WinSyslog

ユーザーマニュアル

Ver.13

Rev. 1.0

2016.10.5

ジュピターテクノロジー

目次

更新履歴:

このドキュメントの更新履歴は以下の通りです。

版	発行日	更新内容
第 1.0 版	2016/10/05	新規

1　WinSyslog について

1.1　概要

WinSyslog は、Windows 上で稼動する Syslog サーバーです。
（Unix の Syslog デーモンと同じ役目を果たします。）

ネットワーク管理において WinSyslog をご使用頂けば、継続的にシステムを監視することができます。さらに、重要なイベントが発生した場合には、即座に通知を受け取るようにも設定できます。
Syslog は、システム・イベントの集中レポート作成のための標準プロトコルです。そのルーツは UNIX 環境にありますが、例えば、Cisco のような最新のデバイスは Syslog プロトコルを使用しています。

それらのデバイスは、重要なイベントやオペレーティングのパラメーター、デバッグのメッセージでさえ Syslog でレポートを作成します。残念ながら Microsoft Windows は syslog サーバーを含んでいません。（Syslog サーバーは「Syslog デーモン」や Syslogd などと呼ばれたりしています）

Adiscon の WinSyslog はこのギャップを埋めるものです。
バージョン 3.0 以前、WinSyslog は「NTSLog」の名で知られていました。最初のバージョンは、Cisco ルーターのステータスメッセージを受け取るために 1996 年に作成されました。
製品は、これまで断続的に開発されています。この製品は、バージョン3で飛躍的に機能性が高まりました。それがバージョン3で製品名を変えるきっかけとなりました。

また WinSyslog は、Adiscon 製品の MonitorWare エージェント、EventReporter、ActiveLogger とともに、Windows のイベントログをトータル的に集中監視するツールとして使うこともできます。WindowsNT/2000/XP の中央モニタリングに関しては、http://www.monitorware.com.（英語）で詳細を確認できます。
（現在のところ、弊社では WinSyslog のほか EventReporter を販売しております。）
ほとんどのユーザーは Syslog のデバイス（例：ルーター、スイッチ、ファイアーウォールやプリンターなど）から発生したイベントログを集め、それらを持続的に Windows のシステムに保存することなどに WinSyslog を利用しています。

WinSyslog は、インタラクティブにスクリーン上で syslog メッセージを表示することができるだけでなく、さらに収集した Syslog メッセージをフラットな ASCII ファイル、ODBC データベース、または Windows イベントログに保存することもできます。
この製品は、はじめに設定を行いさえすれば、信頼できるサービスとして動作し、オペレーターの操作を介する必要もありません。サービスは、Windows の起動時に自動的に起動することができます。

バージョン 4 で改良されたサービス、ルールによって、WinSyslog の設定はより融通性の高いものになりました。

WinSyslog は、入って来るメッセージ内の文字列照合などの状態を発見したり、それに従って能動的に動作を行ったりといったことが可能です。例えば、優先順位の高いメッセージを発見した場合は、E メールメッセージを送信することなどが可能です。複数の Syslog サーバーで同時にこの動作を行えます。さらにそれぞれ別のポートを使用できます。

1.2 機能

■ ログの集中管理

これは WinSyslog のキーとなる機能です。
WinSyslog は、それぞれのソース（デバイス）から送られた全ての Syslog メッセージをまとめて、ローカルの Windows システムに保存します。デバイスが Syslog に対応していれば、WinSyslog でそれらを処理できます。
今日においては、事実上、すべてのデバイスで Syslog を使うことができます。
顕著な実例として、Cisco ルーターが挙げられます。

■ 使い易さ

WinSyslog 設定クライアントにより、セットアップやカスタマイズが容易に行えます。
さらに、大規模な環境で利用できる GUI を利用しないインストール方法についてもサポートしております。

■ 効果的なアクション

受信した各メッセージは、WinSyslog の効果的で柔軟性の高いルールエンジンによって処理されます。
各ルールでは、メッセージがルールの「フィルタの条件」に一致したときに、どのアクションが実行されるのかを設定します。（例えば、E メールでメッセージを送信するのか、データベースに保存するのかなど）
「フィルタの条件」では、メッセージ内の文字列や Syslog ファシリティ、プライオリティなど、お客様のニーズに合わせた条件設定が可能となっており、不要なメッセージの絞込みが容易に行えます。
なお、利用できるルールの「フィルタの条件」とアクションには数には制限がありません。

■ インタラクティブ Syslog ビューア

受信したメッセージをインタラクティブに表示したい場合は、インタラクティブ Syslog ビューア（インタラクティブ Syslog サーバーの後継ツール）を使用します。（日本語メッセージも処理できます。）
その他、WinSyslog・EventReporter で作成したデータベースを読み込み、確認する機能も追加されました。

■ Syslog テストメッセージの送信

WinSyslog 設定クライアントには、「Syslog テストメッセージを送信」 の機能が実装されております。（ツールメニュー内にあります）
このオプションにより、ルールセットの設定を確認したり、インタラクティブ Syslog ビューアへ送信テストを行なったりすることができます。ただし、このオプションによるメッセージ送信に利用できるプロトコルは、UDP のみです。（RFC 3195 には対応しておりません）

■ **試用期間に関して**

WinSyslog、インタラクティブ Syslog ビューアとも、セットアップ後 30 日間 は試用モードとして全ての機能を送信デバイス数の制限なく、ご利用になれます。
Ver.13.3（日本仕様BasicFree版）のみ、試用期間終了後、送信元IP3台、Basic機能のフリー版へと移行します。
Ver.13.2 以前のバージョンは、試用期間の終了後、WinSyslog はサービスが開始できなくなります。
インタラクティブ Syslog ビューアは、ver.13.3 は試用期間終了後、送信元 IP3 台のみ表示が可能となります。
Ver.13.2 以前のバージョンでは試用期間終了後、閲覧開始 5 分後に停止するようになります。
正規版として、Syslog サーバーサービスやその他の拡張機能を送信元 IP4 台以上でご利用になりたい場合は、ライセンスをご購入頂く必要があります。

■ **標準の互換性**

WinSyslog は、Syslog RFC3164、RFC5424 に対応しています。
WinSyslog は、送信者（デバイス）やサーバーや中継マシンとして稼動します。
全ての仕様のオペレーションモードがサポートされます。

REC に対応していない部分は、ローカルの環境に合うように管理者が設定することが可能です。
（例えば、デバイス時計が信頼できない場合には、タイムスタンプは報告しているデバイスの代わりにローカル・システムから取り出すことができます）

■ **Syslog 階層**

WinSyslog は大きな組織で必要となるカスケードされた設定をサポートします。
カスケード設定には、重要なイベントを本部の中心となる WinSyslog に報告する、部やサイトレベルで動作するローカルの WinSyslog が例としてあげられます。
カスケードされたシステムには、レベルの数に対する制限がありません。

■ **E メールでの通知**

ユーザーが設定したルールに基づいて、受信した Syslog メッセージを E メールで送ることができます。
この E メールによる通知は、どんな E メールアドレスにも送る事ができます。
（携帯電話でも受信できます）

また、E メールの題名は完全にカスタマイズすることができます。
オリジナルのメッセージを本文内に含むように設定する事もできます。
従って、携帯電話であっても十分に情報を受信する事ができます。

■ **継続的メッセージの保存**

WinSyslog の Syslog サーバーは、継続して全てのメッセージを保存することができます。

したがって、後の監査や重要なシステム･イベントの再調査などが容易にできます。
メッセージは、フラットな ASCII ファイル、ODBC データソースと Windows のイベントログ などに書くことができます。

■ 複数のインスタンス

WinSyslog は、同じマシン上で複数の Syslog サーバーサービスを稼動させることができます。
それぞれが違う Syslog ポートを使用し、TCP または UDP を選択できます。
そして、別々のルールセットを作成できます。
ただし、同じ設定内容（ポート･プロトコル）の Syslog サーバーサービスを複数稼動させることはできません。

■ 完全なログ収集

WinSyslog は、受け取った Syslog メッセージに加え、送信者のシステムの IP アドレスや日付だけでなく、そのプライオリティやファシリティコードをも記録します。
それは、さらに正常なフォーマットがされていないパッケージ（無効なプライオリティ/ファシリティがある、またはそれら自体ない）を記録することができるので、メッセージは消失しません。

■ 安定性

WinSyslog は、正常でない環境のもとでも実行できるように作成されています。
その信頼性は、1996 年以降、顧客サイトで証明されています。

■ 最小限のリソース使用

WinSyslog には、システムリソースへの顕著な影響がありません。
それは、最小限のリソース使用ということを念頭において、作成されたからです。
これは、負荷の大きなサーバーに対してもインストールできるということを保証しています。

■ ファイアーウォールサポート

セキュリティ･ポリシーなどにより、標準でない Syslog ポートを使う必要にせまられた場合でも、WinSyslog は、Syslog メッセージに対して、いかなる TCP/IP ポートでも使用できるように設定できます。

■ NT サービス

WinSyslog サービスは、 マルチスレッドな Windows NT サービスとして実行されます。
それは、コントロールパネルやコンピューターの管理 -MMC (Windows 2000)で制御できます。

■ IPv6

ネットワークに関連のある全てのサービスやアクションで IPv6 をご利用になれます。
IPv6 のアドレスが有効ならば、DNS 名前解決も可能です。[Syslog サーバー]サービスなど、IP アドレスを設定する項目では、IPv4 か IPv6 かを選択することができます。（[IP タイプ]の設定）

IPv4 と IPv6 が混在する環境では、その IP タイプ別にサービスを設定する必要があります。(RELP サービスだけは、IP タイプが自動検知されますので、サービスを分けて設定する必要はありません。

■ Windows 2000、2003、2008、XP、Vista、Windows 7、8、8.1、10、2012 完全対応

WinSyslog は、その出荷当時から、完全に Windows 2000、Server 2003、XP、Vista、Server 2008、7、8、8.1、10、Windows Server 2012 (R2 含む)に対応しています。
そして、WinSyslog はバージョン 3.6 以降、Windows XP 対応するようデザインされています。
さらに新しい「テーマ」機能や「fast user switching(高速ユーザー切り替え)」機能にも対応しています。

■ 複数の言語対応のクライアント

WinSyslog クライアントは、多言語対応になっています。
ボックスから、旧クライアントの場合は、英語、フランス語、ドイツ語、スペイン語、日本語を選択できます。新クライアントは、英語、ドイツ語、日本語を選択できます。
言語は、すぐに切り替えることができ、ユーザー自身が自由に選択できます。

■ 親しみやすく、カスタマイズも可能なユーザーインターフェイス

(旧クライアントのみ)WinSyslog クライアントに新たにスキン機能が加わりました。ーデフォルトにより 5 種類の新しいスキンがインストールされ、選択できます。 これらのスキンは、色、彩度と RGB カラーによりカスタマイズすることが可能です。参考画面。

また、クローンの機能が追加されました。クリックするだけで、ルールセット、ルール、アクション、サービスそれぞれのクローンを作成できます。
「上へ」、「下へ」の機能がアクションで使用できるようになりました。ドラッグ &ドロップでの移動もできます。
また、アクション、サービス、ルールセット作成のウィザードが改良されました。

■ ローメモリへの対応

WinSyslog は、起動時に非常用のメモリを割り当てます。システムのメモリ制限に達した場合、非常用のメモリが解除され、キューはロックされます。
それにより、それ以上どんな項目もキューに入れられなくなります。
結果として、サービスの停止(crash)を防ぐことができるようになります。

1.3 構成

1.3.1 中核となる構成

■ WinSyslog 設定クライアント

WinSyslog 設定クライアント(クライアントと呼ばれます)は、WinSyslog サービスの全ての要素と機能の設定に使用されます。

また、多数のマシンで同じ設定でWinSyslogを使用したい場合など、ベース・システム上のクライアントで設定ファイルを作成・エクスポートし、対象システムに組み込むこともできます。

■ WinSyslog サービス

WinSyslog サービス（サービスと呼ばれます）は、Windows サービスとして稼動し、実際の処理を実行します。
WinSyslog を動作させるためには、サービスがインストールされていなければなりません。
WinSyslog サービスは、製品の「エンジン」と呼ばれています。したがって、サービスのみインストールされたシステムを「エンジンだけの」インストールと呼びます。
サービスは、ユーザーの操作の必要なしにバックグラウンドで動作します。
それは、コントロールパネルやコンピューターの管理 –MMC (Windows 2000)で制御できます。

＜x64 対応バージョン の組み込み＞
x64 対応版は、セットアップ時に自動的にご利用の OS に合わせてインストールされます。（インストーラーが判断します）
主に変更された箇所は、サービスのコアの部分です。
詳細は、以下をご覧下さい：

- [ODBC データベース] アクションが x64 システム上で動作するようになりました。
  但し、各データベースの ODBC 接続 32bit 版 Driver のインストールが必要です。
- 設定情報（レジストリ）の保存に関して、DWORD 値が QWORD 値としてレジストリに保存されるようになりました。けれども、設定クライアントと Win32 バージョンのサービスでは、これらのデータタイプを処理でき、必要に応じて自動的に値が DWORD 値に変換されます。

x64 版であっても設定クライアントは、win32 アプリケーションのままとなっております。

通常のセットアップ更新手順で直接 32bit 版 OS から 64bit 版 OS へのアップグレードはできません。
この問題はマイナーアップグレードにより、必要なすべての x64 コンポーネントがインストールされないことです。
唯一、フルインストールの場合、実施が可能です。したがって、クロスアップデートを実行するために、以下の手順に従ってください：

1. レジストリまたは xml ファイルとして設定ファイルをバックアップします。（設定クライアントのコンピュータメニューをご確認ください）
2. WinSyslog をアンインストールします。
3. WinSyslog 最新版をインストールします。.
4. レジストリまたは xml ファイルから旧設定をインポートします。

1.3.2 アドオン構成

■ インタラクティブ Syslog ビューア

インタラクティブ Syslog ビューアは、Syslog イベントを受信し、リアルタイムに表示する Windows GUI アプリケー

ションです。一般的には WinSyslog や EventReporter とともに使用されます。しかし、独立した Syslog サーバーとしても使用することができます。

インタラクティブ Syslog ビューアは、インタラクティブ Syslog サーバーの後継ツールです。
インタラクティブ Syslog ビューアでは、日本語文字列を含むメッセージの処理も可能です。また、WinSyslog や EventReporter で作成したデータベースの内容を読み込み、確認する機能も追加されました。

■ **Adiscon LogAnalyzer (旧 PhPLogCon)**

Adiscon LogAnalyzer は、収集されたメッセージを web 上に表示できる便利な機能を持っています。
このツールは、たいていのブラウザに対応しています。
Adiscon LogAnalyzer は、Syslog メッセージ、Windows イベントログデータ、その他のネットワークイベントを簡単に web で閲覧することができます。このツールを使用することにより、システム管理者は、迅速に容易にログをチェックすることが可能となります。
Adiscon LogAnalyzer は、WinSyslog のインストールフォルダに含まれています。現在のところ、日本語マニュアルなどはございません。
詳細は、http://loganalyzer.adiscon.com/doc/ または 「Adiscon LogAnalyzer」フォルダ配下の「doc」フォルダ内の英語マニュアルをご参照下さい。
※LogAnalyzer は別途、IIS や Apache などの WebServer、PHP、データベースを構築する必要があります。
弊社では現在サポート対象外です。

■ **MonitorWare コンソール**

＊弊社では、現在のところ、MonitorWare コンソールの販売は行っておりません。
＊詳細は、www.mwconsole.com （英語）にてご覧頂けます。
MonitorWare コンソールは、ネットワークから役に立つ情報を容易に集めることができ、また、その集めた情報に対して、セキュリティ違反を含む無数の問題を調査することが可能です。
MonitorWare コンソールの表示、レポートモジュールを使用することで、能率的に問題を含む範囲をネットワーク上で検出することができます。

1.3.3 これらの構成要素を共に動作させるには

■ **これらの構成要素を共に動作させるには**

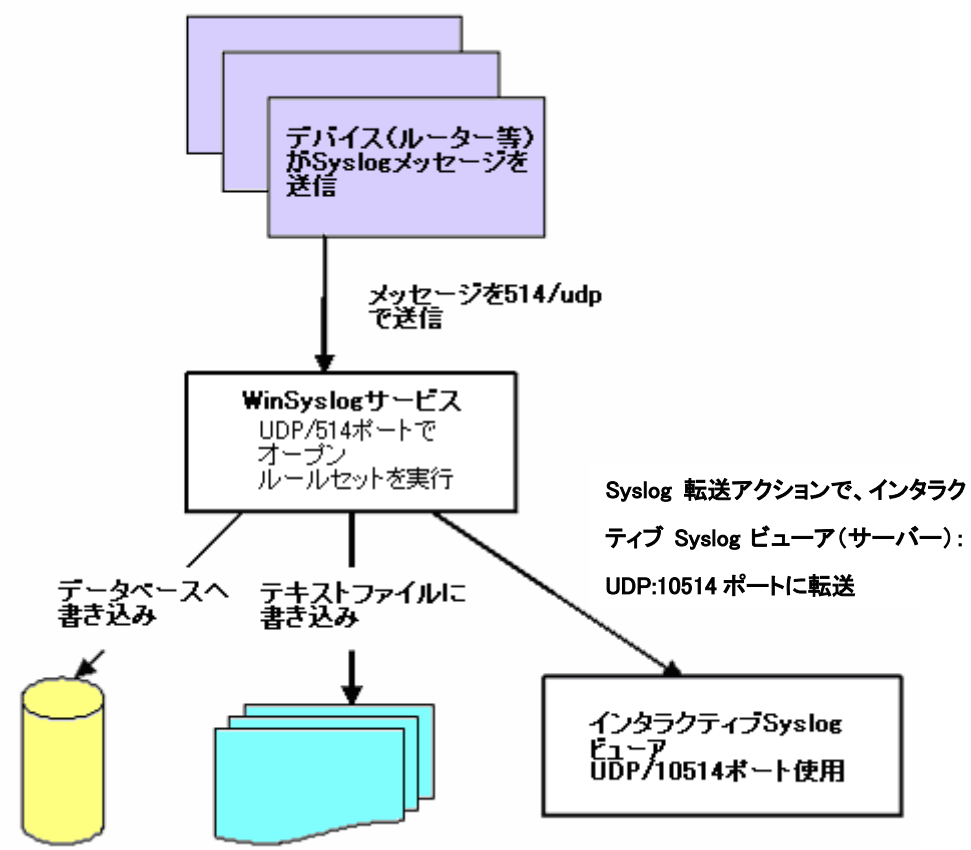
前途の構成要素は、共に密接に動作します。
中核の構成は WinSyslog サービスであり、 これは継続的にバックグラウンドで動作しています。

WinSyslog 設定クライアントでは、サービスの設定を行います。
クライアント自体は、サービスの設定を行うことがクライアントの唯一のタスクであり、一旦 WinSyslog の設定を行った後は、継続してクライアントを起動させて置く必要はありません。

一度サービスの設定を行えば、サービスはバックグラウンドで動作し、設定のとおり実行されます。最も重要な処理として、サービスは Syslog メッセージの受信や、ルールベースによるそれらのメッセージの処理、それらをデータベースやテキストファイルに保存すること、アラートを出すことなどがあります。
WinSyslog サービス自体には、インタラクティブな構成はありません。

Syslog メッセージを Windows GUI で表示する場合には、インタラクティブ Syslog ビューアが必要となります。インタラクティブ Syslog ビューアは、インタラクティブにメッセージを表示する以外は、機能が制限された負荷の軽い Syslog サーバーとして実行されます。
インタラクティブ Syslog ビューアは、起動した時のみ処理を実行します。
インタラクティブ に Syslog メッセージを表示させるために WinSyslog サービスはメッセージをインタラクティブ Syslog ビューアへ転送します。デフォルトでは、標準ポートでない UDP のポート 10514 で処理されます。従って、WinSyslog サービス、およびインタラクティブ Syslog サーバーは、ポートの衝突なしに同一のマシン上で稼動します。

メッセージの流れについては、下図を参照して下さい:



典型的な設定では、ルーターやスイッチなどの Syslog デバイスは、WinSyslog サービスへ 514 ポートを使用して Syslog メッセージを送信します。サービスは、メッセージを受信し、ルールセットでの設定に基づき、それらを

処理します。上図の例では、入ってくる全てのメッセージに対して、データベースへの書き込み、テキストファイルへの書き込み、インタラクティブ Syslog ビューアへの転送の 3 つのアクションが設定されています。

デフォルトでは、メッセージは 10514 ポートを使用して、ローカル(127.0.0.1)のインタラクティブ Syslog ビューアへ転送されます。インタラクティブサーバーは、順次ポートを開き、サーバーから転送された syslog メッセージを受け取ります。

UNIX-用語では、WinSyslog Service は syslog リレーと同様に受信機としての機能を果たします。一方、インタラクティブ Syslog ビューアは、ただの受信機としての機能のみで、中継の機能はありません。

従って、実際はカスケードされた syslog サーバーの設定がここではなされています。インタラクティブ Syslog ビューアは、その機能を可能にする Syslog プロトコルに対して共通の機能拡張が守られるので、メッセージソースとして、オリジナルのメッセージアドレスを表示することが可能です。

WinSyslog 設定クライアントは、サービスの設定を行う際にだけ必要とされます。一旦その設定がなされると、クライアントは使用する必要がなく、メッセージの処理に必要な要素にはなりません。

上図の設定は、WinSyslog 構成要素がいかに共に機能するかを示すためのものであり、あくまでサンプルです。WinSyslog の設定は、その必要性に応じて他にも多数の方法があります。

1.4 システム必要条件

WinSyslog には最小ではありますが、必要条件があります。

■ WinSyslog クライアント

WinSyslog クライアント は、以下の環境でご利用いただけます:

OS	**Windows 2000 SP3 以降のシステム** **(Windows XP/Server 2003/Server 2008/7/8/8.1/10/Server 2012 (R2 を含む))** ワークステーション、サーバーを問わず 32 ビット版、64 ビット版の両方に対応
メモリ	**6MB**
ハードディスク	およそ 10MB の空き容量が必要
必要ソフトウェア	**インターネット・エクスプローラ 5.5 以降のバージョン** (クライアントは XML を使用します。)

■ WinSyslog サービス

WinSyslog サービスは、以下の環境で動作いたします:

OS	Windows 2000 SP3 以降のシステム (Windows XP/Server 2003/Server 2008/7/8/8.1/10/Server 2012 (R2 を含む)) ワークステーション、サーバーを問わず 32 ビット版、64 ビット版の両方に対応
メモリ	4MB その使用環境により 64MB のメモリ追加を推奨 *1
ハードディスク	およそ 1MB の空き容量が必要 *1

サービスはより小さな必要条件で稼動します。
最も重要な違いは、サービスはシステム上にインターネット・エクスプローラを必要としないということです。

***1 使用にされる実際のリソースは、主に設定されるサービスに左右されます。**
サービスが受信するメッセージが 1 秒間に数件である場合は、パフォーマンスへの影響は顕著ではありません。もし 1 秒間に何百件のメッセージを WinSyslog サービスが受信するならば、より大きなリソースを必要とします。それでも、実際の負荷は実行されるアクションに左右されています。

テキストファイルにメッセージを保存することは、データベース・テーブルにそれらを書き込むこと(特にデータベース・エンジンが同じマシンに置かれる場合)よりもパフォーマンスは大きくありません。
このように、**システム必要条件はハードウェアのサイズというよりは、処理の大きさに左右される**と言えます。

けれども、サービスが(Syslog メッセージなどの)メッセージ・バースト(大量にデータをまとめて伝送する)を含む高度な処理を行う場合は、最も負荷がかかるという点に注意してください。

大量のバーストが予想される場合、それから時間のかかるアクション(例:データベースに書き込む)を実行する場合は、マシンにメモリを追加することをお勧めします。その場合、64MB のメモリの追加を推奨します。典型的に見て一つのメッセージのサイズを約 1.5KB と仮定すると、 64MB 追加した場合は、50, 000 メッセージをバッファリングできます。

また、WinSyslog は、マシンがあまりに時間がかかり過ぎてメッセージの処理を行えない場合、一時的にそのようなバーストをメモリに保存することができます。

■ Adiscon LogAnalyzer

Adiscon LogAnalyzer でログを閲覧するには、IIS(バージョン 4 以降)、または Apache、PHP、データベース

(MySQL 等)が必要です。現在、弊社ではサポート対象外です。

2　はじめに

2.1　インストール

WinSyslog のインストールは単純で簡単です。
WinSyslog のインストーラーは、こちら (https://www.jtc-i.co.jp/support/download/index.php)からダウンロードできます。
インストールセット(ダウンロードした zip ファイル)には、wnsyslog.exe(wnsyslogjp.exe)が含まれています。
アーカイブをどこかのディレクトリ(特に場所は問いません)に解凍してください。解凍先は、ローカルドライブ、リムーバルドライブそれからリモートシェアのファイルサーバーなどでも構いません。なお、Win32 Unzip プログラムは、http://www.winzip.com./ で入手できます。 解凍した後は、「wnsyslog.exe」をダブルクリックし、画面上の指示に従って行ってください。
もしも、wnsyslog.exe ファイルを直接ダウンロードした場合には、解凍の部分の説明は無視してください。(exe ファイルか zip ファイルかは、どこでダウンロードしたかによります)

＜ Adiscon LogAnalyzer のインストール ＞
Adiscon LogAnalyzer は、 WinSyslog や EventReporter で作成したデータベースを web 上に表示するため、Adiscon 社によって開発されたフリーツールです。このツールは、たいていのブラウザに対応しています。
Adiscon LogAnalyzer のご利用には、Web サーバー(IIS 4 以降、Apache など)、および PHP がインストールされたマシンをご用意いただき、PHP スクリプトが実行可能な環境設定を行って頂く必要があります。

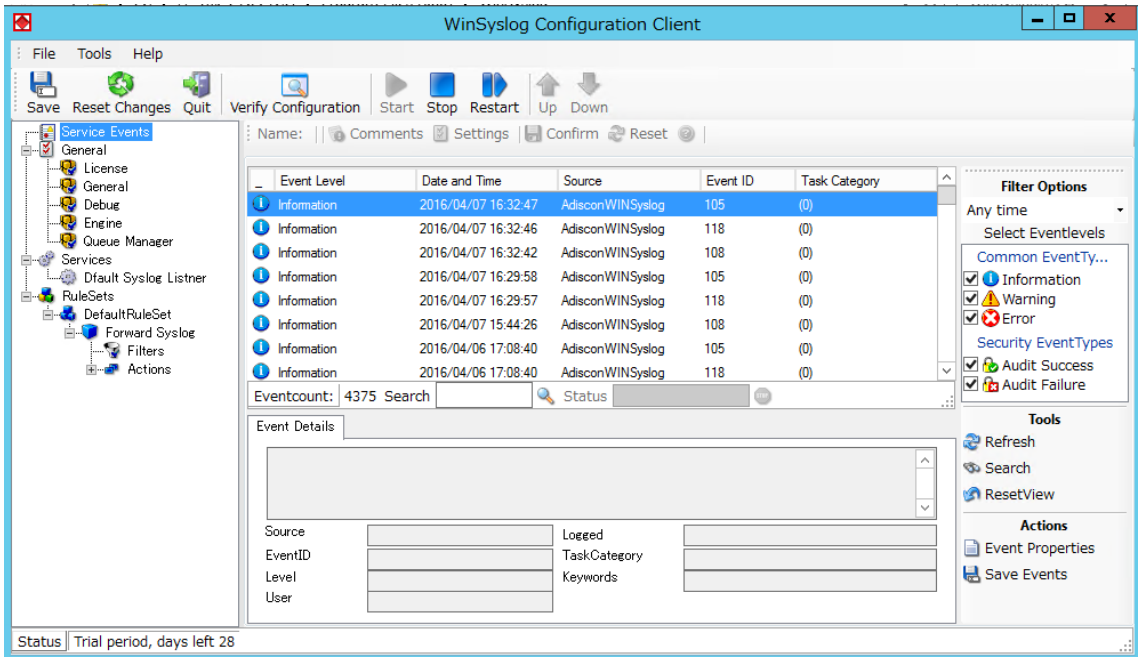
Adiscon LogAnalyzer は、WinSyslog のインストーラーに含まれています。
弊社では、Adiscon LogAnalyzer の直接サポートは行っておりません。
そのため、現在のところ、日本語マニュアルなどはございません。
詳細は、「Adiscon LogAnalyzer」フォルダ配下の「doc」フォルダ内の英語マニュアルをご参照下さい。

2.2　GUI で使用する言語の設定

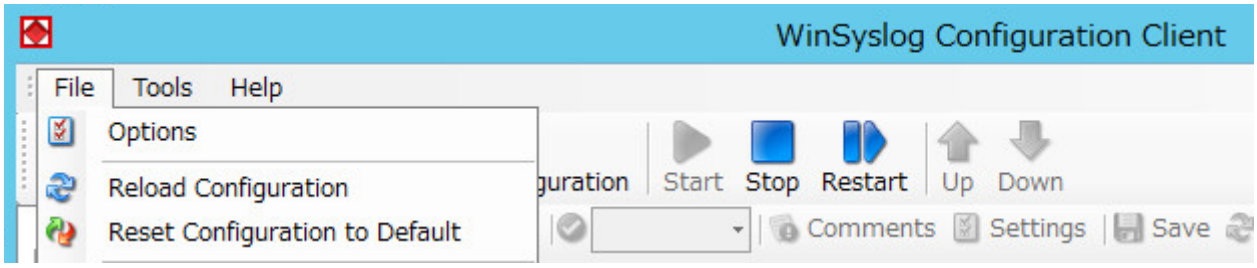
WinSyslog 設定クライアントを起動するには、
アプリケーション一覧より WinSyslog Configuration をダブルクリックして、設定クライアント(WinSyslog Configuration Client)を起動します。
(C:¥Program Files (x86)¥ WINSyslogClient.exe)

すると、下図のようなウィンドウが現れます。:

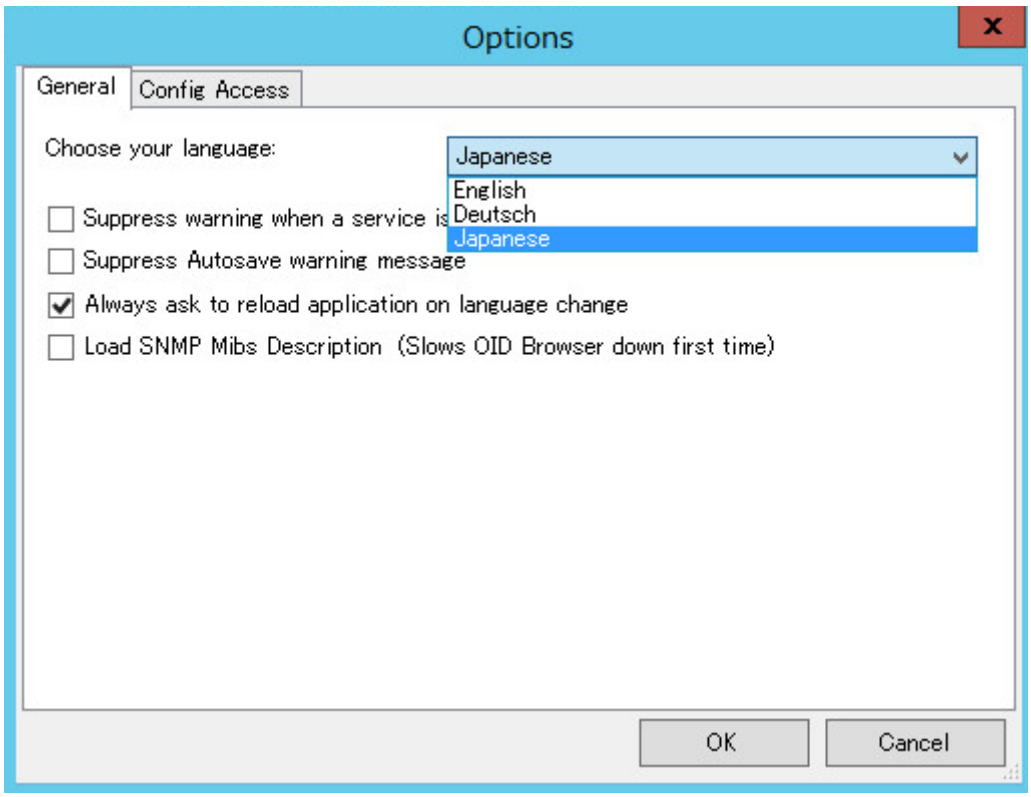
設定クライアント(Configuration Client)　が起動します。最初は英語 GUI　で起動されます。



