を手動で入力する必要があります。
version	はい	ストリング	アプリケーションのバージョン・ストリング。ここでは任意の形式を使用できます。

次の表で、領域タイプのフィールドについて説明します。

フィールド	必須	タイプ	記述
id	はい	文字列	(アプリケーション内で) 固有の領域 ID
name	はい	文字列	Guardium UI レイアウトのナビゲーションのエントリー・ポイントの短い記述名
description	いいえ	文字列	領域の説明 (オプション)
url	はい	文字列	ロードする URL (アプリケーション・ルートからの相対 URL)。Guardium アプリケーション内に存在する URL のみを参照できます。
default_roles	いいえ	文字列配列	このナビゲーション領域へのアクセス権限をデフォルトで持つ必要があるロール名の配列。指定されない場合は「admin」が想定されます。

親トピック: アプリケーション・ファイル構造

ソース依存関係としての Node.js のインストール

Node.js を Web アプリケーション・フレームワークとしてインストールして、IBM Guardium GUI Application Framework SDK に含まれている Flask フレームワークを置き換えることができます。

手順

1. 使用する Node.js アーカイブ (.tar) をダウンロードし、そのアーカイブを app/src_deps/init ディレクトリーにコピーします。
2. 同じフォルダー内に、使用するアーカイブ (この場合は node-v6.3.0-linux-x64.tar.gz) を参照する以下の例のようなインストール・スクリプトを作成します。

```
#!/bin/bash
##
## install node and npm from source tarball, and make available on the path
##
cd /usr/local
tar --strip-components 1 -xzf /src_deps/init/node-v6.3.0-linux-x64.tar.gz
```

3. 同じフォルダー内に、以下のコンテンツを含むファイル ordering.txt を作成します。

```
/src_deps/init/install_nodejs_npm.sh
```

ordering.txt は、Node.js インストール・スクリプトを実行するように IBM Guardium に通知します。

親トピック: アプリケーション・ファイル構造

アプリケーションのメモリー使用量の最適化

Guardium アプリケーション・フレームワークを調整して、アプリケーションのメモリー使用量を最適化します。

Guardium コンソールで実行されるアプリケーションは、メモリーが 200MB に制限されます。アプリケーションがこのしきい値を超過すると、/var/log/guardium.error ファイル内にログが生成されます。最終的に、アプリケーションがこのしきい値を超過するさらに多くのメモリーを使用し続けると、アプリケーションをホストするコンテナーがシャットダウンして再始動します。

以下の 3 つの方法で、アプリケーションが 200 MB のしきい値を超過しないように調整できます。

- 小さいメモリー占有スペースに処理をチャンク化する (または重ならないように調整する) ことで、大容量のメモリーの割り振りを回避する。
- アプリケーション・フレームワークで使用されるメモリー・モデルを変更する。
- 大容量のメモリーを使用するコードが終了した時点でガーベッジ・コレクションを呼び出す。

アプリケーション・フレームワークのメモリー・モデルの変更

デフォルトでは、アプリケーション・フレームワークにより、Flask が使用する Werkzeug WSGI Web アプリケーション・サーバーは単一プロセスとして稼働するように構成されます。各要求を処理するためにスレッドが使用されます。新しい各要求を処理するための個別プロセスを作成するようにアプリケーション・サーバーを構成できます。要求が完了すると、プロセスは破棄されます。この要求を処理するために、Python インタープリターによって割り振られたすべてのメモリーが解放されます。

この動作をオーバーライドするには、run.py ファイルを編集し、threading=False および process=N (N は 1 より大きい数値) を設定します。例えば、値が process=3 の場合、インタープリターごとに約 25 MB が割り振られるため、メモリーが追加で必要になった場合に対応できます。

```
__author__ = 'IBM'

from app import app
from app.gpylib import gpylib

gpylib.create_log()
app.run(debug = True, host='0.0.0.0',
threaded=False,
 process=3)
```

アプリケーション ZIP アーカイブ・ファイル内の template フォルダーに run.py のソースを含めます。インストール中に作成される run.py は、ユーザーの設定により上書きされます。

ガーベッジ・コレクションの呼び出し

Python は、メモリーをいつ解放するか認識しない場合があります。大容量のメモリーが不要になるセクションの直後に以下のコードを配置することで、ガーベッジ・コレクションを高速化できます。

```
import gc
gc.collect()
```

注: Python では、コードで使用されていたメモリーが OS に返されるという保証はありません。ガーベッジ・コレクションで保証されるのは、オブジェクトで使用されていたメモリーが収集され、将来のいずれかの時点において別のオブジェクトで使用されるために解放されるということのみです。前のセクションで説明したアプリケーション・フレームワークのメモリー・モデル・オプションを変更することが、長時間実行されるアプリケーションには重要です。プロセスを強制終了すると、他のコンポーネントで使用するためにメモリーが確実に解放されます。

ツール

メモリーの問題の特定を支援するいくつかのツールを以下に示します。

Memory Profiler
: プロセスのメモリー消費をモニターするための Python モジュール。詳しくは、Python Memory Profiler を参照してください。

Linux ユーティリティー
: コマンド・ライン・ユーティリティー top を使用して、マシンで実行されているすべての Python プロセスをモニターできます。

```
top -p $(pgrep -d',' python)
```

また、以下のコマンドを使用して、システムのすべての Python インタープリターで使用されている合計 MB を取得できます。

```
ps -e -o pid,comm,rss |
grep python | awk '/python/{print $3}' |
awk '{sum+=$1} END
```

リソース・モジュール
: 以下のコードをモジュールに追加することで、プロセスで使用されているメモリーの容量をログに記録できます。

```
import resource
print 'Memory usage: %s (kb)' % resource.getrusage
(resource.RUSAGE_SELF).ru_maxrss
```

親トピック: アプリケーション・ファイル構造

GUI Application Framework の基礎

Guardium GUI Application Framework アプリケーションは、Flask マイクロフレームワークで実行され、Flask Web サーバーから提供されるスタンドアロンの Web アプリケーションです。

インストールの概要

すべてのアプリケーションは、固有の Flask サーバーで実行されます。各 Flask サーバーは、セキュアな Linux コンテナー内で実行されます。コンテナーは、Flask アプリケーション・コードベースを安全に含めるための実装スタックです。

各アプリケーションは、RESTful API エンドポイントを使用してインストールされます。このインストール・エンドポイントでは、以下のタスクが処理されます。

- アプリケーションのマニフェストの検証
- 非同期タスク固有
  - アプリケーションを Guardium に登録して、Web トラフィック・プロキシー、および Guardium からアプリケーションへの HTTP 要求/応答ライフサイクルを有効にする。
  - コンテナー・イメージを、その内部に組み込まれているアプリケーション・コードを使用して自動的に作成する。
  - 永続ストレージのために使用されるデータ専用のセカンダリー・コンテナーにバインドされているコンテナー・イメージからコンテナーを自動的に実行する。

表 1. GUI Application Framework REST API エンドポイント

エンドポイント	パラメーター	記述

エンドポイント	パラメーター	記述
GET /gui_app_framework/application_creation_task	アプリケーション・アイデンティティー	アプリケーション作成のためのすべての非同期要求の状況詳細のリストを取得します。
GET /gui_app_framework/application_creation_task/{application_id}	アプリケーション・アイデンティティー	アプリケーション作成のための 1 つの非同期要求の状況詳細のリストを取得します。
POST /gui_app_framework/application_creation_task	アプリケーション (.zip) バンドル・ファイル	Application Framework 内でアプリケーションを作成し、そのアプリケーションを Guardium に登録します。アプリケーションは非同期に作成されます。application_id への参照が返されるので、後続の API 呼び出しではこの参照を使用してアプリケーションのインストールの状況を判別する必要があります。
POST /gui_app_framework/application_creation_task/{application_id}	アプリケーション・アイデンティティー、キャンセルの状況	Application Framework 内で新規アプリケーションのインストールを更新します。application_id パラメーターおよび status パラメーターは必須です。
GET /gui_app_framework/applications		Guardium コンソールにインストールされているアプリケーションのリストと、そのマニフェスト JSON 構造および状況を取得します。
GET /gui_app_framework/applications/{application_id}	アプリケーション・アイデンティティー	コンソールにインストールされている特定のアプリケーションと、そのマニフェスト JSON 構造および状況を取得します。
POST /gui_app_framework/applications/{application_id}	アプリケーション・アイデンティティー、開始と停止の状況	アプリケーションを更新します。状況を RUNNING に設定してアプリケーションを開始するか、状況を STOPPED に設定してアプリケーションを停止します。
DELETE /gui_app_framework/applications/{application_id}	アプリケーション・アイデンティティー	アプリケーションを削除します。

Python

Guardium アプリケーションを開発するには、Python 2.7.9 を使用する必要があります。

詳しくは、Python の Web サイト (https://www.python.org/doc/) を参照してください。

Flask

Flask は、Python で記述されたマイクロ Web アプリケーション・フレームワークです。

Flask は、アプリケーション・コード・エンドポイントに対応する Web サーバーです。ユーザーは、Python 関数を使用してユース・ケースを提供します。Flask アプリケーションでは、Python メソッドごとにルート注釈を使用できます。Flask Web サーバーの始動後、そのルートについて Flask により HTTP/HTTPS バインド要求が送信され、Python 関数が実行されます。

Docker コンテナー内から実行される各 Flask サーバーでは、ポート *5000* が使用されます。Docker は、コンテナーから外部向けに、その内部ポート 5000 を 49152 から 65535 の一時範囲内の次の空きポートにマップします。登録フェーズ中、この外部向けにマップされたポートが Guardium で保管されます。これにより、Guardium を通じてアプリケーションに対する Web 要求が、正しいコンテナーにプロキシー処理されます。

以下のコードは、Python ルートの例です。

```
@app.route('/')
def hello_world():
    return 'Hello World!'
```

スタンドアロン Flask Web サーバーで、ブラウザーを通じた `http://localhost:5000` に対する Web 要求により、`Hello World!` が返されます。

次の表で、Flask の特定バージョンとその依存関係について概説します。

パッケージ	バージョン	記述
Flask	0.10.1	マイクロフレームワーク、またはマイクロ Web アプリケーション・フレームワーク
itsdangerous	0.24	データに署名し、暗号化するためのユーティリティー・パッケージ
jinja2	2.7.3	Python のテンプレート・エンジン
markupsafe	0.23	Jinja2 と併用される Unicode エスケープ・ライブラリー
Werkzeug	0.96	Python の WSGI (Web サーバー・ゲートウェイ・インターフェース) ユーティリティー・ライブラリー

詳しくは、Flask の Web サイト (http://flask.pocoo.org/) を参照してください。

Jinja2

Jinja2 は Python ライブラリーです。これを使用すると、コア・テンプレート・テキスト・ファイルから各種出力フォーマットのテンプレートを作成できます。Guardium アプリケーションに対して使用されるフォーマットは HTML です。Jinja2 には、高度な API が含まれており、またテンプレート・ファイルにコンテンツを動的に取り入れるために使用する構文ディレクティブ (ステートメント、式、変数、タグ) が多数含まれます。

以下の例に示されているように、ルートで提供されるPython メソッドから Jinja2 HTML テンプレートにデータを挿入する最も簡単な方法は、Flask の組み込み `render_template()` メソッドです。

```
@app.route('/')
def hello_world():
    return render_template('hello.html', title='Guardium')
```

テンプレート hello.html には、以下のコードが含まれます。

```
<!doctype html>
<title>Hello from Flask</title>
  <h1>Hello {{ title }}!</h1>
```

以下の HTML 出力が生成されます。

```
<!doctype html>
<title>Hello from Flask</title>
  <h1>Hello guardium!</h1>
```

詳しくは、Jinja2 の Web サイト (http://jinja.pocoo.org/docs/dev/) を参照してください。

HTTP 要求/応答のライフサイクル

アプリケーションが正常にインストールされている場合、アプリケーションに対する要求は、Guardium への確立済み接続を使用することでのみプロキシー処理されます。直接 URL 要求またはその他の方法を使用して、アプリケーションに直接アクセスすることはできません。

アプリケーションは、Guardium に対してセキュアな認証済みおよび許可済みのセッションを確立できます。セッションの整合性を検証するために作成される許可トークンは再使用できます。アプリケーションは、Guardium のすべての機能、セキュリティー、および認証性のファセットを取得します。GuardiumGuardium システムに対してデータをプルまたはプッシュするために、アプリケーションはユーザー・セッション状態を使用して、すべての Guardium RESTful API エンドポイントへのアクセス権限を取得できます。

コンテナー化されたアプリケーションおよびネットワーク

GUI Application Framework では、トラフィックは、コンテナーからコンテナー、コンテナーからその公開 IP アドレスのホスト (非ローカルホスト)、およびコンテナーから外部に流れます。

各アプリケーションがソース・コードのアーカイブ (.zip ファイル) として Guardium エンドポイントに渡されると、Guardium は、アプリケーション・コードベース固有の初期イメージを作成します。各イメージは、個別のコンテナーとして実行されます。コンテナーが実行または開始されると、Guardium は、内部 Flask サーバー・ポート (5000) を外部一時ポートにマップします。この外部一時ポートは Guardium に登録されるため、プロキシー処理されるアプリケーション・コードに対する要求は、正しいコンテナーにルーティングされます。

Python スレッド

キープアライブ・パケットの送信や定期的なガーベッジ・コレクションの実行などのバックグラウンド・タスクを実行するスレッドには、`thread.daemon=True` を設定する必要があります。デーモン・スレッドとして設定することにより、これらのスレッドは、プログラムの終了時に自動的に強制終了されます。

親トピック: SDK の処理

サンプル・アプリケーション

SDK には、いくつかのサンプル・アプリケーションが含まれています。

これらは /sample_apps にあります。sample_apps フォルダー内の README.md にアプリケーションの説明が記載されています。サンプル・アプリケーションの使用を開始する前に、必ずこのファイルをお読みください。サンプル・アプリケーションをデプロイすることも、独自のアプリケーションの開発時にサンプル・アプリケーションの一部を使用することもできます。

親トピック: SDK の処理

サポート関数

Guardium GUI Application Framework には、いくつかの組み込み経路、カスタム Jinja2 Flask 関数、およびアプリケーション開発をサポートするその他のヘルパー・ユーティリティーが付属しています。

概要

クライアント・サイド・ブラウザーからアプリケーションへのすべての HTTP 要求は、以下のフォーマットを使用します。

```
https://<sqlguard_ip>:8443/guardapp/{application_id}/{my_route}
```

application_id は、GUI アプリケーション作成のためにインストール RESTful エンドポイントを使用するプロセス中に割り当てられる整数値です。*application_id* 値は、grd_app_creator deploy コマンドを実行したときに返される Application Creation Task state 出力内に記録されます。

以下の例では、アプリケーション・アイデンティティーは `1023` です。

```
Application Creation Task state:
{'status': 'COMPLETED', 'application_id': '1023', 'error_messages': '[]'}
```

経路

以下の表の経路については、アプリケーションをターゲットとする独自の Web 要求を作成できます。

表 1. 要求経路

経路	フォーマット	記述
GET /debug	GET https://<sqlguard_ip>:8443/guardapp/ {application_id}/debug	検査のためにコンテナーの内部から /store/log/app.log ファイルをダウンロードします。
GET /debug_view	GET https://<sqlguard_ip>:8443/guardapp/ {application_id}/debug_view	ブラウザー・ウィンドウの内部に /store/log/app.log ファイルのコンテンツを表示します。
POST /log_level	POST https://<sqlguard_ip>:8443/guardapp/ {application_id}/log_level form body: level = 'INFO' 'DEBUG' 'ERROR' 'WARNING' 'CRITICAL'	アプリケーションがキャプチャーするロギングのレベルを動的に定義します。いずれかのログ・レベル値に設定された属性レベルを含むフォームをこのエンドポイントに通知します。Guardium は、/store/log/app.log ファイル内のログ収集レベルを動的にリセットします。

views.py 内の Flask エンドポイントへのアクセス

エンドポイントの相対パスを使用する必要があります。Flask エンドポイントにアクセスする URL を作成するために、ユーティリティー・メソッド q_url_for および get_console_ip を使用しないでください。コンソールが IP アドレスの代わりに Web URL を使用すると、要求がクロスドメインであるためにすべての Flask 要求は拒否されます。その場合、アプリケーションは機能しないことがあります。例えば、AJAX 呼び出しのために、メイン・アプリケーション・テンプレート・レベルからエンドポイントにアクセスして、何らかの JSON データを返したい場合は、以下の URL フォーマットを使用します。

```
url_cpu_data = 'cpu_data'
```

このフォーマットは、views.py 内のメソッドを経路指定します。

```
@app.route('/cpu_data', methods=['GET'])
```

例えば、深いレベルのフォルダーから作業している場合は、../cpu_data を使用して 1 レベル上に移動してから、このエンドポイントに到達します。

エンドポイントにアクセスする URL を作成するために、以下のフォーマットを使用しないでください。

```
url_cpu_data = "{{ q_url_for('cpu_data') }}"
```

上記の場合と同様に、Web URL から IP アドレスに移動すると、要求がクロスドメインであるためにこの要求は拒否されることがあり、アプリケーションは機能しないことがあります。

URL をスラッシュで開始すると、URL は `https://g-machineIPaddress/` のルートから開始されます。

カスタム Flask メソッド

以下の表は、使用できる Flask カスタム・メソッドについて説明しています。

表 2. カスタム Flask メソッド

メソッド	フォーマット	記述
g_url_for()	def g_url_for(endpoint,append_csrf_token=true, **values):	アプリケーションの views.py または Jinja2 テンプレートの経路の内部でこの Python メソッドを使用します。 このメソッドは、基本的には Flask url_for(..) メソッドを包含するラッパーとして機能します。このメソッドは、アプリケーションのエンドポイントに到達できるように、正しいアプリケーション固有のパス URL 部分に関連する接頭部を適用します。また、必要な CSRF トークンを URL に付加します。Guardium CSRF トークンを参照してください。 Guardium は、URL に CSRF トークンが含まれている要求をブロックします (HTTP 本文の隠されたパラメーターとして転送された場合でもブロックします)。オプション・パラメーター append_csrf_token=true を設定して接頭部および URL を追加したり、append_csrf_token=false を使用して接頭部のみを追加したりすることができます。HTML フォームの例を参照してください。HTTP POST で url_for を使用する場合、システムは、「HTTP エラー 403 アクセス禁止」で応答して、セッションを終了します。 以下のスニペットは、Jinja2 テンプレート内のメソッドを使用して、イメージ・リソースの Guardium URL 接頭部を追加する方法を示しています。 <img src="/dcs/markdown/workspace/Transform/htmlout/0/nl/ja/com.ibm.guardium.doc.admin/ecosystem/{{ g_url_for('static', filename = 'images/come_image.png') }}" width="256" height="256" alt="previous" title="Previous" border="0"> Flask url_for(..) メソッドについて詳しくは、Flask Web サイトを参照してください。
getAppBaseURL()	def getAppBaseUrl():	Guardium を通じて完全な URL を取得する関数であり、該当するアプリケーション・プラグイン・サーブレットへのすべての要求をプロキシー処理します。このルーチンは、アプリケーション内のリソースを参照する URL を作成するために後に付加できる URL ストリングを返します。通常、q_url_for() 関数がこの目的で使用されますが、getAppBaseURL() も便宜を図るために提供されています。

Guardium CSRF トークン

Guardium CSRF (クロスサイト・リクエスト・フォージェリー) トークンは、Guardium によって生成され、すべてのフォーム送信、リンク、または Ajax 要求に追加する必要があります。以下の例に示すように、組み込みユーティリティー関数 grdlib.get_CSRF_token() を使用して、このトークンを HTTP 要求から取得し、URL に追加することができます。

JavaScript 要求

AJAX 要求を送信する場合、その要求に X-CSRF-Token ヘッダーを追加します。例えば、jQuery では、トークンを送信するためのすべての要求を構成できます。

```
<script type="text/javascript">
    var csrf_token = "{{ gpylib.get_CSRF_token() }}";

    $.ajaxSetup({
        beforeSend: function(xhr, settings) {
            if (!/^(GET|HEAD|OPTIONS|TRACE)$/i.test(settings.type) && !this.crossDomain) {
                xhr.setRequestHeader("X-CSRF-Token", csrf_token);
            }
        }
    });
</script>
```

HTML フォーム

フォームでトークンと共に隠された入力をレンダリングします。

```
<form method="post">
    <input type="hidden" name="org.apache.catalina.filters.CSRF_NONCE" value="{{ gpylib.get_CSRF_token() }}"/>
    <input type="submit" value="Submit">
</form>
```

フォームを別のページに送信する場合は、相対リンクを使用するか、g_url_for を呼び出してください。

```
<form method="post" action={{g_url_for("differentPage.html",append_csrf_token=false)}}>
    <input type="hidden" name="org.apache.catalina.filters.CSRF_NONCE" value="{{ gpylib.get_CSRF_token() }}"/>
    <input type="submit" value="Submit">
</form>
```

リンク

```
<a href={{g_url_for("debug_view" }}">Show Logs</a>
```

親トピック: SDK の処理

Python ヘルパー・ライブラリー関数

Guardium Python ヘルパー・ライブラリー (gpylib) には、ロギングを追加し、REST API 呼び出しを行い、JSON オブジェクトを Python ディクショナリーに変換するために使用できるいくつかの役立つ関数が含まれています。

アプリケーションの views.py ファイル内にインポートするすべての関数をグローバルに呼び出すことができます。

以下の表は、アプリケーションの views.py ファイル内にインポートできる関数について説明しています。

機能	フォーマット	記述

log(0)	`def log(message, level='info'):`
	
	以下に例を示します。
	
	`from gpylib import gpylib`
	`..`
	`#in precedence order from lowest level to highest`
	`log('debug message' ,'debug')`
	`log('info message' ,'info')`
	`log('warning message' ,'warning')`
	`log('error message' ,'error')`
	`log('critical message' ,'critical')| Copy Copy`
		log() 関数を使用するには、アプリケーションの views.py 内に gpylib ヘルパー・ライブラリーをインポートします。この関数は、選択したログ・レベルのメッセージを /store/log/app.log ファイルに書き込みます。デフォルトでは、ロギングはオンになっていて、INFO レベルに設定されます。これより低いレベルのロギング・メッセージは無視されます。変更するには、POST /log_level エンドポイントを使用します。

set_log_level(log_level)	`def set_log_level(log_level='info'):`	現在のログ・レベルを設定します。POST /log_level エンドポイントによって使用されますが、プログラマチックに呼び出すこともできます。

REST()	`def REST( RESTtype, requestURL, headers={},`
	`data=None, params=None, json=None,`
	`version=None ):`
	
	以下に例を示します。
	
	`try:`
	`  headers = {'content-type' : 'text/plain'}`
	`  arielOptions = gpylib.REST( 'get',`
	`  '/api/ariel/databases',`
	`headers = headers )`
	`except Exception as e:`
	`  gpylib.log( "Error "  + str(e) )`
	`  raise`
		この関数を使用して Guardium REST API エンドポイントへの呼び出しを行うために、gpylib ライブラリーをインポートします。エンドポイントは、Guardium から要求で渡されるセキュリティー・トークンを再使用して、認証および許可を処理します。

機能	フォーマット	記述
to_json_dict(JSON)	`def to_json_dict(python_obj):`	JSON オブジェクトを Python ディクショナリーに変換します。

- GrdAPI クラスと GdrConnection クラス

親トピック: SDK の処理

GrdAPI クラスと GdrConnection クラス

GRDApi の自動生成される Python API は、ファイル ¥myNewApp¥app¥guardapi¥GRDApi.py にあります。

GrdConnection クラスは、Guardium マシンへのアクティブな接続を保持します。

GrdApi クラスは、SDK でユーザーが実行できる、利用可能な Guardium 操作をすべて保持します。

GrdAPI 関数を使用する前に、以下のコマンドを使用して、Guardium システムへの接続を確立します。

```
grd_connection = GrdConnection(GMachineIP, GMachinePort, username, password, client_secret, client_id)

grd_api = GRDApi(grd_connection)
```

引数を指定せずに GrdConnection インスタンスを作成すると、アプリケーションをその環境で初めて実行するときに、必要なすべての情報はコマンド行で要求されます。

これで、grd_api オブジェクトにアクセスすることによって、GRDApi 関数を使用できます。例えば、create_online_report() API 関数を呼び出して、「Guardium ログイン」レポートを作成する場合は、以下を入力します。

```
result = grd_api.create_online_report('Guardium logins', reportParameter=report_parameters)
```

以下に、JSON 形式で出力するレポートの実行例を示します。

```
sql_errors = grd_api.create_online_report('SQL Errors', 'NOW -10 HOUR', 'NOW +3 HOUR')
```

実際の使用例については、『REST API サンプル・アプリケーション』を参照してください。

上級ユーザーのみ

コマンド `regenerate_grd_api_lib` は、GrdAPI クラスを再ビルドし、既存のライブラリーを上書きします。ある Guardium バージョンで作成したアプリケーションを別のバージョンで使用する場合に、このコマンドを使用します。例えば、`regenerate_grd_api_lib-w workspace` などです。

親トピック: Python ヘルパー・ライブラリー関数
関連情報:
GuardAPI リファレンス

Jinja2 テンプレート

Jinja2 は Python ライブラリーです。これを使用すると、コア・テンプレート・テキスト・ファイルから各種出力フォーマットのテンプレートを作成できます。Jinja2 は、Guardium アプリケーション用の HTML テンプレートを作成するために使用できます。

Jinja2 には、高度な API が含まれており、またテンプレート・ファイルにコンテンツを動的に挿入できる構文ディレクティブ (ステートメント、式、変数、タグ) が多数含まれます。

アプリケーションの views.py ファイルで Flask render_template() メソッドを使用すると、ルートで提供される Python メソッドからのデータを Jinja2 テンプレート HTML ファイルに挿入できます。例:

```
__author__ = 'IBM'

from flask import render_template
from app import app

@app.route('/')
def hello_world():
    return render_template("hello.html", title = "Guardium")
```

hello.html テンプレートは /app/templates フォルダーに保管する必要があります。hello.html ファイルは、以下のセクションで示されます。

```
<!doctype html>
<title>Hello from Flask</title>
<h1>Hello {{ title }}!</h1>
```

テンプレートにより、以下の出力が生成されます。

```
<!doctype html>
<title>Hello from Flask</title>
<h1>Hello Guardium</h1>
```

注: アプリケーションの Jinja2 テンプレート内で Flask Jinja2 強制の url_for 機能は使用しないでください。Guardium GUI Application Framework では、要求パスに相対アドレッシングが使用されます。url_for を使用する場合、コンテナー自体から絶対要求パスが作成されます。

Jinja2 テンプレートについて詳しくは、Jinja2 の資料を参照してください。

Eclipse での Jinja2 テンプレートの編集

Eclipse の Django テンプレート・エディター・プラグインを使用して Jinja2 テンプレートを作成できます。

PyDev Eclipse には、デフォルトでは Jinja2 テンプレート・エディターは付属しません。Django テンプレート・エディター・プラグインには、アプリケーション用の Jinja2 テンプレートを作成するために使用できる便利な機能が備わっています。

Eclipse のメインの「ヘルプ」パネルで「ヘルプ」 > 「新規ソフトウェアのインストール」をクリックして Django リポジトリーをインストールします (http://pydev.org/updates)。

このプラグインには、Jinja2 テンプレート作成のための便利な構文強調表示機能および自動入力機能が備わっています。

- テンプレートへの JavaScript ライブラリーの統合
  アプリケーションのユーザー・インターフェースをスタイル設定したり拡張したりするために、HTML テンプレートに CSS および JavaScript ライブラリーを追加します。

親トピック: SDK の処理

テンプレートへの JavaScript ライブラリーの統合

アプリケーションのユーザー・インターフェースをスタイル設定したり拡張したりするために、HTML テンプレートに CSS および JavaScript ライブラリーを追加します。

CSS スタイル設定、ウィジェット、およびその他の UI 機能をアプリケーションに追加するために、Guardium に統合される JavaScript ライブラリー Dojo および JQuery を使用できます。

Dojo

JavaScript の Dojo ツールキットをアプリケーションの HTML テンプレートに統合するには、次のタグをテンプレートの HEAD エレメントに追加します。

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>My app!</title>
   <link type="text/css" rel="stylesheet"

href="#_dcs_markdown_workspace_Transform_htmlout_0_nl_ja_com.ibm.guardium.doc.admin_ecosystem_console_idt_dojo_resources_dojo.css"
></link>
   <link type="text/css" rel="stylesheet"

href="#_dcs_markdown_workspace_Transform_htmlout_0_nl_ja_com.ibm.guardium.doc.admin_ecosystem_console_idt_dijit_themes_dijit.css">
</link>
   <link type="text/css" rel="stylesheet"

href="#_dcs_markdown_workspace_Transform_htmlout_0_nl_ja_com.ibm.guardium.doc.admin_ecosystem_console_idt_dijit_themes_claro_claro
.css"></link>
   <link type="text/css" rel="stylesheet"

href="#_dcs_markdown_workspace_Transform_htmlout_0_nl_ja_com.ibm.guardium.doc.admin_ecosystem_console_idt_idx_themes_oneui_oneui.c
ss"></link>
  <script type="text/javascript"
   src="/dcs/markdown/workspace/Transform/htmlout/0/nl/ja/com.ibm.guardium.doc.admin/ecosystem/console/idt/dojo/dojo.js"
   data-dojo-config="async:true, parseOnLoad: true"></script>
  </head>

<body></body>

</html>
```

Guardium では Dojo 1.9.3 が使用されます。

JQuery

JavaScript ライブラリーの JQuery をアプリケーションの HTML テンプレートに統合するには、次のタグをテンプレートの HEAD エレメントに追加します。

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>My app!</title>
  <script type="text/javascript"

src="/dcs/markdown/workspace/Transform/htmlout/0/nl/ja/com.ibm.guardium.doc.admin/ecosystem/console/core/js/jquery/jquery.min.js">
</script>
  </head>

<body></body>

</html>
```

親トピック: Jinja2 テンプレート

アプリケーションのログ

アプリケーションのログは、アプリケーションのコンテナーの /store/log ディレクトリーに保管されます。

/store/log ディレクトリーには、2 つのログ・ファイルが格納されます。

- startup.log は、アプリケーションの初期始動のログです。 このログは、アプリケーションの app/src_deps/ フォルダーに追加される依存関係のインストールをチェックする場合に便利です。

- app.log は、gpylib ライブラリーによって作成されるログ・ファイルです。gpylib.log() メソッドの呼び出しのログは、この app.log ファイルに書き込まれます。

- アプリケーションへのロギングの追加
  Guardium Python ヘルパー・ライブラリー (gpylib) には、アプリケーションにロギングを追加するために使用できる便利な 2 つの関数が含まれています。
- アプリケーション・ログの表示
  ファイル・サーバーの CLI コマンドを使用して、Web ブラウザーから Guardium アプライアンスへの接続を許可します。これにより、アプリケーション・ファイルをダウンロードできます。

親トピック: SDK の処理

アプリケーションへのロギングの追加

Guardium Python ヘルパー・ライブラリー (gpylib) には、アプリケーションにロギングを追加するために使用できる便利な 2 つの関数が含まれています。

log() 関数

log() 関数を使用するには、アプリケーションの views.py 内に gpylib ヘルパー・ライブラリーをインポートします。この関数は、選択されたログ・レベルのメッセージを /store/log/app.log ファイルに書き込みます。

デフォルトでは、ロギングはオンになっており、*INFO* レベルに設定されています。これより低いレベルのロギング・メッセージは無視されます。ログ・レベルを変更するには、POST /log_level エンドポイントを使用します。

log() 関数では、以下のフォーマットが使用されます。

```
def log(message, level='info'):
```

例:

```
from gpylib import gpylib

..

#in precedence order from lowest level to highest

gpylib.log('debug message' ,'debug')

gpylib.log('info message' ,'info')

gpylib.log('warning message' ,'warning')

gpylib.log('error message' ,'error')

gpylib.log('critical message' ,'critical')
```

注:

アプリケーションの views.py ファイルにインポートするすべての gpylib 関数は、グローバルに呼び出すことができます。このため、他の関数との衝突を回避するために、インポートする gpylib 関数に名前空間を追加することをお勧めします。

set_log_level() 関数

この関数を使用すると、現行のログ・レベルを設定できます。この関数は、POST /log_level エンドポイントで使用されますが、プログラムで呼び出すこともできます。

```
def set_log_level(log_level='info'):
```

親トピック: アプリケーションのログ

アプリケーション・ログの表示

ファイル・サーバーの CLI コマンドを使用して、Web ブラウザーから Guardium アプライアンスへの接続を許可します。これにより、アプリケーション・ファイルをダウンロードできます。

このタスクについて

重要: ファイル・サーバーの稼働中は、ログインしなくても Guardium ログおよびその他の製品固有のファイルが Guardium サーバーで使用可能になります。 ファイルのダウンロードが完了した後、直ちにファイル・サーバーをオフにしてください。
ヒント: ファイル・サーバーは 800 秒後に自動的に停止します。タイムアウト値を変更するには、CLI コマンド store timeout fileserver_session M を使用します。ここで、M は、ファイル・サーバーがタイムアウトするまでの 60 から 1200 までの秒数です。

手順

1. CLI にログインします。
2. コマンド fileserver <ip_address> を入力します。ここで、IP はファイル・サーバーへの接続をオープンするコンピューターの IP アドレスです。IP アドレスを入力しない場合、CLI への接続をオープンしたコンピューターのみが、ファイル・サーバーへの接続を許可されます。物理 Guardium アプライアンスで CLI にアクセスする場合、IP アドレスは必須です。 システムは次のように応答します。
   ```
   xxxx.ibm.com> fileserver Creating the index file. Starting the file server. You can find it at http://xxxx.ibm.comPress ENTER to stop the file server
   ```
   Web サーバーがコレクターで起動し、ブラウザーからログ・ファイル・ディレクトリーにアクセスできます。
3. 「Sqlguard logs」をクリックします。アプライアンスのログ・ディレクトリーに、ファイル・リストがあります。任意のファイルをクリックしてダウンロードできます。最上部の行にある /var/log/guard のリンクはクリックしないでください。これは、現行ページです。ファイルおよびディレクトリーへのその他のリンクは正

常に機能します。

親トピック: アプリケーションのログ

Guardium UI でのアプリケーションの操作

アプリケーションのアップロード方法、管理方法、および出力の表示方法について説明します。

- アプリケーション・ライフサイクルのユーザーおよびユーザー権限に関するガイドライン
  アプリケーションのアップロードと実行に必要な 3 種類のユーザーについて説明します。
- アプリケーションのアップロードおよび管理
  アプリケーションを 1 つ以上の Guardium システムにデプロイし、アプリケーションを実行および管理します。
- アプリケーション出力の表示
  インストールされているアプリケーションはすべて、「Guardium アプリケーション」の下にあるナビゲーション・メニューに表示されます。

親トピック: Guardium アプリケーション

アプリケーション・ライフサイクルのユーザーおよびユーザー権限に関するガイドライン

アプリケーションのアップロードと実行に必要な 3 種類のユーザーについて説明します。

アプリケーションのアップロードと実行には、以下の 3 種類のユーザーが関与します。

- アプリケーションをアップロードするユーザー。通常は admin ですが、**エコシステム・アプリケーション・ライフサイクル**・ロールの権限を持つ任意の Guardium ユーザーにすることができます。このユーザーは、「設定」 > 「ツールとビュー」 > 「アプリケーション・ライフサイクル」にアクセスできます。
- アプリケーションを (Guardium 内で) 実行するユーザー。 「設定」 > 「ツールとビュー」 > 「アプリケーション・ライフサイクル」ページでアプリケーションをインストールするときに、このユーザーを指定します。このユーザーには、GrdAPI、およびアプリケーションに必要なすべての機能に対するアクセス権が必要です。アプリケーションは、アプリケーションをアップロードしたユーザーで実行しないでください。これはベスト・プラクティスに反するものです。アプリケーションごとにユーザーを作成して、各アプリケーション・ユーザーに必要最小限のロールを付与することを強く推奨します。アプリケーション開発者は、パブリッシュしたアプリケーションに必要なアクセス権を文書化してください。
- ナビゲーション・メニューの「Guardium アプリケーション」でアプリケーション出力を表示するユーザー。「設定」 > 「ツールとビュー」 > 「アプリケーション・ライフサイクル」の「ロールの割り当て」ボタンを使用して、セキュリティー・ロールを選択します。指定されたセキュリティー・ロールの 1 つ (以上) を持つユーザーは、アプリケーション出力を表示できます。
  注意:
  アプリケーションでは、アプリケーションのアクセス権がなければアプリケーション・ビューアーに対してブロックされていた機能やデータが、アプリケーションのアクセス権を使用してアプリケーション・ビューアーに公開される可能性があります。

親トピック: Guardium UI でのアプリケーションの操作
関連概念:
アクセス管理の概要

アプリケーションのアップロードおよび管理

アプリケーションを 1 つ以上の Guardium システムにデプロイし、アプリケーションを実行および管理します。

始める前に

- Guardium UI (「設定」 > 「ツールとビュー」 > 「アプリケーション・ライフサイクル」) でアプリケーション・ライフサイクル・ページを有効にするには、次の CLI コマンドを実行します。
  store system ecosystem on
  注意:
  コマンド grd_sdk deploy を使用して、実稼働環境にアプリケーションをデプロイしないでください。
- アプリケーション・ライフサイクルのユーザーおよびユーザー権限に関するガイドラインを参照してください。

重要: CM で複数のアプリケーションを実行しないでください。また、コレクターで複数のアプリケーションを実行しないでください。個々のシステムで複数のアプリケーションを実行すると、問題が発生する可能性があります。

手順

1. 「セットアップ」 > 「ツールとビュー」 > 「アプリケーション・ライフサイクル」にナビゲートします。
2. アプリケーションをアップロードします。
   a.  をクリックし、アプリケーションの .zip ファイルまたは .gz ファイルを参照して選択します。
   b. ユーザーおよびパスワードを入力します。これは、Guardium 内でアプリケーションを実行する Guardium ユーザーです。
      注: このユーザーには、アプリケーション内のアクションに対する権限が必要です。権限がない場合、アプリケーションを完全には実行できません。
   c. 「インストール先」で、1 つ以上の Guardium システムを選択します。
   d. 「アップロード」をクリックします。
3. 1 つ以上のアプリケーションにロールを割り当てます。これらのロールは、アプリケーション出力を表示するユーザーであるため、アクセス権限が制限されている必要があります。
   a. アプリケーション (1 つまたは複数) が停止していることを確認し、「ロールの割り当て」をクリックします。
   b. アプリケーション (1 つまたは複数) を選択して、「ロールの割り当て」をクリックします。
   c. ロールを選択して、「OK」をクリックします。
4. 1 つ以上のアプリケーションを選択して、「開始」または「停止」をクリックし、それらのアプリケーションを開始または停止します。
5. アプリケーションのより新しいバージョンに更新します。
   a. 「更新」をクリックします。

b. .zip ファイルまたは .gz ファイルを参照して選択します。
c. アプリケーションのインストール時に指定したのと同じユーザーおよびパスワードを入力して、「アップロード」をクリックします。
d. 「状況」列でインストール状況を確認します。

6. 1 つのアプリケーションに対するアクションのログを開くには、そのアプリケーションを選択して、「アクション・ログ」をクリックします。
7. 資格情報を変更するには、1 つ以上のアプリケーションを選択して、「資格情報を更新」をクリックし、新しいユーザーとパスワードを入力します。
8. 「アプリケーション・ライフサイクル」ページの情報の CSV をダウンロードするには、「CSV 形式でダウンロード」をクリックします。
9. オプションで、アプリケーションの行の「アプリケーション・ログ」列で  をクリックして、アプリケーションのアクティビティーのログをダウンロードします。

親トピック: Guardium UI でのアプリケーションの操作
関連情報:
システム CLI コマンド

アプリケーション出力の表示

インストールされているアプリケーションはすべて、「Guardium アプリケーション」の下にあるナビゲーション・メニューに表示されます。

手順

1. 「Guardium アプリケーション」をクリックして、マシンにインストールされているアプリケーションのメニューを開きます。
2. アプリケーションのタイトルをクリックして、その名前をクリックし、出力を表示します。

親トピック: Guardium UI でのアプリケーションの操作

アプリケーションに関する FAQ

Guardium システムでアプリケーションが問題を起こしているようです。

Guardium システムでエコシステムが問題を引き起こしていると思われる場合は、次のようにします。

次の CLI コマンドを使用してエコシステムをオフにします。
store system ecosystem off
次のコマンドを使用して、オンに戻すことができます。
store system ecosystem on

Guardium システムでは、アプリケーションをいくつ実行できますか?

CM で 1 つのアプリケーションを実行でき、コレクターで 1 つのアプリケーションを実行できます。

アプリケーションを複数の Guardium アプライアンスで同時に実行できますか?

いいえ。どの時点においても、一度に 1 つの Guardium アプライアンスのみアプリケーションを実行してください。

Guardium からログアウトされるのはなぜですか?

HTTP 40x および 50x のエラーが発生すると、Guardium は常に現行セッションをクローズします。HTTP post で URL は許可されないことに注意してください。HTTP エラー 403 が発生します。Guardium CSRF トークンを参照してください。その他のエラーについては、ログ・ファイルを参照してください (ログ・ファイルはどこにありますか?)。

/store フォルダーに保存されたデータは、アプリケーションまたはマシンの再始動後も保持されますか?

Guardium システムでは、アプリケーションまたはマシンの再始動後も /store フォルダーは保持されます。このフォルダーは、Guardium システムに保管されている Docker 「ボリューム」にマウントされます。 ただし、Docker (grd_sdk run -d) を使用してアプリケーションをローカルでテストするたびに、新規ボリュームが割り当てられます (また、ローカルで実行されるたびに新しい Docker イメージが作成されます)。

ログ・ファイルはどこにありますか?

- UI から、「設定」 > 「ツールとビュー」 > 「アプリケーション・ライフサイクル」にナビゲートして、アプリケーションの行の「アプリケーション・ログ」列で  をクリックし、アプリケーションのアクティビティーのログをダウンロードします。
- ファイル・サーバー・コマンド = /store/log。アプリケーションのログを参照してください。
- サポート関数で /debug および /debug_view を参照してください。

Docker コンテナーを介してアプリケーションを実行すると、アプリケーションが停止しません。なぜですか。

キープアライブ・パケットの送信や定期的なガーベッジ・コレクションの実行などのバックグラウンド・タスクを実行するスレッドには、`thread.daemon=True` を設定する必要があります。デーモン・スレッドとして設定することにより、これらのスレッドは、プログラムの終了時に自動的に強制終了されます。

Windows で Docker Toolbox を使用してアプリケーションを実行中に、「[エラー -2] 名前またはサービスが不明です ([Error -2] Name or service not known)」というエラーが発生します。

ご使用の Docker Toolbox 環境で DNS 解決の問題が発生しています。DNS サービスを修正するか、ホスト名の代わりに Guardium マシンの IP アドレスを使用してください。

アプリケーションの実行時に、「keyring.backends._OS_X_API.Error: (-25293, "システムからパスワードを取り出せません") (keyring.backends._OS_X_API.Error: (-25293, "Can't fetch password from system"))」というエラーが発生します。

このエラーは、SDK がパスワード・マネージャーにアクセスできないことを示します。

以下の手順を実行します。

1. `pip2 uninstall <package name>` を入力して、使用していない Python パッケージをアンインストールします。
2. `brew uninstall python` を入力して、Python をアンインストールします。
3. `xcode-select --install` を入力して、開発ツールを有効にします。
4. `brew install python@2` を入力して、Python を再インストールします。
5. SDK を再インストールします。SDK のインストールを参照してください。

別の方法として、`codesign -f -s - /usr/local/etc/GuardiumAppSDK/guard_sdk_venv/bin/python` を入力することで、SDK Python 環境にトラステッドのマークを付けます。

親トピック: Guardium アプリケーション

リソース

IBM GuardiumGUI Application Framework でのアプリケーション作成を支援する各種リソースを使用します。

Flask API

- http://www.flaskapi.org/ (http://www.flaskapi.org/)
- Flask チュートリアル (https://blog.miguelgrinberg.com/post/the-flask-mega-tutorial-part-i-hello-world)

Jinja2 テンプレート

- http://jinja.pocoo.org/docs/dev/ (http://jinja.pocoo.org/docs/dev/)

親トピック: Guardium アプリケーション

問題のトラブルシューティング

IBM 製品の問題を切り分けて解決するために、トラブルシューティングとサポートの情報を使用することができます。 この情報には、IBM Guardium を含む IBM 製品に付属する問題判別リソースの使用方法が掲載されています。

- 問題のトラブルシューティング手法
  *トラブルシューティング*は、問題解決のための系統的なアプローチです。 トラブルシューティングの目的は、ある部分が予期したとおりに機能しない理由および問題を解決する方法を判別することです。確立されている一般的な手法でタスクのトラブルシューティングを行うことができます。
- 問題および解決策
  このトピックで、発生した問題の解決策を検索してください。

問題のトラブルシューティング手法

*トラブルシューティング*は、問題解決のための系統的なアプローチです。 トラブルシューティングの目的は、ある部分が予期したとおりに機能しない理由および問題を解決する方法を判別することです。確立されている一般的な手法でタスクのトラブルシューティングを行うことができます。

トラブルシューティング・プロセスの最初のステップは、問題を完全に記述することです。問題を記述することで、ユーザーと IBM 技術サポート担当者が、問題の原因をどこから探し始めるか認識しやすくなります。このステップでは、次の基本的な質問をご自身で検討します。

- 問題の症状はどのようなものか。
- 問題が発生する場所はどこか
- 問題が発生したのはいつか。
- どのような条件下で問題が発生するか
- 問題を再現できるか。

通常は、これらの質問に回答することで問題が適切に記述され、問題解決につながります。

問題の症状はどのようなものか。

問題は何か。この質問は単純なように思われますが、これをいくつかのさらに絞り込んだ質問に分解し、問題をさらに具体的に記述することができます。次のような質問が考えられます。

- 誰が、または何が問題を報告しているか。
- どのようなエラー・コードまたはメッセージが出ているか。
- どのような障害がシステムに起こったか。例えば、ループ、ハング、異常終了、性能低下、結果が正しくない、など。

問題が発生する場所はどこか

問題がどこで発生しているかの判断は、簡単にできるとは限りませんが、問題解決のための最も重要なステップの 1 つです。問題を報告しているコンポーネントと障害が起こっているコンポーネントの間には、多数のテクノロジー層が存在することがあります。問題を調査するときは、ネットワーク、ディスク、ドライバーを始めとして多くのコンポーネントを考慮する必要があります。

問題が発生している部分に焦点を当てて問題となっているレイヤーを切り分ける上で、次の質問が役立ちます。

- 問題は 1 つのプラットフォームまたはオペレーティング・システムに固有か、それとも複数のプラットフォームまたはオペレーティング・システムに共通か。
- 現在の環境および構成がサポートされているか。
- ユーザー全員に問題が発生しているか。
- (マルチサイト・インストール済み環境の場合。)すべてのサイトに問題が発生しているか。

ある層で問題が報告されたとしても、必ずしもその層内で問題が発生しているとは限りません。問題がどこで発生したかを突き止めるには、問題が存在する環境を理解することが不可欠です。しばらく時間を割いて、問題の環境を完全に記述してください。これにはオペレーティング・システムとそのバージョン、対応するすべてのソフトウェアとそのバージョン、およびハードウェア情報を含める必要があります。サポートされている構成の環境で実行していることを確認してください。問題の多くは、ソフトウェアのレベルが非互換 (一緒に実行することが意図されていないソフトウェアまたはその組み合わせでのテストが完全になされていないソフトウェア) であることが原因で生じている可能性があります。

問題が発生したのはいつか。

障害に至るまでのイベント (特に発生が 1 回限りのイベント) の詳しい時系列表を作成してください。最も簡単に時系列表を作成する方法は、逆方向にたどることです。エラーが報告された時点 (ミリ秒単位に至るまで可能な限り精密に) から開始して、使用可能なログと情報を通じて逆方向にたどります。通常、確認する必要があるのは、診断ログで見つけた最初の疑わしいイベントまでの部分のみです。

イベントの詳細な時刻表を作成するには、以下の質問に答えてください。

- 問題が発生するのは、日中または夜間の特定の時刻のみか。
- 問題が発生する頻度はどの程度か。
- 問題が報告された時刻までにイベントがどのような順序で発生したか
- 問題が発生したのは環境変更 (ソフトウェアまたはハードウェアのアップグレードまたはインストールなど) の後か。

このような質問に回答することで、問題を調査するための基準枠を設定できます。

どのような条件下で問題が発生するか

問題が発生したときに実行中だったシステムおよびアプリケーションがどれかを知ることは、トラブルシューティングの重要部分です。お客様の環境に関する以下の質問は、問題の根本原因を識別するために役立ちます。

- 問題は同じタスクの実行中に常に起こるか
- 特定の一連のイベントが発生した場合にのみ、その問題が発生するか
- ほかのアプリケーションにも同時に障害が起こるか。

これらのタイプの質問に回答することは、問題が発生している環境を説明し、依存関係の相関付けをするのに役立ちます。ほぼ同時に複数の問題が発生したとしても、それらの間に関連があるとは限らないことに注意してください。

問題を再現できるか。

トラブルシューティングの観点からすると、理想的な問題とは、再現できる問題であるということです。通常、問題を再現できる場合は、調査に役立てるために自由に使用できるツールまたは手順の数が多くなります。そのため、再現できる問題は多くの場合、デバッグや解決がより容易です。

しかし、問題を再現できることが、デメリットになることもあります。問題がビジネスに大きな影響を及ぼす場合は、問題が再発するのは望ましくありません。可能な場合は、テスト環境または開発環境で問題を再現してください。通常、そのような環境では、より柔軟で制御の利いた調査ができます。

- 問題をテスト・システムで再現することができるか。
- 複数のユーザーまたはアプリケーションで、同じタイプの問題が検出されているか。
- 1 つのコマンド、一連のコマンド、または特定のアプリケーションを実行することで、問題を再現できるか。

- Fix Central からのフィックスの入手
  Fix Central を使用して、Guardium を含むさまざまな製品について、IBM サポートが推奨するフィックスを見つけることができます。Fix Central では、ご使用のシステム用のフィックスを検索、選択、注文、およびダウンロードすることができ、その際に配信オプションを選択できます。以下は英語のみの対応となります。 お客様の問題の解決に、プロダクトのフィックスが有効な場合があります。
- IBM サポートへの問い合わせ
  IBM サポートでは、製品の問題に関する支援や、よくある質問への回答、製品の問題のユーザーによる解決のための支援を提供します。
- IBM サポートのための基本情報
  IBM サポートに連絡する前に、IBM Guardium (コレクター、アグリゲーター、中央マネージャー、UNIX/Linux S-TAP、Windows S-TAP) に関する基本情報を収集します。
- IBM との情報の交換
  問題を診断または特定するために、システムのデータおよび情報を IBM サポートに提供する必要がある場合があります。問題判別に使用するツールまたはユーティリティーを IBM サポートから提供される場合もあります。
- サポート更新のサブスクライブ
  使用する IBM 製品に関する重要な情報を常に入手するために、更新にサブスクライブできます。

親トピック: 問題のトラブルシューティング

Fix Central からのフィックスの入手

Fix Central を使用して、Guardium を含むさまざまな製品について、IBM サポートが推奨するフィックスを見つけることができます。Fix Central では、ご使用のシステム用のフィックスを検索、選択、注文、およびダウンロードすることができ、その際に配信オプションを選択できます。以下は英語のみの対応となります。 お客様の問題の解決に、プロダクトのフィックスが有効な場合があります。

このタスクについて

手順

フィックスを見つけてインストールするには、次のようにします。

1. フィックスを入手するために必要なツールを取得します。インストールされていない場合は、 製品の更新インストーラーを入手します。このインストーラーは、Fix Central からダウンロードできます。 このサイトには、更新インストーラーのダウンロード、インストール、および構成の手順が示されています。
2. 製品として Guardium を選択し、 解決する問題に関連したチェック・ボックスを 1 つ以上選択します。
3. 必要なフィックスを特定して選択します。
4. フィックスをダウンロードします。
    a. ダウンロード・マニュアルを開き、Download package セクション内のリンクに従います。
    b. ファイルをダウンロードするときに、メンテナンス・ファイルの名前が変更されていないことを確認してください。 この変更は意図的である場合や、特定の Web ブラウザーやダウンロード・ユーティリティーに起因する意図しない変更である場合があります。
5. フィックスを適用します。
    a. ダウンロード資料の『Installation Instructions』セクションに記載されている説明に従ってください。
    b. 詳しくは、製品資料のトピック『更新インストーラーを使用したフィックスのインストール』を参照してください。
6. オプション: フィックスおよびその他の IBM サポートの更新に関する電子メール通知を毎週受信するようにサブスクライブします。

親トピック: 問題のトラブルシューティング手法

IBM サポートへの問い合わせ

IBM サポートでは、製品の問題に関する支援や、よくある質問への回答、製品の問題のユーザーによる解決のための支援を提供します。

始める前に

技術情報などのその他の自己解決型の選択肢を使用して答えまたは解決策を見つけようとした後に、IBM サポートに問い合わせることができます。 IBM サポートにお問い合わせいただくには、会社または組織が有効な IBM *保守契約名*を保持し、お問い合わせいただくユーザーが IBM に問題を送信する権限を持っている必要があります。利用できるサポートの種類については、「"*Software Support Handbook*"」の『Support portfolio』トピックを参照してください。

手順

問題について IBM サポートに問い合わせるための手順は以下のとおりです。

1. 問題を明確にし、バックグラウンド情報を収集して、問題の重大度を判別します。 詳細については、「*Software Support Handbook*」の『Getting IBM support』のトピックを参照してください。
2. 診断情報を収集します。
3. 以下のいずれかの方法で、IBM サポートに問題を報告します。
    - IBM サポート・ポータルからオンラインで報告:「サービス・リクエスト」ページの「サービス・リクエスト」ポートレットから、お客様のすべてのサービス要求を開いて更新、表示することができます。
    - 電話: お客様の地域の連絡先電話番号については、Directory of worldwide contacts の Web ページを参照してください。

タスクの結果

ユーザーが提出した問題が、ソフトウェア障害または資料の不正確や欠落が原因である場合、IBMサポートがプログラム診断依頼書 (APAR) を作成します。 APAR では問題を詳細に記述します。 IBM Support は、APAR が解決されてフィックスが 配信されるまで、ユーザーが実施できる次善策を可能な限り提供します。 IBM は、解決された APAR を IBM サポート Web サイトに毎日公開し、同じ問題を経験した他のユーザーが、同じ解決方法を利用できるようにしています。

親トピック: 問題のトラブルシューティング手法

関連情報:

サポート・チケット (PMR) にデータをアップロードする方法 (ビデオ)

Guardium のトラブルシューティングおよびサポート (ビデオ)

IBM サポートのための基本情報

IBM サポートに連絡する前に、IBM Guardium (コレクター、アグリゲーター、中央マネージャー、UNIX/Linux S-TAP、Windows S-TAP) に関する基本情報を収集します。

support must_gather commands を使用します。これを CLI を通じて実行すると、任意の Guardium システムの状態に関する特定の情報を生成することができます。この情報は、Guardium GUI を介して収集することもできます。

この情報は、問題管理レポート (PMR) が記録されているときであればいつでも、Guardium システムからアップロードして IBM サポートに送信できます。

サポート情報の結果の収集

サポート情報を収集するには、「管理」 > 「メンテナンス」 > 「サポート情報の収集」をクリックします。以下のセクションの内容を実行します。

1. サポート情報収集セッションを記述します。
2. PMR 番号を入力します。
3. E メール・アドレスに結果を送信するには、「E メール:」を指定し、E メール・アドレスを入力します。
4. カレンダー・アイコンをクリックして、開始時刻をスケジュールします。
5. 以下のカテゴリーに関連する Must Gather ログ情報をチェックします。
    - 統合
    - ユーザー・インターフェース
    - backup

- データベース・ユーザー
- Scheduler
- システム DB
- ネットワーク
- 適用状態
- モニター・エージェントのデプロイ
- ネイティブ監査
- データ・マイニング
- ELB
- アラート
- 監査
- 中央マネージャー
- パージ
- スニファー
- パッチ・インストール
- 高度な脅威スキャン
- 資格最適化
- クイック・スタート・コンプライアンス・モニター
- エコシステム
- ビッグデータ

6. 情報を収集する対象となる特定の期間を示す値 (分) を入力します。デフォルト値は 10 分です。この値は、ログが収集される期間です。E メールを指定した場合、プロセスを開始した時刻から 10 分間ログが収集され、その後 E メールが送信されます。問題のトラブルシューティングに必要なデバッグ情報が、ログに含まれるようにするために、問題を再現し、指定された期間中のログ情報を生成する必要があります。
7. 結果ログ・ファイルに表示される最大行数を入力します。
8. 構成が終了したら、「開始」をクリックします。
9. 「サポート情報の結果」に移動して、結果を表示します。.tgz ファイルを開くか、または保存することができます。

CLI を使用した Guardium アプライアンスの Must Gather

IBM Guardium のコレクター、アグリゲーター、中央マネージャー

must_gather コマンドは、ユーザーが CLI を通じていつでも実行できます。以下の手順を実行します。

1. 問題のコレクター、アグリゲーターまたは中央マネージャーに対して Putty セッション (または同様のセッション) を開きます。
2. ユーザー `cli` でログインします。
3. 問題のタイプに応じて、適切な must_gather コマンドを CLI プロンプトに貼り付けます。問題を診断するために、複数の must_gather コマンドが必要となる場合があります。コマンドを、以下のリストに説明と共に示します。
   - support must_gather agg_issues (統合プロセス)
   - support must_gather alert_issues (アラート)
   - support must_gather app_issues (アプリケーション)
   - support must_gather audit_issues (監査プロセス)
   - support must_gather backup_issues (バックアップ・プロセス)
   - support must_gather big_data_issues (バックアップ・プロセス)
   - support must_gather cm_issues (中央マネージャー)
   - support must_gather datamining_issues (データ・マイニング)
   - support must_gather deploy_agents_issues
   - support must_gather deployment_issues
   - support must_gather eagle_eye_issues (脅威検出分析)
   - support must_gather ecosystem_issues
   - support must_gather enterprise_load_balancer_issues
   - support must_gather entitlement_issues (資格最適化)
   - support must_gather miss_dbuser_prog_issues (システム・データベース・ユーザー)
   - support must_gather native_auditing_issues
   - support must_gather network_issues (ネットワーク体系)
   - support must_gather patch_install_issues (パッチのインストールおよびアップグレード)
   - support must_gather purge_issues (パージ・プロセス)
   - support must_gather quick_start_issues
   - support must_gather scheduler_issues (スケジューラー機能)
   - support must_gather sniffer_issues (スニファー機能)
   - support must_gather system_db_info (Guardium システムのデータベース・パフォーマンスまたは操作スペース・パフォーマンス)

   出力は、以下の例のようなファイル名で must_gather ディレクトリーに書き込まれます。

   must_gather/system_logs/.tgz

4. 結果の出力を IBM サポートに送信してください。

   fileserver <ip address> を使用すると、.tgz ファイルをアップロードして、IBM サポートに送信できます。

   E メールでファイルを送信するか、標準的なデータ・アップロードを使用して ECUREP にアップロードします。PMR 番号と、アップロードするファイルを指定します。

UNIX/Linux S-TAP の must gather

guard_diag スクリプトは、Guardium の診断に役立つ統計をサーバー上で作成します。

guard_diag の説明:

診断スクリプト (guard_diag)

概要:

診断スクリプト (guard_diag) は、GUI で S-TAP ロギングのレベルを 7 に設定しているときに、/usr/local/guardium/guard_stap/guard_diag から実行されます。S-TAP を実行しているマシンにこのスクリプトを転送することも可能です。

使用法: ./guard_diag output_dir

スクリプトが S-TAP のインストール場所を自動的に判別できない場合は、場所を尋ねるプロンプトが出されます。実行時間は約 1.5 分です。出力ディレクトリーを指定しない場合、スクリプトは、生成される .tar ファイルを /tmp に格納します。スクリプトが GUI からロギングを実行し、有効化する場合、.tar ファイルは /var/tmp に入れられます。ファイル名は、マシン名および実行時刻/日付から派生しますが、先頭は常に diag.ustap になります。

収集される一般システム・データ:

- Uname -a
- インストールされているカーネル・モジュールのリスト
- 1 つのサイクルの出力
- アップタイム
- プロセッサーの番号とタイプ
- 最新の syslog のダンプ
- Netstat 出力
- IPC リスト
- ディスクの空きの統計
- /etc/services のコピー
- /etc のディレクトリー・リスト
- さまざまなプラットフォーム固有の情報
- /etc/inittab の内容

収集される S-TAP データ:

- S-TAP バージョン
- guard_tap.ini の内容
- K-TAP デバイス・ノードでの ls -l
- A-TAP 診断 (atap_must_gather.sh の出力) (S-TAP が A-TAP トラフィックをキャプチャーしていないを参照)
- Db2 出口診断 (db2_exit_health_check.sh の出力) (S-TAP が Db2 出口トラフィックをキャプチャーしていないを参照)
- S-TAP の 30 秒のトレース
- K-TAP 統計
- インストール・ディレクトリー内のすべてのファイルのリスト
- K-TAP khash
- K-TAP (2) および S-TAP (4) の詳細デバッグ・ログ

既知の問題:

- Tusc がすべての HP-UX オペレーティング・システムにインストールされているわけではないため、S-TAP PID をトレースできません。
- システムに gzip が必ずインストールされているとは限りません。圧縮 (最終的な拡張子は .tar.Z) することを試みますが、失敗した場合は .tar ファイルが出力ディレクトリーに格納されます。
- AIX での Topas 出力には制御コードが含まれており、エディターで開くとほとんど理解できなくなるため、端末で解釈するのが最善です。
- 非 root S-TAP には、診断スクリプトに関するいくつかの問題があります。
- Linux では、/var/log/messages は、root のみが読み取り可能です。
- 一部の Solaris オペレーティング・システムは、正しく構成されていない可能性があり、そのために netstat がエラーを表示します。
- 非 root ユーザーのパスはかなり基本的なものであり、その結果、一部のコマンドはまったく実行されない可能性があります。特に HP-UX の gzip で、この既知の問題が発生します。

サポートされているプラットフォーム:

- Linux
- HP-UX
- AIX
- Solaris

プラットフォーム固有の要件:

- STAP: なし
- Linux: なし
- AIX: topas
- Solaris: top、prtdiag、psrinfo
- HP-UX: tusc

Windows S-TAP の must gather

このスクリプトを実行すると、current ディレクトリー内に以下のテキスト・ファイルが生成されます。

- stap.txt
- tasks.txt
- system.txt
- evtlog.txt または evtlog2008.txt
- reg.txt

注:

1. この診断スクリプトは、どの S-TAP バージョンでも実行できます。
2. 診断スクリプトの名前を diag.bat に変更して、S-TAP のインストール・ディレクトリーの下に配置してください これにより、そのスクリプトを手動で実行できるようになります。診断情報を含むテキスト・ファイルが生成されます。
3. 結果を Guardium L3 Support または Research & Development に送信してください。

このスクリプトは以下のデータを収集します。

- %system%guard_tap.ini の内容
- Guardium S-TAP インストール・ログ
- すべての実行中のタスク
- インストールされている全カーネル・ドライバーのリスト
- システム情報ユーティリティーから収集される OS 情報
- ipconfig /all
- netstat -nao
- データベース・サーバーから Guardium システムへの ping と trace の結果
- guardium_stapr の CPU 使用量
- 全体のシステム CPU 使用量
- guardium_stapr のプロセス・ハンドル・カウントとメモリー使用状況
- S-TAP によって生成されるイベント・ログ・メッセージ
- システム・イベント・ログ・メッセージ
- 以下のレジストリー項目:
  - HKLMSOFTWAREMicrosoftWindowsCurrentVersionUninstall
  - HKLMSYSTEMCurrentControlSetServices
  - HKLMSYSTEMCurrentControlSetControlGroupOrderList
  - HKEY_LOCAL_MACHINESOFTWAREMicrosoftMSSQLServer

Must Gather の暗号化

「Must Gather の暗号化 (Encrypt Must Gather)」が、「グローバル・プロファイル」画面に追加されました。「グローバル・プロファイル」画面に移動するには、「設定」 > 「グローバル・プロファイル」をクリックします。デフォルトでは、クリアされています (暗号化しない)。これがクリアされている場合、Must Gather 出力は圧縮され、暗号化されません (現行の機能)。チェック・ボックスにチェック・マークが付けられている場合、以降のすべての Must Gather 出力は暗号化されます。暗号化は、store encrypt_must_gather on CLI コマンドによって設定したり、store encrypt_must_gather off コマンドによってクリアしたりすることもできます。

GuardAPI の must gather

GuardAPI コマンドを使用して、スクリプトから GuardAPI Must Gather 情報収集を実行します。

grdapi must_gather --help=true。

以下の関数パラメーターがリストされます。

```
ID=0
function parameters :
commandsList - String -required - Constant values list
description - String
email - String
maxLogLength - Integer - Constant values list
pmrNumber - String
runDuration - Integer - Constant values list
startRun - Date
To get a Constant values list for a parameter, call the function with --get_param_values=<param-name>
```

--commandsList には文字列が必要です。--description も、必須の文字列です。--runDuration は、must_gather がどれだけの期間実行されるかを示します。must_gather レポートを送信する E メール・アドレスを入力します。--maxLogLength パラメーターは、ログ・レポートの最大長を設定する必須の整数です。--pmrNumber は、IBM サポートが顧客のレポートを追跡して解決するために使用する問題管理レポート番号です。--startRun は、必須の日付 (now など) です。grdapi must_gather --get_param_values=<param-name> 関数を呼び出すことによって、パラメーターごとの値のリストを取得することができます。

Windows S-TAP のプレカーネル・ダンプ検証ユーティリティー

カーネル・ダンプを実行する前にこのユーティリティーを実行し、問題なくダンプを作成できることを確認します。

S-TAP のインストール・ディレクトリー内の該当する場所からユーティリティー SystemVerificationTool.exe を実行します。

ウィンドウが開き、状況、構成されているダンプ・タイプ、CPU サイズ、および空きディスク・スペースが表示されます。カーネル・ダンプの構成が適切な場合、「システムはカーネル・ダンプ用に正しく構成されています (The system is correctly configured for a kernel dump)」と表示されます。カーネル・ダンプの構成が適切ではない場合は、その理由が通知されます。

メッセージ	対処方法
ダンプ・タイプがカーネル・タイプとして構成されていません (Dump type is not configured kernel)。	リンクをクリックしてダンプ・タイプを変更する手順に従い、ユーティリティーを再実行します。
CPU が 1 つしかないため、ダンプを生成すると CPU がロックされます (There is only one CPU, generating a dump may lock up the CPU)。	カーネル・ダンプは推奨されません。
ディスク・スペースが小さすぎて、ダンプの作成を処理できません。ダンプを生成すると、CPU がロックされます(Disk space is too low to handle the creation of the dump, generating a dump may lock up the CPU)。	ディスク・スペースを解放して、ユーティリティーを再実行します。ユーティリティーの実行には、物理メモリーの 3 分の 1 または 10 GB のいずれか小さい方の容量が必要になります。

親トピック: 問題のトラブルシューティング手法
関連情報:
Guardium のトラブルシューティングおよびサポート (ビデオ)

IBM との情報の交換

問題を診断または特定するために、システムのデータおよび情報を IBM サポートに提供する必要がある場合があります。問題判別に使用するツールまたはユーティリティーを IBM サポートから提供される場合もあります。
親トピック: 問題のトラブルシューティング手法

IBM サポートへの情報の送信

問題解決に必要な時間を短縮するために、トレースおよび診断情報を IBM サポートに送信することができます。

手順

診断情報を IBM サポートに送信するには、次のようにします。

1. 問題管理レコード (PMR) を開きます。
2. 必要な診断データを収集します。診断データは、PMR を解決するまでの時間を短縮するのに役立ちます。診断データは、次のように手動でも自動でも収集できます。
   - データを手動で収集する。
   - データを自動的に収集する。
3. ファイルを .zip または .tar ファイル形式を使用して圧縮します。
4. ファイルを IBM に転送します。 以下のいずれかの方法を使用して、ファイルを IBM に転送できます。
   - サービス・リクエスト・ツール
   - 標準的なデータのアップロード方法: FTP、HTTP
   - セキュアなデータ・アップロード方法: FTPS、SFTP、HTTPS
   - E メール

   これらすべてのデータ交換方法については、IBM サポートの Web サイトで説明されています。

IBM サポートからの情報の受信

IBM 技術サポート担当者から、診断ツールやその他のファイルのダウンロードをお願いする場合があります。これらのファイルは FTP を使用してダウンロードできます。

始める前に

IBM 技術サポート担当者から、ファイルのダウンロードに使用する推奨サーバー、およびアクセスするディレクトリーとファイルの正確な名前について、必ず指定を受けてください。

手順

IBM サポートからファイルをダウンロードするには、次のようにします。

1. FTP を使用して、IBM 技術サポート担当者が指定したサイトに接続し、`anonymous` としてログインします。電子メール・アドレスをパスワードとして使用します。
2. 次のようにして、適切なディレクトリーに移動します。
   a. /fromibm ディレクトリーに移動します。

      ```
      cd fromibm
      ```

   b. IBM 技術サポート担当者が指定したディレクトリーに移動します。

      ```
      cd nameofdirectory
      ```

3. セッションでバイナリー・モードを有効にします。

   ```
   binary
   ```

4. get コマンドを使用して、IBM 技術サポート担当者が指定したファイルをダウンロードします。

   ```
   get filename.extension
   ```

5. FTP セッションを終了します。

   ```
   quit
   ```

サポート更新のサブスクライブ

使用する IBM 製品に関する重要な情報を常に入手するために、更新にサブスクライブできます。

このタスクについて

Guardium に関する更新を受け取るようにサブスクライブすることで、特定の IBM サポート・ツールおよびリソースに関する重要な技術情報と更新を受け取ることができます。次の 2 つの方法のうちいずれかを使用して、更新にサブスクライブできます。

RSS フィードとソーシャル・メディアのサブスクリプション
Guardium については、次の RSS フィードとソーシャル・メディアのサブスクリプションを利用できます。

- *RSS feed 1*
- *RSS feed 2*
- *RSS feed 3*

RSS に関する一般情報 (RSS を使用するための設定の手順や、RSS に対応した IBM Web ページのリストなど) については、IBM Software Support RSS feeds の Web サイトを参照してください。

My Notifications
「My Notifications」を使用すると、任意の IBM 製品のサポート更新にサブスクライブすることができます。(「My Notifications」は、これまでにご利用いただいた可能性のある類似のツール「My Support」に代わるものです。) 「My Notifications」を利用すると、電子メールによる告知を毎日または毎週受け取るように指定できます。 また、受信する情報のタイプ (資料、ヒント、製品フラッシュ (アラートとも呼ばれる)、ダウンロード、およびドライバーなど) を指定できます。「My Notifications」を利用して、情報を受け取りたい製品やニーズに最適な配信方法を、カスタマイズしたりカテゴリー化したりすることができます。

手順

サポート更新にサブスクライブするには、以下の手順に従ってください。

1. Guardium RSS フィードをサブスクライブします。
2. IBM® サポート・ポータルにアクセスし、「通知」ポートレットの「My Notifications」をクリックすることによって、「My Notifications」にサブスクライブします。
3. IBM ID およびパスワードを使用してサインインし、「送信」をクリックします。
4. 更新を受け取る対象と方法を指定します。
    a. 「サブスクライブ」タブをクリックします。
    b. 該当するソフトウェア・ブランドまたはハードウェアのタイプを選択します。
    c. 1 つ以上の製品名を選択して、「続行」をクリックします。
    d. 更新を受け取る方法 (電子メールで受信、指定したフォルダーにオンライン受信、RSS または Atom フィードとして受信) の設定を選択します。
    e. 例えば、製品ダウンロードについての新しい情報とディスカッション・グループのコメントなど、受け取る資料の更新のタイプを選択します。
    f. 「送信」をクリックします。

タスクの結果

RSS フィードと My Notifications の設定を変更するまで、要求した更新情報に関する通知を受け取ることになります。設定は必要に応じて (ある製品の使用を中止して、別の製品の使用を開始する場合など) 変更できます。
親トピック: 問題のトラブルシューティング手法
関連情報
IBM Software Support RSS feeds
Subscribe to My Notifications support content updates
My Notifications for IBM technical support overview
My Notifications Questions and Answers

問題および解決策

このトピックで、発生した問題の解決策を検索してください。

- ユーザー・インターフェース
- ポリシー
- レポート
- 評価および強化
- Guardium システムの構成
- アクセス管理
- 統合
- 一元管理
- S-TAP およびその他のエージェント
- GIM
- ファイル・アクティビティーのトラブルシューティング
- Guardium システムのインストール

親トピック: 問題のトラブルシューティング

ユーザー・インターフェース

- 検査エンジンの追加時に変更内容が保存されない
  検査エンジンの追加時に変更内容が保存されない場合は、パラメーターが有効であることを確認します。
- HTTP エラー 403
  HTTP エラー 403 を受け取った場合は、Cross-Site Request Forgery (CSRF) 保護機構を無効にすると、このエラーを回避できます。
- Java.lang.IllegalStateException
  java.lang.IllegalStateException エラーを受け取った場合は、Java サーブレットをクリーンアップします。
- ページが正しくロードされない
  ページが正しくロードされない場合は、GUI を再始動するか、別のブラウザーを使用します。

親トピック: 問題および解決策

検査エンジンの追加時に変更内容が保存されない

検査エンジンの追加時に変更内容が保存されない場合は、パラメーターが有効であることを確認します。

症状

検査エンジンを追加したときに、新規の設定が数分間だけ残り、その後消失します。

原因

S-TAP 構成ファイル guard_tap.ini 内で、新規検査エンジンまたは別の検査エンジンの 1 つ以上のパラメーター値にエラーがあります。

環境

Guardium コレクター・ユーザー・インターフェースが影響を受けます。

問題の解決

検査エンジンに設定する必要のあるすべてのパラメーターに、有効な値が設定されていることを確認してください。例えば、一部のデータベース・タイプでは、db_install_dir を、サーバー上のインストール・ディレクトリーのパスに設定する必要があります。ただし、その他のデータベース・タイプでは、このパラメーターを設定してはならないか、または NULL に設定する必要があります。ご使用のデータベース・タイプに固有の要件を S-TAP ヘルプ・ブックでチェックし、すべてのパラメーターが正しく設定されていることを確認してください。

親トピック: ユーザー・インターフェース

HTTP エラー 403

HTTP エラー 403 を受け取った場合は、Cross-Site Request Forgery (CSRF) 保護機構を無効にすると、このエラーを回避できます。

症状

システムのメインページから IBM Security Guardium GUI をリフレッシュすると、以下のエラーを受け取ります。

```
HTTP Status 403-
type Status report
message
description Access to the specified resource () has been forbidden
```

原因

これは、Cross-Site Request Forgery (CSRF) を回避するように設計されている Guardium の機能が原因です。CSRF 保護は、デフォルトで有効になっています。

環境

すべての Guardium 構成 (コレクター、アグリゲーター、中央マネージャー) が影響を受けます。

問題の解決

この機能を無効にするには、CLI コマンド store gui csrf_status off を使用します。

注: CSRF 保護をオフにすると、Guardium システムのセキュリティー・レベルは低下します。

以下のコマンドによって、Cross-Site Request Forgery に対する保護が有効になります。デフォルトでは有効になっています。store gui csrf_status on

状況を確認するには、CLI コマンド show gui csrf_status を実行します。

親トピック: ユーザー・インターフェース

Java.lang.IllegalStateException

java.lang.IllegalStateException エラーを受け取った場合は、Java サーブレットをクリーンアップします。

症状

以下のエラー・メッセージを受け取ります。

エラーが発生しました。 システム管理者に連絡してください
(java.lang.IllegalStateException)

原因

このエラーが発生するのは、メソッドが呼び出され、Java VM がそのメソッドと不整合な状態である場合です。また、デッドロックが原因で Java サーブレットが破損している場合もあります。

環境

Guardium システムが影響を受けます。

問題の解決

数分待ってからやり直してください。エラーが続く場合は、ユーザー cli としてログインした後、コマンド restart GUI を実行して GUI を再始動します。

Java サーブレットをクリーンアップするには、コマンド support clean sevlets を実行します。

問題が解決しない場合は、以下の tomcat ログを収集し、IBM Security Guardium 技術サポートにお問い合わせください。

```
tomcat_log/localhost.<date_stamp>.log
tomcat_log/catalina.<date_stamp>.log
```

親トピック: ユーザー・インターフェース

ページが正しくロードされない

ページが正しくロードされない場合は、GUI を再始動するか、別のブラウザーを使用します。

症状

空白画面または別のエラーが表示されることがあります。この問題は特定のシステム上の特定のブラウザーで発生しますが、他では発生しません。

原因

この原因は、ローカライズされたブラウザーに限定されるか、Java 仮想マシンに問題がある可能性があります。

環境

コレクター、アグリゲーター、および中央マネージャーが影響を受けます。

問題の解決

問題を解決するには、Guardium システムで CLI プロンプトから restart GUI を実行します。それでも解決しない場合は、以下のアクションを試してください。

- システムを再始動します。
- Java 仮想マシンをアンインストールし、再インストールします。
- ブラウザーをアンインストールしてから、再インストールします。
- 異なるブラウザーを使用します。

親トピック: ユーザー・インターフェース

ポリシー

- 相関アラート定義内に照会が表示されない
  相関アラート定義内に照会が表示されない場合は、カウント・フィールドにチェック・マークを付けて、タイム・スタンプでソートします。
- ルールがトリガーされない
  ポリシー・コマンド・フィールドに値を持つルールが予期されるようにトリガーされない場合は、ルールを再構成します。
- 編集機能によって結果が過度にマスクされる
  編集機能によって結果が過度にマスクされる場合は、正規表現 [¥x0c]{1}[0-9]{8}([0-9]{4}) を使用します。
- Oracle データベースにログインする SSH セッションおよび自動化 CRON ジョブが失敗したログインとして表示される
  Oracle データベースにログインする SSH セッションおよび自動化 CRON ジョブが失敗したログインとして表示される場合は、ポリシーを修正します。
- Guardium 内部データベースがいっぱいになる
  Guardium 内部データベースがいっぱいになった場合は、手動で、または通常のパージ戦略の一環としてパージすることができます。

親トピック: 問題および解決策

相関アラート定義内に照会が表示されない

相関アラート定義内に照会が表示されない場合は、カウント・フィールドにチェック・マークを付けて、タイム・スタンプでソートします。

症状

相関アラートを作成するために、アクセス照会を作成しました。しかし、相関アラート定義内で、この照会はドロップダウン・リストに表示されません。

原因

レポートでの相関アラート検索は、タイム・スタンプに基づいています。

環境

コレクターおよびアグリゲーターが影響を受けます。

問題の解決

「カウントの追加」チェック・ボックスにマークを付け、タイム・スタンプでソートします。

親トピック: ポリシー

ルールがトリガーされない

ポリシー・コマンド・フィールドに値を持つルールが予期されるようにトリガーされない場合は、ルールを再構成します。

症状

ポリシーの「コマンド」フィールドに値を持つルールが予期されるようにトリガーされません。

原因

この原因は、コマンド・フィールドの構成の誤りです。 Guardium パーサーは、コマンド修飾子をコマンドの一部と見なしません。

環境

Guardium コレクター - ワイルドカード (%) を使用する場合、ポリシー・ルール内のコマンド・フィールドも影響を受けます。

問題の解決

ルールの「コマンド」フィールド内の値は、SQL 動詞に表示される値と正確に一致する必要があり、必要に応じてワイルドカード (%) が追加されます。正しい例は次のとおりです。

```
GRANT
GRANT%
```

この例は正しくありません。

```
GRANT% TO PUBLIC
%GRANT% ADMIN OPTION%
```

ADMIN OPTION と TO PUBLIC は一致せず、ルールをトリガーできません。これは、Guardium パーサーがこれらをコマンドの一部と認識しないためです。一般に、パーサーはコマンド修飾子をコマンドの一部と見なしません。 代わりに、ポリシーがモニターするトラフィックを調べるためのレポートを作成し、そのレポートにコマンド・エンティティーからの「SQL 動詞」フィールドを組み込みます。 「SQL 動詞」フィールドにリストされたものはすべてパーサーに認識され、ポリシー・ルールの「コマンド」フィールドに使用できます。複数のコマンドをグループに追加して、そのグループをルール内で単一コマンドの代わりに使用することができます。この場合、各グループ・メンバーは SQL 動詞内のエントリーに一致する必要があります。Guardium には、ユーザーが使用したりコピーを作成したりできるコマンド・グループがいくつかあります。
親トピック: ポリシー
親トピック: レポート

編集機能によって結果が過度にマスクされる

編集機能によって結果が過度にマスクされる場合は、正規表現 [¥x0c]{1}[0-9]{8}([0-9]{4}) を使用します。

症状

編集機能によって結果が過度にマスクされるか、Oracle トラフィックに ORA-03106 エラーが発生します。

原因

Guardium ポリシー・ルールの編集機能は、結果セットとのパターン・マッチングを行います。これには、一致した文字列をユーザーが指定した文字に置き換える機能があります。

環境

Guardium コレクターが影響を受けます。

問題の解決

正規表現 `[¥x0c]{1}[0-9]{8}([0-9]{4})` を使用します。 この正規表現では、結果が列の長さで始まり、その後に 12 桁が続くようになり、最後の 4 桁が置き換えられます。

親トピック: ポリシー

Oracle データベースにログインする SSH セッションおよび自動化 CRON ジョブが失敗したログインとして表示される

Oracle データベースにログインする SSH セッションおよび自動化 CRON ジョブが失敗したログインとして表示される場合は、ポリシーを修正します。

症状

`/as sysdba` を指定して SQLPLUS および RMAN から Oracle データベースにログインする SSH セッションおよび自動化 CRON ジョブが、失敗したログインとして表示されます。

原因

画面に表示されない場合でも、Oracle はこうしたログインの試みに対して以下のエラーで応答します。

```
ORA-01-17: invalid username/password; logon denied.
```

このエラーによって、失敗したログインのアラートがトリガーされます。例えば、データベース・ユーザー WRONGLOGIN が DBA グループのメンバーであり、`sqlplus WRONGLOGIN as sysdba` としてログインした場合、WRONGLOGIN のデータベース認証が失敗します。この失敗によって ORA-01-17 エラー・アラートがトリガーされ、Guardium ログに反映されます。ただし、sysdba 特権を持つユーザーはデータベース認証なしでもデータベースに接続できるため、セッションの続行が許可されます。どちらのイベントもキャプチャーされ、記録されます。

環境

Guardium コレクターが影響を受けます。

問題の解決

失敗したログインについてアラートするルールの前に許可アクションを組み込むように、ポリシーを修正することができます。以下の条件を使用して、ポリシー内に例外ルールを作成します。

```
Client IP=<Server IP>
Source program = SQLPLUS
DB user in trusted group
OS user in group of Oracle DBAs
Net protocol = BEQUEATH (if local BEQUEATH, not TCP)
```

このルールにより、ORA-01-17 エラーが原因のログイン失敗アラートはスキップされます (ただしログには記録されます)。ログイン失敗アラートをレポートからフィルターに掛けて除外するには、条件リストの最後に以下の条件を追加します。

```
AND
        (
    client IP<>server IP OR
    src prg <> SQLPLUS OR
    db user NOT IN group of trusted OR
    os user NOT IN group of oracle DBAs OR
    net protocol <>BEQUEATH (if this is local BEQUEATH, not TCP )
   )
```

親トピック: ポリシー

Guardium 内部データベースがいっぱいになる

Guardium 内部データベースがいっぱいになった場合は、手動で、または通常のパージ戦略の一環としてパージすることができます。

症状

Guardium 内部データベースがいっぱいになり、データのほとんどが GDM_POLICY_VIOLATIONS_LOG 表内にある。

原因

ポリシーへの変更によって、ポリシー違反ルールが頻繁にトリガーされる場合があります。データのほとんどが GDM_POLICY_VIOLATIONS_LOG 表内に見つかります。

環境

Guardium コレクターが影響を受けます。

問題の診断

CLI コマンド support show db-top-tables all を実行します。

問題の解決

「ポリシー違反 / インシデント管理」レポートにチェック・マークを付けて、常にトリガーされるポリシー・ルールを識別します。次に、ポリシー・ルールがそれほど頻繁にトリガーされないように調整します。

GDM_POLICY_VIOLATIONS_LOG 表内の余分なデータは、通常のパージ戦略の一環としてパージされます。ただし、GDM_POLICY_VIOLATIONS_LOG 表からデータを手動で消去する場合は、コマンド support clean DAM_data policy_violations<start_date><end_date> を使用できます。

親トピック: ポリシー

レポート

- 少なくとも 1 回監査プロセスが実行された後に監査プロセスの受信者の表を変更できない
  監査プロセスの受信者の表を変更できない場合は、監査プロセスをコピーし、元の監査プロセスを置き換えます。
- マルチバイト文字が表示されない
  PDF にエクスポートした Guardium レポートの文字が正しくない場合は、PDF フォント構成を切り替えます。
- ファイル・システムがほとんどいっぱいである
  Guardium ファイル・システムがほとんどいっぱいである場合は、ログ・ローテーション戦略を変更します。
- Guardium 監査レポートを Microsoft Excel で表示すると予期しない文字を含む行が表示される
  監査レポートを .csv 形式で表示したときに予期しない文字を含む行が表示される場合は、別の .csv ビューアーを使用するか、.pdf ファイルとして表示します。
- レポートに IP アドレスが 0.0.0.0 と表示される

- 「要求が中断されたか、割り当て量を超えました」エラー・メッセージ
  レポートの実行時に、要求が中断されたか、割り当て量を超えたことを示すエラー・メッセージを受け取る場合は、より短いレポート間隔にレポートを分割します。
- ルールがトリガーされない
  ポリシー・コマンド・フィールドに値を持つルールが予期されるようにトリガーされない場合は、ルールを再構成します。
- 5 分おきのスケジュールされたジョブの例外
  スケジュールされたジョブの例外を 5 分おきに受け取る場合は、「異常検出」ページからのアラートを非アクティブ化します。
- スケジュール済みジョブ例外: マージが必要です。プロセスの実行を延期してください (merge required, delay executing process)
  マージが必要であるというエラー・メッセージを受け取った場合は、プロセスの実行を延期し、監査プロセスのスケジュールを変更します。
- Teradata のモニター時に Guardium レポートにデータベース・ユーザーが正しく表示されない
  Teradata のモニター時に、Guardium レポートにデータベース・ユーザーが正しく表示されない場合は、Teradata Database を構成します。
- 埋め込みコマンドによる Guardium レポートが予期しない結果になる
  予期しない結果の Guardium レポートを受け取った場合は、タプルを使用して深さを処理するようにポリシー・ルールを構成します。

親トピック: 問題および解決策

少なくとも 1 回監査プロセスが実行された後に監査プロセスの受信者の表を変更できない

監査プロセスの受信者の表を変更できない場合は、監査プロセスをコピーし、元の監査プロセスを置き換えます。

症状

監査プロセスを少なくとも 1 回実行した後には、受信者を削除することも追加することもできなくなります。また、受信者の以下のプロパティーを変更することもできません。

- 必要なアクション
- 続行
- 空の場合は承認

原因

監査プロセスが少なくとも 1 回実行されると、受信者の表はロックされ、ほとんどのプロパティーを変更できなくなります。

環境

すべての Guardium 構成 (コレクター、アグリゲーター、中央マネージャー) が影響を受けます。

問題の解決

以下の手順によって、受信者の表を変更できます。

1. 監査プロセスをコピーします。
2. コピーされた監査プロセスに変更を加えます。
3. 元の監査プロセスを削除します。ただし、監査プロセスの履歴を残したい場合は、その監査プロセスの名前を変更することができます。
4. コピーされた監査プロセスの名前を、元の監査プロセスの名前に変更します。

親トピック: レポート

マルチバイト文字が表示されない

PDF にエクスポートした Guardium レポートの文字が正しくない場合は、PDF フォント構成を切り替えます。

症状

GUI ではレポートを表示できます。しかし、レポートを PDF にエクスポートすると、文字が正しく表示されないか、欠落します。 PDF レポートで、文字が疑問符 (?) またはその他の記号で表示されます。

原因

Guardium PDF エクスポートのデフォルトのフォントでは、マルチバイト文字が正しく表示されません。例えば、ギリシャ語、キリル文字、中国語の文字は正しく表示されません。

環境

コレクター、アグリゲーター、および中央マネージャーが影響を受けます。

問題の解決

バージョン 9 以降では、PDF フォント構成を切り替えて問題を解決します。

1. CLI のユーザーとしてログインします。
2. コマンド store pdf-config multilanguage_support を実行します。
3. 2 Multi-language. を選択します。

親トピック: レポート

ファイル・システムがほとんどいっぱいである

Guardium ファイル・システムがほとんどいっぱいである場合は、ログ・ローテーション戦略を変更します。

症状

ファイル・システムがいっぱいになりつつあり、100% に達しようとしています。

原因

syslog に送信されるアラートとレポートによって、ファイル・システムがいっぱいになる場合があります。

環境

コレクターまたはアグリゲーターが影響を受ける場合があります。

問題の解決

デフォルトでは、ログ・ファイルは毎週循環され、5 つのファイルが保持されます。ただし、ログ・ファイルのログ・ローテーション戦略を変更することができます。以下のコマンドを使用して、システム内のメッセージをより少なく保持するようにします。

```
store logrotate [agg|message] [daily|weekly|monthly] [# of rotations]
```

親トピック: レポート

Guardium 監査レポートを Microsoft Excel で表示すると予期しない文字を含む行が表示される

監査レポートを .csv 形式で表示したときに予期しない文字を含む行が表示される場合は、別の .csv ビューアーを使用するか、.pdf ファイルとして表示します。

症状

Microsoft Excel で監査レポート (.csv 形式) を表示すると、特定の行に予期しない文字が含まれていることに気付きます。 これらの文字は、完全な SQL 列内で見られる文字に似ています。この問題は、.pdf レポートまたは GUI レポートでは発生しません。

原因

Microsoft Excel では、セルに含めることができる文字数の制限は 32,767 文字です。キャプチャーした SQL がこの制限を超える場合は、次の行にまたがります。

環境

コレクター、アグリゲーター、および中央マネージャーが影響を受けます。

問題の解決

セルあたりの文字数の制限がより大きい、別の .csv ビューアーを使用するか、.pdf ファイルとして監査レポートを表示します。

親トピック: レポート

レポートに IP アドレスが 0.0.0.0 と表示される

症状

Guardium で IP アドレスが 0.0.0.0 と表示されます。

原因

Guardium がトラフィックを暗号化解除するとき、IP アドレスは最初は 0.0.0.0 として記録されます。これは、スニファーが実際の IP アドレスを認識していないことが原因です。暗号化解除が完了すると、別のスレッドによって正しい IP アドレスがセッション・テーブルに再設定されます。

環境

データベース・トラフィックを暗号化するすべてのデータベースが影響を受けます。

問題の解決

数分後に同じレポートを実行します。より新しいトラフィックの正しいクライアント IP を表示するには、クライアント/サーバー・ドメインのフィールド「分析済みのクライアント IP」をレポートに追加します。 一部の行では、「分析済みのクライアント IP」がブランクになる可能性があります。 ブランクの場合、その部分のトラフィックの暗号化解除は完了していません。

親トピック: レポート

「要求が中断されたか、割り当て量を超えました」エラー・メッセージ

レポートの実行時に、要求が中断されたか、割り当て量を超えたことを示すエラー・メッセージを受け取る場合は、より短いレポート間隔にレポートを分割します。

症状

Guardium でレポートを実行すると、「要求が中断されたか、割り当て量を超えました」というエラー・メッセージを受け取ります。

原因

エラー・メッセージ「要求が中断されたか、割り当て量を超えました」が表示されるのは、対話式レポートが 3 分の時間制限内に完了しない場合です。基になる原因は、一般的にレポートのサイズです。

環境

コレクターおよびアグリゲーターが影響を受けます。

問題の解決

この問題を解決するには、以下のオプションのいずれかを実行します。

- レポートを、より短いレポート間隔に分割します。 このアクションは、最も推奨される方式です。レポートが 4 GB を超える場合、MYSQL 表データのポインター・サイズがなくなる原因になります。
- 照会タイムアウト値を大きい値にします。「管理」 > 「アクティビティー・モニター」 > 「実行照会モニター」をクリックして、「実行照会モニター」を開きます。
- ブラウザーをアンインストールしてから、再インストールします。「レポート/モニター照会タイムアウト」ボックスに秒数を入力し、「更新」をクリックします。
- バックグラウンドでレポートを実行します。バックグラウンドで実行されるレポートは、照会タイムアウトの影響を受けません。
- 監査プロセスとしてレポートを実行します。

親トピック: レポート

ルールがトリガーされない

ポリシー・コマンド・フィールドに値を持つルールが予期されるようにトリガーされない場合は、ルールを再構成します。

症状

ポリシーの「コマンド」フィールドに値を持つルールが予期されるようにトリガーされません。

原因

この原因は、コマンド・フィールドの構成の誤りです。 Guardium パーサーは、コマンド修飾子をコマンドの一部と見なしません。

環境

Guardium コレクター - ワイルドカード (%) を使用する場合、ポリシー・ルール内のコマンド・フィールドも影響を受けます。

問題の解決

ルールの「コマンド」フィールド内の値は、SQL 動詞に表示される値と正確に一致する必要があり、必要に応じてワイルドカード (%) が追加されます。正しい例は次のとおりです。

```
GRANT
GRANT%
```

この例は正しくありません。

```
GRANT% TO PUBLIC
%GRANT% ADMIN OPTION%
```

ADMIN OPTION と TO PUBLIC は一致せず、ルールをトリガーできません。これは、Guardium パーサーがこれらをコマンドの一部と認識しないためです。一般に、パーサーはコマンド修飾子をコマンドの一部と見なしません。 代わりに、ポリシーがモニターするトラフィックを調べるためのレポートを作成し、そのレポートにコマンド・エンティティーからの「SQL 動詞」フィールドを組み込みます。 「SQL 動詞」フィールドにリストされたものはすべてパーサーに認識され、ポリシー・ルールの「コマンド」フィールドに使用できます。複数のコマンドをグループに追加して、そのグループをルール内で単一コマンドの代わりに使用することができます。この場合、各グループ・メンバーは SQL 動詞内のエントリーに一致する必要があります。Guardium には、ユーザーが使用したりコピーを作成したりできるコマンド・グループがいくつかあります。

親トピック: ポリシー
親トピック: レポート

5 分おきのスケジュールされたジョブの例外

スケジュールされたジョブの例外を 5 分おきに受け取る場合は、「異常検出」ページからのアラートを非アクティブ化します。

症状

定期的な短い間隔 (通常 5 分おき) で、スケジュール済みジョブ例外レポート内に同じメッセージを受け取ります。 この間隔は、異常検出が実行されるポーリング間隔と同じです。

スケジュール済みジョブ例外レポートの例は以下のとおりです。

```
Timestamp            Exception Description                                Count of Exceptions
2013-12-05 15:51:22.0 java.lang.NumberFormatException: empty String       1
```

同じ例外が 5 分おきに発生します。

原因

アクティブ・アラートのいずれかがエラーの原因です。

環境

Guardium のコレクターおよびアグリゲーターが影響を受けます。

問題の診断

ポーリング間隔とアクティブ・アラートは、「異常検出」ページで確認できます。「保護」 > 「データベースの侵入検出」 > 「異常検出」をクリックして、「異常検出」ページを開きます。

問題の解決

問題の原因になっているアラートを正確に特定し、非アクティブ化します。

1. 「異常検出」ページで 1 つのアラートを非アクティブ化します。
2. ポーリング間隔が経過するまで待ちます。
3. そのアラートを非アクティブ化したことによってエラーがなくなったかどうかを確認します。
4. エラーが続く場合は、そのアラートを再アクティブ化して、別のアラートを非アクティブ化します。
5. ステップ 2 から 5 を繰り返して、すべてのアラートを試します。

問題の原因になっているアラートが見つかり、そのエラーを把握して停止するために支援が必要な場合は、IBM Guardium 技術サポートに問い合わせて、以下のアイテムを提供してください。

1. 正確なエラー・テキストおよび画面キャプチャー。
2. 以下の CLI コマンドの出力。要求された場合、1 ポーリング間隔の長さを指定します。

   support must_gather app_issues

   support must_gather alert_issues

親トピック: レポート

スケジュール済みジョブ例外: マージが必要です。プロセスの実行を延期してください (merge required, delay executing process)

マージが必要であるというエラー・メッセージを受け取った場合は、プロセスの実行を延期し、監査プロセスのスケジュールを変更します。

症状

以下のメッセージを受け取ります。「マージが必要です。プロセスの実行を延期してください (merge required, delay executing process)」。短期間でこのようなメッセージをいくつか受信する可能性があります。

原因

監査プロセスが実行されるためには、その前にマージ・プロセスが終了している必要があります。

環境

アグリゲーターが影響を受けます。

問題の診断

「レポート」 > 「Guardium 運用レポート」 > 「統合/アーカイブ・ログ」をクリックして、「統合/アーカイブ・ログ」を開きます。agg_progress.log で問題を診断することもできます。

問題の解決

監査プロセスが、マージ・プロセス後に少なくとも 10 分経過してから実行されるようにスケジュールを変更します。

親トピック: レポート

Teradata のモニター時に Guardium レポートにデータベース・ユーザーが正しく表示されない

Teradata のモニター時に、Guardium レポートにデータベース・ユーザーが正しく表示されない場合は、Teradata Database を構成します。

症状

Guardium レポートで、モニター対象の Teradata Database からのレコードを表示すると、データベース・ユーザー名のフィールドが予期されるとおりに表示されません。ユーザー名が切り捨てられるか、欠落します。

原因

Teradata Database では、完全なユーザー名を返すことができません。

環境

Teradata Database からデータをキャプチャーするすべての Guardium コレクターが影響を受けます。

問題の解決

以下のコマンドを使用して、Teradata Database が完全なユーザー名を正しい文字セットでモニター・アプリケーションに返すことができるようにします。他のアプリケーションは影響を受けません。

gtwcontrol -u yes -d

-d コマンドによって、更新された GDO 設定が表示されます。

注: この設定では、ユーザー名が暗号化されない形式で返されます。暗号化が有効になっている場合は、システムからエラー・メッセージが返されます。
親トピック: レポート

埋め込みコマンドによる Guardium レポートが予期しない結果になる

予期しない結果の Guardium レポートを受け取った場合は、タプルを使用して深さを処理するようにポリシー・ルールを構成します。

症状

レポートの結果が、予期しないものであるか、ポリシーによってフィルターに掛ける必要があると思われます。逆に、キャプチャーしようとしていたステートメントがキャプチャーされません。

原因

通常、SQL には複数のオブジェクトおよびコマンドがステートメント内に埋め込まれています。ポリシー定義またはレポート定義が、異なる深さのオブジェクトまたはコマンドを処理するように構成されていません。

環境

Guardium コレクターが影響を受けます。

問題の解決

条件が正しいオブジェクト名と一致していることを確認します。正しいメイン・エンティティーを使用して、異なる深さのオブジェクトまたは SQL 動詞を表示します。それでも予期しない動作が見られる場合は、グループ・ビルダーを使用して、ポリシー内で使用するタプルのグループを定義します。タプルでは、複数の属性を組み合わせて 1 つのグループ・メンバーを形成することができます。

注: タプルには、1 つのスラッシュおよび 1 つのワイルドカード文字 (%) を使用できます。 ダブルスラッシュは使用できません。
親トピック: レポート

評価および強化

- Windows で CAS が Java 1.7 と連携しない
  Windows 上で Guardium 変更監査システムが Java バージョン 1.7 と連携しない場合、msvcr100.dll を CAS の bin フォルダーにコピーします。
- 失敗したテストに脆弱性評価の例外グループ・メンバーが表示される
  失敗した脆弱性評価テストにテスト例外グループのメンバーが表示される場合は、円記号 (¥) にエスケープ・シーケンスを使用します。

親トピック: 問題および解決策

Windows で CAS が Java 1.7 と連携しない

Windows 上で Guardium 変更監査システムが Java バージョン 1.7 と連携しない場合、msvcr100.dll を CAS の bin フォルダーにコピーします。

症状

Guardium CAS は、古いバージョンの Java とは連携しますが、Java 1.7 とは連携しません。

原因

<GUARDIUM STAP directory>¥cas¥bin¥ に、msvcr100.dll がありません。

環境

Windows 上の Guardium CAS が影響を受けます。

問題の解決

この問題を解決するには、以下の手順を実行します。

1. ご使用のシステムで Java 1.7 がインストールされたパス (C:¥Program Files (x86)¥Java¥jre7¥bin など) を見つけます。
2. 前のステップで見つけた Java パス内で、ライブラリー jvm.dll のロケーションを見つけます。
3. <CAS directory>¥conf ディレクトリー内の cas.cfg ファイルを編集します。例えば、C:¥Program Files (x86)¥GUARDIUM_STAP¥cas¥conf¥cas.cfg が標準的なファイル・パスです。
4. JVM に対応する行 (;JVM=c:¥program files¥java¥jre1_2_3¥bin¥client¥jvm.dll など) を見つけます。
5. 行の先頭からセミコロンを削除します。次に、JVM を、ステップ 2 のライブラリー jvm.dll のパスに設定します。`JVM=C:¥Program Files (x86)¥Java¥jre7¥bin¥server¥jvm.dll`
6. msvcr100.dll を、Java 7 インストール・ディレクトリー内の bin フォルダーから、<CAS directory>¥bin フォルダーにコピーします。例えば、C:¥Program Files (x86)¥Java¥jre7¥bin¥msvcr100.dll を C:¥Program Files (x86)¥Guardium¥GUARDIUM_STAP¥cas¥bin¥msvcr100.dll にコピーします。
7. 変更監査システムを再始動します。

注: これは、Java バージョン 1.7 の場合のみ必要です。Java のそれより古いバージョンでは、このステップは必要ありません。
親トピック: 評価および強化

失敗したテストに脆弱性評価の例外グループ・メンバーが表示される

失敗した脆弱性評価テストにテスト例外グループのメンバーが表示される場合は、円記号 (¥) にエスケープ・シーケンスを使用します。

症状

脆弱性評価を実行すると、詳細フィールドにテスト例外グループの一部のメンバーが表示されます。このグループには、円記号 (¥) と REGEX タグを持つメンバーが含まれています (例: `(R)US¥John Doe`)。

原因

Guardium による例外グループの解析時に、特殊文字によってエラーがトリガーされる場合があります。

環境

Guardium コレクターが影響を受けます。

問題の解決

円記号 (¥) にエスケープ・シーケンスを使用すること、および REGEX タグを使用しないようにします (完全一致を使用します)。以下の例はどちらも機能します。

```
US¥John Doe
```

```
(R)US¥¥John Doe
```

REGEX タグ (R) は、詳細フィールドの正規表現検索をトリガーするために使用され、正規表現に一致するすべての文字列が削除されます。 正規表現において意味を持つ円記号 (¥) やその他の文字は、構文解析エラーを回避するために円記号 (¥)のエスケープ・シーケンスが必要です。(R) タグを使用しない場合、Guardium がマッチングを行う際には、グループ・メンバーは詳細フィールド内の行全体と完全に一致する必要があります。脆弱性テストを通過するには、テストの詳細フィールドを空にする必要があります。

親トピック: 評価および強化

Guardium システムの構成

- アップグレード後に STAP を構成できない
  S-TAP をアップグレードした後に Guardium 内で S-TAP を構成します。
- Guardium がネットワーク・デバイス VMXNET x を認識できない
  Guardium がネットワーク・デバイス VMXNET x を認識できない場合は、Guardium を仮想マシンにインストールし、ネットワーク・アダプターを追加します。
- システム・ボードの交換後に Guardium ネットワーク・インターフェース・エラーが発生する
  ハードウェアの修理後にエラー・メッセージを受け取った場合は、ネットワーク・パラメーターをリセットします。
- ネットワークから Guardium 仮想マシンにアクセスできない
  ネットワークから Guardium 仮想マシンにアクセスできない場合は、store network interface inventory コマンドを実行し、システムを再始動します。
- SSLv3 が有効になっている
  `SSLv3 is enabled` という警告を受け取った場合は、SSLv3 を無効にして POODLE 攻撃を防止します。

親トピック: 問題および解決策

アップグレード後に STAP を構成できない

S-TAP をアップグレードした後に Guardium 内で S-TAP を構成します。

症状

Guardium Installation Manager (GIM) を使用して S-TAP をアップグレードした後に、モジュールのインストール結果で正常に完了したと表示されたにも関わらず、Guardium 内の検査エンジンでデータベース・パス・パラメーターを構成できません。

原因

新規 S-TAP がまったく新しいモジュールとしてインストールされた場合、K-TAP は正しくアップグレードされません。古い K-TAP モジュールが削除されないため、古い K-TAP モジュールと新しい S-TAP の間にプロトコルの不一致が生じます。

環境

AIX、HP-UX、Linux、および Solaris などの UNIX および Linux にインストールされた S-TAP。

問題の診断

問題を診断するには、guard_diag ユーティリティーを実行して、Guardium S-TAP の Must Gather データを収集します。

以下の行が syslog ファイル内にあります。

```
STAP and KTAP Protocol Version Mismatch,
Exit!!!!!: No such file or directory
Tap_controller::init failed
GUARD-01: Error Initializing STap
```

モジュールのログ・ファイルには、古い K-TAP がリストされます。例: `ktap_24276 338760 0`

問題の解決

この問題を解決するには、GIM モジュール・インストール・ペインで以下の手順を実行します。

1. K-TAP Live Update を Y に設定します。
2. K-TAP_ENABLED を Y に設定し、新規 S-TAP を再インストールします。

親トピック: Guardium システムの構成

Guardium がネットワーク・デバイス VMXNET x を認識できない

Guardium がネットワーク・デバイス VMXNET x を認識できない場合は、Guardium を仮想マシンにインストールし、ネットワーク・アダプターを追加します。

症状

VMware でのインストール中に、Guardium がネットワーク・デバイス VMXNET x を認識できません。ゲストとして VMware で Guardium をインストールすると、「`eth0: unknown interface: No such device`」というエラーを受けとります。システムの再始動後にこのエラー・メッセージが表示されます。

原因

VMXNET x 仮想ネットワーク・アダプターは、VMware ツールのみに含まれる特定のドライバーを必要とします。どのオペレーティング・システムにもそのドライバーはありません。Guardium が Linux で実行されており、インストーラーには VMXNET x 用のドライバーは含まれていません。

環境

Guardium システムが影響を受けます。

問題の解決

以下の手順を実行して、この問題を解決します。

1. E1000 または Flexible などのデフォルトのネットワーク・アダプターを使用して、VMware 上に仮想マシンを作成します。
2. 仮想マシンに Guardium をインストールします。
3. Guardium 用の現行の GPU 累積パッチをインストールします。
4. インストール後に、CLI コンソールにログオンし、setup vmware_tools install コマンドを実行して、VMware ツールをインストールします。
5. CLI コンソールから stop system コマンドを使用して Guardium システムをシャットダウンします。
6. VMware Infrastructure Client などの VMware クライアント・ツールを使用して仮想マシン設定を編集します。現行のネットワーク・アダプターを選択し、それを削除します。
7. VMXNET というネットワーク・アダプターを追加します。
8. Guardium システムを再始動します。

親トピック: Guardium システムの構成

システム・ボードの交換後に Guardium ネットワーク・インターフェース・エラーが発生する

ハードウェアの修理後にエラー・メッセージを受け取った場合は、ネットワーク・パラメーターをリセットします。

症状

Guardium アプライアンスでシステム・ボードの交換などのハードウェアの修理を行った後に、ネットワーク接続が失われます。アプライアンスのリブート時に、ネットワーク・インターフェースごとに以下のエラー・メッセージが出されます。

```
rtnetlink answers: no such device
```

原因

システム・ボードを交換すると、MAC アドレスが変わります。アドレスが変わることで、実際の MAC アドレスと、インターフェース構成ファイルに保管されているアドレスが一致しなくなります。

環境

システム・ボードが交換された Guardium アプライアンス (コレクター、アグリゲーター、または中央マネージャー) と、すべての Guardium バージョンが影響を受けます。

問題の解決

ユーザー CLI としてコンソールからアプライアンスにログインし、以下のコマンドを実行してネットワーク・パラメーターをリセットしてください。

```
store network interface inventory
restart network
store network interface ip<IP_address>
store network interface mask<netmask>
store network routes defaultroute<gateway_address>
restart network
```

それでも問題が解決されない場合は、Guardium サポートに連絡して、手動操作を依頼してください。

親トピック: Guardium システムの構成

ネットワークから Guardium 仮想マシンにアクセスできない

ネットワークから Guardium 仮想マシンにアクセスできない場合は、store network interface inventory コマンドを実行し、システムを再始動します。

症状

新規 Guardium システムを仮想マシンとして実装し、すべての必要な初期ネットワーク構成を実行しました。しかし、IP アドレスを使用してシステムを ping することができず、ネットワーク内でそのシステムにアクセスできません。

原因

仮想環境によって仮想マシンに割り当てられた MAC アドレスが、Guardium での MAC アドレスと一致していません。

環境

コレクター、アグリゲーター、および中央マネージャーが影響を受けます。

問題の診断

この問題を診断するには、ネットワークで IP アドレスを ping します。 ping<appliance's ip address> コマンドを使用します。失敗した場合は、システムの MAC アドレスを表示します。

1. ユーザー "cli" でログインします。
2. show network macs コマンドを実行して、Guardium 構成に保管されている MAC アドレスを表示します。
3. ご使用の仮想環境の管理ユーティリティーから、仮想マシンの MAC アドレスを確認します。
    a. VMware Workstation を開きます。
    b. 仮想マシンを右クリックし、「設定 (Settings)」または「プロパティー (Properties)」を選択して「仮想マシンの設定 (Virtual Machine Settings)」を開きます。
    c. 「ハードウェア (Hardware)」の下で「ネットワーク・アダプター (Network Adapter)」を選択します。
    d. 「拡張 (Advanced)」をクリックして、「ネットワーク・アダプター拡張設定 (Network Adapter Advanced Settings)」を開きます。
    e. ステップ 2 と 3 の MAC アドレスを比較します。

問題の解決

この問題を解決するには、以下の手順を実行します。

1. ユーザー "cli" として Guardium システムにログインします。
2. store network interface inventory コマンドを実行します。
3. y と入力して、NIC をリセットします。
4. restart system コマンドを使用して、システムを再始動します。

親トピック: Guardium システムの構成

SSLv3 が有効になっている

`SSLv3 is enabled` という警告を受け取った場合は、SSLv3 を無効にして POODLE 攻撃を防止します。

症状

以下の警告を受け取ります: `SSLv3 is enabled`。

原因

SSLv3 には、Padding Oracle On Downgraded Legacy Encryption (POODLE) と呼ばれるプロトコル脆弱性が存在します。システムで SSLv3 が有効になっている場合、この脆弱性によりアタッカーは SSL/TLS を強制的に SSLv3 にフォールバックさせ、暗号を解除して、ネットワーク・トラフィックをプレーン・テキストで傍受することが可能となります。この脆弱性の詳細は、National Vulnerability Database の CVE-2014-3566 で説明されています。

Guardium® では、POODLE 攻撃を防止するためにすべてのシステムで SSLv3 を無効にすることを推奨しており、新しい Guardium システムではデフォルトで SSLv3 が無効になっています。ただし、古いシステムや一部のアップグレード・シナリオでは SSLv3 が有効なままになっている場合があります。

このトピックでは、SSLv3 の状況を確認し、必要な場合に無効にする方法について説明します。

重要: SSLv3 を無効にすると、Guardium v10 中央マネージャーと GPU 500 より前の Guardium v9 を実行する管理対象ユニットの一部との間の接続が切れる可能性があります。GPU 500 より前の Guardium v9 を実行する管理対象ユニットが存在する混合環境を使用している場合、SSLv3 を無効にする前に、管理対象ユニットを GPU 500 にアップグレードするか、またはパッチ 9501 を適用してください。

問題の解決

1. CLI コマンド `show sslv3` を使用して SSLv3 の状況を確認します。
   - 出力が `SSL setting is disabled` の場合、SSLv3 は無効です。SSLv3 を無効にするための追加手順は不要です。
   - 出力が `SSL setting is enabled` の場合、SSLv3 は有効です。SSLv3 を無効にするための手順を続けてください。
2. CLI コマンド `store sslv3 off` を使用して SSLv3 を無効にします。コマンド出力は、以下のようになります。

```
Current SSL setting is enabled.  Will change to disabled.
Restarting
gui (GUI を再始動しています)
Changing to port 8443 (ポート 8443 に変更しています)
From port 8443
Stopping..... (停止しています.....)
ok
```

3. `show sslv3` と入力して SSLv3 が無効になったことを確認します。出力は `SSL setting is disabled` となるはずです。

親トピック: Guardium システムの構成

アクセス管理

- admin または accessmgr 以外で Guardium にログインできない
  admin または accessmgr としてログインする場合を除いて Guardium GUI にログインできない場合は、認証構成設定を確認します。
- Guardium accessmgr のパスワードのリセット
  accessmgr パスワードが分からなくなり、ログインできない場合は、Guardium サポートに連絡してください。

親トピック: 問題および解決策

admin または accessmgr 以外で Guardium にログインできない

admin または accessmgr としてログインする場合を除いて Guardium GUI にログインできない場合は、認証構成設定を確認します。

症状

admin または accessmgr 以外のユーザーで Guardium にログインすることができません。accessmgr が定義した正しいユーザーおよびパスワードを使用しているにも関わらず、ユーザー名またはパスワードが無効であるというエラーが表示されます。 以下のエラー・メッセージを受け取ります。ユーザー名/パスワードのいずれか （または両方） が無効です。資格情報を再入力してください。

原因

認証設定がローカルとして構成されていません。

環境

コレクター、アグリゲーター、および中央マネージャーが影響を受けます。

問題の解決

問題を解決するには、認証設定をローカルに変更します。このアクションにより、accessmgr によって定義されたどのユーザーとしてもログインできます。

親トピック: アクセス管理

Guardium accessmgr のパスワードのリセット

accessmgr パスワードが分からなくなり、ログインできない場合は、Guardium サポートに連絡してください。

症状

Guardium accessmgr のパスワードが分からなくなり、GUI にログインできません。連続してログイン試行に失敗すると、アカウントのロックも行われます。

原因

Guardium では、複数回のログイン試行の失敗を許容しません。

環境

コレクター、アグリゲーター、および中央マネージャーが影響を受けます。

問題の解決

CLI にログインし、support reset-password accessmgr<N>|random というコマンドを実行します。

<N> または random を使用できます。<N> は、10000000 から 99999999 までの範囲内の数値です。random と指定すると10000000 から 99999999 までの範囲内の数値が自動的に生成されます。IBM Guardium サポートを利用し、PMR を開いて、以下の出力を送信します。

```
G10.ibm.com> support reset-password accessmgr random
Password for accessmgr account have been successfully reset using keyword:<passkey>
Please provide these number to Guardium Customer Service to receive actual account password.
ok
```

新しいパスワードを受け取ったら、アカウントをアンロックします。

1. アカウントをアンロックするには、以下のコマンドを使用します。unlock accessmgr。
2. accessmgr としてログインし、accessmgr の詳細を編集して、一時的なパスワードを入力します。
3. 一時的なパスワードを使用して再度ログインします。
4. プロンプトが出されたら、新規パスワードを入力します。

親トピック: アクセス管理

統合

- Guardium コレクターをアグリゲーターに変換できない
  Guardium コレクターを中央マネージャー・アグリゲーターに変換できない場合は、Guardium を再インストールし、インストール時にアグリゲーターを選択してください。
- Guardium 管理対象システムの GUI からのデータ・エクスポートの構成変更がエラーのため失敗する
  データ・エクスポートの構成変更が失敗した場合は、コレクターとアグリゲーターで共有パスワードが同一になるようにしてください。
- 監査プロセスの結果とレポートの違い
  監査プロセスの結果とレポートの間に違いがある場合は、すべてのアプライアンスが同じタイム・ゾーンに設定されていることを確認してください。
- アグリゲーターで構成を復元した後に HY000 エラーが発生する
  アグリゲーターで構成をリストアした後に HY000 エラーを受け取った場合は、ダミー・インポートを実行します。

親トピック: 問題および解決策

Guardium コレクターをアグリゲーターに変換できない

Guardium コレクターを中央マネージャー・アグリゲーターに変換できない場合は、Guardium を再インストールし、インストール時にアグリゲーターを選択してください。

症状

store unit type manager aggregator というコマンドを使用して、Guardium コレクターをアグリゲーターに変換しようとします。

しかし、以下のコマンドで、ユニット・タイプがまだマネージャーとしてリストされます。

```
> show unit type
Manager
```

原因

CLI コマンドを使用してコレクターをアグリゲーターに変換することはできません。

環境

Guardium コレクターが影響を受けます。

問題の解決

コレクターをアグリゲーターに変換するには、Guardium 製品を再インストールし、インストール時にユニット・タイプとしてアグリゲーターを選択します。アグリゲーターのインストール後に、コマンド store unit type manager を使用して、アグリゲーターを中央マネージャー・アグリゲーターに変換できます。

中央マネージャー/アグリゲーターの制約
v9.5 (v9.0 パッチ 500) 以降、アプリケーションには、中央マネージャーがアグリゲーター・タイプのアプライアンスでなければならないという制約があります。つまり、v9.5 以降では、アグリゲーター・タイプのアプライアンスのみを中央マネージャー・アプライアンスにプロモートできます。v9.5 より前の既存の CM アプライアンスは、この変更の対象ではありません。

親トピック: 統合

Guardium 管理対象システムの GUI からのデータ・エクスポートの構成変更がエラーのため失敗する

データ・エクスポートの構成変更が失敗した場合は、コレクターとアグリゲーターで共有パスワードが同一になるようにしてください。

症状

データ・エクスポートの新規の設定を保存しようとして、構成を保存するために「適用」をクリックしたときに、エラーが発生します。

以下のエラーを修正したうえで再試行してください:
指定されたパラメーターを使用してテスト・データ・ファイルをこのホストに送信することができませんでした。 ホスト名または IP アドレスが正しく入力されていること、ホストがオンラインであること、ターゲット・ディレクトリーが存在し指定したユーザーが書き込めること、そのユーザーのパスワードが正しく指定されていることを確認してください。

原因

Guardium は、データ・エクスポート構成で指定されたユーザーおよびパスワードを使用して、ターゲット・ホストに scp によってログインしようとします。次に、Guardium はテスト・ファイルをターゲット・ディレクトリーにコピーしようとします。このシステム上の共有パスワードは、このシステムからのエクスポート先として設定しようとしているアグリゲーター上の共有パスワードと一致しません。

環境

Guardium 構成: コレクターとアグリゲーターが影響を受けます。

問題の解決

必ず、コレクターとアグリゲーターで共有パスワードが同一になるようにしてください。以下のいずれかの方式を使用できます。

1. アグリゲーター上の共有パスワードを知っている場合は、コレクター上の共有パスワードを同じ値に設定します。以下のいずれかの方式を使用できます。
   - CLI から store system shared secret コマンドを使用して、共有パスワードを設定します。
   - GUI から、「設定」 > 「システム」 > 「システム構成」で共有パスワードを設定します。
2. アグリゲーター上の現在の共有パスワードをバックアップし、コレクターにリストアします。
   - アグリゲーターで、CLI コマンドを実行します。

     ```
     aggregator backup keys file <user@host:/path/filename>
     パラメーター
     user@host:/path/filename
     ```

     ファイル転送操作において、バックアップ鍵ファイルのユーザー、ホスト、および絶対パス名を指定します。 指定するユーザーには、指定したディレクトリーに対する書き込み権限が必要です。
   - コレクターで、次のコマンドを使用して、共用パスワードをリストアします。

     ```
     aggregator restore keys file<user@host:/path/filename>
     ```

3. 両方のアプライアンスの共有パスワードを、同一になるようにリセットします。
   注: アグリゲーターの共有パスワードを変更した場合、アグリゲーターをエクスポート先とする他のすべての Guardium システムの共有パスワードをリセットする必要があります。

親トピック: 統合

監査プロセスの結果とレポートの違い

監査プロセスの結果とレポートの間に違いがある場合は、すべてのアプライアンスが同じタイム・ゾーンに設定されていることを確認してください。

症状

時間パラメーター (例えば「最終日の始め (Start of Last Day)」および「最終日の終わり (End of Last Day)」など) を使用して監査プロセスの一部としてアグリゲーターで実行されるように、レポートを設定します。 そのレポートの結果を調べると、最初のタイム・スタンプは常に、00.00 よりも後の規定の時刻 (例えば 02.00) になります。さらに、最後のタイム・スタンプは、常に 23.59 よりも前の規定の時刻 (例えば 21.59) になります。ただし、レポートを対話式に実行すると、タイム・スタンプは予期したとおりに表示されます。

原因

コレクターとアグリゲーターのタイム・ゾーンが同一の設定になっていない可能性があります。

環境

アグリゲーターが影響を受けます。

問題の診断

すべてのアプライアンスが同じタイム・ゾーンに設定されていることを確認してください。次のコマンドを使用します。`show system clock timezone.`

問題の解決

コレクターとアグリゲーターが同じタイム・ゾーンで設定されていない場合は、CLI を使用してアプライアンスのタイム・ゾーンを構成してください。

```
store system clock timezone list
store system clock timezone <timezone>
```

以下のコマンドを使用して、アプライアンスでの時刻が正しいことを確認します。

```
show system clock datetime
store system clock datetime
```

日時は、以下のコマンドによって NTP サーバーを使用して同期化することもできます。

```
show system ntp all
store system ntp state
store system ntp server
```

親トピック: 統合

アグリゲーターで構成を復元した後に HY000 エラーが発生する

アグリゲーターで構成をリストアした後に HY000 エラーを受け取った場合は、ダミー・インポートを実行します。

症状

アグリゲーターまたは中央マネージャーの構成をリストアしたときに、以下のメッセージのいずれかまたは両方を受け取ります。

```
ERROR 1031 (HY000) at line 1: Table storage engine for 'GUARD_USER_ACTIVITY_AUDIT' doesn't have this option
ERROR 1031 (HY000) at line 1: Table storage engine for 'AGGREGATOR_ACTIVITY_LOG' doesn't have this option
```

原因

このエラー条件は、内部データベースで一時的な不一致がある場合に発生することがあります。

環境

コレクターおよびアグリゲーターが影響を受けます。

問題の解決

この問題を解決するには、ダミー・インポートを実行します。

親トピック: 統合

一元管理

親トピック: 問題および解決策

ユーザーが Guardium 管理対象ユニットで無効になっているが中央マネージャーで有効として表示される

あるユーザーが、Guardium 管理対象ユニットで無効になっているが、中央マネージャーで有効として表示される場合は、ポータル・ユーザー同期を実行します。

症状

あるユーザーが管理対象ユニットで無効になっています。そのユーザーのアカウントが中央マネージャーで再有効化されましたが、管理対象ユニットでは引き続き無効として表示されています。そのユーザーのアカウントは、中央マネージャーでは有効であるものとして表示されます。

原因

中央マネージャー内のユーザーのアカウントが、管理対象ユニットと同期化されていません。

環境

中央マネージャー、コレクター、またはアグリゲーターの組み合わせが影響を受ける可能性があります。

問題の解決

中央マネージャーと管理対象ユニットの間で現在のユーザー状況を同期化するには、ポータル・ユーザー同期を実行します。

1. admin ユーザーとして中央マネージャーにログインします。
2. 「管理」 > 「一元管理」 > 「ポータル・ユーザー同期」をクリックして、「ポータル・ユーザー同期 (Portal User Synchronization)」を開きます。
3. 「今すぐ 1 回実行」をクリックします。

このようにしても、管理対象ユニットと中央マネージャーの間でユーザーのアカウントが同期化されない場合は、IBM Guardium 技術サポートにお問い合わせください。

親トピック: 一元管理

アップグレードされたユニットの新規バージョンを中央マネージャーが認識しない

中央マネージャーが、アップグレードされたユニットの新規バージョンを認識しない場合は、アップグレードされたユニットを選択し、ページをリフレッシュします。

症状

中央マネージャーは、管理対象の、アップグレードされたアグリゲーターまたはコレクターの新規バージョンを、直ちに認識しない場合があります。中央マネージャーから、新規バージョンを必要とするパッチをプッシュすると、ユニットがまだ以前のバージョンであることを示すエラーが出される可能性があります。

管理対象ユニットの古いバージョンが、引き続き GUI の「一元管理」ビューに表示されます。そのビュー内のユニットの ping 時間は、中央マネージャーと管理対象ユニットの間の通信が良好であることを示しています。

原因

新規バージョンの情報をプルするには、GUI をリフレッシュする必要があります。

環境

Guardium 中央マネージャーが影響を受けます。

問題の解決

GUI の「一元管理」ビューで、アップグレードされたユニットを選択し、「リフレッシュ」を押します。このアクションによって、ユニットから新規バージョンの情報がプルされます。

親トピック: 一元管理

スケジュールされたタスクが予定の時刻に起動しない

スケジュールされたタスクが予定の時刻に起動しない場合は、ポータル・ユーザー同期の後に実行されるように、インポートの時刻をスケジュールします。

症状

インポートが失敗し、agg_progress.log で以下のメッセージを受け取ります。

```
* 05/20 04:00:01 --- Import cannot start
(guard_agg|turbine_backup.sh|restore_from_file.pl already running)
* 05/20 20:00:46 --- Merge cannot start - aggregation still active
```

原因

中央マネージャーのポータル・ユーザー同期との競合があります。

環境

アグリゲーターが影響を受けます。

問題の診断

バックグラウンドで実行されているタスクを見つけます。「レポート」 > 「Guardium 運用レポート」 > 「統合/アーカイブ・ログ」をクリックして、「統合/アーカイブ・ログ」を開きます。

問題の解決

問題を解決するには、ポータル・ユーザー同期の後に実行されるように、インポートの時刻をスケジュールします。ポータル・ユーザー同期を 1 時間ごとに実行し、インポートの時刻を、その時刻の 30 分後にします。

親トピック: 一元管理

GUI の「一元管理」ビューでのトルク例外

「一元管理」でトルク例外が発生した場合は、カスタム・グループを削除して新規グループを作成します。

症状

Guardium GUI の「一元管理」ビューで特定のカスタム・グループを選択すると、グループ内の管理対象ユニットではなく、エラーが表示されます。

```
org.apache.torque.TorqueException: Failed to select one and only one row.
```

例外の発生後に、「一元管理」タブの下のどのグループまたはビューにもその例外が表示されます。GUI からログアウトして再度ログインするまで、その例外は以前に作業していたグループに対しても表示されます。

原因

このトルク例外は、グループ内の管理対象ユニットのうちの 1 つが、中央マネージャーではなく管理対象ユニットから登録抹消された場合に、発生する可能性があります。

環境

Guardium 中央マネージャーが影響を受けます。

問題の解決

カスタム・グループを削除して、同じメンバーを含む新規グループを作成します。

親トピック: 一元管理

S-TAP およびその他のエージェント

- IBM Security Guardium S-TAP のインストール時またはアップグレード時に AIX 6.1 で障害が発生する
  AIX 6.1 上で Guardium S-TAP をインストールまたはアップグレードするときにオペレーティング・システムで障害が発生する場合は、フィックスパック AIX 6.1 を適用します。
- Guardium COMM_EXIT_LIST for Db2 の構成時に共有メモリー領域を開くとエラーが発生する
  Guardium COMM_EXIT_LIST の構成時にエラー・メッセージを受け取った場合は、guardctl コマンドを使用して Db2 インスタンス所有者を許可します。
- Guardium が Informix から共有メモリー・トラフィックを収集できない
  Guardium が Informix から共有メモリー・トラフィックを収集できない場合は、検査エンジン構成を確認します。
- Guardium STAP ホスト内で CPU および I/O 使用量が高い
  CPU または I/O 使用量が高い場合は、すべての検査エンジンの構成を確認します。
- ログイン・パケットからの情報の欠落
  ログイン・パケットからの情報が欠落している場合は、S-TAP デバッグ・トレースおよび slon トレースを収集します。
- Nanny プロセスによってスニファーが強制終了される
  Nanny プロセスによってスニファーが強制終了される場合、着信するトラフィックが多すぎる可能性があります。
- スニファーが UNIX S-TAP に接続できない
- UNIX S-TAP を開始できない
  UNIX S-TAP を開始できない場合は、バッファー・サイズが大きすぎる可能性があります。
- Linux 上で S-TAP が自動的に開始されない
  Linux 上で Db2 または Oracle 向けの S-TAP エージェントが自動的に開始されない場合は、/etc/event.d/ ディレクトリーを確認します。
- S-TAP からの戻りが FIPS 140-2 準拠ではない
  FIPS 140-2 に関するエラーを受け取った場合は、「S-TAP 制御」ページで構成を変更します。
- S-TAP のアンインストール後も K-TAP カーネル・モジュールが存在している
  S-TAP のアンインストール後も K-TAP カーネル・モジュールが存在する場合は、手動で削除します。
- UNIX S-TAP が 16 を超える検査エンジンを読み取れない
  UNIX S-TAP が 16 を超える検査エンジンを読み取れない場合は、listen ポートのパラメーターを変更するか、PCAP を使用します。
- Windows S-TAP サービスが始動時にクラッシュする (エラー ID 1000)
  エラー ID 1000 により S-TAP がクラッシュする場合は、guard_tap_ini 構成ファイル内の SOFTWARE_TAP_IP パラメーターを確認します。
- Guardium システム上で z/OS S-TAP がアクティブと表示されない
  Guardium システム上で z/OS S-TAP がアクティブと表示されない場合は inspection-core を再始動します。
- S-TAP が A-TAP トラフィックをキャプチャーしていない
  A-TAP 構成に関する詳細情報を出力するには、A-TAP スクリプトを使用します。
- S-TAP が Db2 出口トラフィックをキャプチャーしていない
  Db2 IE パラメーターを確認し、オプションで修正するには、Db2 出口ヘルス・チェック・スクリプトを使用します。

親トピック: 問題および解決策

IBM Security Guardium S-TAP のインストール時またはアップグレード時に AIX 6.1 で障害が発生する

AIX 6.1 上で Guardium S-TAP をインストールまたはアップグレードするときにオペレーティング・システムで障害が発生する場合は、フィックスパック AIX 6.1 を適用します。

症状

AIX 6.1 上で Guardium S-TAP をインストールまたはアップグレードするときに、オペレーティング・システムで障害が発生します。AIX クラッシュ・メモリー・ダンプに、以下のスタック・トレースが表示されます。

```
Error ID: DD11B4AF Resource Name: SYSPROC
Detail Data: 00007FFFFFFFD080 0000000000473260
0000000000020000 8000000000029032

Symptom Information:
Crash Location: [0000000000473260] execvex_common+1880
Component: COMP Exception Type: 131

Stack Trace:
[0000000000473260] execvex_common+1880
[000000000047744C] execve+A8
[F1000000C083E84C] my_execve+424
```

原因

このクラッシュは、AIX バージョン 6.1 の既知の問題であり、execvex_common コード・パスでのシステム・クラッシュが原因で発生します。

環境

AIX 6.1 オペレーティング・システムにインストールされるすべての S-TAP が影響を受けます。

問題の解決

フィックスパック AIX 6.1 6100-08-04 を適用して問題を解決するには、http://www-01.ibm.com/support/docview.wss?uid=isg1IV50179 を参照してください。

親トピック: S-TAP およびその他のエージェント

Guardium COMM_EXIT_LIST for Db2 の構成時に共有メモリー領域を開くとエラーが発生する

Guardium COMM_EXIT_LIST の構成時にエラー・メッセージを受け取った場合は、guardctl コマンドを使用して Db2 インスタンス所有者を許可します。

症状

Guardium libguard を使用するように DB2 COMM_EXIT_LIST を構成して Db2 サーバーを再始動した後、Db2 diag ログで以下のエラーを受け取ります。

```
2013-06-28-11.41.12.306169-300 E870950E486 LEVEL: Severe
PID : 15764 TID : 139905833363200 PROC : db2sysc 0
INSTANCE: db2001 NODE : 000
APPHDL : 0-16
HOSTNAME: dbhost1
EDUID : 54 EDUNAME: db2agent () 0
FUNCTION: DB2 UDB, DRDA Communication Manager, sqljcCommexitLogMessage,
probe:234
DATA #1 : String with size, 91 bytes
WARNING: Shmem_access /.guard_writer0 failed Error opening shared memory area errno=2 err=8
```

原因

以下のメッセージは、Guardium ライブラリーが必要な共有メモリー・デバイスを作成できなかったことを示しています。

```
Shmem_access /.guard_writer0 failed
Error opening shared memory area
errno=2
err=8
```

guardctl コマンドを使用して、Db2 インスタンス所有者を許可されたユーザーとして追加する必要があります。

環境

Db2 Exit (バージョン 10) と S-TAP の統合を使用する Guardium コレクターが影響を受けます。

問題の解決

guardctl コマンドを使用して、Db2 インスタンス所有者を許可されたユーザーとして追加する必要があります。

1. Db2 インスタンスを停止します。
2. Db2 インスタンス所有者を許可します。

3. Db2 インスタンスを開始します。

Guardium Installation Manager (GIM) がインストールされていない場合は、以下のコマンドを使用して Db2 インスタンス所有者を許可します。

```
<guardium_installdir>/bin/guardctl authorize-user<db2 instance owner>
```

Guardium Installation Manager (GIM) がインストールされている場合は、以下のコマンドを使用して Db2 インスタンス所有者を許可します。

```
<guardium_installdir>/modules/ATAP/current/files/bin/guardctl authorize-user<db2 instance owner>
```

例えば、Db2 インスタンス所有者が db2001 であり、GIM が /usr/local/guardium にインストールされている場合、コマンドは `/usr/local/gim/modules/ATAP/current/files/bin/guardctl authorize-user db2001` です。

親トピック: S-TAP およびその他のエージェント

Guardium が Informix から共有メモリー・トラフィックを収集できない

Guardium が Informix から共有メモリー・トラフィックを収集できない場合は、検査エンジン構成を確認します。

症状

Guardium S-TAP が Informix から共有メモリー・トラフィックを収集できません。

原因

検査エンジンが正しく構成されていません。

環境

Informix システムからの S-TAP 収集がすべて影響を受ける可能性があります。

問題の解決

「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」で、検査エンジン構成を確認します。「プロセス名」フィールドの値が、データベース・サーバーでの以下のコマンドの結果と一致することを確認します。

```
ls -lrt /INFORMIXTMP/.inf.*
```

Informix: /INFORMIXTMP/.inf.sqlexec は、すべての Informix プラットフォームに適用します (Linux を除く)。Linux での Informix の例: /home/informix11/bin/oninit

このコマンドが値を返すためには、Informix が実行されていることが必要です。

A-TAP を使用する Linux サーバーの場合は、共有メモリー・トラフィックを収集するように A-TAP が構成されている必要があります。A-TAP 構成内の --db-info パラメーターと同じ値に設定してから、A-TAP をアクティブ化します。

親トピック: S-TAP およびその他のエージェント

Guardium STAP ホスト内で CPU および I/O 使用量が高い

CPU または I/O 使用量が高い場合は、すべての検査エンジンの構成を確認します。

症状

Guardium S-TAP プロセスによる CPU または I/O 使用量が高くなっています。

原因

以下の項目が一般的な原因です。

1. いずれかの検査エンジンの構成エラー。 検査エンジンにエラーがある場合は、S-TAP プロセスが頻繁に再始動するか、検査エンジンに繰り返し再接続しようとします。
2. S-TAP の K-TAP 部分が、S-TAP への確認要求とともに接続情報を送信している。このステップが遅延の原因となっています。
3. ORACLE RAC が使用されているが、量が多い可能性がある Oracle RAC トラフィックをモニターしないようにするための unix_domain_socket_marker パラメーターが、S-TAP 構成ファイル内に設定されていない。
4. ユーザー ID チェーン (UID チェーン) 機能が有効になっている (例えば、S-TAP 構成ファイル内のパラメーター hunter_trace=1)。ハンター・トレースは UID チェーンで使用され、S-TAP に対してかなり CPU を使用する場合があります。
5. ファイアウォールが有効になっている (firewall_installed=1)。 このファイアウォールによって、監視対象の新規セッションごとに判定が要求されるため、S-TAP のパフォーマンスが低下する可能性があります。

環境

AIX にインストールされている S-TAP

問題の解決

原因に応じて、対応するアクションを実行します。

1. すべての検査エンジンの構成を確認し、どのパラメーターにもエラーがないようにします。例えば、データベース・インストール・ディレクトリー、実行可能ファイル、ポート、および検査エンジンに使用可能なその他のパラメーターが、つづりや値の誤りがなく正しく設定されていることを確認します。
2. Set S-TAP 構成パラメーター ktap_fast_tcp_verdict を 1 に設定し (guard_tap.ini 構成ファイルで ktap_fast_tcp_verdict = 1 を設定)、S-TAP を再始動します。以下に使用可能な設定を示します。

   ktap_fast_tcp_verdict=0: KTAP は、ポートと IP をチェックして、セッションが検査エンジンによって構成されたデータベース接続であることを確認します。

   ktap_fast_tcp_verdict=1: KTAP は、セッションのポートが範囲内にある間は S-TAP に要求を送信しません。

3. UID チェーン機能が不要な場合は、hunter_trace=0 を設定し、S-TAP を再始動することによって無効にします。
4. SGATE が不要な場合は、firewall_installed=0 を設定し、S-TAP を再始動します。

親トピック: S-TAP およびその他のエージェント

ログイン・パケットからの情報の欠落

ログイン・パケットからの情報が欠落している場合は、S-TAP デバッグ・トレースおよび slon トレースを収集します。

症状

Guardium で、ログイン・パケットからの情報 (データベース・ユーザー名、ソース・プログラム、データベース名など) の欠落に関する問題が発生します。

原因

セッションが短すぎる場合、ログイン・パケットの情報が欠落することがあります。

環境

Guardium コレクターが影響を受けます。

問題の解決

Guardium S-TAP がインストールされているデータベース・サーバーから S-TAP デバッグ・トレースを収集し、コレクターから slon トレースを収集します。

これらのトレースの収集について詳しくは、『関連 URL』セクションの『技術情報』を参照してください。

1. 両方のトレースを同時に実行します。
2. 両方のトレースの実行中に、問題を再現する新規データベース・セッションを生成します。ログイン・パケットが送信されるのは、データベース接続が開いているときのみです。
3. 既存のレポートに、セッション開始、クライアント・ポート、およびサーバー・ポートを追加します。新規接続を使用して問題を再現したら、レポートをリフレッシュします。
4. セッション開始をチェックして、セッション中にトレースが実行されていることを確認します。
5. セッションを 5 分以上開いたままにして、スニファーがログイン・パケットを分析できるようにします。
6. フィールドが欠落しているセッションを送信します。セッションの生成に使用したアプリケーション名、データベース名、接続 DB ユーザー、接続タイプ、SQL ステートメント、およびその他の関連する詳細を明らかにします。
7. データベース・サーバーから S-TAP デバッグ・トレース・ファイルを収集し、Guardium コレクターから slon トレースを収集します。また、現在のスニファー関連の「Must Gather」サポート情報を収集します。

親トピック: S-TAP およびその他のエージェント

Nanny プロセスによってスニファーが強制終了される

Nanny プロセスによってスニファーが強制終了される場合、着信するトラフィックが多すぎる可能性があります。

症状

Guardium システム・ログ (メッセージ) またはアラートに、以下のようなメッセージが 1 回以上報告されます。

`Nanny プロセスのエラー状態。(Nanny process error condition.) Nanny プロセスによってスニファーが強制終了されました。(The nanny process killed the sniffer.) VmData は number であり、制限を超えました。(VmData was number and was over the limit.)`

原因

スニファー・メモリー使用量が使用可能メモリーの 90% を超えたため、Nanny プロセスがスニファーを再始動しました。これは製品の予期される動作です。

環境

Guardium コレクター

問題の解決

このメッセージが頻繁に表示される場合、Guardium システムに着信するトラフィックが多すぎます。このメッセージが表示されないようにするには、この Guardium システムへのトラフィックを減らします。例えば、一部の STAP を負荷が少ないコレクターに移動したり、ポリシー内で一部のトラフィックを無視したり、ロード・バランシングを実装してトラフィックを複数のコレクターに広げたりします。

メッセージがめったに表示されない場合は、トラフィックが一時的にスパイク状態であると考えられます。 メッセージが表示されないようにするには、スパイクの原因を特定し、トリガーを回避します。例えば、その時点で実行されていたプロセスを確認し、より多くのトラフィックを生成するものを識別します。このメッセージが常に特定のプロセスの実行と同時に表示される場合は、その時点の同時トラフィックを減らします。例えば、最も負荷が大きいプロセスを移動して別の時刻に実行することも、ポリシーを通じてこのトラフィックの一部を無視することもできます。

親トピック: S-TAP およびその他のエージェント

スニファーが UNIX S-TAP に接続できない

症状

別のスレッド数を指定すると (例えば、コマンド snif -t 20 を使用して 20 を指定する)、スニファーは UNIX S-TAP に接続できなくなります。GUI コンソールで、S-TAP の状況は非アクティブになっています。

原因

デフォルトでは、スニファーは 6 つのスレッドで開始されます。スレッドの数が制限を超えると、動作が未定義であるため、スニファーは UNIX S-TAP に接続できません。

環境

UNIX S-TAP が影響を受けます。

問題の解決

スレッドの数を減らして、接続を正常に確立できるようにします。

親トピック: S-TAP およびその他のエージェント

UNIX S-TAP を開始できない

UNIX S-TAP を開始できない場合は、バッファー・サイズが大きすぎる可能性があります。

症状

S-TAP を開始できず、以下のメッセージが表示されます。

```
mmap: 十分なスペースがありません (Not enough space)
初期化できません (Can't initialize): バッファー・ファイル /tmp/stapbuf/192.168.100.107.0.buf を mmap できません (Can't mmap buffer file /tmp/stapbuf/192.168.100.107.0.buf)
初期化エラー (Error Initializing): Stap は SQLGuard キューを初期化できません (Stap cannot initialize SQLGuard queue)
```

原因

S-TAP が、バッファー・ファイルに合う十分なメモリーを割り振ることができません。

問題の解決

S-TAP のバッファー・ファイル・サイズを小さくします。 サイズは、guard_tap.ini ファイルの buffer_file_size パラメーターで指定します。

親トピック: S-TAP およびその他のエージェント

Linux 上で S-TAP が自動的に開始されない

Linux 上で Db2 または Oracle 向けの S-TAP エージェントが自動的に開始されない場合は、/etc/event.d/ ディレクトリーを確認します。

症状

/etc/inittab ファイルに正しい U-TAP エントリーが表示されているにもかかわらず、Linux 上で S-TAP プロセスが自動的に開始されません。

原因

RedHat 6 などの各種 Linux ディストリビューションでは、etc/inittab ファイルを使用する従来の init デーモンの使用は非推奨になりました。 それらのディストリビューションでは、代わりに upstart と呼ばれる init プロセスを使用するようになりました。Upstart では、U-TAP などのプロセスの自動開始、停止、および respawn に /etc/event.d ディレクトリーと /etc/init ディレクトリーを使用します。

S-TAP インストーラーは、/etc/event.d ディレクトリーが存在するかどうかを検査するようになりました。 存在する場合は、upstart で使用するために /etc/init 内にエントリーが作成されます。存在しない場合は、従来の init デーモンで使用するために /etc/inittab 内にエントリーが作成されます。

upstart を持つシステムに何らかの理由で /etc/event.d がない場合は、代わりに inittab ファイルにデータが設定されます。S-TAP プロセスは、必要な場合に始動も respawn も行いません。

環境

Linux で稼働している S-TAP が影響を受けます。

問題の解決

/etc/event.d ディレクトリーが存在するかどうかを確認します。

/etc/event.d/ ディレクトリーが存在しない場合は、以下の手順を実行して状態を解決します。

1. 既存の S-TAP インストール済み環境をアンインストールします。
2. ユーザー root として /etc/event.d ディレクトリーを作成します (mkdir /etc/event.d)。
3. S-TAP をインストールします。

親トピック: S-TAP およびその他のエージェント

S-TAP からの戻りが FIPS 140-2 準拠ではない

FIPS 140-2 に関するエラーを受け取った場合は、「S-TAP 制御」ページで構成を変更します。

症状

サポート対象: - Solaris X86 - Linux x86/64 - Linux x86/32 - Linux S390X - Linux IA64
サポート対象外: - Solaris SPARC - AIX PowerPC - HPUX RISC - HPUX IA64 - Linux PowerPC

S-TAP イベント・ログに以下のメッセージが表示されます。

```
LOG_ERR: FIPS 140-2 モードを有効にするには、use_tls=1 を設定してください (To enable FIPS 140-2 mode set use_tls=1)
```

原因

FIPS 140-2 は、暗号モジュールに関する米国政府のセキュリティー基準です。このメッセージが表示される場合は、S-TAP 構成が政府の要件を満たしていないことを示しています。

注: このメッセージは、S-TAP にエラーが発生していることを示すわけではありません。

環境

Guardium S-TAP が影響を受けます。

サポート対象: Solaris X86; Linux x86/64; Linux x86/32; Linux S390X; Linux IA64

非サポート対象: Solaris SPARC; AIX PowerPC; HPUX RISC; HPUX IA64; Linux PowerPC

問題の解決

FIPS 準拠を有効にするには、guard_tap.ini ファイルに以下を設定する必要があります。

```
use_tls=1
```

以下のいずれかの方法を使用して、構成を変更できます。

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」をクリックします。
2. 関連する S-TAP の詳細セクションを変更し、TLS チェック・ボックスを使用します。
3. S-TAP を再始動します。

DB サーバー上で guard_tap.ini ファイルを直接編集して、S-TAP を再始動することもできます。

親トピック: S-TAP およびその他のエージェント

S-TAP のアンインストール後も K-TAP カーネル・モジュールが存在している

S-TAP のアンインストール後も K-TAP カーネル・モジュールが存在する場合は、手動で削除します。

症状

Solaris サーバーから S-TAP をアンインストールした後も、K-TAP カーネル・モジュールが存在しています。

原因

Solaris サーバーから K-TAP カーネル・モジュールを削除するために、サーバーが正しく再始動されませんでした。

環境

S-TAP をアンインストールした後の Solaris サーバーが影響を受けます。

問題の診断

modinfo | grep ktap および ls -al /dev/*tap* の両方を実行して、Solaris サーバーで確認します。

問題の解決

以下の手順を実行して、K-TAP カーネルを手動で削除します。

1. /etc/init.d/upguard が削除されていることを確認します。
2. /kernel/drv/sparcv9/ktap* および /kernel/drv/ktap* を削除します。
3. modinfo | grep ktap を実行して、ロードされたドライバーの名前を取得します。
4. 次に、rem_drv<loaded driver> を実行します。例: `rem_drv ktap_36821`。
5. /dev/ktap* および `/dev/guard_ktap` を削除します。
6. サーバーを再始動します。
7. modinfo | grep ktap を実行して、以降はドライバーがロードされないようにします。
8. /etc/inittab から GIM および gsvr のエントリーを削除します (GIM のみを使用している場合)。
9. /usr/local/guardium 内に残っているファイルを手動でクリーンアップします。

親トピック: S-TAP およびその他のエージェント

UNIX S-TAP が 16 を超える検査エンジンを読み取れない

UNIX S-TAP が 16 を超える検査エンジンを読み取れない場合は、listen ポートのパラメーターを変更するか、PCAP を使用します。

症状

UNIX S-TAP が、検査エンジン設定内の最初の 16 個の `port_range` 定義しか読み取りません。

原因

設計上、K-TAP が読み取れる `port_range` 定義は 16 個のみです。

環境

K-TAP と、IPC トラフィックがあるデータベースを使用し、16 を超える検査エンジンを定義する UNIX S-TAP。

問題の解決

パラメーター `port_range_start` および `port_range_end` を使用して、必要なすべてのポートを最初の検査エンジン定義に組み込みます。このアクションによって、指定したポート範囲からのすべてのトラフィックがインターセプトされます。範囲内の一部のポートを無視する必要がある場合は、不要なサーバー・ポートを無視するようにポリシーを定義できます。

以下の例では、モニター対象のターゲット・ポートとして 50000 から 50020 の listen ポートを定義しています。

```
[DB_0]
port_range_end=50020
port_range_start=50000
```

あるいは、`ktap_local_tcp=1` および `devices=<device_name>` を設定して、TCP 接続に PCAP を使用します。

```
[TAP]
ktap_local_tcp=1
devices=<Network Device Name>
```

親トピック: S-TAP およびその他のエージェント

Windows S-TAP サービスが始動時にクラッシュする (エラー ID 1000)

エラー ID 1000 により S-TAP がクラッシュする場合は、guard_tap_ini 構成ファイル内の SOFTWARE_TAP_IP パラメーターを確認します。

症状

Windows サーバー上の S-TAP が始動しません。Windows イベント・ログに、Guardium S-TAP からのイベント ID 1000 で示されたエラーが表示されます。

```
Log Name:      Application
Source:        Application Error
Event ID:      1000
Task Category: (100)
Level:         Error
Keywords:      Classic
記述:
Faulting application name: guardium_stapr.exe, version: 9.0.0.0
Exception code: 0x40000015
```

原因

guard_tap.ini ファイルに誤った SOFTWARE_TAP_IP が指定されていることが原因で、S-TAP は Windows システムに接続できません。

環境

Windows でのすべての Guardium S-TAP が影響を受けます。

問題の解決

guard_tap.ini 構成ファイル内の SOFTWARE_TAP_IP パラメーターが、Windows サーバーの正しい IP アドレスと一致することを確認します。このパラメーターは、インストール CLI パラメーターまたは IBM Guardium Installation Manager (GIM) パラメーターで渡されます。

親トピック: S-TAP およびその他のエージェント

Guardium システム上で z/OS S-TAP がアクティブと表示されない

Guardium システム上で z/OS S-TAP がアクティブと表示されない場合は inspection-core を再始動します。

症状

Guardium システム上で z/OS S-TAP を初めて始動した後、z/OS S-TAP がアクティブと表示されません。ポリシーは、Db2 または IMS コレクション・プロファイルを使用して正しく構成され、インストールされています。z/OS S-TAP は、ポート 16022 を使用するように適切に構成されています。メインフレーム上のすべてのメッセージは接続を示しています。

原因

コレクターが、作成および構成されて以降コレクターとしてアクティブに使用されていない場合は、スニファーがポート 16022 でタイムアウトするようです。

環境

z/OS が影響を受けます。

問題の解決

CLI コマンド restart inspection-core を使用して、inspection-core を再始動します。

親トピック: S-TAP およびその他のエージェント

S-TAP が A-TAP トラフィックをキャプチャーしていない

A-TAP 構成に関する詳細情報を出力するには、A-TAP スクリプトを使用します。

症状

A-TAP トラフィックが Guardium に報告されていません。

原因

不正確に構成された IE パラメーターにより、Guardium システムへの A-TAP トラフィックの送信が妨げられる場合があります。

環境

Linux タイプのデータベース・サーバー。

問題の解決

A-TAP スクリプトを使用して、A-TAP 構成の詳細を出力します。スクリプトの名前は atap_must_gather.sh で、guard_stap bin ディレクトリーにあります。絶対パスを使用すると、どこからでも実行できます。
このスクリプトはアクティブになっている各 DB に対してヘルス・チェックを実行します。実行内容は次のとおりです。

1. Guardium グループを調べて、データベース・ユーザーが許可されているかどうかを確認します (v10.5 以上では不要)
2. システム・ライブラリー・ディレクトリーと現在インストールされているディレクトリー内の ATAP ライブラリーを調べて、ライブラリーが更新されているかどうかを確認します
3. 各 IE の実行可能ファイルを調べて、-orig ファイルと -instrument ファイルがあるかどうかを確認します
4. ATAP 構成ファイル
5. データベース・ユーザーの ID を調べて、データベースのグループを確認します (v10.5 以上では不要)
6. データベース・バージョンとその構成ファイルを収集します
7. guard_tap.ini 内の実行パスと、db_home の下の ATAP データベース実行可能ファイルを比較します
8. ps 出力 (guard_diag からの呼び出し以外)
9. guard_tap.ini ファイル (guard_diag からの呼び出し以外)
10. STAP ログ (guard_diag からの呼び出し以外)
11. システム・ログ (guard_diag からの呼び出し以外)

スクリプトは、それぞれ名前が付いたセクションに詳細を出力します。最初のセクションには、データベース・ユーザーの詳細が出力されます。Exec Files Configured in IE というタイトルの 2 つめのセクションには、guard_tap.ini 構成と DB ホーム構成の両方の詳細が出力されます。この 2 つのセクションで、一致しない詳細内容を確認してください。3 つめのセクション ATAP Configuration には、A-TAP 構成の詳細が出力されます。
親トピック: S-TAP およびその他のエージェント

S-TAP が Db2 出口トラフィックをキャプチャーしていない

Db2 IE パラメーターを確認し、オプションで修正するには、Db2 出口ヘルス・チェック・スクリプトを使用します。

症状

Db2 出口トラフィックが Guardium に報告されていません。

原因

不正確に構成された IE パラメーターにより、Guardium システムへの Db2 出口トラフィックの報告が妨げられる場合があります。

環境

DB2 出口を使用する Linux 環境。

問題の解決

Db2 出口スクリプトを使用して、Db2 IE 構成を確認し、オプションで、スクリプトによって識別された必要な修正を加えます。スクリプトは guard_stap bin ディレクトリーにあります。絶対パスを使用すると、どこからでも実行できます。このスクリプトには、次のように 2 つのオプションがあります: ./db2_exit_health_check.sh [check | fix]
デフォルトでは、ヘルス・チェックのみを実行します。fix オプションを指定すると、スクリプトによって識別された IE パラメーター・エラーが修正されます。このスクリプトは各 DB セクションに対してヘルス・チェックを実行します。実行内容は次のとおりです。

1. guard_tap.ini ファイルで出口 IE を検出します
2. db_install_dir を検証します
3. データベース・ユーザーとデータベース・グループを検出します
4. データベース・ユーザーが許可されているかどうかを確認します
5. db_install_dir が DB2_HOME または Db2 ホームと一致するかどうかを確認します
6. Db2 出口ライブラリーが正しく配置されているかどうかを確認します
7. Db2 出口ライブラリー・パスと Db2 出口 ライブラリーの許可を確認します
8. Db2 出口ライブラリーが、現在インストールされている S-TAP で更新されていることを確認します
9. Db2 出口ライブラリーがロードされているかどうかを確認します

スクリプト出力には、各 IE で見つかった問題がリストされ、fix オプションを使用した場合は、問題 (`ERROR`)、実行された修正アクション (`ACTION`)、および修正アクションの結果が提示されます。適切に構成されている Db2 IE の出力は、`DB2 Exit IE in <DB name> has a GOOD setup` になります。

ユーザーの処置: オプション。特定のユーザーが実行する特定のアクションがある場合は、1 つ以上の ts*Response エレメントを使用します。

親トピック: S-TAP およびその他のエージェント

GIM

親トピック: 問題および解決策

Guardium Installation Manager (GIM) のインストール時にエラーが発生する

GIM が正しくインストールされない場合は、ディレクトリーを手動で作成します。

症状

Guardium Installation Manager (GIM) を RHEL6 にインストールしようとしたときに、以下のエラー・メッセージが表示されます。

```
cp: cannot stat `/usr/local/GIM/modules/central_logger.log': No such file or directory Installation failed
```

原因

RedHat 6 などの各種 Linux ディストリビューションでは、etc/inittab ファイルを使用する従来の init デーモンの使用は非推奨になりました。それらのディストリビューションでは、代わりに Upstart と呼ばれる init プロセスを使用するようになりました。Upstart は、プロセスを自動的に開始、停止、および respawn するために、/etc/event.d ディレクトリーおよび /etc/init ディレクトリーを使用します。

環境

Guardium Installation Manager (GIM) が影響を受けます。

問題の解決

この問題を修正するには、以下の手順を実行します。

- 部分的な GIM インストールを削除します。
- mkdir /etc/event.d コマンドによって手動で /etc/event.d ディレクトリーを作成します。

- GIM インストーラーを実行します。

親トピック: GIM

Windows で Guardium Installation Manager (GIM) サービスが開始しない

Windows で Guardium Installation Manager (GIM) サービスが開始しない場合は、32 ビット・アプリケーション用に予約されているフォルダーに GIM を再インストールします。

症状

Guardium Installation Manager (GIM) を Windows に正常にインストールした後で、サービスが実行されていないことに気付きます。

原因

GIM は 32 ビット・アプリケーションです。64 ビットの Windows を使用している場合、Program Files(x86) ではなく Program Files に GIM がインストールされている可能性があります。

環境

GIM が影響を受けます。

問題の解決

GIM を Program Files(x86) にインストールしてください。これが 32 ビット・アプリケーション用に予約された Windows フォルダーです。

親トピック: GIM

ファイル・アクティビティー

- ファイル・アクティビティーが調査ダッシュボードにもレポートにも記録されない
- 取り外し可能ディスクのファイル・アクティビティーが調査ダッシュボードのログに記録されない
- ファイル・アクティビティーがレポートで表示されるが、調査ダッシュボードで表示されない
- 分類結果で一部のファイルが欠落する
- レポートおよび調査ダッシュボードにファイル・ディスカバリー (資格) 結果が一部しか表示されない
  レポートおよび調査ダッシュボードにディスカバリー (資格) 結果が一部しか表示されない。
- レポートおよび調査ダッシュボードでファイル分類結果が欠落する
- ファイル・アクティビティー・ログ
  ファイル・アクティビティー・モニター (FAM) ログ・ファイルの配置について説明します。
- FAM バンドルをインストールできない
  GIM クライアントをインストールした後に FAM バンドルのインストールが失敗する。

親トピック: 問題および解決策

ファイル・アクティビティーが調査ダッシュボードにもレポートにも記録されない

症状

調査ダッシュボードおよび事前定義レポート (「ファイル・アクティビティー」、「ファイル・ライセンス」、「ファイル: クライアントあたりのアクティビティー数」、「ファイル: サーバーあたりのアクティビティー数」、「ファイル: ユーザーあたりのアクティビティー数」、「ファイル: 特権」など) にファイル・アクティビティーが記録されない。

問題の解決

以下を確認します。

- FAM ライセンスがインストールされているか、および S-TAP がアクティブか確認します。
- アクティビティーのファイル・サーバーに root としてログインしていないことを確認します。root (UID0) からのアクティビティーは、デフォルトではログに記録されません。
- Linux/AIX では、ポリシー・ルールで指定されているファイル・パスを確認します。例えば、/testdir/ は、testdir という名前のディレクトリー内のファイルではなくて、testdir という名前のファイルをモニターします。/testdir/* を指定して、testdir ディレクトリー内のファイルをモニターします。
- Windows では、ドメインを使用し、ポリシー・ルールでユーザーを指定する場合、ドメインが指定されていることを確認します。例えば、Maryjane のみではなく svldev¥Maryjane を使用します。

親トピック: ファイル・アクティビティーのトラブルシューティング

取り外し可能ディスクのファイル・アクティビティーが調査ダッシュボードのログに記録されない

症状

取り外し可能ディスクのファイル・アクティビティーが調査ダッシュボードのログに記録されない

環境

FAM_SCAN_EXCLUDE_REMOTE_DIRECTORIES が true に設定されています

問題の解決

取り外し可能ディスクをマウントする*前に*、ファイル・アクティビティー・モニター・ポリシーをインストールしてください。
親トピック: ファイル・アクティビティーのトラブルシューティング

ファイル・アクティビティーがレポートで表示されるが、調査ダッシュボードで表示されない

症状

ファイル・アクティビティーが事前定義レポートには表示されるが、調査ダッシュボードには表示されない。

問題の解決

Guard API を使用して構成を検査します。

- クイック検索にクロール・データを送信するには、以下を使用します: grdapi enable_fam_crawler activity_schedule_interval=2 activity_schedule_units=MINUTE entitlement_schedule_interval=10 entitlement_schedule_units=MINUTE
- (違反も含めるオプションを指定して) クイック検索を有効にするには、以下を使用します: grdapi enable_quick_search includeViolations=true schedule_interval=2 schedule_units=MINUTE

親トピック: ファイル・アクティビティーのトラブルシューティング

分類結果で一部のファイルが欠落する

症状

分類結果で一部のファイルが欠落する。

原因

以下は、分類でサポート*されない*ファイル・タイプです: DAT、JPG、JPEG、GIF、TIF、TIFF、BMP、WAV、MOV、MP3、MP4、AVI、MPG、WMA、WMV、P7S、XFDL、XFD、FRM、JAR。

親トピック: ファイル・アクティビティーのトラブルシューティング

レポートおよび調査ダッシュボードにファイル・ディスカバリー (資格) 結果が一部しか表示されない

レポートおよび調査ダッシュボードにディスカバリー (資格) 結果が一部しか表示されない。

症状

レポートおよび調査ダッシュボードに表示されるディスカバリー (資格) 結果が不完全である。一部のファイルの結果が表示されない。

問題の解決

ディスカバリー用の GIM 構成に、文書のタイプおよびロケーションが含まれていることを確認します。以下の GIM 構成パラメーターを確認します。

- FAM_SCAN_EXCLUDE_FILES
- FAM_SCAN_EXCLUDE_DIRECTORIES
- FAM_SCAN_EXCLUDE_EXTENSIONS
- FAM_SCAN_EXCLUDE_FILES
- FAM_SCAN_MAX_DEPTH

親トピック: ファイル・アクティビティーのトラブルシューティング

レポートおよび調査ダッシュボードでファイル分類結果が欠落する

症状

レポートおよび調査ダッシュボードでファイル分類結果が欠落する。

原因

分類は、メタデータ・ディスカバリーの後に実行される追加プロセスです。

問題の解決

分類に使用される IBM Content Classification エンジン (http://www-01.ibm.com/support/docview.wss?uid=swg27020838) のソフトウェア要件が満たされていると仮定した場合、以下の GIM 構成を確認します。

- GIM パラメーターの FAM_IS_DEEP_ANALYSIS が TRUE であることを確認します。
- FAM_ICM_CLASS_DECISION_PLANS 設定で判定プラン名が正しいこと、および判定プランのリストがセミコロンで区切られていることを確認します。
- リストされているすべての判定プラン (.dpn) ファイルが、ファイル・サーバーの次の場所に存在することを確認します: %FAM_HOME%¥conf¥ContentClassification

ユーザーの処置: オプション。特定のユーザーが実行する特定のアクションがある場合は、1 つ以上の ts*Response エレメントを使用します。

親トピック: ファイル・アクティビティーのトラブルシューティング

ファイル・アクティビティー・ログ

ファイル・アクティビティー・モニター (FAM) ログ・ファイルの配置について説明します。

症状

FAM ログが見つからない、またはデバッグ・レベルを変更できない

問題の解決

FAM ログは以下の場所にあります。

- Windows: FAM エージェント・ログ・ファイルは StapAT.ctl という名前で、C:¥Program Files¥IBM¥Windows S-TAP¥Logs フォルダーにあります。FAM ログは /tmp/guard_stap.fam.txt にあります。これらのログには S-TAP のデバッグ・レベルが反映されますが、このレベルは S-TAP の開始時にのみ変更できます。.ini ファイルで、または FAM デバッグを取得する S-TAP を手動で実行することで、デバッグ・レベルを 4 に設定します。FAM のログおよびユーティリティー出力は V. 10 guard_diag スクリプトで収集されます。
- Linux: FAM エラーおよびデバッグ・ログは guard_stap.fam.txt という名前が付けられています。UNIX でのデフォルトの場所は ¥tmp であり、tap_log_dir によって構成されます。デバッグ・レベルは、tap_debug_output_level によって構成されます。

親トピック: ファイル・アクティビティーのトラブルシューティング

FAM バンドルをインストールできない

GIM クライアントをインストールした後に FAM バンドルのインストールが失敗する。

症状

FAM バンドルをインストールしようとすると、システムが次のようなメッセージで応答する。
```
-1、GIM - 障害点 : dependancy_violation (従属関係違反 (FAM) : 必須の従属関係が欠落しています - GIM.pm 行 3176 の STAP、<MYFILE> 行 20。
(-1,GIM - Failure point : dependancy_violation (Dependancy violation (FAM) : Missing mandatory dependency - STAP at GIM.pm line
3176, <MYFILE> line 20.)
```

原因

FAM バンドルをインストールする前に S-TAP バンドルをインストールする必要があります。

問題の解決

S-TAP for FAM がインストールされていることを確認してから、FAM バンドルをインストールしてください。ファイル・アクティビティー・モニター・コンポーネントのインストールおよびアクティブ化を参照してください。
親トピック: ファイル・アクティビティーのトラブルシューティング

Guardium システムのインストール

- S-TAP のインストール中にチェックサム・エラーが発生する
  チェックサム・エラーを受け取った場合は、FTP クライアントで転送モードをバイナリーに設定します。
- Guardium S-TAP が cp: illegal option - f のエラー・メッセージを返す
  S-TAP のインストールが cp: illegal option - f で失敗した場合は、which cp コマンドを実行し、ファイル・パスを変更します。
- 新規 Guardium パッチのインストールが完了しない
  新規 Guardium パッチのインストールを完了できない場合は、プロセスへの介入を停止し、パッチを再インストールします。
- 新規 Guardium S-TAP のインストール後にファイルまたはディレクトリーが欠落している
- Guardium のインストール時にパーティション・エラーが発生する
  パーティション・エラーを受け取った場合は、「カスタム」インストールを選択し、ディスクのロケーションとサイズを明示的に指定します。
- パッチ・インストールが失敗する: No such file or directory
  パッチ・インストールが失敗した場合は、ファイルが、ダウンロードされたパッチの MD5SUM と一致することを確認します。

親トピック: 問題および解決策

S-TAP のインストール中にチェックサム・エラーが発生する

チェックサム・エラーを受け取った場合は、FTP クライアントで転送モードをバイナリーに設定します。

症状

UNIX または Linux で Guardium S-TAP をインストールするために S-TAP インストーラーを実行したときに、以下のようなエラーを受け取ります。

```
./guard-stap-v81_r26808_1-aix-6.1-aix-powerpc.sh
Verifying archive integrity...Error in checksums: 2082112805 is
different from 3728267449
```

原因

インストーラー・ファイルが破損しています。ファイルがデータベース・サーバーに転送されたとき、または製品がダウンロードされたときに、ファイルが破損しました。

環境

UNIX または Linux 上の S-TAP が影響を受けます。

問題の解決

この問題を解決するには、FTP クライアントで転送モードがバイナリーに設定されていることを確認してください。次に、再度データベースへの転送を試みてください。プロセスが失敗する場合は、製品を再度ダウンロードしてください。

親トピック: Guardium システムのインストール

Guardium S-TAP が cp: illegal option - f のエラー・メッセージを返す

S-TAP のインストールが cp: illegal option - f で失敗した場合は、which cp コマンドを実行し、ファイル・パスを変更します。

症状

S-TAP のインストールが失敗し、以下のエラー・メッセージが表示されます。

```
A directory called 'guardium' containing Guardium software needs to be created under a path provided.
Enter the path prefix [/usr/local]? /opt/guardium
Directory /opt/guardium/guardium/guard_stap does not exist, would you like to create it [Y/n]? Y
Run STAP as root, or as user 'guardium' [R/u]? R
Please be patient... This might take more than a minute.
Copying installation files...
cp: illegal option -- f
UX:vxfs cp: INFO: V-3-21462: Usage: cp [-i] [-p] f1 f2
cp [-i] [-p] f1 ... fn d1
cp [-i] [-p] [-r|-R] [-e { force | ignore | warn}] d1 d2
```

原因

/usr/bin/cp へのパスが、インストーラーが予期していたパスとは異なります。

環境

UNIX/Linux データベース・サーバーが影響を受けます。

問題の解決

which cp コマンドを実行します。

which cp を実行して /usr/bin/cp 以外の値が返された場合は、export PATH=/usr/sbin:/usr/bin:$PATH コマンドを実行します。

which cp コマンドを再実行して、パスが /usr/bin/cp になっていることを確認してください。

親トピック: Guardium システムのインストール

新規 Guardium パッチのインストールが完了しない

新規 Guardium パッチのインストールを完了できない場合は、プロセスへの介入を停止し、パッチを再インストールします。

症状

新規パッチのインストール時に、インストールが完了しません。CLI コマンド show system patch installed の状況列に、以下のいずれかのメッセージが表示されます。

```
STEP: Setting “java” off
STEP: Setting “amei” off
STEP: Setting “sqlw” off
```

原因

マシン上の Tomcat、検査コア、または別のプロセスが、パッチ・インストールに干渉します。

環境

コレクター、アグリゲーター、および中央マネージャーが影響を受けます。

問題の解決

新規 Guardium パッチをインストールするには、すべてのプロセスがインストールに干渉しないようにします。

1. delete scheduled-patch コマンドを使用して、正常にインストールできなかったパッチを削除します。
2. restart system コマンドを使用して、システムを再始動します。
3. システムの再始動後に、stop gui コマンドと stop inspection-core コマンドを使用して、GUI および検査コアを停止します。
4. パッチを再インストールし、restart gui コマンドおよび start inspection-core コマンドを使用して、GUI および検査コアを再始動します。

親トピック: Guardium システムのインストール

新規 Guardium S-TAP のインストール後にファイルまたはディレクトリーが欠落している

症状

S-TAP をインストールしようとしたときに、以下のエラー・メッセージを受け取ります。

```
Tap_controller::init failed Opening pseudo device /dev/guard_ktap No such file or directory
```

さらに、/dev/*ktap* が存在しません。

原因

K-TAP デバイス作成の失敗については、多数の理由が考えられます。最も一般的な原因を以下に示します。

- Linux カーネル用の K-TAP モジュールを含め、モジュール・ファイルを使用しなかった。
- モジュール・ファイルから K-TAP モジュールをロードするための Flex Loading オプションを指定しなかった。
- 古いインストール済み環境の以前の K-TAP モジュールが、引き続き実行されているかインストールされている。

環境

IBM Guardium S-TAP 製品をインストールできるすべての Linux および UNIX オペレーティング・システムが影響を受けます。

問題の解決

この問題を解決するには、以下の手順を実行します。

1. これらのコマンドは、root として実行してください。

   ```
   <STAP directory>/KTAP/guard_ktap_loader stop
   <STAP directory>/KTAP/guard_ktap_loader uninstall
   <STAP directory>/KTAP/guard_ktap_loader install
   <STAP directory>/KTAP/guard_ktap_loader start
   ```

2. ls /dev/*ktap* コマンドによって、K-TAP デバイスが作成されたかどうかを確認します。作成された場合、問題は解決しています。作成されていない場合は、次のステップに進みます。
3. S-TAP プロセス guard_stap が実行されている場合は、これを停止します。ps -ef | grep guard_stap コマンドを使用して、このプロセスが実行されているかどうかを確認できます。
4. ps -ef | grep guard_stap コマンドを使用して、S-TAP プロセスが実行されていないことを確認します。
5. S-TAP をアンインストールします。
6. S-TAP ディレクトリーがなくなっていることを確認します。
7. 古いインストール済み環境の K-TAP モジュールがまだ実行されているかどうかを確認します。ご使用のオペレーティング・システムに該当するコマンドを使用してください。

   ```
   Linux    : lsmod | grep ktap
   Solaris  : modinfo | grep  tap
   HP-UX    : lsdev | grep tap
   AIX      : genkex | grep tap
   ```

   ktap_<release> などのデバイスがリストされている場合、K-TAP モジュールが実行されています。
8. 前のステップで、K-TAP モジュールが実行されていることが分かった場合、以下のステップを実行して、K-TAP モジュールを停止し、アンインストールします。

   ```
   <STAP directory>/KTAP/guard_ktap_loader stop
   <STAP directory>/KTAP/guard_ktap_loader uninstall
   ```

   サーバーを再始動します。
9. Guardium Installation Manager (GIM) を使用する場合、「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」に移動し、クライアントを選択して「接続のリセット」をクリックします。サーバーがクライアント・リストに再表示されるまで待ちます。
10. S-TAP を再インストールします。GIM を使用して S-TAP をインストールする場合は、GIM および以下のコマンドを使用して S-TAP バンドルを再インストールしてください。

    ```
    KTAP-ALLOW_COMBOS=Y
    KTAP_LIVE_UPDATE=Y
    KTAP_ENABLED=Y
    ```

親トピック: Guardium システムのインストール

Guardium のインストール時にパーティション・エラーが発生する

パーティション・エラーを受け取った場合は、「カスタム」インストールを選択し、ディスクのロケーションとサイズを明示的に指定します。

症状

VMWare で Guardium アプライアンスをインストールすると、以下のエラーを受け取ります。

```
Error Partitioning
Could not allocate requested partitions:
Partitioning failed: Could not allocate partitions as primary partitions.
Not enough space left to create partition for /boot.
```

原因

VMWare で Guardium システムをインストールするときに「標準 (Typical)」を選択した場合、VMWare は、VMWare で OS タイプに対して事前定義された構成パラメーターを使用します。これらの構成パラメーターは、このインストールには適さない場合があります。

環境

すべての Guardium 構成 (コレクター、アグリゲーター、中央マネージャー) が影響を受けます。

問題の解決

「カスタム」インストールを選択し、ディスクのロケーションとサイズを明示的に指定します。モニターおよび監査のニーズを満たすために十分な大きさのディスク・サイズを指定します。これが構成された後には、Guardium で、システムにディスク・スペースを追加する操作はサポートされなくなります。

親トピック: Guardium システムのインストール

パッチ・インストールが失敗する: No such file or directory

パッチ・インストールが失敗した場合は、ファイルが、ダウンロードされたパッチの MD5SUM と一致することを確認します。

症状

Guardium でのパッチ・インストールが失敗し、「`patch.reg: No such file or directory`」というエラーが表示されます。

原因

以下のケースでは、パッチ・インストールが失敗する可能性があります。

- パッチがバイナリー・モードでダウンロードされず、ファイルが破損した。
- 圧縮ファイル自体が Guardium システムにアップロードされた。
- Guardium サポートからパッチを受け取り、パッチのファイル名の接頭部として PMR 番号が付加されている。
- パッチが Windows FTP サーバーから Guardium にアップロードされた。

環境

コレクター、アグリゲーター、および中央マネージャーが影響を受けます。

問題の解決

ファイルの内容が、ダウンロードされたパッチの MD5SUM と一致することを確認します。圧縮ファイルを解凍できないか、MD5SUM が一致しない場合は、ファイルをバイナリー・モードでダウンロードしてください。

圧縮ファイル自体が Guardium システムにアップロードされた場合は、圧縮ファイルを解凍し、パッチのみをアップロードしてください。

ファイル名の接頭部として PMR 番号が付加されている場合は、その番号を削除してから、パッチを Guardium システムにアップロードしてください。

パッチが Windows FTP サーバーからアップロードされる場合は、大/小文字を正しく区別して正確なファイル名を指定してください。

親トピック: Guardium システムのインストール

Windows: S-TAP ユーザーズ・ガイド

Guardium S-TAP は、データベース・サーバーおよびファイル・サーバーにインストールされる軽量のソフトウェア・エージェントです。S-TAP によって収集される情報は、Guardium のトラフィック・レポート、アラート、可視化など、すべての基礎となります。

データ・アクティビティーのモニターでは、S-TAP は、クライアントとデータベースの間のアクティビティーをモニターして、その情報を Guardium コレクターに転送します。データベース・トラフィックは、セキュリティー・ポリシーで指定されている基準に基づいてコレクターに記録されます。また、信頼できる接続を無視したり、特定の IP からのトラフィックを無視したりすることで、最初にコレクターに送信されるトラフィックの量を減らすこともできます。

ファイル・アクティビティーのモニターでは、データ・アクティビティーとは異なり、ポリシー・ルールがファイル・サーバーにプッシュダウンされるため、セキュリティー・ポリシーで指定されているデータのみがコレクターに転送されます。

- Windows: S-TAP のインストール、アップグレード、アンインストール
  S-TAP をインストール、アップグレード、およびアンインストールするにはいくつかの方法があります。それぞれについて確認し、どれが最適かを判断してください。
- Windows: S-TAP の構成
  S-TAP の構成について説明します。
- Windows: S-TAP の操作とパフォーマンス

Windows: S-TAP のインストール、アップグレード、アンインストール

S-TAP をインストール、アップグレード、およびアンインストールするにはいくつかの方法があります。それぞれについて確認し、どれが最適かを判断してください。

- Windows: S-TAP モニター・メカニズムのサポート・マトリックス
  モニターまたはブロックするデータに応じて、使用する S-TAP セットアップを選択します。以下の表を使用して、オペレーティング・システムおよびデータベースごとに、必要な操作を実行可能なモニター・メカニズムを判別してください。
- Windows: 前提条件: S-TAP のインストール
  S-TAP をインストールする前に、ディスク・スペースおよびポートの前提条件を確認してください。
- Windows: S-TAP エージェントのインストール
  Guardium Installation Manager (GIM) モニター・エージェント・ツール、GIM の「クライアント別の設定」、対話式インストーラー、またはコマンド行インストーラーを使用して、S-TAP を Windows にインストールします。
- Windows: Oracle RAC での S-TAP のインストールの流れ
  Oracle RAC で S-TAP を構成します。
- Windows: S-TAP のアップグレードと削除
  ここでは、Windows 上で S-TAP のアップグレードと削除を行う方法について説明します。
- Windows: S-TAP のインストール後またはアップグレード後にデータベースを再始動またはリブートするタイミング
  Windows S-TAP のインストールやアップグレードでは、リリース・ノートや本書で例外として他の記述がない限り、データベース・サーバーのリブートは必要ありません。
- Windows: データベースをアップグレードする際の S-TAP の管理
  以下の指針を使用して、データベースのアップグレード時に Windows S-TAP を管理します。
- Windows: データベースのオペレーティング・システムをアップグレードする際の S-TAP の管理
  以下のガイドラインを使用して、データベースのオペレーティング・システム (OS) のアップグレード時に、対話式インストーラーまたは CLI を使用してインストールされた S-TAP を管理します。

親トピック: Windows: S-TAP ユーザーズ・ガイド

Windows: S-TAP モニター・メカニズムのサポート・マトリックス

モニターまたはブロックするデータに応じて、使用する S-TAP セットアップを選択します。以下の表を使用して、オペレーティング・システムおよびデータベースごとに、必要な操作を実行可能なモニター・メカニズムを判別してください。

例えば、以下の項目の 1 つ以上をトラッキングする必要が生じる場合があります。

- ローカル・トラフィックのみ
- ローカル・トラフィックおよびネットワーク・トラフィック
- 共有メモリー
- 暗号化されたデータ
- モニターおよびブロック
- モニターのみ

以下の表では、Guardium のモニター・メカニズムによってサポートされる、最も一般的なプラットフォーム、データベース・タイプ、およびプロトコルを取り上げています。この表は一般ガイドラインを示しています。ここには示されていない他のサポート対象の組み合わせが存在する場合があります。ここに示されているサポート対象のセットアップの一部は、特定の構成に依存する場合があります。特定のニーズに最も適したセットアップを確認するには、技術サポートにお問い合わせください。空のセルは、その組み合わせがサポートされないことを示しています。

OS	データベース	ネットワーク・トラフィック	ローカル・トラフィック	暗号化されたトラフィック	プロトコル	Kerberos	ブロッキング	編集	インスタンス・ディスカバリー
Windows	MS SQL Server	サポート	サポート	TCP と NMP に対してサポート	TCP, NMP	サポート	サポート	サポート	サポート
Windows	Db2	サポート (Db2 出口でもサポート)	サポート (Db2 出口でもサポート)	Db2 出口	TCP, SHM		サポート (Db2 出口を除く)	サポート (Db2 出口を除く)	サポート
Windows	Oracle	サポート	サポート	サポート (ASO、SSL)	TCP、NMP、BEQ		サポート	サポート	サポート
Windows	Informix	サポート	サポート		TCP		サポート	サポート	サポート
Windows	Sybase	サポート	サポート		TCP		サポート	サポート	
Windows	MySQL	サポート	サポート		TCP		サポート	サポート	
Windows	PostgreSQL	サポート	サポート		TCP		サポート	サポート	
Windows	MongoDB	サポート	サポート		TCP		サポート	サポート	サポート
Windows	CouchDB	サポート	サポート		TCP		サポート	サポート	サポート

親トピック: Windows: S-TAP のインストール、アップグレード、アンインストール

Windows: 前提条件: S-TAP のインストール

S-TAP をインストールする前に、ディスク・スペースおよびポートの前提条件を確認してください。

- Windows: S-TAP のディスク・スペース所要量
  S-TAP をインストールする前に、ディスク・スペース所要量を確認してください。
- Windows: S-TAP の Guardium ポート要件
  Guardium® のコンポーネント間 (例えば、Guardium システムと Windows データベース・サーバー上の S-TAP との間) にファイアウォールがある場合、それらのコンポーネント間の接続に使用されるポートがブロックされていないことを確認する必要があります。

親トピック: Windows: S-TAP のインストール、アップグレード、アンインストール

Windows: S-TAP のディスク・スペース所要量

S-TAP をインストールする前に、ディスク・スペース所要量を確認してください。

ディスク・スペース	記述
S-TAP プログラム・ファイル	S-TAP は、Microsoft .NET Framework を使用します。これがまだインストールされていない場合、5 GB の空きスペースを必要とします。 GIM インストール: 300 MB 非 GIM インストール: 180 MB
バッファー・ファイル	50 MB

親トピック: Windows: 前提条件: S-TAP のインストール

Windows: S-TAP の Guardium ポート要件

Guardium® のコンポーネント間 (例えば、Guardium システムと Windows データベース・サーバー上の S-TAP との間) にファイアウォールがある場合、それらのコンポーネント間の接続に使用されるポートがブロックされていないことを確認する必要があります。

ファイアウォール管理ユーティリティーを使用して、以下にリストされているポートを確認し、必要に応じてオープンにします。

表 1. Windows サーバーのポート要件

ポート	プロトコル	Guardium システムの接続先
9500/9501	TCP	アライブ・メッセージ
9500	TCP	クリア S-TAP
9501	TLS	暗号化された S-TAP

親トピック: Windows: 前提条件: S-TAP のインストール

Windows: S-TAP エージェントのインストール

Guardium Installation Manager (GIM) モニター・エージェント・ツール、GIM の「クライアント別の設定」、対話式インストーラー、またはコマンド行インストーラーを使用して、S-TAP を Windows にインストールします。

ライセンス・キーによっては、ファイルとデータベースの両方のアクティビティー・モニターに同じ S-TAP エージェントを使用できます。FAM の固有 S-TAP パラメーターはありません。

S-TAP のインストールには、Base Filtering Engine (BFE) サービスが実行されている必要があります。サービスが存在しているが実行されていない場合、Guardium はそれを開始しようとします。

S-TAP には .NET Framework 4.5 以上が必要です。.NET 4.5 以上の環境が存在しない場合、S-TAP は .NET 4.5.2 をインストールします。

非 ASCII 環境 (例えば、日本語) に Windows S-TAP をインストールする際、その言語パックが含まれているサーバーを使用するか、システム・ロケールをその場所 (日本) に設定します。

S-TAP のインストールによって、C:¥IBM Windows S-TAP.ctl の下に 1 つのインストール・ログが作成されます。

データベース・インスタンスのオートディスカバリー

S-TAP をインストールするときに、データベース・インスタンスのオートディスカバリーを指定するオプションと、ディスカバーされたインスタンスの検査エンジンを作成するオプションを選択できます。有効な場合、オートディスカバリー・プロセスは、S-TAP のインストール時に 1 回だけ実行されます。自動的に繰り返し実行されることはありません。オートディスカバリーはデフォルトで無効になっています。

オートディスカバリーは、データベース・タイプとして MS SQL Server、Db2、Oracle、Informix、MongoDB、CouchDB をサポートします。ディスカバーされた他のデータベースで検査エンジンを作成するには、「ディスカバーされたインスタンス」レポートを参照してください。

アップグレード中に、オートディスカバリーによって追加のデータベース・インスタンスがディスカバーされますが、新規インスタンス用の検査エンジンは作成されません。

S-TAP のインストールにおいてインストール時またはアップグレード時にデータベースのオートディスカバリーを実行したくない場合は、それぞれの Windows S-TAP インストーラーに対して記述されている手順を実行すると、S-TAP のインストール・プロセスでデータベースのオートディスカバリーが実行されなくなります。

エンタープライズ・ロード・バランシング

S-TAP を Windows にインストールする際に、エンタープライズ・ロード・バランシング機能を使用するように S-TAP を構成することができます。詳しくは、エンタープライズ・ロード・バランシングを参照してください。

- Windows: GIM の「クライアント別の設定」を使用した S-TAP エージェントのインストール
  GIM の「クライアント別の設定」を使用して S-TAPs をデータベース・サーバーにインストールする場合、個別のサーバーまたはサーバーのグループに対して、エージェントのインストール、アップグレード、および管理を行うことができます。これには、その制御下でインストールされた各種プロセスのモニター、S-TAP パラメーターの変更、その他の管理タスクの実行が含まれます。
- Windows: S-TAP の GIM インストールのパラメーター
  GIM のインストールで一般的に使用されるパラメーター (およびその簡略説明) を示します。
- Windows: 対話式インストーラーを使用して S-TAP エージェントをインストールする
  小規模なデプロイメントの場合や、ガイドに従い手順を追ってインストールを行う必要がある場合は、対話式インストーラーを使用すると便利です。
- Windows: コマンド行インターフェースを使用して S-TAP エージェントをインストールする
  コマンド行インストーラーを使用すると、スクリプト可能ソリューションが提供されます。このソリューションは、大規模なデプロイメント環境を管理する場合に特に便利です。
- Windows: S-TAP コマンド・ライン・インストールのパラメーター
  スクリプトと GIM のインストールで使用できるパラメーター (それぞれに簡略説明あり) について説明します。

親トピック: Windows: S-TAP のインストール、アップグレード、アンインストール
関連概念:
モニター・エージェントをデプロイするためのクイック・スタート
Guardium Installation Manager

Windows: GIM の「クライアント別の設定」を使用した S-TAP エージェントのインストール

GIM の「クライアント別の設定」を使用して S-TAPs をデータベース・サーバーにインストールする場合、個別のサーバーまたはサーバーのグループに対して、エージェントのインストール、アップグレード、および管理を行うことができます。これには、その制御下でインストールされた各種プロセスのモニター、S-TAP パラメーターの変更、その他の管理タスクの実行が含まれます。

始める前に

インストールを開始する前に、以下の点を確認してください。

- Windows: 前提条件: S-TAP のインストールに記載されている Windows S-TAP のインストール要件を確認します。
- サポート対象のデータベース・サーバーとオペレーティング・システムを使用しています。
- 予定している S-TAP のインストール・ディレクトリーが空になっているか存在していません。
- GIM クライアントは、S-TAP をインストールするデータベース・サーバーにインストールされます。
- データベース・サーバー上の GIM クライアントは、Guardium システムと通信を行います。
- Fix Central または Guardium 担当員から S-TAP モジュールを入手します。

このタスクについて

GIM クライアントをデータベース・サーバーにインストール後、S-TAP for Windows のインストールを Guardium システムからスケジュールします。

必須パラメーターは WINSTAP_INSTALL_DIR のみです。

パラメーター WINSTAP_INSTALL_DIR はインストール後に変更できません。他のすべてのパラメーターは、インストール後に変更できます。

パラメーターの入力は、「クライアント別の設定」ページまたは「パラメーターの選択」リボンで行うか、[TAP] パラメーターの場合はコマンド WINSTAP_CMD_LINE で構文 parameter=value を使用して行うか、CLI パラメーターの場合は構文 -param value を使用して行います (Windows: S-TAP コマンド・ライン・インストールのパラメーターを参照)。これにより、guard_tap.ini にパラメーターが追加されるか、パラメーターが更新されます。

注意:
WINSTAP_CMD_LINE の使用時には入力の検証が行われません。

手順

1. インストールする Windows S-TAP モジュールをアップロードします。
   a. Guardium システムで、「管理」 > 「モジュール・インストール」 > 「モジュールのアップロード」にナビゲートします。
   b. 「ファイルの選択 (Choose File)」をクリックし、インストールする S-TAP モジュールを選択します。
   c. 「アップロード」をクリックして、モジュールを Guardium システムにアップロードします。 アップロードが完了すると、「アップロード済みモジュールのインポート」表にモジュールが表示されます。
   d. 「アップロード済みモジュールのインポート」表で、インストールする S-TAP モジュールの横にあるチェック・ボックスをクリックします。 モジュールがインポートされ、インストール可能な状態になります。モジュールのインポートが完了すると、「モジュールのアップロード」ページがリセットされ、「アップロード済みモジュールのインポート」表が空になります。
2. クライアント別の設定 の GIM の説明に従って操作し、Windows: S-TAP の GIM インストールのパラメーターを確認します。
   - ほとんどのインストールではデフォルトのパラメーターを適用できますが、WINSTAP_INSTALL_DIR 値は指定する必要があります。このデフォルト値は C:/Program Files/IBM/Windows S-TAP です。これは唯一の必須パラメーターです。
   - WINSTAP_TAP_IP (コマンド行パラメーターの -taphost と同じ) を指定しなかった場合、GIM_CLIENT_IP 値が使用されます。
   - WINSTAP_SQLGUARD_IP (コマンド行パラメーターの -appliance と同じ) を指定しなかった場合、GIM_URL 値が使用されます。
   - オプションで、エンタープライズ・ロード・バランシングを有効にします。Windows: S-TAP の GIM インストールのパラメーターでパラメーターの説明を参照してください。
   - データベース・インスタンスのオートディスカバリーを有効にするには、WINSTAP_NOAUTODISCOVERY を 0 に設定します。

次のタスク

「成功」ポップアップで、「状況の表示」をクリックして、「状況」ウィンドウを開き、ソフトウェアのインストール/アップグレードをモニターします。 をクリックして、結果を最新表示します。インストール/アップグレードが失敗状況の場合、ボタンが表示されていれば、「アンインストール」をクリックします。表示されていない場合は、「接続のリセット」をクリックします。「管理」 > 「レポート」 > 「インストール管理」 > 「GIM クライアント状況」でレポートを表示して、モジュールのインストール状況を確認することもできます。

S-TAP が Guardium システムと通信していることを確認します。これを行うには、「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」にナビゲートし、S-TAPs の状況と構成を確認します。

親トピック: Windows: S-TAP エージェントのインストール
関連概念:
Guardium Installation Manager

Windows: S-TAP の GIM インストールのパラメーター

GIM のインストールで一般的に使用されるパラメーター (およびその簡略説明) を示します。

すべてのパラメーターはWindows: S-TAP 構成パラメーターの編集にリストされています。
注意:
熟練したユーザーである場合や IBM 技術サポートに相談済みの場合を除いて、拡張パラメーターは変更しないでください。

表 1. すべての .NET インストーラーに適用可能なパラメーター

GIM パラメーター	記述
QUIET	サイレント・インストールを実行します (値は必要ありません)。
WINSTAP_INSTALL_DIR	これはインストール・ディレクトリーです。デフォルトのインストール・パスは C:/Program Files/IBM/Windows S-TAP です。
WINSTAP_ENABLEGAM	Guardium Agent Monitor (GAM) サービスを有効にします。

表 2. その他の S-TAP パラメーター

GIM パラメーター	記述
WINSTAP_ENABLEGAM	Guardium Agent Monitor (GAM) サービスを有効にします。
WINSTAP_TAP_IP	ローカル/クライアント IP。無人インストールには必須です。
WINSTAP_SQLGUARD_IP	SQLGUARD IP。このパラメーターを複数回、それぞれ固有の値で指定することにより、複数のアプライアンスをセットアップできます。
WINSTAP_FAM_ENABLED	FAM サービスを有効にします。デフォルトでは無効です。アップグレードする際に、v10.1.4 以前のバージョンで guard_tap.ini のパラメーター fam_enable が有効になっていた場合は、アップグレード時にこのパラメーターが有効になります。

表 3. 適用可能な値「ON」を持つ S-TAP パラメーター. 以下のパラメーターの値は、デフォルトで「ON」に設定されており、有効になっています。特に記載がない限り、これらのパラメーターを「ON」以外の値に設定すると、そのパラメーターが無効になります。

GIM パラメーター	記述
TCP_DRIVER_INSTALLED	TCP_DRIVER_INSTALLED=1。TCP ドライバーを使用します。
NAMED_PIPE_DRIVER_INSTALLED	NAMED_PIPE_DRIVER_INSTALLED=1。ローカル・アクセス用に MS SQL Server で使用される名前付きパイプを指定します。名前付きパイプが使用される場合、このパラメーターに何も指定しなければ、S-TAP はレジストリーから名前付きパイプ名を取得しようとします。
DB2_TAP_INSTALLED	Db2 共有メモリー・トラフィックのスニッフィングを有効にします。
DB2_EXIT_DRIVER_INSTALLED	Db2 と S-TAP の統合を有効にします。
FAM_DRIVER_INSTALLED	FAM S-TAP を有効にします。
ORA_DRIVER_INSTALLED	Oracle ASO および SSL のトラフィックのスニッフィングを有効にします。
KRB_MSSQL_DRIVER_INSTALLED	v10.1.4 から非推奨になっています。guard_tap.ini ファイルには出現しますが、構成には影響しません。 このパラメーターは、MSSQL SSL および Kerberos の暗号化トラフィックの暗号化解除に使用されます。MSSQL 暗号化トラフィックおよび Kerberos チケットを収集するには、1 または 2 に設定します。1 に設定される場合、STAP の始動時に、SID と相関関係のあるユーザー名を事前収集します。krb_mssql_driver_user_collect_time に定義されている秒数にわたって収集します。2 に設定される場合、事前収集は行われず、ユーザー名の相関は実行時に行われます。

表 4. エンタープライズ・ロード・バランシング・パラメーター

GIM パラメーター	記述
WINSTAP_LOAD_BALANCER_IP	ロード・バランシングを構成する場合は必須です。 このオプションにより、この S-TAP がロード・バランシングで使用する中央マネージャーまたは管理対象ユニットの IP アドレスを指定します。 • S-TAP パラメーターは、アップグレード中に対話式インストーラーを使用して変更することはできません。S-TAP パラメーターを変更するには、アップグレード後に Guardium UI を使用します。 • エンタープライズ・ロード・バランサーを管理対象ユニット上で実行するように構成する場合、S-TAP が V10.1 以上でなければなりません。
WINSTAP_INITIAL_BALANCER_TAP_GROUP	オプション。この S-TAP が属するエンタープライズ・ロード・バランシング用のアプリケーション・グループ名。 重要: スペースまたは特殊文字を含むグループ名はサポートされません。

GIM パラメーター	記述
WINSTAP_INITIAL_BALANCER_MU_GROUP	オプション。app-group を関連付ける MU グループ名。定義されている LB-APP-GROUP を必要とします。S-TAP のインストール中に使用できるようにするには、前もって中央マネージャーに MU グループが存在している必要があります。 重要: スペースまたは特殊文字を含むグループ名はサポートされません。
WINSTAP_LOAD_BALANCER_NUM_MUS	エンタープライズ・ロード・バランサーがこの S-TAP に割り振る管理対象ユニットの数。

親トピック: Windows: S-TAP エージェントのインストール

Windows: 対話式インストーラーを使用して S-TAP エージェントをインストールする

小規模なデプロイメントの場合や、ガイドに従い手順を追ってインストールを行う必要がある場合は、対話式インストーラーを使用すると便利です。

始める前に

インストールを開始する前に、以下の点を確認してください。

- Windows: 前提条件: S-TAP のインストールに記載されている Windows S-TAP のインストール要件を確認します。
- サポート対象のデータベース・サーバーとオペレーティング・システムを使用していることを確認します。『System Requirements/ Platforms supported for IBM Guardium』(http://www-01.ibm.com/support/docview.wss?uid=swg27047801) およびWindows: S-TAP モニター・メカニズムのサポート・マトリックスを参照してください。
- S-TAP のインストール先となるデータベース・サーバーまたはドメイン・コントローラーの IP アドレスを特定します (仮想 IP アドレスを含む)。
- S-TAP を制御する Guardium システムの IP アドレスを特定します。
- 予定している S-TAP のインストール・ディレクトリーが空になっているか存在しないことを確認します。
- Fix Central または Guardium 担当員から S-TAP モジュールを入手します。

このタスクについて

データベース・サーバーに S-TAP をインストールする場合は、S-TAP からデータを受信する Guardium システムの IP アドレスまたはホスト名を指定する必要があります。S-TAP が Guardium システムに接続されたら、「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」ページにナビゲートして S-TAP を構成します。

注: Windows S-TAP パラメーターは、アップグレード中に対話式インストーラーを使用して変更することはできません。ユーザーは、アップグレードの後に GUI を使用して Windows S-TAP パラメーターを変更することができます。

手順

1. システム管理者のアカウントを使用して、データベース・サーバーにログオンします。
2. S-TAP モジュールをデータベースにコピーし、Guardium の Windows S-TAP インストール・ウィザードを開始します。
   重要: S-TAP を Windows 2012 以降にインストールする場合は、管理者特権を使用する必要があります。その場合は、インストーラーを右クリックして「管理者として実行」を選択します。
3. 「Guardium ライセンス (Guardium License)」画面で使用条件を確認します。インストールを続行する場合は、「使用条件の条項に同意します」を選択して「次へ」をクリックします。
4. 必要な情報を「お客様情報 (Customer Information)」画面で入力し、「次へ」をクリックして操作を続行します。ほとんどのインストールでは、デフォルト値が適しています。
5. 以下に示すいずれかのインストール・タイプを選択し、「次へ」をクリックして操作を続行します。
   - 標準 (Typical): 標準インストールは、ほとんどのユーザーに適しています。
   - 簡易 (Compact): 簡易インストールでは、エンタープライズ・ロード・バランシングなどの追加機能が必要ないことが想定されます。
   - カスタム: カスタム・インストールを選択すると、追加の S-TAP インストール・オプション (ソフトウェアの選択、インストール・ディレクトリーの指定、Windows S-TAP プロセスを実行するユーザー・アカウントの指定など) を変更することができます。
6. 必要に応じて、「ロード・バランシング・オプション (Load Balancing Options)」画面の「ロード・バランシングを有効にする (Enable Load Balancing)」チェック・ボックスを選択してエンタープライズ・ロード・バランシングを有効にします。「次へ」をクリックして先に進みます。
   a. エンタープライズ・ロード・バランシングを有効にする場合は、「ロード・バランサーのホスト・アドレス (Load Balancer Host Address)」フィールドで、ロード・バランサーの IP アドレスを指定します。
   b. 「拡張オプション (Advanced Options)」ボタンをクリックし、追加のエンタープライズ・ロード・バランシング・オプションを指定します。 詳しくは、エンタープライズ・ロード・バランシングを参照してください。
7. 「ネットワーク・アドレス」画面で、「S-Tap ホスト・アドレス (Software Tap Host Address)」を確認して「アプライアンス・アドレス (Appliance Address(es))」を指定し、「次へ」をクリックして操作を続行します。
   - S-Tap ホストのアドレスにより、S-TAP のインストール先となるローカル・マシンのアドレスが指定されます。
   - アプライアンスのアドレスにより、S-TAP を制御する Guardium システムのアドレスが指定されます。S-TAP のフェイルオーバー・システムを設定する場合や、participate_in_load_balancing パラメーターを使用して S-TAP のロード・バランシングを構成する場合は、複数のアドレス (通常は 3 つ以内) を個別の行に指定します。

   重要: インストール後に S-TAP サービスを有効にしたくない場合は、「S-Tap サービスの開始 (Start S-Tap Service)」チェック・ボックスの選択を解除します。「S-Tap サービスの開始 (Start S-Tap Service)」チェック・ボックスの選択を解除すると、データベースのオートディスカバリーと検査エンジンの作成も無効になります。

   インストールが正常に完了すると、「インストール・ウィザードの完了 (Install Wizard Completed)」画面が表示されます。
8. 「完了」をクリックして、インストーラーを閉じます。

次のタスク

S-TAP が Guardium システムと通信していることを確認します。これを行うには、「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」にナビゲートし、S-TAPs の状況と構成を確認します。

親トピック: Windows: S-TAP エージェントのインストール

Windows: コマンド行インターフェースを使用して S-TAP エージェントをインストールする

コマンド行インストーラーを使用すると、スクリプト可能ソリューションが提供されます。このソリューションは、大規模なデプロイメント環境を管理する場合に特に便利です。

始める前に

インストールを開始する前に、以下の点を確認してください。

- Windows: 前提条件: S-TAP のインストールに記載されている Windows S-TAP のインストール要件を確認します。
- サポート対象のデータベース・サーバーとオペレーティング・システムを使用していることを確認します。『System Requirements/ Platforms supported for IBM Guardium』(http://www-01.ibm.com/support/docview.wss?uid=swg27047801) およびWindows: S-TAP モニター・メカニズムのサポート・マトリックスを参照してください。
- S-TAP のインストール先となるデータベース・サーバーまたはドメイン・コントローラーの IP アドレスを特定します (仮想 IP アドレスを含む)。
- S-TAP を制御する Guardium システムの IP アドレスを特定します。
- 予定している S-TAP のインストール・ディレクトリーが空になっているか存在しないことを確認します。
- Fix Central または Guardium 担当員から S-TAP モジュールを入手します。

手順

1. システム管理者のアカウントを使用して、データベース・サーバーにログオンします。
2. インストーラーをデータベースにコピーし、Windows コマンド・プロンプトを使用して、Windows S-TAP インストーラーのディレクトリーに移動します。 例えば、以下のように入力します。

   ```
   cd c:¥Windows-STAP-V10.6.0.0.89
   ```

   このインストーラー・ディレクトリーに setup.exe 実行可能ファイルが格納されています。
3. setup.exe 実行可能ファイルで適切なパラメーターを指定して、S-TAP をインストールします。 必須パラメーターは以下のとおりです。
   - INSTALLPATH。指定しない場合はデフォルトが使用されます
   - TAPHOST
   - APPLIANCE