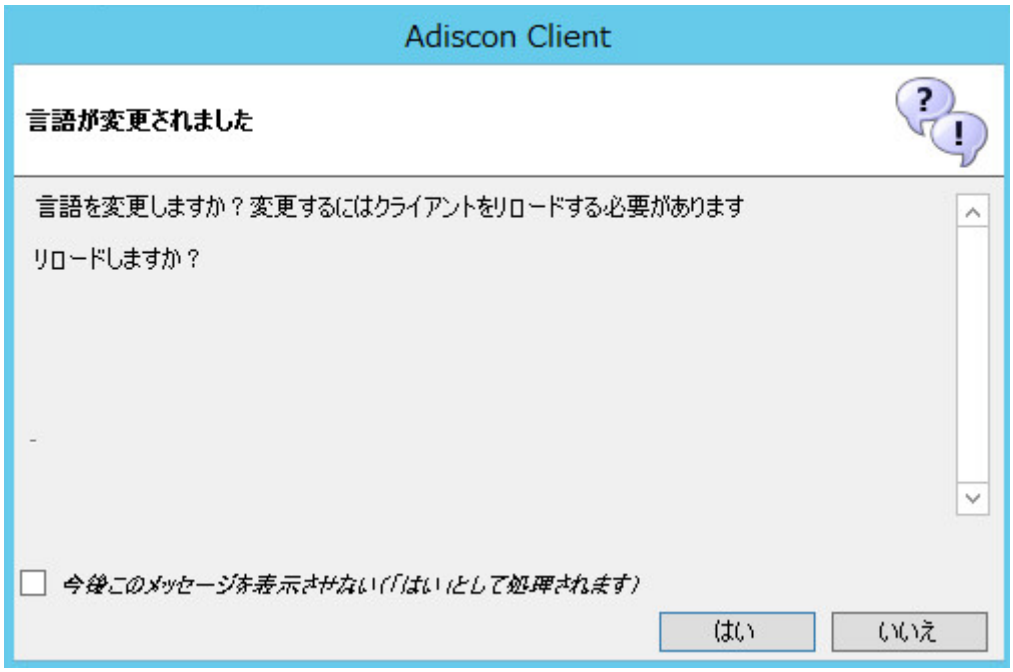
File>Options　を選択します。



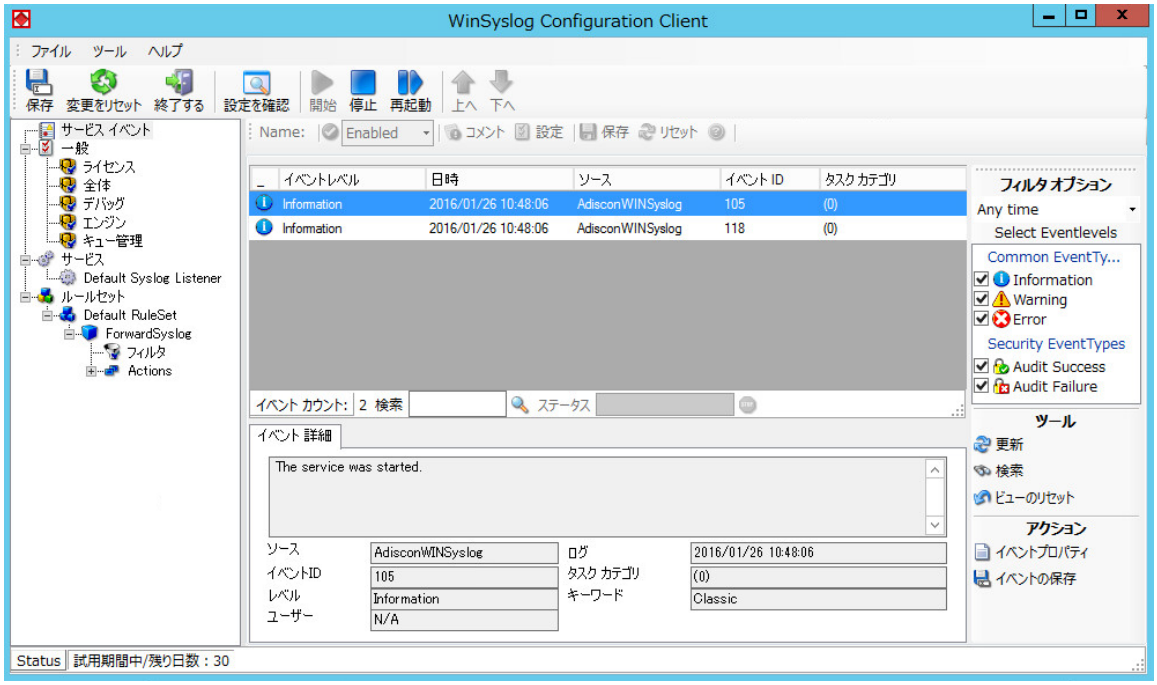
General タブの Choose your language: で Japanese を選択し、OK をクリックします。



はい をクリックすると、クライアントが再起動されます。



設定画面が日本語表示になります。



2.3 初期設定を行う

WinSyslog のインストール後は、動作設定を行う必要があります。
（インストール直後のデフォルト設定では、WinSyslog は、UDP514 で受信したすべての Syslog をインタラクティブ Syslog ビューア 10514 ポートへ転送するルールセットのみ設定されています。）
他の基本的な処理を行うために、以下の作業を行って下さい：

2.3.1 基本のルールセットの作成

最もベーシックなルールセットには、フィルタの条件が設定されておりません。
それは、メッセージを絞り込まずに、WinSyslog で受信した全てのメッセージを処理することを意味します。
はじめは、「ファイルログ」のアクションだけを使用することをお勧めします。
このアクションは、受信したメッセージをローカルのディスクに書き込みます。

2.3.2 最低でも1つの Syslog サーバーサービスを設定

Syslog メッセージを受信するために、Syslog サーバーサービスを設定し、作成したルールセットをこの Syslog サーバーサービスと関連付けるようにして下さい。
（UDP514 ポートで受信する Syslog サーバーサービスは、デフォルトで作成されておりますので、新たに作成する必要はございません）

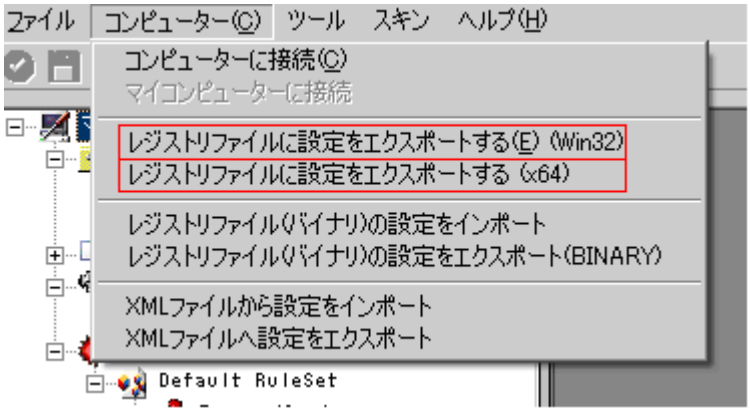
2.3.3 WinSyslog サービスの起動

これでメッセージの受信、保存の準備が整いました。
詳しくは、基本設定解説『標準ログサーバー設定』をご参照ください。

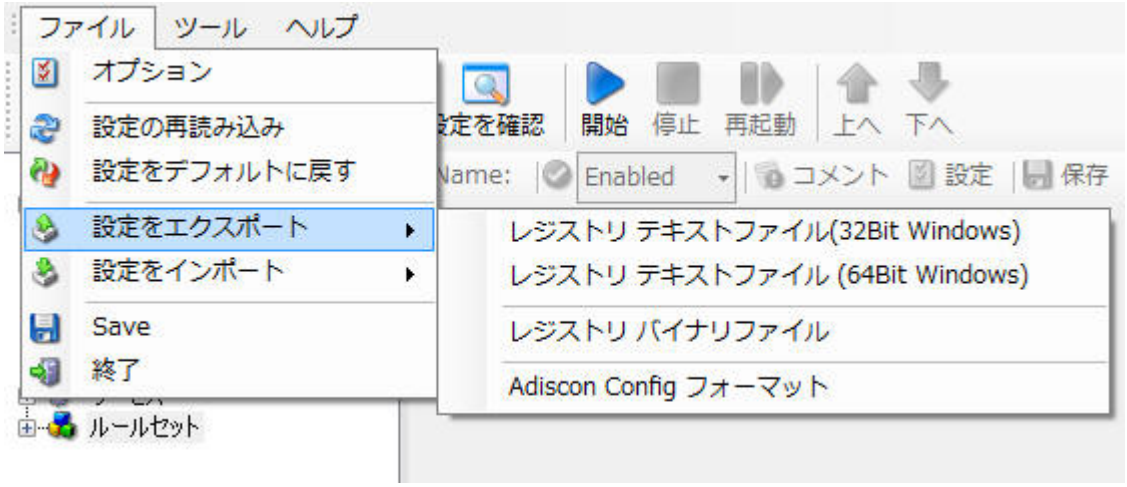
3 設定情報のエクスポート

■ レジストリファイルに設定をエクスポートする

弊社カスタマーポータル宛にお問い合わせ頂いた際、その内容によっては、お客様の設定内容を確認させて頂く場合があります。
(「レジストリファイルに設定をエクスポートする」を実行し、保存して頂いたものを弊社カスタマーポータルより送付してください；下図参照)



▲ 旧設定クライアント―「コンピューター」メニュー



▲ 新しい設定クライアント―「ファイル」メニュー

32bit 版 OS で WinSyslog をご利用の場合には**「レジストリファイルに設定をエクスポートする（Win32）」**、
64bit 版 OS でご利用の場合には**「レジストリファイルに設定をエクスポートする（x64）」**を選択してください。

設定情報のエクスポートは、サポートの場合だけでなく、設定のバックアップを行いたい場合や、複数のマシンで同じ設定をご利用になりたい場合など、様々な状況で役立ちます。

なお、「レジストリファイルの設定をインポート」というオプションはありません。
作成したレジストリファイルをダブルクリックすることで、「レジストリファイルに設定をエクスポート」で保存した設定が読み込まれます。
ダブルクリックすると、「…内の情報をレジストリに追加しますか？」という確認画面が出てきますので、「はい」を選択して下さい。

設定情報のエクスポートの詳細につきましては、簡易マニュアル『設定のエクスポートとインポート』をご参照下さい。

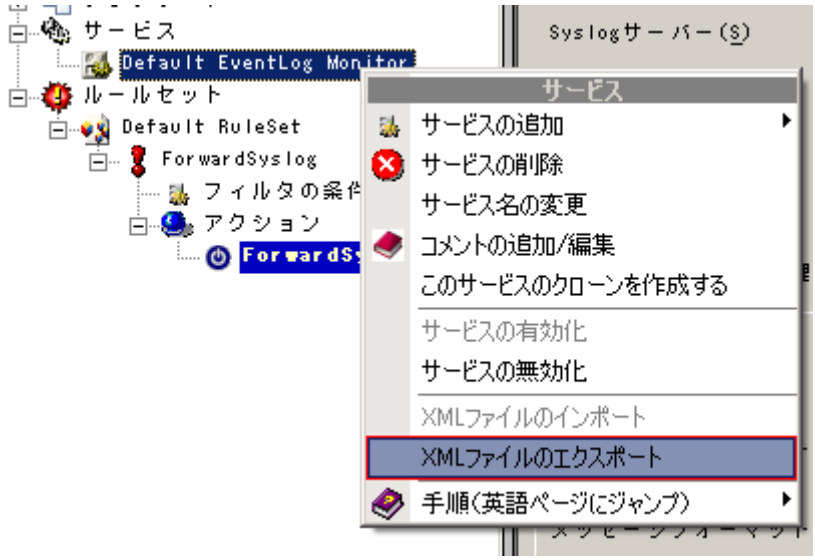
なお、WinSyslog の設定情報は、下記のレジストリキーに保存されております；
32bit 版：HKEY_LOCAL_MACHINE¥SOFTWARE¥Adiscon¥WinSyslog
64bit 版：HKEY_LOCAL_MACHINE¥SOFTWARE¥Wow6432Node¥Adiscon¥WinSyslog
レジストリエディタでもエクスポートは可能です。

■ レジストリファイル（バイナリ）に設定をインポート（エクスポート）する

バイナリフォーマットの設定情報エクスポートは、特別な場合（開発元から送付の依頼が来ない限り）使用しないでください。

■ XML 形式による設定情報のエクスポート・インポート （旧設定クライアント）

XML 形式で設定情報の保存（エクスポート）・読み込み（インポート）を行うことも可能です。
「コンピューター」メニューの「XML ファイルへ設定をエクスポート」では、全てのデータ（ライセンス登録を含む）がエクスポートされます。



▲(旧クライアント)サービスの補助メニュー

もしも、サービスだけ、またはルールセットだけをエクスポートしたい場合には、サービスの補助メニュー(右クリックすると表示される)で「XML ファイルのエクスポート」を実行してください。(上図参照)
ルールセットの場合にも、同様にルールセットの補助メニューから「XML ファイルのエクスポート」を実行してください。

4 InterActive SyslogViewer の使用

4.1 InterActive SyslogViewer(インタラクティブ Syslog ビューア)について

インタラクティブ Syslog ビューアは、Syslog データを簡単に表示させるためのツールです。
EventReporter や WinSyslog から転送された Syslog メッセージを受信し、表示・確認することができるので、監視マシンで起こっていることをリアルタイムに把握することができます。

4.1.1 機能

■ 早くて簡単な Syslog 表示

Syslog ビューアにより、Syslog メッセージの表示・確認が簡単に行えます。従って、監視システムで起こっている問題をより早く発見し、対処することも可能となります。

■ データのエクスポート

受信したデータのうち必要なものだけを選択し、エクスポートすることができます。
データは、テキストファイル、または CSV ファイルとして保存できます。

■ **データベースの読み込み**

データベースビュー機能により、ODBC（32bit）経由で指定したデータベースの内容を表示させることができます。
このデータベースビュー機能では、フィルタ設定を行えます。
それにより、テキストを指定して対象データのみハイライト表示させたり、ファシリティやプライオリティを指定して、対象データのみを表示させたりすることも可能です。

4.1.2 システム必要条件

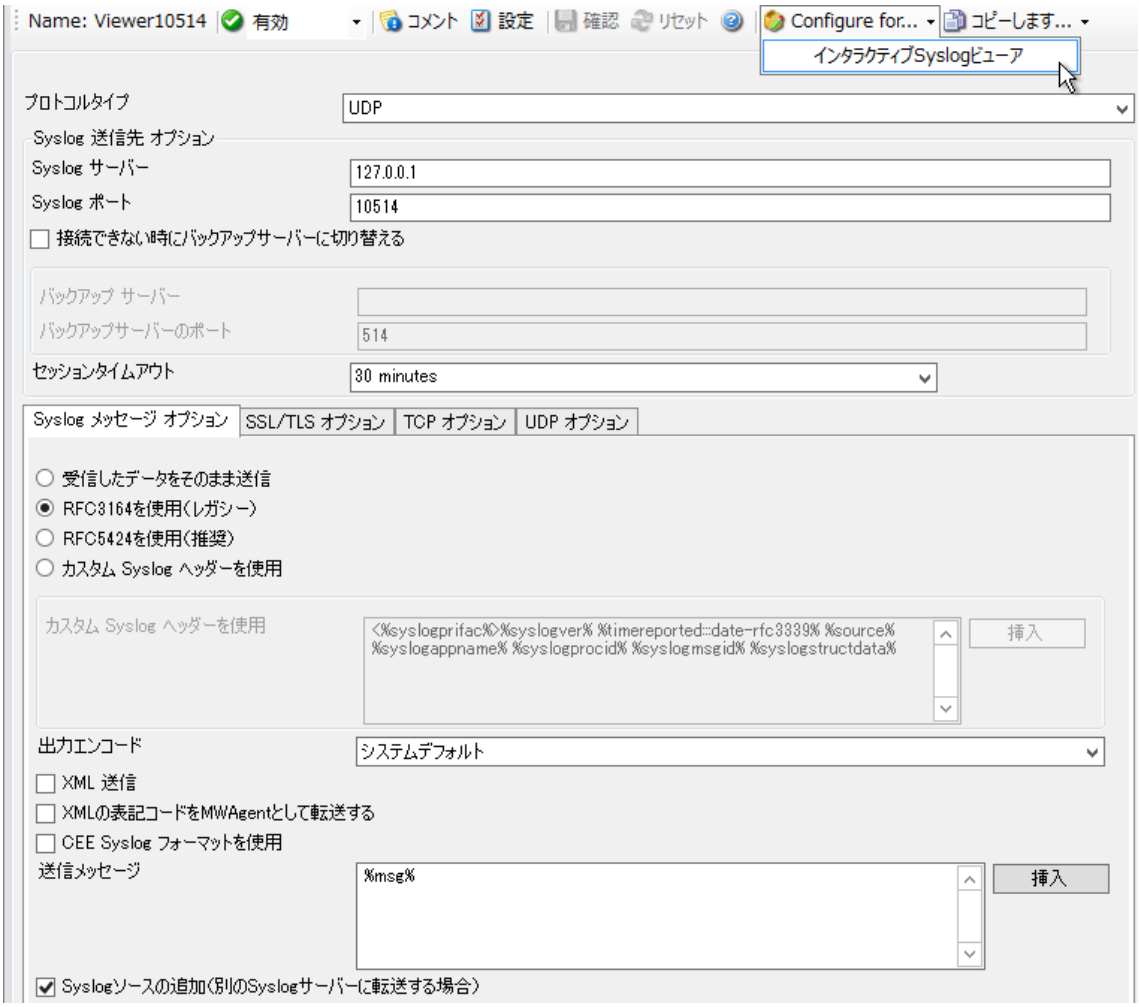
インタラクティブ Syslog ビューア の必要条件は、以下のとおりです；

- Windows 2000、XP、Vista など **Windows NT ベースの OS** 上で動作します
- インタラクティブ Syslog ビューアを起動するには **.NET Framework 2.0（または、それ以降のバージョン）** をインストールする必要があります
- **32MB RAM** 以上のメモリが必要です

4.2 インタラクティブ Syslog ビューアの役割

インタラクティブ Syslog ビューアは、WinSyslog のアドオンツールです。（インタラクティブ Syslog サーバーの後継ツール）
このツールはデフォルト：10514 ポートで受信する Syslog サーバーとして機能しますが、受信 Syslog メッセージをビューアでリアルタイムに表示することを目的としたツールであり、受信ログの保存や転送機能はありませんので、継続してログの監視や保存管理をするには、WinSyslog をご利用ください。

WinSyslog の Syslog 転送アクションで、ローカル IP（127.0.0.1）のポート（デフォルト：10514、複数ポート転送可）に転送することで、インタラクティブ Syslog ビューアでのリアルタイム表示が可能となります。
このアクションは、インストール直後の Default RuleSet のアクションにデフォルトで設定されています。



▲(新クライアント)Syslog 転送アクション

▲(旧クライアント)Syslog 転送アクション

4.3　インタラクティブ Syslog ビューアの起動

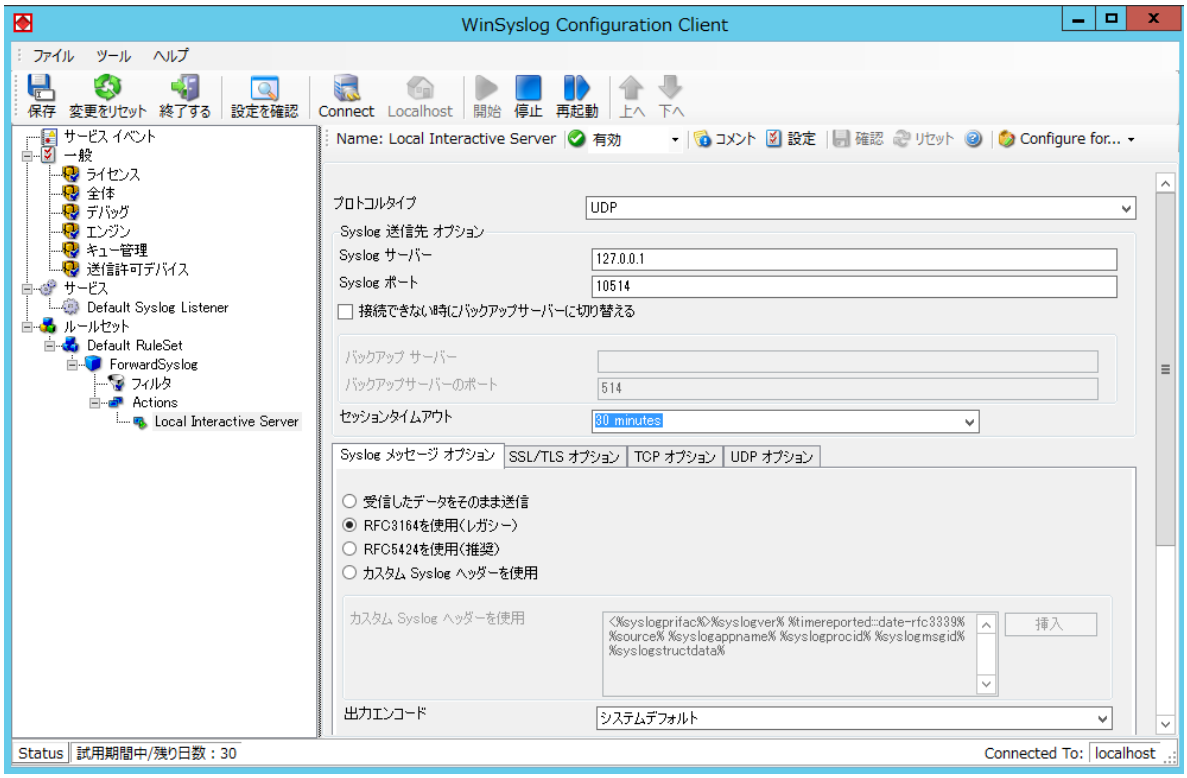
インタラクティブ Syslog ビューアを起動するには、スタートメニューの WinSyslog 配下の InterActive SyslogViewer アイコンをクリックしてください。
コマンドプロンプトからも以下の手順で起動可能です:
・コマンドプロンプトを起動します
・WinSyslog がインストールされているドライブ・ディレクトリに変更します
・InteractiveSyslogViewer.exe と入力しエンターキーを押します
詳細は、インタラクティブ Syslog ビューアのマニュアルをご確認ください。

5　WinSyslog の設定

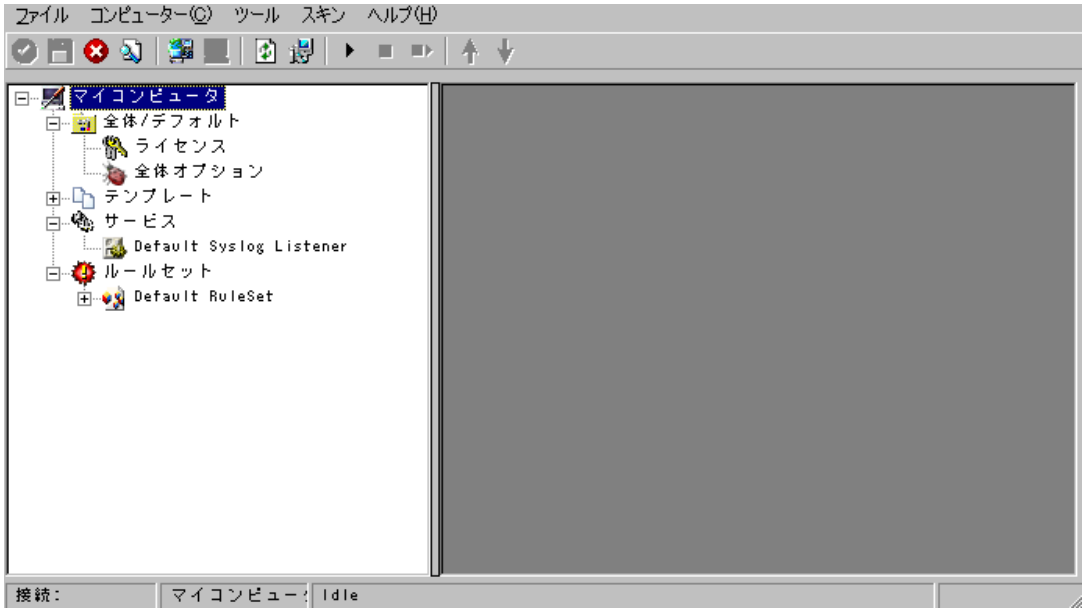
WinSyslog は簡単に操作でき、効果的な製品です。
この章においては、WinSyslog の設定方法について説明します。
WinSyslog の最も重要な部分であるサービスは、一旦設定されるとバックグランドで動作します。
そして、それはユーザーの操作の必要がありません。
従って、この章では、WinSyslog 設定クライアントに焦点を当てて説明します。
クライアントは、サービスの設定を行うために使用されます。

WinSyslog 設定クライアントを起動するには、
スタートメニューまたは、アプリケーション一覧より WinSyslog Configuration をダブルクリックして、設定クライアント（WinSyslog Configuration Client）を起動します。
（C:¥Program Files (x86)¥ WINSyslogClient.exe）

すると、下図のようなウィンドウが現れます。:



▲新設定クライアント



▲旧クライアント（旧設定クライアント）

設定クライアントには、2 つの要素があります。

左側には、WinSyslog システムのそれぞれの要素を選択するツリー表示があります。

右側には、ツリー表示で選択されたパラメーターが表示されます。

（旧クライアントの）ツリー表示には、トップに「全体/デフォルト 」・「テンプレート」・「サービス」
・「ルールセット」の 4 つの要素があります。
「全体/デフォルト」では、基本的な操作上のパラメーターの設定とライセンス情報の登録を行います。

「テンプレート」では、デフォルト値を設定します。（よく使用する値を入力することをお勧めします）
ここでは、特殊なインスタンス（例）は決定しません。
デフォルト値は、実際の設定で個別のサービスやアクションを変更（上書き）することができます。

「サービス」のツリー表示には、設定されたサービスとそのパラメーターがあります。
WinSyslog には、「Syslog サーバー」、「ハートビート」、「MonitorWare Echo Reply」、「SNMP トラップ受信」、「SETP サーバー（エンタープライズエディションのみ対応）」の 5 つのサービスを作成できます。
サービスの作成数に制限はありませんが、同じ設定内容のサービスは、複数稼動させることはできません。
同じ種類のサービスを複数作成する場合には、ポートの衝突を避けるように設定を行って下さい。

Syslog サーバーサービスならば、同じポート（例：514）を使用しているがプロトコルタイプが違うもの、違うポート（例：515）を使用しているものを設定すれば、同じ種類のサービスを同一システム上で3つ作成し、稼動させることも可能です。

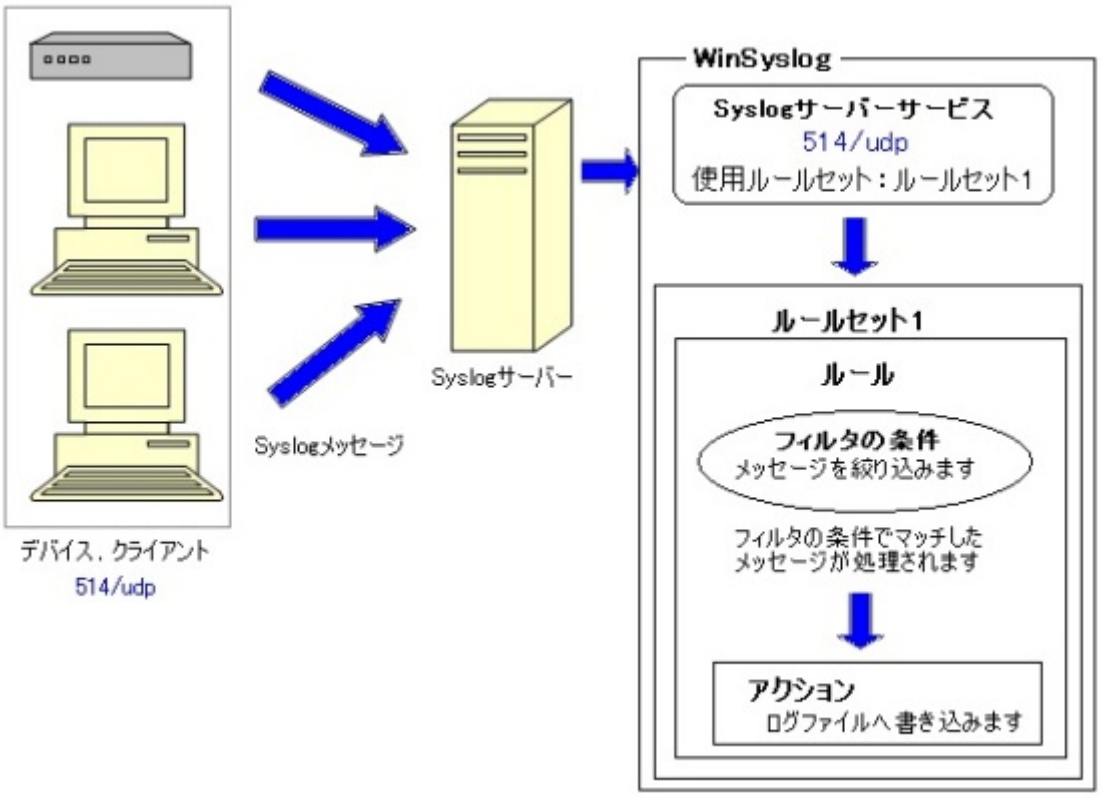
もしも、同じポートで同じプロトコルを設定したSyslogサービスが存在する場合には、複数のSyslogサービスのインスタンスが実行されているという内容のエラーが Windows のイベントログに記録されます。
例として、以下のような エラーがイベントログに記録されます。

イベントの種類	警告
イベント ソース	AdisconWinSyslog
イベント カテゴリ	なし
イベント ID	1001
説明	A configured syslog server service can not be started. Most often, this happens when more than one syslog server service is configured to use the same port and protocol, e.g. 514/UDP. Please make sure that only a single syslog server is defined to listen on the same port and protocol. If you would like to do multiple actions, this can be done within a single rule set that is bound to a single syslog server service. The socket subsystem reported the following reason: "Can't bind to socket - will keep retrying..." Additional help might be available at http://www.adiscon.com/EventHelp.asp （設定した Syslog サーバーサービスは、実行できません。これは、同じポートやプロトコルを使用するよう（例：514/UDP）設定された Syslog サーバーサービスが複数存在する場合に、起こりえます。もし、複数のアクションを実行したい場合には、一つのルールセットを一つのサービスに関連付けるように設定を行ってください ・・・）

理論的には、数百ほどのサービスを追加できますが、オペレーティングシステムのリソースや取り扱いについての観点から、最大でも20から30までのサービスに数を制限することをお勧めします。もちろん、この制限より多くのサービスが有効である場合もあります。
WinSyslog 自体は、サービスの作成数に制限はありません。
数多くのサービスを必要とし、ハードウェアがその処理に耐えうる場合は、数の制限は必要ありません。
実際のパラメーターは、サービスの種類に左右されます。
サービスの有効化・無効化は全てのサービスに共通です。
サービスの有効化によって、サービスは機能します。無効化は、サービスの設定がされていても、それを実行しません。それにより、削除をしなくても簡単にサービスを一時的に使用不能にすることができます。

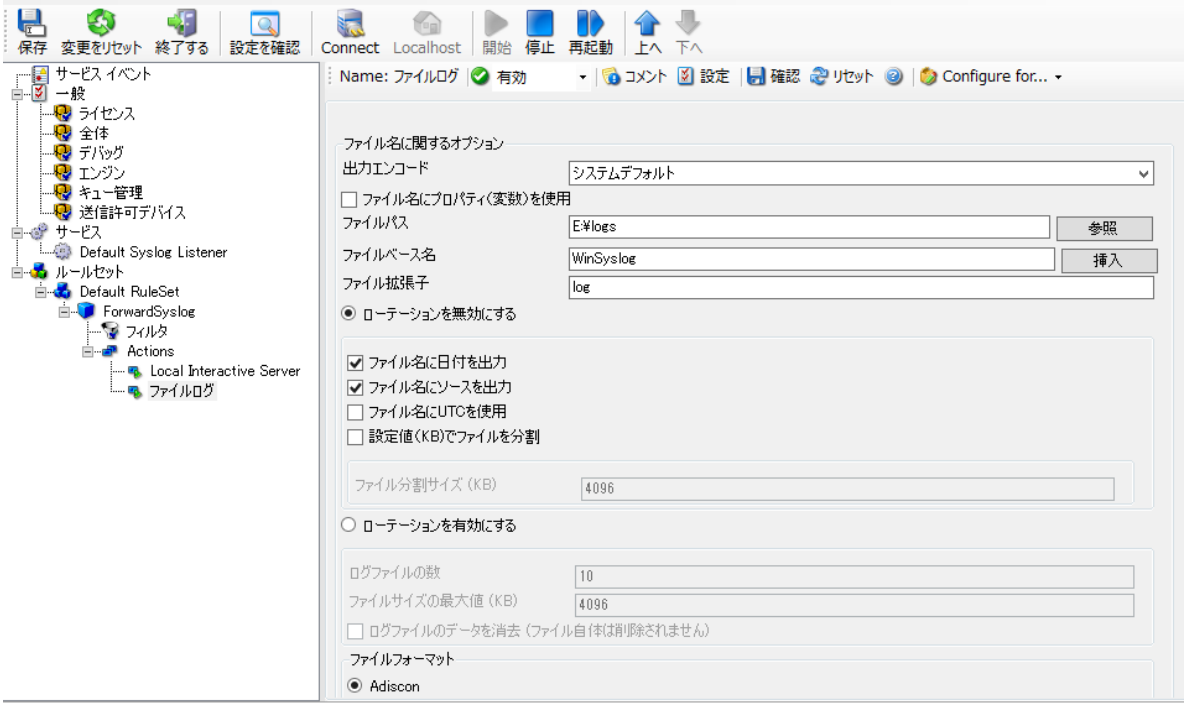
同様に、右側の設定ダイアログの下にある「使用するルールセット」も、サービスの種類に関係なく共通のものです。どのサービスに対して、どのルールセットを実行するのかをここで指定します。
新しくサービスを作成する場合には、「サービス」を右クリックして下さい。
それから、「サービスの追加」を選択し、さらにポップアップメニューからサービスの種類を選択します。
サービスを削除したい場合は、その対象のサービスを右クリックし、「サービスの削除」を選択します。
一時的に削除したい場合には、「サービスの無効化」の設定を行って下さい。

ツリー表示の最後の要素は**「ルールセット」**です。
ここで全てのルールセットの設定を行います。それぞれのルールセットは、完全にお互いから独立しています。
ルールセットは、サービスと組み合わせて使用します。ルールセットの配下には、ルールを作成します。さらに、ルールの下には、それに関連するフィルタとアクションの条件があります。
別途ご説明しますが、ルールは非常に重要な機能です。ルールは最上位のあるものから順に実行されます。
ルールを上や下に移動するには、移動したいルールをクリックして「上へ  」や「下へ  」のボタンをクリックするか、ドラッグ＆ドロップで移動します。 下図を例に、WinSyslog で Syslog メッセージが処理される流れの概要を説明します:



例として、WinSyslog で受信した Syslog メッセージをログファイルに書き込む場合、Syslog メッセージは下記のように処理されます：

1. Syslog サーバーサービスで Syslog メッセージを受信
   （Syslog メッセージ送信側と受信側で通信の設定を合わせてください）
2. 1 のサービスの「使用するルールセット」で指定されたルールセットへメッセージが渡されます
3. 2 のルールセットのルール（複数ある場合には上にあるものから順に）に渡されます
4. 3 のルール内のフィルタの条件を適用
   （設定されていない場合には、全てのメッセージでアクションが実行されます。）
5. 4 のフィルタ条件に合致したメッセージに対して、その配下にあるアクション（上図の場合「ファイルログ」アクション）が実行されます



上のスクリーンショットは、前頁のサンプルの設定画面です。

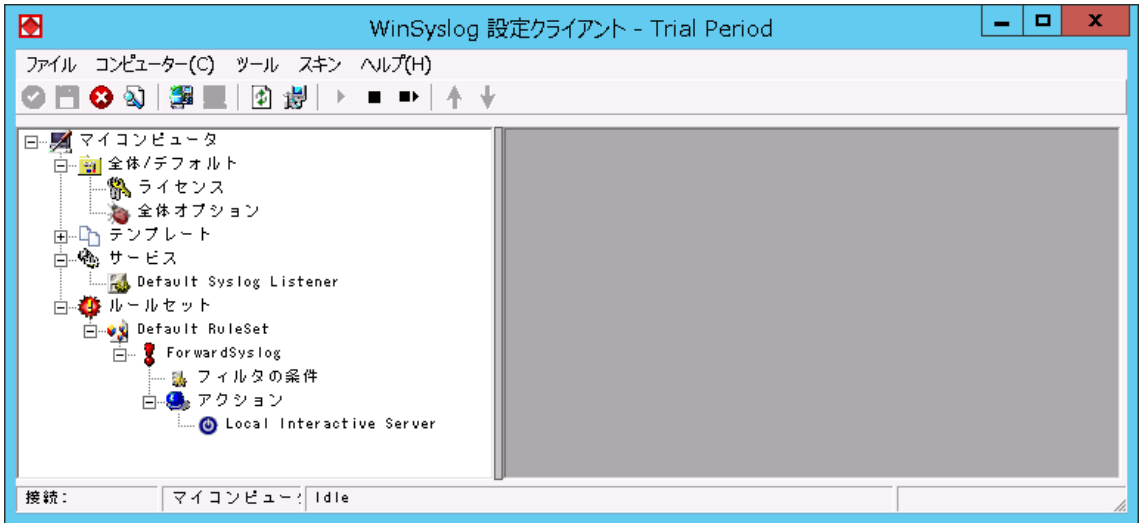
5.1 クライアントオプション

WinSyslog では、WinSyslog 13.1 で実装された現在のクライアント（新クライアント）と旧クライアント（レガシークライアント）が利用できます。
この章では、これら設定クライアントの「ファイル」内にある「オプション」につきまして、以下に説明します。

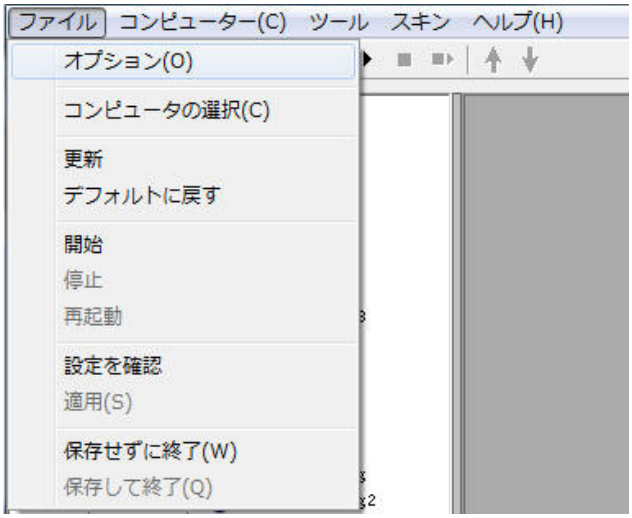
5.1.1 レガシークライアントのオプション（Legacy Client をインストールした場合）

スタートから、「WinSyslog Legacy Client」をクリックします。

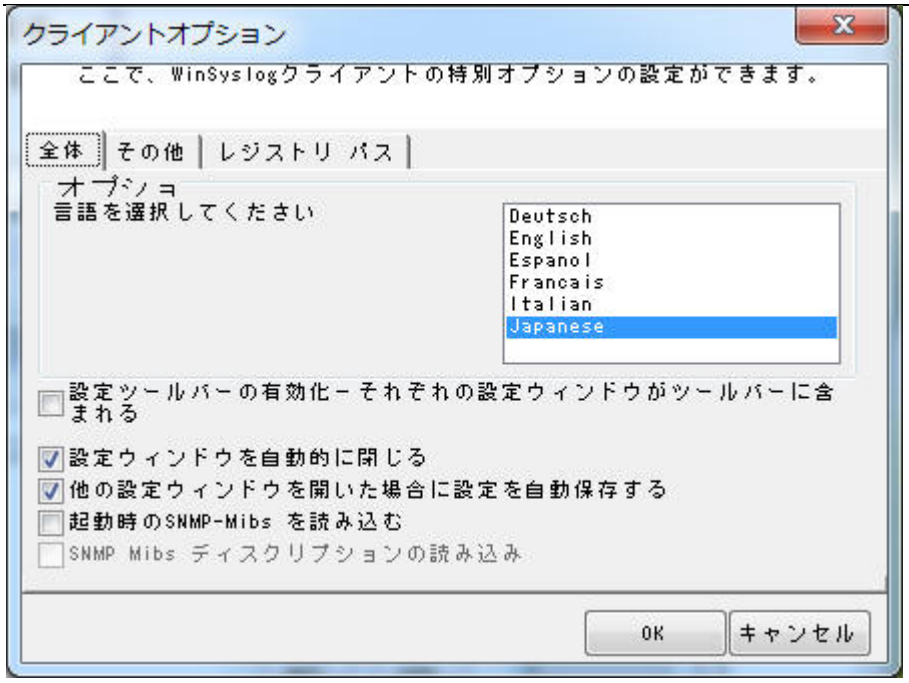
旧クライアント（旧設定クライアント）が起動します。



旧クライアントの「ファイル」内にある「オプション」につきまして、説明します。



＜「全体」タブ＞



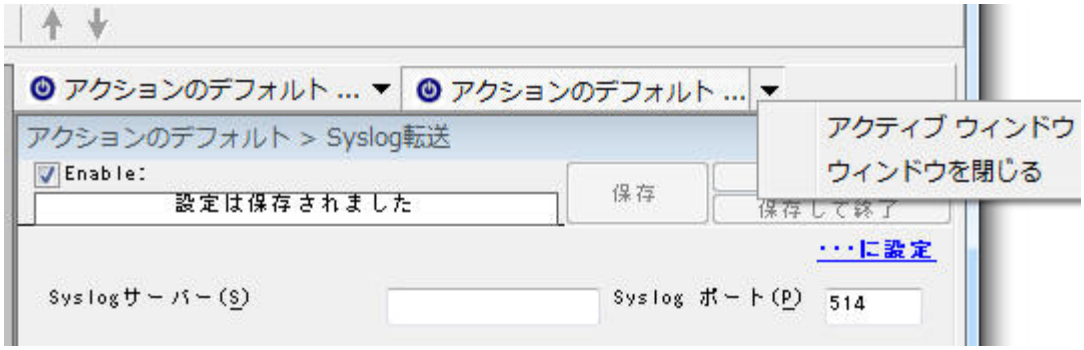
■ 言語を選択してください

設定クライアントで表示させる言語をここで指定します。

■ 設定ツールバーの有効化

ここを有効にすると、クライアント右側上部に開いた設定ウィンドウがツールバーとして表示されるようになります。（下図参照）ツールバーで特定の設定を選択すると、それが最前面に表示されます。

この機能は、「設定ウィンドウを自動的に閉じる」が無効と時に役立ちます。



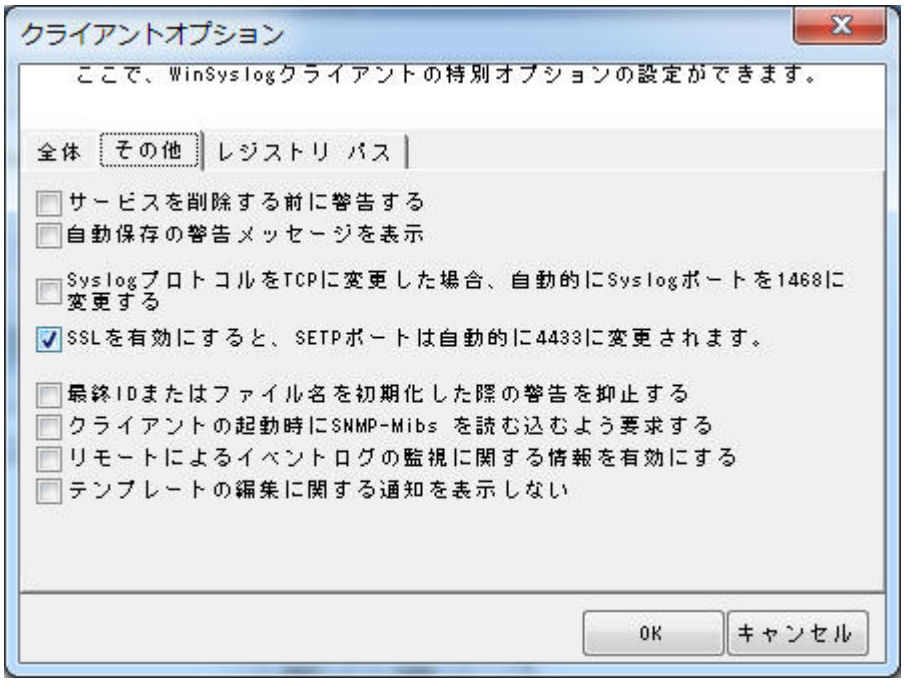
■ 設定ウィンドウを自動的に閉じる

ここを有効にすると、一つの設定が終わり、別の設定に移るときに、前の設定ウィンドウが自動的に閉じるようになります。

■ 他の設定ウィンドウを開いた場合に設定を自動保存する

ここを有効にすると、設定中に別のウィンドウを開いた際、それまで設定していた未保存のデータが自動的に保存されるようになります。

＜「その他」タブ＞



■ サービスを削除する前に警告する

ここを有効にすると、サービスを削除する際に表示される警告画面が表示されなくなります。

■ 自動保存の警告メッセージを表示

ここを有効にすると、設定ウィンドウを切り替える際未保存のデータがあることを警告する画面が表示されます。

■ Syslog プロトコルを TCP に変更した場合、自動的に Syslog ポートを 1468 に変更する

ここを有効にすると、WinSyslog の「Syslog サーバー」サービスで受信プロトコルを TCP に変更した際、ポート番号を 1468 に変更する警告画面が表示されるようになります。

■ SSL を有効にすると、SETP ポートは自動的に 4433 に変更されます

ここを有効にすると、SETP で SSL を有効にした際にポート番号を変更する警告画面が表示されるようになります。

■ 最終 ID またはファイル名を初期化した際の警告を抑止する

ここを有効にすると、最終 ID またはファイル名を初期化した際の警告画面が表示されなくなります。

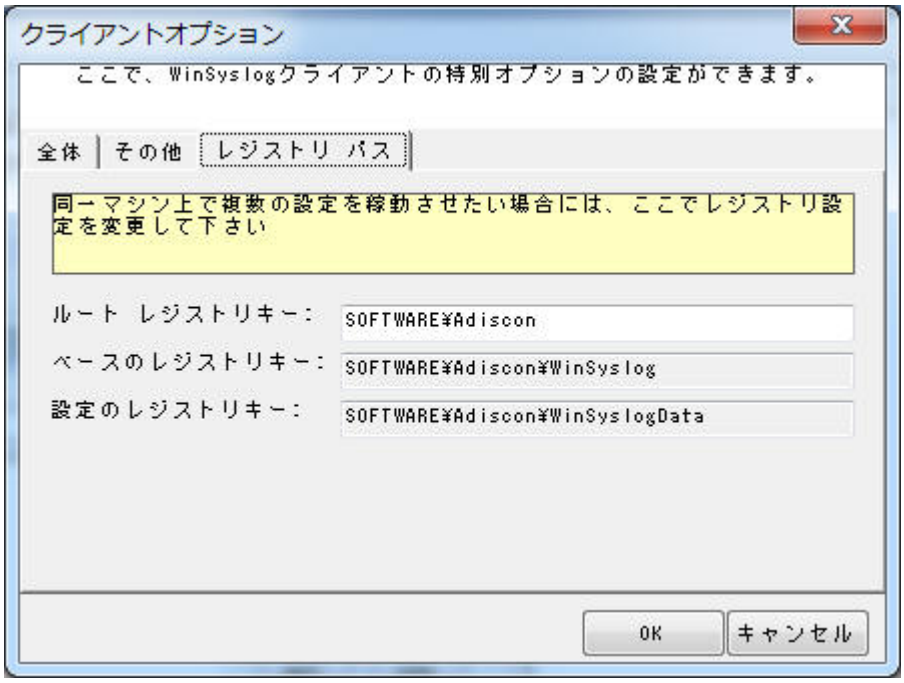
■ リモートによるイベントログの監視に関する情報を有効にする

ここを有効にすると、イベントログの監視サービスでリモートのイベントログを設定した際の警告画面が表示されます。

■ テンプレートの編集に関する通知を表示しない

ここを有効にすると、テンプレート（各設定のデフォルト値）を編集する際の警告画面が表示されなくなります。

＜「レジストリ　パス」タブ＞

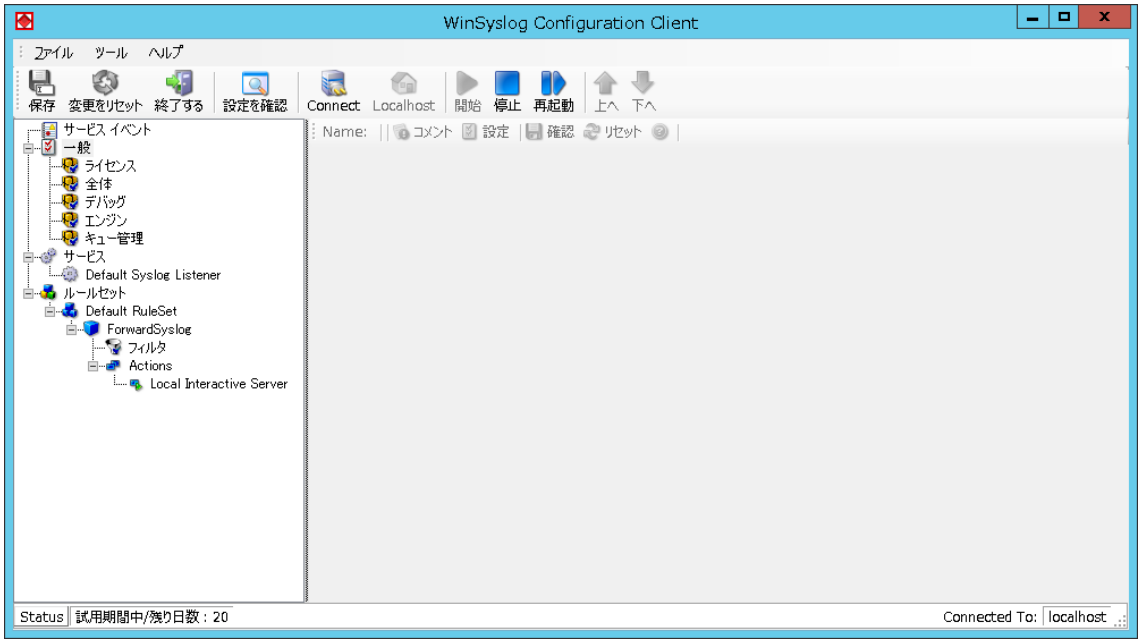


設定を読み込むレジストリのパスを変更することができます。

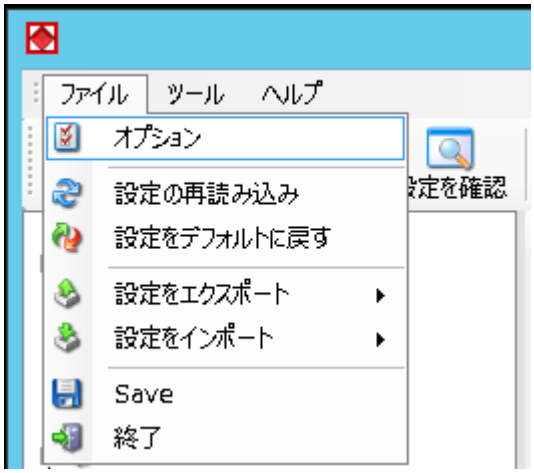
5.1.2　設定クライアントのオプション（新設定クライアント）

スタートから、「WinSyslog Configuration」をクリックします。

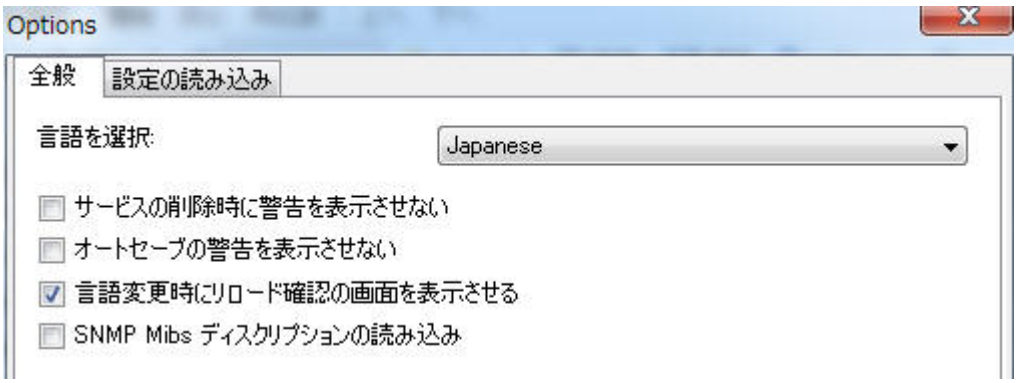
設定クライアント(WinSyslog Configuration Client)が起動します。



設定クライアントの「ファイル」内にある「オプション」につきまして、以下に説明します。



＜「全般」タブ＞



■ 言語を選択

設定クライアントの表示言語をここで指定します。

デフォルトは英語になっています。

■ サービスの削除時に警告を表示させない

サービスを削除する際の確認メッセージを表示させたくない場合には、ここを有効にしてください。

無効の時は、削除時に「サービスを削除してもよろしいでしょうか？」という確認メッセージが表示されます。

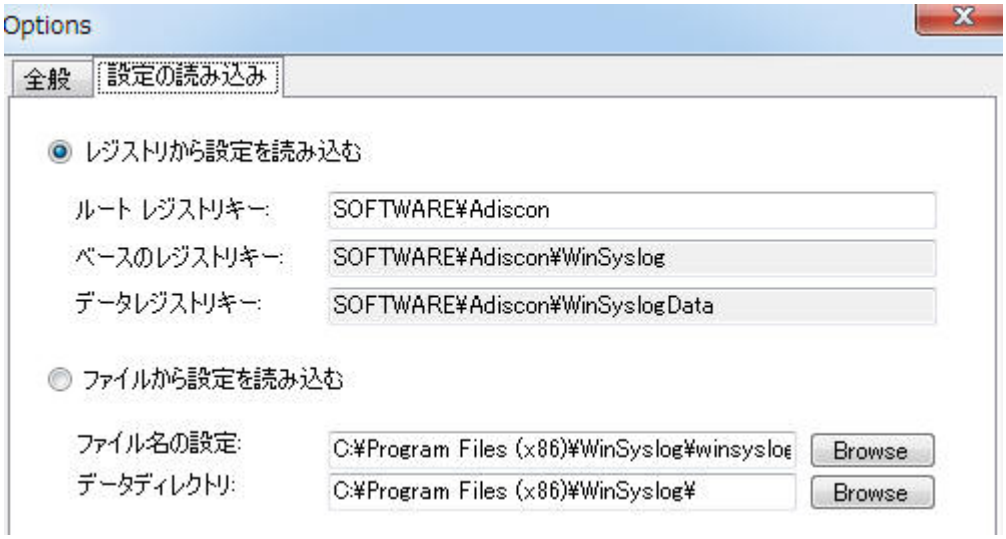
■ オートセーブの警告を表示させない

設定変更後、保存せずに別の設定に移るときに表示される確認メッセージを表示させたくない場合には、ここを有効にしてください。

■ 言語変更時にリロード確認の画面を表示させる

設定クライアントの表示言語変更時の確認画面を表示させたくない場合には、ここを無効にしてください。

＜「設定の読み込み」タブ＞



■ レジストリから設定を読み込む

ここで指定したレジストリキーから設定を読み込みます。(デフォルト)

■ ファイルから設定を読み込む

ここで指定したファイル(.cfg ファイル)から設定を読み込ませることも可能です。

5.2 全体オプション

5.2.1 ライセンスの設定

試用期間(30 日間)終了後、WinSyslog を製品版として継続して使用するには、ライセンス登録をして頂く必要があります。

ライセンス情報は、ご購入後にメールで送付させていただきます。(図 1 参照)

ライセンスは、テキストファイルで、以下のような形式で支給されます。

Product: WinSyslog Professional

Version: 13

Licensee Name: "XXX Corp." (without the quotes)

License Key(s): 11111-22222-333333-444444-555555

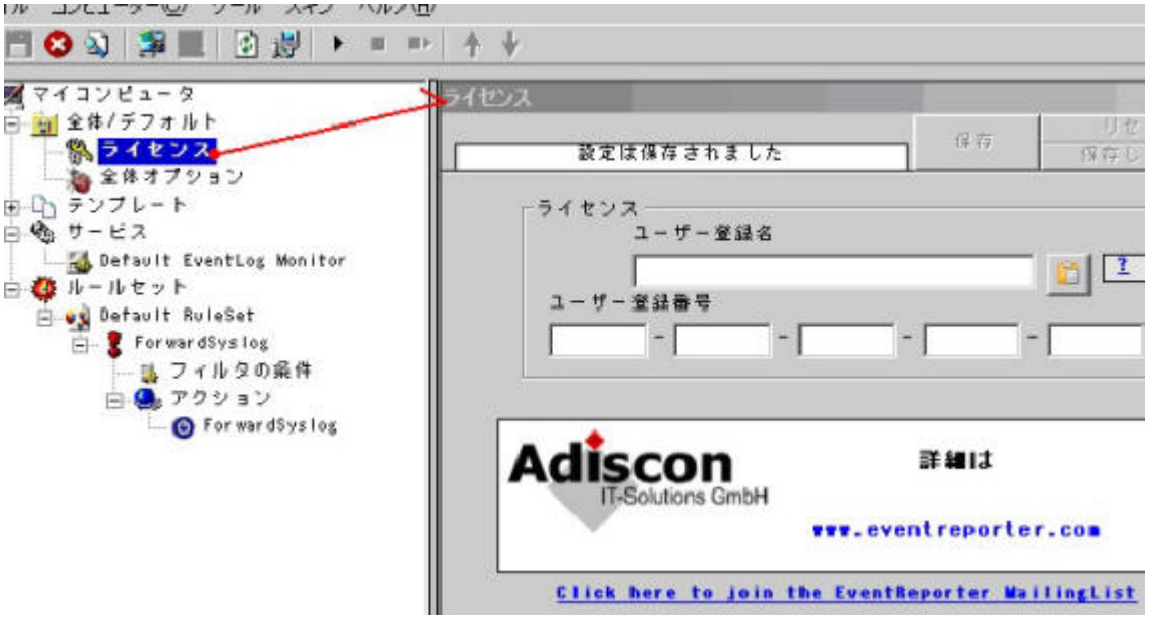
Licensed Copies: 1

Licensed Clients: 100

▲図 1 ライセンス情報のサンプル(抜粋)

ライセンス情報メールにある「ユーザー登録名」と「ユーザー登録番号」を設定クライアントの「ライセンス」で入力・保存することで製品版として動作するようになります。(図 2、図 3 参照)

登録後は、WinSyslog サービスを再起動してください。



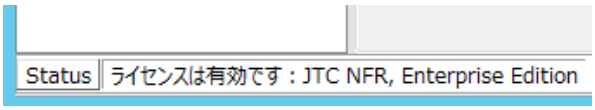
▲図2 ライセンス登録画面（旧設定クライアント）

▲ 図 3 ライセンス登録画面（新設定クライアント）

正常に登録されると、アプリケーションログにイベントが出力されます。（例 / ID：118、詳細：EventReporter Basic is running in registered mode・・・）←下線部には製品名が入ります。

▲ライセンス登録画面（旧設定クライアント）

また、「？」にカーソルを合わせると、上図のように登録された
ライセンス情報が表示されます。
ただし、こちらは従来の旧クライアントのみの機能となります。



▲ライセンス登録画面（新設定クライアント）

新設定クライアントでは、ライセンス情報は下部 Status バーに表示され、バージョン情報は、
ヘルプ＞About WinSyslog をクリックすると表示されます。

■ ユーザー登録名

ユーザー登録名は、ご注文時に「エンドユーザーライセンス申請フォーム」に記載していただきましたユーザー情報の「会社名（英語）」となります。
ライセンス発行後は、登録名の変更はできませんのでご注意ください。

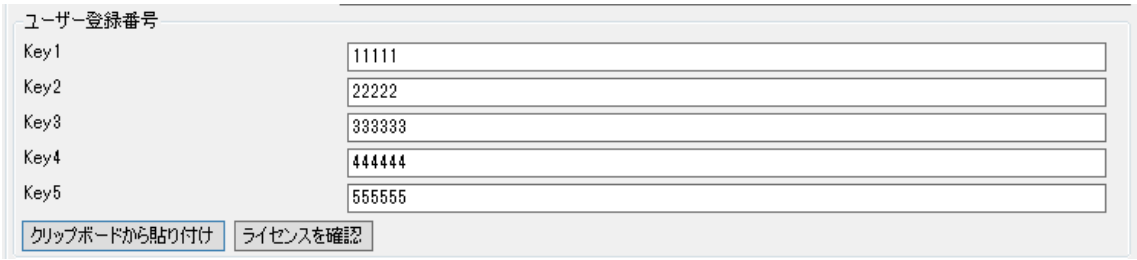
■ ユーザー登録番号

この番号は、ご注文後 Adiscon 社によって発行されます。（ユーザー登録名に対して、固有のユーザー登録番号が発行されます。）
ユーザー登録名とユーザー登録番号がライセンスキーとなり、これらを設定クライアントにてご登録頂くことで、製品版として試用期間終了後も継続して動作するようになります。

ライセンス登録では、必ず正しい登録番号を入力するようにしてください。
ご登録内容に間違いがある場合には、製品版として登録されません。
設定クライアントの「クリップボードからの貼り付け」を使用するのが便利です。

例：支給されたライセンス情報内の License Key で、
License Key(s): 11111-22222-333333-444444-555555

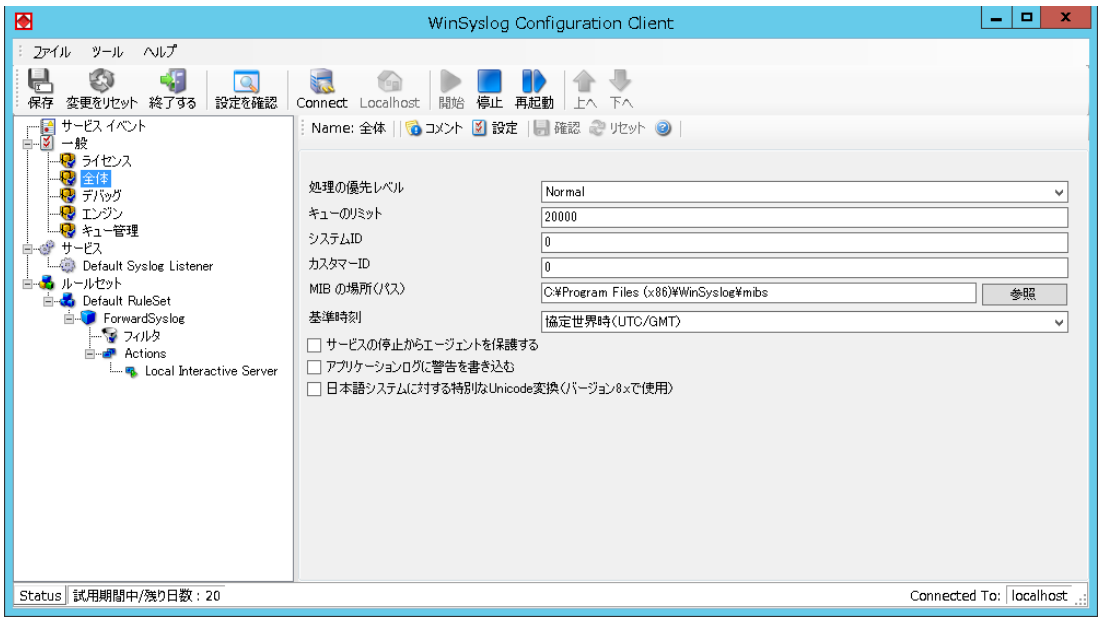
四角で囲んだ数字と － の部分をクリップボードへコピーし、
「クリップボードから貼り付け」をクリックします。
以下のように、番号がペーストされます。



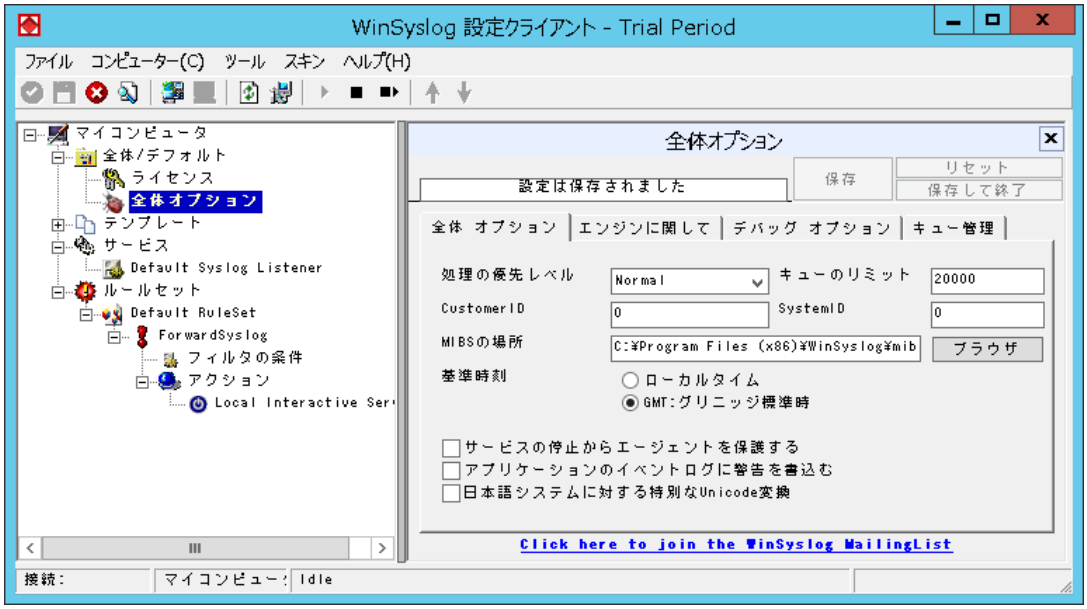
「ライセンスを確認」をクリックし、「有効なライセンスです」とウィンドウに表示されれば、認証されました。保存アイコンをクリックして、変更を保存します。
「インストールとライセンス登録」もご参照下さい。

5.2.2 「全体」オプション

設定クライアント： 一般＞全体

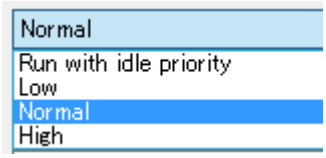


旧クライアント：マイコンピュータ＞全体/デフォルト＞全体オプション
＜「全体 オプション」タブ＞



■ 処理の優先レベル

ここでは、WinSyslog の処理の優先レベルを設定できます。



■ キューのリミット

ここでは、WinSyslog の処理時における キューの最大値が設定できます。

デフォルトは、200000 です。「0」にすると、無制限になります。

■ Customer ID

顧客によって Customer ID を変更したい場合には、ここで 整数値を入力します。

例えば、顧客のサーバーを監視する際に、会社ごとに違う ID を設定することが可能です。

サーバー A と B を監視しているとして、5 台あるサーバー A は Customer ID を 1 、2 台あるサーバー B の Customer ID を 2 といった具合で設定することが可能です。

サーバー A と B のサーバー名が同じ場合でも、Customer ID を設定すれば別の定義を行うことが可能です。

■ System ID

System ID を変更したい場合には、ここで整数値を入力します。

■ MIB の場所(パス) / MIBS の場所

ここでは、MIB ファイル の場所を指定します。パスを指定するか、またはブラウザより選択してください。

■ **基準時刻**

ここでは、ファイルログやデータベースログ、E メール送信など、WinSyslog 全体で使用するタイムスタンプの設定を行えます。UTC、またはローカルタイムの指定を行えない設定項目に対しても、ここで選択したタイムスタンプが設定されます。

ただし、プロパティ値（%timegenerated% や %timereported%）の基準時刻には、この設定値は反映されません。

■ **サービスの停止からエージェントを保護する**

サービスは、未処理のイベントをインメモリ キューに保存します。サービスが停止すると、このインメモリ キューは、空になります。この場合、未処理イベントは消失してしまいます。