   INSTALLPATH を除くすべてのパラメーターは、インストール後に更新できます。標準的なインストール・コマンドは以下のとおりです。

   ```
   setup.exe -UNATTENDED -APPLIANCE 10.0.147.234 -TAPHOST 10.0.145.41
   ```

   ここで:
   - -UNATTENDED: これは、コマンド行インストーラーを起動するための必須パラメーターです。
   - -APPLIANCE: これは、S-TAP を制御する Guardium システムの IP アドレスを指定するためのパラメーターです。
   - -TAPHOST: これは、S-TAP のインストール先となるクライアントの IP アドレスを指定するための必須パラメーターです。

   setup.exe 実行可能ファイルとそのパラメーターの詳細な説明については、Windows: S-TAP コマンド・ライン・インストールのパラメーターを参照してください。

次のタスク

S-TAP が Guardium システムと通信していることを確認します。これを行うには、「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」にナビゲートし、S-TAPs の状況と構成を確認します。

親トピック: Windows: S-TAP エージェントのインストール
関連資料:
Windows: S-TAP コマンド・ライン・インストールのパラメーター

Windows: S-TAP コマンド・ライン・インストールのパラメーター

スクリプトと GIM のインストールで使用できるパラメーター (それぞれに簡略説明あり) について説明します。

CLI インストールでは、適切なパラメーターを次の形式で指定して、setup.exe 実行可能ファイルを使用することで、S-TAP をインストールします。

Setup.exe -PARAMETER value

値をパラメーターに割り当てる場合、「=」記号は使用しないでください。「=」を使用するのは、コマンド行でパラメーターを入力するように、guard_tap.ini ファイルの TAP セクションにパラメーターを直接追加する場合だけです。

ここに指定されていないパラメーターを guard_tap.ini ファイルに追加する必要がある場合は、以下のように、「=」記号を使用してそのパラメーターと値を指定することにより、[TAP] セクションを追加できます。

setup.exe -UNATTENDED -INSTALLPATH "C:/Program Files/IBM/Windows S-TAP" -APPLIANCE 10.0.148.160 -TAPHOST 10.0.146.160 QRW_INSTALLED=0 QRW_DEFAULT_STATE=0

重要: TAPHOST、APPLIANCE、INSTALLPATH は必須属性です。

表 1. すべての .NET インストーラーに適用可能なパラメーター

コマンド行パラメーター	GIM パラメーター	記述
UNATTENDED	QUIET	サイレント・インストールを実行します (値は必要ありません)。
INSTALLPATH	WINSTAP_INSTALL_DIR	これはインストール・ディレクトリーです。デフォルトのインストール・パスは C:/Program Files/IBM/Windows S-TAP です。
ENABLEGAM	WINSTAP_ENABLEGAM	Guardium Agent Monitor (GAM) サービスを有効にします。
UNINSTALL		アンインストールします。値は不要です。

コマンド行パラメーター	GIM パラメーター	記述
CUSTOMER		カスタマー名を変更する場合に使用します。
COMPANY		会社名を変更する場合に使用します。
SERVICEUSER		サービスを実行するユーザーを指定する場合に使用します。
SERVICEPASSWORD		ユーザーのパスワードを指定します。

表 2. その他の S-TAP パラメーター

コマンド行パラメーター	記述
NOAUTODISCOVERY	インストール時にオートディスカバリーを実行したくない場合に使用します。値は不要です。
ENABLEGAM	Guardium Agent Monitor (GAM) サービスを有効にします。
START	インストール後に S-TAP を開始するかどうかを制御します。 重要: このパラメーターは、デフォルトで有効になっています。このパラメーターは、値を 0 に設定することでのみ無効にできます。0 以外の値を設定すると、このパラメーターは有効になります。
TAPHOST	ローカル/クライアント IP。無人インストールには必須です。
APPLIANCE	SQLGUARD IP。このパラメーターを複数回、それぞれ固有の値で指定することにより、複数のアプライアンスをセットアップできます。
FAM	FAM を有効にします。デフォルトでは無効です。アップグレードする際に、v10.1.4 以前のバージョンで CLI のパラメーター FAM が有効になっていた場合は、アップグレード時にこのパラメーターが有効になります。

表 3. 適用可能な値「ON」を持つ S-TAP パラメーター. 以下のパラメーターの値は、デフォルトで「ON」に設定されており、有効になっています。特に記載がない限り、これらのパラメーターを「ON」以外の値に設定すると、そのパラメーターが無効になります。

コマンド行パラメーター	記述
TCP	TCP ドライバーを使用します。
NMP	ローカル・アクセス用に MS SQL Server で使用される名前付きパイプを指定します。名前付きパイプが使用される場合、このパラメーターに何も指定しなければ、S-TAP はレジストリーから名前付きパイプ名を取得しようとします。
DB2SHMEM	Db2 共有メモリー・トラフィックのスニッフィングを有効にします。
DB2EXIT	Db2 と S-TAP の統合を有効にします。
ORACLEPLUGIN	Oracle ASO および SSL のトラフィックのスニッフィングを有効にします。
MSPLUGIN	v10.1.4 から非推奨になっています。guard_tap.ini ファイルには出現しますが、構成には影響しません。 このパラメーターは、MSSQL SSL および Kerberos の暗号化トラフィックの暗号化解除に使用されます。MSSQL 暗号化トラフィックおよび Kerberos チケットを収集するには、1 または 2 に設定します。1 に設定される場合、STAP の始動時に、SID と相関関係のあるユーザー名を事前収集します。krb_mssql_driver_user_collect_time に定義されている秒数にわたって収集します。2 に設定される場合、事前収集は行われず、ユーザー名の相関は実行時に行われます。

表 4. エンタープライズ・ロード・バランシング・パラメーター

コマンド行パラメーター	GIM パラメーター	記述
LOAD-BALANCER-IP	WINSTAP_LOAD_BALANCER_IP	ロード・バランシングを構成する場合は必須です。 このオプションにより、この S-TAP がロード・バランシングで使用する中央マネージャーまたは管理対象ユニットの IP アドレスを指定します。 • S-TAP パラメーターは、アップグレード中に対話式インストーラーを使用して変更することはできません。S-TAP パラメーターを変更するには、アップグレード後に Guardium UI を使用します。 • エンタープライズ・ロード・バランサーを管理対象ユニット上で実行するように構成する場合、S-TAP が V10.1 以上でなければなりません。
LB-APP-GROUP	WINSTAP_INITIAL_BALANCER_TAP_GROUP	オプション。この S-TAP が属するエンタープライズ・ロード・バランシング用のアプリケーション・グループ名。 重要: スペースまたは特殊文字を含むグループ名はサポートされません。
LB-MU-GROUP	WINSTAP_INITIAL_BALANCER_MU_GROUP	オプション。app-group を関連付ける MU グループ名。定義されている LB-APP-GROUP を必要とします。S-TAP のインストール中に使用できるようにするには、前もって中央マネージャーに MU グループが存在している必要があります。 重要: スペースまたは特殊文字を含むグループ名はサポートされません。
LB-NUM-MUS	WINSTAP_LOAD_BALANCER_NUM_MUS	エンタープライズ・ロード・バランサーがこの S-TAP に割り振る管理対象ユニットの数。

親トピック: Windows: S-TAP エージェントのインストール

Windows: Oracle RAC での S-TAP のインストールの流れ

Oracle RAC で S-TAP を構成します。

手順

1. すべてのノードに S-TAP をインストールします。GIM が使用されている場合は、すべてのノードに GIM クライアントをインストールしてから、すべてのノードに S-TAP をインストールします。

2. STAP パラメーター STAP_TAP_IP (ノード用に構成されるパブリック IP) を構成します。(GIM UI を使用して構成できます。)
    - パラメーター STAP_ALTERNATE_IPS は不要です。
    - Oracle データベースが暗号化 (ASO/SSL) されている場合は、パラメーターが ORA_DRIVER_INSTALLED=1 であることを確認してください。
    - Oracle 検査エンジンが自動検出された場合、INSTANCE_NAME を含むすべての必須パラメーターが既に含まれています。

親トピック: Windows: S-TAP のインストール、アップグレード、アンインストール

Windows: S-TAP のアップグレードと削除

ここでは、Windows 上で S-TAP のアップグレードと削除を行う方法について説明します。

親トピック: Windows: S-TAP のインストール、アップグレード、アンインストール

このタスクについて

コマンド行を使用して Windows S-TAP をアップグレードする

このタスクについて

旧バージョンの Windows S-TAP がインストールされている場合は、設定プログラムを使用して、コマンド行からアップグレードを実行できます。

STAP の一部として実行されるオートディスカバリーがあります。これはローカル・データベースを検索し、次の 2 つの処理を行います。最初に、一部のサポート対象データベース・タイプ用の検査エンジンを作成します。次に、後で使用するためにデータベース情報をアプライアンスにアップロードします。現在の動作では、この情報は一定の間隔でアプライアンスに送信されます。ただし、v10 の早期では、STAP をアップグレードすると、既存の検査エンジンも上書きされていました。この動作は修正され、オートディスカバリーで検査エンジンが更新されるのではなく、データベース情報が一定の間隔でアプライアンスに送信され続けるようになりました。

手順

1. システム管理者のアカウントを使用して、データベース・サーバー・システムにログオンします。
2. S-TAP の設定プログラムが格納されているディレクトリーに移動します。
3. オプション setup -UNATTENDED を使用して、設定プログラムを実行します。
   重要: 以前のリリースの一部のファイルは、次回にスケジュールされているリブートが実行されるまで、完全には削除されません。

「プログラムの追加と削除」を使用して Windows S-TAP を削除する

このタスクについて

この手順では、構成ファイルを将来使用できるように確実に保存し、インストールされている S-TAP を削除します。

手順

1. システム管理者のアカウントを使用して、データベース・サーバー・システムにログオンします。
2. 現在の S-TAP 構成ファイルを安全なロケーション (Guardium 以外のディレクトリー) にコピーします。このファイルは C:Program Files (x86)¥IBM¥Windows S-TAP¥Bin¥guard_tap.ini で見つけてください。
3. 「プログラムの追加と削除 」制御パネルから、「GUARDIUM_STAP」を削除します。
   重要: 一部のファイルは、次回にスケジュールされているリブートが実行されるまで、完全には削除されません。

コマンド行を使用して Windows S-TAP を削除する

このタスクについて

この手順では、構成ファイルを将来使用できるように確実に保存し、インストールされている S-TAP を削除します。

手順

1. システム管理者のアカウントを使用して、データベース・サーバー・システムにログオンします。
2. 現在の S-TAP 構成ファイルを安全なロケーション (Guardium 以外のディレクトリー) にコピーします。このファイルは C:Program Files (x86)¥IBM¥Windows S-TAP¥Bin¥guard_tap.ini で見つけてください。
3. S-TAP の設定プログラムが格納されているディレクトリーに移動します。
4. setup -UNINSTALL オプションを指定して設定プログラムを実行します。
   重要: 一部のファイルは、次回にスケジュールされているリブートが実行されるまで、完全には削除されません。

Windows: S-TAP のインストール後またはアップグレード後にデータベースを再始動またはリブートするタイミング

Windows S-TAP のインストールやアップグレードでは、リリース・ノートや本書で例外として他の記述がない限り、データベース・サーバーのリブートは必要ありません。

ご使用の特定のバージョンのリブート要件が不明な場合は、技術サポート担当に確認してください。再始動およびリブートの要件は、GIM による実装と GIM を使用しない実装のどちらの場合も同じです。

ドライバーをアップグレードする必要があるときは、データベース・サーバーのみをリブートしてください。

親トピック: Windows: S-TAP のインストール、アップグレード、アンインストール

Windows: データベースをアップグレードする際の S-TAP の管理

以下の指針を使用して、データベースのアップグレード時に Windows S-TAP を管理します。

手順

1. データベースをアップグレードします。
2. 出口を使用している場合: 出口ライブラリーが適切な場所にあることを確認します (新しい DB ロケーション・ディレクトリーがある場合など)。
3. データベースの検査エンジンが正しいかどうか検査します (バージョン番号など)。
4. IE を変更した場合は S-TAP を再始動します。

親トピック: Windows: S-TAP のインストール、アップグレード、アンインストール

Windows: データベースのオペレーティング・システムをアップグレードする際の S-TAP の管理

以下のガイドラインを使用して、データベースのオペレーティング・システム (OS) のアップグレード時に、対話式インストーラーまたは CLI を使用してインストールされた S-TAP を管理します。

このタスクについて

このタスクは、対話式インストーラーまたは CLI を使用してインストールされた S-TAP エージェントのみに関連しています。GIM を使用してインストールされた S-TAP エージェントについては、データベース・サーバーのオペレーティング・システムをアップグレードするときを参照してください。

手順

1. S-TAP エージェントをアンインストールします。
2. データベースのオペレーティング・システムをデータベースをアップグレードします。
3. アップグレードされたデータベース・オペレーティング・システム用の S-TAP インストーラーを Fix Central からダウンロードして、インストールします。

親トピック: Windows: S-TAP のインストール、アップグレード、アンインストール
関連概念:
Windows: S-TAP エージェントのインストール
関連タスク:
Windows: S-TAP のアップグレードと削除

Windows: S-TAP の構成

S-TAP の構成について説明します。

- Windows: GUI からの S-TAP の構成
  この Guardium システムで管理されるすべての S-TAP を表示したり、個々の STAP を管理したり、すべての STAP に対するいくつかの操作を実行したりします。
- Windows: データベース・インスタンスのディスカバー
  Guardium S-TAP ディスカバリー・アプリケーションは、データベース・インスタンスを定期的にディスカバーし、その詳細を 1 次 (現在アクティブな) S-TAP システムに送信します。
- Windows: 検査エンジンの構成
  「S-TAP 制御」ペインで検査エンジンを構成または変更します。
- Windows: 検査エンジンの検査
  S-TAP 検査では、ご使用の環境の STAP とのその検査エンジンが実行されていて、データベース・アクティビティーをアクティブにモニターしていることを確認します。検査について理解し、S-TAP を定期的に検査するためのスケジュールを定義します。
- Windows: S-TAP のロード・バランシング・モデルと構成ガイドライン
  S-TAP のロード・バランシング・モデルについて理解し、セットアップに適したモデルを選択してください。
- Windows: SSL 証明書を使用する S-TAP 認証のセットアップ
  S-TAP サーバーと Guardium システムの間の認証をセットアップします。
- Windows: Db2 出口ライブラリーの使用
  Db2 出口メカニズムを使用すると、Guardium は、トラフィックが暗号化されているかどうか、およびローカルかリモートかに関係なく、すべての Db2 トラフィックを取得できます。このソリューションは、S-TAP 構成を単純化し、ネイティブの Db2 サポートを提供します。
- Windows: S-TAP 構成パラメーターの編集
  S-TAP 構成は、インストール後に GIM または UI を使用して変更できます。上級者の場合は、データベース上の構成ファイルで変更できます。

親トピック: Windows: S-TAP ユーザーズ・ガイド

Windows: GUI からの S-TAP の構成

この Guardium システムで管理されるすべての S-TAP を表示したり、個々の STAP を管理したり、すべての STAP に対するいくつかの操作を実行したりします。

このタスクについて

前提条件: S-TAP のアクティブ・ホストである Guardium システムにログインする必要があります。

ユーザーが、S-TAP のインストール・プロセス中に決定できないことがあったり、誤った決断をして、それがインストール・プロセスの完了後に検出されたりする場合があります。例えば、SQL Guard IP を定義する際に、IP アドレスを入力し忘れたり間違った IP アドレスを使用したりすることがあります。 このような誤りは、S-TAP 構成

を変更することにより修正できます。

GUI のパラメーターは安全に変更できます。GUI に含まれていないパラメーターは変更の必要がほとんどないため、通常は未変更のままにしてください。これらは Guardium 技術サポートまたは上級者によって使用されます。

構成の変更内容によっては、パラメーターの記述で示されるように、手動で S-TAP エージェントを再始動する必要があります。

S-TAP を Guardium Installation Manager (GIM) を使用してインストールした場合、GIM GUI または API を使用して一部のパラメーターを更新することができます。

S-TAP 状況は次のいずれかになります。

- 緑: オンライン
- 黄: ローカル (guard_tap.ini) とリモート (MySql Tap プロパティーに保管されている) の構成パラメーターの不一致を示します。通常、接続が失われたか、コレクターから S-TAP に新規構成パラメーターを送信できないことが原因です。
- 赤: オフライン

手順

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」をクリックして、「S-TAP 制御」を開きます。
2. このページで、すべての S-TAP に対する操作を実行します。
   - リフレッシュ: S-TAP の表示をリフレッシュします。
   - スケジュールにすべて追加: 表示されているすべての S-TAP を S-TAP 検査のスケジュールに追加します。Windows: 検査エンジンの検査を参照してください。
   - スケジュールからすべて削除: 表示されているすべての S-TAP を S-TAP 検査のスケジュールから削除します。
   - コメント: コメントを追加します。『コメント』を参照してください。
3. S-TAP の IP アドレス、または S-TAP がインストールされているデータベース・サーバーのシンボリック・ホスト名で、構成する S-TAP を特定します。個々の S-TAP に対する操作を表示して実行します。

オプション	説明
削除:	S-TAP を除去するには、「削除」をクリックします。 S-TAP の削除は、ある S-TAP が非アクティブになったことが分かった場合、またはその S-TAP の構成ファイルで Guardium 装置がホストとしてリストされなくなった場合に、表示をクリーンアップする際に役立ちます。どちらの場合でも、S-TAP を削除しない限り、S-TAP はオフライン状況で無期限に表示されます。 アクティブな S-TAP はリストから削除できません。削除をクリックしても、S-TAP は情報の送信を停止せず、その S-TAP の構成ファイルに保管されているホストのリストからこの Guardium ホストが削除されることもありません。
リフレッシュ:	「リフレッシュ」をクリックすると、S-TAP 構成の最新のコピーがエージェントから取り出されます。(S-TAP の表示は自動でリフレッシュされません。)
送信コマンド:	「S-TAP コマンド」ポップアップが開きます。ここから各種コマンドを S-TAP ホストに対して実行できます。 ○ 再始動: S-TAP を再始動します。通常は必要ありません。再始動が必要な場合は、データベース・サーバーから停止できます。 ○ S-TAP ロギング ○ バッファーの再初期化: K-TAP 統計をリセットし、S-TAP バッファーを削除します。 ○ 診断の実行: S-TAP の診断スクリプトを実行します (その後、結果を Guardium システムにアップロードします)。 ○ リプレイ・ログの記録: すべてのデータを DB サーバー上のファイルに記録し (RECORD)、データをコレクターに送信します (REPLAY)。 ○ 無視の取り消し: 取り消し可能な無視ポリシーによって無視されたすべてのセッションの無視が取り消され、それらのセッションのトラフィックのキャプチャーが再度開始されます。 ○ データベース・インスタンス・ディスカバリーの実行: ディスカバリー・プロセスを直ちに 1 回実行します。(自動実行が有効になっている場合、デフォルトでは 24 時間ごとに実行されます。)
S-TAP 構成の編集:	「S-TAP 構成」ウィンドウが開きます。GUI に表示されないパラメーターは拡張パラメーターです。上級者でない場合、または Guardium 技術サポートから変更するよう指示されていない場合は、それらのパラメーターを変更しないでください。以下の GUI パラメーターを参照してください。 ○ Windows: 一般パラメーター ○ Windows: 構成監査システム (CAS) パラメーター ○ Windows: Guardium ホスト (SQLGuard) パラメーター ○ Windows: ファイアウォール・パラメーター ○ Windows: 検査エンジン・パラメーター
S-TAP イベント・ログを表示:	クリックすると S-TAP イベント・ログが開きます。ここで、接続、切断、GIM サーバー構成のイベントを確認できます。このログは、トラブルシューティングに非常に役立ちます。
「スケジュールに追加」チェック・ボックス	個々の S-TAP をスケジュールが設定された検査に追加します。
「無視されたセッションをすべて取り消す」チェック・ボックス	データベースは多数のセッションを実行している可能性があり、そのうちのいくつかは現在無視されています。そのサーバーからのトラフィックの無視を停止するには、このオプションをクリアします。

親トピック: Windows: S-TAP の構成

Windows: データベース・インスタンスのディスカバー

Guardium S-TAP ディスカバリー・アプリケーションは、データベース・インスタンスを定期的にディスカバーし、その詳細を 1 次 (現在アクティブな) S-TAP システムに送信します。

Guardium のディスカバリー・エージェントは、データベース・サーバーに S-TAP パッケージとともに自動的にインストールされるソフトウェア・エージェントです。インスタンス・ディスカバリー・エージェントは、データベース・インスタンス、リスナー、およびポートの情報を Guardium システムに報告します。ディスカバリーは、サーバー上の DB インスタンスのすべての詳細を検出して報告するわけではありません。

オートディスカバリーはデフォルトで有効になっています。guard_tap.ini パラメーター winstap_discovery_interval を使用して、実行間隔を構成します。

S-TAP ディスカバリーでサポートされるデータベース・タイプ
MS SQL Server、Db2、Oracle、Informix、MongoDB、CouchDB。

新たにディスカバーされたデータベース・インスタンスは「ディスカバーされたインスタンス」レポートで確認できます。このレポートから、「アクション」メニューを使用して、データ・ソースや検査エンジンを Guardium に素早く追加できます。

データベース・サーバー上のデータベースが作動可能 (開始済み) ではない場合や、後から追加された場合でも、ディスカバリー・エージェントは、STAP 制御ウィンドウからディスカバリー・エージェントの実行コマンドを実行 (「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」。 をクリックし、「データベース・インスタンス・ディスカバリーの実行」を選択) することで、これらのインスタンスをディスカバーできます。

S-TAP のディスカバリーは手動で実行できますが、このアクションは推奨されません。 手動で実行する主な理由は、デバッグのためです。スケジュールされたディスカバリーの実行中に、ユーザー・インターフェースから新しい要求が届いた場合、その新しい要求は無視されます。

注: S-TAP ディスカバリーで Informix データベースがオープンされないという状況が発生しないようにするには、実行可能ファイルの絶対パスを使用して Informix データベースを開始することをお勧めします。

S-TAP ディスカバリー・アプリケーションのパラメーターは、上級ユーザーの場合を除き、デフォルト値のままにしておく必要があります。ディスカバリー・アプリケーションについては、Linux システムおよび UNIX システム: discovery パラメーターで説明しています。

ディスカバリーは、以下のパラメーターも使用します。

- Software_tap_host: S-TAP がインストールされているデータベース・サーバーの IP アドレスまたはホスト名
- sqlguard_ip: S-TAP のディスカバリーの結果が、この IP に送信されます。 (SQLguard パラメーターに primary=1 が指定されている Guardium システム。)

親トピック: Windows: S-TAP の構成

Windows: 検査エンジンの構成

「S-TAP 制御」ペインで検査エンジンを構成または変更します。

始める前に

S-TAP を管理する Guardium システムにログインする必要があります。

このタスクについて

S-TAP をホストしている Guardium システム、または同じ Guardium システムにレポートしている別の S-TAP によって直接モニターされているネットワーク・トラフィックを、S-TAP 検査エンジンでもモニターするように構成しないでください。そのように構成すると、Guardium システムが重複する情報を受け取り、セッションを再構成できずに、そのトラフィックを無視する可能性があります。

手順

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」に移動します。
2. S-TAP の行で、 をクリックします。 「S-TAP 構成」ウィンドウが開きます。
3. 検査エンジンの下部までスクロールして、「検査エンジンの追加...」の横にある をクリックします。
4. プロトコルを選択して、ポート範囲を入力します。ウィンドウが関連パラメーターとそのデフォルト値 (一部) で最新表示されます。
5. すべての必須パラメーターを構成して、「追加」をクリックします。パラメーターが欠落している場合は、システムにより、欠落しているパラメーターが通知されます。

親トピック: Windows: S-TAP の構成
関連資料:
Windows: 検査エンジン・パラメーター

Windows: 検査エンジンの検査

S-TAP 検査では、ご使用の環境の STAP とのその検査エンジンが実行されていて、データベース・アクティビティーをアクティブにモニターしていることを確認します。検査について理解し、S-TAP を定期的に検査するためのスケジュールを定義します。

検査では、Guardium システムと検査エンジンとの間のスニファー操作と通信を検査します。検査は、システム上のすべての S-TAP クライアント、個々の S-TAP クライアント、または個々の検査エンジンについて有効にできます。

検査は、以下のデータベース・タイプでサポートされています。

- Db2
- Db2 Exit (Db2 バージョン 10)
- FTP
- Kerberos
- Mysql
- Oracle
- PostgreSQL
- Sybase

- IE を除外
- MSSQL

検査には、以下の 2 つのタイプがあります。

標準検査
: S-TAP と検査エンジンとの間のスニファー操作と通信を検査します。 検査プロセスは、間違ったユーザー ID とパスワードを使用してデータベースの STAP クライアントへのログインを試行することで、この試行が認識され、Guardium システムに通知されることを確認します。 次に、検査プロセスが、データベース・サーバー上の選択された検査エンジンに接続できるかどうか検査します。 失敗ログインを示す応答を受信することが想定されています。 異なる応答が受信される場合は、さらに調査を行わなければならない可能性があります。
個々のデータベースからの一部エラー・メッセージは、特定の 1 つの問題を示しているわけではありません。 例えば、いくつかのサポートされるデータベース上で、ポートが間違っているためにエラー・コードが返される場合、データベース自体が開始していないことも意味する可能性があります。

詳細検査
: 失敗ログイン要求を避けて、個々の IE を管理するには、詳細検査を使用します。失敗ログイン要求を避けるためには、ターゲット・データベースに関連付けられているデータ・ソース定義を特定または作成する必要があります。 データ・ソース定義には、検査プロセスがデータベースにログインするために使用する資格情報が含まれています。 次に、エラー・メッセージを生成するために、存在しない表からデータを取得するための要求が送信されます。

両方のタイプの検査要求について、実行されたテストと、失敗したテストの推奨アクションについての情報を提供する新しいダイアログに結果が表示されます。

- Windows: S-TAP 検査
  S-TAP 検査プロセスはいくつかの構成パラメーターを確認し、検査エンジンに接続しようとします。
- Windows: 標準検査の構成
  このタスクを使用して、特定の S-TAP クライアント・ホスト上のすべての検査エンジンを構成します。
- Windows: 詳細検査の構成
  このタスクを使用して、特定の S-TAP クライアント・ホスト上のすべての検査エンジンを構成します。
- Windows: S-TAP 検査スケジュールの構成
  S-TAP 検査を実行するためのスケジュールを構成できます。

親トピック: Windows: S-TAP の構成

Windows: S-TAP 検査

S-TAP 検査プロセスはいくつかの構成パラメーターを確認し、検査エンジンに接続しようとします。

データベースに接続する前に、検査プロセスは、Guardium システム上でスニファー・プロセスが実行されているかどうかを検査します。スニファーは、各 S-TAP との通信と、受信されるデータの処理を担当します。スニファーが実行されていない場合、S-TAP からの応答は認識されません。

検査プロセスは、間違ったユーザー ID とパスワードを使用してデータベースの STAP クライアントへのログインを試行することで、この試行が認識され、Guardium システムに通知されることを確認します。

次に、検査プロセスが、データベース・サーバー上の選択された検査エンジンに接続できるかどうか検査します。 失敗ログインを示す応答を受信することが想定されています。 異なる応答が受信される場合は、さらに調査を行わなければならない可能性があります。

個々のデータベースからの一部エラー・メッセージは、特定の 1 つの問題を示しているわけではありません。 例えば、いくつかのサポートされるデータベース上で、ポートが間違っているためにエラー・コードが返される場合、データベース自体が開始していないことも意味する可能性があります。

「S-TAP 検査」ページで検査結果を確認します (「管理」 > 「レポート」 > 「アクティビティー・モニター」 > 「S-TAP 検査」ページ)。失敗した検査が最初に示され、次のステップのための推奨事項が示されます。 リスト末尾の縮小表示されたセクションに、成功した検査が表示されていることを確認します。 状況によっては、成功した検査を調べることが、考えられる複数の次のステップから選択するのに役立つ場合があります。

親トピック: Windows: 検査エンジンの検査

Windows: 標準検査の構成

このタスクを使用して、特定の S-TAP クライアント・ホスト上のすべての検査エンジンを構成します。

このタスクについて

この手順の代わりに、GRDAPI コマンド verify_stap_inspection_engine_with_sequence を使用できます。

手順

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」にアクセスします。
2. 次のオプションを使用します。
   - スケジュールにすべて追加: 表示されているすべての S-TAP に対するすべての検査エンジンを検査に追加します。
   - スケジュールからすべて削除: 表示されているすべての S-TAP に対するすべての検査エンジンを検査から削除します。
   - スケジュールに追加: 選択した S-TAP クライアントのすべての検査エンジンをスケジュールに追加します。

   S-TAP でオプション「すべてが制御可能」が有効になっていない場合は、この S-TAP に対して Guardium システムが 1 次システムである場合、状況の変更のみを実行できます。
3. 「リフレッシュ」をクリックします。
4. 今すぐ検査を実行するには、「管理」 > 「アクティビティー・モニター」 > 「S-TAP 検査スケジューラー」に移動し、「今すぐ 1 回実行」をクリックします。
5. 検査結果を表示する前に、システムはデフォルトで 5 秒待機します。 ネットワーク待ち時間が長い場合、これは、データベース・サーバーからの予期される応答を受信するには、十分の時間ではない可能性があります。 より長い時間が必要な場合は、store stap network_latency CLI コマンドを使用して、期間を変更できます。

次のタスク

「S-TAP 検査」ページで検査結果を確認します (「管理」 > 「レポート」 > 「アクティビティー・モニター」 > 「S-TAP 検査」ページ)。失敗した検査が最初に示され、次のステップのための推奨事項が示されます。 リスト末尾の縮小表示されたセクションに、成功した検査が表示されていることを確認します。 状況によっては、成功した検査を調べることが、考えられる複数の次のステップから選択するのに役立つ場合があります。

親トピック: Windows: 検査エンジンの検査

Windows: 詳細検査の構成

このタスクを使用して、特定の S-TAP クライアント・ホスト上のすべての検査エンジンを構成します。

このタスクについて

手順

1. 「管理」 > 「システム・ビュー」 > 「S-TAP 状況モニター」にアクセスします。
2. S-TAP の行の任意の場所をクリックします。
   このホストの個々の検査エンジンによってウィンドウがリフレッシュされます。
3. 詳細検査を構成します。
   a. 1 つの検査エンジンをクリックし、「詳細検査」をクリックします。
   b. オプションで、「データ・ソース」で、「一致する S-TAP ホストのみを表示」を選択するか、「名前ドロップダウン・リストから名前を選択して、特定の検査エンジンを検索します。
   c. 「閉じる」をクリックします。
4. 今すぐ検査を実行するには、検査エンジンを 1 つ以上選択し、「検査」をクリックします。「S-TAP 検査の結果」ウィンドウが開きます。
5. 検査結果を表示する前に、システムはデフォルトで 5 秒待機します。 ネットワーク待ち時間が長い場合、これは、データベース・サーバーからの予期される応答を受信するには、十分の時間ではない可能性があります。 より長い時間が必要な場合は、store stap network_latency CLI コマンドを使用して、期間を変更できます。
6. 検査の追加または削除を実行するには、次のようにします。
   a. 1 つ以上の検査エンジンを選択します。
   b. 「スケジュールに追加」または「スケジュールから削除」をクリックします。

次のタスク

「S-TAP 検査」ページで検査結果を確認します (「管理」 > 「レポート」 > 「アクティビティー・モニター」 > 「S-TAP 検査」ページ)。失敗した検査が最初に示され、次のステップのための推奨事項が示されます。 リスト末尾の縮小表示されたセクションに、成功した検査が表示されていることを確認します。 状況によっては、成功した検査を調べることが、考えられる複数の次のステップから選択するのに役立つ場合があります。

親トピック: Windows: 検査エンジンの検査

Windows: S-TAP 検査スケジュールの構成

S-TAP 検査を実行するためのスケジュールを構成できます。

このタスクについて

検査がスケジュールされているすべての S-TAPs に、同じスケジュールが使用されます。

スケジュールが定義されると、「S-TAP 検査スケジューラー」の「一時停止」ボタンをクリックして、検証プロセスをアクティブのまま一時的に停止できます。 リアルタイムで 1 回検証を実行するには、「今すぐ 1 回実行」ボタンを使用します。

手順

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 検査スケジューラー」をクリックして、「S-TAP 検査スケジューラー」を開きます。
2. このページの「S-TAP 検査スケジューラー」の部分で、「スケジュールの変更」をクリックします。
3. 「スケジュール定義」ダイアログで、ドロップダウン・リストとチェック・ボックスを使用して、検査実行のスケジュールを設定します。 このスケジュールは、検査がスケジュールされているすべての S-TAPs に適用されます。
4. 「保存」をクリックして、変更を保存します。

親トピック: Windows: 検査エンジンの検査

Windows: S-TAP のロード・バランシング・モデルと構成ガイドライン

S-TAP のロード・バランシング・モデルについて理解し、セットアップに適したモデルを選択してください。

各ロード・バランシング・モデルとその具体的なパラメーター要件について、以下に説明します。

注: このトピックでは、エンタープライズ・ロード・バランシングではなく、S-TAP ロード・バランシングについて説明しています。

フェイルオーバー

S-TAP は、1 つのコレクター (1 次) にトラフィックを送信し、必要に応じて 2 次コレクターにフェイルオーバーします。S-TAP エージェントは、1 つの 1 次コレクター IP と 1 つ以上の 2 次コレクター IP で構成されます。さまざまな理由で S-TAP エージェントが 1 次コレクターにトラフィックを送信できない場合、S-TAP エージェントは自動的に 2 次コレクターにフェイルオーバーします。2 次ホスト・システムが使用できなくなるか、1 次ホストが再び使用可能になるか、S-TAP が再始動 (S-TAP はその 1 次ホストへの接続を最初に試行します) されるまで、S-TAP は 2 次ホストにデータを送信し続けます。2 次ホスト・システムが使用不可になると、別の 2 次ホストにフェイルオーバーします (定義されている場合)。2 つ目のケースでは、S-TAP は 2 次 Guardium ホストから 1 次 Guardium ホストに再びフェイルオーバーします。1 次コレクターを 1 つと、2 次コレクターを最大 2 つセットアップすることをお勧めします。1 つのコレクターのみをスタンバイ・フェイルオーバー・コレクターとして定義することも、複数のフェイルオーバー・コレクターを定義することも可能です。1 つのスタンバイ・フェイルオーバーを使用する場合、通常は 4 つから 5 つのコレクターに対して 1 つのコレクターで十分です。複数のフェイルオーバー・コレクターを使用する場合、各コレクターは最大 50% の容量で稼働し、追加の負荷用にリソースが常に存在す

るようにする必要があります。ご使用のアーキテクチャー、データベース、およびデータ・センターのレイアウトに最適なセットアップを選択してください。1 次ホストが使用可能になると、S-TAP は 2 次 Guardium ホストから 1 次 Guardium ホストに再びフェイルバックします。

S-TAP はアクティブ・ホストから構成変更が適用されるたびに再始動されます。

「S-TAP制御」ウィンドウの「詳細」セクションで、「ロード・バランシング」 を 0 に設定します。次に「Guardium ホスト」セクションで、少なくとも 1 つの 2 次 Guardium ホストを追加します。

上級者以外は、追加のフェイルオーバー構成をデフォルト値のままにしておく必要があります。

Guardium システムを S-TAP の 2 次ホストとして指定する前に、以下の項目を確認してください。

- Guardium システムが、S-TAP がインストールされているデータベース・サーバーに接続可能であること。 複数の Guardium システムが使用されている場合、それらはしばしば、ネットワーク上で切り離された分岐に接続されています。
- その Guardiumシステムが、S-TAP がインストールされているデータベース・サーバーからのセッション・データを無視するようなセキュリティー・ポリシーを持たないこと。多くの場合、Guardium® セキュリティー・ポリシーは、監視可能なデータベース・トラフィックの狭いサブセットに重点を置き、その他すべてのセッションは無視するように構築されます。2 次ホストが S-TAP からのセッション・データを無視しないことを確認するか、Guardiumシステムのセキュリティー・ポリシーを必要に応じて変更します。

ロード・バランシング

この構成により、1 つのデータベースから複数のコレクターへのトラフィックのバランスを取ることができます。このオプションは、アクティブ・データベースのすべてのトラフィックをモニターする場合 (広範なモニター) に適しています。(異常値を検出するためには、アグリゲーターですべての関連データが処理されるように、同じアグリゲーターおよび中央マネージャーの下にコレクターが配置されている必要があります。) 生成されたトラフィックが大きく、データを長期間にわたってコレクター上にオンラインで格納する必要がある場合、この方法は最良の選択肢と言えます。なぜなら、複数のコレクター間でセッション・ベースのロード・バランシングが実行されるからです。S-TAP は、最大 10 個のコレクターを使用してこのように構成できます。

ロード・バランシングを行うには、「S-TAP制御」ウィンドウの「詳細」セクションで、「ロード・バランシング」 を 1 に設定します。

グリッド

グリッドでは、S-TAP は f5 や Cisco などのロード・バランサーを介してコレクターと通信します。S-TAP エージェントは、ロード・バランサーにトラフィックを送信するように構成されます。ロード・バランサーは、コレクターのプール内のいずれかのコレクターに S-TAP トラフィックを転送します。ロード・バランサーに障害が発生した場合に継続してモニターできるように、ロード・バランサー間のフェイルオーバーを構成することも可能です。

S-TAP の持続性は、以下のフェイルオーバー・パラメーターによって構成されます。

- TAP_MIN_TIME_BEFOREFAILOVER: S-TAP が 1 次 Guardium システムに接続できない場合、または 1 次 Guardium システムに接続はできるが、そのバッファーに書き込むことができない場合に、S-TAP が 2 次 Guardium システムに切り替えるまでの時間間隔 (分単位)。デフォルトは 5です。
- TAP_MIN_HEARTBEAT_INTERVAL: S-TAP が 2 次 Guardium バッファーへの書き込みを試行する前に、1 次 Guardium システム・バッファーへの書き込みを試行する最大時間。デフォルトは 30 秒です。つまり、フェイルオーバーの前に少なくとも 5 * 60/30 回、書き込みを試行します。

F5 環境の S-TAP は、ログ・ファイルおよび診断を実行した結果 (メモリー・ダンプを除く ..¥Logs フォルダー内のすべてのファイル) をロケーション ./var/IBM/Guardium/log/stap_diagnostic/ でアクティブなコレクターおよび中央マネージャー (存在する場合) にアップロードします。

グリッド・モデルを使用するには、「S-TAP制御」ウィンドウの「詳細」セクションで、「ロード・バランシング」 を 3 に設定します。

さらに、以下のように設定します。

- 「すべてが制御可能」= 1
- 「Guardium ホスト」 =<すべての S-TAP データベース・クライアントが指し示すバランサーの仮想 IP の IP>

冗長

冗長では、S-TAP はそのペイロード全体を複数のコレクターに送信します。S-TAP は複数のコレクター (通常は 2 つのみ) で構成され、同じ内容を両方のコレクターに送信します。このオプションにより、ログに記録された同一データの、複数のコレクター間における完全な冗長性が実現します。また、異なる細分度レベルのアクティビティーに関するデータおよびアラートをログに記録するためにこのオプションを使用することもできます。

冗長設定を使用するには、「S-TAP制御」ウィンドウの「詳細」セクションで、「ロード・バランシング」 を 2 に設定します。

親トピック: Windows: S-TAP の構成

Windows: SSL 証明書を使用する S-TAP 認証のセットアップ

S-TAP サーバーと Guardium システムの間の認証をセットアップします。

S-TAP は、指定の証明書または証明書セットを使用して認証される特定のマシンのグループのみに接続するように構成できます。これらの証明書は、Guardium システム上でローカルに生成して認証局 (CA) に署名のために送信するか、または CA 側で作成して、Guardium システム全体にインストールできます。

- Windows: Guardium システムでの証明書署名要求 (CSR) の生成
  次の手順を実行して、Guardium システム上でローカルに証明書署名要求を生成し、署名のために認証局 (CA) に送信します。
- Windows: Guardium システム外部で生成された SSL 証明書のインストール
  次の手順を実行して、CA によって作成された SSL 証明書をインストールします。
- Windows: x.509 証明書認証を使用するための S-TAP の構成

親トピック: Windows: S-TAP の構成

Windows: Guardium システムでの証明書署名要求 (CSR) の生成

次の手順を実行して、Guardium システム上でローカルに証明書署名要求を生成し、署名のために認証局 (CA) に送信します。

手順

1. CLI を使用して Guardium システムにログインします。
2. cli> create csr sniffer と入力します。
3. 要求されたデータを入力します。

```
temp4> create system csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [GB]:CA
State or Province Name (full name) [Berkshire]:BC
Locality Name (eg, city) [Newbury]:
Organization Name (eg, company) [My Company Ltd]:QA_Sample1
Organizational Unit Name (eg, section) []:Sample_QA
Common Name (eg, your name or your server's hostname) []:sample1_qa.victoria
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
```

終了すると、次のようになります。

```
Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject: C=CA, ST=BC, L=Newbury, O=QA_Sample1, OU=Sample_QA, CN=sample1_
qa.victoria
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (2048 bit)
                Modulus (2048 bit):
                    00:e9:5e:a2:81:53:dc:e9:b5:f7:54:33:17:6f:15:
                    9c:00:5c:ff:b2:64:c6:e5:48:be:36:a7:a4:55:f4:
                    b1:df:c9:81:a4:41:fe:29:80:d6:fd:d4:c4:b5:97:
                    b7:c1:3d:42:6c:c0:f8:09:cd:ea:36:f6:3b:b9:d9:
                    ce:15:87:2d:2f:b3:f2:5f:f9:42:06:e2:a0:62:56:
                    06:6d:cc:65:69:62:db:36:34:09:95:5b:c3:d0:e6:
                    85:ee:64:76:3e:ed:d6:47:bb:49:f2:08:81:14:c2:
                    e3:93:db:20:ba:86:e7:60:24:80:01:7f:3d:b7:60:
                    16:ba:06:d4:a1:e0:18:39:73:ca:1e:24:15:56:6d:
                    97:79:81:04:f7:fd:37:06:42:d7:15:82:34:aa:51:
                    2c:cc:e2:f0:d1:42:dd:b5:71:bb:10:19:a0:0a:5c:
                    4f:77:b9:bb:36:95:ed:a4:77:07:e3:50:f9:36:20:
                    13:e2:e1:78:d2:0a:36:8a:b9:39:90:1f:a4:82:12:
                    4f:50:29:3f:19:7d:16:a6:b3:23:7b:b8:7b:85:60:
                    04:21:39:84:1d:9e:81:e5:20:2c:8a:51:f3:52:f7:
                    3c:4f:e6:f2:a5:88:dc:2e:99:0a:b3:65:1e:bf:33:
                    5f:be:dc:53:1c:a6:69:18:c4:c7:75:bf:20:e3:cf:
                    29:af
                Exponent: 65537 (0x10001)
        Attributes:
            challengePassword        :guardium
    Signature Algorithm: sha1WithRSAEncryption
        06:4a:b9:db:04:a1:8d:4c:f7:3f:8f:24:fa:7c:ec:a6:70:77:
        8b:b9:38:7c:b6:e6:51:aa:ed:96:20:16:37:85:a7:44:26:2b:
        87:4c:a4:db:0c:f3:d3:87:e3:68:4a:8e:de:f6:0a:09:58:8f:
        68:98:4f:f3:8a:e2:37:5c:d6:42:32:8f:d9:01:56:41:88:df:
        1a:ba:63:03:62:08:89:06:13:88:74:6f:cd:eb:26:f0:67:a4:
        26:9b:a3:4c:ff:7b:c9:19:2c:12:58:06:ce:22:3c:e6:cd:52:
        b0:d0:da:6a:c9:02:df:02:e6:25:77:39:cf:50:80:e7:1d:01:
        fc:40:17:a2:98:04:bf:8b:24:f6:55:46:99:7b:17:05:01:d3:
        09:3d:a2:f0:e0:ba:5d:15:b8:28:74:d2:a3:fe:fd:86:7d:e0:
        60:e8:e4:38:6a:17:9c:80:80:e3:50:11:5e:35:f5:02:2b:65:
        60:41:2a:dc:ed:a8:a9:9a:6f:24:b4:7a:9c:39:01:a4:fc:cf:
        e6:94:86:f1:18:3a:f5:99:6b:f8:66:a2:ff:04:08:7e:ca:6b:
        2a:aa:cf:72:26:d0:c9:96:a0:98:fd:91:bb:b1:e4:8d:6d:10:
        08:ea:56:de:07:20:d3:e6:9a:bf:de:cf:c3:a4:e8:43:60:4f:
        b4:53:aa:d5
-----BEGIN CERTIFICATE REQUEST-----
MIIC0TCCAbkCAQAwczELMAkGA1UEBhMCQ0ExCzAJBgNVBAgTAkJDMRAwDgYDVQQH
EwdOZXdidXJ5MRMwEQYDVQQKDApRQV9TYW1wbGUxMRIwEAYDVQQLDAlTYW1wbGVf
UUExHDAaBgNVBAMME3NhbXBsZTFfcWEudmljdG9yaWEwggEiMA0GCSqGSIb3DQEB
AQUAA4IBDwAwggEKAoIBAQDpXqKBU9zptfdUMxdvFZwAXP+yZMblSL42p6RV9LHf
yYGkQf4pgNb91MS1l7fBPUJswPgJzeo29ju52c4Vhy0vs/Jf+UIG4qBiVgZtzGVp
Yts2NAmVWBPQ5oXuZHY+7dZHu0nyCIEUwuOT2yC6hudgJIABfz23YBa6BtSh4Bg5
c8oeJBVWbZd5gQT3/TcGQtcVgjSqUSzM4vDRQt21cbsQGaAKXE93ubs2le2kdwfj
UPk2IBPi4XjSCjaKuTmQH6SCEk9QKT8ZfRamsyN7uHuFYAQhOYQdnoH1ICyKUfNS
9zxP5vKliNwumQqzZR6/M1++3FMcpmkYxMd1vyDjzymvAgMBAAGgGTAXBgkqhkiG
9w0BCQcxChMIZ3VhcmRpdW0wDQYJKoZIhvcNAQEFBQADggEBAAZKudsEoY1M9z+P
JPp87KZwd4u50Hy25lGq7ZYgFjeFp0QmK4dMpNsM890H42hKjt72CglYj2iYT/OK
4jdc1kIyj9kBVkGI3xq6YwNiCIkGE4h0b83rJvBnpCabo0z/e8kZLBJYBs4iPObN
UrDQ2mrJAt8C5iV3Oc9Qg0cdAfxAF6KYBL+LJPZVRpl7FwUB0wk9ovDgul0VuCh0
0qP+/YZ94GDo5DhqF5yAg0NQEV419QIrZWBBKtztqKmabyS0epw5AaT8z+aUhvEY
OvWZa/hmov8ECH7Kayqqz3Im0MmWoJj9kbux5I1tEAjqVt4HINPmmr/ez8Ok6ENg
T7RTqtU=
-----END CERTIFICATE REQUEST-----
ok
temp4>
```

4. -----BEGIN CERTIFICATE REQUEST----- から -----END CERTIFICATE REQUEST----- までをファイルにコピーして、署名のために CA に送信します。

   CA によって証明書が署名され、次のような公開鍵が送られてきます。

```
enance@enance1 Latest $ cat sample1_qa.victoria.pem
-----BEGIN CERTIFICATE-----
MIID1jCCAsCgAwIBAgIBCDALBgkqhkiG9w0BAQUwgYAxCzAJBgNVBAYTAkNBMRkw
FwYDVQQIExBCcml0aXNoIENvbHVtYmlhMREwDwYDVQQHEwhWaWN0b3JpYTEUMBIG
A1UEChMLUUFfdGVzdF92aWMxFDASBgNVBAsTC1ZpY3Rvcmlhx1FBMRcwFQYDVQQD
Ew5WaWN0b3JpYV9RQV9DQTAeFw0xMDExMDEyMTA5MzhaFw0xNTExMDEyMTA5Mzha
MHMxCzAJBgNVBAYTAkNBMQswCQYDVQQIEwJCQzEQMA4GA1UEBxMHTmV3YnVyeTET
MBEGA1UECgwKUUFfU2FtcGxlMTESMBAGA1UECwwJU2FtcGxlX1FBMRwwGgYDVQQD
DBNzYW1wbGUxX3FhLnZpY3RvcmlhMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA6V6igVPc6bX3VDMXbxWcAFz/smTG5Ui+NqekVfSx38mBpEH+KYDW/dTE
tZe3wT1CbMD4Cc3qNvY7udnOFYctL7PyX/lCBuKgYlYGbcxlaWLbNjQJlVvD0OaF
7mR2Pu3WR7tJ8giBFMLjk9sguobnYCSAAX89t2AWugbUoeAY0XPKHiQVVm2XeYEE
9/03BkLXFYI0qlEszOLw0ULdtXG7EBmgClxPd7m7NpXtpHcH41D5NiAT4uF40go2
irk5kB+kghJPUCk/GX0WprMje7h7hWAEITmEHZ6B5SAsilHzUvc8T+bypYjcLpkK
s2UevzNfvtxTHKZpGMTHdb8g488prwIDAQABo2swaTAfBgNVHSMEGDAWgBR0SB68
8syKm4CUQ27LGB9ftHRZyTAMBgNVHRMBAf8EAjAAMA8GA1UdDwEB/wQFAwMHuAAw
JwYDVR0lBCAwHgYIKwYBBQUHAwQGCCsGAQUFBwMCBggrBgEFBQcDATALBgkqhkiG
9w0BAQUDggEBAJe1Dlh623uD9mBjfB3YDKO3agm3vbdMd2vcdKI8TA5dsxMhmHvm
8E+gVsV0rNVbupLoc6OYeJLPvWQ54j9wZnKavBbma067C1QJJ2jEhOhjzoiEDIqT
l/qBhvqabhTG3vIMFSIwOu0zmQD/2iFu9cykKiru8A8djfZWjjfZiHO4dkkiCInP
/dor+Cm5RokGZ+OXhZ/5hxTuGeSAWJIhObVnrnPLZ2c2uYGh6LYip+2GU6L/rp8z
tmLYfldjtTMGYeP4Ivo1s7KHJqqDlATOBwe2XVR98O8SrHI7toSpAbdIqP+f77zv
pb5xv0SfmqLuV6eUvJW8d/Wj2mvgw1qLvqY=
-----END CERTIFICATE-----
```

5. 内容をコピーしたり、Guardium システムにインポートするのに便利な場所にこのファイルを配置してください。cli> store certificate sniffer [console | import] と入力します。
6. console の場合、**-----BEGIN CERTIFICATE-----** から **-----END CERTIFICATE-----** まですべて (コピー内のものを含む) をコピーし、プロンプトが出されたら CLI に貼り付けます。import を選択した場合、ファイルのインポート元を Guardium システムに指示します。

```
temp4 > store system certificate console
Please paste your new system certificate, in PEM format.
Include the BEGIN and END lines, then press CTRL-D.

-----BEGIN CERTIFICATE-----
MIID1jCCAsCgAwIBAgIBCDALBgkqhkiG9w0BAQUwgYAxCzAJBgNVBAYTAkNBMRkw
FwYDVQQIExBCcml0aXNoIENvbHVtYmlhMREwDwYDVQQHEwhWaWN0b3JpYTEUMBIG
A1UEChMLUUFfdGVzdF92aWMxFDASBgNVBAsTC1ZpY3Rvcmlhx1FBMRcwFQYDVQQD
Ew5WaWN0b3JpYV9RQV9DQTAeFw0xMDExMDEyMTA5MzhaFw0xNTExMDEyMTA5Mzha
MHMxCzAJBgNVBAYTAkNBMQswCQYDVQQIEwJCQzEQMA4GA1UEBxMHTmV3YnVyeTET
MBEGA1UECgwKUUFfU2FtcGxlMTESMBAGA1UECwwJU2FtcGxlX1FBMRwwGgYDVQQD
DBNzYW1wbGUxX3FhLnZpY3RvcmlhMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA6V6igVPc6bX3VDMXbxWcAFz/smTG5Ui+NqekVfSx38mBpEH+KYDW/dTE
tZe3wT1CbMD4Cc3qNvY7udnOFYctL7PyX/lCBuKgYlYGbcxlaWLbNjQJlVvD0OaF
7mR2Pu3WR7tJ8giBFMLjk9sguobnYCSAAX89t2AWugbUoeAY0XPKHiQVVm2XeYEE
9/03BkLXFYI0qlEszOLw0ULdtXG7EBmgClxPd7m7NpXtpHcH41D5NiAT4uF40go2
irk5kB+kghJPUCk/GX0WprMje7h7hWAEITmEHZ6B5SAsilHzUvc8T+bypYjcLpkK
s2UevzNfvtxTHKZpGMTHdb8g488prwIDAQABo2swaTAfBgNVHSMEGDAWgBR0SB68
8syKm4CUQ27LGB9ftHRZyTAMBgNVHRMBAf8EAjAAMA8GA1UdDwEB/wQFAwMHuAAw
JwYDVR0lBCAwHgYIKwYBBQUHAwQGCCsGAQUFBwMCBggrBgEFBQcDATALBgkqhkiG
9w0BAQUDggEBAJe1Dlh623uD9mBjfB3YDKO3agm3vbdMd2vcdKI8TA5dsxMhmHvm
8E+gVsVOrNVbupLoc6OYeJLPvWQ54j9wZnKavBbma067C1QJJ2jEhOhjzoiEDIqT
l/qBhvqabhTG3vIMFSIwOu0zmQD/2iFu9cykKiru8A8djfZWjjfZiHO4dkkiCInP
/dor+Cm5RokGZ+OXhZ/5hxTuGeSAWJIhObVnrnPLZ2c2uYGh6LYip+2GU6L/rp8z
tmLYfldjtTMGYeP4Ivo1s7KHJqqDlATOBwe2XVR98O8SrHI7toSpAbdIqP+f77zv
pb5xvOSfmqLuV6eUvJW8d/Wj2mvgw1qLvqY=
-----END CERTIFICATE-----
```

証明書を保管するかどうかを確認するよう求められます。確認すると、証明書が保管されます。

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 8 (0x8)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=CA, ST=British Columbia, L=Victoria, O=QA_test_vic, OU=Victori
a_QA, CN=Victoria_QA_CA
        Validity
            Not Before: Nov  1 21:09:38 2010 GMT
            Not After : Nov  1 21:09:38 2015 GMT
        Subject: C=CA, ST=BC, L=Newbury, O=QA_Sample1, OU=Sample_QA, CN=sample1_
qa.victoria
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (2048 bit)
                Modulus (2048 bit):
                    00:e9:5e:a2:81:53:dc:e9:b5:f7:54:33:17:6f:15:
                    9c:00:5c:ff:b2:64:c6:e5:48:be:36:a7:a4:55:f4:
                    b1:df:c9:81:a4:41:fe:29:80:d6:fd:d4:c4:b5:97:
                    b7:c1:3d:42:6c:c0:f8:09:cd:ea:36:f6:3b:b9:d9:
                    ce:15:87:2d:2f:b3:f2:5f:f9:42:06:e2:a0:62:56:
                    06:6d:cc:65:69:62:db:36:34:09:95:5b:c3:d0:e6:
                    85:ee:64:76:3e:ed:d6:47:bb:49:f2:08:81:14:c2:
                    e3:93:db:20:ba:86:e7:60:24:80:01:7f:3d:b7:60:
                    16:ba:06:d4:a1:e0:18:39:73:ca:1e:24:15:56:6d:
                    97:79:81:04:f7:fd:37:06:42:d7:15:82:34:aa:51:
                    2c:cc:e2:f0:d1:42:dd:b5:71:bb:10:19:a0:0a:5c:
                    4f:77:b9:bb:36:95:ed:a4:77:07:e3:50:f9:36:20:
                    13:e2:e1:78:d2:0a:36:8a:b9:39:90:1f:a4:82:12:
                    4f:50:29:3f:19:7d:16:a6:b3:23:7b:b8:7b:85:60:
                    04:21:39:84:1d:9e:81:e5:20:2c:8a:51:f3:52:f7:
                    3c:4f:e6:f2:a5:88:dc:2e:99:0a:b3:65:1e:bf:33:
                    5f:be:dc:53:1c:a6:69:18:c4:c7:75:bf:20:e3:cf:
                    29:af
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Authority Key Identifier:
                keyid:74:48:1E:BC:F2:CC:8A:9B:80:94:43:6E:CB:18:1F:5F:B4:74:59:C
9

            X509v3 Basic Constraints: critical
                CA:FALSE
            X509v3 Key Usage: critical
                Digital Signature, Key Encipherment, Data Encipherment, Key Agre
ement
            X509v3 Extended Key Usage:
                E-mail Protection, TLS Web Client Authentication, TLS Web Server
 Authentication
    Signature Algorithm: sha1WithRSAEncryption
        97:b5:0e:58:7a:db:7b:83:f6:60:63:7c:1d:d8:0c:a3:b7:6a:
        09:b7:bd:b7:4c:77:6b:dc:74:a2:3c:4c:0e:5d:b3:13:21:98:
        7b:e6:f0:4f:a0:56:c5:4e:ac:d5:5b:ba:92:e8:73:a3:98:78:
        92:cf:bd:64:39:e2:3f:70:66:72:9a:bc:16:e6:6b:4e:bb:0b:
        54:09:27:68:c4:84:e8:63:ce:88:84:0c:8a:93:97:fa:81:86:
        fa:9a:6e:14:c6:de:f2:0c:15:22:30:3a:ed:33:99:00:ff:da:
        21:6e:f5:cc:a4:2a:2a:ee:f0:0f:1d:8d:f6:56:8e:37:d0:88:
        73:b8:76:49:22:08:89:cf:fd:da:2b:f8:29:b9:46:89:06:67:
        e3:97:85:9f:f9:87:14:ee:19:e4:80:58:92:21:39:b5:67:ae:
        73:cb:67:67:36:b9:81:a1:e8:b6:22:a7:ed:86:53:a2:ff:ae:
        9f:33:b6:62:d8:7e:57:63:b5:33:06:61:e3:f8:22:fa:35:b3:
        b2:87:26:aa:83:94:04:ce:07:07:b6:5d:54:7d:f0:ef:12:ac:
        72:3b:b6:84:a9:01:b7:48:a8:ff:9f:ef:bc:ef:a5:be:71:bc:
        e4:9f:9a:a2:ee:57:a7:94:bc:95:bc:77:f5:a3:da:6b:e0:c3:
        5a:8b:be:a6

Do you want to store this certificate? (y/n)
```

7. 新しい証明書を有効にするために、inspection-core を再始動します。

親トピック: Windows: SSL 証明書を使用する S-TAP 認証のセットアップ

Windows: Guardium システム外部で生成された SSL 証明書のインストール

次の手順を実行して、CA によって作成された SSL 証明書をインストールします。

このタスクについて

インストールする証明書全体が CA から送信される場合、PKCS#8 (パスワード保護) フォーマットの秘密鍵と PEM フォーマットの公開鍵の 2 つのファイルが必要です。生成される証明書は、2048 ビットの RSA 鍵である必要があります。

CA から 2 つのファイルと、CA の公開証明書が送信されます。

CA の公開証明書は次のようになります。

```
enance@enance1 Latest $ cat Victoria_QA_CA.pem
-----BEGIN CERTIFICATE-----
MIID2zCCAsWgAwIBAgIBATALBgkqhkiG9w0BAQUwgYAxCzAJBgNVBAYTAkNBMRkw
FwYDVQQIExBCcml0aXNoIENvbHVtYmlhMREwDwYDVQQHEwhWaWN0b3JpYTEUMBIG
A1UEChMLUUFfdGVzdF92aWMxFDASBgNVBAsTC1ZpY3RvcmlhX1FBMRcwFQYDVQQD
Ew5WaWN0b3JpYV9RQV9DQTAeFw0xMDA4MTIxODMzMDJaFw0zMDA4MTIxODMzMDJa
MIGAMQswCQYDVQQGEwJDQTEZMBcGA1UECBMQQnJpdGlzaCBDb2x1bWJpYTERMA8G
A1UEBxMIVmljdG9yaWExFDASBgNVBAoTC1FBX3R1c3Rfdmljb3RQwEgYDVQQLEwtW
aWN0b3JpYV9RQTEXMBUGA1UEAxM0VmljdG9yaWFfUUFfQ0EwggEgMAsGCSqGSIb3
DQEBAQOCAQ8AMIIBCgKCAQEA0x3iAXs1KGNoJThXkO+jcNyMBlfwKWRMTOq9PkF4
piznXCRwPz2nQWk5/fpslchmuVYXJtfZi7umDxp2FEMvMmhJfFZiqbCn1Rb5yH+1
V3RsIerB0DFpoWkdt+wD6BufndO5P9eOlv14bmTl+F0dUM0TxAwtX73CMQOX/n+i
/WrzpWU41U71KkyWUfJl2Pm8TLEMr5awpzt2rEJ/Q1qItHcksQDbGY0MNLaNojEU
XBzUpu9ezbv+zVH+5iorFYkrH0NQI0NK+YoR1b3Tto0HLDh6istsMfHdNEEQb9BB
vMjqUz4tGB2HDguYTaNbQJj9Yw8uv7/tfwW/cesrqm8DiQIDAQABo2QwYjAPBgNV
HRMBAf8EBTADAQH/MA8GA1UdDwEB/wQFAwMHBgAwHQYDVR0OBBYEFHRIHrzyzIqb
gJRDbssYH1+0dFnJMB8GA1UdIwQYMBaAFHRIHrzyzIqbgJRDbssYH1+0dFnJMAsG
CSqGSIb3DQEBBQOCAQEAbrImEbRyBka0wO/ZuPd0Hw9jpbxIuaYEskakv7aM4TUQ
awf1CiqwwAymmKb2REItLajhjmBFbXBun7d137vBU2KX104I7WGw0xgI5rmlELa+
2fizugY+Bc6mh+5c0haizkyudKzo8mLz2p/IS7SPh21J9rnuBleSt9zfiYanPxX1
Q6z1+WRKIRSUmK+ho4bmtgr5F0+ejmZb9nfzeOBj23H9i0mWoaQ/SO+O2iDIvDY0
KYwqCeS2UNEdTcnfuczbBkqnAsf5/GBPIhnWX3onuLkOsHdY0HHPjoqgHauoxPK8
pOsEvlCK8EFOD6wkp0vtNhFQCykxRimHR6Pz9jEVjw==
-----END CERTIFICATE-----
```

ユーザー/この Guardium システムに固有の公開証明書は、次のようになります。

```
enance@enance1 Latest $ cat Sample_givenCert.victoria.qa_pub.pem
-----BEGIN CERTIFICATE-----
MIID5jCCAtCgAwIBAgIBCTALBgkqhkiG9w0BAQUwgYAxCzAJBgNVBAYTAkNBMRkw
FwYDVQQIExBCcml0aXNoIENvbHVtYmlhMREwDwYDVQQHEwhWaWN0b3JpYTEUMBIG
A1UEChMLUUFfdGVzdF92aWMxFDASBgNVBAsTC1ZpY3RvcmlhX1FBMRcwFQYDVQQD
Ew5WaWN0b3JpYV9RQV9DQTAeFw0xMDExMTUyMDUwNThaFw0xNTExMTUyMDUwNTha
MIGEMQswCQYDVQQGEwJDQTEZMBcGA1UECBMQQnJpdGlzaCBDb2xvbWJpYTESMBAG
A1UEBxMJVmFuY291dmVyMQswCQYDVQQKEwJRQTESMBAGA1UECxMJUUFfU0FNUExF
MSUwIwYDVQQDExxTYW1wbGVfZ2l2ZW5DZXJ0LnZpY3RvcmlhLnFhMIIBIDALBgkq
hkiG9w0BAQEDggEPADCCAQoCggEBALdt2XhJEJIAognblgFNoiomVvcGIe9PHs0t
3fP5DxAL5PVm+UCRS0xnnCoKe3pdJNagepDwBa2K5UsHbK4vkHJEgWWEd4bx2Ks7
kRoHr83TXK+W8al1HGKRwUSwN0hfm/KV4v8ZX3AFWs2c/BJ2iA77a0mhU0JuWa1/
/scXitJB5ykPjFb2EuReA6ELphaQ/itjZIQTTeVxbamyx42ia9J5B07lFTp3qGdU
yXUwlEFX0HFPmknAAaQrSnpMdQjkOKgzxU0NTV4ILA66hCVkX+ezQeJA90s/AHA5
hAY2fyurKZsOowoE0xlEpFWCGf87zxfkmTaWthqCS0NzEcJ6efECAwEAAaNrMGkw
HwYDVR0jBBgwFoAUdEgevPLMipuAlENuyxgfX7R0WckwDAYDVR0TAQH/BAIwADAP
BgNVHQ8BAf8EBQMDB7gAMCcGA1UdJQQgMB4GCCsGAQUFBwMEBggrBgEFBQcDAgYI
KwYBBQUHAwEwCwYJKoZIhvcNAQEFA4IBAQB+P1m22B72eRESk9rgNdOB+l48k5xw
QHf64TJcoI/Vcz0vtGBC1jDfZzVDIHJaX6ZMtd023QO6rtjQSnhjhbOt/EiimaNO
tysJOE999E+HQ7UpKYKvdoznwpYEthyoQJ9quKqQc1YW/l8pQMYDEdK8c7yMbJpv
mrxO8h+G3YYQMT6A9KAk/GPA+yKl0fCuogBhPyWM28q35EAIw6H9ahQigWhNkzrL
DCYO8VCLi3BX+OohLvurkyzJm2lnbm7BSb5QB4jISlRBGPoIWhK6VPypboyABfVs
yWHFVs10EUb0Maa/XNZILTAoYAbYK5sX7Ri5hr5KhxnmDd9AJMQsLfik
-----END CERTIFICATE-----
```

秘密鍵 (pkx#8 で暗号化) は次のようになります。

```
enance@enance1 Latest $ cat Sample_givenCert.victoria.qa_pri.pem
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIE4jAcBgoqhkiG9w0BDAEDMA4ECAKuMvSla9T4AgIBYgSCBMB/RzDgm4xpRDnA
Cd//wgD9c8junXmVpyJWXWgXhlj3Sgv3toKt2yBQY7Rv0peP1y68nuHKZcp/QZDj
QksDvIJHacpz886KQ2p9xxOCrIO3RTVNULPzrucUNuJ2a5W4IGPGwnrNLKsriRH6
CXQjRu19+kbczmPtrlHu7gj6MuCp58sIyXbsOPCnsZBaAOcf72qeKfVa4tKlpobI
e5YDg+UiJXYv1Kfj9t9tftSfaCym0jK+uE3++y5+ah/k+u/VcEb9b6DrlfG8KaJI
/OYVVmRWVIHEY2QjVOsVEXAARjpugf6cXIn8epw/CMaC0yVg4O9OGDrIz/41DVgk
/b3jlr1oWMRVASSqgOQxE2LrNNAF/XiFwpiLUCBHR0fXX9iKJM7nK6AXlf4hrd31
sMm1hICJvQwuRP+1xCHHs3qKk7oFAdyEdMiYthcOoAaynGmP4vPEMB6ZdOSxd/yN
7IfBT1U7nCtgo4FpAn4FTnkXRd1DqPpvKLqe0kkQCKo9ZWRnnIS1hAs9VJXS6zou
1IohZewrS/TWnPU+ArE/QJ4WkX2dBAe0Q2HMvuXs6J6Rfn+H2eZC+zTh0kvqA0ic
UiQAgJB0ohXWEah9BJbIYNULLpbssUySDLPrxYdZRPn5MFKqCXmWw5tAEV8mIcg4
obB6DxVGHz93iPD0XJyBGWpCdcwItzdngnMR316tX4R3jMy2U0hrzk5o01GoZQQ/
eGYsuPzm4zqVJ8S9gUMFR8ocoPWqwhU+Sq4QPCIQEGf5gwxxt9j7L2TtRbD7kxnD
23aPL/wuk45EFJKTQCeZ3kjUqCCuSbqBJXGEjiiUsahus+yJULJ8boFWu1T8zW7m
/CBK1TzY9Fh+gbdOtN4b+zWUrpk5E7ZctFhEwojVFBwgPzUIZEkap4gH+/F5d3rI
uWFBcN7QER35w08au/k5kLKHDh+5U15ltkJ38P1pJYFE0p5/K81YWnfvUaJmynJY
DfnK0X6Au69/F6+QR4S+bchbrqk73plIiXbkwqapQ26QH5ztIK3T6/nY6RKa7u6N
WvnskpAE1uxf+soM+BpEzpTl2gQQmEaZnh9Ci3ZnpTZE5tkAL0oTqXL9sToyeynz
Z91m+uKxYRjFvc1KPh+I8dUR0E5qXhBdRR7heIlm7/Dg+UqCUkXoyW89G14u/mJT
jB50CHxLAFOSeXnKn8G/9r0HA565r8L/z8D12sli5T1zHicZJ80211zCSezdF+GG
mI12jfOZmNIBgCVUtpOBNX5IDbr7L+0bM6vuDmXRhnhX4f34+5HGfj3ntLAV7bmn
IOEGIhyoGFnoxyWaT0xxa6MVQJazcKSyyyhO8UixFEoKI3drpFXB0tXVb0CLFq+y
QVp9dHgO30tzw/yCmCpbQQe8UFRYyC74wBf9yPWDtN6MW/IeEZCBoYaeMR+WyonM
mdq5xbriETXD4zha3NGAv2qnzkfkKMBcVeUULulyixMsHmNNJkINNEsj5deVYtOt
4j5EK/aZsTqTgrVq3oGlwgXM1YptvlVR+HPWZfpjwbL+7/NhmK7AhEPo+qQCVD7t
ZGRbdbB7FEN23m88IMSd8xkhS2M1pop1Cj7lP/dA+DD/dgcjP2bw7K923d4r3CcS
1yxPhLKM
-----END ENCRYPTED PRIVATE KEY-----
```

これらのファイルを準備して、Guardium システムにインポートする (scp、ftp などを使用) か、Guardium システム上の cli インターフェースにコピー・アンド・ペーストします。

手順

1. CLI を使用して Guardium システムにログインします。
2. cli> store certificate keystore [import | console] と入力して秘密鍵を保管します。 import は保存されたファイルを取得し、ファイルの内容をコンソール・インターフェースにコピー・アンド・ペーストします。ファイルと共に保存されたパスワードを尋ねられます。これは、証明書を作成するために CA に提供したものか、あるいは多くの場合、CA からファイルが送信されたときに、CA から提供されたパスワードです。Guardium システムでは次のように表示されます。

```
temp4> store system key console
Please paste your new system key, in PEM format.
Include the BEGIN and END lines, then press CTRL-D.

-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIE4jAcBgoqhkiG9w0BDAEDMA4ECAKuMvS1a9T4AgIBYgSCBMB/RzDgm4xpRDnA
Cd//wgD9c8junXmVpyJWXWgXhlj3Sgv3toKt2yBQY7Rv0peP1y68nuHKZcp/QZDj
QksDvIJHacpz886KQ2p9xxOCrIO3RTVNULPzrucUNuJ2a5W4IGPGwnrNLKsriRH6
CXQjRu19+kbczmPtrlHu7gj6MuCp58sIyXbsOPCnsZBaAOcf72qeKfVa4tKlpobI
e5YDg+UiJXYv1Kfj9t9tftSfaCym0jK+uE3++y5+ah/k+u/VcEb9b6DrlfG8KaJI
/0YVVmRWVIHEY2QjVOsVEXAARjpugf6cXIn8epw/CMaC0yVg4O9OGDrIz/41DVgk
/b3jlr1oWMRVASSqgOQxE2LrNNAF/XiFwpiLUCBHR0fXX9iKJM7nK6AXlf4hrd31
sMm1hICJvQwuRP+1xCHHs3qKk7oFAdyEdMiYthcOoAaynGmP4vPEMB6ZdOSxd/yN
7IfBT1U7nCtgo4FpAn4FTnkXRdlDqPpvKLqe0kkQCKo9ZWRnnIS1hAs9VJXS6zou
1IohZewrS/TWnPU+ArE/QJ4WkX2dBAe0Q2HMvuXs6J6Rfn+H2eZC+zThOkvqA0ic
UiQAgJB0ohXWEah9BJbIYNULLpbssUySDLPrxYdZRPn5MFKqCXmWw5tAEV8mIcg4
obB6DxVGHz93iPD0XJyBGWpCdcwItzdngnMR316tX4R3jMy2U0hrzk5o01GoZQQ/
eGYsuPzm4zqVJ8S9gUMFR8ocoPWqwhU+Sq4QPCIQEGf5gwxxt9j7L2TtRbD7kxnD
23aPL/wuk45EFJKTQCeZ3kjUqCCuSbqBJXGEjiiUsahus+yJULJ8boFWu1T8zW7m
/CBK1TzY9Fh+gbdOtN4b+zWUrpk5E7ZctFhEwojVFBwgPzUIZEkap4gH+/F5d3rI
uWFBcN7QER35wO8au/k5kLKHDh+5Ul5ltkJ38P1pJYFE0p5/K81YWnfvUaJmynJY
DfnK0X6Au69/F6+QR4S+bchbrqk73plIiXbkwqapQ26QH5ztIK3T6/nY6RKa7u6N
WvnskpAE1uxf+soM+BpEzpTl2gQQmEaZnh9Ci3ZnpTZE5tkAL0oTqXL9sToyeynz
Z91m+uKxYRjFvc1KPh+I8dUR0E5qXhBdRR7heIlm7/Dg+UqCUkXoyW89G14u/mJT
jB50CHxLAFOSeXnKn8G/9r0HA565r8L/z8D12sli5T1zHicZJ8O211zCSezdF+GG
mI12jfOZmNIBgCVUtpOBNX5IDbr7L+0bM6vuDmXRhnhX4f34+5HGfj3ntLAV7bmn
IOEGIhyoGFnoxyWaT0xxa6MVQJazcKSyyyhO8UixFEoKI3drpFXB0tXVb0CLFq+y
QVp9dHgO3Otzw/yCmCpbQQe8UFRYyC74wBf9yPWDtN6MW/IeEZCBoYaeMR+WyonM
mdq5xbriETXD4zha3NGAv2qnzkfkKMBcVeUULulyixMsHmNNJkINNEsj5deVYtOt
4j5EK/aZsTqTgrVq3oGlwgXM1YptvlVR+HPWZfpjwbL+7/NhmK7AhEPo+qQCVD7t
ZGRbdbB7FEN23m88IMSd8xkhS2M1pop1Cj71P/dA+DD/dgcjP2bw7K923d4r3CcS
1yxPhLKM
-----END ENCRYPTED PRIVATE KEY-----
Enter pass phrase for /var/tmp/key.pem:
writing RSA key

ok
temp4>
```

3. cli> store certificate sniffer [import | console] と入力して署名済み証明書をインポートします。 証明書に関する情報が表示され、証明書を保管するかどうかを確認するよう求められます。次のようになります。

```
temp4> store system certificate console
Please paste your new system certificate, in PEM format.
Include the BEGIN and END lines, then press CTRL-D.

-----BEGIN CERTIFICATE-----
MIID5jCCAtCgAwIBAgIBCTALBgkqhkiG9w0BAQUwgYAxCzAJBgNVBAYTAkNBMRkw
FwYDVQQIExBCcml0aXNoIENvbHVtYmlhMREwDwYDVQQHEwhWaWN0b3JpYTEUMBIG
A1UEChMLUUFfdGVzdF92aWMxFDASBgNVBAsTC1ZpY3RvcmlhX1FBMRcwFQYDVQQD
Ew5WaWN0b3JpYV9RQV9DQTAeFw0xMDExMTUyMDUwNThaFw0xNTExMTUyMDUwNTha
MIGEMQswCQYDVQQGEwJDQTEZMBcGA1UECBMQQnJpdGlzaCBDb2xvbWJpYTESMBAG
A1UEBxMJVmFuY291dmVyMQswCQYDVQQKEwJRQTESMBAGA1UECxMJUUFfU0FNUExF
MSUwIwYDVQQDExxTYW1wbGVfZ2l2ZW5DZXJ0LnZpY3RvcmlhLnFhMIIBIDALBgkq
hkiG9w0BAQEDggEPADCCAQoCggEBALdt2XhJEJIAognblgFNoiomVvcGIe9PHsOt
3fP5DxAL5PVm+UCRS0xnnCoKe3pdJNagepDwBa2K5UsHbK4vkHJEgWWEd4bx2Ks7
kRoHr83TXK+W8al1HGKRwUSwN0hfm/KV4v8ZX3AFWs2c/BJ2iA77a0mhU0JuWa1/
/scXitJB5ykPjFb2EuReA6ELphaQ/itjZIQTTeVxbamyx42ia9J5B07lFTp3qGdU
yXUwlEFX0HFPmknAAaQrSnpMdQjkOKgzxU0NTV4ILA66hCVkX+ezQeJA90s/AHA5
hAY2fyurKZsOowoE0xlEpFWCGf87zxfkmTaWthqCS0NzEcJ6efECAwEAAaNrMGkw
HwYDVR0jBBgwFoAUdEgevPLMipuAlENuyxgfX7R0WckwDAYDVR0TAQH/BAIwADAP
BgNVHQ8BAf8EBQMDB7gAMCcGA1UdJQQgMB4GCCsGAQUFBwMEBggrBgEFBQcDAgYI
KwYBBQUHAwEwCwYJKoZIhvcNAQEFA4IBAQB+Plm22B72eRESk9rgNdOB+l48k5xw
QHf64TJcoI/Vcz0vtGBC1jDfZzVDIHJaX6ZMtdO23QO6rtjQSnhjhbOt/EiimaNO
tysJOE999E+HQ7UpKYKvdoznwpYEthyoQJ9quKqQc1YW/l8pQMYDEdK8c7yMbJpv
mrxO8h+G3YYQMT6A9KAk/GPA+yKl0fCuogBhPyWM28q35EAIw6H9ahQigWhNkzrL
DCYO8VCLi3BX+OohLvurkyzJm2lnbm7BSb5QB4jISlRBGPoIWhK6VPypboyABfVs
yWHFVs10EUb0Maa/XNZILTAoYAbYK5sX7Ri5hr5KhxnmDd9AJMQsLfik
-----END CERTIFICATE-----
```

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 9 (0x9)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=CA, ST=British Columbia, L=Victoria, O=QA_test_vic, OU=Victoria_QA, CN=Victoria_QA_CA
        Validity
            Not Before: Nov 15 20:50:58 2010 GMT
            Not After : Nov 15 20:50:58 2015 GMT
        Subject: C=CA, ST=British Colombia, L=Vancouver, O=QA, OU=QA_SAMPLE, CN=Sample_givenCert.victoria.qa
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (2048 bit)
                Modulus (2048 bit):
                    00:b7:6d:d9:78:49:10:92:00:a2:09:db:96:01:4d:
                    a2:2a:26:56:f7:06:21:ef:4f:1e:c3:ad:dd:f3:f9:
                    0f:10:0b:e4:f5:66:f9:40:91:4b:4c:67:9c:2a:0a:
                    7b:7a:5d:24:d6:a0:7a:90:f0:05:ad:8a:e5:4b:07:
                    6c:ae:2f:90:72:44:81:65:84:77:86:f1:d8:ab:3b:
                    01:1a:07:af:cd:d3:5c:af:96:f1:a9:75:1c:62:91:
                    c1:44:b0:37:48:5f:9b:f2:95:e2:ff:19:5f:70:05:
                    5a:cd:9c:fc:12:76:88:0e:fb:6b:49:a1:53:42:6e:
                    59:ad:7f:fe:c7:17:8a:d2:41:e7:29:0f:8c:56:f6:
                    12:e4:5e:03:a1:0b:a6:16:90:fe:2b:63:64:84:13:
                    4d:e5:71:6d:a9:b2:c7:8d:a2:6b:d2:79:07:4e:e5:
                    15:3a:77:a8:67:54:c9:75:30:94:41:57:d0:71:4f:
                    9a:49:c0:01:a4:2b:4a:7a:4c:75:08:e4:38:a8:33:
                    c5:4d:0d:4d:5e:08:2c:0e:ba:84:25:64:5f:e7:b3:
                    41:e2:40:f7:4b:3f:00:70:39:84:06:36:7f:2b:ab:
                    29:9b:0e:a3:0a:04:d3:19:44:a4:55:82:19:ff:3b:
                    cf:17:e4:99:36:96:b6:1a:82:4b:43:73:11:c2:7a:
                    79:f1
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Authority Key Identifier:
                keyid:74:48:1E:BC:F2:CC:8A:9B:80:94:43:6E:CB:18:1F:5F:B4:74:59:C9

            X509v3 Basic Constraints: critical
                CA:FALSE
            X509v3 Key Usage: critical
                Digital Signature, Key Encipherment, Data Encipherment, Key Agreement
            X509v3 Extended Key Usage:
                E-mail Protection, TLS Web Client Authentication, TLS Web Server Authentication
    Signature Algorithm: sha1WithRSAEncryption
        7e:3e:59:b6:d8:1e:f6:79:11:12:93:da:e0:35:d3:81:fa:5e:
        3c:93:9c:70:40:77:fa:e1:32:5c:a0:8f:d5:73:3d:2f:b4:60:
        42:d6:30:df:67:35:43:20:72:5a:5f:a6:4c:b5:d3:b6:dd:03:
        ba:ae:d8:d0:4a:78:63:85:b3:ad:fc:48:a2:99:a3:4e:b7:2b:
        09:38:4f:7d:f4:4f:87:43:b5:29:29:82:af:76:8c:e7:c2:96:
        04:b6:1c:a8:40:9f:6a:b8:aa:90:73:56:16:fe:5f:29:40:c6:
        03:11:d2:bc:73:bc:8c:6c:9a:6f:9a:bc:4e:f2:1f:86:dd:86:
        10:31:3e:80:f4:a0:24:fc:63:c0:fb:22:a5:d1:f0:ae:a2:00:
        61:3f:25:8c:db:ca:b7:e4:40:08:c3:a1:fd:6a:14:22:81:68:
        4d:93:3a:cb:0c:26:0e:f1:50:8b:8b:70:57:f8:ea:21:2e:fb:
        ab:93:2c:c9:9b:69:67:6e:6e:c1:49:be:50:07:88:c8:4a:54:
        41:18:fa:08:5a:12:ba:54:fc:a9:6e:8c:80:05:f5:6c:c9:61:
        c5:56:cd:74:11:46:f4:31:a6:bf:5c:d6:48:2d:30:28:60:06:
        d8:2b:9b:17:ed:18:b9:86:be:4a:87:19:e6:0d:df:40:24:c4:
        2c:2d:f8:a4

Do you want to store this certificate? (y/n)
y
ok
temp4 >
```

4. 新しい証明書を有効にするために、inspection-core を再始動します。

親トピック: Windows: SSL 証明書を使用する S-TAP 認証のセットアップ

Windows: x.509 証明書認証を使用するための S-TAP の構成

このタスクについて

最初に、証明書の CA および CN として割り当てた内容を記録します。覚えていない場合は、CLI コマンド show system certificate を使用して値を表示します。

```
temp4 > show system certificate
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 8 (0x8)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=CA, ST=British Columbia, L=Victoria, O=QA_test_vic, OU=Victoria_QA, CN=Victoria_QA_CA
        Validity
            Not Before: Nov  1 21:09:38 2010 GMT
            Not After : Nov  1 21:09:38 2015 GMT
        Subject: C=CA, ST=BC, L=Newbury, O=QA_Sample1, OU=Sample_QA, CN=sample1_qa.victoria
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (2048 bit)
                Modulus (2048 bit):
```

Guardium システムにインストールされた証明書の CN と、Guardium システム上の証明書に署名した CA の公開鍵が必要です。Guardium システムの証明書に署名したのと同じ CA により署名された証明書取り消しリストも必要になることがありますが、ここでは必須ではありません。

guard_tap.ini の関連パラメーターは以下のとおりです。

```
; Where is the CA certificate
guardium_ca_path=NULL
; What's the CN to expect from the SqlGuard certificate?
sqlguard_cert_cn=NULL
; Path to crls file or dir
guardium_crl_path=NULL
```

パラメーターに対して値を使用しない場合、そのパラメーターを guard_tap.ini に含めません。これは特に CRL パスに関連します。あるいは、証明書認証を中止して TLS に戻るような場合です。

手順

1. CA から送信された CA の公開鍵 (および必要な場合は CRL) を、S-TAP ホスト上のディレクトリーにコピーします。このディレクトリーを記録しておきます。
2. `guardium_ca_path=[CA へのパス.pem]` を設定します。
3. `sqlguard_cert_cn=[Guardium システムの完全 CN または部分 CN (* をワイルドカードとして使用)]` を設定します。
4. この時点で証明書失効リストを使用する場合は、`guardium_crl_path=[crl へのパス.crl]` を設定します。 次のようになります。

```
guardium_ca_path=/var/tmp/pki/Victoria_QA_CA.pem
sqlguard_cert_cn=sample1_qa.victoria
guardium_crl_path=/var/tmp/pki/Victoria_QA_CA.crl
```

5. tls=1 を変更します。
6. S-TAP を再始動します。 これで Openssl を使用して接続されます。

親トピック: Windows: SSL 証明書を使用する S-TAP 認証のセットアップ

Windows: Db2 出口ライブラリーの使用

Db2 出口メカニズムを使用すると、Guardium は、トラフィックが暗号化されているかどうか、およびローカルかリモートかに関係なく、すべての Db2 トラフィックを取得できます。このソリューションは、S-TAP 構成を単純化し、ネイティブの Db2 サポートを提供します。

このタスクについて

Db2 出口は、DB2_Exit メカニズムを介して Guardium ライブラリーを Db2 に組み込みます。DB2_Exit は直接 Guardium S-TAP と通信し、トラフィックが暗号化されているかどうかに関係なく、ローカルとリモートの両方の Db2 トラフィックをすべて転送します。Db2 出口は TCP トラフィックと SHM トラフィックをキャプチャーします。

Db2 出口は、強制終了、および UID チェーンをサポートします。

制限事項:

- DB2 Exit は Guardium データ・マスキング (修正/編集) をサポートしません。
- Guardium ファイアウォール (V10.1.2 以降) には Db2 バージョン 10.1 以降が必要です。
- ストアード・プロシージャー: DB2-Exit はストアード・プロシージャーをモニターします。Guardium はストアード・プロシージャーに何が含まれているかを認識しないため、プロシージャー内からの SQL はキャプチャーされません。

手順

1. 各インスタンスごとに、DB2 SQLLIB フォルダー内に新しいフォルダー $DB2PATH¥security¥plugin¥commexit¥instance_name を作成します。 例: C:¥Program Files¥IBM¥SQLLIB¥security¥plugin¥commexit¥DB2_01
2. 対応する DLL を、S-TAP のインストール・ディレクトリーから作成したディレクトリーにコピーします。
   - 32 ビットの Db2 の場合:
     - db2fexitx86.dll
     - db2exitx86.dll
   - 64 ビットの Db2 の場合:
     - db2exitx64.dll
     - db2fexitx64.dll
3. Db2 インスタンスを停止し、次のコマンドを実行します。
   - 32 ビットの場合: UPDATE DBM CFG USING COMM_EXIT_LIST db2fexitx86
   - 64 ビットの場合: UPDATE DBM CFG USING COMM_EXIT_LIST db2fexitx
4. Db2 インスタンスを開始します。
5. Db2 出口の検査エンジンを、プロトコル Db2 出口で追加します。「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」に移動します。Windows: 検査エンジン・パラメーターで、パラメーターの記述を参照してください。guard_tap.ini を変更することもできますが、GUI では一部の情報が自動的に設定され、所定の検証も行われるため、GUI を使用する方が大幅に簡単になります。guard_tap.ini を変更する場合は、
   - [DB_DB2_EXIT1]
   - DB_TYPE=DB2_EXIT
   - INSTANCE_NAME=Service_name

   TAP セクションで、パラメーター DB2_EXIT_DRIVER_INSTALLED=1 を設定してください。

   サービス名は、インスタンス名ではありません。S-TAP インストール・フォルダーの db2tap ユーティリティー、または制御パネルを使用して、サービス名を判別できます。 2 番目のダッシュ (-) 区切り文字の後ろに続くサービス名の部分をインスタンス名に設定します。例えば、コントロール・パネル内でサービス名が `DB2 - DB2COPY1 - DB2-01-0` である場合、INSTANCE_NAME を `DB2-01-0` に設定します。
6. この機能の使用を停止し、Db2 を停止するには、次のコマンドを実行してから、Db2 を再始動します。db2 UPDATE DBM CFG USING COMM_EXIT_LIST NULL

親トピック: Windows: S-TAP の構成

Windows: S-TAP 構成パラメーターの編集

S-TAP 構成は、インストール後に GIM または UI を使用して変更できます。上級者の場合は、データベース上の構成ファイルで変更できます。

注: GUI のパラメーターは安全に変更できます。GUI にないパラメーターは高度なパラメーターであり、変更する必要はほとんどありません。これらは Guardium のサポート担当員または上級者が使用します。
注意:
熟練したユーザーである場合や IBM 技術サポートに相談済みの場合を除いて、拡張パラメーターは変更しないでください。

GUI でパラメーターを変更することができます。Windows: GUI からの S-TAP の構成を参照してください。

S-TAP バンドルが GIM を使用してインストールされている場合は、GIM を使用すると簡単にパラメーターを変更できます。クライアント別の設定を参照してください。「クライアント別の設定」ページの「パラメーターの選択」リボンに、[TAP] パラメーターの構文 parameter=value のコマンド WINSTAP_CMD_LINE を使用して、任意のパラメーターを入力できます。これは、guard_tap.ini で追加または更新されます。

注意:
コマンド WINSTAP_CMD_LINE の使用時には入力の検証が行われません。このコマンドは注意して使用してください。熟練したユーザーである場合や IBM 技術サポートに相談済みの場合を除いて、拡張パラメーターは変更しないでください。

データベース・サーバーから構成ファイルを変更する必要がある場合は、このセクションで説明する手順に従ってください。

guard_tap.ini を変更した後、S-TAP を再始動する必要があります。GIM を使用している場合は、GIM により S-TAP が自動的に再始動されます。

注意:
パラメーターは、[Version]、[TAP]、[SQLGuard]、[DB_<name>] のそれぞれの関連セクションに追加する必要があります。

1. root アカウントを使用して、データベース・サーバー・システムにログオンします。
2. S-TAP を停止します。
3. 構成ファイル (guard_tap.ini) のバックアップ・コピーを作成します。デフォルトのファイル・ロケーションは ¥Program Files¥IBM¥Windows S-TAP¥Bin¥ です。
4. 構成ファイルをテキスト・エディターで開きます。
5. 必要に応じてファイルを編集します。
6. ファイルを保存します。
7. S-TAP を再始動して、変更が取り込まれているかどうかを確認します。

- Windows: Guardium ホスト (SQLGuard) パラメーター
  以下のパラメーターは、この S-TAP が接続できる Guardium システムを示します。このセクションのパラメーターはすべて、基本パラメーターであり、[SQL_GUARD] セクションに表示されます。
- Windows: 一般パラメーター
  これらのパラメーターは、Windows サーバー上で実行されている S-TAP および S-TAP のインストール先のサーバーの基本プロパティーを定義し、他のどのカテゴリーにも分類されません。
- Windows: 検査エンジン・パラメーター
  これらのパラメーターは、Windows サーバー上のデータ・リポジトリーをモニターするために S-TAP が使用する検査エンジンの動作に影響を与えます。
- Windows: ファイアウォール・パラメーター
  これらのパラメーターは、ファイアウォールに関する S-TAP の動作に影響を与えます。
- Windows: 照会再書き込みパラメーター
  照会再書き込みパラメーターは、ディスカバリーに関する S-TAP の動作に影響を与えます。
- Windows: discovery パラメーター
  discovery パラメーターは、データベース・インスタンスのディスカバーと現在アクティブな S-TAP への結果の送信を行うオートディスカバリー機能の動作を定義します。
- Windows: デバッグ・パラメーター
  これらのパラメーターは、S-TAP デバッグの動作に影響を与えます。
- Windows: 構成監査システム (CAS) パラメーター
  これらのパラメーターは、CAS の動作に影響を与えます。
- Windows: ドライバー・パラメーター
  これらのパラメーターは、S-TAP が対話するいくつかのドライバーの動作に影響を与えます。

親トピック: Windows: S-TAP の構成

Windows: Guardium ホスト (SQLGuard) パラメーター

以下のパラメーターは、この S-TAP が接続できる Guardium システムを示します。このセクションのパラメーターはすべて、基本パラメーターであり、[SQL_GUARD] セクションに表示されます。

GUI	GIM	guard_tap.ini	デフォルト値	記述
✔ (チェック・マークは 1 次ホストを示します)		PRIMARY		この S-TAP の 1 次 Guardium システムを示します。guard_tap.ini での設定: 0 = 2 次、1 = 1 次
		TAP_GUARD_TCP_PORT	9500	読み取り専用。S-TAP が Guardium システムに接続するために使用するポート。
Guardium ホスト	WINSTAP_SQLGUARD_IP	SQLGUARD_IP	NULL	S-TAP のホストとしての役割を果たす Guardium システムの IP アドレスまたはホスト名。[SQLGuard_1]、[SQLGuard_2]、以降同様を追加することで、複数のホストを定義できます。

親トピック: Windows: S-TAP 構成パラメーターの編集

Windows: 一般パラメーター

これらのパラメーターは、Windows サーバー上で実行されている S-TAP および S-TAP のインストール先のサーバーの基本プロパティーを定義し、他のどのカテゴリーにも分類されません。

これらのパラメーターは、S-TAP プロパティー・ファイルの [VERSION] セクションに格納されています。

表 1. [VERSION] セクションの S-TAP 構成パラメーター

GUI	guard_tap.ini	記述
	STAP_CLIENT_BUILD	読み取り専用。インストールされている S-TAP のビルド・バージョン。
バージョン	PROTOCOL_VERSION	読み取り専用。Guardium システムのバージョン。

これらのパラメーターは、S-TAP プロパティー・ファイルの [TAP] セクションに格納されています。

表 2. [TAP] セクションの S-TAP 構成パラメーター

GUI	GIM	guard_tap.ini	デフォルト値	記述
		TAP_TYPE	wstap	読み取り専用。インストールされている S-TAP エージェントのタイプ。
バージョン		TAP_VERSION		読み取り専用。サーバーにインストールされている S-TAP のバージョン。
S-TAP ホスト		TAP_IP		読み取り専用。ファイル・システムのモニター・サービスによって SOFTWARE_TAP_HOST パラメーターの代わりに使用されます。両方のパラメーターに同じ値を指定する必要があります。
すべてが制御可能	WSTAP_ALL_CAN_CONTROL	ALL_CAN_CONTROL	0	0= S-TAP は 1 次 Guardium システムからのみ制御できます。1= S-TAP は任意の Guardium システムから制御できます。
ロード・バランシング	WINSTAP_PARTICIPATE_IN_LOAD_BALANCING	PARTICIPATE_IN_LOAD_BALANCING	0	Guardium システムに対する S-TAP のロード・バランシング (エンタープライズ・ロード・バランシングではない) を制御します。 • 0: ロード・バランシングなし。 • 1: ロード・バランシング。SQLGuard セクションで定義されている 1 次サーバーと 2 次サーバーの間でトラフィックのバランスを取ります。 • 2: 冗長。完全にミラーリングされた S-TAP によって、SQLGuard セクションで定義されているすべての 1 次サーバーと 2 次サーバーにすべてのトラフィックが送信されます。 • 3: ハードウェア・ロード・バランシング。Guardium では、F5 や Cisco などのロード・バランサーが使用されます。S-TAP はトラフィックをロード・バランサーに送信し、ロード・バランサーはそれをプール内のいずれかのコレクターに転送します。 1 次サーバー、2 次サーバーなどのサーバーを指定するには、SQLGUARD セクションでプライマリー・パラメーターを使用します。 このパラメーターが 0 に設定されているときに、複数の Guardium システムでトラフィックをモニターしている場合は、1 次以外の Guardium システムをフェイルオーバー用に使用することができます。
TLS の使用		USE_TLS	0	1=SSL を使用して、エージェントと Guardium システムとの間のトラフィックを暗号化します。 0=暗号化しません。警告 - エージェントと Guardium システム間のトラフィックは平文です。 Guardium では、可能な場合は常に S-TAP とコレクター間のネットワーク・トラフィックを暗号化することを推奨しています。この暗号化を無効にする必要があるのは、パフォーマンスの優先順位がセキュリティーより高い場合のみです。
TLS フェイルオーバー		FAILOVER_TLS	1	V10.5 で非推奨になっています。1= 何らかの理由で SSL 接続を使用できない場合は、非セキュア接続を使用するようにフェイルオーバーします。0= セキュア接続のみを使用します。
		NUMBER_OF_PROCESSORS	4	読み取り専用。マシンのプロセッサー数。
		ALTERNATE_IPS		このデータベース・サーバーへの接続に使われる代替または仮想 IP アドレスのコンマ区切りのリスト。これが使用されるのは、複数の IP または仮想 IP を持つ複数のネットワーク・カードがサーバーにある場合だけです。この S-TAP 用に定義された S-TAP ホスト IP、またはここにリストされるいずれかの代替 IP に宛先 IP が一致する場合にのみ、S-TAP はトラフィックをモニターします。このため、すべての仮想 IP をここにリストすることをお勧めします。
		STAP_STATISTIC		S-TAP が S-TAP に関する統計情報をスニファーに送信する間隔。 • 正の整数は時間を表します • 負の整数は分を表します • 0=送信しない デフォルトは -1 (毎分) です。
		DB2_TAP_INSTALLED	0	Db2 共有メモリー・トラフィックをスニッフィングするには 1 に設定します。 1 に設定する場合、Db2 TAP サービスを開始します。

GUI	GIM	guard_tap.ini	デフォルト値	記述
		DB2_EXIT_DRIVER_INSTALLED		S-TAP への Db2 の統合: Db2 出口ライブラリー統合を有効にするには、1 に設定します。1) S-TAP が Db2 エンジンからすべての Db2 トラフィックをキャプチャーします。これは特定の Db2 リリース (10.1 以降) でのみ使用できることに注意してください。2) この方式を使用する場合、ファイアウォール、および修正機能と編集機能はサポートされません。また、ストアード・プロシージャーはキャプチャーされません。3) 暗号化プロトコルとネットワーク・プロトコルに関係なく、すべての Db2 トラフィックをピックアップすることができます。4) このソリューションは、このバージョンの Db2 を導入するユーザーの S-TAP 構成を簡素化し、固有の Db2 サポートをそれらのユーザーに提供します。
		DB2_SHMEM_DRIVER_INSTALLED		非推奨。db2_tap_installed に置き換えられています。
		DB2_SHMEM_DRIVER_LEVEL		推奨されません
		DC_COLLECT_FREQ	24	v10.5 で非推奨になっています。収集の頻度を時間単位で指定します。最小値は 1、最大値は 24 です。GuardiumDC は、ユーザー・アカウント (SID およびユーザー名) の更新を 1 次ドメイン・コントローラーから収集し、その後、Guardium_S-TAP にその変更内容をシグナル通知して、S-TAP 内部の SID/UserName のマップを更新するサービスです。S-TAP は、マップから解決済みの SID を見つけられない場合、1 次ドメイン・コントローラーからこれを取得しようとします。その場合、S-TAP は、デバッグ・ログ (レベル 7) にメッセージ「SID *** のアカウント名 *** を取得しました (`The account name *** has been retrieved for SID ***`)」を記録します。
		DC_COLLECT_MAXUSERS	200,000	v10.5 で非推奨になっています。収集するユーザーの最大数。最小値は 10,000 です。
		DOMAIN_CONTROLLER		SID/ユーザー名のマップを読み取る特定のコントローラーの名前。
		HIGH_RESOLUTION_TIMER	0	0: ミリ秒単位のタイム・スタンプを送信します。1: マイクロ秒単位のタイム・スタンプを送信しますが、ミリ秒のシステム・タイマーを使用します (これはシステム・パフォーマンス・ヒットを減らすためであり、ミリ秒数を 1000 倍します)。 2: マイクロ秒単位のタイム・スタンプを送信し、高解像度 Windows タイマーを使用します (最も正確)。 1 および 2 の場合、S-TAP は、PacketData 内の予約済みバイトを 1 に設定することにより、マイクロ秒が送信されていることを Guardium システムに示します。
		BUFFER_FILE_SIZE	50	拡張機能。バッファーの初期サイズ。範囲は 5 から 1000 (MB) です。
		BUFFER_FILE_NAME		v10.5 で非推奨になっています。BUFFER_MMAP_FILE=1 の場合、メモリー・マップ・ファイルの絶対パス。デフォルトは、WSTAP 作業フォルダー /StapBuffer/STAP_buffer.dtx です。
		BUFFER_MMAP_FILE	0	1=メモリー・マップ・ファイルのオプション。0=仮想メモリーの割り振り。
		BUFFER_FILE_MAX_SIZE	250	拡張機能。メモリー・コミットが拡張される最大サイズ (MB 単位)。 最大値は 1000 です。
		BUFFER_FILE_MEM_FOOTPRINT	8	拡張機能。合計メモリーのうち、動的バッファー増加のために割り振られる最大部分。デフォルト値の 8 は、合計メモリーの 1/8 を意味します。最小パラメーター値は 2 であり、合計メモリーの 1/2 より多くを割り振ることはできないことを意味します。
		DYNAMIC_BUFFER_INCREASE	0	拡張機能。動的バッファー機能を有効にします。現行の S-TAP セッションでバッファーの使用率が 75% になると、バッファー・サイズは 50MB 単位で増えます。この機能は、buffer_file_size、buffer_file_max_size、buffer_file_mem_footprint によって制御されます。この機能は、buffer_file_size、buffer_file_max_size、buffer_file_mem_footprint によって制御されます。 0: 無効、1: 有効
		SOFTWARE_TAP_HOST		S-TAP がインストールされているデータベース・サーバー・ホスト。DNS サーバーによって認識されている IP アドレスまたは名前を指定できます。デフォルトはありません。構成が無効な SOFTWARE_TAP_HOST は、有効なローカル IP に自動的に置き換えられます。
		TCP_ALIVE_MESSAGE	1	このパラメーターは、Guardium v10.x 以降、非推奨になりました。Guardium コレクターは UDP アライブ・メッセージを送信しなくなりました。
圧縮レベル		COMPRESSION_LEVEL	0	1 から 9 までの圧縮レベル。 0=圧縮なし。
		DISABLE_SHARED_MEMORY_IF_TURNED_ON	0	
		FILE_SNIFFER_FREQUENCY	45	次の項目の頻度 (秒): • 前の試行が成功しなかった場合の、Guardium システムへの登録の試行 • Program Files¥IBM¥Windows S-TAP¥Logs からコレクターにアップロードできる新しいログの S-TAP による確認
		MAXIMUM_PACKET_NUM	300,000	推奨されません
		MIN_BYTES_TO_COMPRESS	500	拡張機能。圧縮するメッセージの最小サイズ。
		NOT_SEND_TO_SQLGUARD	0	拡張機能。Guardium システムに何も送信しません。
		RECV_LEVEL	0	拡張機能。
メッセージ: リモート		REMOTE_MESSAGES	1	1=アクティブな Guardium システムにメッセージを送信します。0=メッセージを送信しません。
		SEND_LEVEL	0	高。スレッドの優先順位付けに使用されます。

GUI	GIM	guard_tap.ini	デフォルト値	記述
		SNIFFED_UDP_PORTS	88	推奨されません。
		SYNCH_FLAG	1	読み取り専用。v10.0 で非推奨になっています。パラメーターが UI と同期されているかどうかを示します。
		TAP_DBSERVER_NAMES		
		TAP_MIN_HEARTBEAT_INTERVAL	30	S-TAP が 2 次 Guardium バッファーへの書き込みを試行する前に、1 次 Guardium システム・バッファーへの書き込みを試行する最大時間。デフォルトは 30 秒です。つまり、デフォルトではフェイルオーバーの前に少なくとも 5 * 60/30 回、書き込みを試行します (TAP_MIN_TIME_BEFOREFAILOVER も使用)。
		TAP_MIN_TIME_BEFOREFAILOVER	5	S-TAP が 1 次 Guardium システムに接続できない場合、または 1 次 Guardium システムに接続はできるが、そのバッファーに書き込むことができない場合に、S-TAP が 2 次 Guardium システムに切り替えるまでの時間間隔 (分単位)。
		TCP_BUFFER_SIZE	60000	拡張機能。Guardiumにメッセージを送信する前に収集する最小バイト数。
		TIME_NETWORK	0	拡張機能。デバッグにのみ使用します。
		WEB_SERVER_CONNECTIONS	1	.net アプリケーションによる DB 接続の最大数。
		WEB_SERVER_INSTALLED	0	推奨されません。以前は、IIS Tap の有効化に使用されていました。
		WEB_SERVER_PORT	9000	Web サーバーのポート。
		GUARDIUM_CA_PATH	NULL	認証局証明書の場所。
		SQLGUARD_CERT_CN	NULL	Sqlguard 証明書で予期される共通名。
		GUARDIUM_CRL_PATH	NULL	証明書失効リストのファイルまたはディレクトリーへのパス。
		TAP_FAILOVER_SESSION_QUIESCE	240	以前のアクティブ・サーバーからのフェイルオーバー・リストの未使用セッションを、現在のアクティブ・サーバーから削除できるフェイルオーバー後の秒数。
		TAP_FAILOVER_SESSION_SIZE	8192	フェイルオーバー・セッション・リストのサイズ (MB 単位)。0=フェイルオーバー・セッションは保存されません。
		DB_IGNORE_RESPONSE		検査レベルの応答の無視。この機能を使用して、S-TAP レベルのすべてのデータベース応答を無視します。このとき、Guardium システムには何も送信されません。クライアントとのトランザクションのみが必要とされる特定の環境では、この機能を使用すると、S-TAP および Guardium システムの処理能力および処理時間が節約できます。この機能は、データベースからの不要な応答を、ネットワークからのロードを行わずに無視するより簡単な構成を行うために使用します。データベース・タイプをコンマ区切りでリストできます。または、すべてのタイプのデータベースからの応答を無視する場合は ALL を指定できます。 例えば、DB_IGNORE_RESPONSE=ALL、または DB_IGNORE_RESPONSE=MSSQL,DB2 のように指定します。サポートされる DB タイプ: ALL、MSSQL_NP、MSSQL、MYSQL、TRD、PGRS、MSSYB、 ORACLE、Db2、DB2_EXIT、INFORMIX、KERBEROS、FTP、CIFS。
		DB_IGNORE_RESPONSE_FILTER	0.0.0.0/0.0.0.0	応答を無視する IP/マスクのコンマ区切りリスト。指定された IP/マスクに対する、DB_IGNORE_RESPONSE で指定されたタイプのデータベース応答はすべて無視されます。 NULL: 応答のフィルタリングなし 0.0.0.0/0.0.0.0: すべての IP がフィルタリングされる
		DB_IGNORE_RESPONSE_LOCAL	1	ローカル DB 応答をフィルタリングするかどうか。0: いいえ、1: はい 注: このパラメーターでは TCP トラフィックはローカル・トラフィックと見なされません。
		DB_IGNORE_RESPONSE_BYPASS_BYTES	65535	このバイパス・バイトに達したときに DB_IGNORE_RESPONSE が開始されます。
		DB_IGNORE_RESPONSE_RESETS_PER_REQUEST	1	要求ごとに DB_IGNORE_RESPONSE_BYPASS_BYTES をリセットします。
	WSTAP_FAM_ENABLED	FAM_ENABLE	0	FAM モニター (クローラー) のグローバルな有効化/無効化。 0: 無効 1: 有効 このパラメーターはアップグレード後も引き続き適用されます。
		UPLOAD_FEATURE	1	Program Files¥IBM¥Windows S-TAP¥Logs からコレクターへのすべてのログ・ファイルへのアップロードを制御します。

親トピック: Windows: S-TAP 構成パラメーターの編集

Windows: 検査エンジン・パラメーター

これらのパラメーターは、Windows サーバー上のデータ・リポジトリーをモニターするために S-TAP が使用する検査エンジンの動作に影響を与えます。

これらのパラメーターは、データ・リポジトリー名を使用して、S-TAP プロパティー・ファイルの個々の [DB_<name>] 検査エンジン・セクションに保管されます。プロパティー・ファイルには、複数のセクションが存在する場合があり、各セクションは、この S-TAP によって使用される 1 つの検査エンジンを記述しています。

GUI	guard_tap.ini	デフォルト値	記述

GUI	guard_tap.ini	デフォルト値	記述
プロトコル	DB_TYPE		モニター中のデータ・リポジトリーのタイプ。
インスタンス名	INSTANCE_NAME		このサーバー上のデータベース・インスタンスの名前。MS SQL Server で暗号化を使用している場合、MS SQL Server で Kerberos 認証を使用している場合、Db2 出口のトラフィック収集の場合、Db2 SHM トラフィックの場合は、必須です。(デフォルトは MSSQLSERVER です。)
ポート範囲	PORT_RANGE_START		データベース・インスタンスに固有のポート範囲の先頭。TAP_DB_PORT_MAX とともに、このデータベース・インスタンスについてモニターされるポートの範囲を定義します。通常は、単一のポートだけが範囲に含まれます。Kerberos 検査エンジンの場合は、先頭と終了の値を 88-88 に設定します。範囲を使用する場合は余分なポートを範囲に含めないでください。含めた場合、S-TAP が不要なトラフィックの分析を試みたときにリソースが過剰に消費されることがあります。
ポート範囲	PORT_RANGE_END		データベース・インスタンスに固有の終了ポート範囲。
名前付きパイプ	NAMED_PIPE	sql¥query、sqllocal、¥MSSQLSERVER	ローカル・アクセス用に MS SQL Server で使用される名前付きパイプを指定します。名前付きパイプが使用される場合、このパラメーターに何も指定しなければ、S-TAP はレジストリーから名前付きパイプ名を取得しようとします。
クライアント IP/マスク	NETWORKS		IP アドレス/マスク形式 (n.n.n.n/m.m.m.m) のアドレスのリストを使用して、モニターされるクライアントを識別します。不適切な IP アドレス/マスクが入力された場合、S-TAP は始動しません。有効な値: • null=すべてのクライアントを選択 • 127.0.0.1/255.255.255.255=ローカル・トラフィックのみ 「クライアント IP /マスク」(networks) と「除外クライアント IP/マスク」(exclude networks) を同時に指定することはできません。 IP アドレスがデータベース・サーバーの IP アドレスと同じで、マスク **255.255.255.255** が使用される場合には、**ローカル**・トラフィックだけがモニターされます。アドレス/マスク値 **1.1.1.1/0.0.0.0** は、すべてのクライアントをモニターします。
除外クライアント IP/マスク	EXCLUDE_NETWORKS		モニターから除外されるクライアント IP アドレスと対応するマスクのリスト。このオプションを使用すると、特定のクライアントやサブネット (またはこれらの集合) を除くすべてのクライアントをモニターするよう S-TAP を構成できます。「クライアント IP /マスク」(networks) と「除外クライアント IP/マスク」(exclude networks) を同時に指定することはできません。
プロセス名	TAP_DB_PROCESS_NAMES		モニター対象のデータベース・サービス実行可能ファイル。例えば、Db2 IE の場合は TAP_DB_PROCESS_NAMES=DB2SYSCS.EXE となります。Oracle または MS SQL Server のみ (名前付きパイプが使用される場合)。Oracle では、2 つの項目 oracle.exe、tnslsnr.exe がリストに含まれます。MS SQL Server では、リストは 1 つの項目 sqlservr.exe だけです。
	PRIORITY_COUNT	20	セッション作成時に、最初の priority_count パケットは、高優先度フラグのマークを付けられ、コレクター上の特別な高優先度キューに転送されます。有効な範囲は、0 (無効) から 50 です。
ID	TAP_IDENTIFIER	NULL	オプション。検査エンジンを相互に識別するために使用します。このフィールドに値を指定しない場合、Guardium は、データベース・タイプと GUI 表示シーケンス番号を使用して、固有の名前をこのフィールドに自動入力します。

以下の追加のパラメーターは、IBM Db2 データベースで使用します。

表 1. Db2 検査エンジン用の追加の S-TAP 構成パラメーター

GUI	guard_tap.ini	デフォルト値	記述
Db2 共有メモリー・ 調整	DB2_FIX_PACK_ADJUSTMENT	80	データベース・タイプとして Db2 が選択され、共有メモリー接続がモニターされる場合に必須です。共有メモリー領域のサーバー部分へのオフセット。 Db2 共有メモリー・パケットの開始位置へのオフセットで、Db2 のバージョンによって異なります。8.2.1 より前では 32、8.2.1 以上では 80 です。
	DB2_LOG_SIZE		拡張機能。機能している DLL が、ログ項目を削除し始める前にバッファーに保持できる最大ファイル・サイズ (MB 単位)。
Db2 共有メモリー・クライアント位置	DB2_CLIENT_OFFSET	61440	共有メモリー領域のクライアント部分へのオフセット。データベース・タイプとして Db2 が選択され、共有メモリー接続がモニターされる場合に必須です。 クライアント・オフセットは、Db2 パラメーター ASLHEAPSZ の値を取得して 4096 を乗算することで適切なオフセットを計算できます。このパラメーターのデフォルトは、10 進数の 61440 です。このパラメーターは、Db2 データベース構成値 ASLHEAPSZ を使用して計算され、4096 で乗算されます。ASLHEAPSZ の値を取得するには、Db2 コマンド `db2 get dbm cfg` を実行して、ASLHEAPSZ の値を探します。通常、この値は 15 で、その結果、デフォルトの 61440 が算出されます。これが 15 ではない場合は、その値を使用し、4096 で乗算して、適切なクライアント・オフセットを算出します。
Db2 共有メモリー・ サイズ	DB2_SHMEM_SIZE	131072	Db2 共有メモリー・セグメント・サイズ。データベース・タイプとして Db2 が選択され、共有メモリー接続がモニターされる場合に必須です。

Windows: ファイアウォール・パラメーター

これらのパラメーターは、ファイアウォールに関する S-TAP の動作に影響を与えます。

これらのパラメーターは、S-TAP プロパティー・ファイルの [TAP] セクションに格納されています。

注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。

GIM	guard_tap.ini	デフォルト値	記述
WSTAP_FIREWALL_INSTALLED	FIREWALL_INSTALLED	0	ファイアウォール機能を有効にします。1=有効、0=無効。
WSTAP_FIREWALL_TIMEOUT	FIREWALL_TIMEOUT	2	ファイアウォールがタイムアウトになった場合に Guardium システムからの判定を待機する時間 (秒単位)。接続をブロックするのか、許可するのかを知るために、firewall_fail_close 値を調べます。任意の整数値を指定できます。
WSTAP_FAIL_CLOSE	FIREWALL_FAIL_CLOSE	0	Guardium システムから判定が返されず、firewall_timeout が期限切れになった場合、firewall_close = 0 では接続が許可され、firewall_close=1 では接続がブロックされます。

GIM	guard_tap.ini	デフォルト値	記述
WSTAP_DEFAULT_STATE	FIREWALL_DEFAULT_STATE	2	• 0=インストール済みポリシーのルールでトリガーされると、セッションごとにファイアウォールがアクティブ化されます。 • 1=デフォルトですべてのトラフィックに対してファイアウォール・ポリシー違反の監視が行われます。 • 2=デフォルトですべてのトラフィックに対してファイアウォール・ポリシー違反の監視が行われますが、最初の priority_count パケットでイベントによって監視がトリガーされない場合、セッションの監視はオフにされます。(V10.5 で導入。) 2 に設定された場合、ファイアウォール操作は Watch、Drop、Watch & Drop、および Unwatch の各コマンドで変更できます。FIREWALL_DEFAULT_STATE=2 のときに Watch コマンドを受け取った場合、2 から 1 に変更されて、接続は永続的にファイアウォール操作または照会再書き込み操作の対象となります。Drop または Watch & Drop を受け取った場合、接続は即時に強制終了します。FIREWALL_DEFAULT_STATE=2 のときに Unwatch コマンドを受け取った場合、2 から 0 に変更されて、接続はファイアウォール操作および照会再書き込み操作の対象から外れます。 このパラメーターを変更した後、S-TAP を再始動してください。
WSTAP_FORCE_WATCH	FIREWALL_FORCE_WATCH	NULL	ファイアウォール機能が有効になっていて、firewall_default_state が 0 の場合、セッションは、そのクライアント IP がこの IP/MASK 値リストのいずれかと一致すると、自動的に監視されます。リスト自体は、コンマで区切られます。例: 1.1.1.1/1.1.1.1,2.2.2.2/2.2.2.2
WSTAP_FORCE_UNWATCH	FIREWALL_FORCE_UNWATCH	NULL	ファイアウォール機能が有効になっていて、firewall_default_state が 1 の場合、セッションは、そのクライアント IP がこの IP/MASK 値リストのいずれかと一致すると、自動的に監視されません。リスト自体は、コンマで区切られます。例: 1.1.1.1/1.1.1.1,2.2.2.2/2.2.2.2,

親トピック: Windows: S-TAP 構成パラメーターの編集

Windows: 照会再書き込みパラメーター

照会再書き込みパラメーターは、ディスカバリーに関する S-TAP の動作に影響を与えます。

これらのパラメーターは、S-TAP プロパティー・ファイルの [TAP] セクションに格納されています。

注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。

GIM	guard_tap.ini	デフォルト値	記述
WINSTAP_QRW_INSTALLED	QUERY_REWRITE_INSTALLED	0	データベースの動的データ・マスキング機能を有効または無効にします。0 に設定すると、このグループの他のすべてのパラメーターが無視されます。 • 0=無効 • 1=有効

GIM	guard_tap.ini	デフォルト値	記述
WINSTAP_QRW_DEFAULT_STATE	QUERY_REWRITE_DEFAULT_STATE	0	照会再書き込みのアクティベーション・トリガーを設定します。firewall_default_state=1 の場合は 0 を指定する必要があります。 • 0=インストール済みポリシーのルールでトリガーされると、セッションごとに QRW がアクティブ化されます。 • 1=インストール済みポリシーに関係なく、すべてのセッションに対して QRW がアクティブ化されます。 • 2=デフォルトですべてのトラフィックに対して QRW ポリシー違反の監視が行われますが、最初の PRIORITY_COUNT パケットでイベントによって監視がトリガーされない場合、セッションの照会再書き込みはオフにされます。 2 に設定された場合、QRW 操作は Watch、Drop、Watch & Drop、および Unwatch の各コマンドで変更できます。状態 2 が有効なときに Watch コマンドを受け取った場合、この状態は 2 から 1 に変更されて、接続は永続的にファイアウォール操作または照会再書き込み操作の対象となります。Drop または Watch & Drop を受け取った場合、接続は即時に強制終了します。状態 2 が有効なときに Unwatch コマンドを受け取った場合、この状態は 2 から 0 に変更されて、接続はファイアウォール操作および照会再書き込み操作の対象から外れます。 このパラメーターを変更した後、S-TAP を再始動してください。
WINSTAP_QRW_FORCE_WATCH	QUERY_REWRITE_FORCE_WATCH	NULL	自動的に監視するクライアント IP/マスクのコンマ区切りリスト (例えば、1.1.1.1/1.1.1.1,2.2.2.2/2.2.2.2 など)。qrw_default_state が 0 の場合に有効です。 firewall_force_watch と同じ範囲には構成できません。
WINSTAP_QRW_FORCE_UNWATCH	QUERY_REWRITE_FORCE_UNWATCH	NULL	監視から除外するクライアント IP/マスクのコンマ区切りリスト (例えば、1.1.1.1/1.1.1.1,2.2.2.2/2.2.2.2 など)。firewall_default_state が 1 の場合に有効です。 firewall_force_unwatch と同じ範囲には構成できません。
WINSTAP_QUERY_REWRITE_FAIL_CLOSE	QUERY_REWRITE_FAIL_CLOSE	8	Guardium システムから判定が返されず、QUERY_REWRITE_TIMEOUT が期限切れになった場合に、QUERY_REWRITE_CLOSE = 8 であれば、照会再書き込み操作が続行されます。 QUERY_REWRITE_CLOSE = 12 であれば、接続が終了します。
WINSTAP_QUERY_REWRITE_TIMEOUT	QUERY_REWRITE_TIMEOUT	10	Guardium システムから判定が返されず、QUERY_REWRITE_TIMEOUT が期限切れになった場合に、QUERY_REWRITE_CLOSE = 0 であれば、照会再書き込み操作が続行されます。 QUERY_REWRITE_CLOSE = 1 であれば、接続が終了します。

親トピック: Windows: S-TAP 構成パラメーターの編集

Windows: discovery パラメーター

discovery パラメーターは、データベース・インスタンスのディスカバーと現在アクティブな S-TAP への結果の送信を行うオートディスカバリー機能の動作を定義します。

これらのパラメーターは、S-TAP プロパティー・ファイルの [TAP] セクションに格納されています。

注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。

GIM	guard_tap.ini	デフォルト値	記述
WINSTAP_DISCOVERY_INTERVAL	DISCOVERY_INTERVAL	24	オートディスカバリーが実行される時間間隔 (時間)。使用不可にするには 0 に設定します。

親トピック: Windows: S-TAP 構成パラメーターの編集

Windows: デバッグ・パラメーター

これらのパラメーターは、S-TAP デバッグの動作に影響を与えます。

注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。
これらのパラメーターは、S-TAP プロパティー・ファイルの [DEBUG_OPTIONS] セクションに格納されています。

guard_tap.ini	デフォルト値	記述
DEBUG_BUFFER	1	1=ローカル・パケットの内容をログに記録します
DEBUG_FIREWALL	1	1=ファイアウォール・イベントをログに記録します

これらのパラメーターは、S-TAP プロパティー・ファイルの [TAP] セクションに格納されています。

表 1. デバッグ用の追加の S-TAP 構成パラメーター

guard_tap.ini	デフォルト値	記述
DEBUG_MAX_FILE_SIZE	200	

guard_tap.ini	デフォルト値	記述
DEBUGLEVEL	0	格納するデバッグ・メッセージのレベル。IBM 技術サポートの指示がない限り、0 のままにしてください。 0 クリティカル・エラー情報のみ v10.1.4 以降: 2 つの「始動」デバッグ・ログが bin¥..¥logs に保存されます。ファイル名の構文: startup_hostname_timestamp.new および startup_hostname_timestamp.old。bin¥..¥logs のファイルは、upload_feature がオンの場合は自動的にアップロードされます。 1 前のすべてのメッセージ、および反復可能なクリティカル・エラー情報 v10.1.4 以降: 2 つの「通常」デバッグ・ログが bin¥StapBuffer に保存されます。ファイル名の構文: stap_hostname_timestamp.new および stap_hostname_timestamp.old。bin¥StapBuffer のファイルはアップロードされません。 2 使用されていない 3 レベル 1 のすべてのメッセージ、および Guardium システムに送信されたパケットの要約情報 4 レベル 3 のすべてのメッセージ、およびローカル・スニッフィング・ログ 5 レベル 4 のすべてのメッセージ、およびネットワーク・スニッフィング・ログ 6 レベル 5 のすべてのメッセージ、およびハートビート受信ログ 7 レベル 6 のすべてのメッセージ、および各種デバッグ情報
DUMP_FILE_MODE	0	S-TAP が異常終了した場合に、ダンプ・ファイルの取り込みを有効にします。 パラメーターがゼロではないときは、S-TAP が始動するたびに新しいダンプ・ファイルが開かれます。異常終了が発生していない場合は空になります。 • 0: クラッシュ・ダンプは生成されません • 1: クラッシュ・ダンプが生成され、ファイル stap.diag に書き込まれます。このファイルは、S-TAP の作業ディレクトリーに作成されます。 S-TAP は stap.diag ファイルを上書きする前に、既存の stap.diag ファイルをすべてバックアップ・ファイルにコピーします。 • 2: タイム・スタンプ付きのクラッシュ・ダンプが生成され、ファイル stap-TIMESTAMP.diag に書き込まれます。このファイルは、S-TAP の作業ディレクトリーに作成されます。ここで、TIMESTAMP は、クラッシュ・ダンプが生成されたタイミングを示します。 異常終了で問題がある場合は、このオプションを使用して最新のダンプのみではなく、すべてのダンプを取り込んでください。 タイム・スタンプもデバッグに役立ちます。ただし、このオプションでは、使用するディスク・スペースが多くなります。
DEBUG_FILE_MODE		S-TAP デバッグ・ファイルのロケーション。は、デフォルトは <install folder>/StapBuffer/stap.txt でした。 v10.1.4 以上: debuglevel > 0 の場合、以前の S-TAP セッション (存在する場合) からのログは %STAP_DIR%¥Bin¥StapBuffer¥stap_%HOSTNAME%%YY-MM-DD%%HHMMDD%.old として保存され、新しいログは %STAP_DIR%¥Bin¥StapBuffer¥stap_%HOSTNAME%%YY-MM-DD%%HHMMDD%.new として作成されます。 また、S-TAP の始動に関連するメッセージのみが格納された始動ログは常に、%STAP_DIR%¥Logs に startup_%HOSTNAME%%YY-MM-DD%%HHMMDD%.old および startup_%HOSTNAME%%YY-MM-DD%%HHMMDD%.new として生成されます。
STACK_TRACE_FILE_MODE		dump_file_mode と同様です。
KERNEL_DEBUG_LEVEL	0	
SYSLOG_MESSAGES	1	1 = EventViewer にメッセージを送信します。0=メッセージは送信されません。
WER_DUMP	1	
WER_DUMP_FOLDER	なし	パラメーターが設定されていない場合、以下の値が使用されます。STAP インストール・フォルダーが「C:¥Program Files (x86)¥...」以外のどこかのルート・フォルダーである場合、WER ダンプ・フォルダーは、末尾に「...¥Windows S-TAP¥Bin¥..¥Logs」がある絶対パスに設定されます。STAP インストール・フォルダー名にテキスト「(x86)」が含まれている場合、ダンプ・フォルダーは「C:¥Guardium¥Dumps」に設定され、STAP プロセスでパスが作成されます。 例えば、Windows S-TAP が C:¥PROGRAM FILES¥IBM¥WINDOWS S-TAP にインストールされ、WER_DUMP_FOLDER と WER_DUMP_COUNT のデフォルト値を使用する場合、Windows S-TAP は以下のレジストリー設定を使用します。Windows S-TAP がクラッシュした場合は、Windows S-TAP クラッシュ・ダンプが Windows Error Reporting (WER) 機能を介して生成されます。 HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Windows¥Windows Error Reporting¥LocalDumps¥guardium_stapr.exe DumpCount REG_DWORD 0x1 DumpFolder REG_EXPAND_SZ C:¥PROGRAM FILES¥IBM¥WINDOWS S-TAP¥Bin¥..¥LOGS¥ DumpType REG_DWORD 0x2
WER_DUMP_COUNT	1	最大値は 5 です。

親トピック: Windows: S-TAP 構成パラメーターの編集

Windows: 構成監査システム (CAS) パラメーター

これらのパラメーターは、CAS の動作に影響を与えます。

GUI	guard_tap.ini	デフォルト値	記述
	CAS_SERVER_PORT	16017	CAS エージェントとの通信用のポート。非暗号化の場合は 16017、暗号化の場合は 16019。

親トピック: Windows: S-TAP 構成パラメーターの編集

Windows: ドライバー・パラメーター

これらのパラメーターは、S-TAP が対話するいくつかのドライバーの動作に影響を与えます。

注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。

guard_tap.ini	デフォルト値	記述
WFP_DRIVER_INSTALLED	1	v10.5 で非推奨になっています。
TCP_DRIVER_INSTALLED	1	TCP ドライバーを使用します。
ORA_DRIVER_INSTALLED	1	Oracle ASO および SSL トラフィックをスニッフィングするには 1 に設定します。
ORA_DRIVER_LEVEL	0	高。スレッドの優先順位付けに使用されます。
NAMED_PIPES_DRIVER_INSTALLED	1	ローカルの名前付きパイプをスニッフィングするには 1 に設定します。
NAMED_PIPES_DRIVER_LEVEL	0	高。スレッドの優先順位付けに使用されます。
SHARED_MEMORY_DRIVER_LEVEL	0	高。スレッドの優先順位付けに使用されます。
KRB_MSSQL_DRIVER_INSTALLED	2	v10.1.4 から非推奨になっています。guard_tap.ini ファイルには出現しますが、構成には影響しません。 このパラメーターは、MSSQL SSL および Kerberos の暗号化トラフィックの暗号化解除に使用されます。MSSQL 暗号化トラフィックおよび Kerberos チケットを収集するには、1 または 2 に設定します。1 に設定される場合、STAP の始動時に、SID と相関関係のあるユーザー名を事前収集します。krb_mssql_driver_user_collect_time に定義されている秒数にわたって収集します。2 に設定される場合、事前収集は行われず、ユーザー名の相関は実行時に行われます。
KRB_MSSQL_DRIVER_LEVEL	0	このパラメーターは v10.1.4 から非推奨になっています。
KRB_MSSQL_DRIVER_NONBLOCKING	0	このパラメーターは v10.1.4 から非推奨になっています。guard_tap.ini ファイルには出現しますが、構成には影響しません。
KRB_MSSQL_DRIVER_USER_COLLECT_TIME	30	このパラメーターは v10.1.4 から非推奨になっています。10.1 で導入された相関ドライバーを使用してください。
CORRELATION_TIMEOUT	5	WFP スニファーおよび NMP スニファーが相関の発生を待機する秒数。この秒数が過ぎると、断念してアプライアンスへのトラフィックのフローを再開します。デフォルトは 5 秒です。

親トピック: Windows: S-TAP 構成パラメーターの編集

Windows: S-TAP の操作とパフォーマンス

- Windows: GIM を使用した S-TAP の停止
  GIM を使用すると、データベース・サーバーにログインせずに S-TAP を停止できます。
- Windows: GIM を使用した S-TAP の始動
  GIM を使用すると、データベース・サーバーにログインせずに S-TAP を始動できます。
- Windows: GIM を使用しない S-TAP の始動
  データベース・サーバーから S-TAP を開始する方法について説明します。
- Windows: GIM を使用しない S-TAP の停止
  データベース・サーバーから S-TAP を停止する方法について説明します。
- Windows: GUI からの S-TAP のモニター
  次の標準レポートとビューを使用して、GUI で STAP 状況をモニターします。
- Windows: S-TAP の統計
  S -TAP の統計は、コレクター上のデータベース表 STAP_Statistic に格納されます。この表は、S-TAP から送信された統計をスニファーに格納します。
- Windows: Guardium Agent Monitor によるモニター
  Guardium Agent Monitor (GAM) プロセスは、Guardium エージェントのパフォーマンスと反応性をモニターします。これは、トラブルシューティング中の詳細分析に有効です。
- Windows: S-TAP の問題のトラブルシューティング
  「システム・ビュー」の「S-TAP 状況モニター」タブを使用して、問題の調査を行うことができます。 他のツールを使用しなければならない場合があります。特に、検査エンジンを検査できないデータベースをモニターしている場合です。

親トピック: Windows: S-TAP ユーザーズ・ガイド

Windows: GIM を使用した S-TAP の停止

GIM を使用すると、データベース・サーバーにログインせずに S-TAP を停止できます。

このタスクについて

次のステップを使用して、WINSTAP_ENABLED パラメーターを変更し、データベース・サーバー上での S-TAP の始動をスケジュールします。

手順

1. 「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」にナビゲートします。
2. 「クライアントの選択」セクションで、S-TAP を停止するデータベース・サーバーを選択します。表内のチェック・ボックスを使用して個々のクライアントを選択するか、「クライアント・グループを選択してください」メニューを使用してクライアントのグループを選択します。 「次へ」をクリックして先に進みます。
3. 「バンドルの選択」セクションで、S-TAP バンドルを選択します。「次へ」をクリックします。 ソフトウェア・バンドルを選択すると、「選択されたバンドルのアクション」列に、各クライアントに対して実行されるアクションが示されます。状況が「更新」のパラメーターがある S-TAP を停止できます。
4. 「パラメーターの選択」セクションで、`WINSTAP_ENABLED` を入力し、値 0 を入力します。「次へ」をクリックします。
5. 「クライアントの構成」セクションで、表を使用して、加える変更を検討します。
6. 「インストール」をクリックします。
7. 「OK」をクリックして S-TAP を今すぐ停止するか、 アイコンを使用して、停止時刻をスケジュールし、「OK」をクリックします。

親トピック: Windows: S-TAP の操作とパフォーマンス

Windows: GIM を使用した S-TAP の始動

GIM を使用すると、データベース・サーバーにログインせずに S-TAP を始動できます。

このタスクについて

次のステップを使用して、WINSTAP_ENABLED パラメーターを変更し、データベース・サーバー上での S-TAP の始動をスケジュールします。

手順

1. 「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」にナビゲートします。
2. 「クライアントの選択」セクションで、S-TAP を始動するデータベース・サーバーを選択します。表内のチェック・ボックスを使用して個々のクライアントを選択するか、「クライアント・グループを選択してください」メニューを使用してクライアントのグループを選択します。 「次へ」をクリックして先に進みます。
3. 「バンドルの選択」セクションで、S-TAP バンドルを選択します。「次へ」をクリックします。 ソフトウェア・バンドルを選択すると、「選択されたバンドルのアクション」列に、各クライアントに対して実行されるアクションが示されます。状況が「更新」のパラメーターがある S-TAP を開始できます。
4. 「パラメーターの選択」セクションで、`WINSTAP_ENABLED` を入力し、値 `1` を入力します。「次へ」をクリックします。
5. 「クライアントの構成」セクションで、表を使用して、加える変更を検討します。
6. 「インストール」をクリックします。
7. 「OK」をクリックして S-TAP を今すぐ始動するか、 アイコンを使用して、開始時刻をスケジュールし、「OK」をクリックします。

親トピック: Windows: S-TAP の操作とパフォーマンス

Windows: GIM を使用しない S-TAP の始動

データベース・サーバーから S-TAP を開始する方法について説明します。

このタスクについて

注: Windows S-TAP の始動中に構成の問題 (不明なローカル IP アドレス、複数の 1 次 SQL-Guard が定義されているなど) による致命的エラーが発生したときは、Windows イベント・ログにその理由が記録されます。場合によっては、障害後の終了によって異常終了したり、別のイベントがログに 記録されたりする可能性があります。障害の原因を説明するイベントの後でこの異常終了が 起こる場合は、心配ありません。

手順

1. システム管理者のアカウントを使用して、データベース・サーバー・システムにログオンします。
2. 「サービス」制御パネルで、IBM Security Guardium S-TAP を始動します。
3. この S-TAP の報告先となっている Guardium システムにログインします。 S-TAP 制御パネルの状況ランプがグリーンになっていることを確認します。

親トピック: Windows: S-TAP の操作とパフォーマンス

Windows: GIM を使用しない S-TAP の停止

データベース・サーバーから S-TAP を停止する方法について説明します。

手順

1. システム管理者のアカウントを使用して、データベース・サーバー・システムにログオンします。
2. 「サービス」制御パネルで、IBM Security Guardium S-TAP を停止します。
3. この S-TAP のレポート先となった Guardium システムの UI にログインし、S-TAP 制御パネルの「状況」ライトが赤色であることを確認します。

親トピック: Windows: S-TAP の操作とパフォーマンス

Windows: GUI からの S-TAP のモニター

次の標準レポートとビューを使用して、GUI で STAP 状況をモニターします。

S-TAP によって作成された例外に基づいてアラートを作成できますが、S-TAP レポートが使用する他のドメインは、システム専用であり、ユーザーはアクセスできません。

システム・ビュー

「システム・モニター」ウィンドウの**S-TAP 状況モニター**: このレポートは、この Guardium システムへの各 S-TAP レポートについて、S-TAP ホスト、S-TAP バージョン、データベース・サーバー・タイプ、状況 (アクティブまたは非アクティブ)、最後に受信した応答 (日時)、インスタンス名、1 次ホスト名、および (MS SQL サーバー共有メモリー、DB2® 共有メモリー、Win TCP、ローカル TCP モニター、名前付きパイプの使用、暗号化、ファイアウォール、データベース・インストール・ディレクトリー、DB ポート (最小)、および DB ポート (最大) の) true/false インジケーターを示します。

行をクリックすると、この S-TAP 用に構成されている検査エンジンが表示されます。階層リンクを参照すると、現在位置が分かります。「すべての S-TAP」をクリックして、S-TAP のリストに戻ります。詳しくは、Windows: 検査エンジンの検査を参照してください。

注: Db2 共有メモリー・ドライバーは Db2 Tap フィーチャーに置き換えられました。

S-TAP 状況モニター: このレポートは、この Guardium システムへの各 S-TAP レポートについて、S-TAP ホスト、データベース・サーバー・タイプ、S-TAP バージョン、状況 (アクティブまたは非アクティブ)、検査エンジンの状況、最後に受信した応答 (日時)、1 次ホスト名、および (ファイアウォールおよび暗号化の) true/false インジケーターを示します。すべての検査エンジンの検査状況を表示するには、「S-TAP 状況」と「検査エンジンの状況」をクリックします。

S-TAP イベント: このレポートは、この Guardium システムへの各 S-TAP レポートについて、S-TAP ホスト、タイム・スタンプ、イベント・タイプ (成功、エラー・タイプなど)、およびタップ・メッセージを示します。

「S-TAP イベント」パネルにメッセージが表示されない場合は、その S-TAP® の構成ファイル内でイベント・メッセージの生成が無効になっている可能性があります。その場合は、ホスト・システム上のイベント・ログ内に S-TAP イベント・メッセージが出力されている場合があります。

TAP モニター

プライマリー Guardium® ホスト変更ログ: S-TAP の 1 次ホスト変更のログ。1 次ホストとは、 S-TAP がデータを送信する Guardium システムです。このレポートの各行には、S-TAP ホスト、Guardium ホスト名、期間の開始、期間の終了がリストされます。

S-TAP 状況: 各 S-TAP ホストで定義されている各検査エンジンについて、状況情報を表示します。 このレポートは現在の状況を報告するため、開始日付と終了日付のパラメーターはありません。このレポートの各行は、S-TAP ホスト、データベース・サーバー・タイプ、状況、最後の応答、1 次ホスト名、および属性 (共有メモリー・ドライバー (インストール済み)、Db2 共有メモリー・ドライバー (インストール済み)、名前付きパイプ・ドライバー (インストール済み)、およびアプリケーション・サーバー (インストール済み)) の Yes/No インディケーターをリストします。さらに、ハンター DBS をリストします。

非アクティブな S-TAP: システムで定義されている非アクティブな S-TAP をすべてリストします。これには 1 つだけ、QUERY_FROM_DATE というランタイム・パラメーターがあり、デフォルトでは now -1 hour に設定されています。このパラメーターを使用して、*非アクティブ*をどのように定義するのかを制御します。このレポートには、S-TAP 状況レポートと同じデータの列が含まれており、レポートの各行のカウントが追加されています。

親トピック: Windows: S-TAP の操作とパフォーマンス

Windows: S-TAP の統計

S -TAP の統計は、コレクター上のデータベース表 STAP_Statistic に格納されます。この表は、S-TAP から送信された統計をスニファーに格納します。

アクセスするには、GUI を使用します。結果に基づいてアラートを作成できます。S-TAP が S-TAP に関する統計情報をスニファーに送信する間隔は、guard_tap.ini パラメーター STAP_STATISTIC によって制御されます。有効な値:

- 正の整数は時間を表します
- 負の整数は分を表します
- 0=送信しない

デフォルトは -1 (毎分) です。

S-TAP 統計表のフィールド:

- TIMESTAMP
- SOFTWARE_TAP_HOST
- TOTAL_BYTES_SO_FAR
- BUFFER_RECYCLED
- STAP_BUFFER_USAGE_PERCENT

親トピック: Windows: S-TAP の操作とパフォーマンス

Windows: Guardium Agent Monitor によるモニター

Guardium Agent Monitor (GAM) プロセスは、Guardium エージェントのパフォーマンスと反応性をモニターします。これは、トラブルシューティング中の詳細分析に有効です。

注: GAM サービスはインストールされている環境に固有の構成を必要とするため、デフォルトではオフにする必要があります。不適切な構成は、操作上の重大な問題を発生させる可能性があります。これはトラブルシューティングに役立つツールであり、それ以外の場合は不要です。

モニター対象は次のとおりです。

- CPU 使用率
- メモリー
- 処理
- スレッドの数
- アライブ - 反応性 (サポートされるエージェントのみ。現在は S-TAP が唯一のサポート対象エージェント)(反応性を参照)

モニター対象エージェントが構成されたしきい値を超える場合、あるいはコンソール要求に応答しない場合、次のアクションが任意の組み合わせで実行される可能性があります。

- 自動的に diag.bat を実行
- サービスの自動停止/再始動
- コア・ダンプの自動実行

Guardium Agent Monitor は S-TAP のインストール時にインストールされますが、デフォルトでは有効になりません。S-TAP をアンインストールすると、GAM はアンインストールされます。

注: S-TAP と同様に、GAM には管理特権が必要です。インストール時には、「管理者として実行」を使用して管理ユーザーとして実行します。

GAM のデフォルトのインストール・ロケーションは、S-TAP の親フォルダーです (C:¥Program Files¥IBM¥Guardium Agent Monitor¥)。

GAM 出力のデフォルト・ロケーションは ¥Bin¥ サブフォルダーです。

GAM を有効にしたら、そのプロセスがデータベース・サーバーで実行されていることを確認してください (resmon.exe)。

GAM 構成

Guardium Agent Monitor は、その構成ファイル resmon.ini を引数として実行します。このモニターは、resmon.ini ファイルを使用して制御されます。サンプルの resmon を参照してください。サンプル ini の末尾に、すべてのパラメーターのデフォルト値が記載されています。

グローバル構成

NUMBER_OF_SERVICES: モニターされるサービスの数

UPDATE_INTERVAL: 各ポーリング・メトリック間のインターバルの長さ (秒単位)

DEBUG: 1 は GAM デバッグ・ログを有効にし、0 はログを無効にします

NUMBER_BYTES_IN_LOG: GAM ログの最大 KB 数

CPU しきい値構成

CPU_LOAD_LIMIT: アクションが実行されるか、UPDATE_INTERVAL がしきい値に到達したオカレンスのカウントを開始する CPU パーセンテージのしきい値

CPU_INTERVALS_ALLOWED: アクションがトリガーされるまでに、CPU がしきい値を超えることができるインターバルの数 (時間制限を設定するために UPDATE_INTERVAL と一緒に使用)

UPDATE_INTERVAL: 0 = CPU がロード制限に到達したときにアクションが実行されます。1 = CPU が、CPU_INTERVALS_ALLOWED によって指定した回数、ロード制限に到達したときにアクションが実行されます。

CPUAVE: CPU 平均のタイプを定義します。1 = すべての CPU コアの平均使用量 (システム平均)、0 = プロセスによって使用されたコアのパーセンテージ。

メモリー使用量、ハンドル数、およびスレッド数のしきい値構成

これらのメトリックには、制限とピーク制限の 2 つのしきい値があります。許可されるより多くの間隔で制限しきい値を通過したか、ピーク制限しきい値を通過したときに、アクションがトリガーされます。メトリックとは、CPU、メモリーなどを指します。

[METRIC]_LIMIT: 下位しきい値。 [METRIC]_INTERVALS_ALLOWED より多くの間隔でこの制限を超えると、アクションがトリガーされます

[METRIC]_INTERVALS_ALLOWED: アクションがトリガーされるまでに、下限しきい値で許可されるインターバルの数 (時間制限のための UPDATE_INTERVAL と一緒に使用)

[METRIC]_PEAK_LIMIT: 上位しきい値。このしきい値を 1 回超えると、アクションがトリガーされます

注: [METRIC]_INTERVALS_ALLOWED は、しきい値の制限時間を設定するために UPDATE_INTERVAL と一緒に使用されます。(例えば UPDATE_INTERVAL=1、CPU_INTERVALS_ALLOWED=10、CPU_LOAD_LIMIT=10 は、CPU ロードが 10 秒以上 10 %を超える場合にアクションがトリガーされることを意味します)。

反応性

NAMEDPIPE_INTERVAL: S-TAP エージェントの反応性を確認するために ping が実行される間隔 (秒単位)。無効にするには「0」に設定します。

アクション構成

トリガーできるアクションは、「コア・ダンプ構成」および「診断構成」で説明されています。2 番目と 3 番目のアクションは、前のアクションの ACTION_RESET_INTERVAL 内にトリガーされた場合にのみ開始されます。新しいトリガーがないまま ACTION_RESET_INTERVAL 時間が経過した場合は、次のトリガーによって FIRST_ACTION で新しいサイクルが開始されます。

FIRST_ACTION: 0 = アクションなし。1 = サービスを停止して再開します。2 = サービスを停止します。

SECOND_ACTION: ACTION_RESET_INTERVAL 中に 2 回目のトリガーがあったときにアクションが開始されます。0 = アクションなし。1 = サービスを停止して再開します。2 = サービスを停止します。

THIRD_ACTION: ACTION_RESET_INTERVAL 中に 3 回目のトリガーがあったときにアクションが開始されます。0 = アクションなし。1 = サービスを停止して再開します。2 = サービスを停止します。

ACTION_RESET_INTERVALS: アクションをリセットするまでの秒数。

コア・ダンプ構成

コア・ダンプは、アクションがトリガーされるたびに取ることができます。

ACTION: 1 = アクションがトリガーされるときに必ずコア・ダンプを取ります。0 = コア・ダンプは取られません。

MAX_NUM_DUMP: ダンプ・ディレクトリーに保管されるコア・ダンプの最大数 (最新のものを保持)。

MDTIMEOUT: コア・ダンプのタイムアウト時間 (ミリ秒単位)

診断構成

アクションがトリガーされるたびに診断ファイルを実行できます。サービスの実行可能ファイル・パスと同じフォルダーにある diag.bet 診断スクリプトは、DIAG_PARAMETER パラメーターを使用して実行されます。

DIAGACTION: 1 = アクションがトリガーされるときに必ず診断スクリプトを実行します。0 = 診断スクリプトは実行されません。

DIAGNAME: 実行する診断ファイルの名前 (サービス実行可能ファイルと同じフォルダー内になければなりません)

DIAG_PARAMETER: 診断ファイルの実行時に使用されるパラメーター

resmon.ini の例

```
;行の先頭にあるセミコロンはコメントを示します
;
[Global]
NUMBER_OF_SERVICES=1
;
;しきい値を確認するインターバル (秒)
UPDATE_INTERVAL=1
;
;モニター・ログの有効化
DEBUG=1
;
;「0」はアクションのミニダンプを取らないことを意味します。「1」はミニダンプを取得します
ACTION=1
;
;ダンプ・ディレクトリーに保管されるダンプの最大数
MAX_NUM_DUMP=3
;
;平均 CPU 時間。「0」は 1 つのコアに対するパーセンテージ、「1」はシステム内のすべてのコアの平均パーセンテージです。
CPUAVE=1
;
;ミニダンプのタイムアウト (ミリ秒単位)
MDTIMEOUT=1000
;モニター・ログの最大バイト数 (KB 単位)
NUMBER_BYTES_IN_LOG=200
;
;サービスの構成
[Service1]
Name=GUARDIUM_STAP
;
;生存を確認するインターバル (サポートされるエージェントのみ)。無効にするには「0」に設定します。
NAMEDPIPE_INTERVAL=30
;
;アクションへの診断の実行。有効にするには「1」に設定します。
DIAGACTION=0
;
;診断ファイル名
DIAGNAME=diag.bat
;
;診断パラメーター。パラメーターにスペースが含まれている場合は、引用符で囲む必要があります。
DIAG_PARAMETER=
;
;CPU 制限のパーセンテージ
CPU_LOAD_LIMIT=10
;
;CPU_LOAD_LIMIT が許容される最大連続インターバル数
CPU_INTERVALS_ALLOWED=10
;
;メモリー制限 (KB)
MEM_USAGE_LIMIT=150000
MEM_USAGE_PEAK_LIMIT=200000
MEM_USAGE_INTERVALS_ALLOWED=30
;
;ハンドル制限
HANDLE_COUNT_LIMIT=500
HANDLE_COUNT_PEAK_LIMIT=1000
HANDLE_COUNT_INTERVALS_ALLOWED=20
;
;スレッド制限
THREAD_COUNT_LIMIT=200
THREAD_COUNT_PEAK_LIMIT=300
THREAD_COUNT_INTERVALS_ALLOWED=20
;
;「1」はアクションを実行し、サービスを再始動します。
;「2」はアクションを実行し、サービスを停止して始動しません。
FIRST_ACTION=1
SECOND_ACTION=1
THIRD_ACTION=2
```

```
;
;リセット間隔 (秒単位)
ACTION_RESET_INTERVALS=60
```

親トピック: Windows: S-TAP の操作とパフォーマンス

Windows: S-TAP の問題のトラブルシューティング

「システム・ビュー」の「S-TAP 状況モニター」タブを使用して、問題の調査を行うことができます。 他のツールを使用しなければならない場合があります。特に、検査エンジンを検査できないデータベースをモニターしている場合です。

S-TAP が Guardium システムに接続されていない場合
IBM Security Guardium S-TAP サービスがデータベース・サーバーで実行されているかどうかを確認します。
IBM Security Guardium S-TAP サービスを調べて、実行中であることを確認します。
S-TAP のバージョンを調べる方法

- GUI の「管理」 > 「システム・ビュー」 > 「S-TAP 状況モニター」に、S-TAP® バージョン番号が表示されます。
- 別の方法として、データベース・サーバーのコマンド行で、S-TAP バージョン番号を表示できます。

コマンド行からデバッグを実行して、構成の問題を迅速に特定
GIM GUI またはコマンド行でデバッグをオンにします。Windows: デバッグ・パラメーターで、デバッグ・レベルを参照してください。
データベース・サーバーと Guardium システムの間の接続を確認します。

- データベース・サーバーから *sqlguard_ip* で Guardium システムを ping できることを確認します。
- ping が成功した場合は、Guardium システム上のポート 16016/16018 に Telnet でログインできることを確認します。

データベース・サーバーと Guardium システムの間にファイアウォールがある場合
これら 2 つのシステムの間でのトラフィック用に、TCP ポート 16016 または TLS ポート 16018 (暗号化接続の場合) が開かれていることを確認します。
注: ポートが使用可能かどうかを確認するには、コマンド `nmap -p port guardium_hostname_or_ip` を使用します。
sqlguard_ip パラメーターが、接続先の Guardium システムの正しい *guardium_hostname_or_ip* に設定されていることを確認します。

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」をクリックして、「S-TAP 制御」を開きます。
2. データベース・サーバーに対応する IP アドレスの S-TAP ホストを見つけます。
3. 「Guardium ホスト」サブセクションを展開して、アクティブな Guardium ホストが正しく構成されていることを確認します。
4. 必要に応じて、「変更」をクリックして、Guardium ホストを更新します。

デバッグ・ファイルはどこにありますか?
debuglevel > 0 の場合、以前の S-TAP セッション (存在する場合) からのログは %STAP_DIR%¥Bin¥StapBuffer¥stap_%HOSTNAME%%YY-MM-DD%%HHMMDD%.old として保存され、新しいログは %STAP_DIR%¥Bin¥StapBuffer¥stap_%HOSTNAME%%YY-MM-DD%%HHMMDD%.new として作成されます。
また、S-TAP の始動に関連するメッセージのみが格納された始動ログは常に、%STAP_DIR%¥Logs に startup_%HOSTNAME%%YY-MM-DD%%HHMMDD%.old および startup_%HOSTNAME%%YY-MM-DD%%HHMMDD%.new として生成されます。
トラフィックに重大なスパイクが発生して、トラフィックがドロップされる
この症状の原因としてバッファー・オーバーフローが考えられます。デバッグ・ログを調べて、バッファー・オーバーフローを示すメッセージがないか確認してください。上級者のみ: 動的バッファー機能を有効にすることを検討してください。Windows: 一般パラメーターの dynamic_buffer_increase を参照してください。
S-TAP プロセスが繰り返し再始動していないことを確認します。
データベース・サーバー上で、コマンド `ps -eaf | grep stap` を実行して、S-TAP のプロセスが変更されていないことを確認します。
S-TAP 承認がオンになっていないことを確認します。
S-TAP 承認がオンになっていると、Guardium システムに接続されている新規 S-TAP は、すべて拒否されます。

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 認証」をクリックして、「S-TAP 認証」を開きます。
2. 「S-TAP 承認が必要」チェック・ボックスを調べます。このボックスにチェック・マークが付いている場合、新規 S-TAP がこの Guardium システムに接続できるのは、承認済み S-TAP のリストに追加されてからになります。
3. S-TAP 承認がオンになっている場合は、「日次モニター」 > 「承認された Tap クライアント」を選択して、承認済み S-TAP のリストを表示します。 調査対象の S-TAP がこのリストにない場合は、「S-TAP 認証」ペインに戻り、「クライアント・ホスト」フィールドに S-TAP の IP アドレスを入力して、「追加」をクリックします。

S-TAP 検査の問題

検査プロセスは、間違ったユーザー ID とパスワードを使用してデータベースの STAP クライアントへのログインを試行することで、この試行が認識され、Guardium システムに通知されることを確認します。要求が行われた Guardium システムに検査エンジンのメッセージが到達しないように、S-TAP が構成されている可能性があります。

このような構成の詳細には、以下が含まれます。

- ロード・バランシング: 複数の Guardium システムに応答を返すように S-TAP が構成されている場合は、エラー・メッセージをさまざまな Guardium システムに送信できます。
- フェイルオーバー: 2 次 Guardium システムが S-TAP 用に構成されていると、1 次 Guardium システムがビジー状態である場合に、エラー・メッセージを 2 次 Guardium システムに送信できます。
- Db_ignore_response: データベースからのすべての応答を無視するように S-TAP が構成されている場合、エラー・メッセージは Guardium システムに送信されません。
- クライアント IP/マスク: 0.0.0.0 ではないマスクを定義する場合、エラー・メッセージが送信されない可能性があります。
- 除外 IP/マスク: 0.0.0.0 ではないマスクを定義する場合、エラー・メッセージが送信されない可能性があります。

関連トピック:

- Windows: GUI からの S-TAP のモニター
- Windows: Guardium Agent Monitor によるモニター
- Windows: 検査エンジンの検査

親トピック: Windows: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: S-TAP ユーザーズ・ガイド

Guardium S-TAP は、データベース・サーバーおよびファイル・サーバーにインストールされる軽量のソフトウェア・エージェントです。S-TAP によって収集される情報は、Guardium のトラフィック・レポート、アラート、可視化など、すべての基礎となります。このセクションでは、Linux、Solaris、AIX、および HP-UX の各サーバーでの S-TAP について説明します。

データ・アクティビティーのモニターでは、S-TAP は、クライアントとデータベースの間のアクティビティーをモニターして、その情報を Guardium コレクターに転送します。データベース・トラフィックは、セキュリティー・ポリシーで指定されている基準に基づいてコレクターに記録されます。また、信頼できる接続を無視したり、特定の IP からのトラフィックを無視したりすることで、最初にコレクターに送信されるトラフィックの量を減らすこともできます。

ファイル・アクティビティーのモニターでは、データ・アクティビティーとは異なり、ポリシー・ルールがファイル・サーバーにプッシュダウンされるため、セキュリティー・ポリシーで指定されているデータのみがコレクターに転送されます。

S-TAP はブート時に S-TAP カーネル・コンポーネントのアップグレードを処理して、Linux 環境のカーネル・アップグレードに対応します。

- Linux システムおよび UNIX システム: S-TAP の機能
  UNIX システムでの S-TAP のインストールを開始する前に、次の概念を把握しておいてください。
- Linux システムおよび UNIX システム: S-TAP のインストール、アップグレード、およびアンインストール
  S-TAP をインストール、アップグレード、およびアンインストールするにはいくつかの方法があります。それぞれについて確認し、どれが最適かを判断してください。
- Linux システムおよび UNIX システム: S-TAP の構成
  S-TAP の構成について説明します。
- Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: S-TAP の機能

UNIX システムでの S-TAP のインストールを開始する前に、次の概念を把握しておいてください。

- Linux システムおよび UNIX システム: S-TAP モニター・メカニズムのサポート・マトリックス
  モニターまたはブロックするデータに応じて、使用する S-TAP セットアップを選択します。以下の表を使用して、オペレーティング・システムおよびデータベースごとに、必要な操作を実行可能なモニター・メカニズム (出口ライブラリー、K-TAP、A-TAP) を判別してください。
- Linux システムおよび UNIX システム: Linux、Solaris、AIX、および HP-UX の S-TAP のモニター・メカニズム
  Guardium UNIX S-TAP は、データベース・トラフィックを収集するためにいくつかの異なるモニター・メカニズムを使用します。構成時に、要件に最も適した方式を選択できます。すべてのメカニズムは、トラフィックをフィルターに掛けて、ネットワークのオーバーヘッドを軽減し、パフォーマンスを向上させます。
- Linux システムおよび UNIX システム: S-TAP からコレクターへの暗号化
  ネットワークを介して暗号化 (TLS) 方式でコレクターと通信するように S-TAP エージェントを構成できます。
- Linux システムおよび UNIX システム: UID チェーン
  UID チェーンは、それを使用することで、S-TAP が (K-TAP を介して)、データベース接続前に発生したユーザーのチェーンをトラッキングできるメカニズムです。それは、Solaris ゾーン、AIX WPAR、Solaris 8/9、Solaris 11 SPARC でサポートされています。
- Linux システムおよび UNIX システム: プロキシー・ファイアウォール
  プロキシー・サーバーから発生するトラフィックのモニター方法について説明します。

親トピック: Linux システムおよび UNIX システム: S-TAP ユーザーズ・ガイド

Linux システムおよび UNIX システム: S-TAP モニター・メカニズムのサポート・マトリックス

モニターまたはブロックするデータに応じて、使用する S-TAP セットアップを選択します。以下の表を使用して、オペレーティング・システムおよびデータベースごとに、必要な操作を実行可能なモニター・メカニズム (出口ライブラリー、K-TAP、A-TAP) を判別してください。

例えば、以下の項目の 1 つ以上をトラッキングする必要が生じる場合があります。

- ローカル・トラフィックのみ
- ローカル・トラフィックおよびネットワーク・トラフィック
- 共有メモリー
- 暗号化されたデータ
- モニターおよびブロック
- モニターのみ

以下の表では、Guardium のモニター・メカニズムによってサポートされる、最も一般的なプラットフォーム、データベース・タイプ、およびプロトコルを取り上げています。この表は一般ガイドラインを示しています。ここには示されていない他のサポート対象の組み合わせが存在する場合があります。ここに示されているサポート対象のセットアップの一部は、特定の構成に依存する場合があります。特定のニーズに最も適したセットアップを確認するには、お客様サポートにお問い合わせください。空のセルは、その組み合わせがサポートされないことを示しています。

出口ライブラリーは、他のすべてのモニター・メカニズムよりも優先されます。出口ライブラリーを使用できない場合は、次に K-TAP、その次に A-TAP、最後に PCAP が選択されます。

OS	データベース	ネットワーク・トラフィック	ローカル・トラフィック	暗号化されたトラフィック	共有メモリー	Kerberos	ブロッキング	編集	UID チェーン	圧縮	照会再書き込み	インスタンス・ディスカバリー
AIX	Oracle	K-TAP	K-TAP	A-TAP (ASO、SSL)		K-TAP	K-TAP、A-TAP	K-TAP	K-TAP		K-TAP	はい

OS	データベース	ネットワーク・トラフィック	ローカル・トラフィック	暗号化されたトラフィック	共有メモリー	Kerberos	ブロッキング	編集	UID チェーン	圧縮	照会再書き込み	インスタンス・ディスカバリー
AIX	Sybase ASE	K-TAP	K-TAP	A-TAP (SSL)		K-TAP	K-TAP、A-TAP	K-TAP	K-TAP、A-TAP (実際の IP に対して構成されている場合は A-TAP のみ)			はい
AIX	Sybase IQ	K-TAP	K-TAP	A-TAP (ログイン・パケットの暗号化解除のみ。TLS サポートなし)	A-TAP (Sybase 16.1 は DB ユーザー名をサポートしません)		K-TAP、A-TAP	K-TAP	K-TAP			はい
AIX	Db2	Db2 出口、K-TAP	Db2 出口、K-TAP	Db2 出口	Db2 出口、K-TAP	K-TAP	Db2 出口、K-TAP	K-TAP	Db2 出口、K-TAP			はい
AIX	Informix	Informix 出口、K-TAP	Informix 出口、K-TAP	Informix 出口	Informix 出口、K-TAP		Informix 出口、K-TAP	Informix 出口、K-TAP	Informix 出口、K-TAP			はい
HP-UX	Oracle	K-TAP	K-TAP	A-TAP (ASO、SSL)		K-TAP	K-TAP、A-TAP	K-TAP	K-TAP		K-TAP	はい
HP-UX	Sybase ASE	K-TAP	K-TAP	A-TAP (Sybase 15 のみ)			K-TAP、A-TAP	K-TAP	K-TAP、A-TAP (実際の IP に対して構成されている場合は A-TAP のみ)			はい
HP-UX	Sybase IQ	K-TAP	K-TAP				K-TAP	K-TAP	K-TAP			はい
HP-UX	Db2	Db2 出口、K-TAP	Db2 出口、K-TAP	Db2 出口	Db2 出口、K-TAP	K-TAP	Db2 出口、K-TAP	K-TAP	Db2 出口、K-TAP		K-TAP	はい
HP-UX	Informix	Informix 出口、K-TAP	Informix 出口、K-TAP	Informix 出口	Informix 出口、K-TAP		Informix 出口、K-TAP	Informix 出口、K-TAP	Informix 出口、K-TAP			はい
Linux	Db2	Db2 出口、K-TAP		Db2 出口	Db2 出口、A-TAP	K-TAP	Db2 出口、K-TAP、A-TAP (Linux 2.6.36 以降の A-TAP のみ)	K-TAP	Db2 出口、K-TAP		K-TAP	はい
Linux	Informix	Informix 出口、K-TAP	Informix 出口、K-TAP	Informix 出口	Informix 出口、A-TAP		Informix 出口、K-TAP、A-TAP (Linux 2.6.36 以降の A-TAP のみ)	Informix 出口、K-TAP	Informix 出口、K-TAP			はい
Linux	Oracle	K-TAP	K-TAP	A-TAP (ASO、SSL)		K-TAP	K-TAP、A-TAP (Linux 2.6.36 以降の A-TAP のみ)	K-TAP	K-TAP		K-TAP	はい
Linux	Postgres	K-TAP	K-TAP	A-TAP			K-TAP、A-TAP (Linux 2.6.36 以降の A-TAP のみ)	K-TAP	K-TAP、A-TAP (実際の IP に対して構成されている場合は A-TAP のみ)			はい
Linux	Sybase IQ	K-TAP		A-TAP (x86_64 のみ)	A-TAP (Sybase 16.1 は DB ユーザー名をサポートしません)		K-TAP、A-TAP (Linux 2.6.36 以降の A-TAP のみ)	K-TAP	K-TAP、A-TAP (実際の IP に対して構成されている場合は A-TAP のみ)			はい

OS	データベース	ネットワーク・トラフィック	ローカル・トラフィック	暗号化されたトラフィック	共有メモリー	Kerberos	ブロッキング	編集	UID チェーン	圧縮	照会再書き込み	インスタンス・ディスカバリー
Linux	Sybase ASE	K-TAP	K-TAP	A-TAP			K-TAP、A-TAP (Linux 2.6.36 以降の A-TAP のみ)	K-TAP	K-TAP、A-TAP (実際の IP に対して構成されている場合は A-TAP のみ)			はい
Linux	MongoDB	K-TAP	K-TAP	A-TAP			K-TAP、A-TAP (Linux 2.6.36 以降の A-TAP のみ)	K-TAP	K-TAP、A-TAP (実際の IP に対して構成されている場合は A-TAP のみ)			
Linux	Teradata	Teradata 出口、K-TAP		Teradata 出口、A-TAP			Teradata 出口、K-TAP、A-TAP (Linux 2.6.36 以降の ATAP のみ)	K-TAP	K-TAP、A-TAP (実際の IP に対して構成されている場合は A-TAP のみ)			はい
Linux	Netezza	K-TAP					K-TAP	K-TAP	K-TAP			はい
Linux	Cassandra	K-TAP					K-TAP	K-TAP	K-TAP	K-TAP		
Linux	Cassandra / Datastax	暗号化トラフィックのためのネイティブ監査ロギング・サポート。FileAppender に記録するための Cassandra 監査の構成を参照してください。		暗号化トラフィックのためのネイティブ監査ロギング・サポート。FileAppender に記録するための Cassandra 監査の構成を参照してください。								
Linux	SAP HANA	K-TAP					K-TAP	K-TAP	K-TAP			
Linux	MySQL	K-TAP	K-TAP				K-TAP	K-TAP	K-TAP	K-TAP		はい
Linux	MemSQL	K-TAP	K-TAP	K-TAP			K-TAP	K-TAP	K-TAP			はい
Linux	Vertica	K-TAP	K-TAP				K-TAP	K-TAP	K-TAP			
Linux	Hadoop (Cloudera/ Hortonworks)	Cloudera Navigator、Hortonworks および Apache Ranger		Cloudera Navigator、Hortonworks および Apache Ranger			Hortonworks および Apache Ranger					はい
Linux	Greenplum	K-TAP	K-TAP	A-TAP			K-TAP、A-TAP (Linux 2.6.36 以降のみ)	K-TAP	K-TAP			
Linux	MariaDB	K-TAP	K-TAP				K-TAP	K-TAP	K-TAP			
Linux	Aster	K-TAP	K-TAP				K-TAP	K-TAP	K-TAP			
Linux	Couch	K-TAP					K-TAP	K-TAP	K-TAP			
Linux	Hive	K-TAP					K-TAP	K-TAP	K-TAP			
Linux	Accumulo	K-TAP					K-TAP	K-TAP	K-TAP			
Linux	Impala	K-TAP					K-TAP	K-TAP	K-TAP			
Linux	Hue	K-TAP	K-TAP				K-TAP	K-TAP	K-TAP			
Linux	WebHDFS	K-TAP					K-TAP	K-TAP	K-TAP			
Linux	Solr	K-TAP					K-TAP	K-TAP	K-TAP			
Solaris	Oracle	K-TAP	K-TAP	A-TAP (ASO、SSL)		K-TAP	K-TAP、A-TAP	K-TAP	K-TAP		K-TAP	はい

OS	データベース	ネットワーク・トラフィック	ローカル・トラフィック	暗号化されたトラフィック	共有メモリー	Kerberos	ブロッキング	編集	UID チェーン	圧縮	照会再書き込み	インスタンス・ディスカバリー
Solaris	Sybase ASE	K-TAP	K-TAP	A-TAP (Sparc のみ)		K-TAP	K-TAP、A-TAP	K-TAP	K-TAP、A-TAP (実際の IP に対して構成されている場合は A-TAP のみ)			はい
Solaris	Postgres	K-TAP	K-TAP	A-TAP (9.3 以上)			K-TAP、A-TAP	K-TAP	K-TAP、A-TAP (実際の IP に対して構成されている場合は A-TAP のみ)			はい
Solaris	Sybase IQ	K-TAP	K-TAP				K-TAP	K-TAP	K-TAP			はい
Solaris	Db2	Db2 出口、K-TAP	Db2 出口、K-TAP	Db2 出口	Db2 出口、K-TAP	K-TAP	Db2 出口、K-TAP	K-TAP	Db2 出口、K-TAP		K-TAP	はい
Solaris	Informix	Informix 出口、K-TAP	Informix 出口、K-TAP	Informix 出口	Informix 出口、K-TAP		Informix 出口、K-TAP	Informix 出口、K-TAP	Informix 出口、K-TAP			はい

親トピック: Linux システムおよび UNIX システム: S-TAP の機能

Linux システムおよび UNIX システム: Linux、Solaris、AIX、および HP-UX の S-TAP のモニター・メカニズム

Guardium UNIX S-TAP は、データベース・トラフィックを収集するためにいくつかの異なるモニター・メカニズムを使用します。構成時に、要件に最も適した方式を選択できます。すべてのメカニズムは、トラフィックをフィルターに掛けて、ネットワークのオーバーヘッドを軽減し、パフォーマンスを向上させます。

メカニズムは、インストール中に選択します。すべてのメカニズムは、トラフィックをフィルターに掛けて、クライアントとサーバーの IP アドレスの特定のセットに対するデータベース関連トラフィックのみが収集されるようにします。ここでは、メカニズムを優先順位の高い方から出口ライブラリー、K-TAP、A-TAP、PCAP の順に示しています。Linux システムおよび UNIX システム: S-TAP モニター・メカニズムのサポート・マトリックスを参照して、ニーズに合ったメカニズムを選択してください。

出口ライブラリー
: 出口ライブラリーは、推奨のモニター・メカニズムです。最適なパフォーマンスを実現し、暗号化されていても暗号化されていなくても、ローカルとネットワークの両方のトラフィックを処理できます。DB_USER を常にキャプチャーします。 唯一の欠点は、出口ライブラリーは一部のデータベースでしか使用できないことです。
出口ライブラリーには、データベースでの構成が必要です。また、S-TAP バージョンをアップグレードする場合は、出口ライブラリーも更新する必要があります。
出口ライブラリーは、Db2、Informix、および Teradata でのみサポートされます。

K-TAP
: K-TAP は、オペレーティング・システムにインストールされるカーネル・モジュールです。これは、すべてのプロトコルと接続方式 (TCP、TLI、SHM、名前付きパイプなど) をサポートします。有効にすると、これは、データベース・クライアントとデータベース・サーバーとの間の通信に使用するメカニズムに接続することで、データベース・サーバーへのアクセスを監視します。

  Db2 サーバーと Informix サーバーで共有メモリー・トラフィックをキャプチャーするには、Db2 と Informix の出口ライブラリーを K-TAP とともに使用します。A-TAP を使用するよりもこの方法をお勧めします。

  Linux ではカーネルが頻繁に更新され、多くのカーネル・バージョンが存在します。K-TAP バージョンは Linux バージョンに依存します。Linux システムおよび UNIX システム: K-TAP の作成を参照してください。

  K-TAP は、S-TAP のインストール中にインストールされます。K-TAP のインストールに失敗すると、代わりに PCAP がインストールされます。インストール後、構成ファイル設定を使用して、有効または無効にできます。S-TAP のインストール中に K-TAP をロードせずに、後で使用することを決定した場合は、S-TAP を再構成して再始動する必要があります。

A-TAP
: A-TAP (application-level tap) はアプリケーション層に配置されて、暗号化されたデータベース・トラフィックのモニターをサポートします。このモニターは、K-TAP によってカーネルで実行することはできません。A-TAP により、データベース・サーバーの内部コンポーネント間の通信がモニターされます。これは、アプリケーション層の暗号化されていないデータを取り出し、それを K-TAP に送信します。K-TAP は、データを S-TAP に渡すためのプロキシーです。渡されたデータは次に Guardium コレクターに送信されます。

  A-TAP を使用すると、まだ暗号化されているカーネルからデータをキャプチャーする代わりに、元のデータベース・バイナリーの実行前に TAP ライブラリーをロードすることで、Guardium がデータをキャプチャーします。A-TAP ライブラリーは、ノーオペレーション (インターフェースなし) です。ライブラリーは、データが暗号化解除された後、またはデータベースによって暗号化される前に、アプリケーション・モードでデータベースからデータを取得します。このため、暗号化されたトラフィックが Guardium によって収集されるようになったことを除き、データベースの通常の動作には変更はありません。つまり、Oracle コードを実行する前に、Guardium コードを呼び出すスクリプトやツールを更新する必要はありません。

  A-TAP はすべての S-TAP に含まれていますが、モニターするデータベース・インスタンスごとに別々に構成する必要があります。Linux システムおよび UNIX システム: A-TAP の管理を参照してください。

  制約事項:

  - 32 ビット・データベースが 64 ビット・サーバーにある環境では A-TAP がサポートされていません。

- モニター : A-TAP を使用するときは、編集がサポートされません。ブロッキングは、2.6.36 以降のリリースで Linux カーネルでサポートされます。

どのようなときに A-TAP を使用するか

A-TAP は、動作中の DBMS 暗号化が使用されているときに必要になりますが、その他の内部的なデータベース実装の詳細 (A-TAP を必要とする共有メモリーなど) がある場合があります。

Linux での Informix と Db2 は、出口を使用してより緊密に Guardium と統合するため、共有メモリーのサポートに対して推奨される方式です (適用可能な場合)。

PCAP

PCAP は、あるデータベース・サーバーに出入りするネットワーク・トラフィックを listen するパケット取り込みメカニズムです。UNIX 環境では、K-TAP がすべてのネットワーク・トラフィックを取り込むので、PCAP が使用されることはまれです。 デバイスでのローカル TCP/IP トラフィックの取り込みに PCAP が使用されます。
制約事項:

- PCAP はポートでのみ機能します (共有メモリーなどはありません)。

ヒント: PCAP は、すべてのローカル検査エンジンのためのクライアント IP/マスク値を使用して、モニターおよびレポートの対象を判別します。 異なるクライアント IP/マスク値を持つ複数の検査エンジンがある S-TAP とともにインストールされた PCAP は、すべての検査エンジンに定義されているすべてのクライアントからトラフィックを収集します。 PCAP は、意図したよりも多くの情報を処理し、Guardium システムに送信している可能性があります。

親トピック: Linux システムおよび UNIX システム: S-TAP の機能

Linux システムおよび UNIX システム: S-TAP からコレクターへの暗号化

ネットワークを介して暗号化 (TLS) 方式でコレクターと通信するように S-TAP エージェントを構成できます。

Guardium では、可能な場合は常に S-TAP とコレクター間のネットワーク・トラフィックを暗号化することを推奨しています。この暗号化を無効にする必要があるのは、パフォーマンスの優先順位がセキュリティーより高い場合のみです。暗号化を有効にすると、パフォーマンスに若干の影響が及びます。デフォルトの S-TAP 構成は、パフォーマンス上の影響を回避するために暗号化なしになっています。

以下の要因を考慮した上で、ご使用の環境に最適な選択肢を決定してください。

- TLS を指定して S-TAP を構成すると暗号化のための時間が追加で必要になるため、S-TAP エージェントがインストールされているデータベース・サーバーのパフォーマンスに影響を及ぼすおそれがあります。アプライアンス (コレクター) 側でも、このトラフィックの暗号化解除に時間を要します。
- アプリケーションとデータベースのユーザーが暗号化されていない方式でデータベースと通信している場合、暗号化を使用してネットワーク経由の通信を行うように S-TAP エージェントを構成しても、ネットワークの安全性は確保されません。

通常、S-TAP トラフィックの暗号化が効果的なのは、別のネットワーク上のアプライアンスに送信されるデータが暗号化されている場合、あるいはモニター対象のデータベース・トラフィックがネットワーク上で暗号化されている場合です。

暗号化は、検査エンジンの構成時に有効化され、いつでも変更可能です。

親トピック: Linux システムおよび UNIX システム: S-TAP の機能

Linux システムおよび UNIX システム: UID チェーン

UID チェーンは、それを使用することで、S-TAP が (K-TAP を介して)、データベース接続前に発生したユーザーのチェーンをトラッキングできるメカニズムです。それは、Solaris ゾーン、AIX WPAR、Solaris 8/9、Solaris 11 SPARC でサポートされています。

あるユーザーが、例えば、ssh informix@barbet、su - db2inst1、su -、su - oracle9 を実行してから、sqlplus scott/tiger@onora1 を実行することで、何回かユーザー名を変更してから、データベースに接続する場合があるとします。Guardium では、UID チェーンを使用して、プロセスを呼び出したプロセスに戻ってプロセスをトレースし、元の (問題の) ユーザーまで戻ることができます。

- Solaris ゾーンの場合は、ユーザー名の代わりにユーザー ID が報告される可能性があります。
- SSH クライアントの IP アドレスとポートが UID チェーンに追加されます。
- ゾーンを使用する Solaris 11 での Postgres はサポートされていません。一部のディレクトリーにマスター・ゾーンからスレーブ・ゾーンへのアクセスを許可しないゾーン構成があるためです。
- Solaris ゾーンおよび AIX® WPAR: guard_tap.ini ファイル内の db2bp_path を、db2bp 実行可能ファイルの絶対パス (グローバル・ゾーン/wpar から見た、関連する db2bp の絶対パス) に設定します。
- Solaris 8/9 では、プロセス間通信 (IPC) 用の UID チェーンはありません。
- Hadoop データベースでは、UID チェーンは検出されません。
- hunter_trace パラメーターは、UNIX S-TAP® での TCP/IP 接続には必須です。インストール時に hunter_trace = 1 を設定して、ローカル TCP/IP 接続の uid_chain を使用可能にします。
- セッションを開始したプロセスが、STAP がそれを調査する前に終了した場合、UID チェーンは機能しません。
- Linux for Db2 では、UID チェーンでローカル TCP はサポートされません。さらに、Db2 出口は、UID チェーンをサポートするために特定バージョンのデータベースを必要とします。
- 非 root ユーザーとして実行する場合、UID チェーンは S-TAP を使用した Db2 共有メモリー (SHM) に対しては機能しません。
- Guardium は、ネットワーク・トラフィックの UID チェーンをログに記録しません。
- Guardium は UID チェーンを判別するためにアプリケーションのプロセス ID に依存するため、Guardium は非常に短時間のセッションの UID チェーンをログに記録しない可能性があります。セッションを開始したプロセスが、STAP がそれを調査する前に終了した場合、UID チェーンは機能しません。

制約事項: トラフィックのインターセプトに A-TAP を必要とするシナリオでは、UID チェーンはサポートされません。以下のものがあります。

- Oracle ASO 暗号化トラフィックをインターセプトする ATAP
- Sybase 暗号化トラフィックをインターセプトする ATAP
- Teradata 暗号化トラフィックをインターセプトする ATAP
- Linux 上の Db2 または Informix 共有メモリー・トラフィック (ATAP が必要)

UID チェーン・レコードのパージ

2 時間を経過した UID チェーン・レコードは、通常の推論プロセスが実行されるときにパージされます。 経過時間が 1 日を超えるレコードは、毎夜パージされます。

親トピック: Linux システムおよび UNIX システム: S-TAP の機能

Linux システムおよび UNIX システム: プロキシー・ファイアウォール

プロキシー・サーバーから発生するトラフィックのモニター方法について説明します。

S-TAP は通常データベース・サーバーにデプロイされますが、K-TAP ベースのファイアウォールをプロキシー・サーバーにデプロイすることができます。S-GATE を使用することにより、プロキシー・サーバーから発生するトラフィックをモニターできます。appserver パラメーターの設定およびポリシー内での S-GATE の使用について詳しくは、「ポリシー」ヘルプ・トピックで、Linux システムおよび UNIX システム: アプリケーション・サーバー・パラメーターおよび『S-GATE アクション』(ブロッキング・アクション) を参照してください。

親トピック: Linux システムおよび UNIX システム: S-TAP の機能

Linux システムおよび UNIX システム: S-TAP のインストール、アップグレード、およびアンインストール

S-TAP をインストール、アップグレード、およびアンインストールするにはいくつかの方法があります。それぞれについて確認し、どれが最適かを判断してください。

- Linux システムおよび UNIX システム: S-TAP エージェントのインストール
  前提条件を確認し、デプロイ・モニター・エージェント・ツール、Guardium Installation Manager (GIM)、RPM、またはシェル・インストーラーを使用して、Linux、Solaris、AIX、および HP-UX の各サーバーに S-TAP をインストールします。
- Linux システムおよび UNIX システム: S-TAP のインストールまたはアップグレード後に再始動またはリブートが必要なもの
  このトピックでは、S-TAP のインストールまたはアップグレード後にリブートまたは再始動が必要なものについて詳しく説明します。再始動およびリブートの要件は、GIM による実装と GIM を使用しない実装のどちらの場合も同じです。
- Linux システムおよび UNIX システム: S-TAP エージェントのアンインストール
  古い構成ファイルを保存する必要がある場合、S-TAP の新規バージョンをインストールする前にこの手順を実行します。
- Linux システムおよび UNIX システム: S-TAP エージェントのアップグレード
  アップグレードのワークフローは、ご使用のモニター・メカニズムによって異なります。
- Linux システムおよび UNIX システム: データベースをアップグレードする際の S-TAP の管理
  以下の指針を使用して、データベースのアップグレード時に UNIX S-TAP を管理します。
- Linux システムおよび UNIX システム: データベースのオペレーティング・システムをアップグレードする際の S-TAP の管理
  以下のガイドラインを使用して、データベースのオペレーティング・システム (OS) のアップグレード時に、シェルまたは RPM を使用してインストールされた S-TAP を管理します。

親トピック: Linux システムおよび UNIX システム: S-TAP ユーザーズ・ガイド

Linux システムおよび UNIX システム: S-TAP エージェントのインストール

前提条件を確認し、デプロイ・モニター・エージェント・ツール、Guardium Installation Manager (GIM)、RPM、またはシェル・インストーラーを使用して、Linux、Solaris、AIX、および HP-UX の各サーバーに S-TAP をインストールします。

ライセンス・キーによっては、ファイル・サーバーとデータベースの両方のアクティビティー・モニターに同じ S-TAP エージェントを使用できます。FAM は、特定の S-TAP 構成を必要としません。

S-TAP の Linux、Solaris、AIX、および HP-UX へのインストール・フロー

このフローでは、1 つのコレクターに報告する単一のデータベースへの S-TAP のインストールについて説明します。クラスターとゾーンでの S-TAP の追加情報については、関連トピックを参照してください。

1. インストールを計画します。以下のトピックを確認してください。
   - Linux システムおよび UNIX システム: S-TAP モニター・メカニズムのサポート・マトリックス
   - Linux システムおよび UNIX システム: Linux、Solaris、AIX、および HP-UX の S-TAP のモニター・メカニズム
   - Linux システムおよび UNIX システム: S-TAP からコレクターへの暗号化
   - エンタープライズ・ロード・バランシング
2. 前提条件を確認します。
   - Linux システムおよび UNIX システム: データベース・バージョンとディレクトリーの要件
   - データベースに、十分な使用可能ディスク・スペースがあること (Linux システムおよび UNIX システム: S-TAP のディスク・スペース所要量)。
   - コレクターと S-TAP との間の通信に必要なポートがオープンであること (Linux システムおよび UNIX システム: S-TAP のポート要件)。
   - 必要な IP アドレスを特定し、データベース接続を確認します (Linux システムおよび UNIX システム: システムの詳細および検査)。
   - GIM を使用してインストールする場合、ターゲット・データベース・サーバーに GIM クライアントがインストールされている必要があります。 UNIX サーバーへの GIM クライアントのインストールを参照してください。
3. 次のいずれかの方法を使用して、S-TAP をインストールします。
   - モニター・エージェントをデプロイするためのクイック・スタート
   - Linux システムおよび UNIX システム: GIM の「クライアント別の設定」を使用した S-TAP クライアントのインストール
   - Linux システムおよび UNIX システム: RPM を使用した S-TAP のインストール、アンインストール、および更新
   - Linux システムおよび UNIX システム: シェル・インストーラーを使用した S-TAP のインストール

   S-TAP のインストール中に、オートディスカバリーが有効に設定されていると、データベースのオートディスカバリーが実行され、ディスカバーされたデータベースに対して検査エンジンが作成されます。オートディスカバリー・プロセスは、S-TAP のインストール時に 1 回だけ実行されます。自動的に繰り返し実行されることはありません。インストールの完了後に構成を変更できます。
4. システムでの必要性に応じて、オプションのコンポーネントを構成します。
   - Linux システムおよび UNIX システム: Kerberos 認証データベース・トラフィック

    - Linux システムおよび UNIX システム: Solaris ゾーンの S-TAP 構成
    - Linux システムおよび UNIX システム: Oracle RAC の S-TAP 構成
5. 必要に応じてリブートまたは再始動します (Linux システムおよび UNIX システム: S-TAP のインストールまたはアップグレード後に再始動またはリブートが必要なもの)。
6. S-TAP 構成を完了します。
    - Linux システムおよび UNIX システム: GUI からの S-TAP の構成
    - 上級者のみ: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集
7. 必要に応じて、エンタープライズ・ロード・バランシングを構成します。

- Linux システムおよび UNIX システム: S-TAP のインストール前提条件
  S-TAP のインストールを開始する前に、すべての前提条件を確認してください。
- Linux システムおよび UNIX システム: S-TAP エージェントのインストール
  Guardium Installation Manager (GIM)、GIM デプロイ・モニター・エージェント・ツール、RPM、シェル・インストーラー、またはネイティブ・インストーラーのうちニーズに最も適したものを使用して、S-TAP クライアントを Linux サーバー、Solaris サーバー、AIX サーバー、および HP-UX サーバーにインストールします。
- Linux システムおよび UNIX システム: 特別な環境での構成
  ゾーン、RAC、WPAR、クラスターがあるシステムでは、以下の該当する手順を使用してください。

親トピック: Linux システムおよび UNIX システム: S-TAP のインストール、アップグレード、およびアンインストール

Linux システムおよび UNIX システム: S-TAP のインストール前提条件

S-TAP のインストールを開始する前に、すべての前提条件を確認してください。

- Linux システムおよび UNIX システム: データベース・バージョンとディレクトリーの要件
  S-TAP や、関連付けられているエージェントをインストールする前に、次のデータベース・リリース、パッチ・レベル・コンポーネント、およびディレクトリーを確認してください。
- Linux システムおよび UNIX システム: S-TAP のディスク・スペース所要量
  S-TAP や、関連付けられているエージェントをインストールする前に、次のディスク・スペース所要量を確認してください。
- Linux システムおよび UNIX システム: S-TAP のポート要件
  Guardium システムと S-TAP エージェントとの間にファイアウォールが配置されている場合、それらのコンポーネント間の接続に使用されるポートがオープンであることを確認します。
- Linux システムおよび UNIX システム: システムの詳細および検査
  以下のシステムの詳細が手元にあり、データベースが Guardium システムと通信していることを確認します。

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのインストール

Linux システムおよび UNIX システム: データベース・バージョンとディレクトリーの要件

S-TAP や、関連付けられているエージェントをインストールする前に、次のデータベース・リリース、パッチ・レベル・コンポーネント、およびディレクトリーを確認してください。

表 1. Linux、Solaris、AIX、および HP-UX のデータベース・バージョン要件

データベース・タイプ	バージョン
Linux	MAKE バージョン 3.81 以降。MAKE ユーティリティーのバージョンを確認するには、`make -v` コマンドを実行します。
Oracle ASO、HP-UX 11.11	LD_PRELOAD のインストールが必須。パッチ PHSS_28436 以降でインストールされる。
TLS	サーバー上の S-TAP® では、/dev/random または /dev/urandom がサーバー上にあること。Linux システムおよび UNIX システム: S-TAP のポート要件の TLS ポート要件を参照してください。

注: GIM または S-TAP をインストールする root ユーザーには、ユーザーおよびグループを作成および削除する許可が必要です。

表 2. プラットフォームごとの必須ディレクトリー

要件タイプ	Linux	Solaris	AIX	HP-UX
インストール・フォルダーが存在しないか空	/usr/local/guardium/guard_stap	/usr/local/guardium/guard_stap	/usr/local/guardium/guard_stap	/usr/local/guardium/guard_stap
ファイルが存在	/bin/sh	/bin/sh	/bin/sh	/bin/sh
ファイルが存在	/bin/sed または /usr/bin/sed	/bin/sed または /usr/bin/sed	/bin/sed または /usr/bin/sed	/bin/sed または /usr/bin/sed
ファイルが存在	tar、awk、grep、tr	tar、awk、grep、tr	tar、awk、grep、tr	tar、awk、grep、tr
ファイルが存在	dd および /dev/zero	dd および /dev/zero	dd および /dev/zero	prealloc
ファイルが存在	uudecode が /usr/bin または /tmp にあるか、Perl が存在する	uudecode が /usr/bin または /tmp にあるか、Perl が存在する	uudecode が /usr/bin または /tmp にあるか、Perl が存在する	uudecode が /usr/bin または /tmp にあるか、Perl が存在する

親トピック: Linux システムおよび UNIX システム: S-TAP のインストール前提条件

Linux システムおよび UNIX システム: S-TAP のディスク・スペース所要量

S-TAP や、関連付けられているエージェントをインストールする前に、次のディスク・スペース所要量を確認してください。

表 1. Linux、Solaris、AIX、および HP-UX: S-TAP のディスク・スペース所要量

ディスク・スペース	GIM インストール	非 GIM インストール:
S-TAP プログラム・ファイル	• AIX: 400 MB • HP-UX: 500 MB • Linux: 450 MB • Solaris: 400 MB	• AIX: 300 MB • HP-UX: 400 MB • Linux: 350 MB • Solaris: 300 MB
FAM プログラム・ファイル	最小で 600 MB	
バッファー・ファイル	デフォルトでは、S-TAP は、匿名メモリーを使用して、Guardium システムに送信するためにデータをステージングします。バッファー・ファイルを使用するように S-TAP を構成する場合、サイズはデフォルトで 50 MB に設定されます。サイズは、guard_tap.ini ファイルの buffer_file_size パラメーターにより制御されます。	

親トピック: Linux システムおよび UNIX システム: S-TAP のインストール前提条件

Linux システムおよび UNIX システム: S-TAP のポート要件

Guardium システムと S-TAP エージェントとの間にファイアウォールが配置されている場合、それらのコンポーネント間の接続に使用されるポートがオープンであることを確認します。

ファイアウォール管理ユーティリティーを使用して、リストされているポートを確認し、必要に応じてオープンにします。

表 1. Linux、Solaris、AIX、および HP-UX のサーバーのポート要件

ポート	プロトコル	Guardium システムの接続先
16016	TCP	クリアな S-TAP
16018	TLS	暗号化された S-TAP
16020	TCP	プールされた通常接続
16021	TLS	プールされた TLS 接続
16022	TCP	フィード・プロトコル
16023	TLS	暗号化された S-TAP TLS

親トピック: Linux システムおよび UNIX システム: S-TAP のインストール前提条件

Linux システムおよび UNIX システム: システムの詳細および検査

以下のシステムの詳細が手元にあり、データベースが Guardium システムと通信していることを確認します。

- S-TAP のインストール先となっているデータベース・サーバーの IP アドレスを取得します。仮想 IP を使用する場合は、これらもメモしてください (これらは、構成を完了するときに、後で構成する必要があります)。
- 中央マネージャーでインストールする場合、この S-TAP を制御するコレクターであり、かつ S-TAP が報告を行う先のコレクターの IP アドレスを特定します。
- データベース・サーバーとコレクターの間の接続を確認します。データベースで、nmap -p <port> <ip_address> を入力します。例えば、ポート 16018 (Guardium® で TLS に使用されるポート) に IP アドレス 192.168.3.104 で到達できるか検査するには、次のコマンドを入力します。
  nmap -p 16018 192.168.3.104

通常の出力は、以下のとおりです。

```
Starting nmap V. 3.00 Interesting ports on g4.guardium.com (192.168.3.104): Port State Service 16018/tcp open unknown
```

親トピック: Linux システムおよび UNIX システム: S-TAP のインストール前提条件

Linux システムおよび UNIX システム: S-TAP エージェントのインストール

Guardium Installation Manager (GIM)、GIM デプロイ・モニター・エージェント・ツール、RPM、シェル・インストーラー、またはネイティブ・インストーラーのうちニーズに最も適したものを使用して、S-TAP クライアントを Linux サーバー、Solaris サーバー、AIX サーバー、および HP-UX サーバーにインストールします。

GIM のデプロイ・モニター・エージェント・ツールを使用して、GIM クライアントを自動的にアクティブ化し、S-TAP をインストールして、データベース・トラフィックのモニターを開始できます。モニター・エージェントをデプロイするためのクイック・スタートを参照してください。

S-TAP クライアントをインストールする場合、インストール・プログラムによって Guardium グループが存在するかどうかが確認されます。 このグループが存在しない場合は、インストール・プログラムによって作成されます。A-TAP や Db2 出口などの特定のコンポーネントや機能を使用する場合は、適切に機能するように、ユーザーをこのグループに追加する必要があります。これらの要件については、関連セクションで説明します。

インストール・プロセスは、STAP パッケージ全体 (S-TAP、K-TAP、A-TAP、Tee、P-CAP、Discovery) に対するログ・ファイルを作成します。ログ・ファイルは、失敗したインストールのトラブルシューティングに役立ちます。場所には、/var/tmp、/tmp、および /var/log があります。

インストール・プロセスによって、inittab、upstart、および rc の各スクリプトが更新されます。

S-TAP は /usr/local/guardium にインストールされます。

まれに、S-TAP を Guardium として (root ではなく) 実行する必要があります。これは他の問題を引き起こす可能性があるため、必要な場合にのみ使用してください。Guardium ユーザーとして S-TAP を実行すると、許可レベルが原因で、一部のデータベースまたはプロトコルが機能しなくなる場合があります。データベース・パスまたは exec ファイルに、Guardium ユーザーに読み取りを許可する権限が付与されていることを確認します。ご使用の環境に応じて、代表的な制限事項は以下のようになります。

- ディスカバリーは、機能が限定されています。
- AIX® WPAR および Solaris Zones のデータベースが機能しない可能性があります。インストール・パスまたは実行ファイルへのアクセス権限を確認してください。
- Oracle BEQ の場合は、データベースの始動後、または再始動後に S-TAP を再始動してください。
- Informix® 共有メモリーの場合は、データベースの始動後、または再始動後に S-TAP を再始動してください。
- Db2 共有メモリーの場合
  - ktap_fast_shmem が 0 に設定されていると、許可の問題が原因で shmctl が失敗した場合、ほとんどの場合に S-TAP が root として実行されるように変更する必要があります。
  - ktap_fast_shmem が 1 に設定されていて、グループによる読み取りが共有メモリー・セグメントで許可されている場合は、Db2 インスタンスがユーザー (Guardium) グループに追加されていることを確認してください。ただし依然として、サーバーごとに、Db2® の構成は 1 セットのみサポートされます。
  - Db2 ユーザーによる読み取りのみが共有メモリー・セグメントで許可されている場合は、S-TAP を root として実行する必要があります。(Db2 共有メモリー・セッションを開き、コマンド ipcs -ma を実行し、出力で MODE を確認します)

- Linux システムおよび UNIX システム: GIM の「クライアント別の設定」を使用した S-TAP クライアントのインストール
  Guardium Installation Manager の「クライアント別の設定」を使用して、スタンドアロン Guardium アプライアンスまたは中央マネージャーから S-TAP エージェントをインストールし、1 つ以上のデータベースでのインストールのスケジュールを設定します。
- Linux システムおよび UNIX システム: S-TAP の GIM インストールのパラメーター
  GIM のインストールで一般的に使用されるパラメーター (およびその簡略説明) を示します。
- Linux システムおよび UNIX システム: RPM を使用した S-TAP のインストール、アンインストール、および更新
  RPM を使用して、Linux サーバーで S-TAP をインストール、アンインストール、および更新できます。RPM によるインストールの利点は、データベース・サーバー上の他のすべてのソフトウェアを管理するとの同じ方法で STAP をインストールおよび管理する点です。
- Linux システムおよび UNIX システム: シェル・インストーラーを使用した S-TAP のインストール
  シェル・インストーラーを使用して、対話モードまたは非対話モードのいずれかで、Linux、Solaris、HPUX、AIX の各データベース・サーバーに S-TAP クライアントをインストールします。
- Linux システムおよび UNIX システム: S-TAP インストール・スクリプトのパラメーター
  S-TAP をインストールするためのスクリプト・パラメーターについて説明します。
- Linux システムおよび UNIX システム: ネイティブ・インストーラーを使用した S-TAP のインストールとアンインストール
  ネイティブ・インストーラーは、シェル・インストーラーにシェルを提供します。唯一の利点は、S-TAP がオペレーティング・システムの資産リポジトリーに確実に登録されることです。この登録は、Guardium で S-TAP をインストールする場合の要件ではありませんが、企業の要件になっている場合があります。ネイティブ・インストーラーは、必要な場合にのみ使用してください。
- Linux システムおよび UNIX システム: K-TAP の処理
  K-TAP について説明します。

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのインストール

Linux システムおよび UNIX システム: GIM の「クライアント別の設定」を使用した S-TAP クライアントのインストール

Guardium Installation Manager の「クライアント別の設定」を使用して、スタンドアロン Guardium アプライアンスまたは中央マネージャーから S-TAP エージェントをインストールし、1 つ以上のデータベースでのインストールのスケジュールを設定します。

始める前に

- すべての Linux システムおよび UNIX システム: S-TAP のインストール前提条件を確認します。
- Fix Central または Guardium の担当員から、S-TAP の正しいインストーラー・スクリプトを入手します。スクリプト名によって、データベース・サーバーのオペレーティング・システムが識別されます。

このタスクについて

インストール後、すべてのパラメーターを管理し、その制御下でインストールされたプロセスをモニターできます。他のいずれかのインストール方法を使用してインストールすると、GIM を使用して変更できるエージェント・パラメーターの数が少なくなります。

手順

1. GIM クライアントがデータベース・サーバーにインストールされていることを確認します。UNIX サーバーへの GIM クライアントのインストールを参照してください。
2. 適切な S-TAP モジュールを Guardium Installation Manager アプライアンスにアップロードします。
   a. 「管理」 > 「モジュール・インストール」 > 「モジュールのアップロード」に移動します。
   b. 「ファイルの選択 (Choose File)」をクリックし、インストールする S-TAP モジュールを選択します。
   c. 「アップロード」をクリックして、モジュールをアプライアンスにアップロードします。 モジュールが「アップロード済みモジュールのインポート」表に表示されます。
   d. 「アップロード済みモジュールのインポート」表で、インストールする S-TAP モジュールの横にあるチェック・ボックスをクリックします。 モジュールがインポートされ、インストール可能な状態になります。「モジュールのアップロード」ページがリセットされ、「アップロード済みモジュールのインポート」表が空になります。
3. 「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」にナビゲートします。
4. 「クライアントの選択」セクションで、S-TAP モジュールをインストールするデータベース・サーバーを選択します。表内のチェック・ボックスを使用して個々のクライアントを選択するか、「クライアント・グループを選択してください」メニューを使用してクライアントのグループを選択します。 「次へ」をクリックして先に進みます。
   重要:
   - クライアント・グループを作成するには、 をクリックして、「クライアント・グループの作成」ダイアログを開きます。「クライアントの追加」をクリックして、「既存のクライアント」ウィンドウを開き、クライアントを選択して、「OK」を選択します。「CSV からインポート」をクリックして CSV ファイルを選択し、CSV ファイルからインポートすることもできます。
   - クライアント・リストを変更した後、 をクリックしてクライアント・リストを更新します。
   - クライアントを再登録する前に Guardium システムから GIM クライアント情報を削除するには、「接続のリセット」を使用します。「接続のリセット」をクリックした後、GIM クライアント・プロセスの現在の状況が反映されるまで数分かかることがあります。
   - クライアントを選択して、「インストール済みモジュールの表示」をクリックし、「インストール済みモジュールの表示」ウィンドウを開きます。このウィンドウには、このクライアントにインストールされているすべてのモジュール (S-TAP など)、それぞれのバージョン、および選択されたすべてのクライアントでいずれかのモジュールが保留状態であるかどうかが表示されます。
   - グループを作成または更新し、GIM クライアントの「クライアント名」を編集する場合、ホスト名とアドレスは、Guardium システムに接続されている GIM システムの有効な値を反映している必要があります。無効なホスト名が指定された場合、編集後のクライアントはグループのメンバーとして表示されません。IP アドレスによるクライアントの追加はサポートされていません。
5. 「バンドルの選択」セクションで、「バンドルを選択してください」メニューを使用して、インストールするソフトウェアを特定します。 ソフトウェア・バンドルを選択すると、「選択されたバンドルのアクション」列に、各クライアントに対して実行されるアクションの「インストール」が示されます。
   ヒント:
   - 名前、モジュール、選択されたバンドルのアクション、およびクライアント OS などでクライアントをフィルタリングできます。結果の選択内容は保持されます。アクションは、**クライアントのフィルタリングされたリストにのみ適用されます**。「クライアントの選択」セクションに表示されるクライアントの数が「クライアントの構成」セクションに表示される数よりも大きくなります。
   - バンドルの旧バージョンを表示して操作するには、「最新バージョンのみを表示」チェック・ボックスをクリアします。
   - バンドル内の個々のモジュールを特定するには、「バンドルのみを表示」チェック・ボックスをクリアします。
   - 選択したバンドルと互換性がないクライアントを非表示にするには、「互換性のあるクライアントのみを表示」チェック・ボックスを選択します。
   重要:
   - デフォルトでは、「バンドルの選択」メニューには、プラットフォームや選択したクライアントとの互換性に関係なく、アップロードされた最新のバンドル・バージョンのみが表示されます。特定のプラットフォームまたはクライアントに対して異なるバンドル・バージョンをインストールするには、「最新バージョンのみを表示」チェック・ボックスをクリアし、必要なバンドルを選択してください。
   - 「クライアント別の設定」ツールを使用中に新しいバンドルをアップロードしてインポートした場合、その新しいバンドルを表示するには、ブラウザーをリフレッシュします。
   - 既にバンドルのインストールのスケジュールが設定されている場合、新しいバンドルをインストールすると、既存のスケジュールが削除されます。
   「次へ」をクリックして先に進みます。
6. 「パラメーターの選択」セクションで、必須パラメーターとオプション・パラメーターの値を指定します。オプション・パラメーターを追加または削除するには、 または を使用します。名前または説明でパラメーターを検索するには、 アイコンを使用します。 以下のパラメーターは必須です。
   - STAP_TAP_IP: STAP がインストールされているデータベース・サーバーまたはノードの IP アドレスまたは FQDN (-taphost コマンド行パラメーターと同じ)。指定されていない場合、GIM_CLIENT_IP 値が使用されます。
   - STAP_SQLGUARD_IP: この STAP の通信先である 1 次コレクターの IP アドレスまたは FQDN (-appliance コマンド行パラメーターと同じ)。指定されていない場合、GIM_URL 値が使用されます。
   重要: Linux システムおよび UNIX システム: S-TAP の GIM インストールのパラメーターで、エンタープライズ・ロード・バランシング・パラメーターを参照してください。
   重要: クライアント固有のパラメーターとして特定された場合を除き、「パラメーターの選択」セクションで指定された値は、すべてのクライアントのインストールに適用されます。クライアント固有のパラメーターについては、値のフィールドが無効になり、「クライアントの構成」セクションでクライアントごとに値が定義されます。
   「次へ」をクリックして先に進みます。
7. 「クライアントの構成」セクションで、表を使用して、各クライアントのパラメーター値を検討し、編集します。 編集可能なパラメーターには、パラメーター値の横に アイコンが表示されます。その アイコンをクリックして、値を編集します。
8. 「インストール」をクリックして、ソフトウェアのインストールを開始します。 アイコンを使用して、インストールをスケジュールし、「OK」 をクリックして続行します。