このオプションを有効にすると、サービスの停止前に全てのイベントが確実に処理されます。

しかし、その処理中は、サービスがハングアップしたかのような状態になります。

このオプションは、大きな インメモリ キューがある場合には、有効なケースとなります。

■ **アプリケーションのイベントログに警告を書き込む**

ここを有効にすると、Windows アプリケーション イベントログへ警告を記録できるようになります。

■ **日本語システムに対する特別な Unicode 変換**

日本語のシステムにおいては、文字の処理方法が異なります。

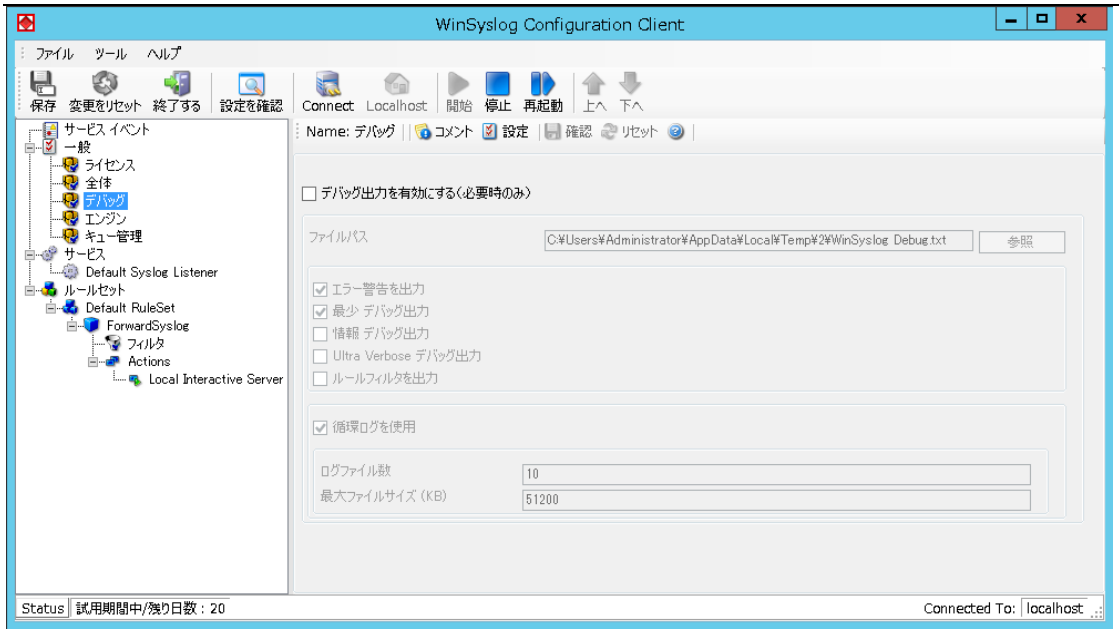
日本語文字列がメッセージに含まれる場合、以下のように設定してください。

WinSyslog 8 をご利用の場合には、ここを有効にしてください。

それ以外のバージョンでは無効のままご利用ください。

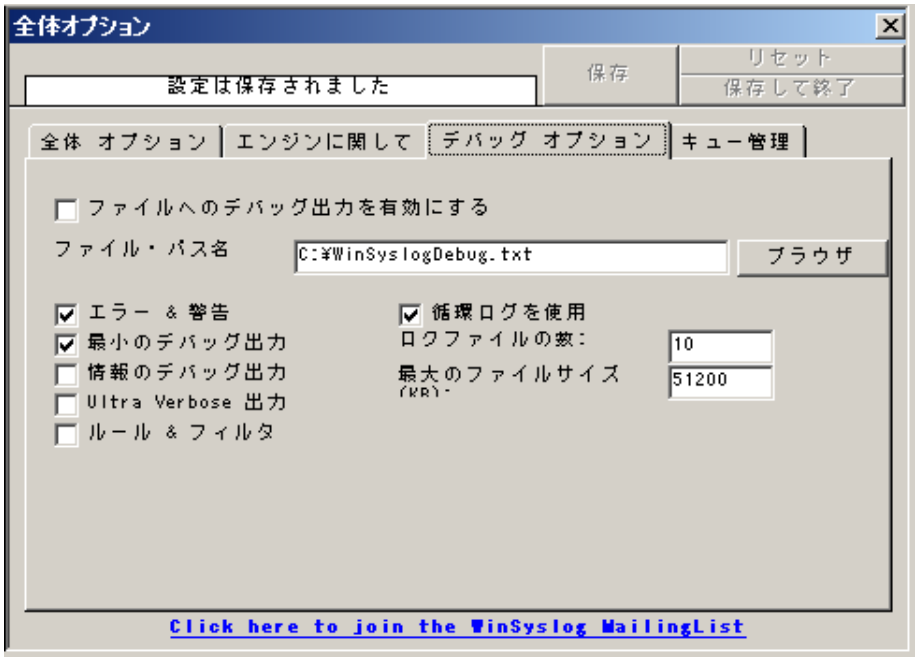
5.2.3 「デバッグ」オプション

設定クライアント：一般＞デバッグ



▲新クライアント「デバッグ」オプション

旧クライアント：マイコンピュータ＞全体/デフォルト＞デバッグ　オプション



▲旧クライアント「デバッグ」オプション

ここでは、ルールベースのデバッグを行うことが可能です。
複雑なルールベースの場合は特に、その処理が実行されている間 WinSyslog が内部で何を行っているかを知る必要があります。デバッグのログにより WinSyslog の内部での働きを知る事ができます。
ルールベースのテストとは別に、このデバッグのログは技術的なお問い合わせの際に役に立ちます。
状況によっては、お問い合わせ時に問題を解決するため、特定のレベルに設定を変更していただく場合があります。

重要: デバッグ出力は、かなりのシステムリソースを必要とします。
ログのレベルが上がるにつれ、より大きなリソースが必要となります。
しかし、最も低いレベルに設定しても、WinSyslog の処理はかなり遅くなります。
従って、**必要な時以外は、このオプションは使用しないでください**

■ デバッグ出力を有効にする（必要時のみ）
（ファイルへのデバッグ出力を有効にする）
ここを有効にすると、デバッグのログが可能になり、サービスが稼動する際に書き込まれます。
＊パフォーマンスを考慮して、普段は ここを無効のままにして置いてください

■ ファイルパス（ファイルパス名）
書き込みを行うログファイルの完全な名前を設定します。
ドライブを含めた完全なパス名を指定するようにして下さい。
ファイルまたはパス名だけを入力すると、その入力情報はローカルのサービスのデフォルト・ディレクトリを参照します。整合性を考慮して、ドライブを含めた完全で適したファイル名を指定するようにして下さい。

＜記録される内容＞
■ エラー警告を出力（エラー＆警告）
ここを有効にすると、エラーと警告がデバッグログへ出力されます。

■ ルールフィルタを出力（ルール＆フィルタエンジン）
ここを有効にすると、ルールとフィルタエンジンがデバッグログへ出力されます。

■ 最小 デバッグ出力
ここを有効にすると、最小のデバッグログが出力されます。

■ 情報 デバッグ出力
ここを有効にすると、情報のデバッグログが出力されます。

■ Ultra Verbose デバッグ出力

ここを有効にすると、詳細なデバッグログが出力されます。

■ 循環ログを使用

デバッグログに循環ログ機能が追加されました。

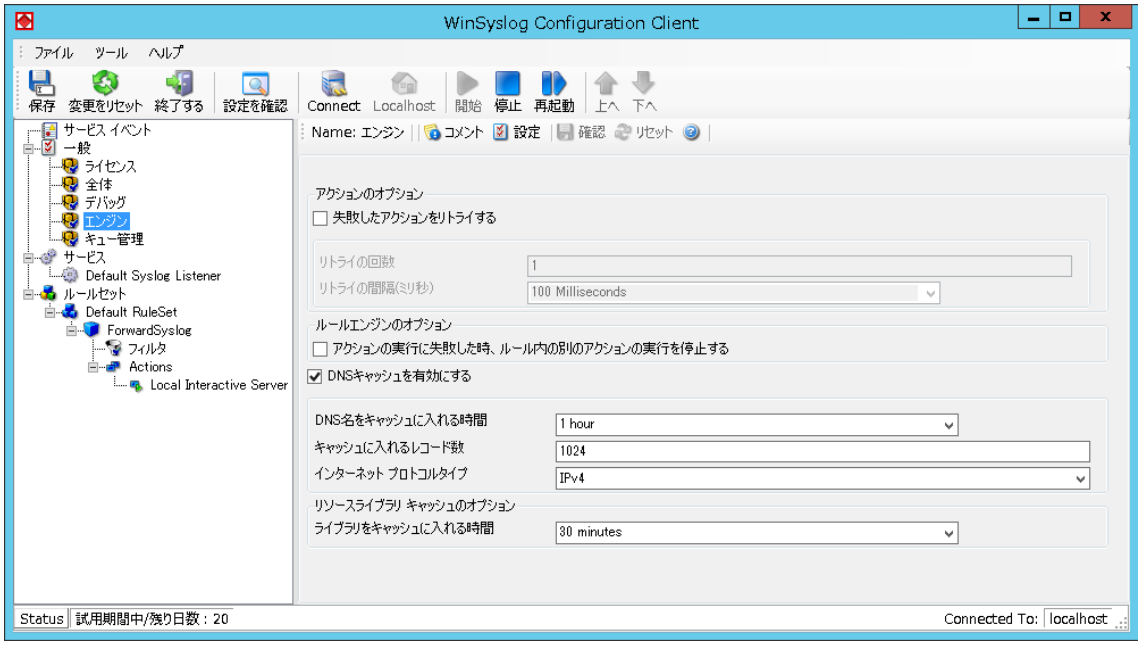
この機能を有効にすると、指定したログファイルの数・ログサイズでログを循環させることが可能です。

ログファイルの数: 10 (デフォルト値)

最大のファイルサイズ(KB): 51200 (デフォルト値)

5.2.4 「エンジン」オプション

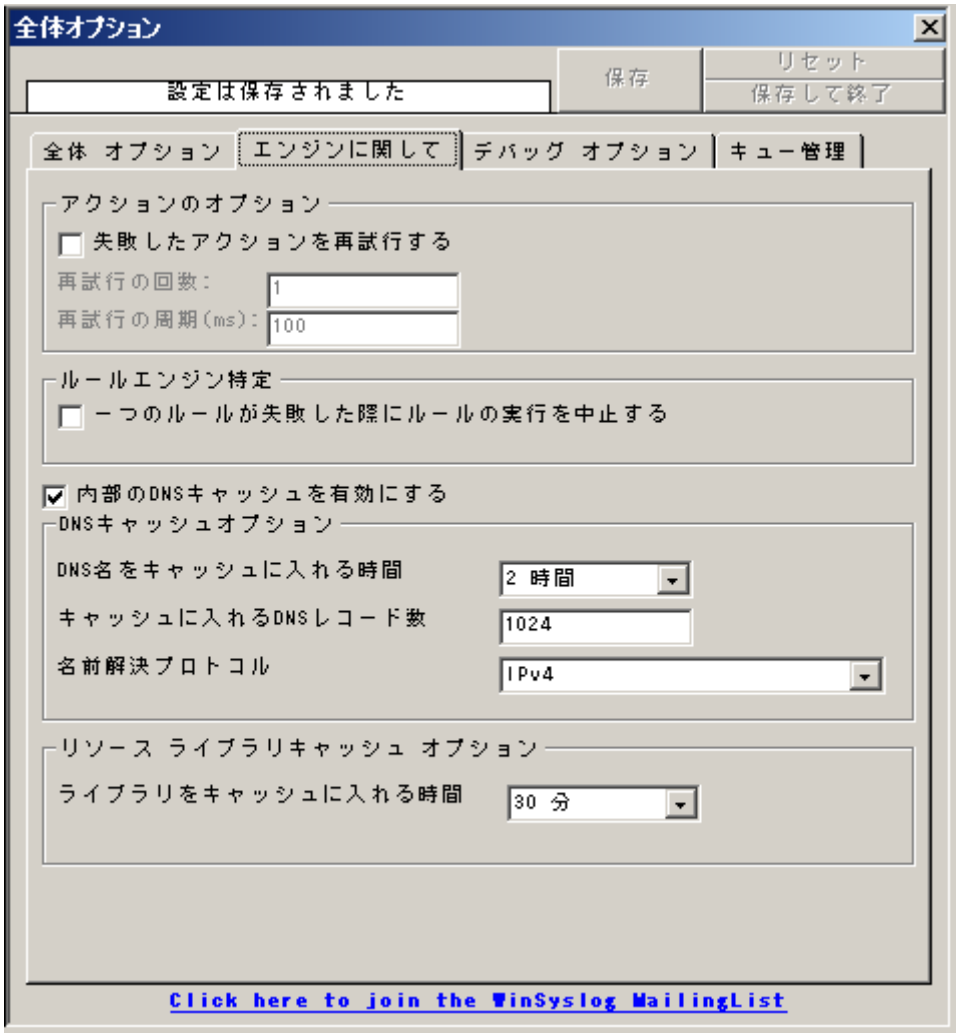
設定クライアント: 一般＞エンジン



▲新クライアント「エンジン」オプション

旧クライアント:マイコンピュータ＞全体/デフォルト＞エンジンに関して

＜「エンジンに関して」タブ＞



▲旧クライアント「エンジンに関して」オプション

＜アクションのオプション＞

■ 失敗したアクションをリトライする（失敗したアクションを再試行する）

この機能を有効にすると、サービスは「リトライ（再試行 ）の回数」の設定値に達するまで、失敗したアクションを実行します。

なお、エラーログは最後の失敗に対してのみイベントログ（ID114）に記録 されます。

再試行中のエラーは、EventReporter のデバッグログ（エラー＆警告）に記録されます。

＜ルールエンジンのオプション＞（ルールエンジン特定）

■ アクションの実行に失敗した時、ルール内の別のアクションの実行を停止する

（一つのルールが失敗するとき、ルールの実行を中止する）

ここを有効にすると、ルールに定義されているアクションのうちの一つが失敗した場合に、そのルールの実行を中止します。

ここが無効になっている場合には、ルール内に複数あるアクションのうち一つが失敗してもルールは停止せず、

それ以下に定義されているアクションが実行されます。

■ DNS キャッシュを有効にする（内部の DNS キャッシュを有効にする）

DNS キャッシュは、逆引き DNS の検索に使用します。
逆引き検索は、IP アドレスをコンピューター名に変換するために使用され、これは「ホスト名解決」アクションで実行されます。検索を実行する際、毎回 DNS は照会されるため、比較的システムへの負荷が大きくなってしまいます。それで、検索結果をキャッシュに入れます。検索が行われるたびに、システムは、まずローカルキャッシュに既に検索結果が入っていないかどうかをチェックします。検索結果がない場合、DNS クエリが実行され、その結果がキャッシュに入れられます。それにより、ホスト名の解決を実行する速度が格段に上がります。
しかし、コンピューター名や IP アドレスは変更される場合もあります。その場合、DNS は更新されます。もし、DNS を常にキャッシュに入れ、そこから検索するようにしていると、変更された情報を得ることができません。これを回避するために、DNS 名をキャッシュに入れる時間に制限を設けました。時間切れとなった DNS 名は、キャッシュ内にレコードが存在していないと認識され、新たに検索されるようになります。
また、キャッシュレコードは、システムメモリを使います。数多く名前解決をしたい場合には、より多くのメモリを割り当てる必要が出てくるでしょう。これを解決するために、キャッシュに入れる DNS レコード数を設定できるようにしました。この設定値に達すると、それ以降は新しくキャッシュレコードは割り当てられなくなります。

＜DNS キャッシュオプション＞

■ DNS 名をキャッシュに入れる時間

ここでは、DNS 名をキャッシュに入れる制限時間を設定します。
名前解決に問題が生じる可能性がありますので、ここでは高すぎる値を設定しないようにして下さい。
24 時間以上の値を設定することは、お奨めできません。

■ キャッシュに入れるレコード数

ここでは、キャッシュに入れるレコードの最大値を設定します。
名前解決をするレコード数が多くなると、システムが割り当てるメモリも大きくなります。ここで大きな値を設定しても名前解決を実行するホストの数が少ない場合には、キャッシュは大きくなりません。
しかし、数多くのホストの名前解決を行なう場合には、キャッシュに入れるレコード数に上限を設けることをお奨めします。ですが、その場合、頻繁に DNS の問い合わせが行なわれます。
1 つのキャッシュレコードにつき およそ 1～2KB として数値を設定して下さい。

■ インターネットプロトコルタイプ（名前解決プロトコル）

ここでは、名前解決の際に IPv4、IPv6 のうちどちらを優先するかを指定します。

＜リソース ライブラリキャッシュ オプション＞

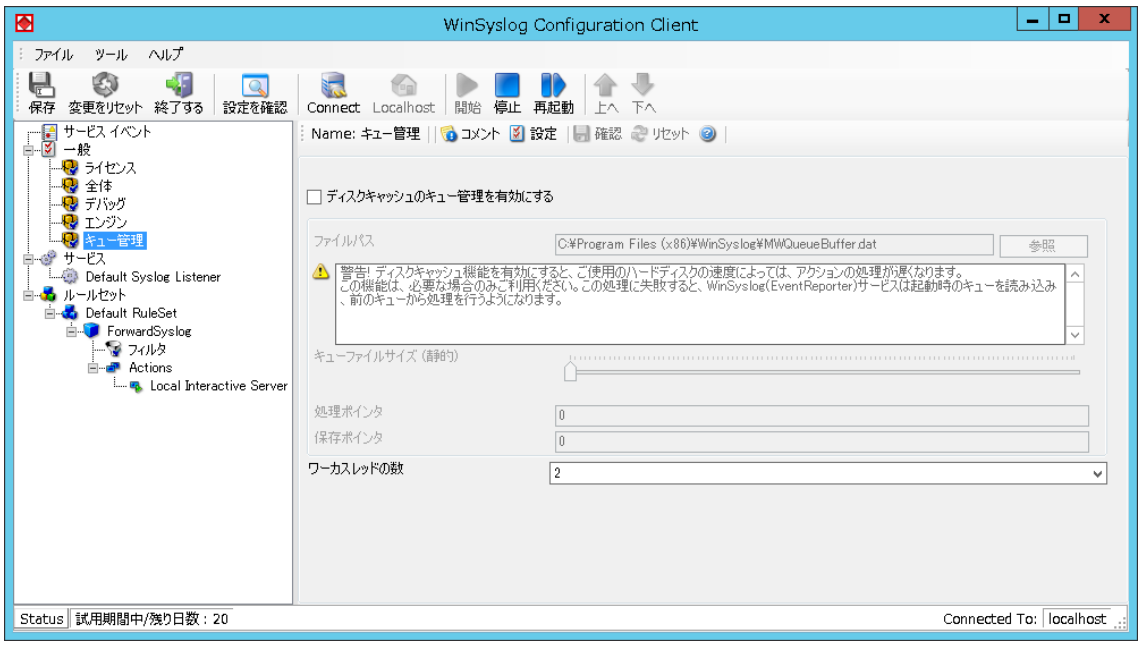
■ ライブラリをキャッシュに入れる時間

このオプションは、EventReporter のイベントログの監視機能において特に役立つ機能です。

同じイベントが何度も処理される状況では、このオプションを使用することで、パフォーマンスが上がることが期待できます。
デフォルトでは、全てのライブラリが 30 分間キャッシュに入れられます。

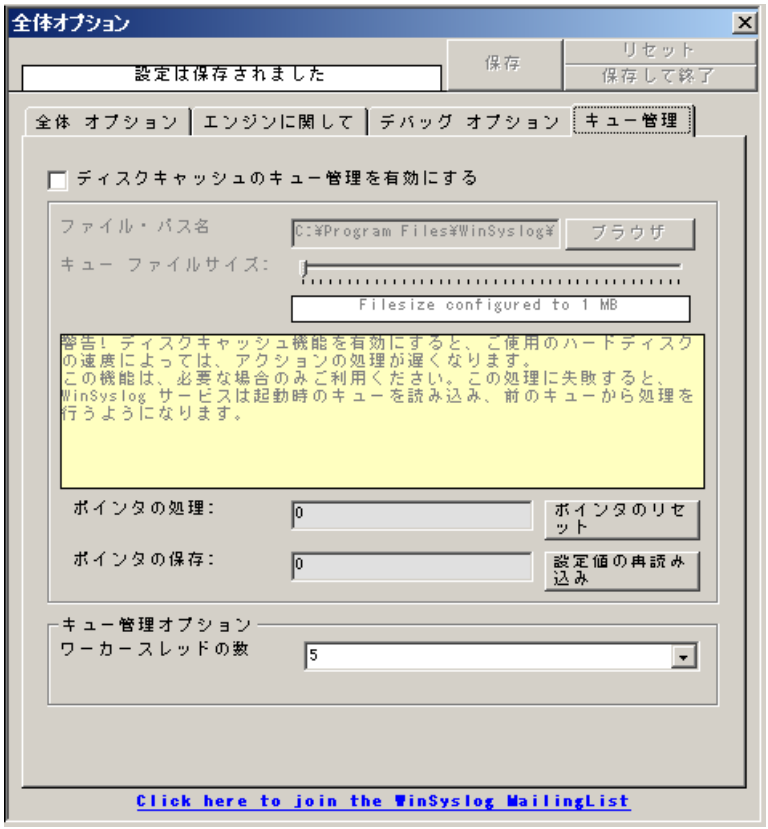
5.2.5 「キュー管理」オプション

設定クライアント：一般＞キュー管理



▲新クライアント「キュー管理」オプション

旧クライアント：マイコンピュータ＞全体/デフォルト＞キュー管理



▲旧クライアント「キュー管理」オプション

■ ディスクキャッシュのキュー管理を有効にする

この機能により、項目（Items）をディスク（指定したファイル）の内部キューに蓄えて置くことが可能となります。

この機能は、確実に必要な場合のみ使用するようにしてください。

ご利用のマシンのハードディスクの速度により、アクションの処理速度が下がる場合があります。最悪の場合には、マシンが IO 読み込みを実行できなくなり、キューがいっぱいになってしまいます。ディクスキャッシュは、受信した Syslog メッセージで未処理のものを確実に処理させたい場合の追加機能です。

ディスクキャッシュでは、イベントログの監視サービスのようにアクションが成功している間、継続して動作するタイプのサービスのインフォメーションユニットはキャッシュに入れません。Syslog サーバーサービスのような、その他全てのデータがキャッシュに入れられます。もしも、サービスやサーバーがダウンした時、次回のサービス起動時にキューが自動的に読み込まれるようになります。従って、キューにあるメッセージは消失されません。ただし、キャッシュに入れる処理中にダウンした場合には、そのデータは残りません。

■ ファイルパス（ファイル・パス名）

キューファイルの保存場所をここで指定できます。

■ キューファイルサイズ（静的）（キューファイルサイズ）

キューのサイズをここで指定します。

システムメモリを超えたサイズのキューファイルサイズは設定しないでください。システムメモリの合計よりも小さいサイズ（512MB）でご利用になることをおすすめします。最大値は 2048MB に設定されています。
キューファイルのサイズを変更すると、次回のサービス起動時に新たにキューファイルが作成されます。
それまで作成されていたキューファイルの内容は新しいファイルに移行されませんので、サイズ変更後はそれまでのキューがきちんと処理されたかどうかを確認してください。
また、一般的にサービスを停止した際にも同様にそれまでのキューが処理されます。

＜キューのリミット（全体オプション）＞

ディスクキャッシュのキュー管理を有効にした場合、全体オプションの「キューのリミット」が重要になります。キューファイルのサイズがどんなに大きくても、ここでの値がキューに保存されるアイテムの最大値になります。

＜Race conditions　競合状態について＞

サーバーやコンピューターがクラッシュした際、たいていの場合はファイルシステム、またはキューファイルが壊れてしまいます。その場合は、サービスの起動時に壊れたファイルを探し出し、可能であればキャッシュされたキューを初期化し、キューファイルをリセットします。

■ 処理ポインタ（ポインタの処理）

処理ポインタをここで指定します。（ディスクキャッシュのどの位置にポインタが来るかを指定します）

■ 保存ポインタ（ポインタの保存）

このポインタは、前回どこの位置で項目が保存されたかを表します。

＜キュー管理 オプション＞

■ ワーカースレッドの数

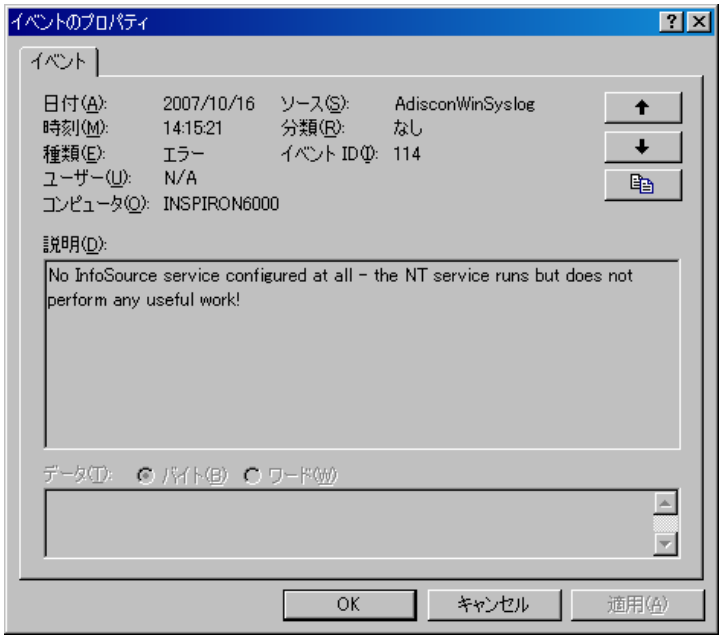
WinSyslog がキューの処理に使用しているワーカースレッドの数をここで指定します。

5.3 サービス

5.3.1 サービスについて

WinSyslog サービスは、Syslog メッセージの受信や SNMP トラップの受信などを行います。

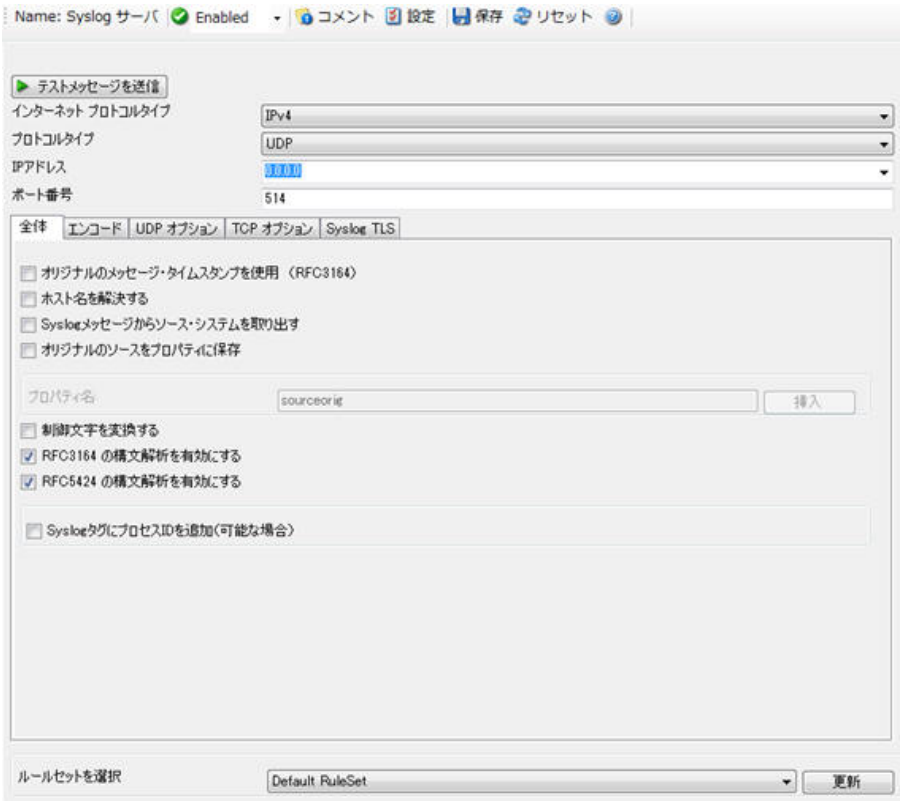
サービスの作成数に制限はありませんが、同じ設定内容（使用ポート・プロトコル）のサービスを複数稼動させることはできません。（その場合、正常に動作しません。）
異なった設定内容であれば、同じタイプのサービスを複数稼動させることが可能です。
ただし、サービスを一つも設定しなければ、WinSyslog は全く機能しません。その際には、イベントログに以下のようなエラーが記録されます。



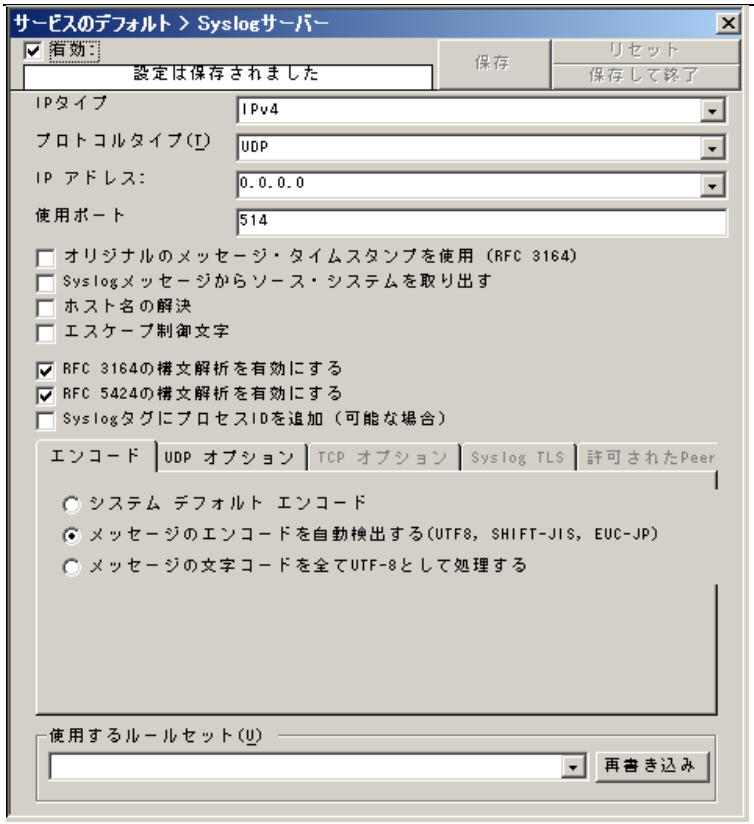
「サービス」と「サービスのデフォルト」が取り違われる場合もありますが、「サービスのデフォルト」は、予めサービスの設定を定義するもので、それ自体は何も実行しません。(旧設定クライアントのみ)

5.3.2 Syslog サーバー

ここでは Syslog サーバーサービスの設定を行います。設定することで、Syslog メッセージを収集できます。



▲新クライアント「Syslog サーバー」サービス



▲旧クライアント「Syslog サーバー」サービス

■ インターネット プロトコルタイプ(IP タイプ)

受信する Syslog が IPv4 なのか IPv6 なのかを指定します。

IPv4 と IPv6 のデバイスが混在する場合、Syslog サーバーサービスを IPv4 用、IPv6 用として分けて作成・設定しなければならないので、ご注意ください。

■ プロトコルタイプ

Syslog メッセージは、UDP または TCP、RFC3195 ベースの TCP で受信されます。

一つのサービスで一つのプロトコルを使用できます。

デフォルトは、UDP となっております。

Syslog サーバーは、UDP 以外にも TCP、そして RFC3195 RAW スタンダードを使用した TCP で Syslog メッセージを受信することが可能です。

＊「TCP」、「RFC3195(TCP)」は、プロフェッショナル、エンタープライズエディションの限定機能となります

■ IP アドレス

Syslog サーバーの IP アドレスを指定できます。(ここで指定したアドレスを宛先とした Syslog のみ処理されます)

この機能は、マルチホームの環境で別のサービスを別の IP アドレスで設定したい場合に役立ちます。

注)デフォルトとして設定されている「0.0.0.0」はすべて(ANY)の IP アドレスを意味します

■ **ポート番号（使用ポート）**

syslog サーバーが使用するポート番号を指定します。一般的な値は 514 です。
変更しなければいけない理由が明確である場合のみ、変更を行って下さい。
そのような必要性は、概してセキュリティに対する懸念から生じます。
＊ポートの変更を行った場合は、報告を行っているデバイス全ての設定をその標準でないポートを使用するように変更しなければなりません。

<全体>タブ

■ **オリジナルのメッセージ・タイムスタンプを使用　（RFC3164）**

ここをチェックすると、WinSyslog はメッセージ受信の時刻の代わりに、Syslog メッセージ内のタイムスタンプを使用します。これは、Syslog RFC3164 に対応しています。
チェックしない場合は、ローカルのシステム時刻を使用します。

メッセージ内のタイムスタンプを使用することには、いくらかの欠点もあります。
タイムゾーンの情報を持たないことなどが挙げられます。
もしも、複数のタイムゾーンのデバイスを処理している場合は、WinSyslog の時間の記録は、ぐちゃぐちゃになってしまいます。このような場合は、メッセージ受信時のタイムスタンプを使用することをお勧めします。

■ **ホスト名を解決する（ホスト名の解決）**

ここをチェックすると、メッセージソースの IP アドレスは（DNS によって）ホスト名の解決が実行されます。
チェックしない場合は、単に IP アドレスが使用されます。
“Syslog メッセージからソースシステムを取り出す” 設定が有効になっている場合には、この設定は機能しませんので、ご注意ください。この場合には、そのメッセージは常に syslog メッセージから取り出されます。

■ **Syslog メッセージからソースシステムを取り出す**

このボックス がチェックされる場合、ソースシステムの名前または IP アドレスは（RFC 3164 による） syslog メッセージから取り戻されます。
チェックしない場合は、それはメッセージを受信したアドレスに基づいて生成されます。

■ **オリジナルのソースをプロパティに保存（新クライアント）**

ここを有効にすると、オリジナル（元）のネットワークソースがプロパティ（デフォルトでは%sourceorig%）に保存されるようになります。

■ **制御文字を変換する（エスケープ制御文字）**

Syslog メッセージに制御文字が含まれている場合、ここをクリックするとその文字が 5 バイトのシーケンスのアスキー文字 ID に置換されるようになります。
例えば、ビープ音：BEL の場合には、アスキーの文字コードで 7 となっておりますので、この機能を有効にした場

合には〈007〉と表示されるようになります。

但し、日本語などダブルバイトの文字を使用している場合には、メッセージが壊れてしまう可能性が高いので、この機能は使用しないようにして下さい。

■ RFC3164 の構文解析を有効にする

このチェックボックスを有効にすると、RFC3164 に対応した Syslog メッセージの構文解析が可能となります。
無効にすると、旧メッセージ構文解析（Adiscon 仕様）が使用されます。
RFC3164 を基準にしていないメッセージを受信する際は、ここを無効にしてください。
（送信ホスト名やタイムスタンプが正常に処理されない場合には、ここを無効にしてみてください）