次のタスク

S-TAP 状況を確認します。

- 「成功」ポップアップで、「状況の表示」をクリックして、「状況」ウィンドウを開き、ソフトウェアのインストール/アップグレードをモニターします。 をクリックして、結果を最新表示します。インストール/アップグレードが失敗状況の場合、ボタンが表示されていれば、「アンインストール」をクリックします。表示されていない場合は、「接続のリセット」をクリックします。
- 「管理」 > 「レポート」 > 「インストール管理」 > 「GIM クライアント状況」のレポートで、モジュール状況を確認します。
- 「モニター」 > 「保守」 > 「S-TAP ログ」 > 「S-TAP 状況」 で、S-TAP の行の状況 (最初の列) が緑色であることを確認します。

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのインストール
関連概念:
Guardium Installation Manager
関連タスク:
クライアント別の設定

Linux システムおよび UNIX システム: S-TAP の GIM インストールのパラメーター

GIM のインストールで一般的に使用されるパラメーター (およびその簡略説明) を示します。

すべてのパラメーターはLinux システムおよび UNIX システム: S-TAP 構成パラメーターの編集にリストされています。
注意:
熟練したユーザーである場合や IBM 技術サポートに相談済みの場合を除いて、拡張パラメーターは変更しないでください。

表 1. その他の S-TAP パラメーター

GIM パラメーター	記述
STAP_TAP_IP	STAP がインストールされているデータベース・サーバーまたはノードの IP アドレスまたは FQDN (-taphost コマンド行パラメーターと同じ)。指定されていない場合、GIM_CLIENT_IP 値が使用されます。
STAP_SQLGUARD_IP	この STAP の通信先である 1 次コレクターの IP アドレスまたは FQDN (-appliance コマンド行パラメーターと同じ)。指定されていない場合、GIM_URL 値が使用されます。
STAP_ADDITIONAL_SQLGUARD_IPS	スペースで区切られた追加の SQLGUARD IP アドレスのリスト。
STAP_ENABLED	インストールの完了時に STAP を有効化します。デフォルト = 1 (はい)
STAP_FAM_ENABLED	FAM モニターを有効化します。デフォルトでは無効です。アップグレードする際に、v10.1.4 以前のバージョンで CLI のパラメーター FAM が有効になっていた場合は、アップグレード時にこのパラメーターが有効になります。
KTAP_ENABLED	Kernel TAP モジュールを制御します。デフォルト = 1 (はい)
KTAP_ALLOW_MODULE_COMBOS	Linux の場合のみ。バンドルに完全一致のカーネルがない場合、最も一致率が高いカーネルがインストールされます。K-TAP をインストールできない場合や、K-TAP が開始されない場合、ユーザーはインストールを続行するかどうかの確認を求められます。デフォルト = N
KTAP_LIVE_UPDATE	サーバーのリブートを必要としない KTAP の更新を有効にします。デフォルト = Y

表 2. エンタープライズ・ロード・バランシング・パラメーター

GIM パラメーター	記述
STAP_LOAD_BALANCER_IP	ロード・バランシングを構成する場合は必須です。ブランクの場合、エンタープライズ・ロード・バランシングは無効になります。 このオプションにより、この S-TAP がロード・バランシングで使用する中央マネージャーまたは管理対象ユニットの IP アドレスを指定します。 • エンタープライズ・ロード・バランサーを管理対象ユニット上で実行するように構成する場合、S-TAP が V10.1 以上でなければなりません。
STAP_INITIAL_BALANCER_TAP_GROUP	オプション。この S-TAP が属するエンタープライズ・ロード・バランシング用のアプリケーション・グループ名。 重要: スペースまたは特殊文字を含むグループ名はサポートされません。
STAP_INITIAL_BALANCER_MU_GROUP	オプション。app-group を関連付ける MU グループ名。定義されている LB-APP-GROUP を必要とします。S-TAP のインストール中に使用できるようにするには、前もって中央マネージャーに MU グループが存在している必要があります。 重要: スペースまたは特殊文字を含むグループ名はサポートされません。
STAP_LOAD_BALANCER_NUM_MUS	エンタープライズ・ロード・バランサーがこの S-TAP に割り振る管理対象ユニットの数。

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのインストール

Linux システムおよび UNIX システム: RPM を使用した S-TAP のインストール、アンインストール、および更新

RPM を使用して、Linux サーバーで S-TAP をインストール、アンインストール、および更新できます。RPM によるインストールの利点は、データベース・サーバー上の他のすべてのソフトウェアを管理するとの同じ方法で STAP をインストールおよび管理する点です。

始める前に

- すべての Linux システムおよび UNIX システム: S-TAP のインストール前提条件を確認します。
- Fix Central または Guardium の担当員から、S-TAP の正しいインストーラー・スクリプトを入手します。スクリプト名によって、データベース・サーバーのオペレーティング・システムが識別されます。

このタスクについて

RPM 名のフォーマットは、`guard-stap-10.6.0.0.89165-1-rhel-6-linux-x86_64.x86_64.rpm` です。ここで、最初の 3 つの番号は STAP のリリース番号 (10.0.0、10.1.2 など) であり、4 番目の番号はコード改訂 (89165) です。その直後に続く番号はパッケージの世代を表し、RPM に K-TAP モジュールが追加される場合に増加されます。

32 ビット S-TAP の RPM は 1 つですが、64 ビット S-TAP の RPM は 2 つあるため、32 ビット出口ライブラリーが必要でなければ、64 ビット S-TAP が 32 ビット・ライブラリーに依存することはありません。追加の RPM は guard-stap-32bit-exit-libs-10.1.0.89165-1-rhel-6-linux-x86_64.x86_64.rpm のような名前であり、メインの RPM に依存します。

デフォルトでは、インストール・プロセスは、Linux カーネルをチェックして、そのカーネルで処理を実行するための K-TAP モジュールが作成済みかどうかを判別します。モジュールが存在する場合、それがインストールを行います (ktap_installed = 1 が設定されます)。モジュールがない場合、ローダー柔軟性を有効にしない限り、K-TAP はインストールを行いません。ローダー柔軟性は、完全一致が存在しない場合に、現在作成されているモジュールのインストールに役立ちます。ローダー柔軟性が有効になっている場合、Linux カーネルに対応する K-TAP の作成が試行されます。

RPM は S-TAP を /opt/guardium にインストールします。この場所は変更できません。tap_ip は自動的にシステムのホスト名に設定されます。sqlguard_ip は、目的の構成のプレースホルダーとして 127.0.0.1 に設定されます。以下の手順で説明されているように、インストール後に構成を完了します。

RPM ログは /opt/guardium/rpm_logs に保存されます。

guard-config-update スクリプトは、root ユーザーとしても非 root ユーザーとしても実行できます。許可されている機能を確認するには、help コマンドを使用してください。

手順

1. S-TAP パッケージを unzip し、データベース・サーバーの /tmp に RPM をコピーします。
2. ローダー柔軟性を有効にするために、Linux 環境変数 `NI_ALLOW_MODULE_COMBOS="Y"` を設定します。
3. RPM をインストールします。
   a. RPM 名を取得するために、`rpm -qa | grep guard_stap` を実行します。
   b. コマンド `rpm -i <RPM_NAME>` を実行します。
   S-TAP がインストールされます。
   c. 4 で説明されているパラメーターを使用したスクリプト `guard-config-update` を実行して、構成を完了します。
   RPM が既にインストールされている場合、S-TAP シェル・インストーラーはインストールを行いません (二重インストールを防ぎます)。
4. 構成または更新するには、`root` としてシステムにログインし、ディレクトリーを /opt/guardium に切り替え、次のリストの該当するオプションとアクションを使用して、スクリプト `guard-config-update` を実行します。

[--stap-dir]	デフォルトでない場合の S-TAP インストール・ディレクトリー (デフォルト: /usr/local/guardium)
[--set-tap-ip [IP またはホスト名]]	S-TAP 構成ファイル /usr/local/guardium/guard_stap/guard_tap.ini の tap_ip を設定します (デフォルト: rh5u9x64t.guard.swg.usma.ibm.com)
[--set-sqlguard-ip [IP またはホスト名]]	S-TAP 構成ファイル /usr/local/guardium/guard_stap/guard_tap.ini の SQLGuard_0 セクションの sqlguard_ip を設定します (デフォルト: 127.0.0.1)
[--add-sqlguard [ID] [IP またはホスト名]] (V10.1.4 以上)	S-TAP 構成ファイル /usr/local/guardium/guard_stap/guard_tap.ini へ SQLGuard_ID セクションを追加します
[--remove-sqlguard [ID]] (V10.1.4 以上)	S-TAP 構成ファイル /usr/local/guardium/guard_stap/guard_tap.ini から SQLGuard_ID セクションを削除します
[--modify-sqlguard [ID] [parameter] [value]] (V10.1.4 以上)	S-TAP 構成ファイル /usr/local/guardium/guard_stap/guard_tap.ini の SQLGuard_ID セクションのパラメーターを設定します。パラメーター: sqlguard_ip SQLGuard ユニットの IP アドレスまたはホスト名 sqlguard_port SQLGuard ユニットへの接続に使用されるポート (デフォルト: 16016) primary 優先順位 (1=1 次、2=2 次、3=3 次、以降同様) num_main_thread SQLGuard で使用するメイン接続の数。participate_in_load_balancing = { 1、4 } (デフォルト: 1) と併用 connection_pool_size SQLGuard ユニットへのメイン接続ごとのデータ接続の (デフォルト: 0)
[--modify-tap [parameter] [value]] (V10.1.4 以上)	S-TAP 構成ファイル /usr/local/guardium/guard_stap/guard_tap.ini の TAP セクションのパラメーターを設定します。パラメーター: tap_debug_output_level デバッグ・レベルを設定します (0 以上の整数にする必要がありますが、2 または 3 にはできません) participate_in_load_balancing ロード・バランシングへの参加を設定します (値: 1、2、3、4)。(Linux システムおよび UNIX システム: S-TAP のロード・バランシング・モデルと構成ガイドラインを参照) use_tls TLS を有効にします [0、1] failover_tls

	v10.5 から非推奨になっています。非 TLS への TLS 接続のフェイルオーバー [0、1] hunter_trace UID チェーン・レポートを有効にします [0、1] buffer_file_size バッファー・ファイル・サイズ (MB) alternate_ips STAP 用の代替 IP/ホスト名のコンマ区切りリスト firewall_installed ファイアウォールを有効にします [0、1] firewall_fail_close 判断がないとき (SQLGuard に到達不能な場合やタイムアウトに達した場合など) に実行するアクション [0: 何もしない、1: 接続をブロック] firewall_default_state デフォルトの状態を設定します [0: 監視しない、1: 監視する] firewall_timeout ファイアウォール・タイムアウトを秒単位で設定します firewall_force_watch firewall_default_state=0 の場合でも監視する IP/マスクのコンマ区切りリスト firewall_force_unwatch firewall_default_state=1 の場合でも監視しない IP/マスクのコンマ区切りリスト
[--help-config [option]]	使用可能な場合、ini 内のオプションに関する情報を表示します (オプションが何も指定されない場合、使用可能なすべてを表示します)。
[--set-flexload [0 または 1]]	K-TAP フレックス・ロードを有効または無効にします
[--retry-ktap-load]	KTAP ロードを再試行します (S-TAP が自動的に再始動される、dev パッケージのインストール、KTAP 要求による更新、または flexload の変更の後に役立ちます。)
[--discover-ies]	ディスカバリーを実行し、すべての検査エンジンをディスカバーされたものと置換し
[--stop [service]]	サービス (S-TAP、または monitor) を一時的に停止します (Solaris サービスおよび inittab はこれを永久無効として処理し、再度有効にするまでは起動時に自動始動しません)
[--start [service]]	サービス (S-TAP、または monitor) がまだ実行されていない場合、それを開始します (有効化を意味します)
[--restart [service]]	サービス (S-TAP、または monitor) が既に実行されている場合、それを再始動します
[--disable [service]]	サービス (S-TAP、または monitor) が再実行されないようにします
[--enable [service]]	サービス (S-TAP、または monitor) に自動始動を設定します
[--status]	開始されているサービスと、自動的に開始するように設定されているかどうかを表示します
--show-tap [option]	guard_tap.ini ファイルの TAP セクション内のパラメーターについて現在格納されている値を表示します
--show-ies	guard_tap.ini ファイル内の現在構成されている検査エンジンを表示します

5. アップグレードするには、RPM パッケージを /opt/guardium にコピーし、コマンド `rpm -U <RPM_NAME>` を実行します。
6. アンインストールするには、以下のようにします。
   a. RPM 名を取得するために、`rpm -qa | grep guard_stap` を実行します。
   b. `rpm -e <RPM_NAME>` を実行します。

アンインストール後も、/opt/guardium ディレクトリーは引き続き存在しますが、/opt/guardium/guard_stap/guard_tap.ini.rpmsave と /opt/guardium/rpm_logs のみが含まれます。

次のタスク

インストールが完了したら、次のように S-TAP 状況を確認します。

- 「モニター」 > 「保守」 > 「S-TAP ログ」 > 「S-TAP 状況」 で、S-TAP の行の状況 (最初の列) が緑色であることを確認します。

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのインストール

Linux システムおよび UNIX システム: シェル・インストーラーを使用した S-TAP のインストール

シェル・インストーラーを使用して、対話モードまたは非対話モードのいずれかで、Linux、Solaris、HPUX、AIX の各データベース・サーバーに S-TAP クライアントをインストールします。

始める前に

- すべての Linux システムおよび UNIX システム: S-TAP のインストール前提条件を確認します。
- Fix Central または Guardium の担当員から、S-TAP の正しいインストーラー・スクリプトを入手します。スクリプト名によって、データベース・サーバーのオペレーティング・システムが識別されます。

このタスクについて

インストールとアンインストールを行うには、対話モードが簡単な方法ですが、システムごとに個別に実行する必要があります。対話式インストーラーを使用すると、ステップごとに検証処理が実行されるため、エラーが発生しにくくなります。小規模なデプロイメントの場合や、ガイドに従い手順を追ってインストールを行う必要がある場合は、対話式インストーラーを使用すると便利です。非対話モードでは、1 つのスクリプトで複数の S-TAP をインストールできるため、大規模なデプロイメントを管理する場合は特に便利です。

いずれかの段階でインストールが失敗した場合は、その時点までのすべてのステップを取り消します。 S-TAP を部分的にインストールしたまま放置しないでください。

S-TAP パッケージ名の形式は `guard-stap-guard-10.6.0.0_r79927_1-rhel-5-linux-x86_64.sh` です。最初の 3 つの数字はリリース番号で、その後に改訂番号が続きます (この例では r79927)。

一部のディレクトリーには、以下のように、より多くのオープン権限があります。

- guardium/bin および guardium/etc には、すべてに対する読み取り権限と実行権限があります

インストールされている一部のファイルには、以下のように、より多くのオープン権限があります。

- guard_stap/guardctl、guard_stap/guard-config-update、guard_stap/common.sh、guard_stap/platform_checks.sh、guardium/bin/guard-executor-32、guardium/bin/guard-executor-64、guardium/bin/guard-util、guardium/bin/guard-tag、guardium/etc/guard-stap-build.conf には、すべてに対する読み取り権限と実行権限があります
- guard_stap/guard-stap-build.conf、guard-stap-install.conf、guard_stap/guard_tap.iniには、すべてに対する読み取り権限があります

個々の S-TAP のインストールには、対話モードをお勧めします。システムにより、基本構成を尋ねるプロンプトが出され、ユーザーの入力がすぐに検証されるので、エラーは発生しません。デフォルトでは、S-TAP インストール時に K-TAP が自動的にインストールされます。S-TAP インストーラーは、カーネルのバージョンに合った K-TAP を使用できるかどうか検査します。インストール・プロセスは、対応する K-TAP を検出できない場合、当該 Linux カーネルに対応する K-TAP の作成を試みます。K-TAP をインストールできない場合や、K-TAP が開始されない場合、ユーザーはインストールを続行するかどうかの確認を求められます。

単一コマンドを実行し、tapfile パラメーター --tapfile <path to ini file> と、データベースおよびその詳細を指定する guard_tap.ini ファイルを使用して、複数のデータベースおよび複数のシステムにインストールする場合は、非対話モードを使用します。複数のデータベースにインストールする場合は、非対話モードの代わりに GIM を使用することを検討してください。

手順

1. `root` アカウントを使用して、データベース・サーバーにログオンします。
2. インストール・ディレクトリーを指定し、そこに十分なディスク・スペース (合計で、約 400 MB から 500 MB) があることを確認します。
3. S-TAP .tgz をデータベース・サーバー上のローカル・ディスク (通常は /tmp) にコピーします。
4. 非対話モードによる標準インストールの場合、最小限のパラメーターは以下のとおりです。

   ```
   ./guard-stap-guard-<release number>_<revision number>_1-rhel-5-linux-x86_64.sh -- --ni --dir
   <guardium_installation_directory>   --tapip <tap_ip or host_name>  --sqlguardip < sqlguard_ip or host_name>
   ```

   注: S-TAP インストーラーには、さまざまな Linux カーネルに固有の、考えられるすべてのモジュールが含まれています。まれに、S-TAP パッケージに、該当する K-TAP モジュールが含まれていない場合があります。この場合は、以下のコマンドを使用して、K-TAP モジュールを /tmp にコピーします。K-TAP モジュール・ファイルは、インストール時に S-TAP インストール・ディレクトリーにコピーされます。

   ```
   ./guard-stap-guard-10.0.0_r79927_1-rhel-5-linux-x86_64.sh --
   --modules /tmp/modules-guard-10.0.0_r79927_1.tgz"
   ```

5. 対話モードの場合、インストールされているスクリプトを実行します。場合によっては、S-TAP を Guardium として実行する必要があります。これは他の問題を引き起こす可能性があるため、本当に必要な場合にのみ使用してください。入力する必要がある唯一の値は、SQL Guard ユニットの IP アドレスです。その他の値はすべてデフォルトのままにすることができます。インストーラーによって次のプロンプトが表示されます。

   ```
   Enter the path prefix [/usr/local]?
   Directory /usr/local/guardium/guard_stap does not exist, would you like to create it? [Y/n]
   System library path [/usr/lib]?
   Run STAP as root, or as user 'guardium'? [R/u]
   Install STAP as root, or as user 'guardium'? [r/U]
   Would you like to run guard_discovery? [Y/n]
   Do you want to configure load balancer functionality? [y/N]
   IP address of the SQL Guard unit:
   Do you want to edit the parameters file? [y/N]

   If you later update your kernel to another version, we can
   try to load the closest fitting delivered module.  This
   feature is not enabled by default, but we recommend enabling
   it to reduce delays in support.  Note that if all the
   packages require to build natively are installed, a local
   build to generate an exact matching module will be attempted
   prior to looking for non-exact matches.
   Do you wish to enable this feature (y/N/h)?
   ```

   スクリプトで「Would you like to run guard_discovery? [Y/n]」と尋ねられたときに yes を選択すると、guard_discovery が --update-tap-flag を設定して 1 回実行され、検査エンジンが初期構成されます。いずれの場合も、guard_discovery の --send-to-sqlguard-flag が構成され、guard_discovery が 24 時間ごとに 1 回実行されます。

次のタスク

S-TAP 状況を確認します。

- 「モニター」 > 「保守」 > 「S-TAP ログ」 > 「S-TAP 状況」 で、S-TAP の行の状況 (最初の列) が緑色であることを確認します。

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのインストール

Linux システムおよび UNIX システム: S-TAP インストール・スクリプトのパラメーター

S-TAP をインストールするためのスクリプト・パラメーターについて説明します。

インストール・スクリプトのコマンド行構文

使用法: guard-stap-setup [options]

--ni	非対話式インストール
-k | -p	K-TAP、または PCAP を使用したインストール
--ignore-compat	スクリプト互換性検査を無視します。
-u	以前のインストール済み環境が見つかった場合、その環境を更新します。
--user | --root	ユーザーまたは root として S-TAP を実行。
--userinst | --rootinst	ユーザーまたは root として S-TAP をインストール。
--overwrite-existing	既存のインストール済み環境が見つかった場合、その環境を上書きします。
--libdir	システム・ライブラリー・パス。ライブラリー・ファイルは、64 ビット・システムでも、システムによって信頼できるものとして構成されているディレクトリー内 (例えば、Linux では /usr/lib) に配置する必要があります。デフォルト = /usr/lib
--tls force | none	S-TAP の TLS 設定。failover オプションは v10.5 から非推奨になっています。
--dir <dir>	S-TAP のインストール・ディレクトリー。
--tapfile <file>	インストール・プロセスでは、この guard_tap.ini ファイルが読み取られ、インストールしている STAP に対してそのパラメーターが使用されます。例: /var/tmp/guard-stap-10.0.0_r103368_v10_5_1-rhel-5-linux-x86_64.sh --ni --dir /usr/local --tapfile /var/tmp/guard_tap.ini

--ipfile <file>	ホスト名、IP アドレス、および Guardium システム・アドレスを単一のスペースで区切ったリストを指定するテキスト・ファイル。例:
	`database-01 10.10.10.1 gmachine-01`
	`database-02 10.10.10.2 gmachine-01`
	`database-03 10.10.10.3 gmachine-02`
	
	コマンドは次のようになります。/var/tmp/guard-stap-10.0.0_r103368_v10_5_1-rhel-5-linux-x86_64.sh --ni --dir /usr/local --ipfile /var/tmp/ipfile.txt これらのパラメーターを構成する方法として GIM を使用する方が大幅に簡単になります。

--tapip <tapip>	S-TAP のインストール先となるマシンの IP を指定します。
--sqlguardip <sqlguardip>	S-TAP の通信相手となる Guardium システムの IP を指定します。
--presets <file> | <preset-options>	インストール設定の読み取り、またはインストール設定のファイルへの書き込みを指定します。
--no-discovery	検査エンジンを構成するためにディスカバリー・ユーティリティーを使用しません。
--modules <module-bundles>	外部の K-TAP モジュールのバンドルを指定します。
--ktap_allow_module_combos	K-TAP のロードで、カーネルのあいまい一致を許可します。
--load-balancer-ip <load_balancer_ip>	この S-TAP がエンタープライズ・ロード・バランシングに使用する中央マネージャーまたは管理対象ユニットの IP アドレス。
--lb-app-group <app_group>	オプション。この S-TAP が属するエンタープライズ・ロード・バランシング用のアプリケーション・グループ名。 重要: スペースまたは特殊文字を含むグループ名はサポートされません。
--lb-mu-group <mu_group>	オプション。app-group を関連付ける MU グループ名。定義されている LB-APP-GROUP を必要とします。このパラメーターは、初期インストール時に 1 回しか指定できません。S-TAP のインストール中に使用できるようにするには、前もって中央マネージャーに MU グループが存在している必要があります。 重要: スペースまたは特殊文字を含むグループ名はサポートされません。
--lb-num-mus <number_of_mus>	エンタープライズ・ロード・バランサーがこの S-TAP に割り振る管理対象ユニットの数。

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのインストール

Linux システムおよび UNIX システム: ネイティブ・インストーラーを使用した S-TAP のインストールとアンインストール

ネイティブ・インストーラーは、シェル・インストーラーにシェルを提供します。唯一の利点は、S-TAP がオペレーティング・システムの資産リポジトリーに確実に登録されることです。この登録は、Guardium で S-TAP をインストールする場合の要件ではありませんが、企業の要件になっている場合があります。ネイティブ・インストーラーは、必要な場合にのみ使用してください。

ネイティブ・インストーラーを使用すると、S-TAP がオペレーティング・システムの資産リポジトリーに確実に登録されます。この登録は、Guardium で S-TAP をインストールする場合の要件ではありませんが、企業の要件になっている場合があります。OS タイプごとに別個のネイティブ・インストーラーがあります。

- Linux システムおよび UNIX システム: AIX ネイティブ・インストーラーを使用した S-TAP のインストールおよびアンインストール
- Linux システムおよび UNIX システム: HP-UX ネイティブ・インストーラーを使用した S-TAP のインストールおよびアンインストール
- Linux システムおよび UNIX システム: Solaris ネイティブ・インストーラーを使用した S-TAP のインストールおよびアンインストール

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのインストール

Linux システムおよび UNIX システム: AIX ネイティブ・インストーラーを使用した S-TAP のインストールおよびアンインストール

始める前に

すべての Linux システムおよび UNIX システム: S-TAP のインストール前提条件を確認します。

手順

1. S-TAP のインストール先となっているデータベース・サーバーの IP アドレスを取得します。仮想 IP を使用する場合は、これらもメモしてください (これらは、構成を完了するときに、後で構成する必要があります)。
2. この S-TAP を制御するコレクターであり、かつ S-TAP が報告を行う先のコレクターの IP アドレスを特定します。
3. データベース・サーバーとコレクターの間の接続を確認します。データベースで、nmap -p <port> <ip_address> を入力します。例えば、ポート 16018 (Guardium® で TLS に使用されるポート) に IP アドレス 192.168.3.104 で到達できるか検査するには、次のコマンドを入力します。
   nmap -p 16018 192.168.3.104
   通常の出力は、以下のとおりです。
   ```
   Starting nmap V. 3.00 Interesting ports on g4.guardium.com (192.168.3.104): Port State Service 16018/tcp open unknown
   ```
4. S-TAP のインストール DVD から、ご使用の AIX® のバージョン向けの該当するネイティブ・インストーラー・ファイル (.bff ファイル) を見つけます。
5. クリーン・サーバー (以前に S-TAP インストールを行っていない環境) で、以下のコマンドを入力し、AIX 用のシェル・インストーラーを抽出します。ファイル名は該当する .bff ファイル名に置き換えてください。

   ```
   installp -aX -d/var/tmp<filename> SqlGuardInstaller
   例:
   installp -aX -d/var/tmp/guard-stap-guard-8.0.00rc1_r20934_1-aix-5.2-aix-powerpc.bff SqlGuardInstaller
   ```

   シェル・インストーラーが抽出され、/usr/local の下に guardium という名前で置かれます。

6. インストール手順の対話式インストーラーの実行に進み、オペレーティング・システム・バージョンのデフォルトのインストール・スクリプトではなく、生成されたインストール・スクリプトを実行します。

親トピック: Linux システムおよび UNIX システム: ネイティブ・インストーラーを使用した S-TAP のインストールとアンインストール

ネイティブ・インストーラーを使用した AIX S-TAP の削除

手順

ネイティブ・インストーラーを使用して AIX S-TAP を削除するには、以下のコマンドを使用します。

```
/usr/lib/instl/sm_inst installp_cmd -u -f 'filename'
```

例

```
/usr/lib/instl/sm_inst installp_cmd -u -f'SqlGuardInstaller'
```

Linux システムおよび UNIX システム: HP-UX ネイティブ・インストーラーを使用した S-TAP のインストールおよびアンインストール

始める前に

すべての Linux システムおよび UNIX システム: S-TAP のインストール前提条件を確認します。

手順

1. S-TAP のインストール先となっているデータベース・サーバーの IP アドレスを取得します。仮想 IP を使用する場合は、これらもメモしてください (これらは、構成を完了するときに、後で構成する必要があります)。
2. この S-TAP を制御するコレクターであり、かつ S-TAP が報告を行う先のコレクターの IP アドレスを特定します。
3. データベース・サーバーとコレクターの間の接続を確認します。データベースで、nmap -p <port> <ip_address> を入力します。例えば、ポート 16018 (Guardium® で TLS に使用されるポート) に IP アドレス 192.168.3.104 で到達できるか検査するには、次のコマンドを入力します。
   nmap -p 16018 192.168.3.104
   通常の出力は、以下のとおりです。
   ```
   Starting nmap V. 3.00 Interesting ports on g4.guardium.com (192.168.3.104): Port State Service 16018/tcp open unknown
   ```
4. Guardium S-TAP® のインストール DVD で、ご使用の HPUX のバージョン向けの該当するネイティブ・インストーラー・ファイル (.depot.gz ファイル) を見つけます。
5. 以下のコマンドを使用して、ファイルを解凍します。

   ```
   gzip -d <filename>.depot.gz
   ```

6. 選択したファイル名 (該当するネイティブ・インストーラー・ファイル) とデータベース・サーバーのホスト名指定して、以下のようにswinstall コマンドを入力します。このコマンドは対話式プログラムを開始します。 プロンプトに従い、該当するコントロールを使用して、該当する S-TAP インストール・プログラム (.sh ファイル) をインストールします。プログラムは /var/spool/sw/var/tmp にインストールされます。

   ```
   swinstall -s /var/tmp/<filename>.depot @ ,hostname>:/var/spool/sw
   ```

7. インストール手順の対話式インストーラーの実行に進み、オペレーティング・システム・バージョンのデフォルトのインストール・スクリプトではなく、生成されたインストール・スクリプトを実行します。

親トピック: Linux システムおよび UNIX システム: ネイティブ・インストーラーを使用した S-TAP のインストールとアンインストール

ネイティブ・インストーラーを使用した HPUX S-TAP の削除

手順

ネイティブ・インストーラーを使用して HPUX S-TAP を削除するには、次のコマンドを使用します。

```
swremove @<hostname>:/var/spool/sw
```

Linux システムおよび UNIX システム: Solaris ネイティブ・インストーラーを使用した S-TAP のインストールおよびアンインストール

始める前に

すべての Linux システムおよび UNIX システム: S-TAP のインストール前提条件を確認します。

手順

1. S-TAP のインストール先となっているデータベース・サーバーの IP アドレスを取得します。仮想 IP を使用する場合は、これらもメモしてください (これらは、構成を完了するときに、後で構成する必要があります)。
2. この S-TAP を制御するコレクターであり、かつ S-TAP が報告を行う先のコレクターの IP アドレスを特定します。
3. データベース・サーバーとコレクターの間の接続を確認します。データベースで、nmap -p <port> <ip_address> を入力します。例えば、ポート 16018 (Guardium® で TLS に使用されるポート) に IP アドレス 192.168.3.104 で到達できるか検査するには、次のコマンドを入力します。
   nmap -p 16018 192.168.3.104
   通常の出力は、以下のとおりです。
   ```
   Starting nmap V. 3.00 Interesting ports on g4.guardium.com (192.168.3.104): Port State Service 16018/tcp open unknown
   ```
4. Guardium S-TAP® のインストール DVD で、以下のように、ご使用の Solaris のバージョン向けの該当するネイティブ・インストーラー・ファイル (.pkg ファイル) を見つけます。
5. 以下のように、pkgadd コマンドを入力し、選択したファイルを使用してインストーラーを実行します。

   ```
   pkgadd -d <filename>.pkg
   ```

   シェル・インストーラーが /usr/local/guardium の下に抽出されます。
6. インストール手順の対話式インストーラーの実行に進み、オペレーティング・システム・バージョンのデフォルトのインストール・スクリプトではなく、抽出されたシェル・インストーラー・スクリプトを実行します。

親トピック: Linux システムおよび UNIX システム: ネイティブ・インストーラーを使用した S-TAP のインストールとアンインストール

ネイティブ・インストーラーを使用した Solaris S-TAP の削除

手順

ネイティブ・インストーラーを使用して S-TAP を削除するには、以下のコマンドを使用します。

```
pkgrm GrdTapIns
```

Linux システムおよび UNIX システム: K-TAP の処理

K-TAP について説明します。

K-TAP は、オペレーティング・システムにインストールされるカーネル・モジュールです。それは、S-TAP インストール時にインストールされます。インストール後、構成ファイル設定を使用して、使用可能にしたり使用不可にしたりすることができます。使用可能に設定された場合、これは、データベース・クライアントとデータベース・サーバーとの間の通信に使用するメカニズムをフックすることにより、データベース・サーバーへのアクセスを監視します。 K-TAP では、データベース・クライアントのサーバーとの接続方法を変更する必要はありません。

インストール時に、サーバーのオペレーティング・システムに K-TAP カーネル・モジュールをロードするかどうかを選択します。これは、そのモジュールをロードする唯一の方法です。 最初に K-TAP をロードせずに、後で K-TAP を使用することにした場合は、S-TAP® を削除してから、再インストールする必要があります。
注: インストール中に K-TAP のロードが適切に行われなかった場合、ハードウェアまたはソフトウェアの互換性が原因である可能性があります。デフォルトの収集メカニズムとして P-CAP がインストールされます。
注: セッション内トラフィックは、コールバックを使用することで古い KTAP から新規 KTAP に転送されます。つまり、ほとんどのデータベースでは、既存のセッションに対する新しい KTAP を使用したインターセプトが再開されるまでに、2 つの SQL 要求を受け取ることができます。Sybase IOCP の場合は、セッションの性質上、3 つの SQL 要求を受け取ります。

- Linux システムおよび UNIX システム: K-TAP の概要
  S-TAP のインストール時、正しい K-TAP バージョンのロードが試行されます。
- Linux システムおよび UNIX システム: K-TAP の作成
  使用可能な何百もの Linux ディストリビューションが存在し、そのリストは増え続けています。このことは、ご使用の Linux ディストリビューションで使用可能な K-TAP がまだ存在しない可能性があることを意味します。適切な K-TAP が使用可能でない場合、S-TAP インストール・プロセスにより自動で作成することができます。
- Linux システムおよび UNIX システム: GIM を使用した K-TAP モジュールのコピー
  Linux データベース・サーバー用のカスタム K-TAP モジュールを作成する場合、GIM を使用して、そのモジュールを他の Linux データベース・サーバーにコピーできます。

- Linux システムおよび UNIX システム: 新規 K-TAP モジュールの他のシステムへのコピー
  Linux データベース・サーバー用の新しい K-TAP モジュールをビルドしたら、そのモジュールを同じ Linux ディストリビューションを稼働する他のデータベース・サーバーにコピーすることができます。
- Linux システムおよび UNIX システム: P-CAP がデフォルトでインストールされた場合のインストール後の K-TAP の有効化
  インストール・プロセスの実行中に、K-TAP が正しくロードを行えない場合 (おそらく、ハードウェアまたはソフトウェアの非互換性が原因)、デフォルトの収集メカニズムとして P-CAP がインストールされます。互換性の問題が解決した後に K-TAP に切り替えるには、以下の手順を行います。

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのインストール

Linux システムおよび UNIX システム: K-TAP の概要

S-TAP のインストール時、正しい K-TAP バージョンのロードが試行されます。

KTAP ローダー・メカニズム

KTAP ローダー・メカニズムは、Linux S-TAP のインストール (GIM および GIM 以外を使用) で以下のシーケンスを使用します。

注: KTAP ローダー・メカニズムは前のステップが成功しなかった場合、次のステップに自動的に進みます。

1. KTAP ローダーは、オペレーティング・システム・レベルに完全に一致するカーネル・モジュールを探し、見つかった場合は、それをロードします。
2. KTAP ローダーが一致するものを見つけられなかった場合、KTAP モジュールをローカルにコンパイルし、それをロードします。これは、システムに必要なパッケージ (ブートされたカーネルの場合は gcc および kernel-devel) がインストールされている場合にのみ発生する可能性があります。
3. KTAP ローダーが正しいカーネル・モジュールをまだロードできない場合で、かつ FlexLoad メカニズムがオンの場合、KTAP ローダーは最も一致率が高いカーネル・モジュールを見つけ、それをロードします。

   FlexLoad メカニズムをオンにするには、以下のフラグを使用します。
   - シェル・インストールの場合: --ktap_allow_module_combos
   - GIM インストールの場合: KTAP_ALLOW_MODULE_COMBOS=Y
4. KTAP がカーネル・モジュールをロードできない場合、「ロードに失敗しました」メッセージで通知します。それは、KTAP なしで S-TAP をインストールしたか、S-TAP インストールが失敗したかのいずれかです。その場合、一致するモジュールを Guardium サポートに要求できます。これは、準備するのに約 2 週間かかります。

注: KTAP パラメーター・トピックの CUSTOM BUNDLES に関する情報を参照してください。
親トピック: Linux システムおよび UNIX システム: K-TAP の処理

Linux システムおよび UNIX システム: K-TAP の作成

使用可能な何百もの Linux ディストリビューションが存在し、そのリストは増え続けています。このことは、ご使用の Linux ディストリビューションで使用可能な K-TAP がまだ存在しない可能性があることを意味します。適切な K-TAP が使用可能でない場合、S-TAP インストール・プロセスにより自動で作成することができます。

Linux システムに S-TAP をインストールすると、インストール・プロセスは Linux カーネルをチェックして、そのカーネルで処理を実行するための K-TAP が作成されているかどうかを判別します。過去に KTAP がロードされていないカーネルが実行されている場合は、一致するモジュールを検索してそれをロードします。インストール・プロセスは、対応する K-TAP を検出できない場合、当該 Linux カーネルに対応する K-TAP の作成を試みます。

ほとんどの K-TAP コードは、カーネルから独立しています。インストーラーによってコードはカーネルと対話できるようになります。この層は、独自仕様のソース・コードとして提供されます。インストーラーは、この独自仕様のソース・コードをユーザーの Linux カーネルに対してコンパイルすることで、完全な K-TAP を作成します。 これにより、ご使用の Linux ディストリビューションに固有の K-TAP が作成されます。

このプロセスでは、Linux ディストリビューションで提供されている標準のカーネル開発ユーティリティーが、K-TAP が作成されるデータベース・サーバーに存在している必要があります。開発パッケージはカーネルに完全に対応していなければなりません。必要なパッケージは、以下のとおりです。

- RedHat ベースのサーバー:
  - ブートされたカーネルの kernel-devel-`uname -r`
  - gcc コンパイラー・パッケージ
- Suse ベースのサーバー:
  - ブートされたカーネルの kernel-devel または kernel-source (利用可能な場合)
  - ブートされたカーネルの kernel-default-devel
  - gcc コンパイラー・パッケージ
- Ubuntu ベースのサーバー:
  - ブートされたカーネル・イメージの linux-headers
  - gcc コンパイラー・パッケージ

同じ Linux ディストリビューションを実行している複数のシステムがある場合は、1 つのシステム上で K-TAP を作成し、それをその他のシステムにコピーすることができます。例えば、テスト・システムで K-TAP を作成し、テストを行った後に、1 つ以上の実動データベース・サーバーにその K-TAP をコピーできます。 Guardium Installation Manager (GIM) を使用して S-TAP をインストールする場合、GIM は、新規 K-TAP が含まれているバンドルを、他のデータベース・サーバーへの配布元とすることができる Guardium システムに自動的にコピーできます。

インストーラーが K-TAP モジュールの作成を試みる際、guard-ktap-loader によって発行されるメッセージが表示されます。例えば、以下のメッセージがあります。

- 作成を試みています (It is attempting to build)
- 作成が完了しました (The build has completed)
- K-TAP がロードされました (K-TAP has been loaded)
- カーネル開発パッケージが見つからないため、作成を行うことができません (The build cannot be attempted, because the kernel development package is not found)

親トピック: Linux システムおよび UNIX システム: K-TAP の処理

Linux システムおよび UNIX システム: 新規 K-TAP モジュールの他のシステムへのコピー
Linux システムおよび UNIX システム: GIM を使用した K-TAP モジュールのコピー

Linux システムおよび UNIX システム: GIM を使用した K-TAP モジュールのコピー

Linux データベース・サーバー用のカスタム K-TAP モジュールを作成する場合、GIM を使用して、そのモジュールを他の Linux データベース・サーバーにコピーできます。

始める前に

使用可能な何百もの Linux ディストリビューションが存在し、そのリストは増え続けています。このことは、ご使用の Linux ディストリビューションで K-TAP がまだ使用可能になっていない可能性があることを意味します。適切な K-TAP が使用可能でない場合、S-TAP インストール・プロセスにより K-TAP を作成することができます。

カスタム K-TAP モジュールは、現行カーネル用に事前作成された K-TAP がない Linux サーバー上に S-TAP をインストールするときに作成されます。カスタム K-TAP モジュールは、kernel-devel パッケージがインストールされている場合にのみ作成されます。S-TAP バンドルをインストールするときは、GIM UI を使用して、GIM パラメーター STAP_UPLOAD_FEATURE の値を 1 に設定してください。この設定は、カスタム K-TAP モジュールが作成された後、Guardium システムにそのモジュールをアップロードし、次に、カスタム S-TAP バンドルを自動的に作成するよう GIM クライアントに指示します。

GIM GUI または CLI を使用して、カスタム K-TAP が作成されたサーバーと同じ Linux ディストリビューションを実行しているその他のデータベース・サーバーに新規バンドルを配布します。

手順

1. GIM を使用して S-TAP を Linux データベース・サーバーにインストールします。 インストーラーは、カスタム K-TAP モジュールが必要と判断し、そのモジュールを作成します。 カスタム K-TAP モジュールは、S-TAP が構成されている Guardium システムに自動的にアップロードされます。 これは、GIM サーバーとして使用する Guardium システムと同じシステムではない場合があることに注意してください。その Guardium システムで、バンドル STAP が自動的に作成され、GIM GUI で表示されるようになります。バンドル番号には、_800 で始まり、バンドルが追加されるたびに 1 増える番号が付加されます (例: BUNDLE_STAP (10.1.4_r102728_800))。ロードされたバンドルは /var/gim_dist_packages に保管されます。 新規バンドルは、/var/dump に配置されます。
2. 新規バンドルが、ご使用の GIM サーバーではない Guardium システム上にある場合、新規バンドルを GIM サーバーにコピーします。

親トピック: Linux システムおよび UNIX システム: K-TAP の処理
新規 K-TAP モジュールの他のシステムへのコピー

Linux システムおよび UNIX システム: 新規 K-TAP モジュールの他のシステムへのコピー

Linux データベース・サーバー用の新しい K-TAP モジュールをビルドしたら、そのモジュールを同じ Linux ディストリビューションを稼働する他のデータベース・サーバーにコピーすることができます。

始める前に

Linux データベース・サーバー上で K-TAP モジュールをビルドし、テストした後、以下の手順を実行します。

このタスクについて

データベース・サーバー上でのエージェントの管理に Guardium Installation Manager (GIM) を使用している場合は、GIM を使用してモジュールをコピーします。使用する手順については、以下のリンクを参照してください。

手順

1. テスト済みの K-TAP のあるデータベース・サーバーにログインします。
2. /usr/local/guardium/guard_stap/ktap/current/ ディレクトリーに移動して、 ./guard_ktap_append_modules を実行し、ローカルにビルドしたモジュールを modules.tgz に追加します。
3. 更新された modules.tgz ファイルをターゲット・サーバーにコピーします。
4. ターゲット・サーバーにログインして、/usr/local/guardium/guard_stap/ktap/current/ ディレクトリーに移動します。
5. retry パラメーターと、更新された modules.tgz ファイルへの絶対パスを指定して、K-TAP ローダーを実行します。例:

```
guard_ktap_loader retry /tmp/modules-9.0.0_r55927_v90_1.tgz
```

6. S-TAP を再始動して、この新しい K-TAP モジュールに接続します。

タスクの結果

ターゲット・システム上で、カスタム K-TAP モジュールを使用する準備ができました。この K-TAP モジュールのデプロイ先となる対応する Linux システムごとに、この手順を繰り返します。

親トピック: Linux システムおよび UNIX システム: K-TAP の処理
Linux システムおよび UNIX システム: GIM を使用した K-TAP モジュールのコピー
Linux システムおよび UNIX システム: K-TAP パラメーター

Linux システムおよび UNIX システム: P-CAP がデフォルトでインストールされた場合のインストール後の K-TAP の有効化

インストール・プロセスの実行中に、K-TAP が正しくロードを行えない場合 (おそらく、ハードウェアまたはソフトウェアの非互換性が原因)、デフォルトの収集メカニズムとして P-CAP がインストールされます。互換性の問題が解決した後に K-TAP に切り替えるには、以下の手順を行います。

手順

1. K-TAP の問題をすべて解決します。例えば、モジュール要求を IBM に送信するか、カーネル開発パッケージをインストールします。
2. KTAP ディレクトリーでスクリプト guard_ktap_loader retry を実行します。
   - シェル/RPM インストール: guardium/guard_stap/ktap/current
   - GIM インストール: modules/KTAP/current

   K-TAP が正常にロードされると、guard_tap.ini パラメーター ktap_installed は自動的に 1 (はい) に設定されます。

親トピック: Linux システムおよび UNIX システム: K-TAP の処理

Linux システムおよび UNIX システム: 特別な環境での構成

ゾーン、RAC、WPAR、クラスターがあるシステムでは、以下の該当する手順を使用してください。

- Linux システムおよび UNIX システム: Solaris ゾーンの S-TAP 構成
  S-TAP をマスター・ゾーン (グローバル・ゾーン) にインストールして構成します。他のすべてのゾーン (ローカル・ゾーン) は、マスター・ゾーンとリソースを共有します。
- Linux システムおよび UNIX システム: Oracle RAC の S-TAP 構成
- Linux システムおよび UNIX システム: S-TAP for Db2 WPAR の構成
- Linux システムおよび UNIX システム: Db2 クラスターのすべてのノードでの A-TAP のアクティブ化
  A-TAP は、Db2 サーバーが Db2 クラスターのノードによって共有されているすべてのノードでアクティブにする必要があります。
- Linux システムおよび UNIX システム: 遅延クラスター・ディスク・マウントの構成
  このトピックは、Oracle、Informix®、および DB2® の各データベース・サーバーにのみ適用されます。

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのインストール

Linux システムおよび UNIX システム: Solaris ゾーンの S-TAP 構成

S-TAP をマスター・ゾーン (グローバル・ゾーン) にインストールして構成します。他のすべてのゾーン (ローカル・ゾーン) は、マスター・ゾーンとリソースを共有します。

このタスクについて

手順

1. ローカル・ゾーンはマスター・ゾーンの情報を共有するため、データベースが実行されているゾーンに関係なく、S-TAP をマスター・ゾーン (グローバル・ゾーン) にインストールします。
2. 検査エンジンを構成するとき、db_install_dir パスと tap_db_process_names にグローバル・ゾーン値を使用します。(S-TAP はグローバル・ゾーンからすべてのゾーン内のデータベースへのアクセスをモニターします。)
3. PCAP を使用している場合、モニターするすべてのゾーンの IP アドレスを、Solaris データベースの guard_tap.ini ファイル内の alternate_ips パラメーターに追加します。
4. インストールの終了時には、以下のようになります。
   - K-TAP は、グローバル・ゾーンにのみロードされるため、ローカル・ゾーンにはロードされません。それはローカル・ゾーンで表示できます。
   - S-TAP はローカル・ゾーンでは実行されません。

親トピック: Linux システムおよび UNIX システム: 特別な環境での構成

Linux システムおよび UNIX システム: Oracle RAC の S-TAP 構成

このタスクについて

Oracle RAC (Real Application: Clusters) を使用すると、複数のコンピューターが単一のデータベースにアクセスする一方で、Oracle RDBMS ソフトウェアを同時に実行でき、クラスタリングが行われます。

RAC 以外の Oracle データベースでは、単一データベースにアクセスするのは単一インスタンスです。データベースは、ディスク上に保管されたデータ・ファイル、制御ファイル、および再実行ログの集合で構成されます。インスタンスは、Oracle 関連メモリーと、コンピューター・システムで実行されるオペレーティング・システム・プロセスとの集合で構成されます。

Oracle RAC 環境では、2 つ以上のコンピューター (それぞれに Oracle RDBMS インスタンスが含まれています) が単一データベースに同時にアクセスします。これにより、アプリケーションまたはユーザーはいずれかのコンピューターに接続し、調整された単一のデータ・セットにアクセスできます。

手順

1. すべてのノードに S-TAP をインストールします。GIM が使用されている場合は、すべてのノードに GIM クライアントをインストールしてから、すべてのノードにバンドル S-TAP をインストールします。
2. STAP パラメーターを構成します。すべてのパラメーターは、GIM UI を使用して構成できます。
   - STAP_TAP_IP: ノード用に構成されたパブリック IP
   - STAP_ALTERNATE_IPS: ノード用に構成された VIP (仮想 IP) のコンマ区切りリストと、スキャン・リスナー
     ヒント: alternate_ips に入れる仮想ホスト名の値を取得するには、コマンド su – grid –c 'cat $ORACLE_HOME/network/admin/*.ora'|grep –i host を使用してください。

     例:

     ```
     [root@racvm121 ~]# su - grid -c 'cat $ORACLE_HOME/network/admin/*.ora'|grep -i host
     LISTENER_RACVM121=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=racvm121-vip.guard.swg.usma.ibm.com)(PORT=1521)
     ```

- STAP 検査エンジン・パラメーター unix_domain_socket_marker=<key> を構成します。<key> 値は、IPC プロトコル定義内の listener.ora にあります。
  ヒント: unix_domain_socket の値を取得するコマンドは、`su - grid -c 'cat $ORACLE_HOME/network/admin/*.ora'|grep -i KEY` です。
  - 例: listener.ora にある記述が LISTENER=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=ORCL)))) である場合、unix_domain_socket_marker=ORCL です。
  - 例: listener.ora 内に複数の IPC 行がある場合、すべてのキーの共通項を使用します。

    ```
    su - grid -c 'cat $ORACLE_HOME/network/admin/*.ora'|grep -i KEY
    LISTENER=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER))))
    LISTENER_SCAN1=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER_SCAN1))))
    LISTENER_SCAN2=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER_SCAN2))))
    LISTENER_SCAN3=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER_SCAN3))))
    ```

    Guardium はパスで文字列検索を使用します。「LISTENER」は上記の 4 つすべてで機能します。この場合は、unix_domain_socket_marker=LISTENER を使用してください
  - 例: 共通項がない場合は、特定の IPC キーに対応する unix_domain_socket_marker を使用して追加の検査エンジンを作成します。例えば、guard_tap.ini は、末尾でこの例に似ている場合があります。

    ```
    [DB_0]
    ...
    unix_domain_socket_marker=EXTPROC1522
    ...
    [DB_1]
    ...
    unix_domain_socket_marker=LISTENER
    ```

3. Oracle データベースが暗号化 (ASO/SSL) されている場合は、すべてのノード (アクティブとスタンバイ) で ATAP をアクティブ化します。
   a. すべての Oracle サービス (Clusterware を含む) を停止し、ohasd.bin がダウンしていることを確認します。
      i. `crsctl stop cluster -all` を実行します。
      ii. `ohasd.bin` がダウンしていることを確認します。
   b. ユーザーに Oracle およびグリッドを許可します (リスナーがユーザー・グリッドに属している場合)。
   c. A-TAP パラメーターを構成します。
   d. A-TAP をアクティブにします。
   e. クラスター内のすべての Oracle サービスを開始します。
4. Oracle RAC 環境では、どのユーザーがリスナーを開始するかを検査します。それがユーザー・グリッドに関するものであれば、ユーザーにグリッドを許可します。

親トピック: Linux システムおよび UNIX システム: 特別な環境での構成
関連資料:
Linux システムおよび UNIX システム: A-TAP の guardctl ユーティリティー・コマンド
Linux システムおよび UNIX システム: データベース固有の guardctl パラメーター

Linux システムおよび UNIX システム: S-TAP for Db2 WPAR の構成

このタスクについて

スクリプト **find_db2_shmem_parameters.sh** の出力は、検査エンジンで定義されている Db2 共有メモリー・パラメーターをリストします。これは任意のディレクトリーから実行できます。パラメーターとして db2 インスタンス名が必要です。

ktap_fast_shmem が 1 に設定され、guard_tap.ini ファイル内で 1 つの WPAR に対して複数の Db2 インスタンスが構成されていて、これらの Db2 インスタンスの db2_shmem_size が同じである場合、その WPAR の最初の Db2 セクションから db2_fix_pack_adjustment と db2_shmem_client_position が取得されます。したがって、WPAR 上で複数の Db2 インスタンスが実行されている場合は、以下のようになります。

- すべての Db2 インスタンスの db2_shmem_size、db2_fix_pack_adjustment、および db2_shmem_client_position が同じである場合は、構成されているインスタンスが 1つだけであっても、すべてのインスタンスからのパケットが収集されます。
- すべての Db2 インスタンスで db2_shmem_size は同じであるが、db2_fix_pack_adjustment または db2_shmem_client_position が異なる場合は、最初に構成された Db2 インスタンスからのパケットのみが収集されます。

手順

1. 「find_db2_shmem_parameters.sh <instance name>」という構文を使用して、任意のディレクトリーから **find_db2_shmem_parameters.sh** スクリプトを実行します。出力には、検査エンジンで定義されている Db2 共有メモリー・パラメーターがリストされます。
2. Db2 共有メモリー・トラフィックをキャプチャーするには、以下のパラメーターを設定します。

表 1. Db2 パラメーター

パラメーター	STAP 名	ATAP 名
パケット・ヘッダー・サイズ	db2_fixed_pack_adjustment	db2_header_offset
クライアント入出力域オフセット	db2_shmem_client_position	db2_c2soffset
DB2® 共有メモリー・セグメント・サイズ	db2_shmem_size	db2_shmsize

親トピック: Linux システムおよび UNIX システム: 特別な環境での構成

Linux システムおよび UNIX システム: Db2 クラスターのすべてのノードでの A-TAP のアクティブ化

A-TAP は、Db2 サーバーが Db2 クラスターのノードによって共有されているすべてのノードでアクティブにする必要があります。

手順

1. ノード 1 で Db2 ユーザーを許可します。<guardium_base>/xxx/guardctl authorize-user <user-name>
   例:

   ```
   # /usr/local/guardium/bin/guardctl authorize-user db2inst1

   # /usr/local/guardium/bin/guardctl is_user_authorized db2inst1
   ```

   ユーザー「db2inst1」が許可されます。

2. ノード 1 で A-TAP をアクティブにします。
   <guardium_base>/xxx/guardctl db_instance=<instance> activate
   例:

   ```
   # /usr/local/guardium/guard_stap/guardctl db_instance=db2inst1 activate

   # /usr/local/guardium/guard_stap/guardctl list-active
   db2inst1
   ```

3. ノード 1 で ATAP をアクティブにした後でノード 1 で元の Db2 サーバーを復元します。これにより、他のノードが ATAP をアクティブにできるようにします。(すべてのノードは実行可能ファイルを共有します。(db2 adm ディレクトリーで、db2sysc-guard-original を db2sysc にコピーします (最初にそれぞれのコピーを作成し、保管してください))。例:
   ```
   # > cp db2sysc-guard-original db2sysc
   ```
4. db2sysc-guard-original を削除します (そうしないと、ノード 2 でアクティベーションが失敗します)。例:
   ```
   # rm -rf db2sysc-guard-original
   ```
5. クラスター・リソースをノード 2 に移動します。例:
   ```
   # pcs resource move resource_id <destination node>
   ```
6. ノード 2 で Db2 ユーザーを許可し、アクティブにします (ステップ 1 とステップ 2)。これにより、ノード 2 でライブラリーが作成され、削除された db2sysc-guard-original が置換されます。 現在の状況は以下のようになります。
   Node01:

   ```
   # /usr/local/guardium/guard_stap/guardctl list-active

   db2inst1
   ```

   Node02:

   ```
   # /usr/local/guardium/guard_stap/guardctl list-active

   db2inst1
   ```

親トピック: Linux システムおよび UNIX システム: 特別な環境での構成

Linux システムおよび UNIX システム: 遅延クラスター・ディスク・マウントの構成

このトピックは、Oracle、Informix®、および DB2® の各データベース・サーバーにのみ適用されます。

これらのデータベース・タイプでは、S-TAP は開始時に、データベース・ホームへのアクセス権限を持っている必要があります。ご使用の環境でクラスタリング・スキームを使用しており、パッシブ・ノードではなく、アクティブ・ノードにマウントされている単一ディスクを複数のノードが共有している場合、パッシブ・ノード上ではフェイルオーバーが発生するまでデータベース・ホームを使用できません。

親トピック: Linux システムおよび UNIX システム: 特別な環境での構成

Linux システムおよび UNIX システム: S-TAP のインストールまたはアップグレード後に再始動またはリブートが必要なもの

このトピックでは、S-TAP のインストールまたはアップグレード後にリブートまたは再始動が必要なものについて詳しく説明します。再始動およびリブートの要件は、GIM による実装と GIM を使用しない実装のどちらの場合も同じです。

EXIT の使用時に UNIX/Linux S-TAP のインストール後に再始動が必要なもの
- Teradata: データベースを再始動する必要があります
- Db2: データベースを再始動する必要があります
- Informix: 再始動は不要です。ifxserver が実行中の場合は再始動します。ifxserver が実行中でない場合は、再始動する必要はありません。

A-TAP の使用時に UNIX/Linux S-TAP のインストール後に再始動が必要なもの
- A-TAP を使用する場合、データベースを再始動する必要があります。
- データベース・ソフトウェアを更新する前にはいつでも、A-TAP の非アクティブ化およびインスツルメンテーションの削除を行う必要があります。

K-TAP の使用時に UNIX/Linux S-TAP のインストール後に再始動が必要なもの

OS / データベース	**Oracle**		**Db2**		**Sybase**		**MS-SQL**		**Informix**	
	TPC/IPC	SHM	TPC/IPC	SHM	TPC/IPC	SHM	TPC/IPC	SHM	TPC/IPC	SHM
Linux	NR	NR	NR	REQ	NR	NR	NR	NR	NR	REQ
AIX	REQ	NR	REQ	NR	REQ	NR	NA	NR	REQ	NR
Solaris	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR
HP-UX	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR

NR = 再始動/リブートは不要 (ライブ・アップデート・メカニズムの使用と、ライブ・アップデート・リンクの参照 (ライブ・アップデート・リンクがある場合) に基づく)
REQ = 再始動が必要
NA = 適用されない

UNIX/Linux の S-TAP のライブ・アップグレード後に再始動が必要なもの
A-TAP および出口ライブラリーを含まないライブ・アップグレードでは、再始動はまったく不要です。
S-TAP をアップグレードする前にはいつでも、A-TAP の非アクティブ化およびインスツルメンテーションの削除を行う必要があります。
v10.6.0 以上からのアップグレード: 出口ライブラリーを使用するデータベースを再始動する必要はありません。
v10.5 以前からのアップグレード: 出口ライブラリーを使用するデータベースを再始動する必要があります。

リブートのガイドライン
データベース・サーバーをリブートする必要があるのは、K-TAP をアンインストールする場合だけ (K-TAP が使用中であるかどうかに関係なく) です。

親トピック: Linux システムおよび UNIX システム: S-TAP のインストール、アップグレード、およびアンインストール

Linux システムおよび UNIX システム: S-TAP エージェントのアンインストール

古い構成ファイルを保存する必要がある場合、S-TAP の新規バージョンをインストールする前にこの手順を実行します。

このタスクについて

以前に S-TAP® がインストールされている場合は、/usr/local/guardium/guard_stap という名前のディレクトリーがあります。

A-TAP がインストールされている場合は、アップグレード/インストール操作を行う前に、それを非アクティブにする必要があります。『Linux システムおよび UNIX システム: A-TAP の非アクティブ化』の A-TAP 非活動化コマンドの説明を参照してください。

K-TAP を使用していた S-TAP の旧バージョンを削除している場合は、データベース・サーバーをリブートする必要があります。K-TAP がインストールされている場合は、/dev/guard_ktap という名前のデバイスがあります。

手順

1. root アカウントを使用して、データベース・サーバー・システムにログオンします。
2. オプションで、S-TAP 構成ファイルを安全な場所 (Guardium 以外のディレクトリー) にコピーします。 デフォルトでは、絶対パス名は /usr/local/guardium/guard_stap/guard_tap.ini となります。 このファイルは、このソフトウェア・バージョンを再インストールする必要がある場合に後で使用することも、S-TAP の更新されたバージョンを構成する際に参照することもできます。 古い構成ファイルは、新バージョンのソフトウェアでは絶対に直接使用しないでください。新しいプロパティーが欠落している可能性があり、デフォルトを使用することにより、S-TAP の始動時に予期しない動作になることがあります。
3. アンインストール・スクリプトを実行します。例えば、デフォルト・ディレクトリーが使用されている場合は、以下のようにします。`[root@yourserver ~]# /usr/local/guardium/guard_stap/uninstall`
4. 前のバージョンの S-TAP に K-TAP が含まれていた場合は、ここでデータベース・サーバーのリブートを行います。

a. アンインストール・スクリプトを再実行します。

5. このステップは、AIX® WPAR と Solaris ゾーンにのみ適用されます (その他すべての場合はスキップします)。K-TAP が含まれていた前のバージョンの S-TAP をアンインストールする場合は、マスター・ノードから次のコマンドを実行します: rm -f /wpars/<server>/dev/ktap* および rm -f /wpars/<server>/dev/guard_ktap*。ここで、/wpars/<server> はマスター・ノードから WPAR へのパスです。

親トピック: Linux システムおよび UNIX システム: S-TAP のインストール、アップグレード、およびアンインストール

Linux システムおよび UNIX システム: S-TAP エージェントのアップグレード

アップグレードのワークフローは、ご使用のモニター・メカニズムによって異なります。

- Linux システムおよび UNIX システム: S-TAP と K-TAP のアップグレード
  S-TAP をアップグレードして、リブートせずに、既存のセッションからのデータのキャプチャーを続行し、その構成を保守します。S-TAP アップグレードの一環として、K-TAP もアップグレードできます。
- Linux システムおよび UNIX システム: A-TAP を使用するデータベースでの S-TAP のアップグレード
  A-TAP モニターを使用する S-TAP をアップグレードするときはこのワークフローを使用します。
- Linux システムおよび UNIX システム: 出口ライブラリーを使用するデータベースでの S-TAP のアップグレード
  データベースが出口ライブラリー (Db2、Informix、Teradata) を使用する S-TAP をアップグレードするためのフローについて説明します。

親トピック: Linux システムおよび UNIX システム: S-TAP のインストール、アップグレード、およびアンインストール

Linux システムおよび UNIX システム: S-TAP と K-TAP のアップグレード

S-TAP をアップグレードして、リブートせずに、既存のセッションからのデータのキャプチャーを続行し、その構成を保守します。S-TAP アップグレードの一環として、K-TAP もアップグレードできます。

このタスクについて

- S-TAP をアップグレードする前に、S-TAP ホストとして機能する Guardium システムをアップグレードします。
- K-TAP を使用していた S-TAP® の旧バージョンを削除している場合は、データベース・サーバーをリブートする必要があります。
- K-TAP がインストールされている場合は、/dev/guard_ktap という名前のデバイスがあります。

GIM を使用して S-TAP をアップグレードする場合、回避する必要がある 2 つのシナリオがあります。1 つは A-TAP ユーザーがアクティブである状況 (アップグレードが失敗する原因となります)、もう 1 つは DB で保守が実行されている状況です。代表的なシナリオは、午前 0 時にデータベース・サーバーの保守が計画されていて、午前 1 時に S-TAP をアップグレードするような場合です。S-TAP を正常にアップグレードするために、以下の一般的なフローを使用してください。

1. GIM を使用してアップグレードをスケジュールします。
2. GIM のアップグレードのデプロイを無効にします。configurator.sh --delayed_bundle_deployment enable
3. データベースの保守が完了するまで待ちます。

4. ATAP ユーザーがアクティブであるかどうかを検査します (DB が停止していることを想定して、それらが存在する場合は無効にします)。<GIM INSTALL DIR>/ATAP/current/files/bin/guardctl list-active
5. GIM のアップグレードを有効にします。configurator.sh --delayed_bundle_deployment disable

手順

1. root アカウントを使用して、データベース・サーバー・システムにログオンします。
2. システムに A-TAP があり、暗号化ボックスが使用されない場合、次のようにします。
   a. データベースを停止します。
   b. guardctl を使用して A-TAP を非アクティブにします。
3. システムに A-TAP があり、暗号化ボックスが使用される場合、DB を停止します。(DB の実行中は常に、ATAP がアクティブになります。暗号化ボックスは、ATAP を自動的にアクティブにするように設定されると、単にボックスのチェック・マークを外すことでは無効にできません。設定をクリアするには、機能を無効にしてシステムをリブートする必要があります。)
4. GIM またはシェル・インストーラーを使用してライブ更新を実行する前に、S-TAP 以外のプロセスが K-TAP デバイスを使用していないことを確認してください。S-TAP が実行中であり、A-TAP が非アクティブ化されている必要があります。fuser /dev/ktap_xxx または lsof | grep ktap_xxx (ここで、xxx は古いバージョン番号) を実行し、デバイスを開いているプロセスがあるかどうかを確認します。 これを実行しないと、予期しない動作になる可能性があります。
5. S-TAP バージョン 6.0 以降をアンインストールする場合:
   a. Red Hat Enterprise Linux 6 の場合: stop utap コマンドを使用して S-TAP を停止します。
   b. For Red Hat Enterprise Linux 7 の場合: systemctl stop guard_utap コマンドを使用して S-TAP を停止します。
   c. その他すべての場合:
      i. /etc/inittab ファイル内の UTAP エージェント項目を削除します (コメント化されているかどうかは関係ありません)。デフォルト・インストールでは、このステートメントは以下のようになります。utap:<nnnn>:respawn:/usr/local/guardium/guard_stap/guard_stap /usr/local/guardium/guard_stap/guard_tap.ini
      ii. /etc/inittab ファイルを保存します。
      iii. init q コマンドを実行します。
   d. ps - ef | grep stap を実行して、S-TAP がもう実行されていないことを確認します。
6. S-TAP 構成ファイルを安全なロケーション (Guardium 以外のディレクトリー) にコピーします。
7. アンインストール・スクリプトを実行します。例えば、デフォルトのディレクトリーを使用すると、次のようになります。[root@yourserver ~]# /usr/local/guardium/guard_stap/uninstall
   注: S-TAP が実行されている状態では、アンインストール・プログラムを実行しないでください。S-TAP が停止されていることを確認してください。
8. 前のバージョンの S-TAP に K-TAP が含まれていた場合は、ここでデータベース・サーバーのリブートを行います。
9. HP-UX サーバーのみ (他のすべてのサーバーの場合はスキップ): K-TAP が含まれている前のバージョンの S-TAP をアンインストールする場合、リブート後にアンインストール・スクリプトを再度実行します。
10. AIX WPAR のみ (他のすべての場合はスキップ): K-TAP が含まれている前のバージョンの S-TAP をアンインストールする場合、アンインストール後にマスター・ノードでコマンド rm -f /wpars/<server>/dev/ktap* と rm -f /wpars/<server>/dev/guard_ktap* を発行します。ここで、/wpars/<server> はマスター・ノードから WPAR へのパスです。
11. 次のいずれかの方法を使用して、S-TAP をアップグレードします。
    - Linux システムおよび UNIX システム: GIM の「クライアント別の設定」を使用した S-TAP クライアントのインストール。手順の最後に「アップグレード」オプションを使用します。
      - K-TAP をアップグレードする場合は、KTAP_LIVE_UPDATE を yes に設定します。 他のパラメーターを必要に応じて変更します。変更しないパラメーターは、アップグレードで引き継がれます。
      - ATAP がアクティブの場合に、バンドル S-TAP のインストールを遅らせるには、GIM スクリプト configurator.sh --delayed_bundle_deployment enable を実行します。A-TAP ユーザーが非アクティブになるのを GIM が待機する間、GIM は他のインストールやパラメーター更新の要求を処理しません。
    - Linux システムおよび UNIX システム: RPM を使用した S-TAP のインストール、アンインストール、および更新。Linux システムおよび UNIX システム: S-TAP インストール・スクリプトのパラメーターにリストされている -u フラグや他の関連アップグレード・パラメーターを使用します。K-TAP をアップグレードするには、--live_update Y を指定します。
    - Linux システムおよび UNIX システム: シェル・インストーラーを使用した S-TAP のインストール
12. K-TAP ライブ・アップグレードの後は、以下のようになります。
    - K-TAP をアップグレードした後の既存のセッションに対する最初の SQL は、取り込まれません。
    - Solaris ローカル・ゾーン上の既存の A-TAP セッションはログに記録されません。
    - 一部のプロセスでは、古い K-TAP モジュール内のメモリーを引き続き参照する場合があります。このシナリオでは、将来不安定にならないように、モジュールはリソースを解放しません。 このような場合、ユーザーは、それらのリソースが使用されなくなった後で、guard_ktap_cleanup (ktap ディレクトリーにあります) を実行して、手動でクリーンアップする必要があります。
    - HP-UX 11.11 では、古い K-TAP モジュールはインストールされなくなりますが、kmadmin -s | grep tap を実行したときに、引き続き登録済みとして表示されます。kmmodreg -U ktap_<version> を使用して、このモジュールを手動で登録抹消してください。
    - Solaris および AIX® では、リブート後に古いデバイス・ノードが自動的に削除されないので、それらを手動で削除する必要があります。

    例外:
    - GIM を介してインストールされなかったバージョンの DB サーバーがインストールされていて、その非 GIM K-TAP のバージョンが、インストールされる K-TAP のバージョンと同じではない場合、KTAP_LIVE_UPDATE の値は無視されます。これは、非 GIM バージョンからのアップグレードではシステム・リブートが必要であるためです。
    - 非 GIM バージョンから同じ GIM バージョンへのアップグレード時には、システムをリブートする必要はありません。
    - マシンをリブートせずに、前にインストールした K-TAP バージョンを再インストールすることはできません。

    エラー処理:
    - 障害が発生した場合、障害によっては、完全なリカバリーのためにシステムのリブートが必要になるため、「GIM イベント・リスト」レポートを確認することが極めて重要です。

    注: ODMDIR 環境が定義されていない場合、S-TAP のインストールまたはアップグレード時に AIX 用の K-TAP のみがロードに失敗します。ODMDIR とは、オブジェクト・データ・マネージャー・ディレクトリーを指します。ODM は、OS に統合されるシステムおよびデバイスの構成情報が含まれるデータベースです。これは、システム情報、ソフトウェア情報、およびデバイス情報を格納するためのものです。 すべての ODM コマンドでは、/etc/environment ファイルに設定されている ODMDIR 環境変数が使用されます。ODMDIR のデフォルトの値は /etc/objrepos です。

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのアップグレード
関連資料:
Linux システムおよび UNIX システム: A-TAP の guardctl ユーティリティー・コマンド

Linux システムおよび UNIX システム: A-TAP を使用するデータベースでの S-TAP のアップグレード

A-TAP モニターを使用する S-TAP をアップグレードするときはこのワークフローを使用します。

手順

1. データベースを停止します。
2. すべての ATAP のインスツルメンテーションの削除および非アクティブ化を行います (guardctl を使用します)。
3. S-TAP をアップグレードします。
4. すべての A-TAP でインスツルメンテーションを実行してアクティブ化します。
5. データベースを開始します。

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのアップグレード

Linux システムおよび UNIX システム: 出口ライブラリーを使用するデータベースでの S-TAP のアップグレード

データベースが出口ライブラリー (Db2、Informix、Teradata) を使用する S-TAP をアップグレードするためのフローについて説明します。

手順

1. データベースを停止します。
2. S-TAP をアップグレードします。
3. 新しい出口ライブラリーを、新しい STAP からデータベース用の適切な場所にコピーします。
4. データベースを再始動します。

親トピック: Linux システムおよび UNIX システム: S-TAP エージェントのアップグレード

Linux システムおよび UNIX システム: データベースをアップグレードする際の S-TAP の管理

以下の指針を使用して、データベースのアップグレード時に UNIX S-TAP を管理します。

手順

1. A-TAP を使用している場合: その A-TAP のインスツルメンテーションの削除および非アクティブ化を行います。
2. データベースをアップグレードします。
3. A-TAP を使用している場合: その A-TAP でインスツルメンテーションを実行してアクティブ化します。
4. 出口を使用している場合: 出口ライブラリーが適切な場所にあることを確認します (新しい DB ロケーション・ディレクトリーがある場合など)。
5. データベースの検査エンジンが正しいかどうか検査します (バージョン番号など)。
6. IE を変更した場合は S-TAP を再始動します。

親トピック: Linux システムおよび UNIX システム: S-TAP のインストール、アップグレード、およびアンインストール

Linux システムおよび UNIX システム: データベースのオペレーティング・システムをアップグレードする際の S-TAP の管理

以下のガイドラインを使用して、データベースのオペレーティング・システム (OS) のアップグレード時に、シェルまたは RPM を使用してインストールされた S-TAP を管理します。

このタスクについて

このタスクは、シェルまたは RPM を使用してインストールされた S-TAP エージェントのみに関連しています。GIM を使用してインストールされた S-TAP エージェントについては、データベース・サーバーのオペレーティング・システムをアップグレードするときを参照してください。

手順

1. S-TAP エージェントをアンインストールします。
2. データベースのオペレーティング・システムをデータベースをアップグレードします。
3. アップグレードされたデータベース・オペレーティング・システム用の S-TAP インストーラーを Fix Central からダウンロードして、インストールします。

親トピック: Linux システムおよび UNIX システム: S-TAP のインストール、アップグレード、およびアンインストール
関連概念:
Linux システムおよび UNIX システム: S-TAP エージェントのインストール
関連タスク:
Linux システムおよび UNIX システム: S-TAP エージェントのアンインストール

Linux システムおよび UNIX システム: S-TAP の構成

S-TAP の構成について説明します。

- Linux システムおよび UNIX システム: GUI からの S-TAP の構成
  この Guardium システムで管理されるすべての S-TAP を表示したり、個々の S-TAP を管理したり、すべての S-TAP に対するいくつかの操作を実行したりします。
- Linux システムおよび UNIX システム: データベース・インスタンスのディスカバー
  S-TAP が定期的にデータベース・インスタンスを検出し、現在のアクティブな S-TAP システムにその結果を送信できるようにします。
- Linux システムおよび UNIX システム: 検査エンジンの構成
  「S-TAP 制御」ペインで検査エンジンを構成または変更します。
- Linux システムおよび UNIX システム: 検査エンジンの検査
  S-TAP 検査では、ご使用の環境の STAP とのその検査エンジンが実行されていて、データベース・アクティビティーをアクティブにモニターしていることを確認します。検査について理解し、S-TAP を定期的に検査するためのスケジュールを定義します。
- Linux システムおよび UNIX システム: S-TAP のロード・バランシング・モデルと構成ガイドライン
  S-TAP のロード・バランシング・モデルについて理解し、セットアップに適したモデルを選択してください。
- Linux システムおよび UNIX システム: SSL 証明書を使用する S-TAP 認証のセットアップ
  S-TAP サーバーと Guardium システムの間の認証をセットアップします。
- Linux システムおよび UNIX システム: Kerberos 認証データベース・トラフィック
  Kerberos は、ネットワーク上で暗号化されていないパスワードの伝送を排除するネットワーク認証プロトコルです。ここでは、これが Guardium で機能する仕組みを説明します。
- Linux システムおよび UNIX システム: A-TAP の管理
  A-TAP は application-level tapping です。A-TAP はアプリケーション層に配置され、暗号化されたデータベース・トラフィックのモニターをサポートします。このモニターは、K-TAP によってカーネルで実行することはできません。
- Linux システムおよび UNIX システム: 出口ライブラリーの使用
  出口ライブラリーは、出口メカニズムを使用して Guardium ライブラリーをデータベースに組み込みます。出口ライブラリー、つまり出口モジュールは、Guardium S-TAP と直接通信してデータベース・トラフィックを転送します。
- Linux システムおよび UNIX システム: FileAppender に記録するための Cassandra 監査の構成
  ネイティブ監査ロギングでの Cassandra/Datastax のファイル・アペンダーへのロギングを構成します。
- Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集
  S-TAP 構成は、インストール後に GIM または GUI を使用して変更できます。上級者の場合は、データベース上の構成ファイルで変更できます。

親トピック: Linux システムおよび UNIX システム: S-TAP ユーザーズ・ガイド

Linux システムおよび UNIX システム: GUI からの S-TAP の構成

この Guardium システムで管理されるすべての S-TAP を表示したり、個々の S-TAP を管理したり、すべての S-TAP に対するいくつかの操作を実行したりします。

このタスクについて

前提条件: S-TAP のアクティブ・ホストである Guardium システムにログインする必要があります。

ユーザーが、S-TAP のインストール・プロセス中に決定できないことがあったり、誤った決断をして、それがインストール・プロセスの完了後に検出されたりする場合があります。例えば、SQL Guard IP を定義する際に、IP アドレスを入力し忘れたり間違った IP アドレスを使用したりすることがあります。 このような誤りは、S-TAP 構成を変更することにより修正できます。

GUI のパラメーターは安全に変更できます。GUI に含まれていないパラメーターは変更の必要がほとんどないため、通常は未変更のままにしてください。これらは Guardium 技術サポートまたは上級者によって使用されます。GUI または GIM で S-TAP パラメーターを変更すると、S-TAP はパラメーター値を保存する前に検査を実行します。誤っていることが確認された値は保存されません。S-TAP は S-TAP のオンライン状態を維持するために、誤っていることが確認された値の修正を試みます。例えば、ポート範囲開始がポート範囲終了よりも大きい場合は、終了として構成された値を開始として設定します。S-TAP はパラメーター値を変更すると、LOG_CONF_ERR を S-TAP イベント・ログに記録します。をクリックして S-TAP イベント・ログを開き、各 LOG_CONF_ERR エラーを評価します。S-TAP によって行われた変更を受け入れるか、S-TAP が受け入れる値に値を変更すると、状況は緑になってタイム・スタンプが更新されます。S-TAP は、構成を修正する際にバックアップの guard_tap.ini を作成します。これは、S-TAP ディレクトリーの下に guard_tap.ini.bak として保存されます。

構成の変更内容によっては、パラメーターの記述で示されるように、手動で S-TAP エージェントを再始動する必要があります。

S-TAP を Guardium Installation Manager (GIM) を使用してインストールした場合、GIM GUI または API を使用して一部のパラメーターを更新することができます。

S-TAP 状況は次のいずれかになります。

- 緑: オンライン
- 黄: 構成エラー (LOG_CONF_ERROR を含む) を示します。構成に設定の誤りがあり、対処されたか、システムに対して正確または不正確なデフォルト値が入力されていたか、コレクターに接続するために構成が実行可能でなかったために構成がバックアップにロールバックされています。
- 赤: オフライン

手順

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」をクリックして、「S-TAP 制御」を開きます。
2. このページで、すべての S-TAP に対する操作を実行します。
   - リフレッシュ: S-TAP の表示をリフレッシュします。
   - スケジュールにすべて追加: 表示されているすべての S-TAP を S-TAP 検査のスケジュールに追加します。Linux システムおよび UNIX システム: 検査エンジンの検査を参照してください。
   - スケジュールからすべて削除: 表示されているすべての S-TAP を S-TAP 検査のスケジュールから削除します。
   - コメント: コメントを追加します。『コメント』を参照してください。
3. S-TAP の IP アドレス、または S-TAP がインストールされているデータベース・サーバーのシンボリック・ホスト名で、構成する S-TAP を特定します。個々の S-TAP に対する操作を表示して実行します。

オプション	説明

オプション	説明
削除:	S-TAP を除去するには、「削除」をクリックします。 S-TAP の削除は、ある S-TAP が非アクティブになったことが分かった場合、またはその S-TAP の構成ファイルで Guardium 装置がホストとしてリストされなくなった場合に、表示をクリーンアップする際に役立ちます。どちらの場合でも、S-TAP を削除しない限り、S-TAP はオフライン状況で無期限に表示されます。 アクティブな S-TAP はリストから削除できません。削除をクリックしても、S-TAP は情報の送信を停止せず、その S-TAP の構成ファイルに保管されているホストのリストからこの Guardium ホストが削除されることもありません。
リフレッシュ:	「リフレッシュ」をクリックすると、S-TAP 構成の最新のコピーがエージェントから取り出されます。(S-TAP の表示は自動でリフレッシュされません。)
送信コマンド:	「S-TAP コマンド」ポップアップが開きます。ここから各種コマンドを S-TAP ホストに対して実行できます。 ○ 再始動: S-TAP を再始動します。通常は必要ありません。再始動が必要な場合は、データベース・サーバーから停止できます。 ○ S-TAP ロギング ○ バッファーの再初期化: K-TAP 統計をリセットし、S-TAP バッファーを削除します。 ○ KTAP ロギング: S-TAP ロギングと同様、KTAP からのデバッグ出力を増やします。 ○ 診断の実行: S-TAP の診断スクリプトを実行します (その後、結果を Guardium システムにアップロードします)。 ○ Linux モジュールのアップロード: Linux のみ。K-TAP のローカル・カスタム・ビルド・モジュールをアップロードします。 ○ リプレイ・ログの記録: すべてのデータを DB サーバー上のファイルに記録し (RECORD)、データをコレクターに送信します (REPLAY)。 ○ 無視の取り消し: 取り消し可能な無視ポリシーによって無視されたすべてのセッションの無視が取り消され、それらのセッションのトラフィックのキャプチャーが再度開始されます。 ○ データベース・インスタンス・ディスカバリーの実行: ディスカバリー・プロセスを直ちに 1 回実行します。(自動実行が有効になっている場合、デフォルトでは 24 時間ごとに実行されます。)
S-TAP 構成の編集:	「S-TAP 構成」ウィンドウが開きます。GUI に表示されないパラメーターは拡張パラメーターです。上級者でない場合、または Guardium 技術サポートから変更するよう指示されていない場合は、それらのパラメーターを変更しないでください。以下の GUI パラメーターを参照してください。 ○ Linux システムおよび UNIX システム: 一般パラメーター ○ Linux システムおよび UNIX システム: 構成監査システム (CAS) パラメーター ○ Linux システムおよび UNIX システム: アプリケーション・サーバー・パラメーター ○ Linux システムおよび UNIX システム: Guardium ホスト (SQLGuard) パラメーター ○ Linux システムおよび UNIX システム: ファイアウォール・パラメーター ○ Linux システムおよび UNIX システム: 検査エンジン・パラメーター
S-TAP イベント・ログを表示:	クリックすると S-TAP イベント・ログが開きます。ここで、接続、切断、GIM サーバー構成のイベントを確認できます。このログは、トラブルシューティングに非常に役立ちます。上記のとおり、構成エラー (LOG_CONF_ERR) を特定するには、イベント・ログを使用します。S-TAP によって行われた変更を受け入れる場合は「同意する」をクリックし、パラメーター値を修正できる変更ウィンドウを開くには「変更」をクリックします。
「スケジュールに追加」チェック・ボックス	個々の S-TAP をスケジュールが設定された検査に追加します。
「無視されたセッションをすべて取り消す」チェック・ボックス	データベースは多数のセッションを実行している可能性があり、そのうちのいくつかは現在無視されています。そのサーバーからのトラフィックの無視を停止するには、このオプションをクリアします。

親トピック: Linux システムおよび UNIX システム: S-TAP の構成

Linux システムおよび UNIX システム: データベース・インスタンスのディスカバー

S-TAP が定期的にデータベース・インスタンスを検出し、現在のアクティブな S-TAP システムにその結果を送信できるようにします。

Guardium のディスカバリー・エージェントは、データベース・サーバーに S-TAP パッケージとともに自動的にインストールされるソフトウェア・エージェントです。インスタンス・ディスカバリー・エージェントは、データベース・インスタンス、リスナー、およびポートの情報を Guardium システムに報告します。ディスカバリーは、サーバー上の DB インスタンスのすべての詳細を検出して報告するわけではありません。

オートディスカバリーはデフォルトで有効になっています。guard_tap.ini パラメーター discovery_interval を使用して、実行間隔を構成します。

S-TAP ディスカバリーでサポートされるデータベース・タイプ
Oracle、Db2、Informix、MySQL、PostgreSGL、Enterprise PostgreSQL、Sybase、Hadoop、Teradata、Netezza、MemSQL。

ディスカバリー・バンドルは、スレーブ・ゾーンや WPAR にはインストールされません。グローバル・ゾーンで実行されるディスカバリー・エージェントが、他のゾーンの情報も収集します。
注: Solaris ゾーン・アーキテクチャーでは、Db2 インスタンスがスレーブ・ゾーンで実行されている場合、ディスカバリーは Db2 共有メモリー・パラメーターをディスカバーしません。

新たにディスカバーされたデータベース・インスタンスは「ディスカバーされたインスタンス」レポートで確認できます。このレポートから、「アクション」メニューを使用して、データ・ソースや検査エンジンを Guardium に素早く追加できます。

データベース・サーバー上のデータベースが作動可能 (開始済み) ではない場合や、後から追加された場合でも、ディスカバリー・エージェントは、STAP 制御ウィンドウからディスカバリー・エージェントの実行コマンドを実行 (「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」。 をクリックし、「データベース・インスタンス・ディスカバリーの実行」を選択) することで、これらのインスタンスをディスカバーできます。

S-TAP のディスカバリーは手動で実行できますが、このアクションは推奨されません。 手動で実行する主な理由は、デバッグのためです。スケジュールされたディスカバリーの実行中に、ユーザー・インターフェースから新しい要求が届いた場合、その新しい要求は無視されます。

データベース・サーバー上のローカル・コマンド・ラインからディスカバリーを実行するには (/usr/local/guardium/guard_stap/guard_discovery)、以下の 3 つのうちいずれかの方法を使用します。

- -update-tap フラグを使用: guard_tap.iniを編集して、検査エンジンを追加または更新します。
- --send-to-sqlguard フラグを使用 (またはフラグなし、これがデフォルト): 検出された変更を Guardium システムに送信します。これらの変更は Guardium システムの「ディスカバーされたインスタンス」レポートに表示されます。
- --print-output フラグを使用: 検出された変更を stdout に出力します (デバッグ用)

S-TAP が Guardium ではなく「ユーザー」として実行されている場合、ディスカバリー機能は制限されます。以下のメッセージが表示されます。

警告: ディスカバリーが有効化され、STAP がユーザー guardium として実行されています。(WARNING: Discovery is enabled and STAP is running as user guardium.)
STAP がユーザー guardium として実行されている場合、ディスカバリー機能は制限されます。(The discovery function is limited when STAP runs as user guardium.)
ディスカバリーが最も有効なのは、「tap_run_as_root=1」の場合です。(Discovery is most effective when 'tap_run_as_root=1')

注: S-TAP のディスカバリーは AIX 5.3 ではサポートされていません。これは、そのプラットフォームに静的ライブラリーが必要であるためです。
注: S-TAP ディスカバリーで Informix データベースがオープンされないという状況が発生しないようにするには、実行可能ファイルの絶対パスを使用して Informix データベースを開始することをお勧めします。

S-TAP ディスカバリー・アプリケーションのパラメーターは、上級ユーザーの場合を除き、デフォルト値のままにしておく必要があります。ディスカバリー・アプリケーションについては、Linux システムおよび UNIX システム: discovery パラメーターで説明しています。

ディスカバリーは、以下のパラメーターも使用します。

- tap_ip: データベース・インスタンスが関連付けられている S-TAP。
- sqlguard_ip: S-TAP のディスカバリーの結果が、この IP に送信されます。 (SQLguard パラメーターに primary=1 が指定されている Guardium システム。)

親トピック: Linux システムおよび UNIX システム: S-TAP の構成

Linux システムおよび UNIX システム: 検査エンジンの構成

「S-TAP 制御」ペインで検査エンジンを構成または変更します。

始める前に

S-TAP を管理する Guardium システムにログインする必要があります。

このタスクについて

S-TAP をホストしている Guardium システム、または同じ Guardium システムにレポートしている別の S-TAP によって直接モニターされているネットワーク・トラフィックを、S-TAP 検査エンジンでもモニターするように構成しないでください。そのように構成すると、Guardium システムが重複する情報を受け取り、セッションを再構成できずに、そのトラフィックを無視する可能性があります。

手順

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」に移動します。
2. S-TAP の行で、 をクリックします。 「S-TAP 構成」ウィンドウが開きます。
3. 検査エンジンの下部までスクロールして、「検査エンジンの追加...」の横にある をクリックします。
4. プロトコルを選択して、ポート範囲を入力します。ウィンドウが関連パラメーターとそのデフォルト値 (一部) で最新表示されます。
5. すべての必須パラメーターを構成して、「追加」をクリックします。パラメーターが欠落している場合は、システムにより、欠落しているパラメーターが通知されます。

親トピック: Linux システムおよび UNIX システム: S-TAP の構成
関連資料:
Linux システムおよび UNIX システム: 検査エンジン・パラメーター

Linux システムおよび UNIX システム: 検査エンジンの検査

S-TAP 検査では、ご使用の環境の STAP とのその検査エンジンが実行されていて、データベース・アクティビティーをアクティブにモニターしていることを確認します。検査について理解し、S-TAP を定期的に検査するためのスケジュールを定義します。

検査では、Guardium システムと検査エンジンとの間のスニファー操作と通信を検査します。検査は、システム上のすべての S-TAP クライアント、個々の S-TAP クライアント、または個々の検査エンジンについて有効にできます。

検査は、以下のデータベース・タイプでサポートされています。

- Db2 および Db2 出口
- Greenplum
- Informix
- MSSQL (クラスター構成の場合、詳細検査のみをサポート)
- MySQL
- Netezza
- Oracle
- PostgreSQL
- Teradata (詳細検査のみ)

検査には、以下の 2 つのタイプがあります。

標準検査
: S-TAP と検査エンジンとの間のスニファー操作と通信を検査します。 検査プロセスは、間違ったユーザー ID とパスワードを使用してデータベースの STAP クライアントへのログインを試行することで、この試行が認識され、Guardium システムに通知されることを確認します。 次に、検査プロセスが、データベース・サーバー上の選択された検査エンジンに接続できるかどうか検査します。 失敗ログインを示す応答を受信することが想定されています。 異なる応答が受信される場合は、さらに調査を行わなければならない可能性があります。
個々のデータベースからの一部エラー・メッセージは、特定の 1 つの問題を示しているわけではありません。 例えば、いくつかのサポートされるデータベース上で、ポートが間違っているためにエラー・コードが返される場合、データベース自体が開始していないことも意味する可能性があります。

詳細検査
: 失敗ログイン要求を避けて、個々の IE を管理するには、詳細検査を使用します。失敗ログイン要求を避けるためには、ターゲット・データベースに関連付けられているデータ・ソース定義を特定または作成する必要があります。 データ・ソース定義には、検査プロセスがデータベースにログインするために使用する資格情報が含まれています。 次に、エラー・メッセージを生成するために、存在しない表からデータを取得するための要求が送信されます。

両方のタイプの検査要求について、実行されたテストと、失敗したテストの推奨アクションについての情報を提供する新しいダイアログに結果が表示されます。

- Linux システムおよび UNIX システム: S-TAP 検査
  S-TAP 検査プロセスはいくつかの構成パラメーターを確認し、検査エンジンに接続しようとします。
- Linux システムおよび UNIX システム: 標準検査の構成
  このタスクを使用して、特定の S-TAP クライアント・ホスト上のすべての検査エンジンを構成します。
- Linux システムおよび UNIX システム: 詳細検査の構成
  このタスクを使用して、個々の検査エンジンについてすべての検証を構成し、詳細検査を構成します。
- Linux システムおよび UNIX システム: S-TAP 検査スケジュールの構成
  S-TAPs を検査するデフォルトのスケジュールは、毎日 1 時間に 1 回です。このスケジュールは変更可能です。

親トピック: Linux システムおよび UNIX システム: S-TAP の構成

Linux システムおよび UNIX システム: S-TAP 検査

S-TAP 検査プロセスはいくつかの構成パラメーターを確認し、検査エンジンに接続しようとします。

データベースに接続する前に、検査プロセスは、Guardium システム上でスニファー・プロセスが実行されているかどうかを検査します。スニファーは、各 S-TAP との通信と、受信されるデータの処理を担当します。スニファーが実行されていない場合、S-TAP からの応答は認識されません。

検査プロセスは、間違ったユーザー ID とパスワードを使用してデータベースの STAP クライアントへのログインを試行することで、この試行が認識され、Guardium システムに通知されることを確認します。

次に、検査プロセスが、データベース・サーバー上の選択された検査エンジンに接続できるかどうか検査します。 失敗ログインを示す応答を受信することが想定されています。 異なる応答が受信される場合は、さらに調査を行わなければならない可能性があります。

個々のデータベースからの一部エラー・メッセージは、特定の 1 つの問題を示しているわけではありません。 例えば、いくつかのサポートされるデータベース上で、ポートが間違っているためにエラー・コードが返される場合、データベース自体が開始していないことも意味する可能性があります。

「S-TAP 検査」ページで検査結果を確認します (「管理」 > 「レポート」 > 「アクティビティー・モニター」 > 「S-TAP 検査」ページ)。失敗した検査が最初に示され、次のステップのための推奨事項が示されます。 リスト末尾の縮小表示されたセクションに、成功した検査が表示されていることを確認します。 状況によっては、成功した検査を調べることが、考えられる複数の次のステップから選択するのに役立つ場合があります。

親トピック: Linux システムおよび UNIX システム: 検査エンジンの検査

Linux システムおよび UNIX システム: 標準検査の構成

このタスクを使用して、特定の S-TAP クライアント・ホスト上のすべての検査エンジンを構成します。

このタスクについて

この手順の代わりに、GRDAPI コマンド verify_stap_inspection_engine_with_sequence を使用できます。

手順

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」にアクセスします。
2. 次のオプションを使用します。
   - スケジュールにすべて追加: 表示されているすべての S-TAP に対するすべての検査エンジンを検査に追加します。
   - スケジュールからすべて削除: 表示されているすべての S-TAP に対するすべての検査エンジンを検査から削除します。
   - スケジュールに追加: 選択した S-TAP クライアントのすべての検査エンジンをスケジュールに追加します。

   S-TAP でオプション「すべてが制御可能」が有効になっていない場合は、この S-TAP に対して Guardium システムが 1 次システムである場合、状況の変更のみを実行できます。
3. 「リフレッシュ」をクリックします。
4. 今すぐ検査を実行するには、「管理」 > 「アクティビティー・モニター」 > 「S-TAP 検査スケジューラー」に移動し、「今すぐ 1 回実行」をクリックします。
5. 検査結果を表示する前に、システムはデフォルトで 5 秒待機します。 ネットワーク待ち時間が長い場合、これは、データベース・サーバーからの予期される応答を受信するには、十分の時間ではない可能性があります。 より長い時間が必要な場合は、store stap network_latency CLI コマンドを使用して、期間を変更できます。

次のタスク

「S-TAP 検査」ページで検査結果を確認します (「管理」 > 「レポート」 > 「アクティビティー・モニター」 > 「S-TAP 検査」ページ)。失敗した検査が最初に示され、次のステップのための推奨事項が示されます。 リスト末尾の縮小表示されたセクションに、成功した検査が表示されていることを確認します。 状況によっては、成功した検査を調べることが、考えられる複数の次のステップから選択するのに役立つ場合があります。

親トピック: Linux システムおよび UNIX システム: 検査エンジンの検査

Linux システムおよび UNIX システム: 詳細検査の構成

このタスクを使用して、個々の検査エンジンについてすべての検証を構成し、詳細検査を構成します。

このタスクについて

手順

1. 「管理」 > 「システム・ビュー」 > 「S-TAP 状況モニター」にアクセスします。
2. S-TAP の行の任意の場所をクリックします。
   このホストの個々の検査エンジンによってウィンドウがリフレッシュされます。
3. 詳細検査を構成します。
   a. 1 つの検査エンジンをクリックし、「詳細検査」をクリックします。
   b. オプションで、「データ・ソース」で、「一致する S-TAP ホストのみを表示」を選択するか、「名前ドロップダウン・リストから名前を選択して、特定の検査エンジンを検索します。
   c. 「閉じる」をクリックします。
4. 今すぐ検査を実行するには、検査エンジンを 1 つ以上選択し、「検査」をクリックします。「S-TAP 検査の結果」ウィンドウが開きます。
5. 検査結果を表示する前に、システムはデフォルトで 5 秒待機します。 ネットワーク待ち時間が長い場合、これは、データベース・サーバーからの予期される応答を受信するには、十分の時間ではない可能性があります。 より長い時間が必要な場合は、store stap network_latency CLI コマンドを使用して、期間を変更できます。
6. 検査の追加または削除を実行するには、次のようにします。
   a. 1 つ以上の検査エンジンを選択します。
   b. 「スケジュールに追加」または「スケジュールから削除」をクリックします。

次のタスク

「S-TAP 検査」ページで検査結果を確認します (「管理」 > 「レポート」 > 「アクティビティー・モニター」 > 「S-TAP 検査」ページ)。失敗した検査が最初に示され、次のステップのための推奨事項が示されます。 リスト末尾の縮小表示されたセクションに、成功した検査が表示されていることを確認します。 状況によっては、成功した検査を調べることが、考えられる複数の次のステップから選択するのに役立つ場合があります。

親トピック: Linux システムおよび UNIX システム: 検査エンジンの検査

Linux システムおよび UNIX システム: S-TAP 検査スケジュールの構成

S-TAPs を検査するデフォルトのスケジュールは、毎日 1 時間に 1 回です。このスケジュールは変更可能です。

このタスクについて

検査がスケジュールされているすべての S-TAPs に、同じスケジュールが使用されます。

スケジュールが定義されると、「S-TAP 検査スケジューラー」の「一時停止」ボタンをクリックして、検証プロセスをアクティブのまま一時的に停止できます。 リアルタイムで 1 回検証を実行するには、「今すぐ 1 回実行」ボタンを使用します。

手順

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 検査スケジューラー」をクリックして、「S-TAP 検査スケジューラー」を開きます。
2. このページの「S-TAP 検査スケジューラー」の部分で、「スケジュールの変更」をクリックします。
3. 「スケジュール定義」ダイアログで、ドロップダウン・リストとチェック・ボックスを使用して、検査実行のスケジュールを設定します。 このスケジュールは、検査がスケジュールされているすべての S-TAPs に適用されます。
4. 「保存」をクリックして、変更を保存します。

親トピック: Linux システムおよび UNIX システム: 検査エンジンの検査

Linux システムおよび UNIX システム: S-TAP のロード・バランシング・モデルと構成ガイドライン

S-TAP のロード・バランシング・モデルについて理解し、セットアップに適したモデルを選択してください。

各ロード・バランシング・モデルとその具体的なパラメーター要件について、以下に説明します。

フェイルオーバー

S-TAP は、1 つのコレクター (1 次) にトラフィックを送信し、必要に応じて 1 つ以上のコレクター (2 次、3 次など) にフェイルオーバーします。S-TAP エージェントは、1 つの 1 次コレクター IP と 1 つ以上の 2 次コレクター IP で構成されます。さまざまな理由で S-TAP エージェントが 1 次コレクターにトラフィックを送信できない場合、S-TAP エージェントは自動的に 2 次コレクターにフェイルオーバーします。2 次ホスト・システムが使用できなくなるか、1 次ホストが再び使用可能になるまで、S-TAP は 2 次ホストにデータを送信し続けます。1 つ目のケースでは、3 次ホストが定義されている場合は、それにフェイルオーバーします。2 つ目のケースでは、S-TAP は 2 次 Guardium ホストから 1 次 Guardium ホストに再びフェイルオーバーします。フェイルオーバー・コレクターは必要な数だけ構成できますが、3 つより多くのコレクターを定義する必要性はありません。1 つのコレクターのみをスタンバイ・フェイルオーバー・コレクターとして定義することも、複数のフェイルオーバー・コレクターを定義することも可能です。1 つのスタンバイ・フェイルオーバーを使用する場合、通常は 4 つから 5 つのコレクターに対して 1 つのコレクターで十分です。複数のフェイルオーバー・コレクターを使用する場合、各コレクターは最大 50% の容量で稼働し、追加の負荷用にリソースが常に存在するようにする必要があります。ご使用のアーキテクチャー、データベース、およびデータ・センターのレイアウトに最適なセットアップを選択してください。

S-TAP はアクティブ・ホストから構成変更が適用されるたびに再始動されます。

「S-TAP 制御」ウィンドウの「詳細」セクションで Load Balancing を 0 に設定し、同じウィンドウの「Guardium ホスト」セクションに 1 つ以上の 2 次 sqlguard_ip を追加します。

上級者以外は、追加のフェイルオーバー構成をデフォルト値のままにしておく必要があります。

Guardium システムを S-TAP の 2 次ホストとして指定する前に、以下の項目を確認してください。

- その Guardium システムが、S-TAP を管理するように構成されていること。これを確認し、必要な場合に再構成するには、『エージェントを管理するための Guardium システムの構成』を参照してください。
- Guardium システムが、S-TAP がインストールされているデータベース・サーバーに接続可能であること。 複数の Guardium システムが使用されている場合、それらはしばしば、ネットワーク上で切り離された分岐に接続されています。
- その Guardiumシステムが、S-TAP がインストールされているデータベース・サーバーからのセッション・データを無視するようなセキュリティー・ポリシーを持たないこと。多くの場合、Guardium® セキュリティー・ポリシーは、監視可能なデータベース・トラフィックの狭いサブセットに重点を置き、その他すべてのセッションは無視するように構築されます。2 次ホストが S-TAP からのセッション・データを無視しないことを確認するか、Guardiumシステムのセキュリティー・ポリシーを必要に応じて変更します。

ロード・バランシング

この構成により、1 つのデータベースから複数のコレクターへのトラフィックのバランスを取ることができます。このオプションは、アクティブ・データベースのすべてのトラフィックをモニターする場合 (広範なモニター) に適しています。(異常値を検出するためには、アグリゲーターですべての関連データが処理されるように、同じアグリゲーターおよび中央マネージャーの下にコレクターが配置されている必要があります。) 生成されたトラフィックが大きく、データを長期間にわたってコレクター上にオンラインで格納する必要がある場合、この方法は最良の選択肢と言えます。なぜなら、複数のコレクター間でセッション・ベースのロード・バランシングが実行されるからです。S-TAP は、最大 10 個のコレクターを使用してこのように構成できます。

ロード・バランシングを選択する場合は、participate_in_load_balancing を 1 に設定します。

グリッド

グリッドでは、S-TAP は f5 や Cisco などのロード・バランサーを介してコレクターと通信します。S-TAP エージェントは、ロード・バランサーにトラフィックを送信するように構成されます。ロード・バランサーは、コレクターのプール内のいずれかのコレクターに S-TAP トラフィックを転送します。ロード・バランサーに障害が発生した場合に継続してモニターできるように、ロード・バランサー間のフェイルオーバーを構成することも可能です。

グリッド・モデルを選択する場合は、participate_in_load_balancing を 3 に設定します。

冗長

冗長では、S-TAP はそのペイロード全体を複数のコレクターに送信します。S-TAP は複数のコレクター (通常は 2 つのみ) で構成され、同じ内容を両方のコレクターに送信します。このオプションにより、ログに記録された同一データの、複数のコレクター間における完全な冗長性が実現します。また、異なる細分度レベルのアクティビティーに関するデータおよびアラートをログに記録するためにこのオプションを使用することもできます。

冗長を選択する場合は、participate_in_load_balancing を 2 に設定します。

複数の K-TAP バッファー

このモードでは、追加のスレッドと K-TAP バッファーを利用してスループットを増やします。participate_in_load_balancing を 4 に設定します。Linux システムおよび UNIX システム: S-TAP スループットの増加を参照してください。

親トピック: Linux システムおよび UNIX システム: S-TAP の構成

Linux システムおよび UNIX システム: SSL 証明書を使用する S-TAP 認証のセットアップ

S-TAP サーバーと Guardium システムの間の認証をセットアップします。

S-TAP は、指定の証明書または証明書セットを使用して認証される特定のマシンのグループのみに接続するように構成できます。これらの証明書は、Guardium システム上でローカルに生成して認証局 (CA) に署名のために送信するか、または CA 側で作成して、Guardium システム全体にインストールできます。

- Linux システムおよび UNIX システム: Guardium システムでの証明書署名要求 (CSR) の生成
  次の手順を実行して、Guardium システム上でローカルに証明書署名要求を生成し、署名のために認証局 (CA) に送信します。
- Linux システムおよび UNIX システム: Guardium システム外部で生成された SSL 証明書のインストール
  次の手順を実行して、CA によって作成された SSL 証明書をインストールします。
- Linux システムおよび UNIX システム: x.509 証明書認証を使用するための S-TAP の構成

親トピック: Linux システムおよび UNIX システム: S-TAP の構成

Linux システムおよび UNIX システム: Guardium システムでの証明書署名要求 (CSR) の生成

次の手順を実行して、Guardium システム上でローカルに証明書署名要求を生成し、署名のために認証局 (CA) に送信します。

手順

1. CLI を使用して Guardium システムにログインします。
2. cli> create csr sniffer と入力します。
3. 要求されたデータを入力します。

```
temp4> create system csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [GB]:CA
State or Province Name (full name) [Berkshire]:BC
Locality Name (eg, city) [Newbury]:
Organization Name (eg, company) [My Company Ltd]:QA_Sample1
Organizational Unit Name (eg, section) []:Sample_QA
Common Name (eg, your name or your server's hostname) []:sample1_qa.victoria
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
```

終了すると、次のようになります。

```
Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject: C=CA, ST=BC, L=Newbury, O=QA_Sample1, OU=Sample_QA, CN=sample1_
qa.victoria
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (2048 bit)
                Modulus (2048 bit):
                    00:e9:5e:a2:81:53:dc:e9:b5:f7:54:33:17:6f:15:
                    9c:00:5c:ff:b2:64:c6:e5:48:be:36:a7:a4:55:f4:
                    b1:df:c9:81:a4:41:fe:29:80:d6:fd:d4:c4:b5:97:
                    b7:c1:3d:42:6c:c0:f8:09:cd:ea:36:f6:3b:b9:d9:
                    ce:15:87:2d:2f:b3:f2:5f:f9:42:06:e2:a0:62:56:
                    06:6d:cc:65:69:62:db:36:34:09:95:5b:c3:d0:e6:
                    85:ee:64:76:3e:ed:d6:47:bb:49:f2:08:81:14:c2:
                    e3:93:db:20:ba:86:e7:60:24:80:01:7f:3d:b7:60:
                    16:ba:06:d4:a1:e0:18:39:73:ca:1e:24:15:56:6d:
                    97:79:81:04:f7:fd:37:06:42:d7:15:82:34:aa:51:
                    2c:cc:e2:f0:d1:42:dd:b5:71:bb:10:19:a0:0a:5c:
                    4f:77:b9:bb:36:95:ed:a4:77:07:e3:50:f9:36:20:
                    13:e2:e1:78:d2:0a:36:8a:b9:39:90:1f:a4:82:12:
                    4f:50:29:3f:19:7d:16:a6:b3:23:7b:b8:7b:85:60:
                    04:21:39:84:1d:9e:81:e5:20:2c:8a:51:f3:52:f7:
                    3c:4f:e6:f2:a5:88:dc:2e:99:0a:b3:65:1e:bf:33:
                    5f:be:dc:53:1c:a6:69:18:c4:c7:75:bf:20:e3:cf:
                    29:af
                Exponent: 65537 (0x10001)
        Attributes:
            challengePassword        :guardium
    Signature Algorithm: sha1WithRSAEncryption
        06:4a:b9:db:04:a1:8d:4c:f7:3f:8f:24:fa:7c:ec:a6:70:77:
        8b:b9:38:7c:b6:e6:51:aa:ed:96:20:16:37:85:a7:44:26:2b:
        87:4c:a4:db:0c:f3:d3:87:e3:68:4a:8e:de:f6:0a:09:58:8f:
        68:98:4f:f3:8a:e2:37:5c:d6:42:32:8f:d9:01:56:41:88:df:
        1a:ba:63:03:62:08:89:06:13:88:74:6f:cd:eb:26:f0:67:a4:
        26:9b:a3:4c:ff:7b:c9:19:2c:12:58:06:ce:22:3c:e6:cd:52:
        b0:d0:da:6a:c9:02:df:02:e6:25:77:39:cf:50:80:e7:1d:01:
        fc:40:17:a2:98:04:bf:8b:24:f6:55:46:99:7b:17:05:01:d3:
        09:3d:a2:f0:e0:ba:5d:15:b8:28:74:d2:a3:fe:fd:86:7d:e0:
        60:e8:e4:38:6a:17:9c:80:80:e3:50:11:5e:35:f5:02:2b:65:
        60:41:2a:dc:ed:a8:a9:9a:6f:24:b4:7a:9c:39:01:a4:fc:cf:
        e6:94:86:f1:18:3a:f5:99:6b:f8:66:a2:ff:04:08:7e:ca:6b:
        2a:aa:cf:72:26:d0:c9:96:a0:98:fd:91:bb:b1:e4:8d:6d:10:
        08:ea:56:de:07:20:d3:e6:9a:bf:de:cf:c3:a4:e8:43:60:4f:
        b4:53:aa:d5
-----BEGIN CERTIFICATE REQUEST-----
MIIC0TCCAbkCAQAwczELMAkGA1UEBhMCQ0ExCzAJBgNVBAgTAkJDMRAwDgYDVQQH
EwdOZXdidXJ5MRMwEQYDVQQKDApRQV9TYW1wbGUxMRIwEAYDVQQLDAlTYW1wbGVf
UUExHDAaBgNVBAMME3NhbXBsZTFfcWEudmljdG9yaWEwggEiMA0GCSqGSIb3DQEB
AQUAA4IBDwAwggEKAoIBAQDpXqKBU9zptfdUMxdvFZwAXP+yZMblSL42p6RV9LHf
yYGkQf4pgNb91MS117fBPUJswPgJzeo29ju52c4Vhy0vs/Jf+UIG4qBiVgZtzGVp
Yts2NAmVW8PQ5oXuZHY+7dZHu0nyCIEUwu0T2yC6hudgJIABfz23YBa6BtSh4Bg5
c8oeJBVWbZd5gQT3/TcGQtcVgjSqUSzM4vDRQt21cbsQGaAKXE93ubs2le2kdwfj
UPk2IBPi4XjSCjaKuTmQH6SCEk9QKT8ZfRamsyN7uHuFYAQhOYQdnoHlICyKUfNS
9zxP5vKliNwumQqzZR6/M1++3FMcpmkYxMd1vyDjzymvAgMBAAGgGTAXBgkqhkiG
9w0BCQcxChMIZ3VhcmRpdW0wDQYJKoZIhvcNAQEFBQADggEBAAZKudsEoY1M9z+P
JPp87KZwd4u5OHy251Gq7ZYgFjeFp0QmK4dMpNsM89OH42hKjt72CglYj2iYT/OK
4jdc1kIyj9kBVkGI3xq6YwNiCIkGE4h0b83rJvBnpCabo0z/e8kZLBJYBs4iPObN
UrDQ2mrJAt8C5iV3Oc9QgOcdAfxAF6KYBL+LJPZVRp17FwUB0wk9ovDgul0VuCh0
0qP+/YZ94GDo5DhqF5yAg0NQEV419QIrZWBBKtztqKmabyS0epw5AaT8z+aUhvEY
OvWZa/hmov8ECH7Kayqqz3ImGMmWoJj9kbux5I1tEAjqVt4HINPmmr/ez8Ok6ENg
T7RTqtU=
-----END CERTIFICATE REQUEST-----
ok
temp4>
```

4. -----BEGIN CERTIFICATE REQUEST----- から -----END CERTIFICATE REQUEST----- までをファイルにコピーして、署名のために CA に送信します。

   CA によって証明書が署名され、次のような公開鍵が送られてきます。

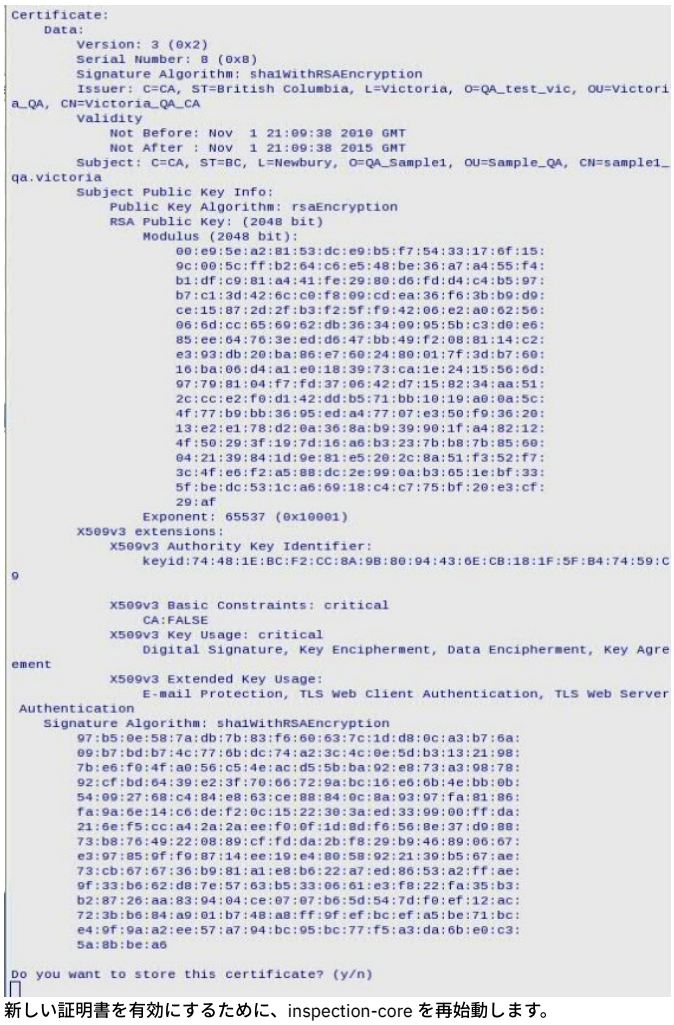
```
enance@enance1 Latest $ cat sample1_qa.victoria.pem
-----BEGIN CERTIFICATE-----
MIID1jCCAsCgAwIBAgIBCDALBgkqhkiG9w0BAQUwgYAxCzAJBgNVBAYTAkNBMRkw
FwYDVQQIExBCcml0aXNoIENvbHVtYmlhMREwDwYDVQQHEwhWaWN0b3JpYTEUMBIG
A1UEChMLUUFfdGVzdF92aWMxFDASBgNVBAsTC1ZpY3Rvcmlhx1FBMRcwFQYDVQQD
Ew5WaWN0b3JpYV9RQV9DQTAeFw0xMDExMDEyMTA5MzhaFw0xNTExMDEyMTA5Mzha
MHMxCzAJBgNVBAYTAkNBMQswCQYDVQQIEwJCQzEQMA4GA1UEBxMHTmV3YnVyeTET
MBEGA1UECgwKUUFfU2FtcGxlMTESMBAGA1UECwwJU2FtcGxlX1FBMRwwGgYDVQQD
DBNzYW1wbGUxX3FhLnZpY3RvcmlhMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA6V6igVPc6bX3VDMXbxWcAFz/smTG5Ui+NqekVfSx38mBpEH+KYDW/dTE
tZe3wT1CbMD4Cc3qNvY7udnOFYctL7PyX/lCBuKgYlYGbcxlaWLbNjQJlVvD0OaF
7mR2Pu3WR7tJ8giBFMLjk9sguobnYCSAAX89t2AWugbUoeAY0XPKHiQVVm2XeYEE
9/03BkLXFYI0qlEszOLw0ULdtXG7EBmgClxPd7m7NpXtpHcH41D5NiAT4uF40go2
irk5kB+kghJPUCk/GX0WprMje7h7hWAEITmEHZ6B5SAsilHzUvc8T+bypYjcLpkK
s2UevzNfvtxTHKZpGMTHdb8g488prwIDAQABo2swaTAfBgNVHSMEGDAWgBR0SB68
8syKm4CUQ27LGB9ftHRZyTAMBgNVHRMBAf8EAjAAMA8GA1UdDwEB/wQFAwMHuAAw
JwYDVR0lBCAwHgYIKwYBBQUHAwQGCCsGAQUFBwMCBggrBgEFBQcDATALBgkqhkiG
9w0BAQUDggEBAJe1Dlh623uD9mBjfB3YDKO3agm3vbdMd2vcdKI8TA5dsxMhmHvm
8E+gVsV0rNVbupLoc6OYeJLPvWQ54j9wZnKavBbma067C1QJJ2jEhOhjzoiEDIqT
l/qBhvqabhTG3vIMFSIwOu0zmQD/2iFu9cykKiru8A8djfZWjjfZiHO4dkkiCInP
/dor+Cm5RokGZ+OXhZ/5hxTuGeSAWJIhObVnrnPLZ2c2uYGh6LYip+2GU6L/rp8z
tmLYfldjtTMGYeP4Ivo1s7KHJqqDlATOBwe2XVR98O8SrHI7toSpAbdIqP+f77zv
pb5xv0SfmqLuV6eUvJW8d/Wj2mvgw1qLvqY=
-----END CERTIFICATE-----
```

5. 内容をコピーしたり、Guardium システムにインポートするのに便利な場所にこのファイルを配置してください。cli> store certificate sniffer [console | import] と入力します。
6. console の場合、**-----BEGIN CERTIFICATE-----** から **-----END CERTIFICATE-----** まですべて (コピー内のものを含む) をコピーし、プロンプトが出されたら CLI に貼り付けます。import を選択した場合、ファイルのインポート元を Guardium システムに指示します。

```
temp4 > store system certificate console
Please paste your new system certificate, in PEM format.
Include the BEGIN and END lines, then press CTRL-D.

-----BEGIN CERTIFICATE-----
MIID1jCCAsCgAwIBAgIBCDALBgkqhkiG9w0BAQUwgYAxCzAJBgNVBAYTAkNBMRkw
FwYDVQQIExBCcml0aXNoIENvbHVtYmlhMREwDwYDVQQHEwhWaWN0b3JpYTEUMBIG
A1UEChMLUUFfdGVzdF92aWMxFDASBgNVBAsTC1ZpY3Rvcmlhx1FBMRcwFQYDVQQD
Ew5WaWN0b3JpYV9RQV9DQTAeFw0xMDExMDEyMTA5MzhaFw0xNTExMDEyMTA5Mzha
MHMxCzAJBgNVBAYTAkNBMQswCQYDVQQIEwJCQzEQMA4GA1UEBxMHTmV3YnVyeTET
MBEGA1UECgwKUUFfU2FtcGxlMTESMBAGA1UECwwJU2FtcGxlX1FBMRwwGgYDVQQD
DBNzYW1wbGUxX3FhLnZpY3RvcmlhMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA6V6igVPc6bX3VDMXbxWcAFz/smTG5Ui+NqekVfSx38mBpEH+KYDW/dTE
tZe3wT1CbMD4Cc3qNvY7udnOFYctL7PyX/lCBuKgYlYGbcxlaWLbNjQJlVvD0OaF
7mR2Pu3WR7tJ8giBFMLjk9sguobnYCSAAX89t2AWugbUoeAY0XPKHiQVVm2XeYEE
9/03BkLXFYI0qlEszOLw0ULdtXG7EBmgClxPd7m7NpXtpHcH41D5NiAT4uF40go2
irk5kB+kghJPUCk/GX0WprMje7h7hWAEITmEHZ6B5SAsilHzUvc8T+bypYjcLpkK
s2UevzNfvtxTHKZpGMTHdb8g488prwIDAQABo2swaTAfBgNVHSMEGDAWgBR0SB68
8syKm4CUQ27LGB9ftHRZyTAMBgNVHRMBAf8EAjAAMA8GA1UdDwEB/wQFAwMHuAAw
JwYDVR0lBCAwHgYIKwYBBQUHAwQGCCsGAQUFBwMCBggrBgEFBQcDATALBgkqhkiG
9w0BAQUDggEBAJe1Dlh623uD9mBjfB3YDKO3agm3vbdMd2vcdKI8TA5dsxMhmHvm
8E+gVsVOrNVbupLoc6OYeJLPvWQ54j9wZnKavBbma067C1QJJ2jEhOhjzoiEDIqT
l/qBhvqabhTG3vIMFSIwOu0zmQD/2iFu9cykKiru8A8djfZWjjfZiHO4dkkiCInP
/dor+Cm5RokGZ+OXhZ/5hxTuGeSAWJIhObVnrnPLZ2c2uYGh6LYip+2GU6L/rp8z
tmLYfldjtTMGYeP4Ivo1s7KHJqqDlATOBwe2XVR98O8SrHI7toSpAbdIqP+f77zv
pb5xvOSfmqLuV6eUvJW8d/Wj2mvgw1qLvqY=
-----END CERTIFICATE-----
```

証明書を保管するかどうかを確認するよう求められます。確認すると、証明書が保管されます。



7. 新しい証明書を有効にするために、inspection-core を再始動します。

親トピック: Linux システムおよび UNIX システム: SSL 証明書を使用する S-TAP 認証のセットアップ

Linux システムおよび UNIX システム: Guardium システム外部で生成された SSL 証明書のインストール

次の手順を実行して、CA によって作成された SSL 証明書をインストールします。

このタスクについて

インストールする証明書全体が CA から送信される場合、PKCS#8 (パスワード保護) フォーマットの秘密鍵と PEM フォーマットの公開鍵の 2 つのファイルが必要です。生成される証明書は、2048 ビットの RSA 鍵である必要があります。

CA から 2 つのファイルと、CA の公開証明書が送信されます。

CA の公開証明書は次のようになります。

```
enance@enance1 Latest $ cat Victoria_QA_CA.pem
-----BEGIN CERTIFICATE-----
MIID2zCCAsWgAwIBAgIBATALBgkqhkiG9w0BAQUwgYAxCzAJBgNVBAYTAkNBMRkw
FwYDVQQIExBCcml0aXNoIENvbHVtYmlhMREwDwYDVQQHEwhWaWN0b3JpYTEUMBIG
A1UEChMLUUFfdGVzdF92aWMxFDASBgNVBAsTC1ZpY3RvcmlhX1FBMRcwFQYDVQQD
Ew5WaWN0b3JpYV9RQV9DQTAeFw0xMDA4MTIxODMzMDJaFw0zMDA4MTIxODMzMDJa
MIGAMQswCQYDVQQGEwJDQTEZMBcGA1UECBMQQnJpdGlzaCBDb2x1bWJpYTERMA8G
A1UEBxMIVmljdG9yaWExFDASBgNVBAoTC1FBX3R1c3RfdmljMRQwEgYDVQQLEwtW
aWN0b3JpYV9RQTEXMBUGA1UEAxMOVmljdG9yaWFfUUFfQ0EwggEgMAsGCSqGSIb3
DQEBAQOCAQ8AMIIBCgKCAQEA0x3iAXs1KGNoJThXkO+jcNyMBlfwKWRMTOq9PkF4
piznXCRwPz2nQWk5/fpslchmuVYXJtfZi7umDxp2FEMvMmhJfFZiqbCn1Rb5yH+1
V3RsIerB0DFpoWkdt+wD6BufndO5P9eOlv14bmTl+F0dUM0TxAwtX73CMQOX/n+i
/WrzpWU4lU71KkyWUfJl2Pm8TLEMr5awpzt2rEJ/Q1qItHcksQDbGY0MNLaNojEU
XBzUpu9ezbv+zVH+5iorFYkrH0NQI0NK+YoR1b3Tto0HLDh6istsMfHdNEEQb9BB
vMjqUz4tGB2HDguYTaNbQJj9Yw8uv7/tfwW/cesrqm8DiQIDAQABo2QwYjAPBgNV
HRMBAf8EBTADAQH/MA8GA1UdDwEB/wQFAwMHBgAwHQYDVR0OBBYEFHRIHrzyzIqb
gJRDbssYH1+0dFnJMB8GA1UdIwQYMBaAFHRIHrzyzIqbgJRDbssYH1+0dFnJMAsG
CSqGSIb3DQEBBQOCAQEAbrImEbRyBka0wO/ZuPd0Hw9jpbxIuaYEskakv7aM4TUQ
awf1CiqwwAymmKb2REItLajhjmBFbXBun7d137vBU2KX104I7WGw0xgI5rmlELa+
2fizugY+Bc6mh+5c0haizkyudKzo8mLz2p/IS7SPh2lJ9rnuBleSt9zfiYanPxX1
Q6z1+WRKIRSUmK+ho4bmtgr5F0+ejmZb9nfzeOBj23H9i0mWoaQ/SO+O2iDIvDY0
KYwqCeS2UNEdTcnfuczbBkqnAsf5/GBPIhnWX3onuLkOsHdY0HHPjoqgHauoxPK8
p0sEvlCK8EFOD6wkp0vtNhFQCykxRimHR6Pz9jEVjw==
-----END CERTIFICATE-----
```

ユーザー/この Guardium システムに固有の公開証明書は、次のようになります。

```
enance@enance1 Latest $ cat Sample_givenCert.victoria.qa_pub.pem
-----BEGIN CERTIFICATE-----
MIID5jCCAtCgAwIBAgIBCTALBgkqhkiG9w0BAQUwgYAxCzAJBgNVBAYTAkNBMRkw
FwYDVQQIExBCcml0aXNoIENvbHVtYmlhMREwDwYDVQQHEwhWaWN0b3JpYTEUMBIG
A1UEChMLUUFfdGVzdF92aWMxFDASBgNVBAsTC1ZpY3RvcmlhX1FBMRcwFQYDVQQD
Ew5WaWN0b3JpYV9RQV9DQTAeFw0xMDExMTUyMDUwNThaFw0xNTExMTUyMDUwNTha
MIGEMQswCQYDVQQGEwJDQTEZMBcGA1UECBMQQnJpdGlzaCBDb2xvbWJpYTESMBAG
A1UEBxMJVmFuY291dmVyMQswCQYDVQQKEwJRQTESMBAGA1UECxMJUUFfU0FNUExF
MSUwIwYDVQQDExxTYW1wbGVfZ2l2ZW5DZXJ0LnZpY3RvcmlhLnFhMIIBIDALBgkq
hkiG9w0BAQEDggEPADCCAQoCggEBALdt2XhJEJIAognblgFNoiomVvcGIe9PHs0t
3fP5DxAL5PVm+UCRS0xnnCoKe3pdJNagepDwBa2K5UsHbK4vkHJEgWWEd4bx2Ks7
kRoHr83TXK+W8al1HGKRwUSwN0hfm/KV4v8ZX3AFWs2c/BJ2iA77a0mhU0JuWa1/
/scXitJB5ykPjFb2EuReA6ELphaQ/itjZIQTTeVxbamyx42ia9J5B07lFTp3qGdU
yXUwlEFX0HFPmknAAaQrSnpMdQjkOKgzxU0NTV4ILA66hCVkX+ezQeJA90s/AHA5
hAY2fyurKZsOowoE0xlEpFWCGf87zxfkmTaWthqCS0NzEcJ6efECAwEAAaNrMGkw
HwYDVR0jBBgwFoAUdEgevPLMipuAlENuyxgfX7R0WckwDAYDVR0TAQH/BAIwADAP
BgNVHQ8BAf8EBQMDB7gAMCcGA1UdJQQgMB4GCCsGAQUFBwMEBggrBgEFBQcDAgYI
KwYBBQUHAwEwCwYJKoZIhvcNAQEFA4IBAQB+P1m22B72eRESk9rgNdOB+l48k5xw
QHf64TJcoI/Vcz0vtGBC1jDfZzVDIHJaX6ZMtdO23QO6rtjQSnhjhbOt/EiimaNO
tysJOE999E+HQ7UpKYKvdoznwpYEthyoQJ9quKqQc1YW/l8pQMYDEdK8c7yMbJpv
mrxO8h+G3YYQMT6A9KAk/GPA+yKl0fCuogBhPyWM28q35EAIw6H9ahQigWhNkzrL
DCYO8VCLi3BX+OohLvurkyzJm2lnbm7BSb5QB4jISlRBGPoIWhK6VPypboyABfVs
yWHFVs10EUb0Maa/XNZILTAoYAbYK5sX7Ri5hr5KhxnmDd9AJMQsLfik
-----END CERTIFICATE-----
```

秘密鍵 (pkx#8 で暗号化) は次のようになります。

```
enance@enance1 Latest $ cat Sample_givenCert.victoria.qa_pri.pem
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIE4jAcBgoqhkiG9w0BDAEDMA4ECAKuMvSla9T4AgIBYgSCBMB/RzDgm4xpRDnA
Cd//wgD9c8junXmVpyJWXWgXhlj3Sgv3toKt2yBQY7Rv0peP1y68nuHKZcp/QZDj
QksDvIJHacpz886KQ2p9xxOCrIO3RTVNULPzrucUNuJ2a5W4IGPGwnrNLKsriRH6
CXQjRu19+kbczmPtrlHu7gj6MuCp58sIyXbsOPCnsZBaAOcf72qeKfVa4tKlpobI
e5YDg+UiJXYv1Kfj9t9tftSfaCym0jK+uE3++y5+ah/k+u/VcEb9b6DrlfG8KaJI
/OYVVmRWVIHEY2QjVOsVEXAARjpugf6cXIn8epw/CMaC0yVg4O9OGDrIz/41DVgk
/b3jlr1oWMRVASSqgOQxE2LrNNAF/XiFwpiLUCBHR0fXX9iKJM7nK6AXlf4hrd31
sMm1hICJvQwuRP+1xCHHs3qKk7oFAdyEdMiYthcOoAaynGmP4vPEMB6ZdOSxd/yN
7IfBT1U7nCtgo4FpAn4FTnkXRdlDqPpvKLqe0kkQCKo9ZWRnnIS1hAs9VJXS6zou
1IohZewrS/TWnPU+ArE/QJ4WkX2dBAe0Q2HMvuXs6J6Rfn+H2eZC+zThOkvqA0ic
UiQAgJB0ohXWEah9BJbIYNULLpbssUySDLPrxYdZRPn5MFKqCXmWw5tAEV8mIcg4
obB6DxVGHz93iPD0XJyBGWpCdcwItzdngnMR316tX4R3jMy2U0hrzk5o01GoZQQ/
eGYsuPzm4zqVJ8S9gUMFR8ocoPWqwhU+Sq4QPCIQEGf5gwxxt9j7L2TtRbD7kxnD
23aPL/wuk45EFJKTQCeZ3kjUqCCuSbqBJXGEjiiUsahus+yJULJ8boFWu1T8zW7m
/CBK1TzY9Fh+gbd0tN4b+zWUrpk5E7ZctFhEwojVFBwgPzUIZEkap4gH+/F5d3rI
uWFBcN7QER35w08au/k5kLKHDh+5U15ltkJ38P1pJYFE0p5/K81YWnfvUaJmynJY
DfnK0X6Au69/F6+QR4S+bchbrqk73plIiXbkwqapQ26QH5ztIK3T6/nY6RKa7u6N
WvnskpAE1uxf+soM+BpEzpTl2gQQmEaZnh9Ci3ZnpTZE5tkAL0oTqXL9sToyeynz
Z91m+uKxYRjFvc1KPh+I8dUR0E5qXhBdRR7heIlm7/Dg+UqCUkXoyW89G14u/mJT
jB50CHxLAFOSeXnKn8G/9r0HA565r8L/z8D12sli5T1zHicZJ80211zCSezdF+GG
mI12jfOZmNIBgCVUtpOBNX5IDbr7L+0bM6vuDmXRhnhX4f34+5HGfj3ntLAV7bmn
IOEGIhyoGFnoxyWaT0xxa6MVQJazcKSyyyhO8UixFEoKI3drpFXB0tXVb0CLFq+y
QVp9dHgO30tzw/yCmCpbQQe8UFRYyC74wBf9yPWDtN6MW/IeEZCBoYaeMR+WyonM
mdq5xbriETXD4zha3NGAv2qnzkfkKMBcVeUULulyixMsHmNNJkINNEsj5deVYtOt
4j5EK/aZsTqTgrVq3oGlwgXM1YptvlVR+HPWZfpjwbL+7/NhmK7AhEPo+qQCVD7t
ZGRbdbB7FEN23m88IMSd8xkhS2M1pop1Cj7lP/dA+DD/dgcjP2bw7K923d4r3CcS
1yxPhLKM
-----END ENCRYPTED PRIVATE KEY-----
```

これらのファイルを準備して、Guardium システムにインポートする (scp、ftp などを使用) か、Guardium システム上の cli インターフェースにコピー・アンド・ペーストします。

手順

1. CLI を使用して Guardium システムにログインします。
2. cli> store certificate keystore [import | console] と入力して秘密鍵を保管します。 import は保存されたファイルを取得し、ファイルの内容をコンソール・インターフェースにコピー・アンド・ペーストします。ファイルと共に保存されたパスワードを尋ねられます。これは、証明書を作成するために CA に提供したものか、あ

るいは多くの場合、CA からファイルが送信されたときに、CA から提供されたパスワードです。Guardium システムでは次のように表示されます。

```
temp4> store system key console
Please paste your new system key, in PEM format.
Include the BEGIN and END lines, then press CTRL-D.

-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIE4jAcBgoqhkiG9w0BDAEDMA4ECAKuMvSla9T4AgIBYgSCBMB/RzDgm4xpRDnA
Cd//wgD9c8junXmVpyJWXWgXhlj3Sgv3toKt2yBQY7Rv0peP1y68nuHKZcp/QZDj
QksDvIJHacpz886KQ2p9xxOCrIO3RTVNULPzrucUNuJ2a5W4IGPGwnrNLKsriRH6
CXQjRu19+kbczmPtrlHu7gj6MuCp58sIyXbsOPCnsZBaAOcf72qeKfVa4tKlpobI
e5YDg+UiJXYv1Kfj9t9tftSfaCym0jK+uE3++y5+ah/k+u/VcEb9b6Drlfg8KaJI
/OYVVmRWVIHEY2QjVOsVEXAARjpugf6cXIn8epw/CMaC0yVg4O9OGDrIz/41DVgk
/b3jlr1oWMRVASSqgOQxE2LrNNAF/XiFwpiLUCBHR0fXX9iKJM7nK6AXlf4hrd31
sMm1hICJvQwuRP+1xCHHs3qKk7oFAdyEdMiYthcOoAaynGmP4vPEMB6ZdOSxd/yN
7IfBT1U7nCtgo4FpAn4FTnkXRdlDqPpvKLqe0kkQCKo9ZWRnnIS1hAs9VJXS6zou
1IohZewrS/TWnPU+ArE/QJ4WkX2dBAe0Q2HMvuXs6J6Rfn+H2eZC+zThOkvqA0ic
UiQAgJB0ohXWEah9BJbIYNULLpbssUySDLPrxYdZRPn5MFKqCXmWw5tAEV8mIcg4
obB6DxVGHz93iPD0XJyBGWpCdcwItzdngnMR316tX4R3jMy2U0hrzk5o01GoZQQ/
eGYsuPzm4zqVJ8S9gUMFR8ocoPWqwhU+Sq4QPCIQEGf5gwxxt9j7L2TtRbD7kxnD
23aPL/wuk45EFJKTQCeZ3kjUqCCuSbqBJXGEjiiUsahus+yJULJ8boFWu1T8zW7m
/CBK1TzY9Fh+gbdOtN4b+zWUrpk5E7ZctFhEwojVFBwgPzUIZEkap4gH+/F5d3rI
uWFBcN7QER35wO8au/k5kLKHDh+5Ul5ltkJ38P1pJYFE0p5/K81YWnfvUaJmynJY
DfnK0X6Au69/F6+QR4S+bchbrqk73plIiXbkwqapQ26QH5ztIK3T6/nY6RKa7u6N
WvnskpAE1uxf+soM+BpEzpTl2gQQmEaZnh9Ci3ZnpTZE5tkAL0oTqXL9sToyeynz
Z91m+uKxYRjFvc1KPh+I8dUR0E5qXhBdRR7heIlm7/Dg+UqCUkXoyW89G14u/mJT
jB50CHxLAFOSeXnKn8G/9r0HA565r8L/z8D12sli5T1zHicZJ80211zCSezdF+GG
mI12jf0ZmNIBgCVUtpOBNX5IDbr7L+0bM6vuDmXRhnhX4f34+5HGfj3ntLAV7bmn
IOEGIhyoGFnoxyWaT0xxa6MVQJazcKSyyyhO8UixFEoKI3drpFXB0tXVb0CLFq+y
QVp9dHgO3Otzw/yCmCpbQQe8UFRYyC74wBf9yPWDtN6MW/IeEZCBoYaeMR+WyonM
mdq5xbriETXD4zha3NGAv2qnzkfkKMBcVeUULulyixMsHmNNJkINNEsj5deVYtOt
4j5EK/aZsTqTgrVq3oGlwgXM1YptvlVR+HPWZfpjwbL+7/NhmK7AhEPo+qQCVD7t
ZGRbdbB7FEN23m88IMSd8xkhS2M1pop1Cj71P/dA+DD/dgcjP2bw7K923d4r3CcS
1yxPhLKM
-----END ENCRYPTED PRIVATE KEY-----
Enter pass phrase for /var/tmp/key.pem:
writing RSA key

ok
temp4>
```

3. cli> store certificate sniffer [import | console] と入力して署名済み証明書をインポートします。 証明書に関する情報が表示され、証明書を保管するかどうかを確認するよう求められます。次のようになります。

```
temp4> store system certificate console
Please paste your new system certificate, in PEM format.
Include the BEGIN and END lines, then press CTRL-D.

-----BEGIN CERTIFICATE-----
MIID5jCCAtCgAwIBAgIBCTALBgkqhkiG9w0BAQUwgYAxCzAJBgNVBAYTAkNBMRkw
FwYDVQQIExBCcml0aXNoIENvbHVtYmlhMREwDwYDVQQHEwhWaWN0b3JpYTEUMBIG
A1UEChMLUUFfdGVzdF92aWMxFDASBgNVBAsTC1ZpY3RvcmlhX1FBMRcwFQYDVQQD
Ew5WaWN0b3JpYV9RQV9DQTAeFw0xMDExMTUyMDUwNThaFw0xNTExMTUyMDUwNTha
MIGEMQswCQYDVQQGEwJDQTEZMBcGA1UECBMQQnJpdGlzaCBDb2xvbWJpYTESMBAG
A1UEBxMJVmFuY291dmVyMQswCQYDVQQKEwJRQTESMBAGA1UECxMJUUFfU0FNUExF
MSUwIwYDVQQDExxTYW1wbGVfZ2l2ZW5DZXJ0LnZpY3RvcmlhLnFhMIIBIDALBgkq
hkiG9w0BAQEDggEPADCCAQoCggEBALdt2XhJEJIAognblgFNoiomVvcGIe9PHsOt
3fP5DxAL5PVm+UCRS0xnnCoKe3pdJNagepDwBa2K5UsHbK4vkHJEgWWEd4bx2Ks7
kRoHr83TXK+W8al1HGKRwUSwN0hfm/KV4v8ZX3AFWs2c/BJ2iA77a0mhU0JuWa1/
/scXitJB5ykPjFb2EuReA6ELphaQ/itjZIQTTeVxbamyx42ia9J5B07lFTp3qGdU
yXUwlEFX0HFPmknAAaQrSnpMdQjkOKgzxU0NTV4ILA66hCVkX+ezQeJA90s/AHA5
hAY2fyurKZsOowoE0xlEpFWCGf87zxfkmTaWthqCS0NzEcJ6efECAwEAAaNrMGkw
HwYDVR0jBBgwFoAUdEgevPLMipuAlENuyxgfX7R0WckwDAYDVR0TAQH/BAIwADAP
BgNVHQ8BAf8EBQMDB7gAMCcGA1UdJQQgMB4GCCsGAQUFBwMEBggrBgEFBQcDAgYI
KwYBBQUHAwEwCwYJKoZIhvcNAQEFA4IBAQB+Plm22B72eRESk9rgNdOB+l48k5xw
QHf64TJcoI/Vcz0vtGBC1jDfZzVDIHJaX6ZMtdO23QO6rtjQSnhjhbOt/EiimaNO
tysJOE999E+HQ7UpKYKvdoznwpYEthyoQJ9quKqQc1YW/l8pQMYDEdK8c7yMbJpv
mrxO8h+G3YYQMT6A9KAk/GPA+yKl0fCuogBhPyWM28q35EAIw6H9ahQigWhNkzrL
DCYO8VCLi3BX+OohLvurkyzJm2lnbm7BSb5QB4jISlRBGPoIWhK6VPypboyABfVs
yWHFVs10EUb0Maa/XNZILTAoYAbYK5sX7Ri5hr5KhxnmDd9AJMQsLfik
-----END CERTIFICATE-----
```

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 9 (0x9)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=CA, ST=British Columbia, L=Victoria, O=QA_test_vic, OU=Victoria_QA, CN=Victoria_QA_CA
        Validity
            Not Before: Nov 15 20:50:58 2010 GMT
            Not After : Nov 15 20:50:58 2015 GMT
        Subject: C=CA, ST=British Colombia, L=Vancouver, O=QA, OU=QA_SAMPLE, CN=Sample_givenCert.victoria.qa
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (2048 bit)
                Modulus (2048 bit):
                    00:b7:6d:d9:78:49:10:92:00:a2:09:db:96:01:4d:
                    a2:2a:26:56:f7:06:21:ef:4f:1e:c3:ad:dd:f3:f9:
                    0f:10:0b:e4:f5:66:f9:40:91:4b:4c:67:9c:2a:0a:
                    7b:7a:5d:24:d6:a0:7a:90:f0:05:ad:8a:e5:4b:07:
                    6c:ae:2f:90:72:44:81:65:84:77:86:f1:d8:ab:3b:
                    01:1a:07:af:cd:d3:5c:af:96:f1:a9:75:1c:62:91:
                    c1:44:b0:37:48:5f:9b:f2:95:e2:ff:19:5f:70:05:
                    5a:cd:9c:fc:12:76:88:0e:fb:6b:49:a1:53:42:6e:
                    59:ad:7f:fe:c7:17:8a:d2:41:e7:29:0f:8c:56:f6:
                    12:e4:5e:03:a1:0b:a6:16:90:fe:2b:63:64:84:13:
                    4d:e5:71:6d:a9:b2:c7:8d:a2:6b:d2:79:07:4e:e5:
                    15:3a:77:a8:67:54:c9:75:30:94:41:57:d0:71:4f:
                    9a:49:c0:01:a4:2b:4a:7a:4c:75:08:e4:38:a8:33:
                    c5:4d:0d:4d:5e:08:2c:0e:ba:84:25:64:5f:e7:b3:
                    41:e2:40:f7:4b:3f:00:70:39:84:06:36:7f:2b:ab:
                    29:9b:0e:a3:0a:04:d3:19:44:a4:55:82:19:ff:3b:
                    cf:17:e4:99:36:96:b6:1a:82:4b:43:73:11:c2:7a:
                    79:f1
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Authority Key Identifier:
                keyid:74:48:1E:BC:F2:CC:8A:9B:80:94:43:6E:CB:18:1F:5F:B4:74:59:C9

            X509v3 Basic Constraints: critical
                CA:FALSE
            X509v3 Key Usage: critical
                Digital Signature, Key Encipherment, Data Encipherment, Key Agreement
            X509v3 Extended Key Usage:
                E-mail Protection, TLS Web Client Authentication, TLS Web Server Authentication
    Signature Algorithm: sha1WithRSAEncryption
        7e:3e:59:b6:d8:1e:f6:79:11:12:93:da:e0:35:d3:81:fa:5e:
        3c:93:9c:70:40:77:fa:e1:32:5c:a0:8f:d5:73:3d:2f:b4:60:
        42:d6:30:df:67:35:43:20:72:5a:5f:a6:4c:b5:d3:b6:dd:03:
        ba:ae:d8:d0:4a:78:63:85:b3:ad:fc:48:a2:99:a3:4e:b7:2b:
        09:38:4f:7d:f4:4f:87:43:b5:29:29:82:af:76:8c:e7:c2:96:
        04:b6:1c:a8:40:9f:6a:b8:aa:90:73:56:16:fe:5f:29:40:c6:
        03:11:d2:bc:73:bc:8c:6c:9a:6f:9a:bc:4e:f2:1f:86:dd:86:
        10:31:3e:80:f4:a0:24:fc:63:c0:fb:22:a5:d1:f0:ae:a2:00:
        61:3f:25:8c:db:ca:b7:e4:40:08:c3:a1:fd:6a:14:22:81:68:
        4d:93:3a:cb:0c:26:0e:f1:50:8b:8b:70:57:f8:ea:21:2e:fb:
        ab:93:2c:c9:9b:69:67:6e:6e:c1:49:be:50:07:88:c8:4a:54:
        41:18:fa:08:5a:12:ba:54:fc:a9:6e:8c:80:05:f5:6c:c9:61:
        c5:56:cd:74:11:46:f4:31:a6:bf:5c:d6:48:2d:30:28:60:06:
        d8:2b:9b:17:ed:18:b9:86:be:4a:87:19:e6:0d:df:40:24:c4:
        2c:2d:f8:a4

Do you want to store this certificate? (y/n)
y
ok
temp4 >
```

4. 新しい証明書を有効にするために、inspection-core を再始動します。

親トピック: Linux システムおよび UNIX システム: SSL 証明書を使用する S-TAP 認証のセットアップ

Linux システムおよび UNIX システム: x.509 証明書認証を使用するための S-TAP の構成

このタスクについて

最初に、証明書の CA および CN として割り当てた内容を記録します。覚えていない場合は、CLI コマンド show system certificate を使用して値を表示します。

```
temp4 > show system certificate
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 8 (0x8)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: C=CA, ST=British Columbia, L=Victoria, O=QA_test_vic, OU=Victoria_QA, CN=Victoria_QA_CA
        Validity
            Not Before: Nov  1 21:09:38 2010 GMT
            Not After : Nov  1 21:09:38 2015 GMT
        Subject: C=CA, ST=BC, L=Newbury, O=QA_Sample1, OU=Sample_QA, CN=sample1_qa.victoria
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (2048 bit)
                Modulus (2048 bit):
```

Guardium システムにインストールされた証明書の CN と、Guardium システム上の証明書に署名した CA の公開鍵が必要です。Guardium システムの証明書に署名したのと同じ CA により署名された証明書取り消しリストも必要になることがありますが、ここでは必須ではありません。

guard_tap.ini の関連パラメーターは以下のとおりです。

```
; Where is the CA certificate
guardium_ca_path=NULL
; What's the CN to expect from the SqlGuard certificate?
sqlguard_cert_cn=NULL
; Path to crls file or dir
guardium_crl_path=NULL
```

パラメーターに対して値を使用しない場合、その値を NULL に設定します。これは特に CRL パスに関連します。あるいは、証明書認証を中止して TLS に戻るような場合です。

手順

1. CA から送信された CA の公開鍵 (および必要な場合は CRL) を、S-TAP ホスト上のディレクトリーにコピーします。このディレクトリーを記録しておきます。
2. `guardium_ca_path=[CA へのパス.pem]` を設定します。
3. `sqlguard_cert_cn=[Guardium システムの完全 CN または部分 CN (* をワイルドカードとして使用)]` を設定します。
4. この時点で証明書失効リストを使用する場合は、`guardium_crl_path=[crl へのパス.crl]` を設定します。 次のようになります。

```
guardium_ca_path=/var/tmp/pki/Victoria_QA_CA.pem
sqlguard_cert_cn=sample1_qa.victoria
guardium_crl_path=/var/tmp/pki/Victoria_QA_CA.crl
```

5. tls=1 を変更します。
6. S-TAP を再始動します。 これで Openssl を使用して接続されます。

親トピック: Linux システムおよび UNIX システム: SSL 証明書を使用する S-TAP 認証のセットアップ

Linux システムおよび UNIX システム: Kerberos 認証データベース・トラフィック

Kerberos は、ネットワーク上で暗号化されていないパスワードの伝送を排除するネットワーク認証プロトコルです。ここでは、これが Guardium で機能する仕組みを説明します。

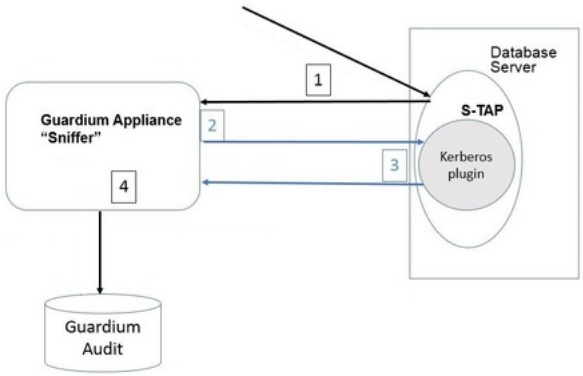
相互認証モードで機能し、認証を要求しているユーザーの ID と、要求された認証を提供するサーバーの両方を検証します。Kerberos 認証メカニズムでは、ネットワーク・サービスにアクセスするためのチケットが発行されます。 これらのチケットには、要求されたサービスに対するユーザーの ID を裏付ける暗号化されたデータ (暗号化されたパスワードなど) が含まれます。

監査およびアラートでは、アクションを実行したデータベース・ユーザーが誰かを把握することが重要です。Kerberos チケットを使用してログインする場合、データベース・ユーザーの判別は必ずしも簡単ではありません。

Guardium S-TAP はネットワーク・トラフィックのみを確認し、それを Guardium アプライアンス上のスニファーに渡します。Kerberos チケットがログインに使用されると、S-TAP はその Kerberos チケットをスニファーに渡します。一部のデータベース・サーバー・タイプでは、スニファーが Kerberos ログイン・トラフィックからデータベース・ユーザーを判別できるため、追加情報は必要ありません。その他のデータベース・サーバー・タイプでは、スニファーはいくらかサポートを必要とします。そのような機能は、S-TAP Kerberos プラグインによって実行されます。

S-TAP Kerberos プラグインはデフォルトでは使用可能になっていないため、追加の構成が必要です。

Kerberos を使用する以上は、プラグインを構成してください。プラグインを構成してもパフォーマンスへの影響やその他のマイナス面はないため、必要になったときのために構成しておいてください。



データベース、Guardium スニファー、および Guardium 監査データの間でのデータ・フローは以下のとおりです。

1. S-TAP が Kerberos 化されたデータベース・ログイン・パケットを (他のアクティビティーとともに) キャプチャーし、Guardium アプライアンスに送信します。
2. スニファーが Kerberos チケットからユーザー名を判別できる場合は、それを解析します。
3. スニファーが Kerberos チケットからユーザー名を判別できない場合は、データベース・ユーザーの要求とともに Kerberos チケットを S-TAP に送信します。S-TAP は、Kerberos プラグインが構成されているかどうかを確認します。Kerberos プラグインが構成されている場合、S-TAP はそのチケットをプラグインに与え、プラグインはチケットから DB_USER を割り出そうとします。プラグインは、データベース・ユーザー名を S-TAP に戻します。(戻されない場合、データベース・ユーザー名は提供されず、レポートにデータベース・ユーザー名が表示されません。)
4. スニファーはそのチケットのデータベース・ユーザーにユーザー名を取り込み、監査でそのユーザー名とそのユーザーの以降のデータベース・アクティビティーとを相互に関連付けられるようになりました。

- Linux システムおよび UNIX システム: Kerberos 認証がサポートされるデータベース
  Kerberos 認証がサポートされているデータベース・サーバーと、それらに Kerberos プラグインが必要かどうかのリストを確認してください。
- Linux システムおよび UNIX システム: Kerberos プラグインの使用可能化
- Linux システムおよび UNIX システム: Kerberos プラグインの構成
  Kerberos 認証 (DB_USER の識別を含む) を使用するサーバー上のデータベース・トラフィックをモニターするには、guardtap.ini ファイルと guardkerbplugin.conf ファイルを適切に構成する必要があります。
- Linux システムおよび UNIX システム: Oracle の Kerberos 構成パラメーターの検索
  Oracle Kerberos の場合、Kerberos キータブと構成ファイルの場所は sqlnet.ora で見つけます。
- Linux システムおよび UNIX システム: Sybase の Kerberos 構成パラメーターの検索

親トピック: Linux システムおよび UNIX システム: S-TAP の構成

関連情報:
標準 S-TAP を使用した Hadoop 統合

Linux システムおよび UNIX システム: Kerberos 認証がサポートされるデータベース

Kerberos 認証がサポートされているデータベース・サーバーと、それらに Kerberos プラグインが必要かどうかのリストを確認してください。

データベース	Kerberos プラグインが必要か
Db2	いいえ
Oracle	はい
Cassandra	はい
Sybase ASE	はい
HBase	はい
MongoDB	いいえ
HDFS	いいえ
Big SQL	いいえ
Hive	はい
Impala	いいえ

親トピック: Linux システムおよび UNIX システム: Kerberos 認証データベース・トラフィック

Linux システムおよび UNIX システム: Kerberos プラグインの使用可能化

このタスクについて

プラグインを使用可能にするには、guard_tap.ini 構成ファイルを編集し、**kerberos_plugin_dir** 項目が、プラグイン自体 (libguardkerbplugin.so) および構成ファイル (guardkerbplugin.conf) が配置されているディレクトリーを指すように変更します。

手順

1. デフォルト・シェル・インストールの場合: `kerberos_plugin_dir=/usr/local/guardium/guard_stap`
2. デフォルト GIM インストールの場合: (正確なパスは、使用されているソフトウェア・リリースによって異なります)
   `kerberos_plugin_dir=/usr/local/IBM/modules/STAP/10.1.3_r101299_1-1495145548`
3. デフォルト (プラグインが使用不可の場合): `kerberos_plugin_dir=NULL`

親トピック: Linux システムおよび UNIX システム: Kerberos 認証データベース・トラフィック

Linux システムおよび UNIX システム: Kerberos プラグインの構成

Kerberos 認証 (DB_USER の識別を含む) を使用するサーバー上のデータベース・トラフィックをモニターするには、guardtap.ini ファイルと guardkerbplugin.conf ファイルを適切に構成する必要があります。

このタスクについて

Kerberos プラグインのすべてのカスタマイズ設定は guardkerbplugin.conf ファイル内にあります。 このファイルのデフォルトのコンテンツは以下のとおりです。

```
# Kerberos の値
KRB5RCACHETYPE=none
KRB5_KTNAME=/path/to/kerberos/krb5.keytab
KRB5_CONFIG=/path/to/kerberos/krb5.conf
# プラグインの値
KRB5_PLUGIN_CCACHE=/path/to/kerberos/krb5cc_*
KRB5_PLUGIN_GSSAPI_LIBRARY=/path/to/lib/libgssapi_krb5.so
#KRB5_PLUGIN_DEBUG=0
```

ブランク行と # で始まる行はコメントとして扱われ、無視されます。無効な項目があると、エラーが発生し、Kerberos プラグインを実行できなくなります。

構成項目を変更した場合は、更新後の値を有効にするために、S-TAP を再始動する必要があります。

構成項目は以下のとおりです。

KRB5RCACHETYPE
: `KRB5RCACHETYPE=none`

KRB5_KTNAME
: これは、キータブ・ファイルへのパスです。これは、システムで既に使用されているキータブ・ファイル、またはプラグイン専用に Kerberos ユーティリティーで生成されたキータブ・ファイルのいずれでもかまいません。通常、このファイルの名前は krb5.keytab になります。以下に例を示します。
`KRB5_KTNAME=/home/oracle11/krb5/keytabKRB5_KTNAME=/home/sybase15/kerberos/keytab`

KRB5_CONFIG
: これは、システムで使用されている Kerberos 構成ファイルへのパスです。通常、このファイルの名前は krb5.conf になります。以下に例を示します。
`KRB5_CONFIG=/home/oracle11/krb5/krb5.conf KRB5_CONFIG=/home/sybase15/kerberos/krb5.conf`

KRB5_PLUGIN_CCACHE
: これは、Kerberos システム・キャッシュ・ファイルが配置されている場所へのワイルドカード・パスです。以下に例を示します。
`KRB5_PLUGIN_CCACHE=/tmp/krb5cc*`

標準ライブラリー・パス上にある場合は、この値を名前にすることもできます (例:
KRB5_PLUGIN_CCACHE =<library name>.so)。
複数のパスをコロン (「:」) で区切って指定できます。以下に例を示します。
KRB5_PLUGIN_CCACHE=/home/sybase16/krb5cc*:/tmp/krb5cc*
注: 必要以上に多くのファイルを指定する (例えば、/tmp/* を指定する) と、パフォーマンスが影響を受けます。

KRB5_PLUGIN_GSSAPI_LIBRARY
これは、Kerberos GSSAPI 動的ライブラリーの場所です。ほとんどのシステムでは、この名前は libgssapi_krb5.so になります。
この場所は、以下のように絶対パスで指定できます。
KRB5_PLUGIN_GSSAPI_LIBRARY=/usr/lib64/libgssapi_krb5.so KRB5_PLUGIN_GSSAPI_LIBRARY=/opt/freeware/lib64/libgssapi_krb5.so
または、ライブラリーがシステムの標準ライブラリー検索パスに配置されている場合は、以下のようにファイル名のみを指定できます。
KRB5_PLUGIN_GSSAPI_LIBRARY=libgssapi_krb5.so
注: GSSAPI ライブラリー (通常、libkrb5.so、libk5crypto.so、libkrbsupport.so) で必要なライブラリーもすべて、システム上に配置されている必要があります。
重要: Kerberos ライブラリーが標準ライブラリー・パスにない場合は、パラメーター KRB5_PLUGIN_GSSAPI_LIBRARY を使用する必要があります。コメントを外し、値を libgssapi_krb5.so の絶対パスに更新します。

KRB5_PLUGIN_DEBUG
このパラメーターは、プラグインのデバッグのみに使用されます。通常運用時には、この行をコメント化する必要があります。そうしないと、プラグイン・パフォーマンスが影響を受けます。

手順

1. guard_tap.ini ファイルで、kerberos_plugin_dir パラメーターの値を Guardium S-TAP への絶対パスに変更します。プラグインはここに配置されているためです。
   - GIM インストール済み環境: kerberos_plugin_dir=<guardium_base>/modules/STAP/current
   - S-TAP シェル・インストール済み環境: kerberos_plugin_dir=<guardium_base>/guard_stap
2. S-TAP インストール・ディレクトリーにもある guardkerbplugin.conf ファイルで、以下を構成します。
   - KRB5_KTNAME=<kerberos krb5.keytab ファイルへの絶対パス>
   - KRB5_CONFIG=<kerberos krb5.conf ファイルへの絶対パス>
   - 上記のオプション・パラメーター。チケット・キャッシュの構成パラメーターは、Kerberos プラグインがユーザーを認識しない場合に必要になることがあります。通常は複数のキャッシュ・ファイルがあるため、このパラメーターにはワイルドカードを使用できます。複数のパスをコロンで区切って指定できます。KRB5_PLUGIN_CCACHE=<kerberos krb5cc_* ファイルへの絶対パス:kerberos krb5cc_* ファイルへの追加の絶対パス:以降同様>

   注: V.10.1.2 より前の Guardium リリースでは、パラメーター allow_weak_crypto = 1 および clockskew = 600 が必要でした。これらのパラメーターは、ほとんどの場合において不要になりました。

親トピック: Linux システムおよび UNIX システム: Kerberos 認証データベース・トラフィック

Linux システムおよび UNIX システム: Oracle の Kerberos 構成パラメーターの検索

Oracle Kerberos の場合、Kerberos キータブと構成ファイルの場所は sqlnet.ora で見つけます。

このタスクについて

手順

1. grep –i KERBEROS $ORACLE_HOME/network/admin/sqlnet.ora と入力します。
   次のような出力が表示されます。

```
SQLNET.AUTHENTICATION_KERBEROS5_SERVICE = oracle
SQLNET.KERBEROS5_CONF = /home/oracle11/krb5/krb5.conf
SQLNET.KERBEROS5_REALMS = /home/oracle11/krb5/krb.realms
SQLNET.AUTHENTICATION_SERVICES= (BEQ,KERBEROS5)
SQLNET.KERBEROS5_CLOCKSKEW = 600
SQLNET.KERBEROS5_KEYTAB = /home/oracle11/krb5/keytab
SQLNET.KERBEROS5_CONF_MIT = TRUE
```

2. Kerberos キャッシュ・パラメーターを見つけるには、oklist|grep -i cache と入力します。

   次のような出力が表示されます。

```
Ticket cache: /tmp/krb5cc_500
```

親トピック: Linux システムおよび UNIX システム: Kerberos 認証データベース・トラフィック

Linux システムおよび UNIX システム: Sybase の Kerberos 構成パラメーターの検索

手順

1. klist -k と入力します。
   次のような出力が表示されます。

```
env|grep -i KRB
KRB5_KTNAME=/home/sybase15/kerberos/keytab
KRB5_CONFIG=/home/sybase15/kerberos/krb5.conf
```

2. Kerberos キャッシュ・パラメーターを見つけるには、klist -c と入力します。

   次のような出力が表示されます。

```
Ticket cache: FILE:/tmp/krb5cc_533
```

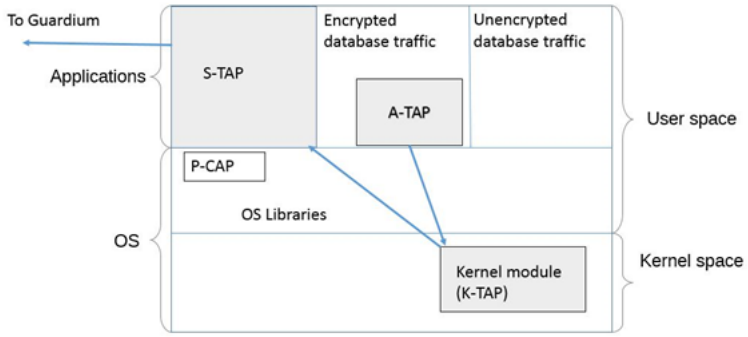
親トピック: Linux システムおよび UNIX システム: Kerberos 認証データベース・トラフィック

Linux システムおよび UNIX システム: A-TAP の管理

A-TAP は application-level tapping です。A-TAP はアプリケーション層に配置され、暗号化されたデータベース・トラフィックのモニターをサポートします。このモニターは、K-TAP によってカーネルで実行することはできません。

A-TAP メカニズムにより、データベース・サーバーの内部コンポーネント間の通信がモニターされます。データはアプリケーション層で暗号化解除され、そこで A-TAP がそのデータを取得して K-TAP に送信します。K-TAP は、データを S-TAP に渡すためのプロキシーです。そのデータは、続いて S-TAP から Guardium コレクターに送信されます。

以下の図は、データベース・サーバー上の全体のアーキテクチャーで A-TAP が配置される場所を示しています。



A-TAP はすべての S-TAP に含まれますが、A-TAP を必要とするデータベースごとに固有に構成する必要があります。

どのようなときに A-TAP を使用するか

A-TAP は、動作中の DBMS 暗号化が使用されているときに必要になりますが、その他の内部的なデータベース実装の詳細 (A-TAP を必要とする共有メモリーなど) がある場合があります。

Linux での Informix と Db2 は、出口を使用してより緊密に Guardium と統合するため、共有メモリーのサポートに対して推奨される方式です (適用可能な場合)。

制限: 32 ビット・データベースが 64 ビット・サーバーにある環境では A-TAP がサポートされていません。

モニターの制限: A-TAP は編集をサポートしていません。ブロッキングは、2.6.36 以降のリリースで Linux カーネルでサポートされます。

- Linux システムおよび UNIX システム: A-TAP の構成および保守の準備
  A-TAP を構成および保守するには、データベース管理者とシステム管理者の両方との調整が必要です。
- Linux システムおよび UNIX システム: A-TAP の構成とアクティベーション
  各 A-TAP を構成し、アクティブにします。
- Linux システムおよび UNIX システム: A-TAP アクティブ化、非アクティブ化、および DB 停止、再始動のガイドライン
  A-TAP のアクティブ化と非アクティブ化、および DB の停止と再始動のタイミングを把握します。
- Linux システムおよび UNIX システム: A-TAP の guardctl ユーティリティー・コマンド
  guardctl ユーティリティーは、A-TAP の管理ツールです。A-TAP の使用を開始する前に、以下のコマンドを理解してください。
- Linux システムおよび UNIX システム: guardctl の戻りコード
  guardctl エラー・コードは、発生したエラー条件を明確にします。特に、ATAP インスタンスを管理するために別のスクリプト経由で guardctl スクリプトを呼び出す場合に役立ちます。
- Linux システムおよび UNIX システム: データベース固有の guardctl パラメーター
  各データベース・タイプには、固有の guardctl 要件があります。
- Linux システムおよび UNIX システム: A-TAP の非アクティブ化
  データベース OS をアップグレードする前に、A-TAP を非アクティブにする必要があります。STAP のアップグレードまたはアンインストールを行う前にも (GIM、RPM、またはシェル・インストーラーによってインストールされているかどうかに関係なく) ATAP を非アクティブにする必要があります。
- Linux システムおよび UNIX システム: 特殊な環境での A-TAP の構成とアクティブ化
  ゾーン、WPAR、Teradata、および Oracle には、追加の構成が必要です。
- Linux システムおよび UNIX システム: A-TAP 構成の問題のトラブルシューティング
  このセクションでは、A-TAP の構成中に起こる一般的な失敗、それらの症状、およびそれらを回避する方法をまとめます。

親トピック: Linux システムおよび UNIX システム: S-TAP の構成

Linux システムおよび UNIX システム: A-TAP の構成および保守の準備

A-TAP を構成および保守するには、データベース管理者とシステム管理者の両方との調整が必要です。

A-TAP を構成およびアクティブにするには、以下の権限が必要です。

- データベース・サーバーに対する root アクセス権限
- データベースを停止および再始動する権限

さらに、DBA と連携して、ユーティリティーに入力する必須パラメーターを取得する必要があります。 必要なパラメーターの詳細については、Linux システムおよび UNIX システム: データベース固有の guardctl パラメーターを参照してください。継続的な保守のためには、OS およびデータベースのアップグレード時に A-TAP のアクティブ化と非アクティブ化を処理するための文書化された手順が組織に用意されている必要があります。Linux システムおよび UNIX システム: A-TAP アクティブ化、非アクティブ化、および DB 停止、再始動のガイドラインを参照してください。クラスター環境では、すべてのノードで A-TAP を構成し、アクティブにする必要があります。

ほとんどの場合、A-TAP のアクティブ化、アップグレード、または非アクティブ化には Guardium の guardctl ユーティリティーを使用します。また、guardctl を ATAP に対するユーティリティー・インターフェースとして使用するラッパー・スクリプトを実装し、独自のユーザー・エクスペリエンスを提供することもできます。guardctl ユーティリティーの構文とオプションについて詳しくは、Linux システムおよび UNIX システム: A-TAP の guardctl ユーティリティー・コマンドを参照してください。

作業を開始する前に、以下を実行してください。

- S-TAP がインストール済みで、K-TAP が有効であることを確認します。
- データベース・サーバーに対する root 権限があることを確認します。
- ご使用のデータベースに当てはまるLinux システムおよび UNIX システム: データベース固有の guardctl パラメーターを参照し、ユーティリティーの実行に必要なパラメーターがあることを確認します。

親トピック: Linux システムおよび UNIX システム: A-TAP の管理

Linux システムおよび UNIX システム: A-TAP の構成とアクティベーション

各 A-TAP を構成し、アクティブにします。

このタスクについて

前提条件:

- S-TAP がインストールされていること。
- ソフトウェアが GIM でインストールされている場合は、GIM_ROOT_DIR がモジュールの絶対パス (/usr/local/guardium/modules など) であることを確認します。
- ユーザーが guardium グループに対して許可されている必要があります。guardium グループが LDAP で作成されている場合は、同じ GID を使用して guardium というローカル・グループを作成します (データベース・ユーザーを許可するときにこのグループに追加されます)。あるいは、guardium グループ ID を /etc/passwd のデータベース・ユーザーに追加します。v10.5 以上のシェル・インストールでは、検査エンジン db_user が指定されている場合は、LDAP 環境であってもユーザーを許可する必要はありません。GIM インストールでは、引き続きデータベース・ユーザーを許可する必要があります。

A-TAP を管理するには、すべての機能をユーザー root として管理する方法と、db ユーザーとして管理する方法の 2 とおりがあります。db ユーザーのオプションでは、A-TAP の構成、アクティブ化、非アクティブ化、およびインスツルメンテーションを行うことができますが、すべての機能を実行できるわけではありません。つまり、非 root ユーザーは、root ユーザーを必要とすることなく、A-TAP の日常のアクティビティーを処理できます。guardctl ヘルプ・ウィンドウに、ログインしているユーザーに許可されているコマンドがリストされます。機能は以下のとおりです。

- 非 root として A-TAP インスタンスをアクティブ化する場合、現行ユーザーは、インスタンス構成で指定された db_user でなければならず、S-TAP 構成で一致する検査エンジンの db_user として指定されている必要があります。
- 非 root ユーザーは、最初に root ユーザーによって構成された A-TAP インスタンスの管理 (構成、アクティブ化、非アクティブ化、およびインスツルメンテーション) を行うことはできません。
- root ユーザーは、非 root ユーザーによって作成された A-TAP インスタンスをアクティブ化および非アクティブ化することができますが、インスタンス名を ${DB_USER}/${DB_INSTANCE} として指定する必要があります。

ユーザーを許可することはオプションです。guard_tap.ini に db_user が指定されている場合は、ユーザーを許可する必要はありませんが、許可することもできます。guard_tap.ini に db_user が指定されていない場合は、ユーザーを許可する必要があります。guardctl で非 root ユーザーとしてどのアクションを実行することもできません。

手順

1. guard_tap.ini ファイルで ktap_installed=1 であることを確認します。
2. すべてのアクティブ・データベース・セッションからログオフし、データベースを停止します。データベース管理ユーザーのプロセスがすべて停止されることが重要です。例えば Oracle の場合は ps -ef | grep oracle を実行します
3. root ユーザーとして、以下のように guardctl ユーティリティーに authorize-user コマンドを指定して使用し、データベース管理ユーザーにトラフィックを記録することを許可します。
   ```
   <guardium_base>/xxx/guardctl authorize-user <user-name>
   ```

   シェル・インストーラー、postgres 許可ユーザー
   `/usr/local/guardium/guard_stap/guardctl authorize-user postgres` ユーザー「postgres」にトラフィックのログ記録を許可します。
   シェル・インストーラー、postgres の許可を確認
   `/usr/local/guardium/guard_stap/guardctl is_user_authorized postgres` ユーザー「postgres」は許可されています。
   GIM インストール、postgres 許可ユーザー
   `/usr/local/guardium/modules/ATAP/current/files/bin/guardctl authorize_user postgres` ユーザー「postgres」にトラフィックのログ記録を許可します
   シェル・インストーラー、Greenplum 許可ユーザー
   `/usr/local/guardium/guard_stap/guardctl authorize-user <gpadmin>` ユーザー「<gpadmin>」にトラフィックのログ記録を許可します

4. 以下のようにして構成パラメーターを格納します。
   a. ご使用のデータベース・タイプおよびプラットフォームに必要なパラメーターを判別するには、Linux システムおよび UNIX システム: データベース固有の guardctl パラメーターを参照してください。
   b. 以下のように guardctl ユーティリティーの store-conf コマンドを使用してデータベース・インスタンスの構成を格納します。
   ```
   <guardium_base>/xxx/guardctl db_instance=<instance> [<name>=<value> ...] store-conf
   ```

   シェル・インストーラー、Linux 上の Oracle の store-conf
   ```
   /usr/local/guardium/guard_stap/guardctl --db-user=oracle11 --db-type=oracle --db-instance=on12rh60 --db-
   home=/home/oracle11/product/11.1.0/db_1 --db-version=11.2 store-conf
   ```
   GIM インストール、Linux 上の Oracle の store-conf
   ```
   /usr/local/guardium/modules/ATAP/current/files/bin/guardctl db_instance=$ORACLE_SID db_home=$ORACLE_HOME db_type=oracle
   db_user=oracle12 db_version=12 store-conf
   ```
   シェル・インストーラー、Linux 上の Greenplum の store-conf
   ```
   /usr/local/guardium/guard_stap/guardctl --db-user=<gpadmin> --db-type=greenplum -db-home=<db_user home directory> --db-
   instance=<greenplum> --db-base==<db_user home directory> store-conf
   ```

   注: Guardium V10.1 以上では、インスツルメンテーションはアクティブ化中に自動的に行われるため、明示的なインスツルメンテーションはありません。
5. A-TAP をアクティブにします。
   a. `<guardium_base>/xxx/guardctl db_instance=<instance> activate` と入力します。

シェル・インストーラー、Linux 上の Oracle の activate
```
/usr/local/guardium/guard_stap/guardctl --db-instance=onrh60x activate
```
GIM インストール、Linux 上の Oracle の activate
```
/usr/local/guardium/modules/ATAP/current/files/bin/guardctl --db-instance=onrh60x activate
```
シェル・インストーラー、Linux 上の Greenplum の activate
```
/opt/guardium/guard_stap/guardctl --db-type=greenplum --db-home=/usr/local/greenplum-db-4.3.4.0 --db-user=gpadmin
--db-instance=greenplum --db-base=<db_user home directory> activate
```

注: A-TAP は、Guardium GUI の検査エンジン構成の「暗号化」チェック・ボックスを使用して任意でアクティブにすることができますが、GUI でアクティブ化する利点はありません。このオプションは、Linux プラットフォームでは使用できません。

b. 次のように guardctl ユーティリティーの list-active コマンドを使用して、インスタンスがアクティブ化されていることを確認します:
<guardium_base>/xxx/guardctl list-active

例: `<guardium_base>/xxx/guardctl list-active oracle`

6. データベース・サーバーを再始動します。

親トピック: Linux システムおよび UNIX システム: A-TAP の管理

Linux システムおよび UNIX システム: A-TAP アクティブ化、非アクティブ化、および DB 停止、再始動のガイドライン

A-TAP のアクティブ化と非アクティブ化、および DB の停止と再始動のタイミングを把握します。

A-TAP のための再始動/ロード/アクティブ化の要件

シナリオ	説明
Oracle クラスター環境に UNIX A-TAP をインストールした後	すべてのデータベース・インスタンス、およびすべてのクラスター間プロセスを再始動する必要があります
A-TAP をアクティブにする前	データベースを停止します
A-TAP をアクティブにした後	データベースを再始動します
A-TAP を非アクティブにする前	データベースを停止します
データベースをアップグレードする前 (フィックスパックの適用など)	A-TAP を非アクティブにします
S-TAP をアップグレードする前	A-TAP を非アクティブにします

親トピック: Linux システムおよび UNIX システム: A-TAP の管理

Linux システムおよび UNIX システム: A-TAP の guardctl ユーティリティー・コマンド

guardctl ユーティリティーは、A-TAP の管理ツールです。A-TAP の使用を開始する前に、以下のコマンドを理解してください。

guardctl ユーティリティー

A-TAP を管理するには、すべての機能をユーザー root として管理する方法と、db ユーザーとして管理する方法の 2 とおりがあります。db ユーザーのオプションでは、A-TAP の構成、アクティブ化、非アクティブ化、およびインスツルメンテーションを行うことができますが、すべての機能を実行できるわけではありません。つまり、非 root ユーザーは、root ユーザーを必要とすることなく、A-TAP の日常のアクティビティーを処理できます。guardctl ヘルプ・ウィンドウに、ログインしているユーザーに許可されているコマンドがリストされます。機能は以下のとおりです。

- 非 root として A-TAP インスタンスをアクティブ化する場合、現行ユーザーは、インスタンス構成で指定された db_user でなければならず、S-TAP 構成で一致する検査エンジンの db_user として指定されている必要があります。
- 非 root ユーザーは、最初に root ユーザーによって構成された A-TAP インスタンスの管理 (構成、アクティブ化、非アクティブ化、およびインスツルメンテーション) を行うことはできません。
- root ユーザーは、非 root ユーザーによって作成された A-TAP インスタンスをアクティブ化および非アクティブ化することができますが、インスタンス名を ${DB_USER}/${DB_INSTANCE} として指定する必要があります。

ユーザーを許可することはオプションです。guard_tap.ini に db_user が指定されている場合は、ユーザーを許可する必要はありませんが、許可することもできます。guard_tap.ini に db_user が指定されていない場合は、ユーザーを許可する必要があります。guardctl で非 root ユーザーとしてどのアクションを実行することもできません。

guardctl ユーティリティーは、<guardium_base>/guard_stap ディレクトリーの下にインストールされます。ここで、<guardium_base> は Guardium ソフトウェアがインストールされているディレクトリーです。GIM インストール済み環境の guardctl の場合は、<guardium_base>/modules/ATAP/current/files/bin の下にインストールされます。

構文

<guardium_base>/xxx/guardctl [<parameter>=value>] [<parameter>=<value> ...] <command> [-q | -v | -qv]

Linux システムおよび UNIX システム: データベース固有の guardctl パラメーターに記載されているパラメーターを参照してください。

-q、-v、-qv のフラグ

以下のフラグを使用して出力を管理します。

- -q (抑制): 名前/値ペアを除き、すべての出力を抑制します
- -v (値のペア): 各コマンドに関連する名前/値ペアを追加します

- -qv: 名前/値ペアのみを出力します

出力は、コマンドのタイプによって異なります。

- 構成済みのすべてのインスタンスに対してアクションを実行するコマンド
  - overall_rv および overall_msg を除き、各インスタンスのすべての名前/値ペアを出力します
  - 末尾に overall_rv の名前/値ペアを出力します。値は以下のいずれかです。
    - 0 (成功)、すべての報告が成功した場合のみ
    - 1 (失敗)、いずれかの報告に何らかの失敗があった場合
  - 末尾に overall_msg の名前/値ペアを出力します。
  - 「overall_rv」 の名前/値ペアで報告された値を戻します。
- 1 つのインスタンスでアクションを実行するコマンド
  - overall_rv および overall_msg を除き、すべての名前/値ペアを出力します
  - 「rv」の名前/値ペアで報告された値を戻します。
- パラメーターの格納、パラメーターの出力、または状況の確認を行うコマンド
  - 名前/値ペアを出力しません。

名前/値ペアの出力は以下のようになります。

```
db_instance: ${db_instance}
db_user: ${db_user}
db_base: ${db_base}
db_home: ${db_base}
db_version: ${db_version}
db_type: ${db_type}
is_active: ${is_active} (「yes」または「no」)
is_instrumented: ${is_db_instrumented} (「yes」または「no」)
msg: some string
rv: ${retval}
overall_rv: ${retval}
overall_msg: (string)
```

commands

コマンド	記述
activate	保管されたパラメーターを使用して、指定されたデータベース・インスタンスの A-TAP をアクティブにします。v10.1.3 以上: -v または -qv を指定した場合は、名前/値ペアを出力します。 V10.1.3 では、既にアクティブなインスタンス (DB が実行中であるかどうかは関係ない) をアクティブ化してもエラーは生成されません。
authorize-user	ユーザーを「guardium」許可グループに追加します。
deactivate	指定された単一のデータベース・インスタンスの A-TAP を非アクティブにします。v10.1.3 以上: -v または -qv を指定した場合は、名前/値ペアを出力します。 Guardium V10.1.3 からは、既に非アクティブなインスタンス (DB が実行中であるかどうかにかかわらず) を非アクティブ化してもエラーは生成されません。
deactivate-all	指定されたデータベース・インスタンスのリストの A-TAP を非アクティブにします。データベース・インスタンスを指定しないと、すべてのアクティブ A-TAP が非アクティブにされます。v10.1.3 以上: -v または -qv を指定した場合は、各インスタンスの名前/値ペアを出力します。オプションで db-type を指定し、グループ (例えば、すべての Oracle など) を非アクティブにできます。追加の名前/値ペアについては、最後に「overall_rv={0、1}」を指定します。すべてのインスタンスに対して rv=0 の場合、成功 (0) を戻します。少なくとも 1 つのインスタンスで rv != 0 が報告される場合は、失敗 (1) を戻します。
deinstrument	指定した Oracle DB のインスツルメンテーションを削除します。v10.1 以上は不要です。インスツルメンテーションの削除が必要な場合は、非アクティブ化する際に自動的に行われます。V10.1.3 以上: -v または -qv を指定した場合は、名前/値ペアを出力します。 V10.1.3 以上では、DB が実行中であっても、アクティベーション状況を問わず、インスツルメンテーションされていないインスタンスからインスツルメンテーションを削除してもエラーが生成されません。
dump-params	パラメーターの現行値をダンプします。
get-statistics	A-TAP の統計を取得します。統計には、どの ATAP がアクティブであるのか、非アクティブであるのか、および誤った中間の状態にあるのか (この状態が発生することはまずありませんが、通常、ATAP がアクティブになっているときに DB が更新されると発生します) に関する情報が含まれます。
help	デフォルトのコマンドで、サポートされるコマンド、パラメーターおよびそれらのデフォルト値のリストを印刷します。 この情報を使用して、ユーザー・タイプに応じて実行できるコマンドを確認します。
instrument	再リンクされ、インスツルメンテーションされた Oracle を明示的に作成します。インスツルメンテーションが必要な場合は、通常はアクティブ化する際に自動的に行われます。手動でのインスツルメンテーションは、AIX 上の Oracle バージョン <= 10 でのみ必要です。 既にインスツルメンテーションされているインスタンスをインスツルメンテーションすると、エラーが返されます。v10.1.3 以上: -v または -qv を指定した場合は、名前/値ペアを出力します。
is-active	A-TAP がアクティブ化されているインスタンスが少なくとも 1 つある場合、1 を戻します。それ以外の場合は、0 を返します。
is-user-authorized	db-user (A-TAP を実行しているユーザー) が guardium グループに対して許可されていてデータベース・トラフィックを K-TAP/S-TAP に記録できるかどうかを検査します。
list-active	すべてのアクティブな A-TAP データベース・インスタンスのデータベース・インスタンス・ユーザー名をリストします。v10.1.3 以上: -v または -qv を指定した場合は、名前/値ペアを出力します。
list-configured	構成はされているが、非アクティブな A-TAP を持つデータベース・インスタンスをリストします。v10.1.3 以上: -v または -qv を指定した場合は、名前/値ペアを出力します。
oracle-relink	DB バイナリーを再リンクするために、Oracle 提供のユーティリティーを呼び出します。
prepare-libs	Zone/WPAR インストール済み環境で使用するライブラリーを準備します。

コマンド	記述
repair	このコマンドは、A-TAP がアクティブである間に DB が (誤って) アップグレードされた場合に実行します。-guard-original ファイルと -guard-instrumented ファイルの名前を変更します。修復が成功した場合、または修復が必要ない場合は成功を戻します。現在の DB 実行可能ファイルには影響しません。V10.1.3 以上: -v または -qv を指定した場合は、名前/値ペアを出力します。v10.1.4 からは、アクティブ化時および非アクティブ化時に自動的に呼び出されます。
restore-active-ataps	save-active-ataps によって以前に保存された A-TAP のアクティブ状態を復元します。インスタンスがアクティブ化に失敗した場合 (DB が実行中であったり他のエラーが原因で)、残りのインスタンスは引き続きアクティブ化を試みます。このコマンドでは、既にアクティブなインスタンスのアクティブ化がエラーにならないため、問題なく複数回実行することができます。v10.1.4 で導入されました。
save-active-ataps	現在アクティブな A-TAP の構成を単一のファイルに保存し、後でアクティブ状態に復元できるようにします。DB のアップグレードを準備するときに、deactivate-all の前に使用すると役立ちます。v10.1.4 で導入されました。
store-conf	特定のデータベース・インスタンスの構成を保管します。
store-system-conf	システム構成パラメーターを保管します。

親トピック: Linux システムおよび UNIX システム: A-TAP の管理

Linux システムおよび UNIX システム: guardctl の戻りコード

guardctl エラー・コードは、発生したエラー条件を明確にします。特に、ATAP インスタンスを管理するために別のスクリプト経由で guardctl スクリプトを呼び出す場合に役立ちます。

コード	記述	使用法
0	成功	すべてのコマンドによって返されます。 deactivate への応答として返される場合、すべてのインスタンスが非アクティブ化されています。 is-active への応答として返される場合、アクティブ・インスタンスはありません。
1	誤ったパラメーター	パラメーターが無効であるか欠落している場合に、すべてのコマンドによって返されます。
2	認識されないインスタンスで is-active が呼び出された	指定された db-instance が guardctl で認識されず、そのためアクティブかどうかを判別できない場合に、is-active によって返されます。
20	データベースの実行中にインスタンスをアクティブ化しようとしたが、まだアクティブでない	activate によって返され、DB インスタンスが実行中であるためアクティブ化を行えなかったことを示します。
21	データベースの実行中にインスタンスを非アクティブ化しようとしたが、まだ非アクティブでない	deactivate によって返され、DB インスタンスが実行中であるため非アクティブ化を行えなかったことを示します。
22	ユーザーは許可されていない	instrument と activate によって返され、指定された db-user が「guardium」グループのメンバーとして許可されていないことを示します。修正するには、authorize-user を実行します。
23	db-home パラメーターが guard_tap.ini の db_install_dir パラメーターと一致しない	store-conf および activate によって返され、現在の guard_tap.ini に、db_install_dir ATAP パラメーターに一致する db_home で構成された IE がないことを示します。これらのいずれかを正しい値に調整しないと、STAP が実行されない恐れがあります。
24	実行可能ファイルが ATAP 実行プログラムでもインスツルメンテーション・バイナリーでもないインスタンスを非アクティブにしようとした	deactivate によって返されます。このインスタンスはアクティブにしておく必要があると思われますが、バイナリーが本来あるべき姿ではありません。ATAP がアクティブな間に DB 実行可能ファイルが更新された可能性があります。repair コマンドを実行して問題を修正し、再度アクティブ化してください。
25	guard_tap.ini に encryption=1 が設定されているときに ATAP をアクティブ化しようとした	IE で encryption パラメーターが 1 に設定されている場合に、activate によって返されます。guardctl でアクティブ化せず、ini の encryption パラメーターを使用します。
26	DB 実行可能ファイルが見つからない	activate、deactivate、instrument、deinstrument、store-conf、prepare-libs、および repair によって返されます。DB 実行可能ファイルがありません (例: Oracle バイナリー自体が指定されたパスにない)。 インスタンスの構成時に使用されたパス・パラメーターを確認してください。
27	インスツルメンテーションが必要だが実施されていない	インスツルメンテーションが必要であるが、まだ実施されていない場合に、activate と store-conf によって返されます。Oracle インスツルメンテーションは、ほとんどの場合は自動的に実行されるようになりましたが、AIX および Oracle のバージョン <= 10 に対しては引き続き手動で指定する必要があります。
28	is-active でインスタンスがアクティブでないと報告される	is-active によって返されます。情報提供のみです。指定された db-instance がアクティブではありません。または、インスタンスが指定されていない場合は、アクティブなインスタンスがありません。
29	deactivate-all が正常に完了しない	少なくとも 1 つのアクティブ・インスタンスを非アクティブ化できなかった場合に、deactivate-all によって返されます。
30	is-instrumented でインスタンスがインスツルメンテーションされていないと報告される	コマンドではエクスポートされません。
40	内部インスツルメンテーション・エラー	インスツルメンテーションを完了できなかった場合に、instrument によって返されます。
41	内部インスツルメンテーション・エラー	インスツルメンテーションを完了できなかった場合に、instrument によって返されます。
42	内部インスツルメンテーション・エラー	インスツルメンテーションを完了できなかった場合に、instrument によって返されます。

コード	記述	使用法
43	インスツルメンテーション・エラー。元のバイナリーを保存できない	-guard-original ファイルが既に存在する場合に、instrument によって返されます。A-TAP がインスツルメンテーションとともに現在アクティブであるか、A-TAP は非アクティブであるがインスツルメンテーションがまだアクティブであるかのいずれかです。 後続の instrument および activate が実施される前に、非アクティブ化してインスツルメンテーションを削除します。
44	インスタンスの実行中にインスツルメンテーションを試行し、まだインスツルメンテーションされていない	DB インスタンスが現在実行中のときに instrument によって返されます。DB インスタンスを停止してから、再びインスツルメンテーションを試行してください。
45	A-TAP がアクティブな状態でインスツルメンテーションを試行し、まだインスツルメンテーションされていない	A-TAP は既にアクティブであるが、インスツルメンテーションがアクティブでない場合に、instrument によって返されます。これは、インスツルメンテーションを必要としない Oracle 構成から、必要とする構成に切り替える場合に発生する可能性があります。A-TAP を非アクティブにしてから、インスツルメンテーションを再試行してください。
46	インスツルメンテーションを試みたが、インスタンスが既にインスツルメンテーションされていた	インスタンスが既にインスツルメンテーションされている場合に instrument によって返されます。インスツルメンテーションを再実行する必要がある場合は、まずインスツルメンテーションを削除します。
93	アクティブ化中、非アクティブ化中、またはインスツルメンテーション中を除く、DB が実行中 (例: repair コマンドの実行中) であることによる詳細不明のエラー (例: repair コマンドを実行している場合)	
94	このデータベースをサポートする ATAP ライブラリーがない	instrument、deinstrument、prepare-libs、activate、deactivate、repair、list-active、および list-configured によって返されます。通常は、不明なエラーが発生したことを示します。
95	システム・エラー。グループが見つからない	activate によって返されます。guardium グループは、このシステムに認識されていない可能性があります。
96	システム・エラー。グループを作成できない	authorize-user によって返されます。guardium グループが存在しなかったため、このグループを作成しようとしましたが、失敗しました。
97	ファイル・システム・エラー。ディレクトリーまたはファイルを作成できないか、スペースの不足が発覚した	
98	サポートされないプラットフォーム	instrument、deinstrument、prepare-libs、activate、deactivate、repair、list-active、list-configured、store-conf によって返されます。ATAP で使用しようとしている DB は、このプラットフォームではサポートされていません (例えば、Linux 以外のプラットフォームでの Db2、Informix、Teradata、または mongo など)。
99	その他の詳細不明なエラー	

親トピック: Linux システムおよび UNIX システム: A-TAP の管理

Linux システムおよび UNIX システム: データベース固有の guardctl パラメーター

各データベース・タイプには、固有の guardctl 要件があります。

- Linux システムおよび UNIX システム: Db2 固有の guardctl パラメーター
  Db2 データベースに A-TAP を構成する場合には、以下の guardctl パラメーターを使用します (Linux のみ)。
- Linux システムおよび UNIX システム: Greenplumb 固有の guardctl パラメーター
  Greenplumb データベースに A-TAP を構成する場合には、以下の guardctl パラメーターを使用します。
- Linux システムおよび UNIX システム: Informix 固有の guardctl パラメーター
  Informix データベースに A-TAP を構成する場合には、以下の guardctl パラメーターを使用します。
- Linux システムおよび UNIX システム: Oracle 固有の guardctl パラメーター
  Oracle データベース用に A-TAP を構成する場合には、以下の guardctl パラメーターを使用します。
- Linux システムおよび UNIX システム: Postgres 固有の guardctl パラメーター
  Postgres データベースに A-TAP を構成する場合には、以下の guardctl パラメーターを使用します。
- Linux システムおよび UNIX システム: Sybase 固有の guardctl パラメーター
  Sybase データベースに A-TAP を構成する場合には、以下の guardctl パラメーターを使用します。

親トピック: Linux システムおよび UNIX システム: A-TAP の管理

Linux システムおよび UNIX システム: Db2 固有の guardctl パラメーター

Db2 データベースに A-TAP を構成する場合には、以下の guardctl パラメーターを使用します (Linux のみ)。

Db2 (Linux のみ) の必須パラメーター

例:

/usr/local/guardium/guard_stap/guardctl --db-user=db2inst1 --db-type=db2 --db-instance=dn0rh7x6 --db-version=10.5 store-conf

stap_directory/bin にあるスクリプト **find_db2_shmem_parameters.sh** は、検査エンジンで定義されている Db2 共有メモリー・パラメーターを出力します。root または Db2 ユーザーとして、find_db2_shmem_parameters.sh <instance name> という構文を使用して実行してください。これは任意のディレクトリーから実行できます。

必須パラメーター	値	判別方法
db_user	Db2 ユーザー名	DB インスタンス・ユーザー名を提示します
db_instance	Db2 インスタンス名	$ db2 LIST DATABASE DIRECTORY

必須パラメーター	値	判別方法
db_type	db2	
db_version	データベース・バージョン	Db2 ユーザーとして実行: $ db2level

Db2 (Linux のみ) のオプション・パラメーター

オプション・パラメーター	値	判別方法	必要なとき
db_home	DB バージョンがインストールされている場所のパス	db_base と同じ	
db_base	データベース・インスタンス・ユーザーのホーム・ディレクトリー	db_base の値は、DB インスタンス・ユーザーのホーム・ディレクトリーの正しいパスに一致している必要があります。DB_USER にすることはできません。	db_base が db_home と同じでない場合
db_bits	32 または 64	DB インスタンス・アーキテクチャー (32 ビットの場合は 32、64 ビットの場合は 64)	A-TAP がアーキテクチャーを認識できない場合にのみ必要。
db2-shmsize	131072	Db2 共有メモリー・サイズ	値がデフォルト値と異なる場合
db2-c2soffset	61440	Db2 共有メモリー・クライアント域のオフセット	値がデフォルト値と異なる場合
db2-header-offset	20	Db2 共有メモリー・ヘッダーのオフセット	値がデフォルト値と異なる場合

親トピック: Linux システムおよび UNIX システム: データベース固有の guardctl パラメーター

Linux システムおよび UNIX システム: Greenplumb 固有の guardctl パラメーター

Greenplumb データベースに A-TAP を構成する場合には、以下の guardctl パラメーターを使用します。

重要: A-TAP が構成されていると、ブロッキング (S-GATE) および編集はサポートされません。

例:

/opt/guardium/guard_stap/guardctl --db-type=greenplum --db-home=/usr/local/greenplum-db-4.3.4.0 --db-user=gpadmin --db-instance=greenplum --db-base=/usr/local/greenplum-db-4.3.4.0 activate

Greenplumb の必須パラメーター

必須パラメーター	値	判別方法
db_base	データベース・インスタンス・ユーザーのホーム・ディレクトリー	db_user が DB に対する OS ユーザーと異なる場合は必須です。この場合、対応する検査エンジンに対して、db_base 値を guard_tap.ini パラメーター db_install_dir にも設定する必要があります。
db_home	データベース実行可能ファイルがインストールされている場所のパス。	GUI の検査エンジン・パラメーター: 「プロセス名」、または guard_tap.ini パラメーター: db_exec_file
db_instance	Greenplumb インスタンス名	このインスタンスを識別するユーザー定義文字列
db_type	Greenplumb	
db_user	Greenplumb ユーザー名	データベース・インスタンス・ユーザー名。guard_tap.ini パラメーター db_user の値を使用します

親トピック: Linux システムおよび UNIX システム: データベース固有の guardctl パラメーター

Linux システムおよび UNIX システム: Informix 固有の guardctl パラメーター

Informix データベースに A-TAP を構成する場合には、以下の guardctl パラメーターを使用します。

Informix の必須パラメーター

例:

/usr/local/guardium/guard_stap/guardctl --db-user=informix --db-type=informix --db-instance=in17rh7x --db-version=11.70 store-conf

必須パラメーター	値	判別方法
db_user	Informix ユーザー名	DB インスタンス・ユーザー名を提示します
db_instance	Informix インスタンス名	Informix サーバー・インスタンス名
db_type	informix	
db_version	データベース・バージョン	Informix ユーザーとして実行: dbaccess -V

Informix のオプション・パラメーター

オプション・パラメーター	値	判別方法	必要なとき
db_home	DB バージョンがインストールされている場所を提示します。	db_base と同じ	
db_base	db_user のホーム・ディレクトリー	DB インスタンス・ユーザーのホーム・ディレクトリー。db_base の値は、DB インスタンス・ユーザーのホーム・ディレクトリーの正しいパスに一致している必要があります。DB_USER にすることはできません。	db-base が db-home と同じでない場合
db_bits	32 または 64	DB インスタンス・アーキテクチャー (32 ビットの場合は 32、64 ビットの場合は 64)	

親トピック: Linux システムおよび UNIX システム: データベース固有の guardctl パラメーター

Linux システムおよび UNIX システム: Oracle 固有の guardctl パラメーター

Oracle データベース用に A-TAP を構成する場合には、以下の guardctl パラメーターを使用します。

例:

/usr/local/guardium/guard_stap/guardctl --db-user=oracle11 --db-type=oracle --db-instance=on12rh60 --db-home=/home/oracle11/product/11.1.0/db_1 --db-version=11.2 store-conf

Oracle の必須パラメーター

必須パラメーター	値	判別方法
db_user	Oracle ユーザー名	データベース・インスタンス・ユーザー名を使用します。
db_instance	Oracle インスタンス名	$ORACLE_SID の値を使用します。
db_type	Oracle	
db_home	データベース実行可能ファイルのインストール場所。	
db_base	データベース・インスタンス・ユーザーのホーム・ディレクトリー	db_base の値は、$ORACLE_BASE またはデータベース・インスタンス・ユーザーのホーム・ディレクトリーの正しいパスと一致しなければなりません。DB_USER にすることはできません。
db_version	データベース・バージョン	次の SQL を実行: > SELECT * FROM V$VERSION

Oracle のオプション・パラメーター

オプション・パラメーター	値	判別方法	必要なとき
db_relink	no/yes	A-TAP の活動化方式	
db_use_instrumented	no/yes	A-TAP の活動化では、以前に guardctl の instrument コマンドで作成された、Oracle の再リンク済みバージョンが使用されます。	以下の場合にインスツルメンテーションが必要です。 • Windows 以外のすべてのプラットフォームでの Oracle 12 SSL • AIX での Oracle 11.2 SSL • 11.2 より前の AIX での Oracle ASO と SSL **重要:** レベル 10.1 のS-TAP では、インスツルメンテーションは「activate」コマンドまたは Guardium UI を使用して自動的に行われます。
db_bits	32 または 64	DB インスタンス・アーキテクチャー (32 ビットの場合は 32、64 ビットの場合は 64)	A-TAP がアーキテクチャーを認識できない場合にのみ必要。

親トピック: Linux システムおよび UNIX システム: データベース固有の guardctl パラメーター

Linux システムおよび UNIX システム: Postgres 固有の guardctl パラメーター

Postgres データベースに A-TAP を構成する場合には、以下の guardctl パラメーターを使用します。

Postgres の必須パラメーター

例:

/usr/local/guardium/guard_stap/guardctl --db-user=postgres --db-type=postgres --db-instance=guardium_qa --db-version=9.4 --db-base=/home/postgres94 store-conf

必須パラメーター	値	判別方法

必須パラメーター	値	判別方法
db-user	Postgres ユーザー名	DB インスタンス・ユーザー名を提示します
db_instance	Postgres インスタンス名	Postgres サーバー・インスタンス名
db_type	postgres	
db_version	データベース・バージョン	Postgres ユーザーとして以下を実行します。 pg_ctl --version

Postgres のオプション・パラメーター

オプション・パラメーター	値	判別方法	必要なとき
db_home	DB バージョンがインストールされている場所を提示します。	db_base と同じ	
db_base	db_user のホーム・ディレクトリー	DB インスタンス・ユーザーのホーム・ディレクトリー。db_base の値は、DB インスタンス・ユーザーのホーム・ディレクトリーの正しいパスに一致している必要があります。DB_USER にすることはできません。	db-base が db-home と同じでない場合
db_bits	32 または 64	DB インスタンス・アーキテクチャー (32 ビットの場合は 32、64 ビットの場合は 64)	
db-tcp-min-port	0 から任意の整数	インターセプトする TCP ポート範囲の下限	実際の IP を使用する場合
db-tcp-max-port	0 から任意の整数	インターセプトする TCP ポート範囲の上限	実際の IP を使用する場合

親トピック: Linux システムおよび UNIX システム: データベース固有の guardctl パラメーター

Linux システムおよび UNIX システム: Sybase 固有の guardctl パラメーター

Sybase データベースに A-TAP を構成する場合には、以下の guardctl パラメーターを使用します。

Sybase の必須パラメーター

例:

/usr/local/guardium/guard_stap/guardctl --db-user=sybase15 --db-type=sybase --db-instance=sn57rh7x --db-version=15 store-conf

必須パラメーター	値	判別方法	
db_home	データベースがインストールされている場所を提示します。	db_base と同じ	DB バイナリーを探すための基礎。通常は db_base の値を使用できますが、それが誤っている場合、アクティブ化する際にすぐに判明します (guardctl で、DB バイナリーが見つからないことが示されます)。
db_instance	Sybase インスタンス名	Sybase サーバー・インスタンス名。このパラメーターは、guardctl 内で ATAP インスタンスに名前を付けるために使用されます。	
db_type	sybase		
db_user	Sybase ユーザー名	データベース・インスタンス・ユーザー名。guard_tap.ini パラメーター db_user の値を使用します	
db_version	データベース・バージョン	Sybase ユーザーとして以下を実行します。 > select @@version >go	

Sybase のオプション・パラメーター

オプション・パラメーター	値	判別方法	必要なとき
db_base	データベース・インスタンス・ユーザーのホーム・ディレクトリー	DB インスタンス・ユーザーのホーム・ディレクトリー。これは、一部の IE では、guard_tap.ini 内の db_install_dir と一致している必要があります。~DB_USER のショートカットは使用せず、絶対パスを使用してください。	db_home を別個に指定しない場合は、db_base の値を db_home の値として使用してください。

オプション・パラメーター	値	判別方法	必要なとき
db_bits	32 または 64	DB インスタンス・アーキテクチャー (32 ビットの場合は 32、64 ビットの場合は 64)	A-TAP がアーキテクチャーを認識できない場合にのみ必要。
db-tcp-min-port	0 から任意の整数	インターセプトする TCP ポート範囲の下限	暗号化されたセッションで実際の IP が報告されるようにするかどうかを指定します。このモードでは、パフォーマンスに影響が及ぶ可能性があるほか、ポート範囲を指定することで ATAP セットアップが複雑化する可能性があります。 非特定の IP モードを使用するには、ブランクのままにします。
db-tcp-max-port	0 から任意の整数	インターセプトする TCP ポート範囲の上限	暗号化されたセッションで実際の IP が報告されるようにするかどうかを指定します。このモードでは、パフォーマンスに影響が及ぶ可能性があるほか、ポート範囲を指定することで ATAP セットアップが複雑化する可能性があります。 非特定の IP モードを使用するには、ブランクのままにします。

親トピック: Linux システムおよび UNIX システム: データベース固有の guardctl パラメーター

Linux システムおよび UNIX システム: A-TAP の非アクティブ化

データベース OS をアップグレードする前に、A-TAP を非アクティブにする必要があります。STAP のアップグレードまたはアンインストールを行う前にも (GIM、RPM、またはシェル・インストーラーによってインストールされているかどうかに関係なく) ATAP を非アクティブにする必要があります。

手順

1. データベースが停止していることを確認します。すべてのアクティブ・データベース・セッションからログオフします。
2. データベースの A-TAP を非アクティブにします。

   一般的な例
   ```
   <guardium_base>/xxx/guardctl -db-instance=<instance-name> deactivate
   ```
   Greenplum の例
   ```
   /opt/guardium/guard_stap/guardctl --db-type=greenplum --db-home=/usr/local/greenplum-db-4.3.4.0 --db-user=gpadmin --db-instance=greenplum --db-base=/usr/local/greenplum-db-4.3.4.0 deactivate
   ```
3. あるいは、以下を実行してすべてのアクティブ・インスタンスを非アクティブにします。
   ```
   <guardium_base>/xxx/guardctl deactivate-all
   ```

親トピック: Linux システムおよび UNIX システム: A-TAP の管理

Linux システムおよび UNIX システム: 特殊な環境での A-TAP の構成とアクティブ化

ゾーン、WPAR、Teradata、および Oracle には、追加の構成が必要です。

- Linux システムおよび UNIX システム: ゾーン環境および WPAR 環境での A-TAP のインストールとアクティブ化
- Linux システムおよび UNIX システム: ゾーン環境および WPAR 環境での A-TAP の非アクティブ化とアンインストール
- Linux システムおよび UNIX システム: ゾーン環境および WPAR 環境での A-TAP のアップグレード
- Linux システムおよび UNIX システム: Teradata データベースでの A-TAP の構成とアクティブ化の手順
- Linux システムおよび UNIX システム: A-TAP の Oracle 構成

親トピック: Linux システムおよび UNIX システム: A-TAP の管理

Linux システムおよび UNIX システム: ゾーン環境および WPAR 環境での A-TAP のインストールとアクティブ化

このタスクについて

手順

1. 通常の方式で、STAP/KTAP をマスターまたはグローバルのゾーン/WPAR にインストールします。
2. Solarisゾーンの場合は、Oracle がインストールされているサブゾーンごとに、以下のようにして Guardium デバイスがマップされていること確認してください。
   - zoneadm -z <zonename> halt
   - zonecfg -z <zonename>
   - <zonename>> add device
   - <zonename>device> set match=/dev/ktap_xxx (Solaris 10 の場合) (katp_xxx はファイル名)
   - <zonename>device> set match=/dev/guard_ktap (Solaris 11 の場合)
   - <zonename>device> end
   - <zonename>> verify
   - <zonename>> exit
   - zoneadm -z <zonename> boot
3. KTAP デバイスが複数の場合、名前 ktap_xxxx (Solaris 10) または guard_ktap_x (Solaris 11) を使用して KTAP デバイスごとに手順を繰り返してください。
4. A-TAP インストール・ディレクトリー全体をサブゾーン/サブ WPAR にコピーします。仮に Guardium ソフトウェアが /usr/local/guardium の下のマスター・ゾーン/WPAR にインストールされており、サブゾーン/サブ WPAR に十分な空き領域がある書き込み可能ディレクトリー /usr/local が存在するとした場合には、マスタ

ーまたはグローバルのゾーン/WPAR で次のコマンドを実行します: `cd /usr/local; tar -cvf - guardium | ssh root@subzonehost 'cd /usr/local && tar -xvf -'`

5. A-TAP ライブラリーを各サブゾーン/サブ WPAR にコピーし、活動化します。
   - A-TAP をマスター・ゾーン/WPAR で活動化する場合は、guardctl を使用して通常どおりに活動化します。
     注: 活動化は guardctl を使用して行う必要があります。GUI インターフェースの検査エンジン・セクションで暗号化ボックスを有効にしたり、guard_tap.ini ファイルで encryption=1 を設定したりすることで活動化することはできません。
   - A-TAP がマスター・ゾーン/WPAR で使用されない場合は、guardctl を使用して、ライブラリーの使用準備をします。マスター・ゾーン/WPAR で以下のようにします。`/usr/local/guardium/bin/guardctl --db_instance=<instance-name> --db_type=<database-type> --db_version=<database-version> prepare-libs`
     注: A-TAP の活動化後、データベースで libguard-xxx.so が検出できないことが示される場合、このステップを再確認してください。
6. 任意の各サブゾーン/サブ WPAR で、ステップ 1 から 5 を使用してデータベース・インスタンスに A-TAP をインストールして活動化します。
   注: A-TAP (guardctl) の活動化では、以下に関する指摘と警告が出されることがあります。
   - /usr/lib の下にライブラリーをインストールする際のエラー (そのディレクトリーがグローバル/マスター・ゾーンに属しているため)
   - guard_tap.ini を、oracle ではなく oracle-guard をモニターするように変更できない (ファイルがグローバル・ゾーンにあるため)
   - S-TAP を再始動できない (マスター・ゾーンでのみ実行されているため)
7. 手動で guard_tap.ini ファイルを編集して、マスターまたはグローバルのゾーン/WPAR で guard_tap.ini ファイルを調整します。
   - 以下のように、対応する db_exec_path 行を変更します。
     - Solaris 上の Oracle の場合: db_exec_path を oracle ではなく oracle-guard-original に設定します
     - AIX 上の Oracle の場合: db_exec_path を oracle ではなく oracle-guard-instrumented に設定します
   - IE 定義で参照されるファイルおよびディレクトリー (db_install_dir および db_exec_file) を変更し、グローバル区画ではなく、WPAR のルート・ディレクトリーを基準とするようにします。 (IE オーダー、tap_identifier のストリングなどは、すべての guard_tap.ini ファイルで同じでなければなりません。)
8. S-TAP を再始動します。
9. Solaris の場合、各々のサブゾーンで guard_ktap リンクと許可を確認します。この操作は、グローバル/マスター・ゾーンから **root** として実行する必要があります。
   a. サブゾーン・デバイス・ディレクトリーに移動します。例: `cd /export/home2/zones/iris3/dev`
   b. KTAP デバイスが存在することを確認します (存在しない場合、ステップ 2 のインストールに問題があります): `ls -l kmmodreg*`
   c. guard_ktap シンボリック・リンクが存在することを確認します。`ls -l guard_ktap`
   d. 存在しない場合は、作成してください。(注: ktap_xxxxx はリストされたデバイスです): `ln -fs ktap_xxxx guard_ktap`

      以下に例を示します。

      ```
      -bash-3.00# ln -fs ktap_83164_0 guard_ktap
      -bash-3.00# ln -fs ktap_83164_1 guard_ktap1
      -bash-3.00# ln -fs ktap_83164_2 guard_ktap2
      -bash-3.00# ln -fs ktap_83164_3 guard_ktap3
      -bash-3.00# ln -fs ktap_83164_4 guard_ktap4
      -bash-3.00# ln -fs ktap_83164_5 guard_ktap5
      ```

   e. guard_ktap と ktap_xxxxx をすべてのユーザーが使用できるようにします。

      ```
      chmod 0666 ktap_xxxxx_0
      chmod 0666 ktap_xxxxx_1
      chmod 0666 ktap_xxxxx_2
      chmod 0666 ktap_xxxxx_3
      chmod 0666 ktap_xxxxx_4
      chmod 0666 ktap_xxxxx_5
      chmod 0666 guard_ktap
      chmod 0666 guard_ktap1
      chmod 0666 guard_ktap2
      chmod 0666 guard_ktap3
      chmod 0666 guard_ktap4
      chmod 0666 guard_ktap5
      ```

注: ATAP、WPAR/ゾーンを使用する場合、暗号化されたトラフィックと暗号化解除されたトラフィックでは、アナライザーに送られる際に IP が異なります。したがって、WPAR/ゾーンの db_user は無意味です。

親トピック: Linux システムおよび UNIX システム: 特殊な環境での A-TAP の構成とアクティブ化

Linux システムおよび UNIX システム: ゾーン環境および WPAR 環境での A-TAP の非アクティブ化とアンインストール

このタスクについて

手順

1. A-TAP がインストール/活動化されているすべてのサブゾーン/サブ WPAR で、以下のようにします。
   a. Linux システムおよび UNIX システム: A-TAP の非アクティブ化 のステップに従い、guardctl を使用してすべての A-TAP を非アクティブにします (AIX 上の Oracle の場合は、必要に応じてインスツルメンテーションの削除も行います)。
   b. インストール・ディレクトリーを手動で削除 (`rm -rf`) します
   c. 以下のようにして、ATAP ライブラリーを手動で削除します `find /usr/lib -type f -name 'libguard-*.so' | xargs rm -f`

   注: ライブラリーを削除する際にエラーが出されることがありますが、無視できます。
2. 通常の方式で、STAP/KTAP をアンインストールします
   a. 以下のようにして、ライブラリーを削除します `find /usr/lib -type f -name 'libguard-*.so' | xargs rm -f o`
   b. Solaris では、以下のようにして各ゾーンの構成から ktap デバイスを削除します。

      ```
      zoneadm -z <zonename> halt
      zonecfg -z <zonename>
      <zonename>> info
      ```

      ktap デバイスが検出された場合は、それを削除します。