■ RFC5424 の構文解析を有効にする

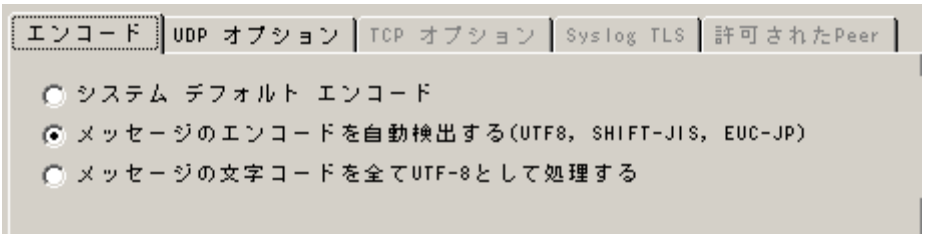
このチェックボックスを有効にすると、RFC5424 に対応した Syslog メッセージの構文解析が可能となり、RFC5424 のヘッダの検出とデコードができるようになります。また、新しい Syslog プロパティが含まれるようになります。
無効にすると、従来のメッセージ構文解析（Adiscon 仕様）が使用されます。
＊RFC5424 は、2009 年 3 月に発表された比較的新しい Syslog の基準です。
（WinSyslog と EventReporter は、RFC5424 に対応しております。）
RFC5424 を基準にしていないメッセージを受信する際は、ここを無効にしてください。
（送信ホスト名やタイムスタンプが正常に処理されない場合には、ここを無効にしてみてください）

■ Syslog タグにプロセス ID を追加（可能な場合）

このオプションは、[RFC5424 の構文解析を有効にする] を有効にした際に選択できます。
ここを有効にすると、Syslog タグに、プロセス名だけでなくプロセス ID（pid）も追加されるようになります。

＜エンコード＞タブ

▲新クライアント「エンコード」



▲旧クライアント「エンコード」

■ システム デフォルト エンコード（旧クライアントのみ）

受信する Syslog をシステムのデフォルトエンコードで処理する場合には、ここを有効にしてください。

■ メッセージのエンコードを自動検出する（UTF-8, SHIFT_JIS, EUC-JP）

9.1 バージョンより設けられた機能です。

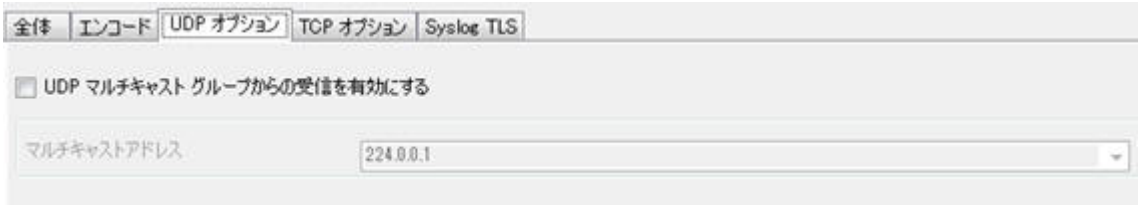
マルチバイト文字（日本語文字列など）を含む Syslog メッセージを処理する場合には、ここを有効にしてください。

■ メッセージの文字コードを全て UTF-8 として処理する

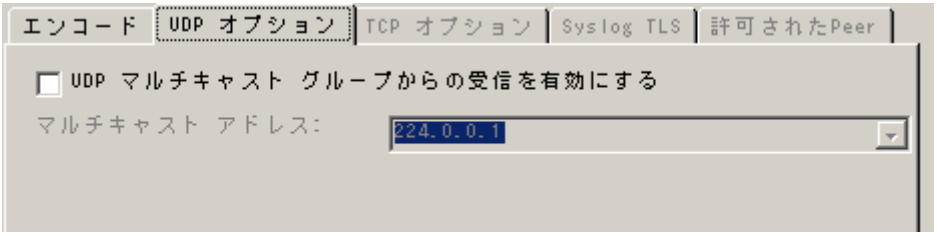
BOM を含まない UTF-8 の Syslog メッセージを受信する際は、ここを有効にしてください。

ただし、全てのメッセージが UTF-8 として扱われますので、他の文字コードで送信されてきた Syslog は正しく処理されない可能性がありますので、ご注意ください。

＜UDP オプション＞タブ



▲新クライアント「UDP オプション」



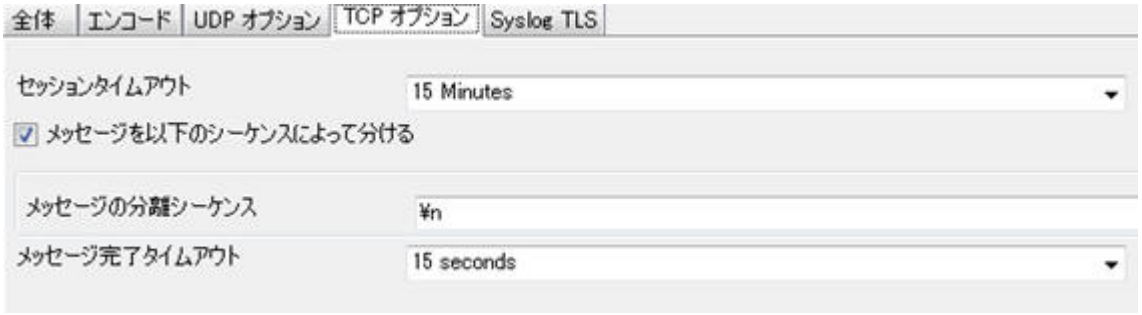
▲旧クライアント「UDP オプション」

■ UDP マルチキャスト グループからの受信を有効にする

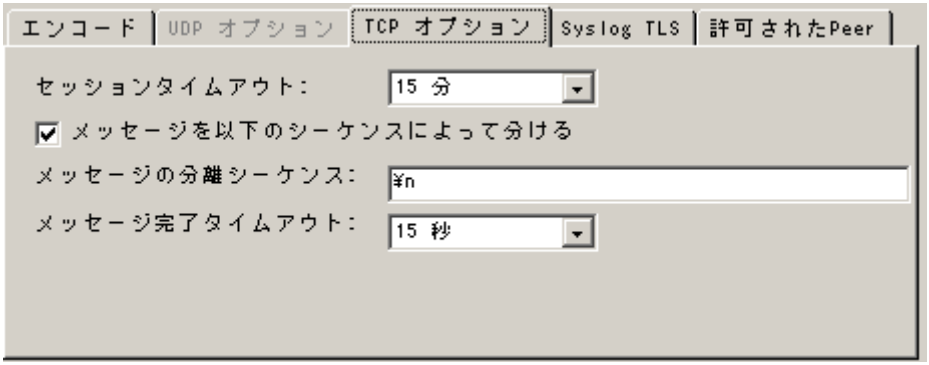
このオプションを有効にすることで、マルチキャスト IP アドレス（例；224.0.0.1）からの Syslog メッセージが受信で

きるようになります。

＜TCP オプション＞タブ



▲新クライアント「TCP オプション」



▲旧クライアント「TCP オプション」

■ セッションタイムアウト

ここでは、最後のパケットが送信された後、どれだけの時間 TCP のセッションをオープンしておくのかを設定します。プルダウンメニューから値（1 秒～1 日）を選択するか、「カスタム」を選択し、任意の値（ミリ秒で 2147483646 が最大値）を入力してください。カスタムで 0 を入力すると、セッションタイムアウトは無効になります。

■ メッセージを以下のシーケンスによって分ける

ここを有効にすると、受信メッセージは以下の設定で分けることができます；

メッセージ分離のシーケンス:

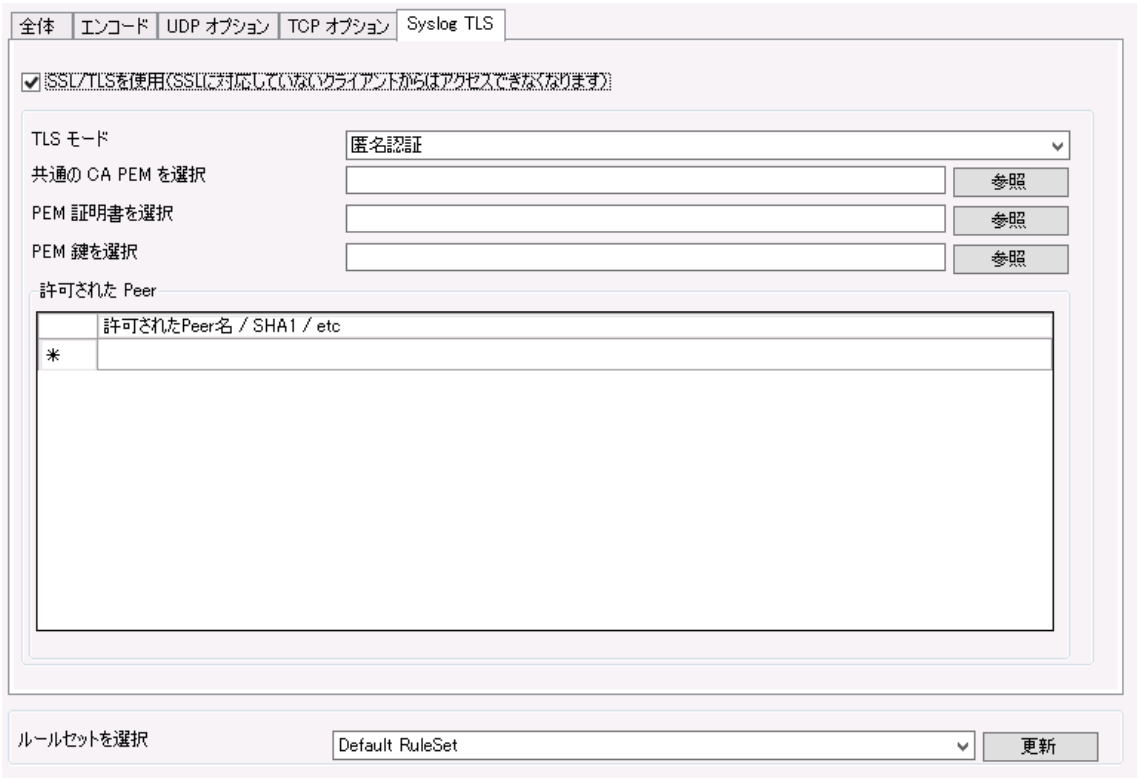
ここで入力したシーケンスでメッセージを分けます。デフォルトは「¥r¥n」（改行コード）に設定されております。

メッセージ完了タイムアウト:

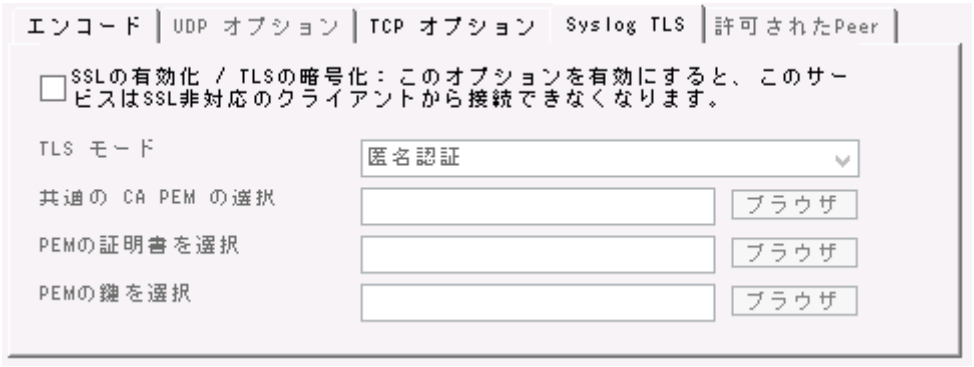
ここではメッセージ完了の時間を設定します。ここで設定した時間内に処理されなかったメッセージは、次の（新しい）メッセージとして分けて処理されます。

プルダウンメニューから値（1 秒～1 日）を選択するか、「カスタム」を選択し、任意の値（ミリ秒で 2147483646 が最大値）を入力してください。

＜Syslog TLS＞タブ



▲新クライアント「Syslog TLS」（「SSL/TLS を使用」オプションを有効にした場合）



▲旧クライアント「Syslog TLS」

■ SSL/TLS を使用（SSL の有効化／TLS の暗号化：）

ここを有効にすると、SSL に対応していないデバイス（クライアント）からのメッセージを受信できなくなります。（デフォルトは無効）

■ TLS モード

次のモードから選択できます；

・匿名認証

デフォルトでは、このモードになっています。

このモードでは、どんな証明書でも（証明書がない場合でも）受信できます。

・x509/name（証明書の確認と名前認証）

このモードを使用すると、「許可された Peer」でクライアントの証明書の subject がチェックされるようになります。

それによりセキュリティで保護された接続のみ許可されるようになります。

・x509/fingerprint（証明書とフィンガプリント）

このモードは、SHA1 フィンガプリント（指紋）を作成し、「許可された Peer」のフィンガプリントと比較します。

デバッグログを取得すれば、許可されなかったクライアント証明書のフィンガプリントを確認することができます。

・x509/certvalid（証明書の確認のみ）

クライアント証明書が有効でありさえすれば接続が許可されます。

■ 共通の CA PEM の選択

ここでは、CA（Certificate Authority；認証局）を指定します。

Syslog を送信する側・受信する側で同じ CA を使用しなければなりません。

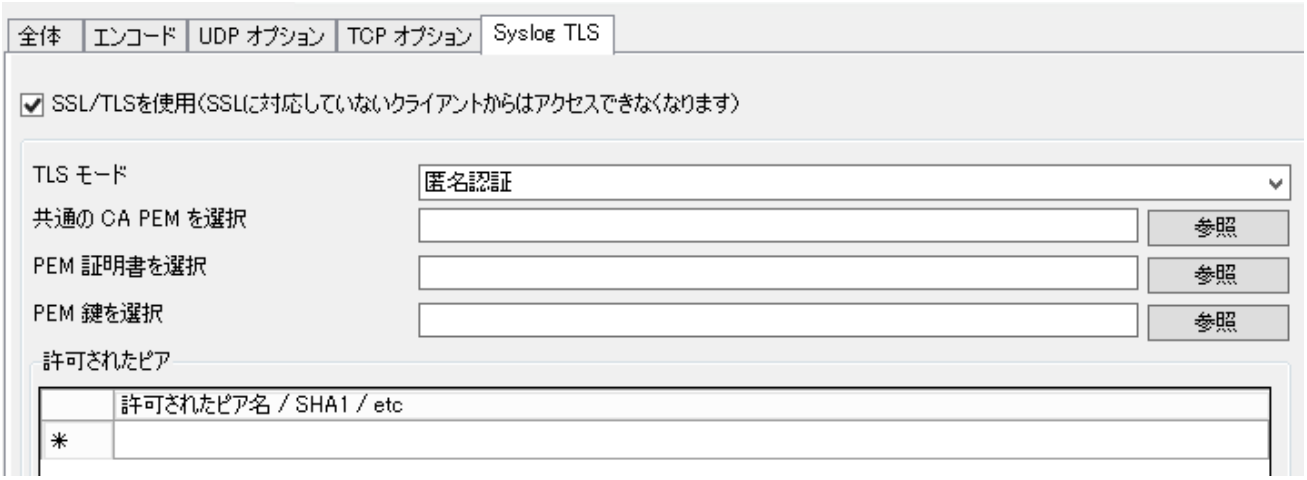
■ PEM の証明書を選択

クライアントの証明書（PEM フォーマット）を選択します。

■ PEM の鍵を選択

クライアント証明書の鍵（キーファイル）を選択します。

■ 許可されたピア



▲新クライアント「許可されたピア」

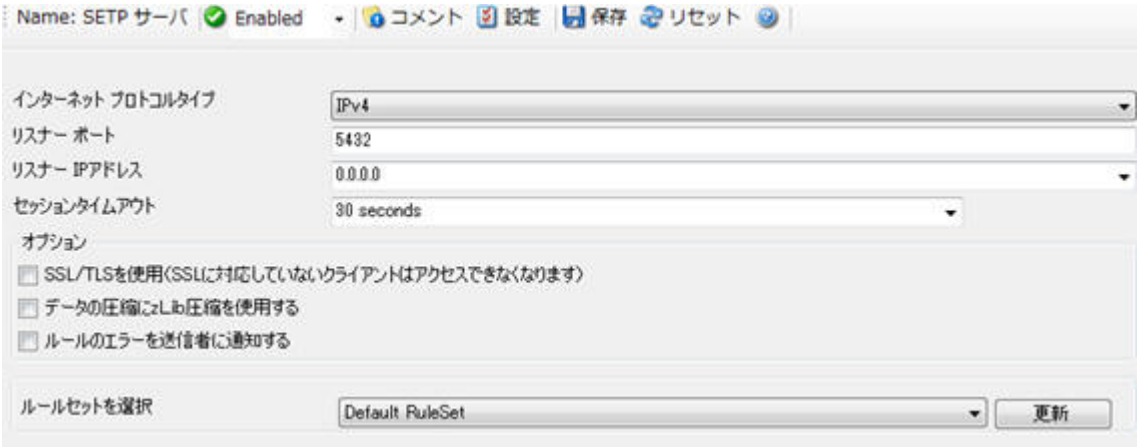
ここでは、許可されたピア名を入力します。
x509／name を使用する場合、証明書の subject を入力します。
例えば、証明書の subject が CN = secure.syslog.msg である場合、secure.syslog.msg を許可されたピア として入力します。
x509/fingerprint を使用する場合、ここでは許可された SHA1 指紋（fingerprint）を入力します。
指紋（fingerprint）は、OpenSSL ツールで生成することもできますが、デバッグログファイルから入手することもできます。
以下は、フォーマット（RFC5425 参照）のサンプルです；
SHA1:2C:CA:F9:19:B8:F5:6C:37:BF:30:59:64:D5:9A:8A:B2:79:9D:77:A0

■ ルールセットを選択（使用するルールセット）
syslog サーバーのサービスで使用されるルールセット名を選択します。
当然のことながら、そのルールセット名は、有効でなければなりません。

5.3.3 SETP サーバー

ここでは、SETP サーバーサービスを設定します。
※このサービスは、エンタープライズ エディションのみご利用可能となっております※

SETP サーバーは、他システムからイベントを MonitorWare（弊社では扱っておりません） 製品ライン内で確実に受信するのに用いられます。ここには、SETP が送り主からオリジナルのメッセージを受け取って、送り側で設定した設定を正確に使用するための設定が設けられています。
変更は、SETP サーバー側で発生しません; メッセージ・フォーマットに対して設定すべき値はありません。



▲新クライアント「SETP サーバー」サービス

▲旧クライアント「SETP サーバー」サービス

■ インターネットプロトコルタイプ(IP タイプ)

受信する Syslog が IPv4 なのか IPv6 なのかを指定します。

IPv4 と IPv6 のデバイスが混在する場合、Syslog サーバーサービスを IPv4 用、IPv6 用として分けて作成･設定しなければならないので、ご注意ください。

■ リスナーポート(使用ポート)

SETP サーバーが使用するポートを指定します。デフォルトは、5432 です。

変更しなければいけない理由が明確である場合のみ、変更を行って下さい。

そのような必要性は、概してセキュリティに対する懸念から生じます。

SETP サーバーとの通信には、TCP が使用されます。

ポートの変更を行った場合は、送信側の設定を送信側と合わせるように変更しなければなりません。

■ リスナーIP アドレス

SETP サーバーサービスでも、Syslog サーバーサービスと同様に特定の IP アドレスで処理させることが可能となりました。

この機能は、マルチホームの環境で別のサービスを別の IP アドレスで設定したい場合に役立ちます。

注)デフォルトとして設定されている「0.0.0.0」はすべて(ANY)の IP アドレスを意味します

■ セッションタイムアウト

サーバー側のセッションがオープン状態である場合の最大の待ち時間を指定します。

■ SSL/TSL を使用（SSL の有効化／TLS の暗号化）

ここを有効にした場合、SSL、または TLS、STEP サーバーに接続することができます。
ただし、SSL に対応していない SETP サーバーには接続できなくなります。

■ データの圧縮に zLib 圧縮を使用する

ここを有効にすると、WinSyslog は SETP 送信により送信されたzLib 圧縮データを解凍します。今まで通り、通常のデータも受信することが可能です、zLib 圧縮は、WAN 環境において通信量を減少させることに役立ちます。

■ ルールのエラーを送信者に通知する

ここを有効にすると、アクションの結果を SETP メッセージの送信者へ通知されるようになります。

例えば、イベントログの監視を実行していて、これらのイベントを SETP で送信、一方でデータベースへ収集した全てのイベントを書き込むよう設定しているとします。

もしも、データベースがオフラインの場合、イベントの書き込みは実行できないので、SETP サーバーは、アクションの実行に失敗したという内容のメッセージを最後のメッセージとして送信し、イベントログに ID：1005 のエラーを作成します。（その後、このアクションが成功した場合には、ID：1012 のイベントログが記録されます。）送信者は、それから停止して、再度イベントの送信を試みます。

これは、SETP が TCP と同じようにデータ転送を確実にするためですが、さらに、アクションが成功した場合にも送信者にステータスを返信することもできます。これは、イベントログの監視サービスが再試行可能（restartable）なイベントソースだからです。同じソースでアクションが再試行するかどうかを決定するために、アクションの結果が使用されます。他のイベントソースは、違う動作をします。例えば、Syslog サーバーサービスは、失敗したアクションを再試行しません。これは、Syslog メッセージが消失する可能性があるという性質によるものです。

注意：この機能をご利用になる場合、それ以前のバージョン（7.2.x 以前）の WinSyslog へは、このデータが正常に送信できない可能性があります。従って、この機能をご利用の際は、全ての WinSyslog を最新版にバージョンアップするようにして下さい。

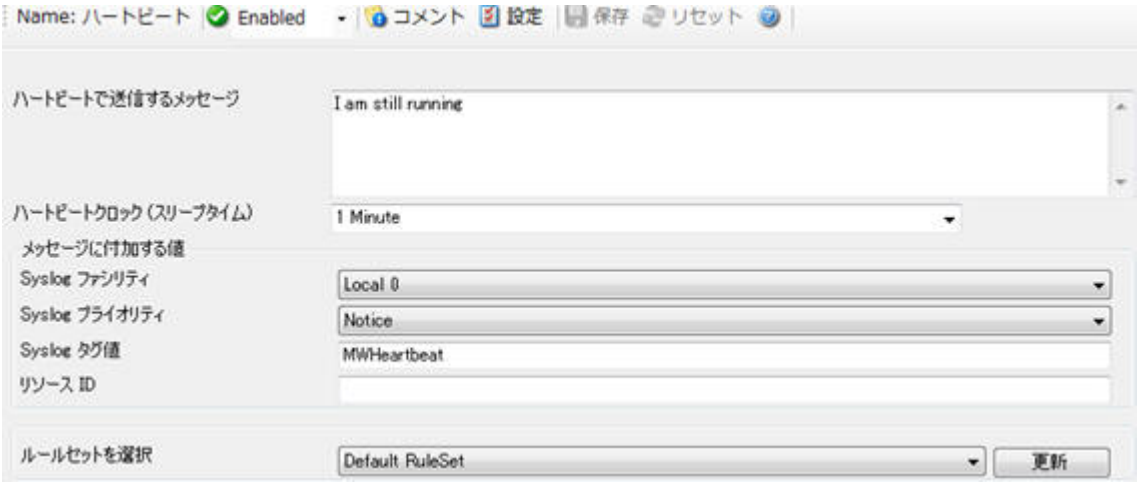
■ ルールセットを選択（使用するルールセット）

SETP サーバーのサービスで使用されるルールセット名を選択します。
当然のことながら、そのルールセット名は、有効でなければなりません。

5.3.4　ハートビート

ハートビートサービスを使用することで、WinSyslog サービスが稼動しているかどうかを確認することができます。
ハートビートクロックで設定した時間ごとにハートビートメッセージが作成されます。
ハートビートメッセージは、Syslog メッセージと同様にメール送信や Syslog 転送など、お好みのアクションを組み合わせ、通知させることが可能です。
（このサービスの「使用するルールセット」で選択したルールセット内のアクションで処理されます）

＊ハートビートメッセージが指定の時間に届かない場合には、WinSyslog で何かトラブルが起きているか、動作が停止しているということが疑われます。



▲新クライアント「ハートビート」サービス

▲旧クライアント「ハートビート」サービス

■ ハートビートで送信するメッセージ（ハートビート中に受信するメッセージ）

ここで設定したメッセージがハートビートからのログに記録されます。

入力する内容は、どんなものでも構いません。

■ ハートビート クロック(スリープタイム)

ハートビートメッセージを生成する間隔を指定します。(**ミリ秒で設定**します)

ハートビートメッセージを受信するマシンのスペック等を考慮し、値を設定してください。

重い負荷がかかっている時は、メッセージの生成間隔が設定より遅くなる場合もあります。

＜メッセージに付加する値＞

■ Syslog ファシリティ

ハートビートメッセージに割り当てられる syslog ファシリティ。

メッセージを syslog サーバーに転送する際に役立ちます。

■ Syslog プライオリティ

ハートビートメッセージに割り当てられる syslog プライオリティ(Severity)。

メッセージを syslog サーバーに転送する際に役立ちます。

■ Syslog タグ値

ハートビートメッセージに割り当てられる syslog タグ値。
メッセージを syslog サーバーに転送する際に役立ちます。

■ リソース ID

ハートビートメッセージに割り当てられるリソース ID。
メッセージを syslog サーバーに転送する際に役立ちます。

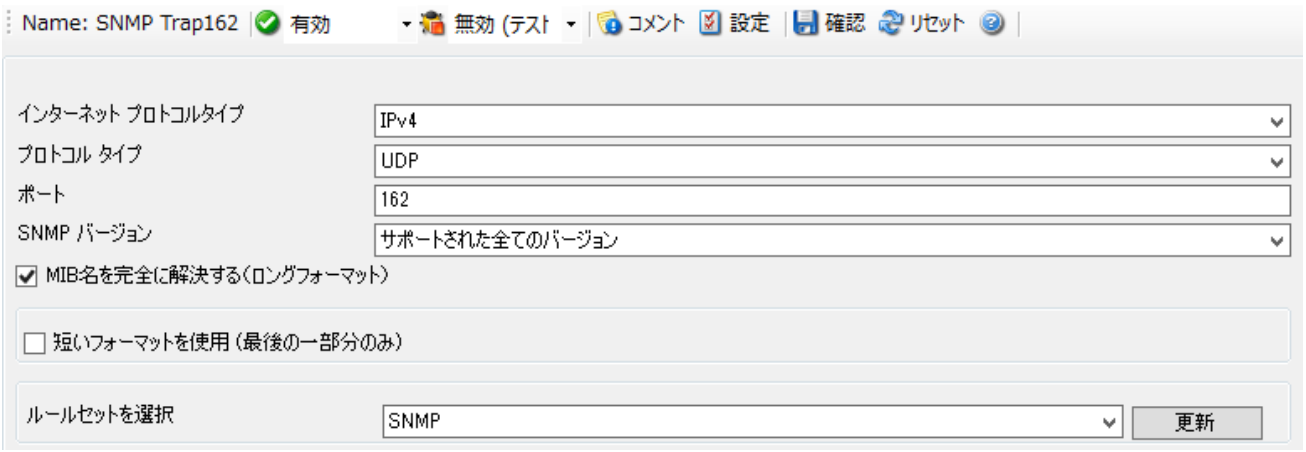
■ ルールセットを選択（使用するルールセット）

このサービスのために使用されるルールセット名。ルールセット名は、有効でなければなりません。

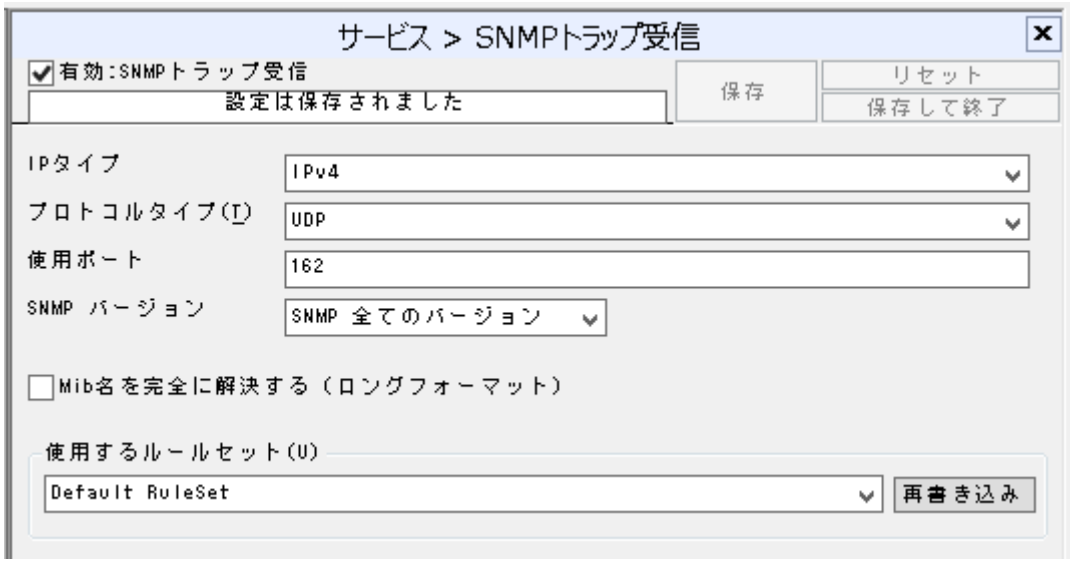
5.3.5 SNMP トラップ受信

SNMP トラップ受信によって、SNMP メッセージを受信することができます。

トラップは、別のプロトコル（SNMP）において Syslog メッセージのような役割を果たすものです。
デバイスが送信すべき情報がある場合などにトラップが生成されます。その情報には、バージョンやコミュニティなどの標準的な項目も含まれます。



▲新クライアント「SNMP トラップ受信」サービス



▲旧クライアント「SNMP トラップ受信」サービス

■ インターネット プロトコルタイプ （IP タイプ）

受信する SNMP トラップが IPv4 なのか IPv6 なのかを指定します。

IPv4 と IPv6 のデバイスが混在する場合、SNMP トラップ受信サービスを IPv4 用、IPv6 用として分けて作成・設定しなければならないので、ご注意ください。

■ プロトコルタイプ

受信する SNMP トラップのプロトコルを指定します。UDP、TCP のどちらかを選択してください。

■ ポート （使用ポート）

SNMP リスナーが使用するポートを指定します。

確信がない場合には、デフォルトのポートである 162 のままにして置いて下さい。

■ SNMP バージョン

ここでは、SNMP バージョンを限定します。設定可能な値は、下記のとおりです:

1.サポートされた全てのバージョン （SNMP 全てのバージョン）

2. SNMP バージョン 1 のみ

3. SNMP バージョン 2c のみ

■ MIB 名を完全に解決する（ロングフォーマット）

このオプションを有効にすると、MIB ブラウザにあるように MIB 名が解決されるようになります。

■ 短いフォーマットを使用（最後の一部分のみ）

完全な名前解決が長く読みづらい場合には、この機能を有効にしてください。

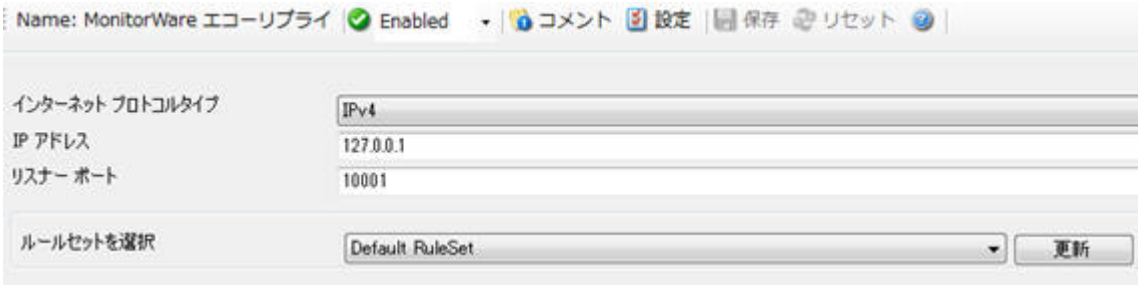
MIB 名の最後の部分だけになるようフォーマットが変更されます。
こちらは新クライアントの機能になります。

■ ルールセットを選択（使用するルールセット）
このサービスのために使用されるルールセット名。ルールセット名は、有効でなければなりません。

5.3.6 MonitorWare Echo Reply

この機能は、MonitorWare エージェントに関連するものです。
現在、ジュピターテクノロジー（株）では、MonitorWare の販売は行っておりません。

MonitorWare Echo Reply サービスは、多少変わったサービスです。
これは、単独では何のイベントも生成しません。このサービスは、MonitorWare Echo Request サービスに受動的に対応する物です。
これらは、同時に失敗しているエージェントを見つけるのに使用されます。設定項目は、使用ポートのみで Echo Request サービスが接続するポートと同じポートである必要があります。



▲新クライアント「MonitorWare エコーリプライ」サービス

▲旧クライアント「MonitorWare エコーリプライ」サービス

5.3.7 RELP リスナー（WinSyslog 10.1 で追加された機能）

RELP リスナーサービスは、新しいプロトコルである「Reliable Event Logging Protocol」に対応しています。

このプロトコルを使用することで、従来の TCP Syslog プロトコルより確実な転送が可能となります。
（このサービスでは、RELP に対応した送信元からのメッセージを受信します）

RELP プロトコルを使用すること以外は、機能的には Syslog サーバーサービスと同じような働きをします。

▲新クライアント「RELP リスナー」サービス

▲旧クライアント「RELP リスナー」サービス

■ リスナーポート（使用ポート）

RELP リスナー サービスで使用するポート番号をここで指定します。
デフォルトは、20514 です。
この値は変更できますが、その際は、メッセージを送信している機器のポートも合わせて変更するようにしてください。

■ セッションタイムアウト

ここでは、サーバー側のセッションがオープン状態である場合の最大の待ち時間を指定します。

■ ルールセットを選択（使用するルールセット）

このサービスのために使用されるルールセット名。ルールセット名は、有効でなければなりません。

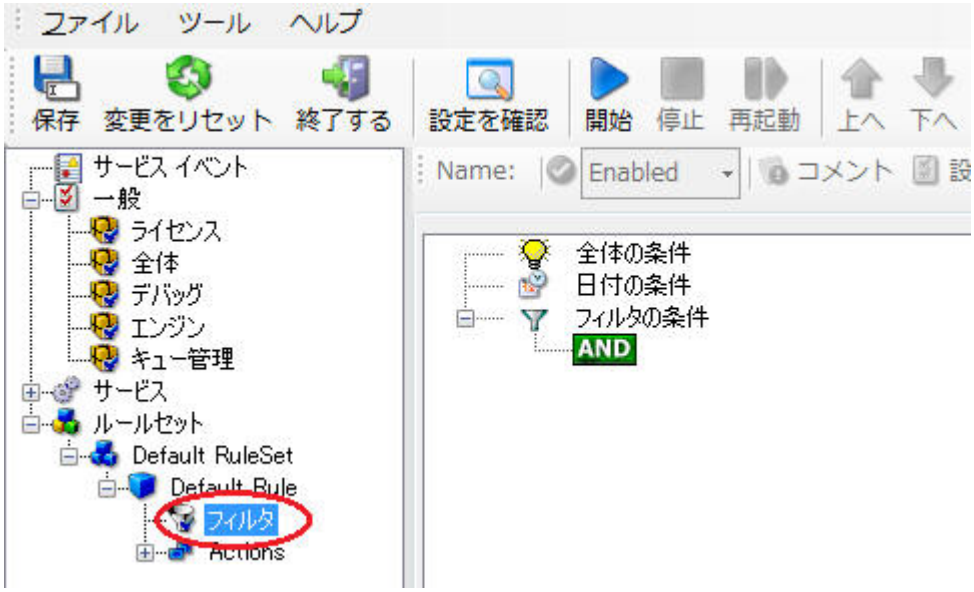
※RELP とは

Reliable Event Logging Protocol の略であり、転送中のメッセージがロストしないよう設計されたプロトコルです。
現行バージョンの RELP プロトコルは、接続が切れた場合でも、メッセージを複製できる可能性があります。

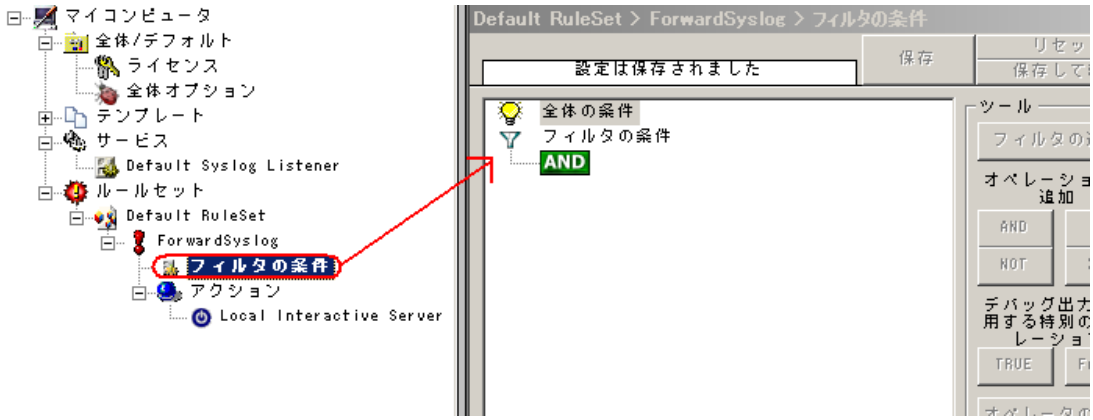
5.4 フィルタ

5.4.1 フィルタの条件

フィルタ（フィルタの条件）は、受信したログの絞込みをしたい（特定の条件に合ったログだけを処理、または破棄したい）場合に設定します。
フィルタ（フィルタの条件）は、ルールの中、アクションの上にあります。
（下図では、Default RuleSet（ルールセット）の下、さらに Default Rule （ForwardSyslog）の下、アクションの上にあります）



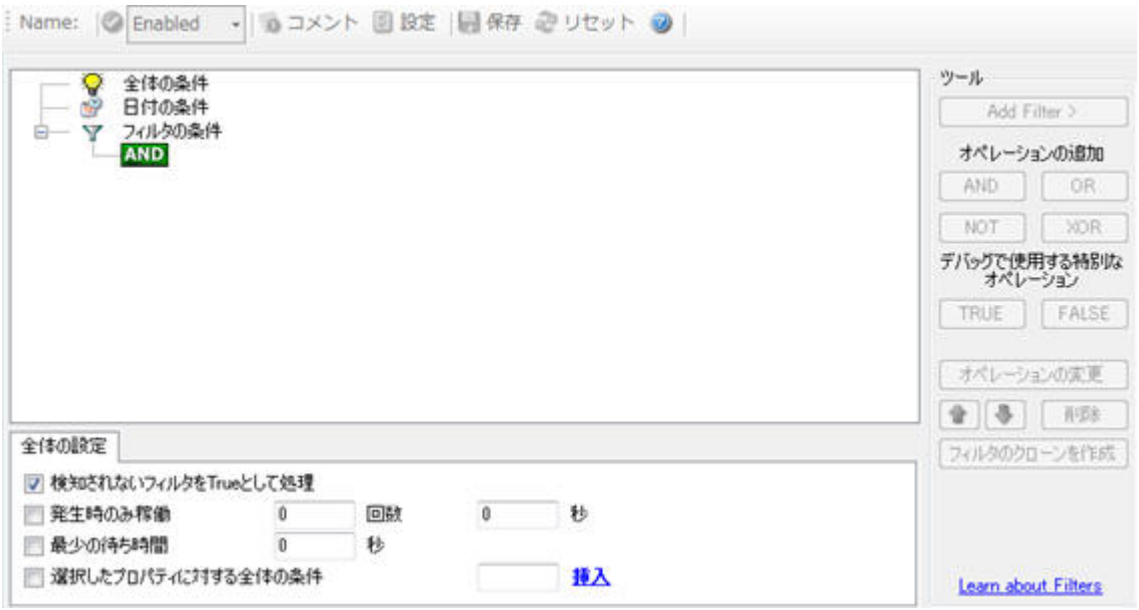
▲新クライアント「フィルタ（フィルタの条件）」



▲旧クライアント「フィルタの条件」

WinSyslog で受信したログは、 フィルタ（フィルタの条件）で設定した内容（フィルタ）に一致した場合、その下で設定されたアクションにより処理されます。

フィルタ（フィルタの条件）は、必要に応じて複雑にすることが可能です。
ブール演算と条件のネスティングがサポートされています。



▲新クライアント「フィルタ（フィルタの条件）」

▲旧クライアント「フィルタの条件」

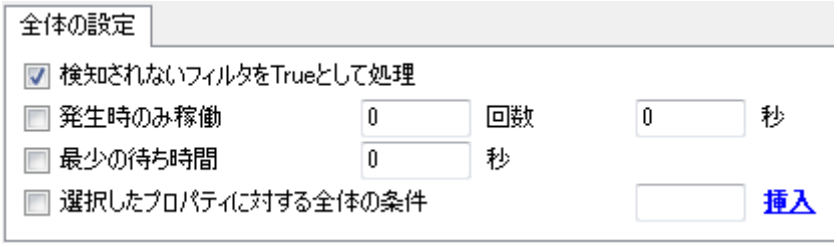
デフォルトでは、 上図のように「AND」のみが設けられています。
この状態では、全てのデータが「真（True）」となり、それらに対してアクションが実行されます。
（すなわち、フィルタリングされず、全てのメッセージが処理されます）

メッセージの絞込みを行わない（例えば、データベースやテキストファイルに全てのメッセージを書き込みたい場合）場合には、このデフォルトの設定のままご利用下さい。

一方、特定の条件において（メッセージを絞り込んで）アクションを実行させたい場合には、その条件によって、様々な設定を行うことができます。

5.4.2 全体の条件

全体の条件は、全体としてルールに適用します。
それは、フィルタのツリーの中で条件と共に、理論的な「AND」と組み合わせられます。



▲新クライアント「全体の条件」

▲旧クライアント「全体の条件」

■ 検出されないフィルタを TRUE として処理する

フィルタの条件で問い合わせたプロパティがイベントに存在しない場合、通常は「偽（False）」として処理されます。
上記のイベントを「真（True）」として処理したい場合、ここを有効にすることで可能となります。

■ 発生時のみ稼働（イベントが発生した場合のみ稼動）

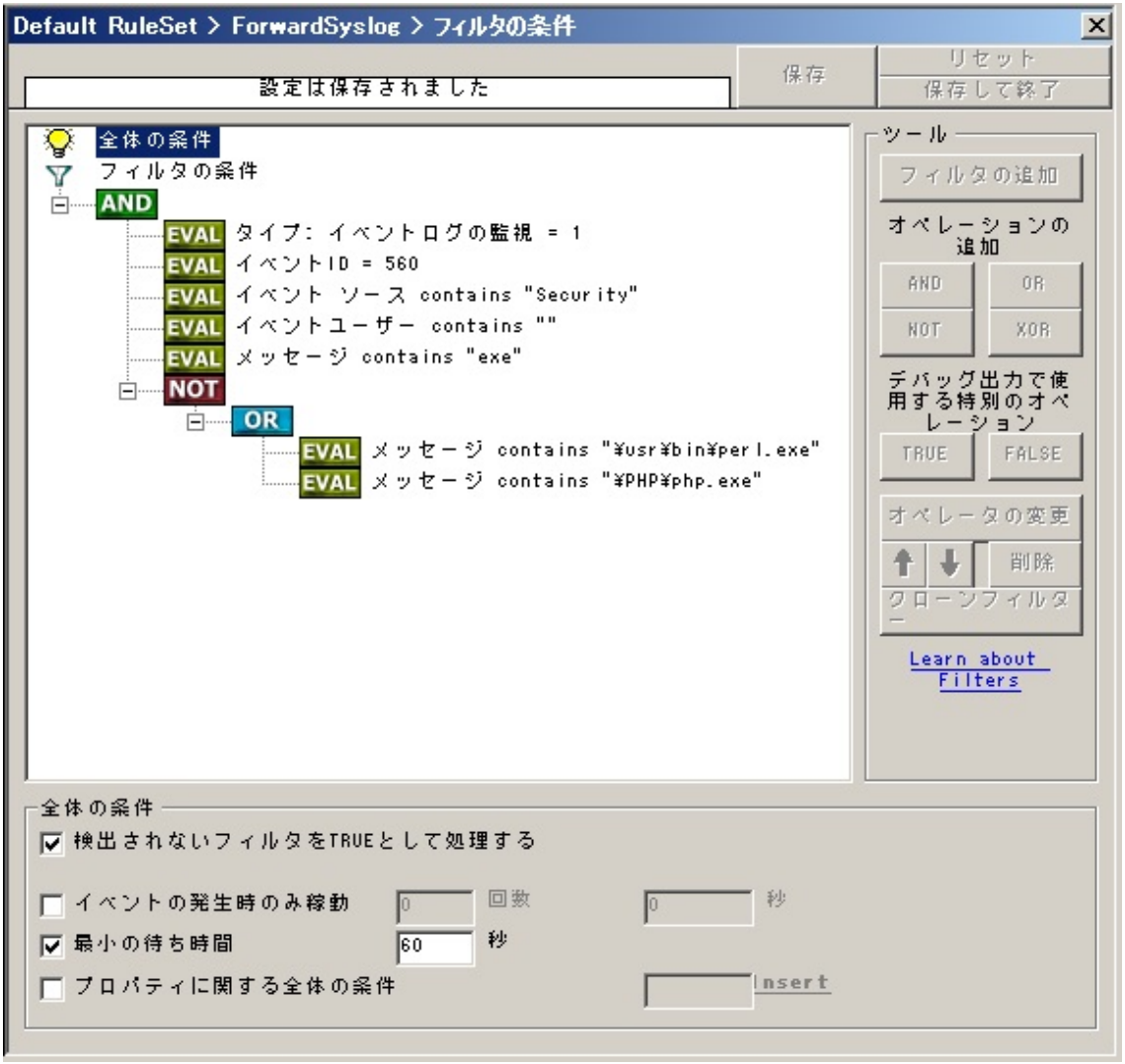
これは、ある意味「最小の待ち時間」と正反対の機能と言えます。
ここでは、設定した回数だけイベントが発生しないとルールが起動しません。

この機能は、指定した時間内に、指定されただけの回数のイベントが発生しなければルールが稼動しません。これは、以前は「発生」と呼ばれていた機能を改名したものです。

■ 最小の待ち時間

このフィルタの条件は、ルールが過度に作動するのを防ぐために使用ができます。
（ここで設定した時間間隔で、ルールが稼動するようになります。）
＜例＞ ここで 60 と入力すれば、60 秒毎にルール（アクション）が実行されます。

下図は、侵入検知を行うための 「フィルタ条件」設定例です。



ここでは、**IIS で許可されていない exe ファイルが実行された場合に生成されるイベントログを受け取った際にアクションを実行させるよう**設定されております。

具体的には、 下記の条件を満たすイベントが「真（True）」と評価され、アクションが実行されます。

・イベントログの監視サービスで処理したもので、イベント ID：「560」、ソース：「Security」、ユーザー：「P15111116¥IUSR_ROOTSERVER」、文字列：「exe」 を含む

・ 「¥usr¥bin¥perl.exe」または「¥PHP¥php.exe」の文字列が含まれていない

頻繁にアラートが発生しないように、さらに「最小の待ち時間」を 60 秒と設定しました。
したがって、全ての条件一致した場合でも、フィルタの条件は 60 秒の間隔を置いて「真（True）」と評価し、アクションを実行します。

■ 選択したプロパティに関する全体の条件（プロパティに関する全体の条件）

この機能により、プロパティをベースにして全体の条件を管理することが可能となりました。

例えば、メッセージのソースをプロパティとして処理する場合、それぞれ個々のメッセージソースに対して最小の待ち時間が適用されるようになります。

5.4.3 日付の条件（新クライアントのみ）

▲新クライアント「日付の条件」

■ 常にルールを処理

ここが有効になっている時には、条件はなく、常にルールは実行されます。

■ インストール直後のみ処理

ここが有効の場合、インストール直後、処理対象のメッセージがあった際にルールが実行されます。

■ 設定した日にのみ処理

ここが有効の場合、設定した日付で、処理対象のメッセージがあった際にルールが実行されます。

5.4.4 オペレーション

オペレーションでは、フィルタの条件がどのように互いにリンクしているかを設定します。
以下のオペレーションを使用することが可能です。

AND	全てのフィルタが一致した場合のみ、結果が「真（True）」になります。

	＜例＞ フィルタ a、 b、 c を設定した場合 a かつ b かつ c のいずれも満たす場合に真（True）となり、アクションが実行される
OR	ひとつでも一致するフィルタがあるならば、結果が「真（True）」となります。 ＜例＞ フィルタ a、 b、 c を設定した場合 a または b または c のいずれか１つでも満たす場合に真（True）となり、アクションが実行される
NOT	NOT 演算には、ひとつのフィルタしか作成できません。 フィルタが一致した場合、「NOT」は「偽（False）」となります。 ＜例＞ フィルタ a を設定した場合 a を満たす場合、アクションが実行されない
XOR	XOR 演算は、２つのフィルタのうち一方が一致した場合のみ、「真（True）」となります。 ＜例＞ フィルタ a、 b を設定した場合 a と b のどちらか一方のみ満たす場合に真（True）となり、アクションが実行される a と b の両方を満たす、両方を満たさない場合は、偽（False）となる。
TRUE	デバッグを行う際に役立ちます。結果は「真」となります。
FAULSE	デバッグを行う際に役立ちます。結果は「偽」となります。

5.4.5 フィルタ

フィルタは、各オペレーションノードの下に追加することが可能です。
全てのサービスに使用できる共通のフィルタは、数種類あります。

また、特別な種類のインフォメーション ユニットに対してのみ適用されるフィルタも存在します。
インフォメーション ユニットでマッチしない全てのフィルタは、フィルタリング処理で無視されます。
このような場合には、サービスごとに別のルールセットを作成し、関連付けするようにして下さい。

フィルタには、色々なタイプのものが存在します。
従って、それらのタイプと値を比較する方法もそれぞれ存在します。
以下のタイプが使用可能です。

文字列（String）

「＝」、「Not ＝」、「範囲内の一致」で別の文字列と比較されます。

番号（Number）

「＝」、「Not ＝」、「＜」、「＞」で別の番号と比較されます。

ブール演算子（Boolean）

「＝」、「Not ＝」で「真」か「偽」のいずれかと比較されます。

時間（Time）

「＝」のみで別の時間と比較されます。

5.4.6 一般

▲新クライアント

▲旧クライアント

■ ソース

このフィルタの条件は、インフォメーション ユニットを作成するシステムをチェックします。
例えば、Syslog サーバーの場合、これは syslog メッセージを送っている syslog デバイスになります。

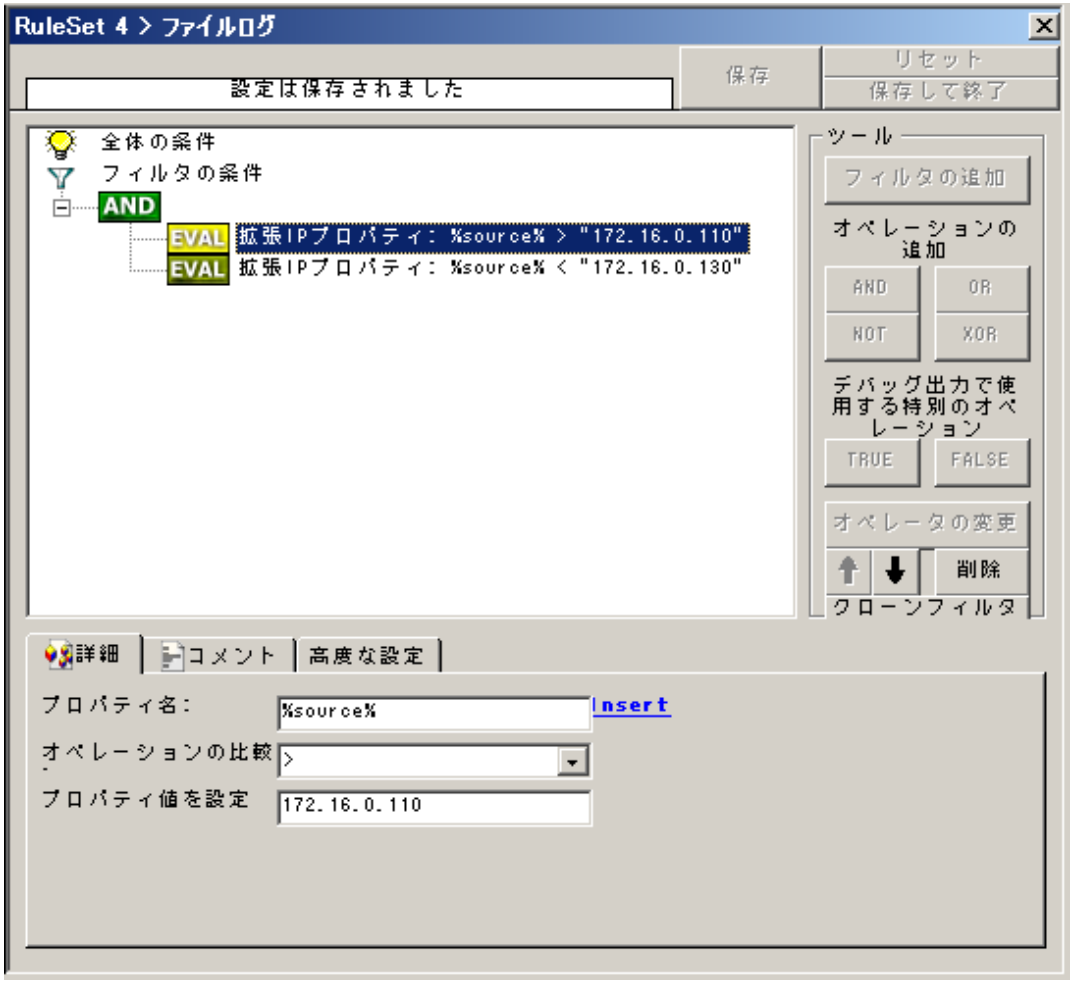
このフィルタは、文字列で指定します。
ソースシステム名か IP アドレスを含むように指定します。

■ ソース(IP タイプ)

このフィルタは、IP アドレスとホスト名に対してフィルタをかけることができます。
(ホスト名は DNS キャッシュにより名前解決されます)
%source%だけでなく別のプロパティとも組み合わせて使用できます。
ですが、常にこの値を含むと考えられる%source%と組み合わせてご利用になることをおすすめします。

このフィルタは、「<」や「>」の比較オペレーションを利用できるので、それにより IP レンジのフィルタを作成することも可能です。

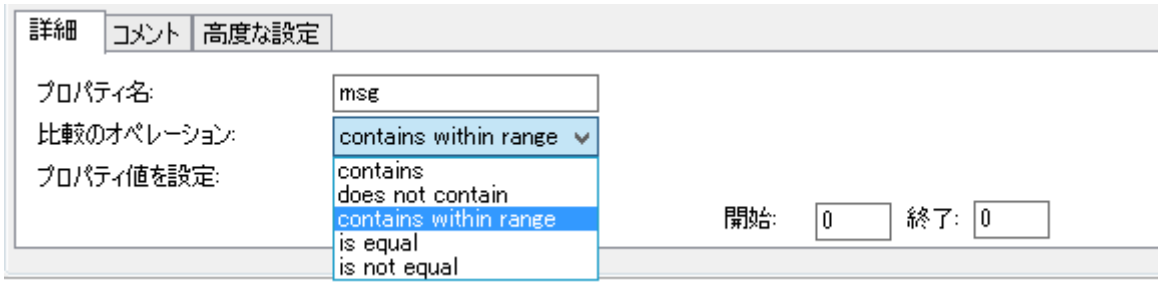
(例)下図では、172.16.0.110 から 172.16.0.130 までの IP をフィルタリングするよう設定しています。



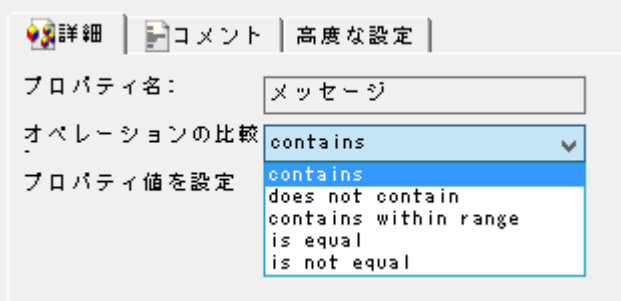
■ メッセージ

このフィルタは、処理されるイベントのメッセージに含まれる文字列をもとにフィルタリングしたい場合に使用します。

「プロパティ値を設定」のテキストボックスに検索したい文字列を直接入力し、「比較のオペレーション」(旧クライアント「オペレーションの比較」)(contains:含む、does not contain:含まない・・・など)と組み合わせて使用します。



▲新クライアント「フィルタの条件:一般>メッセージ」



▲旧クライアント「フィルタの条件：一般＞メッセージ」

「比較のオペレーション」（旧クライアント「オペレーションの比較」）は、以下の値から選択します；

contains	「プロパティ値を設定」に入力した文字列がイベントのメッセージに含まれる
does not contain	「プロパティ値を設定」に入力した文字列がイベントのメッセージに含まれない
contains within range	「プロパティ値を設定」に入力した文字列がイベントのメッセージの指定した範囲内にある *1
is equal	イベントのメッセージが「プロパティ値を設定」に入力した文字列と全く同じ内容である
is not equal	イベントのメッセージが「プロパティ値を設定」に入力した文字列と異なる

*1： 範囲の指示は、「比較のオペレーション」（旧クライアント「オペレーションの比較」）から「contain within range」を選択します。
すると「開始」「終了」（旧クライアント「範囲開始」「範囲終了」）のテキストボックスが現れます。
「開始」と「終了」のボックスにはそれぞれ数値を入力します。
デフォルトでは、「開始」と「終了」ともに 0 になっています。
範囲指定において、「開始」を 0、「終了」を 10 とした場合、最初の文字を 0 として数えますので、9 文字目までが対象となります。
下記のように設定を行った場合、「192.168.0.」が検索の範囲となります。
このように設定を行うことで、「192.168.0.0」から「192.168.0.254」までのデバイスで作成されるログを検出することも可能です。

プロパティ値 を設定= 192.168.0.0
開始 = 0
終了 = 10

上記の設定で、「終了」を 9 としてしまうと、例えば「192.168.010」なども検出されてしまうので、ご注意下さい。

タイプ＝文字列

■ CustomerID

顧客によって CustomerID を変更したい場合に、ここで 整数値を入力します。例えば、顧客のサーバーを監視する際に、エージェント別に違う ID を設定することが可能です。

サーバーA とB を監視しているとして、5 台あるサーバーA は Customer ID を1、2 台あるサーバーB の Customer ID を 2 といった具合で設定することが可能です。

サーバーA と B のサーバー名が同じであっても、CustomerID を設定すれば別の定義を行うことが可能となります。

タイプ＝数字

■ SystemID

SystemID を変更したい場合には、ここで整数値を入力します。

タイプ＝数字

■ ステータス名および値

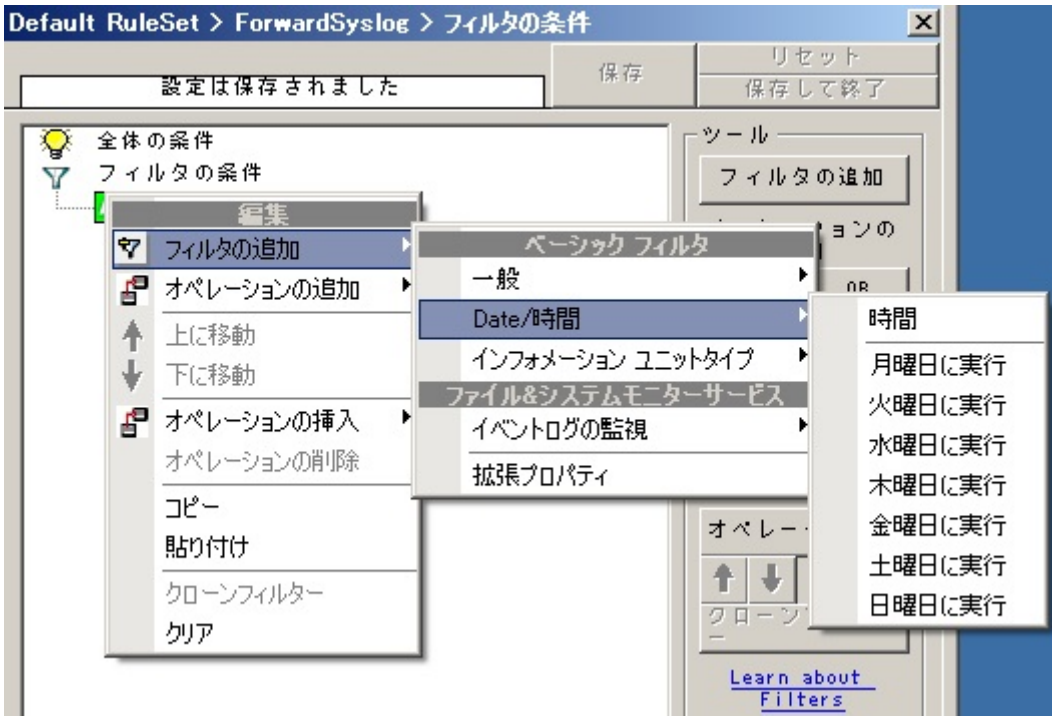
このフィルタのタイプは、「ステータスの設定」アクションに対応しています。

タイプ＝文字列

5.4.7 曜日/時間（Date/時間）

このフィルタの条件は、時間の枠（イベントが発生した曜日）をチェックするために使用されます。

▲新クライアント「フィルタの条件：曜日/時間」



▲旧クライアント「フィルタの条件：一般」

■ 時間

このフィルタの条件は、イベントが発生した時間をチェックするのに使用されます。

例えば、営業時間中であれば、ダイヤルアップしたという内容の Cisco ルーターからメッセージを受信したとしても全く問題ありません。
ですが、それが夜に起こった場合は、警告すべきことなので管理者はこのイベントの通知を受信するでしょう。ここでは、そのような時間を設定できます。

■ 曜日

このフィルタの条件は、上記の時間のフィルタと良く似ていますが、こちらは日をベースにしています。

例えば、週末にイベントが発生し、不審な動きをしていることなどを見つけ出すのに役立ちます。
具体的には以下のフィルタが利用可能です：

月曜に実行（タイプ＝ブール演算）
火曜に実行（タイプ＝ブール演算）
水曜に実行（タイプ＝ブール演算）
木曜に実行（タイプ＝ブール演算）
金曜に実行（タイプ＝ブール演算）
土曜に実行（タイプ＝ブール演算）

日曜に実行（タイプ＝ブール演算）

5.4.8 インフォメーション ユニット タイプ

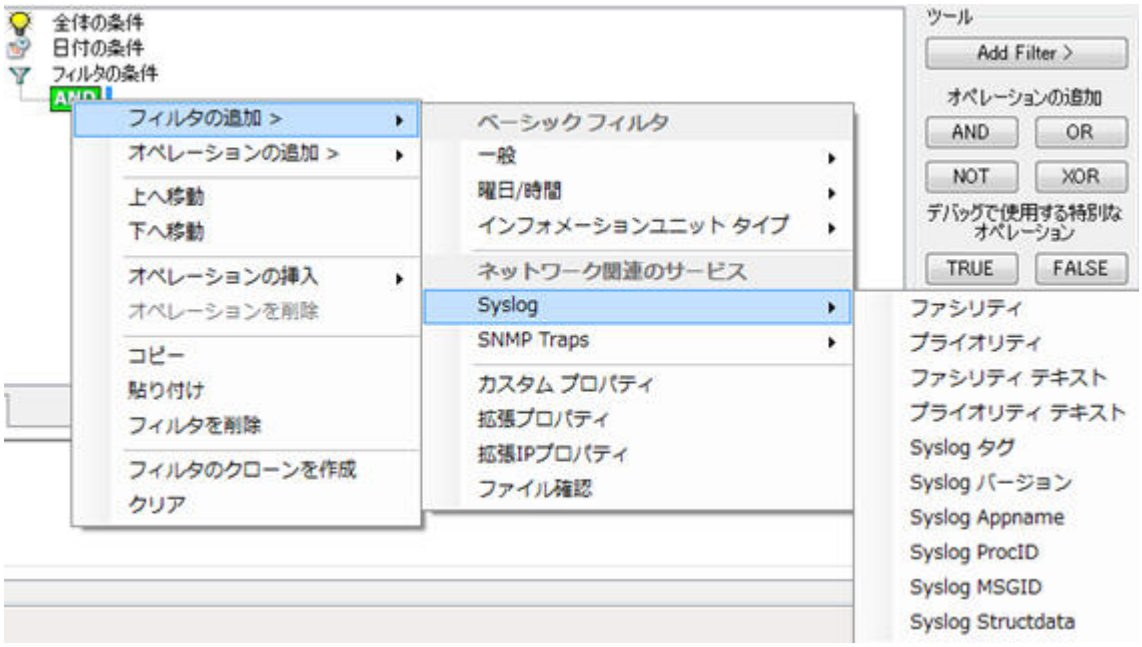
いつくかのインフォメーションユニットタイプに対して、1つのルールを処理させたい場合、そのインフォメーションの種類を選択します。
これは、標準でない処理を必要とする特定のタイプにとって、特に役立ちます。
利用可能な各インフォメーションユニットタイプには、以下のフィルタが定義されています。（下図参照）

具体的には以下のフィルタが利用可能です:

Syslog（タイプ＝ブール演算）
ハートビート（タイプ＝ブール演算）
SNMP トラップ（タイプ＝ブール演算）
RELP リスナー（タイプ＝ブール演算）

5.4.9 Syslog

Syslog 特有のデータでフィルタリングを行う際、以下のフィルタを使用します。



▲新クライアント「Syslog」フィルタ

▲旧クライアント「Syslog」フィルタ

■ Syslog ファシリティ

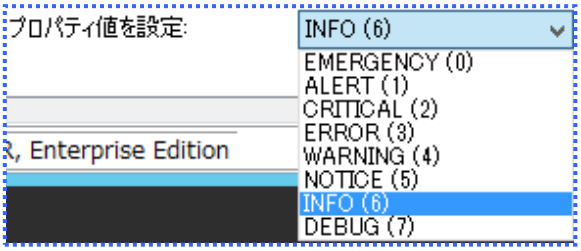
Syslog ファシリティによりフィルタリングを行う場合に設定します。

デフォルトは、LOCAL 0 です。(つまり、ファシリティ値 16)

ファシリティ値は、「詳細」の「プロパティ値を設定」タブから選択し、変更することができます。
Syslog 以外のメッセージ（ハートビートなど）は、ベストエフォート型を基礎としてマッピングされた値となります。

タイプ＝数字

■ **Syslog プライオリティ**



Syslog プライオリティ（Severity）によりフィルタリングを行う場合に設定します。デフォルトは、INFO（6）です。（Value = 6）
※プライオリティ値は数値が小さいほど重要度（Level）が高いメッセージとなります。
プライオリティ値は、「詳細」の「プロパティ値を設定」タブから選択し、変更することができます。
Syslog 以外のメッセージ（ハートビートなど）は、ベストエフォート型を基礎としてマッピングされた値となります。
「詳細」の「オペレーションの比較」では、マッチングモードを選択 することができます。
そこでは、「未満(＜)」、「より大きい（＞）」、「等しい（＝）」、「異なる（Not＝）」が選択可能です。
ここで「＜」と設定すると、「プロパティ値を設定」で入力したプライオリティ値より小さいもの全てがフィルタリングの対象となります。
注意: この場合、（未満なので）「プロパティ値を設定」で入力したプライオリティ値は対象値に含まれません。
もし、その値を含めたい場合は、次に高い値を指定するようにして下さい。

タイプ＝数字

■ **Syslog タグ**

Syslog タグ（メッセージを生成したプログラムまたはプロセスの名前）によりフィルタリングしたい場合に設定します。
デフォルトでは、「オペレーションの比較」が「contains」となっているのみで、「プロパティ値を設定」には何も設定されておりません。