```
/<zonename> remove device match=/dev/ktap_xxxx (Solaris 10 の場合)
/<zonename> remove device match=/dev/guard_ktap (Solaris 11 の場合)
<zonename>> verify
<zonename>> exit
zoneadm -z <zonename> boot
```

c. それぞれのサブゾーン/サブ WPAR デバイス・ディレクトリーから、以下の例のように ktap デバイス・ファイルとリンクを削除します。

```
/export/home2/zones/iris3/dev cd /export/home2/zones/iris3/dev
rm -f ktap_xxxx guard_ktap
```

d. KTAP デバイスが複数の場合、名前 ktap_xxxx (Solaris 10) または guard_ktap_x (Solaris 11) を使用して KTAP デバイスごとに手順を繰り返してください。

親トピック: Linux システムおよび UNIX システム: 特殊な環境での A-TAP の構成とアクティブ化

Linux システムおよび UNIX システム: ゾーン環境および WPAR 環境での A-TAP のアップグレード

手順

1. Solaris Zone の場合:
   a. マスター/グローバル・ゾーンで、以前にインストールした K-TAP デバイスを削除します。

```
zoneadm -z <zonename> halt
zonecfg -z <zonename>
<zonename>> info
```

   b. K-TAP デバイスが検出された場合は、それを削除します。

```
/<zonename> remove device match=/dev/ktap_xxxx (Solaris 10 の場合)
/<zonename> remove device match=/dev/guard_ktap (Solaris 11 の場合)


<zonename>> verify
<zonename>> exit
zoneadm -z <zonename> boot
```

   c. Solaris サブゾーンについては、サブゾーン・デバイス・ディレクトリーから以前の K-TAP デバイス・ファイルとリンクを削除します。サブゾーン・デバイス・ディレクトリー (例えば /export/home2/zones/iris3/dev) に移動します。

```
cd /export/home2/zones/iris3/dev
rm -f ktap_xxxx guard_ktap
```

2. Solaris Zone の場合:
   a. マスター/グローバル・ゾーンで、以下のようにして新しい K-TAP デバイスをゾーン構成に追加します。

```
zoneadm -z <zonename> halt
zonecfg -z <zonename>
<zonename>> add device

<zonename>device> set match=/dev/ktap_xxxx (Solaris 10 の場合)
<zonename>device> set match=/dev/ktap_xxxx (Solaris 11 の場合)

<zonename>device> end
<zonename>> verify
<zonename>> exit
zoneadm -z <zonename> boot
```

   b. guard_ktap リンクを追加し、アクセス権を変更します。サブゾーン・デバイス・ディレクトリー (例えば、サブゾーン・デバイス・ディレクトリーは /export/home2/zones/iris3/dev です) に移動します。

```
cd /export/home2/zones/iris3/dev
ln -fs ktap_xxxx guard_ktap
chmod 0666 ktap_xxxx
chmod 0666 guard_ktap
```

   c. 複数の K-TAP デバイスがあるため、名前 ktap_xxxx_x (solaris 10) または guard_ktap_x (solaris 11) を使用して、ktap デバイスごとにステップを繰り返します。
3. AIX WPAR の場合は、WPAR 上で K-TAP デバイスに対するアクセス権を変更します。WPAR デバイス・ディレクトリー (例えば、WPAR デバイス・ディレクトリーは /wpars/odin3/dev です) に移動します。

```
ln -fs ktap_xxxx guard_ktap
chmod 0666 ktap_xxxx
chmod 0666 guard_ktap
```

親トピック: Linux システムおよび UNIX システム: 特殊な環境での A-TAP の構成とアクティブ化

Linux システムおよび UNIX システム: Teradata データベースでの A-TAP の構成とアクティブ化の手順

ステップ 1: gtwgateway を実行しているユーザーおよびパスを判別します。

以下に例を示します。

su11u1x64-tera:~ # ps -ef | grep gtwgateway

teradata 5000 4608 0 Jan03 ? 00:00:05 /usr/tgtw/bin/gtwgateway

root 20128 20063 0 12:35 pts/0 00:00:00 grep gtwgateway

ユーザー teradata として gtwgateway を実行します。

guardctl に対してパラメーター *--db-user=teradata* を設定します。

gtwgateway のパスは /usr/tgtw/bin/gtwgateway です。これは、パラメーター tdc_gtwgateway のデフォルト値であり、この値自体は指定する必要はありません。

そうでない場合、このパラメーターは *--tdc_gtwgateway=/usr/tgtw/bin/gtwgateway* と指定する必要があります。

ステップ 2: pdemain のパスを判別します。

通常、これは /usr/pde/bin/pdemain です。

以下に例を示します。

su11u1x64-tera:~ # ps -ef | grep pdemain

root 4608 1 0 Jan03 ? 00:00:25 pdemain –debug

su11u1x64-tera:~ # ls -l /proc/4608/exe

lrwxrwxrwx 1 root tdtrusted 0 2015-01-03 01:20 /proc/4608root 20620 20063

0 12:40 pts/0 00:00:00 grep pdemain/exe ->

/opt/teradata/tdat/pde/15h.00.00.07/bin/pdemain

このファイルおよび /usr/pde/bin/pdemain の inode を調べ、それらの inode は同じであることが分かりました。

su11u1x64-tera:~ # ls -li /opt/teradata/tdat/pde/15h.00.00.07/bin/pdemain

1638875 -r-xr-xr-x 1 teradata tdtrusted 1294666 2014-01-22 01:40

/opt/teradata/tdat/pde/15h.00.00.07/bin/pdemain

su11u1x64-tera:~ # ls -li /usr/pde/bin/pdemain

1638875 -r-xr-xr-x 1 teradata tdtrusted 1294666 2014-01-22 01:40

/usr/pde/bin/pdemain

inode が同一であり、*--db-home* のデフォルト値が */usr/pde* であるため、この場合にはこのパラメーターを指定する必要はありません。そうでない場合、--db-home=/opt/teradata/tdat/pde/15h.00.00.07 または --db-home=/usr/pde を指定できます。このケースでは、両方のパスの bin/pdemain が、ハードリンクされた同一ファイルであるためです。

ステップ 3: Teradata インスタンスを停止します。

以下に例を示します。

su11u1x64-tera:~ # /etc/init.d/tgtw stop

tgtw Shutdown complete

su11u1x64-tera:~ # /etc/init.d/tpa stop

PDE stopped for TPA shutdown

ステップ 4: DB ユーザーに対して Guardium グループの権限を付与します。

以下に例を示します。

/usr/local/guardium/guard_stap/guardctl --db-instance=teradata authorize-user

ステップ 5: ステップ 1 および 2 で決定したパラメーターを使用して、A-TAP の構成を保管します。

以下に例を示します。

/usr/local/guardium/guard_stap/guardctl --db-instance=teradata

--tdc_gtwgateway=/usr/tgtw/bin/gtwgateway --db-type=teradata

--db-home=/opt/teradata/tdat/pde/15h.00.00.07 --db-user=teradata store-conf

ステップ 6: A-TAP を活動化します。

以下に例を示します。

/usr/local/guardium/guard_stap/guardctl --db-instance=teradata activate

ステップ 7: Teradata インスタンスを再始動します。

以下に例を示します。

su11u1x64-tera:~ # /etc/init.d/tpa start

Teradata Database Initiator service is starting...

Teradata Database Initiator service started successfully.

su11u1x64-tera:~ # /etc/init.d/tgtw start

tgtw Startup complete

親トピック: Linux システムおよび UNIX システム: 特殊な環境での A-TAP の構成とアクティブ化

Linux システムおよび UNIX システム: A-TAP の Oracle 構成

Oracle パッチ・インストールを処理する場合の A-TAP の手順

Oracle パッチは relink を起動して、Oracle 実行可能ファイルを置換する場合がありますが、その結果 A-TAP の機能が停止します。

正しい手順は、以下のとおりです。

1. すべての A-TAP インスタンスが非活動化されていることを確認します
2. Oracle パッチを適用します
3. A-TAP を活動化します

ただし、Oracle パッチ・インストールの前に A-TAP が正しく非活動化されなかった場合、パッチ・インストールの後でそれを非活動化しようとしないでください。代わりに、以下の手順を実行します。

1. A-TAP に問題がないことを確認します。

   ```
   grep guardium $ORACLE_HOME/bin/oracle >& /dev/null && echo "ATAP IS OK"
   ```

   a. `ATAP IS OK` が表示された場合、A-TAP は引き続きアクティブなので、何もする必要はありません。
   b. `ATAP IS OK` が表示されない場合、$ORACLE_HOME/bin/oracle-guard を削除し、A-TAP を活動化します。

すべての方法が失敗した場合は、以下のようにします。

- $ORACLE_HOME/bin/oracle-guard を削除します。
- relink all を実行します。

Oracle のアクセス許可に関する A-TAP の問題と解決策

ユーザーとグループのアクセス権に関連したいくつかの問題が発生することがあります。

- データベースをインストールしたユーザー以外のユーザーからの「BEQUEATH」アクセスでは、以下のように、アクセス権を手動で設定する必要があります。
  - sqlplus を実行しているユーザーをグループ「guardium」に追加します
  - 以下の 2 つのディレクトリーで「chmod a+rx」により読み取り権限を開きます

    ```
    /usr/local/guardium/xxx/etc/guard
    /usr/local/guardium/xxx/etc/guard/executor
    ```

  - ${ORACLE_HOME}/bin/oracle で、SUID ビットと SGID ビットがオンであることを確認します。
    - オンでない場合は、コマンド chmod ug+s ${ORACLE_HOME}/bin/oracle') を実行します。
- UID または EUID が OWNER グループ GID のメンバーでない場合、Permission denied の理由は、UID または EUID に一致するユーザーが、OWNER GID に一致するグループに属していないことです。
- グループ Guardium への自動追加を無効にする一方で、ユーザーおよびグループの追加にさまざまな OS 構文を処理せずに済むようにして、処理を簡易化するために、guardctl 内で次の 2 つのコマンドを使用できます。これらは、ATAP のアクティブ化に使用する方法 (guardctl または guard_tap.ini) にかかわらず、使用できます。
  - #/path/to/guardium/bin/guardctl is-user-authorized
  - #/path/to/guardium/bin/guardctl authorize-user ...

注: グループ Guardium は、groupdel guardium を使用して、ほとんどの OS で削除できます。ただし、削除した後で、それを正しく再作成して K-TAP デバイスのアクセス権を変更できるのは、guard_ktap_loader パラメーターだけです。
親トピック: Linux システムおよび UNIX システム: 特殊な環境での A-TAP の構成とアクティブ化

Linux システムおよび UNIX システム: A-TAP 構成の問題のトラブルシューティング

このセクションでは、A-TAP の構成中に起こる一般的な失敗、それらの症状、およびそれらを回避する方法をまとめます。

表 1. Oracle の一般的な失敗

症状	失敗	プラットフォーム	エラー・メッセージ	回避方法
活動化コマンドが失敗する。	誤った db_home パラメーター	すべて		db_home 名として必ず $ORACLE_HOME の値を指定してください。
活動化コマンドが失敗する。	OS ユーザーがログインした	すべて		OS ユーザーがログインしていないことを常に確認してください。 どのユーザーがログインしているかを確認するには、w コマンドを使用します。

症状	失敗	プラットフォーム	エラー・メッセージ	回避方法
データベースが始動しない。	誤ったインスタンス名	すべて	oracleon1jumbo-guard を実行できませんでした。該当するファイルまたはディレクトリーが存在しません。エラー: 該当するファイルまたはディレクトリーが存在しません。ORA-12547: TNS: 接続が失われました (Failed to execute oracleon1jumbo-guard: No such file or directory: No such file or directory ERROR: ORA-12547: TNS:lost contact)	db_instance 名として必ず $ORACLE_SID の値を指定してください。
トラフィックがログに記録されない。	db_version の誤りまたは欠落	AIX		バージョンは、必ず数値 (例えば 10.2 または 9.2) で指定してください。バージョン番号の小数点の後には、1 桁しか指定できません。
活動化に失敗する。	Oracle-guard-instrumented が欠落している。	AIX	Missing Oracle-guard-instrumented.	最初に instrument コマンドを実行し、再リンクされ、インスツルメンテーションされた Oracle 実行可能ファイルを作成してください。
ATAP の活動化中にエラーが発生する。	不十分なディスク・スペース、インストールの終了		一致するモジュールが見つかりました。Oracle は /ngs/lpp/guardium/modules/ATAP/current/files/lib/libguard-atap-oraclestatic-any でサポートされています。ディスク・スペースをテストしています... cp : 0653-447 131072 バイトの書き込みを要求しましたが、126976 バイトしか書き込めませんでした。(Matching module found - oracle is supported by /ngs/lpp/guardium/modules/ATAP/current/files/lib/libguard-atap-oraclesta tic-any Testing for disk space... cp : 0653-447 Requested a write of 131072 bytes, but wrote only 126976.) ディスク・スペースが不足しています。ファイルをいくつか削除し、再試行してください。(Insufficient disk space - please delete some files and try again.)	Oracle ファイルをクリーンアップして再試行してください。db_space=8 を db_space=1 に変更してください。
guard_stap ログに、guard-atap-ctl の失敗が示される。	GIM_ROOT_DIR がモジュールへの絶対パスに設定されていない (例: /usr/local/guardium/modules)			guard_tap.ini ファイルを介して A-TAP をアクティブ化すると、encryption=1 が通知なく失敗します。これは、guard_stap を手動で実行する場合に、特に重要です。guard_stap を実行する際は、この環境変数が定義されていることを確認してください。

表 2. Db2 共通の失敗

症状	失敗	プラットフォーム	エラー・メッセージ	回避方法
トラフィックがログに記録されない。	db2_* パラメーターの誤りまたは欠落	Linux		IE 構成が正しいことを確認します。stap_directory/bin の下にあるスクリプト `find_db2_shmem_parameters.sh` を実行します。Db2 インスタンス名をパラメーターとして使用し、root ユーザーまたは Db2 ユーザーとして実行してください。Db2 共有メモリー・サイズ、クライアント位置、およびヘッダー・サイズなど、共有メモリー・パラメーターが返されます。Guardium で定義されている IE パラメーターが、返された値と一致していることを確認してください。

表 3. Informix の一般的な失敗

症状	失敗	プラットフォーム	エラー・メッセージ	回避方法
トラフィックが正しくログに記録されない。	db_version の誤りまたは欠落	Linux		バージョンは、必ず数値 (例えば 7 または 11) で指定してください。

親トピック: Linux システムおよび UNIX システム: A-TAP の管理

Linux システムおよび UNIX システム: 出口ライブラリーの使用

出口ライブラリーは、出口メカニズムを使用して Guardium ライブラリーをデータベースに組み込みます。出口ライブラリー、つまり出口モジュールは、Guardium S-TAP と直接通信してデータベース・トラフィックを転送します。

- Linux システムおよび UNIX システム: Db2 Exit と S-TAP の統合
  Db2 出口メカニズムを使用すると、Guardium は、トラフィックが暗号化されているかどうか、およびローカルかリモートかに関係なく、すべての Db2 トラフィッ

クを取得できます。A-TAP も K-TAP も必要としません。

- Linux システムおよび UNIX システム: Informix 出口と UNIX S-TAP の統合
  Informix 出口の ifxguard ユーティリティー (Informix 12.10 以上) は、Informix データベースへの接続をモニターします。
- Linux システムおよび UNIX システム: Teradata 出口の統合
  Teradata 出口モジュールを使用すると、暗号化されているかいないか、およびローカルかリモートかに関係なく、Guardium が Teradata トラフィックを取得できるようになります。A-TAP も K-TAP も必要としません。

親トピック: Linux システムおよび UNIX システム: S-TAP の構成

Linux システムおよび UNIX システム: Db2 Exit と S-TAP の統合

Db2 出口メカニズムを使用すると、Guardium は、トラフィックが暗号化されているかどうか、およびローカルかリモートかに関係なく、すべての Db2 トラフィックを取得できます。A-TAP も K-TAP も必要としません。

このタスクについて

Db2 出口は、DB2_Exit メカニズムを介して Guardium ライブラリーを Db2 に組み込みます。DB2_Exit は直接 Guardium S-TAP と通信し、トラフィックが暗号化されているかどうかに関係なく、ローカルとリモートの両方の Db2 トラフィックをすべて転送します。Db2 出口は TCP トラフィックと SHM トラフィックをキャプチャーします。Db2 とともに UID チェーンを有効にした場合に消費される CPU リソースは、KTAP および UID チェーンよりも大幅に少なくなります。

Db2 出口ライブラリーは、動的リンク・ライブラリーです。Db2 データベースは、データベースの始動中にロードされます。

Db2 出口は、ファイアウォール (STAP 10.1.2 以降は、Db2 バージョン 10.1 以降も必要)、強制終了、および UID チェーンをサポートします。

K-TAP を必要とする S-TAP の他の検査エンジン (IE) がない場合、K-TAP をロードする必要はありません。guard_tap.ini で ktap_installed=0 を設定するか、その STAP の GIM ダイアログで GIM を使用して ktap_enabled を no に設定します。Linux OS と STAP のアップグレードは、K-TAP モジュールの互換性を気にすることなく実行できます。ただし、K-TAP モジュールを必要とする S-TAP の他の IE がある場合は、Linux バージョンのアップグレード時に互換性のある K-TAP モジュールが確実に使用可能になっているようにしてください。

Db2 IE の構成時に使用するための情報を Db2 サーバーから収集するには、Db2 出口ヘルス・チェック・スクリプトを使用します。このスクリプトは guard_stap bin ディレクトリーにあります。このスクリプトは絶対パスを使用して任意の場所から実行できます。このスクリプトの名前は ./db2_exit_health_check.sh [check | fix] です。デフォルトでは、各 Db2 出口 IE の一部の IE パラメーターを出力し、IE 構成の検査を実行します。IE パラメーターを修正するには修正オプションを使用します。

制限事項:

- DB2 Exit は Guardium データ・マスキング (修正/編集) をサポートしません。
- Guardium ファイアウォール (V10.1.2 以降) には Db2 バージョン 10.1 以降が必要です。
- ストアード・プロシージャー: DB2 Exit はストアード・プロシージャーをモニターします。Guardium はストアード・プロシージャーに何が含まれているかを認識しないため、プロシージャー内からの SQL はキャプチャーされません。

S-TAP を v10.6.0 以上からアップグレードする場合、データベースを停止して再始動する必要はありません。データベース・トラフィックは、アップグレード中も継続してモニターされます。アップグレード後、データベースは、引き続き完全にモニターされ、完全に作動可能ですが、前のバージョンの出口エンジンを使用しています。データベースが次に再始動するときに、最新バージョンの出口エンジンが自動的に使用されます。再始動は任意の時点で実行でき、数週間後であっても構いません。データベースが複数ある場合、またはデータベースの複数のインスタンスがある場合、それらを個別に再始動できます。同時に再始動する必要はありません。

Guardium インストーラーには、32 ビットと 64 ビットの 2 つのバージョンの Db2 出口ライブラリーがあります。インストールされている Db2 に一致したバージョンを使用してください。どちらのバージョンも、lib サブディレクトリーの Guardium インストール・ディレクトリーにあります。Linux サーバー上では、64 ビット・バージョンは lib64 にあります。

Db2 の V101FP4 バージョンと V105FP3 バージョンは UID チェーンをサポートします。

ライブラリー名

- libguard_db2_exit_32.so
- libguard_db2_exit_64.so

手順

1. Db2 のビット単位を判別します。`root` としてログインし、`db2level` を実行します。出力は、以下のようになります。
   `DB21085I インスタンス db2inst1 は、64 ビットおよび Db2 コード・リリース SQL09070 をレベル ID 08010107 で使用します (DB21085I Instance db2inst1 uses 64 bits and DB2 code release SQL09070, with level identifier 08010107)`
2. 通信バッファー出口ライブラリーの場所 (DB2PATH) を確認します。
   a. Db2 ユーザー `trip` としてログインします
   b. Db2 clp で、db2 `get database manager configuration` を実行します。
   c. 出力で、デフォルトのデータベース・パスを見つけます。デフォルトのデータベース・パス:
   `(DFTDBPATH) = /DB2/trip`
   DFTDBPATH は、環境パラメーター DB2PATH に必要な値です。
3. Db2 ユーザーとして、以下のいずれかのコマンドを入力してディレクトリーを作成します。(これはライブラリーを最初にインストールした時にのみ実行します (ディレクトリーが存在しないため))。
   - 32 ビット環境: mkdir $DB2_PATH/sqllib/security/plugin/commexit
   - 64 ビット環境: mkdir $DB2_PATH/sqllib/security64/plugin/commexit
4. Db2 ユーザーとして、コマンド ln -fs /usr/lib64/libguard_db2_exit_64.so $DB2_PATH/sqllib/security64/plugin/commexit/libguard_db2_exit_64.so を実行します。
5. root ユーザーとして、Db2 OS ユーザーを Guardium グループに追加します。 Guardium グループは、S-TAP のインストール中に作成されます。 この要件により、S-TAP によって作成される共有メモリー領域のセキュリティーが強化されます。
   a. Db2 ユーザーが「trip」の場合、「trip」が既に許可されているかどうか確認します。ATAP フォルダーの下の guardctl を使用します。

```
# /opt/IBM/guardium/module/modules/ATAP/current/files/bin/guardctl is-user-authorized trip
User 'trip' is authorized.
```

b. ユーザー trip が許可されていない場合、次のように、ここで許可します。

```
# /opt/IBM/guardium/module/modules/STAP/10.1.0_r88469_1-1468880597/guardctl authorize-user trip
guardctl authorize-user guardium
```

6. Db2 で db2 出口を有効にします (これにより、SQL トラフィックを S-TAP に送信するようにします)。
   a. db2 ユーザーとしてログインし、次のように db2 clp コマンドを使用して有効にします。
   ```
   db2 UPDATE DBM CFG USING COMM_EXIT_LIST libguard_db2_exit_64
   ```
   b. 有効になると、db2 は SQL トラフィックを STAP に送信します。以下のコマンドを入力して、db2 出口が正常に有効化されたかどうかを確認します。

   ```
   db2 get database manager configuration
   ```

   出力には以下が含まれます。

   ```
   Communication buffer exit library list (COMM_EXIT_LIST) = libguard_db2_exit_64
   ```

7. Db2 を再始動します。
   a. `db2 user` としてログインし、以下を入力します。

   ```
   # db2stop force; ipclean; db2start
   ```

   b. 応答に以下が含まれていることを確認します。

   ```
   The DB2START command completed successfully
   ```

   c. 再始動が失敗した場合、以下のコマンドを入力して、db2 出口を停止して Db2 の警告をクリアします。

   ```
   db2 UPDATE DBM CFG USING COMM_EXIT_LIST NULL
   ```

   次に、以下のコマンドを入力して再始動します。

   ```
   db2 restart
   ```

   d. 再始動しなかった場合、次のログ・ファイルを調べて手掛かりを探してください。~/sqllib/db2dump/db2diag.log
8. A-TAP がアクティブになっていない場合、DB2_EXIT に対して STAP を構成します。(A-TAP がアクティブになっている場合は、9 に進んでください)
   a. 通常通り guard_tap.ini または GIM で Db2 に対して IE を構成します。識別を容易にするために、db_type=db2 を設定してください。
   b. DB2_EXIT IE のパラメーター db_install_dir が Db2 環境変数の $DB2_HOME または $HOME の値に設定されていることを確認します。
   c. 新規構成で S-TAP を再始動します。
9. 始動時に A-TAP がアクティブになっていた場合
   a. 以下のコマンドを入力して Db2 を停止します。
   ```
   # db2stop force;ipclean
   ```
   b. 以下のコマンドを入力して A-TAP を非アクティブにします。
   ```
   # /opt/IBM/guardium/module/modules/ATAP/10.1.0_r88469_1-1468880597/files/bin/guardctl db_instance=<db_instance> [--
   force-action=yes ] deactivate
   ```
   c. 通常通り guard_tap.ini または GIM で Db2 に対して IE を構成します。識別を容易にするために、db_type=db2 を設定してください。
   d. DB2_EXIT IE のパラメーター db_install_dir が Db2 環境変数の $DB2_HOME または $HOME の値に設定されていることを確認します。
   e. 新規構成で S-TAP を再始動します。
10. ゾーン/WPAR をセットアップします。
    a. S-TAP をゾーン/WPAR にコピーします。
       i. マスター/グローバル・ゾーン/WPAR 上で (Guardium ソフトウェアが /usr/local/guardium の下のマスター・ゾーン/WPAR にインストールされており、サブゾーン/サブ WPAR 上に十分な空き領域がある書き込み可能ディレクトリー /usr/local が存在することが前提)、以下のコマンドを入力します。

       ```
       cd /usr/local
       tar -cvf - guardium | ssh root@subzonehost 'cd /usr/local && tar -xvf -'
       ```

       ii. ゾーン/WPAR 上で、以下を指定して guard_tap.ini に DB2_EXIT IE を追加します。
           - -- ktap_installed = 0
           - -- tap_run_as_root = 1
           - -- tap_ip = ゾーン/WPAR のローカル IP アドレス
           - ゾーンで S-TAP を開始するために、他の IE を指定しないでください。
    b. /var/guard ディレクトリーを作成します。
    c. S-TAP を開始します。
       - WPAR では、inittab ファイル内の utap サーバー項目を手動でコピー/追加します。
       - Solaris ゾーンでは、コマンド svcadm -v enable guard_utap を使用します。
    d. 必要に応じて tap_debug_output_level を構成します。
       注: **データベース・サーバーでのデバッグ・ロギングの影響:** ロギングは Db2 出口モジュールによって実行されます。このモジュールは Db2 によってロードされ、診断はログ・ファイルにパイピングされます。データベース・サーバーは実際のロギングを実行しているため、実行されるロギングの量に応じて何らかの影響があります。S-TAP ロギングはトラブルシューティングの一環として使用するように意図されており、標準機能ではないため、影響があるのはロギングがオンのときのみであることに注意してください。
       - S-TAP のログ・レベルが 10 の場合、S-TAP のログと db2_exit ログ (db2diag.log) の両方にデバッグ情報が記録されます。
       - S-TAP のログ・レベルが 11 の場合、db2_exit ログ (db2diag.log) のみにデバッグ情報が記録されます。

注: WPAR 環境で、ディスカバリーの実行時にインスタンス名がスレーブ・ゾーンとマスター・ゾーンで同じである場合は、マスター・ゾーンに属する 1 つの検査エンジン項目のみが追加されます。
注: 検査エンジンで tap_identifier を変更する場合、変更を Informix 出口または Db2 出口で有効にするには、データベースを再始動する必要があります。ATAP が有効になっている状態では、データベースを停止し、ATAP を非アクティブ化し、再アクティブ化し、そして最後にデータベースを再始動する必要があります。Informix 出口の場合は、ifxguard を停止してからデータベースを再始動し、その後 ifxguard を開始します。

親トピック: Linux システムおよび UNIX システム: 出口ライブラリーの使用

Linux システムおよび UNIX システム: Informix 出口と UNIX S-TAP の統合

Informix 出口の ifxguard ユーティリティー (Informix 12.10 以上) は、Informix データベースへの接続をモニターします。

このタスクについて

Informix 出口を使用すると、Guardium は Informix SQL アクティビティーのすべてのプロトコルを監査できます。これには、TCP プロトコル、共有メモリー・プロトコル、および名前付きパイプ・プロトコルが含まれます。それは、Guardium のすべての機能 (S-gate、UID チェーン、編集、照会再書き込みなど) をサポートします。Linux プラットフォームでは、A-TAP の代わりに Informix 出口を使用して、共有メモリー・トラフィックをキャプチャーできます。Informix 出口は、暗号化トラフィックもキャプチャーします。

共有ライブラリー、Informix 出口は、Guardium Unix S-TAP インストールの一部です。S-TAP には 32 ビットと 64 ビットが含まれています。それらは、
<guardium_installation_directory>/guard_stap の下にあります。以下に例を示します。
/usr/local/guardium/guard_stap/libguard_informix_exit_32.so
/usr/local/guardium/guard_stap/libguard_informix_exit_64.so.

標準ライブラリー・パスにリンクも作成されます
/usr/lib64/libguard_informix_exit_64.so -> libguard_informix_exit_64.so.10.6.0.0
/usr/lib/libguard_informix_exit_32.so -> libguard_informix_exit_32.so.10.6.0.0

この出口はリンクを使用するため、S-TAP のアップグレード後に S-TAP を再始動する必要はありません。データベースの実行中に S-TAP をアップグレードできます。データベースが次回再始動されると、更新済みの出口ライブラリーが使用されます。

手順

1. データベース・サーバーで S-TAP をインストールして開始し、informix_exit プロトコル用に検査エンジンを構成します。『Linux システムおよび UNIX システム: S-TAP エージェントのインストール』および『Linux システムおよび UNIX システム: 検査エンジン・パラメーター』を参照してください。
2. データベースにユーザー informix としてログインし、以下の UNIX コマンドを実行して、それのインスタンス名 (INFORMIXSERVER) とそれのインストール・ディレクトリー (INFORMIXDIR) を探します。
   ```
   $ echo $INFORMIXSERVER
   INFORMIXSERVER=test117
   $ echo $INFORMIXDIR
   INFORMIXDIR=/home/informix
   ```
3. ユーザー root として、ユーザー informix が guardium グループに属していることを確認します。ユーザーがグループ guardium 内にない場合は、guardctl ユーティリティーを使用して、ユーザーをグループに追加します。以下に例を示します。
   ```
   /usr/local/guardium/bin/guardctl authorize-user informix
   ```
   UNIX (AIX のみ) では以下のようにします。
   ```
   # chgroup users=informix guardium.
   ```
4. ユーザー informix として、次のコマンドを実行し、informix_exit ライブラリーへのリンクを作成します。
   ln -fs /usr/lib64/libguard_informix_exit_64.so $INFORMIXDIR/lib/libguard_informix_exit_64.so
5. Informix_exit モニターを有効にするには、ifxguard プロセスを開始する必要があります。このプロセスを初めて開始する場合は、ユーザー Informix として次のコマンドを実行します。
   ifxguard -p $INFORMIXDIR/lib/libguard_informix_exit_64.so -l $INFORMIXDIR/tmp/ifxguard.msg.txt
   ifxguard プロセスが開始された後、構成用とメッセージング用の 2 つのファイルが自動的に作成されます。構成ファイルは以下の場所に作成されます。
   $INFORMIXDIR/etc/ifxguard.$INFORMIXSERVER
   ファイルには以下の行が含まれています。
   NAME in2rh5u7_guard
   LOGFILE /home/informix12/tmp/ifxguard.msg.txt
   WORKERS 4
   LIBPATH /home/informix12/lib/libguard_informix_exit_64.so
6. ifxguard を無効にするには、コマンド ifxguard -k を実行します。 以下の出力が示されます。`ifxguard in2rh5u7_guard successfully shut down.`
7. 正常にセットアップした後に ifxguard を再開するには、コマンド ifxguard を実行します。 以下の出力が示されます。`ifxguard in2rh5u7_guard successfully shut down.`
   ```
   [informix@rh5u7x64t ~]$ ifxguard
   17:36:21 ifxguard set instance name in2rh5u7_guard Starting ifxguard in2rh5u7_guard ...
   check log file: /home/informix12/tmp/ifxguard.msg.txt.
   ```
8. S-TAP を再始動します。

親トピック: Linux システムおよび UNIX システム: 出口ライブラリーの使用

Linux システムおよび UNIX システム: Teradata 出口の統合

Teradata 出口モジュールを使用すると、暗号化されているかいないか、およびローカルかリモートかに関係なく、Guardium が Teradata トラフィックを取得できるようになります。A-TAP も K-TAP も必要としません。

このタスクについて

Teradata 16.10 以上での S-TAP には、この構成が必要です。

Teradata 出口は、出口モジュールを介して Db2 に Guardium ライブラリーを組み込みます。出口モジュールは Guardium S-TAP と直接通信して、すべての Teradata トラフィックを転送します。

Teradata 出口は、終端とファイアウォールをサポートします。UID チェーンと編集はサポートしません。

libguard_teradata_exit_64.so およびその他の Guardium ファイルのロケーションは、インストール方式や選択したディレクトリーによって異なります。

S-TAP を v10.6.0 以上からアップグレードする場合、データベースの再始動は必要ありません。 データベースの実行中に S-TAP をアップグレードできます。データベースが次回再始動されると、更新済みの出口ライブラリーが使用されます。

手順

1. 次のようにして、Teradata サービスを停止します。

```
 /etc/init.d/tpa stop
/etc/init.d/tgtw stop
```

2. Guardium で、Teradata 出口検査エンジンを以下のように構成します。

```
[DB_0]
connect_to_ip=127.0.0.1
db_exec_file=/opt/teradata/tdat/tgtw/16.00.00.05sks/bin/gtwgateway
db_install_dir=/root
db_type=trd_exit
intercept_types=NULL
tap_identifier=NULL
networks=0.0.0.0/0.0.0.0
exclude_networks=
```

3. DB で、次のようにしてディレクトリー「site」を作成します。`mkdir /opt/teradata/tdat/tgtw/site`
4. DB で、次のようにしてシンボリック・リンクを作成します。`ln -s`
   `/usr/local/guardium/modules/STAP/current/files/lib/libguard_teradata_exit_64.so`
   `/opt/teradata/tdat/tgtw/site/libtgtwmonitoring.so`
5. DB で、トラフィックをキャプチャーするために、ユーザーを guardium グループに対して許可します。root として、以下を入力します。

```
/usr/local/guardium/guard_stap/guardctl --db-user=tdatuser authorize-user
/usr/local/guardium/guard_stap/guardctl --db-user=teradata authorize-user
/usr/local/guardium/guard_stap/guardctl --db-user=root authorize-user
```

6. DB で、次のようにして出口ライブラリーを Teradata データベースにロードします。`/usr/tgtw/bin/gtwcontrol --monitorlib load=yes`
7. 次のようにして、Teradata サービスを開始します。

```
 /etc/init.d/tpa start
/etc/init.d/tgtw start
```

親トピック: Linux システムおよび UNIX システム: 出口ライブラリーの使用

Linux システムおよび UNIX システム: FileAppender に記録するための Cassandra 監査の構成

ネイティブ監査ロギングでの Cassandra/Datastax のファイル・アペンダーへのロギングを構成します。

手順

1. データベースで、ファイル dse.yaml をテキスト・エディターで開き、「Audit logging options」セクションを以下のように更新します。

```
# Audit logging options
audit_logging_options:
    enabled: true
logger: SLF4JAuditWriter
```

2. ファイルを保存して閉じます。
3. ファイル logback.xml をテキスト・エディターで開き、次のアペンダーを追加します。

```
<appender name="GuardiumAuditWriterAppender" class="ch.qos.logback.core.FileAppender">
    <file>/usr/local/guardium/guard_stap/.cassandra_audit</file>
    <encoder>
      <pattern>%msg{}GUARD_DELIM</pattern>
      <immediateFlush>true</immediateFlush>
    </encoder>
  </appender>
```

4. 新規アペンダーを監査ロガーに追加します。

```
<logger name="SLF4JAuditWriter" level="INFO" additivity="false">
    <appender-ref ref="SLF4JAuditWriterAppender"/>
    <appender-ref ref="GuardiumAuditWriterAppender"/>
  </logger>
```

5. ファイルを保存して閉じます。
6. guard_tap.ini パラメーター cassandra_audit_enabled=1 が設定されていることを確認します。 これにより、ネイティブ監査ロギングでの Cassandra/Datastax 用のファイル・アペンダー・パイプが作成されます。
7. guard_tap.ini パラメーター cassandra_audit_delimiter がデフォルト以外に設定されている場合は、ロガー構成の値「GUARD_DELIM」に同じ値が指定されていることを確認します。
8. Guardium システムで、以下を入力して、Cassandra ユーザーを「guardium」グループに追加します。

```
/usr/local/guardium/guard_stap/guard_ctl authorize-user <cassandra>
```

   これにより、cassandra ユーザーはパイプ・ファイル・アペンダーに書き込むことができるようになります。パイプに対するアクセス権により、S-TAP ユーザーは読み取り/書き込みを実行でき、guardium グループ内の全員が書き込みを実行できるようになります。

親トピック: Linux システムおよび UNIX システム: S-TAP の構成
関連資料:
Linux システムおよび UNIX システム: 一般パラメーター

Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

S-TAP 構成は、インストール後に GIM または GUI を使用して変更できます。上級者の場合は、データベース上の構成ファイルで変更できます。

注: GUI のパラメーターは安全に変更できます。GUI にないパラメーターは高度なパラメーターであり、変更する必要はほとんどありません。これらは Guardium のサポート担当員または上級者が使用します。
注意:
熟練したユーザーである場合や IBM 技術サポートに相談済みの場合を除いて、拡張パラメーターは変更しないでください。

GUI でパラメーターを変更することができます。Linux システムおよび UNIX システム: GUI からの S-TAP の構成を参照してください。

S-TAP バンドルが GIM を使用してインストールされている場合は、GIM を使用すると簡単にパラメーターを変更できます。クライアント別の設定を参照してください。

構成変更アラート: 更新された構成を S-TAP がコレクターに送信すると、コレクターは、コレクター上に最後に保管された構成と照合して新規構成を検査します。変更内容の詳細を示すアラートが生成されます。このアラートは、S-TAP イベント・ログに表示されます。セクションの追加または削除が行われた場合、そのセクションのパラメーターがリストされます。セクションのパラメーターが変更された場合、セクション名、パラメーター名、以前の値、および新しい値が表示されます。セクションのパラメーターの追加または削除が行われた場合、セクション名、パラメーター名、および値が表示されます。例:

```
UTAP 'rh7-docker1' configuration changed, differences:
Section 'TAP' parameter 'discovery_debug' changed from '0' to '1'
Section 'TAP' parameter 'stap_statistic' changed from '1' to '0'
Section 'SQLGuard_1' added
Section 'SQLGuard_1' parameter 'connection_pool_size'='0' added
Section 'SQLGuard_1' parameter 'num_main_thread'='1' added
Section 'SQLGuard_1' parameter 'primary'='2' added
Section 'SQLGuard_1' parameter 'sqlguard_ip'='gibm39' added
Section 'SQLGuard_1' parameter 'sqlguard_port'='16016' added
```

データベース・サーバーから構成ファイルを変更する必要がある場合は、このセクションで説明する手順に従ってください。guard_tap.ini ファイルには、パラメーターの多くについて説明するコメントが含まれています。

guard_tap.ini を変更した後、S-TAP を再始動する必要があります。GIM を使用している場合は、GIM により S-TAP が自動的に再始動されます。

guard_tap.ini ファイルには、Guardium 外部 S-TAP コンテナー専用に使用される [Proxy] セクションがあります。このセクションにあるパラメーターを変更しないでください。

注意:
パラメーターは、[TAP]、[SQLGuard]、[DB_<name>] のそれぞれの関連セクションに追加する必要があります。

1. root アカウントを使用して、データベース・サーバー・システムにログオンします。
2. S-TAP を停止します。
3. 構成ファイル (guard_tap.ini) のバックアップ・コピーを作成します。デフォルトのファイル・ロケーションは /usr/local/guardium/guard_stap/guard_tap.ini です。
4. 構成ファイルをテキスト・エディターで開きます。
5. 必要に応じてファイルを編集します。
6. ファイルを保存します。
7. S-TAP を再始動して、変更が取り込まれているかどうかを確認します。

- Linux システムおよび UNIX システム: Guardium ホスト (SQLGuard) パラメーター
  以下のパラメーターは、この S-TAP が接続できる Guardium システムを示します。このセクションのパラメーターはすべて、基本パラメーターであり、[SQL_GUARD] セクションに表示されます。
- Linux システムおよび UNIX システム: 一般パラメーター
  これらのパラメーターは、DB サーバー上で実行されている S-TAP および S-TAP のインストール先のサーバーの基本プロパティーを定義し、他のどのカテゴリーにも分類されません。
- Linux システムおよび UNIX システム: 検査エンジン・パラメーター
  これらのパラメーターは、DB サーバー上のデータ・リポジトリーをモニターするために S-TAP が使用する検査エンジンの動作に影響を与えます。
- Linux システムおよび UNIX システム: ファイアウォール・パラメーター
  これらのパラメーターは、ファイアウォールに関する S-TAP の動作に影響を与えます。
- Linux システムおよび UNIX システム: 照会再書き込みパラメーター
  照会再書き込みパラメーターは、ディスカバリーに関する S-TAP の動作に影響を与えます。
- Linux システムおよび UNIX システム: サーバー・サイド・マスキング (SSM) パラメーター
  サーバー・サイド・マスキング・パラメーターは、ディスカバリーに関する S-TAP の動作に影響を与えます。
- Linux システムおよび UNIX システム: discovery パラメーター
  discovery パラメーターは、データベース・インスタンスのディスカバーと現在アクティブな S-TAP への結果の送信を行うオートディスカバリー機能の動作を定義します。
- Linux システムおよび UNIX システム: アプリケーション・サーバー・パラメーター
  アプリケーション・ユーザー名をデータベース・アクティビティーとバインドする必要がある場合、これらのパラメーターは S-TAP の動作に影響を与えます。
- Linux システムおよび UNIX システム: Hadoop パラメーター
  Guardium は、Apache Ranger を使用した統合 Hortonworks ディストリビューションをサポートします。S-TAP と Ranger エージェントの間の接続に必要な S-TAP パラメーターについて説明します。
- Linux システムおよび UNIX システム: 構成監査システム (CAS) パラメーター
  これらのパラメーターは、CAS の動作に影響を与えます。
- Linux システムおよび UNIX システム: デバッグ・パラメーター
  これらのパラメーターは、S-TAP デバッグの動作に影響を与えます。
- Linux システムおよび UNIX システム: K-TAP パラメーター
  これらのパラメーターは、K-TAP の動作に影響を与えます。

親トピック: Linux システムおよび UNIX システム: S-TAP の構成

Linux システムおよび UNIX システム: Guardium ホスト (SQLGuard) パラメーター

以下のパラメーターは、この S-TAP が接続できる Guardium システムを示します。このセクションのパラメーターはすべて、基本パラメーターであり、[SQL_GUARD] セクションに表示されます。

GUI	GIM	guard_tap.ini	デフォルト値	記述
プール・サイズ		connection_pool_size	0	S-TAP と Guardium ホスト上のスニファー・プロセスとの間で開かれる接続の数。値を高くすると、TLS などの暗号化を有効にする場合に必要になる可能性があるスループットが増えます。プールされた接続の最大数は 50 です。総合計は、guard_tap.ini 内のすべての [SQLGuard_n] セクションの (connection_pool_size x num_main_threads) の合計です。 有効な値: • 0: プーリングを無効にする • 1 から 10 (定義されたホストごと) デフォルト = 0
メイン・スレッド		num_main_threads	1	S-TAP と 1 つ以上の Guardium ホストとの間で使用されるスレッドの数。 有効な値: 1 から 510 (定義されているすべての Guardium ホストの最大合計数が 510) デフォルト = 1 注: エンタープライズ・ロード・バランシングでは、1 つの管理対象ユニットに対して複数のスレッドを使用することがサポートされていません。エンタープライズ・ロード・バランシングを使用する場合には、このパラメーターを 1 に設定してください。
✔ (チェック・マークは 1 次ホストを示します)		primary		この S-TAP の 1 次 Guardium システムを示します。guard_tap.ini での設定: 1 = 1 次、2 = 2 次、3 = 3 次など
		sqlguard_port	16016	読み取り専用。S-TAP が Guardium システムに接続するために使用するポート。
Guardium ホスト	STAP_SQLGUARD_IP	sqlguard_ip	NULL	S-TAP のホストとしての役割を果たす Guardium システムの IP アドレスまたはホスト名。[SQLGuard_1]、[SQLGuard_2]、以降同様を追加することで、複数のホストを定義できます。

親トピック: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

Linux システムおよび UNIX システム: 一般パラメーター

これらのパラメーターは、DB サーバー上で実行されている S-TAP および S-TAP のインストール先のサーバーの基本プロパティーを定義し、他のどのカテゴリーにも分類されません。

これらのパラメーターは、S-TAP プロパティー・ファイルの [TAP] セクションに格納されています。

表 1. [TAP] セクションの S-TAP 構成パラメーター

GUI	GIM	guard_tap.ini	デフォルト値	記述
		tap_type		インストールされている S-TAP エージェントのタイプ。 stap=UNIX ztap=Z/OS
バージョン		tap_version		読み取り専用。DB サーバーにインストールされ、インストール時またはアップグレード時にのみファイルに追加される S-TAP のバージョン。
S-TAP ホスト	STAP_TAP_IP	tap_ip		読み取り専用。S-TAP がインストールされているデータベース・サーバー・システムの IP アドレスまたはホスト名
デバイス	STAP_DEVICES	devices	なし	listen 対象のインターフェース。ifconfig を使用して、正しいインターフェースを見つけます。
すべてが制御可能	STAP_ALL_CAN_CONTROL	all_can_control	0	0= S-TAP は 1 次 Guardium システムからのみ制御できます。1= S-TAP は任意の Guardium システムから制御できます。

GUI	GIM	guard_tap.ini	デフォルト値	記述
ロード・バランシング	STAP_PARTICIPATE_IN_LOAD_BALANCING	participate_in_load_balancing	0	以下のように、Guardium システムへのロード・バランシングを制御します。 • 0: ロード・バランシングなし。 • 1: ロード・バランシング。SQLGuard セクションで定義されている 1 次サーバーと 2 次サーバーの間でトラフィックのバランスを取ります。 • 2: 冗長。完全にミラーリングされた S-TAP によって、SQLGuard セクションで定義されているすべての 1 次サーバーと 2 次サーバーにすべてのトラフィックが送信されます。 • 3: ハードウェア・ロード・バランシング。Guardium では、F5 や Cisco などのロード・バランサーが使用されます。S-TAP はトラフィックをロード・バランサーに送信し、ロード・バランサーはそれをプール内のいずれかのコレクターに転送します。 • 4: トラフィックの分割に複数の KTAP バッファーと S-TAP スレッドが使用されます。 1 次サーバー、2 次サーバーなどのサーバーを指定するには、SQLGUARD セクションでプライマリー・パラメーターを使用します。 このパラメーターが 0 に設定されているときに、複数の Guardium システムでトラフィックをモニターしている場合は、1 次以外の Guardium システムをフェイルオーバー用に使用することができます。 注: Guardium は、v10.x S-TAP と v9.x コレクターを使用したフェイルオーバーをサポートしていません。
		connection_timeout_sec	10	S-TAP が Guardium サーバーは使用不可であると見なすまでの秒数。任意の整数値を指定できます。
TLS の使用	STAP_USE_TLS	use_tls	0	1=SSL を使用して、エージェントと Guardium システムとの間のトラフィックを暗号化します。 0=暗号化しません。**エージェントと Guardium システム間のトラフィックは平文です。** Guardium では、可能な場合は常に S-TAP とコレクター間のネットワーク・トラフィックを暗号化することを推奨しています。この暗号化を無効にする必要があるのは、パフォーマンスの優先順位がセキュリティーより高い場合のみです。 TLS が有効である場合、ログイン・パケットの暗号化解除はサポートされていません。これは、DB_USER が取り込まれず、失敗したログインがアクセスに関連付けられないことを意味します。
TLS フェイルオーバー	STAP_FAILOVER_TLS	failover_tls	0	v10.5 で非推奨になっています。 1= 何らかの理由で SSL 接続を使用できない場合は、非セキュア接続を使用するようにフェイルオーバーします。 0= セキュア接続のみを使用します。
	STAP_WAIT_FOR_DB_EXEC	wait_for_db_exec	-1	v10.5 で非推奨になっています。 S-TAP の再始動後に、そのデータベースのモニターが開始される方法を指定します。 1 以上: システムのリブートまたはユーザーが開始した S-TAP 停止/始動コマンドによって S-TAP が再始動されると、S-TAP はモニター対象として構成されているすべてのデータベースをポーリングし、それらが使用可能な場合にモニターを開始します。S-TAP によるデータベースのモニター機能を制限する構成異常 (データベース側または S-TAP 側) がある場合でも、有効な構成を備えた他のデータベースに対する S-TAP のモニターが制限されることはありません。S-TAP は正常に始動し、すべての有効な構成をモニターして、他のデータベースを継続的にポーリングし、それらが使用可能になったらモニターを開始します。障害を起こした S-TAP の状況のモニターとレポートには、既存のアラートおよびレポートの使用を推奨します。 例えば、Oracle の再リンク後、Oracle BEQ トラフィックは 15 分間ログに記録されません。これは、S-TAP が定期的に稼働し、Oracle デバイス・ノードが変更されたかどうかを検査するのにかかる時間です。 0 以下: S-TAP が db_install_dir にアクセスできない場合は、エラー・メッセージを出して終了します。STAP に複数の IE がある場合は、DB に到達できない現象が最初に発生したときに終了します。

GUI	GIM	guard_tap.ini	デフォルト値	記述

	STAP_RUN_AS_ROOT	tap_run_as_root	TAPUSER	S-TAP を通常のユーザーとして実行できるようにします。0 = Guardium ユーザーとして実行し、1= root として実行します。
				
				場合によっては、S-TAP を Guardium として (root ではなく) 実行する必要があります。これは他の問題を引き起こす可能性があるため、必要な場合にのみ使用してください。S-TAP を Guardium ユーザーとして実行すると、許可レベルが原因でデータベースまたはプロトコルが機能しなくなる場合があります。Guardium ユーザーにデータベース・パスまたは exec ファイルの読み取り許可が付与されていることを確認してください。ご使用の環境に応じて、代表的な制限事項は以下のようになります。
				
				• wait_for_db_exec が機能しない可能性があります。クラスターの場合は、Guardium ユーザー読み取り許可のデータベース・パスまたは exec ファイルを確認してください。
				
				• AIX® WPAR および Solaris Zones のデータベースが機能しない可能性があります。インストール・パスまたは実行ファイルへのアクセス権限を確認してください。
				• Oracle BEQ の場合は、データベースの始動後、または再始動後に S-TAP を再始動してください。
				• Informix® 共有メモリーの場合は、データベースの始動後、または再始動後に S-TAP を再始動してください。
				• Db2 共有メモリーの場合、許可の問題が原因で shmctl が失敗したら、ほとんどの場合に S-TAP® が root として実行されるように変更する必要があります。
				  ○ グループによる読み取りが共有メモリー・セグメントで許可されている場合は、Db2 インスタンスがユーザー (Guardium) グループに追加されていることを確認してください。ただし依然として、サーバーごとに、DB2® の構成は 1 セットのみサポートされます。
				  ○ db2 ユーザーによる読み取りのみが共有メモリー・セグメントで許可されている場合は、S-TAP を root として実行する必要があります。(Db2 共有メモリー・セッションを開き、コマンド ipcs -ma を実行し、出力で MODE を確認します)

		tap_buf_dir	NULL	S-TAP バッファー・ファイルの場所。デフォルトの場所は、$inidir/buffers です。
		tap_log_dir	NULL	S-TAP ログ・ファイルの場所: guard_stap.stdout.tx、guard_stap.stderr.txt、guard_stap.fam.txt。デフォルトでは、ログ・ファイルは /tmp に書き込まれます。
代替 IP	STAP_ALTERNATE_IPS	alternate_ips	NULL	このデータベース・サーバーへの接続に使われる代替または仮想 IP アドレスのコンマ区切りのリスト。これが使用されるのは、複数の IP または仮想 IP を持つ複数のネットワーク・カードがサーバーにある場合だけです。この S-TAP 用に定義された S-TAP ホスト IP、またはここにリストされるいずれかの代替 IP に宛先 IP が一致する場合にのみ、S-TAP はトラフィックをモニターします。このため、すべての仮想 IP をここにリストすることをお勧めします。
	TEE_ENABLED	tee_installed	0	非推奨
		tee_msg_buf_len	128	非推奨
	STAP_BUFFER_FILE_SIZE	buffer_file_size	50	拡張機能。パケット・キューに割り振られているバッファーのサイズ (MB 単位)。バッファー・サイズの設定値が大きすぎると、S-TAP を始動できないことがあります。 ファイルが 2560 MB より大きいと、この問題が生じることが認識されています。
		buffer_mmap_file	0	1=メモリー・マップ・ファイルのオプション。0=仮想メモリーの割り振り。
トレース・ファイル・ディレクトリー		tracefiles_dir		アクセス・トレーサー・ファイルが格納されるディレクトリー。デフォルトは INSTALLDIR です。
圧縮レベル	STAP_COMPRESSION_LEVEL	compression_level	0	拡張機能。圧縮レベル。 1 から 9。 0: 圧縮なし 1: 最高速度 9: 最高圧縮率 0: 圧縮なし -1: デフォルトの圧縮
		min_bytes_to_compress	500	拡張機能。圧縮するメッセージの最小サイズ。
		tap_min_heartbeat_interval	180	S-TAP がフェイルオーバーするまでの秒数。

GUI	GIM	guard_tap.ini	デフォルト値	記述
		msg_aggregate_timeout	100	K-TAP が、バッファーに累積したパケットを S-TAP に送信する時の時間 (ミリ秒単位)。任意の整数値を指定できます。
		msg_count_watermark	64	K-TAP が、バッファーに累積したパケットを S-TAP に送信する時のパケット数。任意の整数値を指定できます。
		log_program_name	0	パフォーマンスを向上させるために、ソース・プログラム名の取得を無効にすることはできますが、接続を使用していたプログラム名を識別できなくなります (ユーザーやクライアント・アドレスなどの他のすべての接続情報は入手可能です)。0 = source_program 名を Guardium システムに送信しません。1= source_program 名を Guardium システムに送信します。
		max_server_write_size	16384	S-TAP が Guardium システムに一度に送信する最大バイト数。任意の整数値を指定できます。
		guardium_ca_path	NULL	認証局証明書の場所。
		sqlguard_cert_cn	NULL	Sqlguard 証明書で予期される共通名。
		guardium_crl_path	NULL	証明書失効リストのファイルまたはディレクトリーへのパス。
		tap_failover_session_size	1024	Guardium システムごとのリスト内のフェイルオーバー・セッションの最大数。0= フェイルオーバー機能は無効です。任意の整数値を指定できます。
		tap_failover_session_quiesce	60	以前のアクティブ・サーバーからの S-TAP フェイルオーバー・リストの未使用セッションを、現在のアクティブ・サーバーから削除するフェイルオーバー後の分数。これには、セッションのポリシーの消去と、ファイアウォール・リストおよび修正リストからのセッションの削除が含まれます。
	STAP_KERBEROS_PLUGIN_DIR	kerberos_plugin_dir	NULL	Kerberos ファイルの場所。
	STAP_DB_IGNORE_RESPONSE	db_ignore_response	NULL	応答を無視するデータベース・タイプのコンマ区切りリスト。 none に設定した場合、どの応答も無視されません。all に設定した場合、すべてのデータベースからの応答が無視されます。 注: db_ignore_response=all を使用して Oracle データベースの応答が無視されるように (トラフィック負荷を削減するためキャプチャーされないように) 設定する場合、関係するのはデータベース・サーバー応答だけではないことに注意してください。 データベース・サーバー応答には、アプリケーションが以下のデータベース要求解釈のために使用する重要なデータベース・プロトコル・メタデータ情報も含まれている可能性があります。
	STAP_STATISTIC	stap_statistic	0	S-TAP が S-TAP/K-TAP についての統計情報をスニファーに送信する間隔。0= 送信しません。時間の場合は正の整数、分の場合は負の整数を指定します。
		stap_statistic_version	1	STAP 統計は、コレクターに固有のバージョンです。 1: Guardium V10 以上 0: Guardium V9
	STAP_UPLOAD_FEATURE	upload_feature	1	1 の場合、新規 K-TAP が作成されると、この S-TAP の報告先である Guardium システムに自動的にアップロードされます。
	STAP_UPLOAD_SNAPSHOTS	upload_snapshots	1	ファイル・アップロード・メカニズムを使用してスナップショットをアップロードします。
		add_to_verification schedule	0	guard_tap.ini で定義した検査エンジンを S-TAP 検査スケジュールに追加します。 S-TAP 検査はトラフィック・キャプチャーをテストします。0=OFF、1=ON でデフォルトは 0 です。
	STAP_DB_IGNORE_BYPASS_BYTES	db_ignore_response_bypass_bytes	4096	結果セットのバイト・サイズの整数。結果セットがこのサイズより大きい場合応答を無視します。

GUI	GIM	guard_tap.ini	デフォルト値	記述
	STAP_DB_IGNORE_RESETS_PER_REQUEST	db_ignore_response_resets_per_request	0	要求ごとに db_ignore_response_bypass_bytes をリセットします。 0=いいえ; 1=はい
	STAP_DB_IGNORE_RESPONSE_FILTER	db_ignore_response_filter	0.0.0.0/0.0.0.0	応答を無視する IP/マスクのコンマ区切りリスト。デフォルトでは、すべてのトラフィックがフィルタリングされます。 指定された IP/マスクに対する、DB_IGNORE_RESPONSE で指定されたタイプのデータベース応答はすべて無視されます。 0= 応答のフィルタリングは行われない 0.0.0.0/0.0.0.0= すべての IP がフィルタリングされる
	STAP_DB_IGNORE_RESPONSE_LOCAL	db_ignore_response_local	1	ローカル DB 応答のフィルタリング。このパラメーターでは TCP トラフィックはローカル・トラフィックと見なされません。 0= いいえ 1= はい
		debug_snapshot	0	拡張機能。デバッグ・ダンプを STAP から収集します。GUI (「S-TAP 制御」 > 「S-TAP コマンド」) から起動する必要があります。GUI からダンプをトリガーした後、このパラメーターはデフォルトの 0 に戻ります。
		debug_snapshot_level	1	拡張機能。デバッグ・ダンプに対して実行される tap_debug_output_level の値: • 1: 基本デバッグ • 4: 詳細デバッグ
		debug_snapshot_time	60	拡張機能。診断が実行される時間間隔 (秒単位)。任意の整数値を指定できます。
		force_log_limited	0	コレクターに対する特定のタイプの情報の送信を制御します。Guardium コレクターに個人データが保管される可能性を懸念している場合に役立ちます。 0=無制限。デフォルト 1 = 制限付きログ。個人データは削除されます。
		hunter_trace	0	UID_CHAIN を有効にします。 0: 無効にする。 1: 有効にする。Solaris ゾーンや AIX WPAR を含む、ローカル TCP/IP 接続の場合。または、appserver_installed = 1 の場合のリモート TCP/IP 接続の場合。
ロード・バランサー IP	STAP_LOAD_BALANCER_IP	load_balancer_ip		ロード・バランサー・ユニットの IP アドレス。定義されない場合、S-TAP はエンタープライズ・ロード・バランシングを使用しません。
管理対象ユニット	STAP_LOAD_BALANCER_NUM_MUS	load_balancer_num_mus	1	ロード・バランサーから要求する管理対象ユニットの数。
		merge_with_template	0	コレクターからの構成が STAP に対してプッシュされたときに、それをテンプレート構成ファイルとマージするかどうかを指定します。 0= いいえ 1= はい
		shmid_blacklist	NULL	KTAP がフィルタリングする共有メモリー ID のコンマ区切りリスト。
		shmid_blacklist_wait	0	shmid_blacklist 項目がディスカバーされるまで、インターセプトをアクティブ化するのを待機します。0: いいえ、1: はい (0)
		blacklist_shmem_ops_by_proc	NULL	KTAP は blacklist_shmem_ops_by_proc を使用して、指定されたプロセス (コンマ区切りリスト) の shmem インターセプトをフィルタリングします。

GUI	GIM	guard_tap.ini	デフォルト値	記述
		fam_enable	1	FAM モニター (クローラー) のグローバルな有効化/無効化。 0: 無効 1: 有効 FAM を実行するには、FAM ルールが定義されている必要があります。 ルールが定義されていない場合は、このパラメーターを有効に設定すると、Guardium システムへの接続がポート 16022 (暗号化を使用している場合は 16023) で開きますが、FAM は基本的に無効なままになります。
SSH デーモンに対する UID チェーンにクライアント IP を含める	STAP_UID_CHAIN_TRACE	uid_chain_sshd_ip	0	v10.1.4 で導入されました。ssh が UID チェーン内のプロセスの 1 つとして識別された場合、クライアント IP を UID チェーンにエンコードします。 0= 無効、1= 有効
		cassandra_audit_enabled	0	0: ネイティブ監査ロギングで Cassandra/Datastax 用のファイル・アペンダー・パイプを作成しません。 1: ネイティブ監査ロギングで Cassandra/Datastax 用のファイル・アペンダー・パイプを作成します。 このパラメーターを変更した後、S-TAP を再始動してください。
		cassandra_audit_delimiter	GUARD_DELIM	監査と Cassandra を区切る文字列。有効な値は、ASCI 印刷可能文字 a-zA-Z0-9-_!@#$%^&*() です。 このパラメーターを変更した後、S-TAP を再始動してください。

親トピック: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

Linux システムおよび UNIX システム: 検査エンジン・パラメーター

これらのパラメーターは、DB サーバー上のデータ・リポジトリーをモニターするために S-TAP が使用する検査エンジンの動作に影響を与えます。

これらのパラメーターは、データ・リポジトリー名を使用して、S-TAP プロパティー・ファイルの個々の [DB_<name>] 検査エンジン・セクションに保管されます。プロパティー・ファイルには、複数のセクションが存在する場合があり、各セクションは、この S-TAP によって使用される 1 つの検査エンジンを記述しています。

GUI	guard_tap.ini	デフォルト値	記述
プロトコル	db_type		モニター中のデータ・リポジトリーのタイプ。
ポート範囲	port_range_start		データベース・インスタンスに固有のポート範囲の先頭。port_range_end とともに、このデータベース・インスタンスについてモニターされるポートの範囲を定義します。通常は、単一のポートだけが範囲に含まれます。Kerberos 検査エンジンの場合は、先頭と終了の値を 88-88 に設定します。範囲を使用する場合は余分なポートを範囲に含めないでください。含めた場合、S-TAP が不要なトラフィックの分析を試みたときにリソースが過剰に消費されることがあります。
ポート範囲	port_range_end		データベース・インスタンスに固有の終了ポート範囲。
KTAP データベース実ポート	real_db_port	4100	K-TAP および PCAP では、モニター対象のデータベース・ポートまたはポート範囲を識別します。 出口ライブラリーの場合、db_home 用の値を使用します

GUI	guard_tap.ini	デフォルト値	記述
クライアント IP/マスク	networks		IP アドレス/マスク形式 (n.n.n.n/m.m.m.m) のアドレスのリストを使用して、モニターされるクライアントを識別します。不適切な IP アドレス/マスクが入力された場合、S-TAP は始動しません。有効な値: • null=すべてのクライアントを選択 • 127.0.0.1/255.255.255.255=ローカル・トラフィックのみ 「クライアント IP /マスク」(networks) と「除外クライアント IP/マスク」(exclude networks) を同時に指定することはできません。 IP アドレスがデータベース・サーバーの IP アドレスと同じで、マスク **255.255.255.255** が使用される場合には、**ローカル**・トラフィックだけがモニターされます。アドレス/マスク値 **1.1.1.1/0.0.0.0** は、すべてのクライアントをモニターします。
除外クライアント IP/マスク	exclude_networks		モニターから除外されるクライアント IP アドレスと対応するマスクのリスト。このオプションを使用すると、特定のクライアントやサブネット (またはこれらの集合) を除くすべてのクライアントをモニターするよう S-TAP を構成できます。「クライアント IP /マスク」(networks) と「除外クライアント IP/マスク」(exclude networks) を同時に指定することはできません。
TEE 聴取ポート - 実ポート	tee_listen_port	12344	推奨されません。K-TAP モニター・メカニズムが使用される場合、パラメーター real_db_port に置き換えられます。 TEE モニター・メカニズムの場合は必須でした。聴取ポートは、S-TAP がローカル・データベース・トラフィックを聴取して受け入れるポートです。実ポートは、S-TAP がトラフィックを転送する先のポートです。
接続先 IP	connect_to_ip	127.0.0.1	S-TAP がデータベースへの接続に使用する IP アドレス。一部のデータベースは、デフォルト (127.0.0.1) ではなく、マシンの実際の IP アドレスでのみローカル接続を受け入れます。K-TAP が有効になっている場合、このパラメーターは Solaris Zones および AIX WPAR に使用され、トラフィックを取り込むには、ゾーン IP アドレスでなければなりません。
データベース・ユーザー	db_user	NULL	DB サーバー・プロセスの所有者の OS ユーザー名 (大/小文字の区別あり) (例えば、oracle)。 このパラメーターは、atap_request_handler ソケットの使用を許可されるユーザーを指定します。ユーザー root を使用していない場合は必須です。有効な値に設定されない場合、A-TAP は、K-TAP にアクセスする許可を取得するためにソケットにアクセスすることができません。そのため、暗号化解除されたトラフィックを K-TAP に記録するためにグループ・メンバーシップを介した許可が必要になる可能性があります (guardctl authorize-user コマンドを使用します)。 このパラメーターを変更した後、S-TAP を再始動してください。
データベース・インストール・ディレクトリー	db_install_dir	NULL	Db2、Informix、または Oracle の場合、データベース・インストール・ディレクトリーの絶対パス名を入力します。例: /home/oracle10。その他すべてのデータベース・タイプでは、NULL と入力します。Db2 出口および Informix 出口の場合、db_install_dir がデータベースの $HOME 値 (Db2 出口の場合は $DB2_HOME 値) と完全に同じであることが必要です。そうでない場合、tap_identifier は適切に機能しません。
プロセス名	db_exec_file	NULL	Db2、Oracle、または Informix のデータベースでは、データベース実行可能ファイルの絶対パス名を入力します。次に例を示します。 • Oracle: 標準パスはありません。データベースのインストール先のディレクトリーによって異なります。 • Informix: /INFORMIXTMP/.inf.sqlexec。これは Linux を除くすべての Informix プラットフォームに適用されます。 • Linux での Informix の例: /home/informix11/bin/oninit • MYSQL: mysql • その他すべてのデータベース・タイプ: NULL
暗号化	encryption	0	Oracle (バージョン 11 と 12) および Sybase (Solaris、HPUX、および AIX 上) 用に ASO または SSL 暗号化トラフィックをアクティブ化します。 Oracle の場合、ini ファイルに db_version を指定します (例: db_version=12)。 Oracle12 SSL の場合、すべてのプラットフォームでインスツルメンテーションを実行します。 Oracle11 SSL の場合、AIX でインスツルメンテーションを実行します。 インスツルメンテーションを必要とする Oracle に対して、guard_tap.ini 内で encryption=1 (Linux ではサポートされません) を使用する場合、 そのパラメーターを設定する前にインスツルメンテーションを実行する必要があります。 一部の DB では、暗号化を有効にした後に再始動する必要があります。

GUI	guard_tap.ini	デフォルト値	記述
	load_balanced	1	1=データベース・トラフィックはロード・バランシングに参加します。0=データベース・トラフィックはロード・バランシングに参加しません。
	priority_count	20	セッション作成時に、最初の priority_count パケットは、高優先度フラグのマークを付けられ、コレクター上の特別な高優先度キューに転送されます。有効な範囲は、0 (無効) から 50 です。
インターセプト・タイプ	intercept_types	NULL	IE がインターセプトするプロトコル・タイプ。有効な値: • NULL: データベースがサポートしているすべてのプロトコルを自動的にインターセプトします。 • コンマ区切りリスト: IE はこれらのプロトコル・タイプのみをインターセプトします。
ID	tap_identifier	NULL	オプション。検査エンジンを相互に識別するために使用します。このフィールドに値を指定しない場合、Guardium は、データベース・タイプと GUI 表示シーケンス番号を使用して、固有の名前をこのフィールドに自動入力します。
データベース・バージョン	db_version	9	データベース・バージョン。
Unix ソケット・マーカー	unix_domain_socket_marker	Null	Oracle、MySQL、および Postgres の UNIX ドメイン・ソケットのマーカーを指定します。通常は、デフォルト値が正しいですが、名前付きパイプまたは UNIX ドメイン・ソケット・トラフィックが動作しない場合は、この値が正しく設定されていることを確認する必要があります。例えば、Oracle では、unix_domain_socket_marker を tnsnames.ora で定義されている IPC のキーに設定する必要があります。NULL の場合、または設定しない場合、S-TAP は、次のような定義済みのデフォルト・マーカーを使用します。* MySQL - "mysql.sock" * Oracle - "/.oracle/" * Postgres - ".s.PGSQL.5432"

以下の追加のパラメーターは、IBM Db2 データベースで使用します。

stap_directory/bin にあるスクリプト **find_db2_shmem_parameters.sh** は、検査エンジンで定義されている Db2 共有メモリー・パラメーターを出力します。root または Db2 ユーザーとして、find_db2_shmem_parameters.sh <instance name> という構文を使用して実行してください。これは任意のディレクトリーから実行できます。

表 1. Db2 検査エンジン用の追加の S-TAP 構成パラメーター

GUI	guard_tap.ini	デフォルト値	記述
Db2 共有メモリー・ 調整	db2_fix_pack_adjustment	20	データベース・タイプとして Db2 が選択され、共有メモリー接続がモニターされる場合に必須です。共有メモリー領域のサーバー部分へのオフセット。 Db2 共有メモリー・パケットの開始位置へのオフセットで、Db2 のバージョンによって異なります。8.2.1 より前では 32、8.2.1 以上では 80 です。
Db2 共有メモリー・クライアント位置	db2_shmem_client_position	61440	共有メモリー領域のクライアント部分へのオフセット。データベース・タイプとして Db2 が選択され、共有メモリー接続がモニターされる場合に必須です。値を見つけるには、スクリプト **find_db2_shmem_parameters.sh** を使用します。
	db2bp_path	Null	Db2 で ATAP を使用する場合にのみ使用します。プログラム「db2bp」(Db2 の一部) が標準の場所にある場合は、これを設定する必要はありません。非標準の場合は、このパラメーターはその場所を指します。 このパラメーターの値は、グローバル Zone/Wpar から見えるように関連 db2bp の絶対パスである必要があります。例えば、ファイルが /data/db2inst1/sqllib/bin/db2bp であり、ゾーンが /data/zones/oracle2nd/root/ にインストールされている場合、db2bp_path パラメーターに設定する必要がある db2bp への絶対パスは /data/zones/oracle2nd/root/data/db2inst1/sqllib/bin/db2bp です。
Db2 共有メモリー・ サイズ	db2_shmem_size	131072	Db2 共有メモリー・セグメント・サイズ。データベース・タイプとして Db2 が選択され、共有メモリー接続がモニターされる場合に必須です。

親トピック: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

Linux システムおよび UNIX システム: ファイアウォール・パラメーター

これらのパラメーターは、ファイアウォールに関する S-TAP の動作に影響を与えます。

これらのパラメーターは、S-TAP プロパティー・ファイルの [TAP] セクションに格納されています。

注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。

GUI	guard_tap.ini	デフォルト値	記述

GIM	guard_tap.ini	デフォルト値	記述
STAP_FIREWALL_INSTALLED	firewall_installed	0	ファイアウォール機能を有効にします。1=有効、0=無効。
STAP_FIREWALL_TIMEOUT	firewall_timeout	2	ファイアウォールがタイムアウトになった場合に Guardium システムからの判定を待機する時間 (秒単位)。接続をブロックするのか、許可するのかを知るために、firewall_fail_close 値を調べます。任意の整数値を指定できます。
STAP_FIREWALL_FAIL_CLOSE	firewall_fail_close	0	Guardium システムから判定が返されず、firewall_timeout が期限切れになった場合、firewall_close = 0 では接続が許可され、firewall_close=1 では接続がブロックされます。

GIM	guard_tap.ini	デフォルト値	記述
STAP_FIREWALL_DEFAULT_STATE	firewall_default_state	2	• 0=インストール済みポリシーのルールでトリガーされると、セッションごとにファイアウォールがアクティブ化されます。 • 1=デフォルトですべてのトラフィックに対してファイアウォール・ポリシー違反の監視が行われます。 • 2=デフォルトですべてのトラフィックに対してファイアウォール・ポリシー違反の監視が行われますが、最初の priority_count パケットでイベントによって監視がトリガーされない場合、セッションの監視はオフにされます。(V10.5 で導入。) 2 に設定された場合、ファイアウォール操作は Watch、Drop、Watch & Drop、および Unwatch の各コマンドで変更できます。firewall_default_state=2 のときに Watch コマンドを受け取った場合、2 から 1 に変更されて、接続は永続的にファイアウォール操作または照会再書き込み操作の対象となります。Drop または Watch & Drop を受け取った場合、接続は即時に強制終了します。firewall_default_state=2 のときに Unwatch コマンドを受け取った場合、2 から 0 に変更されて、接続はファイアウォール操作および照会再書き込み操作の対象から外れます。 このパラメーターを変更した後、S-TAP を再始動してください。
STAP_FIREWALL_FORCE_WATCH	firewall_force_watch	NULL	ファイアウォール機能が有効になっていて、firewall_default_state が 0 の場合、セッションは、そのクライアント IP がこの IP/MASK 値リストのいずれかと一致すると、自動的に監視されます。リスト自体は、コンマで区切られます。例: 1.1.1.1/1.1.1.1,2.2.2.2/2.2.2.2

GIM	guard_tap.ini	デフォルト値	記述
STAP_FIREWALL_FORCE_UNWATCH	firewall_force_unwatch	NULL	ファイアウォール機能が有効になっていて、firewall_default_state が 1 の場合、セッションは、そのクライアント IP がこの IP/MASK 値リストのいずれかと一致すると、自動的に監視されません。リスト自体は、コンマで区切られます。例: 1.1.1.1/1.1.1.1,2.2.2.2/2.2.2.2,

親トピック: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

Linux システムおよび UNIX システム: 照会再書き込みパラメーター

照会再書き込みパラメーターは、ディスカバリーに関する S-TAP の動作に影響を与えます。

これらのパラメーターは、S-TAP プロパティー・ファイルの [TAP] セクションに格納されています。

注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。

GIM	guard_tap.ini	デフォルト値	記述
STAP_QRW_INSTALLED	qrw_installed	0	データベースの動的データ・マスキング機能を有効または無効にします。0 に設定すると、このグループの他のすべてのパラメーターが無視されます。 • 0=無効 • 1=有効
STAP_QRW_DEFAULT_STATE	qrw_default_state	0	照会再書き込みのアクティベーション・トリガーを設定します。firewall_default_state=1 の場合は 0 を指定する必要があります。 • 0=インストール済みポリシーのルールでトリガーされると、セッションごとに QRW がアクティブ化されます。 • 1=インストール済みポリシーに関係なく、すべてのセッションに対して QRW がアクティブ化されます。 • 2=デフォルトですべてのトラフィックに対して QRW ポリシー違反の監視が行われますが、最初の PRIORITY_COUNT パケットでイベントによって監視がトリガーされない場合、セッションの照会再書き込みはオフにされます。 2 に設定された場合、QRW 操作は Watch、Drop、Watch & Drop、および Unwatch の各コマンドで変更できます。状態 2 が有効なときに Watch コマンドを受け取った場合、この状態は 2 から 1 に変更されて、接続は永続的にファイアウォール操作または照会再書き込み操作の対象となります。Drop または Watch & Drop を受け取った場合、接続は即時に強制終了します。状態 2 が有効なときに Unwatch コマンドを受け取った場合、この状態は 2 から 0 に変更されて、接続はファイアウォール操作および照会再書き込み操作の対象から外れます。 このパラメーターを変更した後、S-TAP を再始動してください。
STAP_QRW_FORCE_WATCH	qrw_force_watch	NULL	自動的に監視するクライアント IP/マスクのコンマ区切りリスト (例えば、1.1.1.1/1.1.1.1,2.2.2.2/2.2.2.2 など)。qrw_default_state が 0 の場合に有効です。firewall_force_watch と同じ範囲には構成できません。
STAP_QRW_FORCE_UNWATCH	qrw_force_unwatch	NULL	監視から除外するクライアント IP/マスクのコンマ区切りリスト (例えば、1.1.1.1/1.1.1.1,2.2.2.2/2.2.2.2 など)。firewall_default_state が 1 の場合に有効です。firewall_force_unwatch と同じ範囲には構成できません。

親トピック: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

Linux システムおよび UNIX システム: サーバー・サイド・マスキング (SSM) パラメーター

サーバー・サイド・マスキング・パラメーターは、ディスカバリーに関する S-TAP の動作に影響を与えます。

これらのパラメーターは、S-TAP プロパティー・ファイルの [TAP] セクションに格納されています。

注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。

パラメーター	デフォルト値	記述
server_side_masking_installed	0	サーバー・サイド・マスキング機能を有効にします。 • 0=無効 • 1=有効
server_side_masking_default_state	0	サーバー・サイド・マスキングのアクティベーション・トリガーを設定します。 • 0=インストール済みポリシーのルールでトリガーされると、セッションごとに SSM がアクティブ化されます。 • 1=インストール済みポリシーに関係なく、すべてのセッションに対して SSM がアクティブ化されます。
server_side_masking_force_watch	NULL	クライアント IP/マスクのコンマ区切りリスト (例えば、1.1.1.1/1.1.1.1,2.2.2.2/2.2.2.2 など)。このリストのセッションは自動的に監視されます。server_side_masking_installed=1 かつ qrw_default_state=0 の場合に有効です。 firewall_force_watch と同じ範囲には構成できません。
server_side_masking_force_unwatch	NULL	クライアント IP/マスクのコンマ区切りリスト (例えば、1.1.1.1/1.1.1.1,2.2.2.2/2.2.2.2 など)。このリストのセッションは監視されません。server_side_masking_installed が 1 で、かつ firewall_default_state が 1 の場合に有効です。 firewall_force_unwatch と同じ範囲には構成できません。

親トピック: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

Linux システムおよび UNIX システム: discovery パラメーター

discovery パラメーターは、データベース・インスタンスのディスカバーと現在アクティブな S-TAP への結果の送信を行うオートディスカバリー機能の動作を定義します。

注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。

GIM	guard_tap.ini	デフォルト値	記述
STAP_DISCOVERY_INTERVAL	discovery_interval	24	オートディスカバリーが実行される時間間隔 (時間)。使用不可にするには 0 に設定します。
DISCOVERY_DBS	discovery_dbs	oracle:db2:informix:mysql:postgres:sybase:hadoop:teradata:netezza:memsql	ディスカバーするデータベース・タイプのコロン (:) 区切りリスト。
DISCOVERY_DEBUG	discovery_debug	0	ディスカバリー・デバッグ・レベル 0 = エラーのみ 1 = エラーとデバッグ・ステートメント
DISCOVERY_ORA_ALT_LOCATIONS	discovery_ora_alt_locations		listener.ora ファイルを検索する代替場所
STAP_DISCOVERY_PORT	discovery_port	8443	S-TAP のディスカバリーが Guardium システムへの接続に使用する Guardium ポート。

親トピック: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

Linux システムおよび UNIX システム: アプリケーション・サーバー・パラメーター

アプリケーション・ユーザー名をデータベース・アクティビティーとバインドする必要がある場合、これらのパラメーターは S-TAP の動作に影響を与えます。

これらのパラメーターは、guard_tap.ini ファイルの [TAP] セクションにあります。

GUI	GIM	guard_tap.ini	デフォルト値	記述

GUI	GIM	guard_tap.ini	デフォルト値	記述
	STAP_APPSERVER_INSTALLED	appserver_installed	0	0 がデフォルトで、S-TAP は通常どおり機能します。 1=S-TAP は「クライアント・モード」に設定され、S-TAP がデータベース・サーバーではなくクライアントにインストールされていることを反映するように S2C と C2S のパケットを切り替えます。 また、1 の場合、他の appserver_* パラメーターが入力されているかどうかを検査し、入力されている場合は、提供されているポートの http パケットを調べて、クライアント・システムに常駐する java アプリケーションのエンド・ユーザーについてのセッション情報を入手します。
ポート	STAP_APPSERVER_PORTS	appserver_ports	8080	Java アプリケーションが Web ブラウザーを介してアクセスされるポートのコンマ区切りリスト、またはそれらのポートの包含範囲を示すハイフン。
ログイン・パターン	STAP_APPSERVER_LOGIN_PATTERN	appserver_login_pattern		アプリケーションに渡されるログイン・パターンを指定するストリングのコンマ区切りリスト。 これは、Java アプリケーションに渡されるユーザー・ログインを示すパターンです。
ユーザー名接頭部	STAP_APPSERVER_USERNAME_PREFIX	appserver_username_prefix		指定のセッションのユーザー名の接頭部を指定するストリングのコンマ区切りリスト。 これは、Java アプリケーションが、指定のセッションのユーザー名を示すために使用するパターンです。
ユーザー名接尾部	STAP_APPSERVER_USERNAME_POSTFIX	appserver_username_postfix		指定のセッションのユーザー名の接尾部を指定するストリングのコンマ区切りリスト。 これは、Java アプリケーションが、ユーザー名を示す特定の変数の値の終わりを示すために使用するパターン (または文字)です。
セッション・パターン	STAP_APPSERVER_SESSION_PATTERN	appserver_session_pattern		特定のデータベース・セッションを使用するエンド・ユーザー・セッションの開始を指定するストリングのコンマ区切りリスト。
セッション接頭部	STAP_APPSERVER_SESSION_PREFIX	appserver_session_prefix		セッション ID が開始する場所を指定するストリングのコンマ区切りリスト
セッション接尾部	STAP_APPSERVER_SESSION_POSTFIX	appserver_session_postfix		セッション ID が終了する場所を指定するストリングのコンマ区切りリスト。

GUI	GIM	guard_tap.ini	デフォルト値	記述
セッション ID パターン	STAP_APPSERVER_USER SESS_PATTERN	appserver_usersess_pattern		指定の接続が継続しているエンド・セッションをマーキングするための ID を指定するストリングのコンマ区切りリスト。
セッション ID 接頭部	STAP_APPSERVER_USER SESS_PREFIX	appserver_usersess_prefix		指定の usersess 標識パケットの session_id を識別する (session_id に先行する) 対象を指定するストリングのコンマ区切りリスト。
セッション ID 接尾部	STAP_APPSERVER_USER SESS_POSTFIX	appserver_usersess_postfix		セッション ID が終了する場所を指定するストリングのコンマ区切りリスト。

親トピック: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

Linux システムおよび UNIX システム: Hadoop パラメーター

Guardium は、Apache Ranger を使用した統合 Hortonworks ディストリビューションをサポートします。S-TAP と Ranger エージェントの間の接続に必要な S-TAP パラメーターについて説明します。

Apache Ranger を使用する Hortonworks の guard_tap.ini のパラメーター

注: 一部のパラメーターは、Guardium ユーザー・インターフェースまたは Guardium Installation Manager を介して構成できます。すべてのパラメーターは、Guardium API を使用して構成できます。
注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。

表 1. Apache Ranger 統合を使用する Hortonworks の guard_tap.ini のパラメーター

パラメーター	値	記述
log4j_reader_enabled	`0` または `1` `0` は無効 (デフォルト) です。 `1` は有効です。	Ranger トラフィックに対する log4j listen モードを有効にします。
log4j_port	整数。 デフォルトは `5555` です。	Guardium S-TAP が Ranger 監査を listen するポート。
log4j_listen_address	IP アドレス `0.0.0.0` は、システムの任意の IP アドレス (デフォルト) を示します。 `localhost` は、システムのループバック・アドレスを示します。	このアドレスには、Ranger プラグインが接続します。 デフォルト値の `0.0.0.0` は、S-TAP が任意のホストからトラフィックを受信できるようになるため、お勧めです。 システムを高可用性のために構成する場合は、`localhost` を使用します。 特定のアドレスへのアクセスを制限することを選択した場合は、モニターのために必要なトラフィックを除外しないようにしてください。
log4j_num_connections	整数 デフォルト値は `20` です。	この S-TAP 用に定義されているサービスに期待される同時接続数。
ranger_dynamic_policy_port	integer デフォルト = 5556	Ranger プラグインが接続するポート。S-TAP はここで Ranger 動的ポリシーを listen します。

パラメーター	値	記述
ranger_dynamic_policy_listen_address	IP アドレス 0.0.0.0 は、システムの任意の IP アドレス (デフォルト) を示します。	このアドレスには、Ranger 動的ポリシー・プラグインが接続します。HA の場合は localhost を使用します。
ranger_dynamic_policy_num_connections	整数 デフォルト = 20	動的ポリシー・プラグインからサポートする接続の最大数。
ranger_dynamic_policy_timeout	整数 デフォルト = 10	デフォルトの判断結果を送信するまで判断を待機する秒数。
ranger_dynamic_policy_default_verdict	0 または 1 1 = 一致、0 = 不一致 デフォルト = 1	Guardium が到達不能である場合、または判断がタイムアウトになった場合の動作。
ranger_dynamic_policy_reader_enabled	0 または 1 0 は無効 (デフォルト) です。 1 は有効です。	Hortonworks 動的ポリシー・ロギングを有効にします。

Kafka メッセージングを使用する Cloudera Navigator の guard_tap.ini のパラメーター

Guardium は、Kafka メッセージング・システムを使用して監査データを収集する Cloudera Navigator をサポートします。
注: 一部のパラメーターは、Guardium ユーザー・インターフェースまたは Guardium Installation Manager を介して構成できます。すべてのパラメーターは、Guardium API を使用して構成できます。
注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。

表 2. Kafka メッセージング統合を使用する Cloudera Navigator の guard_tap.ini のパラメーター

パラメーター	値	記述
kafka_reader_enabled	`0` または `1` `0` は無効 (デフォルト) です。 `1` は有効です。	Kafka のパブリッシュとコンシュームを使用した Cloudera Navigator 統合の有効化。
kafka_bootstrap_servers	`host name:port` のペアのコンマ区切りリスト。 形式: `host:port,host:port` 例: `hostnameofbroker1:9092,hostnameofbroker2:9092`	`host name:port` のリストは、Kafka クラスターへの初期接続を確立するために使用されます。初期接続が確立されると、クラスター内のすべてのサーバーが使用されます。ダウンした場合に備えて、複数のブートストラップを指定しておくことができます。
kafka_use_tls	`0` または `1` `0` は無効 (デフォルト) です。 `1` は有効です。	Kafka クラスターが TLS を使用するかどうかを示します。
kafka_topic_name	文字列 デフォルト値は `NavigatorAuditEvents` です。	監査イベントを Kafka にパブリッシュするために Cloudera Navigator が使用するトピック名。
kafka_principal	文字列 デフォルト値は `NULL` です。	Kafka クラスターで Kerberos 認証を必要とする場合に使用される、S-TAP の Kerberos プリンシパル名。
kafka_keytab	NULL	S-TAP サーバー上の Kerberos キータブ・ファイルへのパス。

親トピック: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

Linux システムおよび UNIX システム: 構成監査システム (CAS) パラメーター

これらのパラメーターは、CAS の動作に影響を与えます。

GUI	guard_tap.ini	デフォルト値	記述
タスク・チェックポイント	cas_task_checkpoint	task_checkpoint	ホスト障害の場合の内部ハンドル・プログラム・マシンの状態。
クライアント・チェックポイント	cas_client_checkpoint	client_checkpoint	処理の再開に使用されるファイル。一連のファイルが作成されます。ファイルの各バージョンの末尾は固有の番号です。デフォルトは、task_checkpoint および client_checkpoint です。
チェックポイント期間	cas_checkpoint_period	60	チェックのための時間間隔 (秒単位)。

GUI	guard_tap.ini	デフォルト値	記述
フェイルオーバー・ファイル	cas_fail_over_file	fail_over_file	出力メッセージ・バッファーの名前。Guardium システムに到達できない場合、データベースはこのファイルに書き込みます。この期間に、ファイルは指定された最大サイズまで大きくなる可能性があります。制限に達した場合、同じ名前を使用し、名前の末尾に数字 2 を付加して、2 番目のファイルが作成されます。(これは CAS が 2 次サーバーへの接続試行を開始する時点です。) そのファイルもまた最大サイズに達した場合、最初のファイルが上書きされます。最初のファイルが再び満杯になると 2 番目のファイルが上書きされます。したがって、障害が長く続いた後にはデータをいくらか失う可能性がありますが、「フェイルオーバー・ファイルのサイズ制限」の 2 倍までの量のデータが確保されることになります。
フェイルオーバー・ファイルのサイズ制限	cas_fail_over_file_size_limit	50000	フェイルオーバー・ファイルの最大サイズ。該当するファイルが 2 つ存在するため、ディスク・スペース所要量はここで指定する値の 2 倍です。-1 を指定するとファイル・サイズは無制限になりますが、ファイル・サイズに上限を設定することを推奨します。
再接続最大試行回数	cas_max_reconnect_attempts	5000	接続が失われた場合の再接続の試行回数。Guardium システムとの接続を失った後、CAS が再接続を試みる最大回数。この値を -1 に設定すると、最大回数が除去されます (CAS はいつまでも再接続を試行します)。デフォルトの cas_max_reconnect_attempts および cas_reconnect_interval は、約 3.5 日間の間隔を定義します。最大値に達すると、CAS は フェイルオーバー・ファイルに書き込みながら実行を続けますが、Guardium ホストへの再接続は試行しなくなります。
再接続間隔	cas_reconnect_interval	60	再接続試行間の待機時間 (秒単位)。
生データ制限	cas_raw_data_limit	1000	項目テンプレートの「データを保持」チェック・ボックスを選択した場合に 1 項目に対して書き込まれる最大キロバイト数。-1 を指定すると無制限になります。
Md5 データ制限	cas_md5_size_limit	1000	MD5 チェックサム計算が実行されるデータ項目の最大サイズ (キロバイト単位)。-1 を指定すると無制限になります。
	cas_command_wait	300	長期実行データ収集プロセスを強制終了するまでの待機時間 (秒単位)。
	cas_server_failover_delay	60	別の Guardium システムへの接続を試行するまでの待機時間 (分単位)。

表 1. CAS の非推奨パラメーター

guard_tap.ini
cas_task_baseline
cas_client_baseline

親トピック: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

Linux システムおよび UNIX システム: デバッグ・パラメーター

これらのパラメーターは、S-TAP デバッグの動作に影響を与えます。

注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。
これらのパラメーターは、guard_tap.ini ファイルの [TAP] セクションにあります。

表 1. デバッグ用の S-TAP 構成パラメーター

GUI	GIM	guard_tap.ini	デフォルト値	記述
メッセージ: syslog	STAP_SYSLOG_MESSAGES	syslog_messages	1	1= メッセージを syslog に送信します。0=メッセージは送信されません。
		tap_debug_output_level	0	S-TAP ログ・レベル。ログは、tap_log_dir パラメーターで指定されたディレクトリー (デフォルトでは stderr.txt) にある stderr.txt、guard_stap.fam.txt です。各 S-TAP ログ・レベルは以下のとおりです。 • 0: 無効 • 1: 基本デバッグ • 4: 詳細デバッグ • 6: アプリケーション・サーバーのデバッグ • 10: 出口エンジンのデバッグ。S-TAP のログと db2_exit ログ (db2diag.log) の両方にデバッグ情報が記録されます。 • 11: 出口エンジンのデバッグ。db2_exit ログ (db2diag.log) のみにデバッグ情報が記録されます。
メッセージ: リモート	STAP_REMOTE_MESSAGES	remote_messages	1	アクティブな Guardium ホストにメッセージを送信します。 • 0=メッセージを送信しません。 • 1=アクティブな Guardium システムにメッセージを送信します。

親トピック: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

Linux システムおよび UNIX システム: K-TAP パラメーター

これらのパラメーターは、K-TAP の動作に影響を与えます。

これらのパラメーターは、S-TAP プロパティーの [TAP] セクションにあります。

注意:
これらは拡張パラメーターであるため、通常は IBM 技術サポートのみが変更を行います。

表 1. K-TAP 構成パラメーター

guard_tap.ini	デフォルト値	記述
ktap_installed	1	カーネル・モニター・モジュールがインストールされているかどうか。0=いいえ、1=はい。
ktap_request_timeout	5	K-TAP の応答を待機するタイムアウト時間 (秒)。K-TAP は、ioctl を S-TAP に送信して情報を求め、S-TAP からの応答を待機します。任意の値を指定できます。
ktap_dbgev_ev_list	0	GUI または guard_tap.ini ファイルによって、K-TAP トレース・ログを有効にするために使用されます。0 は、/var/tmp ディレクトリーに置かれる ktap トレース・ログを無効にし、1 は有効にします。
ktap_dbgev_func_name	すべて	K-TAP トレース・ログに記録する関数のリスト。all を指定すると、すべての関数が記録されます。accept などの特定の関数を指定すると、accept 関数のみがログ・ファイルに記録されます。 K-TAP トレース・ログとは関係のない関数を指定する場合、ログには何も記録されません。
ktap_fast_tcp_verdict	1	TCP 接続用。 0: 「低速」判定。KTAP はセッションに関する情報を STAP に送信して、トラフィックをインターセプトするかどうかを確認します。 1: 「高速」判定。KTAP が単独で判断します。 どちらの場合も、network/exclude network パラメーターが着信 IP に対して検査されます。 10.1.4 からは、この値がアップグレード後に 1 になります。
ktap_fast_file_verdict	1	TLI 接続の場合、K-TAP は ioctl を S-TAP に送信し、ポートおよび Ips を検査することによって、セッションが IE で構成されたデータベース接続であることを確認します。ktap_fast_file_verdict を 1 に設定すると、セッションのポートが範囲内にある限り、K-TAP は要求を S-TAP に送信しません。 1 または 0 の値を指定できます (1)。
ktap_buffer_size	4194304	拡張機能。K-TAP バッファーのバイト単位のサイズ。値の範囲は 1 MB から 32 MB までです。このパラメーターを変更した後、サーバーをリブートしてください。
ktap_buffer_flush	0	拡張機能。K-TAP から S-TAP にメッセージを送信する手段。1 の場合、S-TAP は K-TAP バッファー全体を読み取り、バッファー内のすべてのパケットを処理します。ktap_flush_buffer=0 の場合、S-TAP はバッファー全体ではなく、一定量を読み取ります。
ktap_local_tcp	0	1=ローカル接続のみをインターセプトします (以前にインターセプトされた接続は引き続き取り込まれます) (このパラメーターは、TCP 接続に使用されます)
khash_table_length	24593	Khash 表に格納できるセッションの数。任意の整数値を指定できます。
khash_max_entries	8192	特定のセッションのすべての情報を入れる表の長さ。任意の整数値を指定できます。
ktap_fast_shmem	1	Db2 共有メモリー接続の場合 • 0=KTAP: ioctl を STAP に送信して、プロセス ID を確認することで、セッションが IE で構成されたデータベース接続であることを確認します。 • 1=K-TAP: セッションの db2_shmem_size が、接続されている共有メモリー・セグメントと一致するまで、S-TAP に要求を送信しません。
ktap_fsmon_buffer_size	4194304	拡張機能。FAM バッファーのバイト単位のサイズ。値の範囲は 128 KB から 32 MB までです。このパラメーターを変更した後、サーバーをリブートしてください。

表 2. A-TAP および PCAP の構成パラメーター

パラメーター	デフォルト値	記述
atap_exec_location	/var/guard	検査エンジン・セクションの暗号化ボックスを有効にして A-TAP をアクティブ化する場合に使用される実行可能ファイルの場所。
atap_request_handler_enable	1	手動による構成なしに A-TAP が K-TAP にアクセスすることを許可します (IE セクションで db_user が定義されている必要があります)。 0: 無効、1: 有効
pcap_read_timeout	0	PCAP トラフィックのみ (K-TAP ではない) です。PCAP サンプリング間の S-TAP の待機時間を示します。この値を変更する場合は、変更する前に必ず技術サポートに問い合わせ、問題を検証し、PCAP/S-TAP に関連するボトルネックを原因とする損失 (取り込まれないトラフィックがある) を特定してください。
pcap_dispatch_count	16	PCAP による取り込みを最適化します。S-TAP にレポートを返す前にバンドル (グループ化) するパケットの数です。パケットをグループ化することによって、PCAP と S-TAP の間の通信を削減して、パフォーマンスを向上させることができます。この値を変更する場合は、変更する前に必ず技術サポートに問い合わせ、問題を検証し、PCAP/S-TAP に関連するボトルネックを原因とする損失 (取り込まれないトラフィックがある) を特定してください。

パラメーター	デフォルト値	記述
pcap_buffer_size	-1	PCAP ソケット・バッファーのサイズ。このパラメーターは、LINUX でのみ使用されます。この整数のデフォルト値は -1 です。これは、可能な最大バッファーを取得することを示します。その他の値は、キロバイト単位のバッファー・サイズを示します。0 は無効です。0 の場合は 60 を意味します。これ以外であれば、65535 までの任意の値を指定できます。 バッファーが大きいほど、トラフィックが急激に増大した際に、損失が発生する可能性が低くなります。高トラフィックが発生すると、PCAP はすべてを取り込みますが、S-TAP (または PCAP から S-TAP へのフロー) は速度が十分ではないため、トラフィックについていくことができません。損失を回避するために、未処理のパケットがバッファーに入れられます。 バッファーが大きいほど、トラフィックが急激に増大した場合に、より高いトラフィックや、より長時間継続するトラフィックの増大に対する回復力が高くなります。この値を変更する場合は、変更する前に必ず技術サポートに問い合わせ、問題を検証し、PCAP/S-TAP に関連するボトルネックを原因とする損失 (取り込まれないトラフィックがある) を特定してください。
pcap_backup_ktap	1	このパラメーターを有効にすると、IE で定義された Db2 がある限り、ktap_installed が有効かどうかに関係なく、常に PCAP を開始します。

GIM GUI を介してカスタム KTAP モジュール配布の使用を制御するためにパラメーターを追加

GIM ユーザー - カスタム・ビルド KTAP をカスタム・バンドルにコンパイルし、他のデータベース・サーバー上でそれを使用します。

GIM ユーザー以外 - カスタム・バンドルは必要ありません。手動でカスタム KTAP をコンパイルし、データベース・サーバー間でコピーできます。

パラメーター名: GIM_ALLOW_CUSTOMED_BUNDLES

有効な値: 「1」 - カスタム・バンドル・インストールを許可します。 「0」 - カスタム・バンドル・インストールを拒否します。

デフォルト値: 1

GIM のスクラッチ・インストール時 (DB サーバー) - ユーザーはオプションの新規インストール・パラメーター (install_custom_bundles) を指定できます。

このパラメーターを指定した場合、カスタム・バンドル・インストール (カスタム・バンドル STAP など) が、その DB サーバー上で許可されます (GIM_ALLOW_CUSTOMED_BUNDLES は「1」に設定されます)。 指定しない場合は許可されません (GIM_ALLOW_CUSTOMED_BUNDLES は「0」に設定されます)。

このパラメーターが含まれていなかった GIM バージョンから (GIM GUI を使用して) GIM をアップグレードする場合 - (その時点までこのカスタム・バンドルのフィーチャーを使用している可能性がある顧客のため、この機能を無効にしないように) デフォルトの値は「1」になります 。

DB サーバー上でコンフィギュレーター・ユーティリティーを使用する場合、このパラメーターは「1」または「0」のどちらにも設定できます。

重要:

GIM_ALLOW_CUSTOMED_BUNDLES を UI または API から変更することはできません。GIM をインストールした後、オペレーティング・システム管理者のみが値を 0 から 1 に変更できます。

注: この機能は (DB サーバー上への) インストール時にはチェックされますが、バンドル・インストールやパラメーター更新の割り当てまたはスケジューリングを行っているときには (他のすべてのパラメーターが検証されているようには) チェックされません 。

影響する機能: BUNDLE-GIM、configurator.sh、統合インストーラー

GuardAPI コマンドおよびカスタム KTAP バンドル

v10 の場合

1. STAP_UPLOAD_FEATURE インディケーターはデフォルトでオン (1) になっています。 そのため、カスタム KTAP はコンパイル時に自動的にアプライアンスにアップロードされます。
2. 新規カスタム KTAP を組み込むため、カスタム GIM バンドルをコンパイルするには、ユーザーは GrdAPI make_bundle_with_uploaded_kernel_module コマンド (正確な構文のコマンドが必要) を実行する必要があります。
3. どのサーバーでも既にコンパイル済みのカスタム・バンドルを使用できるようにするには、お客様は GIM_ALLOW_CUSTOMED_BUNDLES インディケーターをオンにして 1 にする必要があります (これはセキュリティー上の理由により、各 DB サーバー上で手動で実行する必要があります)。GIM_ALLOW_CUSTOMED_BUNDLES インディケーターをオフに戻すことは、アプライアンスから実施できます。

親トピック: Linux システムおよび UNIX システム: S-TAP 構成パラメーターの編集

Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

- Linux システムおよび UNIX システム: GIM を使用した S-TAP の停止
  GIM を使用すると、データベース・サーバーにログインせずに S-TAP を停止できます。
- Linux システムおよび UNIX システム: GIM を使用した S-TAP の始動
  GIM を使用すると、データベース・サーバーにログインせずに S-TAP を始動できます。
- Linux システムおよび UNIX システム: GIM を使用しない S-TAP の停止
  この手順は、S-TAP がスクリプトまたは RPM によってインストールされている場合に使用します。
- Linux システムおよび UNIX システム: GIM を使用しない S-TAP の再始動
  この手順は、S-TAP がスクリプトまたは RPM によってインストールされている場合に使用します。
- Linux システムおよび UNIX システム: S-TAP ログ
  UNIX S-TAP には、いくつかのログ・ファイルがあります。
- Linux システムおよび UNIX システム: さまざまな OS タイプ/バージョンによる S-TAP/GIM プロセスの初期設定の方法
- Linux システムおよび UNIX システム: S-TAP バージョンの判別

- Linux システムおよび UNIX システム: S-TAP スループットの増加
  複数の Guardium システムに報告する S-TAP を構成すると、データのスループットを増やすことができます。
- Linux システムおよび UNIX システム: GUI からの S-TAP のモニター
  次の標準レポートとビューを使用して、GUI で STAP 状況をモニターします。
- Linux システムおよび UNIX システム: S-TAP の統計
  S-TAP の統計は、事前定義された S-TAP 統計レポートで簡単に確認できます。
- Linux システムおよび UNIX システム: S-TAP モニター (guard_monitor)
  S-TAP Watchdog (guard_monitor) は、S-TAP のパフォーマンスと応答性をモニターします。S-TAP が特定のしきい値を超えたときにトリガーされる特定のアクションを構成できます。
- Linux システムおよび UNIX システム: S-TAP の問題のトラブルシューティング
  「システム・ビュー」の「S-TAP 状況モニター」タブを使用して、問題の調査を行うことができます。 他のツールを使用しなければならない場合があります。特に、検査エンジンを検査できないデータベースをモニターしている場合です。

親トピック: Linux システムおよび UNIX システム: S-TAP ユーザーズ・ガイド

Linux システムおよび UNIX システム: GIM を使用した S-TAP の停止

GIM を使用すると、データベース・サーバーにログインせずに S-TAP を停止できます。

このタスクについて

次のステップを使用して、STAP_ENABLED パラメーターを変更し、データベース・サーバー上での S-TAP の始動をスケジュールします。

手順

1. 「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」にナビゲートします。
2. 「クライアントの選択」セクションで、S-TAP を停止するデータベース・サーバーを選択します。表内のチェック・ボックスを使用して個々のクライアントを選択するか、「クライアント・グループを選択してください」メニューを使用してクライアントのグループを選択します。 「次へ」をクリックして先に進みます。
3. 「バンドルの選択」セクションで、S-TAP バンドルを選択します。「次へ」をクリックします。 ソフトウェア・バンドルを選択すると、「選択されたバンドルのアクション」列に、各クライアントに対して実行されるアクションが示されます。状況が「更新」のパラメーターがある S-TAP を停止できます。
4. 「パラメーターの選択」セクションで、`STAP_ENABLED` を入力し、値 0 を入力します。「次へ」をクリックします。
5. 「クライアントの構成」セクションで、表を使用して、加える変更を検討します。
6. 「インストール」をクリックします。
7. 「OK」をクリックして S-TAP を今すぐ停止するか、  アイコンを使用して、停止時刻をスケジュールし、「OK」をクリックします。

親トピック: Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: GIM を使用した S-TAP の始動

GIM を使用すると、データベース・サーバーにログインせずに S-TAP を始動できます。

このタスクについて

次のステップを使用して、STAP_ENABLED パラメーターを変更し、データベース・サーバー上での S-TAP の始動をスケジュールします。

手順

1. 「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」にナビゲートします。
2. 「クライアントの選択」セクションで、S-TAP を始動するデータベース・サーバーを選択します。表内のチェック・ボックスを使用して個々のクライアントを選択するか、「クライアント・グループを選択してください」メニューを使用してクライアントのグループを選択します。 「次へ」をクリックして先に進みます。
3. 「バンドルの選択」セクションで、S-TAP バンドルを選択します。「次へ」をクリックします。 ソフトウェア・バンドルを選択すると、「選択されたバンドルのアクション」列に、各クライアントに対して実行されるアクションが示されます。状況が「更新」のパラメーターがある S-TAP を開始できます。
4. 「パラメーターの選択」セクションで、`STAP_ENABLED` を入力し、値 `1` を入力します。「次へ」をクリックします。
5. 「クライアントの構成」セクションで、表を使用して、加える変更を検討します。
6. 「インストール」をクリックします。
7. 「OK」をクリックして S-TAP を今すぐ始動するか、 アイコンを使用して、開始時刻をスケジュールし、「OK」をクリックします。

親トピック: Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: GIM を使用しない S-TAP の停止

この手順は、S-TAP がスクリプトまたは RPM によってインストールされている場合に使用します。

手順

1. root アカウントを使用して、データベース・サーバー・システムにログオンします。
2. Red Hat の場合
   a. ps -fe | grep guard_stap | grep -v grep を使用して S-TAP プロセス ID を見つけます。
   b. コマンド kill を使用してそのプロセスを強制終了します。
3. Solaris の場合:

```
-bash-3.00# svcadm -v disable guard_utap
svc:/site/guard_utap:default disabled.
```

```
-bash-3.00# ps -eaf | grep stap
    root  2375  1930   0 14:25:36 pts/2       0:00 grep stap
```

4. この S-TAP® の報告先となっている Guardiumシステムから、S-TAP 制御パネルの状況ライトが赤になっていることを確認します。

親トピック: Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: GIM を使用しない S-TAP の再始動

この手順は、S-TAP がスクリプトまたは RPM によってインストールされている場合に使用します。

手順

1. root アカウントを使用して、データベース・サーバー・システムにログオンします。
2. Red Hat Enterprise Linux 以外のすべての場合
   a. /etc/inittab ファイルを編集用に開きます。
   b. 各行の先頭のコメント文字 (AIX® の場合は :、その他すべての場合は #) を削除して、以下の 2 つのステートメントをアンコメントします。

   ```
   #utap:2345:respawn:/usr/local/guardium/guard_stap/guard_stap /usr/local/guardium/guard_stap/guard_tap.ini
   ```

   c. init q コマンドを実行して、S-TAP® プロセスを再開します。
3. Red Hat Enterprise Linux の場合
   a. オペレーティング・システム・コマンド initctl list を使用して、現在実行中のエージェントをリストします。出力に、エージェントが次の例のように示されます。

   ```
   gim_33264 start/running, process 910
   gsvr_33264 start/running, process 2552
   ```

   b. 各エージェントを、start <agent> コマンドを使用して開始します。ここで、agent は ステップ a のリストにある先頭の項目です。以下の例を参照してください。

   ```
   start gim_33264
   start gsvr_33264
   start guard_utap
   ```

4. Solaris サービスを使用して再始動するには、次のようにします。

```
bash-3.00# svcadm -v enable guard_utap
svc:/site/guard_utap:default enabled.
-bash-3.00# ps -eaf | grep stap
 root  2379     1   0 14:25:57 ?           0:00
/usr/local/guardium/guard_stap/guard_stap /usr/local/guardium/guard_stap/guard_
    root  2396  1930   0 14:26:00 pts/2       0:00 grep stap
-bash-3.00# svcs guard_utap
STATE          STIME    FMRI
online         14:25:56 svc:/site/guard_utap:default
-bash-3.00#
```

5. ps -ef | grep stap を実行して、S-TAP が実行されていることを確認します。
6. この S-TAP の報告先となっている Guardiumシステムの管理者ポータルから、S-TAP 制御パネルの状況ライトが緑になっていることを確認します。

親トピック: Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: S-TAP ログ

UNIX S-TAP には、いくつかのログ・ファイルがあります。

- guard_stap* logs: tap_log_dir パラメーターで指定されたファイル・パスにあります。
  - guard_stap.stderr.txt: STAP のすべての出力 (および追加のデバッグ出力)。
  - guard_stap.fam.txt: FAM が有効である場合にのみ存在します。FAM モニターのすべての出力 (および追加のデバッグ出力) が含まれます。
  - guard_stap.stdout.txt: v10.1.4 以降、これはシステム内に存在しますが、使用されません。
- UNIX システム・ログ (/var/adm/syslog および /var/log/messages。名前と場所は特定のシステムに関連しています): K-TAP モジュールの出力メッセージ (および他のすべてのカーネル・タスクからの出力メッセージ) が含まれます。

親トピック: Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: さまざまな OS タイプ/バージョンによる S-TAP/GIM プロセスの初期設定の方法

OS	バージョン		初期化方式
AIX	6.1	PowerPC	inittab
AIX	7.1	PowerPC	inittab
AIX	7.2	PowerPC	inittab
HP-UX	11.11	pa9000	inittab
HP-UX	11.23	IA-64	inittab
HP-UX	11.23	pa9000	inittab

OS	バージョン		初期化方式
HP-UX	11.31	IA-64	inittab
HP-UX	11.31	pa9000	inittab
RHEL	4	i686	inittab
RHEL	4	IA-64	inittab
RHEL	4	x86_64	inittab
RHEL	5	i686	inittab
RHEL	5	IA-64	inittab
RHEL	5	ppc64	inittab
RHEL	5	s390x	inittab
RHEL	5	x86_64	inittab
RHEL	6	i686	upstart
RHEL	6	ppc64	upstart
RHEL	6	s390x	upstart
RHEL	6	x86_64	upstart
RHEL	7	ppc64le	systemd
RHEL	7	ppc64	systemd
RHEL	7	s390x	systemd
RHEL	7	x86_64	systemd
SUSE	11	i686	inittab
SUSE	11	ppc64	inittab
SUSE	11	s390x	inittab
SUSE	11	x86_64	inittab
SUSE	12	ppc64le	systemd
SUSE	12	s390x	systemd
SUSE	12	x86_64	systemd
Ubuntu	10.04	x86_64	inittab
Ubuntu	12.04	x86_64	upstart
Ubuntu	14.04	x86_64	upstart
Ubuntu	16.04	x86_64	systemd
Solaris	5.10	i386	サービス
Solaris	5.10	i386_64	サービス
Solaris	5.10	SPARC	サービス
Solaris	5.11	i386_64	サービス
Solaris	5.11	SPARC	サービス

Upstart サーバー

Upstart サーバーの使用時、データベース・サーバーで使用する start コマンドと stop コマンドは以下のとおりです。

S-TAP プロセスを停止する場合:

stop utap

S-TAP プロセスを開始する場合:

start utap

GIM と監視プログラムのプロセスを停止する場合:

stop gim_revision#

stop gsvr_revision#

例： stop gim_46743

GIM と監視プログラムのプロセスを開始する場合:

start gim_revision#

start gsvr_revision#

例: start gim_46743

システムの Guardium 製品の状況を確認する場合:

initctl list

status utap

Systemd サーバー

systemd サーバーの使用時、データベース・サーバーで使用するコマンドは以下のとおりです。

S-TAP プロセスを停止する場合:

systemctl stop guard_utap.service

S-TAP プロセスを開始する場合:

systemctl start guard_utap.service

GIM と監視プログラムのプロセスを停止する場合:

systemctl stop guard_gim.service

systemctl stop guard_gsvr.service

GIM と監視プログラムのプロセスを開始する場合:

systemctl start guard_gim.service

systemctl start guard_gsvr.service

システムの Guardium 製品の状況を確認する場合:

systemctl -t service -a|grep guard

Services サーバー

services サーバーの使用時、データベース・サーバーで使用するコマンドは以下のとおりです。

S-TAP プロセスを停止する場合:

svcadm -v disable guard_utap

S-TAP プロセスを開始する場合:

svcadm –v enable guard_utap

GIM と監視プログラムのプロセスを停止する場合:

svcadm –v disable guard_gim

svcadm –v disable guard_gsvr

GIM と監視プログラムのプロセスを開始する場合:

svcadm –v enable guard_gim

svcadm –v enable guard_gsvr

サーバーの Guardium 製品の状況を確認する場合:

svcs | grep guard

親トピック: Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: S-TAP バージョンの判別

手順

1. GUI の「管理」 > 「システム・ビュー」 > 「S-TAP 状況モニター」に、S-TAP® バージョン番号が表示されます。
2. 別の方法として、データベース・サーバーの UNIX コマンド行で guard_stap バイナリーに -version 引数または --version 引数を指定して実行することで、S-TAP バージョン番号を表示できます。 例えば、the S-TAP がデフォルトのインストール・ディレクトリーにインストールされているとすると、次のいずれかのコマンドを入力します。

```
-bash-3.2#  <guardium_base>/modules/STAP/current/guard_stap --version

-bash-3.2#  <guardium_base>/guard_stap/guard_stap --version
STAP-doberman_r20511_1-20100728_0514
```

親トピック: Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: S-TAP® スループットの増加

複数の Guardium システムに報告する S-TAP を構成すると、データのスループットを増やすことができます。

複数のスレッドを作成するように S-TAP を構成すると、データのスループットを増やすことができます。 S-TAP 構成ファイルに複数の Guardium システムが定義されている場合、Guardium システムごとにスレッドを作成できます。 S-TAP は、Guardium システムの数に合わせて、追加のスレッドを (v10.1.4 以上で、最大 10 スレッド) 作成します。participate_in_load_balancing パラメーターが 4 に設定されていると、K-TAP は、Guardium システムの数に合わせて、ほぼ同じの数のバッファーを 5 スレッド

まで作成します。K-TAP バッファーの数は、sqlguard メイン・スレッドの総数に依存します。K-TAP は、バッファー間を行き来して、各バッファーにパケット全体を配置します。 各 S-TAP スレッドは、異なる K-TAP バッファーから読み取りを行い、単一の Guardium システムにトラフィック・データを送信します。

この構成では、S-TAP からすべてのデータを受信する Guardium は 1 つもありません。 配布は、participate_in_load_balancing が 1 に設定されている場合に使用されるのに似ています。
重要: V10 GPU200 より前は、Guardium システムが使用不可になった場合、フェイルオーバーは行われません。Guardium システムに送信されていたデータは、システムが使用可能になるか、構成が変更されるまで失われます。
重要: V10 GPU300 より前に、S-TAP 構成ファイルに複数の Guardium システムが定義されている場合、Guardium システムごとにスレッドを作成できます。 この機能がアクティブになるのは、participate_in_load_balancing パラメーターが 4 に設定されている場合のみです。

A-TAP の暗号化されたトラフィックと暗号化されていないトラフィックを同じ Guardium システムに送信することはできません。これは、participate_in_load_balancing が 1 に設定されている場合と似ています。

親トピック: Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: GUI からの S-TAP のモニター

次の標準レポートとビューを使用して、GUI で STAP 状況をモニターします。

S-TAP によって作成された例外に基づいてアラートを作成できますが、S-TAP レポートが使用する他のドメインは、システム専用であり、ユーザーはアクセスできません。

システム・ビュー

「システム・モニター」ウィンドウの**S-TAP 状況モニター**: このレポートは、この Guardium システムへの各 S-TAP レポートについて、S-TAP ホスト、S-TAP バージョン、データベース・サーバー・タイプ、状況 (アクティブまたは非アクティブ)、最後に受信した応答 (日時)、インスタンス名、1 次ホスト名、および (KTAP、MS SQL サーバー共有メモリー、DB2® 共有メモリー、Win TCP、ローカル TCP モニター、名前付きパイプの使用、暗号化、ファイアウォール、データベース・インストール・ディレクトリー、DB ポート (最小)、および DB ポート (最大) の) true/false インジケーターを示します。

行をクリックすると、この S-TAP 用に構成されている検査エンジンが表示されます。階層リンクを参照すると、現在位置が分かります。「すべての S-TAP」をクリックして、S-TAP のリストに戻ります。詳しくは、Linux システムおよび UNIX システム: 検査エンジンの検査を参照してください。

注: Db2 共有メモリー・ドライバーは Db2 Tap フィーチャーに置き換えられました。

S-TAP 状況モニター: このレポートは、この Guardium システムへの各 S-TAP レポートについて、S-TAP ホスト、データベース・サーバー・タイプ、S-TAP バージョン、状況 (アクティブまたは非アクティブ)、検査エンジンの状況、最後に受信した応答 (日時)、1 次ホスト名、および (ファイアウォールおよび暗号化の) true/false インジケーターを示します。すべての検査エンジンの検査状況を表示するには、「S-TAP 状況」と「検査エンジンの状況」をクリックします。

S-TAP イベント: このレポートは、この Guardium システムへの各 S-TAP レポートについて、S-TAP ホスト、タイム・スタンプ、イベント・タイプ (成功、エラー・タイプなど)、およびタップ・メッセージを示します。

「S-TAP イベント」パネルにメッセージが表示されない場合は、その S-TAP® の構成ファイル内でイベント・メッセージの生成が無効になっている可能性があります。その場合は、ホスト・システム上の syslog ファイル内に S-TAP イベント・メッセージが出力されている場合があります。

TAP モニター

S-TAP 構成変更履歴: このレポートは、検査エンジンが追加または変更されたときだけ表示されます。S-TAP の構成変更がリストされます。個々の検査エンジンの変更は別々の行に表示されます。 各行には、S-TAP ホスト、データベース・サーバー・タイプ、データベース・ポート (始まり)、データベース・ポート (終わり)、データベース・クライアント IP、データベース・クライアント・マスク、および変更のタイム・スタンプがリストされます。

プライマリー Guardium® ホスト変更ログ: S-TAP の 1 次ホスト変更のログ。1 次ホストとは、 S-TAP がデータを送信する Guardium システムです。このレポートの各行には、S-TAP ホスト、Guardium ホスト名、期間の開始、期間の終了がリストされます。

S-TAP 状況: 各 S-TAP ホストで定義されている各検査エンジンについて、状況情報を表示します。 このレポートは現在の状況を報告するため、開始日付と終了日付のパラメーターはありません。このレポートの各行は、S-TAP ホスト、データベース・サーバー・タイプ、状況、最後の応答、1 次ホスト名、および属性 (K-TAP (インストール済み)、共有メモリー・ドライバー (インストール済み)、Db2 共有メモリー・ドライバー (インストール済み)、名前付きパイプ・ドライバー (インストール済み)、およびアプリケーション・サーバー (インストール済み)) の Yes/No インディケーターをリストします。さらに、ハンター DBS をリストします。

非アクティブな S-TAP: システムで定義されている非アクティブな S-TAP をすべてリストします。これには 1 つだけ、QUERY_FROM_DATE というランタイム・パラメーターがあり、デフォルトでは now -1 hour に設定されています。このパラメーターを使用して、*非アクティブ*をどのように定義するのかを制御します。このレポートには、S-TAP 状況レポートと同じデータの列が含まれており、レポートの各行のカウントが追加されています。

親トピック: Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: S-TAP の統計

S-TAP の統計は、事前定義された S-TAP 統計レポートで簡単に確認できます。

S-TAP 統計コレクションは stap_statistic パラメーターで構成されます。これは拡張パラメーターであるため Guardium 技術サポートまたは上級者のみが変更する必要があります。

アクセスするには、GUI を使用します。結果に基づいてアラートを作成できます。

表のフィールド

- TIMESTAMP
- SOFTWARE_TAP_HOST
- TOTAL_BYTES_SO_FAR
- TOTAL_BYTES_DROPPED_SO_FAR

- TOTAL_BYTES_IGNORED
- TOTAL_BUFFER_INIT
- IOCTL_REQUESTS
- TOTAL_RESPONSE_BYTES_IGNORED
- SYSTEM_CPU_PERCENT
- SYSTEM_CPU_IDLE_PERCENT
- STAP_CPU_PERCENT
- BUFFER_RECYCLED

親トピック: Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: S-TAP モニター (guard_monitor)

S-TAP Watchdog (guard_monitor) は、S-TAP のパフォーマンスと応答性をモニターします。S-TAP が特定のしきい値を超えたときにトリガーされる特定のアクションを構成できます。

注: HP-UX 11.11 では、process コマンドに関する情報は 64 文字に制限されています。 つまり、 guard_stap バイナリーへの絶対パスが 64 文字を超える場合、Guardium モニターはそのパスを認識できなくなります。

モニター対象は次のとおりです。

- CPU 使用率: *ps* コマンドまたは procfs の CPU 時間によって検査されます。
- ポーリングに対する CPU 応答性: S-TAP プロセスにコンソール要求を送信し、応答を待機することによって検査されます。

S-TAP の CPU 使用率が構成されたしきい値を超える場合、あるいは S-TAP がコンソール要求に応答しない場合、以下のアクションが実行されることがあります。

- 自動的に guard_diag を実行。
- 自動的に S-TAP プロセスを強制終了。
- 自動的にコア・ダンプが行われ、S-TAP プロセスを強制終了。
- 自動的に S-TAP プロセスをトレース。

Guard モニターは、S-TAP インストールの最後に自動的にインストールされます。ユーザー・プロンプトはなく、インストールの進行状況は表示されません。S-TAP のアンインストール時、Guard モニターは自動的にアンインストールされます。 そのため、ユーザーはインストーラー内のリブートのオプションが使用できなくなり、代わりに、アンインストールを完了するためにリブートが必要という通知を受けます。 このリブートは重大ではありませんが、システムに S-TAP を再インストールしたい場合には必要です。 アンインストールし、リブートを行わずに再インストールを試みると、インストールをブロッキングするポップアップが表示されて、S-TAP が部分的にインストールされており、サーバーにはリブートが必要ということが通知されます。

guard_monitor は、その構成ファイル (guard_monitor.ini) を引数として稼働します。 このモニターは、guard_monitor.ini ファイルを使用して制御されます。 シェル・インストールの場合、構成に関するすべての変更は構成ファイルで直接行うことができます。 GIM の場合、GUI のインターフェースを使用して変更を行います。

Guard_monitor はデフォルトでは有効になっていません。シェル・インストールでは、「umon」行をアンコメントするか、特定のオペレーティング・システムのサービス制御機能 (RedHat 6 の場合は initctl、RedHat 7 の場合は systemctl、Solaris 10 以上の場合は SMF) を使用することで、inittab から有効にします。GIM インストールの場合、`STAP-UTILS_START_MONITOR=y` と設定することで guard_monitor を有効にします。

注: guard_monitor は管理特権 (root) を必要とします。

S-TAP モニター出力のデフォルトの場所は /var/tmp/monitor です。この場所は、guard_monitor.ini (構成ファイル) で構成可能です。このトピックの最後にある guard_monitor.ini ファイルの例を参照してください。

guard_monitor を有効にしたら、そのプロセスがデータベース・サーバーで稼働中であることを確認してください。

設定例

各関数には、デフォルトのしきい値があります。 例えば、CPU 使用率をモニターするときに、診断情報の収集のために 1 つしきい値 (75%) を設定し、S-TAP を強制終了する高いしきい値 (85%) を設定することが考えられます。診断情報の収集を有効にするには auto_diag=1 を設定し、CPU 使用率が 75% に達した場合に診断情報を収集するには diag_high_cpu_level=7500 を設定します。 次いで、S-TAP プロセスの自動強制終了を有効にするために auto_kill_on_cpu_enable=1 を設定し、CPU 使用率が 85% に達した場合にプロセスを強制終了するために auto_kill_on_cpu_level=8500 を設定します。

S-TAP プロセスを繰り返し強制終了することを避けるために、制限を設けることもできます。kill_num_in_hour=5 を設定することによって、1 時間のうちにプロセスを強制終了する回数を制限できます。 次いで、制限に達した場合の処置を指定します。S-TAP を無効にするには final_action=1 をコーディングし、実行を継続するには final_action=2 をコーディングします。

Guard_monitor CPU ポーリング・パラメーター

guard_monitor.ini	GIM	記述	デフォルト
poll_cpu_interval	STAP-UTILS_MONITOR_POLL_CPU_INTERVAL	guard_monitor が S-TAP CPU 使用率を検査する間隔 (秒単位)。 guard_monitor は CPU 使用率をチェックする際、*ps* を使用して guard_stap プロセスの存続期間中の平均の CPU 使用率を測定します。つまり、S-TAP は guard_monitor が問題を検出するまで、しばらくの間 CPU しきい値を超えて実行されることになります。	10
cpu_measurement_timeslice		CPU 消費を測定する間隔 (秒単位)。0 に設定すると、消費がプロセスの存続時間全体にわたって測定されます。	5
poll_stap_interval	STAP-UTILS_MONITOR_POLL_STAP_INTERVAL	guard_monitor がコンソールの S-TAP 要求を送信する間隔 (秒単位)。	10

guard_monitor.ini	GIM	記述	デフォルト
cpu_measurement_mode	NA	CPU 消費を計算する方法: 0: 1 つのコアに対する CPU 消費を測定 1: Guardium システムの合計 CPU キャパシティーから CPU 消費を測定します。	0
nonresponsive_action	NULL	S-TAP がポーリングに応答しないときに実行されるアクション。 • diags • trace • NULL	

Auto-Diag アクション

S-TAP の CPU 使用率が構成されたしきい値を超える場合、guard_monitor により実施される最も基本的なアクションは自動 guard_diag です。

デフォルトでは、guard_diag の出力は /var/tmp に配置されます。ファイル名はマシン名と実行された時刻/日付から派生し、常に diag.ustap で始まります。

guard_monitor.ini	GIM	記述	デフォルト
auto_diag	STAP-UTILS_MONITOR_AUTO_DIAG	自動 guard_diag を有効にします。0=いいえ、1=はい。	1
diag_high_cpu_level	STAP-UTILS_MONITOR_DIAG_HIGH_CPU_LEVEL	guard_monitor が guard_diag を開始する S-TAP CPU しきい値。(%CPU しきい値 * 100) を入力します。v10.1.4 以上では、cpu_measurement_mode=1 のときは、% が 100 より高くなる可能性があります。	7500
diag_num	STAP-UTILS_MONITOR_DIAG_NUM	複数の guard_diag 出力の作成を有効にします。整数。	2

Auto-Kill アクション

S-TAP の自動的な強制終了を構成するには、次のパラメーターを使用します。

guard_monitor.ini	GIM	記述	デフォルト
auto_kill_on_cpu_enable	STAP-UTILS_MONITOR_AUTO_KILL_ON_CPU_ENABLE	S-TAP の自動的な強制終了を有効にします。0=いいえ、1=はい。	
auto_kill_on_cpu_level	STAP-UTILS_MONITOR_AUTO_KILL_ON_CPU_LEVEL	guard_monitor が S-TAP を強制終了する S-TAP CPU しきい値。(%CPU しきい値 * 100) を入力します。v10.1.4 以上では、cpu_measurement_mode=1 のときは、% が 100 より高くなる可能性があります。	8500
kill_num_in_hour	STAP-UTILS_MONITOR_KILL_NUM_IN_HOUR	guard_monitor を強制終了する 1 時間あたりの最大回数。整数値。	5
final_action	STAP-UTILS_MONITOR_FINAL_ACTION	1 時間あたりの最大強制終了数に到達したときに実行するアクション。 • 1 = S-TAP を無効にします。 • 2 = S-TAP の強制終了を停止して、続行させます。	

強制終了前の S-TAP のコア・ダンプ

S-TAP がループに陥っている場合など、S-TAP の問題によっては guard_diag 出力に示されている内容より詳しい情報が必要になります。

guard_monitor は、S-TAP プロセスの自動コア・ダンプを実行します。guard_monitor は、プロセスを強制終了する前に S-TAP のコア・ダンプを行います (S-TAP の自動強制終了が有効な場合)。

guard_monitor で作成されたコア・ダンプは /var/tmp/monitor/coredumps にあります。

次のパラメーターは、自動コア・ダンプを構成します。

guard_monitor.ini	記述	デフォルト

guard_monitor.ini	記述	デフォルト
force_core_before_kill	生成するコア・ダンプのタイプ: sigsegv: これは、最も便利なオプションですが、SA はコア・ダンプを有効にするために ulimit を構成する必要があります。 gcore: 最も便利なオプションですが、システムに gcore がインストールされている必要があります。Linux プラットフォームのみ。 pstack: 一番便利でないオプションですが、一部のシステムでは使用できる唯一のユーティリティーです。Linux プラットフォームのみ。 NULL: 無効。	
force_core_when	コア・ダンプを収集するタイミング: limitsexceeded: リソース制限を超えたために S-TAP が強制終了されたときにコアを収集 nonresponsive: 応答しなくなったために S-TAP が強制終了されたときにコアを収集 always: 常にコアを収集	always
kill_oldcore_saved	整数。生成されるコア・ダンプを保存するかどうかを指定します。ゼロ以外に設定すると、guard_diag は、生成されたすべてのコア・ダンプを維持します。そうではない場合、新規コア・ダンプが生成されるたびに、古いコア・ダンプを削除します。	

guard_monitor.ini の例

```
The following section header is required for GIM to recognize this .ini file.
; それ以外の目的はありません

[TAP]
;
モニター・ログ、診断、トレースなどの出力ディレクトリー。
monitor_output_dir=/var/tmp
;
guardium インストール済み環境の場所 (モニターのインストール場所である必要はありません。例： /usr/local)
stap_dir=/usr/local
; 構成ファイルをダウンロードしたり、診断やトレースの出力をアップロードするために接続する IP
; これは guard_tap.ini から解析されますが、ここのバックアップ値は同期が維持されます。
sqlguard_ip=NULL
;
サーバー・エンドが引き続き有効であることを確認するためのポーリング間隔 (秒単位)
poll_server_interval=20
;
CPU レベルをチェックするためのポーリング間隔                    (秒単位)
poll_cpu_interval=10
;
STAP との通信のためのポーリング間隔                    (秒単位)
poll_stap_interval=10
;
モニター・ログ・ファイルの最大ファイル・サイズ                    (KB)
monitor_log_rotate_size=1024
;
保持する循環モニター・ログの数
monitor_log_rotate_num_kept=5
;
ログ・ファイルの最大ファイル・サイズ                    (KB)
log_rotate_size=4096
;
保持する循環ログの数
log_rotate_num_kept=5
;
循環するログ
logs_to_rotate=/tmp/guard_stap.stderr.txt,/tmp/guard_stap.stdout.txt,/usr/local/guardium/guard_stap/ktap/ktap_install.log,/usr/local/guardium/guard_stap/guard_discovery.stderr.log
;
1 時間あたりの STAP 強制終了の最大数 (auto_kill_on_intercept により行われた強制終了はカウントしない)
kill_num_in_hour=5
; 1 時間あたりの強制終了回数が上限に達した場合に STAP を無効にするか、強制終了を無効にして STAP を続行する
;
STAP を無効にする: 1; 強制終了を無効にする: 2
final_action=2
; CPU レベルで STAP を自動的に強制終了する        オン/オフ (1/0)
auto_kill_on_cpu_enable=0
;
強制終了のための CPU レベル        (% * 100)
auto_kill_on_cpu_level=8500
;
強制終了のための snif タイムアウト      (秒単位、0 は無効)
auto_kill_on_snif_timeout=0
;
強制終了のための KTAP タイムアウト      (秒単位、0 は無効)
auto_kill_on_ktap_timeout=0
;
強制終了のための PCAP タイムアウト      (秒単位、0 は無効)
```

```
auto_kill_on_pcap_timeout=0
;
強制終了のための TEE タイムアウト      (秒単位、0 は無効)
auto_kill_on_tee_timeout=0
;
強制終了のための SHMEM タイムアウト      (秒単位、0 は無効)
auto_kill_on_shmem_timeout=0
;
自動診断                    オン/オフ (1/0)
auto_diag=1
;
診断の実行回数
diag_num=2
; 診断実行間の時間          (分)
diag_interval=2
; 古い診断ファイルを保持するかどうか      はい/いいえ (1/0)
diag_oldrun_saved=0
; 診断後に STAP を強制終了する  はい/いいえ (1/0)
diag_auto_kill=0
;
診断トリガーのための CPU レベル         (% * 100)
diag_high_cpu_level=7500
;
診断トリガーのための snif タイムアウト      (秒単位、0 は無効)
diag_snif_timeout=0
;
診断トリガーのための KTAP タイムアウト      (秒単位、0 は無効)
diag_ktap_timeout=0
;
診断トリガーのための PCAP タイムアウト      (秒単位、0 は無効)
diag_pcap_timeout=0
;
診断トリガーのための TEE タイムアウト      (秒単位、0 は無効)
diag_tee_timeout=0
;
診断トリガーのための SHMEM タイムアウト      (秒単位、0 は無効)
diag_shmem_timeout=0
;
自動トレース                              オン/オフ (1/0)
auto_trace=0
;
トレース実行の最大時間                        (秒)
trace_max_time=30
;
トレースの最大ログ・ファイル・サイズ               (MB)
trace_max_log_size=10
;
古いトレース・ログ・ファイルの保持                   はい/いいえ (1/0)
trace_oldlog_saved=0
;
トレースが実行完了をしたときに STAP を強制終了する はい/いいえ (1/0)
; (例えば、低 CPU によりキャンセルされない場合)
trace_kill_on_complete=0
;
トレースをトリガーするための CPU レベル         (% * 100)
trace_high_cpu_level=6000
;
トレースをキャンセルするための低 CPU レベル            (% * 100)
trace_low_cpu_level=3500
;
snif 通信トリガーのタイムアウト  (秒単位、0 は無効)
trace_snif_timeout=0
;
KTAP 通信トリガーのタイムアウト  (秒単位、0 は無効)
trace_ktap_timeout=0
;
トレースをトリガーするための PCAP タイムアウト      (秒単位、0 は無効)
trace_pcap_timeout=0
;
トレースをトリガーするための TEE タイムアウト      (秒単位、0 は無効)
trace_tee_timeout=0
;
トレースをトリガーするための SHMEM タイムアウト      (秒単位、0 は無効)
trace_shmem_timeout=0
;
構成されたデータベースをインターセプトしていない場合に STAP を自動的に強制終了する
       はい/いいえ(1/0)
; guard_tap.ini が STAP がルートとして実行されていることを示す場合にも機能は無効になります。
auto_kill_on_intercept=0
; STAP の要求された強制終了間の最小時間          (分)
intercept_min_time_interval=15
;
1 時間あたりのインターセプトの強制終了の最大数
intercept_max_num_in_hour=0
; number of seconds across which CPU consumption is measured    (secs, 0 disabled)
; when disabled, CPU consumption is measured across the life of the process
cpu_measurement_timeslice=0
; method for calculating CPU consumption (0 or 1)
; 0: measure CPU consumption relative to one core
```

```
; 1: measure CPU consumption taking number of cores into account
cpu_measurement_mode=0
; when to collect a core dump (always, limitsexceeded, nonresponsive)
; limitsexceeded: collect core when STAP is killed due to exceeding a resource limit
; nonresponsive: collect core when STAP is killed due to it being nonresponsive
; always: always collect core
force_core_when=always

; STAP nonresponsive action
; run diags before killing STAP    : diags
; collect trace before killing STAP: trace
; no collection, just kill STAP    : NULL
nonresponsive_action=diags
```

親トピック: Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Linux システムおよび UNIX システム: S-TAP の問題のトラブルシューティング

「システム・ビュー」の「S-TAP 状況モニター」タブを使用して、問題の調査を行うことができます。 他のツールを使用しなければならない場合があります。特に、検査エンジンを検査できないデータベースをモニターしている場合です。

S-TAP が Guardium システムに接続されていない場合
: IBM Security Guardium S-TAP サービスがデータベース・サーバーで実行されているかどうかを確認します。
データベース・サーバー上で、コマンド行でコマンド ps -ef | grep stap を実行して、S-TAP® プロセスが実行されていることを確認します。プロセス・リストの中で、/guardium/guard_stap を探します。

S-TAP のバージョンを調べる方法

- GUI の「管理」 > 「システム・ビュー」 > 「S-TAP 状況モニター」に、S-TAP バージョン番号が表示されます。
- 別の方法として、データベース・サーバーのコマンド行で、S-TAP バージョン番号を表示できます。

コマンド行からデバッグを実行して、構成の問題を迅速に特定
: 構文 <stap_program> <parameter_file> <debug_level> を使用します。ここで、通常デバッグのレベルは 4 です。(他の値では実行内容が異なり、デバッグではないこともあります)。例: /usr/local/guardium/guard_stap/guard_stap /usr/local/guardium/guard_stap/guard_tap.ini 4

データベース・サーバーと Guardium システムの間の接続を確認します。

- データベース・サーバーから *sqlguard_ip* で Guardium システムを ping できることを確認します。
- ping が成功した場合は、Guardium システム上のポート 16016/16018 に Telnet でログインできることを確認します。

データベース・サーバーと Guardium システムの間にファイアウォールがある場合
: これら 2 つのシステムの間でのトラフィック用に、TCP ポート 16016 または TLS ポート 16018 (暗号化接続の場合) が開かれていることを確認します。
注: ポートが使用可能かどうかを確認するには、コマンド `nmap -p port guardium_hostname_or_ip` を使用します。

sqlguard_ip パラメーターが、接続先の Guardium システムの正しい *guardium_hostname_or_ip* に設定されていることを確認します。

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」をクリックして、「S-TAP 制御」を開きます。
2. データベース・サーバーに対応する IP アドレスの S-TAP ホストを見つけます。
3. 「Guardium ホスト」サブセクションを展開して、アクティブな Guardium ホストが正しく構成されていることを確認します。
4. 必要に応じて、「変更」をクリックして、Guardium ホストを更新します。

S-TAP プロセスが繰り返し再始動していないことを確認します。
: データベース・サーバー上で、コマンド `ps -eaf | grep stap` を実行して、S-TAP のプロセスが変更されていないことを確認します。

S-TAP 承認がオンになっていないことを確認します。
: S-TAP 承認がオンになっていると、Guardium システムに接続されている新規 S-TAP は、すべて拒否されます。

1. 「管理」 > 「アクティビティー・モニター」 > 「S-TAP 認証」をクリックして、「S-TAP 認証」を開きます。
2. 「S-TAP 承認が必要」チェック・ボックスを調べます。このボックスにチェック・マークが付いている場合、新規 S-TAP がこの Guardium システムに接続できるのは、承認済み S-TAP のリストに追加されてからになります。
3. S-TAP 承認がオンになっている場合は、「日次モニター」 > 「承認された Tap クライアント」を選択して、承認済み S-TAP のリストを表示します。 調査対象の S-TAP がこのリストにない場合は、「S-TAP 認証」ペインに戻り、「クライアント・ホスト」フィールドに S-TAP の IP アドレスを入力して、「追加」をクリックします。

S-TAP に緑の状況が表示されているがデータが処理されていない場合
: A-TAP の状況を確認してください。

Db2 共有メモリー・トラフィックがキャプチャーされない
: IE 構成が正しいことを確認してください。stap_directory/bin の下にあるスクリプト `find_db2_shmem_parameters.sh` を実行します。Db2 インスタンス名をパラメーターとして使用し、root ユーザーまたは Db2 ユーザーとして実行してください。Db2 共有メモリー・サイズ、クライアント位置、およびヘッダー・サイズなど、共有メモリー・パラメーターが返されます。S-TAP で定義されている IE パラメーターが、返された値と一致していることを確認してください。

S-TAP 検査の問題

検査プロセスは、間違ったユーザー ID とパスワードを使用してデータベースの STAP クライアントへのログインを試行することで、この試行が認識され、Guardium システムに通知されることを確認します。要求が行われた Guardium システムに検査エンジンのメッセージが到達しないように、S-TAP が構成されている可能性があります。

このような構成の詳細には、以下が含まれます。

- ロード・バランシング: 複数の Guardium システムに応答を返すように S-TAP が構成されている場合は、エラー・メッセージをさまざまな Guardium システムに送信できます。
- フェイルオーバー: 2 次 Guardium システムが S-TAP 用に構成されていると、1 次 Guardium システムがビジー状態である場合に、エラー・メッセージを 2 次 Guardium システムに送信できます。
- Db_ignore_response: データベースからのすべての応答を無視するように S-TAP が構成されている場合、エラー・メッセージは Guardium システムに送信されません。
- クライアント IP/マスク: 0.0.0.0 ではないマスクを定義する場合、エラー・メッセージが送信されない可能性があります。

- 除外 IP/マスク: 0.0.0.0 ではないマスクを定義する場合、エラー・メッセージが送信されない可能性があります。

関連トピック:

- Linux システムおよび UNIX システム: GUI からの S-TAP のモニター
- Linux システムおよび UNIX システム: S-TAP モニター (guard_monitor)
- Linux システムおよび UNIX システム: 検査エンジンの検査

親トピック: Linux システムおよび UNIX システム: S-TAP の操作とパフォーマンス

Db2 for IBM i S-TAP

Guardium Db2 for i S-TAP を使用して、IBM i 上のあらゆるデータベース・アクセスをモニターおよびレポートすることができます。これには、ネイティブのデータベース入出力操作または SQL アクセスを使用するあらゆるプログラム (RPG など) が含まれます。

Guardium Db2 for i S-TAP によって収集された情報を使用して、アクティビティー・レポートを作成したり、監査要件を満たしたり、無許可アクティビティーに関するアラートを生成したりすることができます。詳細な監査情報には、以下の内容が含まれます。

- セッションの開始時刻と終了時刻
- TCP/IP アドレスおよびポート
- オブジェクト名 (例えば、表またはビュー)
- ユーザー
- SQLSTATE
- ジョブおよびジョブ番号
- SQL ステートメントおよび変数
- クライアントの特殊レジスター値
- インターフェース情報 (ODBC、ToolboxJDBC、ネイティブ JDBC、.NET など)

S-TAP は、以下の 2 つのソースからデータを受け取ります。

- SQL アプリケーション用の SQL パフォーマンス・モニター (別名はデータベース・モニター) データ
- SQL 以外のインターフェースを使用したアプリケーション用の QSYS/QAUDJRN 監査ジャーナルからの監査項目

これらのソースからのデータには、以下が含まれます。

- あらゆる SQL アクセス。これには、IBM i サーバーで開始されたアクセスも、クライアントから開始されたアクセスも含まれます。
- 監査ジャーナルで取得されたネイティブ・アクセス。

S-TAP は、このデータを Guardium システムにリアルタイムで送信します。
Db2 for i S-TAP および関連するトピックについて詳しくは、以下の情報源を参照してください。

- Using IBM Security Guardium for monitoring and auditing IBM DB2 for i database activity: この developerWorks の記事は、IBM Guardium、Db2 for i S-TAP、および関連する主要な詳細情報を紹介しています。
- IBM i の IBM Knowledge Center: IBM i、監査ジャーナル、およびその他の関連トピックについての情報は、この Web サイトを参照してください。

暗号化、ロード・バランシング、フェイルオーバー用の i S-TAP

IBM i S-TAP では、TLS 暗号化、S-TAP セッションのロード・バランシング、S-TAP セッションのフェイルオーバーがサポートされています。

注: i S-TAP の TLS とロード・バランシングは、IBM i 7.1 と 7.2 でのみサポートされます。

UNIX S-TAP の場合と同様に、i S-TAP の構成パラメーターは、IBM i サーバー上の /usr/local/guardium ディレクトリー内の guard_tap.ini ファイルに保存されます。

管理者は、他の UNIX S-TAP と同じ API と UI (S-TAP 制御) を使用して S-TAP を構成します。GUI または API を使用して S-TAP の構成を変更すると、Guardium のスニファーがメッセージを S-TAP に送信します。メッセージを受信した S-TAP は、古い .ini ファイルのバックアップを作成し、変更された構成情報を新しい .ini ファイルに保存してから、自分自身を再起動します。

管理者は、S-TAP 構成制御を使用して、S-TAP とアプライアンス間の暗号化通信を設定することができます。また、各種のロード・バランシング・オプションを設定することもできます。

S-TAP のフェイルオーバー機能とロード・バランシング機能の使用

i S-TAP のフェイルオーバー・オプションとロード・バランシング・オプションは、UNIX S-TAP のオプションと似ています。participate_in_load_balancing パラメーターを使用して、フェイルオーバー機能またはロード・バランシング機能を使用するかどうかを指定し、S-TAP の SQLGuard セクションを使用して、1 次 Guardium ホスト、2 次 Guardium ホスト、3 次 Guardium ホストを設定します。

i S-TAP と UNIX S-TAP との違いは、i S-TAP の場合は participate_in_load_balancing=3 が必要ないということです。これは、各メッセージについて完全なセッション情報を使用できるように I S-TAP の通信方法が構築されているためです。そのため、このパッチに含まれている拡張機能を適用しなくても、構成ファイルの 1 次 SQLGuard セクションに記述されている participate_in_load_balancing=1 パラメーターと仮想 IP アドレスを使用すれば、F5 などのハードウェア・バランシング機能を使用できるようになっています。

フェイルオーバー構成では、複数のコレクターで登録するように S-TAP が構成されますが、1 回に 1 つのコレクターに対してのみ、トラフィックが送信されます (participate_in_load_balancing=0)。このように構成されている S-TAP は、送信先となる 1 つのコレクターで接続に関する問題が発生しない限り、すべてのトラフィックをそのコレクターに送信します。このコレクターで接続に関する問題が発生すると、2 次コレクターに対してフェイルオーバーがトリガーされます。

IMS からの AppEvent の使用方法

APP_EVENT DLI 呼び出しのユーザー情報を保持するデータは、GuardAppEvent API に類似した構文を使用する必要があります。

先頭の 2 バイトは、それに続くバイトのエンコード方式の CCSID を示します。 例えば、0x04B8 は CCSID 1208 を意味します。 これに続くバイトは、以下のような構文を使用する必要があります。

SELECT

'GuardAppEvent:Start',

'GuardAppEventType:type',

'GuardAppEventUserName:name',

'GuardAppEventStrValue:string',

'GuardAppEventNumValue:number',

'GuardAppEventDateValue:date'

FROM DUAL

type (タイプ)、name (名前)、string (文字列)、number (数値)、date (日付) についての詳細は、「GuardAppEvent API」をご確認ください。

現在サポートされているのは、UTF8 エンコード方式のみです。

- モニター戦略
  法規制およびその他の要件を認識してそれらを満たすための戦略を作成し、モニターおよび監査を効果的かつ効率的に実行できるようにします。
- IBM i 用の S-TAP のインストール
  以下の手順に従って、S-TAP のインストールまたはアンインストールを行います。
- IBM i 用の S-TAP の定義
  S-TAP をインストールした後、S-TAP が Guardium システムと通信できるようにします。

モニター戦略

法規制およびその他の要件を認識してそれらを満たすための戦略を作成し、モニターおよび監査を効果的かつ効率的に実行できるようにします。

どのようなデータが必要か分かったら、無関係のデータをできる限り省いて、必要なデータを収集するための戦略を作成します。不要なデータのモニターおよびロギングにより、ディスク・スペースと処理能力が消費され、余計なネットワーク・トラフィックが発生します。このような戦略をさまざまな領域で実装可能です。

データベースのモニター
グローバル SQL モニターは、SQL 情報をキャプチャーし、その情報を S-TAP のキューに入れます。 モニターのフィルター機能を使用して、キューに入れるユーザーとオブジェクトのタイプを制御することができます。 デフォルトでは、以下のタイプの項目は、S-TAP から Guardium システムに転送されません。

SQL 省略語	意味
AD	ALLOCATE DESCRIPTOR
CL	CLOSE
DA	DEALLOCATE DESCRIPTOR
DE	DESCRIBE
EX	EXECUTE (実行された SQL ステートメントは監査されます)
FE	FETCH
FL	FREE LOCATOR
GD	GET DIAGNOSTICS
GS	GET DESCRIPTOR
HL	HOLD LOCATOR
PR	PREPARE (許可エラーのキャプチャーは除く)
RE	RELEASE
RG	RESIGNAL
SC	SET CONNECTION
SD	SET DESCRIPTOR
SG	SIGNAL

監査ジャーナル
対象となるオブジェクトまたは対象となるユーザーに関係する項目のみキャプチャーするようにシステム監査ジャーナルを構成できます。デフォルトでは、これらのタイプの項目は、S-TAP から Guardium システムに送信されます。

SQL 省略語	意味
ZR	オブジェクトの読み取り
ZC	オブジェクトの変更
CA	権限変更
AD	監査の変更
AF	権限の障害
CO	オブジェクトの作成
DO	オブジェクトの削除

SV	システム値の変更
GR	汎用監査レコード
OM	オブジェクトの移動と名前変更
PG	1 次グループの変更
PW	パスワードまたはユーザー ID が無効
OW	所有者の変更
OR	オブジェクトの復元
RA	権限の変更を復元
RO	所有者の変更を復元
RZ	1 次グループの変更を復元

データベース・オブジェクトに関係する以下の項目のみ転送されます。

- *FILE (表、ビュー、索引、論理ファイル、別名、またはデバイス・ファイル)
- *SQLUDT (SQL ユーザー定義タイプ)
- *SQLPKG (SQL パッケージ)
- *PGM (プロシージャー、関数、またはプログラム)
- *SRVPGM (プロシージャー、関数、グローバル変数、またはサービス・プログラム)
- *DTAARA (SQL シーケンス)

Guardium システム上で
S-TAP から受信した情報の中でどの情報を無視するか、および他の項目に基づいてどのアクションを実行するかを制御するポリシーを定義できます。

ネットワーク上に送信された後にデータを無視することは非効率的です。可能な場合は必ず、S-TAP 用のキューに入れられる前に不要な情報をフィルターで除去してください。

親トピック: Db2 for IBM i S-TAP

IBM i 用の S-TAP のインストール

以下の手順に従って、S-TAP のインストールまたはアンインストールを行います。

始める前に

Db2 for i S-TAP には、Portable Application Solutions Environment (PASE) が必要です。これは、ユーザーが IBM Guardium ユーザー・インターフェースから Db2 for i S-TAP を開始および停止したときに、必要に応じて自動的に開始および停止される環境です。

この S-TAP が接続する Guardium システムの IP アドレスを知っている必要があります。

S-TAP をダウンロードするときには、正しいパッケージがダウンロードされるように、必ず IBM i プラットフォームでフィルターに掛けるようにしてください。

このタスクについて

IBM i では、Guardium Installation Manager (GIM) はサポートされません。

5250 エミュレーター・ソフトウェアを使用して、IBM i システムにリモート接続することができます。

手順

1. IBM i サーバー上で、call qp2term コマンドを入力して、PASE のシェルを開きます。
2. PASE のシェル環境で、S-TAP のインストール・スクリプトを格納するための一時ディレクトリー (/tmp など) を作成します。
3. FTP を使用して、S-TAP のインストール・シェル・スクリプト guard-itap-9.0.0_*rnnnnn*-aix-5.3-aix-powerpc.sh を、作成した一時ディレクトリーに移します。
4. 同じディレクトリーで、以下のコマンドを実行します。

   ```
   guard-itap-9.0.0_rnnnnn-aix-5.3-aix-powerpc.sh guardium_host_IP
   ```

   ここで、*guardium_host_IP* は Guardium システムの IP アドレスです。

タスクの結果

S-TAP が /usr/local/guardium にインストールされます。インストールが完了すると、S-TAP はアクティビティー・モニターを有効にするプロセスを開始し、インストール・コマンドで指定された IP アドレスを使用して、Guardium システムに接続しようとします。

次のタスク

正常なインストールと監査プロセスの開始を確認するには、IBM Guardium Web コンソールに管理者としてログインし、「システム・ビュー」タブにナビゲートして、S-TAP の状況を確認します。

親トピック: Db2 for IBM i S-TAP

S-TAP のアンインストール

手順

S-TAP を停止し、アンインストールするには、以下のコマンドを実行します。

```
RUNSQL SQL('call SYSPROC/SYSAUDIT_End') COMMIT(*NONE)
RMVDIR DIR('/usr/local/guardium') SUBTREE(*ALL)
```

IBM i 用の S-TAP の定義

S-TAP をインストールした後、S-TAP が Guardium システムと通信できるようにします。

始める前に

IBM i システムのログイン資格情報を知っている必要があります。

このタスクについて

S-TAP を構成するための手順の概要を以下に示します。

1. Db2 for i を IBM Guardium に対する認識されたデータ・ソースとして定義し、接続をテストします。
2. Db2 for i S-TAP のインストール時に作成された IBM i 上の構成ファイルの情報を、カスタム表ビルダーのプロセスを使用して Guardium システムに取り込みます。
3. Db2 for i の構成レポートを作成します。このレポート・インターフェースから、モニター・プロセスの開始と停止、状況情報の取得、フィルタリング値を含む構成パラメーターの更新を可能にする Guardium API を呼び出すことができます。

手順

1. 「設定」 > 「ツールとビュー」 > 「データ・ソース定義」をクリックして、「データ・ソース・ビルダー」を開きます。「アプリケーション選択」ボックスから「カスタム・ドメイン」を選択します。「次へ」をクリックします。
2. 「データ・ソース・ファインダー」で、「新規」をクリックします。これにより、「データ・ソース・ビルダー」が開きます。
3. 「データベース・タイプ」として Db2 for i を選択し、「ホスト」、「サービス名」、「資格情報」に適切な情報を追加します。「適用」をクリックします。
4. 「接続のテスト」をクリックして、構成が正常に行われたことを確認します。
5. 「ツール」 > 「レポートのビルド」をクリックします。
6. 「カスタム表ビルダー」をクリックします。「Db2 for i S-TAP 構成」を選択し、「データのアップロード」をクリックします。「データ・ソース・ファインダー」に、Db2 for i S-TAP のリストが表示されます。
7. 構成した Db2 for i データ・ソースをこのリストから選択し、「追加」をクリックします。
8. 「データのインポート」画面に、Db2 for i データ・ソースが表示されていることを確認します。「適用」をクリックして、「今すぐ 1 回実行」をクリックします。操作が正常に終了し、行が 1 つ挿入されたことを示すメッセージが表示されます。
9. Guardium のタイトル・バーにある「カスタマイズ」をクリックします。次に、「ペインの追加」をクリックします。
10. ペインに My New Reports などの新しい名前を指定して、「適用」をクリックします。
11. 「カスタマイズ」ペインに「My New Reports」が表示されます。名前の横にあるアイコンをクリックします。「レイアウト」のドロップダウン・リストで、「メニュー・ペイン」を選択します。「保存」をクリックします。作成した新しいペインがタブの形で表示されます。
12. ナビゲーションペインで、「レポートのビルド」 をクリックします。
13. 照会のドロップダウン・リストから、「Db2 for i S-TAP 構成」をクリックし、次に「検索」をクリックします。
14. 「Db2 for i S-TAP 構成」を選択し、「My New Reports に追加」 (あるいはステップ 10 でペインに指定した名前) をクリックします。
15. 「My New Reports」タブを開きます。この時点でこのタブには、IBM i のレポート行が表示されています。レポート内の行を 1 つダブルクリックし、「呼び出し」を選択します。 選択できる IBM Guardium API のリストが表示されます。
16. update_istap_config を選択します。
17. Guardium API を選択すると、その API の各パラメーターが表示されます。変更が必要な任意の値を変更することができます。start_monitor パラメーターの値を 1 に変更します。「今すぐ呼び出し」 をクリックします。

タスクの結果

update_istap_config API は、入力されたデータを使用して以下のタスクを実行します。

- S-TAP から Guardium システムに項目を送信するために使用されるメッセージ・キューを作成し、INSTEAD OF トリガー (メッセージ・キューに項目を送信する) による、ビューを使用したグローバル・データベース・モニターを開始します。
- PASE および S-TAP を開始します。
- QAUDJRN からジャーナル項目を受け取り、それをメッセージ・キューに追加します。

親トピック: Db2 for IBM i S-TAP

External S-TAP

IBM® Guardium® 外部 S-TAP® は、Guardium のコンポーネントであり、データベース・サーバーに検査エージェントをインストールすることなく、クラウドおよびオンプレミスのデータベース・サービスのトラフィックをインターセプトできます。外部 S-TAP コンポーネントは、Docker イメージとして利用でき、どのサポート対象環境にでもインストールできます。

External S-TAP は、クライアントとデータベース・サーバーの間のトラフィックをインターセプトして、トラフィックのコピーを分析およびポリシー適用のために Guardium コレクターに転送します。図 1 に示すように、External S-TAP をクラウドおよびオンプレミスのいずれのデータベースでも使用できます。

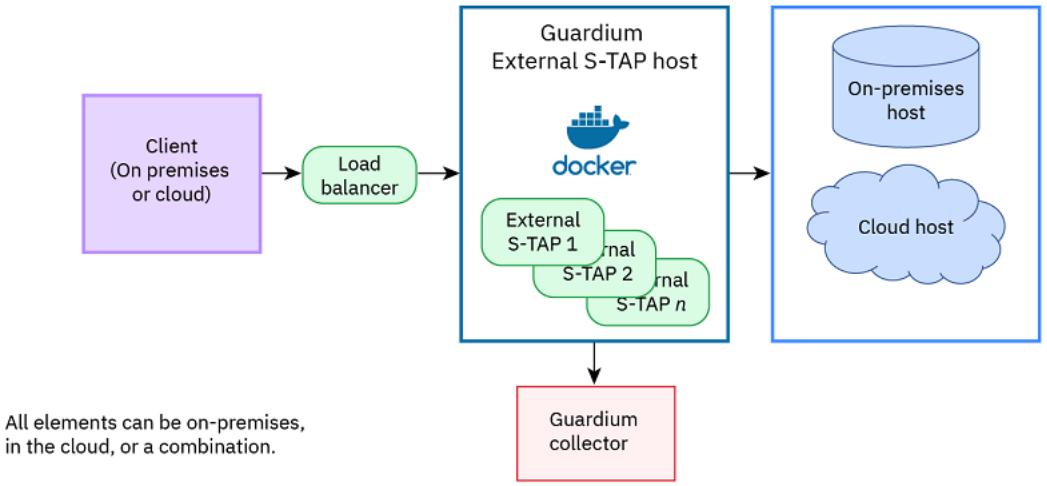


図 1. Guardium External S-TAP の概要

図 1 に示すように、External S-TAP を使用する Guardium システムのエレメントはすべてオンプレミスでもクラウドでも配置できます。External S-TAP は高度な構成が可能です。Guardium External S-TAP のデプロイで説明されているように、デプロイメント時に大半のオプションを構成できます。

ロード・バランシングは、ハードウェア・アプライアンスまたはソフトウェアのどちらを使用しても実行できます。ロード・バランシング・ソリューションのデプロイについて詳しくは、ロード・バランサー・スクリプトの準備を参照してください。

Docker コンテナーの使用

Docker コンテナーは、ソフトウェア・ソリューションを簡単にダウンロードして管理できるようにパッケージする手段となります。ご使用のサイト構成に応じて、Guardium External S-TAP Docker コンテナーを Docker ストアから直接ダウンロードするか、インターネット・アクセスのないコンピューターではプライベート・イメージ・リポジトリーから入手できます。

Docker コンテナーはイメージとして実行されます。これは、ホスト・データベースにインストールできるパッケージ・ソフトウェア・ソリューション (この場合は External S-TAP) です。External S-TAP ホストとして機能するマシン上に複数のコンテナーをインストールできます。

External S-TAP をデプロイする準備

ご使用のサイトで既に Guardium が使用されていることを前提として、External S-TAP コンテナーを実行する各データベースで以下のステップを実行する必要があります。

1. ご使用のサイトで暗号化トラフィックを管理している場合 (つまり、SSL 対応)、認証局 (CA) と協力し、適切なセキュリティー証明書を使用して Guardium コレクターを準備する必要があります。このステップには多少の時間がかかることがあります。外部の企業 (CA) と協力する必要があるためです。詳しくは、External S-TAP の SSL 証明書の取得を参照してください。ご使用の環境で SSL が有効になっていない場合は、このステップをスキップできます。
2. Linux 環境を使用できることを確認します。Docker は、Linux でインストールされて実行されている必要があります。詳しくは、https://www.docker.com/ および Docker コンテナーのダウンロードを参照してください。
3. ロード・バランサー・スクリプトおよびExternal S-TAP デプロイメント・スクリプトを準備します。詳しくは、Guardium External S-TAP のデプロイを参照してください。

External S-TAP は、デプロイされた後、自動的に実行されます。External S-TAP は Guardium から管理できます。詳しくは、External S-TAP ページの操作を参照してください。

External S-TAP の要件

External S-TAP は、オンプレミスまたはクラウド・ベース (Oracle では AWS、MSSQL では Azure) のいずれでも Microsoft SQL (SSL 対応) または Oracle データベース (SSL が使用されているかどうかに関係なく) で使用できます。

ロード・バランシング・ソリューションが必要です。利便性を考慮して、ロード・バランサーの構成に役立つスクリプトが用意されています。

External S-TAP コンテナー・ホストがオンプレミス・マシンまたは仮想マシンである場合、ホストは以下の要件を満たしている必要があります。

- x86_64 プロセッサー。
- Linux カーネル・バージョン 3.10 以上 (最新バージョンを推奨)。
- Iptables 1.4 以上。
- Docker (Docker CE または Docker EE) 1.12.16 以上。
- UNIX ドメイン・ソケットを使用できること。
  重要: オンプレミスのインストール済み環境では、ホスト・システム上でコンテナーを開始するユーザーに対して公開鍵認証を有効にすることをお勧めします。デプロイメント・スクリプトは、ホスト・システムに対して ssh を複数回呼び出します。公開鍵認証を使用すると、このプロセスが簡素化されます。

さらに、オンプレミスでもクラウドでも、すべてのインストール済み環境が以下の要件を満たしている必要があります。

- Docker については、インストールするユーザーに、システム間でコンテナーを作成するために必要な特権があることを確認してください。
- Docker ストア、または管理者が Docker ストアからイメージをプッシュできるプライベート Docker レジストリーへのネットワーク・アクセスを利用できることを確認してください。
- TCP を使用して、データベース・クライアントがExternal S-TAP ホストに接続でき、External S-TAP ホストがデータベース・サーバーに接続できることが必要です。
- すべてのExternal S-TAP ホストをクライアントとデータベース・ホストの間に配置できるネットワーク・トポロジーで配置します。 理想的には、クライアント、External S-TAP ホスト、およびデータベース・サービスの間の待ち時間を可能な限り短くします。

- External S-TAP ホストへのアクセスが保護されていることを確認してください。
- Docker では、コア・ファイルを配置する場所を判別するためにホストのカーネル・コア・パターンを使用します。一部のシステムでは、デフォルトのパスはコンテナーの観点からは適切ではありません。コア・ファイルが正確に保管されるように、以下のパターンを使用します。

  ```
  '/tmp/core.%t.%e.%p'
  ```

  例えば、コンテナーが実行されるExternal S-TAP ホストでは、以下のコマンドを実行してコア・パターンを設定します。

  ```
  echo '/tmp/core.%t.%e.%p' | sudo tee /proc/sys/kernel/core_pattern'
  ```

- External S-TAP の SSL 証明書の取得
  SSL 対応システムを保護するために、Guardium 外部 S-TAP では、VeriSign や Entrust などの信頼された認証局 (CA) から Secure Sockets Layer (SSL) デジタル証明書を取得する必要があります。ご使用のExternal S-TAP 環境で SSL が有効になっていない場合は、このステップをスキップできます。
- Docker コンテナーのダウンロード
  Guardium 外部 S-TAP モニターをデプロイするには、最初に IBM Guardium External S-TAP コンテナーを Docker ストアからダウンロードする必要があります。External S-TAP ホストとして機能するマシン (実、仮想、またはクラウド) にコンテナーをデプロイします。
- Guardium External S-TAP のデプロイ
  システムで Guardium 外部 S-TAP コンテナーを実行する前に、いくつかのデプロイメント・タスクを実行する必要があります。
- External S-TAP ページの操作
  インストールされている Guardium 外部 S-TAP のモニター、開始、停止、および構成を行うには、「外部 S-TAP インスタンス」ページを使用します。

External S-TAP の SSL 証明書の取得

SSL 対応システムを保護するために、Guardium® 外部 S-TAP® では、VeriSign や Entrust などの信頼された認証局 (CA) から Secure Sockets Layer (SSL) デジタル証明書を取得する必要があります。ご使用のExternal S-TAP 環境で SSL が有効になっていない場合は、このステップをスキップできます。

SSL 証明書を取得して使用するには、以下のステップを実行します。

1. 証明書署名要求 (CSR) を生成し、サード・パーティー認証局 (CA) に問い合わせてください。
2. CA から署名付きの証明書が返されます。
3. External S-TAP 環境の Guardium コレクターまたは中央マネージャー (CM) に署名付きの証明書をインポートします。

証明書は Privacy Enhanced Mail (PEM) 形式で、BEGIN および END の区切り文字が含まれている必要があります。証明書を受け取った後、証明書をコンソールからコピーして貼り付けるか、標準インポート・プロトコルを介してインポートすることができます。
注: Guardium は、CA サービスは提供しません。

- 証明書署名要求の作成
  SSL 対応データベース用にデプロイする Guardium 外部 S-TAP ごとに証明書署名要求 (CSR) を作成します。生成された CSR の中に、この手順により、External S-TAP のインストールに必要なトークン (共有パスワード) も作成します。
- SSL 証明書の保管
  認証局 (CA) から SSL 証明書を取得した後、証明書を保管して、Guardium 外部 S-TAP をデプロイすることができます。

親トピック: External S-TAP
関連タスク:
証明書署名要求の作成
SSL 証明書の保管
関連情報:
証明書 CLI コマンド

証明書署名要求の作成

SSL 対応データベース用にデプロイする Guardium® 外部 S-TAP® ごとに証明書署名要求 (CSR) を作成します。生成された CSR の中に、この手順により、External S-TAP のインストールに必要なトークン (共有パスワード) も作成します。

このタスクについて

証明書要求およびトークンを作成するには、以下のステップを実行します。

手順

1. 以下の CLI コマンドを入力します。

   ```
   create csr external_stap
   ```

2. プロンプトが表示されたら、ホスト名を入力します。これは、証明書の別名になります。

   ```
   example.yourdomain.com
   ```

3. プロンプトが表示されたら、以下の情報を証明書に示されているとおりに正確に入力します。
   - この証明書の共通名 (CN=) (データベース・サーバーの名前)。以下に例を示します。

     ```
     example.yourdomain.com
     ```

     注: CN は、ステップ 2 で入力した別名と同じです。
   - 組織単位 (OU=) (例: `external_stap`)。
   - 組織の名前 (O=) (例: `IBM`)。
   - 市区町村または局所性の名前 (L=) (例: `Boston`)。
   - 都道府県のコード (ST=) (例: `Massachusetts`)。
   - 組織の 2 文字の国別コード (C=) (例: `US`)。

- 使用する暗号化アルゴリズム (1 = DSA、2 = RSA [デフォルト]) (例: 2)。

4. 要求された情報を入力して Enter キーを押した後、証明書要求プロセスが開始されます。プロンプトが表示されたら、以下の情報を入力します。
   - 使用する鍵サイズ (1 = 1024、2 = 2048 [デフォルト]) (例: 2)。
   - 完全修飾ドメイン・ネーム形式の最大 9 つの SAN (サブジェクト代替名)。SAN を入力せずに先に進むには、Enter キーを押します。
5. システムは、CSR を生成して表示します。
6. 証明書署名要求 (CSR) 全体をファイルにコピーして貼り付けます (ノートパッドなどのエディターを使用します)。
   注: CSR 出力には、*CN* proxy_keycert *token* 形式の別名が含まれます。この別名は、証明書を保管するために必要です。
7. 次に、証明書要求を CA に送信するためにファイルにコピーして貼り付けます。証明書要求は、BEGIN ステートメントと END ステートメントの中の情報も含めて、両方のステートメントの間にあるすべての情報です。 つまり、以下の行から開始して、必ずすべてのテキストを含めてください。

   ```
   -----BEGIN NEW CERTIFICATE REQUEST-----
   ```

   次の行で終了します。

   ```
   -----END NEW CERTIFICATE REQUEST-----
   ```

次のタスク

CA から証明書が返された後、証明書を Guardium コレクターまたは Guardium 中央マネージャー (CM) に保管できます。詳しくは、SSL 証明書の保管を参照してください。
親トピック: External S-TAP の SSL 証明書の取得
関連概念:
External S-TAP の SSL 証明書の取得
関連タスク:
SSL 証明書の保管

SSL 証明書の保管

認証局 (CA) から SSL 証明書を取得した後、証明書を保管して、Guardium® 外部 S-TAP® をデプロイすることができます。

このタスクについて

SSL 証明書を取得して保管するには、以下のステップを実行します。

手順

1. 作成したファイルを CA に送信して、証明書を取得します。
2. 証明書が PEM 形式でない場合は、OpenSSL または他のサード・パーティー・ツールを使用して変換します。 例えば、PKCS7 形式から PEM に変換するには、以下の OpenSSL コマンドを使用します。

   openssl pkcs7 -print_certs -in certificate.p7b -out certificate.pem

3. 次のコマンドを使用して、CA からのルート証明書および中間証明書を保管します。

   ```
   store certificate keystore_external_stap
   ```

4. 次のコマンドを使用して、署名付きの証明書をご使用のシステムに保管します。

   ```
   store certificate external_stap
   ```

   プロンプトが表示されたら、CSR 出力ファイルに含まれていた署名付きの証明書の別名を入力します。

   注: 別名を覚えていない場合は、show certificate external_stap コマンドを使用して、証明書情報を表示します。詳しくは、証明書 CLI コマンドを参照してください。

次のタスク

証明書の作成、署名、および保管が行われた後、以下のステップを実行します。

1. まだ実行していない場合は、External S-TAP イメージが入った Docker コンテナーをダウンロードします。詳しくは、Docker コンテナーのダウンロードを参照してください。
2. ロード・バランシング・スクリプトを使用している場合は、ロード・バランサー・スクリプトの準備で説明されているようにスクリプトを準備します。
3. External S-TAP デプロイメント・スクリプトを実行して、External S-TAP ホストでExternal S-TAP コンテナーを作成してインストールします。詳しくは、Guardium External S-TAP のデプロイを参照してください。

show certificate external_stap のほかに、SSL 証明書を管理するためにその他の有用なコマンドを使用できます。

親トピック: External S-TAP の SSL 証明書の取得
関連概念:
Guardium External S-TAP のデプロイ
関連タスク:
証明書署名要求の作成
関連情報:
証明書 CLI コマンド

Docker コンテナーのダウンロード

Guardium® 外部 S-TAP® モニターをデプロイするには、最初に IBM® Guardium External S-TAP コンテナーを Docker ストアからダウンロードする必要があります。External S-TAP ホストとして機能するマシン (実、仮想、またはクラウド) にコンテナーをデプロイします。

始める前に

1. Linux 環境をExternal S-TAP ホストで使用できることを確認します。 External S-TAP の場合、Docker は Linux でインストールされて実行される必要があります。
2. SSL 対応サイトの場合は、External S-TAP の SSL 証明書の取得で説明されているように、適切なセキュリティー証明書が用意されていることを確認します。ご使用の環境で SSL が有効になっていない場合は、このステップをスキップできます。

このタスクについて

External S-TAP をデプロイする前に、Docker アカウントを作成してから、External S-TAP Docker コンテナーを Linux 環境にダウンロードします。

手順

1. Docker のアカウントがまだない場合は、www.docker.com で作成します。Docker アカウントを作成するのは簡単です (無償)。
2. ご使用のサイトで Docker が提供されていない場合は、Docker CE をExternal S-TAP ホストにインストールします。詳しくは、About Docker CE/Supported Platforms を参照してください。
3. Docker ストアにログインします。
4. 「IBM External S-TAP」を検索します。
5. 検索結果で Guardium External S-TAP を選択します。
6. 連絡先情報を入力して使用条件に同意し、「Get Content」をクリックします。
    a. 「Resources」セクションに、External S-TAP コンテナーに対する docker pull コマンドが用意されています。
    b. docker pull コマンドを使用して、Docker コンテナーをご使用の環境にダウンロードします。

   注: Docker ホスト・マシンがインターネットにアクセスできない場合は、Docker コンテナーを保管する内部リポジトリーを作成します。内部リポジトリーを作成する方法の 1 つでは複数のステップを使用します。以下に例を示します。
    a. ローカル (プライベート) Docker レジストリーを実行するためにホストを構成します。詳しくは、Deploy a registry server を参照してください。
    b. Docker がインストール済みでローカルの Docker レジストリーと store.docker.com の両方にアクセスできるホストで以下のステップを実行します。
        i. Docker ストアからExternal S-TAP Docker イメージをプルします。
        ii. External S-TAP Docker イメージをローカルの Docker レジストリーにプッシュします。
    c. イメージがローカルのレジストリーに配置された後、そのレジストリーに対するアクセス権限があるホストにExternal S-TAP コンテナーをデプロイできます。

次のタスク

External S-TAP Docker コンテナーをダウンロードした後、コンテナーを Docker ホスト・マシンにデプロイするか、必要に応じてシステムをセキュアに保つためにセキュリティー証明書を作成することができます。詳しくは、Guardium External S-TAP のデプロイまたはExternal S-TAP の SSL 証明書の取得を参照してください。

親トピック: External S-TAP
関連概念:
External S-TAP の SSL 証明書の取得
Guardium External S-TAP のデプロイ

Guardium External S-TAP のデプロイ

システムで Guardium® 外部 S-TAP® コンテナーを実行する前に、いくつかのデプロイメント・タスクを実行する必要があります。

External S-TAP をデプロイする準備

この時点では、デプロイする準備ができたExternal S-TAP コンテナーがあり、SSL が有効になっている各コレクターに有効な SSL 証明書があります。次のステップでは、デプロイメント・スクリプトを準備して実行します。スクリプトは、External S-TAP ホスト、または External S-TAP ホストに対するアクセス権限があり、Linux を実行している別のマシンから実行できます。

1. ロード・バランシング・スクリプトを使用してロード・バランシング・ソリューションを作成する予定の場合は、ロード・バランサー・スクリプトを準備する必要があります。デプロイメント・スクリプトで最初に実行するステップでは、ロード・バランシング・スクリプトを呼び出します。ロード・バランサー・スクリプトについて詳しくは、ロード・バランサー・スクリプトの準備を参照してください。
   注: ロード・バランシング・スクリプトの使用は推奨されますが、必須ではありません。ご使用のサイトでロード・バランシング・スクリプトを使用しない場合は、別のロード・バランシング戦略が準備されていることを確認してください。
2. External S-TAP デプロイメント・スクリプト container_mgmt.sh を対話モードで実行して、ご使用のサイトに適切なオプションを反復的に設定します。

注: デプロイメント・スクリプトおよびロード・バランシング・スクリプトは、GitHub (https://github.com/IBM/Guardium_External_S-TAP) で入手できます。スクリプトを実行または変更する前に、必ず、CONTRIBUTOR.md ファイルおよび README.md ファイルをお読みください。

External S-TAP のデプロイメントの例

以下の例では、External S-TAP コンテナーを対話モードで作成するスクリプトについて説明します。

1. Linux コマンド行から、cd を使用して、container_mgmt.sh スクリプトおよび lb_interface_nginx.sh スクリプトが置かれているディレクトリーに変更します。
2. コマンド行から、container_mgmt.sh スクリプトを、--state-file パラメーターおよび --lb-script パラメーターを指定して呼び出します (表 1 を参照)。以下に例を示します。

```
~> container_mgmt.sh --state-file sample_state --lb-script lb_interface_nginx.sh
```

   注: --state-file パラメーターは常に必須です。--lb-script パラメーターは、ロード・バランシング・スクリプトを使用する場合 (推奨) に必要です。--lb-script を含めない場合、警告メッセージが表示されます。

3. プロンプトが表示されたら、実行するアクションを指定します (表 2 を参照)。選択するアクションに応じて、デプロイメント・スクリプトで必要な情報が求められます。
4. この例では、External S-TAPコンテナーを作成するために、`would you like to` プロンプトで、C を選択して、Enter キーを押します。
   ヒント: External S-TAP をクラウド・ホストにインストールしている場合は、P を選択して、クラウド・インスタンスをExternal S-TAP コンテナー用に構成するために必要な環境変数のリストを出力します。
5. 次のプロンプト `What host do you want to use to host the service containers?` では、1 つ以上のホスト名 (コンマ区切り) を入力するか、Enter キーを押してデフォルト (localhost) を選択することができます。
6. スクリプトは、以下に示すように、実際のパラメーター名と引数をエコー出力します。

   ```
   What host do you want to use to host the service containers? [localhost]
   Non-interactive parameter: --svc-host localhost
   ```

   ヒント: 必須パラメーターを決定する際に、それぞれのパラメーターと引数をノートパッドやその他のエディターにコピーして、コマンド行を作成してください。次に、完成したコマンド行を Linux シェルにコピーして貼り付けることができます。
7. 各プロンプトへの応答を続行します。
   - デフォルトを受け入れるパラメーターでは、Enter キーを押します。
   - デフォルトを変更する (またはデフォルトがない) パラメーターでは、新しい値を入力して、Enter キーを押します。
8. スクリプトを正常に実行した後、ホスト・コンピューターは、Guardium ホストに ssh で接続して、コマンドを実行し、External S-TAP コンテナーおよびロード・バランサーをセットアップします。

オンプレミス環境でデプロイメント・スクリプトを実行する場合は、External S-TAP コンテナーのクラスターが作成されます。External S-TAPs は、すべてのトランザクションをモニターして、TLS の暗号化解除と再暗号化を実行し (データベースで SSL が有効になっている場合)、トラフィックを Guardium システムに転送します。また、(オプションで) クライアントが接続できるロード・バランサーを構成します。ロード・バランサーは、これらの接続をExternal S-TAP コンテナーの 1 つに転送します。

表 3 を使用して、必要なパラメーターを判別してください。パラメーターは、新規クラスターを作成する (つまり、`--c` オプションを選択する) ときに呼び出される順序で示されています。
注: ご使用のインストール済み環境での必要に応じて、どのスクリプトでも名前変更または編集を行うことができます。必ず、1 つのスクリプトに対する変更 (ロード・バランシング・スクリプトの名前変更など) が、そのスクリプトを呼び出す他のスクリプトに伝搬されるようにしてください。 パラメーター名を変更しないでください。

必要なコマンド行パラメーター

表 1 のパラメーターをコマンド行で使用できることが必要です。コマンド行に --lb-script を含めない場合は、独自のロード・バランシング・ソリューションを準備する必要があります。詳しくは、ロード・バランサー・スクリプトの準備を参照してください。

表 1. 必要なコマンド行パラメーター

パラメーター	意味
--state-file *filename*	必須。状態が記録されるファイルの名前。例: `--state-file /ext_stap_state` 注: 状態ファイルは、コマンド行で指定する必要があります。
--lb-script *filename*	ロード・バランサー・デプロイメント・スクリプトの名前。ロード・バランサー・スクリプトの名前を含めない場合は、ロード・バランサーを別に構成する必要があります。利便性を考慮して、以下の 2 つのデフォルトのスクリプトが用意されています。 • lb_interface_echo.sh - 汎用ロード・バランサー用のサンプル・スクリプト (現状のままでは機能しません)。 • lb_interface_nginx.sh - NGINX ベースのロード・バランサーを作成するためのサンプル・スクリプト。 注: ロード・バランサー・スクリプトを使用するには、コマンド行にパラメーターおよびスクリプトの名前を指定する必要があります。

アクションの選択

表 2 のパラメーターを使用して、デプロイメント・スクリプトを実行するたびに行うアクションを指定します。一般に、このスクリプトは、External S-TAP コンテナーのクラスターを作成するために (--c) 使用します。ただし、表 2 で説明されているように、同じスクリプトを使用して、クラスターの削除、Docker コンテナー環境変数の出力、その他のアクションを実行することもできます。

表 2. External S-TAP アクション・パラメーター

パラメーター	意味
--ni	非対話モードでこのスクリプトを実行します。必要に応じてパラメーターを設定するには、対話モードを使用します。 スクリプトが環境用にセットアップされた後、このスクリプトを非対話モードで実行できます。
--c	クラスターを作成します。
--d	既存のクラスターを削除します。
--e	インターセプトを有効にします。つまり、データベース・トラフィックのインターセプトを再開します。ロード・バランサー統合スクリプトが必要です。このコマンドは一般に、サポートとテストのみを目的として使用されます。
--p	クラスターを作成しませんが、Docker コンテナー環境変数を出力します。出力は、状態ファイルに保存されます。クラウドでホスティングしている場合は、クラウド・インスタンスをExternal S-TAP コンテナー用に構成するために必要な環境変数を出力するために、--p パラメーターを使用します。
--r	インターセプトを解除します。つまり、データベース・トラフィックのインターセプトを停止します。ロード・バランサー統合スクリプトが必要です。このコマンドは一般に、サポートとテストのみを目的として使用されます。
--u	既存のクラスターをアップグレードします。
--z	ゾンビ・インスタンスをクリーンアップします。つまり、まだホストで実行されている古いコンテナーを停止して削除します。

デプロイメント・スクリプト・パラメーター

残りのデプロイメント・スクリプト・パラメーターには、ホスト名、リポジトリー名、およびデプロイするコンテナーの数など、デプロイメントに重要な情報を指定します。スクリプトを対話モードで実行する場合、各パラメーターの値を求めるプロンプトが表示されます。オプション・パラメーターでは、または必須パラメーターでデフォルト値を受け入れる場合は、何も情報を入力せずに Enter キーを押して先に進むことができます。

注: 表示されるパラメーターは、External S-TAP デプロイメント・スクリプトの呼び出し時に選択するアクションに応じて異なります。

表 3. External S-TAP デプロイメント・スクリプト・パラメーター

パラメーター	スクリプトの質問/意味
--svc-host *host/ip*	`What host do you want to use to host the service containers?` オプション。コンテナーを作成する 1 つ以上のホストのホスト名または IP アドレス。複数のホスト名は、コンマで区切って入力します。 デフォルトは、`$SVC_HOST` です。

--svc-port-range *m-n*	`What is the port range for the exported service port?`
	
	オプション。Docker コンテナーのエクスポートされるポート番号。使用可能なポートの *m-n* の範囲 (両端を含む) を指定できます。その場合は、ホストは、各 Docker コンテナーのポート番号を動的に決定します。例:
	
	`--svc-port-range 6100-6500`
	
	デフォルトは 0 で、以下の値を使用します。
	
	`/proc/sys/net/ipv4/ip_local_port_range`

--svc-host-user *username*	`What user will be logging in to the host to start the service containers? .` オプション。Docker ホスト・マシン上にコンテナーを作成するユーザーのユーザー名。 デフォルトは、現在のユーザー `$SVC_HOST_USER` です。

--svc-image *image*	`Enter the hash or tag for the service container image:`
	
	Docker からのExternal S-TAP イメージの名前またはハッシュ。例:
	
	`--svc-image  store/ibmcorp/guardium_external_s-tap:v10.6.0`

--repo-user *username*	`What is the username to be used if login is required to pull the service container image?` オプション。External S-TAP Docker イメージがプルされる元のリポジトリーにログインするユーザーのユーザー名。
--repo-pass *password*	`What is the password for user?` オプション。repo-user ユーザー名のパスワード。 ヒント: デプロイメント・スクリプトを実行する前に、docker login を使用して (Docker ホスト・マシン上の) Docker にログインして、ログイン情報を ~/.docker/config.json に保存します。ログインした後は、--repo-user パラメーターおよび --repo-pass パラメーターを入力する必要はありません。
--svc-container-num *num*	`How many service containers would you like to create?` オプション。このデータベース検査クラスター用に作成するExternal S-TAP Docker コンテナーの数。 デフォルトは、1 です。
--uuid *UUID*	`Please enter a UUID for this group:` オプション。External S-TAP クラスターの UUID。デフォルトは、`uuidgen` から生成されるランダムの UUID です。
--proxy-num-workers *n*	`Enter the number of workers for each service container of Guardium External S-TAP:` オプション。使用するExternal S-TAP のワーカー・スレッドの数。クラスターが作成された後に、Guardium コレクターの「外部 S-TAP グループの編集」ページでワーカー・スレッドの数を設定できます。 デフォルトは、1 です。
--db-host *host/ip*	`Enter the hostname or IP to which the DB the Guardium External S-TAP group will be relaying traffic:` オプション。External S-TAP が接続を転送する先のコンピューターのホスト名または IP アドレス。
--db-type *string*	`Enter the type of database for the DB host:` オプション。External S-TAP トラフィックのデータベース・タイプ。クラスターが作成された後に、Guardium コレクターの「外部 S-TAP グループの編集」ページでデータベース・タイプを設定できます。オプションは以下のとおりです。 • mssql • oracle
--db-port *port*	`Enter the port for the DB to which the Guardium External S-TAP group will be relaying traffic:` オプション。このクラスターにアクセスするために必要なポート番号。クラスターが作成された後に、Guardium コレクターの「外部 S-TAP グループの編集」ページでポート番号を設定できます。

パラメーター	スクリプトの質問/意味
--proxy-protocol *n*	`If proxy protocol version 1 is enabled for the DB traffic, enter 1, otherwise enter 0:` オプション。データベース・トラフィックに対してプロキシー・プロトコルが有効になっているかどうかを指定します。詳しくは、https://www.haproxy.org/download/1.8/doc/proxy-protocol.txtを参照してください。 • 0 - 有効になっていません (デフォルト)。 • 1 - プロトコル・バージョン 1。 ロード・バランサーがプロキシー・プロトコルをサポートしている場合、プロトコルは、クライアントの IP アドレスを External S-TAP と共有するためにデータを接続の先頭に挿入します。プロキシー・プロトコルを使用する場合、接続は、クライアントから DB ホストに行われているものとして報告されます。 クラスターが作成された後、Guardium コレクターの「外部 S-TAP グループの編集」ページでこの設定を変更できます。
--invalid-cert-disconnect	`Do you wish to disconnect the clients if the DB server certificate cannot be verified? (y/n)` オプション。データベース・サーバーの証明書を検証できない場合に、クライアント接続を終了します。
--invalid-cert-notify	`"Do you wish to log an error message if the DB server certificate cannot be verified? (y/n)` オプション。データベース・サーバーの証明書を検証できない場合に、実行を続行して、警告をログに記録します。
--proxy-secret *string*	`Enter the secret token which will be used to retrieve the private keys and certificates from the Guardium Collector:` 必須。コレクターからExternal S-TAP 用に鍵を取得するために必要な共有パスワード文字列またはトークン。Guardium コレクターの CLI から `show certificate External S-TAP` を実行して、proxy-secret に入力する必要があるトークンを確認します。
--sqlguard-ip *host/ip*	`Enter the hostname or IP of the Guardium Collector:` 必須。暗号化解除されたトラフィックを中継するためのExternal S-TAP のコレクターのホスト名または IP アドレス。
--kill-after *n*	`Enter the number of seconds to wait before forcefully stopping the old containers (0 is wait 30s, but don't forcefully stop):` オプション。コンテナーの停止時に、コンテナーが *n* 秒以内にシャットダウンしない場合は、強制的に除去します。デフォルトでは、30 秒待ちますが、強制的に除去しません。 注: このパラメーターは、コンテナーの作成時には表示されません。

- ロード・バランサー・スクリプトの準備
  Guardium 外部 S-TAP では、冗長性を確保して単一障害点をなくすために、ロード・バランサーと統合する必要があります。

親トピック: External S-TAP
関連概念:
ロード・バランサー・スクリプトの準備

ロード・バランサー・スクリプトの準備

Guardium® 外部 S-TAP® では、冗長性を確保して単一障害点をなくすために、ロード・バランサーと統合する必要があります。

Guardium では、独自のスクリプトを作成するためのベースとして使用できる 2 つのサンプル・ロード・バランサー統合スクリプトを提供しています。これらのスクリプトには、ご使用の環境のロード・バランサー構成を作成して管理するために Guardium 外部 S-TAP デプロイメント・スクリプトによって呼び出される一連の関数が用意されています。ロード・バランサーは、ロード・バランサー統合スクリプトと連動して、クライアントとサーバーの間のトラフィックに影響を与えることなく、External S-TAP インスタンスをアップグレードできるようにします。
重要: ご使用のサイトでロード・バランシング・スクリプトを使用しない場合は、別のロード・バランシング・ソリューションが準備されていることを確認してください。
注: スクリプトの中で、STATE= という語句を含む行はすべて無視してください。これらの行は内部使用専用です。

サンプル・スクリプトの 1 つをベースとして使用できますが、実際の実装の詳細はニーズによって異なります。いずれのサンプル・スクリプトにも、表 1 で説明されている必要なロード・バランサー関数が含まれています。ご使用のサイトのニーズに対応するために、いずれかのスクリプトを変更することが必要になります。

- lb_interface_nginx.sh サンプル・スクリプトは、NGINX ベースの実装のサンプルを提供します。
- lb_interface_echo.sh サンプル・スクリプトは、汎用の実装に関する情報を提供して、スクリプト内のエレメントに関する情報をエコー出力します。

重要: ロード・バランシング・スクリプトを直接実行しないでください。必ず、External S-TAP デプロイメント・スクリプトからスクリプトを呼び出してください。その際、スクリプト名を最初の引数として指定します。例えば、--lb-script *filename* (*filename* はロード・バランシング・スクリプトの名前) と指定します。

ロード・バランサーは、External S-TAP のデプロイ時にアクティブ化されます。

表 1 で説明されている関数は必須であり、指定された正確な関数名を使用する必要があります。ご使用のサイトの要件に対応するために、必要に応じて各関数を変更してください。

表 1. ロード・バランシング・スクリプトの関数

関数名	意味

関数名	意味

lb_import_state()	デプロイメント・スクリプトによって作成された状態ファイルを使用して、ロード・バランサー構成を作成します。ファイル・フォーマットは次のとおりです。
	`Container 1 information`
	`Container 2 information`
	`...`
	`Container n information`
	各行には、以下の情報がコンマ区切りリストとして含まれます。
	• *コンテナーが実行されているホスト*
	• *ホスト上のコンテナー用の外部ポート*
	• *内部のコンテナーの listen ポート*
	• *データベース上の listen ポート*
	• *コンテナー名*
	lb_import_state 関数は、指定された状態から構成を準備します。状態は lb_import_state に渡され、デプロイメント・スクリプトが実行されるたびに必ず他の関数の前に 1 回呼び出されます。

lb_redirect_around_containers()	lb_import_state によって作成された構成を変更して、トラフィックを一時的に Docker コンテナー (経由ではなく) の周辺に送信します。ターゲット・サーバーの *host* および *port* を記述する 2 つのパラメーターを受け取ります。デバッグおよびテストのためにExternal S-TAP インスタンスによるインターセプトを一時的に解除するために使用されます。
lb_add_one()	追加するExternal S-TAP Docker コンテナーの 2 つのパラメーター *host* および *port* を使用します。lb_add_one 関数は、lb_import_state で準備された構成を使用して、コンテナーを構成に追加します。
lb_remove_one()	除去するExternal S-TAP Docker コンテナーの 2 つのパラメーター *host* および *port* を使用します。lb_remove_one 関数は、lb_import_state で準備された構成を使用して、コンテナーを構成から除去します。
lb_apply_config()	パラメーターを使用しません。構成の現在の状態をロード・バランサーに適用します。デプロイメント・スクリプトが実行されるたびに複数回呼び出すことができます。
lb_teardown_config()	パラメーターを使用しません。ロード・バランサーを非アクティブにします。この関数は、アンインストール・プロセスの一環としてExternal S-TAP コンテナーを除去するために呼び出します。
lb_cleanup()	パラメーターを使用しません。一時ファイルを削除するために必要なクリーンアップを実行します。この関数は、ロード・バランサー統合を二度と呼び出さない場合にのみ 1 回呼び出します。

親トピック: Guardium External S-TAP のデプロイ

External S-TAP ページの操作

インストールされている Guardium® 外部 S-TAP のモニター、開始、停止、および構成を行うには、「外部 S-TAP インスタンス」ページを使用します。

External S-TAPs をデプロイした後、「外部 S-TAP インスタンス」ページから表示して管理することができます。「外部 S-TAP インスタンス」ページでは、現在実行されているExternal S-TAPs の状況を確認して、特定のパラメーターを変更することができます。

このページを開くには、Guardium の「ようこそ」ページから「管理」 > 「アクティビティー・モニター」 > 「外部 S-TAP 制御」を選択します。

「外部 S-TAP インスタンス」ページ

「外部 S-TAP インスタンス」ページには、現在の Docker ホスト・マシンで現在使用可能なすべてのExternal S-TAPs が表示され、それらの管理に役立つツールがいくつか用意されています。

「外部 S-TAP インスタンス」ページでは、以下のツールを選択できます。

- 編集 - External S-TAP を選択して、 アイコンをクリックし、「外部 S-TAP グループの編集」ページを開きます。 詳しくは、外部 S-TAP タブを参照してください。
- 削除 - 停止しているExternal S-TAP を選択して、 アイコンをクリックし、削除します。実行中のExternal S-TAP は削除できません。
- リフレッシュ - アイコンをクリックして、「外部 S-TAP インスタンス」ページのビューを更新します。
- アクション - ドロップダウン・リストにあるアクションの 1 つを実行するExternal S-TAP を選択します。詳しくは、「アクション」メニューを参照してください。
- エクスポート - 「エクスポート」メニューから、使用可能なExternal S-TAPs に関する現行情報を保存するための以下のいずれかのオプションを選択します。
  - CSV 形式でダウンロード - 情報を Excel ファイルに保存します。
  - PDF 形式でダウンロード - 情報を PDF ファイルに保存します。
- フィルター - 「フィルター」テキスト・ボックスに文字列を入力して、指定した文字列を含まないExternal S-TAPs を除外します。例えば、65 と入力すると、ホスト IP アドレスまたはグループ UUID に数字 65 が含まれているExternal S-TAPs のみが表示されます。

External S-TAP ごとに以下の情報が表示されます。

- External S-TAP グループ - External S-TAP クラスターの名前。名前は、データベース・タイプおよび Docker ホスト・マシンの IP アドレスから作成されます。
- グループ UUID - このクラスターの UUID。UUID は、生成された UUID、またはデプロイメント時に UUID として入力された文字列のいずれかです。
- ホスト - Docker ホスト・マシンの IP アドレス。
- データベース・タイプ - このExternal S-TAP のデータベースのタイプ。
- 合計メンバー数 - このクラスター内のコンテナーの総数。各クラスターには、ロード・バランサーと 1 つ以上のExternal S-TAP コンテナーが含まれています。
- 全体状況 - このExternal S-TAP クラスターの状況。
  - すべてのExternal S-TAPs が停止している場合、状況は赤色で示されます。
  - 少なくとも 1 つのExternal S-TAP が実行中の場合、状況は緑色で示されます。

- 正常なメンバー - このクラスター内の正常なメンバーの数。複数のExternal S-TAPs で構成されるクラスターでは、「合計メンバー数」が「正常なメンバー」と異なっている場合、一部のExternal S-TAPs が停止していることが分かります。
- コレクター - このExternal S-TAP が使用している Guardium コレクターの名前。

「アクション」メニュー

External S-TAP を選択した後、「アクション」メニューから以下のいずれかのオプションを選択できます。

- 再始動 - このExternal S-TAP で実行されている S-TAP を再始動します。
- S-TAP ロギング - 「S-TAP ロギング」ウィンドウで、External S-TAP グループ、デバッグ・レベル (表 1 を参照)、および S-TAP の相互作用をモニターする期間を指定して、データを S-TAP ログ・ファイルに保存します。
- 診断の実行 - 「診断の実行」ウィンドウで、External S-TAP グループ、デバッグ・レベル (表 1 を参照)、および S-TAP 診断スクリプトを実行する期間を指定します。診断は、指定されたデバッグ・レベルで実行され、Guardium コレクターにアップロードされます。
- 無視の取り消し - ご使用のインストール済み環境で「S-TAP セッションを無視 (取り消し可能)」ルール・セットが使用されている場合、S-TAP が無視状態にあったセッションに関するデータの送信を開始するために、選択したExternal S-TAP グループに対して「適用」をクリックします。
- 詳細表示 - 選択したExternal S-TAP グループの各メンバーに関する詳細情報とバージョン情報を表示します。
- イベントの表示 - 選択したExternal S-TAP グループに関するイベントを表示します。レポートには、イベントごとにイベント・タイプ、イベントの記述、タイム・スタンプ、およびコンテナー (グループ UUID) が示されます。レポート内の任意の文字列でフィルタリングできます。例えば、エラー・メッセージを確認するには、*ERR* と入力して、イベント・タイプ LOG-ERR のイベントのみを表示します。

External S-TAP ページには以下のタブがあります。

- 外部 S-TAP タブ
  「External S-TAP」タブでは、Guardium External S-TAP に関する情報を表示または変更することができます。
- 「TAP」タブ
  External S-TAPs は、S-TAPs を使用するように設計されています。「TAP」タブでは、多数の S-TAP 関連パラメーターを設定できます。
- 「検査エンジン」タブ
  「検査エンジン」タブでは、Guardium 検査エンジンに関連するパラメーターを表示または変更することができます。
- 「コレクター」タブ
  「コレクター」タブでは、Guardium コレクターの追加、編集、または削除を行うことができます。

親トピック: External S-TAP

外部 S-TAP® タブ

「External S-TAP」タブでは、Guardium External S-TAP に関する情報を表示または変更することができます。

「External S-TAP」タブでは、表 1 で説明されている情報を表示または変更することができます。

表 1. 「External S-TAP」タブ

パラメーター	デフォルト	意味
グループ UUID		このクラスターの UUID。表示専用のパラメーターです。
メンバー数		このクラスター内のExternal S-TAPs の数。表示専用のパラメーターです。
データベース・ホスト		このExternal S-TAP クラスターによってモニターされているデータベース・ホストの名前。別のデータベースをモニターするために名前を変更できます。
聴取ポート		External S-TAP が listen しているデータベース・ホスト上のポート。
デバッグ	0	このExternal S-TAP のデバッグ・レベルを設定します。デバッグ・レベルは以下のとおりです。 • 0: デバッグなし (デフォルト) • 1: 構成デバッグ。構成および統計に関連するログを作成します。 • 2: パケット・ダイジェスト。各パケットに対するアクションの読み取りと書き込みを行います。 • 3: 拡張パケット・ダイジェスト。接続ライフサイクル、オープン、クローズ、または TLS ハンドシェークのログなどの情報、および接続ログごとの情報を組み込みます。 • 4: 完全詳細デバッグ。すべてのログおよび部分的なペイロード・ダンプが含まれます。
ワーカー・スレッド	1	クラスター内の各External S-TAP で使用されているスレッドの数。
プロキシー・プロトコル	0	ロード・バランサー・プロキシー・プロトコルに対するサポートを提供します (有効になっている場合)。例えば、NGNIX ロード・バランシング・ソリューションの場合は、proxy_protocol on を /etc/nginx/nginx.conf 構成ファイルに追加して、このタブでプロキシー・プロトコルを有効にします。
無効な証明書について		SSL 証明書が有効でない場合に実行するアクションを指定します。 注: このリリースでは「無効な証明書について」は無視されます。

親トピック: External S-TAP ページの操作

「TAP」タブ

External S-TAPs は、S-TAPs を使用するように設計されています。「TAP」タブでは、多数の S-TAP 関連パラメーターを設定できます。

「TAP」タブでは、表 1 で説明されている S-TAP 関連パラメーターを設定できます。

表 1. 「TAP」タブ

パラメーター	デフォルト	意味
すべてが制御可能	未チェック	複数のコレクターの構成では、すべてのコレクターが構成を変更できるか、または 1 次コレクターのみが変更を行えるかを指定します。デフォルトでは、1 次コレクターのみが変更を行えます。
メッセージ	未チェック	リモート・メッセージおよび syslog メッセージをオンまたはオフにします。 • リモート - 選択された場合、メッセージをアクティブな Guardium® ホストに送信します。 • Syslog - 選択された場合、システム・メッセージを syslog に記録します。
ロード・バランシング		ロード・バランシング・オプションを選択します。 • 0: ロード・バランシングなし。トラフィックは、1 つのアライブ・サーバーに送信されます。1 次サーバーの優先順位が最も高くなります。 • 1: ロード・バランシング。トラフィックは、サーバー間で分割されます。 • 2: 冗長。トラフィックは、すべてのサーバーに送信されます。 • 3: ハードウェア・ロード・バランシング。ハードウェア・ロード・バランシング・ソリューションを使用できます。 • 4: 複数のロード・バランシング。トラフィックは、複数の S-TAP® スレッドで管理 (および分割) されます。
管理対象ユニット	1	ロード・バランサーから要求する管理対象ユニット (MU) の数
圧縮レベル	1	S-TAP とコレクターの間のデータ圧縮のレベルを選択します。1 (なし) から 9 (最高) の間の圧縮レベルを選択してください。

親トピック: External S-TAP ページの操作

「検査エンジン」タブ

「検査エンジン」タブでは、Guardium 検査エンジンに関連するパラメーターを表示または変更することができます。

「検査エンジン」タブでは、表 1 で説明されているパラメーターを表示または変更することができます。

表 1. 「検査エンジン」タブ

パラメーター	デフォルト	意味
データベース・タイプ		この外部 S-TAP® のデータベースのタイプ。データベース・タイプは表示専用です。
データベース・ポート		データベースが listen しているポート。
ネットワーク	ブランク	モニターする特定のクライアントを IP アドレスで指定します。モニターする IP アドレスの範囲の開始と終了を指定してください。ブランクの場合は、すべてのクライアントがモニターされます。 注: 「ネットワーク」または「ネットワークを除外する」のいずれかを指定できますが、両方は指定できません。
ネットワークを除外する	ブランク	モニター対象から除外する特定のクライアントを指定します。除外する IP アドレスの範囲の開始と終了を指定してください。ブランクの場合は、すべてのクライアントがモニターされます。
優先順位のカウント	20	高優先度として設定するセッションのパケットの数を指定します。最初の *number of packets* がスニファーで高優先度キューに送信されます。範囲は 0 (オフ) から 50 です。

親トピック: External S-TAP ページの操作

「コレクター」タブ

「コレクター」タブでは、Guardium® コレクターの追加、編集、または削除を行うことができます。

- コレクターを追加するには、  アイコンをクリックして、「新規コレクターの追加」ウィンドウを表示します。このウィンドウで、表 1 で説明されている情報を入力してから、「保存」をクリックし、新規コレクターを保存します。

  表 1. 「コレクター」タブ

パラメーター	デフォルト	意味
Guardium ホスト		追加する Guardium コレクターの名前。
メイン・スレッド	1	ロード・バランシング・メソッド 1 または 4 を選択してロード・バランシングを分割する場合に S-TAP® とサーバーの間で開かれるメイン接続の数。詳しくは、「TAP」タブを参照してください。
プール・サイズ	1	S-TAP とサーバー上のスニファー・プロセスの間で開かれるデータ接続の数。
1 次として設定		コレクターが複数の場合に、1 次コレクターとして機能するコレクターを選択します。
フィルター		「フィルター」テキスト・ボックスに文字列を入力して、指定した文字列を含まないコレクターを除外します。

- 既存のコレクターを編集するには、そのコレクターを選択してから、  アイコンをクリックし、「コレクターの編集」ウィンドウを表示します。このウィンドウでは、コレクター名を表示して、「メイン・スレッド」および「プール・サイズ」を変更することができます (表 1 を参照)。

必要な変更を行い、「保存」をクリックして変更を保存するか、「キャンセル」をクリックして、保存せずに終了します。

- 既存のコレクターを削除するには、そのコレクターを選択して、  アイコンをクリックします。コレクターは、削除されます。
  注: コレクターを誤って削除した場合は、「キャンセル」をクリックして、変更を保存せずに「外部 S-TAP グループの編集」ウィンドウを閉じます。

親トピック: External S-TAP ページの操作

Guardium Installation Manager

Guardium® Installation Manager (GIM) を使用して、管理対象サーバー上で Guardium コンポーネントをインストールおよび保守できます。

GIM コンポーネントには GIM サーバーと GIM クライアントが含まれています。GIM サーバーは Guardium システムの一部としてインストールされます。また、GIM クライアントは、モニターするデータベースまたはファイル・システムをホストするサーバー上にインストールされている必要があります。GIM クライアントは、各管理対象サーバー上で実行される一連の Perl スクリプトです。GIM クライアントをインストールすると、これは GIM サーバーと連動して以下のタスクを実行します。

- インストールされたソフトウェアの更新がないか検査する
- 新規ソフトウェアを転送およびインストールする
- ソフトウェアをアンインストールする
- ソフトウェア・パラメーターを更新する
- データベース・サーバー上で実行中のプロセスをモニターおよび停止する

例えば、GIM を使用して S-TAP モジュールをインストールし、これを最新の状態に維持することができます。

GIM クライアントは、ポート 8444 を使用して、GIM サーバーと通信します。

GIM サーバーは、Guardium ユーザー・インターフェースまたはコマンド行インターフェース (CLI) を介して使用できます。

GIM を使用してデプロイできるソフトウェア・モジュールは、GIM バンドルとしてパッケージされます。*バンドル*とは、GIM を使用してデプロイできるソフトウェアを格納する gim タイプのファイルです。

ご使用の環境に、中央マネージャーとして構成されている Guardium システムが含まれている場合、GIM サーバーとして使用する Guardium システムを決定する必要があります。中央マネージャーのような単一の Guardiumシステム から最大 4000 個のすべての GIM クライアントを管理することも、GIM クライアントをグループとして別々の Guardiumシステム から管理することもできます。 単一の Guardiumシステムからすべての GIM クライアントを管理する場合は、その 1 つの UI で、すべての GIM クライアントの状況を表示し、関連するタスクを実行することができます。グループ内の GIM クライアントを別個の Guardium システムから管理することを選択した場合、各 UI を使用して、それが管理する GIM クライアントで作業することができます。全体的なビューは使用できません。

V9.0 GPU パッチ 50 以降からバージョン 10.0 にアップグレードする場合、GIM クライアントに関する情報の表示方法に変更はありません。それより前のバージョンからアップグレードする場合に、次の制限が適用されます。つまり、ご使用の中央マネージャーをアップグレードした後、他の Guardium システムに割り当てられている GIM クライアントに関する情報は引き続き表示できますが、これらの GIM クライアントに対するプロビジョニング作業を中央マネージャーから実行できなくなります。ご使用のすべての Guardium システムをアップグレードした後、各 GIM クライアントは、その GIM サーバーである Guardium システムからしか表示できません。

多数の GIM インストール済み環境を管理する場合、GIM クライアントのグループを作成できます。これにより、そのグループを使用して、ソフトウェア・バンドルとしてインストール、更新、管理することができます。

GIM クライアントは、ユーザーが GIM を使用してインストールしたプロセスをモニターします。GIM クライアントは、1 分に一度、各プロセスのハートビートをチェックし、それらのプロセスの状況変更を GIM サーバーに渡します。各プロセスの状況は、「プロセス・モニター」パネルに表示されます。 変更は、3 分以内に反映されます。GIM クライアント自体の状況の変更は、クライアントがサーバーをポーリングし、その「アライブ・メッセージ」を送信する間隔に従って反映されます。

注: システム・バックアップを実行し、GIM が定義されているサーバーから別のサーバーにバックアップを復元する場合、復元先のサーバーに対する GIM フェイルオーバーを構成する必要があります。この GIM 構成は、バックアップ中央マネージャーまたはシステムのバックアップとリストアに適用されます。

- モニター・エージェントをデプロイするためのクイック・スタート
  デプロイ・モニター・エージェント・ツールを使用すると、GIM クライアントのアクティブ化、S-TAP のインストール、およびデータベース・トラフィックのモニター開始を自動的に行えます。
- GIM によるソフトウェアの管理
- GIM サーバーの割り振り
  事前インストールされた非アクティブな (どのコレクターにも接続されていない) GIM エージェントにリモートで接続し、そのエージェントがデータベース・サーバーにアクセスすることなく何らかのコレクターに接続するようにします。
- Windows サーバーへの GIM クライアントのインストール
  対話式インストーラーまたはサイレント・インストールのいずれかを使用して、GIM クライアントを Windows にインストールする方法を説明します。GIM クライアントのアンインストールについても説明します。
- UNIX サーバーへの GIM クライアントのインストール
  このコマンドを使用して、GIM クライアントを各データベース・サーバーにインストールします。
- UNIX データベース上の GIM およびそのモジュールのアンインストール
  GIM とそのモジュールのアンインストールは、GUI から行う方法と、データベース・サーバー自体で行う方法があります。
- GIM クライアントのアップグレード
  GIM を使用して GIM クライアントを新しいバージョンにアップグレードできます。
- GIM でのグループの使用
  グループを使用することによって、一部の GIM タスクを実行しやすくすることができます。
- GIM の動的更新
  GIM クライアントは、GIM サーバーからの更新がないかを一定の間隔でチェックします。GIM サーバーは、使用する最適なポーリング間隔をシステムの状態に基づいて計算することができます。
- データベース・サーバーのオペレーティング・システムをアップグレードするとき
  データベース・サーバーでオペレーティング・システムをアップグレードするときに、GIM クライアントが、GIM クライアント自体と GIM によってインストールされたモジュール内で必要な変更を行えるようにすることができます。
- 管理対象ユニットへの GIM バンドルの配布
  管理対象ユニットによって管理される GIM クライアント上に GIM バンドルをデプロイするために、管理対象ユニットに GIM バンドルを配布することができます。

- 使用されていない GIM バンドルの削除
  GIM バンドルがデータベース・サーバーで使用されなくなった場合、GIM サーバーから削除することができます。
- GIM 診断の実行
  GIM サーバーが、各 GIM クライアントについて正確なデータを持っているかどうかを確認するために、GIM クライアント上で診断を実行することができます。
- GIM 動作のデバッグ
  問題をトラブルシューティングするためにデバッグをオンにすることが必要な場合があります。
- SMF サポートを備えた Solaris 用の監視プログラムの再始動
  一連の CLI コマンドを使用して、SMF サポートを備えた Solaris サーバーで監視プログラムを再始動します。

モニター・エージェントをデプロイするためのクイック・スタート

デプロイ・モニター・エージェント・ツールを使用すると、GIM クライアントのアクティブ化、S-TAP のインストール、およびデータベース・トラフィックのモニター開始を自動的に行えます。

デプロイ・モニター・エージェント・ツールは、Guardium デプロイメントを確立するプロセスを簡単にします。デプロイ・モニター・エージェント・ツールは、既存の Guardium インストール・マネージャー (GIM) インフラストラクチャーにビルドすることで、データベース・サーバーの検索、モニター・エージェント (S-TAP) のインストール、およびデータベースの検査エンジンの構成を迅速に行えるようにします。また、このツールはデプロイメント状況を追跡および検討するための一元化されたビューを提供します。

- モニター・エージェントをデプロイするための前提条件
  モニター・エージェントのデプロイを開始する前に、前提条件と制限事項を確認してください。
- モニター・エージェントのデプロイ
  S-TAP のデプロイと検査エンジンの構成を迅速に行う方法について説明します。

親トピック: Guardium Installation Manager

モニター・エージェントをデプロイするための前提条件

モニター・エージェントのデプロイを開始する前に、前提条件と制限事項を確認してください。

デプロイ・モニター・エージェント・ツールを使用してデータベース・サーバーに S-TAP をインストールし、検査エンジンを構成する前に、以下の前提条件を確認してください。

ターゲットの S-TAP のインストール・ディレクトリーは、存在しないか、または空である必要があります。
: 既にファイルが格納されているディレクトリーに、S-TAP をインストールすることはできません。

S-TAP の前提条件
: Windows: 前提条件: S-TAP のインストール
: Linux システムおよび UNIX システム: S-TAP のインストール前提条件

GIM クライアントをリスナー・モードでインストールする
: ご使用の環境内の 1 つ以上のデータベース・サーバーに GIM クライアントをリスナー・モードでインストールします。Windows システムで GIM クライアントをリスナー・モードでインストールするには、`--host` パラメーターを省略します。AIX や Linux などのシステムで GIM クライアントをリスナー・モードでインストールするには、`--sqlguardip` パラメーターを省略します。GIM リスナー・モードについて詳しくは、GIM サーバーの割り振りを参照してください。
: 重要: データベース・サーバーの GIM クライアントと、デプロイ・モニター・エージェント・ツールを実行する Guardium システムとの間にポートを開ける必要がある場合があります。GIM クライアントのインストール時に別のポートを指定しない限り、デフォルト・ポート 8445 が使用されます。

Guardium システムへの GIM S-TAP モジュールのアップロード
: アグリゲーターとして構成されていない任意の Guardium システムから、管理ユーザーとしてデプロイ・モニター・エージェント・ツールを実行します。作業を開始する前に、以下の手順を使用して GIM S-TAP モジュールを Guardium システムにアップロードします。

  1. 「管理」 > 「モジュール・インストール」 > 「モジュールのアップロード」にナビゲートします。
  2. 「ファイルの選択 (Choose file)」をクリックし、インストールするモジュールを選択します。
  3. 「アップロード」をクリックして、モジュールを Guardium システムにアップロードします。アップロードが完了すると、「アップロード済みモジュールのインポート」表にモジュールが表示されます。
  4. 「アップロード済みモジュールのインポート」表で、インストールするモジュールの横にあるチェック・ボックスをクリックします。モジュールがインポートされ、インストール可能な状態になります。モジュールがインポートされると「モジュールのアップロード」ページが再ロードされ、モジュールが 「アップロード済みモジュールのインポート」表に表示されなくなります。

  S-TAP オファリングとサポート対象プラットフォームについて詳しくは、System requirements and supported platforms for IBM Security Guardium を参照してください。

検出可能なデータベース・サーバーがすべて実行中であることを確認する
: 検査エンジンは、以下を含む一部のデータベース用に自動的に構成できます。

  - Db2 for Linux, UNIX, and Windows
  - Informix
  - Microsoft SQL Server
  - MySQL
  - Oracle
  - Postgre SQL
  - Sybase
  - Teradata

  検査エンジンの自動構成を許可するには、モニター・エージェントをデプロイする前にデータベース・サーバーが稼働していることを確認してください。

  データベース・インスタンスの自動ディスカバーについて詳しくは、データベース・インスタンスのディスカバーおよび Windows: データベース・インスタンスのディスカバーを参照してください。

親トピック: モニター・エージェントをデプロイするためのクイック・スタート

モニター・エージェントのデプロイ

S-TAP のデプロイと検査エンジンの構成を迅速に行う方法について説明します。

始める前に

アグリゲーターとして構成されていない任意の Guardium システムから、管理ユーザーとしてデプロイ・モニター・エージェント・ツールを実行します。インストールを開始する前に、以下の点を確認してください。

- GIM クライアントがリスナー・モードでインストールされている。
- GIM S-TAP モジュールが Guardium システムにインポートされている。
- 検出可能なデータベース・サーバーが実行されている。

詳しくは、モニター・エージェントをデプロイするための前提条件を参照してください。

このタスクについて

以下の手順は、データベース・トラフィックをモニターするために、デプロイ・モニター・エージェント・ツールを使用して S-TAP のインストールおよび検査エンジンの構成を迅速に行う方法を示しています。

手順

1. 「セットアップ」 > 「クイック・スタート」 > 「モニター・エージェントのデプロイ」にナビゲートし、デプロイ・モニター・エージェント・ツールを開きます。
2. 「データベース・サーバーの識別」セクションの「IP アドレス」フィールドを使用して、リスナー・モードの GIM クライアントを検索する IP アドレスの範囲を指定します。 アイコンを使用して追加の IP アドレスを指定します。検索を拡張するには、ワイルドカード文字 (*) や範囲文字 (–) を含めます。例えば、`10.0.0-5.*` などです。複数の完全な IP アドレスまたは範囲を区切るには、コンマを使用します。例えば、`9.70.145.165,9.70.145-148.165,9.70.145.*` のようにします。
   重要: 大量の IP アドレスをスキャンすると時間がかかり、スキャンが完了する前にタイムアウトになる可能性があります。「IP アドレス」フィールドを使用して、リスナー・モードの GIM クライアントが検出されると思われる狭い範囲の IP アドレスを定義します。
3. 「ディスカバー」をクリックして、リスナー・モードの GIM クライアントのスキャンを開始します。
   ヒント: デフォルトでは、GIM クライアントのディスカバリーとモニター・エージェント (S-TAP) のデプロイメントは、まずディスカバリーして、次にデプロイメントするという 2 つのステップで実行されます。そのため、以下のステップで説明されているように、S-TAP をインストールするデータベース・サーバーを手動で選択できます。

   ただし、IP アドレスのスキャン中にディスカバーされた、互換性のある GIM クライアントすべてに S-TAP を自動的にインストールすることによって、プロセスを簡素化することができます。自動化モードを有効にするには、 をクリックして「設定のカスタマイズ」ダイアログを開き、「ディスカバーされたデータベース・サーバーにエージェントを自動的にデプロイ」を選択します。自動化モードを使用する場合は、スキャンする IP アドレスを指定した後に「ディスカバーおよびデプロイ」ボタンをクリックするだけです。
4. 「データベース・サーバーの状況」セクションで、モニター・エージェントをデプロイするデータベース・サーバーを選択し、「エージェントのデプロイ」をクリックして「モニター・エージェントの構成」ダイアログを開きます。
5. 「モニター・エージェントの構成」ダイアログから、インストール・パラメーターを確認して調整します。「適用」をクリックし、モニター・エージェントのインストールを開始します。

   ほとんどの新規デプロイメントは、デフォルト・パラメーターでうまく機能します。ただし、特定の環境に応じて以下の設定を調整することができます。

   Windows インストール・ディレクトリー

   Windows データベース・サーバーにデプロイされる S-TAP のインストール・ディレクトリーを指定します。他のプラットフォームにデプロイする場合にはこのパラメーターは無視され、デフォルトのインストール・パスが使用されます。S-TAP インストール・パラメーターについて詳しくは、S-TAP のコマンド行パラメーターおよび GIM インストール・パラメーターと S-TAP インストール・スクリプト・パラメーターを参照してください。

   Guardium コレクターの割り当て

   一元的に管理された環境で Guardium コレクターの相対的な負荷や可用性に基づいて自動的に S-TAP を割り当てるには、「エンタープライズ・ロード・バランシングの使用」を選択します。詳しくは、エンタープライズ・ロード・バランシングを参照してください。

   特定の Guardium コレクターに S-TAP を割り当てるには、「コレクターの指定」を選択します。
6. 「データベース・サーバーの状況」セクションで、「S-TAP のインストール状況」列を使用してモジュール・インストールの進行状況をモニターします。`Installed` 状況は、インストールが正常に完了したことを示します。

次のタスク

データベース・サーバーの「S-TAP のインストール状況」に `Failed` のマークが付けられている場合は、 アイコンをクリックして問題の詳細を確認します。モニター・エージェントをデプロイしようとした後に「データベース・サーバーの状況」からデータベース・サーバーが消える場合は、「エラー・ログ」をクリックして問題の詳細を確認します。

ヒント: 「エラー・ログ」には、デプロイ・モニター・エージェント・ツールに関連した問題が収集されます。例えば、インストールに必要なモジュールがデプロイ・モニター・エージェントで見つからない場合は、「エラー・ログ」 にメッセージが追加されます。その他のエラーはコンポーネント固有のログに記録され、「S-TAP のインストール状況」列の アイコンをクリックすることで調査に使用できます。

モニター・エージェントが正常にデプロイされたら、データベース・サーバー上のトラフィックをモニターし、セキュリティー・コンプライアンス要件への適合を始める準備は完了です。コンプライアンス・モニターを構成するには、「セットアップ」 > 「クイック・スタート」 > 「コンプライアンス・モニター」にナビゲートし、詳細について「コンプライアンス・モニターのクイック・スタート」を参照してください。

親トピック: モニター・エージェントをデプロイするためのクイック・スタート

GIM によるソフトウェアの管理

- クライアント別の設定
  Guardium Installation Manager (GIM)「クライアント別の設定」ツールを使用して、S-TAP とその他のソフトウェア・パッケージを迅速にデプロイします。
- GIM ユーザー・インターフェース
  GIM はモジュールの自動インストール機能の提供を目的とし、各データベース・サーバーおよび Guardiumシステムごとに常駐する GIM クライアントと GIM サーバーを活用します。
- GIM コマンド行インターフェース
  データベース・サーバー上でモジュールをインストールまたはアップグレードするために、CLI を使用できます。

親トピック: Guardium Installation Manager

クライアント別の設定

Guardium Installation Manager (GIM)「クライアント別の設定」ツールを使用して、S-TAP とその他のソフトウェア・パッケージを迅速にデプロイします。

始める前に

「クライアント別の設定」ツールを使用する前に、次の事項を確認してください。

- GIM クライアントが、データベース・サーバーにインストールされ、Guardium システムに接続されていること。
- 互換性のある GIM バンドルがアップロードされ、Guardium システムにインポートされていること。

手順

1. 「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」にナビゲートします。
2. 「クライアントの選択」セクションで、GIM を使用してソフトウェアをインストールまたは更新するデータベース・サーバーを選択します。表内のチェック・ボックスを使用して個々のクライアントを選択するか、「クライアント・グループを選択してください」メニューを使用してクライアントのグループを選択します。
   重要:
   - クライアント・グループを作成するには、  をクリックして、「クライアント・グループの作成」ダイアログを開きます。「クライアントの追加」をクリックして、「既存のクライアント」ウィンドウを開き、クライアントを選択して、「OK」を選択します。「CSV からインポート」をクリックして CSV ファイルを選択し、CSV ファイルからインポートすることもできます。
   - クライアント・リストを変更した後、  をクリックしてクライアント・リストを更新します。
   - クライアントを再登録する前に Guardium システムから GIM クライアント情報を削除するには、「接続のリセット」を使用します。「接続のリセット」をクリックした後、GIM クライアント・プロセスの現在の状況が反映されるまで数分かかることがあります。
   - クライアントを選択して、「インストール済みモジュールの表示」をクリックし、「インストール済みモジュールの表示」ウィンドウを開きます。このウィンドウには、このクライアントにインストールされているすべてのモジュール (S-TAP など)、それぞれのバージョン、および選択されたすべてのクライアントでいずれかのモジュールが保留状態であるかどうかが表示されます。
   - グループを作成または更新し、GIM クライアントの「クライアント名」を編集する場合、ホスト名とアドレスは、Guardium システムに接続されている GIM システムの有効な値を反映している必要があります。無効なホスト名が指定された場合、編集後のクライアントはグループのメンバーとして表示されません。IP アドレスによるクライアントの追加はサポートされていません。

   「次へ」をクリックして先に進みます。
3. 「バンドルの選択」セクションで、「バンドルを選択してください」メニューを使用して、インストールまたは更新するソフトウェアを特定します。「次へ」をクリックして先に進みます。 ソフトウェア・バンドルを選択すると、「選択されたバンドルのアクション」列に、各クライアントに対して実行される次のアクションが示されます。

   インストール
   : 選択したバンドルがクライアントにインストールされます。このアクションは、クライアントへのソフトウェアの初回のインストールを示します。

   アップグレード
   : バンドルがクライアント上でアップグレードされます。このアクションは、ソフトウェアの旧バージョンがクライアントに現在インストールされていることを示します。

   パラメーターの更新
   : バンドルのパラメーターがクライアント上で更新されます。このアクションは、選択したソフトウェアと現在インストールされているソフトウェアが同じバージョンであることを示します。

   なし (バンドルが見つかりません)
   : アクションは実行されず、選択したバンドルに対してクライアントに互換性のあるアクションがないことを示します。

   なし (より新しいバージョンがインストールされています)
   : 選択したバンドルは、クライアントに現在インストールされているバージョンより古いため、アクションは実行されません。ソフトウェアの古いバージョンをインストールするには、目的のバージョンをインストールする前に、現在インストールされているバージョンをアンインストールしてください。

   ヒント:
   - 名前、モジュール、選択されたバンドルのアクション、およびクライアント OS などでクライアントをフィルタリングできます。結果の選択内容は保持されます。アクションは、**クライアントのフィルタリングされたリストにのみ適用されます**。「クライアントの選択」セクションに表示されるクライアントの数が「クライアントの構成」セクションに表示される数よりも大きくなります。
   - バンドルの旧バージョンを表示して操作するには、「最新バージョンのみを表示」チェック・ボックスをクリアします。
   - バンドル内の個々のモジュールを特定するには、「バンドルのみを表示」チェック・ボックスをクリアします。
   - 選択したバンドルと互換性がないクライアントを非表示にするには、「互換性のあるクライアントのみを表示」チェック・ボックスを選択します。

   重要:
   - デフォルトでは、「バンドルの選択」メニューには、プラットフォームや選択したクライアントとの互換性に関係なく、アップロードされた最新のバンドル・バージョンのみが表示されます。特定のプラットフォームまたはクライアントに対して異なるバンドル・バージョンをインストールするには、「最新バージョンのみを表示」チェック・ボックスをクリアし、必要なバンドルを選択してください。
   - 「クライアント別の設定」ツールを使用中に新しいバンドルをアップロードしてインポートした場合、その新しいバンドルを表示するには、ブラウザーをリフレッシュします。
   - 既にバンドルのインストールのスケジュールが設定されている場合、新しいバンドルをインストールすると、既存のスケジュールが削除されます。

4. 「パラメーターの選択」セクションで、必須パラメーターとオプション・パラメーターの値を指定します。オプション・パラメーターを追加または削除するには、  アイコンまたは  アイコンを使用します。名前または説明でパラメーターを検索するには、  アイコンを使用します。「次へ」をクリックして先に進みます。
   重要: クライアント固有のパラメーターとして特定された場合を除き、「パラメーターの選択」セクションで指定された値は、ソフトウェアのインストール、アップグレード、または更新先のすべてのクライアントに適用されます。クライアント固有のパラメーターについては、値のフィールドが無効になり、「クライアントの構成」セクションでクライアントごとに値が定義されます。
5. 「クライアントの構成」セクションで、表を使用して、各クライアントのパラメーター値を検討し、編集します。 編集可能なパラメーターには、パラメーター値の横に  アイコンが表示されます。その  アイコンをクリックして、値を編集します。「選択されたバンドルのアクション」列に、各クライアントに対して実行されるアクションが示されます。
6. 「インストール」をクリックして、ソフトウェアのインストールを開始します。  アイコンを使用して、インストールをスケジュールし、「OK」 をクリックして続行します。
7. 「クライアント別の設定」で現在の構成の Guardium API 構文を作成するには、「GuardAPI の生成」をクリックします。使用できる情報が十分な場合、「GuardAPI コマンド」ダイアログで複数のクライアントの API コマンドが生成されます。情報が十分でない場合は、デフォルトのテンプレートが表示されます。

次のタスク

「成功」ダイアログで、「状況の表示」をクリックして、「状況」ウィンドウを開き、ソフトウェアのインストール/アップグレードをモニターします。  をクリックして、結果を最新表示します。インストール/アップグレードが失敗状況の場合、ボタンが表示されていれば、「アンインストール」をクリックします。表示されていない場合は、「接続のリセット」をクリックします。

バンドルまたはモジュールの「失敗」のインストール状況が表示される場合は、「バンドルの選択」セクションを開き、クライアントを選択して、「アンインストール」ボタンをクリックして、  アイコンを使用して、インストール状況をモニターします。「アンインストール」ボタンを使用できない場合は、「クライアントの選択」パネルを開き、影響を受けたクライアントを選択して、「接続のリセット」ボタンをクリックします。接続がリセットされたら、  アイコンを使用して、クライアント・リストをモニターします。

親トピック: GIM によるソフトウェアの管理

GIM ユーザー・インターフェース

GIM はモジュールの自動インストール機能の提供を目的とし、各データベース・サーバーおよび Guardiumシステムごとに常駐する GIM クライアントと GIM サーバーを活用します。

ユーザーは、CLI を介して GIM と対話することもできます。CLI を使用した GIM によるモジュールのインストールおよびアップグレードについては、GIM コマンド行インターフェースを参照してください。

以下のタスクで Guardium Installation Manager (GIM) の GUI を使用できます。

- プロセスの モニター
- モジュール・パッケージの アップロード
- モジュールの構成、インストールまたは更新
- ロールバックのメカニズム

注: A-TAP が使用されている場合、GIM ベースでの S-TAP® のアップグレードまたはアンインストールを実行する前に、データベース・サーバーで最初に A-TAP を無効にする必要があります。
注: GIM はネイティブ S-TAP インストーラー (rpm、dept、bff など) をサポートしていません
注: GIM ユーティリティーを使用して特定のクライアントにモジュールを初めてインストールする場合、バンドルの形式にする必要があります。インストール済みのバンドルに属する特定モジュールの将来のアップグレードは、単一のモジュールまたはバンドルとして提供されます。

プロセスのモニター

サーバー上の GIM プロセスの状況を表示します。

監視プログラム

GIM モニター・プログラムは、Guardium® プロセスのモニターおよび モニターを主な目的としたプロセスです。具体的には、この監視プログラムは常に すべての Guardium プロセスの開始、停止、またこれらのプロセスの稼働を確認し、失敗した場合はプロセスを再始動する役割を担っています。

注: Guardium V9.0 では、Solaris 5.10/5.11 上で GIM と SUPERVISOR が SMF サービスになります。これらは inittab エントリーではなくなります。

gim/supervisor を開始および停止するには、以下を使用します。

svcadm -v enable guard_gim

svcadm -v enable guard_gsvr

svcadm -v disable guard_gim

svcadm -v disable guard_gsvr

GIM

GIM プロセスは GIM クライアント・プロセスです。このプロセスの役割は GIM サーバーへの登録、ソフトウェアの更新チェック要求の開始、新規ソフトウェアのインストール、モジュール・パラメーターの更新、モジュールのアンインストールなどです。

モジュール・パッケージの アップロード

モジュール・パッケージ・ファイル (1 つ以上のモジュールのサブパッケージを含む単一の .gim ファイル) をデータベースにロードします。

1. 「管理」 > 「インストール管理」 > 「アップロード」をクリックして、「アップロード」を開きます。
2. 「表示」をクリックして、該当パッケージ (.gim ファイル) のディスク上の場所を表示します。
3. 「アップロード」をクリックしてパッケージをアップロードします。
4. 「アップロード済みモジュールのインポート」の下にあるアップロード済みパッケージの「インポート」 アイコンをクリックして、パッケージをロードします。

モジュールの構成、インストールまたは更新

最新の GIM ソフトウェア管理ツールについては、クライアント別の設定を参照してください。

GIM における Windows S-TAP パラメーター

S-TAP のインストール時、または S-TAP 構成を更新するために、「クライアント別の設定」ページの WINSTAP_CMD_LINE フィールドを使用できます。

「クライアント別の設定」ページの「パラメーターの選択」リボンに、[TAP] パラメーターの構文 parameter=value のコマンド WINSTAP_CMD_LINE または構文 -param value の CLI パラメーター (Windows: S-TAP コマンド・ライン・インストールのパラメーター) を使用して、任意のパラメーターを入力できます。これは、guard_tap.ini で追加または更新されます。

注意:
このフィールドへの入力の検証は行われません。
例えば、次のコマンド行オプションの場合、CAS および名前付きパイプ・サポートのインストールがスキップされます。

```
CAS=0 NamedPipes=0
```

インストール中の S-TAP が MSSQL データベースを自動的にディスカバーしないようにする場合、WINSTAP_CMD_LINE 列に `START=0` と入力し、インストール時に S-TAP が開始されないようにします。以下のとおり GIM API を使用して、単一のデータベース・サーバーに対してこのパラメーターを指定することもできます。

```
grdapi gim_update_client_params clientIP=xx.xx.xx.xx paramName=WINSTAP_CMD_LINE paramValue="START=0"
```

インストール時に、追加の guard_tap.ini パラメーターも設定しなければならない場合があります。例として、「paramValue="START=1 !client_timeout_sec=120&use_tls=1!"」が挙げられます。

注: GuardAPI コマンドを使用する場合は、上記の例のように、WINSTAP_CMD_LINE の paramValue を引用符で囲み、各パラメーターをスペースで区切る必要があります (例: paramValue="START=1 CAS=0")。スペースを挿入しないと、後続のインストールが予期したように実行されない場合があります。

ロールバックのメカニズム

GIM のロールバック・メカニズムの目的は、インストール中のエラーを処理し、 モジュールをリカバリーして以前の状態に戻すことです。ロールバックのメカニズムは以下のリカバリー・シナリオをサポートします。

1. ライブ・アップグレードのリカバリー
   バンドルの場合
   - バンドルのインストールの場合、そのバンドル内でインストールに失敗したモジュールを ロールバックします。
   - `NO_ROLLBACK` とマークされたモジュール (<MODULE>_NO_ROLLBACK=1 という読み取り専用パラメーターの形式) は、失敗が発生した場合にロールバックされません。S-TAP と KTAP はこのような 2 つのモジュールで、いったん正常にインストールされると、別のモジュールで失敗が発生した場合にロールバックされません。

   バンドル以外の場合
   - スクラッチ・インストールの場合、ロールバックはスタンドアロン・モジュール の削除を伴いますが、アップグレードの場合は、以前のバージョンに戻されます。
2. ブート・タイム・インストールのリカバリー
   システムのリブート時にインストールが失敗した場合は、リカバリーを完了するために 2 回目のシステム・リブートが必要になります。リブート後も、IP-PR 状態のままになり、GIM_EVENT エントリーには、リカバリー・プロセスを完了するために 2 回目のリブートが必要であることが示されます。2 回目のリブート後に、モジュール/バンドルの状態は "FAILED" 状態を示します。
   注: 状態が 'IP-PR' の場合、 データベース・サーバーのブート方法は OS によって異なります (以下の方法以外でシステムをリブートした場合は、 保留中のモジュールは保留状態のままになります)

   ```
   Linux   : shutdown -r
   SuSe    : reboot
   HP      : shutdown -r
   Solaris : shutdown -i [6|0] (注: 「0」を使用できるのは、端末サーバーから shutdown を実行する場合のみです。)
   AIX     : reboot
   Tru64   : reboot
   ```

   注: また、リブートの前に A-TAP インスタンスを使用不可/非活動化する必要があります。

親トピック: GIM によるソフトウェアの管理

GIM コマンド行インターフェース

データベース・サーバー上でモジュールをインストールまたはアップグレードするために、CLI を使用できます。

以下は、一般的なシナリオの一部のみ を示す例です。サポートされるすべての CLI コマンドの完全なリストおよび詳細については、「GuardAPI GIM 関数」を参照してください。

- モジュール・パッケージ のロード
- バンドルを 使用したアップグレードまたはスクラッチ・インストール
- モジュール/バンドルの アンインストール
- インストール状況
- モジュール状態の 照会

モジュール・パッケージ のロード

モジュールを DB サーバーにインストールできるようにするには、まずそれらのモジュールを中央マネージャー GIM データベースにロードする必要があります。中央マネージャーがアーキテクチャーの一部ではない場合、パッケージを各 Guardium システムにロードする必要があります。 データベースにロードされたパッケージを取得するには、GIM UI の「パッケージのロード (Load package)」オプションを使用します。

バンドルを 使用したアップグレードまたはスクラッチ・インストール

注: スクラッチ・インストールは、古い (以前の GIM) S-TAP® がデータベース・サーバーにインストールされているケースも指します。
バンドルとは、 グループ化してまとめられたモジュールのリストです。これによりインストール・プロセスが 容易になります。モジュールのインストールまたはアップグレードには、常にバンドルを使用してください。

1. 以下のとおり、登録済みクライアント (つまり、GIM サーバーに登録済みの GIM クライアントがインストールされているデータベース・サーバー) のリストを取得します。

```
grdapi gim_list_registered_clients
ID=0
####### ENTRY 0 #######
CLIENT_ID:          1
IP:                 192.168.2.204
OS:                 HP-UX
OS_RELEASE:         B.11.00
OS_VENDOR:          hp
OS_VENDOR_VERSION:  B.11.00
OS_BITS:            64
PROCESSOR           9000
####### ENTRY 1 #######
CLIENT_ID:          2
IP:                 192.168.2.210
OS:                 Linux
OS_RELEASE:         2.6.16.54-0.2.5-smp
OS_VENDOR:          suse
OS_VENDOR_VERSION:  10.1
OS_BITS:            64
PROCESSOR           x86_64
```

2. 使用可能な最新のバンドルを 特定クライアントに割り当てます (インストールを準備するもので、実際にそのクライアントへのインストールを要求 するものではありません)

```
grdapi gim_assign_latest_bundle_or_module_to_client  clientIP=198.168.2.210 moduleName=BUNDLE-STAP
```

注: 特定のバンドルまたは特定のモジュールをクライアントに割り当てるには、ステップ 2 を以下の手順に置き換えてください。

```
gim_get_available_modules clientIP=”client ip”
gim_assign_bundle_or_module_to_client_by_version clientIP=”client ip” modulesName=”Bundle/Module name”
moduleVersion=”Bundle/Module version”
```

3. インストールをスケジュールします。

```
grdapi gim_schedule_install clientIP=192.168.2.210 date=now
```

注: 複数のクライアントをインストールする場合は、ステップ 2 からステップ 3 を繰り返してください。
注: フレキシブルな GIM スケジューリングを行う場合は、以下を使用します: now + [1-9][0-9]* minute | hour | day | week | month。例: now + 1 day、now + 3 minutes

GIM スケジューリング

すべての時刻が、Guardium のシステム時刻を基にしています。ここで使用する「now」とは、Guardium システムで指定された現在時刻のことです。 例えば「now +30 minute」となっている場合、Guardium の現在のシステム時刻から 30 分先の時刻になります。データベース・サーバー上の時刻が、インストール用に指定された Guardium システムの時刻を過ぎた場合、インストールが開始されます。

例 1: (a) Guardium のシステム時刻からマイナス 1 時間に設定されているクライアント、(b) Guardium のシステム時刻に設定されているクライアント、(c) Guardium のシステム時刻からプラス 1 時間に設定されているクライアント、という 3 つのクライアントがあるとします。

この状態で、GIM による S-TAP のインストールを「now +30 minute」に設定します。

Guardium システム (a) は、インストール用に設定された時刻よりも既に 30 分先であるため、即時にインストールを開始します。

Guardium システム (b) は、30 分後にインストールを開始します。

Guardium システム (c) は、システム (b) の 1 時間後にインストールを開始します。

例 2: 例 1 と同じ設定で、今度は「now」を指定します。

この場合、すべてのクライアントで、IP に対するインストールの状況が即時に変更されます。

モジュール/バンドルのアンインストール

```
grdapi gim_uninstall_module clientIP=192.168.2.210 module=BUNDLE-STAP date=now
```

`date=now` と指定するか、`YYYY-MM-DD HH:mm` という形式を使用します。アンインストールは、次に GIM クライアントが更新の有無をチェックしたとき (GIM_INTERVAL) に行われます。

インストール状況

クライアントが 送信した最新状況に関する追加情報は、 次のコマンドを実行して取得できます (状況メッセージは GIM_EVENTS 表内の項目として表示され、そのメッセージからレポートを生成できます)

汎用的な 状況メッセージは次の CLI コマンドで取得できます。

```
grdapi gim_get_client_last_event clientIP="client ip"
grdapi gim_get_client_last_event clientIP=winx64
grdapi gim_get_client_last_event clientIP=9.70.144.73
```

このコマンドの出力の例を以下に示します。

```
ID=0
OK
BUNDLE-STAP-8.0_r2609_1 INSTALLED
STAP-UTILS-8.0_r2609_1 INSTALLED
COMPONENTS-8.0_r2609_1 INSTALLED
KTAP-8.0_r2609_1 INSTALLED
STAP-8.0_r2609_1 INSTALLED
TEE-8.0_r2609_1 INSTALLED
ATAP-8.0_r2609_1 INSTALLED
```

モジュール状態の 照会

クライアントごとに インストールされたモジュールの状態を照会するには、次の CLI コマンドを実行してください。

```
grdapi gim_list_client_modules clientIP=”client ip”
```

次のような状態 があります。

INSTALLED
: モジュールはインストール済み。

PENDING-INSTALL
: モジュールのインストールのスケジュール設定を保留中

PENDING-UNINSTALL
: モジュールのアンインストールのスケジュール設定を保留中

PENDING-UPDATE
: モジュールのアップデートのスケジュール設定を保留中

IP
: モジュールのインストールが進行中

FAILED
: モジュールの最終操作が失敗しました

IP-PR
: モジュールのインストール・プロセスを完了するには、クライアントをリブートしてください。リブートする前に、すべての A-TAP インスタンスを非アクティブ化してください。データベース・サーバーのリブート方法は OS によって異なります (以下の方法以外でシステムをリブートした場合は、 保留中のモジュールは保留状態のままになります)

  - AIX: reboot
  - Linux : shutdown -r
  - SuSe: reboot
  - HP-UX: shutdown -r
  - Solaris : shutdown -i [6|0] (注: 「0」を使用できるのは、端末サーバーから shutdown を実行する場合のみです。)
  - Tru64: reboot

出力例

```
ID=0
####### ENTRY 0 #######
MODULE_ID:          11
NAME:               INIT
INSTALLED_VERSION   8.0_r3852_1
SCHEDULED_VERSION   8.0_r3852_1
STATE:              INSTALLED
IS_SCHEDULED:       N
####### ENTRY 1 #######
MODULE_ID:          -1
NAME:               COMMON
INSTALLED_VERSION   8.0_r0_1
SCHEDULED_VERSION   8.0_r0_1
STATE:              INSTALLED
IS_SCHEDULED:       N
####### ENTRY 2 #######
MODULE_ID:          12
NAME:               UTILS
INSTALLED_VERSION   8.0_r3852_1
SCHEDULED_VERSION   8.0_r3852_1
STATE:              INSTALLED
IS_SCHEDULED:       N
####### ENTRY 3 #######
MODULE_ID:          13
NAME:               SUPERVISOR
INSTALLED_VERSION   8.0_r3852_1
SCHEDULED_VERSION   8.0_r3852_1
STATE:              INSTALLED
IS_SCHEDULED:       N
####### ENTRY 4 #######
MODULE_ID:          14
NAME:               GIM
INSTALLED_VERSION   8.0_r3852_1
SCHEDULED_VERSION   8.0_r3852_1
```

```
STATE:                INSTALLED
IS_SCHEDULED:         N
####### ENTRY 5 #######
MODULE_ID:            15
NAME:                 BUNDLE-GIM
INSTALLED_VERSION     8.0_r3852_1
SCHEDULED_VERSION     8.0_r3852_1
STATE:                INSTALLED
IS_SCHEDULED:         N
```

親トピック: GIM によるソフトウェアの管理

GIM サーバーの割り振り

事前インストールされた非アクティブな (どのコレクターにも接続されていない) GIM エージェントにリモートで接続し、そのエージェントがデータベース・サーバーにアクセスすることなく何らかのコレクターに接続するようにします。

概要

以下のプロセス (GIM オートディスカバリーとも呼ばれます) により、事前インストールされた非アクティブな GIM エージェントにリモートで接続し、そのエージェントがデータベース・サーバーにアクセスせずにコレクターに接続するようにすることができます。

1. 非アクティブ GIM クライアントがリスナー・モードで実行され、コレクターからの接続を待機しています。
2. コレクターのグラフィック・ユーザー・インターフェース (GUI) または GuardAPI から、コレクターの IP アドレスを非アクティブな GIM クライアントに送信することができます。
3. 非アクティブな GIM クライアントは、コレクターの IP アドレスを受け入れて、その IP アドレスに接続します。

コレクターの IP アドレス (--sqlguardip) が指定されずに GIM がインストールされている場合、GIM はリスナー・モードで実行されます。GIM エージェントがサーバー・モードで実行されている場合、GIM は、証明書認証および共有パスワード検査を保持する検証済みコレクターからのみ SSL を介してメッセージを受け入れます。30 回以上連続して認証が失敗すると、GIM エージェントは要求の listen を停止し、サーバー・モードで実行されます。このアクションにより、サービス妨害 (DoS) 攻撃が回避されます。

ユーザーは、独自の証明書、共有パスワード、およびポート番号を定義できます。他の証明書を使用するには、証明書と鍵の絶対パス名をインストール・パラメーター (--key_file および --cert_file) に指定します。 GuardAPI コマンド store certificate gim を使用して、証明書をコレクター鍵ストアにロードします。

デフォルト以外の共有パスワードを設定するには、GuardAPI コマンド grdapi gim_set_global_param paramName=gim_listener_default_shared_secret paramValue=<password> を使用します。フォーマットは文字列でなければなりません。共有パスワードは、データベース・サーバーとコレクターで同一でなければなりません。

注: 暗号化されていない共有パスワードをコマンド行で指定しないでください。

デフォルト以外のポートを使用するには、インストール・パラメーター --listener_port にポートを指定します。 「GIM グローバル・パラメーター (GIM Global Parameters)」で、GIM グローバル・パラメーター gim_listener_default_port に新規ポートを設定します。

注: ファイアウォールでデフォルト・ポートまたはユーザー定義ポートを有効化する必要があります。

パラメーター

次のリストは、GIM インストール・パラメーターを説明したものです。

- --sqlguardip - GIM クライアントの接続先のコレクターの IP アドレスまたはホスト名を設定します。これが指定されていない場合、GIM クライアントは「リスナー・モード」で動作します。
- --ca_file - 認証局 PEM ファイルへの完全ファイル名パス。
- --key_file - 秘密鍵 PEM ファイルへの完全ファイル名パス。
- --cert_file - 証明書 PEM ファイルへの完全ファイル名パス。
- --shared_secret - コレクターを検査するための共有パスワードを指定します。
- --listener_port - デフォルトとは異なるポート番号を指定します。
- --no_listener - --sqlguardip が指定されていない場合でも、GIM が「リスナー・モード」で実行されないようにします。

以下の操作を実行しようとするとします。

- パラメーターの更新
- モジュールのインストール
- データベース・サーバーでの GIM の直接アンインストール

GIM エージェントはサーバー・モードを終了して、要求を処理します。 GIM クライアントは、指定されたコレクターに接続できない場合、サーバー・モードに戻ります。 GIM エージェントが有効なコレクターの IP アドレスまたはホスト名に割り当てられた後は、サーバー・モードで再実行されるように GIM サーバーを設定できません。新規の GIM エージェント・サーバー・モード・パラメーターはすべて READ-ONLY と表示されます。
注: 以下のパラメーターは、ファイル・システムに存在している必要があります。存在しない場合、インストールは失敗します。

- ca_file
- key_file
- cert_file

追加のコマンド行パラメーター

GIM の GIM インストーラーと統合インストーラーには、以下に挙げる追加のコマンド行パラメーターがあります。

--allow_ip_hostname_combo <0|1>

パラメーター名 : GIM_ALLOW_IP_HOST_COMBO

パラメーター値: 1 - 有効、0 - 無効

パラメーターのデフォルト値: 0

パラメーターの説明 : このパラメーターが有効に設定され、GIM_CLIENT_IP が DB サーバーのホスト名と異なる場合、GIM_CLIENTS.GIM_CLIENT_NAME は、`hostname`_<GIM_CLIENT_IP> の組み合わせである値を使用して設定されます。

GIM_CLIENT_IP が IP アドレスを使用して設定され、GIM_ALLOW_IP_HOST_COMBO が有効に設定されている場合、GIM のホスト名は <hostname>_<GIM_CLIENT_IP> の組み合わせになります。 これにより、「共通」のホスト名を持つデータベース・サーバー間で GIM クライアントの固有性が確保されます。

制限事項:「共通」のホスト名を使用して GIM_CLIENT_IP を設定することはできません。 「共通」ホスト名を使用した GIM_CLIENT_IP の設定は、重複 ID で登録する試みとみなされます。

サーバー・モード・グローバル・パラメーターでの GIM の設定

以下の GuardAPI コマンドを使用して、サーバー・モード GIM パラメーターを設定できます。

```
grdapi gim_set_global_param
paramName=gim_listener_default_shared_secret
paramValue=<password>
```

この値は暗号化されてデータベースに保管されます。GIM エージェントをデータベース・サーバーにインストールする場合、この値は、共有パスワードとしての暗号化されていない値と同一でなければなりません。

新規のデフォルト・サーバー・モード GIM ポートを設定するには、以下の GuardAPI コマンドを使用します。

```
grdapi gim_set_global_param paramName=gim_listener_default_port paramValue=<port number>
```

GIM エージェントをデータベース・サーバーにインストールする場合、この値は共有パスワードの暗号化されていない値と同一でなければなりません。

注: 異なるポートまたは共有パスワードを使用する場合、コレクター IP またはコレクター・ホスト名をサーバー・モード GIM エージェントに接続するたびに、共有パスワードまたはポートを指定する必要があります。

GIM リモート・アクティベーション

事前インストールされた GIM エージェントにリモートで接続し、GIM リモート・アクティベーションを使用してデータベース・サーバーにアクセスせずにそのエージェントをコレクターに接続します。

1. 「管理」 > 「モジュール・インストール」 > 「GIM リモート・アクティベーション (GIM Remote Activation)」をクリックします。
2. GIM がリスナー・モードで実行されている IP アドレスまたはホスト名を「IP / ホスト名」フィールドに入力します。それ以外の方法では、下にあるリストからサーバー・グループを選択します。
3. GIM リスナー・ポートが GIM グローバル設定と異なる場合、「GIM リスナー・ポート」に数値を入力します。デフォルト値は 8445 です。
4. GIM リスナー・パスワードが GIM グローバル設定と異なる場合、「GIM リスナー・パスワード」フィールドに共有パスワードを入力します。
5. 「実行」をクリックして情報を処理するか、「リセット」をクリックして、情報をクリアします。

注: IP アドレスまたはホスト名を入力するか、サーバー・グループを選択する必要があります。ただし、GIM リスナー・ポートおよび GIM リスナー・パスワードはオプションです。 GIM クライアントをリスナー・モードでインストールすると、共有パスワードおよび証明書の設定は、GIM クライアントを再インストールしない限り変更できません。
注:「GIM リモート・アクティベーション」の「コレクター IP」フィールドがブランクの場合、コレクターのホスト名がサーバーに送信されます。 IP を指定すると、それが代わりに送信されます。

GIM オートディスカバリー・プロセスの作成

GIM オートディスカバリー・プロセスを作成して、リスナー・モードでインストールされている GIM クライアントを特定して関連付けます。モニター・エージェントをデプロイするためのクイック・スタートを使用して、リスナー・モードでインストールされている GIM クライアントをアクティブ化することもできます。

1. 「ディスカバー」 > 「データベース・ディスカバリー」 > 「GIM オートディスカバリーの構成」にナビゲートします。
2.  アイコンをクリックして、新しい GIM オートディスカバリー・プロセスを作成します。
3. 「プロセス名」フィールドを使用してプロセスに名前を指定し、「適用」をクリックします。
4. 「プロセスにホストとポートを追加」セクションを使用して、リスナー・モードでインストールされている GIM クライアントに対するスキャンを行うようにホストを定義します。
   a. 「ホスト」フィールドを使用して、スキャン対象のホストまたはサブネットを指定します。ワイルドカード文字が使用可能です。例えば、192.168.2 で始まるアドレスをすべて選択するには、「`192.168.2.*`」と指定します。
   b. 「スキャンの追加」をクリックして、ホストまたはサブネットを GIM オートディスカバリー・プロセスに追加します。
   c. 上記の手順を繰り返して、GIM オートディスカバリー・プロセスに組み込む複数のホストまたはサブネットを定義します。

   注:
   - デュアル・スタック構成がある場合は、IPV4 アドレスと IPV6 アドレスの両方に対してスキャンを定義します。
   - 既存のホストまたはサブネットのスキャンを変更するには、既存の値を上書き入力で変更し、「適用」をクリックして、変更を保存します。
   - スキャンを削除するには、  アイコンをクリックします。タスクに、それに従属するスキャン結果がある場合は、そのスキャンは削除できません。
5. 「今すぐ 1 回実行」をクリックして GIM オートディスカバリー・プロセスを実行するか、「スケジュールの変更」をクリックして、プロセスを実行するスケジュールを定義します。スケジュールの定義については、スケジューリングを参照してください。
6. プロセスが完了した後、「結果の表示」をクリックして、ディスカバーされた GIM クライアントのリストを表示し、それらのクライアントを Guardium システムに関連付けます。
   a. 関連付ける GIM クライアントを選択します。
   b. 現行の Guardium システムにクライアントを割り当てるには「関連付ける」をクリックします。または、環境内の別の Guardium システムにクライアントを割り当てるには「コレクターの割り当て」をクリックします。
   c. 「結果」ダイアログを使用して、クライアントの関連付けの状況を確認します。関連付けが正常に行われた後、GIM クライアントはリスナー・モードではなくなり、GIM オートディスカバリーの結果のウィンドウに表示されません。
   d. 「閉じる」をクリックして、結果のウィンドウを閉じます。

GIM グローバル・パラメーター

ユーザー独自の共有パスワードまたは GIM リスナー・ポートをユーザー・インターフェースを介して定義します。

1. 「GIM グローバル・パラメーター (GIM Global Parameters)」を開くには、「管理」 > 「モジュール・インストール」 > 「GIM グローバル・パラメーター (GIM Global Parameters)」をクリックします。
2. gim_listener_default_shared_secret を選択して共有パスワードを設定するか、gim_listener_default_port を選択してポートを設定します。
3. アイコンをクリックして、選択したパラメーターを編集します。
4. 値を変更し、「保存」をクリックしてパラメーターを変更するか、「閉じる」をクリックしてページに戻ります。

親トピック: Guardium Installation Manager

Windows サーバーへの GIM クライアントのインストール

対話式インストーラーまたはサイレント・インストールのいずれかを使用して、GIM クライアントを Windows にインストールする方法を説明します。GIM クライアントのアンインストールについても説明します。

このタスクについて

Windows GIM クライアントのインストーラーは、v10.1 で .NET ベースのインストーラーに変更されました。GIM クライアントのインストーラーは、ご使用の GIM クライアントのバージョンに基づいています。

親トピック: Guardium Installation Manager

対話式インストーラーを使用した GIM クライアントのインストール

GIM クライアントを各データベース・サーバーにインストールする際に役立つウィザードが用意されています。

手順

1. GIM クライアント・インストーラーをデータベース・サーバーの任意のフォルダーに配置します。
2. setup.exe ファイルを実行して、GIM クライアントをインストールするウィザードを開始します。
3. インストール・ウィザードの質問に答えます。

次のタスク

インストールの結果は、ログ・ファイル C:¥IBM Windows GIM.ctl で確認できます。

サイレント・インストールを使用した GIM クライアントのインストール

ウィザードを使用する代わりにコマンド行から GIM クライアントをインストールすることもできます。

手順

1. GIM クライアント・インストーラーをデータベース・サーバーの任意のフォルダーに配置します。
2. コマンド・プロンプトを開き、インストーラーを配置したフォルダーの下にある GIM_Installer* フォルダーにナビゲートします。
3. 次のコマンドを改行を入れずに入力します。`setup.exe -UNATTENDED -INSTALLPATH "c:¥Program Files(x86)¥Guardium Installation Manager" -LOCALIP 10.9.876.543`
   重要:
   -
     - UNATTENDED パラメーターと LOCALIP パラメーターは必須です。APPLIANCE はオプションで、指定しない場合は、リスナー・モードがトリガーされます。パラメーター AUTO_ASSIGN_IP を使用する場合、LOCALIP は不要です。
     - クライアントを GIM リスナー・モードでインストールするには、-APPLIANCE パラメーターを省略します。リスナー・モードは、GIM クライアントを Guardium システムからリモート登録できるようにします。リスナーとしてインストールする方法の例: setup.exe -UNATTENDED -INSTALLPATH C:¥program files (x86)¥guardium¥GIM -LOCALIP 10.9.876.543。詳しくは、『GIM リモート・アクティベーション』と『GIM オートディスカバリー・プロセスの作成』を参照してください。
     - データベース・サーバーを複製して大量のデプロイメントを設定する場合は、--auto_assign_ip=1 を使用してデータベース・サーバーの有効な IP アドレスのいずれかからランダム IP アドレスを割り振ります。GIM クライアントのインストール時に auto_assign_ip と localip の両方を指定しないでください。「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」を使用して GIM_AUTO_SET_CLIENT_IP パラメーターを更新する場合は、新規設定を有効にするために GIM クライアント・サービスを再始動する必要があります。
   - **Windows GIM コマンド行インストールのリファレンス**

すべての .NET インストーラーに適用可能なパラメーター

パラメーター	説明
-UNATTENDED	サイレント・インストールを実行します 値は不要です。
-UNINSTALL	アンインストールします。値は不要です。
-INSTALLPATH	これはインストール・ディレクトリーです。デフォルトのインストール・パスは「C:¥Program Files (x86)¥Guardium¥Guardium Installation Manager」です。
-CUSTOMER	カスタマー名を変更する場合に使用します。
-COMPANY	会社名を変更する場合に使用します。
-SERVICEUSER	サービスを実行するユーザーを指定する場合に使用します。

パラメーター	説明
-SERVICEPASSWORD	ユーザーのパスワードを指定します。

GIM .NET インストーラーに固有のパラメーター

パラメーター	説明
-APPLIANCE	GIM が接続するアプライアンスのアドレスを設定するために使用します。このパラメーターを指定しないと、リスナー・モードを使用して GIM がインストールされます。
-LOCALIP	これは、GIM のインストール先サーバーの IP です。
-KEY_FILE	鍵ファイルを非デフォルト・ファイルに設定するために使用します。
-CERT_FILE	証明書ファイルを非デフォルト・ファイルに設定するために使用します。
-CA_FILE	CA ファイルを非デフォルト・ファイルに設定するために使用します。
-SHARED_SECRET	-APPLIANCE パラメーターを使用して共有秘密鍵が指定されない場合の、アプライアンスへの登録用の共有秘密鍵を設定するために使用します。
-LISTENER_PORT	-APPLIANCE パラメーターを使用しない場合、アプライアンスへの登録用のリスナー・ポートを設定します。デフォルト値は 8445 です。
-AUTO_ASSIGN_IP	値を 1 に設定するときは、ローカル IP が自動的に割り当てられるため、-LOCALIP を使用して指定しないでください。デフォルト値は 0 です。

次のタスク

インストールの結果は、ログ・ファイル C:¥IBM Windows GIM.ctl で確認できます。

GIM クライアントのアンインストール

手順

1. コマンド・プロンプトを開き、クライアントをインストールしたフォルダーの下にある GIM_Installer* フォルダーにナビゲートします。
2. 次のコマンドを入力します。

```
setup.exe -UNINSTALL
```

UNIX サーバーへの GIM クライアントのインストール

このコマンドを使用して、GIM クライアントを各データベース・サーバーにインストールします。

このタスクについて

GIM クライアントを Solaris スレーブ・ゾーンまたは AIX ワークロード・パーティション (WPAR) にインストールして使用できます。 これにより、GIM クライアントを使用して、 S-TAP をスレーブ・ゾーンや WPAR にインストールすることができます。S-TAP をスレーブ・ゾーンまたは WPAR にインストールする際には、ktap_enabled パラメーターの設定に関係なく、K-TAP は無効になっています。 GIM クライアントを使用して、CAS (構成監査システム) エージェントをスレーブ・ゾーンや WPAR にインストールすることもできます。スレーブ・ゾーンや WPAR にディスカバリー・バンドルをインストールすることはできません。グローバル・ゾーンで稼働するディスカバリー・エージェントは、他のゾーンから情報を収集できます。GIM クライアントを Solaris スレーブ・ゾーンまたは AIX ワークロード・パーティションにインストールするプロセスは、マスター・ゾーンにインストールするプロセスと同じです。インストールにかかる時間は、マスター・ゾーンへのインストールよりも数秒長くなる場合があります。マスター・ゾーンとスレーブ・ゾーンがある Solaris システムに GIM クライアントをインストールする場合、クライアントをマスター・ゾーンとスレーブ・ゾーンで同じロケーションにインストールする必要があります。このロケーションは共用ディレクトリーにすることはできません。

Solaris では、各スレーブ・ゾーン内の GIM クライアントと監視プログラム は、マスター・ゾーンで実行される GIM 監視プログラムのプロセスによって制御されます。マスター・ゾーンの監視プログラムのプロセスがシャットダウンされると、スレーブ・ゾーンの GIM プロセスもすべてシャットダウンされます。

注: GIM には 300 MB 以上のディスク・スペースが必要ですが、FAM モジュールもインストールする場合は 700 MB 以上必要です。

手順

1. GIM クライアント・インストーラーをデータベース・サーバーの任意のフォルダーに配置します。
2. インストーラーを次のように実行します。`./<installer_name> [-- --dir <install_dir> <--sqlguardip> <g-machine ip> --tapip <db server ip address> --perl <perl dir> -q]` インストーラー名の構文は、guard-bundle-GIM-<リリース・ビルド>-<DB>-<OS>_<bit>.gim.sh です。以下に例を示します。

   ```
   guard-bundle-GIM-10.5.0_r103224_v10_5_1-rhel-6-linux-x86_64.gim.sh
   ```

   重要:
   - クライアントを GIM リスナー・モードでインストールするには、--sqlguardip パラメーターを省略します。 リスナー・モードは、GIM クライアントを Guardium システムからのリモート登録に使用できるようにします。詳しくは、GIM リモート・アクティベーション および GIM オートディスカバリー・プロセスの作成を参照してください。
   - データベース・サーバーを複製して大量のデプロイメントを設定する場合は、--auto_set_gim_tapip を使用してデータベース・サーバーの有効な IP アドレスのいずれかからランダム IP アドレスを割り振ります。GIM クライアントのインストール時に auto_set_gim_tapip と tapip の両方を指定しないでくださ

い。GIM クライアントをインストールした後、「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」を使用して GIM_AUTO_SET_CLIENT_IP パラメーターを更新してください。

3. Red Hat Linux バージョン 6 以降では、次のコマンドを実行して、各ファイルが追加されたことを確認します。

```
ls -la /etc/init/gim*
ls -la /etc/gsvr*
```

Solaris バージョン 10 以降では、次のコマンドを実行します。

```
ls /lib/svc/method/guard_g*
```

その他のすべてのプラットフォームでは、次のコマンドを実行して、以下の新しい項目が /etc/inittab に追加されたことを確認します。

```
gim:2345:respawn:<Perl ディレクトリー>/perl <モジュール・インストール・ディレクトリー>/GIM/<バージョン>/gim_client.pl

gsvr:2345:respawn:<モジュール・インストール・ディレクトリー>/perl<モジュール・インストール・ディレクトリー>/SUPERVISOR/<バージョン
>/guard_supervisor
```

ここで、モジュール・インストール・ディレクトリー は、すべての GIM モジュールのインストール先のディレクトリーで、例えば /usr/local/guardium/modules などです。

4. 次のコマンドを入力して、GIM クライアント、 SUPERVISOR プロセス、および各モジュールが実行されていることを確認します。

```
ps -afe | grep modules
```

5. Guardium システムにログインして、プロセス・モニター状況を確認します。

親トピック: Guardium Installation Manager

UNIX データベース上の GIM およびそのモジュールのアンインストール

GIM とそのモジュールのアンインストールは、GUI から行う方法と、データベース・サーバー自体で行う方法があります。

手順

1. Guardium の GUI を使用してアンインストールするには、以下の手順を実行します。
   a. S-TAP バンドルのアンインストールをスケジュールします (「クライアント別の設定」)。
   b. GIM バンドルのアンインストールをスケジュールします (「クライアント別の設定」)。
   c. データベース・サーバーをリブートして、ドライバーから K-TAP を削除します。
2. 別の方法として、DB サーバー自体でアンインストールすることもできます。
   a. コマンド /full/path/modules/GIM/current/uninstall.pl を root として実行して、GIM バンドルと S-TAP バンドルの両方をアンインストールします。
   b. データベース・サーバーをリブートして、ドライバーから K-TAP を削除します。

親トピック: Guardium Installation Manager

GIM クライアントのアップグレード

GIM を使用して GIM クライアントを新しいバージョンにアップグレードできます。

手順

1. 使用可能な最新の BUNDLE-GIM.gim ファイルを Guardium システムにアップロードします。
2. GIM GUI を使用して、新しい BUNDLE-GIM.gim ファイルのインストールをスケジュールします。
3. 「i」アイコンをクリックし、「リフレッシュ」を押して、インストール・プロセスをモニターします。インストールが正常に完了すると、「INSTALLED」状況が表示されます。

親トピック: Guardium Installation Manager

GIM でのグループの使用

グループを使用することによって、一部の GIM タスクを実行しやすくすることができます。

始める前に

このタスクについて

GIM クライアントのグループを作成し、そのグループを使用して更新を管理対象サーバーに展開することができます。

手順

1. 「設定」 > 「ツールとビュー」 > 「グループ・ビルダー」をクリックします。グループ・ビルダーで、新しいグループを作成します。「グループ・タイプの記述」で、「クライアントのホスト名」を選択します。 新しいグループが、既存グループのリストに追加されます。
2. 「既存グループの変更」リストでその新しいグループを選択し、グループにメンバーを追加します。メンバーを手動で追加するか、または照会からリストにメンバーを追加することもできます。 照会からリストにメンバーを追加する場合は、「照会から取り込み」をクリックして、以下の必要な情報に注意してください。
   a. 「照会」では、「GIM」で始まるレポート名を選択します。
   b. 「列からメンバーをフェッチ」では、「GIM クライアント名」を選択します。
   c. 各「入力してください (Like)」フィールドには、マッチングする値を入力します。このフィールドをクライアントの識別に使用しない場合は、「%」を入力します。

d. グループを保存して、照会を実行するか、照会のスケジュールを設定します。

タスクの結果

そのグループを「管理」 > 「モジュール・インストール」 > 「クライアント別の設定」画面で使用して、個々のクライアントで作業する代わりに、このクライアント・セットをグループとして作業に使用することができます。
親トピック: Guardium Installation Manager

GIM の動的更新

GIM クライアントは、GIM サーバーからの更新がないかを一定の間隔でチェックします。GIM サーバーは、使用する最適なポーリング間隔をシステムの状態に基づいて計算することができます。

各 GIM クライアントは、GIM サーバーに対して定期的に「アライブ」メッセージを送信して、処理する準備ができた更新があるかどうかを確認します。このポーリング間隔は、GIM サーバーの状態に基づいて計算され、更新されます。間隔は定期的に計算され、「アライブ」メッセージに応答して、新しい値が GIM クライアントに渡されます。この機能はデフォルトで有効になりますが、代わりに固定間隔を使用したい場合は、オフに切り替えることができます。

GIM クライアントが GIM サーバーに接続しようとして 5 回連続で失敗した場合、フェイルオーバー・サーバーが指定されていれば、GIM クライアントは自動的にそちらに接続します。元の GIM サーバーが使用可能になると、GIM クライアントはその GIM サーバーへの接続を再開します。GIM サーバーとフェイルオーバー・サーバーは、それぞれ GIM_URL パラメーターと GIM_FAILOVER_URL パラメーターを使用して構成されます。

動的更新は Guardium API コマンド gim_set_global_param に以下のパラメーターを指定して制御されます。

dynamic_alive_enabled
: 動的アライブ機能のコントロール。1 – 有効、0 – 無効。デフォルト = 1

dynamic_alive_check_interval
: ポーリング間隔が再計算される間隔 (分単位)。デフォルト = 5

例:

```
grdapi gim_set_global_param dynamic_alive_enabled=0
```

各 GIM クライアントがサーバーにアライブ・メッセージを送信すると、サーバーは応答として、新しいポーリング間隔を渡すほか、そのクライアント用にスケジュールされたその他の更新があればをそれを送信します。

以下のパラメーターはバージョン 10.0 では有効でしたが、バージョン 10.1 以上からは削除されました。

- dynamic_alive_default_load_factor
- dynamic_alive_cpu_level1_threshold
- dynamic_alive_cpu_level2_threshold
- dynamic_alive_db_conn_level1_threshold
- dynamic_alive_db_conn_level2_threshold
- dynamic_alive_cpu_load_sample_time

親トピック: Guardium Installation Manager

データベース・サーバーのオペレーティング・システムをアップグレードするとき

データベース・サーバーでオペレーティング・システムをアップグレードするときに、GIM クライアントが、GIM クライアント自体と GIM によってインストールされたモジュール内で必要な変更を行えるようにすることができます。

始める前に

必要なバンドルをこのサーバー上で使用できることが必要です。Fix Central からダウンロードしてください。

このタスクについて

この手順は、オペレーティング・システムをアップグレードできるデータベースのみに関連しています。OS をアップグレードできない場合は、S-TAP をアンインストールして、アップグレードする OS と互換性のある S-TAP バンドルをインストールする必要があります。

アップグレードを手動で行うか、自動的に行うかにかかわらず、アップグレードの後すぐに、GIM によってインストールされたモジュールすべてを更新することをお勧めします。デフォルトでは、GIM によってインストールされたモジュールの自動更新は無効になっています。この手順では、自動更新について説明します。

ご使用の環境でソフトウェア・アップグレードを制御する場合は、パラメーター auto_install_on_db_server_os_upgrade=0 をそのままにしておきます。ただし、サーバー上の Guardium ソフトウェアは、アップグレードされるまで機能しません。OS のアップグレードが完了すると、GIM 構成ファイルは新しい OS バージョンを反映するために更新され、K-TAP はロードされず、STAP guard_tap.ini の ktap_installed は 0 に設定され、バンドル STAP は実行されたままになります。詳細については、K-TAP ログを参照してください。GIM_EVENTS レポートに、以降のステップが記載されたアラートが示されます。「アラート: OS ベンダー・バージョンのアップグレード (XXX -> YYY) が確認されました。データベース・サーバー OS のアップグレードが原因で GIM は自動モジュール・アップグレード用にセットアップされていません。新規クライアントの OS 属性が指定されたバンドルがロードされていて、ロードされた新規バンドルのビルド番号がこのクライアントに現在割り当てられている番号以上であることを確認してください。新規バンドルがロードされた後、すべてのモジュールの自動アップグレードを可能にするために auto_install_on_db_server_os_upgrade を有効に設定するか、インストールを手動で割り当ててスケジュールを設定してください。(Alert: OS vendor version upgrade (XXX -> YYY) has been identified ! GIM is NOT setup for automatic module upgrade due to DB server OS upgrade. Please verify that bundles with new client's OS attributes have been loaded and the build number of the new loaded bundles are the same or greater than ones currently assigned to this client. Once new bundles have been loaded, either enable 'auto_install_on_db_server_os_upgrade to allow automatic upgrade of all the modules or manually assign and schedule the installation.)」

手順

1. データベース・サーバーにインストールされている各モジュールについて、新しいオペレーティング・システムのバージョンをサポートする、このモジュールの最新バージョンが含まれている GIM バンドルを探します。各バンドルのビルド番号が、現在インストールされているバンドルの番号と同じか、それより大きい番号である必要があります。各バンドルを GIM サーバーにロードします。
2. gim_set_global_param コマンドを使用して、グローバル・パラメーター auto_install_on_db_server_os_upgrade の値を 1 に設定します。これにより、GIM サーバー上で自動更新オプションが有効になります。

   ```
   grdapi gim_set_global_param paramName="auto_install_on_db_server_os_upgrade" paramValue="1"
   ```

   デフォルトでは、このパラメーターは 0 に設定されています。これはこのオプションが無効であることを示します。
3. その他の準備をすべて完了した後、データベース・サーバー上でオペレーティング・システムをアップグレードします。

タスクの結果

OS のアップグレード後の最初のブート時に、GIM クライアントは、オペレーティング・システムがアップグレードされたことを認識します。また、自動更新オプションが有効になっているため、以下のステップを実行します。

1. GIM によってインストールされたすべてのモジュールの構成ファイルを、新しいオペレーティング・システムの属性がサポートされるように変更します。
2. すべてのモジュールを GIM サーバーに、更新後の属性で再登録します。
3. OS のアップグレードが実行されたことを示し、実行すべきアクションをリストしたアラートを GIM_EVENTS レポートに記録します。

モジュールが再登録されている場合、GIM サーバーは、以前にインストールされたバンドルと同じビルド番号を持つが、アップグレードされた OS と互換性のあるバンドルを最初に探します。このようなバンドルを検出できない場合、サーバーは新しい OS 属性をサポートする最新のバンドルを探します。サーバーは、該当するバンドルを検出できない場合、エラー・メッセージを出します。 サーバーが該当するバンドルを検出した場合は、それらのアップグレードをスケジュールし、アップグレード・プロセスを直ちに実行します。

次のタスク

GIM_EVENTS レポート内のメッセージを確認します。 GIM サーバーによって、モジュールが正常にアップグレードされたことが報告された場合は、更新後と同様に、各モジュールが正しく動作することを確認します。

GIM_EVENTS レポートに、アップグレードが正常に行われなかったことを示すエラー・メッセージが書き込まれている場合は、エラー・メッセージを調べてアドバイスがないか確認します。

スケジュールされた OS のアップグレードが完了したら、GIM サーバーで自動更新オプションを無効にします。 これにより、GIM クライアントで誤って更新プロセスが開始されるのを防ぐことができます。

```
grdapi gim_set_global_param paramName="auto_install_on_db_server_os_upgrade" paramValue="0"
```

別の OS のアップグレードを実行するときに、自動更新オプションを再度有効にすることができます。
親トピック: Guardium Installation Manager

管理対象ユニットへの GIM バンドルの配布

管理対象ユニットによって管理される GIM クライアント上に GIM バンドルをデプロイするために、管理対象ユニットに GIM バンドルを配布することができます。

始める前に

このタスクについて

すべての GIM クライアントを中央マネージャーから管理する場合は、すべての GIM クライアントに中央マネージャーから直接バンドルをデプロイすることができます。クライアントのグループを複数の管理対象ユニットから管理する場合は、中央マネージャーから管理対象ユニットに GIM バンドルを配布することができます。

配布に必要な時間は、バンドルのサイズとネットワークの状態によって異なります。相当な待ち時間があるネットワークでは、転送に数時間かかることがあります。

手順

1. 配布するバンドルを、中央マネージャー上の /var/gim/dist_packages ディレクトリーにコピーします。このディレクトリー内のすべてのファイルが配布されます。配布するバンドルを選択することはできません。
2. バンドルを配布する管理対象ユニットを選択します。
3. 「GIM バンドルの配布」をクリックします。 選択した管理対象ユニットにバンドルがコピーされます。

タスクの結果

各管理対象ユニットから、その管理対象ユニットが管理する GIM クライアントに、バンドルをインストールすることができます。
親トピック: Guardium Installation Manager

使用されていない GIM バンドルの削除

GIM バンドルがデータベース・サーバーで使用されなくなった場合、GIM サーバーから削除することができます。

このタスクについて

この機能を使用すると、GIM バンドルのインベントリーを管理し、インベントリーによってディスク・スペースが無駄に使用されるのを防ぐことができます。

2 つの新しい Guardium API コマンドを使用して、使用されていない GIM バンドルを特定し、削除できます。GIM サーバーとして機能する各 Guardium システム上で、以下の手順を実行します。

手順

1. gim_list_unused_bundles コマンドを実行して、FAM インストールの未使用のバンドルを特定します。includeLatest パラメーターは、コマンドによって返されるリストに、各 GIM バンドルの最新バージョンを含めるかどうかを指定する目的で使用します。まだ配布していないバンドルがある場合があります。また、必要になったときに再インストールできるように旧バージョンを保存しておきたい場合もあります。各バンドルの使用されていない最新のバージョンをコマンドの結果から除外する場合は、includeLatest を 0 に設定します。 使用されていないすべてのバージョンを含める場合は、1 に設定します。このパラメーターは必須で、デフォルト値は提供されていません。例:

   ```
   gim_list_unused_bundles includelatest=0
   ```

   このコマンドにより、GIM サーバー上で見つかったが、この GIM サーバーとともに動作する GIM クライアントが存在するデータベース・サーバー上にはインストールされていない GIM バンドルのリストが返されます。
2. ステップ 1 で使用されていないバンドルがいくつか示されたら、gim_remove_bundle コマンドを使用して、不要な各バンドルを削除します。このコマンドは、削除するバンドルを指定する 1 つのパラメーター bundlePackageName を取ります。このパラメーターは必須で、デフォルト値は提供されていません。
   gim_list_unused_bundles コマンドによって返された名前を指定してください。
   次の条件を満たす場合にのみ、指定されたバンドルが削除されます。
   - bundlePackageName に指定された名前が、特定の 1 つだけの GIM バンドルの名前と一致する場合。
   - この GIM サーバーとともに動作する GIM クライアントが存在するデータベース・サーバーに、bundlePackageName と名前が一致する GIM バンドルがインストールされていない場合。

   例:

   ```
   gim_remove_bundle bundlePackageName=name
   ```

   ここで、name は gim_list_unused_bundles コマンドによって返されたバンドル名です。

タスクの結果

必要のない GIM バンドルが GIM サーバーから削除されます。
親トピック: Guardium Installation Manager

GIM 診断の実行

GIM サーバーが、各 GIM クライアントについて正確なデータを持っているかどうかを確認するために、GIM クライアント上で診断を実行することができます。

このタスクについて

GIM クライアントで問題が発生した場合、最初のステップとして、GIM サーバーがそのクライアントについて正確なデータを持っていることを確認する必要があります。GIM 診断を実行すると、GIM サーバー上でそのクライアントについてリストされたモジュールが、そのクライアントにインストールされているモジュールと一致するかどうか、および GIM クライアントに保管されているパラメーターが、GIM サーバーに保管されているパラメーターと一致するかどうかが検査されます。

GIM 診断は、Guardium ユーザー・インターフェースまたはコマンド行のいずれかから実行できます。コマンド行から実行する場合は、次のコマンドを使用します。

```
grdapi gim_run_diagnostics clientIP=xx.xx.xx.xx
```

clientIP の値には、IP アドレスまたはホスト名を指定できます。このコマンドは、このクライアントの GIM サーバーである Guardium システムで実行する必要があります。

GIM 診断を GUI から実行する場合は、次の手順を使用します。

手順

1. 各クライアントの横にあるチェック・ボックスを使用して、GIM 診断の実行対象とするクライアントを選択します。
2. 「診断の実行」をクリックします。 各クライアントは、次回に更新について GIM サーバーをポーリングするときに、診断コマンドを受け取り、コマンドを直ちに実行します。

タスクの結果

この結果を GIM_EVENT レポートで確認することができます。
親トピック: Guardium Installation Manager

GIM 動作のデバッグ

問題をトラブルシューティングするためにデバッグをオンにすることが必要な場合があります。

このタスクについて

以下のステップを使用して、GIM サーバーで GIM デバッグをオンにします (Guardium システム)。gimserver.log4j.properties を変更するには、Guardium アプライアンスに root ログインする必要があります。必要に応じて、Guardium 技術サポートにお問い合わせください。

手順

1. GIM プロパティー・ファイル /opt/IBM/Guardium/tomcat/gimserver/ROOT/WEB-INF/conf/gimserver.log4j.properties を編集します。
2. 値 ERROR を DEBUG に変更します。
3. ファイルを保存します。

タスクの結果

デバッグは数秒後にオンになり、デバッグ・メッセージは /var/log/guard/debug-logs/ 内の日次デバッグ・ログ・ファイルに書き込まれます。

次のタスク

デバッグが終了したら、ファイルを再び編集して DEBUG を ERROR に戻します。

親トピック: Guardium Installation Manager

GIM クライアントのデバッグの有効化

このタスクについて

GIM クライアントでデバッグを有効にするには、パラメーター module_DEBUG を 1 に変更します。ここで、module は、操作のデバッグ対象となるインストール済みモジュールの名前です。CLI またはユーザー・インターフェースを使用することにより、パラメーターを変更できます。デバッグの完了時に値を 0 に設定します。

SMF サポートを備えた Solaris 用の監視プログラムの再始動

一連の CLI コマンドを使用して、SMF サポートを備えた Solaris サーバーで監視プログラムを再始動します。

このタスクについて

監視プログラムを再始動するには、以下の手順を実行します。この手順は、SMF サポートを備えた Solaris サーバー上でのみ使用してください。

手順

1. コマンド svcadm -v disable guard_gsvr を実行して、監視プログラムを停止します。
2. svccfg delete -f guard_gsvr コマンドを実行します。
3. コマンド svccfg import <gim install dir>/SUPERVISOR/current/guard_gsvr.xml を使用して監視プログラムを再始動します。ここで、<gim install dir> は、GIM インストール・ディレクトリーへのファイル・パスです。

タスクの結果

SMF サポートを備えた Solaris で監視プログラムが再始動されました。
親トピック: Guardium Installation Manager

Guardium システムのインストール

この資料では、IBM Security Guardiumシステムをインストールして構成するのに必要なステップについて詳しく説明します。

また、この資料では、アプライアンスでパーティショニングをカスタマイズする方法や、リモート・ドライブ (SAN) にインストールする方法についても説明します。

具体的なステップは以下のとおりです。

1. 作業を始める前に、必要な構成情報とハードウェアを集めます。
2. 物理アプライアンスまたは仮想アプライアンスを設定します。
3. Guardium® イメージをインストールします。
4. 初期構成と基本構成を設定します。
5. インストールが成功したかどうか検証します。

IBM Security Guardium ソリューションは、以下の形態で提供されます。

- ハードウェア・オファリング – IBM® 提供の物理アプライアンスに組み込んで提供される、完全に構成されたソフトウェア・ソリューション。
- ソフトウェア・オファリング - ユーザーが所有するハードウェアに直接デプロイする、または仮想アプライアンスとしてデプロイするソフトウェア・イメージとして提供されるソリューション。

この資料に記載する要件は、特に指定のないかぎり、物理アプライアンスと仮想アプライアンスの両方のインストールに適用されます。

- 動作モード
  Guardium システムは、複数の動作モードのうち、任意のモードでデプロイすることができます。
- ライセンス・キー
  Guardium システムを機能する状態にするには、基本ライセンスと 1 つ以上の追加ライセンスの両方が必要になります。
- ハードウェア要件
  詳細なハードウェア要件およびサイジングの推奨事項は、IBM サポート・ポータルより入手できます。
- Guardium のポート要件
  各 Guardium システムには、何種類かの通信を行うためのポートが必要です。以下の表に、これらの通信を行うための接続と、その接続に割り当てられているデフォルトのポート番号を示します。
- ステップ 1. 始める前の準備
  Guardium システムのデプロイメントを準備するために、ネットワーク管理者は以下の情報を提供する必要があります。
- ステップ 2. 物理アプライアンスまたは仮想アプライアンスの設定
  このセクションで示すセットアップ手順は、物理アプライアンスでインストールする場合と、仮想アプライアンスでインストールする場合で異なります。
- ステップ 3. Guardium イメージのインストール
  このセクションでは、イメージのインストールおよびディスクのパーティション化を行う方法について説明します。

- ステップ 4. 初期構成および基本構成の設定
  最初のステップとしてネットワーク構成を行います。これは、シリアル・ポートまたはシステム・コンソールからアクセス可能なコマンド行インターフェース (CLI) を使用してローカルに行う必要があります。
- ステップ 5. 次の作業
  このセクションでは、インストールの検証、ライセンス・キーのインストール、および入手可能な保守パッチのインストールの各ステップについて詳しく説明します。
- 仮想イメージの作成
  仮想イメージをインストールする場合は、このセクションを参照してください。
- カスタム・パーティション
  ハード・ディスクのパーティションをカスタマイズする場合は、いくつかの選択を行う必要があります。
- 暗号化された LVM によるパーティション化の方法
  暗号化されたディスクを使用する場合は、以下の手順を実行して、論理ボリューム / と /var を含む暗号化された LVM ボリュームを作成します。
- SAN 構成の例
  この付録では、ハード・ディスクの事前パーティション化 (SAN をインストールする場合に必要) を行うために、コマンド・プロンプトに移動して実行する手順について説明します。

動作モード

Guardium システムは、複数の動作モードのうち、任意のモードでデプロイすることができます。

Guardium の環境を計画する際に、システムを以下のいずれかの動作モードでデプロイすることも、すべての動作モードでデプロイすることもできます。

コレクター
: コレクターは、データベース・サーバーとファイル・サーバー上にデプロイされているエージェントから、データベースのアクティビティーまたはファイルのアクティビティーに関するデータを受信します。 コレクターは、コレクターにインストールされているポリシーに従い、受信したデータを処理して応答します。コレクターは、アグリゲーターにデータをエクスポートすることができます。

アグリゲーター
: アグリゲーターは複数のコレクターからデータを収集し、そのデータの集約ビューを提供します。アグリゲーターは、データベース・サーバーやファイル・サーバーに直接接続されることはありません。場所や機能に応じて、コレクターをアグリゲーターに割り振ることができます。例えば、人事関連リソースのデータベース・サーバーをモニターするコレクターを単一のアグリゲーターに接続すると、それらすべてのサーバーに関連するデータを 1 カ所で表示することができます。必要な場合は、コレクターからではなく、他のすべてのアグリゲーターからデータを収集するアグリゲーターをデプロイすることにより、第 2 の集約層を実装することもできます。
  注: 中央マネージャーとしてアプライアンスを使用する予定の場合、アグリゲーター・オプションを選択する必要があります。

中央マネージャー
: Guardium 環境には中央マネージャーが 1 つしか存在しませんが、バックアップ用の中央マネージャーとして別の Guardium システムを指定することができます。中央マネージャーを使用することにより、単一コンソールからポリシーを定義してそのポリシーをすべてのコレクターに配布したり、すべての Guardium システムに影響する他の構成タスクを実行したり、他のさまざまな管理タスクを実行したりすることができます。 中央マネージャーをアグリゲーターとして動作させて、データをコレクターや他のアグリゲーターから収集することもできます。この場合、全社規模でのアクティビティーのビューを表示したり、すべての Guardium システムから集約されたデータに基づくレポートを表示したりすることができます。

コレクターに割り当てるモニター対象のデータベース・サーバーとファイル・サーバーの数は、サーバーからコレクターに流れるデータ量によって異なります。現在の環境で必要なコレクターとアグリゲーターの数に関する情報と、最良の結果を得るための Guardium システムの配置方法については、「Deployment Guide for IBM Guardium」を参照してください。

Guardium 脆弱性評価コンポーネントを使用する場合は、評価テストを実行する場所を決める必要があります。 一部のユーザーは、この機能のために専用の Guardium システムを個別に設定しています。コレクター、アグリゲーター、または中央マネージャーとしてデプロイされた任意の Guardium システムからテストを実行することもできます。

親トピック: Guardium システムのインストール

ライセンス・キー

Guardium システムを機能する状態にするには、基本ライセンスと 1 つ以上の追加ライセンスの両方が必要になります。

基本ライセンスと追加ライセンスの概要を以下に示します。

- 基本ライセンス・キー (リセット・キーとも呼ばれます) は、システムのマシン・タイプを反映します。例えば、コレクター・システムを設定する場合は、コレクター用の基本ライセンスが必要になります。
- 追加ライセンス・キーを使用すると、特定の機能セットが有効になります。例えば、通常のデータ・アクティビティー・モニター機能を使用する場合は、DAM Standard 追加ライセンスが必要になります。複数の追加ライセンスを組み合わせてインストールすると、Guardium の拡張機能を使用できるようになります。

基本ライセンスを適用するとマシン・タイプがチェックされ、互換性があるかどうかが確認されます。基本ライセンスには、以下の 2 つのタイプがあります。

表 1. 基本ライセンスのタイプ

基本ライセンスのタイプ	ライセンスの説明
コレクター	スタンドアロン・システムやコレクターを設定する場合は、コレクター用の基本ライセンスを使用します。
アグリゲーター	アグリゲーターや中央マネージャー・システムを設定する場合は、アグリゲーター用の基本ライセンスを使用します。

Guardium システムで使用できる機能は、インストールされている追加ライセンスによって異なります。有効な追加ライセンスを以下に示します。これらのライセンスは、組み合わせて使用することができます。

表 2. 追加ライセンスのタイプ

追加ライセンスのタイプ	ライセンスの説明
DAM Standard	データ・アクティビティー・モニターの主要機能を使用するためのライセンス。

追加ライセンスのタイプ	ライセンスの説明
DAM Advanced	DAM Standard 機能、詳細なアクセス制御機能、マスキング機能、隔離機能、ブロッキング機能 (アクティビティーの強制終了機能) を使用するためのライセンス。
FAM Standard	ファイル・アクティビティー・モニターの主要機能を使用するためのライセンス。
FAM Advanced	FAM Standard 機能とブロッキング機能を使用するためのライセンス。
VA Standard	脆弱性評価機能、データベース保護サービス (DPS)、変更監査システム (CAS)、データベース・ライセンス・レポートを使用するためのライセンス。

Guardium ライセンスのインストールについては、ライセンス・キーのインストールを参照してください。

親トピック: Guardium システムのインストール
関連タスク:
ライセンス・キーのインストール

ハードウェア要件

詳細なハードウェア要件およびサイジングの推奨事項は、IBM サポート・ポータルより入手できます。

詳細なハードウェア仕様およびサイジングの推奨事項については、IBM Guardium V10.1 Software Appliance Technical Requirements を参照してください。

親トピック: Guardium システムのインストール

Guardium のポート要件

各 Guardium システムには、何種類かの通信を行うためのポートが必要です。以下の表に、これらの通信を行うための接続と、その接続に割り当てられているデフォルトのポート番号を示します。

オープン・ポート

Guardium システムで使用されるポート。

DB サーバー – コレクター

TCP 8443 - DB サーバーからコレクター

TCP 16016 – Unix STAP、両方向、登録、ハートビート、およびデータ (PASE で稼働中の IBM i S-TAP を含む)

TCP 16017 - Windows/Unix CAS、両方向、テンプレートおよびデータ

TCP 16018 – Unix STAP (TLS)、両方向、登録、ハートビート、およびデータ

TCP 16019 - Windows/Unix CAS (TLS)、両方向、テンプレートおよびデータ

TCP 16020 - UNIX STAP 接続プーリングから

TLS 16021 - STAP エージェントの暗号化された UNIX STAP 接続プーリングから

TCP 8081 – Guardium Installation Manager、両方向、データベース・サーバーからコレクター/中央マネージャー

TCP 9500 – Windows STAP、両方向、DB サーバーからコレクター、STAP 登録およびデータ

TCP 9501 – Windows STAP (TLS)、両方向、DB サーバーからコレクター、STAP 登録およびデータ

コレクター – アグリゲーター (セキュア・シェル – SSL)

TCP 22 – コレクターからアグリゲーター、SCP データ・エクスポート、両方向

中央マネージャー – 管理対象デバイス

TCP 22 – SSH/SCP データ転送、両方向

TCP 8443 – SSL、両方向

TCP 8444 – SSL、STAP から GIM へのファイル・アップロード

TCP 3306 – MySQL、特定ソースに対してオープン (例えば、中央マネージャーはすべての管理対象ユニットに対してオープンであり、管理対象ユニットは中央マネージャーに対してオープンです)

TLS 8447 - フェデレーテッド環境または一元管理環境内の Guardium システム間の通信用に、リモート・メッセージング・サービス・インフラストラクチャー (およびプロファイル配布インフラストラクチャー) で使用されます。構成プロファイルを使用すると、中央マネージャーから構成とスケジューリングの設定を定義して、中央マネージャー自体の構成を変更することなく、これらの設定を管理対象ユニット・グループに容易に配布できます。

ファイル・アクティビティー・モニター (FAM)

TCP/TLS 16022/16023 - 汎用フィード。16022 (FAM モニター、非暗号化) と 16023 (FAM モニター、暗号化) の両方が双方向にオープンである必要があります。スニファーでは、16016 から 16023 までのブロックが双方向にオープンである必要があります

18087 - FAM がインストールされているのと同一マシン上にある IBM Content Classification (ICM) サーバー上の FAM のリスナー・ポート (serverSettings.icmURL=http://localhost:18087)。双方向にオープン。

Guardium Installation Manager (GIM)

8445 - GIM クライアント・リスナー、両方向。GIM クライアントが listen を行っています。中央マネージャーまたはコレクターいずれかの GIM サーバーが、それ (GIM クライアント) に到達できます。

8446 - GIM 認証 TLS、両方向。GIM クライアントと GIM サーバー (中央マネージャーまたはコレクター上) の間で使用します。GIM_USE_SSL が無効に設定されていない場合、gim_client はポート 8446 を介してその証明書に通信しようと試みます。ポート 8446 がオープンされていない場合、デフォルトは 8444 ですが、証明書は渡されません (証明書の検証なしの TLS など)。

8081 - TLS - GIM クライアントを GIM サーバーに接続するために 8081 を使用する場合、GIM_USE_SSL パラメーターを無効にする必要があります。デフォルトではオンになっています。このパラメーターは、GUI の GIM 共通パラメーターに含まれています。 GIM_USE_SSL が無効に設定されていない場合、gim_client はポート 8446 を介してその証明書に通信しようと試みます。ポート 8446 がオープンされていない場合、デフォルトは 8444 ですが、証明書は渡されません (証明書の検証なしの TLS など)。

エンタープライズ・ロード・バランサー

TLS 8443 - S-TAP ロード・バランサー - これは、UNIX/Linux S-TAP でインスタンスをコレクターに通信させる際に必要です。ただし、このポートは中央マネージャーのロード・バランサーにも使用されます。 インストール済み環境でエンタープライズ・ロード・バランサーを使用することが示されると、S-TAP は HTTPS メッセージを送信することにより、8443 で中央マネージャー (ロード・バランサー) に対する要求を開始します。お客様がデータベースから中央マネージャーへの直接的なオープン・ポートを希望しない場合、データベース・サーバーと中央マネージャー間にプロキシー・サーバーを使用する機能が存在します。

Quick Search for Enterprise

TCP 8983 - SOLR - 着信、SSL

TCP 9983 - SOLR - 着信、SSL

ユーザー・インターフェース - Guardium システム (スタンドアロン、アグリゲーター、中央マネージャー)

TCP 22 – ユーザーからシステム、CLI 接続、両方向

TCP 8443 - ユーザーからシステム、GUI 接続 (構成可能)、両方向

システム - SMTP サーバー

TCP 25 - システムから SMTP サーバー、E メール・アラート

システム - SNMP サーバー

UDP 161 - SNMP クライアントからシステム - SNMP ポーリング

UDP 162 - システムから SNMP サーバー、SNMP トラップ

システム - SYSLOG サーバー

UDP/TCP 514 –他のシステムとの送受信リモート syslog メッセージ、通常は SIEM。**注:** ローカル・ポートは 514 ですが、リモート・ポートを構成に入力する必要があります。暗号化を使用する場合、プロトコルは UDP ではなく、TCP でなければなりません。

システム – NTP サーバー

TCP/UDP 123 – システムから Network Time Protocol サーバー

システム - DNS サーバー

TCP/UDP 53 - システムからドメイン・ネーム・サーバー

システム – EMC Centera (バックアップ)

TCP 3218 - システムから EMC Centera

システム - Tivoli LDAP

UDP 389 –システムと Tivoli LDAP 間

システム - メインフレーム

TCP 16022 - S-TAP を Db2 z/OS、S-TAP IMS、S-TAP VSAM (S-TAP データ・セット) に接続

TCP 16023 - TLS 接続、具体的には IBM の Application Transparent Transport Layer Security (AT-TLS)

Windows データベース・サーバーに接続するためのポート

ポート	プロトコル	目的
8075	UDP	Windows S-TAP ハートビート・シグナル (両方向トラフィック)。注: UNIX S-TAP エージェントは、ハートビート・シグナルで UDP を使用しないため、この機能に対応する UNIX ポートはありません。
9500	TCP	クリアな Windows S-TAP
9501	TLS	暗号化された Windows S-TAP (オプション)
16017	TCP	クリアな Windows CAS

16019	TLS	暗号化された Windows CAS (オプション)

Guardium アプリケーションへのアクセスに使用されるデフォルト・ポート

ポート	プロトコル	目的
8443	TCP	Guardium のユーザー・インターフェースに対する Web ブラウザー・アクセス (HTTPS)。注: Guardium の管理者は、このポートを変更することができます。また、このポートを使用して、管理対象ユニットが中央マネージャーに登録されます。
22	TCP	Guardium アプライアンスを管理するための、クライアントからの SSH アクセス
3306	TCP	中央マネージャーと管理対象ユニット間の通信

z/OS データベース・サーバーに接続するためのポート

ポート	プロトコル	目的
16022	TCP	S-TAP for Db2 z/OS、S-TAP for IMS、S-TAP for Data Sets への接続
16023	TCP	TLS 接続、具体的には IBM の Application Transport Layer Security (AT-TLS)
41500	TCP	内部メッセージ・ロギング通信用のデフォルトの開始ポート – LOG_PORT_SCAN_START
39987	TCP	エージェントとエージェントの 2 次アドレス・スペース間のデフォルトのエージェント固有通信ポート – ADS_LISTENER_PORT

他の機能で使用されるデフォルト・ポート

ポート	プロトコル	目的
20, 21	TCP	バックアップ/アーカイブのための FTP サーバー (オプション)
22	TCP	バックアップ/アーカイブ、パッチの配布、ファイル転送のための SCP
25	TCP	アラートおよびその他の通知のための SMTP (E メール・サーバー)
53	TCP	DNS Servers
123	TCP、UDP	時刻の同期のための NTP (タイム・サーバー)
161	TCP、UDP	SNMP ポーリング (オプション)
162	TCP、UDP	SNMP トラップ (オプション)
389	TCP	LDAP (Active Directory や Sun One Directory など)
514	TCP	Syslog サーバー (オプション)

636	TCP	LDAP (SSL 経由の Active Directory や Sun One Directory など) (オプション)
1500	TCP	Tivoli Storage Manager のバックアップ・ホスト (オプション)
3218	TCP、UDP	EMC 中央バックアップ・ホスト (オプション)
ユーザー定義	TCP	Guardium データ・ソース・アクセス用のデータベース・サーバーのリスナー・ポート (例えば、Oracle の場合は 1521、MS-SQL の場合は 1433) (オプション)。S-TAP 検査およびディスカバリーにこのポートを使用します。
16022/16023	TCP/TLS	汎用フィード - ファイル・アクティビティー・モニター (FAM)
18027		IBM Content Classification をローカルに使用した FAM (serverSettings.icmURL=http://localhost:18087)
8445		GIM クライアント・リスナー、両方向 GIM クライアントが listen を行っています。中央マネージャーまたはコレクターいずれかの GIM サーバーが、それ (GIM クライアント) に到達できます。
8446	TLS	GIM 認証 TLS、両方向 GIM クライアントと GIM サーバー (中央マネージャーまたはコレクター上) の間で使用します。 GIM_USE_SSL が無効に設定されていない場合、gim_client はポート 8446 を介してその証明書に通信しようと試みます。ポート 8446 がオープンされていない場合、デフォルトは 8444 ですが、証明書は渡されません (証明書の検証なしの TLS など)。
8447	TLS	フェデレーテッド環境または一元管理環境内の Guardium システム間の通信用に、リモート・メッセージング・サービス・インフラストラクチャー (およびプロファイル配布インフラストラクチャー) で使用されます。構成プロファイルを使用すると、中央マネージャーから構成とスケジューリングの設定を定義して、中央マネージャー自体の構成を変更することなく、これらの設定を管理対象ユニット・グループに容易に配布できます。
8443	TLS	エンタープライズ・ロード・バランサー これは UNIX/Linux S-TAP でインスタンスをコレクターに通信させる際に必要です。 ただし、このポートは中央マネージャーのロード・バランサーにも使用されます。 インストール環境でエンタープライズ・ロード・バランサーを使用する場合、S-TAP は HTTPS メッセージを送信することにより ポート 8443 上で中央マネージャーへの要求を開始します。 そのため、お客様がデータベースから中央マネージャーへの直接的なオープン・ポートを希望しない場合、データベース・サーバーと中央マネージャー間にプロキシー・サーバーを使用する機能が存在します。
8081	TLS	GIM クライアントを GIM サーバーに接続するために 8081 を使用する場合、GIM_USE_SSL パラメーターを無効にする必要があります。デフォルトではオンになっています。 このパラメーターは、GUI の GIM 共通パラメーターに含まれています。 GIM_USE_SSL が無効に設定されていない場合、gim_client はポート 8446 を介してその証明書に通信しようと試みます。ポート 8446 がオープンされていない場合、デフォルトは 8444 ですが、証明書は渡されません (証明書の検証なしの TLS など)。
8983	TCP	SOLR、着信、SSL (Quick Search for Enterprise)
9983	TCP	SOLR、着信、SSL (Quick Search for Enterprise)

親トピック: Guardium システムのインストール

ステップ 1. 始める前の準備

Guardium システムのデプロイメントを準備するために、ネットワーク管理者は以下の情報を提供する必要があります。

- インターフェース・カード (eth0) の IP アドレス。
- 1 次 IP アドレスのサブネット・マスク。
- デフォルトのルーター IP アドレス。
- システムに割り当てるホスト名およびドメイン名。
- DNS サーバーの IP アドレス (最大 3 つのアドレス)、および DNS ドメインへの新規 Guardium システムの追加。
- (オプション) 2 次管理インターフェースの IP アドレス。
- (オプション) 2 次 IP 管理インターフェースのマスク。

- (オプション) 2 次 IP 管理インターフェースのゲートウェイ。
- (オプション) NTP サーバーのホスト名。
- (オプション) SMTP 構成情報 (E メール・アラート用): IP アドレス、ポート、 および (認証使用の場合に) SMTP ユーザー名とパスワード。
- (オプション) SNMP 構成情報 (SNMP アラート用): SNMP サーバーの IP アドレスと使用するトラップ・コミュニティー名。

- SAN ストレージ・デバイス
  インストール処理をストレージ・エリア・ネットワーク (SAN) 上でデプロイする場合は、デプロイメントを実行する前に、SAN で必要な構成情報をすべて準備しておく必要があります。また、SAN ストレージ・デバイスをパーティション化し、Guardium OS をインストールするための追加のインストール・ステップを実行する必要があります。

親トピック: Guardium システムのインストール

SAN ストレージ・デバイス

インストール処理をストレージ・エリア・ネットワーク (SAN) 上でデプロイする場合は、デプロイメントを実行する前に、SAN で必要な構成情報をすべて準備しておく必要があります。また、SAN ストレージ・デバイスをパーティション化し、Guardium OS をインストールするための追加のインストール・ステップを実行する必要があります。

注: SAN へのインストールはサポートされていますが、NAS へのインストールはサポートされていません。
親トピック: ステップ 1. 始める前の準備

ステップ 2. 物理アプライアンスまたは仮想アプライアンスの設定

このセクションで示すセットアップ手順は、物理アプライアンスでインストールする場合と、仮想アプライアンスでインストールする場合で異なります。

- 物理アプライアンス
  ラックにアプライアンスを設置したら、次の方法でアプライアンスをネットワークに接続します。
- eth0 とその他のネットワーク・ポートの識別方法
  次の CLI コマンドを使用して、ネットワーク・ポートをマップします。
- 物理アプライアンスのデフォルト・パスワード
  定義済みユーザーには、デフォルトのパスワードが設定されています。
- 仮想アプライアンス
  IBM Security Guardium 仮想マシン (VM) は、ゲスト仮想マシン (VMware ESX Server など) でライセンス交付およびインストールを行う、ソフトウェア専用ソリューションです。

親トピック: Guardium システムのインストール

物理アプライアンス

ラックにアプライアンスを設置したら、次の方法でアプライアンスをネットワークに接続します。

1. 電源接続部を見つけます。適切な電源コードを、これらの接続部に接続します。
2. ネットワーク・ケーブルを eth0 ネットワーク・ポートに接続します。必要に応じて、オプションの 2 次ネットワーク・ケーブルを接続します。
3. キーボード、ビデオ、マウスを、直接または KVM 接続 (シリアル・ポートまたは USB ポート) 経由でシステムに接続します。
4. システムの電源を入れます。

親トピック: ステップ 2. 物理アプライアンスまたは仮想アプライアンスの設定
関連情報:
Lenovo System x3550 M5 Installation and Service Guide
技術要件に関する資料
eth0 管理ポートの変更点

eth0 とその他のネットワーク・ポートの識別方法

次の CLI コマンドを使用して、ネットワーク・ポートをマップします。

show network interface inventory

この CLI コマンドを使用して、インストールされているすべてのネットワーク・インターフェースのポート名および MAC アドレスを表示します。

```
show network interface inventory
eth0 00:13:72:50:CF:40
eth1 00:13:72:50:CF:41
eth2 00:04:23:CB:11:84
eth3 00:04:23:CB:11:85
eth4 00:04:23:CB:11:96
eth5 00:04:23:CB:11:97
```

show network interface port

この CLI コマンドを使用して、アプライアンスの背面にある物理コネクターを見つけます。show network interface inventory コマンドを使用してすべてのポート名を表示したら、以下のコマンドを使用して、「n」で指定される物理ポートのランプを 20 回明滅させます (「n」は、eth0、eth1、eth2、eth3 などのように、*eth* の後に続く数字です)。

```
show network interface port 1
```

ポート eth1 のランプが 20 回明滅します。

専用コンピューターにソフトウェアを直接インストールする

Guardium ソフトウェアを専用コンピューターのディスクに直接インストールする場合は、物理アプライアンスの手順を実行します。

親トピック: ステップ 2. 物理アプライアンスまたは仮想アプライアンスの設定

物理アプライアンスのデフォルト・パスワード

定義済みユーザーには、デフォルトのパスワードが設定されています。

IBM から物理アプライアンスを受け取ったら、以下のパスワードを使用して初期構成を行ってください。
注: インストールが完了したら、すべてのデフォルト・パスワードを必ず変更してください。

表 1. 定義済みユーザーのデフォルト・パスワード

ユーザー	デフォルトのパスワード
accessmgr	guard1accessmgr
admin	guard1admin
cli	guard1cli

親トピック: ステップ 2. 物理アプライアンスまたは仮想アプライアンスの設定

仮想アプライアンス

IBM Security Guardium 仮想マシン (VM) は、ゲスト仮想マシン (VMware ESX Server など) でライセンス交付およびインストールを行う、ソフトウェア専用ソリューションです。

Guardium VM をインストールするには、『仮想イメージの作成』に記載されているステップに従います。具体的なステップは以下のとおりです。

- システム互換性の検証
- VMware ESX Server のインストール
- ネットワーク・ケーブルの接続
- VM 管理ポータルの構成
- 新規仮想マシンの作成
- IBM Security Guardium 仮想アプライアンスのインストール

VM をインストールした後に、『ステップ 4. 初期構成および基本構成の設定』に戻って Guardium システムの構成方法に関する詳しい説明を参照してください。

親トピック: ステップ 2. 物理アプライアンスまたは仮想アプライアンスの設定

ステップ 3. Guardium® イメージのインストール

このセクションでは、イメージのインストールおよびディスクのパーティション化を行う方法について説明します。

1. UEFI/BIOS の「ブート・シーケンス」が、ハード・ディスクを使用する前に取り外し可能メディア (CD/DVD ドライブ) から始動するように設定されていることを確認します。
   注: インストールは DVD から実行できます。必要に応じて、技術サポートから UEFI/BIOS のパスワードを入手してください。
2. インストール DVD から Guardium イメージをロードします。
3. 次の 2 つのオプションが表示されます。

   標準インストール: これはデフォルトです。ディスクをパーティション化している場合は、ほとんどの場合この項目を使用します。

   カスタム・パーティション・インストール: すべてのパーティションを (ローカルに、または SAN ディスク上で) 詳細にカスタマイズできます。このオプションを実装する方法について詳しくは、『カスタム・パーティショニング』を参照してください。

   注:
   - 標準インストールでは、ディスク内容がすべて消去され、ディスクの再パーティション化と再フォーマットが行われ、新規オペレーティング・システムがインストールされます。
   - インストール後の最初のブート時に、ご使用条件への同意を求められます。Page Down キーでご使用条件を確認するか、Q キーで末尾にスキップします。ご使用条件に同意するには、`q` を入力して終了し、`yes` と入力します。`yes` と入力してご使用条件に同意する必要があります。同意しないとマシンはブートされません。
4. システムは DVD からブートされます。このインストールには約 12 分かかります。

   (d) インストール・プロセスで、コレクターまたはアグリゲーターを選択するよう求められます (10 秒間入力されなかった場合、自動的に「コレクター」に設定されます)。コレクターとアグリゲーターの説明については、製品概要を参照してください。 アグリゲーターを選択しようとしたが 10 秒以内に選択しなかった場合は、アグリゲーターを選択できるこの時点まで戻るために、再インストールを実行する必要があります。

   注: 中央マネージャーとしてアプライアンスを使用する予定の場合、アグリゲーター・オプションを選択する必要があります。
5. この時点でシステムが自動的にリブートされ、インストールが完了します。リブート後の初回ログイン時に、パスワードを変更する必要があります。

親トピック: Guardium システムのインストール

ステップ 4. 初期構成および基本構成の設定

最初のステップとしてネットワーク構成を行います。これは、シリアル・ポートまたはシステム・コンソールからアクセス可能なコマンド行インターフェース (CLI) を使用してローカルに行う必要があります。

以降のステップでは、CLI コマンドを使用して各種ネットワーク・パラメーターを指定して、Guardium システムをご使用の環境に統合します。

CLI 構文では、変数が不等号括弧で示されます (<ip_address> など)。

各変数を、ご使用のネットワークおよびインストール済み環境に適した値で置き換えます。括弧は含めないでください。

- 1 次システムの IP アドレスの設定
  1 次 IP アドレスは eth0 接続用で、以下の 2 つのコマンドを使用して定義されます。
- デフォルト・ルーター IP アドレスの設定
  次の CLI コマンドを使用します。
- DNS サーバーの IP アドレスの設定
  1 つ以上の DNS サーバーの IP アドレスを設定します。これは、アプライアンスがホスト名と IP アドレスの解決に使用します。 最初のリゾルバーは必須で、他はオプションです。
- SMTP サーバー
  システム・アラートを送信するには、SMTP サーバーが必要です。次のコマンドを入力して、ご使用の SMTP サーバー IP アドレスの設定、メッセージのリターン・アドレスの設定、および始動時の SMTP アラートの有効化を行います。
- ホスト名とドメイン・ネームの設定
  アプライアンスのホスト名とドメイン・ネームを構成します。 この名前は、DNS サーバーに登録されたアプライアンスのホスト名と一致する必要があります。
- タイム・ゾーンおよび日時の設定
  アプライアンスの日時を設定するには、以下のオプションがあります。
- 初期ユニット・タイプの設定
  アプライアンスは、スタンドアロン・ユニット、マネージャー・ユニット、または管理対象ユニットとして設定できます。また、アプライアンスがネットワーク検査または S-TAP、またはその両方を介してデータベース・アクティビティーをキャプチャーするように設定できます。標準的な構成はスタンドアロン・アプライアンス用 (すべてのアプライアンス用) で、最も一般的な設定では S-TAP を使用してキャプチャーを行います (コレクター専用)。
- root パスワードのリセット
  次の CLI コマンドを実行することで、独自の専用パス・キーを使用してアプライアンスの root パスワードをリセットします (アクセス・キーが必要: 「*t0Tach*」)。
- すべての設定の検証
  CLI からログアウトして次の構成ステップに進む前に、以下のコマンドを使用して、構成した設定をレビューして検証します。
- システムのリブート
  システムが最終ロケーションにない場合は、ここでシステムをシャットダウンし、最終的なネットワーク・ロケーションに配置して、再始動します。

親トピック: Guardium システムのインストール

1 次システムの IP アドレスの設定

1 次 IP アドレスは eth0 接続用で、以下の 2 つのコマンドを使用して定義されます。

```
store network interface ip <ip_address>
store network interface mask <subnet_mask>
```

デフォルトのネットワーク・インターフェース・マスクは 255.255.255.0 です。これがご使用のネットワークでの正しいマスクである場合は、2 番目のコマンドをスキップできます。

2 次 IP アドレスを割り当てるには、CLI コマンド store network interface secondary [on <interface> <ip> <mask> <gw> | off] を使用します。このコマンドを使用して、2 次インターフェースを有効/無効にすることができます。

次に、CLI コマンド restart network を使用してネットワークを再始動する必要があります。2 次 IP アドレスの割り当ては、CLI からのみ実行でき、GUI を使用して実行することはできません。

アプライアンス上の他のネットワーク・インターフェース・カードは、データベース・トラフィックのモニターに使用でき、IP アドレスは割り当てられません。

親トピック: ステップ 4. 初期構成および基本構成の設定

デフォルト・ルーター IP アドレスの設定

次の CLI コマンドを使用します。

```
store network routes defaultroute <default_router_ip>
```

親トピック: ステップ 4. 初期構成および基本構成の設定

DNS サーバーの IP アドレスの設定

1 つ以上の DNS サーバーの IP アドレスを設定します。これは、アプライアンスがホスト名と IP アドレスの解決に使用します。 最初のリゾルバーは必須で、他はオプションです。

```
store network resolver 1 <resolver_1_ip>
store network resolver 2 <resolver_2_ip>
store network resolver 3 <resolver_3_ip>
```

親トピック: ステップ 4. 初期構成および基本構成の設定

SMTP サーバー

システム・アラートを送信するには、SMTP サーバーが必要です。次のコマンドを入力して、ご使用の SMTP サーバー IP アドレスの設定、メッセージのリターン・アドレスの設定、および始動時の SMTP アラートの有効化を行います。

```
store alerter smtp relay <smtp_server_ip>
store alerter smtp returnaddr <first.last@company.com>
store alerter state startup on
```

注: SMTP サーバーはユーザー・インターフェースで構成することもできます。「設定」 > 「アラート機能」をクリックします。
親トピック: ステップ 4. 初期構成および基本構成の設定

ホスト名とドメイン・ネームの設定

アプライアンスのホスト名とドメイン・ネームを構成します。 この名前は、DNS サーバーに登録されたアプライアンスのホスト名と一致する必要があります。

```
store system hostname <host_name>
store system domain <domain_name>
```

親トピック: ステップ 4. 初期構成および基本構成の設定

タイム・ゾーンおよび日時の設定

アプライアンスの日時を設定するには、以下のオプションがあります。

タイム・ゾーン、日付、時刻、および NTP

1. タイム・ゾーンの設定
2. 日付と時刻の設定。オプション 1 - NTP の設定。オプション 2 - store system clock datetime

日付/時刻のオプション 1: Network Time Protocol

アクセス可能な NTP サーバーの詳細を指定して、 これを使用できるようにします。

```
store system ntp server
store system ntp state on
```

日時オプション 2: タイム・ゾーン、日付、および時刻の設定

次のコマンドを使用して、有効なタイム・ゾーンのリストを表示します。

```
store system clock timezone list
```

リストから適切なタイム・ゾーンを選択し、 同じコマンドを使用して設定します。

```
store system clock timezone <selected time zone>
```

注: 新しいタイム・ゾーンを設定すると、内部サービスが再始動し、その再始動中にデータ・モニターが数分無効になります。

日付と時刻を YYYY-mm-dd hh:mm:ss 形式で保存します。

```
store system clock datetime <date_time>
```

注: 同じ CLI セッションでホスト名とタイム・ゾーンを変更しないでください。

親トピック: ステップ 4. 初期構成および基本構成の設定

初期ユニット・タイプの設定

アプライアンスは、スタンドアロン・ユニット、マネージャー・ユニット、または管理対象ユニットとして設定できます。また、アプライアンスがネットワーク検査または S-TAP、またはその両方を介してデータベース・アクティビティーをキャプチャーするように設定できます。標準的な構成はスタンドアロン・アプライアンス用 (すべてのアプライアンス用) で、最も一般的な設定では S-TAP を使用してキャプチャーを行います (コレクター専用)。

store unit type standalone - このコマンドはすべてのアプライアンスに使用します。

store unit type stap - このコマンドはコレクターに使用します。

スタンドアロン・ユニット・タイプおよび STAP ユニット・タイプは、デフォルトで設定されます。マネージャー・ユニット・タイプは指定する必要があります (必要な場合)。

注: ユニット・タイプの設定は、アプライアンスが完全に作動可能になってから、後で行うことができます。
親トピック: ステップ 4. 初期構成および基本構成の設定

root パスワードのリセット

次の CLI コマンドを実行することで、独自の専用パス・キーを使用してアプライアンスの root パスワードをリセットします (アクセス・キーが必要: 「*t0Tach*」)。

```
support reset-password root <random>
```

資料で使用されているパス・キーを保存して、将来的に技術サポートの担当者が root アクセスできるようにします。 現在のパス・キーを確認するには、次の CLI コマンドを使用します。

```
support show passkey root
```

質問 - Guardium システムの root パスワードはどの程度安全ですか? 誰がアクセスできますか?

Guardium アプライアンスは「ブラック・ボックス」環境です。エンド・ユーザーのみがオペレーティング・システム・アカウントへの限定アクセス権を持ちます。以下に例を挙げます。

cli、guardcli1、guardcli2、guardcli3、guardcli4、guardcli5。

グラフィカル・ユーザー・インターフェースのユーザー・アカウント (admin や accessmgr など) は、Guardium システムのオペレーティング・システムによって定義されるのではなく、 アプリケーション・インターフェース (accessmgr) によって定義および管理されるアプリケーション ID です。

セキュアなサーバーであるため、 事前にすべてのユーザーが root アクセスを使用できるようにはなっていませんが、 多くの場合は、問題のトラブルシューティングおよび解決の目的で Guardium アプライアンスにアクセスするために Guardium サポートによって必要とされます。Guardium サポートは、Guardium アプライアンスにアクセスするために sudo を使用せず、また root 以外のいかなるユーザー ID も使用しません。

root のパスワードは、 「結合パスワード」の仕組みを使用してセキュリティーを確保しています。お客様は、8 桁の数値のパス・キーの形でアプライアンスに対するキーを保持します。IBM はパス・キー・デコーダーを保持します。パス・キーとパス・キー・デコーダーの両方がなければ、IBM もお客様も root としてアプライアンスにアクセスすることはできません。

パス・キーは、 お客様が CLI インターフェースで管理します。お客様は、 以下の CLI コマンドを使用して、IBM に通知せずにいつでもパス・キーを変更できます。

```
support reset-password root
```

CLI アクセス権を持つすべてのユーザーは、以下の CLI コマンドを使用して root のパス・キーを取得できます。

```
support show passkey root
```

Guardium サポートと連携するときには、 リモート・デスクトップ共有セッションで、 サポート・アナリストが問題の Guardium アプライアンスに対する root パス・キーを要求します。パス・キーがデコードされると、Guardium サポートが root パスワードを使用して root としてアプライアンスにアクセスします。リモート・デスクトップ共有セッションが終了したら、お客様は、上記の CLI コマンドを使用してパス・キーを変更することで、IBM が今後そのアプライアンスの root パスワードを所有しないようにすることができます。

パス・キーは 8 桁の数値キーであり、 範囲は 10000000 から 99999999 までです。この範囲で 89,999,999 通りのパスワードを作成できます。エンコード後のパスワードはすべて強固です。一般的なパスワードや辞書にある単語は含まず、 長さはそれぞれ異なり、各国語文字、特殊文字、英字 (大文字と小文字)、および/または数字を含んでいます。

パス・キー・デコーダーへのアクセスは、選ばれた少数の IBM Guardium 従業員 (Guardium R&D、Guardium QA、Guardium サポート・スタッフのメンバーなど) に制限されています。IBM スタッフからは使用できません。

上記の CLI ユーザー ID (cli、guardcli1、guardcli2、guardcli3、guardcli4、guardcli5) はパス・キーの仕組みを使用しません。 これらのユーザーのパスワードは完全にお客様によって管理され、IBM がそのパスワードにアクセスすることはできません。このような理由から、root パス・キーをパスワード・ボールトに保管しておき、CLI アカウント・パスワードを忘れたり紛失したりした場合でもアプライアンスにアクセスできるように備えることを IBM は推奨します。

親トピック: ステップ 4. 初期構成および基本構成の設定

すべての設定の検証

CLI からログアウトして次の構成ステップに進む前に、以下のコマンドを使用して、構成した設定をレビューして検証します。

```
show network interface all
show network routes defaultroute
show network resolver all
show system hostname
show system domain
show system clock timezone
show system clock datetime
show system ntp all
show unit type
```

親トピック: ステップ 4. 初期構成および基本構成の設定

システムのリブート

システムが最終ロケーションにない場合は、ここでシステムをシャットダウンし、最終的なネットワーク・ロケーションに配置して、再始動します。

システムをリブートする前に、インストール DVD を取り出してください。

システムを停止するには、CLI で次のコマンドを入力します。

```
stop system
```

システムがシャットダウンします。システムを最終ロケーションに移動し、システムのケーブル接続を修正し、電源を再びオンにします。システムの電源がオンになると、指定した IP アドレスまたはホスト名を使用して、(CLI および GUI から) ネットワーク経由でシステムにアクセスできるようになります。

親トピック: ステップ 4. 初期構成および基本構成の設定

ステップ 5. 次の作業

このセクションでは、インストールの検証、ライセンス・キーのインストール、および入手可能な保守パッチのインストールの各ステップについて詳しく説明します。

- インストールが成功したかどうかの検証
  インストールを検証するには、次のステップを実行します。
- ユニット・タイプの設定
  フェデレーテッド環境を設定するには、いずれかのアプライアンスを中央マネージャーとして構成し、他のすべてのアプライアンスが中央マネージャーによって管理されるように設定します。
- ライセンス・キーのインストール
  このトピックでは、Guardium のライセンス・キーをインストールして使用条件に同意する手順について説明します。

- 保守パッチのインストール (該当する場合)
  CLI または GUI を使用してパッチをインストールすることができます。
- 追加のステップ (オプション)
  次のセクションでは、ベースラインの英語を別の言語に変更する方法について説明します。これを行うには、S-TAP® エージェントをインストールし、検査エンジンを定義し、CAS エージェントをインストールします。

親トピック: Guardium システムのインストール

インストールが成功したかどうかの検証

インストールを検証するには、次のステップを実行します。

1. CLI にログインします。ssh cli@<ip of appliance>
2. GUI にログインします。https://<hostname of appliance>.<full domain>:8443 (管理ユーザー ID を使用)

リブート後の初回ログイン時に、パスワードを変更することが必要です。

Guardium の Web ベースのインターフェースにログインし、組み込みのオンライン・ヘルプで以下のタスクの詳細を参照してください。

親トピック: ステップ 5. 次の作業

ユニット・タイプの設定

フェデレーテッド環境を設定するには、いずれかのアプライアンスを中央マネージャーとして構成し、他のすべてのアプライアンスが中央マネージャーによって管理されるように設定します。

各 Guardium システムのタイプを設定するには、CLI コマンドの store unit type を使用します。

親トピック: ステップ 5. 次の作業

ライセンス・キーのインストール

このトピックでは、Guardium のライセンス・キーをインストールして使用条件に同意する手順について説明します。

始める前に

- パスポート・アドバンテージからライセンス・キーをダウンロードします。
- Guardium システムのインストールまたはアップグレードを行います。
- マシン・タイプがシステムに対して正しく設定されていることを確認します。

このタスクについて

Guardium のライセンス・キーをインストールする場合は、最初にライセンス・キーをインストールしてから、使用条件を読んで同意する必要があります。Guardium のライセンス・キーがインストールされると、ユーザー・インターフェースが再ロードされ、新しいライセンスで使用できる機能が表示されます。

Guardium システムを機能する状態にするには、基本ライセンスと 1 つ以上の追加ライセンスの両方をインストールする必要があります。基本ライセンスをインストールして使用条件に同意してから、追加ライセンスをインストールして使用条件に同意する必要があります。

Guardium のライセンス・キーについては、ライセンス・キーを参照してください。

重要:

Guardium システムをアップグレードする場合は、ライセンスを適用する必要はありません。既存のインストール済み環境に基づいて、ライセンス・キーが自動的に生成されます。ただし、Guardium システムを使用する前に、使用条件を読んで同意する必要があります。アップグレードするシステムのライセンスの使用条件を読んで同意するには、「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートして「ライセンスを読んで同意してください」をクリックします。

手順

1. `admin` ユーザーとして Guardium システムにログインします。
2. Guardium バナーに表示されている「マシン・タイプ」が、ライセンス対象のシステムに対して正しく設定されていることを確認します。 マシン・タイプは、以下のいずれかになります。
   - スタンドアロン
   - 中央マネージャー
   - アグリゲーター

   重要: 「マシン・タイプ」がアグリゲーターに設定されている状態で中央マネージャーを設定する場合は、CLI コマンドの `store unit type manager` を使用して、システムをアグリゲーターから中央マネージャーに変換してください。
3. 基本ライセンスをインストールします。
   a. 「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートします。
   b. 「ライセンス」ページの「ライセンス・キー」フィールドでシステムの基本キーを入力し、「適用」をクリックして操作を続行します。
      重要: 設定するシステムに応じて、基本コレクター・キーを適用するか基本アグリゲーター・キーを適用するかが異なります。中央マネージャー・システムを設定する場合は、基本アグリゲーター・キーが必要になります。
   c. 「ご使用条件」ダイアログで基本キーに関する使用条件を読み、条件に同意する場合は「同意する」をクリックします。 使用条件に同意すると、Guardium のインターフェースが自動的に更新されます。ただし、基本ライセンス・キーをインストールしても、使用可能な機能は変更されません。
4. 1 つ以上の追加ライセンスをインストールします。 複数の追加ライセンスを購入した場合は、ライセンスごとに以下の手順を繰り返してインストールしてください。
   a. 「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートします。

b. 「ライセンス」ページの「ライセンス・キー」フィールドで追加キーを入力し、「適用」をクリックして操作を続行します。
c. 「ご使用条件」ダイアログで追加キーに関する使用条件を読み、条件に同意する場合は「同意する」をクリックします。 使用条件に同意すると、Guardium のインターフェースが自動的に更新され、追加ライセンスに関連する新しい機能を使用できるようになります。
d. インストールする追加ライセンスごとに、上記の手順を繰り返します。

次のタスク

中央マネージャーが設定されている環境の場合、「管理」 > 「一元管理」 > 「一元管理」ページで アイコンをクリックすると、中央マネージャーから管理対象ユニットに対してライセンスを配布することができます。

中央マネージャーが設定されている環境では、中央マネージャーとその管理対象ユニットで同じ共有パスワードを使用する必要があります。共有パスワードは、「設定」 > 「ツールとビュー」 > 「システム」ページで設定することも、CLI コマンドの store system shared secret を使用して設定することもできます。

ライセンス・テキストは、Guardium fileserver を開始し、opt-ibm-guardium-log/install/[LICR number] に移動することでもダウンロードできます。ここで、[LICR number] はダウンロードするライセンスを特定します。

親トピック: ステップ 5. 次の作業
関連概念:
ライセンス・キー
関連情報:
fileserver CLI コマンド

保守パッチのインストール (該当する場合)

CLI または GUI を使用してパッチをインストールすることができます。

注: フェデレーテッド環境では、保守パッチは Central Manager からすべてのアプライアンスに適用できます。

インストール・データには、保守パッチが含まれていない場合もあります。 含まれている場合は、以下の手順を実行してパッチを適用してください。

1. 前のインストール手順で定義した CLI の一時パスワードを使用して、CLI ユーザーとして Guardium® コンソールにログインします。これは、ssh クライアントを使用して実行できます。
2. 以下のいずれかを実行します。
   - ネットワーク上のロケーションからインストールする場合、次のコマンドを入力します (ftp または scp を選択)。

     ```
     store system patch install [ftp | scp]
     ```

     次のプロンプトに応答します (パッチ・ファイルの絶対パス名を指定してください)。

     Host to import patch from:

     User on <hostname>

     Full path to patch, including name:

     Password:

   - ファイル・サーバー機能を使用してインストールする場合は、次のコマンドを入力します。

     ```
     store system install patch sys
     ```

     適用するパッチを選択するよう求めるプロンプトが出されます。複数のパッチを取得するには、パス名にワイルドカードを使用します。また、パッチ名をコンマで区切ります。
3. 追加でパッチをインストールするには、ステップ 2 を繰り返します。
4. パッチが正常にインストールされたかどうかを確認するには、次の CLI コマンドを使用します。

   ```
   show system patch installed
   ```

   パッチはバックグラウンド・プロセスでインストールされます。インストールが完了するまで数分かかる場合があります。

親トピック: ステップ 5. 次の作業

追加のステップ (オプション)

次のセクションでは、ベースラインの英語を別の言語に変更する方法について説明します。これを行うには、S-TAP® エージェントをインストールし、検査エンジンを定義し、CAS エージェントをインストールします。

言語の変更

IBM Guardium のインストールは、常に英語で行われます。このベースライン言語の英語を別の言語に変更し、データベースをその言語に変換するには、CLI コマンドの store language を使用します。インストールされた Guardium システムは、日本語または中国語 (繁体字または簡体字) にのみ変更することができます。store language コマンドは、Guardium システムのセットアップ処理の一部とみなされ、このシステムの初期セットアップ中に実行されるように設計されています。 特定の言語でアプライアンスをデプロイメントした後にこの CLI コマンドを実行すると、既にキャプチャー、保管、カスタマイズ、アーカイブ、またはエクスポートされた情報が変更される可能性があります。 例えば、psmls (作成済みのペインとポートレット) は新しい言語で再作成する必要があるため、削除されます。

注: Guardium UI に言語が混合されて表示されるのを回避するには、中央マネージャーと管理対象ユニットを同じ言語に設定します。

S-TAP エージェントのインストール

S-TAP エージェントをデータベース・サーバーにインストールし、その検査エンジンを定義します。S-TAP は、データベース・サーバーにインストールされる単純なソフトウェア・エージェントです。このエージェントは、ローカル・データベースとネットワーク・データベースのトラフィックをモニターし、関連する情報を Guardium システム (コレクター) に送信します。この情報を使用して、詳細な分析、レポート作成、アラート処理が実行されます。S-TAP をインストールするには、インフォメーション・センターで S-TAP に関するセクションを参照してください。S-TAP がインストールされて Guardium システムに接続されていることを確認するには、以下の手順を実行します。

1. 管理者ポータルにログインします。
2. 以下のいずれかを実行します。

「管理」 > 「システム・ビュー」にナビゲートし、メニューで「S-TAP 状況モニター」をクリックします。アクティブなすべての S-TAP は、緑色の背景で表示されます。赤色の背景は、S-TAP がアクティブでないことを示しています。

「管理」 > 「アクティビティー・モニター」 > 「S-TAP 制御」にナビゲートし、対象の S-TAP の状況ライトが緑色になっていることを確認します。

検査エンジンの定義

ネットワーク・ベースのアクティビティー・モニター用の検査エンジンの定義

CAS エージェントのインストール

データベース・サーバーへの構成監査システム (CAS) エージェントのインストール

親トピック: ステップ 5. 次の作業

仮想イメージの作成

仮想イメージをインストールする場合は、このセクションを参照してください。

- VMware インフラストラクチャーの概要
  Guardium VM は任意の VMware 製品にインストールできますが、仮想ソリューション用のプラットフォームには VMware ESX Server が推奨されます。VMware ESX Server について、ここで紹介します。
- VM のインストールの概要
  IBM Security Guardium VM をインストールするには、ここで説明するステップに従います。VM をインストールしたら、『ステップ 3. IBM Security Guardium イメージのインストール』および『ステップ 4. 初期構成と基本構成の設定』に戻ってください。
- Hyper-V 仮想マシンの作成

親トピック: Guardium システムのインストール

VMware インフラストラクチャーの概要

Guardium VM は任意の VMware 製品にインストールできますが、仮想ソリューション用のプラットフォームには VMware ESX Server が推奨されます。VMware ESX Server について、ここで紹介します。

Guardium VM をインストールできる VMware ESX Server は、VMware インフラストラクチャーの 1 つのコンポーネントです。Guardium VM をサポートするために VMware インフラストラクチャーのすべてのコンポーネントが必要となるわけではありませんが、インストール済み環境で使用しているコンポーネントは、すべて熟知しておく必要があります。

ESX Server: このコンポーネントは、ESX Server ホストと呼ばれる物理ホスト上の VMware 仮想マシンを構成し、制御するために使用されます。Guardium VM をインストールするには、まず ESX Server ホスト上で仮想マシンを定義し、その仮想マシンに Guardium VM イメージをインストールして構成します。1 つの ESX Server に複数の Guardium VM を作成できます。

VI Client (Virtual Infrastructure Client): このコンポーネントは、スタンドアロン ESX Server または VirtualCenter Server に接続するために使用されます。VirtualCenter Server に接続する場合は、複数の ESX Server ホストに作成された複数の仮想マシンを管理できます。

Web ブラウザー: ESX Server ホストまたは VirtualCenter Server から VI Client ソフトウェアをダウンロードして使用するために使用されます。

VirtualCenter 管理サーバー (オプション): このコンポーネントは、リモートの Windows マシンで実行され、複数の ESX Server ホスト上にある複数の仮想マシンを管理するために使用できます。すべての ESX Server ホストの単一制御点として機能します。

データベース (オプション): VirtualCenter Server では、データベースを使用してインフラストラクチャーの構成情報が保管されます。VirtualCenter Server を使用しない場合、データベースは不要です。

ライセンス・サーバー (オプション): VMware インフラストラクチャーを保守するために必要なライセンスを保管し、管理します。

詳しくは、www.vmware.com にアクセスし、ESX Quick Start を検索してください。

親トピック: 仮想イメージの作成

VM のインストールの概要

IBM Security Guardium VM をインストールするには、ここで説明するステップに従います。VM をインストールしたら、『ステップ 3. IBM Security Guardium イメージのインストール』および『ステップ 4. 初期構成と基本構成の設定』に戻ってください。

VMware VirtualCenter 管理サーバー環境に複数の Guardium VM システムをインストールする場合は、最初に作成する Guardium VM からテンプレート・システムを作成し、必要に応じてそのテンプレートをコピーできます。その後は、コピーした各システムで IP アドレスを設定するだけで済みます。詳しくは、ステップ 7 の後の注を参照してください。

ステップ 1: システム互換性の検証

1. ホストが VMware ESX Server に対応していることを検証します (Guardium システムを実行するには、ESX 4.0 Update 4 以降が最低限必要です)。 詳しくは、VMware の資料「Systems Compatibility Guide for ESX Server」を参照してください (PDF 版がオンラインで提供されています)。
2. ホストにインストールされる仮想マシンが、Guardium システムに推奨最小リソースを提供できるかどうかを検証します (この場合、システムをコレクター、中央マネージャー、アグリゲーターのいずれとして使用するかは関係ありません)。この資料の『ハードウェア要件』セクションに記載している最小/推奨リソースを参照してください。
3. 64 ビット VM を初めて作成する場合、または 32 ビット VM を 64 ビットにアップグレードする場合は、仮想ハードウェアが 64 ビット操作に対応するように正しく構成されていることを確認します。場合によっては、仮想ハードウェアのアップグレード操作を実行するが必要があります。詳しくは、VMware の資料を参照してください。

ステップ 2: VMware ESX Server のインストール

VMware ESX Server をインストールします (まだインストールしていない場合)。VMware では、インストールに関する説明を Web サイトに掲載して、VMware インフラストラクチャーおよび ESX Server のインストールと構成を支援しています。

注: ESX Server は、特定のハードウェア・デバイス・セットでのみサポートされます。詳しくは、VMware Virtual Infrastructure の資料を参照してください。

ステップ 3: ネットワーク・ケーブルの接続

Guardium VM に使用する仮想スイッチを定義する前に、適切な NIC をネットワークに接続する必要があります。NIC を物理的に接続しないと、NIC を仮想ネットワークや仮想スイッチに割り当てることはできません。

Guardium VM がネットワーク・インターフェースをどのように使用するかを次の表で説明します。Guardium VM が使用するように仮想スイッチを構成する前に、この表を参照して適切に接続を行ってください。

表 1. IBM Security Guardium VM ネットワーク・インターフェースの使用

インターフェース	記述
プロキシー・インターフェース (eth0)	このインターフェースは、アプライアンスに対するメインゲートウェイであり、以下の目的で使用されます。 • ソリューションを管理し、構成し、使用するための Web ベースのグラフィカル・ユーザー・インターフェース (GUI) • 初期設定と基本構成を行うためのコマンド行インターフェース (CLI) • 外部システム (バックアップ・システム、データベース・サーバー、LDAP サーバーなど) との接続 • 他の Guardium コンポーネントとの通信。他のコンポーネントとは、他のアプライアンス (アグリゲーターや中央マネージャーなど) や、データベース・サーバーやファイル・サーバーにインストールされたエージェント (S-TAP や CAS クライアントなど) などです。
アプリケーション・サーバー・インターフェース (eth1)	このインターフェースは、Guardium システムを透過プロキシーとして構成する場合に必要となります。このインターフェースは、Guardium システムでコンテンツがマスクされるように構成されているアプリケーション・サーバーに接続します。

ステップ 4: Guardium VM 管理ポータルの構成

VMware ESX Server の新規インストール済み環境のデフォルトの構成では、VMware サービス・コンソールとすべての仮想マシンが使用する 1 つのポート・グループが作成されます。Guardium VM では、VMware コンソールや他の仮想マシンとポートを共有しないことを強くお勧めします。以下の手順に従って、Guardium VM で使用される仮想スイッチを 1 つ以上作成します。

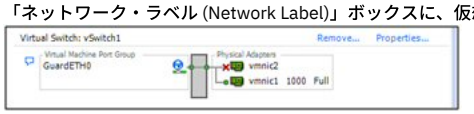
1. VMware VI Client を開き、新規の仮想マシンを作成する VirtualCenter Server または ESX Server ホストにログオンします。
2. VirtualCenter Server にログインした場合は、ナビゲーション・バーで「インベントリ (Inventory)」をクリックし、必要に応じてインベントリーを展開して、Guardium VM をインストールする管理対象ホストまたはクラスターを表示します。
3. インベントリー表示で、Guardium VM をインストールするホストまたはクラスターをクリックします。
4. 「構成 (Configuration)」タブをクリックし、「ハードウェア (Hardware)」ボックスで「ネットワーク (Networking)」をクリックし、「ネットワークの追加 (Add Networking)」をクリックします。

さまざまな目的に使用される「ネットワークの追加ウィザード (Add Network Wizard)」が開きます。

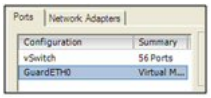
「ネットワークの追加ウィザード (Add Network Wizard)」を使用して、Guardium VM ネットワーク・インターフェース用の新しい仮想スイッチを定義します。この接続を介して、ユーザーは Guardium VM 管理コンソールにアクセスし、Guardium VM は他の Guardium コンポーネント (例えば S-TAP (後で 1 つ以上のデータベース・サーバーにインストールするソフトウェア・エージェント) など) と通信します。

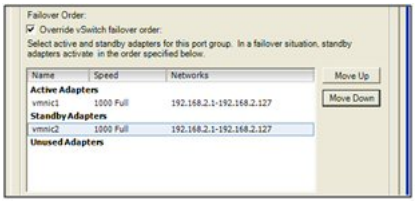
5. 「接続タイプ (Connection Types)」ボックスで、「仮想マシン (Virtual Machine)」をクリックし、「次へ (Next)」をクリックします。
6. 「ネットワーク アクセス (Network Access)」パネルで、「仮想スイッチの作成 (Create a virtual switch)」をクリックし、Guardium VM ネットワーク・インターフェースに使用する未要求ネットワーク・アダプターにマークを付けます。

7. VMware IP チーミング機能を使用して 2 次 (フェイルオーバー) ネットワーク・インターフェースを提供する場合は、オプションとして 2 番目の未要求ネットワーク・アダプターにマークを付けます。この 2 番目のアダプターは、後でスタンバイ・アダプターとして指定します (もちろん、両方の NIC を適切にケーブル接続する必要があります)。
8. 「次へ (Next)」をクリックして「ネットワークの追加ウィザード (Add Network Wizard)」の「接続設定 (Connection Settings)」ページに進みます。
9. 「ネットワーク・ラベル (Network Label)」ボックスに、仮想マシン・ポート・グループの名前 (GuardETH0 など) を入力し、「次へ」をクリックします。



10. 「サマリ (Summary)」ページで「終了 (Finish)」をクリックします。新規の仮想スイッチが「構成 (Configuration)」タブに表示されます。
11. オプション。フェイルオーバーの目的で 2 番目のアダプターを定義した場合は、(a) 作成したばかりの仮想スイッチの「プロパティ リンク (Properties link)」をクリックして仮想スイッチの「プロパティ (Properties)」パネルを開きます。(b) 「ポート (Ports)」タブをクリックし、作成したばかりの仮想ポート・グループ (GuardETH0 など) を選択し、「編集 (Edit)」をクリックします。 (c) 仮想ポート・グループの「プロパティ (Properties)」パネルで、「NIC チーミング (NIC Teaming)」タブをクリックし、「vSwitch のフェイルオーバーの置き換え (Override vSwitch Failover)」ボックスにチェック・マークを付けて、2 番目のアダプターを「スタンバイ アダプタ (Standby Adapters)」リストに移動します。(d) 「OK」をクリックして仮想ポート・グループの「プロパティ (Properties)」ボックスを閉じ、「閉じる (Close)」をクリックして仮想スイッチの「プロパティ (Properties)」ボックスを閉じます。





ステップ 5: 新規仮想マシンの作成

Guardium VM をインストールする新規仮想マシンを作成します (まだ作成していない場合)。

このタスクは、VMware VI Client を使用して実行します。

1. VMware VI Client を開き、新規の仮想マシンを作成する VirtualCenter Server または ESX Server ホストにログオンします。
2. VirtualCenter Server にログインした場合は、ナビゲーション・バーで「インベントリ (Inventory)」をクリックし、必要に応じてインベントリーを展開し、新規仮想マシンを追加する管理対象ホストまたはクラスターを選択します。
3. 「ファイル」メニューで、「新規 – 仮想マシン」をクリックして「新規仮想マシン・ウィザード (New Virtual Machine wizard)」の「構成タイプ (Configuration Type)」パネルを開きます。
4. 構成タイプとして「標準 (Typical)」をクリックし、「次へ (Next)」をクリックして「名前とフォルダ (Name and Folder)」パネルに進みます。
5. 「名前とフォルダ (Name and Folder)」パネルで、以下の作業を行います。

   「仮想マシン名 (Virtual Machine Name)」フィールドに新規仮想マシンの名前を入力します。この名前は VI Client のインベントリーに表示され、仮想マシン・ファイルの名前としても使用されます。

   新規仮想マシンのインベントリーの場所を設定するには、「仮想マシン インベントリの場所 (Virtual Machine Inventory Location)」のリストからフォルダーまたはデータ・センターのルート・ロケーションを選択します。

   「次へ (Next)」をクリックします。

6. ホストまたはクラスターにリソース・プールが含まれている場合は、「リソース プール (Resource Pool)」パネルが表示されるので、仮想マシンを実行するリソース (ホスト、クラスター、またはリソース・プール) を選択する必要があります。「次へ (Next)」をクリックします。
7. 「データストア (Datastore)」パネルで、新規仮想マシン・ファイルを保管するデータ・ストアを必要に応じて選択し、「次へ (Next)」をクリックします。
8. 「ゲスト OS を選択 (Choose the Guest Operating System)」パネルで、インストールする Guardium イメージに対応するオペレーティング・システムを選択します。「バージョン」ボックスで「Linux」 > 「RedHat Enterprise Linux 6、64-bit (RedHat Enterprise Linux 6, 64-bit)」をクリックし、「次へ」をクリックします。.

   この時点でオペレーティング・システムはインストールされていませんが、仮想マシンの適切なデフォルト値を設定するには、OS タイプが必要となります。

   VM の最小リソースについては、『始める前に』セクションの『ハードウェア要件』を参照してください。

9. 「仮想 CPU (Virtual CPU)」パネルで、インストールする Guardium VM のタイプに対して推奨される CPU の数を選択し、「次へ (Next)」をクリックします。
10. 「メモリ (Memory)」パネルで、インストールする Guardium VM のタイプに対して推奨されるメモリー量を選択し、「次へ (Next)」をクリックします。重要: 初期値は 16 GB 以上にする必要があります。ユーザーが必要な範囲を超えて作業することを求めている場合は、技術サポートにお問い合わせください。
11. 「ネットワーク (Network)」パネルで、必要なポート数として「1」をクリックし、「次へ (Next)」をクリックします。
12. 選択したポートに対して、「ネットワーク (Network)」プルダウン・メニューで、仮想ネットワークでの使用のために構成したポート・グループを選択します (このポート・グループは、前の手順で定義したものです)。
13. 選択したポート・グループに対し、「パワーオン時に接続 (Connect at Power On)」チェック・ボックスにマークを付け (デフォルトでマークが付いた状態になっています)、「次へ (Next)」をクリックします。

14. 「仮想ディスク容量 (Virtual Disk Capacity)」パネルの「ディスク サイズ (Disk Size)」フィールドに、新規仮想マシン用に確保するディスク・スペースのサイズを入力します。
15. 「終了準備 (Ready to Complete)」パネルで、設定内容を確認し、「終了 (Finish)」をクリックします。

これで、新規仮想マシンの定義は完了しました。オペレーティング・システムがまだインストールされていないため、仮想マシンを始動しようとしても失敗します。

ステップ 6: Guardium システムのインストール

このタスクは、VMware Virtual Infrastructure Client を使用して実行します。

1. VMware VI Client を開き、新規の仮想マシンを作成する VirtualCenter Server または ESX Server ホストにログオンします。
2. VirtualCenter Server にログインした場合は、ナビゲーション・バーで「インベントリ (Inventory)」をクリックし、必要に応じてインベントリーを展開し、Guardium VM をインストールする仮想マシンを選択します。
3. 「サマリ (Summary)」タブで「設定の編集 (Edit Settings)」をクリックします。
4. 「CD/DVD ドライブ 1 (CD/DVD Drive 1)」をクリックします。
5. 以下のいずれかのオプションを選択して、仮想 DVD デバイスが Guardium® インストール・プログラムを読み取る場所を決定します。**最初のオプションを選択することを強くお勧めします。**:

   「データストア ISO ファイル (Datastore ISO File)」 – データストア上にある Guardium インストール ISO ファイルに接続します。仮想マシンをインストールする ESX Server ホストからアクセス可能なデータ・ストアに、Guardium ISO ファイルをコピーします (まだコピーしていない場合)。「参照 (Browse)」をクリックしてファイルを選択します。

   注意: 他のオプションを選択する場合は、Guardium インストール DVD を DVD ドライブに挿入します。DVD ドライブに Guardium インストール DVD を入れた状態でシステムをリブートすると、そのシステムに Guardium がインストールされ、ホスト・オペレーティング・システムとファイルがすべて消去されます。

   「クライアント デバイス」 – VI Client が実行されているシステムの DVD デバイスに接続します。 このオプションを選択する場合は、VI Client が実行されているシステムの DVD ドライブに Guardium DVD を挿入します。

   「ホスト デバイス」 – 仮想マシンをインストールする ESX Server ホスト・マシンの DVD デバイスに接続します。このオプションを選択する場合は、ドロップダウン・メニューからデバイスを選択し、ESX Server ホスト・マシンの DVD ドライブに Guardium DVD を挿入します。

6. 「OK」をクリックします。
7. 「パワーオン (Power On)」をクリックして仮想マシンを始動します。
8. DVD ドライブのオプションとして「クライアント デバイス」を選択した場合は、ツールバーで「仮想 CD-ROM (ide0:0)」をクリックし、接続先のローカル DVD デバイスを選択します。
9. 「コンソール (Console)」タブをクリックして、仮想マシン・コンソールを表示します。
10. コレクターまたはアグリゲーターを作成するかどうかを尋ねられたら、該当するタイプを選択します。

    注意: DVD ドライブを使用した場合は、インストールが完了すると DVD が排出されます。必ずドライブからインストール DVD を取り出してください。ISO ファイルを使用した場合は、必ず仮想 CD/DVD をクライアント・ドライブまたはホスト・ドライブに変更して ISO CD ROM を削除してください。そうしないと、次回リブートしたときに Guardium がホスト・マシンにインストールされ、ホスト・マシンのオペレーティング・システムとすべてのファイルが消去されます。

    マシンは自動的にリブートされ、CLI ユーザーとしてログインするように求めるプロンプトが出されます。

11. この時点で、『ステップ 4. 初期構成と基本構成の設定』に戻って、Guardium システムの構成に関する包括的な説明を参照してください。

ステップ 7: 複数の VM のインストール

(オプション) Guardium VM を複数インストールする場合は、アプライアンスごとに手順を繰り返してもかまいませんが、最初に作成した Guardium VM をコピーし、以下のステップを実行することで作業を最小限にすることができます。

1. VMware の仮想インフラストラクチャー・サーバー製品を使用して、最初に構成した Guardium VM をテンプレートにコピーします。
2. このテンプレートから、追加で構成する Guardium VM ごとにコピーを作成します。
3. 各コピーで、Guardium VM コンソールに一時 CLI パスワードを使用して CLI ユーザーとしてログインし、前の手順で設定した IP 構成パラメーターをすべてリセットします。IP アドレスのリセット、GLOBAL_ID (GID) のリセット、およびホスト名のリセットが必須の作業です。UNIQUE_ID (UID) は自動的に設定されるため、手動による構成は必要ありません。必ず、前の手順で入力した IP 構成の設定をすべて確認してください。

   ```
   store network interface ip <ip_address>
   store network interface mask <subnet_mask>
   store product gid <n>
   store system hostname <host_name>
   ```

   作業が終了したら、restart network コマンドを入力します。

   ```
   restart network
   ```

   注: 複数のアプライアンスに同じ固有 ID (UID) が設定されないように、ホスト名を変更するたびにアプライアンスの固有 ID が再計算されます。
   注: グローバル ID (GID) には任意の数値を指定できますが、9223372036854775808 未満の固有の数値である必要があります。コピー・プロセスではこの固有の数値が必要です。他のアプライアンスからグローバル ID を取得し、このコピーに対して固有の数値を使用してください。

親トピック: 仮想イメージの作成

Hyper-V 仮想マシンの作成

始める前に

- Hyper-V は Microsoft の仮想化ソリューションです。Hyper-V を使用する Guardium ユーザーは、以前に Hyper-V を使用した経験があることが前提となります。
- インストール対象の Guardium のバージョンに対するシステム要件を確認します。

手順

1. Hyper-V マネージャーを開始し、Hyper-V サーバーに接続します。
2. 「アクション」ペインで、「新規」 > 「仮想マシン」を選択し、仮想マシンの新規作成ウィザードを開始します。「次へ」をクリックします。
3. 名前と場所を指定します。
    a. 名前: これは、Guardium システムが含まれている仮想マシンの名前です。
    b. 「仮想マシンを別の場所に格納する」を選択し、データ・ストアのパスを指定します (該当する場合)。
    c. 完了したら「次へ」をクリックします。
4. 仮想マシンの世代の指定: 「第 1 世代」を選択し、「次へ」をクリックします。
5. メモリの割り当て: 割り振られた RAM が最小システム要件を満たしていることを確認します。オプション「この仮想マシンに動的メモリを使用します」のチェックは外したままにして、「次へ」をクリックします。
6. ネットワークの構成: ご使用の仮想スイッチを選択して、「次へ」をクリックします。 ハードウェアは異なることがあり、複数の選択肢がある場合があります。
7. 仮想ハード ディスクの接続: 「仮想ハード ディスクを作成する」を選択します。
    a. 仮想ディスクへのパスを指定します。
    b. 仮想ハード・ディスクのサイズが最小システム要件を満たしていることを確認します。
    c. 「次へ」をクリックします。
8. インストール オプション: オペレーティング・システムは、物理 CD/DVD ドライブまたはイメージ・ファイル (.ISO) を使用してインストールできます。
    a. DVD のインストール: 「物理 CD/DVD ドライブ」を選択し、正しいドライブ名を選択します。
    b. .ISO のインストール: 「イメージ ファイル (.ISO)」を選択し、ご使用のイメージ・ファイルを参照します。
    c. 「次へ」をクリックします。
9. インストールの完了: 「説明」ボックスですべての選択済みオプションを確認します。「完了」をクリックし、仮想マシンを作成して、ウィザードを閉じます。
10. コンソールのオープン: 仮想マシン・ペインから新規仮想マシンを選択し、「接続」をクリックします。
11. 「ファイル」 > 「設定」をクリックします。
12. 「ハードウェア」ペインで次の手順を実行します。
    a. 「BIOS」を選択します。IDE を「スタートアップ順序」の上に移動します。
    b. 「プロセッサ」セクションを選択します。システム要件に基づいて、必要な最小数の仮想プロセッサーを割り振ります。
    c. 「プロセッサ」セクションを展開します。サブセクションで「NUMA」を選択し、最大メモリー量 (MB) をステップ 5 で入力した割り当て済みメモリーに変更します。
    d. 「SCSI コントローラー」セクションを選択し、「削除」ボタンをクリックします。
13. オプションで、「ハードウェア」ペインの下の「管理」ペインで、自動始動アクションと自動停止アクションの設定を行うことができます。
14. 「適用」をクリックし、警告が出されなければ「OK」をクリックします。
15. 緑色の「開始」ボタンをクリックして、仮想マシンを開始します。
16. Guardium システムをインストールします。詳しくは、『Guardium システムのインストール』を参照してください。インストールが完了したら、電源を切ります。
17. Hyper-v 仮想マシン・コンソールで、「ファイル」 > 「設定」を開きます。
18. 「ハードウェア」ペインで、「ネットワーク アダプター」セクションを展開し、「高度な機能」サブセクションを選択します。
19. 「静的」ラジオ・ボタンを選択して、 「MAC アドレス」を構成します。「適用」をクリックしてから、「OK」をクリックします。
20. 仮想 Guardium システムの電源を入れます。

親トピック: 仮想イメージの作成

カスタム・パーティション

ハード・ディスクのパーティションをカスタマイズする場合は、いくつかの選択を行う必要があります。

1. ブート画面で「カスタム・パーティションのインストール (Custom Partitioning Installation)」を選択します。
   「カスタム・レイアウトの作成 (Create custom layout)」を選択し、以下の表に記載されている推奨パーティション・スキームを使用します。
   注: オペレーティング・システムをメモリーにロードする特殊なプログラムであるブート・ローダーは、すべてのカスタム・パーティションのインストールに含まれています。
2. カスタム・レイアウトを作成します。この場合、ディスク上に既存のパーティションが存在しています。これらのパーティションは削除しないでください。必要なパーティションをディスク上の既存のパーティションに追加するには、「カスタム・レイアウト (custom layout)」を選択してください。以下の表に、カスタム・レイアウトの推奨値を示します。

表 1. カスタム・レイアウトの推奨値

パーティション	値
/	25 GB
スワップ・パーティション	RAM サイズの半分
/boot	5 GB
/var	残りすべて

すべての使用可能なドライブも、この画面に表示されます。パーティション化用のドライブを選択してから、インストールを実行してください。

パーティショニングが完了すると、Guardium® システム・ソフトウェアが自動的にインストールされます。

ディスク上の空きスペースを超える値が作成された場合は、エラー・メッセージが表示されます。

「OK」をクリックしてシステムをリブートし、「カスタム・パーティション (Custom Partitioning)」の最初に戻ります。

Red Hat ディストリビューションでのパーティションの処理方法について詳しくは、Red Hat Enterprise Linux の資料を参照してください。

注: デフォルト以外のパーティションを使用するシステム、つまりカスタム・パーティションを使用するシステムは、アップグレード・パッチを使用してアップグレードできません。代わりに、バックアップ、再ビルド、およびリストアの方法を使用する必要があります。システムのパーティションについて不確実な点がある場合は、Health Check p9997 をダウンロードしてインストールしてください。結果のパッチ・ログに、システムのパーティションに関する情報が含まれます。
親トピック: Guardium システムのインストール

暗号化された LVM によるパーティション化の方法

暗号化されたディスクを使用する場合は、以下の手順を実行して、論理ボリューム / と /var を含む暗号化された LVM ボリュームを作成します。

暗号化された LVM をインストールする場合、暗号鍵の入力を要求されます。その後、リブートのたびにこの暗号鍵を入力して、LVM ボリュームのロックを解除する必要があります (そのため、コンソールを使用してアプライアンスに物理的にアクセスするか、リモートからアクセスする必要があります)。

重要 – 暗号鍵をなくした場合は復元できないため、安全な場所に保管しておく必要があります。

注: オペレーティング・システムをメモリーにロードする特殊なプログラムであるブート・ローダーは、カスタム・パーティションのインストールに含まれています。このトピックの最後に、サンプルのパスワード入力画面を示します。

1. IBM Guardium の DVD を挿入してマシンをブートします。
2. ブート画面で「カスタム・パーティションのインストール (Custom Partition Installation)」を選択します。
3. Enter キーを押します。
4. 最初の RedHat Enterprise Linux 画面で、「すべてのパーティションを削除してデフォルトのレイアウトを作成 (Remove all partitions and create default layout)」をクリックします。 また、「システムの暗号化 (Encrypt system)」チェック・ボックスと「パーティション・レイアウトの確認と変更 (Review and modify partitioning layout)」チェック・ボックスも選択します。
5. 「次へ」をクリックします。
6. 次の画面に、本当にすべてのパーティションを削除するのかを尋ねる警告メッセージが表示されます。「はい」をクリックします。
7. 次の画面で「LogVol00」をクリックし、次に「編集」をクリックすると、「LVM ボリューム・グループの編集 (Edit LVM Volume Group)」ダイアログが表示されます。
8. 前の画面のリストで「LogVol00」をクリックし、次に「編集」をクリックします。
9. 次の画面で、サイズを 10240 に変更して「OK」をクリックします。
10. 次の画面のリストで「LogVol01」をクリックし、次に「編集」をクリックします。
11. システムにインストールされているメモリーの半分のサイズのスワップ・パーティションを割り振ります。このスワップ・パーティションのサイズを指定して「OK」をクリックします。
12. 「追加」をクリックします。「論理ボリュームの作成 (Make Logical Volume)」ダイアログが表示されます。
13. マウント・ポイントとして /var を指定し、残っているサイズをシステムに自動的に設定させます。
14. 各パーティションのサイズを確認します。次に「OK」クリックします。
15. 次に、「LVM ボリューム・グループの編集: VolGroup00 (Edit LVM Volume Group: VolGroup00)」ダイアログで「OK」をクリックします。
16. 次の画面で「次へ」をクリックします。パスフレーズ・ダイアログが表示されます。
17. 任意のパスフレーズを「パスフレーズの入力 (Enter passphrase)」フィールドに入力し、同じパスフレーズを「パスフレーズの確認 (Confirm passphrase)」フィールドに入力します。「OK」をクリックします。
    注: このパスフレーズは、システムをブートするたびに入力する必要があります。LVM のパスフレーズを紛失した場合、復旧することはできません。

    ブート・ローダーの構成ダイアログが表示されます。Red Hat Enterprise Linux が使用可能になっているコンピューターを起動すると、ブート・ローダーと呼ばれる特殊なプログラムにより、オペレーティング・システムがメモリーにロードされます。通常、ブート・ローダーはシステムのプライマリー・ハード・ディスク (または他のメディア・デバイス) 上に存在し、Linux カーネルおよびその必須ファイルまたは (場合によっては) 他のオペレーティング・システムをメモリーにロードします。それ以外の処理は実行しません。

    ほとんどの場合、デフォルトのオプションをそのまま使用して問題ありませんが、場合によっては、デフォルトのオプションを変更しなければならないことがあります。

18. この画面で「次へ」をクリックします。暗号化インストールが開始されます。

    このインストール中とその後のリブート時に、LVM の LUKS (Linux Unified Key Setup) パスフレーズの入力をブート中に要求されます。LUKS パスフレーズを入力すると、システムによってブート・プロセスが実行されます。

親トピック: Guardium システムのインストール

SAN 構成の例

この付録では、ハード・ディスクの事前パーティション化 (SAN をインストールする場合に必要) を行うために、コマンド・プロンプトに移動して実行する手順について説明します。

最初に SAN ストレージ・デバイス上のスペースをパーティション化してから、IBM Security Guardium OS をインストールします。このインストール用のハード・ディスクを 1 つ選択してください。

注: 使用する SAN ハードウェアにより、実際の手順が異なる場合があります。SAN へのインストールはサポートされていますが、NAS へのインストールはサポートされていません。

ステップの要約

1. システム・セットアップに入り (初期ブート時に IBM® サーバーで F1 キーを押す)、開始オプションを変更して、ブート元となる適切な PCI スロット (QLogic カードが挿入されているスロット) を選択します。
2. QLogic BIOS のロード中に Ctrl-Q を押して QLogic カードの BIOS を変更し、ブート・デバイスとして使用可能にします。 次に、ブート・デバイスの LUN (論理装置番号) を選択します。
3. RedHat 5.8 の DVD からブートし、fdisk を実行するために Rescue モードに入り、以下の表の仕様を参照して、SAN デバイス上にパーティションを作成します。

表 1. SAN デバイス上のパーティション

パーティション	スペース
1	/boot 用に 500 MB
2	システム・メモリーの量 + 4 GB
3	/ 用に 25 GB

パーティション	スペース
4	/var 用に残りすべてのスペース

注: RedHat のインストール・プロセスでは、パーティションを作成して OS をロードできますが、fdisk を使用してパーティションを事前に作成しておかないと、インストール後にシステムが正しくブートされません。

4. これまでに定義したパーティションを使用して、OS のインストールを実行します (/dev/sda デバイスのみ使用してください)。
5. システムをリブートし、残りのインストール手順 (ホスト名の指定や IP の構成など) を完了します。

注:

SAN 環境では、SAN 上のネットワーク・スイッチ内の冗長パスが原因で、単一の LUN が複数のデバイスとして RedHat 5.8 に示されます。(SDD ストレージは 8 個のデバイスでした)。

これは SAN ストレージ・ブランド/タイプの機能で、各サイトでこのように構成されます。

IBM Guardium のインストール済み環境で認識されている既存のパーティションのみを編集することが非常に重要です。そのためには、マウント・ポイントを追加してファイル・システム (ext4 または swap) を設定します。サイズなど、他の設定は変更しないでください。また、OS のロード先となるデバイスを選択する際に、/dev/sda 以外のデバイスは、すべて選択を解除してください。

fdisk の実行手順

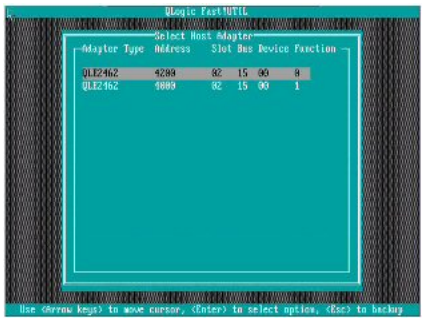
RedHat のレスキュー・モードで fdisk を実行して SAN ストレージの事前パーティション化を行うには、以下の手順を実行します。

1. `fdisk /dev/sda` と入力します (サーバーに接続されているストレージが SAN だけであると想定)。デバイス全体での処理に関する警告が表示された場合は、`y` を入力します。
2. 新しいパーティションとして `n` を入力します。
3. プライマリー・パーティションとして `p` を入力します。
4. パーティション #1 として `1` を入力します。
5. Enter キーを押して、デフォルトの開始位置を受け入れます。
6. `+512M` と入力して、パーティション #1 のサイズを 500MB に設定します (これが /boot パーティションになります)。
7. 新しいパーティションとして `n` を入力します。
8. プライマリー・パーティションとして `p` を入力します。
9. パーティション #2 として `2` を入力します。
10. Enter キーを押して、デフォルトの開始位置を受け入れます。
11. `+12288M` と入力して、パーティション #2 のサイズを 12GB に設定します (物理 RAM のサイズを 8GB と仮定した場合)。推奨サイズは、物理 RAM に 4GB を加算した値です (これがスワップ・パーティションになります)。
12. 新しいパーティションとして `n` を入力します。
13. プライマリー・パーティションとして `p` を入力します。
14. パーティション #3 として `3` を入力します。
15. Enter キーを押して、デフォルトの開始位置を受け入れます。
16. `+10240M` と入力して、パーティション #3 のサイズを 10 GB に設定します。
17. 新しいパーティションとして `n` を入力します。
18. プライマリー・パーティションとして `p` を入力します (デフォルトでパーティション #4 になります)。
19. Enter キーを押して、デフォルトの開始位置を受け入れます。
20. Enter キーを押して、残りのすべてのスペースを割り当てます (これが /var パーティションになります)。
21. `w` と入力して、パーティション・テーブルを SAN に書き込みます。
22. `exit` と入力してレスキュー・モードを終了し、システムをリブートしてカスタム・パーティションのインストールを開始します (ステップ 3: IBM Security Guardium イメージのインストール)。

QLogic のセットアップ画面のサンプル・スクリーン・ショット

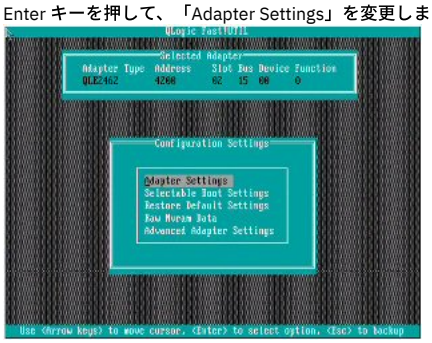
以下の Q-Logic 画面は、必要な手順を示す代表的な画面です。 他のファイバー・チャネル・カードも使用できます。

1. CTRL+D を押して、QLogic カードの BIOS を変更します。構成セットアップ・ユーティリティーに入るようにプロンプトが表示された場合に Ctrl+Q を押すと、最初に以下の画面が表示されます。これは 2 ポート・カードです。適切なポートを選択して、Enter キーを押します。



2. Enter キーを押して、「Configuration Settings」を変更します。

3. Enter キーを押して、「Adapter Settings」を変更します。

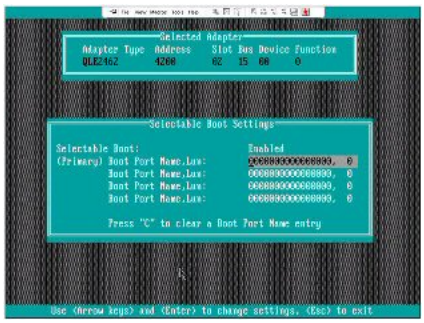


4. 矢印キーを使用して「Host Adapter BIOS」を選択し、Enter キーを押して「Enabled」に切り替えます。

5. Esc キーを押して前の画面に戻り、下矢印キーを使用して「Selectable Boot Settings」を選択し、Enter キーを押します。

6. Enter キーを押して、「Selectable Boot」を「Enabled」に変更します。

7. 最初の「Boot Port Name, LUN」を選択し、Enter キーを押して LUN のリストを表示します。正しいカード/ポートが構成されていれば、ここで LUN 番号が表示されます。リストの先頭に表示されている LUN を選択します。



8. 「Reboot」と表示されている画面に戻るまで Esc キーを押し、「Reboot」を選択してシステムをリブートします。これで、IBM Security Guardium をインストールする準備ができました。

親トピック: Guardium システムのインストール

Guardium システムのアップグレード

ここでは、IBM Security Guardium システムを最新の V10 オファリングにアップグレードする方法について説明します。

アップグレードを開始する前に、アップグレードの計画、アップグレード方法の選択、およびアップグレード中の混合バージョン環境の各セクションを参照してください。

さらに、アップグレード操作をサポートするために、次のリソースを使用できます。

- IBM Security Guardium high-level upgrade roadmap: Guardium の各種リリースからのサポートされるアップグレード・パスの概要について説明しています。
- Hints and tips on upgrading to V10: アップグレードの計画、実行、およびトラブルシューティングに関する情報をビデオで提供しています。

- アップグレードの計画
  ここでは、各種のアップグレード・シナリオについて説明し、 最小限のダウン時間で Guardium システムをアップグレードするための適切な方法を判別します。
- 共通アップグレード・タスク
  システム・データのパージ、インストールのモニター、アップグレード後のクリーンアップなどのタスクは、すべての Guardium アップグレード・シナリオに共通しています。
- 32 ビット環境のアップグレード
  バックアップ中央マネージャーを使用せずに、32 ビットの Guardium 環境をアップグレードします。
- 64 ビット環境のアップグレード
  バックアップ中央マネージャーを使用せずに、64 ビットの Guardium 環境をアップグレードします。
- バックアップ中央マネージャーを使用した 32 ビット環境のアップグレード
  バックアップ中央マネージャーを使用して、32 ビットの Guardium 環境をアップグレードします。
- バックアップ中央マネージャーを使用した 64 ビット環境のアップグレード
  バックアップ中央マネージャーを使用して、64 ビットの Guardium 環境をアップグレードします。

アップグレードの計画

ここでは、各種のアップグレード・シナリオについて説明し、 最小限のダウン時間で Guardium システムをアップグレードするための適切な方法を判別します。

- アップグレード方法の選択
  Guardium をアップグレードするための最良の方法は、アップグレード元のバージョン、ご使用のシステムのハードウェア、特別なパーティショニング要件 (存在する場合) などの複数の要因によって異なります。
- アップグレード中の混合バージョン環境
  アップグレード中に、Guardium 環境は、機能が制限される混合バージョンの状態になります。
- 中央マネージャーおよびアグリゲーターでのアップグレード
  トップダウン・アップグレード方式に従うことにより、ご使用の Guardium 環境への悪影響を最小限に抑えます。

親トピック: Guardium システムのアップグレード

アップグレード方法の選択

Guardium をアップグレードするための最良の方法は、アップグレード元のバージョン、ご使用のシステムのハードウェア、特別なパーティショニング要件 (存在する場合) などの複数の要因によって異なります。

メイン・ユーザー・インターフェースの アイコンをクリックして「Guardium バージョン情報」を選択することによって、 現在の Guardium バージョンおよびパッチ・レベルを判別します。

以下のいずれかの方法を使用して、最新バージョンの Guardium にアップグレードします。

アップグレード・パッチ
: アップグレード・パッチを使用して、管理対象環境内のすべてのシステムをアップグレードします。新しい UI アーキテクチャーのため、UI カスタマイズは例外となりますが、アップグレード・パッチはすべてのデータおよび構成を保持します。デフォルトのパーティションを使用する 64 ビット環境には、バックアップ中央マネージャーを定義せずにアップグレード・パッチを使用することをお勧めします。

バックアップ、再ビルド、リストア
: バックアップ、再ビルド、およびリストアの方法を使用します。これには、システムのフルバックアップ、最新 ISO からのシステムの再ビルド、バックアップからのシステム・データおよび構成のリストアが必要になります。カスタム・パーティションを使用する 32 ビットの環境またはシステムには、バックアップ中央マネージャー付きでバックアップ、再ビルド、およびリストアを使用する方法をお勧めします。

重要: カスタム・パーティションを使用するシステムは、アップグレード・パッチを使用して V10 にアップグレードすることはできません。代わりに、バックアップ、再ビルド、およびリストアの方法を使用する必要があります。システムのパーティションについて不確実な点がある場合は、Health Check p9997 をダウンロードしてインストールしてください。結果のパッチ・ログに、システムのパーティションに関する情報が含まれます。

以下の表を使用して、ご使用のシステムを最新バージョンの Guardium にアップグレードするための最良の方法を判別します。

表 1. アップグレード方法の判別

Guardium システム	**V10 へのアップグレード方法**	
	V9 のバックアップ、最新の V10 へのシステムの再ビルド、V9 バックアップからのリストア	**最新の V10 アップグレード・パッチの適用**
V9 パッチ 600 (64 ビット) 以降	はい	はい
V9 パッチ 600 (32 ビット) 以降	はい	いいえ
V9.0 パッチ 600 未満	はい	いいえ
V8.2 以前	いいえ	いいえ

表 2. V10 アップグレード・パスの概要

現行システムの Guardium レベル	**最新の V10 へのアップグレード・パス**
V8.2	V8.2 システムを V10 システムに直接アップグレードすることはできません。最新の V9 (64 ビット) ISO を使用してアプライアンスを再ビルドしてから、最新の V9 から V10 へのアップグレード・パッチをインストールする必要があります。 1. V8.2 のシステム・バックアップを作成します。 2. 最新の V9 (64 ビット) ISO を使用してアプライアンスを再ビルドします。 3. V9 パッチ 600 以降 (64 ビット) の GPU をインストールします。 4. 元の V8.2 システムからシステム・バックアップをリストアします。 注: コレクターについては、次のステップに進む前に、対応するすべての S-TAP を最新の V9 にアップグレードします。 5. ヘルス・チェック p9997 をインストールします。 6. V9 (64 ビット) のシステム・バックアップを作成します。 7. 最新の V9 から V10 へのアップグレード・パッチをインストールします。
V9 (32 ビット)	1. V9 (32 ビット) のシステム・バックアップを作成します。 2. 最新の V10 (64 ビット) ISO を使用してアプライアンスを再ビルドします。 3. V10 パッチ 100 以降の GPU を適用します。 4. 元の V9 (32 ビット) システムからシステム・バックアップをリストアします。
V9 パッチ 600 (64 ビット) 未満	1. V9 (64 ビット) のシステム・バックアップを作成します。 2. V9 パッチ 600 以降 (64 ビット) の GPU をインストールします。 3. V9 (64 ビット) のシステム・バックアップを作成します。 4. ヘルス・チェック p9997 をインストールします。 5. 最新の V9 から V10 へのアップグレード・パッチをインストールします。
V9 パッチ 600 (64 ビット) 以降	1. ヘルス・チェック p9997 をインストールします。 2. V9 (64 ビット) のシステム・バックアップを作成します。 3. 最新の V9 から V10 へのアップグレード・パッチをインストールします。

親トピック: アップグレードの計画
関連概念:
32 ビット環境のアップグレード
バックアップ中央マネージャーを使用した 32 ビット環境のアップグレード
64 ビット環境のアップグレード
バックアップ中央マネージャーを使用した 64 ビット環境のアップグレード

アップグレード中の混合バージョン環境

アップグレード中に、Guardium 環境は、機能が制限される混合バージョンの状態になります。

すべてのシステム (中央マネージャー、アグリゲーター、およびコレクター) とすべての S-TAPs において、アップグレード・プロセスを同時に完了することはできないため、Guardium 環境は、アップグレード中に混合バージョンの状態になります。例えば、中央マネージャーを最新の V10 にアップグレードした後も、管理対象ユニットは V9 GPU 600 で動作を続行します。混合バージョン環境はサポートされますが、すべてのアップグレード計画の一環として、いくつかの制限事項を考慮する必要があります。例えば、データ収集、データ・アセスメント、およびポリシー (いくつかの制限事項あり) は混合モードでも引き続き機能しますが、新機能と拡張機能は混合環境では機能しません。

重要: ご使用の環境全体を、可能な限り迅速に最新のパッチ・レベルである V10 にアップグレードしてください。アップグレード中に混合バージョン環境で操作する際には、以下の点に注意してください。

- 環境全体が最新の V10 にアップグレードされるまで、Guardium のすべての機能を使用することはできません。
- 混合バージョン環境での操作時には、構成を変更しないでください。
- Guardium V10 は、V9 GPU 600 未満の管理対象ユニットがある混合環境をサポートしません。

構成および設定の配布

V10 中央マネージャーと V9 パッチ 600 以降の管理対象ユニットの間では、構成の配布はサポートされません。この制限には、以下が含まれます。

- V10 中央マネージャーから V9 パッチ 600 の管理対象ユニットにポリシーを配布することはできません。アップグレード前に管理対象ユニットに既にインストール済みのポリシーは変更されません。
- パッチ・バックアップ設定は、V10 中央マネージャーから V9 パッチ 600 以降の管理対象ユニットに配布することはできません。アップグレード前に定義されたパッチ・バックアップ設定は変更されません
- V9 (パッチ 600 以降) の管理対象ユニットがある V10 中央マネージャーでは、UI レイアウトのカスタマイズおよび配布はサポートされません。

管理対象ユニット
: 中央マネージャーを V10 にアップグレードした後は、V9 パッチ 600 以降の管理対象ユニットを追加登録できません。アップグレード前に登録されたユニットは、アップグレード後も登録されたままになります。

クイック検索
: Quick Search for Enterprise は、V10 中央マネージャーおよび V9 パッチ 530 以降の管理対象ユニットで構成される混合環境で機能します。Quick Search for Enterprise を再初期化するために、ユーザー・インターフェースを再始動する必要があります。GPU 500 より前の管理対象ユニットは、Enterprise Search を利用できません。ただし、ローカル・クイック検索は引き続き使用可能です。
中央マネージャーが V9 から最新の V10 にアップグレードされ、管理対象ユニットが V9 のままである場合は、管理対象ユニットが V10 にアップグレードされるまで、V9 管理対象ユニットでクイック検索は無効になります。

レポート
: 一部のレポートを V9 パッチ 600 以降の管理対象ユニットで表示すると、SQL エラーが発生するか、以下のデータが正しく表示されない場合があります。

- 統合/アーカイブ・ログ
- 隔離された接続
- インストール済みのパッチ
- 非アクティブな検査エンジン
- S-TAP 検査
- 接続プロファイル・リスト
- リプレイ統計
- リプレイ・サマリー

エンタープライズ・バッファー使用状況モニター のデータを除き、V9 パッチ 600 以降の管理対象ユニットからのデータは、V10 中央マネージャー上の次のレポートではアクセスできません。

- エンタープライズ S-TAP 検査
- エンタープライズ・ロード・バランシング・イベント

親トピック: アップグレードの計画

中央マネージャーおよびアグリゲーターでのアップグレード

トップダウン・アップグレード方式に従うことにより、ご使用の Guardium 環境への悪影響を最小限に抑えます。

つまり、最初に 1 つの上位システムをアップグレードしてからそのシステムに従属するシステムまたはエージェントをアップグレードした後に、次の上位システムをアップグレードしてからそのシステムに従属するシステムまたはエージェントをアップグレードする、という方法で進めていきます。この方式を使用すると、混合バージョンの Guardium 環境の運用による影響を最小限に抑えられます。

トップダウン方式で行う必要がある理由は、アップグレードされたアグリゲーターは以前のリリースのデータを集約できますが、 以前のアグリゲーターは新しいリリースのデータを集約できないためです。同様に、アップグレードされた中央マネージャーは以前のリリースを実行しているユニットを管理できますが、管理対象ユニットがアップグレードされて中央マネージャーと整合するまで、管理対象ユニットは一部の機能を使用できません。

この問題を回避するには、中央マネージャーをアップグレードしてからその管理対象ユニットをアップグレードします。複数の中央マネージャーが存在する場合は、 最初に 1 つの中央マネージャーをアップグレードしてからその管理対象ユニットをアップグレードした後に、次の中央マネージャーとその管理対象ユニットのアップグレードに進みます。

同様に、1 つのアグリゲーターをアップグレードした後に、そのアグリゲーターにデータをエクスポートするユニットをアップグレードしてください。複数のアグリゲーターが存在する場合は、 最初に 1 つのアグリゲーターをアップグレードしてからそのアグリゲーターに従属するコレクターをアップグレードした後に、次のアグリゲーターとそのコレクターのアップグレードに進みます。

最後に、1 つのコレクターをアップグレードした後に、そのコレクターに登録された S-TAPs をアップグレードします。1 つのコレクターおよびそのコレクターに登録されたすべての S-TAPs をアップグレードした後に、次のコレクターおよびその S-TAPs のアップグレードに進みます。

この方式を使用すると、すべての中央マネージャーまたはアグリゲーターをアップグレードしてからすべてのコレクターをアップグレードする場合と比べて、ご使用の環境の各ブランチでのシステム (中央マネージャーからアグリゲーター、コレクター、および S-TAPs まで) の互換性がより迅速に確保されます。

親トピック: アップグレードの計画

共通アップグレード・タスク

システム・データのパージ、インストールのモニター、アップグレード後のクリーンアップなどのタスクは、すべての Guardium アップグレード・シナリオに共通しています。

- システム・データのパージ
  Guardium システムから不要なデータをパージすると、アップグレード・プロセスを大幅に迅速化することができます。
- パッチのインストール、配布、およびモニター
  アップグレードを開始する前に、パッチをアップロードしてインストールする方法、パッチのインストールのモニター方法、およびインストールが成功したかどうかを検証する方法について把握しておくと役立ちます。
- diag を使用したインストール進行状況のトラッキング
  `diag` コマンドを使用して、アップグレード・ログにアクセスし、アップグレードの進行状況をトラッキングします。

- アップグレード後の検査およびクリーンアップ
  アップグレードが正常に完了したことを確認し、アップグレード後のメンテナンスを実行します。

親トピック: Guardium システムのアップグレード

システム・データのパージ

Guardium システムから不要なデータをパージすると、アップグレード・プロセスを大幅に迅速化することができます。

このタスクについて

最適なパフォーマンスを得るために、また大量のデータのアップグレードに付随するリスクを最小限に抑えるために、不要なシステム・データをパージして、内部データベースの使用率を 20% 未満にしてください。

手順

1. 「管理」 > 「データ管理」 > 「データ・アーカイブ」を開きます。
2. 「パージ」チェック・ボックスをクリックしてパージ操作を定義します。
   重要: 「データ・アーカイブ」のパージ構成に対する変更は、 データ・エクスポートのパージ構成にも適用されます。
3. 「次の期間を経過したデータをパージ」の期間を定義します。 指定した日数、週数、または月数より古いすべてのデータがシステムからパージされます。
4. 「エクスポートまたはアーカイブなしのパージを許可」チェック・ボックスをクリックします。
5. 「保存」をクリックして、構成変更を保存します。
6. 「今すぐ 1 回実行」をクリックし、 パージ操作を実行して古いシステム・データをパージします。

次のタスク

「管理」 > 「レポート」 > 「アクティビティー・モニター」 > 「スケジュール済みジョブ」を開き、 データ・アーカイブ・ジョブの状況をモニターします。
親トピック: 共通アップグレード・タスク

パッチのインストール、配布、およびモニター

アップグレードを開始する前に、パッチをアップロードしてインストールする方法、パッチのインストールのモニター方法、およびインストールが成功したかどうかを検証する方法について把握しておくと役立ちます。

scp を使用したパッチのインストール

Guardium 環境のアップグレード時に、中央マネージャーおよび管理対象ユニットにパッチをアップロードしてインストールするには、いくつかの方法があります。

重要: ZIP 形式でダウンロードしたパッチは、アップロードおよびインストールの前に Guardium システム外部で unzip しておく必要があります。データベース構造の変更を伴うパッチについて以下の制約事項に注意してください。

- 負荷の高いレポート、監査プロセス、バックアップ、インポートなどの長期実行プロセスとの競合を避けるために、パッチのインストールは Guardium システムの負荷が低い時間に実行またはスケジュールしてください。
- パッチ・インストールの正確な所要時間は、データベース使用状況、データ分布などの考慮事項によって異なります。
- パッチのインストールはトップダウン方式、つまり最初に中央マネージャーにパッチを適用してから、アグリゲーターに適用し、最後にコレクターに適用してください。

`scp` を使用してパッチをアップロードしてインストールするには、CLI コマンドの `store system patch install scp` を実行します。

アップロードが完了すると、パッチのインストールを続行するよう自動的にプロンプトが出されます。

fileserver を使用したパッチのインストール

Guardium ファイル・サーバーを使用してパッチをアップロードしてインストールするには、以下の手順に従います。

1. CLI コマンドの `fileserver [ip_address]` を使用して、ファイル・サーバーを初期化します。ここで、`[ip_address]` は、Guardium システムに接続するために使用されるシステムです。
2. Web ブラウザーから、Guardium システムに接続します。
   a. 「パッチのアップロード (Upload Patch)」をクリックします。
   b. パッチ・ファイルを参照して選択し、「アップロード」をクリックします。
3. CLI コマンドの `store system patch install system` を実行してパッチをインストールします。

パッチの配布

中央マネージャーから管理対象ユニットにパッチを配布するには、以下のいずれかが行われている必要があります。

- 中央マネージャーにパッチがインストールされている
- CLI コマンドの `store system patch available` を実行して、中央マネージャー上でパッチが使用可能になっている

中央マネージャーの「一元管理」ページを使用して、管理対象ユニットにパッチを配布します。「管理」 > 「一元管理」 > 「一元管理」にナビゲートし、「パッチ配布」をクリックします。

パッチ・インストールのモニターおよび検証

以下の方法で、パッチのインストールをモニターおよび検証できます。

- CLI コマンドの `show system patch install` を実行します。
- CM の「一元管理」ページを使用します (「管理」 > 「一元管理」 > 「一元管理」 > 「パッチ・インストール状況」)。

重要: Guardium システムを V10 にアップグレードすると、V9 のパッチは使用できなくなります。
親トピック: 共通アップグレード・タスク

diag を使用したインストール進行状況のトラッキング

`diag` コマンドを使用して、アップグレード・ログにアクセスし、アップグレードの進行状況をトラッキングします。

手順

1. Guardium システムの CLI にログインします。
2. `diag` コマンドを実行します。
3. `diag` コマンド・メニューで、以下を行います。
   a. 「1 Output management」を選択し、「OK」をクリックします。
   b. 「3 Export recorded files」を選択し、「OK」をクリックします。
   c. 必要なログ・ファイルを選択し、「OK」をクリックします。
   d. 「1 FTP」または「2 SCP」を選択し、「OK」をクリックします。
   e. アップロード先のホスト名を入力し、「OK」をクリックします。
   f. ユーザー名を入力し、「OK」をクリックします。
   g. パスワードを入力し、「OK」をクリックします。
      注: 「2 SCP」を選択した場合は、パスワードの前に、宛先パスを要求されます。
   h. 宛先パスを入力し、「OK」をクリックします。
   i. 情報を確認し、「OK」をクリックします。 ファイルがターゲット・システムにアップロードされます。
   j. 「OK」を選択して終了します。
   k. 「3 Exit」を選択し、「OK」をクリックします。
      注: 別のファイルをアップロードする必要がある場合は、**3a** に戻ります。必要ない場合は、次のステップに進みます。
   l. 「5 Exit to CLI」を選択し、「OK」をクリックします。

親トピック: 共通アップグレード・タスク

アップグレード後の検査およびクリーンアップ

アップグレードが正常に完了したことを確認し、アップグレード後のメンテナンスを実行します。

手順

1. アップグレード・パッチを使用してアップグレードした場合は、CLI ユーザーとしてログインして、 コマンド `show upgrade-status` を実行します。 このコマンドにより、アップグレード・プロセスの状況に関する詳細な情報が出力され、出力の最終行に「`INFO:Migration Complete`」というメッセージが表示されます。
2. 中央マネージャーをアップグレードした場合は、「管理」 > 「一元管理」 > 「一元管理」ページに管理対象ユニットが表示されていることを確認します。
3. 以前のバージョンの Guardium で作成したカスタム・レポートを 「レポート」 > 「マイ・カスタム・レポート」で使用できることを確認します。

   「マイ・カスタム・レポート」には、新規に作成したレポートと、以前のバージョンの Guardium で変更した事前定義レポートがすべて含まれているはずです。

4. アップグレードされた管理対象ユニットにライセンスを配布できるように、「一元管理」ページですべての管理対象ユニットをリフレッシュします。
5. アップグレード手順またはリストア手順の後に Guardium DPS ファイルを更新する必要がある場合があります。最新の DPS ファイルをダウンロードしてから、「強化」 > 「脆弱性評価」 > 「カスタム・アップロード」ツールを使用し、 新しい DPS ファイルをアップロードしてインポートします。
6. アップグレード手順またはリストア手順の実行前にアップロードした会社ロゴを再ロードする必要がある場合があります。カスタマー・ロゴを再ロードするには、以下のステップを実行します。
   a. 管理ユーザーとしてログインします。
   b. 「設定」 > 「ツールとビュー」 > 「グローバル・プロファイル」にアクセスします。
   c. 会社ロゴ・ファイルを参照します。
   d. ロゴ・ファイルをアップロードします。
7. CLI コマンド show gui csrf_status および show gui xss_status を使用して、クロスサイト・リクエスト・フォージェリー (CSRF) サービスおよびクロスサイト・スクリプティング (XSS) サービスの状況を確認します。

親トピック: 共通アップグレード・タスク

32 ビット環境のアップグレード

バックアップ中央マネージャーを使用せずに、32 ビットの Guardium 環境をアップグレードします。

バックアップ中央マネージャーを使用せずに、ISO を介して 32 ビットの Guardium 環境をアップグレードする前に、以下のチェックリストを確認して各項目を完了してから、アップグレードを試行してください。

重要: V10 システムで `restore db` を実行する前に、システムが V10 にビルドされた後に最新の保守パッチを適用します。32 ビットのコレクター・ベースの中央マネージャーを使用している場合、V10 にアップグレードする前に、64 ビットのコレクター・ベースの中央マネージャーにビルドし直す必要があります。

アップグレード・チェックリスト

- Fix Central から最新のヘルス・チェック・パッチ (p9997) をダウンロードします。詳しくは、「Guardium health check patch release notes」を参照してください。
- 現行システムでは、 Guardium V9 および 32 ビット・アーキテクチャーを使用する必要があります。

- 最新の Guardium V9 リリースをダウンロードするか、後でこれを Fix Central から入手します (オプション)。
- パスポート・アドバンテージからの最新の Guardium V10 ISO のダウンロード
- パスポート・アドバンテージからのすべての基本ライセンスおよび追加ライセンスのダウンロード
- Fix Central からの最新の V10 GPU のダウンロード (入手できる場合)
- 次の Guardium CLI コマンドによって返されるすべてのネットワーク構成パラメーターを記録します。

```
show network interface all
show network route defaultroute
show network resolver 1
show system hostname
show system domain
```

1. 32 ビットの中央マネージャーのアップグレード
   32 ビットの Guardium 環境をアップグレードするには、以下のステップに従って、ヘルス・チェック・パッチを実行し、バックアップ、再ビルド、およびリストアの手順を使用して中央マネージャーをアップグレードします。
2. 32 ビットの管理対象ユニットのアップグレード
   バックアップ、再ビルド、およびリストアの手順を使用して、32 ビットの管理対象ユニットをアップグレードします。

親トピック: Guardium システムのアップグレード
関連概念:
アップグレードの計画

32 ビットの中央マネージャーのアップグレード

32 ビットの Guardium 環境をアップグレードするには、以下のステップに従って、ヘルス・チェック・パッチを実行し、バックアップ、再ビルド、およびリストアの手順を使用して中央マネージャーをアップグレードします。

始める前に

32 ビット環境のアップグレードのアップグレード・チェックリストを完成させます。

手順

1. システムを V9 パッチ 600 以降にアップグレードします。
2. 時刻をローカル・タイム・ゾーンに設定し、NTP サーバーを使用して、すべての Guardium システムにわたって時刻を同期します。
3. 最新のヘルス・チェック・パッチ (p9997) をダウンロードしてインストールし、インストールが正常に完了したことを確認します。その方法については、パッチのインストール、配布、およびモニターを参照してください。
4. 中央マネージャーのシステム・バックアップを取り、バックアップが正常に行われたことを確認します。
   a. 「管理」 > 「データ管理」 > 「システム・バックアップ」にナビゲートします。
   b. 設定に基づいてプロトコルを構成し、すべてのフィールドに入力します。
   c. 構成とデータの両方をバックアップします。
   重要: アップグレード手順を開始する前に、少なくとも 1 つの有効なバックアップを作成してください。
5. 最新の Guardium V10 ISO をマウントします。
   a. Guardium インストーラーに入って最初の 5 秒以内に、システム・タイプを選択します。 「スタンドアロン・コレクター (standalone collector)」ユニット・タイプの「標準インストール (非 CM) (Standard Installation (non CM))」がデフォルトの選択です。中央マネージャーまたはアグリゲーターをアップグレードする場合は、「アグリゲーター」を選択します。
   b. インストールが完了し、システムがリブートされるまで待ちます。
6. ネットワーク・パラメーターを構成します。 Guardium CLI にログインし、以下のコマンドを実行します。

```
store network interface ip
store network route defaultroute
store network resolver 1
store system hostname
store system domain
```

7. Guardium ユーザー・インターフェースにログインし、デフォルトのコンポーネントを検証します。
   注: 初回ログインの場合、デフォルトのパスワードは `guardium` です。
   a. 「ようこそ」および「設定」のナビゲーション項目のみが表示されていることを確認します。
   b. 「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートするか、  アイコンをクリックして、システムにライセンスがインストールされていないことを確認します。
8. ライセンスをインストールします。
   a. 通知リンクに従うか、「設定」 > 「ツールとビュー」 > 「ライセンス」を選択して、ライセンス・ページにナビゲートします。
   b. 関連するすべての基本ライセンスおよび追加ライセンスを適用し、使用条件に同意します。
   c. 必要に応じて、CLI にログインして CLI コマンドの `store unit type <type>` を実行し、システム・ユニット・タイプを変更します。ここで、`<type>` は、`manager`、`standalone`、`netinsp`、`mainframe`、`sink`、または `stap` です。
9. 最新の V10 GPU (最新の V10 ISO より新しい場合) および最新の保守パッチを中央マネージャーにインストールし、それらが正常にインストールされたことを確認します。
10. 中央マネージャーのデータおよび構成をリストアします。
    a. Guardium CLI コマンドの `import file` を実行して、バックアップ・ファイルをインポートします。
    b. データ・ファイルと構成ファイルは個別にインポートします。
    c. CLI コマンドの `restore db-from-prev-version` を実行して、データおよび構成のリストアを実行します。
    ヒント: restore db ログには、`diag` CLI コマンドを実行してアクセスできます。詳しくは、diag を使用したインストール進行状況のトラッキングを参照してください。
11. データおよび構成を中央マネージャーにリストアしたら、関連するすべての管理対象ユニットの情報が「一元管理」ページに表示されていることを確認します。
12. 管理対象環境が予期したとおりに機能していることを確認します。
    a. カスタム・レポートがリストアされたことを確認します。
    b. 管理対象ユニットがオンラインになっており、「一元管理」ページからアクセスできることを確認します。

重要: 混合環境での操作時には、予期される制限事項に注意してください。詳しくは、アップグレード中の混合バージョン環境を参照してください。

次のタスク

32 ビットの Guardium 中央マネージャーのアップグレードが正常に完了したら、32 ビットの管理対象ユニットのアップグレードを行います。
親トピック: 32 ビット環境のアップグレード
次のトピック: 32 ビットの管理対象ユニットのアップグレード

32 ビットの管理対象ユニットのアップグレード

バックアップ、再ビルド、およびリストアの手順を使用して、32 ビットの管理対象ユニットをアップグレードします。

始める前に

32 ビットの管理対象ユニットをアップグレードする前に、以下のタスクを確認して完了します。

- 32 ビット環境のアップグレード
- 32 ビットの中央マネージャーのアップグレード

重要: 最新の V10 にアップグレードする前に、ご使用の環境を V9 パッチ 600 以降にアップグレードする必要があります。

手順

1. 最新のヘルス・チェック・パッチ (p9997) を管理対象ユニットに配布し、正常にインストールされたことを確認します。 詳しくは、パッチのインストール、配布、およびモニターを参照してください。
2. すべての管理対象ユニットのシステム・バックアップを取ります。
3. 以下の手順を使用して、管理対象ユニットを再ビルドします。
   a. 最新の Guardium V10 ISO イメージをマウントします。
   b. Guardium インストーラーに入って 5 秒以内にシステム・タイプを選択します。 デフォルトの選択である「スタンドアロン・コレクター (standalone collector)」ユニット・タイプの「標準インストール (非 CM) (Standard Installation (non CM))」を使用するか、自動ブートするのに任せます。
   c. インストールが完了し、システムがリブートされるまで待ちます。
4. ネットワーク・パラメーターを構成します。 Guardium CLI にログインし、以下のコマンドを実行します。