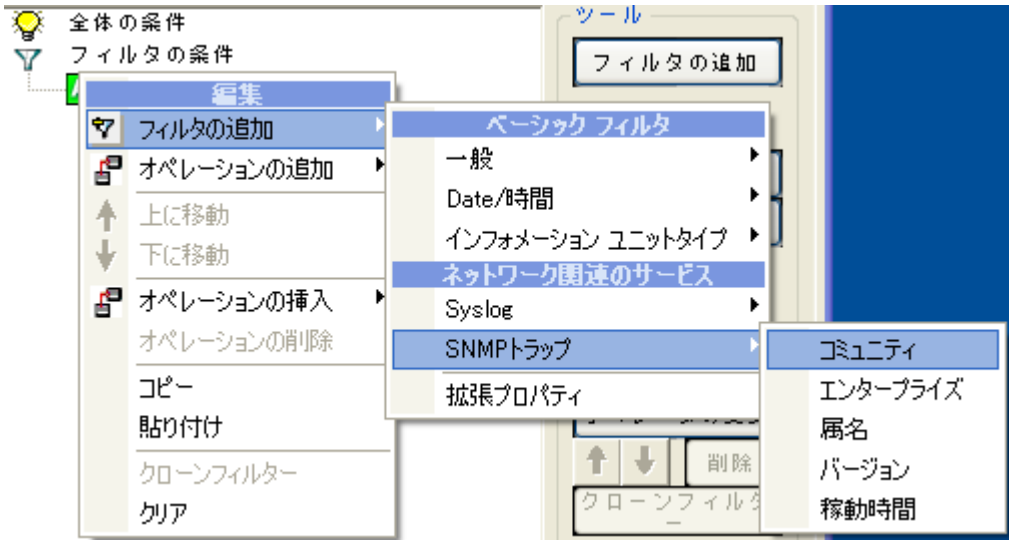
タイプ＝文字列

＊Syslog バージョン、Syslog Appname、Syslog ProcID、Syslog MSGID、Syslog Structdata は、RFC5424（新しい規格）に対応したフィルタです。

5.4.10 SNMP トラップ

SNMP トラップを使用することで、WinSyslog は、コンピューター、ルーター、配線ハブなどを含む いろいろな装置を管理したり、モニターしたりすることが可能になります。
デバイスが送信すべき情報がある場合などにトラップが生成されます。

SNMP トラップ特有のデータによりフィルタリングを行う際、以下のフィルタを使用します。



■ コミュニティ

それぞれの SNMP エンティティに対応します。

フィルタのタイプ＝文字列

■ エンタープライズ

それぞれの SNMP エンティティに対応します。

フィルタのタイプ＝文字列

■ 属名

それぞれの SNMP エンティティに対応します。

フィルタのタイプ＝文字列

■ バージョン

それぞれの SNMP エンティティに対応します。

フィルタのタイプ＝文字列

■ 稼動時間

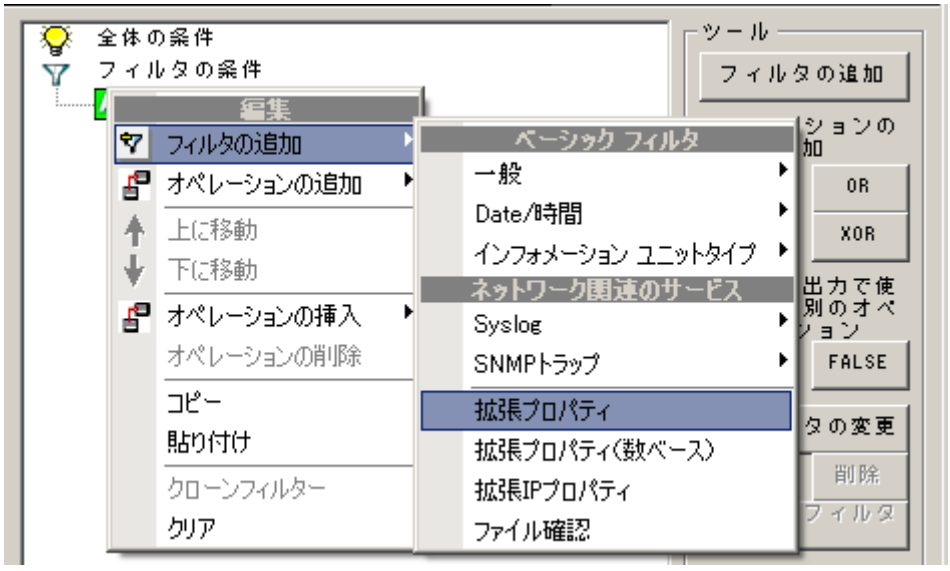
それぞれの SNMP エンティティに対応します。

フィルタのタイプ＝文字列

5.4.11　カスタムプロパティ(拡張プロパティ)

ここでは、プロパティのカスタマイズが行うことが可能です。

▲新クライアント「カスタムプロパティ」フィルタ



▲旧クライアント「拡張プロパティ」フィルタ

WinSyslog の内部で、全ての値はプロパティに保存されています。

例えば、メインのメッセージは「msg」というプロパティに保存されています。

「カスタムプロパティ」で指定することで、直接プロパティにアクセスすることができます。
フィルタのタイプ＝文字列

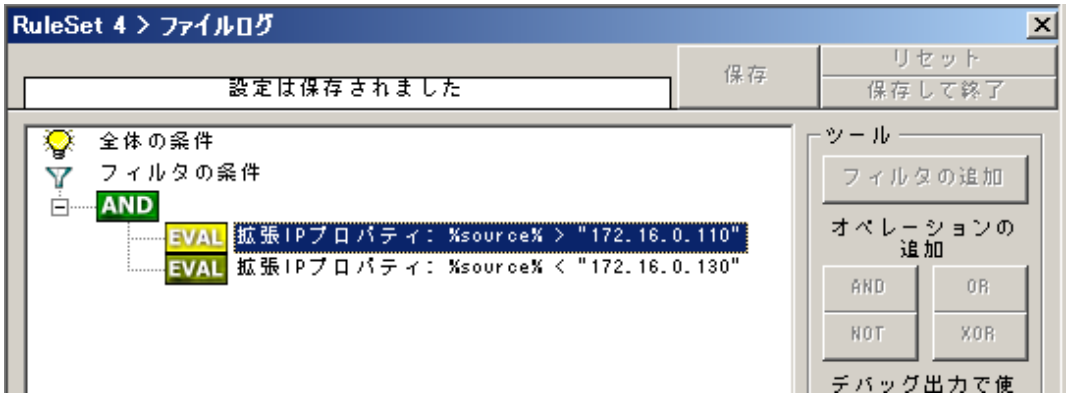
5.4.12　拡張 IP プロパティ

このフィルタは、ホスト名や IP アドレスで絞込みを行う際に使用します。
プロパティ名には、どんなプロパティでも入力できますが、確実にホスト名や IP アドレスが含まれている %source% を使用することをおすすめします。(ホスト名は、DNS キャッシュにより名前解決されます)

このフィルタでは下記オペレーションを使用できます;

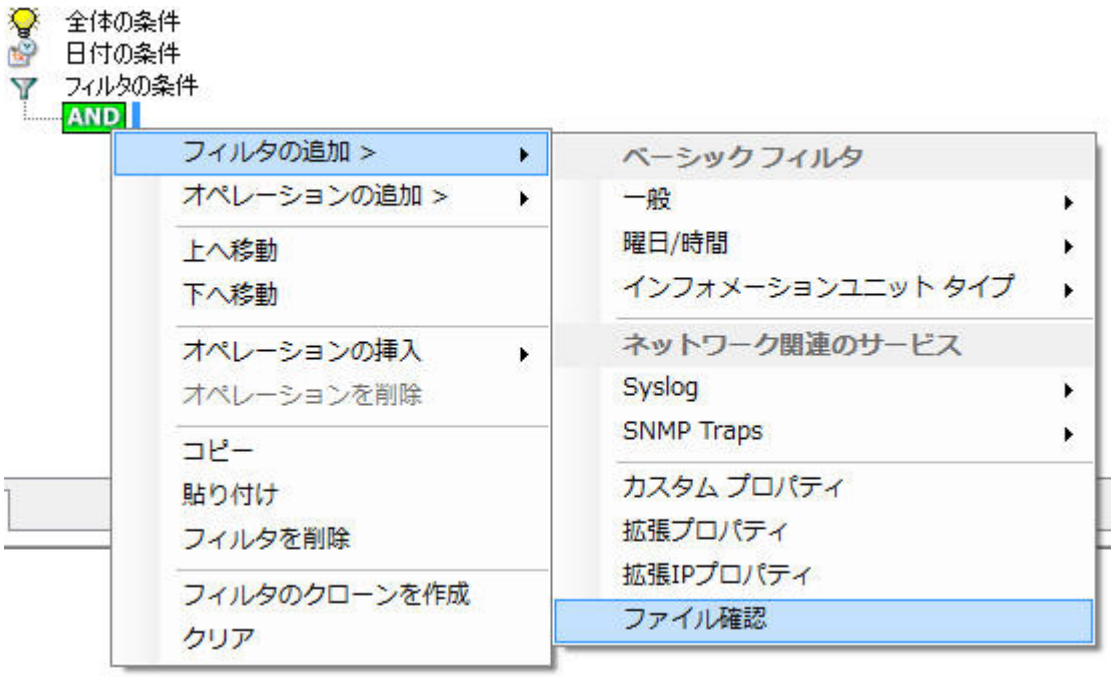
＝	プロパティ値のテキストボックスで入力した IP アドレスに一致すると真(True)になります
Not ＝	プロパティ値のテキストボックスで入力した IP アドレス以外のものが真(True)になります
＞	プロパティ値のテキストボックスで入力した IP アドレスより大きいものが真(True)になります
＜	プロパティ値のテキストボックスで入力した IP アドレスより小さいものが真(True)になります

＞または ＜ のオペレーション使用時は、192.168.0.10、 192.168.0、192.168 や 192 をテキストボックスに入力することができます(どの値を入力するかは、どのようにフィルタをかけるかによります)。

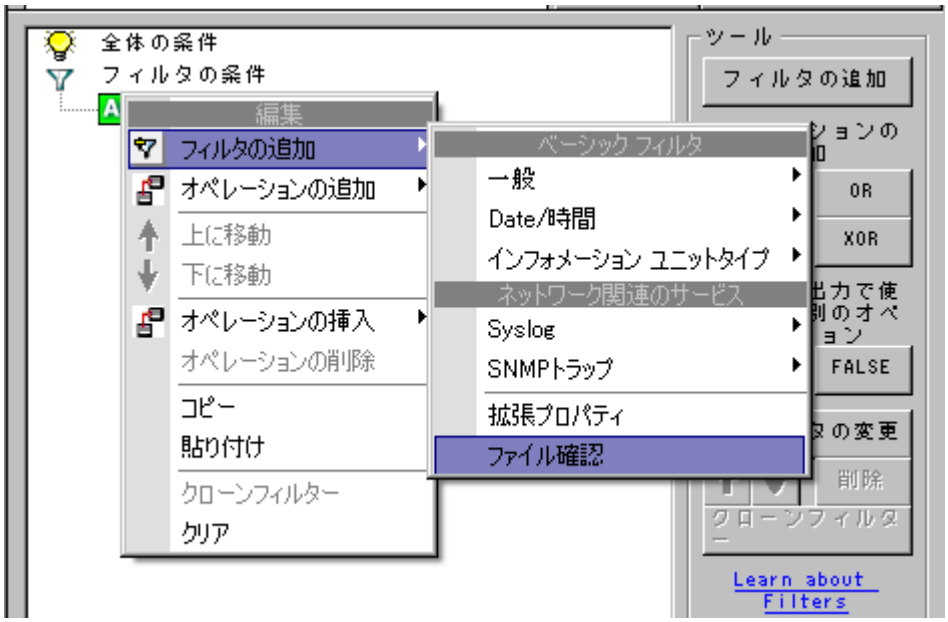


上図のように設定を行うことで、範囲を指定してフィルタリングすることも可能です。
この場合、AND で条件設定されておりますので、「172.16.0.110 より大きい」かつ「172.16.0.130 より小さい」IP アドレスのメッセージが真(True)となり、処理されます。

5.4.13　ファイル確認



▲新クライアント「ファイル確認」フィルタ

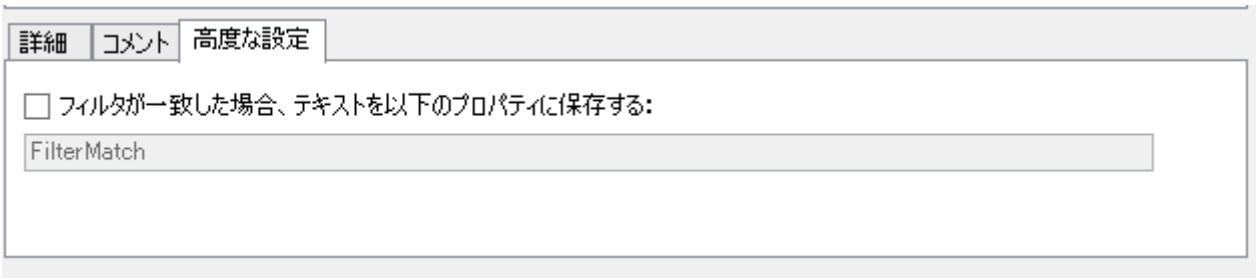


▲旧クライアント「ファイル確認」フィルタ

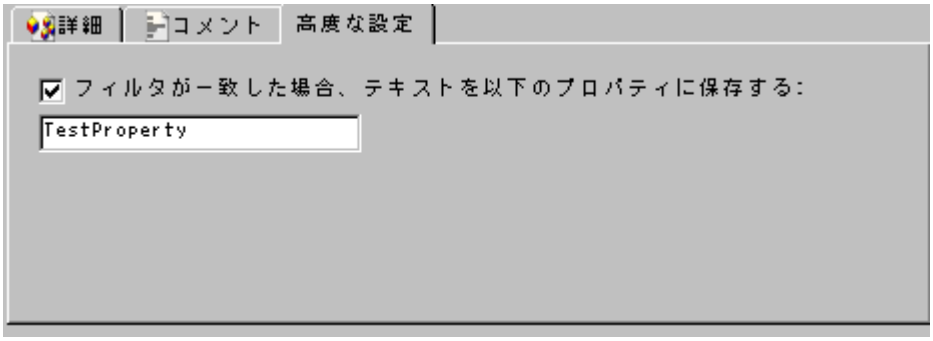
■ ファイル確認

このフィルタでは、設定したファイルが存在するかどうかを確認できます。

5.4.14　フィルタ結果の保存



▲新クライアント



▲旧クライアント

■ フィルタ結果の保存

フィルタがマッチした場合、その結果をカスタムプロパティに保存することが可能です。
さらに、そのカスタムプロパティは、後にアクションで使用することができます。

5.5　アクション

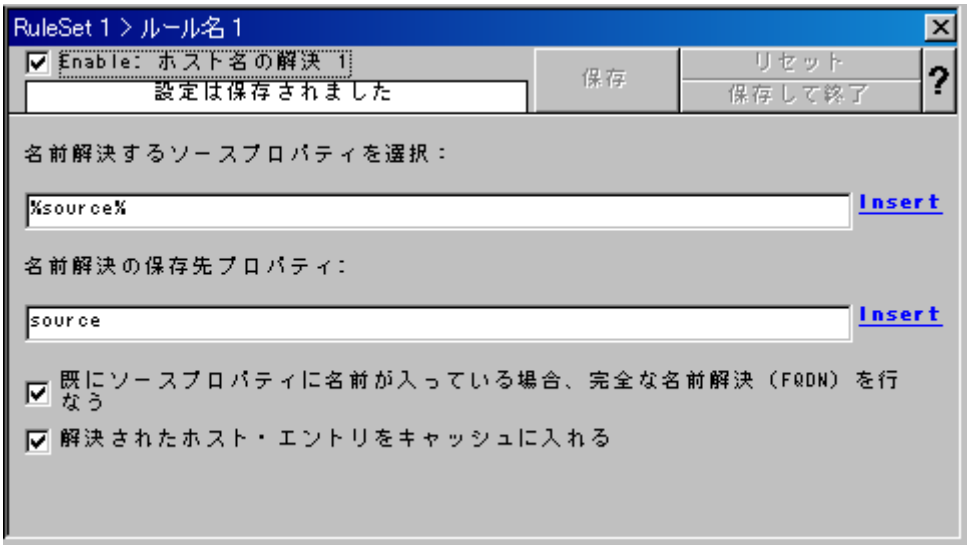
アクションは、そのルール内のフィルタの条件（設定されている場合）が一致したメッセージに対して実行されます。
アクションには、「ファイルログ（受信したログをファイルへ書き込む）」、「メール送信（メールで通知）」、「データベース（データベースへ保存する）」などがあり、お客様のニーズに合わせて設定可能となっております。

アクションは、ルールの配下に作成します。
各ルール内に複数のアクションを作成することもできます。（作成数の制限はありません。）
アクションは、上に表示されているものから順番に処理されてゆきます。
この順番は、「上に移動」、「下に移動」を指示すること（アクションを右クリック）により、変更することも可能です。

5.5.1　ホスト名解決

このアクションにより、全てのサービスでホスト名の解決を実行させることが可能となりました。

▲新クライアント「ホスト名の解決」



▲旧クライアント「ホスト名の解決」

■ 名前解決するソースプロパティを選択

ここでは、名前解決を実行するプロパティを選択します。

テキストボックスの右にある Insert をクリックし、プロパティ値を選択して下さい。

■ 名前解決の保存先プロパティ

ここでも同じようにテキストボックスの右にある Insert をクリックし、名前解決の保存先のプロパティを選択します。

■ 名前解決されたホストをキャッシュに入れる（解決されたホストエントリをキャッシュに入れる）

ここを有効にすると、名前解決されたホストエントリをキャッシュに入れることができます。

■ 既にソースプロパティに名前が入っている場合、完全な名前解決（FQDN）を行なう

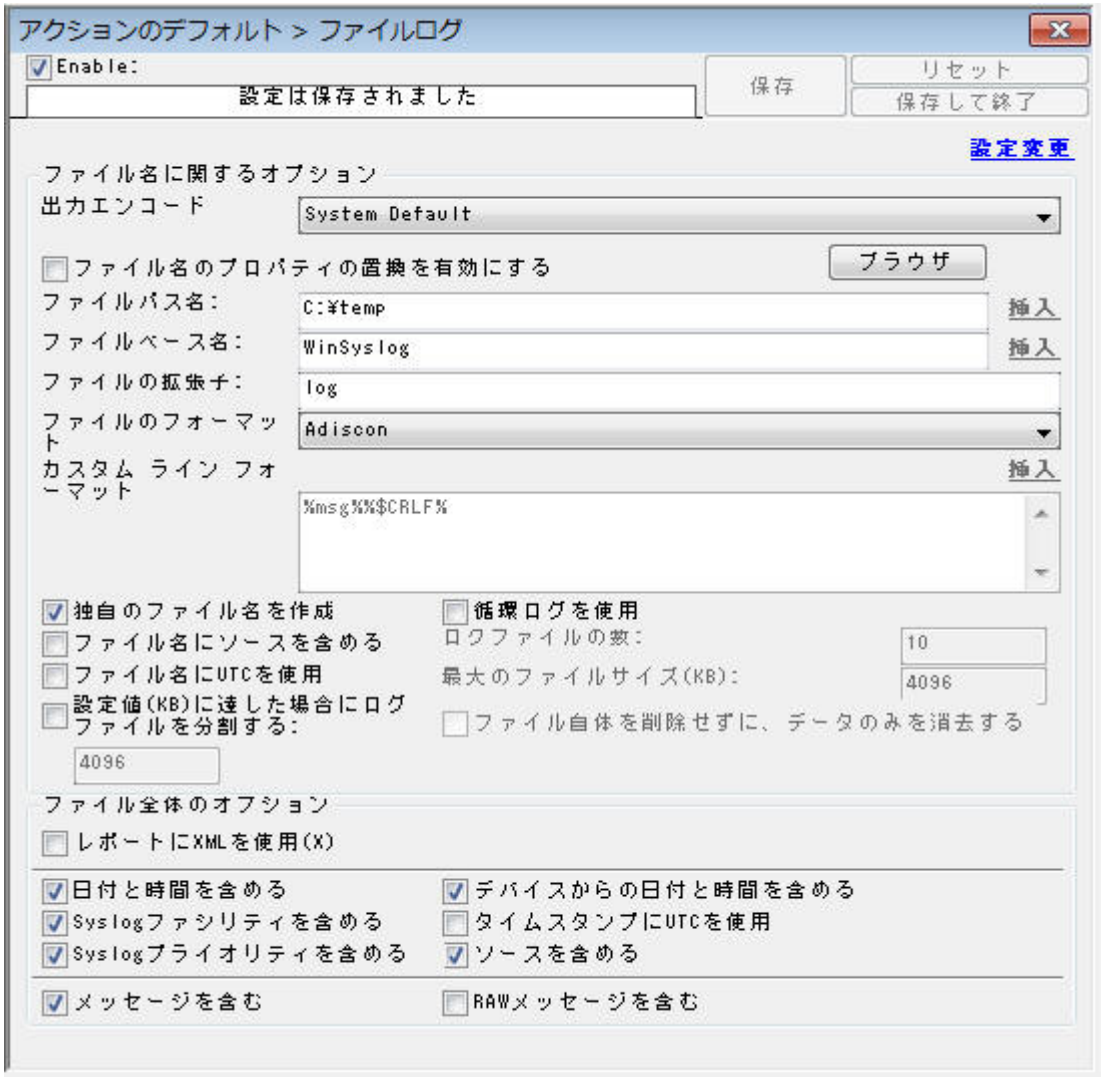
ここを有効にすると、（可能な場合）FQDN を実行します。

例えば、ソースプロパティが既に「servername」と表示されている場合に、この機能を有効にすると完全な名前解決が行なわれ、「servername.mydomain.com」などと表示されるようになります。

5.5.2 ファイルログ

この アクションは、受信したメッセージをテキストファイルに書き込む 際に使用します。

▲新クライアント「ファイルログ」



▲旧クライアント「ファイルログ」

デフォルトでは、**ファイル名に日付を出力(独自のファイル名を作成)**が有効になっておりますので、一日あたりひとつのファイルが記録されます。新しい項目は、そのファイルの最後に付け加えられます。

ログの書き込みを行っていないときには、このログファイルのロックは解除されています。
そのため、WinSyslog サービスが動作している間でも、他のアプリケーションはファイルにアクセスすることができます。
しかし、他のアプリケーションがファイルをロック状態で開いてしまうと、WinSyslog はログの書き込みができなくなり(アクセスできず)、エラーが発生します。
(イベントログに ERROR_SHARING_VIOLATION のエラーが書き込まれます)
もしも、WinSyslog の稼働中に書き込み中のログファイルを開く場合には、ロック状態でオープンしないアプリケーション(notepad.exe など)をご使用ください。
ファイル名は、ダイアログで設定されている括弧の中のパラメーターで、次のように生成されます:

<ファイルパス名><ファイルベース名>-年-月-日.<ファイルの拡張子>

■ 出力エンコード

受信したログをファイルに書き込む際の出力エンコードをここから選択できます。

■ 設定変更（旧クライアント）

この機能は、Adiscon 社の別の製品である MoniLog、および MonitorWare コンソールをご使用になるユーザーのための機能です。

■ ファイル名にプロパティ（変数）を使用（ファイル名のプロパティの置換を有効にする）

ここを有効にすると、ファイルでプロパティを使用したり、%source% などのファイルパス名を使用したりが可能となります。

例えば、ファイルパス名：F:¥syslogs¥%source%、ファイルベース名：IIS-%source%、ソースが 10.0.0.1 の場合、ファイル名は下記のようになります。
F:¥syslogs¥10.0.0.1¥IIS-10.0.0.1.log

この設定では、ソースのプロパティ値である %source% がパスの中に使用されているので、その値がファイル名に置換され、上記のように表現できるようになります。
ファイルパス名とファイルベース名には、その他にも様々なプロパティを指定することが可能です。
WinSyslog のプロパティにつきましては、「WinSyslog プロパティリスト」をご参照下さい。
http://www.jtc-i.co.jp/support/documents/tips/winsyslog_propertylist.pdf

手順につきましては、「標準ログサーバー設定」も合わせてご参照下さい。
http://www.jtc-i.co.jp/support/documents/tips/winsyslog_logserver_setting.pdf

■ ファイルパス

ファイルを保存するフォルダのパス（ディレクトリ）を指定します。
テキストボックスに直接入力するか、「参照」ボタンから保存先のフォルダを選択します。
デフォルトは C:¥Program Files (x86)¥WinSyslog （c:¥temp） です。

「ファイル名にプロパティ（変数）を使用」機能を有効にしている場合、パス名にソースなどのプロパティを入力することが可能です。（ファイルパス名：F:¥syslogs¥%source%など）

■ ファイルベース名

ファイルのベース名を入力します。これは、具体的な日付などの情報以前の部分です。
入力については上図を参照してください。デフォルトは「WinSyslog」 です。

ここでも **「ファイル名にプロパティ(変数)を使用」**機能を有効にしている場合は、「挿入」をクリックすることで、パス名にソースなどのプロパティを入力することが可能です 。

■ ファイルの拡張子

拡張子は、ログファイルを書き込むときに使用されます。
入力については上図を参照してください。デフォルトは「log」 です。

■ ローテーションを無効にする

ここが有効である時は、ログファイルはローテーションされません。

■ ファイル名に日付を出力(独自のファイル名を作成)

ここをチェックすると、ファイル名に日付が含まれるようになります。(例:WinSyslog-2009-04-30.log)
(つまり、ログファイルが毎日作成されるようになります。)

チェックしない場合は、ログファイルは切り替わることなく、設定したファイル(デフォルト:winsyslog.log)に出力され続けることになります。
(ファイルサイズ等の制限はありませんので、テキストエディタで読み込み可能なファイルサイズを超えないよう注意してください。)

■ ファイル名にソースを出力(ファイル名にソースを含める)

ここをチェックすると、ファイル名の中にデバイスのソースが含まれるようになります。
したがって、**ログを報告しているデバイス毎にファイルが作成**されます。

■ ファイル名に UTC を使用

これは、「独自のファイル名を作成」の設定とともに機能します。
新たに日付を含むファイルを作成する場合に、その日付(時間)を UTC(全世界で時刻を記録する際に使われる公式な時刻)を基準にするか、またはローカルタイムを基準にするかを選択します。

UTC は GMT(グリニッジ標準時)とほぼ同じですが、こちらの方がより正確です。
日本の場合、ローカルタイムは UTC より 9 時間進んでいます。UTC で正午ならば、日本は午後9時です。

複数のタイムゾーンのログファイルを作成し、後でそれらをまとめるといった場合は、UTC を基準にすることをお勧めします。
時差の問題に無関係の場合には、ローカルタイムを基準(無効のまま)にして下さい。
注意: これは、ログファイルの作成の切り替わり時刻について設定するものです。ログファイル内で記録される日付に関しては、別の設定になります。

■ 設定値(KB)でファイルを分割(ファイルサイズ (KB)が設定値に達した場合にファイルを分割する)

ここを有効にすると、設定したサイズに達したファイルが分割されます。

その場合、ファイル名には連続した番号が追加されます。(例:WinSyslog-2005-04-26_1.log　ファイルの末尾に「_1」から順番に番号が追加されてゆきます)

■ ローテーションを有効にする(循環ログを使用)

ここを有効にすると、ファイルサイズの最大値(最大のファイルサイズ(KB))で指定したファイルサイズに達する毎にログファイルを作成してゆき、ログファイルの数で指定した本数分作成されるとファイルが循環してゆきます。

■ ログファイルのデータを消去(ファイル自体は削除されません)

(ファイル自体を削除せずに、データのみを消去する)

このオプションは、ローテーションを有効にする(循環ログを使用)オプションとともに使用します。

その名のとおり、このオプションを有効にすると、ファイルがローテーションする際、元のファイルは削除されず、その中身(データ)だけが消去されるようになります。

WinSyslog のログファイルを別のアプリケーションで監視している場合などに有効です。

＜ファイルフォーマット＞

ここは、ログファイルを書き込む際のフォーマットを設定します。デフォルトは「Adiscon」です。

■ Adiscon

Adiscon フォーマットを選択した場合には、以下にある様々な出力オプションを選択することができます。

■ メッセージに XML を出力(レポートに XML を使用)

有効にすると、ログに XML フォーマットされた情報の記録が含まれます。

それには、解析が簡単なフォーマットでタイムスタンプや Syslog ファシリティ、プライオリティやその他追加の情報も含まれます。

XML 出力フォーマットを選択した場合、その他全てのフィールド情報は XML ストリームに既に含まれていますので、それらの機能はオフにしても構いません。しかし、これは必要条件ではありません。

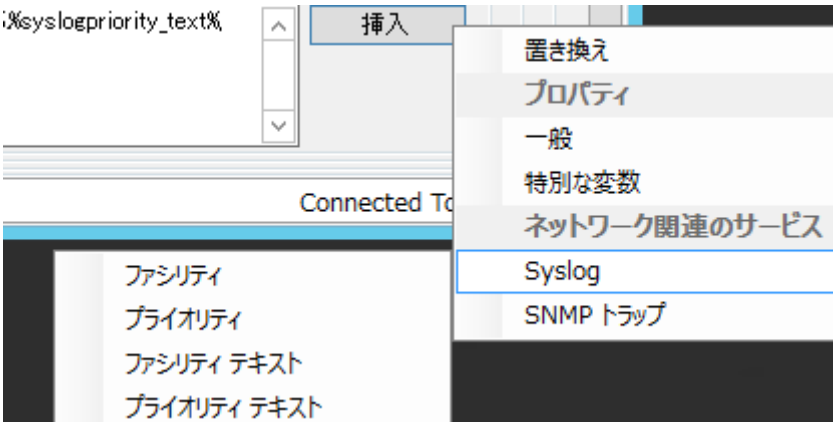
■ 日付と時間を出力(日付と時間を含める)

有効にすると、ログに日付と時間が出力されます。

■ Syslog ファシリティを出力(Syslog ファシリティを含める)

有効にすると、ログに Syslog ファシリティが出力されます。数字で出力されますので、テキストで出力したい場合は、「カスタムフォーマット」を選択し、ファシリティテキスト(%syslogfacility_text%)を挿入します。

挿入＞Syslog＞ファシリティテキスト



■ Syslog プライオリティを出力（Syslog プライオリティを含める）

有効にすると、ログに Syslog プライオリティが出力されます。数字で出力されますので、テキストで出力したい場合は、「カスタムフォーマット」を選択し、プライオリティテキスト（%syslogpriority_text%）を挿入します。

■ 日付と時間（デバイスのタイムスタンプ）を出力（デバイスからの日付と時間を含める）

有効にすると、ログに日付と時間が出力されます。

注意： 出力オプションの一番上にある**日付と時間を出力**は、WinSyslog がメッセージを受信した時間を出力するものです。

一方、**日付と時間（デバイスのタイムスタンプ）を出力**は、受信したメッセージからタイムスタンプを取ります。

従って、報告するデバイスが持っている時間（オフになっている場合もあります）に左右されます。

さらに、Syslog メッセージの場合、デバイスが報告するタイムスタンプにはタイムゾーンの情報がありません。

従って、複数のタイムゾーンで報告するデバイスが存在する場合は、タイムスタンプの情報は、ばらばらになってしまいます。（これは、RFC 3164 の Syslog の仕様によるものです。 ）

■ タイムスタンプに UTC を使用

有効した場合は、ログファイル内のタイムスタンプは全て UTC で書き込まれます。

有効にしない場合には、ローカルタイムでログが書き込まれます。

UTC は複数のタイムゾーンでログを管理したい場合に、特に有効です。

■ ソースを出力（ソースを含める）

有効にすると、ログにソースが出力されます。

■ メッセージを出力（メッセージを含める）

有効にすると、ログに受信したメッセージが出力されます。（Syslog メッセージのメッセージ部分：タグ値（PRI 部）、ホストインフォメーションなど（HEADER 部）を除いたものが出力されます）

■ **RAW メッセージを出力（RAW メッセージを含める）**

有効にすると、ログに受信した RAW メッセージが出力されます。（RAW：全く変更が加えられていないもの）
これは、他のアプリケーションで RAW メッセージが必要とされている場合に選択します。
注意：「メッセージを含む」と「RAW メッセージを含む」のいずれかを選択するようにして下さい。
どちらも有効になっていない場合には、メッセージが全く書き込まれません。
また、両方を選択してしまうと、二重にメッセージが書き込まれますので、ご注意下さい。

その他のフォーマットは、ログファイルの互換性を他のアプリケーションと持たせるために使用されます。

■ **Raw Syslog メッセージ**

ここを有効にすると、ログが Raw Syslog フォーマットで書き込まれます。Syslog メッセージの各ラインは RFC3164 で書かれています。特別のフィールド処理や情報の追加などは行われません。
他の（互換性を持たせたい）アプリケーションの中にはこのフォーマットが必要なものもあります。

■ **WebTrends syslog 互換**

WebTrends アプリケーションが期待するフォーマットを模倣したものです。
このフォーマットは、そのログファイルのフォーマットを模倣しただけに過ぎないということに注意してください。

■ **カスタムフォーマット（カスタム ライン フォーマット）**

ここを有効にすると、ログファイルの出力を完全にカスタマイズすることが可能になります。
ファイルのフォーマットにおいて、「カスタムフォーマット」を選択した場合のみ、この機能は有効になり、「挿入」をクリックすることで項目を追加できます。
デフォルト値は、「%msg%%$CRLF%」です。（%msg%：メッセージ、%$CRLF%：改行コード）