```
store network interface ip
store network route defaultroute
store network resolver 1
store system hostname
store system domain
```

5. Guardium ユーザー・インターフェースにログインし、システムにライセンスがインストールされていないことを確認します。
   ヒント:
   - 初回ログインの場合、デフォルトのパスワードは `guardium` です。
   - 最終的に管理対象ユニットになるスタンドアロン・システムで作業している場合は、ライセンスをインストールする必要はありません。

   a. Guardium のメイン・ナビゲーションで、「ようこそ」および「設定」のナビゲーション項目のみが使用可能であることを確認します。
   b. 「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートするか、  アイコンをクリックして、システムにライセンスがインストールされていないことを確認します。
6. 管理対象ユニットのデータおよび構成をリストアします。
   注: バックアップから管理対象ユニットをリストアする場合、リストア時に中央マネージャーがダウンしていると、その管理対象ユニットのカスタム・レイアウトは失われます。
   a. Guardium CLI コマンドの `import file` を実行して、バックアップ・ファイルをインポートします。
   b. データ・ファイルと構成ファイルは個別にインポートします。
   c. CLI コマンドの `restore db-from-prev-version` を実行して、データおよび構成のリストアを実行します。
7. すべての管理対象ユニットが正常にアップグレードされたら、中央マネージャーから管理対象ユニットにライセンスを配布します。
   a. 中央マネージャーのユーザー・インターフェースにログインします。
   b. 「一元管理」 > 「管理」 > 「一元管理」にナビゲートし、管理対象ユニットがリストされていることを確認します。
   c. 「すべて選択」チェック・ボックスをクリックして、すべての管理対象ユニットを選択します。
   d. 「リフレッシュ」ボタンをクリックして、管理対象ユニットにライセンスを配布します。
   e. リフレッシュ・プロセスが完了するまで待ちます。
   f. 管理対象ユニットのユーザー・インターフェースにログインし、「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートして、正しいライセンスがインストールされていることを確認します。 正しいライセンスがインストールされている場合、以下のようになります。
      - 予期されるナビゲーション・メニュー・オプションが管理対象ユニットで使用可能になります。
      - 管理対象ユニットについてのレポートが機能します。
      - 中央マネージャーからリモート・データ・ソースを介してレポートにアクセスできます。
8. 中央マネージャーに最新の Guardium V10 GPU (最新の V10 ISO より新しい場合) および保守パッチがインストールされた場合、その GPU と保守パッチを管理対象ユニットに配布します。
9. VMware ツールを使用する場合は、アップグレードの完了後にツールを再インストールする必要があります。VMware ツールを再インストールするには、Guardium CLI にログインして `setup vmware_tools install` コマンドを実行し、プロンプトに従います。

タスクの結果

これで、32 ビット Guardium 環境の最新の V10 へのアップグレードが正常に完了しました。Guardium 環境の安定性を確認してください。
親トピック: 32 ビット環境のアップグレード
前のトピック: 32 ビットの中央マネージャーのアップグレード

64 ビット環境のアップグレード

バックアップ中央マネージャーを使用せずに、64 ビットの Guardium 環境をアップグレードします。

バックアップ中央マネージャーを使用せずに、ISO を介して 64 ビットの Guardium 環境をアップグレードする前に、以下のチェックリストを確認して各項目を完了してから、アップグレードを試行してください。

重要: V10 システムで `restore db` を実行する前に、システムが V10 にビルドされた後に最新の保守パッチを適用します。64 ビットのコレクター・ベースの中央マネージャーを使用している場合、アップグレード・パッチによってアップグレードが処理され、システムがコレクター・ベースの中央マネージャーからアグリゲーター・ベースの中央マネージャーに変換されます。

アップグレード・チェックリスト

- 現行システムは V9 パッチ 600 以上で、64 ビット・アーキテクチャーを使用している必要があります。
- 最新の Guardium V9 リリースをダウンロードするか、後でこれを Fix Central から入手します (オプション)。
- アップグレード・パッチ p10000 のダウンロード
- Fix Central からの最新の保守パッチのダウンロード
- Fix Central から最新のヘルス・チェック・パッチ (p9997) をダウンロードします。詳しくは、「Guardium health check patch release notes」を参照してください。

不測の事態に対応するために、以下をダウンロードしてください。

- 必要なすべての基本ライセンスおよび追加ライセンス。
- パスポート・アドバンテージから、最新の V10 ISO。

1. 64 ビットの中央マネージャーのアップグレード
   64 ビットの Guardium 環境をアップグレードするには、以下のステップに従って、ヘルス・チェック・パッチを実行し、中央マネージャーをアップグレードします。
2. 64 ビットの管理対象ユニットのアップグレード
   アップグレード・パッチを使用して、64 ビットの管理対象ユニットをアップグレードします。

親トピック: Guardium システムのアップグレード
関連概念:
アップグレードの計画

64 ビットの中央マネージャーのアップグレード

64 ビットの Guardium 環境をアップグレードするには、以下のステップに従って、ヘルス・チェック・パッチを実行し、中央マネージャーをアップグレードします。

始める前に

64 ビット環境のアップグレードのアップグレード・チェックリストを完成させます。

手順

1. システムを V9 パッチ 600 以降にアップグレードします。
2. 時刻をローカル・タイム・ゾーンに設定し、NTP サーバーを使用して、すべての Guardium システムにわたって時刻を同期します。
3. 最新のヘルス・チェック・パッチ (p9997) をダウンロードしてインストールし、インストールが正常に完了したことを確認します。その方法については、パッチのインストール、配布、およびモニターを参照してください。
4. 中央マネージャーのシステム・バックアップを取り、バックアップが正常に行われたことを確認します。
   a. 「管理」 > 「データ管理」 > 「システム・バックアップ」にナビゲートします。
   b. 設定に基づいてプロトコルを構成し、すべてのフィールドに入力します。
   c. 構成とデータの両方をバックアップします。
   重要: アップグレード手順を開始する前に、少なくとも 1 つの有効なバックアップを作成してください。
5. 中央マネージャーに p10000 をインストールし、そのインストールをモニターします。
   重要: パッチのインストールが完了すると、アップグレード・プロセスが自動的に開始し、システムがリブートされます。システムを手動でリブートしないでください。
6. オペレーティング・システムのインストールが完了するまで待ちます。
   - インストールにかかる時間は、関係するデータの量、およびシステムの仕様と構成によって異なります。
   - オペレーティング・システムのインストールが完了すると、システムは最新の Guardium V10 で初めてリブートされます。
     重要: 最新の V10 が正常にインストールされたら、システムでの最初のブート後に以下が行われます。
     - ネットワーク構成、データベース・データのマイグレーション、データベースの始動。
     - ライセンスのアップグレード、PSML のアップグレード、言語設定。
     - データベースの再始動、証明書および鍵のマイグレーション、パスワードのマイグレーション、およびファイルのクリーンアップ。
7. 中央マネージャーが正常にアップグレードされたことを確認します。
   a. Guardium CLI にログインします。CLI がリカバリー・モードに入る場合、アップグレードはまだ進行中です。
   b. CLI コマンドの show upgrade-status を実行します。 このコマンドは、リカバリー・モードの CLI から実行することもできます。
   c. 出力の最終行が「5.0:INFO:Migration Complete」であることを確認します。
   d. CLI がまだリカバリー・モードの場合は、CLI を終了して再度ログインし、通常の Guardium CLI モードに入ります。
   e. CLI コマンドの show system patch install を実行します。
   f. p10000 の状況が「Phase 5: Migration completed」であることを確認します。
8. Guardium ユーザー・インターフェースにログインし、使用条件に同意して、製品の機能を有効にします。
   a. 「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートします。
   b. 基本ライセンス契約に同意します。
   c. 該当するすべての追加ライセンス契約に同意します。
   注: このステップをスキップすると、Guardium 機能は有効になりません。
9. 管理対象ユニットがオンラインになっており、かつ「一元管理」ページからアクセスできることを確認して、管理対象環境が予期したとおりに機能していることを確認します。
   重要: 混合環境での操作時には、予期される制限事項に注意してください。詳しくは、アップグレード中の混合バージョン環境を参照してください。

10. 中央マネージャーに最新の保守パッチをインストールし、それらが正常にインストールされたことを確認します。

次のタスク

64 ビットの Guardium 中央マネージャーのアップグレードが正常に完了したら、64 ビットの管理対象ユニットのアップグレードを行います。
親トピック: 64 ビット環境のアップグレード
次のトピック: 64 ビットの管理対象ユニットのアップグレード

64 ビットの管理対象ユニットのアップグレード

アップグレード・パッチを使用して、64 ビットの管理対象ユニットをアップグレードします。

始める前に

アップグレード・パッチを使用して 64 ビットの管理対象ユニットをアップグレードする前に、以下のタスクを確認して完了します。

- 64 ビット環境のアップグレード
- 64 ビットの中央マネージャーのアップグレード

重要: 最新の V10 にアップグレードする前に、ご使用の環境を V9 パッチ 600 以降にアップグレードする必要があります。

手順

1. 最新のヘルス・チェック・パッチ (p9997) を管理対象ユニットに配布し、正常にインストールされたことを確認します。 詳しくは、パッチのインストール、配布、およびモニターを参照してください。
2. すべての管理対象ユニットのシステム・バックアップを取ります。
3. p10000 アップグレード・パッチをすべての管理対象ユニットに配布し、パッチのインストールをモニターします。 詳しくは、パッチのインストール、配布、およびモニターを参照してください。
   重要: パッチのインストールが完了すると、アップグレード・プロセスが自動的に開始し、システムがリブートされます。システムを手動でリブートしないでください。
   アップグレードに必要な時間は、関係するデータの量、およびシステムの仕様と構成によって異なります。アップグレードが完了してシステムがリブートされたら、アップグレードされたシステムの最初のブート後に以下が行われます。
   - ネットワーク構成、データベース・データのマイグレーション、データベースの始動。
   - ライセンスのアップグレード、PSML のアップグレード、言語設定。
   - データベースの再始動、証明書および鍵のマイグレーション、パスワードのマイグレーション、およびファイルのクリーンアップ。

   このプロセスの間は、データベースのマイグレーションが完了するまで、アップグレードされた管理対象ユニットにログインできなくなります。
4. 各管理対象ユニットで、アップグレード・プロセスが正常に完了したことを確認します。
   a. アップグレード対象システムの Guardium CLI にログインします。 CLI がリカバリー・モードに入る場合、アップグレードはまだ進行中です。
   b. CLI コマンドの `show upgrade-status` を実行します。 このコマンドは、リカバリー・モードの CLI から実行することもできます。
   c. 出力の最終行が「 `5.0:INFO:Migration Complete`」であることを確認します。
   d. CLI がリカバリー・モードの場合は、CLI を終了して再度ログインし、CLI モードに入ります。
   e. CLI コマンドの `show system patch install` を実行します。
      重要: 最初のリブート後にアップグレードが完了するまで、`show system patch install` は結果を返しません。
   f. アップグレード・パッチのインストール状況が「`Phase 5: Migration completed`」であることを確認します。
5. すべての管理対象ユニットが正常にアップグレードされたら、中央マネージャーから管理対象ユニットにライセンスを配布します。
   a. 中央マネージャーのユーザー・インターフェースにログインします。
   b. 「一元管理」 > 「管理」 > 「一元管理」にナビゲートし、管理対象ユニットがリストされていることを確認します。
   c. 「すべて選択」チェック・ボックスをクリックして、すべての管理対象ユニットを選択します。
   d. 「リフレッシュ」ボタンをクリックして、管理対象ユニットにライセンスを配布します。
   e. リフレッシュ・プロセスが完了するまで待ちます。
   f. 管理対象ユニットのユーザー・インターフェースにログインし、「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートして、正しいライセンスがインストールされていることを確認します。 正しいライセンスがインストールされている場合、以下のようになります。
      - 予期されるナビゲーション・メニュー・オプションが管理対象ユニットで使用可能になります。
      - 管理対象ユニットについてのレポートが機能します。
      - 中央マネージャーからリモート・データ・ソースを介してレポートにアクセスできます。
6. 最新の V10 GPU および保守パッチが中央マネージャーにインストールされた場合は、その GPU および保守パッチを管理対象ユニットに配布します。
7. VMware ツールを使用する場合は、アップグレードの完了後にツールを再インストールする必要があります。VMware ツールを再インストールするには、Guardium CLI にログインして `setup vmware_tools install` コマンドを実行し、プロンプトに従います。

タスクの結果

これで、64 ビット Guardium 環境の最新の V10 へのアップグレードが正常に完了しました。Guardium 環境の安定性を確認してください。
親トピック: 64 ビット環境のアップグレード
前のトピック: 64 ビットの中央マネージャーのアップグレード

バックアップ中央マネージャーを使用した 32 ビット環境のアップグレード

バックアップ中央マネージャーを使用して、32 ビットの Guardium 環境をアップグレードします。

バックアップ中央マネージャーを使用して 32 ビットの Guardium 環境をアップグレードする前に、以下のチェックリストを確認して各項目を完了してから、アップグレードを試行してください。

重要: V10 システムで `restore db` を実行する前に、システムが V10 にビルドされた後に最新の保守パッチを適用します。32 ビットのコレクター・ベースの中央マネージャーを使用している場合、V10 にアップグレードする前に、64 ビットのコレクター・ベースの中央マネージャーにビルドし直す必要があります。

アップグレード・チェックリスト

- 現在の環境で定義されているすべての管理対象ユニットを識別し、記録します。
- Fix Central から最新のヘルス・チェック・パッチ (p9997) をダウンロードします。詳しくは、「Guardium health check patch release notes」を参照してください。
- 現行システムでは、 Guardium V9 および 32 ビット・アーキテクチャーを使用する必要があります。
- 最新の Guardium V9 リリースをダウンロードするか、後でこれを Fix Central から入手します (オプション)。
- パスポート・アドバンテージからの最新の Guardium V10 ISO のダウンロード
- パスポート・アドバンテージからのすべての基本ライセンスおよび追加ライセンスのダウンロード
- Fix Central からの最新の V10 GPU のダウンロード (入手できる場合)
- 次の Guardium CLI コマンドによって返されるすべてのネットワーク構成パラメーターを記録します。

```
show network interface all
show network route defaultroute
show network resolver 1
show system hostname
show system domain
```

1. 32 ビットのバックアップ中央マネージャーのアップグレード
   32 ビットの Guardium 環境をアップグレードするには、以下のステップに従って、ヘルス・チェック・パッチを実行し、バックアップ、再ビルド、およびリストアの手順を使用してバックアップ中央マネージャーをアップグレードします。
2. 以前のプライマリー中央マネージャーのアップグレード (32 ビット)
   バックアップ中央マネージャーを使用する場合は、以下の手順に従って、バックアップ、再ビルド、およびリストアの手順を使用して以前の 32 ビットのプライマリー中央マネージャーをアップグレードします。
3. 32 ビットの管理対象ユニットのアップグレード
   バックアップ、再ビルド、およびリストアの手順を使用して、32 ビットの管理対象ユニットをアップグレードします。

親トピック: Guardium システムのアップグレード
関連概念:
アップグレードの計画

32 ビットのバックアップ中央マネージャーのアップグレード

32 ビットの Guardium 環境をアップグレードするには、以下のステップに従って、ヘルス・チェック・パッチを実行し、バックアップ、再ビルド、およびリストアの手順を使用してバックアップ中央マネージャーをアップグレードします。

始める前に

バックアップ中央マネージャーを使用した 32 ビット環境のアップグレードのアップグレード・チェックリストを完成させます。

手順

1. システムを V9 パッチ 600 以降にアップグレードします。
2. 時刻をローカル・タイム・ゾーンに設定し、NTP サーバーを使用して、すべての Guardium システムにわたって時刻を同期します。
3. 最新のヘルス・チェック・パッチ (p9997) をダウンロードしてインストールし、インストールが正常に完了したことを確認します。その方法については、パッチのインストール、配布、およびモニターを参照してください。
   重要: バックアップ中央マネージャーを指定する前に、プライマリー中央マネージャーとバックアップ中央マネージャー候補の両方に、最新のヘルス・チェック・パッチ (p9997) をインストールする必要があります。
4. バックアップ中央マネージャーを定義します。
   a. プライマリー中央マネージャーの「一元管理」ページにナビゲートします。
   b. 管理対象アグリゲーターを選択します。
   c. プライマリー中央マネージャーとバックアップ中央マネージャーの候補に同じパッチがインストールされていることを確認します。
   d. アグリゲーターをバックアップ中央マネージャーとして指定します。
   e. プライマリー中央マネージャーの「統合/アーカイブ・ログ」を確認して、cm_sync_file.tgz ファイルが作成されたことを確認します。
5. バックアップ中央マネージャーのシステム・バックアップを取り、バックアップが正常に行われたことを確認します。
   a. 「管理」 > 「データ管理」 > 「システム・バックアップ」にナビゲートします。
   b. 設定に基づいてプロトコルを構成し、すべてのフィールドに入力します。
   c. 必ず、構成とデータの両方をバックアップしてください。
   重要: アップグレード手順を開始する前に、少なくとも 1 つの有効なバックアップを作成してください。
6. 最新の V10 ISO を使用して、バックアップ中央マネージャーを再ビルドします。
   a. 最新の V10 ISO をマウントします。
   b. Guardium インストーラーに入って最初の 5 秒以内に、システム・タイプを選択します。 「スタンドアロン・コレクター (standalone collector)」ユニット・タイプの「標準インストール (非 CM) (Standard Installation (non CM))」がデフォルトの選択です。
7. インストールが完了し、システムがリブートされるまで待ちます。
8. ネットワーク・パラメーターを構成します。 Guardium CLI にログインし、以下のコマンドを実行します。

```
store network interface ip
store network route defaultroute
store network resolver 1
store system hostname
store system domain
```

9. Guardium ユーザー・インターフェースにログインし、デフォルトのコンポーネントを検証します。
   注: 初回ログインの場合、デフォルトのパスワードは `guardium` です。
   a. 「ようこそ」および「設定」のナビゲーション項目のみが表示されていることを確認します。
   b. 「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートするか、  アイコンをクリックして、システムにライセンスがインストールされていないことを確認します。

10. ライセンスをインストールします。
    a. 通知内のリンクに従うか、「設定」 > 「ツールとビュー」 > 「ライセンス」を選択して、ライセンス・ページにナビゲートします。
    b. 関連するすべての基本ライセンスおよび追加ライセンスを適用し、使用条件に同意します。
11. 最新の V10 GPU (最新の V10 ISO より新しい場合) および最新の保守パッチを中央マネージャーにインストールし、それらが正常にインストールされたことを確認します。
12. CLI コマンドの `store system shared secret` を使用するか、「設定」 > 「ツールとビュー」 > 「システム」にナビゲートして、バックアップ中央マネージャーに共有パスワードを設定します。
13. 中央マネージャーのデータおよび構成をリストアします。
    a. Guardium CLI コマンドの `import file` を実行して、バックアップ・ファイルをインポートします。
    b. データ・ファイルと構成ファイルは個別にインポートします。
    c. CLI コマンドの `restore db-from-prev-version` を実行して、データおよび構成のリストアを実行します。

    ヒント: restore db ログには、`diag` CLI コマンドを実行してアクセスできます。詳しくは、diag を使用したインストール進行状況のトラッキングを参照してください。
14. プライマリー中央マネージャーから、V10 バックアップ中央マネージャーが使用可能かつオンラインであることを確認します。プライマリー中央マネージャーの配下にある管理対象ユニットの数を確認して記録します (この情報は、バックアップ中央マネージャーへの移行後に使用されます)。
    重要: バックアップ中央マネージャー (最新の Guardium V10 を実行) は、赤色の状況ライトを表示する場合があります。これは、中央マネージャーが V10 システムに V9 シグナルを送信して失敗した場合に発生しますが、バックアップ中央マネージャーの同期ファイルがバックアップ中央マネージャーに存在する限り、引き続きサーバーをプロモートできます。リフレッシュは試みないでください。
15. プライマリー中央マネージャーの「統合/アーカイブ・ログ」を確認して、cm_sync_file.tgz ファイルがプライマリー中央マネージャーからバックアップ中央マネージャーへの転送を少なくとも 2 つ完了したことを確認します。転送は、30 分間隔で発生する必要があります。
16. バックアップ中央マネージャーをプライマリー中央マネージャーにします。 バックアップ中央マネージャーにログインすると、次のメッセージが表示される場合があります。

    ```
    The central manager version is lower than the version of this managed unit. Functionality is limited until the version mismatch is corrected.
    ```

    a. 「設定」 > 「一元管理」にナビゲートします。
    b. 「プライマリー CM に設定」をクリックします。このオプションが表示されない場合、cm_sync_file が正常に転送されていることを確認します。
    c. 「このユニットをプライマリー CM にしますか?」というメッセージに、「はい」と答えます。
    d. 「この変更には数分間かかります。また、GUI を再始動する必要があります。GUI 再始動の実行時にログオフされます。」というポップアップ・メッセージで、「閉じる」をクリックします。 ユーザー・インターフェース・ページに進行状況アイコンが表示されます。

    注: 変換プロセス中は、Guardium ユーザー・インターフェースは一時的に使用不可になります。プロセスが完了すると、ログイン画面は正常に戻ります。
17. 管理対象ユニットを新規プライマリー中央マネージャーに移行します。この処理は、完了までに時間がかかる場合があります。SSH クライアントを使用して、新規プライマリー中央マネージャーに接続して結果ログを表示します。
    a. `fileserver [ip_address] [duration]` コマンドを使用して、ファイル・サーバーを初期化します。
    b. Web ブラウザーから、新規プライマリー中央マネージャーに接続します。
    c. load_secondary_cm_sync_file.log ファイルを表示して、進行状況を確認します。このファイルは、gim-snif-guard-logs ディレクトリーにあります。
    d. 最終行の「Import CM sync info done」が表示されたら、プロセスは正常に終了しています。
    e. この時点で、ユーザー・インターフェースがリフレッシュされ、ログイン・ページが表示されます。
    f. 管理対象ユニットが新規プライマリー中央マネージャーへの移行を開始するため、プロセスが完了するのを正時まで待ちます。
18. Guardium ユーザー・インターフェースにログインし、以下のステップを完了します。
    a. 管理対象ユニットが新規プライマリー中央マネージャーによって管理されるようになったことを確認します。
    b. 以前のプライマリー中央マネージャーを除くすべての管理対象ユニットが移行済みであることを確認します。

次のタスク

バックアップ中央マネージャーのアップグレードおよび管理対象ユニットの移行が正常に完了したら、以前のプライマリー中央マネージャーのアップグレード (32 ビット)を行います。

親トピック: バックアップ中央マネージャーを使用した 32 ビット環境のアップグレード
次のトピック: 以前のプライマリー中央マネージャーのアップグレード (32 ビット)

以前のプライマリー中央マネージャーのアップグレード (32 ビット)

バックアップ中央マネージャーを使用する場合は、以下の手順に従って、バックアップ、再ビルド、およびリストアの手順を使用して以前の 32 ビットのプライマリー中央マネージャーをアップグレードします。

始める前に

バックアップ中央マネージャーが新規プライマリー中央マネージャーになったら、以前のプライマリー中央マネージャーを最新の Guardium V10 にアップグレードできます。以前のプライマリー中央マネージャーをアップグレードする前に、以下のタスクを確認して完了します。

- バックアップ中央マネージャーを使用した 32 ビット環境のアップグレード
- 32 ビットのバックアップ中央マネージャーのアップグレード

手順

1. CLI コマンドの `delete unit type manager` を実行して、以前のプライマリー中央マネージャーを再構成します。 続行する前に、以前のプライマリー中央マネージャーがスタンドアロン・アグリゲーターになったことを確認します。
2. 以前のプライマリー中央マネージャーからシステム・バックアップを取ります。 バックアップにデータと構成の両方を含めます。
3. 以下の手順を使用して、以前のプライマリー中央マネージャーを再ビルドします。
    a. 最新の Guardium V10 ISO イメージをマウントします。
    b. Guardium インストーラーに入って 5 秒以内にシステム・タイプを選択します。 以前のプライマリー中央マネージャーを使用するときは、「アグリゲーター」を選択します。
    c. インストールが完了し、システムがリブートされるまで待ちます。
4. ネットワーク・パラメーターを構成します。 Guardium CLI にログインし、以下のコマンドを実行します。

```
store network interface ip
store network route defaultroute
store network resolver 1
store system hostname
store system domain
```

5. Guardium ユーザー・インターフェースにログインし、システムにライセンスがインストールされていないことを確認します。
   ヒント:
   - 初回ログインの場合、デフォルトのパスワードは `guardium` です。
   - 最終的に管理対象ユニットになるスタンドアロン・システムで作業している場合は、ライセンスをインストールする必要はありません。
   a. Guardium のメイン・ナビゲーションで、「ようこそ」および「設定」のナビゲーション項目のみが使用可能であることを確認します。
   b. 「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートするか、  アイコンをクリックして、システムにライセンスがインストールされていないことを確認します。
6. 以前のバックアップ中央マネージャーをプライマリー中央マネージャーに変換する前に、以前のバックアップ中央マネージャーに最新の V10 GPU (最新の V10 ISO より新しい場合) および保守パッチがインストールされた場合は、同じ GPU および保守パッチを以前のプライマリー中央マネージャーにインストールします。
7. 以前のプライマリー中央マネージャーのデータおよび構成をリストアします。
   a. Guardium CLI コマンドの `import file` を実行して、バックアップ・ファイルをインポートします。
   b. データ・ファイルと構成ファイルは個別にインポートします。
   c. CLI コマンドの `restore db-from-prev-version` を実行して、データおよび構成のリストアを実行します。
8. 「設定」 > 「ツールとビュー」 > 「システム」にナビゲートして、以前のプライマリー中央マネージャーに共有パスワードを設定します。
9. 以前のプライマリー中央マネージャー (アップグレードしたばかりのシステム) を新規プライマリー中央マネージャーに対して登録します。
10. 新規バックアップ中央マネージャーを定義します。
    a. 新規プライマリー中央マネージャーで、「管理」 > 「一元管理」 > 「一元管理」にナビゲートします。
    b. 以前のプライマリー中央マネージャーを選択します。
    c. 以前のプライマリー中央マネージャーを新規バックアップ中央マネージャーとして指定します。
    d. 少なくとも 1 回のバックアップ同期が完了するまで待ちます。 最初のバックアップ同期は、1 時間以内に実行されます。
    e. 新規プライマリー中央マネージャーの「統合/アーカイブ」ログを確認して、cm_sync_file.tgz ファイルが作成されたことを確認します。
11. オプションで、新規バックアップ中央マネージャーをプライマリー中央マネージャーとして再定義することで、元の管理対象環境の構成に戻します。
    a. 「このユニットをプライマリー CM にしますか?」というメッセージに、「はい」と答えます。
    b. 「情報 (Information)」ポップアップ・メッセージで「閉じる」をクリックします。ユーザー・インターフェース・ページに進行状況アイコンが表示されます。
    重要: 変換プロセス中は、ユーザー・インターフェースは一時的に使用不可になります。プロセスが完了すると、ログイン画面は正常に戻ります。
12. 管理対象ユニットを新規プライマリー中央マネージャーに移行します。 このプロセスは、完了までに時間がかかる場合があります。SSH クライアントを使用して、新規プライマリー中央マネージャーに接続して結果ログを表示します。
    a. `fileserver [ip_address] [duration]` コマンドを使用して、ファイル・サーバーを初期化します。
    b. Web ブラウザーから、新規プライマリー中央マネージャーに接続します。
    c. load_secondary_cm_sync_file.log ファイルを表示して、進行状況を確認します。このファイルは、gim-snif-guard-logs ディレクトリーにあります。
    d. 最終行の「Import CM sync info done」が表示されたら、プロセスは正常に終了しています。
    e. この時点で、ユーザー・インターフェースがリフレッシュされ、ログイン・ページが表示されます。
    f. 管理対象ユニットが新規プライマリー中央マネージャーへの移行を開始するため、プロセスが完了するまで 5 分間待ちます。
13. 「管理」 > 「一元管理」 > 「一元管理」にナビゲートし、すべての管理対象ユニットが緑色で表示され、元のプライマリー中央マネージャーによって管理されるようになったことを確認します。 元のバックアップ中央マネージャーは、バックアップ中央マネージャーとして再構成されていない限り、管理対象ユニットのリストには表示されません。

次のタスク

これで、中央マネージャーとバックアップ中央マネージャーがアップグレードされたので、32 ビットの管理対象ユニットのアップグレードを行います。
親トピック: バックアップ中央マネージャーを使用した 32 ビット環境のアップグレード
前のトピック: 32 ビットのバックアップ中央マネージャーのアップグレード
次のトピック: 32 ビットの管理対象ユニットのアップグレード

32 ビットの管理対象ユニットのアップグレード

バックアップ、再ビルド、およびリストアの手順を使用して、32 ビットの管理対象ユニットをアップグレードします。

始める前に

32 ビットの管理対象ユニットをアップグレードする前に、以下のタスクを確認して完了します。

- バックアップ中央マネージャーを使用した 32 ビット環境のアップグレード
- 32 ビットのバックアップ中央マネージャーのアップグレード
- 以前のプライマリー中央マネージャーのアップグレード (32 ビット)

重要: 最新の V10 にアップグレードする前に、ご使用の環境を V9 パッチ 600 以降にアップグレードする必要があります。

手順

1. 最新のヘルス・チェック・パッチ (p9997) を管理対象ユニットに配布し、正常にインストールされたことを確認します。 詳しくは、パッチのインストール、配布、およびモニターを参照してください。
2. すべての管理対象ユニットのシステム・バックアップを取ります。
3. 以下の手順を使用して、管理対象ユニットを再ビルドします。
   a. 最新の Guardium V10 ISO イメージをマウントします。
   b. Guardium インストーラーに入って 5 秒以内にシステム・タイプを選択します。 デフォルトの選択である「スタンドアロン・コレクター (standalone collector)」ユニット・タイプの「標準インストール (非 CM) (Standard Installation (non CM))」を使用するか、自動ブートするのに任せます。
   c. インストールが完了し、システムがリブートされるまで待ちます。

4. ネットワーク・パラメーターを構成します。 Guardium CLI にログインし、以下のコマンドを実行します。

```
store network interface ip
store network route defaultroute
store network resolver 1
store system hostname
store system domain
```

5. Guardium ユーザー・インターフェースにログインし、システムにライセンスがインストールされていないことを確認します。
   ヒント:
   - 初回ログインの場合、デフォルトのパスワードは `guardium` です。
   - 最終的に管理対象ユニットになるスタンドアロン・システムで作業している場合は、ライセンスをインストールする必要はありません。

   a. Guardium のメイン・ナビゲーションで、「ようこそ」および「設定」のナビゲーション項目のみが使用可能であることを確認します。

   b. 「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートするか、  アイコンをクリックして、システムにライセンスがインストールされていないことを確認します。
6. 管理対象ユニットのデータおよび構成をリストアします。
   注: バックアップから管理対象ユニットをリストアする場合、リストア時に中央マネージャーがダウンしていると、その管理対象ユニットのカスタム・レイアウトは失われます。
   a. Guardium CLI コマンドの `import file` を実行して、バックアップ・ファイルをインポートします。
   b. データ・ファイルと構成ファイルは個別にインポートします。
   c. CLI コマンドの `restore db-from-prev-version` を実行して、データおよび構成のリストアを実行します。
7. すべての管理対象ユニットが正常にアップグレードされたら、中央マネージャーから管理対象ユニットにライセンスを配布します。
   a. 中央マネージャーのユーザー・インターフェースにログインします。
   b. 「一元管理」 > 「管理」 > 「一元管理」にナビゲートし、管理対象ユニットがリストされていることを確認します。
   c. 「すべて選択」チェック・ボックスをクリックして、すべての管理対象ユニットを選択します。
   d. 「リフレッシュ」ボタンをクリックして、管理対象ユニットにライセンスを配布します。
   e. リフレッシュ・プロセスが完了するまで待ちます。
   f. 管理対象ユニットのユーザー・インターフェースにログインし、「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートして、正しいライセンスがインストールされていることを確認します。 正しいライセンスがインストールされている場合、以下のようになります。
      - 予期されるナビゲーション・メニュー・オプションが管理対象ユニットで使用可能になります。
      - 管理対象ユニットについてのレポートが機能します。
      - 中央マネージャーからリモート・データ・ソースを介してレポートにアクセスできます。
8. 中央マネージャーに最新の Guardium V10 GPU (最新の V10 ISO より新しい場合) および保守パッチがインストールされた場合、その GPU と保守パッチを管理対象ユニットに配布します。
9. VMware ツールを使用する場合は、アップグレードの完了後にツールを再インストールする必要があります。VMware ツールを再インストールするには、Guardium CLI にログインして `setup vmware_tools install` コマンドを実行し、プロンプトに従います。

タスクの結果

これで、バックアップ中央マネージャーを使用した、32 ビット Guardium 環境の最新の V10 へのアップグレードが正常に完了しました。Guardium 環境の安定性を確認してください。

親トピック: バックアップ中央マネージャーを使用した 32 ビット環境のアップグレード
前のトピック: 以前のプライマリー中央マネージャーのアップグレード (32 ビット)

バックアップ中央マネージャーを使用した 64 ビット環境のアップグレード

バックアップ中央マネージャーを使用して、64 ビットの Guardium 環境をアップグレードします。

バックアップ中央マネージャーを使用して 64 ビットの Guardium 環境をアップグレードする前に、以下のチェックリストを確認して各項目を完了してから、アップグレードを試行してください。

重要: V10 システムで `restore db` を実行する前に、システムが V10 にビルドされた後に最新の保守パッチを適用します。64 ビットのコレクター・ベースの中央マネージャーを使用している場合、アップグレード・パッチによってアップグレードが処理され、システムがコレクター・ベースの中央マネージャーからアグリゲーター・ベースの中央マネージャーに変換されます。

アップグレード・チェックリスト

- 現在の環境で定義されているすべての管理対象ユニットを識別し、記録します。
- 現行システムは V9 パッチ 600 以上で、64 ビット・アーキテクチャーを使用している必要があります。
- 最新の Guardium V9 リリースをダウンロードするか、後でこれを Fix Central から入手します (オプション)。
- アップグレード・パッチ p10000 のダウンロード
- Fix Central からの最新の保守パッチのダウンロード
- Fix Central から最新のヘルス・チェック・パッチ (p9997) をダウンロードします。詳しくは、「Guardium health check patch release notes」を参照してください。

1. 64 ビットのバックアップ中央マネージャーのアップグレード
   64 ビットの Guardium 環境をアップグレードするには、以下のステップに従って、ヘルス・チェック・パッチを実行し、バックアップ、再ビルド、およびリストアの手順を使用してバックアップ中央マネージャーをアップグレードします。
2. 以前のプライマリー中央マネージャーのアップグレード (64 ビット)
   バックアップ中央マネージャーを使用する場合は、以下の手順に従って、アップグレード・パッチを使用して以前の 64 ビットのプライマリー中央マネージャーをアップグレードします。
3. 64 ビットの管理対象ユニットのアップグレード
   アップグレード・パッチを使用して、64 ビットの管理対象ユニットをアップグレードします。

親トピック: Guardium システムのアップグレード
関連概念:

64 ビットのバックアップ中央マネージャーのアップグレード

64 ビットの Guardium 環境をアップグレードするには、以下のステップに従って、ヘルス・チェック・パッチを実行し、バックアップ、再ビルド、およびリストアの手順を使用してバックアップ中央マネージャーをアップグレードします。

始める前に

バックアップ中央マネージャーを使用した 64 ビット環境のアップグレードのアップグレード・チェックリストを完成させます。

手順

1. システムを V9 パッチ 600 以降にアップグレードします。
2. 時刻をローカル・タイム・ゾーンに設定し、NTP サーバーを使用して、すべての Guardium システムにわたって時刻を同期します。
3. 最新のヘルス・チェック・パッチ (p9997) をダウンロードしてインストールし、インストールが正常に完了したことを確認します。その方法については、パッチのインストール、配布、およびモニターを参照してください。
   重要: バックアップ中央マネージャーを指定する前に、プライマリー中央マネージャーとバックアップ中央マネージャー候補の両方に、最新のヘルス・チェック・パッチ (p9997) をインストールする必要があります。
4. バックアップ中央マネージャーを定義します。
   a. プライマリー中央マネージャーの「一元管理」ページにナビゲートします。
   b. 管理対象アグリゲーターを選択します。
   c. プライマリー中央マネージャーとバックアップ中央マネージャーの候補に同じパッチがインストールされていることを確認します。
   d. アグリゲーターをバックアップ中央マネージャーとして指定します。
   e. プライマリー中央マネージャーの「統合/アーカイブ・ログ」を確認して、cm_sync_file.tgz ファイルが作成されたことを確認します。
5. バックアップ中央マネージャーのシステム・バックアップを取り、バックアップが正常に行われたことを確認します。
   a. 「管理」 > 「データ管理」 > 「システム・バックアップ」にナビゲートします。
   b. 設定に基づいてプロトコルを構成し、すべてのフィールドに入力します。
   c. 必ず、構成とデータの両方をバックアップしてください。
   重要: アップグレード手順を開始する前に、少なくとも 1 つの有効なバックアップを作成してください。
6. 中央マネージャーに p10000 をインストールし、そのインストールをモニターします。
   重要: パッチのインストールが完了すると、アップグレード・プロセスが自動的に開始し、システムがリブートされます。システムを手動でリブートしないでください。
7. オペレーティング・システムのインストールが完了するまで待ちます。
   - インストールにかかる時間は、関係するデータの量、およびシステムの仕様と構成によって異なります。
   - オペレーティング・システムのインストールが完了すると、システムは最新の Guardium V10 で初めてリブートされます。
     重要: 最新の V10 が正常にインストールされたら、システムでの最初のブート後に以下が行われます。
     - ネットワーク構成、データベース・データのマイグレーション、データベースの始動。
     - ライセンスのアップグレード、PSML のアップグレード、言語設定。
     - データベースの再始動、証明書および鍵のマイグレーション、パスワードのマイグレーション、およびファイルのクリーンアップ。
8. 以下のステップを使用して、バックアップ CM のアップグレードが正常に完了したことを確認します。
   a. CLI にログインします。
   b. CLI コマンドの show upgrade-status を実行します。
   c. 出力の最終行が「5.0:INFO:Migration Complete」であることを確認します。
   d. CLI コマンドの show system patch install を実行します。
   e. p10000 の状況が「Phase 5: Migration completed」であることを確認します。
9. 中央マネージャーに最新の保守パッチをインストールし、それらが正常にインストールされたことを確認します。
10. プライマリー中央マネージャーが、アップグレードされたバックアップ中央マネージャーを引き続き参照していることを確認します。
    重要: バックアップ中央マネージャー (最新の Guardium V10 を実行) は、赤色の状況ライトを表示する場合があります。これは、中央マネージャーが V10 システムに V9 シグナルを送信して失敗した場合に発生しますが、バックアップ中央マネージャーの同期ファイルがバックアップ中央マネージャーに存在する限り、引き続きサーバーをプロモートできます。リフレッシュは試みないでください。
11. プライマリー中央マネージャーの「統合/アーカイブ・ログ」を確認して、cm_sync_file.tgz ファイルがプライマリー中央マネージャーからバックアップ中央マネージャーへの転送を少なくとも 2 つ完了したことを確認します。転送は、30 分間隔で発生する必要があります。
12. バックアップ中央マネージャーをプライマリー中央マネージャーにします。 バックアップ中央マネージャーにログインすると、次のメッセージが表示される場合があります。

    ```
    The central manager version is lower than the version of this managed unit. Functionality is limited until the version mismatch is corrected.
    ```

    a. 「設定」 > 「一元管理」にナビゲートします。
    b. 「プライマリー CM に設定」をクリックします。このオプションが表示されない場合、cm_sync_file が正常に転送されていることを確認します。
    c. 「このユニットをプライマリー CM にしますか?」というメッセージに、「はい」と答えます。
    d. 「この変更には数分間かかります。また、GUI を再始動する必要があります。GUI 再始動の実行時にログオフされます。」というポップアップ・メッセージで、「閉じる」をクリックします。 ユーザー・インターフェース・ページに進行状況アイコンが表示されます。
    注: 変換プロセス中は、Guardium ユーザー・インターフェースは一時的に使用不可になります。プロセスが完了すると、ログイン画面は正常に戻ります。
13. 管理対象ユニットを新規プライマリー中央マネージャーに移行します。この処理は、完了までに時間がかかる場合があります。SSH クライアントを使用して、新規プライマリー中央マネージャーに接続して結果ログを表示します。
    a. `fileserver [ip_address] [duration]` コマンドを使用して、ファイル・サーバーを初期化します。
    b. Web ブラウザーから、新規プライマリー中央マネージャーに接続します。
    c. load_secondary_cm_sync_file.log ファイルを表示して、進行状況を確認します。このファイルは、gim-snif-guard-logs ディレクトリーにあります。
    d. 最終行の「Import CM sync info done」が表示されたら、プロセスは正常に終了しています。
    e. この時点で、ユーザー・インターフェースがリフレッシュされ、ログイン・ページが表示されます。
    f. 管理対象ユニットが新規プライマリー中央マネージャーへの移行を開始するため、プロセスが完了するのを正時まで待ちます。
14. Guardium ユーザー・インターフェースにログインし、使用条件に同意して、製品の機能を有効にします。
    a. 「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートします。

b. 基本ライセンス契約に同意します。
c. 該当するすべての追加ライセンス契約に同意します。
注: このステップをスキップすると、Guardium 機能は有効になりません。
15. 「一元管理」ページにナビゲートし、管理対象ユニットが、新規プライマリー中央マネージャーによって管理されるようになったことを確認します。以前のプライマリー中央マネージャーは、管理対象ユニットのリストには表示されなくなっています。

次のタスク

バックアップ中央マネージャーのアップグレードおよび管理対象ユニットの移行が正常に完了したら、以前のプライマリー中央マネージャーのアップグレード (64 ビット)を行います。
親トピック: バックアップ中央マネージャーを使用した 64 ビット環境のアップグレード
次のトピック: 以前のプライマリー中央マネージャーのアップグレード (64 ビット)

以前のプライマリー中央マネージャーのアップグレード (64 ビット)

バックアップ中央マネージャーを使用する場合は、以下の手順に従って、アップグレード・パッチを使用して以前の 64 ビットのプライマリー中央マネージャーをアップグレードします。

始める前に

バックアップ中央マネージャーが新規プライマリー中央マネージャーになったら、以前のプライマリー中央マネージャーを最新の Guardium V10 にマイグレーションできます。以前のプライマリー中央マネージャーをアップグレードする前に、以下のタスクを確認して完了します。

- バックアップ中央マネージャーを使用した 64 ビット環境のアップグレード
- 64 ビットのバックアップ中央マネージャーのアップグレード

手順

1. CLI コマンドの `delete unit type manager` を実行して、以前のプライマリー中央マネージャーを再構成します。 続行する前に、以前のプライマリー中央マネージャーがスタンドアロン・アグリゲーターになったことを確認します。
2. 以前のプライマリー中央マネージャーからシステム・バックアップを取ります。 バックアップにデータと構成の両方を含めます。
3. p10000 アップグレード・パッチを使用して以前のプライマリー中央マネージャーをアップグレードし、パッチのインストールをモニターします。 詳しくは、パッチのインストール、配布、およびモニターを参照してください。
   重要: パッチのインストールが完了すると、アップグレード・プロセスが自動的に開始し、システムがリブートされます。システムを手動でリブートしないでください。
   アップグレードに必要な時間は、関係するデータの量、およびシステムの仕様と構成によって異なります。アップグレードが完了してシステムがリブートされたら、アップグレードされたシステムの最初のブート後に以下が行われます。
   - ネットワーク構成、データベース・データのマイグレーション、データベースの始動。
   - ライセンスのアップグレード、PSML のアップグレード、言語設定。
   - データベースの再始動、証明書および鍵のマイグレーション、パスワードのマイグレーション、およびファイルのクリーンアップ。

   このプロセスの間は、データベースのマイグレーションが完了するまで、アップグレードされた管理対象ユニットにログインできなくなります。
4. 以前のプライマリー中央マネージャーで、アップグレード・プロセスが正常に完了したことを確認します。
   a. アップグレード対象システムの Guardium CLI にログインします。 CLI がリカバリー・モードに入る場合、アップグレードはまだ進行中です。
   b. CLI コマンドの `show upgrade-status` を実行します。 このコマンドは、リカバリー・モードの CLI から実行することもできます。
   c. 出力の最終行が「 `5.0:INFO:Migration Complete`」であることを確認します。
   d. CLI がリカバリー・モードの場合は、CLI を終了して再度ログインし、CLI モードに入ります。
   e. CLI コマンドの `show system patch install` を実行します。
   重要: 最初のリブート後にアップグレードが完了するまで、`show system patch install` は結果を返しません。
   f. アップグレード・パッチのインストール状況が「`Phase 5: Migration completed`」であることを確認します。
5. 以前のバックアップ中央マネージャーをプライマリー中央マネージャーに変換する前に、以前のバックアップ中央マネージャーに最新の V10 GPU (最新の V10 ISO より新しい場合) および保守パッチがインストールされた場合は、同じ GPU および保守パッチを以前のプライマリー中央マネージャーにインストールしてください。
6. 「設定」 > 「ツールとビュー」 > 「システム」にナビゲートして、以前のプライマリー中央マネージャーに共有パスワードを設定します。
7. 以前のプライマリー中央マネージャー (アップグレードしたばかりのシステム) を新規プライマリー中央マネージャーに対して登録します。
8. 新規バックアップ中央マネージャーを定義します。
   a. 新規プライマリー中央マネージャーで、「管理」 > 「一元管理」 > 「一元管理」にナビゲートします。
   b. 以前のプライマリー中央マネージャーを選択します。
   c. 以前のプライマリー中央マネージャーを新規バックアップ中央マネージャーとして指定します。
   d. 少なくとも 1 回のバックアップ同期が完了するまで待ちます。 最初のバックアップ同期は、1 時間以内に実行されます。
   e. 新規プライマリー中央マネージャーの「統合/アーカイブ」ログを確認して、cm_sync_file.tgz ファイルが作成されたことを確認します。
9. オプションで、新規バックアップ中央マネージャーをプライマリー中央マネージャーとして再定義することで、元の管理対象環境の構成に戻します。
   a. 「このユニットをプライマリー CM にしますか?」というメッセージに、「はい」と答えます。
   b. 「情報 (Information)」ポップアップ・メッセージで「閉じる」をクリックします。ユーザー・インターフェース・ページに進行状況アイコンが表示されます。
   重要: 変換プロセス中は、ユーザー・インターフェースは一時的に使用不可になります。プロセスが完了すると、ログイン画面は正常に戻ります。
10. 管理対象ユニットを新規プライマリー中央マネージャーに移行します。 このプロセスは、完了までに時間がかかる場合があります。SSH クライアントを使用して、新規プライマリー中央マネージャーに接続して結果ログを表示します。
    a. `fileserver [ip_address] [duration]` コマンドを使用して、ファイル・サーバーを初期化します。
    b. Web ブラウザーから、新規プライマリー中央マネージャーに接続します。
    c. load_secondary_cm_sync_file.log ファイルを表示して、進行状況を確認します。このファイルは、gim-snif-guard-logs ディレクトリーにあります。
    d. 最終行の「Import CM sync info done」が表示されたら、プロセスは正常に終了しています。
    e. この時点で、ユーザー・インターフェースがリフレッシュされ、ログイン・ページが表示されます。
    f. 管理対象ユニットが新規プライマリー中央マネージャーへの移行を開始するため、プロセスが完了するまで 5 分間待ちます。

11. 「管理」 > 「一元管理」 > 「一元管理」にナビゲートし、すべての管理対象ユニットが緑色で表示され、元のプライマリー中央マネージャーによって管理されるようになったことを確認します。 元のバックアップ中央マネージャーは、バックアップ中央マネージャーとして再構成されていない限り、管理対象ユニットのリストには表示されません。

次のタスク

これで、中央マネージャーとバックアップ中央マネージャーがアップグレードされたので、64 ビットの管理対象ユニットのアップグレードを行います。
親トピック: バックアップ中央マネージャーを使用した 64 ビット環境のアップグレード
前のトピック: 64 ビットのバックアップ中央マネージャーのアップグレード
次のトピック: 64 ビットの管理対象ユニットのアップグレード

64 ビットの管理対象ユニットのアップグレード

アップグレード・パッチを使用して、64 ビットの管理対象ユニットをアップグレードします。

始める前に

アップグレード・パッチを使用して 64 ビットの管理対象ユニットをアップグレードする前に、以下のタスクを確認して完了します。

- バックアップ中央マネージャーを使用した 64 ビット環境のアップグレード
- 64 ビットのバックアップ中央マネージャーのアップグレード
- 以前のプライマリー中央マネージャーのアップグレード (64 ビット)

重要: 最新の V10 にアップグレードする前に、ご使用の環境を V9 パッチ 600 以降にアップグレードする必要があります。

手順

1. 最新のヘルス・チェック・パッチ (p9997) を管理対象ユニットに配布し、正常にインストールされたことを確認します。 詳しくは、パッチのインストール、配布、およびモニターを参照してください。
2. すべての管理対象ユニットのシステム・バックアップを取ります。
3. p10000 アップグレード・パッチを中央マネージャーに転送し、管理対象ユニットで使用できるようにします。
   a. アップグレード・パッチを中央マネージャーに転送します。 詳しくは、パッチのインストール、配布、およびモニターを参照してください。
   b. 中央マネージャーから CLI コマンドの `show system patch available` を実行して、管理対象ユニットでアップグレード・パッチを使用できるようにします。
4. p10000 アップグレード・パッチをすべての管理対象ユニットに配布し、パッチのインストールをモニターします。 詳しくは、パッチのインストール、配布、およびモニターを参照してください。
   重要: パッチのインストールが完了すると、アップグレード・プロセスが自動的に開始し、システムがリブートされます。システムを手動でリブートしないでください。
   アップグレードに必要な時間は、関係するデータの量、およびシステムの仕様と構成によって異なります。アップグレードが完了してシステムがリブートされたら、アップグレードされたシステムの最初のブート後に以下が行われます。
   - ネットワーク構成、データベース・データのマイグレーション、データベースの始動。
   - ライセンスのアップグレード、PSML のアップグレード、言語設定。
   - データベースの再始動、証明書および鍵のマイグレーション、パスワードのマイグレーション、およびファイルのクリーンアップ。

   このプロセスの間は、データベースのマイグレーションが完了するまで、アップグレードされた管理対象ユニットにログインできなくなります。
5. 各管理対象ユニットで、アップグレード・プロセスが正常に完了したことを確認します。
   a. アップグレード対象システムの Guardium CLI にログインします。 CLI がリカバリー・モードに入る場合、アップグレードはまだ進行中です。
   b. CLI コマンドの `show upgrade-status` を実行します。 このコマンドは、リカバリー・モードの CLI から実行することもできます。
   c. 出力の最終行が「 `5.0:INFO:Migration Complete`」であることを確認します。
   d. CLI がリカバリー・モードの場合は、CLI を終了して再度ログインし、CLI モードに入ります。
   e. CLI コマンドの `show system patch install` を実行します。
      重要: 最初のリブート後にアップグレードが完了するまで、`show system patch install` は結果を返しません。
   f. アップグレード・パッチのインストール状況が「`Phase 5: Migration completed`」であることを確認します。
6. すべての管理対象ユニットが正常にアップグレードされたら、中央マネージャーから管理対象ユニットにライセンスを配布します。
   a. 中央マネージャーのユーザー・インターフェースにログインします。
   b. 「一元管理」 > 「管理」 > 「一元管理」にナビゲートし、管理対象ユニットがリストされていることを確認します。
   c. 「すべて選択」チェック・ボックスをクリックして、すべての管理対象ユニットを選択します。
   d. 「リフレッシュ」ボタンをクリックして、管理対象ユニットにライセンスを配布します。
   e. リフレッシュ・プロセスが完了するまで待ちます。
   f. 管理対象ユニットのユーザー・インターフェースにログインし、「設定」 > 「ツールとビュー」 > 「ライセンス」にナビゲートして、正しいライセンスがインストールされていることを確認します。 正しいライセンスがインストールされている場合、以下のようになります。
      - 予期されるナビゲーション・メニュー・オプションが管理対象ユニットで使用可能になります。
      - 管理対象ユニットについてのレポートが機能します。
      - 中央マネージャーからリモート・データ・ソースを介してレポートにアクセスできます。
7. 中央マネージャーに最新の Guardium V10 GPU (最新の V10 ISO より新しい場合) および保守パッチがインストールされた場合、その GPU と保守パッチを管理対象ユニットに配布します。
8. VMware ツールを使用する場合は、アップグレードの完了後にツールを再インストールする必要があります。VMware ツールを再インストールするには、Guardium CLI にログインして `setup vmware_tools install` コマンドを実行し、プロンプトに従います。

タスクの結果

これで、バックアップ中央マネージャーを使用した、64 ビット Guardium 環境の最新の V10 へのアップグレードが正常に完了しました。Guardium 環境の安定性を確認してください。
親トピック: バックアップ中央マネージャーを使用した 64 ビット環境のアップグレード
前のトピック: 以前のプライマリー中央マネージャーのアップグレード (64 ビット)

CLI および API

Guardium® コマンド行インターフェース (CLI) は、Guardium システムの構成、トラブルシューティング、および管理を行えるようにするための管理ツールです。 Guardium アプリケーション・プログラミング・インターフェース (API) は、多くの Guardium 関数にコマンド行からアクセスできるようにします。

- CLI の概要
  Guardium コマンド行インターフェース (CLI) は、Guardium システムの構成、トラブルシューティング、および管理を行えるようにするための管理ツールです。
- GuardAPI
  GuardAPI を使用すると、コマンド行から Guardium 機能にアクセスできます。

CLI の概要

Guardium® コマンド行インターフェース (CLI) は、Guardium システムの構成、トラブルシューティング、および管理を行えるようにするための管理ツールです。

本書の規則

すべての CLI コマンドの例は、クーリエ・フォントのテキスト (例えば、show system clock) で書かれています。

構文のルールを図示するために、一部のコマンドではその記述に従属区切り文字が使われています。 そのような区切り文字は、どのコマンド引数が必須であるか、またどのようなコンテキストで使用するかを示します。 各構文の説明では、コマンド引数間の従属関係を以下の特殊文字を使用して示します。

- < および > 記号は必須の引数を表します。
- [および] 記号はオプションの引数を表します。
- | (垂直バー) 記号は、選択項目から 1 つしか選択できない場合に各選択肢を分離するものです。 例:

```
store full-bypass <ON | OFF>
```

CLI コマンドの使用法

- コマンドとキーワードは、コマンドがあいまいにならないだけの十分な文字を入力すれば、省略可能です。 例えば、show は省略して sho にすることができます。
- ほとんどの Guardium CLI コマンドは、コマンド・ワードとそれに続く 1 つ以上の引数で構成されています。 引数は、キーワードの場合もありますし、キーワードに変数値 (例えば、IP アドレス、サブネット・マスク、日付など) が続く場合もあります。
- コマンドとキーワードには大/小文字の区別はありませんが、エレメント名には区別があります。
- コマンド構文と使用法オプションを表示するには、コマンド・ワードの後ろに引数として疑問符 (?) を入力します。
- 語句を引用符で囲むと、検索語が正確に定義されます。

CLI へのアクセス

管理者は次の方法で CLI にアクセスできます。

- 物理的に接続した PC コンソールまたは直列伝送端末
- SSH クライアントを使用したネットワーク接続

物理コンソール・アクセス

Guardium アプライアンスへの対話式アクセスは、シリアル・ポートまたはシステム・コンソールを介して行います。

PC キーボードおよびモニター - アプライアンスのフロント・パネルのビデオ・コネクターまたは背面のビデオ・コネクターのいずれかに、1 台の PC ビデオ・モニターを取り付けることができます。

PS/2 スタイルのコネクターを持つ PC キーボードは、アプライアンスの背面にある PS/2 コネクターに取り付けることができます。 また、USB キーボードは、アプライアンスの前面または背面に配置された USB コネクターに接続できます。

シリアル・ポート・アクセス - ヌル・モデム・ケーブルを使用して、端末装置または別のコンピューターをアプライアンスの背面にある 9 ピン・シリアル・ポートに接続します。 端末装置、または接続したコンピューターの端末エミュレーターは、19200-N-1 (19200 ボー、パリティーなし、ストップ・ビット 1) で通信するよう設定する必要があります。

端末をシリアル・ポートに接続するか、キーボードとモニターをコンソールに接続すると、ログイン・プロンプトが表示されます。 ユーザー名として cli と入力し、続けて、CLI ログイン手順に進みます。

ネットワーク SSH アクセス

CLI へのリモート・アクセスは、SSH クライアントを使用して、管理 IP アドレスまたはドメイン名で使用できます。 SSH クライアントは、ほとんどのデスクトップおよびサーバー・プラットフォームで、無料または商用のものを使用できます。 UNIX SSH 接続コマンドで cli ユーザーとしてログインする場合、以下のようになります。

```
ssh -l cli 192.168.2.16
```

Guardium アプライアンスの暗号指紋を受け入れるかどうか、SSH クライアントから質問が出される場合があります。 指紋を受け入れて、パスワード・プロンプトに進んでください。

注: 初回の接続後に再度指紋について尋ねられた場合、だれか他の人物が不適切なマシンにログインさせようとしている可能性があります。

CLI ログイン

CLI へのアクセスは、admin CLI アカウント cli または 5 つの CLI アカウント (guardcli1、...、guardcli5) のうちの 1 つのいずれかで行います。 5 つの CLI アカウント (guardcli1、...、guardcli5) は、管理責任を分けられるようにするために存在しています。

GuardAPI (繰り返し行うタスクを自動化する際に役立つ一連の CLI コマンド) へアクセスするためには、アクセス・マネージャーでユーザー (GUI username/guiuser) を作成し、それらのアカウントに admin または cli ロールのいずれかを付与する必要があります。 GuardAPI を使用するために CLI に適切にログインするには、5 つの CLI アカウント (guardcli1、...、guardcli5) の 1 つでログインし、さらに「set guiuser」コマンドを発行して guiuser でログインすることが必要です。 詳しくは、『GuardAPI リファレンスの概要』または『set guiuser 認証』を参照してください。

パスワード強化

さまざまな監査およびコンプライアンスの要件を満たすため、CLI アカウントに対して以下のようなパスワード規則が適用されます。

- アカウント cli については、提供されている cli パスワードを使用するか、あるいは十分に強いパスワードを設定してこのアカウントを保護してください。 システムをインストール DVD から再ビルドした直後では、Guardium の cli ユーザーにはデフォルトのパスワード guardium が設定されています。 このパスワードはただちに変更してください。
- CLI および 5 つの CLI アカウントに有効期間を規定する (デフォルトは 90 日)。 パスワードが期限切れになると、ログイン処理の間に、パスワードの変更を要求する処理が呼び出されます。
- パスワードの長さは 8 文字以上でなければならない。
- パスワードには、以下の 4 つのクラスのうち 3 つから、1 つ以上の文字を使用する必要がある。
  - 任意の大文字
  - 任意の小文字
  - 任意の数字 (0、1、2、...)
  - 任意の非英数字 (特殊文字)
- 別の GUI ユーザー名 (guiuser) を使用したアクセスがいったん認可されると、CLI 監査証跡に、ログインに使用された CLI_USER+GUI_USER のペアが示されます。
- CLI ユーザーは、管理アカウントであると見なされ、LDAP サーバーへの接続に関係なくログイン可能でなければならないため、LDAP を介して認証することはできません。

内部データベース保守中の CLI コマンドへの制限

CLI には 3 つのコマンド・セット (汎用コマンド、特殊なサポート・コマンド、およびリカバリー・コマンド) があります。サポート・コマンドは、技術サポートがシステムを分析するために使用します。リカバリー・コマンドは、データベースがダウンした場合に、システムをリカバリーするために使用します。

初期 CLI ログインは次のようになっています。

```
Welcome to CLI - your last login was <date>
```

保守またはアップグレードのため内部データベースが停止している場合、ウェルカム・メッセージにさらに情報が追加されます。

この場合には、使用可能な CLI コマンドの数が制限されています。

```
The internal database on the appliance is currently down and CLI will be working
in "recovery mode";  only a limited set of commands will be available.
```

リカバリー モード時に使用できる CLI コマンドを以下に示します。

```
support reset-password root
restart mysql
restart stopped_services
restart system
restore pre-patch-backup
restore system
```

- アグリゲーター CLI コマンド
  このセクションでは、アグリゲーター CLI コマンドをリストします。
- アラート機能 CLI コマンド
  このセクションでは、アラート機能 CLI コマンドをリストします。
- 証明書 CLI コマンド
  証明書コマンドを使用して、証明書署名要求 (CSR) の作成、および、Guardium システム上へのサーバー証明書、CA (認証局) 証明書、またはトラステッド・パス証明書のインストールを行います。
- 構成および制御 CLI コマンド
  構成および制御用に、以下の CLI コマンドを使用します。
- diag CLI コマンド
  これらの CLI コマンドを使用して、DIAG を介してトラブルシューティング・ユーティリティーおよびメンテナンス・ユーティリティーにアクセスできます。
- ファイル処理 CLI コマンド
  これらのコマンドは、システム情報のバックアップとリストアに使用します。これらのタスクの多くは、Guardium ユーザー・インターフェースから実行できます。
- 検査エンジンの CLI コマンド
  これらの CLI コマンドは、検査エンジンの構成に使用します。
- 調査ダッシュボードの CLI コマンド
  これらの CLI コマンドは、調査ダッシュボードを構成するために使用します。
- ネットワーク構成 CLI コマンド
  ネットワーク構成 CLI コマンドは、IP アドレスの設定、結合/フェイルオーバーの処理、2 次機能の処理、およびネットワークのリセットに使用します。
- サポート CLI コマンド
  以下の CLI コマンドは、技術サポートから指示された場合にのみ使用します。
- システム CLI コマンド
  これらの CLI コマンドは、システム設定の構成に使用します。
- ユーザー・アカウント、パスワード、および認証 CLI コマンド
  これらの CLI コマンドを使用して、ユーザー・アカウント、パスワードおよび認証を構成します。

親トピック: CLI および API
関連情報:
高度な Guardium システム管理および構成 (ビデオ)

アグリゲーター CLI コマンド

このセクションでは、アグリゲーター CLI コマンドをリストします。

aggregator backup keys file

このコマンドは、共有パスワード・ファイルを指定位置にバックアップするために使用します。

構文

aggregator backup keys file <user@host:/path/filename>

パラメーター

user@host:/path/filename ファイル転送操作において、バックアップ鍵ファイルのユーザー、ホスト、および絶対パス名を指定します。 指定するユーザーには、指定したディレクトリーに対する書き込み権限が必要です。

注: 共有パスワードの使用について詳しくは、『システム共有パスワード』を参照してください。

aggregator clean shared-secret

システム共有パスワードの値を NULL に設定します。 NULL の共有パスワードを使用してユニットからアーカイブまたはエクスポートされたすべてのファイルは、共有パスワードが NULL であるシステム上でのみリストアまたはインポートすることができます。

構文

aggregator clean shared-secret

注: 共有パスワードの使用について詳しくは、『システム共有パスワード』を参照してください。

aggregator debug

統合アクティビティーに関連するデバッグ情報の書き込みを開始または停止します。 これらのコマンドは、Guardium® サポートからの指示に従ってのみ使用し、十分な情報を収集した後には必ず stop コマンドを実行してください。

注: デバッグ・モードは、7 日後に自動的に期限切れになります。

構文

aggregator debug <start | stop>

aggregator list failed imports

共有パスワードの不一致が原因でインポート操作に失敗した場合、問題のファイルは /var/importdir ディレクトリーから /var/dump ディレクトリーに移動し、元のファイル名に .decrypt_failed という接尾部が付いた形式で名前変更されます。 このコマンドは、そのようなファイルをすべてリストするために使用します。

構文

aggregator list failed imports

aggregator recover failed import

このコマンドは、障害があるインポート・ファイルの再インポート操作または再リストア操作を試行する前に、これらのファイルを移動して名前変更するために使用します。 障害があるインポート・ファイルは、接尾部 .decrypt_failed が付いて /var/dump ディレクトリーに保管されます。 インポート操作またはリストア操作を再試行する前に、これらのファイルを (.decrypt_failed 接尾部を削除して) 名前変更し、/var/importdir ディレクトリーに移動する必要があります。

構文

aggregator recover failed import <all | filename>

パラメーター

all オプションを使用すると、接尾部 .decrypt_failed が末尾にあるすべてのファイルが /var/dump ディレクトリーから移動します。filename オプションを使用する場合は、移動する 1 つのファイルを指定します。

注: 障害があるファイルの移動後、リストア操作またはインポート操作を実行する前に、エクスポートまたはアーカイブしたファイルの暗号化に使用された共有パスワードとシステム共有パスワードが一致することを確認してください。

aggregator restore keys file

このコマンドは、共有パスワード・ファイルを指定位置からリストアするために使用します。

構文

aggregator restore keys file <user@host:/path/filename>

パラメーター

user@host:/path/filename ファイル転送操作において、バックアップ鍵ファイルのユーザー、ホスト、および絶対パス名を指定します。

注: 共有パスワードの使用について詳しくは、『システム共有パスワード』を参照してください。

store aggregator drop_ad_hoc_audit_db

アグリゲーターに関する監査プロセス・レポート - タスクごとに、そのタスクに関連する日のみが含まれる一時データベースを作成します。これらの一時データベースは、14 日間 (分析用として) 保持することも、使用後すぐに削除することもできます。 この CLI コマンドは、一時データベースのパージ・ポリシーを定義します。 0 または 1 を選択します (0 - 14 日間保持、または 1 - 使用後に削除)。

構文

store aggregator drop_ad_hoc_audit_db [1|0]

Drop ad-hoc merge databases? 0

show aggregator drop_ad_hoc_audit_db

store aggregator orphan_cleanup_flag

この CLI コマンドは、アグリゲーターで静的オーファンのクリーンアップを定期的に実行するために使用します。

3 日より前のデータに対して実行するようにスケジュールされ、パージの終了時に実行されるアグリゲーターで、オーファンを消去するためにこの CLI コマンドを使用します。

この処理はユーザーによってこの CLI コマンドで開始されるため、大規模なデータベースの場合、ユーザーには処理時間の長さがわかります。

アグリゲーター上のデータ全体が網羅されますが、それらすべての実行は別の一時データベースで行われます。

注: コレクターでは、オーファンのクリーンアップは変更されません。small クリーンアップ方針で実行され、エクスポート/アーカイブ前に呼び出されます。

show aggregator orphan_cleanup_flag small、large または analyze を表示します。

store aggregator orphan_cleanup_flag

store aggregator orphan_cleanup_flag <flag>, ここで、flag は < small large analyze > のうちのいずれか 1 語です。

これらのコマンドは、アグリゲーターにのみ適用できます。

small、large または analyze のいずれかを設定した場合、オーファン・クリーンアップ・スクリプトは各マージ処理実行後に呼び出されます。

アグリゲーターのオーファン・クリーンアップでは、最後の 3 日間のオーファン・レコードは削除されませんが、3 日より前のオーファンはすべて削除されます。

small が指定されている場合、マージの完了後に開始可能な監査プロセスが、この処理によって妨げられることはありません。

large が指定されている場合、多数のオーファンがある場合に処理はより高速で実行されますが、この実行によって監査プロセスが妨げられることがあります。large が指定されている場合、監査プロセスはオーファン・クリーンアップが完了するまで開始されません。

analyze が指定されている場合、この処理では最初にオーファンの数が評価され、オーファンが 20% より多い場合は「large」方針が使用されます。「analyze」が指定されている場合、監査プロセスはオーファン・クリーンアップが完了するまで開始されません。

構文

store aggregator orphan_cleanup_flag [small | large | analyze]

表示コマンド

show aggregator orphan_cleanup_flag

store archive_static_table

この CLI コマンドは、アーカイブ静的表のオン/オフを切り換えるために使用します。

使用法: store archive_static_table <state>,

where state is on/off.

表示コマンド

show archive_static_table

store next_export_static

統合ソフトウェアでは、2 つのタイプの表が区別されています。

- 静的表 - ゆっくり拡大していきます。この種の表のデータは、時間に依存していません (GDM_OBJECT、GDM_FIELD、GDM_SENTENCE、GDM_CONSTRUCT など)。
- 動的表 - 急速に拡大していきます。データは、時間に依存しています (GDM_CONSTRUCT_INSTANCE、GDM_SESSION、GDM_CONSTRUCT_TEXT など)。

上記で説明したように、静的表のデータは時間に依存していません。時間に依存する動的表のデータは静的データにリンクされます。 静的表は非常に大きくなる可能性があるため、エクスポート/アーカイブ処理ではすべての静的データが毎日保存されるわけではありません。エクスポート/アーカイブ処理では、その最初の実行時にすべての静的データが保存され、それ以降は毎月 1 日に保存されます。毎月 1 日以外の任意の日においては、その日の間に変更された静的データのみが保存されます。この理由により、任意の日のデータをリストアするときは、その月の 1 日のデータもリストアする必要があります。これにより、確実にすべての静的データが存在するようになり、参照も壊れません。

CLI コマンド store next_export_static は、次のエクスポートにすべての静的データが含まれるようにフラグを設定するときに使用します。

構文

store next_export_static [ON | OFF]

表示コマンド

show next_export_static

store last_used

この CLI コマンドは、 パージおよび統合のときに使用します。

構文

store last_used [size | interval | logging]

表示コマンド

show last_used [size | interval | logging]

LAST_USED SIZE - 整数、デフォルトは 50

LAST_USED INTERVAL - 整数、デフォルトは 60 (分)

LAST_USED LOGGING - 整数

すべての表 - 1

GDM_Object のみ - 2

なし - 0 (デフォルト)

store aggregator static_data

store aggregator static_data [TIMESTAMP | LAST_USED_FOR_OBJECT_ONLY | LAST_USED]

注: このコマンドを使用する前に、CLI コマンド last_used logging を設定してください。

静的表の LAST_USED 列をスニファー によって更新する場合、それらの表のデータをパージするとき、 またはそれらの表のデータをアーカイブしてエクスポートするときに、この列を参照することができます。

この列の値は、 データをアグリゲーターにインポートするときに更新することもできます。

以下の 3 つのオプションがあります。

1. デフォルトでは、システムは前のバージョンと同様に動作します。 つまり、LAST_USED 列は、パージ、アーカイブ、およびエクスポートにおいて考慮されず、 インポート時に更新されることもありません。アーカイブおよびエクスポートは TIMESTAMP によって行われます。
2. LAST_USED_FOR_OBJECT_ONLY が、GDM_OBJECT 表に関してのみ考慮されます。
3. LAST_USED が、GDM_CONSTRUCT、GDM_SENTENCE、GDM_OBJECT、 GDM_FIELD、GDM_JOIN、GDM_JOIN_OBJECT に関して考慮されます。

注: オプション 2 および 3 は、このデータを収集して更新するようにスニファーが構成されている場合にのみ有効です。
注: 検証はコレクター上でのみ行われます。ADMINCONSOLE_PARAMETER.LAST_USED_LOGGING=0 の場合には、TIMESTAMP のみが許可されます。ADMINCONSOLE_PARAMETER.LAST_USED_LOGGING=1 の場合には、 すべてのパラメーターが許可されます。ADMINCONSOLE_PARAMETER.LAST_USED_LOGGING=2 の場合には、 TIMESTAMP および LAST_USED_FOR_OBJECT_ONLY が許可されます。アグリゲーター上では、 すべてのパラメーターが許可されます。

構文

store aggregator static_data <type>

ここで <type> は、 <TIMESTAMP | LAST_USED | LAST_USED_FOR_OBJECT_ONLY> です。これは、last_used logging フラグに応じて異なります。

show/store last_used logging コマンドを使用してください。

表示コマンド

show aggregator static_data

store archive_table_by_date

この CLI コマンドは、アグリゲーターに対してのみ使用します。この CLI コマンドを使用すると、すべての静的表を日次ベースでアーカイブしたり、静的表のデータを最初の実行時にアーカイブしたり、毎月 1 日にアーカイブしたりすることができます。デフォルトでは、アグリゲーター上のデータのアーカイブは、すべての静的表を対象として日次ベースで実行されます。この CLI コマンドを ENABLE に設定すると、静的表は、毎月 1 日またはデータのアーカイブの初回実行時にのみアーカイブされます。

store run_cleanup_orphans_daily

この CLI コマンドを使用して、使用されなくなった古い構成レコードをすべて消去します。この CLI コマンドは、コレクターおよびアグリゲーターに関連し、デフォルトで有効になります。

store run_cleanup_orphans_daily

使用法: store run_cleanup_orphans_daily [on|off]

表示コマンド

show run_cleanup_orphans_daily

store max_number_collector

アグリゲーターによって管理されるコレクターの最大数を設定します。デフォルトは 10 です。

表示コマンド

show max_number_collector

store purge_age_period

パージ基準経過日数の期間を設定します。

表示コマンド

show purge_age_period

親トピック: CLI の概要

アラート機能 CLI コマンド

このセクションでは、アラート機能 CLI コマンドをリストします。

アラート機能サブシステムは、他のコンポーネントによってキューに入れられたメッセージを送信します。このようなメッセージの例には、異常検出サブシステムによってキューに入れられた相関アラートや、セキュリティー・ポリシーによって生成されたランタイム・アラートなどがあります。 アラート機能サブシステムは、SMTP サーバーと SNMP サーバーの両方にメッセージを送信するように構成できます。 アラートは syslog やカスタム・アラート・クラスに送信することもできますが、これら 2 つのオプションについては、アラート機能を開始する以外に特別な構成は必要ありません。 アラート機能コマンドには 4 つのタイプがあります。 リストのリンクを使用するか、このリストに続いて英字順に示されているコマンドを参照してください。

アラート機能の開始およびポーリング・コマンド

- stop alerter
- restart alerter
- store alerter state operational
- store alerter state startup
- store alerter poll
- store anomaly-detection poll
- store anomaly-detection state

SMTP 構成コマンド

- store alerter smtp authentication password
- store alerter smtp authentication type
- store alerter smtp authentication username
- store alerter smtp port
- store alerter smtp relay
- store alerter smtp returnaddr

SNMP 構成コマンド

- store alerter snmp community
- store alerter snmp traphost
- store alerter snmp secondary_traphost
- store alerter snmp secondary_community

restart alerter

アラート機能を再始動します。 次のように store alerter state operational コマンドを使用してアラート機能を停止してから開始すると、同様の機能を実行できます。

store alerter state operational off

store alerter state operational on

構文

restart alerter

stop alerter

アラート機能を停止します。

次のように store alerter state operational コマンドを使用すると、同様の機能を実行できます。

store alerter state operational off

構文

stop alerter

store alerter poll

アラート機能を開始 (on) または停止 (off) します。 インストール時のデフォルト状態は off です。 アラート機能サブシステムを再始動または停止する場合、restart alerter または stop alerter コマンドを使用することもできます。

構文

store alerter state operational <on | off>

表示コマンド

show alerter state operational

store alerter state operational

アラート機能が、SNMP トラップを送信するか SMTP を使用して E メールを送信するためにその出力メッセージ・キューを検査するまでに待機する秒数 n を設定します。デフォルトは 30 です。

構文

store alerter poll <n>

表示コマンド

show alerter poll

store alerter state startup

システム始動時のアラート機能の自動開始を有効または無効にします。 インストール時のデフォルト状態は off です。

構文

store alerter state startup <on | off>

表示コマンド

show alerter state startup

store anomaly-detection poll

異常検出ポーリング間隔を、分単位 (n) で設定します。 これにより、Guardium® がログ・データで異常を検査する頻度を制御します。

構文

store anomaly-detection poll <n>

表示コマンド

show anomaly-detection poll

store anomaly-detection state

異常検出サブシステムを有効または無効にします。このサブシステムには、すべてのアクティブな統計アラートを実行し、ログで異常を検査し、アラート機能サブシステムの必要に応じてアラートをキューに入れる機能があります。

構文

store anomaly-detection state <on | off>

表示コマンド

show anomaly-detection state

store alerter smtp authentication password

アラート機能 SMTP 認証パスワードを、value で指定する値に設定します。 対応する show コマンドはありません。

構文

store alerter smtp authentication <value>

store alerter smtp authentication type

SMTP サーバーが必要とする認証タイプを、以下のいずれかの値に設定します。

none: 認証なしで送信。

auth: ユーザー名/パスワードでの認証。 使用する場合、以下のコマンドを使用してユーザー・アカウントおよびパスワードを設定してください。

store alerter smtp authentication username

store alerter smtp authentication password

構文

store alerter smtp authentication type <none | auth>

表示コマンド

show alerter smtp authentication type

store alerter smtp authentication username

アラート機能 SMTP E メール認証ユーザー名を、name で指定する値に設定します。

構文

store alerter smtp authentication username <name>

表示コマンド

show alerter smtp authentication username

store alerter smtp port

SMTP サーバーで listen するポート番号を、n で指定する値に設定します。 デフォルトは 25 (標準 SMTP ポート) です。

構文

store alerter smtp port <n>

表示コマンド

show alerter smtp port

store alerter smtp relay

Guardium アプライアンスが使用する SMTP サーバーの IP アドレスを設定します。

構文

store alerter smtp relay <ip address>

表示コマンド

show alerter smtp relay

store alerter smtp returnaddr

E メール・アラート返信用の E メール・アドレスを設定します。 送り返されたメッセージや E メールの障害はすべてこのアドレスに返信されます。

構文

store alerter smtp returnaddr <email address>

表示コマンド

show alerter smtp returnaddr

store alerter snmp community

アラート機能が使用する SNMP トラップ・コミュニティーを、name で指定する値に設定します。 対応する show コマンドはありません。

構文

store alerter snmp community <name>

store alerter snmp secondary_community

アラート機能が使用する 2 次 SNMP トラップ・コミュニティーを、name で指定する値に設定します。対応する show コマンドはありません。

構文

store alerter snmp secondary_community <string>

ここで、string はコミュニティーのテキスト文字列です。

store alerter smtp traphost

アラートを受信するアラート機能 SNMP トラップ・サーバーを、指定する IP アドレスまたは DNS ホスト名に設定します。

構文

store alerter snmp traphost <snmp host>

表示コマンド

show alerter snmp traphost

store alerter snmp secondary_traphost

アラートを受信するアラート機能 2 次 SNMP トラップ・サーバーを、指定する IP アドレスに設定します。

構文

store alerter snmp secondary_traphost <arg>

ここで、<arg> は 2 次 SNMP サーバーの IP アドレス、または値をリセットする単語「null」です。

表示コマンド

show alerter snmp secondary_traphost

store syslog-trap

Usage: store syslog-trap ON | OFF

親トピック: CLI の概要

証明書 CLI コマンド

証明書コマンドを使用して、証明書署名要求 (CSR) の作成、および、Guardium® システム上へのサーバー証明書、CA (認証局) 証明書、またはトラステッド・パス証明書のインストールを行います。

注: Guardium は、認証局 (CA) サービスは提供しません。また、デフォルトでインストールされているもの以外の証明書をシステムに添付することもありません。ご使用のサイトに独自の証明書が必要な場合は、サード・パーティー CA (VeriSign や Entrust など) に問い合わせる必要があります。

証明書の有効期限

証明書の有効期限が切れると機能が失われます。show certificate warn_expire コマンドを定期的に実行して、証明書の有効期限が切れていないか確認してください。コマンドにより、6 カ月以内に有効期限が切れるか既に有効期限が切れた証明書が表示されます。ユーザー・インターフェースでも、有効期限切れが迫っている証明書に関する通知が示されます。すべての証明書の要約を表示するには、show certificate summary コマンドを実行します。

新規証明書

新規証明書を取得するには、証明書署名要求 (CSR) を生成し、VeriSign や Entrust などのサード・パーティー認証局 (CA) に問い合わせてください。Guardium は CA サービスは提供しません。また、デフォルトでインストールされているもの以外の証明書をシステムに添付することもありません。証明書の書式は PEM で、BEGIN および END の区切り文字を含む必要があります。証明書は、コンソールから貼り付けるか、標準のインポート・プロトコルのいずれかを介してインポートすることができます。

注: このアクションは、システム・ネットワーク構成パラメーターの設定が終了するまで実行しないでください。

create csr

Guardium システム用の証明書署名要求 (CSR) を作成します。 CSR の作成は、システム・ネットワーク構成パラメーターの設定が終了するまで実行しないでください。生成された CSR の中に、割り当てられたホスト名とドメイン名に基づく共通名 (CN) が自動作成されます。

パラメーター

create csr alias

別名を使用した認証要求を作成します。

create csr external_stap

Guardium External S-TAP Docker コンテナー用の認証要求を作成します。証明書が署名および保管されると、External S-TAP をデプロイして、クラウド内のデータベースや、ローカル・エージェントを使用できないその他の状況にあるデータベースからのトラフィックをモニターすることができます。

create csr gim

gim (GIM リスナー) 用の認証要求を作成します。

create csr gui

tomcat 用の認証要求を作成します。

create csr sniffer

スニファー用の認証要求を作成します。

構文

create csr <alias | external_stap | gim | gui | sniffer>

delete certificate

指定された SSL 証明書を削除します。

パラメーター

delete certificate external_stap

External S-TAP 用のすべての使用可能な証明書の別名を表示して、削除する証明書を選択するようプロンプトを出します。

構文

delete certificate external_stap

restore certificate gim

証明書 gim をレコード上の最新の証明書 gim または最初に提供されたデフォルトの証明書 gim に復元します。

パラメーター

restore certificate gim backup

gim 証明書を最後に保存されたスニファー gim 証明書に復元します。

restore certificate gim default

gim 証明書をシステムに提供されたデフォルトの gim 証明書に復元します。

構文

restore certificate gim <backup | default>

restore certificate keystore

証明書鍵ストアをレコード上の最新の証明書鍵ストアまたは最初に提供されたデフォルトの証明書鍵ストアに復元します。

パラメーター

restore certificate keystore backup

証明書鍵ストアを最後に保存された証明書鍵ストアに復元します。

restore certificate keystore default

証明書鍵ストアをシステムに提供されたデフォルト値に復元します。

構文

restore certificate keystore <backup | default>

restore certificate mysql

クライアント証明書をレコード上の最新の証明書に復元します。

パラメーター

restore certificate mysql backup

最後に保存された mysql 証明書を復元します。

構文

restore certificate mysql <backup>

restore certificate mysql backup client

クライアント証明書をレコード上の最新の証明書に復元します。

パラメーター

restore certificate mysql backup client ca

最後に保存されたクライアント認証局 (CA) 証明書を復元します。

restore certificate mysql backup client cert

最後に保存されたクライアント証明書を復元します。

構文

restore certificate mysql backup client <ca | cert>

restore certificate mysql backup server

サーバー証明書をレコード上の最新の証明書に復元します。

パラメーター

restore certificate mysql backup server ca

最後に保存されたサーバー認証局 (CA) 証明書を復元します。

restore certificate mysql backup server cert

最後に保存されたサーバー証明書を復元します。

構文

restore certificate mysql backup server <ca | cert>

restore certificate mysql default client

mysql クライアント証明書をシステムに提供されたデフォルト・バージョンに復元します。

パラメーター

restore certificate mysql default client ca

mysql クライアント ca 証明書をシステムに提供されたデフォルト・バージョンに復元します。

restore certificate mysql default client cert

mysql クライアント証明書をシステムに提供されたデフォルト・バージョンに復元します。

構文

restore certificate mysql default client <ca | cert>

restore certificate mysql default server

mysql サーバー証明書をシステムに提供されたデフォルト・バージョンに復元します。

パラメーター

restore certificate mysql default server ca

mysql サーバー ca 証明書をシステムに提供されたデフォルト・バージョンに復元します。

restore certificate mysql default server cert

mysql サーバー証明書をシステムに提供されたデフォルト・バージョンに復元します。

構文

restore certificate mysql default server <ca | cert>

restore certificate sniffer

証明書をレコード上の最新の証明書に復元します。

パラメーター

restore certificate sniffer backup

スニファー証明書を最後に保存されたスニファー証明書に復元します。

restore certificate sniffer default

スニファー証明書をデフォルト・スニファー証明書に復元します。

構文

restore certificate sniffer <backup | default>

restore cert_key mysql backup

mysql クライアント認証鍵またはサーバー認証鍵を最後に保存された値に復元します。

パラメーター

restore cert_key mysql backup client

最後に保存された mysql クライアント認証鍵を復元します。

restore cert_key mysql backup server

最後に保存された mysql サーバー認証鍵を復元します。

構文

restore cert_key mysql backup <client | server>

restore cert_key mysql default

mysql クライアント認証鍵またはサーバー認証鍵を、システムに提供されたデフォルト・バージョンに復元します。

パラメーター

restore cert_key mysql default client

システムに提供されたデフォルトの mysql クライアント認証鍵を復元します。

restore cert_key mysql default server

システムに提供されたデフォルトの mysql サーバー認証鍵を復元します。

構文

restore cert_key mysql default <client | server>

show certificate

すべての証明書の要約、証明書情報、別名リスト、鍵ストア内の証明書、有効期限が切れたあるいは間もなく切れる証明書を表示します。

この認証局は、Guardium CA 公開鍵で検証可能です (公開鍵は、クライアント・ソフトウェアと共に配布される CA 証明書に含まれています)。証明書は、顧客企業固有の CN (共通名 - 例えば、acme.com) またはマシン固有の CN (例えば、x4.acme.com) のどちらかを保持します。これによってクライアントは、Guardium システムが有効な証明書を持っている (つまり正式な Guardium システムである) ことだけでなく、それが、クライアントが接続を意図している特定の Guardium システム (または Guardium システムのセット) であることも確証できます。

パラメーター

show certificate all

すべての証明書の要約を表示します。

show certificate alias

別名リストを表示します。

show certificate external_stap

External S-TAP 証明書の概要 (証明書情報、別名、鍵ストア内の証明書、有効期限が切れた証明書または間もなく切れる証明書など) を表示します。

show certificate gim

すべての GIM 証明書情報 (GIM リスナー) を表示します。

show certificate gui

すべての tomcat 証明書情報を表示します。

show certificate keystore

鍵ストア内のすべての証明書と、表示する証明書をユーザーが選択するための別名リストを表示します。

show certificate mysql

クライアントおよびサーバーの mysql 証明書情報を表示します。

show certificate sniffer

すべてのスニファー証明書情報を表示します。

show certificate stap

鍵ストア内のすべての S-TAP 証明書情報を表示します。

すべての external_stap サーバー証明書情報を表示します。

show certificate summary

すべての証明書情報の要約を表示します。

show certificate trusted

すべてのトラステッド証明書情報を表示します。

show certificate warn_expired

有効期限が切れたすべての証明書または 6 カ月以内に有効期限が切れる証明書を表示します。

構文

show certificate <alias | all | external_stap | gim | gui | keystore | mysql | sniffer | stap | summary | trusted | warn_expired >

show certificate keystore

鍵ストア内の証明書情報を表示します。

パラメーター

show certificate keystore all

鍵ストア内のすべての証明書を表示します。

show certificate keystore alias

表示する証明書をユーザーが選択するための別名リストを表示します。

構文

show certificate keystore <all | alias>

show certificate mysql

mysql 証明書情報を表示します。

パラメーター

show certificate mysql client

クライアント mysql 情報を表示します。

show certificate mysql server

サーバー mysql 情報を表示します。

構文

show certificate mysql <client | server>

store certificate

証明書を保管します。証明書を PEM 形式で貼り付け、BEGIN および END 行を追加します。

パラメーター

store certificate alias

CSR が生成された後に証明書を鍵ストアに保管します。この CLI コマンドは、ユーザーが中間のトラステッド証明書を最初から作成することを可能にする CLI コマンド create csr alias をサポートします。これらの両方のコマンドを使用して、中間のトラステッド証明書を作成します。これらの中間のトラステッド証明書は、必要に応じて他の証明書を後で署名するためにも使用できます。

store certificate external_stap

External S-TAP 用に CSR が生成された後にプロンプトが出されたら、SSL 証明書を鍵ストアに保管します。

store certificate gim

証明書、鍵 (オプション)、および CA 証明書 (GIM リスナー) を求めるプロンプトを出すことによって、カスタム gim 証明書を鍵ストア内に保管することができます。

store certificate gui

CSR が生成された後に鍵ストア内に Tomcat 証明書を保管します。

store certificate keystore

トラステッド証明書を一意的に識別するための 1 単語の別名を尋ね、別名を鍵ストア内に保管します。

store certificate keystore_external_stap

External S-TAP 証明書の署名に使用されるルートおよび中間信頼証明書を保管します。

store certificate mysql

mysql クライアント証明書およびサーバー証明書を保管します。

store certificate privatekey gim

GIM 自己署名証明書および秘密鍵を鍵ストアに保管します。

store certificate privatekey gui

GUI 自己署名証明書および秘密鍵を鍵ストアに保管します。

store certificate sniffer

スニファー証明書を保管します。

store certificate stap

S-TAP 証明書を保管します。

構文

store certificate <alias | external_stap | gim | gui | keystore | keystore_external_stap | mysql | sniffer | stap >

store certificate mysql client

mysql クライアント証明書を保管します。

パラメーター

store certificate mysql client ca

クライアント認証局 (CA) 証明書を保管します。

store certificate mysql client cert

クライアント証明書を保管します。

構文

store certificate mysql client <ca | cert>

store certificate mysql server

mysql サーバー証明書を保管します。

パラメーター

store certificate mysql server ca

サーバー認証局 (CA) 証明書を保管します。

store certificate mysql server cert

サーバー証明書を保管します。

構文

store certificate mysql server <ca | cert>

store cert_key

システム認証鍵と mysql クライアントおよびサーバーの認証鍵を保管します。

パラメーター

store cert_key mysql

mysql クライアントおよびサーバーの認証鍵を保管します。

store cert_key sniffer

スニファー認証鍵を保管します。

構文

store cert_key <mysql | sniffer>

store cert_key mysql

mysql クライアントまたはサーバーの認証鍵を保管します。

パラメーター

store cert_key myself client

mysql クライアントの認証鍵を保管します。

store cert_key myself server

mysql サーバーの認証鍵を保管します。

構文

store cert_key sniffer

システム認証鍵を保管します。このコマンドにより、Guardium システムが (S-TAP® との通信に) 使用するシステム証明書を設定できます。証明書は、コンソールから貼り付けるか、標準インポート・プロトコルのいずれかを介してインポートすることができます。証明書の形式は PEM で、BEGIN および END の区切り文字を含む必要があります。この証明書は、guardium_ca_path を通じて S-TAP ソフトウェアから自己署名証明書が使用可能な CA によって署名されている必要があります。

パラメーター

store cert_key sniffer console

鍵をコンソールに貼り付けることによってスニファー認証鍵を保管します。

store cert_key sniffer import

鍵ファイルをインポートすることによってスニファー認証鍵を保管します。

構文

store cert_key sniffer <console | import>

バックアップおよびデフォルトのオプション

バックアップまたはデフォルトのパラメーターを使用して、証明書および認証鍵を復元することを選択できます。証明書を最後に保存された証明書に復元するには、バックアップ・パラメーターを使用します。証明書を Guardium によって提供された元の証明書に復元するには、デフォルト・パラメーターを使用します。

証明書の有効期限および要約コマンド

show certificate warn_expire

コマンドを定期的に実行してください。 このコマンドは、6 カ月以内に有効期限が切れる証明書について警告を出し、有効期限が切れた証明書のリストを表示します。詳細については、show certificate CLI コマンドを参照してください。すべての証明書の要約を表示するには、CLI コマンド show certificate summary を実行します。 コマンドを定期的に実行して、証明書の有効期限を確認してください。

親トピック: CLI の概要

構成および制御 CLI コマンド

構成および制御用に、以下の CLI コマンドを使用します。

? (疑問符)

コマンドを入力するとき、任意の時点で疑問符を入力すると、引数が表示されます。

構文

<コマンドの一部> ?

例

CLI> show account strike ?

使用法: show account strike <arg> ここで、arg は以下のとおりです。

?, count, interval, max

ok

CLI>

clean load_balance_inactive_stap_queue

このコマンドを使用して、非アクティブな S-TAP とその対応するコレクターをロード・バランサー内の非アクティブな S-TAP キューから手動でクリアします。

構文

clean load_balance_inactive_stap_queue <stapHost> <collectorName>

delete unit type

このコマンドを使用して、1 つ以上のユニット・タイプ属性を消去します。なお、このコマンドを使ってすべてのユニット・タイプ属性を消去できるわけではないことに注意してください。詳しくは、store unit type コマンドの後にある表を参照してください。

構文

delete unit type [manager | standalone] [aggregated] [netinsp] [network routes static] [stap] [mainframe]

commands

すべての CLI コマンドをアルファベット順のリストで表示します。

構文

commands

debug

デバッグ・モードを有効/無効にします。引数を指定しない場合、デバッグ状態が切り替わります。オプションで、状態を指定する引数を渡すことができます。

構文

debug <on | off>

eject

このコマンドは CD-ROM を取り外してイジェクトします。これは CD-ROM で配布されたパッチのインストール、またはシステムのアップグレード/再インストールの後で役立ちます。

構文

eject

delete scheduled-patch

パッチ・インストール要求を削除するには、CLI コマンド delete scheduled-patch を使用します。

パッチ・インストールの詳細については、CLI コマンド store system patch install を参照してください。

forward support email

サポート状態オプションが有効の場合 (これがデフォルトです)、このコマンドはシステム・アラートを受信する E メール・アドレスを設定します。

構文

forward support email to <email address>

表示コマンド

show support-email

iptraf

IPTraf は、基礎となるオペレーティング・システムと共に配布されるネットワーク統計ユーティリティーです。これは TCP 接続のパケットとバイトの数、インターフェースの統計とアクティビティーの指標、TCP/UDP トラフィック明細、LAN ステーションのパケットとバイトの数など、さまざまな情報を収集します。 IPTraf ユーザー・マニュアルは、インターネットの以下のロケーションで入手可能です (このリンクが機能しない場合、他のロケーションで入手できる可能性があります):

http://iptraf.seul.org/2.7/manual.html

構文

iptraf

license check

インストール済みライセンスが有効であるかどうかを示します。新しいプロダクト・キーをインストールした後、このコマンドを使用します。

構文

license check

ping

ICMP ping パケットをリモート・ホストに送信します。ネットワーク接続を検査するには、このコマンドが役立ちます。host の値は、IP アドレスまたはホスト名のいずれかです。

構文

ping <host>

quit

コマンド行インターフェースを終了します。

構文

quit

recover failed

失敗した CSV/CEF/PDF 転送ファイルを復元するコマンドです。別のエクスポート試行で使用できるように、ファイルを元のエクスポート・フォルダーに入れます。

構文

recover failed [csv|cef|pdf]

register management

指定された中央マネージャーによる管理の Guardium システムを登録します。この Guardium システムの事前登録構成は保存されます。後でユニットが登録抹消された場合には、この構成が復元されます。

構文

register management <manager ip> <port>

パラメーター

manager ip は中央マネージャーの IP アドレスです。

port は中央マネージャーによって使われるポート番号です (通常は 8443)。

restart gui

IBM® Guardium® Web インターフェースを再始動します。オプションで、GUI の再始動を 1 日に一度 (または週に一度) スケジュールするには、追加のパラメーターを使用します。HH は時間 (01 から 24)、MM は分 (01 から 60) です。W は曜日 (0 から 6) で、日曜日が 0 です。HHMM が 2 度リストされている場合、最後の項目だけが使用されます。パラメーター clear は、スケジュール済みの時間を削除します。

分類およびセキュリティー・アセスメント・プロセスを再始動するには、(GUI からではなく) CLI から restart gui コマンドを実行します。

GUI からの restart GUI の実行は Web サービスだけを再始動させます。分類およびセキュリティー・アセスメント・プロセスを含む、すべてのプロセスを完全に再始動するには、CLI から restart GUI コマンドを実行する必要があります。分類リスナーを再始動するには、管理対象ユニットごとに CLI から restart GUI コマンドを実行する必要があります。

構文

restart gui [HHMM|HHMMW|clear]

restart stopped_services

store auto_stop_services_when_full CLI コマンドで以前に停止したサービスを再始動するには、この CLI コマンドを使用します。

構文

restart stopped_services

restart system

Guardium システムをリブートします。 システムは完全にシャットダウンして再始動します。つまり cli セッションが終了します。

構文

restart system

show buffer

このコマンドは、検査エンジン・プロセスに関するバッファー使用状況のレポートを表示します。ロードで問題が発生する場合、このコマンドを実行するよう IBM 技術サポートから要請されることがあります。

構文

show buffer <log | snif>

show buffer log

この CLI コマンドを使用して、検査エンジン・プロセスのバッファーの使用状況を表示します。

show buffer snif

この CLI コマンドを使用して、スニファーのバッファーの使用状況を表示します。

show build

インストール済みソフトウェアのビルド情報を表示します (ビルド、リリース、スニフ・バージョン)。

構文

show build

show defrag

断片化したパケットを識別して、それらがネットワーク・スニッフィング・プロセスに到達する前にパケットの再構成を試みます。デフラグは SPAM または TAP デバイスを介したネットワーク・スニッフィングにのみ関連があります。

構文

show defrag

パラメーター

Packet size- パケット・サイズ。バイト単位で、最大 217 (131072)

時間間隔 - 時間間隔

トリガー・レベル - トリガー・レベル。

リリース・レベル - 秒数で指定されるリリース・レベル。最大で 2 の 31 乗 (2147483648)。

show load_balance_inactive_stap_queue

このコマンドは、ロード・バランサー内の非アクティブな S-TAP キューに累積した非アクティブな S-TAP および対応するコレクターのリストを表示します。

構文

show load_balance_inactive_stap_queue

show network routes static

ユーザーに対し、所有する IP アドレスが 1 アプライアンスにつき 1 つだけ (eth0 を通じて) であっても、静的ルーティング表を使用することにより、異なるルーターからの直接トラフィックを許可します。 現在の静的ルートとその ID をリストします。

構文

show network routes static

削除コマンド

delete network routes static

show password

この CLI コマンドはパスワード機能を表示します。password disable [0|1] は、値 1 を保管することによりパスワードの使用を解除します。password expiration [CLI|GUI] [日数] はパスワード変更が要求される間隔 (日数) を表示します。デフォルトは 90 日です。password validation [ON|OFF] はパスワードに必要な強さを指定します。

構文

show password disable [0|1]

show password expiration [CLI|GUI] 90

show password validation [ON|OFF]

show security policies

セキュリティー・ポリシーのリストを表示します。

構文

show security policies

show system patch available

既にインストール済みのパッチ、およびインストールするようスケジュールされたパッチを表示します。日時とインストール状況を示します。

構文

show system patch installed

show system patch installed

既にインストール済みのパッチ、およびインストールするようスケジュールされたパッチを表示します。日時とインストール状況を示します。

構文

show system patch installed

show system public key

cli または tomcat 用の公開鍵を表示します。存在しない場合、このコマンドはそれを作成します。

注: 証明書 CLI コマンドの中の show system key、store system key を参照してください。

構文

show system public key <cli | tomcat | grdapi>

stop gui

Web ユーザー・インターフェースを停止します。

構文

stop gui

stop system

アプライアンスを停止して電源を遮断します。

構文

stop system

store apply_user_hierarchy

ユーザー階層を監査受信者に適用するには、この CLI コマンドを使用します。

ON の場合、非監査グループ受信者 (監査グループ受信者以外の受信者 (通常またはロール)) には、受信者の階層 (受信者を含む) 以下のグループ IP に関連する監査結果のみ表示されます。

構文

store apply_user_hierarchy [ON | OFF]

表示コマンド

show apply_user_hierarchy

store alert_timestamp_unit [millisecond | second]

syslog アラートのタイム・スタンプの単位を制御します。デフォルトは秒です。

構文

store alert_timestamp_unit millisecond

store alert_timestamp_unit second

表示コマンド

show alert_timestamp_unit

store allow_simulation

アプライアンスでのポリシー・シミュレーション実行機能を有効 (on) または無効 (off) にします。

シミュレーションを実行するには、ルール・エンジンを介して元のトラフィックを (テストする必要のあるポリシーで) リプレイする必要があります。そのためには、アプライアンス上の元の SQL とその値を部分的に保存する必要があります。allow_simulation を有効にすると、IBM Guardium は SQL や値を保存します。無効にすると、保存しません。

構文

store allow_simulation [on|off]

表示コマンド

show allow_simulation

store alp_throttle

この CLI を使用して、ログに記録されるデータの量を制御します。

使用法: store alp_throttle <num>

ここで、<num> は -2147483647 から 2147483647 までの範囲の数値です。

デフォルトは 0 です。

0 - GDM_FLAT_LOG にログを記録せず、tapks ファイルを作成しません

>0 - GDM_FLAT_LOG にログを記録し、tapks ファイルを作成しません

<0 - GDM_FLAT_LOG にログを記録し、tapks ファイルを作成します

99999 - GDM_FLAT_LOG にログを記録しませんが、tapks ファイルを作成します

例

10 - ステートメントの 10% のログを GDM_FLAT_LOG に記録します。

10 - ステートメントの 10% のログを GDM_FLAT_LOG に記録し、tapks ファイルを作成します

store analyzer

セッションを無視: 現行の要求およびセッションの残りが無視されます。このアクションは、ポリシー違反をログに記録しますが、構成体のロギングを停止し、セッションの残りに対していかなるタイプのポリシー違反もテストしません。 このアクションは、例えば、データベースにテスト領域が含まれ、データベースのその領域に対してポリシー・ルールを適用する必要がない場合などに役立ちます。

このコマンドは「セッションを無視」のタイムアウト値を設定して、「セッションを無視」の期間を設定します。

構文

store analyzer [ignore_sess_timeout | max_open_sess]

表示コマンド

show analyzer

store auto_stop_services_when_full

ON にすると、データベースの充満率がしきい値 90% を超えた場合に内部サービスを停止します。

検査エンジン、分類、その他の収集関連サービスが停止します。また、統合のインポートと復元では新しいファイルが処理されなくなります。

修正を行うには、さまざまなサポート・コマンド (support clean audit_task、support clean log_files、support clean DAM_data、support show large_files) を使って分析し、大きな表を手動でパージしてください。

構文

store auto_stop_services_when_full [ON | OFF]

表示コマンド

show auto_stop_services_when_full

store connect oracle_parser

このコマンドを使用して、Db2 パーサーから Oracle パーサーへの接続と、接続の切断を行います。デフォルトは OFF (切断) です。

構文

store connect oracle_parser [ON | OFF]

使用法: store connect_oracle_parser [state]。state は、ON または OFF です。ON は接続、OFF は切断です。

表示コマンド

show connect oracle_parser

store csv_fetch_size

CSV_FETCH_SIZE および CSV_MAX_SIZE は、CLI を介してのみ変更できる GLOBAL_PROFILE パラメーターです。

Guardium レポートは、CSV ファイル形式でダウンロードできます。

CSV_MAX_SIZE は、レポート・エクスポート・メニューから「レコードをすべてダウンロード」をクリックすると取得される CSV ダウンロードのサイズを制御するために使用されます。

CSV_FETCH_SIZE は、レコードの合計数を制御するために、レポート REST サービスによって使用されます

注: csv_max_size は、変更を有効にするために GUI を再始動する必要があります。csv_fetch_size は、変更を有効にするために GUI を再始動する必要はありません。

表示コマンド

CLI> show csv_fetch_size

使用法

CLI> store csv_fetch_size

使用法: store csv_fetch_size <number>

ここで、number は 0 より大きい数値です。

store csv_max_size

CSV_FETCH_SIZE および CSV_MAX_SIZE は、CLI を介してのみ変更できる GLOBAL_PROFILE パラメーターです。

Guardium レポートは、CSV ファイル形式でダウンロードできます。

CSV_MAX_SIZE は、レポート・エクスポート・メニューから「レコードをすべてダウンロード」をクリックすると取得される CSV ダウンロードのサイズを制御するために使用されます。

CSV_FETCH_SIZE は、レコードの合計数を制御するために、レポート REST サービスによって使用されます

注: csv_max_size は、変更を有効にするために GUI を再始動する必要があります。csv_fetch_size は、変更を有効にするために GUI を再始動する必要はありません。

表示コマンド

CLI> show csv_max_size

使用法

CLI> store csv_max_size

使用法: store csv_max_size <number>

ここで、number は 0 より大きい数値です。

store default_queue_size

この CLI コマンドを使用して、構成パラメーター ADMINCONSOLE_PARAMETER.DEFAULT_QUEUE_SIZE を制御します。 デフォルトは 25 です。値の範囲は 25 から 300 です。

値を変更したら、スニファーを再始動する必要があります。

構文

store default_queue_size <N>。N は、25 から 300 までの範囲の数値です。

表示コマンド

show default_queue_size 25

store defrag

このコマンドを使用すると、デフラグのデフォルトを復元したり、デフラグ・サイズを設定したりすることができます。このコマンドを入力した後、変更内容を有効にするには restart inspection-core コマンドを発行する必要があります。デフラグは SPAM または TAP デバイスを介したネットワーク・スニッフィングにのみ関連があります。

構文

store defrag [default | size <s> interval <i> trigger <t> release <r>]

表示コマンド

show defrag

パラメーター

default - デフォルト・サイズを復元します。

s - パケット・サイズ。バイト単位で、最大 2^17 (131072)

i - 時間間隔

t - トリガー・レベル

r - 秒数で指定されるリリース・レベル。最大で 2 の 31 乗 (2147483648)。

store delayed_firewall_correlation

この CLI コマンドを使用して、暗号化解除の相関が行われるまでユーザー接続を保留します。

構文

store delayed_firewall_collection [on | off]

表示コマンド

show delayed_firewall_correlation

store full-bypass

このコマンドは緊急用です。Guardium システムによってトラフィックが予期せずブロックされている場合にのみ、これを使用します。これを on にすると、すべてのネットワーク・トラフィックはシステムを直接通過するようになり、Guardium システムからは「認識」できません。

このコマンドを使用するときには、admin ユーザー・パスワードを求められます。

構文

store full-bypass <on | off>

store gdm_analyzer_rule

アナライザー・ルール - いくつかのルールをアナライザー・レベルで適用することができます。アナライザー・ルールには、例えば、ユーザー定義の文字セット、ソース・プログラムの変更、ファイアウォールの監視モードや非監視モードなどがあります。以前のリリースでは、ポリシーやルールは、ロギング状態での要求処理の最後に適用されていました。これは、場合によっては、それらのルールに基づく決定に遅れが生じることを意味していました。アナライザー・レベルでルールを適用するということは、より早い段階で決定を行えることを意味します。

注: ソース・プログラムの変更に関するアナライザー・ルールを適用する場合、 ソース・プログラムがパターンに完全一致しない場合は、 パターンの末尾に .* を追加して、ソース・プログラムの末尾にスペースがある (ユーザーには見えない) 可能性に対処してください。

構文

store gdm_analyzer_rule [active_flag | new]

store gdm_analyzer_rule active_flag

使用法: store gdm_analyzer_rule active_flag <id> <on|off>

ここで <id> は、ルール ID です。

GDM アナライザー・ルールのリストを表示するには、CLI コマンド show gdm_analyzer_rule を使用してください。

store gdm_analyzer_rule new

ルールの説明を入力します (オプション)。

ルールのタイプを入力します (必須)。

表示コマンド

show gdm_analyzer_rule

store gdm_analyzer_rule new

Guardium CLI を使用して、直接正規表現のアナライザー・ルールをマスク UID チェーン・パターンに追加します。

CLI> store gdm_analyzer_rule new

ルールの記述を入力してください: new rule 4

ルール・タイプ:

1. ソース・プログラムを変更する

2. 代替文字セットを設定する

3. 判定を送信する

4. HADOOP の除外

5. プロトコルとポートを定義する

6. パケット後のセッションを無視する

7. ログイン情報が欠落している場合に、空の Oracle DB ユーザーを設定する

8. MS SQL ログインを強制する

9. 文字列を変換する

ルール・タイプを選択してください (必須): 9

パターンを入力してください (必須、正規表現文字列): (.*)(-ppassword)(.*)

形式を入力してください (必須、正規表現文字列): ¥¥¥1-p****¥¥¥3

ルールを今すぐアクティブ化しますか? (はい/いいえ)

Y

ok

store gdm_http_session_template

この CLI コマンドを使用して、HTTP セッションのテンプレートを設定します。

使用法

```
store gdm_http_session_template [activate] [add] [deactivate] [remove]
```

表示コマンド

```
show gdm_http_session_template
```

テンプレート情報の取得を試行します。時間がかかる場合があります。お待ちください。

表 1. store gdm_http_session_template

ID#	アクティブな URL の正規表現	セッションの正規表現	ユーザー名正規表現	Login_Session の正規表現	コメント	Logout_Session_ID	Logout_URL_Regex
1	1	Cookie.*PHPSESSID=([[:a	.*user_name=([[:alnum:]	Set-Cookie:.*PHPSESSID=	削除される HTTP セッションの例		
2	1	Cookie.*PSJSESSIONID=([	.*SignOnDefault=([[:aln		HTTP セッションの例	cmd=logout	
3	1	Cookie.*JSESSIONID=([0-	.*username=([[:alnum:]]	Set-Cookie:.*JSESSIONID	HTTP セッションの例		Logout.jsp

外部ログの保管

このコマンドを使用して、外部ログのファイル・サイズ、フラッシュ期間、gdm エラーおよび状態を設定します。

このルールは、以下の CLI コマンドが実行される場合のみ表示されます。

store log external state on

そして外部ログは、ポリシー・アクションとして表示されます

状態をチェックするための CLI コマンド:

show log external state

このアクションを有効および無効にするための CLI コマンド:

store log external state on/off

使用法

```
store log external [file_size] [flush_period] [gdm_error] [state]
```

使用法: store log external gdm_error <state>

where state is on/off. 'on' is to enable and 'off' is to disable.

使用法: store log external file_size <num>

ここで、<num> はファイルのサイズです。

デフォルトは 4096 バイトです。

使用法: store log external flush_period <num>

ここで、<num> はフラッシュ期間です。

デフォルトは 60 秒です。

使用法: store log external state <state>

where state is on/off. 'on' is to enable and 'off' is to disable.

表示コマンド

```
show log external [file_size] [flush_period] [gdm_error] [state]
```

store monitor gdm_statistics

この CLI コマンドは、ユニット使用状況に関する情報を取得するために使用します。デフォルトは 1 (1 時間おきにスクリプトを実行する) です。

構文

```
CLI> store monitor gdm_statistics
USAGE: store monitor gdm_statistics <hour>, where hour is value from 0 to 24.
       Default value is 1, means to run the script every hour.
       Value 0, means not to run the script.
```

表示コマンド

CLI> show monitor gdm_statistics

gdm_statistics モニターを無効にします。

store gui

store gui [port | session_timeout | csrf_status]

IBM Guardium アプライアンス管理インターフェースで接続を受け入れる TCP/IP ポート番号を設定します。デフォルトは 8443 です。**n** は 1024 から 65535 までの範囲の値でなければなりません。別の目的で必須であるか使用中のポートを使用しないようにしてください。

セッションのタイムアウトの設定 - アクティビティーのない状態が何秒続いたら、タイムアウトにするかを指定します。 アクティビティーがないためタイムアウトに達した場合、IBM Guardium に再びログオンする必要があります。デフォルトの長さは 900 秒 (15 分) です。

Cross-site Report Forgery (CSRF) (ON | OFF) の設定 - 『GUI の概要』ヘルプ・トピックで、『**CSRF および 403 アクセス許可エラー**』のセクションを参照してください。アップグレード済みのシステムでは、デフォルト値が有効になっています。特定の Web ブラウザー機能 (例えば、F5/CTRL-R/最新の情報に更新/再読み込みや、戻る/進む) の使用を試みると、403 アクセス許可エラー・メッセージが出されます。

新しいセッション・タイムアウト値は、次回の GUI 再始動後に初めて有効になります。

構文

store gui port <n>

store gui session_timeout <n>

store gui csrf_status [on | off]

表示コマンド

GUI ポート番号、状態、セッション・タイムアウト (秒)、または CSRF 状況の一部または全部を表示します。

構文

show gui [port | state | all | session_timeout | csrf_status]

store gui cache

この CLI コマンドを使用して、Web ブラウザーのキャッシングをオンまたはオフ (有効または無効) に切り替えます。

応答:

The parameter has been changed.(パラメーターが変更されました。)

Restarting gui (GUI を再始動しています)

Changing to port 8443 (ポート 8443 に変更しています)

Stopping..... (停止しています.....)

Safekeeping xregs

ok

ブラウザーのキャッシングのデフォルト設定は「有効」です。

キャッシュの設定を変更する処理によって、自動的に Guardium Web サーバーが再始動されます。

Firefox の場合は、設定を有効にするために、それぞれのブラウザー上のキャッシュをクリアする必要があります。

構文

store gui cache [ON | OFF]

表示コマンド

show gui cache

store gui session_timeout

タイムアウトになるまでの、アクティビティーのない状態の時間の長さ (秒) を設定します。アクティビティーがないためタイムアウトに達した場合、IBM Guardium に再びログオンする必要があります。デフォルトの長さは 900 秒 (15 分) です。

構文

store gui session_timeout

表示コマンド

show gui session_timeout

store gui csrf_status

この CLI コマンドは、 Cross-site Request Forgery (CSRF) 状況を有効化または無効化するときに使用します。

構文

store gui scrf_status [on | off]

表示コマンド

show gui scrf_status

store gui xss_status

この CLI コマンドは、クロスサイト・スクリプティング (XSS) 状況を有効化または無効化するときに使用します。アップグレード済みのシステムでは、デフォルトでこのオプションは有効になっています。

構文

store gui xss_status [on | off]

表示コマンド

show gui xss_status

store gui hsts_status

この CLI コマンドは、HSTS (HTTP Strict Transport Security Filter) を有効化または無効化するときに使用します。アップグレードされたシステムではこのオプションはデフォルトで無効になっており、有効な証明書がインストールされた後にオンにすることが推奨されています。 詳しくは、トピック『ブラウザーの SSL 証明書チャレンジを回避するためのアプライアンス証明書のインストール方法』を参照してください。

構文

store gui hsts_status [on | off]

表示コマンド

show gui hsts_status

store installed security policy

policy-name で指定されるセキュリティー・ポリシーを、インストール済みセキュリティー・ポリシーとして設定します。

構文

store installed security policy <policy-name>

表示コマンド

show installed security policy

store keep_psmls

この CLI コマンドは、Guardium アプリケーションのユーザーの作成時に使用した現在のレイアウト/プロファイル/ポートレットを保持するときに使用します。この CLI コマンドを ON に設定してからアップグレードを実行すると、旧バージョンの psml が保持されます。

構文

store keep_psmls [ON | OFF]

show keep_psmls

store ldap-mapping

LDAP マッピング・パラメーターを保管します。これにより、LDAP サーバー・スキーマ用のカスタム・マッピングが可能になります。このコマンドは、E メール、ファーストネーム、およびラストネーム属性に関する LDAP サーバー・スキーマへのカスタマイズされたマッピングを可能にします。任意の LDAP サーバー・タイプ (Active Directory、Novell Directory、Open LDAP、Sun One Directory、Tivoli® Directory) の間の転送を可能にするために、paging パラメーターが使用されます。paging パラメーターを on に設定しても、サーバーでページングがサポートされない場合には、ページングなしで検索が実行されます。

ページングの例: CLI コマンド **ldap-mapping paging** が ON に設定された場合、Microsoft Active Directory は LDAP インポート構成画面の制限値で定義された最大数のユーザーをダウンロードします。CLI コマンド **ldap-mapping paging** が OFF に設定された場合、Active Directory は制限の設定値にかかわらず、最大 1000 ユーザーだけをダウンロードします。制限の設定値までユーザーをダウンロードするには、他のすべての LDAP サーバー構成で CLI コマンド **ldap-mapping paging off** を使用する必要があります。

注: CLI ldap-mapping 属性を変更するたびに、更新の前に、IBM Guardium GUI の LDAP インポート構成画面の「既存の変更のオーバーライド」 を選択する必要もあります。CLI ldap-mapping の E メール、ファーストネーム、またはラストネーム属性を変更して LDAP ユーザーをインポートするたびに、この操作が必要です。

表示コマンド

show ldap-mapping [email] [firstname][lastname] <名前>

show ldap-mapping paging ON|OFF

新しいパラメーターを有効にするには、CLI の GUI 再始動が必要です。

例

いくつかの例を示します。

store ldap-mapping firstname name

store ldap-mapping lastname sn

store ldap-mapping email mail

store ldap-mapping paging on

属性が次のように指定されている場合、検出される最初の属性がマッピング・プロセスで使われます。これが適切でない場合は、いずれかの例を使って特定の属性にマップしてください。

firstname 属性の値: gn,givenName,name

lastname 属性の値: attribute: sn,surname,name

email 属性の値: userPrincipalName,mail,email,emailAddress,pkcs9email,rfc822Mailbox

paging の値: on, off

store license

このコマンドは、新規ライセンス・キーをアプライアンスに適用します。

ライセンス・キーには、オーバーライド・タイプと追加タイプの 2 種類があります。オーバーライド・タイプは現在インストールされているライセンスを置換し、追加タイプ・ライセンスは現在インストールされているライセンスに追加されます。 追加タイプ・ライセンスでは、機能の追加以外は行われません。新規機能が使用可能になる場合のほか、関連があれば、有効期限が更新されたり、残りのスキャン数やデータ・ソース数が増えたり、特定のライセンスの数値フィールド (管理対象ユニットの数など) が置換されたりします。

構文

store license

表示コマンド

show license

例

store license コマンドを使用するとき、次のように、新しいプロダクト・キーを貼り付けるよう求められます。

CLI> store license

IBM Guardium から受け取った文字列を貼り付けて、Enter キーを押します。

新しいプロダクト・キーをコピーしてカーソル位置に貼り付けた後、Enter を押します。プロダクト・キーには改行や空白文字が含まれず、常に末尾の等号 (これも含まれます) で終わります。一連のメッセージが表示され、その最後は次のようになります。

We recommend that the machine be rebooted at the earliest opportunity in order to complete the license updating process.

ok

CLI>

この時点で restart gui コマンドを実行します。

store log classifier level

分類のデバッグ・レベルを、表示されるいずれかの値に設定します。

構文

store log classifier level DEBUG|INFO|WARN|ERROR|FATAL

表示コマンド

show log classifier level

store log sql parser_errors

構文的に間違った SQL コマンドのロギングを設定します。

構文

store log sql parser_errors [on|off]

注: 保管コマンドを発行した後、変更内容を適用するには検査エンジンを再始動する必要があります。

表示コマンド

show log sql parser_errors

store log object_join_info

object_join のロギングを設定します。

結合表は、多対多の関係を実装するための 1 つの方法です。結合エンティティーは、SELECT SQL ステートメントで表を結合する場合に使用します。

構文

store log object_join_info [on | off]

表示コマンド

show log object_join_info

store log session_info

スニファー関連

構文

store log session_info [on | off]

表示コマンド

show log session_info

store log exception sql

on に設定された場合、例外のロギング時に SQL コマンド全体をログに記録します。

構文

store log exception sql <on | off>

表示コマンド

show log exception sql

store logging granularity

ロギング細分度を、指定された分数に設定します。構文に示されているいずれかの分の値を使用する必要があります。デフォルトは 60 です。

構文

store logging granularity <1、2、5、10、15、30、または 60>

表示コマンド

show logging granularity

store max_audit_reporting

監査レポートしきい値を表示します。デフォルトは 32 です。監査プロセスでレポートを定義するとき、(FROM-TO フィールドで定義される) レポートの日数は特定のしきい値を超えることができません (デフォルトでは 1 カ月)。この CLI コマンドの使用法について詳しくは、『コンプライアンス・ワークフロー自動化』ヘルプ・トピックの『ワークフロー・プロセス、一元管理および統合』のセクションを参照してください。

構文

store max_audit_reporting

表示コマンド

show max_audit_reporting

store max_result_set_size

max_result_set_size を保管します。このデフォルト値は 100 (サイズの範囲は 1 から 65535) で、戻りデータを監視するときの検査エンジンの調整に役立ちます。このコマンドは、結果セットの合計サイズの制限を設定します。このパラメーターはあらゆる種類のデータベースに対して機能します。この値が、定義済みのしきい値を超える

場合には、アナライザーは影響を受けるレコード値を計算するためにデータを取り出しません。

構文

store max_result_set_size <サイズ>

表示コマンド

show max_result_set_size

store max_result_set_packet_size

max_result_set_packet_size を保管します。このデフォルト値は 32 (サイズの範囲は 1 から 65535) で、戻りデータを監視するときの検査エンジンの調整に役立ちます。このコマンドは、応答のパケット・サイズの制限を設定します。このパラメーターはあらゆる種類のデータベースに対して機能します。この値が、定義済みのしきい値を超える場合には、アナライザーは影響を受けるレコード値を計算するためにデータを取り出しません。

構文

store max_result_set_packet_size <サイズ>

表示コマンド

show max_result_set_packet_size

store max_tds_response_packets

max_tds_response_packets を保管します。このデフォルト値は 5 (サイズの範囲は 1 から 65535) で、戻りデータを監視するときの検査エンジンの調整に役立ちます。このコマンドは、応答でのパケット数の制限を設定します。このパラメーターは MS SQL でのみ機能します。この値が、定義済みのしきい値を超える場合には、アナライザーは影響を受けるレコード値を計算するためにデータを取り出しません。

構文

store max_tds_response_packets <サイズ>

注: max_tds_response_packets (表データ・ストリーム) は MS SQL Server および Sybase だけに適用されます。

表示コマンド

show max_tds_response_packets

store maximum query duration

照会の最大秒数を、**n** で指定された値に設定します。デフォルトは 180 です。この値をデフォルトより大きく設定した場合、照会の処理でシステムが過負荷になる可能性が増すため、そのように設定**しない**ことをお勧めします。なお、管理者ポータルの実行状況モニター・パネルからこの値を設定することもできます。

構文

store maximum query duration <n>

表示コマンド

show maximum query duration

store monitor [buffer | custom_db_usage | gdm_statistics]

この CLI コマンドを使用して、「IBM Guardium モニター」タブのバッファー使用状況モニター・レポート内の表示情報を取り出すスクリプトの実行間隔を設定する monitor buffer を保管します。

構文: store monitor buffer

以下の CLI コマンドを使用して、状態をオンに設定してこのジョブの実行時刻を指定する monitor custom_db_usage を保管します。

構文

```
CLI> store monitor custom_db_usage
USAGE: store monitor custom_db_usage <state> <hour>
where state is on/off.
If state is on, specify the hour to run.
Valid value is number from 0 to 23
```

以下の CLI コマンドを使用して、ユニット使用状況に関する情報を取得する monitor gdm_statistics を保管します。デフォルトは 1 (1 時間おきにスクリプトを実行する) です。

構文

```
CLI> store monitor gdm_statistics
USAGE: store monitor gdm_statistics <hour>, where hour is value from 0 to 24.
       Default value is 1, means to run the script every hour.
       Value 0, means not to run the script.
```

表示コマンド

show monitor buffer

show monitor custom_db_usage

show monitor gdm_statistics

store mysql_utf8mb4

4 バイトの UTF-8 エンコード (utf8mb4) のサポートを有効にします。

このコマンドは、4 バイトの UTF-8 文字を正しくキャプチャーして保管するように、Guardium スニファー・プロセスおよび内部データベースを変更します。 ご使用の環境のデータ・ソースに 4 バイト文字が含まれている場合 (中国語、日本語、および韓国語の表意文字に使用されている場合など)、utf8mb4 が有効であると便利な場合があります。

このコマンドを使用するときは、以下のことを確認してください。

- 4 バイト文字をキャプチャーして保管するために必要な追加の処理は、Guardium システムのパフォーマンスに悪影響を与えます。このため、ご使用の環境で 4 バイト文字のサポートを必要としない限り、utf8mb4 を有効にしないでください。
- 集約された環境または一元管理された環境で 4 バイトの UTF-8 エンコードをサポートする必要がある場合は、環境内のすべての Guardium システムで utf8mb4 を有効にする必要があります。環境内の一部のシステムでのみ utf8mb4 を有効にすると、集約が失敗したりレポートが正しく表示されないなど、問題が発生することがあります。
- utf8mb4 を有効にする前に収集されたデータや集約されたデータは、utf8mb4 を有効にした後も引き続き使用可能であり、正しく機能します。

注意:
store mysql_utf8mb4 コマンドを使用して 4 バイトの UTF-8 のサポートを有効にした後で、変更を元に戻すことはできません。Guardium システムで utf8mb を有効にした後、4 バイトの UTF-8 文字のサポートを除去する唯一の方法は、システムを完全に再構築することです。

構文

store mysql_utf8mb4

表示コマンド

show mysql_utf8mb4

例

```
> show mysql_utf8mb4
mysql configuration NOT set with UTF8MB4.
ok

> store mysql_utf8mb4
Attempting to change the mysql config file. It may take time. Please wait.
Start to modify mysql config file
Restarting mysql
Mysql has been restarted. Please exit CLI and log back on.
The parameter IS_UTF8MB4 has been changed to 1.

> show mysql_utf8mb4
mysql configuration set with UTF8MB4.
ok
```

store packet max-size

スニファーからのパケットの最大サイズを制限します。

構文

store packet max-size 1536

表示コマンド

show packet max-size

store pdf-config

このコマンドを使用すると、(ヘッダー/フッターを除く) PDF イメージ本文コンテンツの pdf フォント・サイズと pdf 用紙の向きを変更できます。

サイズは 1 (最小) から 10 (最大) までの範囲で、デフォルト値は 6 です。

用紙の向き (orientation) は 1 (横長) または 2 (縦長) です。デフォルト値は 1 です。

CLI コマンドを入力して Enter キーを押すと、変更内容が直ちに有効になります。

構文

store pdf-config [orientation | size]

表示コマンド

show pdf-config [orientation | size]

store pdf-config multilanguage_support

英語 (英語バージョンで使用) と言語 C/J (中国語/日本語で使用) では、静的 PDF ジェネレーター構成ファイルが異なります。この CLI コマンドを使用して、PDF ジェネレーターのフォントを定義します。Default は英語です。Multi-language は言語 C/J です。

構文

```
CLI> store pdf-config multilanguage_support
Current setting is Default

1  Default
2  Multi-language
Please select the option (1,2, or q to quit)
```

表示コマンド

show pdf-config multilanguage_support

store populate_from_query_maxrecs

照会からのグループおよび別名の取り込みに使用できるレコードの最大数を設定します。

この CLI コマンドを使ってレコード最大数の値を設定するときには、注意が必要です。高く設定し過ぎると、照会からグループに取り込む プロセスが完了しない可能性があります。最大しきい値は動的で、システム負荷とメモリー使用状況に依存します。この CLI コマンドの最大値は 200000 に制限されています。

構文

store populate_from_query_maxrecs 100000

表示コマンド

show populate_from_query_maxrecs

store product gid

保管される固有のプロダクト <n> GID 値を設定します。

構文

store product gid <n>

表示コマンド

show product gid

store purge object

不必要なオブジェクトがパージされる経過日数を設定します。show purge objects age コマンドを使用すると、パージ経過日数を維持する対象となる各オブジェクト・タイプの索引、オブジェクト名、経過日数を示す表が表示されます。次に、その表の適切な索引をコマンド内で使用して、パージ経過日数を設定します。

注: ユニット・タイプが管理対象ユニット、Manager、またはスタンドアロン・ユニットの間で変更されるとき、日数の値はデフォルト (90 日) に設定されます。

構文

store purge object age <索引> <日数>

表示コマンド

show purge object age

例

イベント・ログを 30 日間にわたって保持する必要が生じたとします。まず show purge objects age コマンドを発行して索引を判別します (表を使用しないでください。実際のリストは異なる可能性があります)。次に store purge object コマンドを入力します。

CLI>show purge objects age

Index Name, Age

1. 一元管理永続処理、7
2. S-TAP イベント・ログ、14
4. アセスメント・テスト、7
5. 一元管理一時ポリシー、7
6. S-TAP 変更履歴、14
7. Kerberos 認証情報、1
8. コメント履歴、60
9. コメント・ローカル履歴、60
10. グラフ呼び出し履歴、90

. . .

ok

CLI> **store purge object age 2 30**

ok

store quartz_thread_run

この CLI コマンドは技術サポートによって使用されます。

Java™ 仮想マシンでは、アプリケーションが複数のスレッドを使用できます。スレッドとはプログラム実行の断片です。

同時に実行可能なスレッドの数を設定するには、CLI コマンド store quartz_thread_num を使用します。

このコマンドを使用すると、同時に実行されるスレッドが多すぎる場合の互いの競合を軽減できます。

CLI コマンド show quartz_thread_num は、同時に実行される Quartz スケジューラー・スレッドの数を表示します。

構文

store quartz_thread_run <number>

使用法: store quartz_thread_num <number> (ここで number は 3 から 15 までの範囲、デフォルト値は 5)

表示コマンド

show quartz_thread_num

org.quartz.threadPoll.threadCount= 5

store remotelog

リモート・ロギングの使用を制御します。システム・メッセージに加えて、統計アラートおよびポリシー・ルール違反メッセージを (オプションで) syslog に書き込むことができます。それぞれの **facility.priority** の組み合わせごとに、メッセージを特定のホストに送信することができます。また、このコマンドは、オプションのポート番号を介したリモート・ロギングの使用も制御することができ、必須のプロトコル (UDP または TCP) も指定できます。このコマンドは TCP をサポートする任意の syslog 実装に対して機能します。

リモート・ロギングを有効にする場合、受信側ホストでこの機能が既に有効になっていることを確認してください (『注』を参照)。

構文

store remotelog [help|add|clear] facility.priority host [optional port number:mandatory protocol (UDP または TCP)]

表 2. store remotelog のパラメーター

パラメーター	記述
help	サポートされる機能と優先度を表示します。
add	指定された facility.priority の組み合わせを、指定されたリモート・ホストに送られるメッセージのリストに追加します。
clear	指定された facility.priority の組み合わせを、指定されたホストに送られるメッセージのリストから消去します。
facility	デーモンを使用します。IBM Guardium アプライアンスによって発行されるほとんどのメッセージは daemon 機能から出されます。
priority	これは alert、all、crit、debug、emerg、err、info、notice、warning のいずれか 1 つです。 アラートと違反に関する標準的な IBM Guardium 重大度コードは、次のようにマップされます。 Guardium severity / Syslog priority INFO / info LOW / warning MED / err HIGH / alert
host	この facility.priority の組み合わせを受信するホストを指定します。
optional port number (オプションのポート番号)	
mandatory protocol	UDP または TCP。
format	store remotelog format SIEM 製品によっては、デフォルトよりも IETF RFC 5424 スタイルの syslog メッセージのほうが処理に優れている場合があります。このコマンドは形式を変更します。形式が変更されたら、「restart rsyslog」を実行して変更を有効にする必要があります。 使用法: store remotelog format <default|rfc5424> default - rsyslog の従来形式 rfc5424 - rsyslog の RFC 5424 形式 注: syslog 受信者は RFC5424 形式を受け入れるように構成されている必要があります。そうでない場合、従来形式で受信します。

パラメーター	記述
max_message_size	このコマンドは、5k から 64k の範囲のパラメーターで最大メッセージ・サイズを設定するために使用します。 入力を簡単にするために、値は索引キーに割り当てられます。キーと値のペアは次のとおりです。 • 1 = 5k • 2 = 10k • 3 = 15k • 4 = 20k • 5 = 32k • 6 = 64k **構文** store remotelog max_message_size <1|2|3|4|5|6> 新しい構成を適用するには、restart remotelog を実行します。 **表示コマンド** $MaxMessageSize パラメーターの現在の値を表示するには、このコマンドを使用します。 show remotelog max_message_size
escape_control_characters_on_receive	rsyslog 内のエスケープ制御文字は、on または off に設定できます。デフォルトでは、on に設定されています。 **構文** show remotelog escape_control_characters_on_receive <on|off> 新しい構成を適用するには、restart remotelog を実行します。 **表示コマンド** rsyslog 内の $EscapeControlCharactersOnReceive パラメーターの現在の値を表示するためには、このコマンドを使用します。 show remotelog escape_control_characters_on_receive

注:

リモート・ロギングを受け入れるよう受信側システムを構成するには、そのシステムの /etc/sysconfig/syslog を編集して -r オプションを含めます。例:

```
SYSLOGD_OPTIONS=-r -m 0
```

その後、次のように syslog デーモンを再始動します。

```
/etc/init.d/syslog  restart
```

Linux での標準的な syslog ファイルの名前は、次のとおりです。

```
/var/log/messages
```

コモン・クライテリアでは、Guardium システムからリモート syslog サーバーへのすべての通信は暗号化される必要があります。リモート syslog サーバーへの通信は平文であってはなりません。

CLI コマンド

show remotelog host

store remotelog ?

store remotelog add ?

store remotelog add encrypted

使用法: store remotelog add encrypted <facility.priority> <host[:port]> <tcp|udp>

使用可能な機能: all auth authpriv cron daemon ftp kern local0 local1 local2 local3 local4 local5 local6 local7 lpr mail mark news security syslog user uucp

使用可能な優先順位: alert all crit debug emerg err info notice warning

注:

暗号化されたリモート・ログ・メッセージをサーバーに送信する場合、サーバー内の rsyslog 構成は、暗号化されたメッセージを受け入れる必要があります。

クライアントおよびサーバー上の暗号化設定は TCP モードでのみ機能します。

同じリモート・サーバー上での 1 つのモードから別のモードへの切り替え: 指定されたモードと同期するように構成ファイルを変更する必要があり、リモート・サービスを再始動する必要があります。

例

```
store remotelog add non_encrypted
store remotelog clear
g32.guard.swg.usma.ibm.com> show remotelog host
*.*    @9.70.148.175:10514
```

この例を使用して、証明書を ca.pem として /etc/pki/rsyslog/ に保管します。これにより新しいウィンドウが開き、証明書を貼り付けることをユーザーに求めます。

```
store remote add encrypted all.all <IP address>:<port number> tcp
```

syslog の暗号化

アラートおよびその他のメッセージを SIEM システムなどのリモート syslog 受信者に転送できます。コレクターまたはアグリゲーターからリモート syslog 受信者まで、このメッセージ・トラフィックを暗号化することができます。

注: 暗号化は TCP モードでのみ機能します。デフォルトでは、syslog 転送には UDP が使用されるため、暗号化が必要な場合は CLI コマンド store remotelog で TCP を指定します。

作業を開始する前に、以下を実行してください。

ここに記載されている手順は、暗号化ホストにトラフィックを送信するすべてのコレクターまたはアグリゲーター上で繰り返す必要があります。

リモート syslog 受信者によって使用される証明書が必要です。この証明書を Guardium システム上に保管します。

1. CA (認証局) (Verisign、Thwate、Geotrust、GoDaddy、Comodo、自社内など) から取得した公開証明書を使用できる状態にします。
2. 暗号化された syslog の送信元となる個々の Guardium システムで CLI にログインします。コマンドを実行する前に、該当する証明書 (PEM 形式) を CA から取得して、証明書の「Begin」と「End」の行を含めてクリップボードにコピーします。
3. 次の CLI コマンドを入力します: store remotelog add encrypted daemon.all <暗号化されたリモート・ホストの IP アドレス>:<リモート・ホストのポート番号> tcp
   注: Guardium はデーモンを使用してアプリケーション・イベントを送信するため、この例ではデーモンを使用しています。
4. 次の指示が表示されます。

   ```
   Please paste your CA certificate, in PEM format. Include the BEGIN and END lines, and then press CTRL-D.
   ```

   PEM 形式の証明書をコマンド行に貼り付け、**CRTL-D** を押します。Guardium はこの入力を取り込み、/etc/pki/rsyslog/ca.pem として保管します。

   保管操作の成功または失敗を通知するメッセージが続きます。

   成功した場合、Guardium は正しい鍵を使用して暗号化トラフィックをリモート・システムに送信できます。

5. syslog トラフィックを暗号化ホストに送信するすべてのコレクターおよびアグリゲーターに対してこの手順を繰り返します。

store s2c

ADMINCONSOLE のいくつかの構成可能パラメーターを設定します。これらのパラメーターは、サーバーからクライアントへの (S2C) トラフィックをスロットルするために使用されます。

注: この CLI コマンドは、IBM Guardium 技術サービスから指示された場合にのみ使用してください。

最小値と最大値は次のとおりです:

ANALYZER_S2C_IGNORE = {0,1,2,3}

MAX_S2C_VELOCITY (K バイト/秒) - 数値 >=0 かつ <= 2147483647

MAX_S2C_INTERVAL (秒) - 数値 >=1 かつ <= 2147483647

CLI コマンド store throttle も参照してください。

構文

store s2c

使用法: store s2c ignore I maxrate M maxinterval T

ここで、0<=I<=3 (レベル)、0<=M<=2147483647 (K/秒)、および 1<=T<=2147483647 (秒) です。あるいは store throttle default を使用します

store s2c ignore 3 maxrate 300 maxinterval 5007

新しい構成は、CLI コマンドの restart inspection-core コマンドが実行された後で有効になります。

表示コマンド

show s2c

スロットル S2C パラメーター (デフォルト):

無視: 0

最大速度: 999999

最大間隔: 30

ANALYZER_S2C_IGNORE (0,1,2,3) - シナリオに基づいて、s2c スロットル・メカニズムの on/off を切り替えます。このフラグはビットに基づいています。0 は s2c スロットル・メカニズムを OFF にします。1 は「シナリオ 1」に示されている機能を on に設定し、2 は「シナリオ 2」で示される機能を on にします。3 は両方を on にします。

MAX_S2C_VELOCITY - 最大速度 (K バイト/秒)。この速度を超えた場合、アナライザーは CLI コマンド「セッションを無視」または「セッション応答を無視」要求を S-TAP® またはスニファーに送ります。

MAX_S2C_INTERVAL - CLI コマンド「セッションを無視」または「セッション応答を無視」要求が出される時間間隔 (秒数、デフォルトは 30 秒)。

シナリオ 1

大きな照会の途中で、スニファーが S-TAP またはネットワークからトラフィックを受信し始めます。すべての着信パケットは DB サーバー応答であるため、アナライザーは新しいセッションを作成しません。したがってロガーおよびルール・エンジンには情報が送られません。この種類のトラフィックはスニファーにとって無用です。他方、この種類のトラフィックは追加的な S-TAP およびスニファーの負荷を発生させる可能性があります。スロットル・メカニズムは、S2C 速度が MAX_S2C_VELOCITY より大きい場合にアナライザーからセッションを無視メッセージを送ることで、S-TAP およびネットワーク・スニファーの負荷を軽減するのに役立ちます。何らかの理由で S-TAP またはネットワーク・スニファーにこのメッセージが影響を及ぼしていない場合、アナライザーは MAX_S2C_INTERVAL の秒数が経過した後、「セッションを無視」要求を再び送ります。このスロットル・メカニズムを on に切り替えるには、ANALYZER_S2C_IGNORE フラグを 1 に設定してください。

シナリオ 2

着信トラフィックの S2C 速度が大きい場合 (>MAX_S2C_VELOCITY)、スロットル・メカニズムは、S2C 速度が MAX_S2C_VELOCITY を超える場合のローカル・データベース接続に関して「セッション応答を無視」要求を S-TAP に送ります。何らかの理由で S-TAP にこのメッセージが影響を及ぼしていない場合、アナライザーは MAX_S2C_INTERVAL の秒数が経過した後、セッション応答を無視要求を再び送ります。このスロットル・メカニズムを on に切り替えるには、ANALYZER_S2C_IGNORE フラグを 2 に設定してください。

store sender_encoding

この CLI コマンドは、以前はすべて UTF8 でエンコードされていた出力メッセージ (E メールおよび SNMP トラップ) を異なるエンコード・スキームでエンコードするために使用します。

例えば、Guardium ユーザーがすべての出力 SNMP メッセージを SJIS (代替日本語エンコード方式) でエンコードする場合など。

注: 変換が失敗し、その理由が、(a) 指定されたエンコード・スキームが無効だった、または (b) エンコードする文字を要求されたエンコード・スキームで表せなかった、のいずれかである場合は、デフォルトのエンコード・スキームである UTF8 を使用してメッセージが送信されます。

構文

store sender_encoding <str>,

ここで、str は最大長 16 のエンコード方式です。

表示コマンド

show sender_encoding

store snif_mask_sql_value

このコマンドを使用して、SQL 例外が発生したときに生成される SQL 値をマスクします。

構文

store snif_mask_sql_value on|off

表示コマンド

show snif_mask_sql_value

store stap approval

この機能を使用して、無許可の STAP が Guardium アプライアンスに接続することをブロックします。

ON にすると、STAP は、特定の承認を得ない限り、接続できなくなります。

承認を得ていない STAP は、自身の IP アドレスに特定の権限が与えられない限り、接続してもすぐに切断されます。

承認されたクライアント用の事前定義レポート「承認済み TAP クライアント」があります。このレポートは「日次モニター」タブで表示できます。

注:

ホスト名ではなく、有効な IP アドレスが必要です。

CLI コマンド store stap approval は、IP ロード・バランサーがある環境内では機能しません。

一元管理された環境内では、承認された STAP に IP を追加した後、同期に関連する待ち時間が発生します。この待ち時間は、最大で 1 時間かかる可能性があります。同期が完了すると、承認された STAP 状況は GUI に緑色で表示されます。

構文

store stap approval ON | OFF

表示コマンド

show stap approval

GuardAPI コマンド

grdapi store_stap_approval

新しい構成は、CLI コマンド restart inspection-core の実行後に有効になります。

store stap certificate

IBM Guardium アプライアンス上で、S-TAP ホスト (通常はデータベース・サーバー) からの証明書を保管します。このコマンドの機能は、後で説明する store certificate console コマンドとまったく同じです。

構文

store stap certificate

次のようなプロンプトが出されます。

新規のサーバー証明書を PEM 形式で貼り付けてください。(Please paste your new server certificate, in PEM format.)

BEGIN 行および END 行を含め、CTRL-D キーを押します。

サーバー証明書をクリップボードにまだコピーしていない場合は、コピーします。PEM 形式の証明書をコマンド行に貼り付け、CRTL-D を押すと、保管操作の成功または失敗が通知されます。

完了したら、**restart gui** コマンドを使って IBM Guardium GUI を再始動します。

store stap network_latency

S-TAP 検査は、S-TAP によってデータベース・トラフィックがモニターされているかどうかをユーザーが検査するための機能です。この検査機能は、ユーザーのネットワーク・トラフィック/待ち時間の影響を受けます。待ち時間は各ユーザーごとに異なるため、この検査機能で使用されるデフォルト値をリストおよび変更する手段が必要です。

構文

store stap network_latency

使用法: store stap network_latency <N>

N は 0 より大きい数値 (秒) です。

デフォルト値は 5 秒です。

この数値が大きくなるほど、S-TAP 検査プロセスの速度が低下します。

表示コマンド

show stap network_latency

store set_partitions_for_queries

この CLI コマンドは、照会でパーティション選択を有効/無効にするときに使用します。

使用法:

store set_partitions_for_queries <on|off>

store storage-system

store storage-system

アーカイブ用またはシステム・バックアップ用のストレージ・システム・タイプを追加または削除します。

構文

store storage-system <Centera | TSM> <backup | archive> <on | off>

表示コマンド

show storage-system

例

現在、システム・バックアップ用に Centera を使用していて、TSM システムに切り替えることを決定したとします。(別のオプションとして残しておく場合を除き) Centera バックアップ・オプションを off にして、TSM バックアップ・オプションを on にする必要があります。これを行うためのコマンドが、例に強調表示されています。表示コマンドは必要ありませんが、説明のためにこの例に含まれています。

CLI> show storage-system

NETWORK :

CENTERA : backing-up

TSM :

SCP : archiving and backing-up

FTP : archiving and backing-up

ok

CLI>store storage centera backup off

ok

CLI>store storage tsm backup on

ok

CLI> show storage-system

NETWORK :

CENTERA :

TSM : backing-up

SCP : archiving and backing-up

FTP : archiving and backing-up

ok

CLI>

store support state

サポート E メール・アドレスへの E メール・アラートの送信を有効 (**on**) または無効 (**off**) にします。この E メール・アドレスは **forward support email** コマンドを使って構成可能です。デフォルトではサポート状態が有効 (**on**)、デフォルトのサポート E メール・アドレスは support@guardium.com です。

構文

store support state <on | off>

表示コマンド

show support state

store throttle

この CLI コマンドは、スロットル・パラメーターを保管します。 このコマンドを入力した後、変更内容を有効にするには CLI コマンド restart inspection-core を発行する必要があります。

このコマンドは、大きなパケットをフィルターで除外 (無視) するために使用されます。2 つのスロットル・モードがあります。まず、「しきい値 (セッションごと)」は、大きなパケット (サイズ構成可能) の長すぎるバースト (期間は構成可能) を識別したときにセッションを無視し、トラフィックが特定のしきい値 (これも構成可能) を下回ったときにセッション無視を停止します。次に、「全体」は、特定のサイズ (構成可能) より大きいすべてのパケットをすべてのセッションで無視します。このスロットル・モードは、定義済みサイズよりも小さい、長く過剰な非データベース・パケットを完全に無視します (VNC クライアントおよび他の種類のホワイト・ノイズ・トラフィックに対して役立ちます)。SPAM ポートまたはハードウェア TAP を介したネットワーク・トラフィックに使用します。S-TAP トラフィックの場合は、PCAP によって扱われるネットワーク TCP トラフィックだけです。CLI コマンド store s2c も参照してください。

構文

store throttle [default | size <s> interval <i> trigger <t> release <r>]

使用法: store throttle size S interval I trigger T release R

ここで、0<=S<=2^17 (バイト)、1<=I,T,R,<=2^31 (秒) です。

あるいは、store throttle default を使用します。

表示コマンド

show throttle

Throttle parameters:

Packet size: 228000

Time interval: 604800

Trigger level: 10000000

Release level: 10000000

パラメーター

default - キーワード default を入力すると、システム・デフォルトが復元されます (他のパラメーターは使用されません)。デフォルト・スロットル・パラメーターは、「スロットルなし」です。

s - パケット・サイズ。バイト単位で、最大 2^17 (131072)。

残りのパラメーターは秒単位で、最大 231 (2147483648) です:

i - 時間間隔

t - トリガー・レベル

r - リリース・レベル

注: スロットルのデフォルトを復元するには、CLI コマンド store throttle default を使用してください。

store timeout

CLI セッションまたはファイル・サーバー・セッション (あるいはこの両方) のタイムアウト値を設定します。デフォルト値は 600 秒です。タイムアウトが発生すると、CLI セッションも閉じられます。

タイムアウトが発生したためにファイル・サーバーが停止すると、次のメッセージが表示されます。「警告 ： タイムアウトになったため、ファイル・サーバーが停止しました。ファイルのアップロードは完了していない可能性があります。処理を停止します。」

conf ファイルの socketTimeout 値を表示するには CLI コマンド show timeout db_connection を使用し、タイムアウトの値を設定するには store timeout db_connection <値> を使用します。この値には 0 より大きい値を指定する必要があります。デフォルト値は 25000 秒です。これらの CLI コマンドは、DNS が構成されていない場合に中央マネージャーと管理対象ユニットの間の通信を管理するために使用します。

構文

store timeout cli_session <n>

store timeout fileserver_session <n>

store timeout db_connection <n>

表示コマンド

show timeout cli_session 600

show timeout fileserver_session 600

show timeout db_connection 25000

store transfer-method

CSV/CEF エクスポートで使われるファイル転送方式を設定します。ファイルをエクスポートする場合、CLI コマンド store transfer-method csv を使用して転送方式を設定する必要があります。バックアップまたはアーカイブを実行する場合、CLI コマンド store transfer-method backup を使用して転送方式を設定します。

構文

store transfer-method <FTP | SCP>

表示コマンド

show transfer-method

注: 1 つの IBM Guardium アプライアンスから別のアプライアンスに (例えばコレクターからアグリゲーターに) 送信されるファイルは、常に SCP を使って送られます。

store uid_chain_polling_interval

この CLI コマンドを使用して、UID チェーン・ポーリングの間隔を設定します。UID チェーン・メカニズムを使用すると、S-TAP は (K-Tap を介して)、データベース接続前に発生したユーザーのチェーンをトラッキングできます。

データベースのパフォーマンスを向上させるために UID チェーン処理をオフにするには、間隔を 0 に設定します。 UID チェーン処理がオフになっている場合、UID チェーンの計算および子セッションの更新はスキップされます。

注: データベースを使用するとき、セッションが非常に短い場合には、すべてのセッションで UID チェーンがログに記録されるとは限りません。

構文

store uid_chain_polling_interval <N>

N は分単位の時間です (1 分以上、デフォルトは 2 分)

N を 0 に設定すると、UID チェーン処理はオフになります。

表示コマンド

show uid_chain_polling_interval

store upd_session_end

この CLI コマンドは、セッション終了時刻の更新をスキップするオプションを追加します。

構文

store upd_session_end [enable | disable]

表示コマンド

show upd_session_end

store unit type

この CLI コマンドを使用して、Guardium アプライアンスのユニット・タイプ属性を設定します。このコマンドによって表示できるすべてのユニット・タイプ属性についての説明は、ユニット・タイプ属性表を参照してください。

構文

store unit type [manager | standalone] [netinsp] [stap] [mainframe] [sink]

収集された DRDA トラフィックのタイム・スタンプの細分度を 1 ミリ秒から 1 マイクロ秒に切り替えるには、`store unit type sink` を使用します。

表示コマンド

show unit type

注: リストされているいくつかの属性は store unit type コマンドを使って設定され、delete unit type コマンドを使って消去されます。アグリゲーター属性は、IBM Guardium ソフトウェアのインストール時にのみ設定できます。IBM Guardium ソフトウェアの再インストール以外では、変更できません。

support store ora_tns_errors

処理の早い段階における TNS エラーの処理方法を制御します。これを使用して、TNS エラーがまったくログに記録されないようにすることができます。(v10.6 より前のバージョンでは、TNS 関連のエラーがログに記録され、失敗したログインとしてカテゴリー化されていました。また、これらのエラーは、例外ポリシー・ルールまたはエラー・コード・グループを使用してフィルタリングされていました。)

構文

support store ora_tns_errors [0 | 1]

0: TNS エラーを保管しない

1: TNS エラーを保管する (デフォルト)

コマンドの説明を表示する

show ora_tns_errors

ユニット・タイプ属性

以下の表では、show unit type コマンドによって表示できる Guardium システムのユニット・タイプ属性について説明します。特に明記しない限り、これらの属性は store unit type コマンドを使って設定し、delete unit type コマンドを使って消去することができます。

表 3. ユニット・タイプ属性

属性	記述
mainframe	このユニットはメインフレーム (z/OS®) ネットワーク検査アプライアンスです。
manager	このユニットで中央マネージャー機能が有効になります。
netinsp	ネットワーク・トラフィックの検査が有効になります。
network route static	静的ルーティング表から 1 行を除去します
standalone	ローカル管理 (中央マネージャーから独立)
stap	このユニットは S-TAP および CAS エージェントからデータを受信し、これらを管理することができます。

unregister management

unregister (登録抹消) コマンドは、アプライアンスの一元管理の登録時に保存された構成を復元します。以前のリリースの IBM Guardium ソフトウェアの下で登録が行われた場合、保存済み構成を現在のソフトウェア・リリース・レベルに引き上げる目的でまずパッチを適用せずにその構成を復元した場合、アプライアンスが使用不可になり、これが原因で、そこに保管されているデータがすべて失われる可能性があります。したがって、登録前の構成が現在のソフトウェア・リリース・レベルになっていることを確認するまでは、ユニットを登録抹消しないでください。この確認方法が分からない場合は、ユニットを登録抹消する前に技術サポートに連絡してください。

構文

unregister management

注:

- このコマンドは緊急用です。中央マネージャーが使用不可になった場合にのみ、これを使用します。
- このコマンドを使って登録抹消した後、中央マネージャーからも登録抹消する必要があります (管理コンソールから)。管理対象ユニットの数を減らすには、これが唯一の方法であるためです。許可される管理対象ユニットの数は、プロダクト・キーによって決められています。

親トピック: CLI の概要
関連情報:
Guardium のトラブルシューティングとサポート (ビデオ)

diag CLI コマンド

これらの CLI コマンドを使用して、DIAG を介してトラブルシューティング・ユーティリティーおよびメンテナンス・ユーティリティーにアクセスできます。

技術サポートの指示どおりに diag コマンドを使用します。

このコマンドを使用して定期的に実行する必要がある機能はありません。メインメニューの各項目について、個別のトピックで説明します (『メインメニュー・コマンド』を参照してください)。

DIAG によるトラブルシューティング・ユーティリティーおよびメンテナンス・ユーティリティー:

- Aggregator Fix Schema - インポートされた表のうち、アグリゲーターのスキーマよりも古いスキーマを持つすべての表のスキーマを、アグリゲーターの最新パッチ・レベルに変更します (バックグラウンドで実行され、完了までに数時間かかる場合があります)。注: (a) アグリゲーターのパッチ・レベルが最新ではない、または (b) インポートされた表の中に、パッチ・レベルが最新のものがある、という状況により、インポートされたすべての表が「最新のパッチ・レベル」を持っているわけではない場合があります。
- Aggregator Maintenance - アグリゲーターの完全な分析およびリカバリーです。このユーティリティーは、AGG 関連ログを収集して diag エクスポート・フォルダーに配置し、Aggregator Fix Schema を呼び出してすべてのデータベースのスキーマを同期し、AGG ワークスペースのクリーンを行い、マージ処理を再開して、インポートされたすべての表の完全な分析が確実に行われるようにします (バックグラウンドで実行され、完了までに数時間かかる場合があります)。
- Clean Static Orphans on an Aggregator - このオプションは、静的表が増えすぎたために消去する必要がある場合にのみ、必ず技術サポート限定で使用します。このユーティリティーは、使用されなくなった古い構成レコードをすべて消去します。

診断メインメニューを開く

diag コマンドを使用するには、概説する手順に従ってください。

1. コマンド行プロンプトで、CLI を使用して Guardium® アプライアンスにログインします。

   diag コマンドを使用する Guardium ユーザーには、CLI ロールまたは admin ロールが割り当てられている必要があります。デフォルトで「CLI」ロールを持っているユーザーは、admin のみです。「CLI」ロールまたは「admin」ロールを持つユーザーは、diag コマンドの入力、unlock admin CLI コマンドおよび unlock accessmgr CLI コマンドの使用、および export audit-data CLI コマンドの制限なしの使用を許可されます。CLI ロールを持つユーザーは、GUI ログインに必要なユーザー名とパスワードを入力する必要がなく、それ以降のロールの確認も行われません。

   CLI を使用する Guardium ユーザーが「CLI」ロールまたは「admin」ロールを持っていない場合、CLI は開始しません。CLI ロールおよび admin ロールは、accessmgr により割り当てられます。

2. CLI が開始したら、コマンド行プロンプトで diag コマンド (引数なし) を入力します。
3. diag コマンドを使用する Guardium ユーザーには、Guardium システム上で diag ロールが割り当てられている必要があります。デフォルトでは、admin にのみこのロールが割り当てられています。diag へのアクセスは、このユーザーに割り当てられているロールに基づいて、許可または却下されます (diag へのアクセスは、このユーザーに「diag」ロールが割り当てられている場合にのみ許可されます)。diag ロールは、accessmgr により割り当てられます。
4. メイン・コマンド・メニューが表示されます。以下のいずれかを行って、オプション選択カーソル (例では最初の項目を選択しています) を移動させます。
   - 目的の項目番号を入力します (選択カーソルが選択された項目に移動します)。
   - 上矢印キーまたは下矢印キーを使用して、目的の項目を選択します。
5. スペース・バー、左矢印キー、または右矢印キーを押して、画面にあるコマンド選択カーソル (例では OK コマンドを選択しています) を移動させます。
6. 表示域の適切なオプションを選択して操作を実行し、次に以下のいずれかを実行します。
   - コマンド選択カーソルを使用して、適切なコマンドを選択し、Enter キーを押します。
   - 適切な操作コマンドをクリックします。

diag 出力について

diag コマンドでは、以下の 2 つのディレクトリーに出力が作成されます。

- .../guard/diag/current
- .../guard/diag/depot

この出力にアクセスするには、fileserver CLI コマンドを使用します。詳しくは、『fileserver』を参照してください。

各ディレクトリーについては、以下のサブセクションで説明します。

.../guard/diag/current ディレクトリー

diag コマンドからの出力のほとんどは、テキスト形式で current ディレクトリーに書き込まれます。ほとんどのコマンドでは、このディレクトリーにコマンドごとの個別の出力ファイルが含まれます。同じコマンドを実行する度に、それぞれのコマンド用の単一ファイルに出力が追加されます。幾つかのコマンドでは、実行ごとに個別のファイルが作成され、通常はファイル名に日時スタンプが取り込まれます。

セッションが終了する度に、以降のセッションで古い情報が表示されることがないよう、「クリーンアップ」することをお勧めします。エクスポート用にファイルを単一の圧縮ファイルに圧縮する場合 (以下のトピックを参照)、current ディレクトリー内のファイルがすべて削除されます。または、「Output Management」メニューの「Delete recordings」コマンドを使用して、個別のファイルを削除できます。

current ディレクトリー内のファイルは、メニュー名およびコマンド名から名前が付けられているため、簡単に特定することができます。例えば、「System Interactive Queries」メニューの「File Summary」コマンドを使用した場合、interactive_filesummary.txt という名前のファイルが current ディレクトリーに作成されます。

コマンドを使用している途中に current ディレクトリーを見ると、そのコマンドの出力を含むファイルと同じ名前の隠し一時ファイルが表示されている場合があります。一時ファイルは、コマンドの出力ファイルに出力が追加されると削除されます。

.../guard/diag/depot ディレクトリー

current ディレクトリーで diag 出力ファイルを (例えば、Guardium 技術サポートに送信するために) 圧縮ファイルに圧縮すると、その圧縮ファイルは depot ディレクトリーに保管されます。ファイル名は diag_session_<dd_mm_hhmm>.tgz という形式になります。この名前の変数部分は、ファイルが作成された時を示します。例えば、ファイルが 5 月 20 日の 12:15 PM に作成された場合、名前は diag_session_20_5_1215.tgz になります。

ファイルをエクスポートしたら (『Export recorded files 』トピックを参照)、「Output Management」メニューの「Delete recordings」コマンドを使用して、depot ディレクトリーからエクスポートしたファイルを削除できます。

1 Output Management

「Output Management」コマンドは、diag コマンドによって作成された出力に対する操作を制御します。各「Output Management」コマンドについて、個別に説明します。

1.1 End and pack current session

このコマンドを使用して、current ディレクトリー内の診断ファイルをすべて単一の圧縮ファイルに圧縮し、current ディレクトリーからそれらのファイルを削除します。このコマンドを入力した場合、コマンドが完了したことを示すフィードバックは返されません。depot ディレクトリーのディレクトリーを表示することにより、このコマンドが完了したことを検証できます。コマンドが完了すると、diag_session_<mm_dd_hhmm>.tgz という形式で名前が付けられたファイルが作成されます。前述したとおり、この名前の変数部分は日時スタンプです。「Output Management」メニューの「Export recorded files」コマンドを使用して、別のシステムにファイルを送信します。

1.2 Delete recordings

このコマンドを使用して、depot ディレクトリーまたは current ディレクトリー内のファイルを削除します。(現行セッションのファイルのみを削除するには、「Delete current session files」コマンドを使用します。) このコマンドを入力すると、depot ディレクトリー構造が表示されます。

上矢印キーおよび下矢印キーを使用し、Enter キーを押して、ディレクトリー内をナビゲートできます。例えば、../ を選択して Enter キーを押すと、選択をディレクトリー構造内で 1 つ上のレベルに移動できます。

次に current ディレクトリーを選択して Enter キーを押すと、そのフォルダーまでナビゲートし、個別のコマンド出力ファイルを削除できます。他のディレクトリーにナビゲートすることはできますが、current ディレクトリーおよび depot ディレクトリー以外のディレクトリーからファイルを削除することはできないことに注意してください。

削除するファイルを選択したら、Enter キーを押します。

注意: 削除操作の確認を求めるプロンプトは出されません。

1.3 Export recorded files

このコマンドを使用して、depot ディレクトリーから他のサイトにファイルを送信します。ファイルをエクスポートするには、次のようにします。

1. 「Output Management」メニューから「Export recorded files」を選択します。depot ディレクトリーが表示されます。
2. 送信するファイルを選択するか、../ エントリーおよび ./ エントリーを使用して、ディレクトリー構造内で上または下にナビゲートします。(ただし、エクスポートできるのは depot ディレクトリーのファイルのみであることに注意してください。)
3. 送信するファイルを選択した状態で、Enter キーを押します。
4. FTP を選択するか、終了するよう求めるプロンプトが出されます。FTP を選択し、Enter キーを押します。
5. ホスト名を入力するよう求めるプロンプトが出されます。受信システムのホスト名 (またはその IP アドレス) を入力し、Enter キーを押します。
6. ユーザー名を入力するよう求めるプロンプトが出されます。受信システムのユーザー・アカウント名を入力し、Enter キーを押します。
7. パスワードを入力するよう求めるプロンプトが出されます。受信システムのユーザーのパスワードを入力します。
8. 受信システムで送信されたファイルを受け取るディレクトリーを指定するよう求めるプロンプトが出されます。 受信システム上の、ファイルを格納するディレクトリーの FTP ルートに対する相対パスを入力し、Enter キーを押します。
9. 転送の詳細 (送信されるファイルとその宛先) を確認するよう求めるプロンプトが出されます。Enter キーを押して転送を実行するか、「Cancel」を選択して Enter キーを押し、最初からやり直します。
10. 操作が成功 (または失敗) したことが通知されます。

1.4 Delete current session files

このコマンドを使用して、現行セッション中に作成されたファイルを削除します。

1.5 Exit

「Exit」コマンドを使用して、メインメニューに戻ります。

2 System Static Reports

メインメニューの「System Static Reports」コマンドを使用して、詳細なレポートを作成します。

1. メインメニューから「System Static Reports」を選択します。プロセスが実行中であることが通知されます。
2. レポートの作成が完了すると、表示域にレポートが表示されます。(このレポートは長大であり、デスクトップ・コンピューターにエクスポートしてからテキスト・エディターを使用して表示したほうが見やすい場合があります。)

   上矢印キーおよび下矢印キーを使用して、レポート内をスクロールアップおよびスクロールダウンします。レポートの確認が終わったら、Enter キーを押してメインメニューに戻ります。

System Static Reports の概要

以下のサブトピックでは、「System Static Reports」出力の主要コンポーネントの概要を示します。示されている出力の一部は、実際の内容を詳細に説明するためではなく (本書で扱う範囲を超えています)、レポートに含まれる情報の種類とレベルを説明するためのものです。

システム構成情報

「System Static Reports」出力には、ビルド・バージョン、適用されているパッチ、現在のシステム・アップタイム、およびネーム・サーバーの情報が示されます。

```
Build version: 34e1eb12eb68ba76cb49028251c9a0d6  /opt/IBM/guardium/etc/cvstag
Patches:
2009/02/22 16:16:50: START Installation of 'Update 5.0'
2009/02/22 16:18:04: Installation Done - Successfully Installed

< lines deleted… >

Current uptime:
  09:03:43  up 6 days, 17:34,  1 user,  load average: 0.44, 0.50, 0.41
System nameservers:
192.168.3.20
DB nameservers:
```

```
192.168.3.20
Gateway: 192.168.3.1 (system) 192.168.3.1 (def)
```

次に、ファイル・システム情報が示されます (一部を示します)。

```
Filesystem            Size  Used Avail Use% Mounted on
/dev/hdc3             2.0G  1.1G  813M  58% /
/dev/hdc1              97M  9.2M   83M  10% /boot
none                  504M     0  504M   0% /dev/shm
/dev/hdc2              71G  1.2G   66G   2% /var
        total:    used:    free:  shared: buffers:  cached:
Mem:  1055199232 1041711104 13488128        0 63275008 186220544
Swap: 536698880 295432192 241266688
MemTotal:      1030468 kB
MemFree:         13172 kB

< lines deleted… >
```

続けて、構成されているメール・サーバーおよび SNMP サーバーについての情報が示されます。

```
SMTP server: 192.168.1.7 on port 25 : REACHABLE
SMTP user: undef
SMTP password: undef
SMTP auth: NONE
SNMP trapsink: undef UNREACHABLE
SNMP trap community: undef
SNMP read community: undef
```

システム構成セクションの最後のセクションでは、IP アドレス、ホスト名、およびドメイン名などの、ユニットのネットワーク構成が示されます。

```
eth0:                 192.168.3.101  (system) 192.168.3.101 (def)
hostname:                          (system)  g1 (def)
domain:                              (system)  guardium.com (def)
mac address:      00:04:23:A7:77:F2  (MAC1)  00:04:23:A7:77:F2 (MAC2)
unit type:           548 Standalone STAP
```

内部データベース情報

「System Static Reports」出力の次の主要セクションには、内部データベース状況およびスレッドに関する情報が含まれます (最初の数スレッドのみを示します)。

```
uptime 77097  seconds.
27  threads.
78545028  queries.
+------+------------+----------------------------+---------+---------+------+-----------
| Id   | User       | Host                       | db      | Command | Time | State |  +---------------------------------------
-------------------------------------------
| 1137 | enchantedg | localhost                  | TURBINE | Sleep   | 26   |
| 1257 | enchantedg | localhost.localdomain:33587 | TURBINE | Sleep   | 0    |
| 1258 | enchantedg | localhost.localdomain:60409 | TURBINE | Sleep   | 7716 |
| 1259 | enchantedg | localhost.localdomain:48233 | TURBINE | Sleep   | 322  |

< lines deleted… >
```

スレッドのリストに続けて、表の状況の分析が示されます。

Web サーブレット・コンテナーの情報

「System Static Reports」出力の次の数セクションには、Web サーブレット・コンテナー環境 (Tomcat) に関する情報が含まれます。

```
================================================================================
Currently defined Tomcat port is 8443.
The TOMCAT daemon is running and listening on port(s): 8005 8443.
Currently OPEN ports
java run by tomcat on port *:8443

< lines deleted… >
================================================================================

These are the nanny latest actions:
May 19 14:13:09 guard nanny:[5528]: Also checking tomcat.
May 19 14:13:09 guard nanny:[5528]: Going for my initial nap.

< lines deleted… >

This is the TOMCAT command line:
  463 sh -c ps -o pid,cmd -e | grep Dcatalina.base
21917 grep Dcatalina.base.
```

検査エンジンの情報

「System Static Reports」出力の次の主要セクションには、検査エンジンに関する情報が含まれます。

```
================================================================================
This is the SNIF (pid: 13036) command line:  13036 /opt/IBM/guardium/bin/snif.
This is the SNIF status:
Name:               snif
State:                R (running)
Tgid: 13036

< lines deleted… >
================================================================================
```

```
Current timestamp is 2009-05-20 11:56:41
This is the last timestamp at GDM_CONSTRUCT_INSTANCE: 2009-05-20 11:56:41
This is the last timestamp at GDM_EXCEPTION: 2009-05-20 11:56:41
This is the last timestamp at GDM_POLICY_VIOLATIONS_LOG: 2009-05-20 11:56:41

================================================================================

Snif buf usage at Fri May 20 11:56:44 2009:
100 204800 buffers out of 204800
126 connection used, 32642 unused, 0 dropped (sniffer), 9 ignored (analyzer)
0 bytes lost, 60 connections ended, 601752099 bytes sent, 579063 request sent
Dropped Packets: 0 buffer full, 0 too short , 451 ignored
 time now is 1116604603
Analyzer/Parser buffers size: 6 (66533) 0 (62902)
 ms-tsql-logger  0 (11331)
 syb-tsql-logger  0 (70)
 ora-tsql-logger  79 (67803)
 db2-sql-logger  0 (20544)

< lines deleted… >
```

IP 表の情報

次の主要セクションには、IP 表に関する情報が含まれます。

```
================================================================================
IPTABLES:
-------------
       tcp  --  192.168.2.0/24       192.168.1.0/24      tcp spts:1521:60000  set 0x23
       tcp  --  192.168.1.0/24       192.168.2.0/24      tcp dpts:1521:60000  set 0x22
< lines deleted… >
```

S-TAP の情報

次の主要セクションには、S-TAP® の情報が含まれます。

```
================================================================================
STAP:
----
    0     0 ACCEPT     tcp  --  *      *  0.0.0.0/0      0.0.0.0/0          tcp spt:9500
    0     0 ACCEPT     tcp  --  *      *  0.0.0.0/0      0.0.0.0/0          tcp dpt:9500
 2696  148K ACCEPT     tcp  --  *      *  0.0.0.0/0      0.0.0.0/0          tcp spt:16016
 2835  175K ACCEPT     tcp  --  *      *  0.0.0.0/0      0.0.0.0/0          tcp dpt:16016

< lines deleted… >
```

IP トラフィックの情報

次の主要セクションには、IP トラフィックの情報が含まれます。

```
IP traffic statistics.
OUTPUT OF ETH0
Fri May 20 11:57:04 2012; ******** Detailed interface statistics started ********

*** Detailed statistics for interface eth0, generated Fri May 20 11:58:04 2009

< lines deleted… >

OUTPUT OF ETH1
Fri May 20 11:57:04 2012; ******** Detailed interface statistics started ********

*** Detailed statistics for interface eth1, generated Fri May 20 11:58:04 2009

Total:               82440 packets, 53892382 bytes
        (incoming: 82440 packets, 53892382 bytes; outgoing: 0 packets, 0 bytes)
IP:    82440 packets, 52632747 bytes
        (incoming: 82440 packets, 52632747 bytes; outgoing: 0 packets, 0 bytes)

< lines deleted… >
```

情報エンジンの STDERR および STDOUT の情報

次のセクションには、スニファーによる最終メッセージ出力が含まれます。

```
Snif  STDERR:

< lines deleted… >

Snif STDOUT:
Fri_20-May-2009_04:04:35 : Guardium Engine Monitor starting
Fri_20-May-2009_04:14:37 : Guardium Engine Monitor starting
Fri_20-May-2009_04:24:38 : Guardium Engine Monitor starting

< lines deleted… >
```

インポート・ディレクトリーの情報

次のセクションには、インポート・ディレクトリーの内容がリストされます。

```
These are the contents of the importdir directory:
total 0
```

アグリゲーターのアクティビティーの情報

このセクションには、アグリゲーターのアクティビティーがリストされます (例では何も示されていません)。

```
================================================================================
This is the aggregator last activities:
```

監査レポート

このセクションには、次の要約情報がリストされます (例を参照してください)。

```
================================================================================
Range of time in logs: 01/14/10 13:12:26.348 - 01/18/10 12:48:01.073
Selected time for report: 01/14/10 13:12:26 - 01/18/10 12:48:01.073
Number of changes in configuration: 4    - changes to the audit configuration
Number of changes to accounts, groups, or roles: 0
Number of logins: 22     - logins into the machine - ssh and console
Number of failed logins: 114
Number of authentications: 22 - "su", etc.
Number of failed authentications: 5
Number of users: 2
Number of terminals: 18
Number of host names: 9
Number of executables: 7
Number of files: 0
Number of AVC's: 0
Number of MAC events: 0
Number of failed syscalls: 0
Number of anomaly events: 3
Number of responses to anomaly events: 0
Number of crypto events: 0
Number of keys: 0
Number of process IDs: 9173
Number of events: 98669
================================================================================
```

異常レポート

このセクションには、以下がリストされます (例を参照してください)。

```
================================================================================
# Date Time Type Exe Term Host AUID Event
================================================================================
1. 01/14/10 13:16:02 ANOM_PROMISCUOUS /usr/sbin/brctl (none) ? -1 8 -  this is expected
to appear - it means the bridge is listening to all traffic
```

認証レポート

このセクションには、以下がリストされます (例を参照してください)。

```
================================================================================
# Date Time Type Exe Term Host AUID Event
================================================================================
1. 01/14/10 13:13:22 tomcat ? console /bin/su yes 4
2. 01/14/10 13:16:44 tomcat ? console /bin/su yes 11
3. 01/14/10 13:16:44 tomcat ? console /bin/su yes 17
4. 01/14/10 13:16:45 tomcat ? console /bin/su yes 23
5. 01/14/10 13:16:48 tomcat ? console /bin/su yes 29
6. 01/14/10 13:22:29 tomcat ? ? /bin/su yes 155
7. 01/14/10 13:28:10 ? ? tty1 /bin/login no 252
8. 01/14/10 13:28:20 ? ? tty1 /bin/login no 254
```

ログイン・レポート

このセクションには、以下がリストされます (例を参照してください)。

```
================================================================================
# Date Time Type Exe Term Host AUID Event
================================================================================
1. 01/14/10 13:22:15 root 192.168.2.9 sshd /usr/sbin/sshd no 142
2. 01/14/10 13:22:15 root 192.168.2.9 sshd /usr/sbin/sshd no 143
3. 01/14/10 13:22:17 root 192.168.2.9 sshd /usr/sbin/sshd no 144
4. 01/14/10 13:22:17 root 192.168.2.9 sshd /usr/sbin/sshd no 145
5. 01/14/10 13:22:20 root 192.168.2.9 sshd /usr/sbin/sshd no 146
```

3 Interactive Queries

メインメニューから「System Interactive Queries」を選択し、「Interactive Queries」メニューを開きます。(このメニューの項目をすべて表示するには、下矢印キーを使用して 10 個目の項目より先までスクロールしてください。)

各対話式照会コマンドでは、要求された情報の表示に加え、その出力を含む個別のテキスト・ファイルを current ディレクトリー内に作成します。作成されるファイルについて詳しくは、概要トピックを参照してください。

各コマンドについては、以下のセクションで説明します。

3.1 Files Changed

「Files Changed」コマンドを使用して、指定した日数より前または後に変更されたファイルのリストを表示します。

1. 「Interactive Queries」メニューから、「Files Changed」を選択します。日数を入力するよう求めるプロンプトが出されます。数値を入力し、Enter キーを押します。
2. 入力した日数より前または後に変更されたファイルを表示するかどうかを確認するプロンプトが出されます。1 または 2 を選択し、Enter キーを押します。
3. 変更された各ファイルの絶対ディレクトリー・パスが表示されます。表示域にすべてのデータが収まらない場合は、上矢印キーおよび下矢印キーを使用して、スクロールしてデータを表示してください。ファイル内での現在位置は、ディスプレイの数値により示されます。表示域の白いバーに正符号が表示されている場合、さらにデータが存在することを示します。

3.2 List Folder

このコマンドを使用して、さまざまなディレクトリーの内容をリストします。

1. 「Interactive Queries」メニューから、「List Folder」を選択します。
2. ディレクトリーを選択するよう求めるプロンプトが出されます。ディレクトリーを選択し、Enter キーを押します。選択されたディレクトリーが表示されます。同じ種類のコマンドが複数実行された場合、各コマンドの実行により作成されたデータは、そのコマンド用に維持されている単一のテキスト・ファイルに追加されます。
3. 完了後、Enter キーを押すか、「Exit」をクリックします。

3.3 Summarize Folder

「Summarize Folder」コマンドを使用して、du (ディスク使用状況) コマンドの出力を表示します。

1. 「Interactive Queries」メニューから、「Summarize Folder」を選択します。プロンプトは出されません。さまざまなディレクトリーのディスク使用状況が表示されます。
2. 上矢印キーおよび下矢印キーを使用して、スクロールしてディレクトリーを表示します。
3. 完了後、Enter キーを押すか、「Exit」をクリックします。

3.4 File Summary and Export

このコマンドを使用して、ログ・ファイルのすべてまたは一部分をリストします。

1. 「Interactive Queries」メニューから、「File Summary」を選択します。
2. ファイルを選択するよう求めるプロンプトが出されます。上矢印キーおよび下矢印キーを使用して、表示するファイルまで選択カーソルをスクロールします。
3. Enter キーを押すか、「OK」をクリックします。
4. 表示する行数を選択するよう求めるプロンプトが出されます。選択したら、Enter キーを押します。
5. オプションの検索文字列を入力するよう求めるプロンプトが出されます。特定のログ・メッセージを検索する場合は、このボックスを使用します (正規表現を入力できます)。そうでない場合は、このボックスを空のままにして、Enter キーを押します。
6. プロンプトに従って操作し、Enter キーを押して「Yes」を選択します。これにより、固有のメッセージのみが表示されます。そうでない場合は、「No」を選択して Enter キーを押します (すべてのメッセージが表示されます)。

   「Summary Style」が使用されている場合は、変数がポンド記号文字 (#) で置き換えられることに注意してください。IP アドレスや日付といった変数を含む一部のログ・データでは、より広い範囲で置換が行われる場合があります。

3.5 Test Email

このコマンドを使用して、構成済みの SMTP サーバーを使用してテスト E メールを送信します。

1. 「Interactive Queries」メニューから、「Test Email」を選択します。
2. 宛先を選択するよう求めるプロンプトが出されます。「Custom」を選択し、Enter キーを押します。
3. E メール・アドレスを入力するよう求めるプロンプトが出されます。E メール・アドレスを入力し、Enter キーを押します。操作の出力が通知されます。管理コンソールで、アラート機能構成パネルの SMTP ペイン内の「Test Connection」リンクを使用すると、SMTP ポートが構成されているかどうかのテストのみが行われ、そのサーバー経由で実際にメールを送信できるかどうかはテストされないことに注意してください。このコマンドを使用して、統計アラートやリアルタイム・アラート、または監査プロセスの通知を構成および起動することなく、E メールの送信をテストできます。

3.6 Test SNMP

このコマンドを使用して、構成済みの SNMP サーバーにテスト SNMP トラップを送信します。

1. 「Interactive Queries」メニューから、「Test SNMP」を選択します。
2. アクティビティーとその結果が通知されます。アラート機能構成パネルの、SNMP ペイン内の「Test Connection」リンクを使用すると、SNMP ポートが構成されているかどうかのテストのみが行われ、そのサーバー経由で実際にトラップを送信できるかどうかはテストされないことに注意してください。このコマンドを使用して、統計アラートやリアルタイム・アラート、または監査プロセスの通知を構成 (および起動) することなく、トラップの送信をテストできます。

3.7 Report Query Data

このコマンドを使用して、レポート照会に使用される実際の select ステートメントを表示します。これは、ユーザー作成レポートにより予期しない出力が生成された場合に、役立つことがあります。

1. 「Interactive Queries」メニューから「Report Query Data」を選択します。
2. レポート・タイトルのリストの中から選択するよう求めるプロンプトが出されます。上矢印キーおよび下矢印キーを使用して項目を選択し、Enter キーを押します。このリストの各項目は、レポート・エンティティーです。すべての事前定義レポートが、リストの先頭に表示されます。これらには、100 から 225 までの番号が付けられています (バージョン 3.6.1 の場合。リリースごとにさらに多くの事前定義レポートが作成されるため、通常この数字は大きくなっていきます)。

   ユーザー作成レポートは、事前定義レポートの後に表示され、20001 から番号が付けられます (バージョン 3.6.1 の場合)。

選択したレポートの select ステートメントが表示されます。

3.8 GDM Queries

このコマンドを使用して、100 秒間隔で監視された SQL 呼び出しの数を表示します。

1. 「Interactive Queries」メニューから、「GDM Queries」を選択します。
2. 待機するよう求めるメッセージが表示されます。「Yes」を選択して続行します。表示画面上の CMD_CT 列に、指定したクライアントから指定したサーバーへの SQL 呼び出しの監視数がリストされます。
3. レポートの確認が完了した後で、Enter キーを押します。

3.9 Generate TCP Dump

このコマンドを使用して、TCP ダンプを作成します。このコマンドでは、出力はコマンド・ファイルにのみ書き込まれ、表示画面には書き込まれません。他の多くのコマンドとは異なり、このコマンドを実行する度に、current ディレクトリーに個別のファイルが作成されます。ファイル名は、 tcpdump_<mmyyyy-hhmmss> という形式になります。ここで変数部分は日時スタンプです。mmyyyy は月と年、および hhmmss は時、分、および秒です。

1. 「Interactive Queries」メニューから、「Generate TCP dump」を選択します。
2. インターフェースを選択するよう求めるプロンプトが出されます。ポートを選択し、Enter キーを押します。
3. オプションのフィルター IP アドレスを入力するよう求めるプロンプトが出されます。特定のアドレスからのトラフィックのみが必要な場合は、その IP アドレスを入力して Enter キーを押します。そうでない場合は、そのまま Enter キーを押します。
4. オプションのポート番号を入力するよう求めるプロンプトが出されます。特定のポートからのトラフィックのみが必要な場合は、そのポート番号を入力して Enter キーを押します。そうでない場合は、そのまま Enter キーを押します。
5. 何秒間トラフィックを取り込むのかを選択するよう求めるプロンプトが出されます。秒数を選択し、Enter キーを押します。
6. Enter キーを押して、データの収集を開始するよう求めるプロンプトが出されます。Enter キーを押します。(ほぼ) 指定した秒数後、メニューに戻ります。
7. TCP ダンプ・データを表示するには、「Read TCP dumps」コマンドを選択するか、ファイルをエクスポートします (前述した「Output Management」メニューの「Export Reported Files」を参照してください)。

3.10 Read TCP Dumps

このコマンドを使用して、先ほど作成した TCP ダンプ・ファイルを表示します。

1. 「Interactive Queries」メニューから「Read TCP dumps」を選択します。
2. ファイルを選択するよう求めるプロンプトが出されます。TCP ダンプ・ファイルは、古いものから新しいものの順にリストされます。ファイル名は tcpdump_<mmddyy-hhmmss> という形式になります。ここで変数部分は日時スタンプです。 mmddyy が月、日、および年、hhmmss が時、分、および秒です。表示するファイルを選択し、Enter キーを押します。
3. 選択したファイルが表示されます。 上矢印キーおよび下矢印キーを使用して表示をスクロールし、完了した後で Enter キーを押します。

3.11 Watch Buffer

このコマンドを使用して、Guardium バッファー内のアクティビティーを監視します。

1. 「Interactive Queries」メニューから、「Watch Buffer」を選択します。表示は毎秒更新されます。
2. Ctrl キーを押しながら C を押して、表示を閉じます。

3.12 SLON Utility

このコマンドを使用して、パケットをトラッキングする slon ユーティリティーを実行します。通常は、技術サポートの指示があった場合にのみ、このコマンドを実行します。このコマンドでは、出力は表示画面に書き込まれません。出力は、コマンドの実行ごとに、current ディレクトリー内にある 2 つのコマンド・ファイル、apks.txt.<day_dd-mmm-yyyy_hh.mm.ss.ttt> または requests.txt.<day_dd-mmm-yyyy_hh.mm.ss.ttt> のいずれかに書き込まれます。

ファイル名の変数部分は日時スタンプです。例えば、apks.txt.Fri_20-May-2011_08.52.00.789 です。

1. 「Interactive Queries」メニューから、「Slon Utility」を選択します。
2. 実行する操作を選択し、「OK」をクリックします。選択項目は、以下のとおりです。

   (a) アナライザーのルール情報をダンプする

   (f) IP またはマスク (あるいはこの両方) に基づいてアナライザー・パケットをフィルターに掛ける

   (p) パケットを apks.txt にダンプする

   (l) ロガーの要求を requests.txt にダンプする

   (m) STAP パケットをダンプする (実行時間を選択します。完了するまで待ってから、/var/log/guard/diag/current/tap/ の下の msg-dump ファイルを確認します。)

   (r) IPQ トラフィックを記録する

   (s) 状態マシン情報をダンプする

   (t) スロットル・パラメーターを構成する

3. 選択内容にかかわらず、操作の実行期間を選択するよう求めるプロンプトが出されます。期間を選択し、Enter キーを押します。
4. 指定した期間プログラムが実行されることが通知され、Enter キーを押すよう求めるプロンプトが出されます。 Enter キーを押して待機します。
5. 処理が完了すると、メッセージが表示されます。「File Summary」コマンドを使用して、このコマンドの出力を表示できます。このコマンドにより大量のデータが生成される場合があるため、テキスト・エディターを使用してファイルの内容を表示できる別のシステムにファイルをエクスポートすることをお勧めします。(現行セッション・データを圧縮し、このセクションで前述したように記録をエクスポートします。)

3.13 Show Indexes

このコマンドを使用して、さまざまな内部表の索引を表示します。

1. 「Interactive Queries」メニューから、「Show Indexes」を選択します。
2. 表を選択するよう求めるプロンプトが出されます。表を選択し、Enter キーを押してその表の索引を表示します。
3. 上矢印キーおよび下矢印キーを使用して、表示をスクロールします。完了後、Enter キーを押します。

3.14 S-TAP Check

このコマンドを使用して、S-TAP 定義およびトラフィック情報を表示します。

1. 「Interactive Queries」メニューから、「S-TAP Check」を選択します。
2. システムのユニット・タイプが数値形式で表示されます。Enter キーを押します。
3. S-TAP トラフィックをモニターする秒数を選択するよう求めるプロンプトが出されます。上矢印キーおよび下矢印キーを使用して選択を行い、Enter キーを押します。
4. 出力を待機するおおよその時間が通知され、Enter キーを押すよう求めるプロンプトが出されます。Enter キーを押します。
5. 「S-TAP Definitions」レポートおよび「Server Traffic」レポートが表示されます。レポートの確認が完了した後で、Enter キーを押します。

3.15 Interface Link Status

このコマンドを使用して、インターフェース・リンクの状況を表示します。

1. 「Interactive Queries」メニューから、「Interface link status」を選択します。
2. すべてのインターフェースの状況が表示されます。上矢印キーおよび下矢印キーを使用して、表示をスクロールします。
3. 完了後、Enter キーを押します。このコマンドにより表示されるのは、リンクの状況のみであることに注意してください。インターフェース構成情報を表示するには、show network interface all CLI コマンドを使用します。

3.16 Show Throttle Data

このコマンドを使用して、スロットル・データを表示します。

1. 「Interactive Queries」メニューから、「Show Throttle data」を選択します。
2. Enter キーを押して、スロットル統計を 3 秒間待機します。
3. 上矢印キーおよび下矢印キーを使用して表示をスクロールし、完了した後で「Exit」を押します。

3.17 Generate TCP dump and slon

このコマンドを使用して、TCP ダンプを作成し、パケットをトラッキングする slon ユーティリティーを実行します。通常は、技術サポートの指示があった場合にのみ、このコマンドを実行します。上記の個別トピック Generate TCP dump および Slon Utility を参照してください。

3.18 Generate SSL dump

このコマンドを使用して、SSL ダンプを作成します。

1. 「Interactive Queries」メニューから、「Generate SSL dump」を選択します。
2. インターフェースを選択し、「OK」を押します。フィルター IP アドレスを入力し、「OK」を押します。フィルター・ポート番号を入力し、「OK」を押します。
3. 実行期間を選択し、「OK」を押します。「OK」を押し、TCP ダンプを収集するために指定した時間待機します。
4. SSL ダンプを表示する場合は、「OK」を押します。
5. 完了後、「Exit」を押します。

3.19 View bash history

このコマンドを使用して、bash 履歴を表示します。

1. 「Interactive Queries」メニューから、「View Bash History」を選択します。
2. 「OK」を押します。
3. 上矢印キーおよび下矢印キーを使用して表示をスクロールし、完了した後で「Exit」を押します。

3.20 Generate GDM_Error dump

このコマンドを使用して、GDM_ERROR ダンプを作成します。

1. 「Interactive Queries」メニューから、「Generate GDM_Error dump」を選択します。
2. 「OK」を押し、パスワードを入力します。Enter キーを押します。
3. 上矢印キーおよび下矢印キーを使用して表示をスクロールし、完了した後で「Exit」を押します。

3.21 Prepare Tomcat Memory dump

Tomcat では最初にメモリー不足エラーを検出したときに、/var/tmp/tomcat/tomcat.dmp にメモリー・ダンプを行います。このコマンドを使用して、このファイルを圧縮し、暗号化して、/var/log/guard/diag/tomcat/ に移動し、ファイル・サーバーがこのファイルを取得できるようにします。

1. 「Interactive Queries」メニューから、「Prepare Tomcat Memory dump」を選択します。
2. 「OK」を押します。
3. 上矢印キーおよび下矢印キーを使用して表示をスクロールし、完了した後で「Exit」を押します。

3.22 拡張ネットワーク情報

システムの対話式照会の 下にある「拡張ネットワーク情報」オプションをクリックして、 ネットワーク診断情報を表示します。

例

SQLGuard Diagnostics

Network Parameters from ADMINCONSOLE_PARAMETER:

SYSTEM_NETMASK1: 255.255.255.0

SYSTEM_DOMAIN:

SYSTEM_DEFAULT_ROUTE:

SYSTEM_DNS1:

SYSTEM_DNS2:

SYSTEM_DNS3:

TOMCAT_IP:

MANAGER_IP:

HOST_MAC_ADDRESS:

SECOND_DEVICE:

3.23 Generate TCP dump in rotation

この選択肢は、「Generate TCP Dump」セクションと「Generate TCP dump and slon」にある他の diag の選択肢とは異なります。

「Generate TCP dump in rotation」の場合、フィルター IP アドレスを入力します (すべての IP にブランクを入力します)。次に、フィルターのポート番号を入力します。循環の TCP ダンプが既に実行中である場合は、実行期間を尋ねる質問で、「Rotation OFF」または「Rotation」(ON) のいずれかのオプションを選択します。「Rotation」を選択した場合は、ファイル・サイズを追加してください。

TCP ダンプが、/var/log/guard/tcp.bin1 と /var/log/guard.bin2 に交替で出力されます。

プロセス loop_tcpdump.sh を停止するには、「TCP dump in rotation」をもう一度選択します。

4 Perform Maintenance Actions

メインメニューから「Perform Maintenance Actions」オプションを選択し、「Maintenance」メニューを開きます。これらのコマンドは、必ず技術サポートの指示を受けて使用してください。これらのコマンドを定期的に実行する必要はありません。

4.1 TURBINE analysis (update index cardinality)

このコマンドを使用して、Guardium の内部データベースで索引のカーディナリティーを最適化します。操作の実行中は、進行状況表示バーが表示されます。操作が完了すると、「Maintenance」メニューに戻ります。

4.2 TURBINE optimize (rebuild indexes, takes longer)

このコマンドを使用して、Guardium の内部データベースを分析し、再索引します。

1. 「Maintenance」メニューから、「TURBINE optimize (index cardinality)」を選択します。操作の実行中は、進行状況表示バーが表示されます。操作が完了すると、「Maintenance」メニューに戻ります。

4.3 Clean disk space

このコマンドを使用して、使用されていないディスク・スペースのクリーンを行います。手順が完了すると、「Maintenance」メニューに戻ります。

1. 「Maintenance」メニューから、「Clean disk space」を選択します。ディレクトリーを選択するよう求めるプロンプトが出されます。
2. ファイルを削除するディレクトリーを選択します。ディレクトリーの内容がリストされ、すべてのファイルを削除することの確認を求めるプロンプトが出されます。
3. 操作が完了すると、「Maintenance」メニューに戻ります。

4.4 RAID maintenance

このコマンドは、必ず技術サポートを指示を受けて使用してください。このコマンドでは、RAID ドライブの状況を表示するために使用できる、RAID コントローラー・ユーティリティー・プログラムの管理メニューに対するアクセス権限を提供します。ご使用のシステムに RAID コントローラーがない場合は、このコマンドを選択するとエラー・メッセージが表示されます。RAID コントローラー・ユーティリティー・プログラムで提供される機能の中には、ディスク上のすべての情報を消去するものがあるため、このプログラムを使用する際には十分に注意してください。

4.5 Application Debugging Utility

このコマンドを使用して、デバッグをオンまたはオフにします。ロギングを使用可能または使用不可にするか、システム・デフォルトにリセットするよう求めるプロンプトが出されます。

4.6 Modify TURBINE watchdog threshold

このオプションを使用して、長時間かかる照会に対するタイムアウト制限を変更します。

4.7 Force unrecoverable MySQL to start

このオプションは、技術サポートの指示があった場合にのみ使用してください。

4.8 Transfer backups and system recovery

このコマンドを使用して、バックアップされた内部データベースをリストアします。操作の確認を求めるプロンプトが出されます。

4.9 Tomcat Logging Level

このコマンドを使用して、コンポーネントのデバッグ・レベルを選択します。次のオプションのいずれかを選択してください。

「Classifier」、「Data Level Security」、「Workflow」、または「Other」。

「Classifier」を選択して、デバッグ・レベル・オプション (ERROR、WARN、INFO、DEBUG、ALL) を選択します。

「DLS (data level security)」、「Workflow」、または「Other (text input)」を選択して、デバッグ・レベル・オプション (ERROR、WARN、INFO、DEBUG、ALL) を選択します。

Other を選択する場合は (コンマ区切りによるテキスト入力)、有効なコンポーネント (DLS、ワークフロー、監査、カスタム表、GUI、その他、ジョブ) を入力します。

4.10 Aggregator Maintenance

アグリゲーターの完全な分析およびリカバリーです。このユーティリティーは、AGG 関連ログを収集して diag エクスポート・フォルダーに配置し、Aggregator Fix Schema を呼び出してすべてのデータベースのスキーマを同期し、AGG ワークスペースのクリーンを行い、マージ処理を再開して、インポートされたすべての表の完全な分析が確実に行われるようにします (バックグラウンドで実行され、完了までに数時間かかる場合があります)。

4.11 Aggregator Fix Schema

インポートされたすべての表のスキーマを、最新のパッチ・レベルにします (バックグラウンドで実行され、完了までに数時間かかる場合があります)。

4.12 Clean Static Orphans

このオプションは、静的表が増えすぎたために消去する必要がある場合にのみ、技術サポートが使用する必要があります。このユーティリティーは、関連付けられているインスタンスがない古い構成レコードをすべて消去します。(コレクターまたはアグリゲーターで使用する) 静的オーファンの消去中には、進行状況メッセージが表示されます。

5 Exit to CLI

メインメニューで「Exit to CLI」を選択します。Enter キーを押して diag コマンドを閉じ、CLI に戻ります。

親トピック: CLI の概要

ファイル処理 CLI コマンド

これらのコマンドは、システム情報のバックアップとリストアに使用します。これらのタスクの多くは、Guardium® ユーザー・インターフェースから実行できます。

アーカイブ・データ・ファイル名について

Guardium データがアーカイブ (またはアグリゲーターにエクスポート) されると、日ごとに別のデータ・ファイルができます。エクスポート/パージ操作またはアーカイブ/パージ操作の構成によって、同日のエクスポート・データのコピーが複数できる場合があります。アーカイブ・データ・ファイル名とエクスポート・データ・ファイル名の形式は同じで、次のようになります。

<daysequence>-<hostname.domain>-w<run_datestamp>-d<data_date>.dbdump.enc

daysequence は、アーカイブ・データの日付を表す数値で、0 年からの日数として表現されます。名前の data_date 部分では同じ日付が yyyy-mm-dd 形式で表されます。

hostname.domain は、アーカイブが作成された Guardium アプライアンスのホスト名で、その後にドット文字とドメイン・ネームが続きます。

run_datestamp は、データがアーカイブまたはエクスポートされた日付で、yyyymmdd.hhmmss 形式で表されます。

data_date は、アーカイブ・データの日付で、yyyy-mm-dd 形式で表されます。

例: 732423-g1.guardium.com-w20050425.040042-d2005-04-22.dbdump.enc

backup config

これらのコマンドは、内部管理表にある構成情報のバックアップとリストアを行います。 backup config コマンドは、/media/backup ディレクトリーにデータを保管します。backup config コマンドは、ライセンスなどのマシン固有の情報を削除します。 backup system コマンドは、構成およびシステム全体をさらに包括的にバックアップします。

構文

backup config

restore config

backup system

このトピックでは、Guardium 内部データベースに対するバックアップ操作とリストア操作を説明します。 構成情報のみ、またはシステム全体のどちらかをバックアップまたはリストアできます (システム全体とは、データに構成情報が加わったものです。ただし、共有パスワード・ファイルは除きます。このファイルのバックアップとリ

ストアは別に行われます。aggregator backup keys file および aggregator restore keys file コマンドを参照してください。)。これらのコマンドは検査エンジンと Web サービスをすべて停止し、操作完了後にそれらを再始動します。

ファイルをリストアする前に、そのファイルを作成したシステムのシステム共有パスワードをアプライアンスが使用できるようにしておいてください (そうしないと、情報の暗号化を解除できません)。 「Guardium 管理者ガイド」の『システム共有パスワードについて』を参照してください。
注: システム・リストアは、システム・バックアップと同じパッチ・レベルに対して実行する必要があります。例えばバージョン 7.0 パッチ 7 の時点でアプライアンスをバックアップした後、新しく構築したアプライアンスにこのバックアップをリストアするには、まずバージョン 7.0 のパッチ 1 から 7 までをアプライアンスにインストールした後で、ファイルをリストアする必要があります。
リストア処理には、次の 2 つのコマンドが関係します。

- import file - アーカイブ・バックアップ・ファイルをシステムに返します。
- restore system - import file 操作によって既に返されているバックアップ・ファイルからシステムをリストアします。

backup、import、および restore コマンドのすべてで、どのストレージ・システムが構成されているか、およびリストア操作のタイプに応じて、以下の項目を組み合わせて提供する一連のプロンプトが出されます。 操作に合わせて各プロンプトに応答してください。次の表に、プロンプトが出される対象になる情報を示します。

注:

SCP/FTP/TSM/Centera ファイル転送の 1 コピーが保存されます (転送が成功したか失敗したかは無関係)。ファイルによっては再生成に数時間かかることがあるので (例えばシステム・バックアップ)、すぐに使用できるコピーがあることは (特にファイル転送が失敗した場合)、ユーザーにとって価値があります。 各ファイル・タイプ (アーカイブ/システム・バックアップ/構成バックアップなど) に対して 1 コピーのみ保持されます。

バックアップ・システムは現在のライセンス、課金、およびデータ・ソース数をコピーしてから、データをバックアップします。リストア・システムはデータをリストアしてから、ライセンス、課金、およびデータ・ソース数をリストアします。 このシーケンスは、通常のリストア・システムにも当てはまります。以前のシステムからリストアする場合は、ライセンス、課金、およびデータ・ソース数の再構成が必要になります。

バックアップの構成時にポート番号の値が「0」である場合、デフォルトのポートがそのプロトコルに使用されていてそれを変更する必要がないことを示しています。

表 1. backup system

項目	記述
SCP、FTP、TSM、Centera、Snapshot	ファイルの転送に使用する方式を選択します。 TSM と Centera は、転送に使用するストレージ方式が使用可能に設定されている場合のみ表示されます (store storage-method コマンドを参照)。
Data または Configuration	定義と構成情報のみをバックアップするには、「Configuration」を選択します。構成情報に加えてデータもバックアップするには、「Data」を選択します。
restore from archive または restore from backup	アーカイブ・データをリストアするには、「restore from archive」を選択します。構成情報をリストアするには、「restore from backup」を選択します。
normal または upgrade	同じソフトウェア・バージョンの Guardium からリストアする場合は、「normal」を選択します。 Guardium アプライアンスのソフトウェア・アップグレードの後で構成情報をリストアする場合は、「upgrade」を選択します。
host	バックアップ・ファイルのリモート・ホスト。
remote directory	バックアップ・ファイルのディレクトリー。 FTP の場合は、使用する FTP ユーザー・アカウントの FTP ルート・ディレクトリーからの相対ディレクトリー・パスです。SSH の場合、このディレクトリー・パスは絶対ディレクトリー・パスです。 Windows SSH サーバーの場合は、Windows スタイルの円記号ではなく、Unix スタイルのスラッシュを使用したパス名にします。
username	操作に使用するユーザー・アカウント名 (バックアップ操作の場合、このユーザーには指定したディレクトリーに対する書き込み/実行権限が必要です)。 **注:** Windows の場合、ドメイン・ユーザーは domain¥user の形式にしてください。
password	ユーザー名のパスワード。
file name	アーカイブ・ファイルまたはバックアップ・ファイルのファイル名。 『アーカイブ・データ・ファイル名について』を参照してください。 ファイル名の中にワイルドカード文字 * を使用することにより、複数のファイルを選択できます。転送方式として FTP、SCP、および Snapshot を使用する場合、ワイルドカード文字 * を使用できます。TSM または Centera 転送方式では、ワイルドカード文字 * は使用できません。
Centera server	Centera サーバー名を入力します。PEA ファイルを使用する場合は、形式 <Host name/IP>? <full PEA file name> を使用します。例えば、次のように入力します。 128.221.200.56?/var/centera/us_profile_rwqe.pea.txt
Centera clipID	Centera リストア操作で、バックアップ操作から返されるコンテンツ・アドレス。 例: 6M4B15U4JM4LBeDGKCPF9VQO3UA

バックアップまたはリストア操作に必要な情報をすべて提供すると、操作の結果を通知する一連のメッセージが表示されます。 例えば restore system 操作の場合、メッセージは次のようなものになります (リストアのタイプと使用されるストレージ方式によって異なります)。

```
gpg: Signature made Thu Feb 22 11:38:01 2009 EST using DSA key ID 2348FF9E gpg: Good signature from "Backup Signer
<support@guardium.com>" Proceeding to shutdown services Proceeding to startup services Safekeeping admin.xreg Safekeeping
client.xreg Safekeeping controllers.xreg Safekeeping controls.xreg Safekeeping guardium-portlets.xreg Safekeeping local-
portlets.xreg Safekeeping local-security.xreg Safekeeping local-skins.xreg Safekeeping media.xreg Safekeeping portlets.xreg
Safekeeping security.xreg Safekeeping skins.xreg guard_sniffer.pl -reorder Recovery procedure was successful. ok
```

バックアップ/アーカイブのスクリプトによる /var 容量の使い尽くしの防止

バックアップ・プロセスは、実行前に /var の空き容量をチェックして失敗を防止します。 このプロセスは、バックアップ用のスペースが十分にない場合にも、 ユーザーに警告を出します。

アーカイブ・プロセスは、 静的表のサイズをチェックし、アーカイブを作成できる空き容量が /var にあることを 確認します。

バックアップが 50% を超えると、 ログ・ファイルおよび GUI にエラーが記録されるようになっています。

例:

```
ERROR: /var backup space is at 60% used. Insufficient disk space for backup. CLI> backup system     1. DATA     2. CONFIGURATION
Please enter the number of your choice: (q to quit) 1      1. SCP     2. CONFIGURED DESTINATION  Enter the number of your choice:
(q to quit) 2 Make sure destination is configured in the GUI under the System Backup option Please wait, this may take some time.
```

export audit-data

指定された日付 (yyyy-mm-dd) の監査データを、さまざまな内部 Guardium 表から圧縮アーカイブ・ファイルにエクスポートします。指定した日付のデータは、/var/dump ディレクトリーの圧縮アーカイブ・ファイルに保管されます。 作成されたファイルは、システムが生成するメッセージで示されます。 例を参照してください。このコマンドは、必ず Guardium サポートの指示に従って使用してください。

注: このコマンドを実行できるのは、admin ロールが設定されたユーザーのみです。

構文

export audit-data <yyyy-mm-dd>

例

```
If you enter the audit-data command for the date 2005-09-16, a set of messages similar to the following will be created: CLI>
export audit-data 2005-09-16 2005-09-16 Extracting  GDM_ACCESS  Data ... Extracting  GDM_CONSTRUCT  Data ... Extracting
 GDM_SENTENCE  Data ... Extracting  GDM_OBJECT  Data ... Extracting  GDM_FIELD  Data ... Extracting  GDM_CONSTRUCT_TEXT  Data ...
Extracting  GDM_SESSION  Data ... Extracting  GDM_EXCEPTION  Data ... Extracting  GDM_POLICY_VIOLATIONS_LOG  Data ... Extracting
 GDM_CONSTRUCT_INSTANCE  Data ... Generating tar file ...  /var/csvGenerationTmp ~ GDM_ACCESS.txt GDM_CONSTRUCT.txt
GDM_CONSTRUCT_INSTANCE.txt GDM_CONSTRUCT_TEXT.txt GDM_EXCEPTION.txt GDM_FIELD.txt GDM_OBJECT.txt GDM_POLICY_VIOLATIONS_LOG.txt
GDM_SENTENCE.txt GDM_SESSION.txt ~ Generation completed, CSV Files saved to /var/dump/732570-supp2.guardium.com-w20050919110317-
d2005-09-16.exp.tgz ok
```

名前付の内部データベース表それぞれのデータが、CSV 形式でテキスト・ファイルに書き込まれます。 アーカイブ・ファイルの名前の最後に exp.tgz が付けられ、名前の残りの部分は『アーカイブ・データ・ファイル名について』での説明のとおりに形成されます。

export file コマンドを使用して、このファイルを別のシステムに転送できます。

delete audit-data

このコマンドは、必ず Guardium サポートの指示に従って使用してください。 このコマンドは、圧縮監査データ・ファイルを削除するために使用します。 削除するファイルを特定する索引番号を入力する必要があります。 アーカイブ・データ・ファイル名の形式については、『アーカイブ・データ・ファイル名について』を参照してください。

削除するファイルを特定するようにプロンプトが出されます。

構文

delete audit-data

show audit-data

このコマンドは、CLI コマンド export audit-data を実行して作成されたすべてのファイルを表示するために使用します。監査データ・ファイルについて詳しくは、『export audit-data』を参照してください。

構文

show audit-data <yyyy-mm-dd>

export file

このコマンドは、/var/IBM/Guardium/data/dump、/var/log、または /var/IBM/Guardium/data/importdir ディレクトリーから、filename という名前の単一ファイルをエクスポートします。

このコマンドは、必ず Guardium サポートの指示に従って使用してください。 Guardium データをアグリゲーターにエクスポートするか、データをアーカイブするには、「管理コンソール」パネル上の該当するメニュー・コマンドを使用します。

構文

export file </local_path/filename> <user@host:/path/filename>

local_path は、/var/IBM/Guardium/data/dump、/var/log、または /var/IBM/Guardium/data/importdir のいずれかでなければなりません。

fileserver

このコマンドは、Guardium アプライアンス上で実行される HTTP ベースのファイル・サーバーを開始するために使用します。 このファシリティーは、ユニットへのパッチのアップロード、またはユニットからのデバッグ情報のダウンロードを容易に実行できるようにすることを目的としています。このファシリティーは開始のたびに、パッチのアップロード先のディレクトリーに含まれるすべてのファイルを削除します。

注: ファイル・サーバーがアクセスすることになるファイルを生成する操作は、ファイル・サーバーの開始前に完了する必要があります (ファイルをファイル・サーバーが使用できるようにするため)。

構文

fileserver [IP address] [duration]

`IP address` は、指定された IP アドレスからファイル・サーバーへのアクセスを可能にするオプション・パラメーターです。デフォルト (パラメーターなし) では、アクセスは、ファイル・サーバーを開始した SSH クライアントの IP アドレスに制限されます。

`duration` は、ファイル・サーバーがアクティブである秒数を指定するオプション・パラメーターです。指定された秒数が経過すると、ファイル・サーバーは自動的にシャットダウンします。期間は 60 秒から 3600 秒までの任意の秒数に設定できます。

ブラウザー・セッションがプロキシー・サーバー経由でリダイレクトされるセキュリティー設定では、ファイル・サーバー・クライアントの IP アドレスは、ファイル・サーバーを開始した SSH クライアントと同じにはなりません。その代わり、ファイル・サーバー・クライアントはプロキシー・サーバーの IP アドレスを保持し、このアドレスはオプションの `ip address` パラメーターを渡す必要があります。プロキシー IP アドレスを見つけるには、「Guardium モニター」インターフェースの「Guardium へのログイン」レポートに表示されるブラウザー設定またはクライアント IP アドレスを確認します。

例

```
fileserver 10.0.0.1 3600
Starting the file server...
The file server is ready at https://guardium.system.com:8445
The timeout has been set to 3600 seconds and it may timeout during the uploading.

The upload will only be accessible from the IP you are logged in from: 10.0.0.1
ファイル・サーバーを停止するには ENTER を押してください。
```

ブラウザー・ウィンドウでファイル・サーバーを開き、以下のいずれかを実行します。

- パッチをアップロードするには、「Upload a patch」をクリックし、指示に従います。
- ログ・データをダウンロードするには、「Sqlguard logs」をクリックし、目的のファイルにナビゲートして、他のファイルの場合と同様にダウンロードします。

完了した後で CLI セッションに戻り、Enter を押してセッションを終了します。

fileserver を使用した VA および資格スクリプトへのアクセス方法

操作手順

CLI から、「fileserver <デスクトップ IP> 3600」を実行します。

脆弱性評価:

ブラウザーを開き、https://<アプライアンス ip>/log/debug-logs/gdmmonitor_scripts/ にアクセスします。

ご使用のデータベース・タイプに一致するファイルを選択します。

資格:

ブラウザーを開き、https://<アプライアンス IP>/log/debug-logs/entitlemnts_monitor_role/ にアクセスします。

ご使用のデータベース・タイプに一致するファイルを選択します。

import file

『backup config』および『restore config』を参照してください。

import file CLI コマンドでは、scp、ftp、および snapshot 方式の場合、ファイル名にワイルドカード * を使用できます。

構文

import file

import tsm config

TSM クライアント構成ファイルを Guardium アプライアンスにアップロードします。この操作は、TSM を使用するアーカイブまたはバックアップ操作を実行する前に行う必要があります。 どの場合も、dsm.sys ファイルをアップロードする必要があります。また、このファイルに servername セクションが複数ある場合は、dsm.opt ファイルもアップロードする必要があります。 これらのファイルの作成方法については、お客様の会社の TSM 管理者に確認してください。

指定したホストのユーザー・アカウントのパスワードを求めるプロンプトが出されます。

構文

import tsm config <user@host:/path/[dsm.sys | dsm.opt]>

パラメーター

user@host - 指定したホスト上のファイルにアクセスするためのユーザー・アカウント。

/path/[dsm.sys | dsm.opt] - インポートするファイルの絶対パス・ファイル名。

注: 各コレクターに TSM を設定する場合、初期構成に失敗すると、テスト・ファイルを送信できなかったという通知エラーが出される結果になります。コレクターに root としてログインし、TSM サーバーに対して dsmc アーカイブ・コマンドを実行すると、同じ資格情報で TSM ファイルを構成できます。 GUI に戻り、前に使用したオプションと同じオプションを使用しても構成できます。

tsm config に passwordaccess=generate が含まれている場合、ローカル・ファイルに格納されているパスワードが探索されます。このローカル・パスワード・ファイルを作成するために、root ユーザーは dsmc コマンドを 1 回実行する必要があります。

tsm config ファイルのアップロード後、tsm config に「passwordaccess generate」プロンプトが含まれていれば、「passwordaccess」が生成されるように設定されます。

```
Would you like to run a dsmc command now to ensure password is set locally (y/n)?     If the answer is y, run a "dsmc query
options>>/dev/null" command, which will prompt user for password.
```

import tsm property

この CLI コマンドを使用して、/opt/tivoli/tsm/client/ba/bin/guard_tsm.properties にファイルをアップロードします。

ファイルのサイズは、1K にしてください。

構文

import tsm property user@host:file

このコマンドを実行すると、入力ファイルが /opt/tivoli/tsm/client/ba/bin/guard_tsm.properties にアップロードされます。

restore config

これらのコマンドは、内部管理表にある構成情報のバックアップとリストアを行います。 backup config コマンドは、/media/backup ディレクトリーにデータを保管します。backup config コマンドは、ライセンスなどのマシン固有の情報を削除します。 backup system コマンドは、構成およびシステム全体をさらに包括的にバックアップします。

構成をリストアするときには、バックアップ作成時の元のアプライアンスと同じバージョン、同じパッチ・レベルのバックアップをリストアする必要があります。

構文

backup config

restore config

restore db-from-prev-version

このコマンドは、直前のシステムからバックアップを取り (バックアップ・データの提供が必要、構成バックアップはオプション)、最新のシステム上でリストアを実行します。これには、データやポートレットなどのアップグレードが含まれます。

Guardium システムをアップグレードする前に、システムのフルバックアップを実行します。何らかの理由でアップグレードに失敗し、マシンが使用できない状況になった場合、アップグレードを修正して再実行を試みるのではなく、マシンを最新のシステムとして再構築し、この最新システムを基本ネットワーク情報 (IP、リゾルバー、経路、システム・ホスト名、およびドメイン) のみによって設定します。

結果として、前のシステムのデータとカスタマイズ (構成ファイルが提供された場合) が取り込まれた最新のシステムになります。

まず、以前のシステムから最新システムへの通常のアップグレードを試行してください。正常にアップグレードできなかった場合に、以前のシステムから最新システムにアップグレードするための代替方法として、backup を使用してください。

注: (調査センターではなく) アグリゲーターにリストアされる古いデータで、マージ期間外のデータは、マージ期間が変更されてマージ・プロセスが再実行されるまで表示されません。

このコマンドを実行するには、現在のサーバーのデータと構成の両方をバックアップします。バックアップが完了した後、最新のリリースを同じサーバーにインストールしてください。次に、CLI から import file コマンドを使用して、データ・ファイルと構成ファイルの両方をインポートします。2 つのバックアップ・ファイルがインポートされた後、再度 CLI からコマンド restore db-from-prev-version を実行します。 これにより、古いバージョンからのバックアップ・ファイル (データと構成) が、新しくインストールされたサーバーにリストアされます。
注: Guardium を英語以外の言語で使用している場合には、restore CLI コマンドにより、レポート・ヘッダーなどの一部の文字列が英語に設定されます。このような文字列を英語以外の言語で表示するには、restore CLI コマンドを実行した後に、store language CLI コマンドを実行します。

オプション・パラメーターの「override」は、バックアップからの中央マネージャー・アプライアンスのリストアにのみ適用できます。

デフォルトでは、ユーザーが中央マネージャー・アプライアンス上で「restore db-from-prev-version」コマンドを実行すると、管理対象の管理対象ユニットにリンクする、この中央マネージャーにある既存の構成情報を保存します。

ユーザーが restore コマンドに「override」を追加すると、既存の中央マネージャー/管理対象ユニット構成は、バックアップ・データからの中央マネージャー/管理対象ユニット構成によってオーバーライドされます。

構文

restore db-from-prev-version [override]

例

restore db-from-prev-version

restore db-from-prev-version override

注: この CLI コマンドを使用すると、「S-TAP と管理対象ユニットの関連付け」の管理対象ユニットと S-TAP の関連付けは復元されません。ユーザーは関連付けを再度定義する必要があります。
構文

```
restore db-from-prev-version

This procedure will restore and upgrade a previous backup on a newly-installed latest system. If the older files are currently
located on a remote system, use the "import file" cli command to transfer them locally prior to running this procedure. The
imported files will be put in the /var/dump/ directory. Continue (y/n)?
```

注:

CLI コマンド restore db-from-prev-version の実行中に次の質問に Y (はい) と応答すると、非標準装備/カスタマイズ・タイプのすべてのレポートとペインが圧縮されて「v.x.0 カスタム・レポート」という名前の 1 つのペインに入ります。

これらの同じ質問に N (いいえ) と応答すると、すべてのペインが以前のバージョンの状態にリストアされます。

```
Update portal layout (panes and menus structure) to the new v8 default (current instances of custom reports will be copied to the
new layout, as well as parameter changes on predefined reports) for the user admin? (y/n) n  Update portal layout (panes and menus
structure) to the new v8 default (current instances of custom reports will be copied to the new layout, as well as parameter
changes on predefined reports) for all other users? (y/n)
```

restore keystore

このコマンドは、必ず技術サポートの指示に従って使用してください。

このコマンドは、Web サーブレット・コンテナー環境 (Tomcat) によって使用される認証と秘密鍵をリストアするために使用します。

構文

restore keystore

restore pre-patch-backup

このコマンドは、必ず技術サポートの指示に従って使用してください。

このコマンドは、アプライアンス・データベースが稼働時または停止時に pre-patch-backup をリカバリーするために使用します。

構文

```
restore pre-patchbackup Please enter the information to retrieve the file: Is the file in the local system? (y/n) n Start to
recover with the backup profile parameters. Please check the recovery status in the log
/var/log/guard/diag/depot/patch_installer.log ok ------------------------------------ If answer 'n', abort the operation. If
answer 'y', need to enter the file name.
```

restore system

このトピックでは、Guardium 内部データベースに対するバックアップ操作とリストア操作を説明します。 構成情報のみ、またはシステム全体のどちらかをバックアップまたはリストアできます (システム全体とは、データに構成情報が加わったものです。ただし、共有パスワード・ファイルは除きます。このファイルのバックアップとリストアは別に行われます。aggregator backup keys file および aggregator restore keys file コマンドを参照してください。)。これらのコマンドは検査エンジンと Web サービスをすべて停止し、操作完了後にそれらを再始動します。

ファイルをリストアする前に、そのファイルを作成したシステムのシステム共有パスワードをアプライアンスが使用できるようにしておいてください (そうしないと、情報の暗号化を解除できません)。 「Guardium 管理者ガイド」の『システム共有パスワードについて』を参照してください。

注: システム・リストアは、システム・バックアップと同じパッチ・レベルに対して実行する必要があります。
リストア処理には、次の 2 つのコマンドが関係します。

- import file - アーカイブ・バックアップ・ファイルをシステムに返します。
- restore system - import file 操作によって既に返されているバックアップ・ファイルからシステムをリストアします。

backup、import、および restore コマンドのすべてで、どのストレージ・システムが構成されているか、およびリストア操作のタイプに応じて、以下の項目を組み合わせて提供する一連のプロンプトが出されます。 操作に合わせて各プロンプトに応答してください。次の表に、プロンプトが出される対象になる情報を示します。

注:

SCP/FTP/TSM/Centera ファイル転送の 1 コピーが保存されます (転送が成功したか失敗したかは無関係)。ファイルによっては再生成に数時間かかることがあるので (例えばシステム・バックアップ)、すぐに使用できるコピーがあることは (特にファイル転送が失敗した場合)、ユーザーにとって価値があります。 各ファイル・タイプ (アーカイブ/システム・バックアップ/構成バックアップなど) に対して 1 コピーのみ保持されます。

バックアップ・システムは現在のライセンス、課金、およびデータ・ソース数をコピーしてから、データをバックアップします。リストア・システムはデータをリストアしてから、ライセンス、課金、およびデータ・ソース数をリストアします。 このシーケンスは、通常のリストア・システムにも当てはまります。以前のシステムからリストアする場合は、ライセンス、課金、およびデータ・ソース数の再構成が必要になります。

表 2. restore system

項目	記述
SCP、FTP、TSM、Centera、Snapshot	ファイルの転送に使用する方式を選択します。 TSM と Centera は、転送に使用するストレージ方式が使用可能に設定されている場合のみ表示されます (store storage-method コマンドを参照)。
Data または Configuration	定義と構成情報のみをバックアップするには、「Configuration」を選択します。構成情報に加えてデータもバックアップするには、「Data」を選択します。
restore from archive または restore from backup	アーカイブ・データをリストアするには、「restore from archive」を選択します。構成情報をリストアするには、「restore from backup」を選択します。
normal または upgrade	同じソフトウェア・バージョンの Guardium からリストアする場合は、「normal」を選択します。 Guardium アプライアンスのソフトウェア・アップグレードの後で構成情報をリストアする場合は、「upgrade」を選択します。
host	バックアップ・ファイルのリモート・ホスト。
remote directory	バックアップ・ファイルのディレクトリー。 FTP の場合は、使用する FTP ユーザー・アカウントの FTP ルート・ディレクトリーからの相対ディレクトリー・パスです。SSH の場合、このディレクトリー・パスは絶対ディレクトリー・パスです。Windows SSH サーバーの場合は、Windows スタイルの円記号ではなく、Unix スタイルのスラッシュを使用したパス名にします。
username	操作に使用するユーザー・アカウント名 (バックアップ操作の場合、このユーザーには指定したディレクトリーに対する書き込み/実行権限が必要です)。 **注:** Windows の場合、ドメイン・ユーザーは domain¥user の形式にしてください。
password	ユーザー名のパスワード。

項目	記述
file name	アーカイブ・ファイルまたはバックアップ・ファイルのファイル名。『アーカイブ・データ・ファイル名について』を参照してください。 ファイル名の中にワイルドカード文字 * を使用することにより、複数のファイルを選択できます。転送方式として FTP、SCP、および Snapshot を使用する場合、ワイルドカード文字 * を使用できます。TSM または Centera 転送方式では、ワイルドカード文字 * は使用できません。
Centera server	Centera サーバー名を入力します。PEA ファイルを使用する場合は、形式 <Host name/IP>? <full PEA file name> を使用します。例えば、次のように入力します。 128.221.200.56?/var/centera/us_profile_rwqe.pea.txt サーバー IP と PEA ファイル名の間の ? に注意してください。 この IP アドレスおよび .PEA ファイルは、EMC Centera から取得します。 パスを構成する際には、疑問符が必須です。「.../var/centera/...」を含むパス名でなければバックアップが失敗するため、このパス名は重要です。 .PEA ファイルは、Centera バックアップ要求ごとにアクセス権、ユーザー名、およびパスワード認証を提供します。
Centera clipID	Centera リストア操作で、バックアップ操作から返されるコンテンツ・アドレス。 例: 6M4B15U4JM4LBeDGKCPF9VQO3UA

バックアップまたはリストア操作に必要な情報をすべて提供すると、操作の結果を通知する一連のメッセージが表示されます。 例えば restore system 操作の場合、メッセージは次のようなものになります (リストアのタイプと使用されるストレージ方式によって異なります)。

```
gpg: Signature made Thu Feb 22 11:38:01 2009 EST using DSA key ID 2348FF9E gpg: Good signature from "Backup Signer
<support@guardium.com>" Proceeding to shutdown services Proceeding to startup services Safekeeping admin.xreg Safekeeping
client.xreg Safekeeping controllers.xreg Safekeeping controls.xreg Safekeeping guardium-portlets.xreg Safekeeping local-
portlets.xreg Safekeeping local-security.xreg Safekeeping local-skins.xreg Safekeeping media.xreg Safekeeping portlets.xreg
Safekeeping security.xreg Safekeeping skins.xreg guard_sniffer.pl -reorder Recovery procedure was successful. ok
```

setup help (バックアップ用 2 次ディスク)

R610 R710 アプライアンスにバックアップ用 2 次ディスクを取り付けてください。このディスクはスロット番号 2 に取り付けます。続いて setup snapshotdisk を実行してパーティションを構成し、ドライブをフォーマットした後にマウントします。選択可能な 2 つの CLI は、setup help および setup snapshotdisk です。

構文

setup [help | snapshotdisk | vmware_tools]

store tsm authorization

TSM サーバーで backupinitiationroot が ON に設定されている場合、バックアップおよびアーカイブを実行できるのは root ユーザーと許可されたユーザーのみです。backupinitiationroot が on に設定され、かつ DSM.SYS のパスワード・アクセスが「generate」に設定されていると、TSM に対する Guardium のバックアップおよびアーカイブは次のエラー・メッセージで失敗します。

```
ANS1708E Backup operation failed. Only a root user can do this operation
```

非 root ユーザーは、バックアップおよびアーカイブの実行を許可されている必要があります。

この許可を有効にするには、次の CLI コマンドを実行します。

```
Store tsm authorization backupinitiationroot on
```

この許可を無効にするには、次の CLI コマンドを実行します。

```
Store tsm authorization backupinitiationroot off
```

構文

store tsm authorization backupinitationiroot <on/off>

表示コマンド

show tsm authorization backupinitationiroot <on/off>

この CLI コマンドは、TSM サーバーで backupinitiationroot が ON に設定されている場合に、Guardium の非 root ユーザーがバックアップおよびアーカイブの実行を許可されていると、on と表示します。そうでない場合は、off と表示します。

store language

この CLI コマンドを使用して、ベースライン言語の英語を希望する言語に変更し、データベースをその言語に変換します。 Guardium のインストールは、常に英語で行われます。 Guardium システムは、インストール後に日本語、中国語 (繁体字または簡体字)、フランス語、スペイン語、ドイツ語、またはポルトガル語に変更できます。

CLI コマンド store language はアプライアンスのセットアップと見なされ、アプライアンスの初期セットアップ時に実行されることを目的としています。

特定の言語でアプライアンスをデプロイメントした後にこの CLI コマンドを実行すると、既にキャプチャー、保管、カスタマイズ、アーカイブ、またはエクスポートされた情報を変更することができます。

注: 英語から目的の言語に切り替えた後は、この CLI コマンドを使用してその言語を英語に戻すことはできません。Guardium システムを英語で再インストールする必要があります。

構文

CLI> store language [English | Japanese | SimplifiedChinese | TraditionalChinese | French | German | Spanish | Portuguese]

表示コマンド

show language

setup vmware tools

この CLI コマンドを使用して、ESX インフラストラクチャーで実行される VMware をインストールします。

構文

setup vmware_tools [install | uninstall]

ステップ 1: VM クライアント/コンソールを開き、IBM Guardium アプライアンスが含まれる VM インスタンスを選択します。インスタンスを右クリックし、(ポップアップ・メニューから) ゲスト => VMware ツールのインストール/アップグレードを選択します。これにより、インスタンスがマウント・ポイントを介して VMware ツールにアクセスできるようになります。

ステップ 2: (VM クライアント/コンソール内から) CLI コマンド setup vmware_tools install を実行して、VM ツールをインストールします。

Vmware のリブート後のカーネル・パニック

Guardium を稼働する VMware SX 4.1 仮想マシンが、リブート後にカーネル・パニックを起こすことがあります。

この状況を修正するには、VMware では ESX4.1 の Update 2 をインストールするか、CPU/MMU 仮想化を「Use software for instruction set and MMU Virtualization」に設定することが推奨されています。このオプションは、「Settings」/「Options」/「CPU/MMU」/「Use software for instruction set and MMU Virtualization」にあります。

親トピック: CLI の概要

検査エンジンの CLI コマンド

これらの CLI コマンドは、検査エンジンの構成に使用します。

検査エンジンは 1 つ以上のサーバーからなるサーバー・セットと、1 つ以上のクライアントからなるクライアント・セットとの間の、特定のデータベース・プロトコル (Oracle や Sybase など) を使用したトラフィックをモニターします。 検査エンジンはネットワーク・パケットから SQL を抜き出し、センテンス、要求、コマンド、オブジェクト、およびフィールドを識別する構文解析ツリーをコンパイルして、そのトラフィックについての詳細情報を内部データベースに記録します。

add inspection-engines

検査エンジンのリストの最後に、検査エンジン構成を追加します。 パラメーターを示します。新しい検査エンジンを追加した後に、reorder inspection-engines コマンドを使用して、検査エンジンのリストを再配列できます。検査エンジンを追加しても、その検査エンジンの実行は開始されません。実行を開始するには start inspection-engines コマンドを使用します。

構文

add inspection-engines <name> <protocol>

<fromIP/mask> <port> <toIP/mask>

<exclude client list> <active on startup>

パラメーター

name - 新しい検査エンジンの名前です。これはユニットで固有である必要があります。

protocol - モニター対象のプロトコル。以下のいずれかの値でなければなりません。Windows: CouchDB, DB2, DB2 Exit, Informix, MongoDB, MS SQL, Mysql, Oracle, PostgreSQL, Sybase; UNIX: Aster, Cassandra, CouchDB, DB2, DB2 Exit, exclude IE, FTP, GreenPlumDB, Hadoop, Hive, HTTP, Hue, IBM iSeries, Impala, Informix, Informix Exit, Kerberos, MariaDB, MongoDB, Mysql, Netezza, Oracle, PostgreSQL, SAP HANAVertics, Sybase, Teradata, Vertica, or WebHDFS

fromIP/mask - IP アドレスおよびサブネット・マスクで識別されるクライアントのリスト。 IP アドレスとマスクはそれぞれスラッシュで区切り、複数のエントリーはコンマで区切ります。 アドレスとマスクがすべてゼロの場合は、ワイルドカードです。 除外クライアント・リストのオプションが Y の場合、検査エンジンはこのリストにあるクライアント以外のすべてのクライアントのトラフィックをモニターします。 除外クライアント・リストのオプションが N の場合、検査エンジンはこのリストのクライアントのトラフィックのみをモニターします。

port - 指定したクライアントとデータベース・サーバー間のトラフィックは、このポートまたはポート範囲を使用する場合にモニターされます。 範囲を指定するには、2 つの数字をハイフンでつなぎます。

toIP/mask - トラフィックがモニター対象となるデータベース・サーバーのリスト。IP アドレスとサブネット・マスクで識別されます。 IP アドレスとマスクはそれぞれスラッシュで区切り、複数のエントリーはコンマで区切ります。 アドレスとマスクがすべてゼロの場合は、ワイルドカードです。

exclude client list - Y/N の値をとります。デフォルトは N です。Y の場合、検査エンジンはこのクライアント・リストで識別されるクライアントを除く、すべてのクライアントのトラフィックをモニターします 。N の場合、検査エンジンはクライアント・リストに挙げられたクライアントのトラフィックのみをモニターします。

active on startup - 値は Y または N であり、デフォルトは N です。Y の場合、検査エンジンはシステム始動時に活動化します。

delete inspection-engines

name で識別される、単一の検査エンジンを削除します。 名前は文字、数字、空白のみを含むことができます。検査エンジンの名前に特殊文字が含まれている場合は、管理者ポータルの GUI を使用してこれを削除します。

構文

delete inspection-engines <name>

reorder inspection-engines

検査エンジンの新しい順序を、list inspection-engines コマンドで作成されたリストの索引値を使用して指定します。

構文

reorder inspection-engines <index>, <index>...

例

表示される索引が 1、2、3、4 の場合、次のコマンドによりエンジンの配列が逆になります。

reorder inspection-engines 4,3,2,1

restart inspection-core

検査エンジン・コアを再始動しますが、検査エンジンは再始動しません。 このコマンドが発行されると、データベース・トラフィックの収集は停止します。

構文

restart inspection-core

注: 1 つ以上の特定の検査エンジンのトラフィック収集を再開するには、このコマンドの後に 1 つ以上の start inspection engine コマンドを続けます。 またはすべての検査エンジンのトラフィック収集を再開するには、restart inspection-engines コマンドを使用します。

restart inspection-engines

データベース検査エンジン・コアおよびすべての検査エンジンを再始動します。 これが起こるとデータベース・トラフィックの収集は一時的に停止し、データベース接続が再び開始された場合にのみ、再開されます。

構文

restart inspection-engines

show inspection-engines

以下のような、検査エンジンの構成情報を表示します。

all - すべての検査エンジン。

configuration <index> - 指定した索引 (list inspection-engines コマンドで作成したもの) で識別される検査エンジンのみ。

type <db_type> - 特定のデータベース・タイプの構成を表示。このデータベース・タイプは、サポートされている以下のモニター対象プロトコル・タイプのいずれかである必要がある。Windows: CouchDB, DB2, DB2 Exit, Informix, MongoDB, MS SQL, Mysql, Oracle, PostgreSQL, Sybase; UNIX: Aster, Cassandra, CouchDB, DB2, DB2 Exit, exclude IE, FTP, GreenPlumDB, Hadoop, Hive, HTTP, Hue, IBM iSeries, Impala, Informix, Informix Exit, Kerberos, MariaDB, MongoDB, Mysql, Netezza, Oracle, PostgreSQL, SAP HANAVertics, Sybase, Teradata, Vertica, or WebHDFS。

構文

show inspection-engines <all | configuration <index> | log sqlstrings | type <type> >

注: スパン・ポートなどの非 STAP 検査エンジンを表示するには、CLI コマンド show inspection-engines all を使用します。CLI コマンド list_inspection_engines は、STAP によって作成された検査エンジンを表示します。

start inspection-core

検査エンジン・コアを始動します。

構文

start inspection-core

start inspection-engines

list inspection-engines コマンドで作成されたリストの索引値を使用して識別される、1 つ以上の検査エンジンを始動します。

構文

start inspection-engines <all | id>

start inspection-engines all

すべての検査エンジンを開始します。

構文

start inspection-engine all

start inspection-engines id

使用法: start inspection-engines id <n> (n は数値のスニファー ID)

構文

start inspection-engines id <n>

stop inspection-engines id

使用法: stop inspection-engines id <n> (n は数値のスニファー ID)

stop inspection-core

検査エンジン・コアを停止します。

構文

stop inspection-core

stop inspection-engines

list inspection-engines コマンドで作成されたリストの索引値を使用して識別される、1 つ以上の検査エンジンを停止します。すべての検査エンジンを停止することもできます。

構文

stop inspection-engine <all | id>

stop inspection-engines all

すべての検査エンジンを停止します。

構文

stop inspection-engines all

stop inspection-engines id

list inspection-engines コマンドで作成されたリストの索引値を使用して識別される、1 つ以上の検査エンジンを停止します。

構文

stop inspection-engine <n> (<n> はスニファー ID の数字)

store ignored port list

すべての検査エンジンにより無視されるポート番号の完全なセットを設定します。 指定するリストは既存のリストをすべて置き換えます。リストではそれぞれの数字と次の数字をコンマで区切り、ブランクやその他の空白文字を入れてはなりません。 ハイフンを使用すると、その数字を含めた範囲の範囲指定ができます。

構文

store ignored port list <n>

例

store ignored port list 33,60-70

表示コマンド

show ignored port list

親トピック: CLI の概要

調査ダッシュボードの CLI コマンド

これらの CLI コマンドは、調査ダッシュボードを構成するために使用します。

show solr connection_timeout

このコマンドを使用して、現在の connection_timeout 値を表示します。

```
show solr connection_timeout
```

show solr so_timeout

このコマンドを使用して、現在の so_timeout 値を表示します。

```
show solr so_timeout
```

show solr time_allowed

このコマンドを使用して、現在の time_allowed 値を表示します。

```
show solr time_allowed
```

store solr connection_timeout

このコマンドを使用して、接続タイムアウトを設定します。指定されたタイムアウト期間内に調査ダッシュボードがコレクターに接続できない場合、そのコレクターから結果は返されません。

```
store solr connection_timeout [value]
```

パラメーター	値	記述
connection_timeout	integer	タイムアウトは、0 から 2147483647 ミリ秒の値で表されます。 デフォルト値は 100000 ミリ秒です。

store solr so_timeout

このコマンドを使用して、ソケット・タイムアウトを設定します。

```
store solr so_timeout [value]
```

パラメーター	値	記述
so_timeout	integer	タイムアウトは、0 から 2147483647 ミリ秒の値で表されます。 デフォルト値は 100000 ミリ秒です。

store solr time_allowed

このコマンドを使用して、ソケット・タイムアウトを設定します。

```
store solr time_allowed [value]
```

パラメーター	値	記述
time_allowed	integer	タイムアウトは、0 から 2147483647 ミリ秒の値で表されます。 デフォルト値は 90000 ミリ秒です。 注: 深い検索では、time_allowed の 10 倍の値が使用されます。

親トピック: CLI の概要

ネットワーク構成 CLI コマンド

ネットワーク構成 CLI コマンドは、IP アドレスの設定、結合/フェイルオーバーの処理、2 次機能の処理、およびネットワークのリセットに使用します。

ネットワーク構成 CLI コマンドは、以下の目的で使用します。

- マシンの背面のコネクターを識別する。(show network interface port)
- ネットワーク・カードをインストールした後または移動した後に、ネットワーキングをリセットする。(store network interface inventory)
- IP アドレスを設定する。(store network interface ip、store network interface mask、store network resolver、store network routes defaultroute)
- 高可用性を有効または無効にする。(store network interface high-availability)
- ネットワーク・カードが接続されているスイッチで設定が自動ネゴシエーションされない場合に、ネットワーク・カードを構成する (store network interface auto-negotiation、store network interface speed、store network interface duplex)

制約事項: 一元管理、統合、エンタープライズ・ロード・バランシング、エンタープライズ検索などのすべての Guardium 間通信機能に、ネットワーク・インターフェース ETH0 を使用する必要があります。

restart network

ネットワーク構成のみを再始動します。 例えば、IP アドレスを変更した後に、この CLI コマンドを実行します。

構文

restart network

show network interface all

このコマンドは、Guardium® アプライアンスをデスクトップ LAN に接続するために使用されるネットワークの設定を表示します。IP アドレス、マスク、状態 (有効か無効か) および高可用性状況が表示されます。 IP 高可用性が有効な場合、システムは 2 つのインターフェース (ETH0 および ETH3) を表示します。そうでない場合は ETH0 のみが表示されます。

構文

show network interface all

show network routes operational

使用中の IP ルーティング構成を表示します。

構文

show network routes operational

例

CLI> show net rout ope

Kernel IP routing table

Destination Gateway Genmask Flags Metric Ref Use Iface

192.168.3.0 0.0.0.0 255.255.255.0 U 0 0 0 nic1

169.254.0.0 0.0.0.0 255.255.0.0 U 0 0 0 nic2

0.0.0.0 192.168.3.1 0.0.0.0 UG 0 0 0 nic1

ok

CLI>

show network verify

現行のネットワーク構成を表示します。

構文

show network verify

```
CLI> show network verify

Current Network Configuration
-------------------------------------------------------------------------------
Hostname =
-------------------------------------------------------------------------------
Device     | Address          | Netmask          | Gateway          | Member of
-------------------------------------------------------------------------------
eth0       |
-------------------------------------------------------------------------------
Ethtool Options
-------------------------------------------------------------------------------
Device     | Options (speed,autoneg,duplex)
-------------------------------------------------------------------------------
eth0       |
-------------------------------------------------------------------------------
DNS Servers
-------------------------------------------------------------------------------
Index      | DNS Server
-------------------------------------------------------------------------------
1          |
2          |
-------------------------------------------------------------------------------
Static Routes
-------------------------------------------------------------------------------
Device     | Index      | Address          | Netmask          | Gateway
-------------------------------------------------------------------------------
-------------------------------------------------------------------------------
Basic Network Settings Verified
```

store network interface auto-negotiation

Guardium ポートが接続されているスイッチでオートネゴシエーションが使用可能な場合は、オートネゴシエーションが使用され、このコマンドの再始動オプションのみでは何も影響しません。 このコマンドを使用して、ethN という名前のネットワーク・インターフェースのオートネゴシエーションを有効にしたり、無効にしたり、または再始動したりします。 show network interface inventory コマンドを使用して、すべてのポート名を表示します。

構文

store network interface auto-negotiation <ethN> <on | off | restart>

表示コマンド

show network interface auto-negotiation

store network interface duplex

このコマンドは、Guardium ポートが接続されているスイッチ上でオートネゴシエーションが使用できない場合にのみ使用します。 このコマンドは、ethn という名前のポートに二重モードを構成します。 show network interface inventory コマンドを使用して、すべてのポート名を表示します。

構文

store network interface duplex <ethn> <half | full>

表示コマンド

show network interface duplex <ethn>

store network interface high-availability

IP のチーミング (結合とも言う) を有効または無効にします。IP のチーミングは、Guardium システムの 1 次 IP アドレスのフェイルオーバー機能を提供します。

使用される 2 つのポート (ETH0 および 2 番目のインターフェース) は、同じネットワークに接続されている必要があります。 スイッチがポート構成を再学習することにより、わずかな遅延があります。 デフォルト設定は off です。

1 次 IP アドレスに使用されるポートは常に ETH0 です。高可用性オプションが有効になっている場合、Guardium システムは、必要に応じて、指定した 2 番目のインターフェースに自動的にフェイルオーバーし、実質的に 1 次 IP アドレスを 2 番目のインターフェースに移動します。

注: IP のチーミングと 2 次インターフェースは、同時に実行できません。
構文:

```
store network interface high-availability [on <NIC> | off ]
```

show network interface high-availability コマンドはありません。

store network interface inventory

Guardium 内部の表に保管されているネットワーク・インターフェースの MAC アドレスをリセットします。 このコマンドは、ネットワーク・カードを取り換えたり、移動したりした後にのみ使用する必要があります。

注: store network interface inventory コマンドは、Guardium アプライアンス内のオンボード NIC カードを検出し、これらのカードを eth0 および eth1 として割り当てます。このコマンドは NIC カードの位置を変える可能性があるため、Guardium サポートからそうするよう特に指示された場合にのみ実行してください。

構文

```
CLI> > store network interface inventory
WARNING: Running this function will reorder your NICS and may make the machine unreachable.
WARNING: It is suggested to run this from the console or equivalent.
Are you SURE you want to continue? (y/n)
```

show コマンドは、インストールされているすべてのネットワーク・インターフェースのポート名および MAC アドレスを表示します。

構文

show network interface inventory

例

CLI> show network interface inventory

Current network card configuration:

Device | Mac Address | Member of

eth0 | 00:50:56:3b:c3:73 |

eth1 | 00:50:56:8a:0d:fa |

eth2 | 00:50:56:8a:0d:fb |

eth3 | 00:50:56:8a:00:c1 |

注: 「Member of」は、結合が存在する場合に、どの NIC が結合ペアであるかを示します。

store network interface ip

Guardium アプライアンスの 1 次 IP アドレスを設定します。ネットワーク・インターフェースの IP アドレスを変更する際には、そのサブネット・マスクも変更しなければならない場合があります。 store network interface mask を参照してください。2 次 IP アドレスの作成および管理を行うには、store network interface secondary を参照してください。 結合/フェイルオーバーは、CLI コマンド store network interface high-availability を使用して管理します。

構文

store network interface ip <ip address>

表示コマンド

show network interface ip

store network interface ip6

Guardium アプライアンスの 1 次 IP V6 アドレスを設定します。ネットワーク・インターフェースの IP アドレスを変更する際には、そのサブネット・マスクも変更しなければならない場合があります。 store network interface mask を参照してください。2 次 IP アドレスの作成および管理を行うには、store network interface secondary を参照してください。 結合/フェイルオーバーは、CLI コマンド store network interface high-availability を使用して管理します。

構文

store network interface ip6 <ip address>

表示コマンド

show network interface ip6

store network interface map

ethn で識別されたイーサネット・ポートを MAC アドレス mac にマップします。

構文

store network interface map <ethn> <mac>

store network interface mask

1 次 IP アドレスのサブネット・マスクを設定します。ネットワーク・インターフェース・マスクを変更する際には、その IP アドレスも変更しなければならない場合があります。store network interface ip を参照してください。2 次 IP アドレスのサブネット・マスクは、「セットアップ」 > 「ツールとビュー」 > 「システム」からしか割り当てることができないので、注意してください。

構文

store network interface mask <ip mask>

store network interface mtu

この CLI コマンドは、MTU (最大転送単位) を設定するときに使用します。

```
CLI> store network interface mtu
Usage: store network interface mtu <interface> <mtu>]
       where <interface> is the interface name,
       that is one of ( eth0 )
       and <mtu> is number between 1000 and 9000.
```

表示コマンド

show network interface mtu

eth0 1500

show network interface port

このコマンドを使用して、アプライアンスの背面にある物理コネクターを見つけます。 *show network interface inventory* コマンドを使用してすべてのポート名を表示したら、以下のコマンドを使用して、「n」で指定される物理ポートのランプを 20 回明滅させます (「n」は、eth0、eth1、eth2、eth3 などのように、eth の後に続く数字です)。

構文

show network interface port <n>

例

CLI> show network interface port 1

ポート eth1 のオレンジ色のライトが 20 回明滅します。

store network interface remap

この CLI コマンドは、 NIC を再マップするときに使用します。

構文

store network interface remap

store network interface reset

この CLI コマンドは、 既存の OS ネットワーク構成を消去し、 保存されていた Guardium ネットワーク設定を再適用するときに使用します。

構文

```
CLI> store network interface reset
WARNING: This command will reset the network configuration to the stored Guardium network settings.
Are you SURE you want to continue? (y/n)
```

store network interface secondary

このコマンドは、1 次管理インターフェースとは異なる IP アドレス、ネットワーク・マスク、およびゲートウェイを持つ 2 次管理インターフェースとして Guardium システム上のポートを構成する場合に使用します。

注: IP のチーミングと 2 次インターフェースは、同時に実行できません。
構文:

```
store network interface secondary [on <NIC> <ip> <mask> <gateway> | off ]
```

表示コマンド

show network interface secondary

store network interface speed

このコマンドは、Guardium ポートが接続されているスイッチ上でオートネゴシエーションが使用できない場合にのみ使用します。 このコマンドは ethn という名前のポートの速度設定を構成します。 show network interface inventory コマンドを使用して、すべてのポート名を表示します。

構文

store network interface speed <ethn> <10 | 100 | 1000>

表示コマンド

show network interface speed <ethn>

show network arp-table

アドレス解決プロトコル (ARP) 表を表示します。これは操作可能なシステム値です。 このコマンドはサポート目的としてのみ提供されています。

構文

show network arp-table

例

CLI> sho net arp

IP address HW type Flags HW address Mask Device

192.168.3.1 0x1 0x2 00:0E:D7:98:07:7F * nic1

192.168.3.20 0x1 0x2 00:C0:9F:40:33:30 * nic1

ok

CLI>

show network macs

MAC アドレスのリストを表示します (show network interface inventory コマンドと同様)。

構文

show network macs

例

Network card configuration:

Device | Mac Address | Member of

eth0 | 00:50:56:3b:c3:73 |

eth1 | 00:50:56:8a:0d:fa |

eth2 | 00:50:56:8a:0d:fb |

eth3 | 00:50:56:8a:00:c1 |

注: 「Member of」は、結合が存在する場合に、どの NIC が結合ペアであるかを示します。

ok

store network interface ip6

使用法: store network interface ip <ip> (IP は有効な IP6 アドレス)。

store network resolver

Guardium アプライアンスが使用する第 1、第 2、および第 3 DNS サーバーの IP アドレスを設定します。 それぞれのリゾルバー・アドレスは固有でなければなりません。 DNS サーバーを削除するには、IP アドレスの代わりに null を入力します。

構文

store network resolver <1 | 2 | 3> <ip address | null>

表示コマンド

show network resolver <1 | 2 | 3>

store network routes defaultroute

デフォルト・ルーターの IP アドレスを、指定した値に設定します。

構文

store network routes defaultroute <ip address>

表示コマンド

show network routes defaultroute

store network routes static

ユーザーに対し、所有する IP アドレスが 1 アプライアンスにつき 1 つだけ (eth0 を通じて) であっても、静的ルーティング表を使用することにより、異なるルーターからの直接トラフィックを許可します。 静的ルーティング表に行を追加します。

構文

store network routes static

表示コマンド

現在の静的ルートと ID (デバイス、インデックス、アドレス、ネットマスク、ゲートウェイ) をリストします。

show network routes static

削除コマンド

delete network routes static

store system domain

システム・ドメイン・ネームを指定値に設定します。

構文

store system domain <値>

表示コマンド

show system domain

store system hostname

システムのホスト名を指定値に設定します。

構文

store system hostname <値>

表示コマンド

show system hostname

親トピック: CLI の概要

サポート CLI コマンド

以下の CLI コマンドは、技術サポートから指示された場合にのみ使用します。

技術サポートがマシンの状況を分析し、一般的な問題のトラブルシューティングと修正を行ううえで、これらのコマンドは役立ちます。定期的にこれらのコマンドを使って何らかの機能を実行する必要はありません。

support clean audit_results

監査結果を手動でパージします。このコマンドは本当に必要な場合にのみ使用してください (監査タスクで非常に多数のレコードが生成され、過大なディスク・スペースを占有する場合)。
このコマンドを実行する前に技術サポートに相談することを強くお勧めします。
このコマンドの実行時には警告メッセージが表示され、確認手順が必要になります。
このコマンドは、監査プロセスとタスクについての情報をリストします。
最も大きい結果セットから最も小さい結果セットの順番で、特定の行数が示されます。レポート結果の数は、入力値以上になります。
次に、レポートが表示された後、ユーザーは行番号を 1 つ選んで、その行番号に対応する監査プロセスの結果をパージできます。このように行番号を選択すると、選択されたプロセス名の監査データが削除されます。
構文
support clean audit_results <rows>
入力パラメーター
rows - 表示する行数 (整数)。 デフォルトは 10 です。
注: 監査タスクが非常に多いシステムでは、このコマンドの完了に時間がかかることがあります。

support clean log_files

この CLI コマンドは、指定されたファイルを削除します (その前にユーザーに削除の確認を求めます)。ファイルが見つからない場合、/var/log 内の 10MB より大きいファイルがリストされ、このリストからユーザーは大きなファイルを削除できます。警告メッセージが表示され、確認手順が含まれます。

構文

support clean log_file <filename> >> filename を追加

support clean DAM_data

データベース・アクティビティー・モニター・データを手動でパージします。このコマンドは本当に必要な場合にのみ使用してください。

このコマンドを実行する前に技術サポートに相談することを強くお勧めします。

このコマンドでは警告メッセージが表示され、確認手順が含まれます。

構文

support clean DAM_data <purge_type> <start_date> <end_date>

入力パラメーター

purge_type オプション: agg, exceptions, full_details, msgs, constructs, access, policy_violations, parser_errors, flat_log

start_date: YYYY-mm-dd

end_date: YYYY-mm-dd

support clean centera_files

Centera 内に保管された Guardium のアーカイブ/バックアップには、Guardium によって削除日マーカーが付加されています。ただし、削除処理を呼び出すための後続機能はありません。Centera は、独自のファイルを保守するための GUI を持っていないため、クライアント・アプリケーションからの API 呼び出しに依存します。

この CLI コマンド support clean centera_files を使用して、Centera 内のマークが付けられたファイルを削除してください。

support clean InnoDB-dumps

この CLI コマンドを使用して、InnoDB 表をパージします。

これはパスワードで保護されたコマンドです (技術サポートの場合のみ)。

support clean hosts

使用法: support clean hosts <IP address> <fully qualified domain name>

support clean servlets

jsp.java ファイルおよび *jsp*.class ファイルを削除し、GUI を再始動します。

この CLI コマンドは、生成された Java™ サーブレットとそのクラスを削除するために使用します。

support execute

このユーティリティーは、直接リモート・アクセスが使用できないか許可されていない場合に、Guardium 拡張サポートがリモート診断およびリモート・サポートを行えるように設計されています。

Support Execute は、直接リモート接続に代わるものではありませんが、Guardium サポートが直接アクセスを使用せずに少なくともある程度のルート・アクセスをセキュアな方法で実行できるようにします。

Guardium 拡張サポートから提供されるコマンドには、SQL ステートメント、O/S コマンド、シェル・スクリプト、SQL スクリプトがあります。これらのコマンドは、セキュア・キーとともにお客様に提供され、CLI を使用したコマンドの実行が可能になります。セキュア・キーは、お客様と Guardium サポートが取り組んでいるシステムに結び付けられます。このキーは、他のすべてのシステムでは無効です。このコマンドは、Guardium サポートによって許可された特定の回数のみ実行でき、同意日から 7 日間のみ有効です。

この機能は、デフォルトで無効になっています。通常モードとリカバリー・モードの両方で CLI コマンドを使用して有効にするには、以下を実行します。

support execute [enable | disable]

Guardium 拡張サポート・チームがセキュア・キーを生成できるようにするために、該当のシステムの MAC アドレスを eth0 に対して指定する必要があります。インターフェースおよび MAC アドレスの例を以下に示します。

お客様の使用 / CLI でログイン

support execute <CMD String> <PMR #> <KEY>

Guardium 拡張サポートから提供される主要な実行コマンド

support execute showlog [<Secure Key>|main|files]

使用ログを表示

#「<Secure Key>」: 単一エントリーの完全な詳細を表示

#「main」: 主要な実行ログを表示

「files」: ログのディレクトリー・リストを表示

support execute mac

セキュア・キーを生成するためにサポートが必要とする Eth0 MAC アドレス

support execute info

eth0 MAC アドレス、ルート・パス・キー、およびその他のシステム情報を表示

support execute version

#「Support Execute」の内部 2 進コード・バージョンを表示

support execute help

ユーティリティー情報のヘルプの詳細および目的

Guardium 拡張サポートから提供されるコマンドの例を以下に示します。

support execute "select * from GDM_ACCESS%5CG" 11111,111,111 6254130c0f0c3c504b33687c57f41363e4c00

support reset-password accessmgr

このコマンドは accessmgr アカウントのパスワードをリセットします。

構文

support reset-password accessmgr 10000000-99999999|random

パラメーター

新規パスワードを生成するために使われる 8 桁のキー番号。このキー番号を記録して技術サポートに提示すると、新しい accessmgr アカウント・パスワードを受け取ることができます。random を選択すると 8 桁の乱数が生成されます。

注: accessmgr アカウントの E メールが設定済みの場合、システムはそこに通知を送信しようとします。

support reset-password root

このコマンドは IBM® Guardium® アプライアンス上のルート・パスワードをリセットします。

構文

support reset-password root 10000000-99999999|random

パラメーター

新規パスワードを生成するために使われる 8 桁のキー番号。このキー番号を記録して技術サポートに提示してください。random を選択すると 8 桁の乱数が生成されます。

また、このコマンドを実行するとき、ユーザーはルート・パスワードを変更するために秘密のキーワードを提供する必要があります。ルート・パスワードを変更する必要がある場合には、技術サポートに連絡してください。

注: ビジネス・ルールに従って本当に必要な場合を除き、ルート・パスワードをリセットしないでください。

support schedule find_crashed_tables

この CLI コマンド、support schedule find_crashed_tables [ON/OFF] は、find_crashed_tables.sh スクリプトの日次の cron ジョブを有効または無効にするために使用します。
使用法: support schedule find_crash_tables on ALL|db
support schedule find_crash_tables off
このコマンドは、find_crashed_tables スクリプトの日次スケジュールを有効または無効にします。
注: 入力するデータベースに特に注意してください。ユーザーは、破損した表の 5 つの有効なデータベースすべてを処理するために「ALL」を入力するか、または「TURBINE」、「GDMS」、「CUSTOM」、「DATAMART」、「DIST_INT」の 5 つの有効なデータベースのいずれか 1 つを入力することができます。

support show db-processlist

このコマンドはすべての db プロセスを (実行時間でソートして) リストします。

構文

support show db-processlist all

support show db-processlist locked

support show db-processlist running

support show db-process full

パラメーター:

support show db-processlist [ ]

Where

running は、実行中のすべての sql ステートメントを表示するオプションです

all は、スリープ状態のプロセスも含めるオプションです

locked はロックされているすべてのプロセスと、最も古い 1 つのプロセスを表示します

full [任意指定] は、sql 照会を拡張形式で表示します

support show db-struct-check

このコマンドは、統合プロセスで見つかった構造の違いをすべて表示します。

構文

support show db-struct-check

support show db-top-tables

このコマンドは、サイズの大きい順にソートした上位 20 個のデータベース表をリストします。また、80% を超える空き領域を使用している表については、空き表スペースの使用量 (パーセント単位) でソートして表をリストします。表名によるフィルター処理が可能です。表のサイズはすべて MB で表示され、空き領域の使用

量はパーセントで表示されます。

構文

support show db-top-tables all

support show db-top-tables like

パラメーター

support show db-top-tables all

DB 全体の中から、サイズの大きい表をソートしてリストします

support show db-top-tables like

サイズの大きい表で、表名の任意の部分が基準に一致するものをリストします

support show db-status

このコマンドはデータベースの使用状況を表示します。

free (空き)、used (使用)、メガバイト、パーセントを選択できます。

構文

support show db-status free %

support show db-status used %

support show db-status free m

support show db-status used m

support show hardware-info

このコマンドは、ハードウェア情報を収集し、収集された情報を取り出せるようにディレクトリー内に入れるスクリプトを使用します。

この CLI コマンドの実行後に、以下のメッセージが表示されます。

Collected HW Info as /var/log/guard/Gather_hw_info-2012-06-25-17-43.tgz

その後 CLI コマンド fileserver を実行して、サーバーからこの .tar ファイルを取り出します。

support show iptables

このコマンドはシステム iptables コマンドの出力を表示します。

構文

support show iptables diff

support show iptables list

パラメーター

[diff | list] パラメーターは、通常の iptables 出力表示、または違い/差分だけの表示 (diff) のどちらにするかを制御します

[accept | full] パラメーターは、出力をフィルターに掛けて受け入れる行だけにするか、あるいはフィルターなしでリストします

support show large_files

このコマンドは、/var /tmp /root の各フォルダー内にある、特定の MB より大きく、特定の日数より古いファイルをすべてリストします。

使用法

support show large_files

このコマンドは、/var /tmp /root の各フォルダー内にある、特定の MB より大きく、特定の日数より古いファイルをすべてリストします

入力パラメーター:

* size - 10 より大きい整数 (MB)

* age - ゼロ以上の整数 (日数)

構文:

support show large_files <size> <age>

パラメーター

support show large_files

<size> は表示するファイルの最小サイズです (デフォルトは 100M)

<age> は最後に変更された日以降の日数です

support show netstat

このコマンドはシステム netstat コマンドの出力を表示します。grep パラメーターを使用して、内容に応じて出力をフィルターに掛けることができます。

構文

support show netstat all

support show netstat grep

パラメーター

support show netstat grep

検索対象となる英数字文字列

support show netstat all

support show port open

このコマンドは、telnet を使用して、開いている TCP ポートをローカルで、またはリモート・ホストで検出するのと似ています。

正常に接続できた場合、「Connection to 127.0.0.1 8443 port [tcp/*] succeeded!」のようなメッセージが表示されます。

接続できなかった場合、「connect to 127.0.0.1 port 1 (tcp) failed: Connection refused」のようなメッセージが表示されます。

構文: support show port open

IP port - IP は、127.0.0.1 のような、有効な IPv4 アドレスでなければなりません。

port は、1 から 65535 までの値の整数でなければなりません。

support show top

このコマンドは、システム top コマンドの出力を CPU、メモリー、または実行時間でソートして表示します。構成可能な反復回数 (デフォルトは 1)、表示する行数 (デフォルトは 10) を指定できます。

構文

support show top [cpu | memory | time]

パラメーター

support show top cpu

N は 1 から 10 までの範囲の反復回数、R は表示する行数 (少なくとも 10) です

support show top memory

N は 1 から 10 までの範囲の反復回数、R は表示する行数 (少なくとも 10) です

support show top time

N は 1 から 10 までの範囲の反復回数、R は表示する行数 (少なくとも 10) です

support check tables [DB name] [table name}

表に対して mysqlcheck –c コマンド (表にエラーがないか検査する) を呼び出します。

パラメーターを 1 つも指定しない場合、このコマンドは各検査のタイムアウトを 3 分として TURBINE データベース内のすべての表を検査します。各検査は並行して実行されるため、全体の時間は変動します。コマンドの進行状況がパーセント単位で表示されます。検査の実行時間が 3 分を超えると強制終了されます。タイムアウトによって検査が強制終了された表は、コマンドの完了後に画面上にすべてリストされます。コマンドの処理中に発生したエラーは、ログ・ファイル /var/log/guard/<dbname>_check_tables/errors.<date>.log に報告されます。<date> は現在の日付で、<dbname> はデータベースの名前です。

各表の検査処理で検出されたエラーは、/var/log/guard/<dbname>_check_tables/check_table_child.<tablename>.<date>.log ファイルに報告されます。<date> は現在の日付、<dbname> はデータベースの名前、<tablename> は検査された表の名前です。正常な表のファイルは作成されません。 1 番目のパラメーターとして dbname を指定した場合、このコマンドは指定されたデータベース内のすべての表を同じタイムアウト設定 (3 分) で検査します。パラメーターを 1 つも指定しない場合、TURBINE の表をすべて検査します。

パラメーターとして dbname と tablename を指定した場合、このコマンドは指定されたデータベース内にある指定された表を、タイムアウトなしで検査処理が完了するまで検査します。この方法により、3 分以内に検査が完了しなかった表を手動で検査することができます。パーセント記号 (%) を使用して、tablename パラメーター内でマスクを使用できます。

support shrink innodb-size

この CLI コマンドは、ibdata1 ファイルのサイズを小さくするために使用します。

ここでは、以下のステップが実行されます。

- すべての InnoDB 表をダンプします
- mysql を停止します
- ibdata1、ib_logfile0、ib_logfile1 の各ファイルを削除します
- mysql を開始します
- ダンプした表を復元します

これはパスワードで保護されたコマンドです (技術サポートの場合のみ)。

support show innodb-status

この CLI コマンドは、MySQL の問題をトラブルシューティングする場合に使用します。この CLI コマンドを使用して、実行時に MySQL 表で行われている処理を確認します。 この CLI コマンドを使用して、MySQL 表での検査時間が長い原因は、レコードのロックか表のロックかを判別します。

```
support show innodb-status
```

0 queries inside InnoDB, 0 queries in queue

0 read views open inside InnoDB

Main thread process no. 7959, id 139923805550336, state: sleeping Number of rows inserted 6894, updated 6934, deleted 93, read 24787 0.33 inserts/s, 0.00 updates/s, 0.00 deletes/s, 0.67 reads/s

---

END OF INNODB MONITOR OUTPUT

support analyze static-table

この CLI コマンドは、最大のグループに基づいて値の長さおよび値の出現回数によって静的表をソートすることで、静的表の内容を分析するために使用します。

support must_gather commands

任意の Guardium システムの状態に関する特定の情報を生成する、ユーザー CLI で実行できる単純な must_gather コマンドがいくつかあります。PMR (問題管理レコード) が記録されているときであればいつでも、この情報をアプライアンスからアップロードでき、Guardium 技術サポートに送信できます。

これらのコマンドを実行するには、適切な must_gather パッチがインストールされている必要があります。

正しいパッチをインストールした後は、いつでも以下の手順に従ってユーザー CLI で must_gather コマンドを実行できます。

1. 問題となっている Guardium システムに対する Putty セッション (または同様のセッション) を開きます。
2. ユーザー CLI でログインします。
3. 発生している問題のタイプに応じて、関連する must_gather コマンドを CLI プロンプトに貼り付けます。問題を診断するために、複数の must_gather コマンドが必要な場合があります。

   support must_gather system_db_info

   support must_gather purge_issues

   support must_gather audit_issues

   support must_gather agg_issues

   support must_gather cm_issues

   support must_gather alert_issues

   support must_gather patch_install_issues

   以下は、実行の完了までに数分かかる場合があります。

   support must_gather miss_dbuser_prog_issues

   support must_gather sniffer_issues

   以下のコマンドの場合、問題を再現している間にデバッガーを実行する時間の長さ (分) を入力するように求められます。

   support must_gather backup_issues

   support must_gather scheduler_issues

   出力は、must_gather ディレクトリーに、例えば must_gather/system_logs/.tgz のようなファイル名で書き込まれます。
4. 結果の出力を IBM サポートに送信してください。

ファイル・サーバーを使用すると、tgz ファイルをアップロードして、サポートに送信できます。

E メールで送信するか、ECUREP にアップロード (例えば、PMR 番号とアップロードするファイルを指定してファイル標準データ・アップロードを使用) します。

Guardium for z/OS トラフィック診断コマンド

support store zdiag on [N]

オプションの N は、診断を実行する時間 (分) です (10 から 600。デフォルトは 60)。

Guardium for z/OS トラフィック診断をオンにします。これには、TCPDUMP と SLON の収集が含まれます。この収集は、対応するファイルのサイズが 2 GB に達すると停止します。処理が完了したら、fileserver コマンドを使用して、結果ファイルの tcpdump.tar.gz と slon_all.tar.gz を検索することができます。/var パーティションには、15 GB 以上の空き領域が必要です。

support store zdiag off

Guardium for z/OS トラフィック診断をオフにします。CLI コマンド fileserver を使用して、結果のファイル tcpdump.tar.gz および slon_all.tar.gz をダウンロードできます。

support show zdiag

Guardium for z/OS トラフィック診断の状況を表示します。

SLON 収集コマンド

support store slon on [parameter]

スニファーによってデバッグ用に取得されたパケットを収集する SLON ユーティリティーをオンにします。結果ファイルの slon_packets.tar.gz、slon_messages.tar.gz、slon_all.tar.gz は、fileserver コマンドを使用して検索することができます。/var パーティションには、15 GB 以上の空き領域が必要です。

オプション・パラメーターは、次のとおりです。

packets: アナライザー・パケットをダンプします (デフォルト)。

snifsql: スニファーの SQL アクティビティーをログに記録し、アナライザー・パケットをダンプします。

secparams: セキュア・パラメーター情報をログに記録し、アナライザー・パケットをダンプします。

sgate: S-GATE デバッグ情報をログに記録し、アナライザー・パケットをダンプします。

messages: メッセージ・データ・ダンプをタップします。

support store slon off [parameter]

SLON ユーティリティーをオフにします。結果ファイルの slon_packets.tar.gz、slon_messages.tar.gz、slon_all.tar.gz は、fileserver コマンドを使用して検索することができます。

オプション・パラメーターは、次のとおりです。

packets: パケットのダンプを停止し、セキュア・パラメーター、S-GATE デバッグ情報、スニファーの SQL アクティビティーのロギングを停止します (デフォルト)。

messages: メッセージ・データ・ダンプのタップを停止します。

all: すべてのアクティビティーを停止します。

support show slon

SLON ユーティリティーの状況を表示します。

TCPDUMP 収集コマンド

support store snif_memory_max
使用法: support snif_memory_max <num> (num は数値 | 33 | 50 | 75 |)
このコマンドは、64 ビット・システムにのみ適用されます。
表示コマンド
support show snif_memory_max

support store tcpdump on <type> <period> <loglimit> [interface] [IP] [port] [protocol]

support store tcpdump on <type> <period> <loglimit> [interface] [IP] [port] [protocol]

TCPDUMP ユーティリティーをオンにします。指定された期間が経過すると、fileserver コマンドを使用して結果ファイル tcpdump.tar.gz を検索することができます。/var パーティションには、15 GB 以上の空き領域が必要です。

各部の意味は次のとおりです。

<type> - ダンプのタイプ。「headers」(収集されたヘッダーのみ) または「raw」(収集されたパケット全体) のいずれかです。

<period> - ダンプ期間 NUMBER[SUFFIX]。オプションの SUFFIX は、秒の場合は「s」、分の場合は「m」(デフォルト) です。

<loglimit> - ダンプ・ログ・ファイルのサイズ制限。値の範囲は 1 から 6 ギガバイトです。

オプションのフィルター引数は以下のとおりです。

[interface] - ネットワーク・インターフェース名 (デフォルトは eth0)

[IP] - IP アドレス

[port] - ポート

[protocol] - プロトコル。「tcp」、「udp」、「ip」、「ip6」、「arp」、「rarp」、「icmp」、「icmp6」のいずれかです。

例

support store tcpdump on headers 10m 1

このコマンドでは、10 分間のパケット・ヘッダーを 1GB のログ・ファイルのサイズ制限で保存する TCPDUMP が実行されます。

support show tcpdump

TCPDUMP ユーティリティーの状況を表示します。

support store tcpdump off

TCPDUMP ユーティリティーをオフにします。停止後、fileserver コマンドを使用して結果ファイル tcpdump.tar.gz を検索することができます。

support must_gather datamining_issues

異常値、クイック検索、およびデータマート機能に必要な診断情報を収集します。情報には、対応する内部表のダンプ、必要なログ、対応するプロセスの状態、および標準 must_gather 診断 (一般的なシステムおよび内部データベースの情報) が含まれます。

support must_gather network_issues [--host=<HOST>] (オプション・パラメーター <HOST> は、ホスト名または IP アドレス)

このコマンドは、アプライアンスからすべてのネットワーク情報を収集し、ping、traceroute、対応するポートのプローブ、その他の手段によって Guardium が対話するホストに対してポーリングします。オプション・パラメーターを指定した場合、指定されたホストに対してのみポーリングします (Guardium が当該ホストに対していずれかのアクティビティーを実行するように構成されている場合)。

store antlr3_max

この CLI コマンドは、パーサーとロガーの間のデータ・フローの制御を支援するために使用します。CLI コマンド store antlr3_max は、熟練したユーザーおよびカスタマー・サポートを対象とした拡張パラメーターであり、Oracle、Db2、MySql、および MSSql 用のスニファーのパーサー・コンポーネントとロガー・コンポーネントの間のデータ・フローの制御を支援します。

この値 (デフォルトは 20,000) により、ロガーがキューに入れることができる同時解析 SQL ステートメントの数が変更されます。

これが改善に役立つ可能性がある問題は、スニファーがメモリー不足で再始動する問題や、スニファーがメモリーを十分に使用していない問題です。

スニファーがメモリー不足で再始動することに気付いた場合は、コンテキストの上限を下げると、この問題を軽減するのに役立ちます。あるいは、スニファーが、使用可能なシステム・メモリーを十分に使用していない場合は、コンテキストの上限を上げると、メモリーをさらに使用できるようにすることができます。

store active_parser_engine

この CLI コマンドは、スニファーによって使用されるパーサー・エンジンを制御するために使用されます。この CLI コマンドは、ANTLR3 パーサー (Oracle、Db2、MS SQL、MySQL) によってサポートされるデータベース・タイプにのみ適用されます。
使用法: store active_parser_engine <num>
ここで、<num> は以下のいずれかです。
1: ANTLR2 によって ANTLR3 パーサー・エラーが再解析される (デフォルト)
2: ANTLR2 のみ
3: ANTLR3 のみ
表示コマンド
show active_parser_engine

親トピック: CLI の概要

システム CLI コマンド

これらの CLI コマンドは、システム設定の構成に使用します。

start ecosystem

このコマンドを使用して、エコシステム・プロセス・セット全体を再始動します。これは、パッチ適用、アップグレード、およびその他の操作の後に必要です。

構文

start ecosystem

stop ecosystem

このコマンドを使用して、エコシステム・プロセス・セット全体を一時的かつ正常に停止します。これは、パッチ適用、アップグレード、およびその他の操作に必要です。

構文

stop ecosystem

store system apc

このコマンドを使用すると、UPS 接続時の自動パワーダウン・オプションを構成できます。USB コネクターに UPS を接続する必要があることに注意してください (UPS のシリアル接続はサポートされていません)。

パワーダウンするまでの最小充電パーセント (0 から 100)、またはパワーダウン前にバッテリー電力で稼働する秒数を設定します。デフォルトはそれぞれ 25、ゼロです。

さらに、apc プロセスを開始/停止するコマンドもあります。デフォルトでは apc プロセスが無効になっています。

構文

store system apc [battery-level <percent> | timeout <seconds>]

store system apc start

store system apc stop

表示コマンド

show system apc [battery-level | timeout]

store system auditlog-passthrough

このコマンドは、auditd サービスからローカルの syslog へのシステム監査ログ・データのパススルーを有効または無効にするために使用します。システム監査ログは詳細であるため、auditlog-passthrough 機能はリモート・ロギングと組み合わせて使用するのが最適です。リモート・ロギングについて詳しくは、構成および制御 CLI コマンドを参照してください。

auditlog-passthrough 機能は、デフォルトでは無効になっています。

構文: store system auditlog-passthrough [on | off]

例:

```
> store sys aud on
Restarting auditd service to pick up the change.
Reloading configuration: [ OK ]
Auditd to syslog passthrough is enabled.
ok
```

表示コマンド: show system auditlog-passthrough

store system banner

store system banner [message | clear]

CLI ログイン時のバナー (無許可アクセスなどに関する警告、またはウェルカム・メッセージ) を作成するには、CLI コマンド store system banner [message | clear] を使用します。

構文

store system banner clear - この CLI コマンドを使用して、既存のバナー・メッセージを削除します。

store system banner message - この CLI コマンドを使用して、バナー・メッセージを作成します。 バナー・メッセージを入力して、CTRL-D を押してください。

表示コマンド

show system banner - この CLI コマンドを使用して、既存のバナー・メッセージを表示します。

store system clock datetime

システム・クロックの日時を、指定された値に設定します。**YYYY** は年、**mm** は月、**dd** は日、**hh** は時間 (24 時間形式)、**mm** は分、**ss** は秒です。秒の部分は必須ですが、常に 00 に設定されます。

構文

store system clock datetime <YYYY-mm-dd hh:mm:ss>

表示コマンド

show system clock <all |datetime |timezone>

例

store system clock datetime 2008-10-03 12:24:00

store system clock timezone

使用可能なタイム・ゾーン値をリストします (list オプション)。または、このシステムのタイム・ゾーンを、指定されたタイム・ゾーンに設定します。まず list オプションを使ってすべてのタイム・ゾーンを表示した後、リストから適切なタイム・ゾーンを選んで入力してください。

さらに、IBM® Guardium® では標準の監査証跡にローカル時間帯が記録されます。これにより、別のタイム・ゾーンで収集されたデータの中でデータが使われる (またはそのようなデータと共にデータが統合される) 場合に対処できます。

注: 夏時間が始まっても、タイム・ゾーン設定は自動的に更新されません。マシンを更新するには、ユーザーはタイム・ゾーンをリセットする必要があります。タイム・ゾーンのリセットとは、現在と異なる新しいタイム・ゾーンを設定した後、正しいタイム・ゾーンにリセットすることです。同じタイム・ゾーンにリセットするだけでは効果がなく、「No change for the timezone」というメッセージが出されます。

構文

store system clock timezone <list | timezone>

表示コマンド

show system clock <all | timezone | datetime>

例

まず **list** オプションを指定してコマンドを使用し、すべてのタイム・ゾーンを表示します。その後、適切なゾーンを使ってコマンドを再び入力します。

CLI> store system clock timezone list

Timezone: Description:

--- ---

Africa/Abidjan:

Africa/Accra:

Africa/Addis_Ababa:

...

...output deleted

...

CLI> store system clock timezone America/New_York

store system conntrack

Linux カーネルの接続トラッキング・サブシステムの現在の状況を設定します。状況は ON|OFF のいずれかです。

構文

store system conntrack ON|OFF

表示コマンド

show system conntrack

store system cpu profile

CPU スケーリングをサポートするハードウェアにおいて、CLI コマンドにより、CPU スケーリングの構成を許可します。

この CLI コマンドを使用して、必要に応じた適切な CPU スケーリング・ポリシーを設定します。

- 控えめ = 低い電力使用量、控えめなスケーリング
- 平衡 = 中程度の電力使用量、迅速な拡大
- パフォーマンス = 最大クロック速度での CPU の実行

Guardium ソフトウェアでは、インストール時にスケーリング・ポリシーは「パフォーマンス」に設定されます。

構文

store system cpu profile [min|perf|max]

表示コマンド

show system cpu profile

store system custom_db_size

この CLI コマンドは、カスタム・データベース表の最大サイズ (MB 単位) を設定するときに使用します。デフォルト値は 4000 MB です。

構文

```
CLI> store system custom_db_max_size
USAGE: store system custom_db_max_size <N>
       where N is number larger than 4000.
```

表示コマンド

show system custom_db_size

store system domain

システム・ドメイン・ネームを指定値に設定します。

構文

store system domain <value>

表示コマンド

show system domain

store system ecosystem [on | off]

Guardium Web UI のアプリケーション・ライフサイクル・ページを含む、エコシステム・プロセスとコンポーネントを有効または無効にします (「セットアップ」>「ツールとビュー」>「アプリケーション・ライフサイクル」)。デフォルトは off です。

構文

store system ecosystem [on | off]

store system hostname

システムのホスト名を指定値に設定します。

構文

store system hostname <value>

表示コマンド

show system hostname

store system issue

store system issue [message | clear]

CLI コマンド store system issue message は Ctrl-d までコンソールから入力を受け取り、入力の中の $、¥、￥ の後の 1 文字、および ` 文字をすべて除去した後、それを /etc/motd に書き込みます。これは、このシステムをカスタマーのセキュリティー・ポリシーに準拠させるメッセージを入力する方法の 1 つです。

CLI コマンド store system issue clear は /etc/motd をデフォルト・バージョンに復元します。

バージョンは、/etc/guardium-release から取得されます。 例えば、SG70 -> 7.0、SG80 -> 8.0 です。/etc/guard-release の中に SG が見つからない場合、デフォルト・バージョンは空文字列です。

store system netfilter-buffer-size

netfilter バッファーのサイズを設定します。

構文

store system netfilter-buffer-size

表示コマンド

S-TAP® netfilter バッファー・サイズを表示します。デフォルトは 65536 です。

show system netfilter-buffer-size

show system ntp diagnostics

この CLI コマンドは、ntpq -p および ntptime を実行し、 その出力を直接画面に送信するときに使用します。Guardium システムは、UDP を介してローカル・ホストから NTPD を 照会します。

構文

show system ntp diagnostics

例

```
CLI> show system ntp diagnostics
Output from ntpq -p :
localhost.localdomain:
------------------------------------------------------------------------
Output from ntptime :
(Note that if you have just started the ntp server, it may report an 'ERROR' until it has synchronized.)
------------------------------------------------------------------------
ntp_gettime() returns code 5 (ERROR)
  time d3443c21.47a46000  Thu, Apr 26 2012 17:26:57.279, (.279852),
  maximum error 16384000 us, estimated error 16384000 us
ntp_adjtime() returns code 5 (ERROR)
  modes 0x0 (),
  offset 0.000 us, frequency 0.000 ppm, interval 1 s,
  maximum error 16384000 us, estimated error 16384000 us,
  status 0x40 (UNSYNC),
  time constant 2, precision 1.000 us, tolerance 512 ppm,
```

store system ntp [all | server | state]

store system ntp server

最大で 3 つの NTP (Network Time Protocol) サーバーから成る**ホスト名**を設定します。なお、NTP サーバーの使用を有効にするには、**store system ntp state on** コマンドを使用する必要があります。1 つの NTP サーバーを定義するには、そのホスト名または IP アドレスを入力します。複数の NTP サーバーを定義するには、引数なしでコマンドを入力し、NTP サーバーのホスト名を指定するプロンプトを表示させます。

注: Guardium では、NTP サーバーのホスト名の基底の IP アドレスが動的アドレスである場合は、NTP サーバーにホスト名を使用できません。ホスト名は静的 IP アドレスに解決されなければなりません。その IP アドレスが変更された場合は、新しい NTP サーバー IP と同期するためにネットワークを再始動する必要があります。

構文

store system ntp server

使用法: store system ntp server

サーバーごとに IP またはホスト名のいずれかを入力します。

ストアする NTP サーバーを最大 3 つ入力します。

表示コマンド

show system ntp <all |server>

削除コマンド

delete ntp-server

store system ntp state

NTP (Network Time Protocol) サーバーの使用を有効または無効にします。

構文

store system ntp state <on | off>

表示コマンド

show system ntp <all |state>

store system patch install

1 つのパッチ、または複数のパッチをバックグラウンド・プロセスとしてインストールします。**ftp** および **scp** オプションは、圧縮されたパッチ・ファイルをネットワーク上のロケーションから IBM Guardium アプライアンスにコピーします。圧縮された 1 つのパッチ・ファイルに複数のパッチが含まれることがありますが、一度にインストールできるパッチは 1 つだけであることに注意してください。複数のパッチをインストールするには、インストールする必要があるすべてのパッチをコンマで区切って選択します。CLI は内部的に、リストの各パッチに関する要求を (ユーザーによって指定された順序で) 実行依頼しますが、その際、最初のパッチはユーザーによって指定された要求時間に行われ、後続の各パッチは前のパッチの 3 分後になります。さらに CLI は、指定された (1 つまたは複数の) パッチが既に要求されているかどうか確認し、重複要求を許可しません。

最後のオプション (sys) は、以前にこのコマンドを使って IBM Guardium アプライアンスに既にコピーされた圧縮ファイルに含まれる、2 番目 (またはそれ以降) のいずれかのパッチをインストールするときに使用します。

適用済みのパッチの全リストを表示するには、「管理」>「レポート」>「インストール管理」>「インストール済みのパッチ」、「管理」>「メンテナンス」>「一般」>「インストール済みのパッチ」、または「レポート」>「Guardium 運用レポート」>「インストール済みのパッチ」のいずれかで「インストール済みのパッチ」レポートを参照してください。

CLI コマンド **store system patch install** では、ユーザーはリストから複数のパッチを選択できます。

構文

store system patch install <type> <date> <time>

<type> はインストール・タイプ cd | ftp | scp | sys

<date> および <time> はパッチ・インストールの要求時間で、日付の形式は YYYY-mm-dd、時刻の形式は hh:mm:ss

日時が入力されない場合、または NOW が入力された場合、インストール要求時間は「今すぐ」です。

パラメーター

どのオプションを選択した場合も、適用対象のパッチを選択するようプロンプトが出されます。

Please choose one patch to apply (1-n,q to quit):

cd - パッチを CD からインストールするには、このコマンドを実行する前に IBM Guardium CD-ROM ドライブに CD を挿入してください。CD に含まれるパッチのリストが表示されます。

tp または **scp -** ネットワーク上の任意の場所にある圧縮パッチ・ファイルからパッチをインストールするには、**ftp** オプションまたは **scp** オプションを使用し、示されるプロンプトに応答します。 パッチのファイル名を含む絶対パス名を指定してください。

Host to import patch from:

User on hostname:

Full path to the patch, including name:

パスワード:

CLI コマンド store system patch install scp では、ユーザーはパッチ・ファイル名にワイルドカード * を使用できます。

圧縮パッチ・ファイルが IBM Guardium アプライアンスにコピーされ、ファイルに含まれるパッチのリストが表示されます。

sys - 以前の store system patch 実行により IBM Guardium アプライアンスに既にコピーされたパッチ・ファイルに含まれる、2 番目 (またはそれ以降) のいずれかのパッチを適用するには、このオプションを使用します。

store system patch install コマンドは、インストール後にパッチ・ファイルを IBM Guardium アプライアンスから削除しません。既存のパッチの上に同じパッチを再インストールすることが可能で、パッチ・ファイルを手元に残しておくとさまざまな問題の分析に役立つ可能性があるため、パッチ・ファイルを必ずしも削除する必要はあり

ません。ただし、ユーザーは手操作で、または CLI コマンド diag を使ってパッチ・ファイルを削除できます (なお CLI コマンド diag は特定のユーザーやロールに制限されています)。

パッチ・インストール要求を削除するには、CLI コマンド delete scheduled-patch を使用します。

store system public key reset

CLI コマンド show system public key tomcat または show system public key cli を使用して SSH 公開鍵を生成した後、CLI コマンド store system public key reset を使用すると、SSH 鍵が削除されます。SSH 鍵が生成されなかった場合、この CLI コマンドは何も行いません。このコマンドは、削除前に確認を要求します。

構文

store system public key reset

store system remote-root-login

SSH (ルート・アクセス) を有効/無効にします。SSH つまりセキュア・シェルは、ネットワークで結ばれた 2 つのデバイス間でセキュア・チャネルを使ってデータ交換できるようにするネットワーク・プロトコルです。

構文

store system remote-root-login ON|OFF

表示コマンド

show system remote-root-login

store system serialtty

一部の環境では、シリアル TTY を使用できないため、正常に開始できません。これは、システム・ログに出力され、SIEM に転送される可能性があります。これは、接続を許可するためにデフォルトでは有効になっていますが、後で、システムでシリアル・コンソールを使用できないと判別された場合、無効にできます。

構文

store system serialtty <on, off>

表示コマンド

show system serialtty

システムでシリアル TTY が有効になっているかどうかを報告します。

次のいずれかを報告します。

このシステムではシリアル TTY コンソールが有効になっています。

このシステムではシリアル TTY コンソールが無効になっています。

store system scheduler

スケジューリングは、IBM Guardium アプリケーション内のタイミング・メカニズムによって管理されます。タイミング機能が中断した場合、この CLI コマンドで指定された再始動インターバル後に再始動します。

store system scheduler restart_interval [5 から 1440、または -1] を使用すると、5 分後から 1440 分後までの範囲でタイミング機能を再始動することができます。デフォルトは -1 で、タイミング再始動メカニズムがインストールされていないことを示します。

現在実行中のすべてのジョブが終了した後でスケジューラーを再始動するには、store system scheduler wait_for_shutdown [ON | OFF] を使用します。パラメーターは ON または OFF です。

構文

store system scheduler restart_interval [5 から 1440、または -1]

store system scheduler wait_for_shutdown [ON | OFF]

表示コマンド

show system scheduler

store system shared secret

システムの共有パスワード値を指定値に設定します。この鍵は、中央マネージャーおよび管理されるすべてのアプライアンスの間で同じでなければなりません。または、アグリゲーターとデータ統合対象のすべてのアプライアンスの間で同じでなければなりません。あるアプライアンスを中央マネージャーによる管理対象として登録した後、そのユニットの共有パスワードは使用されなくなります。(この値を変更してユニットを一元管理から「登録抹消」することはできません。)

aggregator OS ユーザーの動的パスワード

aggregator パスワードは、共有パスワードに連結される <現行パスワード> です (つまり、パスワード = <現行パスワード><共有パスワード>)

ユーザーは、コレクターの共有パスワードとアグリゲーターの共有パスワードがまったく同じであることを確認する必要があります。そうでない場合、コレクターからアグリゲーターへの SCP 転送が失敗します。(これは管理対象ユニットとアグリゲーター、コレクターとアグリゲーター、およびエクスポート設定画面での要件です。) 共有パスワードは、CLI、および管理コンソール・タブの「システム」ペインのどちらからでも設定可能です。

構文

store system shared secret <key>

store system signature [on | off]

off の場合、署名を持たないアプリケーションのデプロイメントが可能になります。Guardium システムでアプリケーションをテストするときは、store system signature を off にしてください。そうしないと、アプリケーションがブロックされます。実稼働環境では、App Exchange の認定アプリを使用するため、このパラメーターは on にする必要があります。

構文

store system signature [on | off]

store system snif-alerts-facility

このパラメーターを使用すると、ユーザーはスニフ生成アラートの機能を構成できます。 前もってスニフによって直接生成されたアラートはユーザー機能を使用しますが、間接アラートは (guard_sender ユーティリティーを介して) デーモン機能を使用します。

構文

store system snif-alerts-facility <facility>

使用法: store snif-alerts-facility <facility>

facility は、daemon ftp local0 local1 local2 local3 local4 local5 local6 local7 lpr user のいずれかです。

デフォルトの facility は daemon です。

表示コマンド

show system snif-alerts-facility

store system snif-buffers-reclaim

この CLI コマンドは、IBM Guardium 技術サービスから指示された場合にのみ使用してください。

新しい構成は、CLI コマンド restart inspection-core が実行された後で有効になります。

構文

store system snif-buffers-reclaim [ON | OFF]

表示コマンド

show system snif-buffers-reclaim

store system snif-thread-number

この CLI コマンドは、実行するスレッド数を指定する場合に使用します。

新しい構成は、CLI コマンド restart inspection-core が実行された後で有効になります。

構文

store system snif-thread-number [new | default]

表示コマンド

show system snif-thread-number

snif は、32 ビット・システムでは 6 スレッドで実行されています。

store system snmp contact

IBM Guardium アプライアンスの snmp contact (syscontact) の E メール・アドレスを保管します。デフォルトでは info@guardium.com です。

構文

store system snmp contact <email-address>

表示コマンド

show system snmp contact

store system snmp location

IBM Guardium アプライアンスの snmp システム・ロケーション (syslocation) を保管します。デフォルトでは Unknown です。

構文

store system snmp location <string>

表示コマンド

show system snmp location

store system snmp query community

IBM Guardium アプライアンスの snmp システム照会コミュニティーを保管します。デフォルトでは guardiumsnmp です。

構文

store system snmp query community <string>

表示コマンド

show system snmp query community

store system sshd-max-connection

このコマンドを使用すると、SSHD の最大同時接続数を構成できます。値の範囲は 100 から 500 です。デフォルト値は 250 です。

注: このコマンドを実行すると、既存の接続が打ち切られ、Guardium アプライアンス上で SSH デーモンが再始動されます。

構文

store system sshd-max-connection <value>

表示コマンド

show sys sshd-max-connection

親トピック: CLI の概要

ユーザー・アカウント、パスワード、および認証 CLI コマンド

これらの CLI コマンドを使用して、ユーザー・アカウント、パスワードおよび認証を構成します。

set guiuser 認証

デフォルト CLI アカウント (guardcli1、... guardcli5) のうちの 1 つを使用して CLI 経由でログオンする場合、CLI コマンド set guiuser を実行した後でなければ、GuardAPI コマンドは作動しません。 GUI において制限されたロールしかないユーザーが、GuardAPI コマンドに無許可アクセスを行わないようにするために、この認証が必要です。

guardcli1 ... guardcli5 アカウントを使用するには、ローカル・ユーザーおよびパスワードの設定が必要です。CLI コマンド set guisuer を使用して guardcli1 ... guardcli5 アカウントをリセットしてから、下記の構文に示すように、ローカル・ユーザーおよびパスワードを追加します。

注: LDAP 認証を使用する場合、ローカル・ユーザーとパスワードは、LDAP ユーザーと LDAP パスワードになります。

CLI コマンドの中には、guiuser のロールに依存するものがあります。例えば、grdapi create_user、grdapi set_user_roles、および grdapi update_user にアクセスするには、guiuser のロール (accessmgr ビューから新規ユーザーを作成するときにマークが付けられます) は accessmgr である必要があります。

構文

set guiuser <gui_user or LDAP user> password <password or LDAP password>

例

$ ssh guardcli1@a1.corp.com

IBM Security Guardium, Command Line Interface (CLI)

guardcli1@a1.corp.com's password:

Last login: Thu Nov 4 14:56:34 2012 from 123.a1.corp.com

==========

IBM Security Guardium

Unauthorized access is prohibited

==========

a1.corp.com> set guiuser johny_smith password 3wel9s887s

ok

a1.corp.com>

create_user

例

>grdapi create_user firstName=john lastName=smith

password=pASSW0rd confirmPassword=pASSW0rd email=jsmith@us.ibm.com

userName=john disabled=0

ID=20000

```
>grdapi set_user_roles userName="john"
roles="dba,diag,cas,user"
```

ID=20000

ロール (dba) が追加されました。

ロール (diag) の追加に失敗しました。 診断は、cli または admin のいずれかのロールを持つ必要があります。

ロール (cas) が追加されました。

ロール (user) が追加されました。

```
> grdapi set_user_roles userName="john"
roles="dba,diag,cas,user,cli"
```

ID=20000

ロール (dba) が追加されました。

ロール (diag) が追加されました。

ロール (cas) が追加されました。

ロール (user) が追加されました。

ロール (cli) が追加されました。

```
> grdapi update_user userName="john"
email="john.smith@gmail.com"
```

ID=20000

> grdapi list_users

ID=0

####### User 3 #######

ユーザー名: accessmgr

名: accessmgr

姓: accessmgr

E メール:

無効: false

####### User 1 #######

ユーザー名: admin

名: admin

姓: admin

E メール:

無効: false

####### User 33 #######

ユーザー名: anon

名: anon

姓: anon

E メール:

無効: false

####### User 20000 #######

ユーザー名: john

名: john

姓: smith

E メール: john.smith@gmail.com

無効: false

####### User 2 #######

ユーザー名: bill

名: bill

姓: green

E メール:

無効: true

set_user_roles

set_user_roles

set_user_roles を実行するたびに、ユーザーのロールをリセットします。 ロールには何も追加しないでください。リセットしてください。

GrdAPI を使用してユーザーを作成すると、user ロールを持つユーザーが作成されます。ロールを設定する際には、そのロールのすべてを指定する必要があります。これは、既存のロールの削除と新規ロールの追加を有効にするために行われます。

GUI でも、チェック・マークを付けたり外したりできるロールがすべて表示されます。ロールを保存すると、チェック・マークの付いたすべてのロールが保存されます。

GrdAPI で、ユーザー kevin にロール INV のみを付与します。ユーザーには、user、cli、admin、または accessmgr のいずれかのロールが必要です。

この GrdAPI の正しい呼び出し方法は次のとおりです。

```
grdapi set_user_roles userName="kevin"  roles="user,inv"
```

例

> set guiuser accessmgr password ASDFasdf

ok

> grdapi create_user firstName=kevin

lastName=smith password=pASSW0rd confirmPassword=pASSW0rd

email=ksmith@company.com userName=kevin disabled=0

ID=20000

ok

```
> grdapi set_user_roles userName="kevin" roles="inv"
```

set_user_roles:

ERR=3700

ユーザーには、user、cli、admin、accessmgr のいずれかのロールが必要です。

コマンドを実行中にエラーが発生しました

ok

```
> grdapi set_user_roles userName="kevin"
roles="user,inv"
```

ID=20000

ロール (user) が追加されました。

ロール (inv) の追加に失敗しました。 inv ロールを割り当てる前に、ユーザーの姓を次の 3 つの調査データベースのいずれかの名前に設定する必要があります。

INV_1、INV_2、または INV_3 (大/小文字の区別あり)

ok

```
> grdapi set_user_roles userName="kevin"
roles="dba,diag,cas,user"
```

ID=20000

ロール (dba) が追加されました。

ロール (diag) の追加に失敗しました。 診断は、cli または admin のいずれかのロールを持つ必要があります。

ロール (cas) が追加されました。

ロール (user) が追加されました。

ok

>

show guiuser

これにより、GUI のユーザー (ロール別) が表示されます。

表示コマンド

show guiuser

パスワード制御コマンド

以下のコマンドを使用して、次のようにユーザー・パスワードを制御します。

- store password disable - 非アクティブなアカウントが無効になるまでの日数を設定します。
- store password expiration - パスワードが期限切れになるまでの日数を設定します。
- store password validation - 強化されたパスワードの検証ルールを有効または無効にします。

アカウント・ロックアウト・コマンド

アカウント・ロックアウト・コマンドを使用して、ログイン試行が 1 回以上失敗した後に Guardium® ユーザー・アカウントを無効にします。 これらのコマンドは、以下の目的で使用します。

- 機能を有効または無効にする。 store account lockout を参照してください。
- 1 つのアカウントについて、所定の時間間隔の間に許容されるログイン失敗の最大回数を設定する。 store account strike count および store account strike interval を参照してください。
- 1 つのアカウントについて、Guardium アプライアンスの存続期間中に許容される失敗の最大回数を設定する。 store account strike max を参照してください。
- admin ユーザー・アカウントがロック状態になった場合にアンロックする。 unlock admin コマンドの説明を参照してください。

Guardium ユーザー・アカウントが無効化された後、accessmgr ロールを持つユーザーか admin ユーザーに限り、このアカウントを Guardium ポータルから有効にすることができます。

例

アカウント・ロックアウトを有効にし、10 分以内に 5 回のログインが失敗したらアカウントをロックし、許容される失敗の最大数を 999 に設定します。

store account lockout on

store account strike count 5

store account strike interval 10

store account strike max 999

注:

admin ユーザー・アカウントがロックされている場合、unlock admin コマンドを使用してアンロックします。

アカウント・ロックアウトが有効になっている場合、strike count または strike max をゼロに設定しても、そのタイプのチェックは無効になりません。 それどころか、1 回でも失敗するとそのユーザー・アカウントが無効になることを意味します。

store account lockout

指定回数ログインが失敗したらユーザー・アカウントを無効にする自動アカウント・ロックアウト機能を有効 (on) または無効 (off) にします。

構文

store account lockout <on | off>

表示コマンド

show account lockout

store account strike count

構成されたストライク間隔において、アカウントが無効になるログイン試行失敗回数 (n) を設定します。

構文

store account strike count <n>

表示コマンド

show account strike count

store account strike interval

ここに設定した秒数 (n) の間に、構成されたログイン試行失敗回数に達すると、アカウントが無効になります。

構文

store account strike interval <n>

表示コマンド

show account strike interval

store account strike max

サーバーの存続期間中に、アカウントが無効になるまでに許容されるログイン試行失敗の最大回数 (n) を設定します。

構文

store account strike max <n>

表示コマンド

show account strike max

store disable_sha1_passwords

デフォルトでは、Guardium GUI のユーザー・パスワードは、強力なパスワード・ハッシュ・アルゴリズムを使用してハッシュされます。store disable_sha1_passwords CLI コマンドを使用すると、脆弱な方法でハッシュされている既存のパスワードを、管理者が Guardium アプライアンスから削除できます。

注: アップグレード・シナリオでは、このコマンドを実行すると、アップグレード後にログインしたユーザーの脆弱なパスワードだけが削除されます。

構文

store disable_sha1_passwords [true] [false]

store disable_sha1_passwords true コマンドは、中央マネージャーおよびすべてのバックアップ中央マネージャー (該当する場合) で実行する必要があります。

例:

```
>store disable_sha1_passwords true
> User passwords will now be hashed with a strong password hashing algorithm.

>store disable_sha1_passwords false
> User passwords will now be hashed with a weak password hashing algorithm.
Weak password hashing algorithms may violate your company compliance requirements.
```

表示コマンド

構文:

show disable_sha1_passwords

show コマンドは、パスワード・ハッシュの現在の設定を返します。

例:

```
>show disable_sha1_passwords
>SHA1 passwords are allowed.
```

注: アップグレード・シナリオでは、show コマンドを実行すると、アップグレード後にログインしていないユーザーが返されます。

store password disable

days で設定した日数の間アクティビティーがなければ、ユーザー・アカウントが無効になります。 0 (ゼロ) に設定すると、アクティビティーがなくてもアカウントは無効になりません。 インストール時のデフォルト値はゼロです。 この設定値の変更後には、GUI を再始動する必要があります (restart gui を参照してください)。

構文

store password disable <days>

表示コマンド

show password disable

store password expiration

ユーザー・パスワードの有効期限の存続期間 (日数) を設定します。 -1 に設定すると、パスワードの有効期限が切れることはありません。GUI ユーザーの場合は、0 (ゼロ) に設定すると、パスワードの有効期限が切れることはありません。0 以外の値を設定した場合、アカウント・ユーザーは、現行パスワードが有効期限切れになった後の初回ログイン時にパスワードを再設定する必要があります。 デフォルト値は 90 です。 この設定値の変更後には、GUI を再始動する必要があります。

構文

store password expiration cli <days>

store password expiration gui <days>

表示コマンド

show password expiration

store password validation

パスワードの検証をオンまたはオフに切り替えます。 デフォルト値は on です。 このコマンドを実行すると、GUI が再始動されてこの設定が適用されます。

パスワード検証が有効になっている場合、パスワードは 8 文字以上の長さでなければなりません。さらに、英大文字 (A-Z)、英小文字 (a-z)、数字 (0-9)、および表に示す特殊文字を、それぞれ 1 つ以上含んでいなければなりません。無効になっている (非推奨) 場合は、任意の長さおよび文字の組み合わせが許可されます。

構文

store password validation <on | off>

表示コマンド

show password validation

表 1. Guardium パスワードに使用できる特殊文字

文字	記述
@	アットマーク (単価記号)
#	ナンバー記号
$	ドル記号
%	パーセント記号
^	曲折アクセント記号 (カラット)
&	アンパーサンド
.	終止符 (ピリオド)
;	セミコロン
!	感嘆符
-	ハイフン (マイナス記号)
+	プラス記号
=	等号
_	下線 (アンダースコアー)

store strong_password_enable

このコマンドは、強力なパスワード検査を有効にまたは無効にするために使用します。コマンドが受け入れる構文は次のとおりです。

```
store strong_password_enable [on|off]
```

現在の設定を表示するには、コマンド show strong_password_enable を使用します。

store user password

このコマンドは、cli ユーザー・パスワードをリセットするために使用します。 サポート処理を簡略化するため、初期状態で Guardium によって割り当てられた cli ユーザー・パスワードを覚えておくことを推奨します。 一度設定した cli ユーザー・パスワードを検索する方法はありません。 このパスワードを紛失した場合は、Guardium サポートに連絡し、パスワードのリセットを依頼してください。

構文

store user password

現行のパスワードと、それに続いて新規パスワード (2 回) の入力を求めるプロンプトが出されます。 キーボードで入力したパスワードの値は、画面上には表示されません。

cli ユーザー・パスワード要件は、ユーザー・パスワードの要件とは異なります。 cli ユーザー・パスワードは、 8 文字以上の長さでなければなりません。さらに、次のタイプの文字をそれぞれ 1 つ以上含んでいなければなりません。

- 英小文字
- 英大文字
- Guardium パスワード・テーブルの特殊文字

この CLI コマンドを実行すると、パスワード有効期限ファイル内の変更日時レコードも更新されます。

unlock accessmgr

このコマンドは、無効になっている Guardium accessmgr ユーザー・アカウントを有効にするために使用します。 このコマンドで、accessmgr ユーザー・アカウント・パスワードがリセットされることはありません。

注: この CLI コマンドの実行を許可されるのは、admin ロールを持つユーザーだけです。

構文

unlock accessmgr

restart gui

unlock admin

このコマンドは、無効になっている Guardium admin ユーザー・アカウント を有効にするために使用します。 このコマンドで、admin ユーザー・アカウント・パスワードがリセットされることはありません。

注: この CLI コマンドの実行を許可されるのは、admin ロールを持つユーザーだけです。

構文

unlock admin

restart gui

認証コマンド

以下のコマンドは、使用される認証のタイプを表示または制御します。

store auth

このコマンドは、Guardium アプライアンス、SQL_GUARD へのログオンに使用する認証のタイプをリセットする (つまり、デフォルトのローカル Guardium 認証) ために使用します。

オプションの認証方式 (LDAP や Radius など) の構成および有効化は、管理者ポータルから行うことはできますが、CLI から行うことはできません。詳しくは、『認証の構成』を参照してください。

構文

store auth SQL_GUARD

表示コマンド

show auth

親トピック: CLI の概要

GuardAPI

GuardAPI を使用すると、コマンド行から Guardium® 機能にアクセスできます。

- GuardAPI の使用
  GuardAPI の使用方法について説明します。
- Guardium REST API
  Guardium REST API を使用すると、ご使用のアプリケーション、または REST API を使用できる任意の場所に、多くの Guardium API コマンドを組み込むことができます。
- Guardium API リファレンス (アルファベット順)
  すべての Guardium guardapi コマンドをアルファベット順に示します。guardapi コマンドをクリックすると、そのコマンドの詳細を確認できます。guardapi コマンドに REST API インターフェースがある場合は、REST 呼び出しも示されます。
- GuardAPI アーカイブおよびリストア関数
- GuardAPI アセスメント関数
  以下の CLI コマンドは、アセスメント関数を追加、削除、および更新するために使用します。
- GuardAPI オートディスカバリー関数
  以下の CLI コマンドは、オートディスカバリー関数を作成、変更、リスト、および実行するために使用します。
- GuardAPI Big Data Intelligence 関数
  中央マネージャーで以下のコマンドを実行して、ビッグデータのデータ・ソースへのデータマートの抽出を管理したり、モニターやレポートなどのためにビッグデータを Guardium にプルしたりします。
- GuardAPI カタログ・エントリー関数
  これらの GuardAPI コマンドは、カタログ・エントリー関数の作成、リスト、削除、および更新に使用します。
- GuardAPI 分類関数
  次の GuardAPI コマンドを使用して、分類ポリシー構成、テスト自動化、および前提条件データの準備のスクリプト記述を行います。
- GuardAPI クラウド・データ・ソース関数
  これらのコマンドは、クラウド・データ・ソースを定義、更新、および削除するために使用します。
- GuardAPI データマート関数
  これらのコマンドは、データマートを管理するために使用します。
- GuardAPI データベース・ユーザー関数
  これらの GuardAPI コマンドは、データベース・ユーザー・マッピングの保守、非資格情報スキャン、およびデバッグ・レベルの設定に使用します。
- GuardAPI データ・ソース関数
  これらの GuardAPI コマンドは、データ・ソース関数の作成、リスト、削除、および更新に使用します。
- GuardAPI データ・ソース・リファレンス関数
  これらの GuardAPI コマンドは、データ・ソース・リファレンス関数の作成、リスト、および削除に使用します。
- GuardAPI データ・ユーザー・セキュリティー関数
  以下の GuardAPI コマンドは、データ・ユーザー・セキュリティー関数を作成、リスト、削除、および更新するために使用します。
- GuardAPI エンタープライズ・ロード・バランシング関数
  以下の GuardAPI コマンドを使用して、ロード・バランシング・パラメーターの表示と設定、現在のロード・マップの表示、および S-TAP と管理対象ユニット・グループの関連付けの管理を行います。
- GuardAPI 資格最適化機能
  これらの GuardAPI コマンドは、資格最適化データ・ソースおよびレポート作成を有効化および構成するために使用します。
- GuardAPI 外部フィード関数
  これらの GuardAPI 関数は、外部フィードのマッピングを作成するために使用します。
- GuardAPI External S-TAP 関数
  pull_external_stap_keystore CLI コマンドを使用して、中央マネージャーとその管理対象ユニット間で External S-TAP 鍵ストアを移動します。
- GuardAPI ファイル・アクティビティー・モニター関数
  以下の GuardAPI コマンドは、ファイル・アクティビティー・モニターの有効化および無効化、ファイルの調査ダッシュボードのアクティビティーおよびライセンス抽出のスケジュールの構成、ファイル・アクティビティー・モニターに関する情報の取得を行う場合に使用します。
- GuardAPI GIM 関数
  これらの CLI コマンドは、GIM 関数のリスト、更新、割り当て、削除、およびキャンセルに使用します。
- GuardAPI グループ関数
  これらの GuardAPI コマンドは、データ・ソース・グループ関数の作成、リスト、および削除に使用します。

- GuardAPI Health 関数
  これらの GuardAPI コマンドは、ディスクおよびデータベースの Health Analyzer を構成するために使用します。
- GuardAPI 入力生成
  GuardAPI 入力生成を使用すると、ユーザーは 1 つの Guardium レポートの出力を取得して、それを別の Guardium エンティティーへの入力とすることができます。 つまり、ユーザーは準備済みの呼び出しを使用して素早く API の機能を呼び出すことができます。
- GuardAPI 調査ダッシュボード機能
  これらの GuardAPI コマンドは、調査ダッシュボードの機能とパラメーターを有効化、無効化、または構成するために使用します。
- GuardAPI ネイティブ監査関数
  これらの GuardAPI コマンドを使用して、クラウド・データベースに対する DB 監査 (ネイティブ監査) の有効化、無効化、オブジェクト監査 (監査証跡) に対するオブジェクトの追加と削除、構成、コレクター、およびオブジェクトの取得を実行します。
- GuardAPI 異常値検出機能
  以下の GuardAPI コマンドは、異常値検出機能を有効化、無効化、および構成するために使用します。
- GuardAPI プロセス制御関数
  これらの GuardAPI コマンドは、プロセス制御関数の実行、コピー、アップロード、リスト、および削除に使用します。
- GuardAPI 照会再書き込み関数
  コマンド行インターフェースで Guardium API を使用して、ユーザー・インターフェースから実行できない特定の複雑な照会のテストを自動化したり、そうした照会の定義を作成したりします。
- GuardAPI ロール関数
  これらの GuardAPI コマンドは、ロール関数の付与、リスト、および取り消しに使用します。
- GuardAPI Solr 関数
  これらのコマンドを中央マネージャーまたは管理対象ユニットで実行すると、それぞれの (内部 Guardium) Solr データベースを管理できます。
- GuardAPI S-TAP 関数
  これらの CLI コマンドは、S-TAP 関数の作成、リスト、削除、再始動、および設定に使用します。
- GuardAPI 脅威検出分析機能

親トピック: CLI および API

GuardAPI の使用

GuardAPI の使用方法について説明します。

GuardAPI を使用すると、反復作業の自動化が可能となるため、特に大規模な実装環境においては利用価値があります。これらの GuardAPI 関数を呼び出すことにより、素早くさまざまな操作を行うことができます。 例えば、データ・ソースの作成、ユーザー階層の保守、S-TAP® のような Guardium® 機能の保守などの操作を行えます。

GuardAPI を使用するために CLI に適切にログインするには、5 つの CLI アカウント (guardcli1、...、guardcli5) の 1 つでログインし、さらに、アクセス・マネージャーで作成されて admin または cli ロールを付与されたユーザー (GUI username/guiuser) でログイン (「set guiuser」コマンドを発行) することが必要です。 詳しくは、『set guiuser 認証』を参照してください。

GuardAPI は一連の CLI コマンドで、すべてキーワード grdapi で始まります。

- 使用可能なすべての GuardAPI コマンドをリストするには、引数を指定せずに grdapi コマンドを実行するか、または検索引数を指定せずに「grdapi commands」コマンドを実行します。 例:

```
CLI> grdapi
または
CLI> grdapi commands
```

- 特定のコマンドのパラメーターを表示するには、コマンドに続けて「--help=true」を入力します。例:

```
CLI> grdapi list_entry_location --help=true
ID=0
function parameters :
fileName
hostName - required
path - required
ok
```

- 検索文字列を指定して GuardAPI コマンドを検索するには、CLI コマンド grdapi commands <search-string> を使用します。例:

```
CLI> grdapi commands user
ID=0
Matching API Function list :
create_db_user_mapping
create_user_hierarchy
delete_allowed_db_by_user
delete_db_user_mapping
delete_user_hierarchy_by_entry_id
delete_user_hierarchy_by_user
execute_appUserTranslation
execute_ldap_user_import
list_allowed_db_by_user
list_db_user_mapping
list_user_hierarchy_by_parent_user
update_user_db
```

- パラメーターの値リストを表示するには、コマンドに「--get_param_values=<parameter>;」を付けて入力します。 例:

```
CLI> grdapi create_group --get_param_values=appid
Value for parameter 'appid' of function 'create_group' must be one of:
パブリック
監査プロセス・ビルダー
分類
Db2 zOS グループ
エクスプレス・セキュリティー
```

```
IMS zOS グループ
ポリシー・ビルダー
セキュリティー・アセスメント・ビルダー
ID=0
ok
```

表 1. -get_param_values コマンドの構造をサポートする API

API 関数	パラメーター
create_datasource	application、type、severity、shared
create_group	appid、type

大/小文字の区別

パラメーターの構成要素となるキーワードと値には、いずれも大/小文字の区別があります。

スペースを含むパラメーター値

パラメーター値に 1 つ以上のスペースが含まれる場合は、二重引用符文字で囲む必要があります。

例:

```
grdapi create_datasource type ="MS SQL SERVER" ...
```

NULL 値および空文字列

一般的に、GuardAPI 関数を呼び出すときに、必須ではないパラメーターに値を指定しないか、または空文字列 ("""") を設定すると、そのパラメーターは GuardAPI 関数呼び出し時に GuardAPI によって NULL 値に変換されます。そのため GuardAPI に変換されると、そのパラメーターが指定されていない場合と同様に無視されます。

例えば、ポリシー・ルールからあるグループを消去する場合、そのグループに空文字列 ("" "") ではなく、スペース ("" "") を設定します。空文字列 ("""") を使用すると、そのグループを無視し、そのグループ選択を変更しないように GuardAPI に通知されます。

ポリシー値からグループを消去する例

```
grdapi update_rule fromPolicy=V8 ruleDesc="LogFull Details" dbUserGroup=" " dbUser=" " objectGroup=" " commandsGroup=" "
```

戻りコード

GuardAPI コマンドの結果に関わらず、戻りコードは常に出力の最初の行に次に示すフォーマットで返されます。

表 2. 戻りコード

戻りコード	記述
ID=identifier	成功。identifier は操作対象オブジェクトの ID です。例えば、直前に定義したグループの ID です。
ERR=error_code	エラー。error_code はエラーを識別するためのものです。これに続いてエラーについてのテキストの記述が 1 行以上あります。『概要』に共通エラー表があり、『GuardAPI エラー・コード』にエラー・コードの全リストがあります。

例えば、create_group コマンドを使用して agroup という名前のグループ objects を定義する場合、正常に行われればそのグループの ID が返されます。

```
CLI> grdapi create_group desc=agroup type=objects appid=Public
ID=20001
ok
CLI>
```

この ID を list_group_by_id コマンドで使用すると、グループ定義を表示することができます。

```
CLI> grdapi list_group_by_id id=20001
ID=20001
Group GroupId=20001
Group GroupTypeId=3
Group ApplicationId=0
Group GroupDescription=agroup
Group GroupSubtype=null
Group CategoryName=null
Group ClassificationName=null
Group Timestamp=2008-05-10 07:34:11.0
Group type = OBJECTS
Application Type = Public
Touple Group
ok
```

実行不成功の場合は、エラー・コードが返されます。 例えば、無効な ID を指定して再度 list_group_by_id コマンドを実行した場合、次のメッセージを受け取ります。

```
a1.corp.com> grdapi list_group_by_id id=20123
ERR=140
Could not retrieve Group - check Id.
ok
```

共通エラー・コード

100 より小さい値のエラー・コードは、共通のエラー条件用です。100 より大きなエラー・コードは特定の関数に適用され、各関数の後で説明します。

GuardAPI エラー・コードの全リストを表示するには、CLI コマンド・プロンプトで grdapi-errors と入力します。

表 3. 共通エラー・コード

エラー	記述
0	パラメーターが欠落しているか、予期しない例外などの不明エラーです。
1	例外が発生しました。Guardium のサポートに連絡してください。
2	要求された関数を取得できませんでした。関数名を確認してください。すべての関数をリストするには、CLI コマンド grdapi または grdapi commands を引数なしで入力します。 検索文字列を指定して関数名によって検索するには、CLI コマンド grdapi commands <search-string> を使用します。
3	引数が多すぎます。この関数のパラメーター・リストを取得するには、--help=true を指定してこの関数を呼び出します。
4	必須パラメーターが欠落しています。この関数のパラメーター・リストを取得するには、--help=true を指定してこの関数を呼び出します。
5	パラメーターを暗号化解除できませんでした。正しい共有パスワードを使用して暗号化されたかどうかを確認してください。
6	パラメーター・フォーマットが間違っています。関数名に続けて <name=value> フォーマットを使用してパラメーター・リストを指定してください。
7	パラメーター・タイプに対するパラメーター値が間違っています。
8	パラメーター名が間違っています。パラメーターには大/小文字の区別があります。
9	ユーザーの特権は、要求された API 関数には不十分です。
10	パラメーターの暗号化が有効になっていません。共有パスワードが設定されていません。
11	targetHost に API 呼び出し要求を送信できませんでした。
12	パラメーターの検証中にエラーが発生しました。
13	ターゲット・ホストは中央マネージャーの IP アドレスでなければなりません。
14	ターゲット・ホストはこのマネージャーによって管理されていません。
15	ターゲット・ホストがオンラインではありません。
16	ターゲット・ホストはスタンドアロン・ユニットでは指定できません。
17	ユーザーは指定されたオブジェクトで操作を行うことが許可されていません。
18	ターゲット・ホストを指定できません。
19	終了引用符がありません。
20	ユーザーは grdapi commands を実行することが許可されていません。
21	--username および --source-host は grdapi の予約語であり、コマンド行で渡すことはできません。
22	1 つのパラメーター名を複数回指定することはできません。コマンド行を調べて、重複したパラメーターがないか確認してください。
23	値は定数リストに含まれていません。
24	暗号化された値は有効ではありません。
25	有効なパラメーター・フォーマットではありません。パラメーターは <name=value> として指定する必要があり、スペースは使用できません。

GuardAPI アクティビティー・ログ

Guardium アクティビティー・ログでは、システムで実行されるすべての grdapi コマンドが記録されます。 管理者ポータルからコマンドを表示するには「Guardium モニター」タブにある「ユーザー・アクティビティー監査証跡」レポートにナビゲートします。

すべての grdapi アクティビティーは、cli ユーザーに属するものと見なされます。 そのレポートの cli 行をダブルクリックして、「Guardium ユーザー・アクティビティーの詳細」ドリルダウン・レポートを選択します。入力されたすべてのコマンドが、加えられたすべての変更とともにリストされます。 また、コマンド発行元の IP アドレスもリストされます。

暗号化されたパラメーター

GuardAPI はスクリプトから呼び出されますが、スクリプトにはデータ・ソースのパスワードなどの機密情報が含まれる場合があります。 機密情報を常に暗号化しておくため、grdapi コマンドは 1 つの暗号化されたパラメーターを API 関数に渡すことができます。 この暗号化は、システムの共有パスワードを使用して行われます。共有パスワードは管理者によって設定され、多数のシステムで、またすべての一元管理ユニットと統合クラスターの間で共有されます。このため、暗号化されたパラメーターを使用するスクリプトを、同じ共有パスワードを持つマシン上で実行することができます。

注: 共有パスワードが設定されていないシステムで暗号化されたパラメーターを使用する API 呼び出しを実行しようとすると、次のエラー・メッセージが表示されます。

パラメーター暗号化が有効になっていません – 共有パスワードが設定されていません

GUI を介して生成される GuardAPI スクリプトの場合、暗号化が必要な場合には、スクリプト生成を実行するシステムの共有パスワードを使用して暗号化されます。

すべての grdapi 呼び出しにおいて、オプション・パラメーターの encryptedParam を使用可能です。 このパラメーターは、暗号化された値を別のパラメーターに渡すために使用できます。

手動による暗号化の手順を以下に示します。

1. パラメーター暗号化 API を使用します。

encrypt_value API は、暗号化する値およびターゲット・システムの共有パスワード (鍵) を受け入れた後、暗号化された値をプリント出力します。 鍵がシステムの共有パスワードでない場合は、警告がプリント出力されます。

```
a1.corp.com> grdapi encrypt_value --help=true
ID=0
function parameters :
key - required
valueToEncrypt - required
api_target_host
ok
```

表 4. 暗号化されたパラメーター

パラメーター	記述
key	ターゲット・システムの共有パスワード
valueToEncrypt	暗号化される値
api_target_host	一元管理構成に限り、API が実行されるターゲット・ホストを指定できます。中央マネージャー (CM) では、この値は任意の管理対象ユニットのホスト名または IP です。管理対象ユニット上では、この値は CM のホスト名または IP です。

例

```
a1.corp.com> grdapi encrypt_value valueToEncrypt="some value" key=guard
ID=0
-----BEGIN PGP MESSAGE-----
Version: GnuPG v1.4.7 (GNU/Linux)
jA0EAgMCTEIUShudn0tgyTB9GL7wR79UL9X9DCAa6RkUQRbegG52olA4gwOzmpHF
0qEhsd6Uz7l8rUsheUyX9v4=
=c1Cq
-----END PGP MESSAGE-----
```

2. 生成された内容をコピーして、CLI スクリプト内に組み込みます。

```
cli.gsh  コードの例:
set guiuser johny_smith password 3wel9s887s
 grdapi create_datasource type=oracle name=myOra host=somehost application=AuditTask owner=admin user=sa serviceName=ora
encryptedParam=password
-----BEGIN PGP MESSAGE-----
Version: GnuPG v1.4.7 (GNU/Linux)
jA0EAgMCTEIUShudn0tgyTB9GL7wR79UL9X9DCAa6RkUQRbegG52olA4gwOzmpHF
0qEhsd6Uz7l8rUsheUyX9v4=
=c1Cq
-----END PGP MESSAGE-----
```

3. 次のようにスクリプトを実行して grdapi を呼び出します。

```
user> ssh cli@a1.corp.comuser> ssh cli@a1.corp.com
```

一元管理での注意

一元管理環境で GuardAPI を使用する場合は、中央マネージャーでどのようなコンポーネントが定義されているのか、さらに管理対象ユニットでどのようなコンポーネントが定義されているのかを把握しておく必要があります。このトピックについては、『一元管理』を参照してください。

照会 - レポート・ビルダーにおける特定のユーザーの属性の表示

admin ユーザーはすべての照会属性を照会 - レポート・ビルダーで参照可能であり、非 admin ユーザーは admin のみとして設計されている属性 (ID など) 以外の照会属性を照会 - レポート・ビルダーで参照可能です。

一部のエンティティー (完全な SQL など) には多数の属性があります。

デフォルトでは、すべてのユーザー (admin および非 admin) に関するすべての属性が表示されます。

特定のユーザーに関する特定の属性を表示したり非表示にしたりするために、2 つの GuardAPI コマンドが追加されています。

これらの GuardAPI コマンドは、完全な SQL の特定の属性のグループ (VSAM、ISAM、MApReduce、APEX、Hive、BigInsight) のみ有効化/無効化します。

これら 2 つの新しい GuardAPI の名前は、grdapi enable_special_attributes と grdapi disable_special_attributes です。

両方とも、1 つのパラメーター attributesGroup のみ受け取ります。

このパラメーターの有効値は、VSAM、IMS、MapReduce、APEX、Hive、BI (BigInsights)、IMS/VSAM、Db2 i、F5 です (大/小文字の区別はありません)。

各 Grdapi はグループに対応している属性をすべて有効化 (無効化) します。例えば、VSAM の場合は以下の属性を有効化 (無効化) します。

- VSAM レコード
- 削除済みの VSAM レコード
- 挿入済みの VSAM レコード
- 取得済みの VSAM レコード
- 更新済みの VSAM レコード
- VSAM ユーザー・グループ ID

Hive は以下の属性を有効化 (無効化) します。

- Hive コマンド
- Hive データベース
- Hive エラー

- Hive 解析 SQL
- Hive 表名
- Hive ユーザー

注: ユーザーが admin ロールを持つ場合は、引き続き属性が表示されます。これらの属性の有効化または無効化は非 admin ユーザー (admin ロールを持たないユーザー) のみに適用されます。
注: 変更内容を有効にするために GUI を再始動する必要はありません。 ただし、次のような場合を除きます。すなわち、グループ F5 の属性を持つレポートが作成されていて、それが「My New Reports」に追加されている場合は、その属性が有効化されていても、admin ユーザーはレポートを表示する特権を持っていません。レポート・フィールドを表示するためには、GUI を再始動する必要があります。
親トピック: GuardAPI

Guardium REST API

Guardium® REST API を使用すると、ご使用のアプリケーション、または REST API を使用できる任意の場所に、多くの Guardium API コマンドを組み込むことができます。

Guardium REST API の概要

Guardium REST API は、Guardium の機能豊富なコマンド・ライン・インターフェースの Guardium API (guardapi) 関数のラッパーとして機能します。Guardium コレクターで REST API クライアントを登録すると、多くの guardapi 関数に対して REST API 呼び出しを使用できるようになります。Guardium API への REST インターフェースを提供することで、ご使用のシステムに Guardium をより簡単に統合できるようになります。

REST API 対応の guardapi 関数について詳しくは、Guardium API リファレンス (アルファベット順)を参照してください。

REST API を使用できるように Guardium システムをセットアップするには、図 1 を参照して以下のステップを実行します。

1. コマンド・ライン・インターフェース (CLI) を使用して、コレクター側からクライアント ID を登録します。各クライアントは 1 度だけ登録する必要があります。CLI からクライアント ID のクライアント秘密鍵が返されます。
2. REST クライアントから curl コマンド・ライン・ツールを使用して、アクセス・トークンに対する要求をクライアント秘密鍵とともに Guardium アプライアンスに送信します。
3. REST クライアントが認可されると、アクセス・トークンを使用して、サポートされる guardapi 関数を呼び出すことができるようになります。

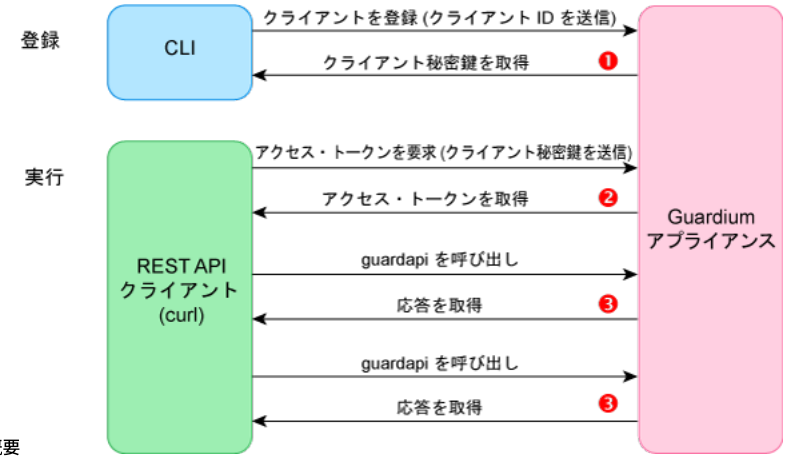


図 1. REST API の概要

REST API の呼び出しの例

以下のシナリオでは、クライアントを登録した後、REST クライアントから guardapi 関数を呼び出す方法を説明します。

1. アプリケーションを登録し、クライアント秘密鍵を取得します。

   CLI からコレクターにアクセスし、以下の grdapi コマンドを実行してアプリケーションを登録します。

   ```
   example.yourdomain.com> grdapi register_oauth_client client_id=client1 grant_types="password"
   redirect_uris="https://TestApp1" scope="read,write"
   ID=0
   ```

   アプリケーションから以下のようなクライアント秘密鍵が返されます。

   ```
   {"client_id":"client1","client_secret":"b1f242a2-1e86-46d6-bf42-
   6298556c2eea","grant_types":"password","scope":"read,write","redirect_uri":"https://TestApp1"}
   ok
   example.yourdomain.com>
   ```

   注: アプリケーションは 1 度だけ登録する必要があります。
2. REST クライアントから、クライアント秘密鍵を指定してアクセス・トークンに対する curl 要求を送信します。

   ```
   C:¥tools¥curl-7.57.0-win64-mingw¥bin>curl -k -X POST -d "client_id=client1&client_secret=b1f242a2-1e86-46d6-bf42-6298556
   c2eea&grant_type=password&username=admin&password=********"  https://example.yourdomain.com:8443/oauth/token
   ```

   Guardium から以下のようなアクセス・トークンが返されます。

   ```
   {"access_token":"29ff4bb4-e622-41cf-97d0-de695ebd756b","token_type":"bearer","expires_in":10799,"scope":"read write"}
   C:¥tools¥curl-7.57.0-win64-mingw¥bin>
   ```

3. このアクセス・トークンを使用して、REST API から Guardium API 関数を呼び出します。以下の例では、REST API から list_datasource_by_id id=20000 に相当する関数を呼び出します。

```
C:¥tools¥curl-7.57.0-win64-mingw¥bin>curl ¥
-k --header "Authorization:Bearer 29ff4bb4-e622-41cf-97d0-de695ebd756b" ¥
--include --header "Content-Type: application/json" ¥
-X GET https://example.yourdomain.com:8443/restAPI/datasource?id=20000
```

この REST API 呼び出しは、指定されたデータ・ソースに関する以下の情報を返します。

```
HTTP/1.1 200 OK
X-FRAME-OPTIONS: SAMEORIGIN
Set-Cookie: JSESSIONID=C7854CAF60CE7B3A6CD585A8173B3222; Path=/; Secure; HttpOnly
Cache-Control: max-age=86400
Expires: Tue, 16 Jan 2018 09:21:52 GMT
Access-Control-Allow-Methods: POST, GET,  PUT, DELETE
Access-Control-Allow-Headers: authorization, origin, X-Requested-With, Content-Type, Accept
Access-Control-Max-Age: 18000
Content-Type: application/json;charset=UTF-8
Content-Length: 914
Date: Mon, 15 Jan 2018 09:21:52 GMT
Server: SQL Guard
[
  {
    "DatasourceId": "https://example.yourdomain.com:8443/restAPI/datasource?id=20000",
    "DatasourceTypeId": "13",
    "Name": "System (9.70.148.141)",
    "Description": "null",
    "Host": "9.70.148.141",
    "Port": "0",
    "ServiceName": "",
      ...
```

API のその他の例

この例では、POST verb を使用して新規データ・ソースを作成します。

```
curl -k --header "Authorization:Bearer 04ce5d90-8d89-4e9c-a060-ec94b4409a71" ¥
--include --header "Content-Type: application/json" ¥
-X POST --data '{"application":"Classifier","host":"192.168.1.54","name":"mydbserver","type":"TEXT:HTTPS"}' ¥
https://192.168.1.10:8443/restAPI/datasource
```

この例では、GET を使用して、すべてのインストール済みアプリケーションの状況を返します。

```
curl -k --header "Authorization:Bearer da186a7e-488d-4cd2-a35b-094b3cc4af86" ¥
--include --header "Content-Type: application/json" ¥
-X GET https://192.168.1.10:8443/restAPI/applications
```

この例では、ID (20001) を指定して既存のデータ・ソースを更新します。

```
curl -k --header "Authorization:Bearer 04ce5d90-8d89-4e9c-a060-ec94b4409a71" ¥
--include --header "Content-Type: application/json" ¥
-X PUT --data '{"id":20001,"description":"My database server."}' ¥
https://192.168.1.10:8443/restAPI/update_datasource_by_id
```

この例では、アプリケーション (1500) を削除します。

```
curl -k --header "Authorization:Bearer 14da5202-f3c6-45d0-bc10-77604e6cede7" ¥
--include --header "Content-Type: application/json" ¥
-X DELETE --data '{"application_id": 1500}' https://192.168.1.10:8443/restAPI/applications
```

REST API 呼び出しでサポートされる guardapi 関数については、Guardium API リファレンス (アルファベット順)を参照してください。

親トピック: GuardAPI

Guardium API リファレンス (アルファベット順)

すべての Guardium guardapi コマンドをアルファベット順に示します。guardapi コマンドをクリックすると、そのコマンドの詳細を確認できます。guardapi コマンドに REST API インターフェースがある場合は、REST 呼び出しも示されます。

GuardAPI および REST API 関数のリスト

A	C	D	E	F	G	I	L
M	N	P	R	S	T	U	V

A

A で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
add_action_to_fam_rule	はい	addActionToFAMRule	POST
add_approved_stap_client	はい	approved_stap_client	POST
add_assessment_datasource	はい	assessment_datasource	POST
add_assessment_test	はい	assessment_test	POST

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
add_assessment_test_by_dsId	はい	assessment_test	POST
add_autodetect_task	はい	add_autodetect_task	POST
add_classifier_datasource	はい	classifier_datasource	POST
add_connection_properties	はい	con_properties	POST
add_datasource_to_entitlement_optimization	はい	addDatasourceToEntitlementOptimization	PUT
add_dm_to_profile	はい	datamartInProfile	PUT
add_group_to_quick_search	いいえ		
add_ip_to_sg	はい	add_ip_to_sg	POST
add_job_dependency	いいえ		
add_objects_native_audit	はい	add_objects_native_audit	POST
add_ranger_config	はい	add_ranger_config	POST
add_ranger_service	はい	add_ranger_service	POST
add_receiver_to_rule_action	はい	receiver_to_rule_action	POST
add_time_period	はい	time_period	POST
applicationRemove	REST のみ	applications	DELETE
assign_load_balancer_groups	いいえ		
assign_qr_condition_to_action	はい	qr_condition_to_action	POST
audit_process_run_status	はい	audit_process_run_status	GET
auto_execute_suggested_dependencies	いいえ		

C

C で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
cancel_distributed_report_target	いいえ		
change_rule_order	はい	change_rule_order	PUT
change_tracker_get_events	はい	change_tracker_get_events	GET
change_tracker_get_params	はい	change_tracker_get_params	GET
change_tracker_get_tasks	はい	change_tracker_get_tasks	GET
change_tracker_reset	はい	change_tracker_reset	PUT
change_tracker_set_params	はい	change_tracker_set_params	PUT
clear_cas_template_set	はい	cas_template_set	PUT
clone_cas_template_set	はい	clone_cas_template_set	POST
clone_extraction_profile	はい	extractionProfile	PUT
clone_policy	はい	clone_policy	POST
close_default_events	いいえ	close_default_events	PUT
configure_archive	はい	configure_archive	PUT
configure_export	はい	configure_export	PUT
configure_purge	はい	configure_purge	PUT
copy_key_file	はい	copy_key_file	PUT
copy_rule	はい	copy_rule	POST
copy_rules	はい	copy_rules	POST
create_ad_hoc_audit_and_run_once	はい	ad_hoc_audit_and_run_once	POST
create_ad_hoc_audit_and_run_with_name	はい	ad_hoc_audit_and_run_once	POST
create_ad_hoc_audit_for_security_assessment	はい	ad_hoc_audit_for_security_assessment	POST
create_alias	はい	alias	POST
create_allowed_db	はい	allowed_db	POST
create_api_parameter_mapping	はい	param_mapping_for_function	POST
create_assessment	はい	assessment	POST
create_autodetect_process	はい	autodetect_processes	POST
create_cas_host_instance	はい	cas_host_instance	POST

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
create_cas_template	はい	cas_template	POST
create_cas_template_set	はい	cas_template_set	POST
create_classifier_action	はい	classifier_action	POST
create_classifier_policy	はい	classifier_policy	POST
create_classifier_process	はい	classifier_process	POST
create_classifier_rule	はい	classifier_rule	POST
create_cloud_datasource	いいえ	cloud_datasource	POST
create_computed_attribute	いいえ	computed_attribute	POST
create_constant_attribute	いいえ	constant_attribute	POST
create_datasource	はい	datasource	POST
create_datasourceRef_by_id	はい	datasource_ref_by_id	POST
create_datasourceRef_by_name	はい	datasource_ref_by_name	POST
create_db_user_mapping	はい	db_user_mapping	POST
create_ef_mapping	はい	create_ef_mapping	PUT
create_entry_location	はい	entry_location	POST
create_fam_rule	はい	famPolicyRule	POST
create_group	はい	group	POST
create_hierarchical_member_to_group_by_desc	はい	hierarchical_member	POST
create_member_to_group_by_desc	はい	group_member	POST
create_member_to_group_by_id	はい	group_member_by_group_id	POST
create_member_to_group_DAMX_Standard_Activity	いいえ		
create_member_to_group_DAMX_Suspicious_Connections	いいえ		
create_online_report	REST のみ	online_report	POST
create_policy	はい	policy	POST
create_qr_action	はい	qr_action	POST
create_qr_add_where	はい	qr_add_where	POST
create_qr_add_where_by_id	はい	create_qr_add_where_by_id	POST
create_qr_condition	はい	qr_condition	POST
create_qr_definition	はい	qr_definition	POST
create_qr_replace_element	はい	qr_replace_element	POST
create_qr_replace_element_byId	はい	create_qr_replace_element_byId	POST
create_quarantine_allowed_until	はい	quarantine_allowed_until	POST
create_quarantine_until	はい	quarantine_until	POST
create_role	はい	create_role	POST
create_rule	はい	rule	POST
create_rule_action	はい	rule_action	POST
create_stap_inspection_engine	はい	inspection_engine	POST
create_test_exception	いいえ	test_exception	POST
create_user	はい	user	POST
create_user_hierarchy	はい	user_hierarchy	POST

D

D で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
datamart_copy_file_bundle	はい	datamart_copy_file_bundle	PUT
datamart_include_file_header	はい	datamart_include_file_header	PUT
datamart_refresh_metadata	はい	datamart_refresh_metadata	PUT
datamart_run_once_now	はい	datamart_run_once_now	PUT
datamart_set_active	はい	datamart_set_active	PUT
datamart_set_date_format	はい	datamart_set_date_format	PUT

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
datamart_set_inactive	はい	datamart_set_inactive	PUT
datamart_update_copy_file_info	はい	datamart_update_copy_file_info	PUT
datamart_validate_copy_file_info	はい	datamart_validate_copy_file_info	PUT
delete_alias	はい	alias	DELETE
delete_allowed_db_by_entry_id	はい	allowed_db	DELETE
delete_allowed_db_by_user	はい	allowed_db	DELETE
delete_analytic_user_feedback	はい	delete_analytic_user_feedback	PUT
delete_api_parameter_mapping	はい	param_mapping_for_function	DELETE
delete_approved_stap_client	はい	approved_stap_client	DELETE
delete_archive_configuration	はい	delete_archive_configuration	DELETE
delete_assessment	はい	assessment	DELETE
delete_assessment_datasource	はい	assessment_datasource	DELETE
delete_assessment_test	はい	assessment_test	DELETE
delete_audit_process_result	はい	audit_process_result	DELETE
delete_autodetect_scans_for_process	はい	autodetect_scans_for_process	DELETE
delete_cas_host	はい	cas_host	DELETE
delete_cas_host_instance	はい	cas_host_instance	DELETE
delete_cas_template	はい	cas_template	DELETE
delete_cas_template_set	はい	cas_template_set	DELETE
delete_classifier_action	はい	classifier_action	DELETE
delete_classifier_policy	はい	classifier_policy	DELETE
delete_classifier_process	はい	classifier_process	DELETE
delete_classifier_rule	はい	classifier_rule	DELETE
delete_computed_attribute	いいえ	computed_attribute	DELETE
delete_constant_attribute	いいえ	constant_attribute	DELETE
delete_datasource_by_id	はい	delete_datasource_by_id	DELETE
delete_datasource_by_name	はい	datasource	DELETE
delete_datasourceRef_by_id	はい	datasource_ref	DELETE
delete_datasourceRef_by_name	はい	datasource_ref	DELETE
delete_db_user_mapping	はい	db_user_mapping	DELETE
delete_distributed_report_result_for_period	いいえ		
delete_ef_mapping	はい	delete_ef_mapping	DELETE
delete_entry_location	はい	entry_location	DELETE
delete_export_configuration	はい	delete_export_configuration	DELETE
delete_group_by_desc	はい	group	DELETE
delete_group_by_id	はい	group	DELETE
delete_group_from_quick_search	いいえ		
delete_hierarchical_member_from_group_by_desc	はい	hierarchical_member	DELETE
delete_imscheckpoint_record	はい	ims_checkpoint	DELETE
delete_inactive_stap	はい	delete_inactive_stap	POST
delete_job_dependencies	いいえ		
delete_member_from_group_by_desc	はい	group_member	DELETE
delete_member_from_group_by_id	はい	group_member_by_group_id	DELETE
delete_policy	はい	policy	DELETE
delete_quarantine	はい	quarantine	DELETE
delete_rule	はい	rule	DELETE
delete_schedule	はい	スケジュール	DELETE
delete_stap_inspection_engine	はい	inspection_engine	DELETE
delete_user	はい	user	DELETE
delete_user_hierarchy_by_entry_id	はい	user_hierarchy	DELETE

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
delete_user_hierarchy_by_user	はい	user_hierarchy	DELETE
deployApplication	REST のみ	application_creation_task	POST
disable_advanced_threat_scanning	はい	disable_advanced_threat_scanning	PUT
disable_auto_execute_suggested_dependencies	いいえ		
disable_big_data_interface	はい	bigDataInterface	DELETE
disable_deprecated_protocols	いいえ		GET
disable_embed_eastern_font	いいえ		
disable_enable_solr_when_non_matched_version	いいえ		
disable_entitlement_optimization	はい	disableEntitlementOptimization	PUT
disable_fam_crawler	はい	disable_fam_crawler	PUT
disable_health_analyzer	はい	disableHealthAnalyzer	PUT
disable_monitoring_ranger_service	はい	disable_monitoring_ranger_service	PUT
disable_native_audit	はい	disable_native_audit	POST
disable_outliers_detection	はい	disableOutliersDetection	PUT
disable_outliers_detection_agg	はい	disableOutliersDetectionOnAgg	PUT
disable_outliers_detection_on_collectors_cross_cm	はい	disableOutliersDetectionOnAggCrossCm	PUT
disable_purge	はい	disable_purge	DELETE
disable_quick_search	はい	disable_quick_search	PUT
disable_special_attributes	いいえ	disable_special_attributes	PUT
disable_threat_detection_use_case	はい	disable_threat_detection_use_case	POST
display_stap_config	はい	display_stap_config	GET

E

E で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
enable_advanced_threat_scanning	はい	enable_advanced_threat_scanning	PUT
enable_big_data_interface	はい	bigDataInterface	PUT
enable_deprecated_protocols	いいえ		GET
enable_embed_eastern_font	いいえ		
enable_entitlement_optimization	はい	enableEntitlementOptimization	PUT
enable_fam_crawler	はい	enable_fam_crawler	PUT
enable_health_analyzer	はい	enableHealthAnalyzer	PUT
enable_monitoring_ranger_service	はい	enable_monitoring_ranger_service	PUT
enable_native_audit	はい	enable_native_audit	POST
enable_outliers_detection	はい	enable_outliers_detection	PUT
enable_outliers_detection_agg	はい	enableOutliersDetectionOnAgg	PUT
enable_outliers_detection_cross_cm_agg	はい	enableOutliersDetectionCrossCMOnAgg	PUT
enable_outliers_detection_cross_cm_collector	はい	enableOutliersDetectionCrossCMOnCollector	PUT
enable_quick_search	はい	enable_quick_search	PUT
enable_special_attributes	いいえ	enable_special_attributes	PUT
enable_threat_detection_use_case	はい	enable_threat_detection_use_case	POST
encrypt_value	はい	encrypt_value	POST
execute_appUserTranslation	はい	app_user_translation	PUT
execute_assessment	はい	execute_assessment	PUT
execute_auditProcess	はい	audit_process	PUT
execute_autodetect_process	はい	execute_autodetect_process	PUT
execute_cls_process	はい	execute_cls_process	PUT
execute_flatLogProcess	はい	execute_flatLogProcess	PUT
execute_incidentGenProcess	はい	execute_incident_gen_process	PUT
execute_incidentGenProcess_byDetails	はい	execute_incident_gen_process_by_details	PUT

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
execute_ldap_user_import	はい	ldap_user	POST
execute_populateGroupFromQuery	はい	populate_group_from_query	PUT
export_definition	いいえ	export_definition	PUT

F

F で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
f5_add_apps_config	いいえ	f5_apps_config	POST
f5_add_data_params	いいえ	f5_data_params	POST
f5_delete_apps_config	いいえ	f5_apps_config	DELETE
f5_delete_data_params	いいえ	f5_data_params	DELETE
f5_list_apps_config	いいえ	f5_apps_config	GET
f5_list_data_params	いいえ	f5_data_params	GET
f5_update_data_params	いいえ	f5_data_params	PUT
flatten_hierarchical_groups	いいえ	flatten_hierarchical_groups	PUT

G

G で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
get_all_modifiable_guard_params	はい	get_all_modifiable_guard_params	GET
get_datamart_info	いいえ	get_datamart_info	PUT
get_debug_level	いいえ		
get_definitions_data_sets	REST のみ	get_definitions_data_sets	GET
get_definitions_items	REST のみ	get_definitions_items	POST
get_distributed_report_target_info	いいえ		
get_eagle_eye_debug_level	はい	get_eagle_eye_debug_level	GET
get_eagle_eye_info	はい	get_eagle_eye_info	GET
get_eagle_eye_scanners_info	はい	get_eagle_eye_scanners_info	GET
get_eagle_eye_symptom_period_hours	はい	get_eagle_eye_symptom_period_hours	GET
get_eagle_eye_symptoms_info	はい	get_eagle_eye_symptoms_info	GET
get_entitlement_optimization_info	はい	getEntitlementOptimizationInfo	PUT
get_expiration_date_for_restored_day	はい	expiration_date_for_restored_day	GET
get_extraction_profile_info	はい	extractionProfile	GET
get_fam_crawler_info	はい	get_fam_crawler_info	GET
get_flatLogProcessType	はい	flatLogProcessType	GET
get_guard_param	はい	get_guard_param	GET
get_hadoop_cluster_status	いいえ		
get_ip_to_alias_overwrites	はい	ip_to_alias_overwrites	GET
get_ip_to_alias_selected	はい	ip_to_alias_selected	GET
get_istap_config	はい	istap_config	GET
get_istap_status	はい	istap_status	GET
get_job_process_concurrency_limit	いいえ		
get_load_balancer_load_map	いいえ		
get_load_balancer_params	いいえ		
get_native_audit_collectors	はい	nau_collectors_list	GET
get_native_audit_configurations	はい	nau_configurations	GET
get_native_audit_objects	はい	nau_objects_list	GET
get_outliers_detection_info	いいえ		
get_purge_batch_size	いいえ	purge_batch_size	GET
get_quick_search_info	いいえ		

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
get_ranger_config	はい	get_ranger_config	GET
get_ranger_services_status	はい	get_ranger_services_status	GET
get_secured_protocols_info	いいえ		GET
get_solr_cluster_info	いいえ		
get_solr_status	いいえ		
get_threat_detection_use_case_info	はい	threat_detection_use_case_info	GET
get_unit_pinger	いいえ	get_unit_pinger	GET
get_ztap_logging_config	はい	ztap_logging_config	GET
getAppStatus	REST のみ	applications	GET
getDeployStatus	REST のみ	application_creation_task	GET
getFieldsTitles	REST のみ	fieldsTitles	GET
getOAuthTokenExpirationTime	いいえ	OAuthTokenExpirationTime	GET
gim_assign_bundle_or_module_to_client_by_version	はい	gim_client_assign	PUT
gim_assign_latest_bundle_or_module_to_client	はい	gim_assign_latest_bundle	PUT
gim_cancel_install	はい	gim_cancel_install	PUT
gim_cancel_uninstall	はい	gim_cancel_uninstall	PUT
gim_get_available_modules	はい	gim_available_modules	GET
gim_get_client_last_event	はい	gim_client_last_event	GET
gim_get_global_param	いいえ		
gim_get_modules_running_status	いいえ		
gim_list_bundles	はい	gim_bundle	GET
gim_list_client_modules	はい	gim_list_client_modules	GET
gim_list_client_params	はい	gim_client_params	GET
gim_list_mandatory_params	いいえ	gim_mandatory_bundle	GET
gim_list_registered_clients	はい	gim_registered_clients	GET
gim_list_unused_bundles	はい	list_unused_bundles	GET
gim_load_package	はい	gim_package	GET
gim_remote_activation	いいえ		
gim_remove_bundle	はい	gim_remove_bundle	DELETE
gim_reset_client	いいえ		
gim_schedule_install	はい	gim_schedule_install	PUT
gim_schedule_uninstall	はい	gim_schedule_uninstall	PUT
gim_set_diagnostics	いいえ		
gim_set_global_param	いいえ		
gim_unassign_client_module	はい	gim_unassign_client_module	PUT
gim_uninstall_module	はい	gim_uninstall_module	DELETE
gim_update_client_params	はい	gim_client_params	PUT
grant_role_to_object_by_id	はい	grant_role_to_object_by_id	PUT
grant_role_to_object_by_Name	はい	grant_role_to_object_by_Name	PUT

I

I で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
import_definitions	REST のみ	import_definitions	POST

L

L で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
list_all_reports	はい	list_all_reports	GET
list_allowed_db_by_user	はい	allowed_db	GET

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
list_approved_stap_client	はい	approved_stap_client	GET
list_assessment_tests	はい	assessment_test	GET
list_assessments	はい	assessment	GET
list_associated_stap_mu_groups	はい	list_associated_stap_mu_groups	GET
list_autodetect_processes	はい	autodetect_processes	GET
list_autodetect_tasks_for_process	はい	autodetect_tasks_for_process	GET
list_available_tests	はい	available_test	GET
list_cas_host_instances	はい	cas_host_instance	GET
list_cas_hosts	はい	cas_host	GET
list_cas_template_sets	はい	cas_template_set	GET
list_cas_templates	はい	cas_template	GET
list_classifier_policy	はい	classifier_policy	GET
list_classifier_process	はい	classifier_process	GET
list_cloud_datasource_by_name	いいえ	cloud_datasource	GET
list_compatibility_modes	はい	compatibility	GET
list_datasource_by_id	はい	datasource	GET
list_datasource_by_name	はい	datasource	GET
list_datasourceRef_by_id	はい	datasource_ref	GET
list_datasourceRef_by_name	はい	datasource_ref	GET
list_db_drivers	はい	db_drivers	GET
list_db_drivers_by_details	はい	list_db_drivers_by_details	GET
list_db_user_mapping	はい	db_user_mapping	GET
list_ef_mapping	はい	list_ef_mapping	GET
list_ef_report	はい	list_ef_report	GET
list_entry_location	はい	entry_location	GET
list_existing_job_dependencies	いいえ		
list_expiration_dates_for_restored_days	はい	expiration_dates_for_restored_days	GET
list_group_by_desc	はい	group	GET
list_group_by_id	はい	group	GET
list_group_members_by_desc	はい	group_members_by_group_desc	GET
list_group_members_by_id	はい	group_members_by_group_id	GET
list_health_node	はい	list_health_node	GET
list_imscheckpoint_records	はい	ims_checkpoint	GET
list_inspection_engines	はい	inspection_engine	GET
list_job_dependencies_tree	いいえ		
list_param_mapping_for_function	いいえ	param_mapping_for_function	GET
list_parameter_names_by_report_name	はい	list_parameter_names_by_report_name	GET
list_policy	はい	policy	GET
list_policy_fam_rule	はい	famPolicyRule	GET
list_policy_rules	はい	rule	GET
list_qr_action	はい	qr_action	GET
list_qr_add_where	はい	qr_add_where	GET
list_qr_add_where_by_id	はい	list_qr_add_where_by_id	GET
list_qr_condition	はい	qr_condition	GET
list_qr_condition_to_action	はい	qr_condition_to_action	GET
list_qr_definitions	はい	list_qr_definitions	GET
list_qr_replace_element	はい	qr_replace_element	GET
list_qr_replace_element_byId	はい	list_qr_replace_element_byId	GET
list_quick_search_groups	いいえ		
list_ranger_configs	はい	list_ranger_configs	GET

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
list_ranger_staps	はい	list_ranger_staps	GET
list_ready_files	いいえ		
list_roles	はい	role	GET
list_roles_granted_to_object_by_id	はい	roles_granted_to_object_by_id	GET
list_roles_granted_to_object_by_Name	はい	role	GET
list_scheduler_jobs	いいえ		
list_schedules	はい	schedules	GET
list_stap_verification_results	いいえ		
list_staps	はい	stap	GET
list_suggested_job_dependencies	いいえ		
list_user_hierarchy_by_parent_user	はい	user_hierarchy	GET
list_user_roles	はい	userRole	GET
list_users	はい	user	GET
list_utilization_thresholds	いいえ	utilization_thresholds	GET
load_mongodb	いいえ		
load_mongodb_by_datasource	いいえ		
local_disable_deprecated_protocols	いいえ		GET
local_enable_big_data_interface	はい	bigDataInterface	PUT

M

M で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
make_bundle_with_uploaded_kernel_module	いいえ		
make_primary_cm	はい	make_primary_cm	POST
modify_autodetect_process	はい	autodetect_processes	PUT
modify_ef_mapping	はい	modify_ef_mapping	PUT
modify_guard_param	はい	modify_guard_param	POST
modify_job_dependency	いいえ		
modify_schedule	はい	スケジュール	PUT
modify_va_summary_key	はい	modify_va_summary_key	PUT
must_gather	はい	must_gather	PUT

N

N で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
non_credential_scan	いいえ	non_credential_scan	PUT
nscd	いいえ		

P

P で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
patch_install	はい	patch_install	PUT
pause_or_resume_job	はい	スケジュール	PUT
policy_fam_rule_delete	はい	famPolicyRule	DELETE
policy_install	はい	policy_install	POST
populate_from_dependencies	いいえ	populate_from_dependencies	POST
populateMembersForGroup	はい	populateMembersForGroup	GET
pull_external_stap_keystore	いいえ	pull_new_proxy_keystore	PUT

R

R で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
refresh_quick_search_groups	いいえ		
refresh_stap_info	はい	refresh_stap_info	GET
register_oauth_client	いいえ	register_oauth_client	POST
register_oauth_internal_client	いいえ		
reinstall_policy	はい	reinstall_policy	POST
reinstall_policy_rule	はい	reinstall_policy_rule	PUT
remove_all_qr_replace_elements	はい	remove_all_qr_replace_elements	DELETE
remove_all_qr_replace_elements_byId	はい	remove_all_qr_replace_elements_byId	DELETE
remove_classifier_datasource	はい	classifier_datasource	DELETE
remove_connection_properties	はい	con_properties	DELETE
remove_datasource_from_entitlement_optimization	はい	removeDatasourceFromEntitlementOptimization	PUT
remove_dm_from_profile	はい	datamartInProfile	DELETE
remove_extraction_profile	はい	extractionProfile	DELETE
remove_objects_native_audit	はい	remove_objects_native_audit	POST
remove_qr_action	はい	qr_action	DELETE
remove_qr_add_where_by_id	はい	remove_qr_add_where_by_id	DELETE
remove_qr_condition	はい	qr_condition	DELETE
remove_qr_definition	はい	qr_definition	DELETE
remove_qr_replace_element_byId	はい	remove_qr_replace_element_byId	DELETE
remove_ranger_config	はい	remove_ranger_config	DELETE
remove_ranger_service	はい	remove_ranger_service	DELETE
replace_active_profile	はい	extractionProfile	POST
reregister_agg_collector	いいえ		
rerun_distributed_report	いいえ		
reset_solr_configuration	いいえ		
reset_unit_utilization_data	はい	reset_unit_utilization_data	PUT
reset_va_summary_by_id	はい	reset_va_summary_by_id	PUT
reset_va_summary_by_key	はい	reset_va_summary_by_key	PUT
rest_export_definitionr	REST のみ	export_definition	POST
restart_cloud_instance	はい	restart_cloud_instance	POST
restart_job_queue_listener	はい	restart_job_queue_listener	PUT
restart_solr	いいえ		
restart_stap	はい	stap	PUT
restart_unit_pinger	いいえ	restart_unit_pinger	PUT
retreiveUpdatedUsers	はい	retreiveUpdatedUsers	GET
retrieveApiParameters	REST のみ	restapi	GET
retrieveAPIs	REST のみ	restapi	GET
revoke_ignore_stap	はい	revoke_ignore_stap	POST
revoke_role_from_object_by_id	はい	revoke_role_from_object_by_id	DELETE
revoke_role_from_object_by_Name	はい	revoke_role_from_object_by_Name	DELETE
revokeOAuthClient	はい	revokeClient	DELETE
revokeOAuthToken	いいえ	revokeToken	DELETE
run_database_instance_discovery	はい	run_database_instance_discovery	POST
	はい	run_diagnostics	PUT

S

S で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
schedule_job	はい	schedule_job	PUT
検索	REST のみ	検索	GET

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
set_debug_level	いいえ		
set_distributed_report_target	いいえ		
set_eagle_eye_debug_level	はい	set_eagle_eye_debug_level	POST
set_eagle_eye_parameter	はい	set_eagle_eye_parameter	POST
set_eagle_eye_scanner_parameter	はい	set_eagle_eye_scanner_parameter	POST
set_eagle_eye_symptom_parameter	はい	set_eagle_eye_symptom_parameter	POST
set_eagle_eye_symptom_period_hours	はい	set_eagle_eye_symptom_period_hours	POST
set_enterprise_search_options	いいえ		
set_entitlement_datasource_parameter	はい	setEntitlementDatasourceParameter	PUT
set_expiration_date_for_restored_day	はい	expiration_date_for_restored_day	PUT
set_flatLogProcessType	はい	flatLogProcessType	PUT
set_import	はい	set_import	PUT
set_ip_to_alias_overwrites	はい	ip_to_alias_overwrites	PUT
set_ip_to_alias_selected	はい	ip_to_alias_selected	PUT
set_job_process_concurrency_limit	いいえ		
set_ktap_debug	はい	stap_debug	PUT
set_load_balancer_param	いいえ		
set_outliers_detection_parameter	いいえ		
set_purge_batch_size	いいえ	purge_batch_size	PUT
set_stap_debug	はい	stap_debug	PUT
set_user_roles	はい	user_roles	PUT
set_zk_tlog_properties	いいえ	set_zk_tlog_properties	PUT
set_ztap_logging_config	はい	ztap_logging_config	PUT
setOAuthTokenExpirationTime	いいえ	OAuthTokenExpirationTime	PUT
show_autodetect_process_status	はい	autodetect_process_status	GET
show_backup_cm_ip	はい	show_backup_cm_ip	GET
show_job_dependency_execution_profile	いいえ		
start_istap_monitor	はい	start_istap_monitor	PUT
stop_audit_process	はい	stop_audit_process	PUT
stop_autodetect_process	はい	stop_autodetect_process	PUT
stop_istap_monitor	はい	stop_istap_monitor	PUT
stop_solr	いいえ		
store_stap_approval	はい	stap_approval	POST

T

T で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
test_datasource_connection	はい	test_connection	POST

U

U で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
unassign_load_balancer_groups	いいえ		
unassign_qr_condition_from_action	はい	qr_condition_to_action	DELETE
uninstall_policy_rule	はい	uninstall_policy_rule	DELETE
unschedule_datamart	はい	datamartSchedule	DELETE
update_alias	はい	alias	PUT
update_assessment	はい	assessment	PUT
update_assessment_test	はい	assessment_test	PUT
update_cas_host_instance	はい	cas_host_instance	PUT

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
update_cas_template	はい	cas_template	PUT
update_classifier_action	はい	classifier_action	PUT
update_classifier_log_level	はい	classifier_log_level	PUT
update_classifier_policy	はい	classifier_policy	PUT
update_classifier_process	はい	classifier_process	PUT
update_classifier_rule	はい	classifier_rule	PUT
update_cloud_datasource	いいえ	cloud_datasource	PUT
update_computed_attribute	いいえ	computed_attribute	PUT
update_constant_attribute	いいえ	constant_attribute	PUT
update_datamart	はい	update_datamart	PUT
update_datasource_by_id	はい	update_datasource_by_id	PUT
update_datasource_by_name	はい	update_datasource_by_name	PUT
update_entry_location	はい	entry_location	PUT
update_group_by_desc	はい	update_group_by_desc	PUT
update_group_by_id	はい	update_group_by_id	PUT
update_istap_config	はい	istap_config	PUT
update_policy	はい	policy	PUT
update_qr_action	はい	qr_action	PUT
update_qr_add_where_by_id	はい	update_qr_add_where_by_id	PUT
update_qr_condition	はい	qr_condition	PUT
update_qr_definition	はい	qr_definition	PUT
update_qr_replace_element_byId	はい	update_qr_replace_element_byId	PUT
update_quarantine_allowed_until	はい	quarantine_allowed_until	PUT
update_quarantine_until	はい	quarantine_until	PUT
update_ranger_config	はい	update_ranger_config	PUT
update_ranger_service	はい	update_ranger_service	PUT
update_rule	はい	update_rule	PUT
update_stap_config	はい	stap_config	PUT
update_user	はい	user	PUT
update_user_db	はい	user_db	PUT
update_utilization_thresholds	いいえ	utilization_thresholds	PUT
upload_custom_data	はい	custom_data	POST

V

V で始まる API コマンド。

Guardapi 関数名	REST が使用可能かどうか	REST リソース名	REST verb
validateManifest	REST のみ	application_creation_task	POST
verify_stap_inspection_engine_with_sequence	いいえ		

親トピック: GuardAPI

GuardAPI アーカイブおよびリストア関数

list_expiration_dates_for_restored_days

すべてのリストア日における有効期限をリストします。

パラメーター	値のタイプ	記述
newExpDate	文字列	必須。 リストア日における新しい有効期限。
restoredDay	文字列	必須。 データのリストア日を指定します。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi list_expiration_dates_for_restored_days

get_expiration_date_for_restored_day

特定のリストア日に関連付けられた有効期限を取得します。

パラメーター	値のタイプ	記述
newExpDate	文字列	必須。 リストア日における新しい有効期限。
restoredDay	文字列	必須。 データのリストア日を指定します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi get_expiration_date_for_restored_day restoredDay=restoredDay
```

ここで、restoredDay は、実際の日 `yyyy-mm-dd hh:mi:ss` または `NOW -10 day` のような相対日のいずれかの形式になります。

set_expiration_date_for_restored_day

特定のリストア日における有効期限を設定します。

パラメーター	値のタイプ	記述
newExpDate	文字列	必須。 リストア日における新しい有効期限。
restoredDay	文字列	必須。 データのリストア日を指定します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi set_expiration_date_for_restored_day newExpDate=newExpDate restoredDay=restoredDay
```

ここで、newExpDate および restoredDay は、実際の日 `yyyy-mm-dd hh:mi:ss` または `NOW -10 day` のような相対日のいずれかの形式になります。

set_import

統合データのインポートを開始または停止します。

パラメーター	値のタイプ	記述
state	文字列	必須。START または STOP
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi set_import [START]
```

configure_export

統合データのエクスポートを構成します。

パラメーター	値のタイプ	記述
aggHost	文字列	必須。アグリゲーター のホスト名。
aggSecHost	文字列	
exportOlderThan	integer	必須。エクスポートするデータの時間別詳細。
exportValues	integer	必須。0, 1
ignoreOlderThan	integer	必須。無視するデータの時間別詳細。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi configure_export [aggHost] [aggSecHost] [exportOlderThan] [exportValues] [ignoreOlderThan]
```

configure_archive

統合データのアーカイブを構成します。

パラメーター	値のタイプ	記述
accessKey	文字列	アグリゲーターの共有パスワード。
archiveOlderThan	integer	必須。アーカイブするデータの時間別詳細。
archiveValues	integer	必須。 0 または 1
bucketName	文字列	
destHost	文字列	アーカイブ先のホスト名。
ignoreOlderThan	integer	必須。無視するデータの時間別詳細。
passwd	文字列	パスワード
passwdRetype	文字列	パスワードの再入力
port	integer	ポート番号
protocol	文字列	必須。SCP、FTP、または AMAZON
retention	integer	保持する期間。
secretKey	文字列	
targetDir	文字列	
userName	文字列	ユーザー名。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi configure_archive [accessKey] [archiveOlderThan] [archiveValues][bucketName][destHost][ignoreOlderThan][passwd]
[passwdRetype] [port] [protocol] [retention] [secretKey] [targetDir] [userName]
```

親トピック: GuardAPI

GuardAPI アセスメント関数

以下の CLI コマンドは、アセスメント関数を追加、削除、および更新するために使用します。

下記の GuardAPI コマンドは、以下の目的で使用します。

- セキュリティー・アセスメント定義の追加、削除、更新
- 既存のセキュリティー・アセスメントでのデータ・ソースの追加、削除
- 既存のセキュリティー・アセスメントでのテストの追加、削除

create_assessment

この GuardAPI コマンドは、セキュリティー・アセスメントを追加するために使用します。

表 1. create_assessment

パラメーター	値のタイプ	記述
assessmentDescription	文字列	必須。フリー・テキスト (固有)。同じ記述のアセスメントが前に存在していないことを確認する必要があります。 既に存在している場合はエラーになります。
fromDate		有効な日付または相対的な日付。必須ではありません。デフォルトは NOW -1 DAY です。
toDate		有効な日付または相対的な日付。必須ではありません。デフォルトは NOW です。
FilterClientIP		有効な IP アドレス。必須ではありません。デフォルトは null です。
FilterServerIP		有効な IP アドレス。必須ではありません。デフォルトは null です。

アクション: すべてのパラメーターが検証されたら、SECURITY_ASSESSMENT 表に新規レコードが作成されます (MODIFIED_FLAG はデフォルトの 0 のままです)

例

```
grdapi create_assessment assessmentDescription=Assess1
```

add_assessment_datasource

この GuardAPI コマンドは、セキュリティー・アセスメントにデータ・ソースを追加するために使用します。

表 2. add_assessment_datasource

パラメーター	値のタイプ	記述
assessmentDescription	文字列	必須。フリー・テキスト。固有。同じ記述のアセスメントが前に存在していないことを確認する必要があります。既に存在している場合はエラーになります。
datasourceName	文字列	必須。フリー・テキスト: 既存のデータ・ソースの名前でなければなりません。既存のデータ・ソースが存在しない場合はエラーになります。

アクション: すべてのパラメーターが検証されたら、ASSESSMENT_DATASOURCE にレコードが追加されます。その際に、アセスメントとデータ・ソースの ASSESSMENT ID と DATASOURCE ID には指定された名前が使用されます。

例

```
grdapi add_assessment_datasource assessmentDescription=Assess1 datasourceName=DS1
```

add_assessment_test

この GuardAPI コマンドは、既存のセキュリティー・アセスメントにテストを追加するために使用します。

パラメーター	値のタイプ	記述
assessmentDescription	文字列	必須 - フリー・テキスト (固有)。同じ記述のアセスメントが前に存在していないことを確認する必要があります。既に存在している場合はエラーになります
testDescription	文字列	必須 - AVAILABLE_TEST 内の既存のテストの TEST_DESC と一致している必要があります。既存のテストが存在しない場合はエラーになります。
severity	文字列	SEVERITY_DESC 表と照合して検証します (DESCRIPTION を使用) - 必須ではありません。デフォルトは INFO です。
thresholdValue		available_test で必須のしきい値が 0 の場合、このパラメーターを無視します。 そうではなく、available_test で必須の値 (しきい値) が 1 の場合、パラメーターは整数である必要があります このパラメーターが指定しないと、AVAILABLE_TEST の DEFAULT_THRESHOLD_VALUE が使用されます。

パラメーター	値のタイプ	記述
exceptionsGroup		AVAILABLE_TEST 内の CAN_HAVE_EXCEPTIONS_GROUP 値を確認します。 このパラメーターは必須ではありません。 0 の場合、(例外グループはこのテストでサポートされない): このパラメーターを指定するとエラーになります (このテストでは例外グループを指定できません)。パラメーターが指定されない場合、-1 を使用してデータを設定します。 そうでない場合 (例外グループがこのテストでサポートされる): パラメーターが指定されない場合、-1 を使用してデータを設定します。パラメーターが指定された場合、グループを検証してグループ ID を使用します。 グループを検証するには、GROUP_DESCRIPTION が指定した記述と一致するレコードを GROUP_DESC から選択し、レコードが存在するかどうか、および GROUP_TYPE_ID を確認します。 そのグループが存在し、GROUP_TYPE_ID != 55 の場合はエラー「例外グループのタイプは「VA 例外」でなければなりません」が出されます。 そのグループが存在し、タイプが 55 である場合は、GROUP_ID が使用されます。

追加の検証: ASSESSMENT_TEST 内に ASSESSMENT_ID および TEST_ID のレコードが既に存在するかどうかを確認します。そのレコードが存在する場合はエラー「このテストは既にアセスメントに存在するため、再度追加することはできません」が出されます。

アクション: すべてのパラメーターが検証されたら、ASSESSMENT_TEST にレコードを追加します (注: 重大度として、記述に指定された値を設定する必要があります)。

例

```
grdapi add_assessment_test  assessmentDescription=Assess1 testDescription="The first test"
```

delete_assessment

この GuardAPI コマンドは、セキュリティー・アセスメントを削除するために使用します。

パラメーター	値のタイプ	記述
assessmentDescription	文字列	必須。フリー・テキスト。固有。同じ記述のアセスメントが前に存在していないことを確認する必要があります。既に存在している場合はエラーになります。

追加の検証: 以下を行って、削除対象のアセスメントの結果が存在していないことを確認する必要があります。

Select count (*) from ASSESSSMENT_RESULT_HEADER where ASSESSMENT_ID = TheIdToRemve

select で 0 より大きい値が返された場合、削除されずに、エラーになります。

アクション: パラメーターが検証されたら (セキュリティー・アセスメント・レコードが特定され、そのアセスメントの結果が存在しない場合)、SECURITY_ASSESSMENT レコード、ASSESSMENT_TEST レコード、および ASSESSMENT_DATASOURCE レコードを削除します (この 3 つはすべて ASSESSMENT_ID を使用して削除します)。

例

```
grdapi delete_asssessment assessmentDescription=Assess1
```

delete_assessment_datasource

この GuardAPI コマンドは、セキュリティー・アセスメントからデータ・ソースを削除するために使用します。

パラメーター	値のタイプ	記述
assessmentDescription	文字列	必須。フリー・テキスト (固有)。同じ記述のアセスメントが前に存在していないことを確認する必要があります。 既に存在している場合はエラーになります。
datasourceName	文字列	必須。フリー・テキスト: 既存のデータ・ソースの名前でなければなりません。既存のデータ・ソースが存在しない場合はエラーになります。

アクション: すべてのパラメーターが検証されたら、指定されたアセスメントとデータ・ソースのレコードが ASSESSMENT_DATASOURCE 内にあるかどうかを確認します。そのレコードがない場合はエラーになります。それ以外の場合、そのレコードを削除します。

例

```
grdapi delete_asssessment_datasource assessmentDescription=Assess1 datasourceName=DS1
```

delete_assessment_test

この GuardAPI コマンドは、既存のセキュリティー・アセスメントからテストを削除するために使用します。

パラメーター	値のタイプ	記述
assessmentDescription	文字列	必須。フリー・テキスト (固有)。同じ記述のアセスメントが前に存在していないことを確認する必要があります。既に存在している場合はエラーになります
testDescription	文字列	フリー・テキスト: AVAILABLE_TEST 内の既存のテストの TEST_DESC と一致している必要があります。既存のテストが存在しない場合はエラーになります。

追加の検証: ASSESSMENT_TEST 内に ASSESSMENT_ID および TEST_ID のレコードがあるかどうかを確認します。そのレコードがない場合はエラー「このテストはアセスメントに存在していません」が出されます。

アクション: すべてのパラメーターが検証されたら、ASSESSMENT_TEST からそのレコードを削除します。

例

```
grdapi delete_asssessment_test assessmentDescription=Assess1
```

list_assessments

この GuardAPI コマンドは、セキュリティー・アセスメントをリストするために使用します。

パラメーター	値のタイプ	記述
assessmentDescription	文字列	必須。フリー・テキスト (固有)。同じ記述のアセスメントが前に存在していないことを確認する必要があります。既に存在している場合はエラーになります

例

```
grdapi list_assessments
```

list_assessment_tests

この GuardAPI コマンドは、セキュリティー・アセスメントのテストのリストを表示するために使用します。

list_available_tests の出力は次の形式になります。TEST=[<test description>], DS_TYPE=[<datasource type>] (実際の値は大括弧内にカプセル化されます)

list_assessment_tests の出力は次の形式になります。TEST_DESC=[<available test description>], DS_TYPE=[<datasourcetype>]

list_assessment_tests API コマンドのパラメーターは必須ではなく、フィルタリングをサポートします。

パラメーター	値のタイプ	検証内容
assessmentDescription		この API は以下を行います。 • その記述が唯一の有効なアセスメントの記述であることを確認し、アセスメントの ID を取得します。(アセスメントがない場合、エラーになります。) • アセスメントのテスト (およびデータ・ソース・タイプ) のリストを表示します。 Select AVAILABLE_TEST.TEST_DESC, DATASOURCE_TYPE.NAME from ASSESSMENT_TEST, DATASOURCE_TYPE, AVAILABLE_TEST, SECURITY_ASSESSMENT where AVAILABLE_TEST.DATASOURCE_TYPE_ID = DATASOURCE_TYPE.DATASOURCE_TYPE_ID and ASSESSMENT_TEST.ASSESSMENT_ID = SECURITY_ASSESSMENT.ASSESSMENT_ID and SECURITY_ASSESSMENT.ASSESSMENT_DESC like “Your Param”

例

```
grdapi list_assessment_tests
```

update_assessment

この GuardAPI コマンドは、セキュリティー・アセスメントのレコードを更新するために使用します。

パラメーター	値のタイプ	記述
assessmentDescription	文字列	SECURITY_ASSESSMENT 内の既存レコードと一致している必要があります。
newAssessmentDescription	文字列	フリー・テキスト - 空の場合、記述を更新しないことを意味し、前のパラメーターの値が使用されます。空でない場合は固有であり、同じ記述のアセスメントが前に存在していないことを確認する必要があります。既に存在している場合はエラーになります。
fromDate	文字列	有効な日付または相対的な日付
toDate	文字列	有効な日付または相対的な日付
filterContentIP	文字列	有効な IP アドレス
filterServerIP	文字列	有効な IP アドレス

アクション: すべてのパラメーターが検証され (さらに、指定された記述を含む SECURITY_ASSESSMENT レコードが特定され) たら、指定された値によってそのレコードを更新します。

例

```
grdapi update_assessment assessmentDescription=Assess1 filterClientIP=192.168.1.1.
```

親トピック: GuardAPI

GuardAPI オートディスカバリー関数

以下の CLI コマンドは、オートディスカバリー関数を作成、変更、リスト、および実行するために使用します。

add_autodetect_task

このコマンドは、指定されたプロセスにタスクを追加します。

パラメーター	値のタイプ	記述
process_name	文字列	必須。プロセスの名前
hosts_list	文字列	必須。ホストのリスト。IP または IP 範囲とワイルドカードのスペース区切りリスト (192.168.0.1 192.168.1.* など)。
ports_list	文字列	必須。ポートのリスト。ポートまたはポート範囲のコンマ区切りリスト (22,23,1400-1600 など)。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi add_autodetect_task process_name=myProcess hosts_list="192.168.1.1 192.168.1.3" ports_list="22,23"
```

create_autodetect_process

このコマンドは自動検出プロセスを作成します。

パラメーター	値のタイプ	記述
check_ICMP_echo		必須。nmap に対する PE パラメーター (*)。値は「true」または「false」
host_timeout	文字列	必須。nmap に対するパラメーター (*)。タイムアウト値。
process_name	文字列	必須。プロセスの名前
run_probe_after_scan		必須。値は「true」または「false」。
use_dns		必須。nmap に対するパラメーター[1]。 値は常に「R」または「true」であり、「n」または「false」はあり得ません。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

注: * nmap オプションは、API からのみアクセス可能であり、GUI からはアクセスできません。nmap パラメーターについて、およびスキャンのパフォーマンスへのそれらの影響について詳しくは、man nmap を参照してください。

例

```
grdapi create_autodetect_process process_name=myProcess
```

modify_autodetect_process

このコマンドは自動検出プロセスを変更します。

パラメーター	値のタイプ	記述
check_ICMP_echo		必須。nmap に対する PE パラメーター (*)。値は「true」または「false」
host_timeout	文字列	必須。nmap に対するパラメーター (*)。タイムアウト値。
process_name	文字列	必須。プロセスの名前
run_probe_after_scan		必須。値は「true」または「false」。
use_dns		必須。nmap に対するパラメーター[1]。 値は常に「R」または「true」であり、「n」または「false」はあり得ません。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

注: * nmap オプションは、API からのみアクセス可能であり、GUI からはアクセスできません。nmap パラメーターについて、およびスキャンのパフォーマンスへのそれらの影響について詳しくは、man nmap を参照してください。

例

```
grdapi modify_autodetect_process process_name=myProcess
```

delete_autodetect_scans_for_process

このコマンドは、プロセスのすべてのタスクを削除しますが、プロセスが実行中またはスケジュールされている場合、またはプロセスに結果がある場合、このコマンドは実行できません。

パラメーター	値のタイプ	記述
process_name	文字列	必須。プロセスの名前
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_autodetect_scans_for_process process_name=myProcess
```

list_autodetect_processes

このコマンドはすべてのプロセスをリストします。

パラメーター	値のタイプ	記述

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi list_autodetect_processes
```

list_autodetect_tasks_for_process

このコマンドは指定されたプロセスのすべてのタスクをリストします。

パラメーター	値のタイプ	記述
process_name	文字列	必須。プロセスの名前
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi list_autodetect_tasks_for_process process_name=myProcess
```

execute_autodetect_process

このコマンドは、指定されたプロセスを実行しますが、プロセスに何もタスクが定義されていない場合またはプロセスが現在実行中である場合、このコマンドは実行できません。

パラメーター	値のタイプ	記述
process_name	文字列	必須。プロセスの名前

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi execute_autodetect_process process_name=myProcess
```

show_autodetect_process_status

このコマンドは、プロセスの状況および進行状況サマリーを表示します。

パラメーター	値のタイプ	記述
process_name	文字列	必須。プロセスの名前
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi show_autodetect_process_status process_name=myProcess
```

stop_autodetect_process

このコマンドは。特定のプロセスの実行を停止します。

パラメーター	値のタイプ	記述
process_name	文字列	必須。プロセスの名前
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi stop_autodetect_process process_name=myProcess
```

GuardAPI Big Data Intelligence 関数

中央マネージャーで以下のコマンドを実行して、ビッグデータのデータ・ソースへのデータマートの抽出を管理したり、モニターやレポートなどのためにビッグデータを Guardium にプルしたりします。

注: v10.5 で、GuardAPI Big Data Intelligence 関数が導入されました。

enable_big_data_interface

このコマンドは、インターフェースのアクティブ・プロファイルとスケジュールの指定、ユニット・タイプに基づくプロファイル内のすべてのデータマート抽出のアクティブ化、およびデータ・ソースとしてのビッグデータ収集の定義を行います。このコマンドは、CM またはスタンドアロン・ユニットで実行できます。各 Guardium システムに定義できるインターフェースは 1 つのみです。

パラメーター	値のタイプ	記述
ds_desc	文字列	オプション。データ・ソースの記述。デフォルトは Big Data Intelligence です。
ds_host	文字列	必須。ビッグデータのストレージ場所のホスト名。ここから、レポートやモニターなどのためにデータが Guardium にプルされます。
ds_password	文字列	必須。データ・ソースからビッグデータをプルする ds_user のパスワード。
ds_port	integer	オプション。Guardium がデータ・ソースからビッグデータをプルするときに使用するポート。デフォルトは 27117 です。
ds_user		必須。データ・ソースからビッグデータをプルするユーザー。このユーザーには、ターゲット・ユーザーとは別のユーザーを指定できます。
profile_name	値リスト	必須。add_dm_to_profile または clone_extraction_profile で定義される、データマート抽出プロファイルの名前。プロファイルのリストを取得するには、grdapi get_extraction_profile_info を実行します。
start_date		オプション。ビッグデータ・データ・ソースへのデータの送信を開始するタイミング。形式は、`NOW -<n> <minute | hour | day | week | month>`; または `yyyy-mm-dd hh:mm:ss` です。デフォルトは、コマンドが実行される時刻です。
target_host	文字列	オプション。抽出されるデータマートのターゲット・データ・ホスト。通常、これは ds_host と同じです。データをホストに移動する前にステージング域が必要な場合は、別の target_host および ds_host を指定します。
target_password	文字列	オプション。target_user のパスワード。空白のままにすると、デフォルトの ds_password が使用されます。
target_path	絶対パス	オプション。抽出されるデータマートのターゲット・フォルダーのロケーション。デフォルトは /local/raid0/sonargd/incoming です。
target_port	integer	オプション。ターゲット・サーバー上のポート。デフォルトは 22 (SCP) です。
target_user	文字列	オプション。target_host のユーザー。空白のままにすると、デフォルトの ds_user が使用されます。
unit_group		オプション。CM またはグループの MU からのデータ・エクスポートを有効にします。デフォルトは ALL です。

add_dm_to_profile

プロファイルとは、まとめてアクティブ化される DM のグループです。Guardium には、変更不可の事前定義プロファイルが複数用意されています。プロファイルのコピーを作成して、そのコピーに変更を加えることができます (または最初からプロファイルを作成します)。

パラメーター	値のタイプ	記述
cron_string	文字列	オプション。データマートのスケジュール設定。デフォルトを使用する場合、このパラメーターは省略します。デフォルトでは Guardium の自動プロセスのすべてのスケジュールが考慮されるため、デフォルトの使用を推奨します。ユーザー定義の DM を追加する場合は、デフォルトの cron_string はありません。
category	文字列	オプション。情報提供用です。
datamart_name	値リスト	必須。「エクスポート」で始まる、事前定義またはユーザー定義のデータマートの名前。
profile_name	値リスト	必須。プロファイル名が Guardium でまだ定義されていない場合は、このコマンドを発行すると作成されます。プロファイルのリストを取得するには、grdapi get_extraction_profile_info を実行します。
unit_type	値リスト	オプション。データマート抽出にデータを含むアプライアンスのタイプ。事前定義データマートの場合は、このパラメーターに値を入力しないでください。事前定義データマートにはそれぞれユニット・タイプが割り当てられています。 ユーザー定義の DM を追加する場合は、デフォルトの unit_type はありません。有効な値は、ANY、CM、CM/STANDALONE、AGGREGATOR、COLLECTOR、STANDALONE です。

パラメーター	値のタイプ	記述
api_target_host	ホスト名またはホスト IP	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

remove_dm_from_profile

プロファイルから DM を削除します。プロファイルがアクティブの場合は、このコマンドによって DM のスケジュール解除も行われます。

パラメーター	値のタイプ	記述
profile_name	値リスト	必須。DM を削除するプロファイル。プロファイルのリストを取得するには、grdapi get_extraction_profile_info を実行します。
datamart_name	値リスト	必須。プロファイルから削除する DM。プロファイルに属するデータマート。
api_target_host	ホスト名またはホスト IP	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

local_enable_big_data_interface

プロファイルがアクティブ化されたときにオフラインであったか MU グループに含まれていなかったコレクター、あるいは MU グループに含まれていないがデータが必要なコレクターに対して、このコマンドをローカルに実行します。

これは上級者専用です。このコマンドは、異なるプロファイルを持つコレクターに対して実行できます。例えば、VA のみを実行するコレクターから VA 結果を抽出する場合などです。メイン・プロファイルのサブセットであるプロファイルを作成し、そのプロファイルを指定されたユニットでのみ実行する必要があります。ただし、ローカル・プロファイルにはデータのターゲットがないため、datamart_update_copy_file_info コマンドを使用してこれを追加する必要があります。

パラメーター	値のタイプ	記述
profile_name	値リスト	必須。既存のプロファイル。プロファイルのリストを取得するには、grdapi get_extraction_profile_info を実行します。
start_date		オプション。ビッグデータ・データ・ソースへのデータの送信を開始するタイミング。デフォルトは、コマンドが実行される時刻です。形式は、`NOW -<n> <minute | hour | day | week | month;` または `yyyy-mm-dd hh:mm:ss` です。

disable_big_data_interface

アクティブ・プロファイルを非アクティブ化して、そのスケジュールを削除します。CM およびスタンドアロン・システムでのみ有効です。

パラメーター	値のタイプ	記述
disable_readback	true、false	false: デフォルト。アクティブ・プロファイルを非アクティブ化して、そのスケジュールを削除しますが、データ・ソースおよびレポート・メタデータは削除しません。 true: データ・ソースとそのスケジュールを削除し、そのメタデータとレポートを非表示にします。

get_extraction_profile_info

すべてのプロファイルとそのスケジュール、および各プロファイルに含まれるデータマートをリストします。

パラメーター	値のタイプ	記述
profile_name	値リスト	オプション。指定すると検索フィルターとして使用され、指定された文字列を含むすべてのプロファイル名が返されます。

パラメーター	値のタイプ	記述
verbose	true、false	false: プロファイル名をリストし、プロファイルがアクティブかどうかを示します。 true: 各プロファイルのデータマートとスケジュールをリストします。

replace_active_profile

インターフェースが有効な場合にのみ、インターフェースのアクティブ・プロファイルを変更します。

パラメーター	値のタイプ	記述
new_active_profile	既存プロファイルの名前	必須。現在のプロファイルを置き換える既存のプロファイル。プロファイルのリストを取得するには、grdapi get_extraction_profile_info を実行します。

clone_extraction_profile

プロファイルのコピーを作成します。

パラメーター	値のタイプ	記述
profile_name	既存プロファイルの名前	必須。コピーを作成するプロファイル。プロファイルのリストを取得するには、grdapi get_extraction_profile_info を実行します。
clone_profile_name	文字列	必須。新規プロファイルの名前。

remove_extraction_profile

非アクティブなユーザー定義プロファイルを削除します。

パラメーター	値のタイプ	記述
profile_name	文字列。既存のユーザー定義プロファイルの名前	必須。削除する非アクティブ・プロファイル。プロファイルのリストを取得するには、grdapi get_extraction_profile_info を実行します。

親トピック: GuardAPI

GuardAPI カタログ・エントリー関数

これらの GuardAPI コマンドは、カタログ・エントリー関数の作成、リスト、削除、および更新に使用します。

create_entry_location

新しいアーカイブ項目を内部カタログ・ロケーション表に追加します。

パラメーター	値のタイプ	記述
entryType	文字列	必須。次のいずれかでなければなりません。 • CollectorDataArchive • AggDataArchive • AggResultArchive
processDesc	文字列	entryType が AggResultArchive である場合のみ使用され、必須となります。
fileName	文字列	必須。ファイルを指定します。
hostName	文字列	必須。ホストを識別します。
path	文字列	必須。FTP の場合、FTP アカウントのホーム・ディレクトリーを基準にした相対パスでディレクトリーを指定します。SCP の場合、絶対パスとしてディレクトリーを指定します。
user	文字列	必須。ホストにアクセスするユーザー・アカウント。
password	文字列	必須。ユーザーのパスワード。
retention	integer	オプション。このエントリーをカタログに保持する日数 (デフォルトは 365)。
storageSystem	文字列	必須。EMC、CENTERA、FTP、SCP、TSM のいずれかでなければなりません。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi  create_entry_location entryType=CollectorDataArchive  fileName=733392-a1.corp.com-w20071223.133546-d2007-12-27.dbdump.enc
password=somePassword user=someUser path=/var/dump/ hostName=192.168.1.241  storageSystem=scp
```

list_entry_location

fileName を指定した場合、1 つのアーカイブ・ロケーションがリストされます。fileName を省略した場合、複数のアーカイブ・ロケーションがリストされます。

パラメーター	値タイプ	記述
fileName	文字列	オプション。リストする単一のファイル・ロケーションを指定します。 省略した場合、指定した hostName および path にあるすべてのファイル・ロケーションがリストされます。
hostName	文字列	必須。ホストを識別します。
path	文字列	必須。FTP の場合、FTP アカウントのホーム・ディレクトリーを基準にした相対パスでディレクトリーを指定します。SCP の場合、絶対パスとしてディレクトリーを指定します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi list_entry_location   path=/mnt/nfs/ogazit/archive_results/ hostName=192.168.1.33
```

delete_entry_location

fileName を指定した場合、1 つのアーカイブ・ロケーションが削除されます。fileName を省略した場合、複数のアーカイブ・ロケーションが削除されます。

パラメーター	値のタイプ	記述
fileName	文字列	オプション。削除する単一のファイル・ロケーションを指定します。 省略した場合、指定した hostName および path にあるすべてのファイル・ロケーションが削除されます。
hostName	文字列	必須。ホストを識別します。
path	文字列	必須。FTP の場合、FTP アカウントのホーム・ディレクトリーを基準にした相対パスでディレクトリーを指定します。SCP の場合、絶対パスとしてディレクトリーを指定します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_entry_location  path=/var/dump/mojgan hostName=192.168.1.18
```

update_entry_location

fileName を指定した場合、1 つのアーカイブ・ロケーションが更新されます。 fileName を省略した場合、複数のアーカイブ・ロケーションが更新されます。

パラメーター	値のタイプ	記述
fileName	文字列	オプション。更新する単一のファイル・ロケーションを指定します。 省略した場合、指定した hostName および path にあるすべてのファイル・ロケーションが更新されます。
hostName	文字列	必須。ホストを識別します。
path	文字列	必須。FTP の場合、FTP アカウントのホーム・ディレクトリーを基準にした相対パスでディレクトリーを指定します。SCP の場合、絶対パスとしてディレクトリーを指定します。
newHostName	文字列	オプション。使用する場合、新しいホスト名を指定します。
newPath	文字列	オプション。使用する場合、新しいパスを指定します。
user	文字列	必須。ホストにアクセスするユーザー・アカウント。
password	文字列	必須。ユーザーのパスワード。
retention	integer	オプション。このエントリーをカタログに保持する日数 (デフォルトは 365)。

パラメーター	値のタイプ	記述
storageSystem	文字列	オプション。EMC、CENTERA、FTP、SCP、TSM のいずれかを使用します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi update_entry_location  fileName=a1.corp.com-1_4_2008-01-10_10:27:24.res.70.tar.gz.enc path=/mnt/nfs/ogazit/archive_results/
hostName=qaserver     storageSystem=SCP newPath=/var/dump/mojgan  newHostName=192.168.1.18
```

親トピック: GuardAPI

GuardAPI 分類関数

次の GuardAPI コマンドを使用して、分類ポリシー構成、テスト自動化、および前提条件データの準備のスクリプト記述を行います。

GuardAPI コマンドの使用方法については、『GuardAPI リファレンスの概要』ヘルプ・トピックを参照してください。

create_classifier_action

パラメーター	値のタイプ	記述
actionName	文字列	必須。文字列
actualMemberContent	文字列	必須。文字列

パラメーター	値のタイプ	記述
actionType	文字列	必須。文字列 参照用に、関連付けられている必須パラメーターを持つアクション・タイプのリストを以下に示します。 必須パラメーターは、選択するアクション・タイプによって異なります。 add_to_group_objects actionName - 文字列 - 必須 actualMemberContent - 文字列 - 必須 objectGroup - 文字列 - 必須 policyName - 文字列 - 必須 ruleName - 文字列 - 必須 add_to_group_object_fields actionName - 文字列 - 必須 objectFieldGroup - 文字列 - 必須 policyName - 文字列 - 必須 ruleName - 文字列 - 必須 create_access_rule accessPolicy - 文字列 - 必須 accessRuleAction - 文字列 - 必須 actionName - 文字列 - 必須 ruleName - 文字列 - 必須 create_privacy_set actionName - 文字列 - 必須 policyName - 文字列 - 必須 privacySet - 文字列 - 必須 ruleName - 文字列 - 必須 log_policy_violation actionName - 文字列 - 必須 policyName - 文字列 - 必須 ruleName - 文字列 - 必須 action_send_alert actionName - 文字列 - 必須 policyName - 文字列 - 必須 receiver - 文字列 - 必須 ruleName - 文字列 - 必須
description	文字列	
objectGroup	文字列	必須。
policyName	文字列	必須。
ruleName	文字列	必須。
replaceGroupContent	ブール値	
objectFieldGroup	文字列	必須。
accessPolicy	文字列	必須。
accessPolicy	文字列	必須。
accessRuleAction	文字列	必須。
commandsGroup	文字列	
includeField	ブール値	
includeServerIP	ブール値	
receiver	文字列	

パラメーター	値のタイプ	記述
privacySet	文字列	必須。
severity	文字列	
notificationType	文字列	
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi create_classifier_action actionType=add_to_group_objects policyName=-policy1 ruleName=-rule1 actionName=-action1
description=desc
objectGroup="DW All Objects" replaceGroupContent=1 actualMemberContent=%FULLLIKE

grdapi create_classifier_action actionType=add_to_group_object_fields policyName=-policy1 ruleName=-rule1 actionName=-action1
description=desc objectFieldGroup="DW All Object-Field" replaceGroupContent=1

grdapi create_classifier_action actionType=create_access_rule policyName=-policy1 ruleName=-rule1 actionName=-action1
description=desc
accessPolicy=pci accessRuleAction="alert daily" commandsGroup="Select command" includeField=1 includeServerIP=0
receiver="syslog,snmp,mail=admin,mail=a b c,custm=-b"

grdapi create_classifier_action actionType=create_privacy_set policyName=-policy1 ruleName=-rule1 actionName=-action1
description=desc
privacySet=-b

grdapi create_classifier_action actionType=log_policy_violation policyName=-policy1 ruleName=-rule1 actionName=-action1
description=desc
severity=MED

grdapi create_classifier_action actionType=send_alert policyName=-policy1 ruleName=-rule1 actionName=-action1 description=desc
notificationType=High receiver="syslog,snmp,mail=admin,mail=a b c,custm=-b"
```

GuardAPI コマンドの値

GUI で使用されるコマンド grdapi create_classifer_action の GuardAPI コマンド値のリストについては、表を参照してください。 これらの値は、グループを作成するときに使用します。

表 1. GrdAPI create_classifer_action

GUI 値	GrdAPI 値
%/%.Name	%/NAME
%/Full	%/FULL
Change/%.Name	CHANGE/NAME
Change/Full	CHANGE/FULL
完全修飾名 (スキーマ.オブジェクト)	FULLNAME
Like %Full	%FULLLIKE
Like %Full%	%FULLLIKE%
Like %Name	%NAMELIKE
Like %Name%	%NAMELIKE%
Like Full%	FULLLIKE%
Like Name%	NAMELIKE%
オブジェクト名のみ	NAMEONLY
Read/%.Name	READ/NAME
Read/Full	READ/FULL

例

```
grdapi create_classifier_action actionName=classgrpobjectseach1 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent=NAMEONLY description="object type NAMEONLY"
```

グループ・オブジェクト・タイプの例

```
grdapi create_group appid=Classifier type=OBJECTS desc="Classifier Group of Each Objects" owner=admin category=classifier
classification=classifier subtype=classifier

grdapi create_datasource type="Oracle (DataDirect)" user=scott password=tiger host="swan.guard.swg.usma.ibm.com"
name="Swan Oracle Object Each" shared=true owner=admin application=Classifier port=1521 serviceName=on8swan0
```

```
grdapi create_classifier_policy policyName="A Group Object Each Type Policy" category="Object Each Process"
classification="Object Each Process"

grdapi create_classifier_rule policyName="A Group Object Each Type Policy" category="Object Each Process"
classification="Object Each Process" ruleName=groupobjects1 ruleType=SEARCH_FOR_DATA dataTypes=TEXT continueOnMatch=1
tableNameLike="EMP_INFORMATION"
columnNameLike="PHONE" tableTypeTable=1

grdapi create_classifier_action actionName=classgrpobjectseach1 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent=NAMEONLY description="object type NAMEONLY"

grdapi create_classifier_action actionName=classgrpobjectseach2 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent=FULLNAME description="object type FULLNAME"

grdapi create_classifier_action actionName=classgrpobjectseach3 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="%NAMELIKE" description="object type %NAMELIKE"

grdapi create_classifier_action actionName=classgrpobjectseach4 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="NAMELIKE%" description="object type NAMELIKE%"

grdapi create_classifier_action actionName=classgrpobjectseach5 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="%NAMELIKE%" description="object type %NAMELIKE%"

grdapi create_classifier_action actionName=classgrpobjectseach6 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="%FULLLIKE" description="object type %FULLLIKE"

grdapi create_classifier_action actionName=classgrpobjectseach7 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="FULLLIKE%" description="object type FULLLIKE%"

grdapi create_classifier_action actionName=classgrpobjectseach8 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="%FULLLIKE%" description="object type %FULLLIKE%"

grdapi create_classifier_action actionName=classgrpobjectseach9 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="Change/Full" description="object type Change/Full"

grdapi create_classifier_action actionName=classgrpobjectseach10 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="CHANGE/NAME" description="object type Change/%.name"

grdapi create_classifier_action actionName=classgrpobjectseach11 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="Read/Full" description="object type Read/Full"

grdapi create_classifier_action actionName=classgrpobjectseach12 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="READ/NAME" description="object type Read/%.name"

grdapi create_classifier_action actionName=classgrpobjectseach13 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="%/Full" description="object type %/Full"

grdapi create_classifier_action actionName=classgrpobjectseach14 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="%/NAME" description="object type %/%.name"

grdapi create_classifier_process policyName="A Group Object Each Type Policy"
processName="A Group Object Each Type Process" datasourceNames="Swan Oracle Object Each"

grdapi create_classifier_action actionName=classgrpobjectseach10 actionType=ADD_TO_GROUP_OBJECTS
policyName="A Group Object Each Type Policy" ruleName=groupobjects1 commandsGroup=groupofobjects
objectGroup="Classifier Group of Each Objects" actualMemberContent="FULLNAME" description="Fully Qualified Name(Schema.Object)
```

create_classifier_policy

パラメーター	値のタイプ	記述
category	文字列	必須。
classification	文字列	必須。
description	文字列	
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi create_classifier_policy policyName=-policy1 classification=class1 description=desc1 category=cat1
```

create_classifier_process

create_classifier_process

注: この GuardAPI を呼び出す前に、分類ポリシーとデータ・ソースを作成してください。

パラメーター	値のタイプ	記述
comprehensive	ブール値	
datasourceNames	文字列	必須。
includeInternalTables	ブール値	この設定は、デフォルトでは使用不可になっています。 includeInternalTables を使用可能にすると、データベース・ソフトウェア・プロバイダーが使用する内部システム・データベースおよびスキーマをスキャンできることを示します。内部システム・データベースおよびスキーマは、機密データを含む可能性が低く、デフォルトではスキャンされません。内部表を組み込む場合は、分類データ・ソース・ユーザーが内部データベースおよびスキーマをスキャンするための十分な特権を持っていることを確認してください。特権が不十分であると、予期しない分類ポリシー・エラーが発生することがあります。 includeInternalTables パラメーターの影響を受けるデータベースおよびスキーマを表示および編集するには、「グループ・ビルダー」を使用して、事前定義の「除外する分類 (Excluded Classification)」グループのいずれかを編集します。
policyName	文字列	必須。
processName	文字列	必須。
sampleSize	integer	
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi create_classifier_process  datasourceNames=sample_cls_0001 policyName=APITEST_Cls_Ply_10001_1
processName=APITEST_Clps_10001_1
```

create_classifier_rule

パラメーター	値のタイプ	記述
policyName	文字列	必須。
ruleName	文字列	必須。
ruleType	文字列	必須。 参照用に、関連付けられている必須パラメーターを持つ有効なルール・タイプのリストを以下に示します。 ルール・タイプに選択する内容に応じて、必須パラメーターが決まります。 catalog_search_add policyName - 文字列 - 必須 ruleName - 文字列 - 必須 search_by_permissions_add policyName - 文字列 - 必須 ruleName - 文字列 - 必須 grantTypes - 文字列 - 必須 search_for_data_add policyName - 文字列 - 必須 ruleName - 文字列 - 必須 search_for_unstructured_data_add policyName - 文字列 - 必須 ruleName - 文字列 - 必須

パラメーター	値のタイプ	記述
category	文字列	
classification	文字列	
continueOnMatch	ブール値	
description	文字列	
columnNameLike	文字列	
fireOnlyWithMarker	文字列	
tableNameLike	文字列	
tableTypeSynonym	ブール値	
tableTypeSystemTable	ブール値	
tableTypeTable	ブール値	
tableTypeView	ブール値	
grantTypes	文字列	
role	文字列	
roleGroup	文字列	
user	文字列	
userGroup	文字列	
withAdminOption	ブール値	
compareToValuesInGroup	文字列	
compareToValuesInSQL	文字列	
dataTypes	文字列	
evaluationName	文字列	
hitPercentage	integer	
maxLength	integer	
minLength	integer	
searchExpression	文字列	
searchLike	文字列	
grantTypes	文字列	
showUniqueValues	True または False	
uniqueValueMask	文字列	
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi create_classifier_rule policyName=-policy1 ruleName=-rule1 category=-cat1 classification=-class1 description=-desc1
ruleType=CATALOG_SEARCH continueOnMatch=1 tableTypeTable=1 tableTypeSynonym=1 tableTypeSystemTable=1 tableTypeView=1
tableNameLike=t1
columnNameLike=c1 fireOnlyWithMarker=m1

grdapi create_classifier_rule policyName=-policy1 ruleName=-rule1 category=-cat1 classification=-class1 description=-desc1
ruleType=SEARCH_BY_PERMISSIONS continueOnMatch=1 tableTypeTable=1 tableTypeSynonym=1 tableTypeSystemTable=1 tableTypeView=1
user=user1
userGroup="suspicious users" role=role1 roleGroup=-role1 withAdminOption=1 grantTypes=CONTROL,DELETE,DROP fireOnlyWithMarker=m1

grdapi create_classifier_rule policyName=-policy1 ruleName=-rule1 category=-cat1 classification=-class1 description=-desc1
ruleType=SEARCH_FOR_DATA continueOnMatch=1 tableTypeSynonym=1 tableNameLike=tl1 dataTypes=DATE,NUMBER,TEXT columnNameLike=cl1
minLength=11 searchLike=sel1 searchExpression=se1 evaluationName=en1 hitPercentage=44 compareToValuesInSQL=cv1sql
compareToValuesInGroup="dw all objects" fireOnlyWithMarker=m1

grdapi create_classifier_rule policyName=-policy1 ruleName=-rule1 category=-cat1 classification=-class1 description=-desc1
ruleType=SEARCH_FOR_UNSTRUCTURED_DATA continueOnMatch=1 searchLike=s1 searchExpression=e1 fireOnlyWithMarker=m1

grdapi create_datasource type="Oracle (DataDirect)" user=scott password=tiger host="swan.guard.swg.usma.ibm.com"
name="Swan Oracle8 all values" shared=true owner=admin application=Classifier port=1521 serviceName=on8swan0
```

```
grdapi create_group appid=Classifier type=OBJECTS desc="AA Classifier ALL Values" owner=admin category=classifier
classification=classifier subtype=classifier

grdapi create_member_to_group_by_desc desc="AA Classifier ALL Values" member=ACCOUNTING

grdapi create_member_to_group_by_desc desc="AA Classifier ALL Values" member=ACCOUNTTING

grdapi create_member_to_group_by_desc desc="AA Classifier ALL Values" member=ACCOUNTTING

grdapi create_member_to_group_by_desc desc="AA Classifier ALL Values" member=AG

grdapi create_classifier_policy policyName="Search ALL DATA SEARCH smoke values" category="ALL" classification="ALL"

grdapi create_classifier_rule policyName="Search ALL DATA SEARCH smoke values" category="ALL" classification=ALL
ruleName=ALL1 ruleType=SEARCH_FOR_DATA dataTypes=TEXT,NUMBER continueOnMatch=1 tableNameLike="DEPT14%" minLength=1 maxLength=100
tableTypeSynonym=1 tableTypeSystemTable=1 tableTypeTable=1 tableTypeView=1 fireOnlyWithMarker=ACCT searchLike="A%"
searchExpression="^AA*" columnNameLike="DNAME" evaluationName="com.guardium.classifier.custom.RichardEvaluation" hitPercentage=10
compareToValuesInGroup="AA Classifier ALL Values" compareToValuesInSQL="select DNAME from SCOTT.DEPT where DNAME like 'A%G'"
showUniqueValues="true" uniqueValueMask="^AA*"

grdapi create_classifier_process policyName="Search ALL DATA SEARCH smoke values"
processName="Search ALL DATA SEARCH smoke values Process" datasourceNames="Swan Oracle8 all values"
```

delete_classifier_action

パラメーター	値のタイプ	記述
actionName	文字列	必須。
policyName	文字列	必須。

例

```
grdapi delete_classifier_action policyName=-policy1 ruleName=-rule1 actionName=-action1
```

delete_classifier_policy

パラメーター	値のタイプ	記述
policyName	文字列	必須。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_classifier_policy policyName=-policy1
```

delete_classifier_process

パラメーター	値のタイプ	記述
processName	文字列	
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_classifier_process processName=APITEST_Clps_10001_1
```

delete_classifier_rule

パラメーター	値のタイプ	記述
policyName	文字列	必須。

パラメーター	値のタイプ	記述
ruleName	文字列	必須。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_classifier_rule policyName=-policy1 ruleName=-rule1
```

execute_cls_process

分類プロセスの実行 (サブミット)

分類プロセスを実行します。 分類プロセス・ビルダーから「今すぐ 1 回実行」を実行することに相当します。 これは、Guardium® ジョブ・キューにプロセスを配置するジョブをサブミットします。このキューからアプライアンスは一度に 1 つのジョブを実行します。 管理者は、「Guardium モニター」>「Guardium ジョブ・キュー」と選択することによりジョブ状況を表示できます。
注: この API を呼び出す前に分類プロセスを作成してください。

パラメーター	値のタイプ	記述
processName	文字列	分類プロセスの名前
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi execute_cls_process processName="classPolicy1"
```

以下は、分類関数およびその各パラメーターのリストです。 パラメーターに有効な項目の設定リストがある場合、このリストが提供されます。

list_classifier_policies

パラメーター	値のタイプ	記述
policyName	文字列	必須。
ruleName	文字列	必須。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi list_classifier_policy policyName=-policy1 ruleName=-rule1 actionName=-action1 recursive=1
```

注: 引数を指定せずにこの関数を実行すると、すべてのポリシーがリストされます。 ポリシーの引数を渡すと、そのポリシーのすべてのルールおよびアクションがリストされます。 ポリシーとルールを渡すと、そのルールのすべてのアクションがリストされます。

list_classifier_process

パラメーター	値のタイプ	記述
processName	文字列	

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi list_classifier_process processName=APITEST_CLPS_30001

get_all_modifiable_guard_params

この汎用の grdapi コマンドは、コマンド modify_guard_param によって変更できるすべてのパラメーターとその値を返します。

パラメーター	値タイプ	記述
paramlike	文字列 任意の数値または文字	指定されたストリングと一致するすべてのパラメーターを返します。これは必須フィールドではありません。指定しない場合は、変更可能なすべてのパラメーターが返されます。
paramdesclike	文字列 任意の数値または文字	指定された説明と一致するすべてのパラメーターを返します。これは必須フィールドではありません。指定しない場合は、変更可能なすべてのパラメーターが返されます。

例:

`grdapi get_all_modifiable_guard_params paramlike=`*`classifier`*

これにより、パラメーター名、パラメーターの記述、パラメーター・タイプ、およびパラメーター値が返されます。

```
Parameter Name : CLASSIFIER_RUNNING_TIMEOUT
Parameter Description : CLS INACTIVE TIMEOUT
Parameter Type : N
Parameter Value: 30
ok
```

get_guard_param

この汎用の grdapi コマンドは、指定されたパラメーターの現行値を返します。

パラメーター	値タイプ	記述
paramName	文字列	パラメーターの名前。 これは必須フィールドではありません。ただし、パラメーター名またはパラメーターの記述のいずれかを指定する必要があります。
paramdesc	文字列	パラメーターの説明 これは必須フィールドではありません。ただし、パラメーター名またはパラメーターの記述のいずれかを指定する必要があります。

構文:

`grdapi get_guard_param paramName=`*`parameter`*

modify_guard_param

この汎用の grdapi コマンドを使用して、変更可能なパラメーターの値を更新します。

変更可能なパラメーター:

パラメーター	値タイプ	記述
paramName	文字列	変更可能なパラメーターの名前。 これは必須フィールドです。
paramValue	整数	変更可能なパラメーターの新規の値 これは必須フィールドです。

構文:

`grdapi modify_guard_param paramName=`*`parameter`* `paramValue=`*`value`*

分類の変更可能なパラメーター	値	記述
classifier_running_timeout	整数 範囲は 5 から 720 分です。 デフォルトは 30 分です。	このパラメーターを変更して、ハウスキーピング・プロセス (Nanny) の制限時間を設定できます。指定された時間が経過すると、Nanny は分類プロセスが非アクティブであると見なし、これを再始動します。

例:

```
grdapi modify_guard_param paramName=classifier_running_timeout paramValue=10
```

set_job_process_concurrency_limit

Guardium は、CPU のパフォーマンスと使用効率を最適化するために、複数のスレッドを並列で実行できます。このマルチスレッド機能を活用するには、set_job_process_concurrency_limit コマンドを使用します。このコマンドは、同時に実行できる評価プロセスと分類プロセスの数を定義します。
構文: grdapi set_job_process_concurrency_limit limit=[value]

パラメーター	値のタイプ	記述
limit	整数: Guardium システムのハードウェア構成に応じて 1 から 100 を指定します。 デフォルト値は 1 です。	limit 値は、同時に実行できるアセスメントおよび分類プロセスの数を定義します。limit 値は、100 よりも小さいか、Guardium システムにインストールされている CPU コアの 2 倍の数になります。 例えば、システムに 8 つの CPU コアがある場合、limit の最大値は 16 です。システムに 64 個の CPU コアがある場合、limit の最大値は 100 です。 limit のデフォルト値は 1 です。

例:

```
grdapi set_job_process_concurrency_limit limit=10
```

値の表示: grdapi get_job_process_concurrency_limit

update_classifier_action

パラメーター	値のタイプ	記述
actionName	文字列	必須。
actualMemberContent	文字列	必須。
description	文字列	
objectGroup	文字列	必須。
policyName	文字列	必須。
ruleName	文字列	必須。文字列
replaceGroupContent	ブール値	
objectFieldGroup	文字列	必須。
accessPolicy	文字列	必須。
accessRuleAction	文字列	必須。
commandsGroup	文字列	
includeField	ブール値	
includeServerIP	ブール値	
receiver	文字列	
privacySet	文字列	必須。
severity	文字列	
notificationType	文字列	
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi update_classifier_action actionType=add_to_group_objects policyName=-policy1 ruleName=-rule1 actionName=-action1
description=desc objectGroup="DW All Objects" replaceGroupContent=1 actualMemberContent=%FULLLIKE

grdapi update_classifier_action actionType=add_to_group_object_fields policyName=-policy1 ruleName=-rule1 actionName=-action1
description=desc objectFieldGroup="DW All Object-Field" replaceGroupContent=1

grdapi update_classifier_action actionType=update_access_rule policyName=-policy1 ruleName=-rule1 actionName=-action1
description=desc accessPolicy=pci accessRuleAction="alert daily" commandsGroup="Select command" includeField=1 includeServerIP=0
receiver="syslog,snmp,mail=admin,mail=a b c,custm=-b"

grdapi update_classifier_action actionType=update_privacy_set policyName=-policy1 ruleName=-rule1 actionName=-action1
description=desc privacySet=-b

grdapi update_classifier_action actionType=log_policy_violation policyName=-policy1 ruleName=-rule1 actionName=-action1
description=desc severity=MED

grdapi update_classifier_action actionType=send_alert policyName=-policy1 ruleName=-rule1 actionName=-action1
description=desc notificationType=High receiver="syslog,snmp,mail=admin,mail=a b c,custm=-b"
```

update_classifier_log_level

パラメーター	値のタイプ	記述
logLevel	文字列	必須 logLevel は、以下のいずれかの値でなければなりません。 • DEBUG • INFO • WARN • FATAL • ERROR 値 DEBUG を指定すると、分類スキャンに関する詳細がログに記録されます。例えば、表名や列名などのスキャン対象データベースからのメタデータはログに記録されますが、データベースからの実際のデータはログに記録されません。 重要: 新しい設定を有効にするには、ログ・レベルの変更後にジョブ・キューを再始動します。ジョブ・キューを再始動するには、restart_job_queue_listener API を使用します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • all_managed: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • all: すべての管理対象ユニットと中央マネージャーで実行します • group:*<group name>*: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば api_target_host=10.0.1.123 です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば api_target_host=10.0.1.123 です。

例

```
grdapi update_classifier_log_level logLevel=INFO
```

update_classifier_policy

パラメーター	値のタイプ	記述
policyName	文字列	必須
category	文字列	必須
classification	文字列	必須
description	文字列	
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • all_managed: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • all: すべての管理対象ユニットと中央マネージャーで実行します • group:*<group name>*: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば api_target_host=10.0.1.123 です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば api_target_host=10.0.1.123 です。

例

```
grdapi update_classifier_policy policyName=-policy1 classification=class1 description=desc1 category=cat1
```

update_classifier_process

update_classifier_process

パラメーター	値のタイプ	記述
comprehensive	ブール値	
datasourceNames	文字列	必須。
includeInternalTables		この設定は、デフォルトでは使用不可になっています。 includeInternalTables を使用可能にすると、データベース・ソフトウェア・プロバイダーが使用する内部システム・データベースおよびスキーマをスキャンできることを示します。内部システム・データベースおよびスキーマは、機密データを含む可能性が低く、デフォルトではスキャンされません。内部表を組み込む場合は、分類データ・ソース・ユーザーが内部データベースおよびスキーマをスキャンするための十分な特権を持っていることを確認してください。特権が不十分であると、予期しない分類ポリシー・エラーが発生することがあります。 includeInternalTables パラメーターの影響を受けるデータベースおよびスキーマを表示および編集するには、「グループ・ビルダー」を使用して、事前定義の「除外する分類 (Excluded Classification)」グループのいずれかを編集します。
newName	文字列	
policyName	文字列	必須
processName	文字列	必須
sampleSize	integer	
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi update_classifier_process datasourceNames=sample_cls_0001,sample_cls_0002 policyName=APITEST_Cls_Ply_10001_1 processName=APITEST_Clps_10001_1 comprehensive=0 sampleSize=3000

update_classifier_rule

パラメーター	値のタイプ	記述
policyName	文字列	必須
ruleName	文字列	必須
ruleType	文字列	必須。値は次のとおりです。 catalog_search search_by_permissions search_for_data search_for_unstructured_data
category	文字列	
classification	文字列	
continueOnMatch	ブール値	
description	文字列	
columnNameLike	文字列	
fireOnlyWithMarker	文字列	
tableNameLike	文字列	
tableTypeSynonym	ブール値	
tableTypeSystemTable	ブール値	
tableTypeTable	ブール値	
tableTypeView	ブール値	
grantTypes	文字列	
role	文字列	
roleGroup	文字列	
user	文字列	
userGroup	文字列	

パラメーター	値のタイプ	記述
withAdminOption	ブール値	
compareToValuesInGroup	文字列	
compareToValuesInSQL	文字列	
dataTypes	文字列	
evaluationName	文字列	
hitPercentage	integer	
maxLength	integer	
minLength	integer	
searchExpression	文字列	
searchLike	文字列	
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi update_classifier_rule policyName=-policy1 ruleName=-rule1 category=-cat1 classification=-class1 description=-desc1
ruleType=CATALOG_SEARCH continueOnMatch=1 tableTypeTable=1 tableTypeSynonym=1 tableTypeSystemTable=1 tableTypeView=1
tableNameLike=t1
columnNameLike=c1 fireOnlyWithMarker=m1

grdapi update_classifier_rule policyName=-policy1 ruleName=-rule1 category=-cat1 classification=-class1 description=-desc1
ruleType=SEARCH_BY_PERMISSIONS continueOnMatch=1 tableTypeTable=1 tableTypeSynonym=1 tableTypeSystemTable=1 tableTypeView=1
user=user1
userGroup="suspicious users" role=role1 roleGroup=-role1 withAdminOption=1 grantTypes=CONTROL,DELETE,DROP fireOnlyWithMarker=m1

grdapi update_classifier_rule policyName=-policy1 ruleName=-rule1 category=-cat1 classification=-class1 description=-desc1
ruleType=SEARCH_FOR_DATA continueOnMatch=1 tableTypeSynonym=1 tableNameLike=tl1 dataTypes=DATE,NUMBER,TEXT columnNameLike=cl1
minLength=11
maxLength=22 searchLike=sel1 searchExpression=se1 evaluationName=en1 hitPercentage=44 compareToValuesInSQL=cv1sql
compareToValuesInGroup="dw all objects" fireOnlyWithMarker=m1

grdapi update_classifier_rule policyName=-policy1 ruleName=-rule1 category=-cat1 classification=-class1 description=-desc1
ruleType=SEARCH_FOR_UNSTRUCTURED_DATA continueOnMatch=1 searchLike=s1 searchExpression=e1 fireOnlyWithMarker=m1
```

親トピック: GuardAPI

GuardAPI クラウド・データ・ソース関数

これらのコマンドは、クラウド・データ・ソースを定義、更新、および削除するために使用します。

注: v10.1.4 で、GuardAPI Cloud Datasource 関数が導入されました。

create_cloud_datasource

パラメーター	値のタイプ	記述
application	文字列。説明を参照	必須。データ・ソースの定義対象のアプリケーション。次のいずれかです。 アクセス・ポリシー アプリケーション・ユーザー・トランスレーション 監査タスク 変更監査システム Classifier カスタム・ドメイン データベース・アナライザー 値のモニター セキュリティー・アセスメント S-TAP 検査
cloudTitle	文字列。説明を参照。	必須。Guardium で既に定義されているクラウド・アカウントの名前
compatibilityMode	文字列	表のモニター時に使用されるモード。

パラメーター	値のタイプ	記述
conProperty	文字列	オプション。 このデータ・ソースとの JDBC 接続を確立するために JDBC URL に追加の接続プロパティーを含める必要がある場合にのみ使用します。 使用するフォーマットは、「property=value」である必要があります。このとき、プロパティーと値の各ペアはコンマで区切ります。
customURL	文字列	オプション。 データ・ソースに対する接続文字列。これを入力しない場合は、前回入力したフィールドのホスト、ポート、インスタンス、プロパティーなどを使用して接続が行われます。これは、例えば Oracle Internet Directory (OID) の接続を作成するときに便利です。
dbInstanceAccount	文字列	オプション。CAS によって使用されるデータベース・アカウント・ログイン名
dbInstanceDirectory	文字列	オプション。CAS によって使用される、データベース・ソフトウェアがインストールされたディレクトリー
dbName	文字列	オプション。Db2® データ・ソースまたは Oracle データ・ソースの場合、スキーマ名を入力します。他の場合は、データベース名を入力します。
description	文字列	オプション。データ・ソースの詳細説明。
host	文字列	必須。ホスト名または IP アドレス。
importServerSSLcert	ブール値	
KerberosConfigName	文字列	オプション。Guardium システムで既に定義されている Kerberos 構成の名前。
name	文字列	必須。Guardium システム内のデータ・ソースに対する固有の名前
objectLimit	0、正の整数	必須。分類プロセスで見つかった機密オブジェクトで、監査対象オブジェクトに自動的に追加される最大数。デフォルトは 20 です。
password	文字列	ユーザーのパスワード。
port	整数	オプション。ポート番号。
primaryCollector	整数	クラウド・データベースから監査データを抽出するコレクター。
region	値リスト	必須。
savePassword	ブール値	Guardium アプライアンス上の認証資格情報を保存して暗号化します。 (オンデマンドではなく) スケジュールされたタスクとして実行されるアプリケーションでデータ・ソースを定義する場合は、必須になります。yes に設定した場合は、ログイン名とパスワードが必須になります。
serviceName	文字列	オプション。Oracle、Informix®、Db2、および IBM® ISeries の場合は必須。Db2 データ・ソースにはデータベース名を入力し、それ以外にはサービス名を入力します。
severity	値リスト	オプション。データ・ソースの重大度分類 (あるいは影響レベル)。以下のいずれか。 低 なし 中 高
shared	値リスト	オプション。他のアプリケーションと共有するには **True** または **Share** に設定します。データ・ソースを他のユーザーと共有する場合は、GUI からロールを割り当てる必要があります。値は次のとおりです。 Share Not Shared True False
type	値リスト	必須。データ・ソース・タイプを識別します。有効な値: Oracle (DataDirect - SID) Oracle (DataDirect - サービス名)
useKerberos	ブール値	オプション (ブール値)。Kerberos 認証を使用する場合は、yes に設定します。yes の場合は、KerberosConfigName を指定する必要があります。
useLDAP	ブール値	オプション (ブール値)。LDAP を使用する場合は、yes に設定します。
user	文字列	オプション。データ・ソースのユーザー。使用する場合、パスワードも使用する必要があります。
useSSL	ブール値	オプション (ブール値)。SSL 認証を使用する場合は、yes に設定します。

list_cloud_datasource_by_name

パラメーター	値のタイプ	記述
name	文字列	必須。Guardium で定義されているクラウド・データ・ソース。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

restart_cloud_instance

指定したクラウド・インスタンスを再開します。

パラメーター	値のタイプ	記述
datasource_name	文字列	必須。Guardium で定義されているクラウド・データ・ソース。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

update_cloud_datasource

クラウド・データ・ソース構成を更新します。

パラメーター	値のタイプ	記述
cloudTitle	値リスト	必須。GRDAPI コマンドで定義されたタイトル
conProperty	文字列	オプション。 このデータ・ソースとの JDBC 接続を確立するために JDBC URL に追加の接続プロパティーを含める必要がある場合にのみ使用します。 使用するフォーマットは、「property=value」である必要があります。このとき、プロパティーと値の各ペアはコンマで区切ります。
customURL	文字列	オプション。 データ・ソースに対する接続文字列。これを入力しない場合は、前回入力したフィールドのホスト、ポート、インスタンス、プロパティーなどを使用して接続が行われます。これは、例えば Oracle Internet Directory (OID) の接続を作成するときに便利です。
dbInstanceAccount	文字列	オプション。CAS によって使用されるデータベース・アカウント・ログイン名
dbInstanceDirectory	文字列	オプション。 CAS によって使用される、データベース・ソフトウェアがインストールされたディレクトリー
dbName	文字列	オプション。Db2® データ・ソースまたは Oracle データ・ソースの場合、スキーマ名を入力します。他の場合は、データベース名を入力します。
description	文字列	オプション。データ・ソースの詳細説明。
host	文字列	必須。ホスト名または IP アドレス。
importServerSSLcert	ブール値	
KerberosConfigName	文字列	Guardium システムで既に定義されている Kerberos 構成の名前。
name	文字列	必須。Guardium システム内のデータ・ソースに対する固有の名前
newName	文字列	オプション。 新規名を指定します。これはシステム上のデータ・ソースで固有でなければなりません。
objectLimit	整数: 0 以上	必須。分類プロセスで見つかった機密オブジェクトで、監査対象オブジェクトに自動的に追加される最大数。
password	文字列	ユーザーのパスワード
port	整数	Guardium で定義されているクラウド・データ・ソース。
primaryCollector	整数	クラウド DB からデータを受信するコレクター
region	値リスト	必須。

パラメーター	値のタイプ	記述
savePassword	ブール値	Guardium アプライアンス上の認証資格情報を保存して暗号化します。(オンデマンドではなく) スケジュールされたタスクとして実行されるアプリケーションでデータ・ソースを定義する場合は、必須になります。yes に設定した場合は、ログイン名とパスワードが必須になります。
serviceName	文字列	オプション。Oracle、Informix®、Db2、および IBM® ISeries の場合は必須。Db2 データ・ソースにはデータベース名を入力し、それ以外にはサービス名を入力します。
severity	値リスト	オプション。データ・ソースの重大度分類 (あるいは影響レベル)。以下のいずれか。 低 なし 中 高
shared	値リスト	オプション。他のアプリケーションと共有するには **True** または **Share** に設定します。データ・ソースを他のユーザーと共有する場合は、GUI からロールを割り当てる必要があります。値は次のとおりです。 Share Not Shared True False
useKerberos	ブール値	オプション (ブール値)。Kerberos 認証を使用する場合は、yes に設定します。yes の場合は、KerberosConfigName を指定する必要があります。
useLDAP	ブール値	オプション (ブール値)。LDAP を使用する場合は、yes に設定します。
user	文字列	オプション。データ・ソースのユーザー。使用する場合、パスワードも使用する必要があります。
useSSL	ブール値	オプション (ブール値)。SSL 認証を使用する場合は、yes に設定します。

親トピック: GuardAPI

GuardAPI データマート関数

これらのコマンドは、データマートを管理するために使用します。

datamart_copy_file_bundle

ファイル・データマートにのみ適用されます。

データマート・バンドルを作成および管理します。

パラメーター	値のタイプ	記述
Action	文字列	必須。以下のいずれか。 • *create*: バンドルを作成します。 • *delete*: バンドルを削除します。 • *include*: バンドルにデータマートを追加します。 • *exclude*: バンドルからデータマートを除去します。 • *info*: バンドルの詳細を返します。
bundle_name	文字列	すべてのアクションに必要です。バンドルの名前。
datamart_name	文字列	include、exclude に必要です。データマートの名前。
main_datamart_name	文字列	バンドルを作成するときに必要です。

datamart_include_file_header

ファイル・データマートにのみ適用されます。

ヘッダー行 (列名) を出力 CSV に含めるかどうかを決定します。

パラメーター	値のタイプ	記述
includeFileHeader	文字列	必須。値: Yes、No
Name	文字列	必須。データマート名

datamart_refresh_metadata

テーブルおよびファイル・データマートに適用されます。

データマートのメタデータは CM に保存され、ユーザー同期プロセスによって各 MU に伝搬されます。このプロセスは、デフォルトでは 30 分ごとに実行されます。データマート定義を指定または変更した場合に、ユーザー同期がまだ実行されていなくても、次回の抽出でその定義が反映されるようにしたい場合は、このユニットに対して

datamart_refresh_metadata grdapi を実行します。

パラメーター	値のタイプ	記述
unit_hostname	文字列	必須

datamart_run_once_now

テーブルおよびファイル・データマートに適用されます。

指定されたデータマートを 1 回実行します (コマンドの実行時に開始)。

パラメーター	値のタイプ	記述
datamart_name	文字列	必須

datamart_set_active

テーブルおよびファイル・データマートに適用されます。

指定されたデータマートの抽出をアクティブ化します。

パラメーター	値のタイプ	記述
Name	文字列	必須。データマート名。

datamart_set_date_format

ファイル・データマートにのみ適用されます。Guardium のデフォルトの日付形式がご使用の Guardium 以外のシステムに適していない場合は、この API を使用して、デフォルトの日付形式を必要な形式に変更します。 それ以外の場合は、日付形式を変更しないでください。

現在の日付形式を調べるには、get_datamart_info datamart_name を使用します。

構文

```
grdapi datamart_set_date_format datamart_name<datamart name> old_date_format=<current date format> new_date_format=<new date format>
```

パラメーター	値のタイプ	記述
datamart_name	文字列	必須
new_date_format	文字列	必須
old_date_format	文字列	必須

例

```
grdapi datamart_set_date_format datamart_name="Export:Full-SQL" old_date_format="%Y-%m-%dT%T" new_date_format="%Y-%m-%dT%TZ"
```

datamart_set_inactive

テーブルおよびファイル・データマートに適用されます。

指定されたデータマートの抽出を非アクティブ化します。

パラメーター	値のタイプ	記述
Name	文字列	必須。データマート名

datamart_update_copy_file_info

ファイル・データマートにのみ適用されます。

エクスポート抽出のターゲット・ホストを定義します。

パラメーター	値のタイプ	記述
destinationHost	文字列	必須。ターゲット・サーバーのホスト名
destinationPassword	文字列	必須。destinationUser で指定されたユーザーのパスワード。
destinationPath	文字列	必須。データマート抽出の保管場所のパス。
destinationUser	文字列	必須。宛先パスへの読み取り権限を持つユーザー。
Name	文字列	必須。データマート名
transferMethod	文字列	必須。 • SCP • FTP

パラメーター	値のタイプ	記述
validate	ブール値	• 0 (false) • 1 (true)
withCOMPLETEfile	ブール値	オプション。COMPLETE ファイルは、データ・ファイルが正常に転送された後に送信されます。有効な値: • 0 (false)。マーカー・ファイルは送信されず、データ・ファイルに「DMv2_」という接頭部が付けられます。例: DMv2_vmappibm_EXP_SYSTEM_INFO_20170508150000.gz • 1 (true)。デフォルト。2 つのファイルが送信されます。1 つはデータ・ファイル、もう 1 つは COMPLETE という語を含む空のファイルです。2 番目のファイルは、結果が完了したことを示すマーカーです。ファイル名は、<グローバルな接頭部 ID>_<アプライアンス名>_<データマート名>_<タイム・スタンプ>.gz で構成されます (例: 1234567890123456789_vmappibm_EXP_SYSTEM_INFO_20170508150000.gz2)。

datamart_validate_copy_file_info

ファイル・データマートにのみ適用されます。

ターゲット・ホストへの接続を検証します。

パラメーター	値のタイプ	記述
destinationHost	文字列	必須。ターゲット・サーバーのホスト名
destinationPassword	文字列	必須。destinationUser で指定されたユーザーのパスワード。
destinationPath	文字列	必須。データマート抽出の保管場所のパス。
destinationUser	文字列	必須。宛先パスへの読み取り権限を持つユーザー。
Name	文字列	必須。データマート名
transferMethod	文字列	必須。 • SCP • FTP

get_datamart_info

テーブルおよびファイル・データマートに適用されます。

データマートに関する詳細 (データマートの基盤となっているレポートおよび照会、作成日、抽出ターゲット、初始動、日付形式、時間間隔など) を返します。

パラメーター	値のタイプ	記述
datamart_name	文字列	必須。データマートの名前。
isExtended	integer	0: 標準の詳細、1: 追加詳細

unschedule_datamart

テーブルおよびファイル・データマートに適用されます。

指定されたデータマートの抽出を停止します。

パラメーター	値のタイプ	記述
datamart_name	文字列	必須。データマート名。
api_target_host	文字列	データマートをスケジュール解除するターゲット Guardium システム。

update_datamart

ファイル・データマートにのみ適用されます。

パラメーター	値のタイプ	記述
comment	文字列	ユーザーが追加する更新に関するコメント。
initial_start	date	• < >: 現在時刻 • YYYY-MM-D hh:mm:ss
name	文字列	データマート名。必須

親トピック: GuardAPI

GuardAPI データベース・ユーザー関数

これらの GuardAPI コマンドは、データベース・ユーザー・マッピングの保守、非資格情報スキャン、およびデバッグ・レベルの設定に使用します。

non_credential_scan

usersGroup に属する使用可能なデフォルト・ユーザーを見つけるために serversGroup 内のデータベースをスキャンするジョブを実行依頼できるようにする API。 実行依頼されたジョブは分類リスナーの下で実行され、分類/アセスメントのジョブ・キュー・レポートを使用してトラッキングできます。 実行依頼されたジョブをキャンセルする場合、分類/アセスメントのジョブ・キュー・レポートでジョブをダブルクリックし、「ジョブの停止」を選択します。

注: serversGroup 内のサーバーに到達できない場合、「スケジュールされたジョブの例外」タイプの例外が追加され、サーバーはスキャンされません。

パラメーター	値のタイプ	記述
databaseType	値リスト	必須。ORACLE、DB2®、SYBASE、MS SQL SERVER、MYSQL、TERADATA、POSTGRESQL、NETEZZA、IBM ISERIES、INFORMIX のいずれかでなければなりません。
serversGroup	値リスト	必須。グループ・ビルダーで定義された、有効なサーバーのグループ (サーバー IP/インスタンス名/ポート) でなければなりません。
usersGroup	値リスト	必須。 グループ・ビルダーで定義された、有効なユーザーのグループ (データベース・ユーザー/データベース・パスワード) でなければなりません。グループ・ビルダーには、デフォルト・グループがあります。

例

```
grdapi non_credential_scan databaseType=ORACLE serversGroup=oracleServers usersGroup="ORACLE Default Users"
```

データベース・マッピングを保守するための API

これらの API は、データベース・ユーザー (違反の原因となった SQL の起動者) とリアルタイム・アラート用 E メール・アドレス間のマッピングを保守するのに役立ちます。 起動者についての詳細は『アラート・アクション』を参照してください。

- create_db_user_mapping
- delete_db_user_mapping
- list_db_user_mapping

create_db_user_mapping

ワイルドカードの使用:

- 「delete」および「list」コマンドでは、4 つのすべてのパラメーターでワイルドカード (「%」) を使用できます。
- 「create」コマンドの場合には、次のようになります。
  - serverIp - ワイルドカードは有効です。IP アドレス・フォーマットの数値の代わりに「%」を指定できます
  - 192.168.2.% - 有効
  - 192.%.2.% - 有効
  - 192.% - 無効
- serviceName - ワイルドカード (%) を使用できます
- dbUserName - ワイルドカードは使用できません。「%」は有効ですが、記号「%」であると見なされます
- emailAddress - ワイルドカードは使用できません。「%」は有効ですが、記号「%」であると見なされます

パラメーター	値のタイプ	記述
serverIp	文字列 (IP アドレス)	必須。形式: IP アドレス (A.B.C.D)
serviceName	文字列	必須。サービス名を識別します。
dbUserName	文字列	必須 (任意の文字列)。データベース・ユーザー名を識別します。
emailAddress	文字列	必須 (任意の文字列で、「@」記号が必要)。E メール・アドレスを識別します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi create_db_user_mapping serverIp=192.168.1.104 serviceName=ora1 dbUserName=scott emailAddress=scott@oracle.com
```

delete_db_user_mapping

ワイルドカードの使用:

- 「delete」および「list」コマンドでは、4 つのすべてのパラメーターでワイルドカード (「%」) を使用できます。
- 「create」コマンドの場合には、次のようになります。
  - serverIp - ワイルドカードは有効です。IP アドレス・フォーマットの数値の代わりに「%」を指定できます
  - 192.168.2.% - 有効

    - 192.%.2.% - 有効
    - 192.% - 無効
- serviceName - ワイルドカード (%) を使用できます
- dbUserName - ワイルドカードは使用できません。「%」は有効ですが、記号「%」であると見なされます
- emailAddress - ワイルドカードは使用できません。「%」は有効ですが、記号「%」であると見なされます

パラメーター	値のタイプ	記述
serverIp	文字列 (IP アドレス)	必須。形式: IP アドレス (A.B.C.D)
serviceName	文字列	必須。サービス名を識別します。
dbUserName	文字列	必須。データベース・ユーザー名を識別します。
emailAddress	文字列	必須 (任意の文字列で、「@」記号が必要)。E メール・アドレスを識別します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi create_db_user_mapping serverIp=192.168.1.104 serviceName=ora1 dbUserName=scott emailAddress=scott@oracle.com
```

list_db_user_mapping

ワイルドカードの使用:

- 「delete」および「list」コマンドでは、4 つのすべてのパラメーターでワイルドカード (「%」) を使用できます。
- 「create」コマンドの場合には、次のようになります。
    - serverIp - ワイルドカードは有効です。IP アドレス・フォーマットの数値の代わりに「%」を指定できます
    - 192.168.2.% - 有効
    - 192.%.2.% - 有効
    - 192.% - 無効
- serviceName - ワイルドカード (%) を使用できます
- dbUserName - ワイルドカードは使用できません。「%」は有効ですが、記号「%」であると見なされます
- emailAddress - ワイルドカードは使用できません。「%」は有効ですが、記号「%」であると見なされます

パラメーター	値のタイプ	記述
serverIp	文字列 (IP アドレス)	必須。形式: IP アドレス (A.B.C.D)
serviceName	文字列	必須。サービス名を識別します。
dbUserName	文字列	必須 (任意の文字列)。データベース・ユーザー名を識別します。
emailAddress	文字列	必須 (任意の文字列で、「@」記号が必要)。E メール・アドレスを識別します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi create_db_user_mapping serverIp=192.168.1.104 serviceName=ora1 dbUserName=scott emailAddress=scott@oracle.com
```

デバッグ・レベルの取得

この GuardAPI コマンドは、IMS 出力のデバッグ・レベルを表示するために使用します。

set_debug_level

この GuardAPI コマンドは、IMS 出力を制御するために使用します。

IMS debug_level = 1 の場合、IMS デバッグ・フィールド (mvs_is_plex、mvs_ipaddr、mvs_dlta_sign、mvs_dlta_val など) が内部データベース表 (GDM_CONSTRUCT_TEXT.FULL_SQL または GDM_EXCEPTION.FULL_SQL) に出力されます。

IMS デバッグ・レベルが 0 の場合、IMS デバッグ・フィールドは配布されません。

親トピック: GuardAPI

GuardAPI データ・ソース関数

これらの GuardAPI コマンドは、データ・ソース関数の作成、リスト、削除、および更新に使用します。

create_datasource

このコマンドは、新規データ・ソースを定義するために使用します。

注: 中央マネージャー環境では、データ・ソースは中央マネージャー上で定義します。 GuardAPI を使用して管理対象ユニットにデータ・ソースを作成することはできますが、それらデータ・ソースを表示または使用することはできません。

クラウド・データ・ソースを作成するには、GuardAPI クラウド・データ・ソース関数を参照してください。

パラメーター	値のタイプ	記述
application	値リスト	必須。 データ・ソースの定義対象となるアプリケーションを指定します。 次のいずれかでなければなりません。 Access_policy アプリケーション・ユーザー・トランスレーション AuditDatabase AuditTask Big Data Intelligence ChangeAuditSystem Classifier CustomDomain DatabaseAnalyzer MonitorValues SecurityAssessment Stap_Verification
compatibilityMode		互換モード: 選択項目は「デフォルト」と「MSSQL 2000」です。表のモニター時に使用する互換モードをプロセッサーに指示します。
conProperty	property=value のコンマ区切りリスト	オプション。 このデータ・ソースとの JDBC 接続を確立するために JDBC URL に追加の接続プロパティーを含める必要がある場合にのみ使用します。 Roman8 をデフォルトの文字セットとする Sybase データベースの場合、次のプロパティーを入力します。 charSet=utf8
customURL		オプション。 データ・ソースに対する接続文字列。これを入力しない場合は、前回入力したフィールドのホスト、ポート、インスタンス、プロパティーなどを使用して接続が行われます。 これは、例えば Oracle Internet Directory (OID) の接続を作成するときに便利です。
dbInstanceAccount	文字列	オプション。 CAS によって使用されるデータベース・アカウント・ログイン名

パラメーター	値のタイプ	記述
dbInstanceDirectory	文字列	オプション。 CAS によって使用される、データベース・ソフトウェアがインストールされたディレクトリー
dbName	文字列	オプション。 DB2® または Oracle データ・ソースの場合、スキーマ名を入力します。他の場合は、データベース名を入力します。
description	文字列	オプション。データ・ソースの詳細説明。
host	文字列	必須。 ホスト名または IP アドレスを入力できます。
KerberosConfigName	文字列	オプション。Guardium システムで既に定義されている Kerberos 構成の名前。
name	文字列	必須。 システム上のデータ・ソースに固有の名前を付けます。
password	文字列	オプション。ユーザーのパスワード。
port	integer	オプション。ポート番号。
savePassword	ブール値	Guardium アプライアンス上の認証資格情報を保存して暗号化します。 (オンデマンドではなく) スケジュールされたタスクとして実行されるアプリケーションでデータ・ソースを定義する場合は、必須になります。yes に設定した場合は、ログイン名とパスワードが必須になります。
serviceName	文字列	Oracle、Informix®、Db2、および IBM® ISeries の場合は必須。Db2 データ・ソースではデータベース名を入力します。それ以外ではサービス名を入力します。
severity		オプション。データ・ソースの重大度分類 (あるいは影響レベル)。
shared	ブール値	オプション。他のアプリケーションと共有する場合は **true** に設定します。データ・ソースを他のユーザーと共有する場合は、GUI からロールを割り当てる必要があります。

パラメーター	値のタイプ	記述
type	値リスト	必須。データ・ソース・タイプを識別します。有効な値: Db2 DB2 for i Db2 for z/OS Informix GBDI MS SQL Server MS SQL サーバー (DataDirect) MySQL NA Netezza Oracle (DataDirect) Oracle (サービス名) Oracle (SID) PostgreSQL Sybase Sybase IQ Teradata アプリケーションが CustomDomain または Classifier である場合、以下も使用できます。 TEXT TEXT:FTP TEXT:HTTP TEXT:HTTPS TEXT:SAMBA
useKerberos	ブール値	オプション。Kerberos 認証を使用する場合は、yes に設定します。yes の場合は、KerberosConfigName を指定する必要があります。
useLDAP	ブール値	オプション。LDAP を使用する場合は、yes に設定します。
user	文字列	オプション。データ・ソースのユーザー。使用する場合、パスワードも使用する必要があります。
useSSL	ブール値	オプション。SSL 認証を使用する場合は、yes に設定します。

例

```
grdapi create_datasource type=DB2 name=chickenDB2  password=guardium user=db2inst1 dbName=dn0chick application=Access_policy
shared=true port=50000 host=chicken.corp.com
```

create_test_exception

このコマンドは、テスト例外にレコードを追加するために使用します。 これは、脆弱性評価の動作に影響を及ぼします。 特定のデータ・ソースのテストが不合格となった場合、該当テスト/データ・ソースのテスト例外表の最終レコードが検査されます。このとき実行日付が最終レコードの開始日付と終了日付の間であれば、テストは PASS に設定され、推奨事項が (例外レコードから) 説明に対して設定されます。さらに結果テキストに次のように設定されます。

テストにパスしました。例外の承認者： 。有効期間 から まで。

注: この API は例外を除去するためにレコードを追加するだけです。必要に応じて新しい日付で新しいレコードを作成してください。

パラメーター	値のタイプ	記述
datasourceName	文字列	必須。定義したデータ・ソースの有効な名前。
testDescription	文字列	必須。セキュリティー・アセスメント内で有効なテスト名。
fromDate		必須。例外が有効である場合の開始日付。
toDate		必須。例外が有効である場合の終了日付。
explanation	文字列	必須。 テストに合格する理由に関する推奨事項。

例

```
grdapi create_test_exception datasourceName=ORAPROD5 testDescription="CVE-2009-0997" fromDate="2012-07-01 08:00:00" toDate="2012-07-31 08:00:00" explanation="Currently in testing stage"
```

get_all_modifiable_guard_params

この汎用の grdapi コマンドは、コマンド modify_guard_param によって変更できるすべてのパラメーターとその値を返します。

パラメーター	値タイプ	記述
paramlike	文字列 任意の数値または文字	指定されたストリングと一致するすべてのパラメーターを返します。これは必須フィールドではありません。指定しない場合は、変更可能なすべてのパラメーターが返されます。
paramdesclike	文字列 任意の数値または文字	指定された説明と一致するすべてのパラメーターを返します。これは必須フィールドではありません。指定しない場合は、変更可能なすべてのパラメーターが返されます。

例:

```
grdapi get_all_modifiable_guard_params paramlike=customtable
```

これにより、パラメーター名、パラメーターの記述、パラメーター・タイプ、およびパラメーター値が返されます。

```
Parameter Name : CUSTOMTABLE_RUNNING_TIMEOUT
Parameter Description : Custom table inactive timeout
Parameter Type : N
Parameter Value: 30
ok
```

get_guard_param

この汎用の grdapi コマンドは、指定されたパラメーターの現行値を返します。

パラメーター	値タイプ	記述
paramName	文字列	パラメーターの名前。 これは必須フィールドではありません。ただし、パラメーター名またはパラメーターの記述のいずれかを指定する必要があります。
paramdesc	文字列	パラメーターの説明 これは必須フィールドではありません。ただし、パラメーター名またはパラメーターの記述のいずれかを指定する必要があります。

構文:

```
grdapi get_guard_param paramName=parameter
```

modify_guard_param

この汎用の grdapi コマンドを使用して、変更可能なパラメーターの値を更新します。

パラメーター	値タイプ	記述
paramName	文字列	変更可能なパラメーターの名前。 これは必須フィールドです。
paramValue	整数	変更可能なパラメーターの新規の値 これは必須フィールドです。

構文:

```
grdapi modify_guard_param paramName=parameter paramValue=value
```

変更可能なパラメーター:

データ・ソースの変更可能なパラメーター	値タイプ	記述
customtable_running_timeout	整数	このパラメーターを変更して、ハングしたデータ・ソースのタイムアウト・メカニズムを設定できます。データ・ソースがハングすると、指定された時間フレーム (分単位で設定) の経過後にカスタム・データのアップロードが停止され、キュー内の次のデータ・ソースにスキップします。

例:

```
grdapi modify_guard_param paramName=customtable_running_timeout paramValue=5
```

list_datasource_by_name

名前で識別されるデータ・ソース定義を表示します。

パラメーター	値のタイプ	記述
name	文字列	必須。データ・ソース名。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • all_managed: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • all: すべての管理対象ユニットと中央マネージャーで実行します • group:<group name>: <group name> によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば api_target_host=10.0.1.123 です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば api_target_host=10.0.1.123 です。

例

```
CLI> grdapi list_datasource_by_name name=chickenDB2
ID=20000
Datasource DatasourceId=20000
Datasource DatasourceTypeId=2
Datasource Name=chickenDB2
Datasource Description=null
Datasource Host=chicken.corp.com
Datasource Port=50000
Datasource ServiceName=
Datasource UserName=db2inst1
Datasource Password=[B@1415de6
Datasource PasswordStored=true
Datasource DbName=dn0chick
Datasource LastConnect=null
Datasource Timestamp=2008-04-18 15:40:58.0
Datasource ApplicationId=2
Datasource Shared=true
Datasource ConProperty=null
Datasource type =DB2
Application Type = Access_policy
ok
```

list_datasource_by_id

ID キーで識別されるデータ・ソース定義を表示します。

パラメーター	値のタイプ	記述
id	integer	必須。リストするデータ・ソースの ID 番号。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi list_datasource_by_id id=2
```

delete_datasource_by_name

データ・ソースがアプリケーションで使用されているのでない限り、指定したデータ・ソース定義を削除します。 この関数は、作成者に関係なくデータ・ソースを削除します。

パラメーター	値のタイプ	記述
name	文字列	必須。データ・ソース名。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_datasource_by_name name=swanSybase
```

delete_datasource_by_id

データ・ソースがアプリケーションで使用されているのでない限り、指定したデータ・ソース定義を削除します。 この関数は、作成者に関係なくデータ・ソースを削除します。

パラメーター	値のタイプ	記述
id	integer	必須。リストするデータ・ソースの ID 番号。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_datasource_by_id id=2
```

update_datasource_by_name

データ・ソース定義を更新します。

パラメーター	値のタイプ	記述
name	文字列	必須。更新するデータ・ソースを指定します。
newName	文字列	オプション。 新規名を指定します。これはシステム上のデータ・ソースで固有でなければなりません。
description	文字列	オプション。データ・ソースの詳細説明。
host	文字列	オプション。 ホスト名または IP アドレスを入力できます。
port	integer	オプション。ポート番号。
savePassword	ブール値	Guardium アプライアンス上の認証資格情報を保存して暗号化します。(オンデマンドではなく) スケジュールされたタスクとして実行されるアプリケーションでデータ・ソースを定義する場合は、必須になります。yes に設定した場合は、ログイン名とパスワードが必須になります。
serviceName	文字列	オプション。 Oracle データ・ソースの場合、サービス名を入力します。
user	文字列	オプション。データ・ソースのユーザー。使用する場合、パスワードも使用する必要があります。
password	文字列	オプション。ユーザーのパスワード。使用する場合、ユーザーも使用する必要があります。
dbName	文字列	オプション。DB2 データ・ソースの場合、データベース名を入力します。

パラメーター	値のタイプ	記述
conProperty	property=value のコンマ区切りリスト	オプション。 このデータ・ソースとの JDBC 接続を確立するために JDBC URL に追加の接続プロパティーを含める必要がある場合にのみ使用します。 Roman8 をデフォルトの文字セットとする Sybase データベースの場合、次のプロパティーを入力します。 CHARSET=utf8
dbInstanceAccount	文字列	オプション。 CAS によって使用されるデータベース・アカウント・ログイン名
dbInstanceDirectory	文字列	オプション。 CAS によって使用される、データベース・ソフトウェアがインストールされたディレクトリー
shared	ブール値	オプション。他のアプリケーションと共有する場合は **true** に設定します。データ・ソースを他のユーザーと共有する場合は、GUI からロールを割り当てる必要があります。
customURL	文字列	オプション。 データ・ソースに対する接続文字列。これを入力しない場合は、前回入力したフィールドのホスト、ポート、インスタンス、プロパティーなどを使用して接続が行われます。 これは、例えば Oracle Internet Directory (OID) の接続を作成するときに便利です。
severity		オプション。データ・ソースの重大度分類 (あるいは影響レベル)。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。
useKerberos	ブール値	オプション。Kerberos 認証を使用する場合は、yes に設定します。yes の場合は、KerberosConfigName を指定する必要があります。
useLDAP	ブール値	オプション。LDAP を使用する場合は、yes に設定します。
useSSL	ブール値	オプション。SSL 認証を使用する場合は、yes に設定します。

例

```
grdapi update_datasource_by_name name=chickenDB2  newName="chicken DB2" user=" " password=" "
```

update_datasource_by_id

データ・ソース定義を更新します。

パラメーター	値のタイプ	記述
id	integer	必須。データ・ソースを指定します。
newName	文字列	オプション。 新規名を指定します。これはシステム上のデータ・ソースで固有でなければなりません。
description	文字列	オプション。データ・ソースの詳細説明。
host	文字列	オプション。 ホスト名または IP アドレスを入力できます。
port	integer	オプション。ポート番号。
savePassword	ブール値	Guardium アプライアンス上の認証資格情報を保存して暗号化します。(オンデマンドではなく) スケジュールされたタスクとして実行されるアプリケーションでデータ・ソースを定義する場合は、必須になります。yes に設定した場合は、ログイン名とパスワードが必須になります。
serviceName	文字列	オプション。 Oracle データ・ソースの場合、サービス名を入力します。
user	文字列	オプション。データ・ソースのユーザー。使用する場合、パスワードも使用する必要があります。
password	文字列	オプション。ユーザーのパスワード。使用する場合、ユーザーも使用する必要があります。
dbName	文字列	オプション。DB2 データ・ソースの場合、データベース名を入力します。
conProperty	property=value のコンマ区切りリスト	オプション。 このデータ・ソースとの JDBC 接続を確立するために JDBC URL に追加の接続プロパティーを含める必要がある場合にのみ使用します。 Roman8 をデフォルトの文字セットとする Sybase データベースの場合、次のプロパティーを入力します。 CHARSET=utf8
dbInstanceAccount	文字列	オプション。 CAS によって使用されるデータベース・アカウント・ログイン名
dbInstanceDirectory	文字列	オプション。 CAS によって使用される、データベース・ソフトウェアがインストールされたディレクトリー
shared	ブール値	オプション。他のアプリケーションと共有する場合は **true** に設定します。データ・ソースを他のユーザーと共有する場合は、GUI からロールを割り当てる必要があります。

パラメーター	値のタイプ	記述
customURL	文字列	オプション。 データ・ソースに対する接続文字列。これを入力しない場合は、前回入力したフィールドのホスト、ポート、インスタンス、プロパティーなどを使用して接続が行われます。 これは、例えば Oracle Internet Directory (OID) の接続を作成するときに便利です。
severity		オプション。データ・ソースの重大度分類 (あるいは影響レベル)。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。
useKerberos	ブール値	オプション。Kerberos 認証を使用する場合は、yes に設定します。yes の場合は、KerberosConfigName を指定する必要があります。
useLDAP	ブール値	オプション。LDAP を使用する場合は、yes に設定します。
useSSL	ブール値	オプション。SSL 認証を使用する場合は、yes に設定します。

例

```
grdapi update_datasource_by_id id=20000 user=" " password=" " newName="chickenDB2hooo"
```

list_db_drivers

現在データ・ソース・タイプとして Oracle (DataDirect) および MS SQL サーバー (DataDirect) をサポートしている データベース・ドライバーの名前のみをリストします。

list_db_drivers_by_details

各データベース・ドライバーの詳細 (名前、クラス、ドライバー・クラス、URL、およびデータ・ソース・タイプ ID) をリストします。

親トピック: GuardAPI

GuardAPI データ・ソース・リファレンス関数

これらの GuardAPI コマンドは、データ・ソース・リファレンス関数の作成、リスト、および削除に使用します。

create_datasourceRef_by_id

特定のアプリケーション・タイプ (特定の分類プロセスなど) の特定のオブジェクトに対して、データ・ソースへの参照を作成します。

パラメーター	値のタイプ	記述
appId	integer	必須。アプリケーションを識別します。このリストのいずれかである必要があります。 • 8 = SecurityAssessment • 47 = CustomTables • 51 = Classifier
datasourceId	integer	必須。データ・ソースを (データ・ソース定義から) 識別します。

パラメーター	値のタイプ	記述
objId	integer	必須。指定された appID タイプのインスタンスを識別します。例えば、apID=51 である場合、これは分類プロセスの ID になります。

例

```
grdapi create_datasourceRef_by_id appId=51 datasourceId=20000 objId=2
```

create_datasourceRef_by_name

特定のアプリケーション・タイプ (特定の分類プロセスなど) の特定のオブジェクトに対して、データ・ソースへの参照を作成します。

表 1. create_datasourceRef_by_name

パラメーター	値のタイプ	記述
application	文字列	必須。アプリケーションを識別します。このリストのいずれかである必要があります。 SecurityAssessment CustomTables Classifier
datasourceName	文字列	必須。データ・ソースを (データ・ソース定義から) 識別します。
objName	文字列	必須。指定されたアプリケーション・タイプのインスタンスを識別します。アプリケーションが Classifier である場合、これは特定の分類プロセスの名前になります。

例

```
grdapi  create_datasourceRef_by_name  application=Classifier  datasourceName=swanSybase  objName="class process1"
```

list_datasourceRef_by_id

特定のアプリケーション・タイプ (特定の分類プロセスなど) の特定のオブジェクトに対して、参照されるすべてのデータ・ソースをリストします。

パラメーター	値のタイプ	記述
appID	integer	必須。アプリケーションを識別します。このリストのいずれかである必要があります。 8 = SecurityAssessment 47 = CustomTables 51 = Classifier
objID	文字列	必須。指定されたアプリケーション・タイプのインスタンスを識別します。 例えば、アプリケーションが Classifier である場合、これは特定の分類プロセスの ID になります。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi list_datasourceRef_by_id appId=13 objId=1
```

list_datasourceRef_by_name

特定のアプリケーション・タイプ (特定の分類プロセスなど) の特定のオブジェクトに対して、参照されるすべてのデータ・ソースをリストします。

パラメーター	値のタイプ	記述
application		必須。アプリケーションを識別します。このリストのいずれかである必要があります。 SecurityAssessment CustomTables Classifier
objName	文字列	必須。指定されたアプリケーション・タイプのインスタンスを識別します。アプリケーションが Classifier である場合、これは特定の分類プロセスの名前になります。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdap list_datasourceRef_by_name application=Classifier  objName="class process1"
```

delete_datasourceRef_by_id

特定のアプリケーション・タイプ (特定の分類プロセスなど) の特定のオブジェクトに対して、データ・ソース参照を削除します。

パラメーター	値のタイプ	記述
appId		必須 (整数)。アプリケーションを識別します。このリストのいずれかである必要があります。 8 = SecurityAssessment 47 = CustomTables 51 = Classifier
datasourceId	integer	必須。データ・ソースを (データ・ソース定義から) 識別します。
objId	integer	必須。指定された appID タイプのインスタンスを識別します。例えば、apID=51 である場合、これは分類プロセスの ID になります。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_datasourceRef_by_id appId=51 datasourceId=2 objId=1
```

delete_datasourceRef_by_name

特定のアプリケーション・タイプ (特定の分類プロセスなど) の特定のオブジェクトに対して、データ・ソース参照を削除します。

パラメーター	値のタイプ	記述
application		必須。アプリケーションを識別します。このリストのいずれかである必要があります。 SecurityAssessment CustomTables Classifier
datasourceName	文字列	必須。データ・ソースを (データ・ソース定義から) 識別します。
objName	文字列	必須。指定されたアプリケーション・タイプのインスタンスを識別します。アプリケーションが Classifier である場合、これは特定の分類プロセスの名前になります。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_datasourceRef_by_name application=Classifier datasourceName=swanSybase objName="class process1"
```

親トピック: GuardAPI

GuardAPI データ・ユーザー・セキュリティー関数

以下の GuardAPI コマンドは、データ・ユーザー・セキュリティー関数を作成、リスト、削除、および更新するために使用します。

create_user_hierarchy

ユーザー・データ・セキュリティー階層にユーザーと親の関係を追加します。

パラメーター	値のタイプ	記述
userName	文字列	必須。ユーザーの名前。
parentUserName	文字列	必須。親ユーザーの名前。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi create_user_hierarchy userName=admin parentUserName=accessmgr

注: 循環的な挿入 (親レポートが子に挿入される) の場合、エラーとなります。

list_user_hierarchy_by_parent_user

ユーザー・データ・セキュリティー階層内の関係をリストします。

パラメーター	値のタイプ	記述
userName	文字列	必須。ユーザーの名前。
create	ブール値	create_user_hierarchy API 呼び出しの create ステートメントを、true を設定すると作成し、false を設定すると生成しません。 このパラメーターは、バッチ・ファイルの生成に必要なすべてのコマンドを取得するときに使用します。 このバッチ・ファイルは、親と子のそれぞれの対を別の Guardium システムに移動するときに使用できます。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi list_user_hierarchy_by_parent_user userName=admin create=true

注: 直接的な親子関係のみがリスト表示されます。「孫」は表示されません。

delete_user_hierarchy_by_entry_id

項目 ID ごとにユーザー・データ・セキュリティー階層にある関係を削除します。

パラメーター	値のタイプ	記述
id	integer	必須。項目を指定します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi delete_user_hierarchy_by_entry_id id=1

注: 項目が存在しない場合、失敗条件はありません。

delete_user_hierarchy_by_user

ユーザー・データ・セキュリティー階層にある関係をユーザーごとに削除します。

パラメーター	値のタイプ	記述

パラメーター	値のタイプ	記述
userName	文字列	必須。 ユーザーの名前。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi delete_user_hierarchy_by_user userName=admin

注:

ユーザーが存在しない場合、失敗条件はありません。

ユーザーが複数の親を持つ場合、削除が複数回行われます。

create_allowed_db

ユーザーとデータベースの関連を作成します。

パラメーター	値のタイプ	記述
userName	文字列	必須。ユーザーの名前。
serverIp		必須。サーバー IP
instanceName	文字列	必須。インスタンス名
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi create_allowed_db userName=admin serverIp=192.168.1.1 instanceName=abcd

list_allowed_db_by_user

ユーザーとデータベースの関連をユーザーごとにリストします。

パラメーター	値のタイプ	記述
userName	文字列	必須。ユーザーの名前。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi list_allowed_db_by_user userName=admin

delete_allowed_db_by_entry_id

ユーザーとデータベースの関連を項目 ID ごとに削除します。

パラメーター	値のタイプ	記述
id	integer	必須。項目を指定します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi delete_allowed_db_by_entry_id id=1

delete_allowed_db_by_user

ユーザーとデータベースの関連をユーザーごとに削除します。

パラメーター	値のタイプ	記述
userName	文字列	必須。ユーザーの名前。
serverIp		サーバー IP。
instanceName	文字列	インスタンス名。 注: インスタンス名を「blank」にする場合、instanceName=[blank] と入力します (instanceName=blank ではありません)。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi delete_allowed_db_by_user userName=scott

update_user_db

アクティブなユーザー - DB 関連付けマップに最近のすべての変更を全面的に適用

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi update_user_db

注: 一元管理構成では、このコマンドは中央マネージャー上で実行してください。
親トピック: GuardAPI

GuardAPI エンタープライズ・ロード・バランシング関数

以下の GuardAPI コマンドを使用して、ロード・バランシング・パラメーターの表示と設定、現在のロード・マップの表示、および S-TAP と管理対象ユニット・グループの関連付けの管理を行います。

get_load_balancer_load_map

現在のロード・マップを表示します。

```
grdapi get_load_balancer_load_map
```

get_load_balancer_params

現在のロード・バランサーの構成パラメーターを表示します。

```
grdapi get_load_balancer_params
```

set_load_balancer_param

ロード・バランサーの構成パラメーターを設定します。

```
grdapi set_load_balancer_param [paramName=value][paramValue=value][paramType=STAP]
```

使用可能なパラメーターおよび許可される値のリストについては、『エンタープライズ・ロード・バランシング構成パラメーター』を参照してください。

例えば、`grdapi set_load_balancer_params paramName=LOAD_BALANCER_ENABLED paramValue=0 paramType=STAP` です。

次の形式を使用して正しく入力します。

grdapi set_load_balancer_param --help=true

ID=0

関数パラメーター : paramName - 文字列 - 必須

paramType - 文字列 - 必須

paramValue - 文字列 - 必須

パラメーターの定数値リストを取得するには、関数に「--get_param_values」を指定して呼び出します。

assign_load_balancer_groups

管理対象ユニット・グループをアプリケーションまたは S-TAP グループに割り当てます。

```
grdapi assign_load_balancer_groups muGroupName=[value] appGroupName=[value]
```

パラメーター	値のタイプ	記述
muGroupName	管理対象ユニット・グループ名	例えば、muGroupName=mu_group_NA です。
appGroupName	アプリケーションまたは S-TAP グループ名	例えば、appGroupName=app_group_NA です。
ifFailoverGroup	1 または 0	例えば、isFailoverGroup=0 です。

unassign_load_balancer_groups

アプリケーションまたは S-TAP グループから管理対象ユニット・グループの割り当てを解除します。

```
grdapi unassign_load_balancer_groups muGroupName=[value] appGroupName=[value]
```

パラメーター	値のタイプ	記述
muGroupName	管理対象ユニット・グループ名	例えば、muGroupName=mu_group_NA です。
appGroupName	アプリケーションまたは S-TAP グループ名	例えば、appGroupName=app_group_NA です。

親トピック: GuardAPI

GuardAPI 資格最適化機能

これらの GuardAPI コマンドは、資格最適化データ・ソースおよびレポート作成を有効化および構成するために使用します。

enable_entitlement_optimization

このコレクターでの資格最適化機能を有効にします。

```
grdapi enable_entitlement_optimization
```

disable_entitlement_optimization

このコレクターでの資格最適化機能を無効にします。

```
grdapi disable_entitlement_optimization
```

add_datasource_to_entitlement_optimization

このソースから資格最適化データ収集にデータを追加し、指定されたとおりに個々のタブにデータを追加します。

```
grdapi add_datasource_to_entitlement_optimization
```

パラメーター	値のタイプ	記述
datasourceName	datasourceName	データ・ソースの名前
isEnabled	true、false のいずれか	資格最適化に対して、データ・ソースが有効化または無効化されます。 デフォルトは false です。
userScope	1 つ以上のコンマ区切りの Guardium ユーザー・グループ ID (グループにはユーザーのみを含める必要があります)	オプション。「資格の推奨」の結果は、このユーザー・グループによってフィルタリングされます。「資格の参照」の結果は、 この範囲内にユーザーが含まれているかどうかを示し、範囲外のユーザーのユーザー・アクティビティー・カウントは表示しません。 デフォルトは NULL です。
objectScope	1 つ以上のコンマ区切りの Guardium オブジェクト・グループ ID (グループにはオブジェクトのみを含める必要があります)	オプション。「資格の推奨」の結果は、このオブジェクト・グループによってフィルタリングされます。 デフォルトは NULL です。
extractActivity	true、false のいずれか	データ・ソース・アクティビティーの抽出を有効または無効にします。 「資格の参照」および「仮定」の場合は、true である必要があります。 デフォルトは false です。
extractEnt	true、false のいずれか	資格データの抽出を有効または無効にします。

itlement		「新機能」、「ユーザーおよびロール」、「推奨」、および「資格の参照」の場合は、true である必要があります。 デフォルトは false です。
generateRoleClusters	`true、false` のいずれか	**「仮定」**タブで使用される、データ・ソースからの動作ロール・クラスタリングの抽出を有効または無効にします。 「仮定」の場合は、true である必要があります。 デフォルトは false です。
generateNews	`true、false` のいずれか	このデータ・ソースからのアクティビティーが**「新機能」**タブに含まれます。 デフォルトは false です。
generateRecommendations	`true、false` のいずれか	このデータ・ソースからのアクティビティーが**「推奨」**タブに含まれます。 デフォルトは false です。
filterTempObjects	`true、false` のいずれか	将来に使用の予定。 一時オブジェクトがデータ・ソースの収集データからフィルタリングされます。 デフォルトは true です。
filterIgnoreVerbs	`true、false` のいずれか	将来に使用の予定。 無視動詞がデータ・ソースの収集データからフィルタリングされます。 デフォルトは true です。

remove_datasource_from_entitlement_optimization

このソースからのすべてのデータを、資格最適化データ収集から削除します。

```
remove_datasource_from_entitlement_optimization
```

set_entitlement_datasource_parameter

資格の最適化に対して既に有効になっているデータ・ソースのパラメーターを変更します。add_datasource_to_entitlement_optimization と同じパラメーターを使用します。

```
grdapi set_entitlement_datasource_parameter
```

get_entitlement_datasource_parameter

このコレクターの各データ・ソースのパラメーター設定を表示します。

```
grdapi get_entitlement_datasource_parameter
```

例:

```
資格最適化は有効です
==================================
Datasource: SCALE-DB16
==================================
isEnabled: true
userScope:
objectScope:
extractActivity: true
extractEntitlement: true
generateRoleClusters: true
generateNews: true
generateRecommendations: true
filterTempObjects: true
filterIgnoreVerbs: true
==================================
Datasource: on12scal
==================================
isEnabled: true
userScope:
objectScope:
extractActivity: true
extractEntitlement: true
generateRoleClusters: true
generateNews: true
generateRecommendations: true
filterTempObjects: true
filterIgnoreVerbs: true
```

親トピック: GuardAPI

GuardAPI 外部フィード関数

これらの GuardAPI 関数は、外部フィードのマッピングを作成するために使用します。

create_ef_mapping

この関数はマッピングを作成し、*reportName* パラメーターで指定されたレポートの名前に基づいて、表にデータを取り込みます。 各マッピングの名前は、EF_MAP_TYPE_HDR.EF_TYPE_DESC に保管されます。この名前は、*reportName* の値と同一になります。 ターゲット表名も、*reportName* パラメーターに基づきますが、単語間に下線が追加されます。例えば、「My Report」は MY_REPORT になります。

パラメーター	値のタイプ	記述
reportName	文字列	外部フィード・マッピングで使用するレポートの名前。このパラメーターは、マッピングの名前およびターゲット表名の決定も行います。

modify_ef_mapping

場合によっては、create_ef_mapping によって生成される名前が特定のデータベースに適さないことがあります。その場合、modify_ef_mapping を使用して、データベース要件に適合するように名前を調整できます。事前定義の Guardium マッピングを保護するため、変更できるマッピングは ID >= 20000 のマッピングのみです。

パラメーター	値のタイプ	記述
reportName	文字列	変更するマッピングの名前。
modifyObj		変更するデータベース・オブジェクト (*table* (表) または *column* (列)) を指定します。 既存の値は、list_ef_mapping 関数を使用して取得できます。
oldName		削除する古い表名を指定します。
newName	文字列	使用する新しい表名を指定します。

delete_ef_mapping

この関数を使用すると、既存のマッピングを削除できます。事前定義の Guardium マッピングを保護するため、削除できるマッピングは ID >= 20000 のマッピングのみです。

パラメーター	値のタイプ	記述
reportName	文字列	削除するマッピングの名前。

list_ef_mapping

パラメーターを指定せずに実行した場合、この関数は、お客様が作成したすべてのマッピングのリストを返します。*reportName* パラメーターを指定して実行した場合、この関数は、指定したマッピングの詳細 (外部フィードで使用される表名や列名など) を返します。

表 1.

パラメーター	値のタイプ	記述
reportName	文字列	オプション。詳細を返すマッピングの名前。

親トピック: GuardAPI

GuardAPI External S-TAP 関数

pull_external_stap_keystore CLI コマンドを使用して、中央マネージャーとその管理対象ユニット間で External S-TAP 鍵ストアを移動します。

pull_external_stap_keystore

中央マネージャーまたは管理対象ユニットから External S-TAP 鍵ストアを取り込むには、grdapi pull_external_stap_keystore を以下のように使用します。

- grdapi pull_external_stap_keystore を管理対象ユニットで実行すると、External S-TAP 鍵ストアが中央マネージャーからその管理対象ユニットにプルされます。
- grdapi pull_external_stap_keystore を CM で実行すると、中央マネージャーから関連付けられた 1 つまたはすべての管理対象ユニットに、External S-TAP 鍵ストアがプルされます。

パラメーター	値	記述
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

- 管理対象ユニットから中央マネージャーに鍵ストアをプルするには、以下のコマンドを管理対象ユニットから実行します。

  ```
  cli>grdapi pull_external_stap_keystore
  ```

- 中央マネージャーからすべての関連付けられた管理対象ユニットに鍵ストアをプルするには、以下のコマンドを中央マネージャーから実行します。

  ```
  cli>grdapi pull_external_stap_keystore
  ```

- 中央マネージャーから特定の管理対象ユニットに鍵ストアをプルするには、以下のコマンドを中央マネージャーから実行します。

  ```
  cli>grdapi pull_external_stap_keystore  api_target_host=hostname
  ```

親トピック: GuardAPI

GuardAPI ファイル・アクティビティー・モニター関数

以下の GuardAPI コマンドは、ファイル・アクティビティー・モニターの有効化および無効化、ファイルの調査ダッシュボードのアクティビティーおよびライセンス抽出のスケジュールの構成、ファイル・アクティビティー・モニターに関する情報の取得を行う場合に使用します。

GuardAPI コマンドの grdapi create_policy を使用して FAM ポリシーを作成します。ポリシーを作成したら、FAM 固有の GuardAPI コマンドを使用します。

例:

grdapi create_policy ruleSetDesc='TEST'

grdapi create_fam_rule policyName='TEST' ruleName=r-test-sles11 actionName="Log As Violation and Audit" serverHost="9.70.144.98:FAM" filePath="/famtest/*"

GuardAPI コマンドの使用手順については、GuardAPIを参照してください。

enable_fam_crawler

クローラー結果およびファイル・アクティビティー・データを処理するように Guardium システムを設定します。 結果は、クイック検索索引ファイルに自動的に追加されます。 各種パラメーターを使用して、ファイル・クイック検索アクティビティー、ライセンス抽出、およびリモート・グループへのデータ取り込みをスケジュールします。

注: 調査ダッシュボードも、コマンド grdapi enable_quick_search schedule_interval=1 を使用して有効にする必要があります。

パラメーター	値のタイプ	記述
extraction_start		ファイル・クイック検索へのデータの抽出を開始する初回の日時。過去 2 日以内に制限されます。デフォルトは現在の時刻です。単位を HOUR に設定した場合、時間単位で丸められます。DAY に設定した場合、日単位で丸められます。
schedule_start		デフォルトは現在の時刻です。
activity_schedule_interval	integer	必須。このパラメーターは、アクティビティー・スケジュール間隔を設定します。推奨間隔は、単位を MINUTE に設定した 2 です。
activity_schedule_units	値リスト	必須。このパラメーターは、アクティビティー・ユニットの単位を設定します。値は MINUTE と HOUR のいずれかです。推奨単位は MINUTE です。
entitlement_schedule_interval	integer	必須。このパラメーターは、ライセンス・スケジュール間隔を設定します。推奨間隔は、単位を DAY に設定した 1 です。

パラメーター	値のタイプ	記述
entitlement_schedule_units	値リスト	必須。このパラメーターは、ライセンス・スケジュールの単位を設定します。指定可能な値は MINUTE、HOUR、および DAY です。推奨単位は DAY です。

例

```
grdapi enable_fam_crawler extraction_start=< > schedule_start=< >
 activity_schedule_interval=2 activity_schedule_units=MINUTE
entitlement_schedule_interval=10 entitlement_schedule_units=MINUTE
```

disable_fam_crawler

ファイル・アクティビティー・モニターを無効にします。ファイル・クイック検索アクティビティーおよびライセンス抽出のスケジューラーが削除されます。この関数は、リモート・グループへのデータ取り込みも無効化します。

例

```
grdapi disable_fam_crawler
```

get_fam_crawler_info

ファイル・アクティビティー・モニターの状況を表示します。有効になっている場合、このコマンドにより、ライセンス抽出およびファイル・クイック検索アクティビティーのスケジュールの設定が表示されます。

```
FAM クローラー（サーバー・サイド）が無効になっています。
```

```
FAM クローラー（サーバー・サイド）が有効になっています。ライセンス（1 DAY）アクティビティー（2 MINUTE）(Entitlement(1 DAY) Activity(2 MINUTE))
```

例

```
grdapi get_fam_crawler_info
```

list_policy_fam_rule

FAM ポリシー内のすべてのルールをリストします。

パラメーター	値のタイプ	記述
policyName	文字列	必須。ポリシー名
ruleName	文字列	オプション。ruleName が指定されていない場合、すべてのポリシー・ルールが詳細とともに表示されます。ruleName が指定されている場合、そのルールの詳細がリストされます。

create_fam_rule

新しい FAM ルールを作成します。

パラメーター	値のタイプ	記述
policyName	文字列	必須。ポリシー名。
ruleName	文字列	必須。ルール名。
filePath	文字列	モニター対象のファイル・パス。filePath と filePathGroup のいずれかを指定する必要があります。
notfilePath	ブール値	「はい」または「いいえ」にする必要があります。「はい」を指定すると、指定されているパス内のファイルを除くすべてのファイルにこのルールが適用されます。

filePathGroup	文字列	ファイル・パスのグループ。filePath と filePathGroup のいずれかを指定する必要があります。
includeSubDirectory	ブール値	「はい」または「いいえ」にする必要があります。「はい」を指定した場合、すべてのサブディレクトリー内のファイルが含められます。
removableMedia	文字列	「はい」または「いいえ」にする必要があります。
osUser	文字列	OS ユーザー名。
osUserGroup	文字列	OS ユーザーのグループ。
notOSUser	文字列	「はい」または「いいえ」にする必要があります。「はい」を指定した場合、指定した osUser を除くすべてのユーザーが使用されます。
serverHost	文字列	ホスト名。
serverHostGroup	文字列	ホスト名のグループ。
command	文字列	ルールに含めるコマンド名。以下のいずれか。 • DELETE • EXECUTE • FILEOP • READ • WRITE
commandGroup	文字列	コマンドのグループ。
notCommand	文字列	「はい」または「いいえ」にする必要があります。「はい」を指定した場合、指定したコマンドを除くすべてのコマンドが使用されます。
actionName	文字列	必須、FAM アクションの名前。
messageTemplate	文字列	メッセージ・テンプレート名。
notificationType	文字列	通知タイプ。以下のいずれか。 • メール • SNMP • カスタム • SYSLOG
userLoginName	文字列	ユーザーのログイン名。
classDestination	文字列	呼び出すカスタム・クラスの名前。

policy_fam_rule_delete

FAM ポリシーからルールを削除します。

パラメーター	値のタイプ	記述
policyName	文字	必須。ポリシー名

	列	
ruleName	文字列	必須。削除するルールの名前。

add_action_to_fam_rule

既存の FAM ルールにアクションを追加します。

パラメーター	値のタイプ	記述
actionName	文字列	必須。FAM アクションの名前。
alertReceiver	文字列	AlertReceiver は、アプライアンスの任意のユーザー (管理者や他のユーザー) です。
command	文字列	ルールに含めるコマンド名。以下のいずれか。 • DELETE • EXECUTE • FILEOP • READ • WRITE
messageTemplate	文字列	文字列。メッセージ・テンプレート名。
notificationType	文字列	通知タイプ。以下のいずれか。 • メール • SNMP • カスタム • SYSLOG
policyName	文字列	必須。有効なポリシー名。
ruleName	文字列	必須。更新するルールの名前。

親トピック: GuardAPI

GuardAPI GIM 関数

これらの CLI コマンドは、GIM 関数のリスト、更新、割り当て、削除、およびキャンセルに使用します。

gim_list_registered_clients

登録済みのすべてのクライアントをリストします。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_list_registered_clients
```

gim_list_client_params

特定のクライアントに割り当てられたすべての (モジュール) パラメーターをリストします。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_list_client_params clientIP=192.168.12.210
```

gim_update_client_params

特定のクライアントの単一モジュール・パラメーターを更新します。

パラメーター	値のタイプ	記述
clientIP	文字列	必須。ターゲット・クライアントの IP
paramName	文字列	必須。パラメーター名
paramValue	文字列	必須。パラメーター値
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_update_client_params clientIP=192.168.1.100 paramName=STAP_TAP_IP paramValue=192.168.1.100
```

gim_list_client_modules

特定のクライアントに割り当てられたすべてのモジュールとその状態をリストします。

パラメーター	値のタイプ	記述

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_list_client_modules clientIP=192.168.2.210
```

gim_load_package

「filename」内のすべてのモジュールをロードします。

注: このコマンドは、ローカル・ファイル・システムにあるファイルをロードします。したがって、このコマンドの前に CM/Guardium アプライアンスへファイルをロードするプロシージャー (cmd='fileserver') が必要です。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_load_package  filename=*.gim
```

注: filename にはワイルドカード「*」を使用することができます。

gim_assign_bundle_or_module_to_client_by_version

クライアントにバンドル/モジュールを割り当てます。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス
module	文字列	必須 - モジュール
moduleVersion	文字列	必須 - モジュール・バージョン

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_assign_bundle_or_module_to_client_by_version clientIP=192.168.1.100  module=BUNDLE-STAP moduleVersion="8.0_r1234_1"
```

gim_schedule_install

お客様に割り当てられていて、まだインストールしていない (保留中など) モジュール/バンドルすべてのインストールをスケジュールします。 パラメーター module が指定される場合、要求されたモジュールのみがスケジュールに入れられます。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス
module	文字列	オプション - モジュール。モジュールがコマンド内で指定されていない場合、指定された clientIP のモジュールすべてがインストール対象としてスケジュールに入れられます。
date		必須 - 日付。形式: 'now' または 'yyyy-MM-dd HH:mm'
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_schedule_install clientIP=192.168.1.100 module=BUNDLE-STAP  date="2008-07-02 14:50"
```

```
grdapi gim_schedule_install clientIP=192.168.1.100 date="2008-07-02 14:50"
```

注: 即時に実行するものがある場合は、過去の日付を使用することができます。

gim_list_client_status

特定のクライアントに対して実行した最新の操作の状況を表示します。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_list_client_status clientIP=192.168.1.100
```

gim_uninstall_module

特定のクライアントのモジュール/バンドルをアンインストールします。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス
module	文字列	必須 - モジュール。
date		必須 - 日付。形式: 'now' または 'yyyy-MM-dd HH:mm'
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_uninstall_module clientIP=192.168.1.100 module=BUNDLE-STAP
```

gim_cancel_install

特定のクライアントへのバンドル/モジュールのインストールをキャンセルします。 インストールのキャンセルは、モジュール/バンドルがクライアントによってインストールのプロセスに入っていない (STATE=IP または IP-PR) 場合のみ行うことができます。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス
module	文字列	必須 - モジュール。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_cancel_install clientIP=192.168.1.100 module=BUNDLE-STAP
```

gim_list_bundles

使用可能なバンドルすべてをリストします。 バンドルとは、クライアントにインストールできるモジュールの集まりです。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_list_bundles
```

gim_list_mandatory_params

1 つのモジュールの必須パラメーターをリストします。

パラメーター	値のタイプ	記述
module	文字列	必須パラメーターを表示する GIM モジュールの名前
version	文字列	必須パラメーターを表示する GIM モジュールのバージョン

例

```
grdapi gim_list_mandatory_params module=name version=number
```

gim_assign_latest_bundle_or module_to_client

特定のクライアントに使用可能な最新 (つまり最上位バージョン) のバンドルまたはモジュールを割り当てます。

パラメーター	値のタイプ	記述

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス
module	文字列	必須 - モジュール。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_assign_latest_bundle_or_module_to_client clientIP=192.168.1.100  module=BUNDLE_STAP
```

gim_schedule_uninstall

クライアントに割り当てられており、まだアンインストールしていない (つまり「PENDING」) モジュール/バンドルすべてのアンインストールをスケジュールに入れます。 パラメーター module が指定される場合、要求されたモジュールのみがスケジュールに入れられます。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス
module	文字列	オプション - モジュール。モジュールがコマンド内で指定されていない場合、指定された clientIP のモジュールすべてがインストール対象としてスケジュールに入れられます。
date		必須 - 日付。形式: 'now' または 'yyyy-MM-dd HH:mm'
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_schedule_uninstall clientIP=192.168.1.100 module=BUNDLE-STAP  date=”2008-07-02 14:50”  grdapi gim_schedule_uninstall
clientIP=192.168.1.100 date=”2008-07-02 14:50”
```

gim_cancel_uninstall

特定のクライアントのバンドル/モジュールのアンインストールをキャンセルします。 アンインストールのキャンセルは、モジュール/バンドルがクライアントによってインストールのプロセスに入っていない (STATE=IP または IP-PR) 場合のみ行うことができます。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス

パラメーター	値のタイプ	記述
module	文字列	必須 - モジュール。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_cancel_uninstall clientIP=192.168.1.100 module=BUNDLE-STAP
```

gim_remove_bundle

このコマンドは、bundlePackageName をデータベースおよびファイル・システム (/var/log/guard/gim_packages。Guardium システムが中央マネージャーである場合はさらに /var/gim_dist_packages も) から削除します。

パラメーター (必須):

bundlePackageName

パラメーター値として、gim_list_unused_bundles の出力に指定されたバンドル・パッケージ名を取ります。 このコマンドは、次の条件を満たす場合にのみ成功します。

2.1 bundlePackageName の値がバンドルを参照している

2.2 bundlePackageName の値がどのクライアントにも割り当てられていない

2.3 bundlePackageName の値が存在する

2.4 bundlePackageName の値を参照するバンドルが 1 つだけ存在する

バンドルをデータベース/ファイル・システムから削除するには、これらのすべての条件 (2.1 から 2.4) が true である必要があります。そうでない場合は、エラーが生成されます。

例

```
grdapi gim_remove_bundle  bundlePackageName= bundlePackageName
```

gim_unassign_client_module

クライアントからモジュールを割り当て解除します。 gim_remove_module とは異なり、このコマンドは CM/Guardium アプライアンス上でモジュールと特定のクライアント間の関連付けを解除します。 このコマンドは実際のデータベース・サーバー・マシンでモジュールをアンインストールしたり、削除したりするものではありません。モジュールの現在状態に関して、データベース・サーバー (つまりクライアント) の情報と CM/Guardium アプライアンスの情報の間で、同期の問題が生じた場合にのみ使用されるものです。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス
module	文字列	オプション。モジュール。モジュールがコマンド内で指定されていない場合、指定された clientIP のモジュールすべてがインストール対象としてスケジュールに入れられます。
date		必須。日付。形式: 'now' または 'yyyy-MM-dd HH:mm'

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_unassign_client_module clientIP=192.168.1.100  module=STAP
```

gim_get_purge_list

以前 Guardium® アプライアンスまたは CM にアップロードした古いソフトウェア・パッケージ (GIM ファイル) をリストします。

パラメーター	値のタイプ	記述
olderThan	文字列	必須 - 日数。 指定された日数より古いファイルがパージされます。 0 以上の任意の数字が有効です。
excludeLatest	ブール値	オプション - true または false (デフォルト値は true)。 true: モジュール、OS ごとの最新バージョンはパージしません。 false: モジュール、OS ごとの最新バージョンをパージします。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_get_purge_list olderThan=30 excludeLatest=true
```

gim_purge

以前 Guardium アプライアンスまたは CM にアップロードした古いソフトウェア・パッケージ (GIM ファイル) を削除します。

パラメーター	値のタイプ	記述
olderThan	文字列	必須 - 日数。 指定された日数より古いファイルがパージされます。 0 以上の任意の数字が有効です。
excludeLatest	ブール値	オプション - true または false (デフォルト値は true)。 true: モジュール、OS ごとの最新バージョンはパージしません。 false: モジュール、OS ごとの最新バージョンをパージします。
filename	文字列	オプション - 削除する特定のファイル。 指定したファイルがバンドル (guard-bundle で始まるなど) である場合、このバンドルの内容が削除されます。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_purge olderThan=30
```

注:

このコマンドには filename パラメーターまたは (olderThan または excludeLatest のどちらか、またはその両方) を指定できます。

gim_purge は、現在インストールのスケジュールに入っているファイルはパージしません。

gim_purge では、「/」文字を含むファイル (パラメーター のファイル名など) を削除できません。

gim_get_available_modules

特定のサーバーにインストール可能なモジュール/バンドルをリストします。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス

例

```
grdapi gim_get_available_modules clientIP=192.168.1.100
```

gim_get_client_last_event

特定のクライアントに対して実行された最新の操作をリストします。

gim_get_client_last_event は、機能が限定された GrdAPI コマンドです。このコマンドが行うのは、最新のインストール試行中に最後に発生したイベントを表示することだけです。例えば、最後に S-TAP をインストールした際にいくつかのエラーが発生した場合は、その grdapi コマンドを実行することで、それが表示されます。ただし、データベース・サーバー上で直接にインストールの問題を手動で修正した場合、(S-TAP が現在実行中であっても) この grdapi コマンドは引き続き元の同じエラー・メッセージを表示します。このコマンドは、データベース・サーバーでの手動修正後に S-TAP 状況を評価するためには使用しないでください。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス

例

grdapi gim_get_client_last_event clientIP=192.168.1.100

grdapi gim_get_client_last_event clientIP=winx64

grdapi gim_get_client_last_event clientIP=9.70.144.73

gim_get_modules_running_status

特定のサーバー上で現在稼働しているモジュールおよびバンドルをリストします。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス
process	文字列	プロセスの名前
status	ONまたはOFF	
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_get_modules_running_status clientIP=192.168.1.100 process= status=
```

gim_list_unused_bundles

このコマンドは、未使用の (どのデータベース・サーバーにもインストールされていない) バンドル、およびアップロード可能な個々の Windows モジュール (Windows CAS、Windows FAM など) のリストを返します。

パラメーター (必須):

includeLatest (valid values 0/1)

1 に設定した場合、最新の未使用のバンドルを含めた、使用されていないバンドルのリストが返されます。

例

```
grdapi gim_list_unused_bundles includeLatest=1
```

gim_reset_client

選択されたクライアントからモジュールを分離します。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_reset_client clientIP=192.168.1.100
```

gim_set_diagnostics

GIM 内に診断収集を設定します。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_set diagnostics clientIP=192.168.1.100
```

gim_set_global_param

GIM 内にグローバル・パラメーターを設定します。

パラメーター	値のタイプ	記述
clientIP	文字列	必須 - クライアントの IP アドレス
paramName	文字列	必須 - マップされる API 関数内のパラメーターの名前
paramValue	文字列	必須 - マップされる API 関数内のパラメーターの値
sqlguardip	文字列	オプション - この GIM エージェントが接続するコレクターの IP アドレス/ホスト名。
ca_file	文字列	オプション - 認証局 PEM ファイルの完全なファイル名パス。

パラメーター	値のタイプ	記述
key_file	文字列	オプション - 秘密鍵 PEM ファイルの完全なファイル名パス。
cert_file	文字列	オプション - 証明書 PEM ファイルの完全なファイル名パス。
gim_listener_default_port	文字列	オプション - GIM エージェント・サーバー・モード用に別のポートを設定します。
gim_listener_default_shared_secret	文字列	オプション - 新しいサーバー・モード GIM エージェントに要求を送信するコレクターを検証するための共有パスワードを設定します。
no_listener	文字列	オプション - サーバー・モードの GIM エージェントを無効化します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_set_global_param clientIP=192.168.1.100 paramName=gim_listener_default_port paramValue=8445
```

gim_remote_activation

コレクターの IP アドレスをサーバー・モードの GIM エージェントまたは GIM エージェントのグループに接続します。

パラメーター	値のタイプ	記述
targetGroup	文字列	オプション - コレクターが接続するすべてのデータベース・サーバーのグループ名。targetHost パラメーターとともに指定することはできません。
sharedSecret	文字列	オプション - インストール時に構成された共有パスワード。
targetPort	文字列	オプション - GIM エージェントのポート・サーバー・モード。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi gim_remote_activation targetGroup=<someGroup> sharedSecret=<password> targetPort=8445
```

親トピック: GuardAPI

GuardAPI グループ関数

これらの GuardAPI コマンドは、データ・ソース・グループ関数の作成、リスト、および削除に使用します。

注: 一元管理環境では、すべてのグループが中央マネージャーで定義され、スケジュールに基づいて管理対象ユニットに送信されます。

グループ関数

create_group

list_group_by_id

list_group_by_desc

delete_group_by_id

delete_group_by_desc

update_group_by_id

update_group_by_desc

flatten_hierarchical_groups

メンバー関数

create_member_to_group_by_id

create_member_to_group_by_desc

list_group_members_by_id

list_group_members_by_desc

delete_member_from_group_by_id

delete_member_from_group_by_desc

create_group

create_group

グループ定義を作成します。

パラメーター	値のタイプ	記述
desc	文字列	必須。新規グループの固有の記述を入力します。
type	値リスト	必須。次のいずれかでなければなりません。 アプリケーション・イベントの値の数値 アプリケーション・イベントの値の文字列 アプリケーション・イベントの値のタイプ アプリケーション項目名 アプリケーション・モジュール アプリケーション・システム ID アプリケーションのトランザクション・コード アプリケーション・ユーザー 監査タスク・タイプ クライアントのホスト名 クライアント IP クライアント IP/データベース・ユーザー クライアント IP/ソース・アプリケーション/データベース・ユーザー クライアント IP/ソース・アプリケーション/データベース・ユーザー/ サーバー IP/サービス名前 クライアントの MAC アドレス

パラメーター	値のタイプ	クライアント OS 記述
		コマンド CVE 定義済みテスト データベース名 データベース・エラー・コード データベース・プロトコル データベース・プロトコル・バージョン データベースのロール データベース・ユーザー/オブジェクト/特権 DB のバージョン/パッチ 例外タイプ フィールド ファイルの許可 グローバル ID Guardium® 監査カテゴリー Guardium のロール Guardium ユーザー ログイン成功コード ネット・プロトコル オブジェクト/コマンド オブジェクト/フィールド オブジェクト 操作タイプ OS ユーザー ポート 修飾されたオブジェクト 影響を受けるレコード スキーマ センテンスの深さ サーバーの記述 サーバーのホスト名 サーバー IP サーバー IP/データベース・ユーザー サーバー IP/サーバー・ポート サーバー IP/サービス名/データベース・ユーザー サーバー OS サーバー・タイプ サービス名 ソース・プログラム SQL ベースの定義済みテスト TeraData プロファイル/データベース・ユーザー

パラメーター	値のタイプ	記述
		TTL ユーザー 脆弱性診断テストの例外 曜日 年
appid	値リスト	必須。グループのアプリケーションを識別します。以下のいずれかの値でなければなりません。 パブリック 監査プロセス・ビルダー Classifier DB2_zOS グループ エクスプレス・セキュリティー IMS zOS グループ ポリシー・ビルダー セキュリティー・アセスメント・ビルダー
subtype	文字列	オプション。サブタイプは、同じグループ・タイプの複数グループをまとめ、各グループのメンバーシップは排他的になるようにするために使用します。 例えば、データベース・サーバーが 3 つのデータ・センターに配置されていて、サーバーをロケーション別にグループ化するとします。 ロケーションごとにデータベース・サーバーの個別のグループを定義して、3 つのグループすべてに同じサブタイプ (例えば datacenter) を定義できます。
category	文字列	オプション。 category は、ポリシー違反とレポート用グループをグループ化するために使用される、オプション・ラベルです。
classification	文字列	オプション。classification は、ポリシー違反とレポート用グループをグループ化するために使用される、もう 1 つのオプションのラベルです。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例 (記載されている大文字と小文字に厳密に従ってください)

```
grdapi create_group desc=agroup type=OBJECTS  appid=Public owner=admin
```

```
grdapi create_group appid=Access_policy owner=admin  type="OBJECTS"  desc=groupName1
```

list_group_by_id

特定グループのプロパティーを表示します。

パラメーター	値のタイプ	記述
id	integer	必須。グループを識別します。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi list_group_by_id id=100003
```

list_group_by_desc

特定グループのプロパティーを表示します。

パラメーター	値のタイプ	記述
desc		必須。表示されるグループの名前。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi list_group_by_desc desc=agroup
```

delete_group_by_id

パラメーター	値のタイプ	記述
id	integer	必須。グループを識別します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_group_by_id id=100005
```

delete_group_by_desc

パラメーター	値のタイプ	記述
desc	文字列	必須。削除されるグループの名前。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_group_by_desc desc=agroup
```

update_group_by_id

指定されたグループのプロパティーを更新します。

パラメーター	値のタイプ	記述
id	integer	必須。更新するグループを識別します。
newDesc	文字列	オプション。新規グループの固有の記述を入力します。
subtype	文字列	オプション。サブタイプは、同じグループ・タイプの複数グループをまとめ、各グループのメンバーシップは排他的になるようにするために使用します。 例えば、データベース・サーバーが 3 つのデータ・センターに配置されていて、サーバーをロケーション別にグループ化するとします。 ロケーションごとにデータベース・サーバーの個別のグループを定義して、3 つのグループすべてに同じサブタイプ (例えば datacenter) を定義できます。
category	文字列	オプション。 category は、ポリシー違反とレポート用グループをグループ化するために使用される、オプション・ラベルです。
classification	文字列	オプション。classification は、ポリシー違反とレポート用グループをグループ化するために使用される、もう 1 つのオプションのラベルです。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi update_group_by_id id=100002 newDesc=beegroup subtype=bee category=be classification=bea
```

update_group_by_desc

指定されたグループのプロパティーを更新します。

パラメーター	値のタイプ	記述
desc	文字列	必須。更新されるグループの名前。
newDesc	文字列	オプション。グループの固有の記述を入力します。
subtype	文字列	オプション。サブタイプは、同じグループ・タイプの複数グループをまとめ、各グループのメンバーシップは排他的になるようにするために使用します。 例えば、データベース・サーバーが 3 つのデータ・センターに配置されていて、サーバーをロケーション別にグループ化するとします。 ロケーションごとにデータベース・サーバーの個別のグループを定義して、3 つのグループすべてに同じサブタイプ (例えば datacenter) を定義できます。
category	文字列	オプション。 category は、ポリシー違反とレポート用グループをグループ化するために使用される、オプション・ラベルです。
classification	文字列	オプション。classification は、ポリシー違反とレポート用グループをグループ化するために使用される、もう 1 つのオプションのラベルです。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi update_group_by_desc desc=beegroup newDesc=beegroupee category=bebebe classification=bebebebe
```

flatten_hierarchical_groups

グループ・ビルダーに存在するすべての階層グループを更新します。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi flatten_hierarchical_groups
```

create_member_to_group_by_id

グループ ID で指定されたグループにメンバーを追加します。

パラメーター	値のタイプ	記述

パラメーター	値のタイプ	記述
id	integer	必須。メンバーを追加する先のグループを識別します。
member	文字列	必須。新規メンバー名。これはグループ内で固有でなければなりません。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi create_member_to_group_by_id  id=100005 member=turkey
```

create_member_to_group_by_desc

指定されたグループにメンバーを追加します。

パラメーター	値のタイプ	記述
desc	文字列	必須。メンバーを追加する先のグループの名前。
member	文字列	必須。新規メンバー名。これはグループ内で固有でなければなりません。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi create_member_to_group_by_desc  desc=bgroup  member=turkey
```

次のコマンドを使用して、メンバーをグループに追加します。

```
grdapi create_member_to_group_by_desc desc=groupName1 member=member_1
grdapi create_member_to_group_by_desc desc=groupName1 member=member_2
grdapi create_member_to_group_by_desc desc=groupName1 member=member_3
grdapi create_member_to_group_by_desc desc=groupName1 member=member_4
grdapi create_member_to_group_by_desc desc=groupName1 member=member_5
```

追加のグループ GuardAPI コマンド

create_hierarchical_member_to_group_by_desc

delete_hierarchical_member_from_group_by_desc

関数パラメーター:

desc - 文字列 - 必須

member - 文字列 - 必須

list_group members_by_id

指定されたグループのメンバーをリストします。

パラメーター	値のタイプ	記述
id	integer	必須。リストされるメンバーのグループを識別します。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi list_group_members_by_id  id=100001
```

list_group_members_by_desc

指定されたグループのメンバーをリストします。

パラメーター	値のタイプ	記述
desc	文字列	必須。メンバーをリストするグループの名前。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi list_group_members_by_desc  desc=bgroup
```

delete_member_from_group_by_id

指定されたグループから 1 メンバーを削除します。

パラメーター	値のタイプ	記述
id	integer	必須。メンバーが削除されるグループを識別します。

パラメーター	値のタイプ	記述
member	文字列	必須。削除するメンバーの名前。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_member_to_group_by_id  id=100005 member=turkey
```

delete_member_from_group_by_desc

指定されたグループから 1 メンバーを削除します。

パラメーター	値のタイプ	記述
desc	文字列	必須。メンバーが削除されるグループの名前。
member	文字列	必須。削除するメンバーの名前。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_member_from_group_by_desc desc=bgroup member=boston
```

親トピック: GuardAPI

GuardAPI Health 関数

これらの GuardAPI コマンドは、ディスクおよびデータベースの Health Analyzer を構成するために使用します。

enable_health_analyzer

事前定義されたスケジュール時刻に実行されるディスクおよびデータベースの Health Analyzer を開始します。

```
grdapi enable_health_analyzer
```

disable_health_analyzer

ディスクおよびデータベースの Health Analyzer を停止します。

```
grdapi disable_health_analyzer
```

modify_guard_param

以下のパラメーターは、ディスクおよびデータベースの Health Analyzer のアラートを構成するために使用します。

```
modify_guard_param paramName=parameter paramValue=value
```

以下のパラメーターを変更する際は注意が必要です。小さな変更でもアラートのパラダイムが影響を受けます。

パラメーター	値	記述
HEALTH_ANALYZER_DB_LOOKAHEAD_DAYS	integer	今後、HEALTH_ANALYZER_DB_LOOKAHEAD_DAYS で設定された日数以内に、予測された HEALTH_ANALYZER_DB_USAGE_THRESHOLD に到達すると、アラートが送信されます。デフォルトは 14 です。
HEALTH_ANALYZER_DB_SAMPLE_DAYS	integer	今後の使用量を予測するために、データベースの増大をモニターする日数。デフォルトは 7 です。
HEALTH_ANALYZER_DB_USAGE_THRESHOLD	整数。1-100	アラート送信の基準となるデータベース・サイズのしきい値 (%)。デフォルトは 50 です。
HEALTH_ANALYZER_VAR_LOOKAHEAD_DAYS	integer	今後、HEALTH_ANALYZER_VAR_LOOKAHEAD_DAYS で設定された日数以内に、予測された HEALTH_ANALYZER_VAR_USAGE_THRESHOLD に到達すると、アラートが送信されます。デフォルトは 14 です。
HEALTH_ANALYZER_VAR_SAMPLE_DAYS	integer	今後の使用量を予測するために、/var の増大をモニターする日数。デフォルトは 7 です。
HEALTH_ANALYZER_VAR_USAGE_THRESHOLD	整数。1-100	アラート送信の基準となる /var のサイズのしきい値 (%)。デフォルトは 50 です。

get_all_modifiable_guard_params

コマンド modify_guard_param によって変更できるすべてのパラメーター (およびその値) を返します。

```
get_all_modifiable_guard_params
```

例: grdapi get_all_modifiable_guard_params paramlike=health_analyze

get_guard_param

指定されたパラメーターの現行値を取得します。

```
get_guard_param paramName=parameter
```

例: grdapi get_guard_param paramName="HEALTH_ANALYZER_VAR_SAMPLE_DAYS"

親トピック: GuardAPI

GuardAPI 入力生成

GuardAPI 入力生成を使用すると、ユーザーは 1 つの Guardium® レポートの出力を取得して、それを別の Guardium エンティティーへの入力とすることができます。 つまり、ユーザーは準備済みの呼び出しを使用して素早く API の機能を呼び出すことができます。

レポート結果への GuardAPI のマッピング

Guardium には定義済みレポートのバッテリーが付属しています。それらの多くは、構成しやすいように既に GuardAPI 関数にマップされています。 さらに、Guardium では、追加レポートを定義でき、ユーザー独自のカスタム・レポートであっても定義できます。作成したレポートは、レポートごとに GuardAPI 関数にマップできます。

1. 「日次モニター」、「Guardium モニター」、または「TAP モニター」の各タブ内の任意の定義済みレポートに移動します。
2. 「呼び出し ...」ボタンをクリックします。
3. 「API マッピングの追加」の選択項目を選択します。
4. 新規ウィンドウの「API マッピングの追加」には、レポートの名前 (例えば Guardium ログイン)、適切な GuardAPI コマンドを検索するための検索/フィルター・メカニズム、および「定義済みレポート」で使用可能な API 関数の選択項目が表示されます。「API 関数」を選択してから、「レポート属性のマップ」をクリックします。
5. 新規ウィンドウの「API - レポート・パラメーター・マッピング」で、パラメーター名をレポート・フィールドにマップします。Guardium レポートに提供されていないデータがある場合もあります。 このような場合には、定数を作成してレポートに追加し、API パラメーター・マッピング内で使用することができます。
   注: 保存すると、現行のマッピングがオーバーライドされます。
   注: 定数が追加された Guardium レポートをエクスポートしても、その定数はエクスポートされません。

GuardAPI パラメーターと Guardium 属性との間のマッピングを簡素化するために、Guardium には事前定義レポートである「照会エンティティーと属性」が作成されています。このレポートには、Guardium の属性がすべてリストされ、ユーザーに GUI インターフェースが示されるため、ユーザーはレポートから簡単にドリルダウンしてリンケージを素早く作成できます。

既存の Guardium 属性またはユーザー定義の定数は、既存の属性または定数の GuardAPI パラメーターにマップできます。

注: GuardAPI パラメーターがレポート属性にマップされる際に、レポート内で同じ GuardAPI パラメーターに対して複数の属性がマップされている場合、その API 呼び出しで選ばれる値は、レポートの表示順序に従って最初に表示される属性です。

既存の属性

1. 「照会エンティティーと属性」レポートに進み、API パラメーターのマッピングを追加します。(「Guardium モニター」->「照会エンティティーと属性」)
2. 「照会エンティティーと属性」レポートは、Guardium 属性をすべてリストするため長くなります。 「カスタマイズ」ボタンを使用して対象となるレコードを絞り込んでください。
3. マッピングを作成するには、パラメーター名を割り当てる属性行をダブルクリックします。
4. 「呼び出し...」オプションをクリックします。
5. create_api_parameter_mapping API 関数を選択します。
6. 「API 呼び出しフォーム」で functionName と parameterName に入力します。
7. 「今すぐ呼び出し」ボタンをクリックして、API - レポート・パラメーター・マッピングを作成します。

GUI を使用して GuardAPI パラメーターをマップする完全なシナリオについては、How-To トピック『カスタム・レポートからの API 呼び出しの使用』を参照してください。

定数

Guardium レポート内に提供されていないデータもあります。 このような場合には、定数を作成してレポートに追加し、API パラメーター・マッピング内で使用することができます。

1. 「照会エンティティーと属性」レポートに進み、API パラメーターのマッピングを追加します。(「Guardium モニター」->「照会エンティティーと属性」)
2. 「照会エンティティーと属性」レポートは、Guardium 属性をすべてリストするため長くなります。 「カスタマイズ」ボタンを使用して対象となるレコードを絞り込んでください。
3. 定数属性を作成するには、定数属性を作成したいエンティティーの任意の行をダブルクリックします。
4. 「呼び出し...」オプションをクリックします。
5. create_constant_attribute API 関数を選択します。
6. 使用する値を constant に、 付ける名前を attributeLabel に入力します。
7. 「今すぐ呼び出し」ボタンをクリックして定数を作成します。
8. マッピングを作成するには、新しく作成した属性行をダブルクリックします。
9. 「呼び出し...」オプションをクリックします。
10. create_api_parameter_mapping API 関数を選択します。
11. 「API 呼び出しフォーム」で functionName と parameterName に入力します。
12. 「今すぐ呼び出し」ボタンをクリックして、API - レポート・パラメーター・マッピングを作成します。
13. 新しく作成した属性はレポートに追加する必要があります。 「照会 - レポート・ビルダー」で照会を変更し、フィールドを追加します。

詳しくは、『API 呼び出しで定数を使用する方法』を参照してください。このトピックには、GUI を使用して定数属性を作成およびマップする例が示されています。

注: 定数が追加された Guardium レポートをエクスポートしても、その定数はエクスポートされません。
注: API マッピングを使用する場合、レポート内の表の列は、その表の列がエンティティーの属性である限り、レポート・フィールドに表示されます。カウント列などの一部の列は、マップできないため、レポート・フィールドには表示されません。

一部の GuardAPI コマンドのオブジェクト・セキュリティー

ロール検証では、一部の GuardAPI コマンドにコントロールを実装して、特定のコンポーネント (アプリケーションだけでなく) のロールを考慮し、ロールが一致しない場合にアクションを禁止します。

これは、ポリシー・ビルダーに対して適切なロールを持つユーザーであれば、どのポリシーにも (その特定のポリシーのロールに関係なく) GuardAPI コマンド delete_rule を実行できることを意味します。

ポリシー・ルールの GuardAPI コマンド change_rule_order、copy_rule、copy_rules、delete_rule、update_rule に対してロール検証が存在します。

グループの記述 GuardAPI コマンド create_member_to_group_by_desc、create_member_to_group_by_id、delete_group_by_desc、delete_group_by_id、delete_member_from_group_by_desc、delete_member_from_group_by_id、update_group_by_id、update_group_by_desc に対してロール検証が存在します。

データ・ソース GuardAPI コマンド delete_datasource_by_id、delete_datasource_by_name、update_datasource_by_id、update_datasource_by_name に対してロール検証が存在します。

監査プロセス GuardAPI コマンド stop_audit_process に対してロール検証が存在します。

表形式レポートおよびグラフィカル・レポートから監査プロセスを実行する API

GuardAPI は、どのレポート・ポートレットからでも自動的に呼び出すことができます。 GuardAPI は呼び出されると新しい監査プロセス・レポートを作成します。

ユーザーに対してそのようなプロセスが存在している場合、パラメーターが更新され、同じプロセスが使用されます。

GuardAPI の振る舞いは以下のようになります。

1 - 新しいプロセスの場合、リスト内に emailContentType パラメーターで示されているコンテンツ・タイプの E メールがあるなら、その E メールごとにレシーバーを 1 つ作成します。また、includeUserReceiver パラメーターが true の場合は、(API を呼び出して) ログインしているユーザーのためのユーザー・レシーバーも作成します。

2 - 既存のプロセスの場合は、すべての E メール・レシーバーが削除され、emailContentType パラメーターに定義されているコンテンツ・タイプで、新規リスト (存在する場合) の E メールに置き換えられます。 リストが空の場合は、E メール・アドレス・レシーバーがすべて削除されます。 ユーザーのレシーバーが既に存在する場合は、includeUserreceiver が false でもそれは削除されませんが、このパラメーターが true で、かつそのようなレシーバーが存在しない場合は、追加されます。

監査プロセスが生成されると、これは (「今すぐ 1 回実行」と同じように) 自動的に実行され、ユーザーはその監査プロセスが自分の To-Do リスト上のアイテムとなることを期待します。

create_ad_hoc_audit_and_run_once

パラメーター:

1 - reportId - 監査プロセスの唯一のタスクに使用される、レポートの ID

2 - isForReportRunOnce - このブール値は、そのプロセスが作成後に 1 回実行する必要があるかどうかを示します。

3 - changeParIfExist ブール値は、プロセスが存在する場合に、タスク・パラメーターを更新するかどうかを示します

4 - taskParameter - それぞれが文字列 ^^ で連結されたすべてのタスク・パラメーターと値は、PAR1=Val1^^PAR2=Val2^^ のようになります。パラメーターを空のままにしても有効です。例えば、PAR2 が空のままにする場合、PAR1=VAL1^^PAR2=^^PAR3=VAL3^^... のようになります。

5 - processNamePar - プロセスの名前。空のときは、名前を持つプロセスが作成されます。

6 - sendToEmails: E メール・アドレスのコンマ区切りリスト

7 - emailContentType 0-PDF または 1-CSV (E メール・レシーバーにのみ適用)

8 - includeUserReceiver ブール値は、ログインしているユーザーのレシーバーを作成するかどうかを示します。

GuardAPI は、どのレポート・ポートレットからでも自動的に呼び出すことができます。GuardAPI は呼び出されると新しい監査プロセス・レポートを作成します。

スケジュール API

modify_schedule パラメーター jobName jobGroup cronString startTime オプション

list_schedule

list_scheduler_jobs

delete_schedule パラメーター jobName jobGroup deleteJob オプション

schedule_job パラメーター jobType objectName optional cronString startTime オプション

注: grdapi schedule_job 関数の一部のジョブ・タイプでは、オブジェクト名は必要ありません。特定のジョブ・タイプ (csvExportJob、systemBackupJob、dataArchiveJob、dataExportJob、dataImportJob、resultsArchiveJob、AppUserTranslation、IpHostToAlias) について objectName パラメーターとして入力した内容を使用してこの関数が実行された場合、オブジェクト名パラメーターに対して検証は実行されず、標準的な「OK」プロンプトが表示されます。

grdapi schedule_job --get_param_values=jobType - 関数「schedule_job」のパラメーター「jobType」の値は、 EagleEyeJob、AuditJob、stapF5Correlation、IpHostToAlias、customAlerting、dataMartExtraction、CustomTableDataUpload、 checkDb2IMonitor、analyticJob、DataExport、mustGather、userSynchronization、InstallPolicy、DataImport、 AppUserTranslation、purgeJob、ResultArchive、customTableStapAssocication、execute_incidentGenProcess、 stapVerification、SystemBackup、UnitUtilization、MonitorValues、secureParamCorrelation、HealthSpaceJob、 updateStapChange、updateToDo、CustomTableDataPurge、flattenHGroups、ReplayConfig、PopulateAlias、 AutoDetectScanJob、cyberRank_fetch、populateAccess、AutoDetectProbeJob、PopulateGrpFromQry、DataArchive、 CSVExport、distributedReportExtraction、LdapImport、FAM、FlatLog、connectivityMonitor のいずれかでなければなりません。

set_purge_batch_size

パージ中に使用されるバッチ・サイズを設定します。このバッチ・サイズはパージのパフォーマンスに貢献し、デフォルト設定は 200,000 です。パフォーマンスとディスク・スペース使用量とのトレードオフには注意が必要です。大きなバッチ・サイズを設定すると、パージの速度は上がりますが、より多くのディスク・スペースが消費されます。小さいバッチ・サイズを設定すると、パージの速度は下がりますが、それほど多くのディスク・スペースを消費することはありません。

関数パラメーター : batchSize - 必須 api_target_host 例 vx29> grdapi set_purge_batch_size batchSize=200000 ID=0 ok

get_purge_batch_size

パージ・バッチ・サイズの現在の設定を取得します

関数パラメーター : api_target_host 例 vx29> grdapi get_purge_batch_size ID=0 パージ・バッチ・サイズ = 200000 ok

patch_install

関数パラメーター : patch_date patch_number - 必須

populate_from_dependencies

関数パラメーター : descOfEndingGroup - 必須 descOfStartingGroup - 必須 flattenNamespace getFunctions getJavaClasses getPackages getProcedures getSynonyms getTables getTriggers getViews isAppend - 必須 isEndingGroupQualified owner - 必須 reverseIt selectedDataSourceName - 必須 api_target_host

create_computed_attribute

レポートで使用。

パラメーター	値のタイプ	記述
attributeLabel	文字列	必須。
entityLabel	文字列	必須。データベース・ユーザー
expression	文字列	必須。サーバー IP。ユーザーは、式の中で tableName.field を指定する必要があります。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

grdapi create_computed_attribute attributeLabel="CustomUserName" entityLabel="App User Name"
expression="SUBSTRING_INDEX(GDM_CONSTRUCT_INSTANCE.APP_USER_NAME,'.',1)"

delete_computed_attribute

レポートで使用。

パラメーター	値のタイプ	記述
attributeLabel	文字列	必須。
entityLabel	文字列	必須。
expression	文字列	必須。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

update_computed_attribute

レポートで使用。

パラメーター	値のタイプ	記述
attributeLabel	文字列	必須。
entityLabel	文字列	必須。
expression	文字列	必須。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

create_constant_attribute

レポートで使用。

パラメーター	値のタイプ	記述
attributeLabel	文字列	必須。
entityLabel	文字列	必須。
constant	文字列	必須。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

delete_constant_attribute

レポートで使用。

パラメーター	値のタイプ	記述
attributeLabel	文字列	必須。
entityLabel	文字列	必須。
constant	文字列	必須。

パラメーター	値のタイプ	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

update_constant_attribute

レポートで使用。

パラメーター	値のタイプ	記述
attributeLabel	文字列	必須。
entityLabel	文字列	必須。
constant	文字列	必須。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

create_ad_hoc_audit_and_run_once

レポートで使用。

パラメーター	値のタイプ	記述
chnageParlfExist	ブール値	必須。
emailContentType	integer	
includeUserReceiver	ブール値	
isForReportRunOnce	ブール値	必須。

パラメーター	値のタイプ	記述
processNamePar	文字列	
reportID	integer	必須
sendToEmails	文字列	
taskParameter	文字列	
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

REST API

JSON (JavaScript Object Notation) 出力オプションは、GuardAPI 関数をサポートします。これは、REST API の一部です。REST は Representational State Transfer の略です。ステートレスのクライアント/サーバーのキャッシュ可能通信プロトコルを利用します。また、ほとんどすべての場合に、HTTP プロトコルが使用されます。REST は、ネットワーク・アプリケーションを設計するためのアーキテクチャー・スタイルです。マシン間の接続に CORBA、RPC、SOAP などの複雑なメカニズムを使用するのではなく、単純な HTTP を使用してマシン間で呼び出しを実行するという考えに基づいています。RESTful アプリケーションでは、HTTP 要求を使用して、データのポスト (作成/更新)、データの読み取り (例えば、照会)、およびデータの削除を行います。そのため、REST では、作成/読み取り/更新/削除の 4 つの操作すべてで HTTP を使用します。REST は、RPC (リモート・プロシージャー・コール) や Web サービス (SOAP、WSDL) などのメカニズムの代わりとなる軽量なメカニズムです。

Guardium REST API の実装

1. アプリケーションを (1 回だけ) 登録し、クライアント・パスワードを取得します。
2. クライアント・パスワードを安全な場所に保管します。
3. 許可のためにアクセス・トークンを要求します。
4. grdAPI コマンドが正しく認証されるように、アクセス・トークンを保管します。
5. アクセス・トークンを使用して GuardAPI コマンドを実行依頼する。

ユース・ケースの例

- Guardium GUI にログインすることなく、特定の IP アドレスの少量の監査データを動的に取得できるようにしたい。
- ポリシーを更新して機密情報への無許可アクセスを防止できるように、既存グループにデータを取り込みたい。
- 特定の許可されたアクセス・グループ内のすべてのユーザーのリストを取得したい。
- モニターする必要がある機密表をアプリケーション開発チームが特定できるようにしたい。
- ターゲット・システムからの応答テキストのコーディングをユーザーに求める、「要求する」スクリプト言語を使用せずに grdAPI にスクリプトでアクセスしたい。

HTTP には操作のボキャブラリー (要求メソッド) があります。

- GET (URL でパラメーターを渡す)
- POST (JSON オブジェクトでパラメーターを渡す)
- PUT (変更するパラメーターを JSON オブジェクトとして渡す)
- DELETE (JSON オブジェクトとしてパラメーターを渡す)

内部 REST API 要求の特殊ユーザー

内部 REST API 要求用に、システム内に特殊なロールとユーザーが事前定義されています。

このユーザーは、accessmgr UI を使用して削除および変更できず、UI でのログインに使用できません。

このユーザーのパスワードは、有効期限切れとなることはありませんが、クライアント ID が取り消された場合は取り消されます。

OAuth クライアント登録時に、新しい関数がこのユーザーとクライアント ID を受け入れます。これは、当該ユーザーに対してランダムで強固なパスワードを生成し、TURBINE_USER 表に保管します。

これは、クライアント・パスワードと、生成されたパスワードを返します。

内部 (S-TAP など) のクライアントでは、クライアント・パスワードとパスワードを保護する必要があります。

accessmgr UI を使用して、各種関数に対する許可をロールに割り当てることができます。

RestAPI と GuardAPI の比較

GET = List

POST = Create

PUT = Update

DELETE = Delete

GuardAPIs

```
list_datasourcename_by_name (parameters - ?name="MSSQL_1")
```

```
-X GET https://10.10.9.239:8443/restAPI/datasource/?name="MSSQL_1"
```

create_datasource

```
-X POST https://10.10.9.239:8443/restAPI/datasource
```

```
update_datasource_by_name - JSON Object '{password:guardium}'
```

```
-X PUT -d '{password:guardium, name:"MSSQL_1}'
```

```
delete_datasource_by_id - JSON Object '{"id":20020}'
```

```
-X DELETE -d '{"id":20020}'
```

詳しくは、DeveloperWorks の記事『Using the IBM Security Guardium REST API』を参照してください。

http://www.ibm.com/developerworks/data/library/techarticle/dm-1404guardrestapi/index.html

get_unit_pinger

この GuardAPI コマンドは、内部 UnitPinger スレッドを照会および再始動するために使用します。

注: この GuardAPI コマンドは、api_target_host=127.0.0.1 パラメーターを使用して呼び出す必要があります。

例

```
grdapi get_unit_pinger api_target_host=127.0.0.1
```

register_oauth_client

この GuardAPI コマンドは、サポートされる GuardAPI 関数を、入出力で JSON (JavaScript Object Notation) を使用する RESTful API にラップする場合に使用します。

GrdAPI コマンド grdapi register_oauth_client は、クライアントを登録し、REST サービスの呼び出しに必要なアクセス・トークンを取得する場合に使用します。

REST は Representational State Transfer の略です。ステートレスのクライアント/サーバーのキャッシュ可能通信プロトコルを利用します。また、ほとんどすべての場合に、HTTP プロトコルが使用されます。

REST は、ネットワーク・アプリケーションを設計するためのアーキテクチャー・スタイルです。マシン間の接続に CORBA、RPC、SOAP などの複雑なメカニズムを使用するのではなく、単純な HTTP を使用してマシン間で呼び出しを実行するという考えに基づいています。

RESTful アプリケーションでは、HTTP 要求を使用して、データのポスト (作成/更新)、データの読み取り (例えば、照会)、およびデータの削除を行います。そのため、REST では、作成/読み取り/更新/削除の 4 つの操作すべてで HTTP を使用します。REST は、RPC (リモート・プロシージャー・コール) や Web サービス (SOAP、WSDL) などのメカニズムの代わりとなる軽量なメカニズムです。

関数パラメーター:

client_id - 文字列 - 必須

grant_types - 文字列 - 必須。サポートされる唯一の権限付与タイプは password です。

redirect_uris - 文字列 - 必須

scope - 文字列 - 必須

fetchSize - 文字列 - オプション。後方互換性を維持するために、デフォルトは 20 レコードです。最大値は 30000 です。

sortColumn - オプション - 指定する場合、いずれかのレポート・フィールドの列タイトルでなければなりません。

sortType - オプション - asc または desc。

構文

grdapi register_oauth_client <client_id> <grant_types> <redirect_uris> <scope>

getOAuthTokenExpirationTime

この GuardAPI コマンドは、REST API トークンの有効期限を取得する場合に使用します。

関数パラメーター:

api_target_host - 文字列

setOAuthTokenExpirationTime

この GuardAPI コマンドは、REST API トークンの有効期限を設定する場合に使用します。

関数パラメーター:

expirationTime - 整数 - 必須。

api_target_host - 文字列

構文

grdapi setOAuthTokenExpirationTime ExpirationTime=10000

親トピック: GuardAPI

GuardAPI 調査ダッシュボード機能

これらの GuardAPI コマンドは、調査ダッシュボードの機能とパラメーターを有効化、無効化、または構成するために使用します。

調査ダッシュボードには、「クイック検索結果表 (Quick Search Results Table)」、「アクティビティー・グラフ」、およびその他のさまざまな事前定義グラフが含まれていることに注意してください。

disable_quick_search

調査ダッシュボード機能を無効にします。

```
grdapi disable_quick_search
```

パラメーター	値	記述
すべて	`true` または `false`	中央マネージャーがある環境では、このパラメーターを使用してすべての管理対象ユニットでの検索を無効にします。例えば、`all=true` です。 このパラメーターはオプションです。
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

enable_quick_search

調査ダッシュボード機能を有効にします。

```
grdapi enable_quick_search schedule_interval=[value] schedule_units=[value]
```

例えば、以下のコマンドは、調査ダッシュボードを 2 分間のデータ抽出間隔で有効にします: `grdapi enable_quick_search schedule_interval=2 schedule_units=MINUTE`。

パラメーター	値	記述
すべて	`true` または `false`	中央マネージャーがある環境では、このパラメーターを使用してすべての管理対象ユニットでの検索を有効にします。例えば、`all=true` です。 このパラメーターはオプションです。
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。
extraction_start	date	検索用監査データ抽出の開始期限の日付を定義します。このパラメーターを省略した場合、抽出は即時に開始されます。 このパラメーターはオプションです。
includeViola	`true` または `false`	検索索引に違反を含めるかどうかを決定します。違反を省略すると、検索索引のサイズを削減できます。

tions		このパラメーターはオプションです。
schedule_interval	integer	schedule_units パラメーターとともに使用して、監査データの抽出の間隔を定義します。例えば、`schedule_interval=2 schedule_units=MINUTE` です。 このパラメーターは必須です。
schedule_start	date	schedule_interval パラメーターと schedule_units パラメーターによって定義された抽出間隔の後に開始する日付。 このパラメーターはオプションです。
schedule_units	`HOUR` または `MINUTE`	schedule_interval パラメーターとともに使用して、監査データの抽出の間隔を定義します。例えば、`schedule_interval=2 schedule_units=MINUTE` です。 このパラメーターは必須です。

set_enterprise_search_options

調査ダッシュボードの検索モードを定義します。

```
grdapi set_enterprise_search_options distributed_search=[value]
```

例えば、以下のコマンドは、all_machines モードで調査ダッシュボードを構成して、Guardium 環境全体にわたるデータの検索を、その環境内の任意の Guardium マシンから実行できるようにします。`grdapi set_enterprise_search_options distributed_search=all_machines`

パラメーター	値	記述
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。
distributed_search	cm_only、local_only、または all_machines	cm_only 中央マネージャーから実行依頼された検索では、Guardium 環境全体からの結果が返されますが、管理対象ユニットから実行依頼された検索では、その管理対象ユニットからのローカルの結果のみが返されます。 local_only 個々のマシンから実行依頼された検索では、そのマシンからの結果のみが返されます。Guardium 環境全体からデータを検索することはできません。 all_machines 任意のマシンから検索を実行依頼でき、Guardium 環境全体からの結果が返されます。 このパラメーターは必須であり、デフォルト値は cm_only です。

親トピック: GuardAPI

GuardAPI ネイティブ監査関数

これらの GuardAPI コマンドを使用して、クラウド・データベースに対する DB 監査 (ネイティブ監査) の有効化、無効化、オブジェクト監査 (監査証跡) に対するオブジェクトの追加と削除、構成、コレクター、およびオブジェクトの取得を実行します。

注: v10.1.4 で、GuardAPI Native Audit 関数が導入されました。

add_ip_to_sg

指定した Guardium IP をクラウド・セキュリティー・グループに追加します。

```
add_objects_native_audit parameter=value
```

パラメーター	値	記述
datasource_name	文字列。	必須。Guardium で定義されているクラウド・データ・ソース
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

add_objects_native_audit

指定したデータ・ソースのオブジェクト監査 (監査証跡) にオブジェクトを追加します。

`add_objects_native_audit parameter=value`

パラメーター	値	記述
datasource_name	文字列	必須。Guardium で定義されているクラウド・データ・ソース
objects	文字列。	オブジェクトのコンマ区切りリスト。オブジェクトの表示は、get_native_audit_objects または GUI で行います。
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

disable_native_audit

指定したクラウド・データ・ソースの DB 監査 (ネイティブ監査) を無効にします。

`disable_native_audit parameter=value`

パラメーター	値	記述
datasource_name	文字列	必須。Guardium で定義されているクラウド・データ・ソース
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

enable_native_audit

指定したデータ・ソースの DB 監査 (ネイティブ監査) を有効にします。

`enable_native_audit parameter=value`

パラメーター	値	記述
datasource_name	文字列	必須。Guardium で定義されているクラウド・データ・ソース
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

get_native_audit_collectors

環境内の、つまり指定したホスト、ポート、およびサービス名からデータを受信するコレクターの名前を返します。

`get_native_audit_collectors parameter=value`

パラメーター	値	記述

host	文字列	必須。AWS ホスト名
port	integer	必須。
service_name	文字列	必須。
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

get_native_audit_configurations

指定したホスト、ポート、サービス名のすべての詳細 (クラウド環境 ID、クラウド環境、プロバイダー、データ・ソース ID、インスタンス名、データベース・エンジン、サービス名、ホスト、ポート、Guardium セキュリティー・グループ、オブジェクト制限、オブジェクト、コレクター) を返します。

`get_native_audit_configurations parameter=value`

パラメーター	値	記述
host	文字列	必須。AWS ホスト名
port	integer	必須。
service_name	文字列	必須。
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

get_native_audit_objects

指定したホスト、ポート、サービス名上の分類プロセスによって検出されたすべてのオブジェクトを返します。

`get_native_audit_objects parameter=value`

パラメーター	値	記述
host	文字列	必須。AWS ホスト名
port	integer	必須。
service_name	文字列	必須。
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

remove_objects_native_audit

指定したデータ・ソース内の指定したオブジェクトのオブジェクト監査 (監査証跡) を無効にします。

`remove_objects_native_audit parameter=value`

パラメーター	値	記述

datasource_name	文字列	必須。Guardium で定義されているクラウド・データ・ソース
objects	文字列	オブジェクトのコンマ区切りリスト。オブジェクトの表示は、get_native_audit_objects または GUI で行います。
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

親トピック: GuardAPI

GuardAPI 異常値検出機能

以下の GuardAPI コマンドは、異常値検出機能を有効化、無効化、および構成するために使用します。

enable_outliers_detection

grdapi enable_outliers_detection

grdapi disable_outliers_detection

このコマンドは、すべての Guardium システム (中央マネージャー、アグリゲーター、スタンドアロン) で使用できます。

- スタンドアロンの Guardium でのローカルの異常値検出を有効化/無効化するために、コレクターで実行します。
- 中央マネージャーで次のように実行します。
  - パラメーターなしで、CM のすべての管理対象ユニット上で有効化/無効化するため。
  - group_description パラメーターを使用して、グループのすべての管理対象ユニット上で有効化/無効化するため。
  - managed_units_hostnames パラメーターを使用して、指定した管理対象ユニット上で有効化/無効化するため。

パラメーター	有効な値	記述
DAM_FAM	DAM または FAM	オプション。異常値のタイプを指定します。デフォルトは DAM です。
extraction_start	形式 yyyy-mm-dd hh:mm:ss の日付	データ抽出の開始を遅らせる場合に使用します。指定しないときは、データ抽出が即時開始されます。
group_descriptions	グループ ID	コマンドの実行対象の特定のグループ。(グループは、「管理」 > 「一元管理」 > 「管理対象ユニット・グループ」で定義されます)CM 上で API を実行するときはオプション。
managed_units_hostnames	ホスト名のコンマ区切りリスト	コマンドの実行対象の特定の管理対象ユニット。CM 上で API を実行するときはオプション。
schedule_interval	1	無視されます。
schedule_units	hour	無視されます。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

enable_outliers_detection_agg

grdapi enable_outliers_detection_agg

grdapi disable_outliers_detection_agg

CM で実行され、指定されたアグリゲーターにデータを送信するコレクターで異常値検出を有効化/無効化します。

パラメーター	有効な値	記述
aggregator_host_name	文字列	必須。
DAM_FAM	DAM または FAM	オプション。異常値のタイプを指定します。デフォルトは DAM です。

パラメーター	有効な値	記述
extraction_start	形式 yyyy-mm-dd hh:mm:ss の日付	データ抽出の開始を遅らせる場合に使用します。指定しないときは、データ抽出が即時開始されます。
schedule_interval	1	無視されます。
schedule_units	hour	無視されます。

enable_outliers_detection_cross_cm_agg

grdapi enable_outliers_detection_cross_cm_agg

grdapi disable_outliers_detection_cross_cm_agg

CM で実行され、指定されたアグリゲーターにデータを送信するコレクターで異常値検出を有効化/無効化します。これは複数 CM 環境 (アグリゲーターはこの CM の環境内にあるが、コレクターは別の CM が管理している) で使用するためのものです。

パラメーター	有効な値	記述
aggregator_host_name	文字列	必須。
DAM_FAM	DAM または FAM	オプション。異常値のタイプを指定します。デフォルトは DAM です。
extraction_start	形式 yyyy-mm-dd hh:mm:ss の日付	データ抽出の開始を遅らせる場合に使用します。指定しないときは、データ抽出が即時開始されます。
schedule_interval	1	無視されます。

disable_outliers_detection_cross_cm_collector

grdapi enable_outliers_detection_cross_cm_collector

grdapi disable_outliers_detection_cross_cm_collector

CM で実行され、この CM の環境内の指定されたコレクターで異常値検出を有効化/無効化します。これは複数 CM 環境 (コレクターはこの CM の環境内にあるが、データの送信先アグリゲーターは別の CM の環境内にある) で使用するためのものです。

パラメーター	有効な値	記述
collector_host_names	文字列	必須。

set_outliers_detection_parameter

`set_outliers_detection_parameter parameter=`*`value`*

注: 異常値のマイニングに精通したユーザーとともに作業する場合以外は、デフォルトを変更しないでください。

パラメーター	有効な値	記述
cleanupKeepDays	integer	コレクター上にモデル・データを保持する日数。デフォルトは 90 日です。
sensitiveObjectGroupIds	オブジェクト・グループ ID (数値) のコンマ区切りリスト	異常値検出アルゴリズムに別のオブジェクト・グループ (表、ビューなど) を追加します。次のコマンドは、グループ ID を検出するために使用します。grdapi list_group_by_desc desc=[group name]
privUsersGroupIds	ユーザー・グループ ID (数値) のコンマ区切りリスト	追加のユーザー・グループを異常値検出アルゴリズムに追加します。次のコマンドは、グループ ID を検出するために使用します。grdapi list_group_by_desc desc=[group name]
minDaysForAlerts	integer	異常値アラートの生成に必要なアクティビティーの日数。デフォルトは 7 です。 このパラメーターの値は、パラメーター budgetTrainingDays を超えないようにする必要があります。超える場合はエラーが発生します。
maxMessageAlertsTopScores	integer	大量に発生したスコアリング・メッセージ・アラートのうち、異常の概要行に表示される数。デフォルトは 20 です。「異常値の概要」行には、いくつかの「詳細」行が含まれます。これらの行は、その時間内に発生した異常のサンプルです。「大量に発生した」異常の詳細は、異常スコアが最も高かった X 行となります。X はこのパラメーターの値です。
maxMessageAlertsSampleSizePerAlertType	integer	これは大量でない異常に適用されます。これは概要アラートに表示される異常サンプルの数です。大量でない異常のサンプル行は Y 行となります。これらの異常ではスコアが問題とならないため、サンプル行は順不同に表示されます (オブジェクト間で新旧の差はない)。Y はこのパラメーターの値です。デフォルトは 5 です。

パラメーター	有効な値	記述
alertsPerDay	integer	1 日に生成される異常値 (概要レベル) の数。デフォルトは 24 です。このパラメーターは、アラートの入力を制御するために使用され、セキュリティー・アナリストが管理できます。これによって数が制御され、スコアが最も高いものが示されます。この数値は、budgetTrainingDays パラメーターで設定された過去の日数 (14 日間など) から得られた統計を基準として適用されます。このプロセスでは、異常がこの数前後になるしきい値の追加分 (常に intervalAlertsThreshold を上回る) が計算されます。重要: • 1 日の異常数が alertsPerDay を下回る場合もあります (単純にその日に発生した異常がそれほど多くなかった場合)。 • 異常が多く発生した場合は、異常数の制限はなくなり、すべての異常が示されます。これは、割り当て量を適用したために差し迫った状況が通知されなくなることを防ぐためです。
budgetTrainingDays	integer	システムが学習のために振り返りを行う日数。デフォルトは 14 です。
intervalAlertsThreshold		異常スコアがこのしきい値を超えると、そのスコアが異常値として報告されます。デフォルトは 0.99 です。この値を小さくすると、システムの異常に対する感度が高くなります。スコアが比較的低い異常も異常値として報告され、誤検出が増える可能性があります。この値を大きくすると、システムの異常に対する感度が低くなります。
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

get_outliers_detection_info

`get_outliers_detection_info`

異常値パラメーターの工場出荷時設定と現在の設定をリストします。

パラメーター	有効な値	記述
api_target_host	文字列	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

親トピック: GuardAPI

GuardAPI プロセス制御関数

これらの GuardAPI コマンドは、プロセス制御関数の実行、コピー、アップロード、リスト、および削除に使用します。

execute_cls_process

分類プロセスを実行 (サブミット) します。 分類プロセス・ビルダーから「今すぐ 1 回実行」を実行することに相当します。 これは、Guardium® ジョブ・キューにプロセスを配置するジョブをサブミットします。このキューからアプライアンスは一度に 1 つのジョブを実行します。 管理者は、「Guardium モニター」>「Guardium ジョブ・キュー」と選択することによりジョブ状況を表示できます。

注: この API を呼び出す前に分類プロセスを作成してください。

パラメーター	値	記述
processName	文字列	分類プロセスの名前

パラメーター	値	記述
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • all_managed: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • all: すべての管理対象ユニットと中央マネージャーで実行します • group:*<group name>*: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば api_target_host=10.0.1.123 です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば api_target_host=10.0.1.123 です。

例

```
grdapi execute_cls_process processName="classPolicy1"
```

execute_assessment

セキュリティー・アセスメントを実行 (サブミット) します。セキュリティー・アセスメント・ファインダーから「今すぐ 1 回実行」を実行することに相当します。 ジョブがサブミットされます。 これによって、Guardium ジョブ・キューにプロセスが配置され、このキューからアプライアンスは一度に 1 つのジョブを実行します。管理者は、「Guardium モニター」>「Guardium ジョブ・キュー」と選択することによりジョブ状況を表示できます。

注: この API を呼び出す前にセキュリティー・アセスメントを作成してください。

パラメーター	値	記述
assessmentDesc		評価の名前
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • all_managed: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • all: すべての管理対象ユニットと中央マネージャーで実行します • group:*<group name>*: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば api_target_host=10.0.1.123 です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば api_target_host=10.0.1.123 です。

例

```
grdapi execute_assessment assessmentDesc="assessment1"
```

execute_auditProcess

監査プロセスを実行します。 指定された監査プロセスを実行します。 監査プロセス・ビルダーから「今すぐ 1 回実行」を実行することに相当します。

注: この API を呼び出す前に監査プロセスを作成してください。
注: 監査レポートによって多くのデータが返される場合、CLI コマンドのヒープ・サイズ制限のため、ユーザーは GUI から監査プロセスを実行する必要があります。

パラメーター	値	記述
auditProcess		監査プロセスの名前
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • all_managed: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • all: すべての管理対象ユニットと中央マネージャーで実行します • group:*<group name>*: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば api_target_host=10.0.1.123 です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば api_target_host=10.0.1.123 です。

例

```
grdapi execute_auditProcess  auditProcess="Appliance Monitoring"
```

stop_audit_process

stop_audit_process API は GuardAPI コマンド行からは使用できません。この関数はドリルダウンからの呼び出しとしてのみ使用可能です。 Workload Automation ヘルプ・トピック『コンプライアンス』のサブトピック『監査プロセスの停止』を参照してください。

パラメーター	値	記述
process		監査プロセスの名前
run		監査プロセスの RunID
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
stop_audit_process
```

execute_populateGroupFromQuery

照会からグループに取り込みを実行します。構成された照会を実行することで選択されているグループを取り込みます。 これは、「「照会からグループに取り込み」の設定」画面から「今すぐ 1 回実行」を実行することに相当します。グループがインポート用に構成されていない場合、エラー・メッセージが表示されます。

注: この grdapi は、「照会設定からのグループに取り込み」画面で既に構成されているグループに対してのみ使用できます (照会が選択されていて、パラメーターが設定されていなければなりません)。

パラメーター	値	記述
groupDesc		グループ名
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi execute_populateGroupFromQuery  groupDesc="A test"
```

execute_appUserTranslation

アプリケーション・ユーザー・トランスレーションを実行します。「アプリケーション・ユーザー・トランスレーション構成」画面で構成済みのすべてのアプリケーションのユーザー定義をインポートします。 これは「アプリケーション・ユーザー・トランスレーション構成」画面から「今すぐ 1 回実行」を実行することに相当します。

注: この grdapi を実行するには、「アプリケーション・ユーザー・トランスレーション構成」画面で「アプリケーション・ユーザー検出」を 1 つ以上定義する必要があります。 定義しないと、メッセージが表示されます。

パラメーター	値	記述
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi execute_appUserTranslation
```

execute_flatLogProcess

未解析ログ情報を内部データベースにマージします。 「未解析ログ処理」画面から「今すぐ 1 回実行」を実行することに相当します。

注: この grdapi は、「未解析ログ処理」画面で「未解析ログ処理」が「処理」として構成されている場合のみ実行できます。 そうでない場合、エラー・メッセージが表示されます。

パラメーター	値	記述
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi execute_flatLogProcess
```

get_flatLogProcessType

現在の未解析ログ・ファイルの processType を表示します。

パラメーター	値	記述
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi get_flatLogProcessType
```

結果:

```
ID=1
FLAT_LOG_PROCESS_TYPE:1 - Process
```

set_flatLogProcessType

未解析ログ・ファイルの動作を設定します。

パラメーター	値	記述
processType	文字列(必須)	このログ・ファイルのタイプおよびアクション。オプションは、以下のとおりです。 • `DEFAULT` - オプションは選択されません。 • `PROCESS` - 未解析ログ情報を内部データベースにマージします。 • `ARCHIVE_AGGREGATE_PURGE` - 未解析ログをアーカイブまたは集約し、オプションでパージします。 • `PURGE_ONLY` - 未解析ログ・データをパージします。
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi set_flatLogProcessType processType="PROCESS"
```

execute_incidentGenProcess

ポリシー違反ログに対し、processId を使用して、選択されたインシデント生成プロセスに定義されている照会を実行します。その照会に基づいてインシデントが生成されます。 「インシデント生成プロセスの編集」画面から「今すぐ 1 回実行」を実行することに相当します。

注: この API を呼び出す前にインシデント生成プロセスを作成してください。

パラメーター	値	記述
processID		インシデントのプロセス ID
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi execute_incidentGenProcess   processId=20003
```

execute_incidentGenProcess_byDetails

ポリシー違反ログに対し、照会名を使用して、選択されたインシデント生成プロセスに定義されている照会を実行します。 その照会に基づいてインシデントが生成されます。 「インシデント生成プロセスの編集」画面から「今すぐ 1 回実行」を実行することに相当します。

注: この API を呼び出す前にインシデント生成プロセスを作成してください。

パラメーター	値	記述
queryName		照会名
categoryName		カテゴリー名
user		ユーザー
threshold		しきい値
severity		重大度レベル
api_target_host	ホスト名または IP アドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi  execute_incidentGenProcess_byDetails  queryName="Policy Violation Count" user=admin severity=info
```

upload_custom_data

分類プロセスを実行 (サブミット) します。tableName で指定されたカスタム表にデータをアップロードします。 これは、カスタム表ビルダーの「データのインポート (Import data)」画面から「アップロード」を実行することに相当します。この grdapi を実行するには、カスタム表ビルダーの「表構造のインポート」画面で、指定されたカスタム表を構成しておく必要があります。 UI から「レポート」/「レポート構成ツール」/「カスタム表ビルダー」と移動し、カスタム表を選択して「データのアップロード」をクリックし、データ・ソースを選択します。

パラメーター	値	記述
tableName	既存のカスタム表	カスタム表の名前

パラメーター	値	記述
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi upload_custom_data tableName="TEST_TABLE"
```

execute_ldap_user_import

LDAP ユーザーをインポートします。「LDAP ユーザーのインポート」画面で構成されている LDAP サーバーから Guardium ユーザー定義をインポートします。 「LDAP ユーザーのインポート」画面から「今すぐ 1 回実行」を実行することに相当します。 (accessmgr としてログインし、「LDAP インポート」を選択します)

注: LDAP を構成する必要があります。構成していないと、エラー・メッセージが表示されます。

パラメーター	値	記述
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi execute_ldap_user_import
```

policy_install

ポリシーを 1 つ、または複数インストールします。 複数のポリシーをインストールする場合、インストールしたい順番でポリシーを指定して、パイプ文字「|」で区切る必要があります。 これは 1 つのポリシーしか変更していない場合でも行う必要があります。

複数のポリシーをインストールする場合は、grdapi policy_install コマンドを使用します。 インストールしたい順番でポリシーを指定して、位置ごとにインストールします。

UI の場合であっても、別のインストール済みポリシーの後にポリシーをインストールすると、それらはすべて再インストールされますが、これは grdapi policy_install コマンドの場合と同じです。

パラメーター	値	記述
policy		ポリシー名
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi policy_install policy="Policy 1|Policy 2"
```

```
grdapi policy_install policy="policy 20|policy 30|policy 40"
```

delete_policy

delete_policy コマンドは、policyDesc パラメーターで指定したポリシーを削除する場合に使用します。

パラメーター	値	記述
policyDesc		ポリシー名。
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_policy policyDesc="Hadoop Policy"
```

list_policy

list_policy コマンドは、使用可能なポリシーのリストを表示する場合、または単一のポリシーに関する詳細を表示する場合に使用します。

パラメーター	値	記述
policyDesc		ポリシー名。未指定の場合、list_policy コマンドは、使用可能なポリシーのリストを返します。
detail		値 true または false を受け入れます。 デフォルト値は true で、ポリシーの詳細を返します。 値 false を指定すると、ポリシー名のみが返されます。
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

特定のポリシーの詳細を表示するには、以下のようにします。

```
grdapi list_policy policyDesc="Hadoop Policy"
```

使用可能なポリシーの詳細リストを表示するには、以下のようにします。

```
grdapi list_policy
```

詳細なしで、使用可能なポリシー名のリストを表示するには、以下のようにします。

```
grdapi list_policy detail=false
```

copy_rule

<fromPolicy> のルール <ruleDesc> を <toPolicy> のルールのリストの最後にコピーします。

注: <fromPolicy> のルールは、<toPolicy> のルールのリストの最後にコピーされます。 この grdapi を実行する前に、<fromPolicy> と <toPolicy> の両方を作成する必要があります。

パラメーター	値	記述
ruleDesc		ルールの記述
fromPolicy		ポリシー名
toPolicy		ポリシー名

パラメーター	値	記述
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi copy_rule ruleDesc="Rule Description" fromPolicy="policy1"  toPolicy=" policy2  "
```

clone_policy

この GuardAPI コマンドは、ポリシーのコピー作成に使用します。

パラメーター	値	記述
policyDesc		ポリシー名
clonedpolicyDesc		コピーしたポリシー名
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi clone_policy policyDesc="Hadoop Policy" clonedPolicyDesc="Hadoop Policy cloned1"
```

update_rule

ポリシー・ルールを更新します。<fromPolicy> のルール <ruleDesc> をルール・パラメーターに従って更新します。

update_rule API 呼び出しで変更できる以下のポリシー・ルール・パラメーターの追加情報については、『ポリシー』を参照してください。

パラメーター	値	記述
ruleDesc		ルールの記述
fromPolicy		ポリシー名
newDesc		新しいルールの記述
clientIP		クライアント IP
clientNetMask		クライアント・ネットマスク
serverIP		サーバー IP
serverNetMask		サーバー・ネットマスク
objectName		オブジェクト名
sourceProgram		ソース・プログラム
dbName		データベース名
dbUser		データベース・ユーザー
command		コマンド
appUserName		アプリケーション・ユーザー名
dateTime		日時
logFlag		ログ・フラグ
exceptionType		例外タイプ
minCount		最小カウント

パラメーター	値	記述
continueToNext		次を続行
resetInterval		リセット間隔
serviceName		サービス名
osUser		O/S ユーザー
dbType		データベース・タイプ
netProtocol		ネット・プロトコル
clientMac		クライアント MAC
fieldName		フィールド名
pattern		パターン
appEventExists		アプリケーション・イベントの存在
eventType		イベント・タイプ
appEventStrValue		アプリケーション・イベントの文字列値
appEventNumValue		アプリケーション・イベントの数値
appEventDate		アプリケーション・イベントの日付
eventUserName		イベント・ユーザー名
errorCode		エラー・コード
severity		重大度
category		カテゴリー
classification		分類
dataPattern		データ・パターン
sqlPattern		SQL パターン
xmlPattern		XML パターン
mvcSystem		MVS™ システム
clientIpNotFlag		「クライアント IP」の「Not」フラグ
serverIpNotFlag		「サーバー IP」の「Not」フラグ
objectNameNotFlag		「オブジェクト名」の「Not」フラグ
sourceProgramNotFlag		「ソース・プログラム」の「Not」フラグ
dbNameNotFlag		「データベース名」の「Not」フラグ
dbUserNotFlag		「データベース・ユーザー」の「Not」フラグ
commandNotFlag		「コマンド」の「Not」フラグ
appUserNameNotFlag		「アプリケーション・ユーザー名」の「Not」フラグ
exceptionTypeIdNotFlag		「例外タイプ ID」の「Not」フラグ
serviceNameNotFlag		「サービス名」の「Not」フラグ
osUserNotFlag		「O/S ユーザー」の「Not」フラグ
clientMacNotFlag		「クライアント MAC」の「Not」フラグ
fieldNameNotFlag		「フィールド名」の「Not」フラグ
errorCodeNotFlag		「エラー・コード」の「Not」フラグ
replacementChar		置換文字
messageTemplate		メッセージ・テンプレート
recordsAffectedThreshold		影響を受けるレコードしきい値
matchedReturnedTreshold		一致戻りしきい値
clientIpGroup		クライアント IP グループ
serverIpGroup		サーバー IP グループ
objectGroup		オブジェクト・グループ
objectCommandGroup		オブジェクト・コマンド・グループ
objectFieldGroup		オブジェクト・フィールド・グループ
dbUserGroup		データベース・ユーザー・グループ
commandsGroup		コマンド・グループ
dbNameGroup		データベース名グループ
sourceProgramGroup		ソース・プログラム・グループ

パラメーター	値	記述
appUserGroup		アプリケーション・ユーザー・グループ
serviceNameGroup		サービス名グループ
osUserGroup		O/S ユーザー・グループ
netProtocolGroup		ネット・プロトコル・グループ
fieldNameGroup		フィールド名グループ
errorGroup		エラー・グループ
appEventStrGroup		アプリケーション・イベントの文字列グループ
clientProgramUserServerInstanceGroup		クライアント・プログラム・ユーザー・サーバー・インスタンス・グループ
quarantineMinutes		隔離分数
clientInfo		DB2 と DB2_COLLECTION_PROFILE に使用します
clientInGroup		DB2_COLLECTION_PROFILE に使用します
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi update_rule ruleDesc="Rule Description" fromPolicy="policy1"  serviceName="ANY"
```

change_rule_order

ポリシー・ルールの順序を変更します。ポリシー内のルールの順序位置を変更します。

パラメーター	値	記述
fromPolicy		ポリシー名
order		ルールの新しい順序位置
ruleDesc		ルールの記述
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi change_rule_order ruleDesc="Copy of policy1 exception1"  fromPolicy="policy1"  order=10
```

list_policy_rules

ポリシーのルールをリストします。

パラメーター	値	記述
policy		ポリシー名

パラメーター	値	記述
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi list_policy_rules  policy="policy1"
```

delete_rule

ポリシーからルールを削除します。

パラメーター	値	記述
fromPolicy		ポリシー名
toPolicy		ポリシー名
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_rule ruleDesc="Copy (3) of policy1 exception1"  fromPolicy="policy1"
```

uninstall_policy_rule

uninstall_policy_rule コマンドは、policy パラメーターおよび ruleName パラメーターで指定したポリシー・ルールをアンインストールする場合に使用します。

パラメーター	値	記述
policy		ポリシー名。
ruleName		ルール名 (複数可)。複数のポリシー・ルールを指定する場合は、パイプ文字を使用します (例えば、`ruleName="rule1|rule2|rule3`)。
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

単一のポリシー・ルールをアンインストールするには、以下のようにします。

```
grdapi uninstall_policy_rule policy="Hadoop Policy" ruleName="Low interest Objects: Allow"
```

複数のポリシー・ルールをアンインストールには、以下のようにします。

```
grdapi uninstall_policy_rule policy="Hadoop Policy" ruleName="Low interest Objects: Allow|Low Interest Commands: Allow"
```

reinstall_policy_rule

reinstall_policy_rule コマンドは、policy パラメーターおよび ruleName パラメーターで指定したポリシー・ルールを再インストールする場合に使用します。

パラメーター	値	記述
policy		ポリシー名。
ruleName		ルール名 (複数可)。複数のポリシー・ルールを指定する場合は、パイプ文字を使用します (例えば、`ruleName="rule1|rule2|rule3"`)。
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

単一のポリシー・ルールを再インストールするには、以下のようにします。

```
grdapi reinstall_policy_rule policy="Hadoop Policy" ruleName="Low interest Objects: Allow"
```

複数のポリシー・ルールを再インストールするには、以下のようにします。

```
grdapi reinstall_policy_rule policy="Hadoop Policy" ruleName="Low interest Objects: Allow|Low Interest Commands: Allow"
```

delete_Audit_process_result

このコマンドは、監査プロセスの結果を削除するために使用します。

パラメーター	値	記述
ExecutionDateFrom		監査プロセスが開始した時期
ExecutionDateTo		監査プロセスが終了した時期
ProcessName		必須。監査プロセスの名前
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_Audit_process_result ExecutionDateFrom=, ExecutionDateTo=, ProcessName=abab
```

create_api_parameter_mapping

API パラメーターをドメイン・エンティティーと属性にマップします。これによって API 呼び出し生成または API 自動化でのレポート値でパラメーターを設定することができます。

注: 『GuardAPI 入力プロセスの生成』の『GuardAPI パラメーターのドメイン・エンティティーおよび属性へのマッピング』には、システムのドメイン、エンティティー、および属性が示され、この API 関数を呼び出す GUI インターフェースがあります。

パラメーター	値	記述
functionName		API 関数の名前
parameterName		マップされる API 関数内のパラメーターの名前
domain		「アクセス」、「アラート」、「ディスカバーされたインスタンス」、「例外」、「グループのトラッキング」など、Guardium レポート・ドメインのいずれか。
entityLabel		レポート・ドメインの任意のエンティティー
attributeLabel		エンティティー内の任意の属性

パラメーター	値	記述
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi create_api_parameter_mapping functionName="create_group" parameterName="desc" domain="Group Tracking" entityLabel="Group"
attributeLabel="Group Description"
```

list_param_mapping_for_function

API 関数のパラメーター・マッピングをリストします。

注: 『GuardAPI 入力プロセスの生成』の『GuardAPI パラメーターのドメイン・エンティティーおよび属性へのマッピング』には、システムのドメイン、エンティティー、および属性が示され、この API 関数を呼び出す GUI インターフェースがあります。

パラメーター	値	記述
functionName		API 関数の名前
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi list_param_mapping_for_function functionName="create_group"
```

delete_api_parameter_mapping

ドメイン・エンティティーと属性への API パラメーターのマッピングを削除します。API 関数のパラメーター・マッピングを削除します。

注: 『GuardAPI 入力プロセスの生成』の『GuardAPI パラメーターのドメイン・エンティティーおよび属性へのマッピング』には、システムのドメイン、エンティティー、および属性が示され、この API 関数を呼び出す GUI インターフェースがあります。

パラメーター	値	記述
functionName		API 関数の名前
parameterName		マップされる API 関数内のパラメーターの名前
domain		「アクセス」、「アラート」、「ディスカバーされたインスタンス」、「例外」、「グループのトラッキング」など、Guardium レポート・ドメインのいずれか。
entityLabel		レポート・ドメインの任意のエンティティー
attributeLabel		エンティティー内の任意の属性
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_api_parameter_mapping functionName="create_group" parameterName="desc" domain="Group Tracking" entityLabel="Group"
attributeLabel="Group Description"
```

close_default_events

特定のプロセス/タスク/実行に定義されているすべてのイベントをクローズします。レポート・タイプのタスクに対する特定のプロセス/タスク/実行に定義されたイベントをすべてクローズします。 特に大量のレコードを返したデフォルトのイベントを持つタスクが存在する場合などに必要です。このようなタスクはすべてのイベントがクローズされない限り割り当てることができません。

パラメーター	値	記述
eventStatus		必須。イベント状況。 監査タスクに定義されたデフォルトのイベントに有効な状況でなければならず、最終状況でなければなりません。
execDate		必須。実行の日時
processDesc		必須。監査プロセスの記述。
taskDesc		必須。監査タスクの記述。
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi close_default_events eventStatus=Done execDate="2010-03-01 08:00:00" processDesc="Audit Process" taskDesc="Task
Description"
```

create_quarantine_allowed_until

ポリシーで使用。

パラメーター	値	記述
allowedUntil		必須。
dbUser		必須。データベース・ユーザー
serverIP		必須。サーバー IP
serverName		必須。サーバー名
タイプ		必須。値は、normal、Db2z、または IMS のいずれかでなければなりません。
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

create_quarantine_until

ポリシーで使用。

パラメーター	値	記述
quarantineUntil		必須。
dbUser		必須。データベース・ユーザー
serverIP		必須。サーバー IP
serverName		必須。サーバー名
タイプ		必須。値は、normal、Db2z、または IMS のいずれかでなければなりません。

パラメーター	値	記述
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

delete_quarantine_until

ポリシーで使用。

パラメーター	値	記述
quarantineUntil		必須。
dbUser		必須。データベース・ユーザー
serverIP		必須。サーバー IP
serverName		必須。サーバー名
タイプ		必須。値は、normal、Db2z、または IMS のいずれかでなければなりません。
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

must_gather

grdapi must_gather コマンドは、Guardium サポートが使用できる Guardium システムの状態に関する情報を収集するために使用します。 詳細は、IBM サポートのための基本情報を参照ください。

パラメーター	値	記述
commandsList		文字列 - 必須
description		文字列 - 必須
duration		整数 - 必須
emailDestination		文字列 - 必須
invokingUser		文字列 - 必須
maxLength		整数 - 必須
pmrNumber		文字列 - 必須
start		日付 - 必須
timestamp		日付 - 必須
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

restart_job_queue_listener

ジョブ・キューの開始に失敗した場合、ジョブ・キューで待機中のジョブが実行されない場合、または長時間にわたってジョブが実行中状況または停止中状況のままになっているように思われる場合は、restart_job_queue_listener コマンドを使用してジョブ・キュー・リスナーを再始動します。このコマンドを発行すると、ジョブ・キューが即時に再始動され、現在実行中のすべてのジョブが停止され、再始動されます。

例:

```
grdapi restart_job_queue_listener
```

restart_job_queue_listener コマンドは、いかなるパラメーターも受け入れません。

update_quarantine_allowed_until

ポリシーで使用。

パラメーター	値	記述
allowedUntil		必須。
dbUser		必須。データベース・ユーザー
serverIP		必須。サーバー IP
serverName		必須。サーバー名
タイプ		必須。値は、normal、Db2z、または IMS のいずれかでなければなりません。
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

execute_incidentGenProcess_byDetails

update_quarantine_until

ポリシーで使用。

パラメーター	値	記述
quarantineUntil		必須。
dbUser		必須。データベース・ユーザー
serverIP		必須。サーバー IP
serverName		必須。サーバー名
タイプ		必須。値は、normal、Db2z、または IMS のいずれかでなければなりません。
api_target_host	ホスト名またはIPアドレス	api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

親トピック: GuardAPI

GuardAPI 照会再書き込み関数

コマンド行インターフェースで Guardium API を使用して、ユーザー・インターフェースから実行できない特定の複雑な照会のテストを自動化したり、そうした照会の定義を作成したりします。

注: API を使用して照会再書き込み定義を作成した場合でも、UI を使用して、「照会再書き込みビルダー」でテストするためにその定義を取得できます。

照会再書き込みに関係した GuardAPI 関数には、以下のものがあります。

assign_qr_condition_to_action

create_qr_action

create_qr_add_where

create_qr_add_where_by_id

create_qr_condition

create_qr_definition

create_qr_replace_element

create_qr_replace_element_byId

list_qr_action

list_qr_add_where

list_qr_add_where_by_id

list_qr_condition

list_qr_condition_to_action

list_qr_definitions

list_qr_replace_element

list_qr_replace_element_byId

remove_all_qr_replace_elements

remove_all_qr_replace_elements_byId

remove_qr_action

remove_qr_add_where_by_id

remove_qr_condition

remove_qr_definition

remove_qr_replace_element_byId

update_qr_action

update_qr_add_where_by_id

update_qr_condition

update_qr_definition

update_qr_replace_element_byId

assign_qr_condition_to_action

照会再書き込み条件と関連アクションの間に関連付けを作成します。

パラメーター	値	記述
actionName		必須。照会再書き込みアクションの名前。
conditionName		必須。指定したアクションに関連付ける照会再書き込み条件の名前。
definitionName		必須。指定した条件およびアクションに関連付ける照会再書き込み定義の名前。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi assign_qr_condition_to_action definitionName="case 15" actionName="qr action15_2" conditionName="qr cond15_2"
```

create_qr_action

指定した照会再書き込み定義に対する照会再書き込みアクションを作成します。

パラメーター	値	記述
actionName		必須。照会再書き込みアクションの固有の名前。

パラメーター	値	記述
definitionName		必須。当該アクションに関連付ける照会再書き込み定義。
description		説明 (オプション)。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi create_qr_action definitionName="case 15" actionName="qr action15_3"
```

create_qr_add_where

照会再書き込み関数を関連付け、指定した照会再書き込みアクションに WHERE 条件を追加します。

パラメーター	値	記述
actionName		必須。照会再書き込みアクションの固有の名前。
definitionName		必須。当該アクションに関連付ける照会再書き込み定義。
whereText		WHERE 節に追加するテキスト。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi create_qr_add_where definitionName="qrw_def_Oracle_1" actionName="qrw_act__addwhere_id2" whereText="id=2"

create_qr_add_where_by_id

照会再書き込み関数を関連付け、指定した照会再書き込みアクションに WHERE 条件を追加します。

パラメーター	値	記述
qrActionId		必須 (整数)。照会再書き込みアクションの固有の ID。
whereText		WHERE 節に追加するテキスト。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi create_qr_add_where_by_id qrActionId=10002 whereText="id=2"

create_qr_condition

照会再書き込み条件を作成します。

パラメーター	値	記述
conditionName		必須。当該照会再書き込み条件の固有の名前。

パラメーター	値	記述
definitionName		必須。当該条件に関連付ける照会再書き込み定義。
depth		当該条件が適用される、解析された SQL の深さを指定する整数 (1 以上)。デフォルトの -1 の場合、照会再書き込み条件が、すべての深さのすべての一致する SQL に適用されます。
isForAllRuleObjects		True または false。このパラメーターは、ポリシー・アクセス・ルール内のオブジェクトに当該条件を関連付ける場合に使用します。true の場合、指定した条件が、実行されたルールのアクセス・ルールのオブジェクト・フィールドまたはオブジェクト・グループ内のすべてのオブジェクトに適用されます。デフォルトは false であり、当該条件で定義されているオブジェクトを使用して照会条件が指定されます。いずれのオプションも、ルールをトリガーする動作に影響しません。
isForAllRuleVerbs		True または false。このパラメーターは、ポリシー・アクセス・ルール内のオブジェクトに当該条件を関連付ける場合に使用します。true の場合、指定した条件が、実行されたルールのアクセス・ルールの verb フィールドまたは verb グループ内のすべての verb に適用されます。デフォルトは false であり、当該条件で定義されている verb を使用して照会条件が指定されます。いずれのオプションも、ルールをトリガーする動作に影響しません。
isObjectRegex		True または false。正規表現を使用して、指定したオブジェクトを指定することを指示します。デフォルトは false です。
isVerbRegex		True または false。正規表現を使用して、指定した verb を指定することを指示します。デフォルトは false です。
object		オブジェクト (表、ビュー)。デフォルトの「*」は、すべてのオブジェクトを意味します。これは、正規表現として指定することもできます。その場合、isVerbRegex を true に設定します。
order		複雑な SQL の複数の関連した照会再書き込み条件を組み立てる順序を指定するために使用します。デフォルトは 1 です。
verb		verb (select (選択)、insert (挿入)、update (更新)、delete (削除))。 デフォルトの「*」は、すべての verb を意味します。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi create_qr_condition definitionName="case 15" conditionName="qr cond15_3" verb=select isForAllRuleObjects=false object=* depth=2 order=3

create_qr_definition

照会再書き込み定義を作成します。

パラメーター	値	記述
dataBaseType		必須。当該照会再書き込み定義を関連付けるデータベースのタイプ。許容値は ORACLE または Db2 です。
definitionName		必須。当該照会再書き込み定義条件の固有の名前。
description		説明 (オプション)。
isNegateQrCond		この定義に関連付けられている照会再書き込み条件セットに NOT フラグがあるかどうかを示します。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi create_qr_definition dataBaseType="ORACLE" definitionName="case 15"

create_qr_replace_element

SQL 文全体や SELECT リストなど、置換要素または置換要素セットを作成します。

パラメーター	値	記述
actionName		必須。当該再書き込み関数を関連付ける照会再書き込みアクションの固有の名前。
definitionName		必須。当該照会再書き込み定義条件の固有の名前。
isFromAllRuleElements		True または false。当該アクションがすべての FROM 要素に適用されることを指示します。デフォルトは false です。
isFromRegex		True または false。正規表現を使用して「from」要素を指定することを指示します。デフォルトは false です。

パラメーター	値	記述
isReplaceToFunction		True または false。「replaceTo」が関数 (ユーザー定義関数など) の名前であることを指示します。
replaceFrom		置き換え元となる、一致するルールの入力文字列。調べる入力照会の要素を具体的に指示するには、replaceType を使用します。
replaceTo		一致する要素の置換文字列。
replaceType		必須。置き換え対象を指示します。 次のいずれかでなければなりません。 • SELECT • VERB • OBJECT • SENTENCE • SELECTLIST
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi create_qr_replace_element definitionName="case 15" actionName="qr action15_2" replaceType=VERB replaceFrom="select" replaceTo="select++"

create_qr_replace_element_byId

指定した照会再書き込みアクションに対して置換仕様を作成します。

パラメーター	値	記述
isFromAllRuleElements		True または false。当該アクションがすべての FROM 要素に適用されることを指示します。デフォルトは false です。
isFromRegex		True または false。正規表現を使用して from 要素を指定することを指示します。デフォルトは false です。
isReplaceToFunction		True または false。「replaceTo」が関数 (ユーザー定義関数など) の名前であることを指示します。
qrActionId		必須 (整数)。照会再書き込みアクションの固有の ID。
replaceFrom		置き換え元となる、一致するルールの入力文字列。調べる入力照会の要素を具体的に指示するには、replaceType を使用します。
replaceTo		一致する要素の置換文字列。
replaceType		必須。置き換え対象を指示します。 次のいずれかでなければなりません。 • SELECT • VERB • OBJECT • SENTENCE • SELECTLIST
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi create_qr_replace_element_byId qrActionID="1116" replaceType=OBJECT replaceFrom="employee" replaceTo="employee_2"

list_qr_action

指定した照会定義の照会アクションをリストします。

パラメーター	値	記述
actionName		照会再書き込みアクションの名前。

パラメーター	値	記述
definitionName		必須。照会再書き込み定義名。
detail		True または false。デフォルトは true であり、アクションの関連付けられているすべての属性がリストされます。false の場合、名前のみが返されます。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi list_qr_action definitionName="case 2"

Output:

```
qrwg1.guard.swg.usma.ibm.com> grdapi list_qr_action definitionName="case 2"
########################################################################


         QR actions of definition 'case 2' - (id = 1 )

########################################################################
qr action ID: 1
qr action name: qr action2
qr action description: add where by id

ok
例:
grdapi list_qr_action definitionName="case 2" detail=false

Output:

qrwg1.guard.swg.usma.ibm.com> grdapi list_qr_action definitionName="case 2" detail=false
################################
         QR actions of definition 'case 2' - (id = 1 )
################################
qr action2
ok
```

list_qr_add_where

指定した照会アクションと照会定義のペアの「add where」関数をリストします。

パラメーター	値	記述
actionName		照会再書き込みアクションの名前。
definitionName		必須。照会再書き込み定義名。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi list_qr_add_where actionName="qrw_act_addwhere_id2" definitionName="qrw_def_Oracle_1"

list_qr_add_where_by_id

指定した照会アクションの「add where」関数をリストします。

パラメーター	値	記述
qrActionId		必須 (整数)。照会再書き込みアクションの固有 ID。

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi list_qr_add_where_by_id qrActionId=20023

list_qr_condition

特定の照会再書き込み定義に関連付けられている照会再書き込み条件をリストします。

パラメーター	値	記述
conditionName		照会再書き込み条件の名前。
definitionName		必須。照会再書き込み定義。
detail		True または false。デフォルトは true であり、条件の関連付けられているすべての属性がリストされます。false の場合、名前のみが返されます。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi list_qr_condition definitionName="case 2" conditionName="qr cond2"

Output:

```
qrwg1.guard.swg.usma.ibm.com> grdapi list_qr_condition definitionName="case 2" conditionName="qr cond2"
########################################################################
         QR Condtions of Definition 'case 2' - (id = 1 )

########################################################################

qr condition id: 1
qr condition name: qr cond2
qr definition ID: 1
qr condition verb: *
qr condition object: *
qr condition dept: -1
is verb regex: false
is object regex: false
is action for all rule verbs: false
is action for all rule objects: false
qr condition order: 1
```

list_qr_condition_to_action

特定の照会定義について、照会再書き込み条件と照会再書き込みアクションの間の関連付けをリストします。

パラメーター	値	記述
actionName		必須 (整数)。照会再書き込みアクションの固有 ID。
definitionName		必須。照会再書き込み定義。
Detail		True または false。デフォルトは true であり、指定したアクションおよび定義の条件の関連付けられているすべての属性がリストされます。false の場合、名前のみが返されます。

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi list_qr_condition_to_action actionName="qr action15_2" definitionName="case 15"

Output:

```
qrwg1.guard.swg.usma.ibm.com> grdapi list_qr_condition_to_action actionName="qr action2" definitionName="case 2"
########################################################################

         QR Condtions of Action 'qr action2' - (id = 1 )

########################################################################

qr condition id: 1
qr condition name: qr cond2
qr definition ID: 1
qr condition verb: *
qr condition object: *
qr condition dept: -1
is verb regex: false
is object regex: false
is action for all rule verbs: false
is action for all rule objects: false
qr condition order: 1
```

list_qr_definitions

照会再書き込み定義をリストします。

パラメーター	値	記述
definitionName		必須。照会再書き込み定義。
Detail		True または false。デフォルトは true であり、指定したアクションおよび定義の条件の関連付けられているすべての属性がリストされます。false の場合、名前のみが返されます。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi list_qr_definitions

Output:

```
qrwg1.guard.swg.usma.ibm.com> grdapi list_qr_definitions
########################################################################
         QR  Definitions

########################################################################
qr definition ID: 1
qr definition name: case 2
qr definition description:
is negation set on qr conditions: false
```

list_qr_replace_element

指定した照会再書き込みアクションと照会再書き込み定義のペアに関する置換をリストします。

パラメーター	値	記述
actionName		必須。照会再書き込みアクション。
definitionName		必須。照会再書き込み定義。

パラメーター	値	記述
Detail		True または false。デフォルトは true であり、指定したアクションおよび定義の置換要素の関連付けられているすべての属性がリストされます。false の場合、名前のみが返されます。
replaceType		指定する場合、以下のいずれかにする必要があります。 • SELECT • VERB • OBJECT • SENTENCE • SELECTLIST
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi list_qr_replace_element actionName="qr action2" definitionName="case 2"

Output:

```
qrwg1.guard.swg.usma.ibm.com> grdapi list_qr_replace_element actionName="qr action2" definitionName="case 2"
######################################################################
         QR replace elements for action 'qr action2' - (qrActionId = 1 )

######################################################################

qr replace element ID: 1
qr replace type: object
qr replace from: emp
qr replace to: NEW_EMP
qr is from regex: false
qr is from all rule elements: false

**********************************************************************
qr replace element ID: 2
qr replace type: selectList
qr replace from: Whole select list
qr replace to: EMPNO,SAL
qr is from regex: false
qr is from all rule elements: false
```

list_qr_replace_element_byId

指定した照会再書き込みアクションに関する置換をリストします。

パラメーター	値	記述
detail		True または false。デフォルトは true であり、指定したアクションおよび定義の置換要素の関連付けられているすべての属性がリストされます。false の場合、名前のみが返されます。
qrActionId		必須 (整数)。照会再書き込みアクションの固有 ID。
replaceType		指定する場合、以下のいずれかにする必要があります。 • SELECT • VERB • OBJECT • SENTENCE • SELECTLIST
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi list_qr_replace_element_byId detail=true qrActionId="22222" replaceType="OBJECT"

remove_all_qr_replace_elements

照会置換仕様をシステムから削除します。

パラメーター	値	記述
actionName		必須。照会再書き込みアクション。
definitionName		必須 (整数)。照会再書き込みアクションの固有 ID。
replaceType		指定する場合、以下のいずれかにする必要があります。 • SELECT • VERB • OBJECT • SENTENCE • SELECTLIST
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi remove_all_qr_replace_elements definitionName="new case 2" actionName="new qr action2“

remove_all_qr_replace_elements_byId

照会置換仕様をシステムから削除します。

パラメーター	値	記述
qrActionId		必須 (整数)。照会再書き込みアクション ID。
definitionName		必須。照会再書き込み定義。
replaceType		指定する場合、以下のいずれかにする必要があります。 • SELECT • VERB • OBJECT • SENTENCE • SELECTLIST replaceType が指定されていない場合、指定したアクションおよび定義に関するすべての置換が削除されます。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi remove_all_qr_replace_elements actionName="qr action15_2" definitionName="case 15" replaceType="OBJECT"

remove_qr_action

指定した照会再書き込みアクションをシステムから削除します。

パラメーター	値	記述
actionName		必須。照会再書き込みアクション。
definitionName		必須。照会再書き込み定義。

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi remove_qr_action actionName="qr action15_2" definitionName="case 15"

remove_qr_add_where_by_id

指定した「add where」関数をシステムから削除します。

パラメーター	値	記述
qrAddWhereId		必須 (整数)。「add where」関数。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi remove_qr_add_where_by_id qrAddWhereId=22666

remove_qr_condition

照会再書き込み条件をシステムから削除します。

パラメーター	値	記述
conditionName		必須。照会再書き込み条件。
definitionName		必須。照会再書き込み定義。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi remove_qr_condition conditionName="qr cond15_1" definitionName="case 15"

remove_qr_definition

照会再書き込み定義をシステムから削除します。

パラメーター	値	記述
definitionName		必須。照会再書き込み定義。

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi remove_qr_definition definitionName="case 15"

remove_qr_replace_element_byId

指定した照会要素置換をシステムから削除します。

パラメーター	値	記述
qrReplaceElementId		必須 (整数)。置換定義 ID。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi qrReplaceElementId=33333

update_qr_action

新しい名前および説明 (オプション) で、既存の照会再書き込みアクションを更新します。

パラメーター	値	記述
actionName		必須。照会再書き込みアクションの固有の名前。
definitionName		必須。当該アクションに関連付ける照会再書き込み定義。
description		説明 (オプション)。
newName		照会再書き込みアクションの新規名。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi update_qr_action definitionName="case 2" actionName="qr action2" newName="new qr action2"

update_qr_add_where_by_id

新しい置換テキストで、既存の「add where」関数を更新できます。

パラメーター	値	記述
qrAddWhereId		必須 (整数)。照会再書き込みの「add where」関数の固有 ID。
whereText		特定された where 節の置換テキスト。

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi update_qr_add_where_by_id 22222 whereText="1=2"

update_qr_condition

既存の照会再書き込み条件を更新します。

パラメーター	値	記述
conditionName		必須。当該照会再書き込み条件の固有の名前。
definitionName		必須。当該条件に関連付ける照会再書き込み定義。
depth		当該条件が適用される、解析された SQL の深さを指定する整数 (1 以上)。デフォルトの -1 の場合、照会再書き込み条件が、すべての深さのすべての一致する SQL に適用されます。
isForAllRuleObjects		True または false。指定した条件が、実行されたルールのすべてのオブジェクトに適用されることを指示します。デフォルトは false です。
isForAllRuleVerbs		True または false。指定した条件が、実行されたルールのすべての verb に適用されることを指示します。デフォルトは false です。
isObjectRegex		True または false。正規表現を使用して、指定したオブジェクトを指定することを指示します。デフォルトは false です。
isVerbRegex		True または false。正規表現を使用して、指定した verb を指定することを指示します。デフォルトは false です。
newName		照会再書き込み条件の新規名。
Object		オブジェクト (表またはビュー)。デフォルトの「*」は、すべてのオブジェクトを意味します。これは、正規表現として指定することもできます。その場合、isVerbRegex を true に設定します。
Order		複雑な SQL の複数の関連した照会再書き込み条件を組み立てる順序を指定するために使用します。デフォルトは 1 です。
verb		verb (select (選択)、insert (挿入)、update (更新)、delete (削除))。 デフォルトの「*」は、すべての verb を意味します。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

grdapi update_qr_condition definitionName="case 16" conditionName="qr cond15_3" newName="qr cond16_3" verb=select object=* dept=2 order=3

update_qr_definition

既存の照会再書き込み定義を更新します。

パラメーター	値	記述
dataBaseType		必須。当該照会再書き込み定義を関連付けるデータベースのタイプ。ORACLE または Db2 のいずれかにする必要があります。
definitionName		必須。当該照会再書き込み定義条件の固有の名前。
description		説明 (オプション)。
isNegateQrCond		この定義に関連付けられている照会再書き込み条件セットに NOT フラグがあるかどうかを示します。
newName		オプション。新しい固有の名前を指定します。
sampleSql		オプション。サンプル SQL ステートメントを指定します。 ほとんどの場合、これを使用することはありません。ただし、UI で入力したサンプル SQL を後で使用する場合を除きます。

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi update_qr_definition dataBaseType="DB2" definitionName="case 15" sampleSql="select EMPNO from EMP where ENAME = (select
ENAME from EMP where SAL = (select SAL from EMP where HIREDATE = to_date('06/09/1981 00:00:00', 'MM/DD/YYYY HH24:MI:SS')))"
newName="DB2_case 15"
```

update_qr_replace_element_byId

指定した照会再書き込みアクションに関する既存の置換仕様を更新します。

パラメーター	値	記述
isFromAllRuleElements		必須。当該照会再書き込み定義を関連付けるデータベースのタイプ。ORACLE または Db2 のいずれかにする必要があります。
isFromRegex		True または false。正規表現を使用して from 要素を指定することを指示します。デフォルトは false です。
isReplaceToFunction		True または false。「replaceTo」が関数 (ユーザー定義関数など) の名前であることを指示します。
qrReplaceElementId		必須 (整数)。照会再書き込みアクションの固有の ID。
replaceFrom		置き換え元となる、一致するルールの入力文字列。調べる入力照会の要素を具体的に指示するには、replaceType を使用します。
replaceTo		一致する要素の置換文字列。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi update_qr_replace_element_byId qrReplaceElementId=1 isFromAllRuleElements=false isFromRegex=false isReplaceToFunction=false
replaceFrom=emp replaceTo=NEW_EMP_UPDATED
```

親トピック: GuardAPI

GuardAPI ロール関数

これらの GuardAPI コマンドは、ロール関数の付与、リスト、および取り消しに使用します。

注: 一元管理環境では、ロールを追加するオブジェクトは、中央マネージャー上または管理対象ユニット上にある場合があります。詳しくは、『統合および一元管理ヘルプ・ブック』の概要を参照してください。

grant_role_to_object_by_id

指定されたオブジェクト (例えば分類プロセス) にロールを追加します。 ロールを追加する前に従属関係がチェックされます。 例えば、分類プロセスにロールを追加するには、その前にその分類プロセスが含むすべてのコンポーネント (分類ポリシーおよび参照されるあらゆるデータ・ソース) にそのロールを割り当てる必要があります。

パラメーター	値	記述

パラメーター	値	記述
objectTypeId		必須 (整数)。ロールを割り当てるオブジェクトのタイプを識別します。 以下のいずれかの整数でなければなりません。 1=Query 2=Report 3=Alert 4=Baseline 5=Policy 6=SecurityAssessment 7=PrivacySet 8=AuditProcess 12=CustomTable 13=Datasource 14=CustomDomain 15=ClassifierPolicy 16=ClassificationProcess
objectId		必須 (整数)。ロールを割り当てるオブジェクトを識別します。
roleId		必須 (整数)。割り当てるロールを識別します。 既存のロール ID または特殊値 -1 (すべてのロールによるアクセスを許可する) を指定できます。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi grant_role_to_object_by_id objectTypeId=13 objectId=2 roleId=3

grant_role_to_object_by_Name

指定されたオブジェクト (例えば分類プロセス) にロールを追加します。 ロールを追加する前に従属関係がチェックされます。 例えば、分類プロセスにロールを追加するには、その前にその分類プロセスが含むすべてのコンポーネント (分類ポリシーおよび参照されるあらゆるデータ・ソース) にそのロールを割り当てる必要があります。パラメーター

パラメーター	値	記述
objectType		必須。ロールを割り当てるオブジェクトのタイプを識別します。 次のいずれかでなければなりません。 Query Report Alert Baseline Policy SecurityAssessment PrivacySet AuditProcess CustomTable Datasource CustomDomain ClassifierPolicy ClassificationProcess

パラメーター	値	記述
objectName		必須。 ロールを割り当てるオブジェクト (例えば照会やレポート) の名前。
role		必須。割り当てるロールの名前。 既存のロール、または **all_roles** (すべてのロールによるアクセスを許可する) を指定できます。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi grant_role_to_object_by_Name objectType=Datasource objectName= “swanSybase” role=admin

list_roles_granted_to_object_by_id

指定されたオブジェクト (例えば分類プロセス) に割り当てられたロールを表示します。

パラメーター	値	記述
objectTypeId		必須 (整数)。ロールを割り当てるオブジェクトのタイプを識別します。 以下のいずれかの整数でなければなりません。 1=Query 2=Report 3=Alert 4=Baseline 5=Policy 6=SecurityAssessment 7=PrivacySet 8=AuditProcess 12=CustomTable 13=Datasource 14=CustomDomain 15=ClassifierPolicy 16=ClassificationProcess
objectId		必須 (整数)。ロールを割り当てるオブジェクトを識別します。
roleId		必須 (整数)。割り当てるロールを識別します。 既存のロール ID または特殊値 -1 (すべてのロールによるアクセスを許可する) を指定できます。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi list_roles_granted_to_object_by_id objectTypeId=7 objectId=1

list_roles_granted_to_object_by_Name

指定されたオブジェクト (例えば分類プロセス) に割り当てられたロールを表示します。

パラメーター	値	記述

パラメーター	値	記述
objectType		必須。ロールを割り当てるオブジェクトのタイプを識別します。 次のいずれかでなければなりません。 Query Report Alert Baseline Policy SecurityAssessment PrivacySet AuditProcess CustomTable Datasource CustomDomain ClassifierPolicy ClassificationProcess
objectName		必須。 ロールを割り当てるオブジェクト (例えば照会やレポート) の名前。
role		必須。割り当てるロールの名前。 既存のロール、または **all_roles** (すべてのロールによるアクセスを許可する) を指定できます。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi list_roles_granted_to_object_by_Name objectType=PrivacySet objectName="privaceSet 1"

revoke_role_from_object_by_id

指定されたオブジェクト (例えば分類プロセス) からロールを削除します。 従属関係は自動的に処理されます。 例えば、ロール foo を特定の照会から削除した場合、その照会に基づくレポートからもロール foo が削除されます。

パラメーター	値	記述
objectTypeId		必須 (整数)。ロールを割り当てるオブジェクトのタイプを識別します。 以下のいずれかの整数でなければなりません。 1=Query 2=Report 3=Alert 4=Baseline 5=Policy 6=SecurityAssessment 7=PrivacySet 8=AuditProcess 12=CustomTable 13=Datasource 14=CustomDomain 15=ClassifierPolicy 16=ClassificationProcess
objectId		必須 (整数)。ロールを割り当てるオブジェクトを識別します。
roleId		必須 (整数)。割り当てるロールを識別します。 既存のロール ID または特殊値 -1 (すべてのロールによるアクセスを許可する) を指定できます。

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi revoke_role_from_object_by_id objectTypeId=13 objectId=5 role=-1

revoke_role_from_object_by_Name

指定されたオブジェクト (例えば分類プロセス) からロールを削除します。 従属関係は自動的に処理されます。 例えば、ロール foo を特定の照会から削除すると、その照会を使用しているレポートからもロール foo が削除されます。

パラメーター	値	記述
objectType		必須。ロールを割り当てるオブジェクトのタイプを識別します。 次のいずれかでなければなりません。 Query Report Alert Baseline Policy SecurityAssessment PrivacySet AuditProcess CustomTable Datasource CustomDomain ClassifierPolicy ClassificationProcess
objectName		必須。 ロールを割り当てるオブジェクト (例えば照会やレポート) の名前。
role		必須。割り当てるロールの名前。 既存のロール、または **all_roles** (すべてのロールによるアクセスを許可する) を指定できます。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

grdapi revoke_role_from_object_by_Name objectType=Datasource objectName="swanSybase" role=admin

親トピック: GuardAPI

GuardAPI Solr 関数

これらのコマンドを中央マネージャーまたは管理対象ユニットで実行すると、それぞれの (内部 Guardium) Solr データベースを管理できます。

get_solr_cluster_info

Solr クラスターに関する情報 (登録済みの管理対象ユニットとその状況) を返します。grdapi は、この出力を /var/IBM/Guardium /log/solr_cluster_info.txt に書き込みます。(多数の MU がある場合は、出力が非常に大きくなる可能性があります。)

このコマンドは CM でのみ実行できます (それ以外の場合はエラー・メッセージが表示されます)。

例

```
grdapi get_solr_cluster_info
```

get_solr_status

現在のマシンの Solr インストール済み環境の状況を返します。いくつかの出力が返される場合があります (これらは ApplicationResources.properties に変換できます)。Solr が使用不可の場合:

- Solr が使用不可の場合: このマシンでは `Solr` が有効になっていません
- Solr は使用可能であるが実行されていない場合: `Solr` は実行されていません
- Solr が使用可能で実行中の場合: `Solr` は実行中です

このコマンドはすべてのタイプの Guardium システムで実行できます。

パラメーター	値のタイプ	記述
get_error_details	true、false	このパラメーターが true に設定されており、何らかの例外のために Solr が実行されていない場合は、この例外もコンソールに書き込まれます。

例

```
grdapi get_solr_status get_error_details=true
```

restart_solr

現在のマシンで Solr プロセスを再始動します。

このコマンドはすべてのタイプの Guardium システムで実行できます。

パラメーター	値のタイプ	記述
restart_all	true、false	このパラメーターは、このコマンドを CM で実行する場合にのみ適用されます。 • true: CM およびすべての管理対象ユニットで Solr が再始動されます。 • false: CM でのみ Solr が再始動されます

例

```
grdapi restart_solr restart_all=true
```

stop_solr

現在のマシンで Solr プロセスを停止します。

このコマンドはすべてのタイプの Guardium システムで実行できます。

パラメーター	値のタイプ	記述
stop_all	true、false	このパラメーターは、このコマンドを CM で実行する場合にのみ適用されます。 • true: CM およびすべての管理対象ユニットで Solr が停止されます。 • false: CM でのみ Solr が停止されます

例

```
grdapi stop_solr stop_all=true
```

親トピック: GuardAPI

GuardAPI S-TAP 関数

これらの CLI コマンドは、S-TAP 関数の作成、リスト、削除、再始動、および設定に使用します。

create_stap_inspection_engine

指定された S-TAP に検査エンジンを追加します。S-TAP 構成は、その S-TAP のアクティブな Guardium® ホストからのみ、S-TAP がオンラインである場合に限って変更できます。

パラメーター	値	記述
stapHost		必須。S-TAP がインストールされているデータベース・サーバーのホスト名または IP アドレスです。

パラメーター	値	記述
protocol		必須。データベース・プロトコルです。以下のいずれかの値でなければなりません。 Aster Cassandra CouchDB Db2 DB2 出口 IE を除外 FTP GreenplumDB HADOOP HIVE HTTP HIVE HP-Vertica IGNORE IMPALA Informix Informix 出口 Kerberos MariaDB MemSQL MongoDB MSSQL Mysql Netezza Oracle PostgreSQL SapHANA Sybase Teradata Teradata 出口 WebHDFS
portMin		必須 (整数)。データベースに構成されている聴取ポート範囲の開始ポート番号です (S-TAP のパフォーマンスが低下するため、大きな包括的範囲を使用しないでください)。
portMax		必須 (整数)。データベースの聴取ポート範囲の終了ポート番号です。
teeListenPort teeRealPort		オプション (整数)。Windows では使用されません。UNIX では、K-TAP モニター・メカニズムが使用される場合、KTAP データベース実ポートに置き換えられます。TEE モニター・メカニズムを使用する場合、これが必須です。聴取ポートは、S-TAP がローカル・データベース・トラフィックを聴取して受け入れるポートです。実ポートは S-TAP がトラフィックを転送するポートです。
connectToIp		オプション (整数)。S-TAP がデータベースへの接続に使用する IP アドレス。デフォルト (127.0.0.1) ではなく、マシンの「実」IP アドレスでのみローカル接続を受け入れるデータベースがあります。
client		必須。モニター対象のクライアントを指定する、クライアント IP アドレスおよび対応するマスクのリスト。IP アドレスがデータベース・サーバーの IP アドレスと同じで、マスク 255.255.255.255 が使用される場合は、ローカル・トラフィックだけがモニターされます。クライアント・アドレス/マスク値 1.1.1.1/0.0.0.0 では、すべてのクライアントがモニターされます (例を参照してください)。
encryption		オプション。ASO 暗号化トラフィックをアクティブにします。encryption=0 (アクティブにしない) または encryption=1 (アクティブにする) です。

パラメーター	値	記述
excludeClient		オプション。除外されるクライアントを指定する、クライアント IP アドレスおよび対応するマスクのリスト。このオプションを使用すると、特定のクライアントやサブネット (またはこれらのオプションの集合) を除く、すべてのクライアントをモニターするように S-TAP を構成できます。
procNames		Windows サーバーの場合: Oracle または MS SQL Server のみ (名前付きパイプが使用される場合)。Oracle では、通常、2 つの項目 oracle.exe、tnslsnr.exe がリストに含まれます。MS SQL Server では、通常、リストは 1 つの項目 sqlservr.exe だけです。
namedPipe		Windows のみ。名前付きパイプの名前を指定します。名前付きパイプを使用し、ここで何も指定しなければ、S-TAP はレジストリーから名前付きパイプ名を取得します。
ktapDbPort		オプション (整数)。Windows では使用されません。UNIX では、K-TAP モニター・メカニズムが使われる場合にのみ使用されます。K-TAP メカニズムによってモニターされるデータベース・ポートを識別します。
dbInstallDir		UNIX のみ。データベース・インストール・ディレクトリーの絶対パス名を入力します。例: /home/oracle10
procName		UNIX サーバーの場合: DB2®、Oracle、または Informix® データベースでは、データベース実行可能ファイルの絶対パス名を入力します。例: /home/oracle10/prod/10.2.0/db_1/bin/oracle
procNames		オプション
db2SharedMemAdjustment db2SharedMemClientPosition db2SharedMemSize		これらの 3 つのパラメーターは、以下の条件下でのみ Db2 検査エンジンに使用されます。 • Db2 サーバーが Linux で稼働している。 • K-TAP モニター・メカニズムがインストールされている。 • 共有メモリーを使ってクライアントが Db2 に接続する。 これらのパラメーターを使用した場合、grdapi は、プロトコルが db2 であることのみを検証し、条件を満たしているかどうかは検証しません。 これらのパラメーターの使用方法の詳細な説明については、トピック『Db2 Linux の S-TAP 構成パラメーター』を参照してください。
instanceName		オプション (文字列)。MSSQL または Oracle 暗号化トラフィックのみに使用されます。このパラメーターを使用する前に、MSSQL または ORACLE 暗号化フラグをオンにする必要があります。
informixVersion		Informix バージョン
ieIdentifier		オプション (文字列)。
interceptTypes		オプション (文字列)。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi create_stap_inspection_engine stapHost=192.168.2.118 protocol=Oracle portMin=1521 portMax=1521 dbInstallDir=/data/oracle10
procName=/data/oracle10/oracle/product/10.2.0/db_1/bin/oracle client=192.168.0.0/255.255.0.0 ktapDbPort=1521
```

注:

構成が拒否されず、正しくインストールされていても、検査エンジンを追加する場合に、「構成は S-TAP によって拒否されました - 詳細は S-TAP イベント・ログを参照してください」という誤ったメッセージが表示されることがあります。

UNIX S-TAP の場合、クライアント IP/マスクは必須ですが、Windows S-TAP の場合はオプションです。

list_inspection_engines

指定されたホスト上のすべての S-TAP のプロパティーを表示します。オプションとして、特定のデータベース・タイプのみのものを表示します。

パラメーター	値	記述
stapHost		必須。S-TAP がインストールされている (およびこの Guardium アプライアンスにレポートするように構成されている) データベース・サーバーのホスト名または IP アドレスです。

パラメーター	値	記述
type		オプション。 使用した場合、指定されたデータベース・タイプのみの検査エンジンがリストされます。タイプは以下のいずれかになります。 db2 informix mssql mssql-np oracle sybase
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
a1.corp.com> grdapi list_inspection_engines stapHost=192.168.2.33 type=oracle
ID=20162
Stap Host: 192.168.2.33 - Not Active
oracle Inspection Engines:
    name =ORACLE2
    type =ORACLE
        connect to IP=127.0.0.1
        install dir = /home/oracle10
        exec file = /home/oracle10/product/10.2.0/db_1/bin/oracle-guard
        instance name = MSSQLSERVER
        encrypted = no
        port range = 1521 - 1521
tee listen port = null, tee rel port = 1521
                client = 127.0.0.1/255.255.255.255
                client = 192.168.0.0/255.255.0.0
    name =ORACLE3
    type =ORACLE
        connect to IP=127.0.0.1
        install dir = /home/oracle9
        exec file = /home/oracle9/bin/oracle
        instance name = MSSQLSERVER
        encrypted = no
        port range = 1521 - 1521
ok
```

list_staps

S-TAP がこの Guardium システムにレポートする元のデータベース・サーバーを表示し、オプションとして、この Guardium システムがアクティブなホストになる S-TAP を持つサーバー (すなわち、S-TAP がデータを送信する先のサーバー、および S-TAP 構成を変更できるサーバー) のみをリストします。

パラメーター	値	記述

パラメーター	値	記述
onlyActive		オプション (ブール値)。この Guardium システムがアクティブなホストになる S-TAP を持つホストのみをリストするには、**true** と入力するか、このパラメーターを省略します。この Guardium システムを 1 次ホストまたはまたは 2 次ホストとして使用するように S-TAP が構成されている、すべてのホストをリストするには、**false** と入力します。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

a1.corp.com> grdapi list_staps onlyActive=false

ID=0

staps:

stap host = FALCON

stap host = 192.168.2.33

stap host = 192.168.2.173

stap host = 192.168.2.248

stap host = jumbo

ok

delete_stap_inspection_engine

S-TAP 検査エンジンを削除します。この Guardium システムは、検査エンジンを削除する S-TAP のアクティブ・ホストでなければなりません。

パラメーター	値	記述
stapHost		必須。S-TAP がインストールされているデータベース・サーバーのホスト名または IP アドレスです。
type		必須。削除する検査のタイプを識別します。タイプは以下のいずれかになります。 Cassandra、CouchDB、Db2、Db2 出口、FTP、GreenPlumDB、Hadoop、HTTP、iSERIES、Informix、KERBEROS、MongoDB、MS SQL、mssql-np、Mysql、名前付きパイプ、Netezza、Oracle、PostgreSQL、SAP Hana、Sybase、Teradata、または Teradata 出口
sequence		必須 (整数)。指定されたタイプの一連の検査エンジンのうち、削除される検査エンジンのシーケンス番号です。最初に type オプションを指定して grdapi list_inspection_engines コマンドを使用して、削除される検査エンジンのシーケンス番号を確認できます。
waitForResponse		オプション。API が S-TAP からの応答を待機するかどうかを指定します。 有効な値は 0 (待機しない) および 1 (応答を待機する) です。 デフォルトは、stapHost が単一のホスト名または IP アドレスの場合は 1 で、その他の場合はすべて 0 です。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi delete_stap_inspection_engine stapHost=192.168.2.118 type=Oracle sequence=1
```

注: 検査エンジンを削除する場合、削除が成功していても「検査エンジンを削除できませんでした - 指定された検査エンジンが見つかりません」という誤ったメッセージが表示されることがあります。

restart_stap

S-TAP 検査エンジンを再始動します。

パラメーター	値	記述

パラメーター	値	記述
stapHost		必須。S-TAP がインストールされているデータベース・サーバーのホスト名または IP アドレスです。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi restart_stap stapHost=192.168.2.118
```

set_stap_debug

すべてのトラフィックをログにダンプするのではなく、データベース、プロトコル、クライアント情報でログの内容をフィルターに掛けます。

関数パラメーター:

stapDebugInterval - required

stapDebugLevel - required

stapDebugOn - required

stapHost - required

api_target_host

store_stap_approval

この機能を使用して、無許可の S-TAP が Guardium システムに接続することをブロックします。

ON にすると、S-TAP は、特定の承認を得ない限り、接続できなくなります。

承認を得ていない S-TAP は、その S-TAP の IP アドレスに特定の権限が与えられない限り、接続してもすぐに切断されます。

承認されたクライアント用の事前定義レポート「承認済み Tap クライアント」があります。この機能は「日次モニター」タブにあります。

注:

ホスト名ではなく、有効な IP アドレスが必要です。

store_stap_approval コマンドは、IP ロード・バランサーがある環境内では機能しません。

一元管理された環境内では、承認された S-TAP に IP アドレスを追加した後、同期に関連する待ち時間が発生します。この待ち時間は、最大で 1 時間かかる可能性があります。同期が完了すると、承認された S-TAP の状況が GUI に緑色で表示されます。

関数: store_stap_approval

function parameters :

isNeeded - ブール値 - 必須

api_target_host - 文字列

構文

grdapi store_stap_approval ON | OFF

CLI コマンド

「store stap approval」および「show stap approval」

add_approved_stap_client

この GuardAPI コマンドは、承認済み S-TAP クライアントを追加するときに使用します。

この GuardAPI コマンドを使用しても、 スニファーは再始動せず、既に接続している S-TAP への影響もありません。このコマンドは、新しい S-TAP 接続にのみ影響を及ぼします。

関数: add_approved_stap_client

function parameters :

stapHost - 文字列 - 必須

api_target_host - 文字列

構文

grdapi add_approved_stap_client <stapHost>

list_approved_stap_client

この GuardAPI コマンドは、承認済み S-TAP クライアントをリストするときに使用します。

関数: add_approved_stap_client

function parameters :

api_target_host - 文字列

構文

grdapi list_approved_stap_client

list_stap_verification_results

この GuardAPI コマンドは、S-TAP の検査結果をリストするために使用します。

関数パラメーター:

stapHost - 文字列。S-TAP がインストールされているデータベース・サーバーのホスト名または IP アドレスです。

構文

grdapi list_stap_verification_results <stapHost>

delete_approved_stap_client

この GuardAPI コマンドは、承認済み S-TAP クライアントを削除するときに使用します。

この GuardAPI コマンドを使用しても、 スニファーは再始動せず、既に接続している他の S-TAP への影響もありません。このコマンドは、指定した S-TAP 接続にのみ影響を及ぼします。

関数: add_approved_stap_client

function parameters :

stapHost - 文字列 - 必須

api_target_host - 文字列

構文

grdapi delete_approved_stap_client <stapHost - 文字列 - 必須>

set_ktap_debug

ID=0

関数パラメーター:

ktapDebugInterval - required

ktapFunctionNames

stapHost - required

api_target_host

display_stap_config

指定されたホスト上のすべての S-TAP のすべてのプロパティーを表示します。

パラメーター	値	記述
stapHost		必須。S-TAP がインストールされていて、この Guardium システムにレポートするように構成されているデータベース・サーバーのホスト名または IP アドレス、あるいはホスト名または IP アドレスのコンマ区切りのリスト。以下の値を使用することもできます。 all_active この Guardium システムにレポートするように構成されているすべての S-TAP all_windows_active この Guardium システムにレポするように構成され、Windows マシン上で稼働しているすべての S-TAP all_unix_active この Guardium システムにレポートするように構成され、UNIX マシン上で稼働しているすべての S-TAP

例:

```
grdapi display_stap_config stapHost=myhost1,myhost2
grdapi display_stap_config stapHost=all_active
```

update_stap_config

指定されたホスト上のすべての S-TAP のプロパティーを更新します。

パラメーター	値	記述
stapHost		必須。Guardium システムのデータベース・サーバーのホスト名または IP アドレス、あるいはホスト名または IP アドレスのコンマ区切りのリスト。以下の値を使用することもできます。 all_active この Guardium システムにレポートするように構成されているすべての S-TAP all_windows_active この Guardium システムにレポするように構成され、Windows マシン上で稼働しているすべての S-TAP all_unix_active この Guardium システムにレポートするように構成され、UNIX マシン上で稼働しているすべての S-TAP
updateValue		必須。1 つ以上の鍵と値のペア (形式は *section.parameter_name:new_value*)。 *section* は、パラメーターが含まれている guard_tap.ini ファイルのセクションを示します。これは TAP または DB_x のいずれかで、DB_x はファイル内のセクション・ヘッダーとして表示される検査エンジンを指定します。 項目をアンパーサンド (&) で区切ることにより、複数のパラメーターに新しい値を指定できます。
waitForResponse		オプション。API が S-TAP からの応答を待機するかどうかを指定します。有効な値は 0 (待機しない) および 1 (応答を待機する) です。 デフォルトは、stapHost が単一のホスト名または IP アドレスの場合は 1 で、その他の場合はすべて 0 です。

例:

```
grdapi update_stap_config stapHost=all_windows_active updateValue=TAP.XXXX
```

verify_stap_inspection_engine_with_sequence

このコマンドは、S-TAP 検査エンジンを検証するために使用します。

パラメーター	値	記述
addToSchedule		文字列。定数値リスト。有効な値は Yes および No です。
datasourceName		文字列。このパラメーターを指定した場合、指定したデータ・ソースに対して詳細検査が実行されます。このパラメーターを省略した場合、標準検査が実行されます。
sequence		必須。整数。検査のための既存の検査エンジンのシーケンス番号。 最初に type オプションを指定して grdapi list_inspection_engines コマンドを使用して、検査される検査エンジンのシーケンス番号を確認できます。
stapHost		必須。文字列。S-TAP がインストールされているデータベース・サーバーのホスト名または IP アドレスです。
protocol		必須。データベース・プロトコル。これは、Db2、Db2 出口 (Db2 バージョン 10)、FTP、Informix、Kerberos、Mysql、Netezza、Oracle、PostgreSQL、Sybase、Teradata、Teradata 出口、Windows ファイル共有、IE を除外、のいずれかの値でなければなりません。Windows S-TAP ホストでは、MSSQL プロトコルと名前付きパイプ・プロトコルも使用できます。

例:

```
grdapi verify_stap_inspection_engine_with_sequence stapHost=9.70.144.212
sequence=3
```

revoke_ignore_stap

このコマンドは、S-TAP セッション・トラフィックを無視する既存の「S-TAP セッションを無視 (取り消し可能)」ポリシー・ルール・アクションを取り消します。このコマンドは、ソフトな無視ルール (「取り消し可能」とマークされているもの) のみを取り消し、ハードなルール (「取り消し可能」とマークされていないもの) を取り消すことはできません。

パラメーター	値	記述
stapHost		必須。S-TAP がインストールされていて、この Guardium システムにレポートするように構成されているデータベース・サーバーのホスト名または IP アドレス、あるいはホスト名または IP アドレスのコンマ区切りのリスト。以下の値を使用することもできます。 all_active この Guardium システムにレポートするように構成されているすべての S-TAP all_windows_active この Guardium システムにレポするように構成され、Windows マシン上で稼働しているすべての S-TAP all_unix_active この Guardium システムにレポートするように構成され、UNIX マシン上で稼働しているすべての S-TAP

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例

```
grdapi revoke_ignore_stap stapHost=myhost1
```

set_ztap_logging_config

このコマンドは、後述のロギング・パラメーターを制御します。

構文: grdapi set_stap_logging_config parameter=[parameter] value=[value]。

パラメーター	値	記述
log_db2z_target	0 (使用不可にする場合) 1 (使用可能にする場合)	`log_db2z_target=1` を使用して使用可能にすると、db2z protobuf メッセージ内のターゲットは、パーサーからのオブジェクトに加えて、GDM_OBJECT にも記録されます。 注: パラメーターは、デフォルトでは使用不可になっています。
log_zkey_to_full_sql	0 (使用不可にする場合) 1 (使用可能にする場合)	`log_zkey_to_full_sql=1` を使用して使用可能にすると、VSAM または IMS キー値が、「全詳細をロギング」を使用したポリシーの完全な SQL ステートメントにログインします。 注: パラメーターは、デフォルトでは使用不可になっています。

例

```
grdapi set_ztap_logging_config parameter=log_db2z_target value=1
```

値の表示: grdapi get_ztap_logging_config。

親トピック: GuardAPI

GuardAPI 脅威検出分析機能

enable_advanced_threat_scanning

特定のデータベース攻撃 (SQL インジェクションや悪意のあるストアード・プロシージャーなど) がないか検査するスキャナー・プロセスを有効にします。

パラメーター	値	記述
すべて		オプション。一元管理構成に限り、すべての管理対象ユニット上のすべての脅威検出スキャナーを有効にします。指定可能な値: `true`、`false`。これは、api_target_host パラメーターに対する「all」オプションと同等です。
schedule_start		オプション。プロセスの実行を開始する日時を指定します。形式は、yyyy-mm-dd hh:mm:ss (24 時間クロック) です。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi enable_advanced_threat_scanning all=true schedule_start="2016-03-24 12:00:05”
```

異常値検出が無効になっているときに脅威分析が有効になっている場合は、以下のメッセージが表示されます。

```
Warning - Enabling advance threat scanning (AKA Eagle Eye) when Analytic anomaly detection is disabled.
Advance threat scanning (AKA Eagle Eye) enabled.
ok
```

disable_advanced_threat_scanning

コレクター上の脅威検出スキャナーを無効にします。

パラメーター	値	記述
すべて		一元管理構成に限り、すべての管理対象ユニット上のすべての脅威検出スキャナーを無効にします。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

get_eagle_eye_info

脅威検出パラメーターの現在の設定を表示します。

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi get_eagle_eye_info
Eagle Eye Parameters Values:
EI_CASES_DISPLAY_LIMIT = 3
EI_CONFIDENCE_PCT_CHANGE_TO_REDISPLAY_CASE = 30
```

```
EI_EAGLE_EYE_ENABLED = 1
EI_PROCESSOR_TIMEOUT_SEC = 420
EI_SCANNER_PATCH_DEF = 10
EI_SCANNER_TIMEOUT_SEC = 300ok
```

set_eagle_eye_parameter

IBM 担当者の指示に従って使用してください。脅威検出の構成パラメーターを変更します。 これらのパラメーターは、以下のように `parameter_name` および `parameter_value` を使用して明示的に設定する必要があります。

```
set_eagle_eye_scanner_parameter parameter_name=[parameter] parameter_value=[value]
```

パラメーター	値	記述
EI_CASES_DISPLAY LIMIT		To-do リスト・レポートに表示されるケースの数。デフォルトは 3 です。
EI_CONFIDENCE_PCT_CHANGE_TO_REDISPLAY CASE		To-do リスト・レポートにこのケースが既に表示されていても、そこに再表示されるようにする「信頼度」変更のパーセンテージ。Guardium が、このパーセンテージ値によって信頼度を引き上げる別の兆候を検出した場合、これが発生する可能性があります。デフォルトは 30 です。
EI_PROCESSOR_TIMEOUT_SEC		このしきい値より長い時間実行されたプロセッサーはオフになります。デフォルトは 420 秒です。
EI_SCANNER_PATCH_DEF		パッチ・インストールの結果として誤検出が発生するのを防ぐために、単一プロセス実行で作成されたストアード・プロシージャーの数がこのパラメーターを越えた場合、そのプロセスはパッチがインストールされたと想定し、兆候の分析を停止します。デフォルトでは、1 回の実行で検出されるストアード・プロシージャーの作成数は 10 です。
EI_SCANNER_TIMEOUT_SEC		このしきい値より長い時間実行されたスキャナーはオフになります。デフォルトは 300 秒です。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

get_eagle_eye_scanners_info

スキャナー設定情報を返します。

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

返されるデータには、以下の情報が含まれます。

フィールド	記述
ID	スキャナー ID。
Name	スキャナー名。
Status	最後の実行以降のスキャナーの状況: I: 進行中 D: 完了 K: 強制終了 E: エラーで終了
Enabled	スキャナーが有効であるかどうかを示します。 True: 有効 False: 無効

フィールド	記述
Permanent disabled	スキャナーが 24 時間で 3 回無効になった場合、そのスキャナーは永続的に無効になります。 True: 無効 False: 有効

例:

```
grdapi get_eagle_eye_scanners_info
ID=0
ID:1, Name:SQLInjectionExceptionsScanner, Status:D, Enabled:true, Permanent disabled:false
ID:2, Name:NumNewConstructScanner, Status:D, Enabled:true, Permanent disabled:false
ID:3, Name:SQLInjectionSuspiciousObjectScanner, Status:D, Enabled:true, Permanent disabled:false
ID:4, Name:SqliQueryScanner, Status:Unknown, Enabled:false, Permanent disabled:true
ID:5, Name:EagleEyeSTPCreateProcedureScanner, Status:D, Enabled:true, Permanent disabled:false
ID:6, Name:EagleEyeSTPCallProcedureScanner, Status:D, Enabled:true, Permanent disabled:false
ID:7, Name:EagleEyeSTPExceptionProcedureScanner, Status:D, Enabled:true, Permanent disabled:false
ID:8, Name:EagleEyePreviousStpUsageProcedureScanner, Status:D, Enabled:true, Permanent disabled:false
ID:9, Name:EagleEyeSTPViolationProcedureScanner, Status:D, Enabled:true, Permanent disabled:false
ID:10, Name:EagleEyeSTPUserOutlierScanner, Status:D, Enabled:true, Permanent disabled:false
ok
```

set_eagle_eye_scanner_parameter

IBM 担当者の指示に従って使用してください。スキャナーをアクティブ化または非アクティブ化します。これらのパラメーターは、以下のように `parameter_name` および `parameter_value` を使用して明示的に設定する必要があります。

```
set_eagle_eye_scanner_parameter parameter_name=[parameter] parameter_value=[value]
```

パラメーター	値	記述
scanner_id		必須。スキャナーの固有 ID。これは、get_eagle_eye_scanners_info GuardAPI コマンドから取得できます。
is_active		スキャナーを実行するかどうかを定義します。タイムアウトになったために自動的に停止されたスキャナーを開始するために使用されます。 0 : スキャナーは停止される 1: スキャナーはアクティブ化される
is_permanent_inactive		スキャナーが 24 時間で 3 回無効になった後に永続的に無効になった場合、この GuardAPI を使用することでのみ再び有効にすることができます。 1: スキャナーは永続的に停止される 0: スキャナーは有効化される
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

以下の例では、永続的に非アクティブ化されたスキャナーを再アクティブ化します。

```
set_eagle_eye_scanner_parameter scanner_id=2 parameter_name=is_permanent_inactive parameter_value=0
```

get_eagle_eye_symptom_period_hours

徴候期間パラメーターの値を時間単位で示します。徴候期間は、1 つのケースについて収集された徴候を、どのくらい前までプロセスが検索して分析するかを決定します。

パラメーター	値	記述
case_name		必須。ケース・タイプ。以下の値を使用できます。 STP: 悪意のあるストアード・プロシージャーのケース SQL_INJECTION: SQL インジェクションのケース

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi get_eagle_eye_symptom_period_hours case_name=STP
The symptoms period for case type: STP is: 168 in hours
ok
```

set_eagle_eye_symptom_period_hours

徴候期間パラメーターの値を時間単位で設定します。徴候期間は、1 つのケースについて収集された徴候を、どのくらい前までプロセスが検索して分析するかを決定します。

パラメーター	値	記述
case_name		必須。ケース・タイプ。以下の値を使用できます。 STP: 悪意のあるストアード・プロシージャーのケース SQL_INJECTION: SQL インジェクションのケース
symptom_period_hours		必須。整数。1 つのケースの兆候を分析するための過去の時間数。
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi set_eagle_eye_symptom_period_hours case_name=STP symptom_period_hours=170
The symptoms period for case type: STP was changed. The old value was: 168. The new value is: 170
ok
```

get_eagle_eye_debug_level

IBM サービス担当員によって使用されます。現在のデバッグ・レベルを表示します。

- 1: オン
- 0: オフ

パラメーター	値	記述
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi get_eagle_eye_debug_level
ID=0
component=EAGLE_EYE level=1
ok
```

set_eagle_eye_debug_level

IBM サービス担当員によって使用されます。現在のデバッグ・レベルを表示します。

パラメーター	値	記述

パラメーター	値	記述
level		整数。必須。指定可能な値: 1: オン 0: オフ
api_target_host		api_target_host は、API の実行場所であるターゲット・ホストを指定するオプション・パラメーターです。受け入れる値は次のとおりです。 • `all_managed`: 管理対象ユニットで実行しますが、中央マネージャーでは実行しません • `all`: すべての管理対象ユニットと中央マネージャーで実行します • `group:<group name>`: *<group name>* によって識別されるすべての管理対象ユニットで実行します • 管理対象ユニットのホスト名または IP アドレス: ある管理対象ユニットで実行するように中央マネージャーから指定されます。例えば `api_target_host=10.0.1.123` です。 • 中央マネージャーのホスト名または IP アドレス: 中央マネージャーで実行するように管理対象ユニットから指定されます。例えば `api_target_host=10.0.1.123` です。

例:

```
grdapi set_eagle_eye_debug_level level=0
ID=0
ok
```

親トピック: GuardAPI

S-TAP for z/OS User's Guides

Welcome to the Security Guardium S-TAP documentation where you can find information about how install, maintain, and use Security Guardium S-TAP for DB2, IMS, and Data Sets on z/OS.

Security Guardium V10.6 uses S-TAP V10.1.3. Follow this link to be redirected to the general landing page for the Security Guardium V10.1.3 documentation: https://www.ibm.com/support/knowledgecenter/en/SSMPHH_10.1.0/com.ibm.guardium.doc.zos/z_plugin_g101-gentopic1.html.

Expand the Table of Contents and scroll down to the Security Guardium S-TAP product of your choice. Product documentation shortcuts are as follows:

- Security Guardium S-TAP for DB2 V10.1.3
- Security Guardium S-TAP for IMS V10.1.3
- Security Guardium S-TAP for Data Sets V10.1.3