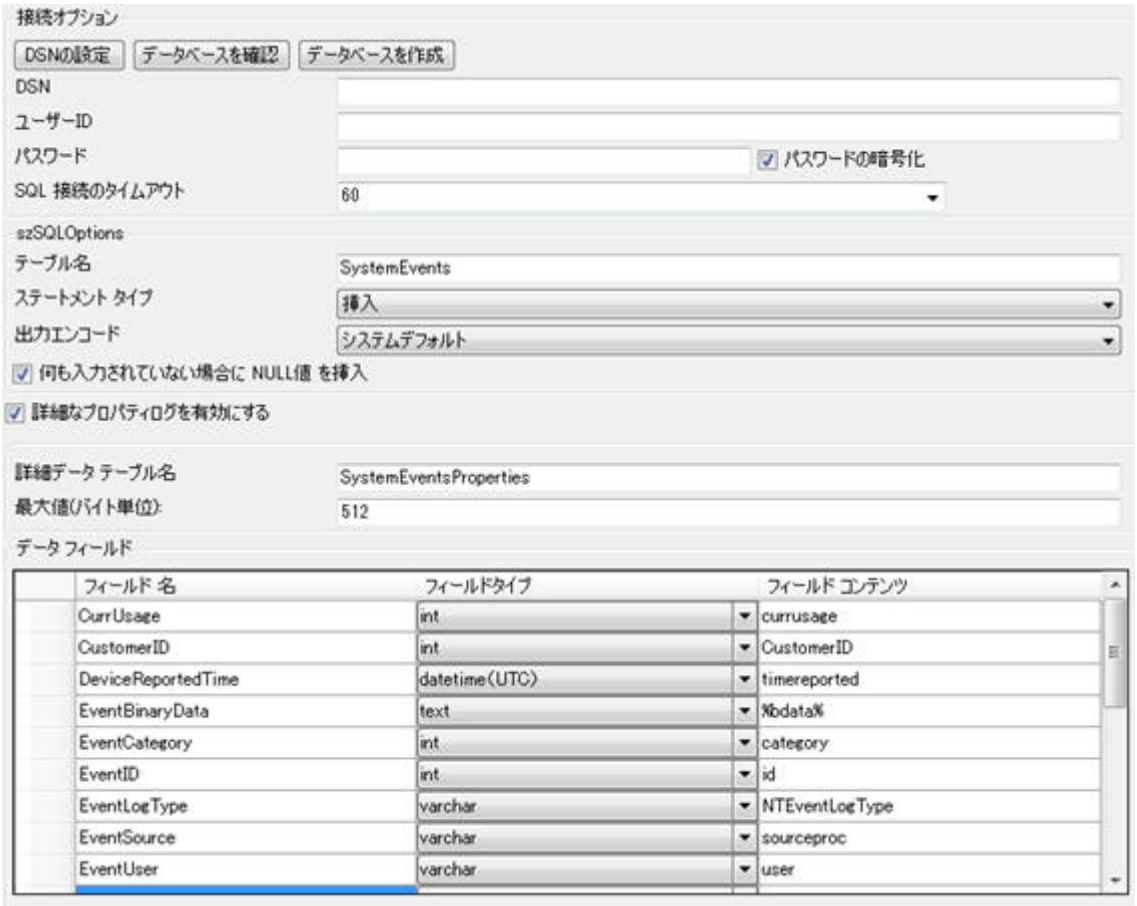
＜表示される時刻につきまして＞

作成時刻（%timegenerated%）、および報告時刻（timereported%）のプロパティをご利用になる場合、そのままでは **UTC タイム（-9 時間）**で記録されます。
ローカルタイムで記録したい場合には、それぞれの値の代わりに下記の値（:::localtime が挿入されています）をご利用下さい：

%timegenerated:::localtime%　　%timereported:::localtime%

5.5.3　ODBC データベース

データベースログをご利用頂くことで、 受信したメッセージをデータベースへ保存できます。
データベースに保存されたログは、メッセージビューアやカスタムアプリケーションで簡単に閲覧することができます。



▲新クライアント「ODBC データベース」

▲旧クライアント「ODBC データベース」

データベースログは、ODBC に対応したデータベース（実際、Windows の OS で使用可能な、いかなるデータベースシステムでも）へ受信した Syslog メッセージを記録することができます。
Microsoft JET データベース（Microsoft Access で使用）と Microsoft SQL サーバーと MySQL は、「データベース生成」でテーブルの作成がサポートされています。 オラクルや Sybase など様々なシステムで正常に稼動している例もあります。

データベースログのアクションで最も重要なのは、フィールドの部分です。
デフォルトは、代表的なイベントプロパティの割り当てをデータベース列へ反映します。
ですが、この割り当ては、自由に変更することが可能です。

「フィールド名（Filedname）」は、データベース列の名称です。予め設定されたフィールド名は、Adiscon のスキーマが使用するものです。必要ならば、名称は変更することが可能です。

「フィールドタイプ（Fieldtype）」は、データベース列のデータタイプです。それは、データベースで選択される列タイプを反映しなければなりません。また、このデータタイプは、保存される実際のプロパティと一致していなければなりません。例えば、syslogpriority のような整数タイプのプロパティは、varchar 列に保存することができます。一方、syslogtag のような文字列データタイプは、整数列に保存することはできません。

「フィールドコンテンツ（Fieldcontent）」は、イベントプロパティです。サポートされたプロパティリストに関しては、「WinSyslog プロパティリスト」をご参照下さい。
http://www.jtc-i.co.jp/support/documents/tips/winsyslog_propertylist.pdf

データベース フィールド内の値を編集は、新クライアントと旧クライアントで操作が異なります。

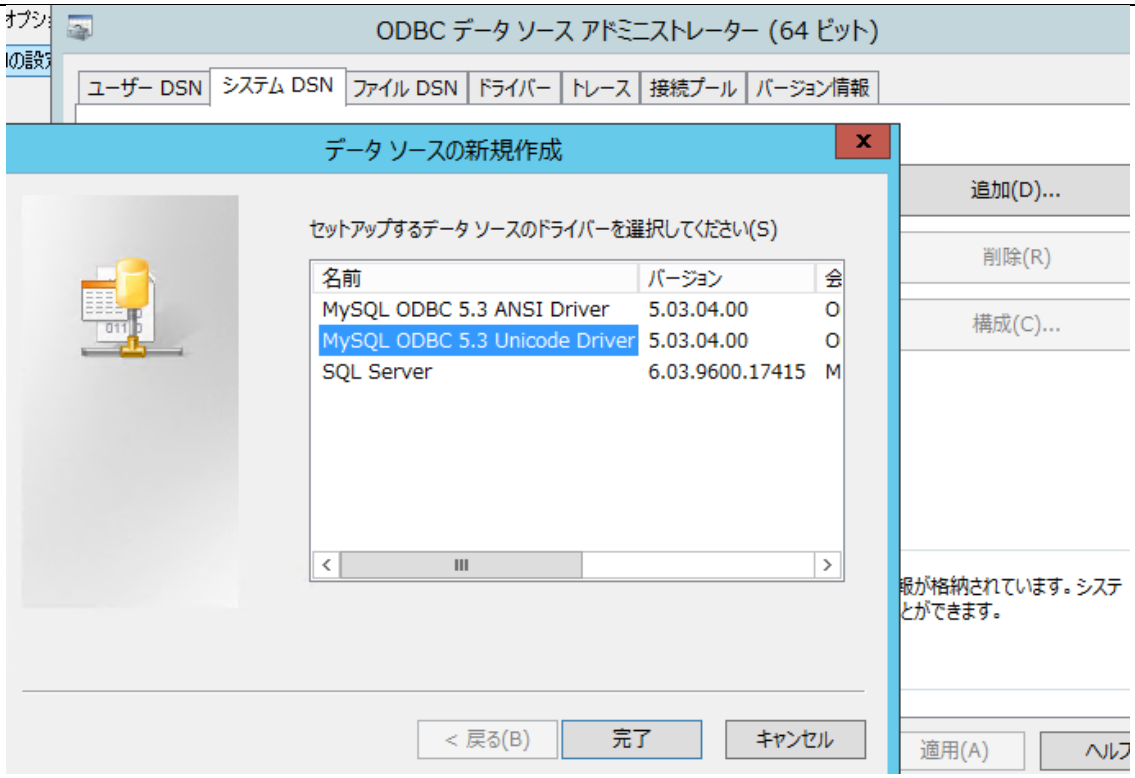
新クライアントでは、表の中の値を直接編集することができます。
項目の追加は、一番下の列に直接入力する形となります。
列を選択し、Delete キーを押すことで項目を削除することも可能です。

一方、旧クライアントで値を編集するには、列を選択します。
選択した列の値は、フィールドリストの上のテキストボックスで変更できます。
また、「挿入」、「削除」のボタンをクリックすることで、フィールドの作成、削除を行うことが可能です。
「削除」ボタンをクリックすると、選択されているフィールドが削除されます。また、上・下の矢印ボタンをクリックすることで、選択したフィールドを移動させることができますが、この移動は表面的なもので、データベースアクションの処理には、何も影響ありません。

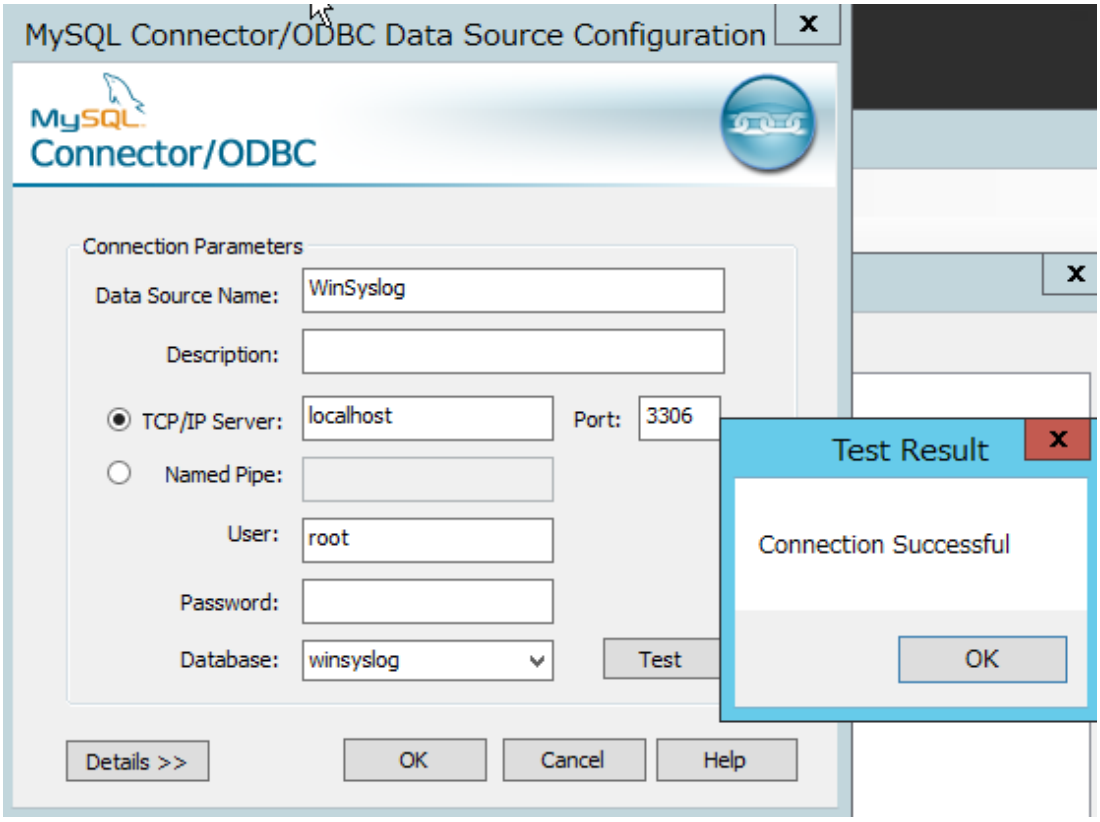
フィールドコンテンツでは、プロパティの置換機能を使用することができます。例えば、メッセージの最初の 200 文字だけを保存したい場合には、"%msg: 1:200%"と設定することで可能です。

■ DSN の設定

ここをクリックすると ODBC データソースアドミニストレーターが表示されます。
システム DSN で設定を行います。
※64bit システムで、Adiscon のサービスを 64bit アプリケーションとして稼動させるために必要なドライバーがあります。 例：MySQL ODBC Connector ver.5.3.4(64bit)

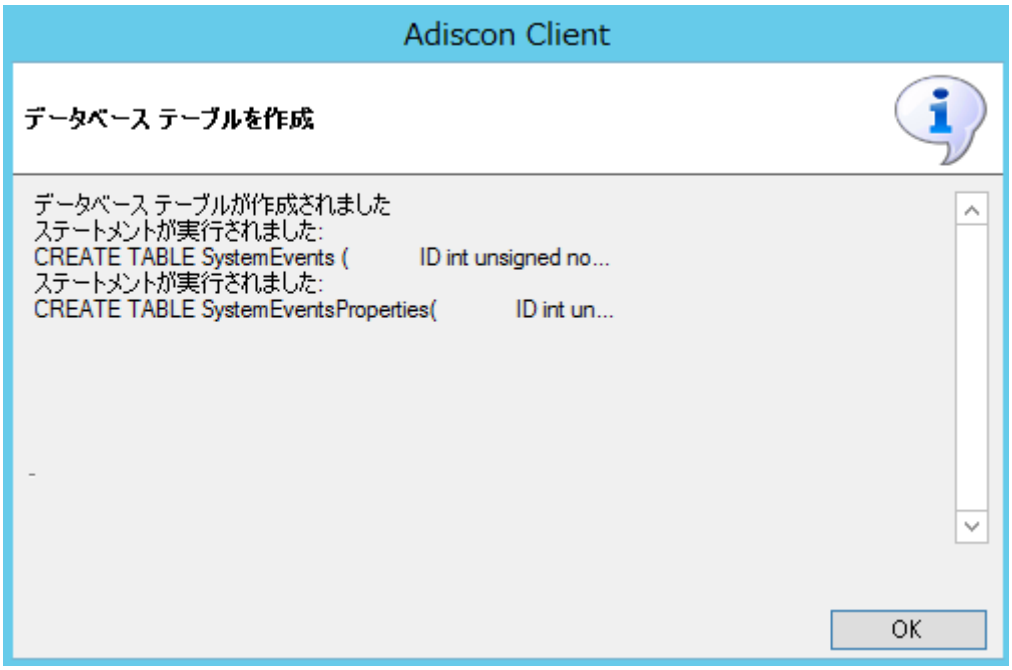


あらかじめ　任意の名前のデータベースを作成しておきます。(ここでは winsyslog　)



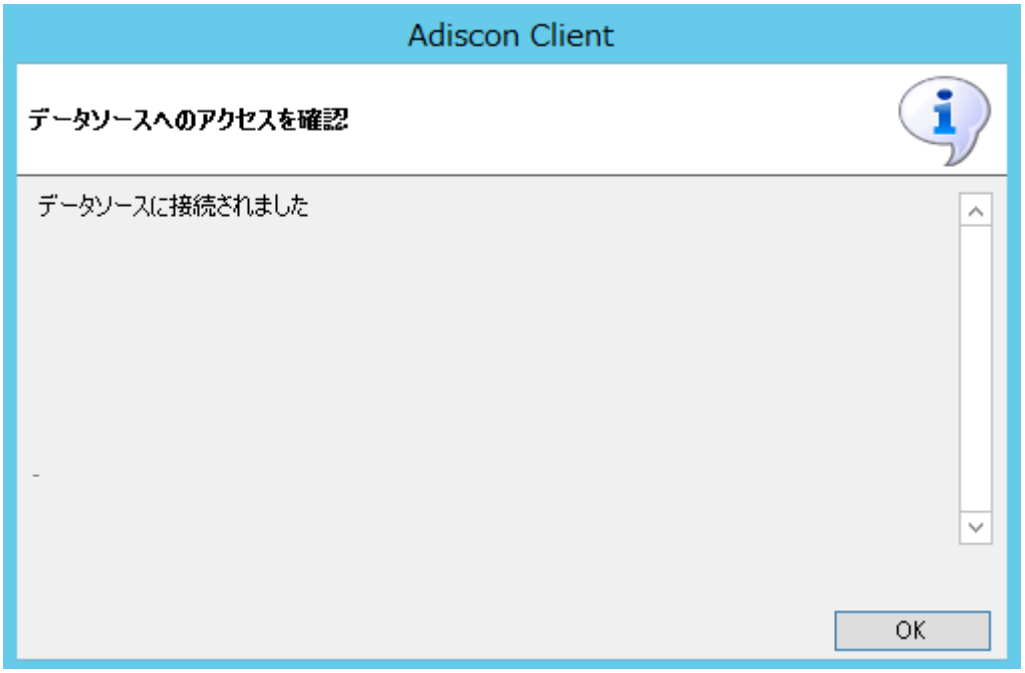
■ **データベースを作成（新クライアント）**

このボタンをクリックすると自動的に SystemEvents と SystemEventsProperties の 2 つのデータベースが作成されます。

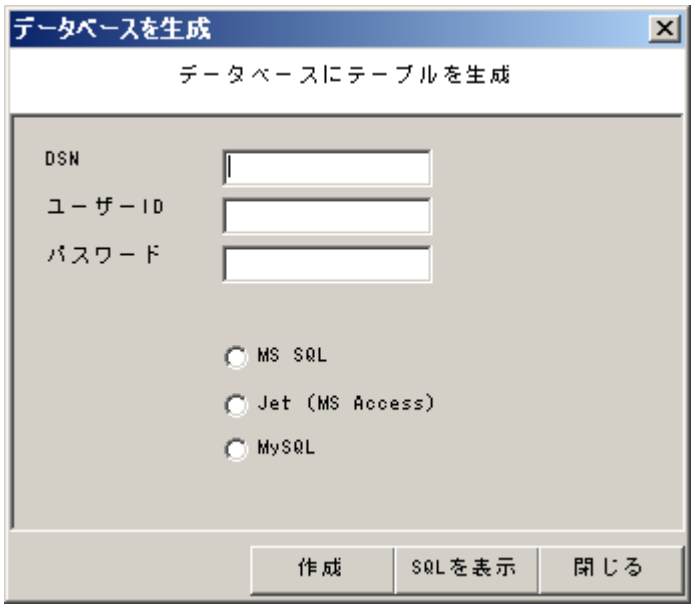


■ **データベースを確認**

このボタンをクリックすると、データベースへの接続を確認することができます。



■ **データベースを生成（旧クライアント）**



▲旧クライアント「データベースを生成」

このフォームでは、基礎となるデータベースの DSN、ユーザーID、パスワードを設定します。
設定後は、データベースにテーブルを作成するために 「作成」ボタンをクリックします。実行される SQL クエリを確認するためには、「SQL を表示」ボタンをクリックします。「閉じる」ボタンにより、このフォームを終了できます。

■ DSN

DSN を入力します。

■ ユーザーID

データベースに接続する際に利用するユーザーID を入力します。
ユーザーID の設定をしなければならないか否かは、データベースシステム次第です。
（例えば、マイクロソフトの Access は設定の必要がなく、一方、マイクロソフトの SQL サーバーはユーザーID を使うように強いられます。）

■ テーブル名

ログを取るテーブルの名前を入力します。
この名前は、SQL のインサート・ステートメントを作成するために使用されるので、データベースの定義と適合しなければなりません。デフォルトは、「SystemEvents」です。

■ パスワード

データベースに接続する際に利用するパスワードを入力します。
それは、「ユーザーID」で指定したアカウントで利用されるパスワードでなければなりません。
ユーザーID のように、パスワードが必要かどうかは、データベースシステムに左右されます。

パスワードは、暗号化しても、暗号化しなくても保存できます。
ですが、暗号化して保存するようお勧めします。

■ 暗号化

ODBC のパスワードを暗号化して保存する際に、ここをチェックします。
チェックしない場合は、パスワードは暗号化されずに保存されます。
なるべくチェックして、暗号化させることをお勧めします。

もし、何らかの理由で、暗号化せずにパスワードの保存を行う場合は、そのセキュリティに気を付けてください。
この場合、限られたアクセス権でアカウントの使用をすることをお勧めします。
暗号化をしない場合であっても、同様に限られたアクセス権でアカウントを使用することをお勧めします。ここでは、強固な暗号を適用することができないからです。

■ SQL ステートメントタイプ

MSSQL のストアドプロシージャ（Stored Procedures）の INSERT、または Call Statement（MSSQL ストアドプロシージャ）のいずれかを選択できるようになりました。この機能は、MSSQL をデータベースソフトとしてご利用になる場合のみ有効です。
また、Call Statement を選択した場合、テーブル名が自動的にプロシージャ名として使用されます。正しくパラメーターがソートされるようにして下さい。そうでないと、アクションが正常に動作しません。また、その場合、この結果はデバッグログで比較しないと分かりません。

■ 出力エンコード

ここでは、データベースへ書き込みを行う際に使用する文字コードを設定します。

■ SQL 接続のタイムアウト

ここでは項目名の通り、接続のタイムアウトを設定します。

■ 何も入植されていない場合に NULL 値を挿入（プロパティが空の場合、NULL 値を挿入）

このオプションを有効にすると、プロパティが空の場合、NULL 値が挿入されるようになります。

■ 詳細なプロパティのログを有効にする

このオプションは、SystemEventProperties テーブルに標準のプロパティ以外のイベントプロパティを記録します。場合によっては、一つのイベントに複数のプロパティがある可能性があるので、このオプションを選択することで、複数のプロパティを書き込むことが可能となります。しかし、Syslog データでは、追加のプロパティはあまり存在しません。（「Post Property」アクションで独自の定義をしている場合を除いては）

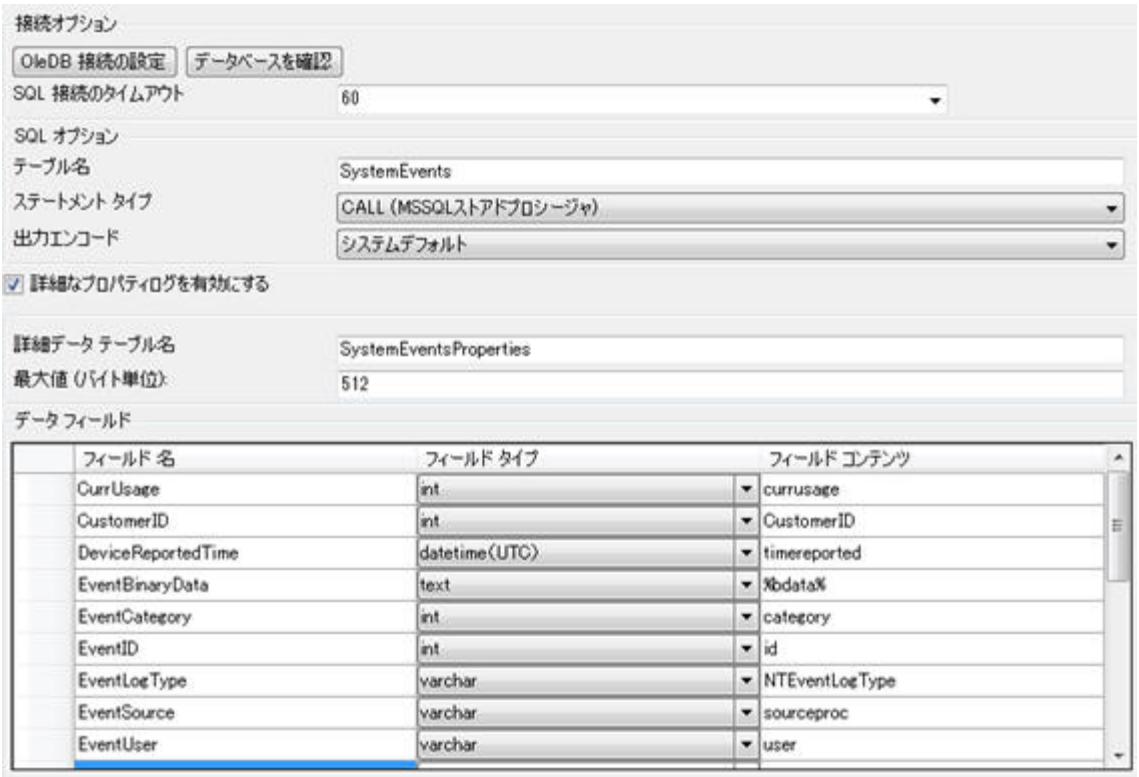
追加のプロパティは、概して イベントログの監視からの SETP 受信データにあります。例えば、SETP で受信した

イベントログのデータには、実際の Windows のイベントプロパティとイベントデータが含まれています。このオプションは、Syslog で受信したイベントログ メッセージには適用されませんので、注意してください。

このオプションは、有効にする前に、必要がどうかを確認するようにして下さい。
（このオプションのデータは、MonitorWare コンソールでは必要となる場合があります。）

5.5.4 OLEDB データベース

OLEDB データベースアクションの設定は、ODBC データベースログアクションと同様に行うことができます。
Win32 版では、MS SQL OLEDB プロバイダと JET4.0 OLEDB プロバイダは、問題なく動作することが確認できておりますが、x64 版では JET4.0 OLEDB プロバイダは現在までのところ対応しておりません。
開発元での内部パフォーマンステストの結果、OLEDB のアクションは、ODBC に比べおよそ 30%の機能強化が確認できております。従って、大量のデータをデータベースに書き込みをしたい場合には、特に役立つと思われます。
このアクションにより、収集したイベントを OLEDB 対応のデータベースに書き込むことが可能となります。保存されたメッセージは、カスタムアプリケーションと同様に、別のメッセージビューアで簡単に閲覧することができます。
データベース・フォーマットは、変更することが可能です。これは、データベース上で更なる分析を行なう場合に非常に役立ちます。



▲新クライアント「OLEDB データベース」

▲旧クライアント「OLEDB データベース」

OLEDB データベースアクションのメイン機能は、フィールドリストです。
デフォルトでは、代表的なイベントプロパティがデータベースの列に割り当てられています。しかし、この値は、必要に応じて変更できます。

フィールド名（Fieldname） は、データベースの列の名称です。
テーブル内には、どんなフィールドでも設定できます。デフォルトとして設定されているフィールド名は、Adiscon のデータベース スキーマで使用されるものです。必要ならば、新たにフィールドを追加することもできます。

フィールドタイプ（Fieldtype） は、データベースの列のデータ型です。
それは、データベースで選ばれる列の型を反映しなければなりません。また、この型は、実際に保存されるプロパティとの間に整合性が取れていなければなりません。例えば、syslogpriority のような整数型のプロパティは、varchar の列に保存することができますが、syslogtag のような文字列のデータ型のプロパティは、integer（整数）型の列に保存することはできません。

フィールドコンテンツ（Fieldcontent） は、イベントプロパティです。
WinSyslog で使用できるプロパティについては、下記 URL の「WinSyslog プロパティリスト」をご参照ください。
http://www.jtc-i.co.jp/support/documents/tips/winsyslog_propertylist.pdf

データベースフィールドの編集は、各行をクリックし、テキストボックスに値を入力、およびドロップダウンリストより値を選択することで行なえます。項目の挿入、削除は、それぞれのボタンをクリックすることで行なうことができます。削除の場合には、選択されている行が削除されます。
また、行を選択して、↑・↓のボタンをクリックすると、上・下へ移動することもできますが、この移動は、表面的なものですので、データベースアクションの書き込みには作用しません。

文字列のデータ型には、プロパティの置換を使用できます。これは、サブストリングを保存したい場合に、特に役立ちます。例えば、各メッセージの先頭から 200 文字を保存したい場合には、"%msg:1:200%" という値を使用します。

■ OleDB 接続の設定（データソースの設定）
ここをクリックすると、OS の OLEDB の設定画面が表示され、データソースの追加、削除、編集などを行なうことができます。

■ データベースを確認（データベースアクセスの確認）
ここでは、指定したデータソースが問題なく動作するかどうかをチェックします。

■ SQL 接続のタイムアウト
接続のタイムアウトを設定します。

■ テーブル名（メインテーブル名）
ログを書き込むテーブル名。この名前は、SQL インサート文の作成に使用されます。また、この名前は、データベース定義にマッチしている必要があります。デフォルトは、「SystemEvents」です。

■ ステートメントタイプ（SQL ステートメントタイプ）
MSSQL のストアドプロシージャ（Stored Procedures）の INSERT、または Call Statement（MSSQL ストアドプロシージャ）のいずれかを選択できるようになりました。この機能は、MSSQL をデータベースソフトとしてご利用になる場合のみ有効です。
また、Call Statement を選択した場合、テーブル名が自動的にプロシージャ名として使用されます。正しくパラメーターがソートされるようにして下さい。そうでないと、アクションが正常に動作しません。また、その場合、この結果はデバッグログで比較しないと分かりません。

■ 出力エンコード

ここでは、データベースへ書き込みを行う際に使用する文字コードを設定します。

■ 詳細なプロパティのログを有効にする

このオプションは、SystemEventProperties テーブルに標準のプロパティ以外のイベントプロパティを記録します。場合によっては、一つのイベントに複数のプロパティがある可能性があるので、このオプションを選択することで、複数のプロパティを書き込むことが可能となります。しかし、Syslog データでは、追加のプロパティはあまり存在しません。(「Post Property」アクションで独自の定義をしている場合を除いては)

追加のプロパティは、概して イベントログの監視からの SETP 受信データにあります。例えば、SETP で受信したイベントログのデータには、実際の Windows のイベントプロパティとイベントデータが含まれています。このオプションは、Syslog で受信したイベントログ メッセージには適用されませんので、注意してください。

このオプションは、有効にする前に、必要がどうかを確認するようにして下さい。

(このオプションのデータは、MonitorWare コンソールでは必要となる場合があります。)

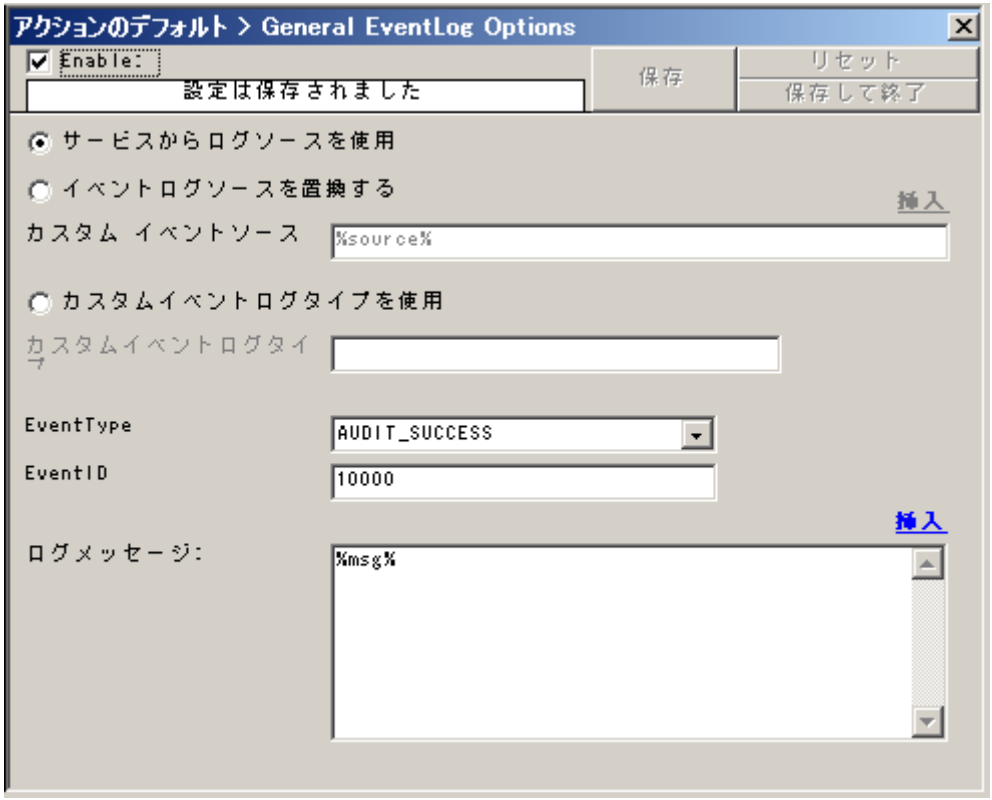
■ 接続の再試行

接続が切れると、WinSyslog は DB 接続をシャットダウンし、次のアクションで接続の再試行を行ないます。

5.5.5 イベントログ記録

ここでは、WinSyslog サービスで処理されるメッセージを Windows のイベントログに記録する設定を行います。

▲新クライアント「イベントログ」



▲旧クライアント「イベントログ」

■ ソースにサービス名を使用（サービスからログソースを使用）

イベントログに書き込む際のログソース名にサービス名を使用したい場合は、ここを有効にします。

■ イベントログのソース名を変更（イベントログソースを置換する）

ここを有効にすると、入力したプロパティ値がカスタムイベントソースとして設定されます。

ここで %source% と設定を行うと、Windows イベントソースは syslog メッセージを送っているシステムの IP アドレスに設定されます。さらに、Syslog ファシリティがイベント ID に設定されます。

しかし、このモードには欠点があります。

上記のように IP アドレスをイベントソースとした場合、それは登録されていないので、イベントビューアは、メッセージ・ライブラリを見つけることができないと以下のメッセージをユーザーに警告をします。

（Windows 2000 の例）

イベント ID (16) (ソース 192.168.1.1 内) に関する説明が見つかりませんでした。

リモート コンピューターからメッセージを表示するために必要なレジストリ情報またはメッセージ DLL ファイルがローカル コンピューターにない可能性があります。次の情報はイベントの一部です:

しかし、その場合でもログメッセージは全て表示されます。

これは、詳細表示においてのみ起こります。

（この問題は、アプリケーションの動作には影響しません。）

■ カスタム イベントソース

ここで「挿入」をクリックすることで イベントログソースをカスタマイズすることができます。
デフォルトでは、「%source%」（ソース）のみ設定されています。
この機能は、「イベントログソースを置換する」機能を有効にしている場合のみ、ご利用頂けます。
ここで挿入可能なプロパティにつきましては、下記 URL の「WinSyslog プロパティリスト」をご参照下さい。
http://www.jtc-i.co.jp/support/documents/tips/winsyslog_propertylist.pdf

■ EventType

このログと共に書き込まれるタイプ―または重要度を設定します。
Windows システムが利用できる値から選択して下さい。

■ イベント ID（EventID）

この ID はイベントログが書き込まれる際に使用されます。
他のプロセスに特定のメッセージに対して整合性のあるインターフェイスを与えるために、違う ID を使用することができます。WinSyslog は、使用する ID を制限しません。
けれども、 もしもオペレーティングシステムで登録されていない ID が書き込まれると、Windows イベントビューアには、実際のメッセージ・テキストより先に未登録を指摘するエラー・メッセージが出ます。
このエラーを避けるために、OS では 10,000 から 10,100 の ID が設けられています。
ですので、カスタマイズした全てのメッセージにはこれらの ID を使用することをお勧めします。

注意： 10,000 以下の ID は、WinSyslog によって生成されるイベントと衝突する可能性があるので、使用すべきではありません。（デフォルトは、10000 です。）

■ 出力メッセージ（ログメッセージ）

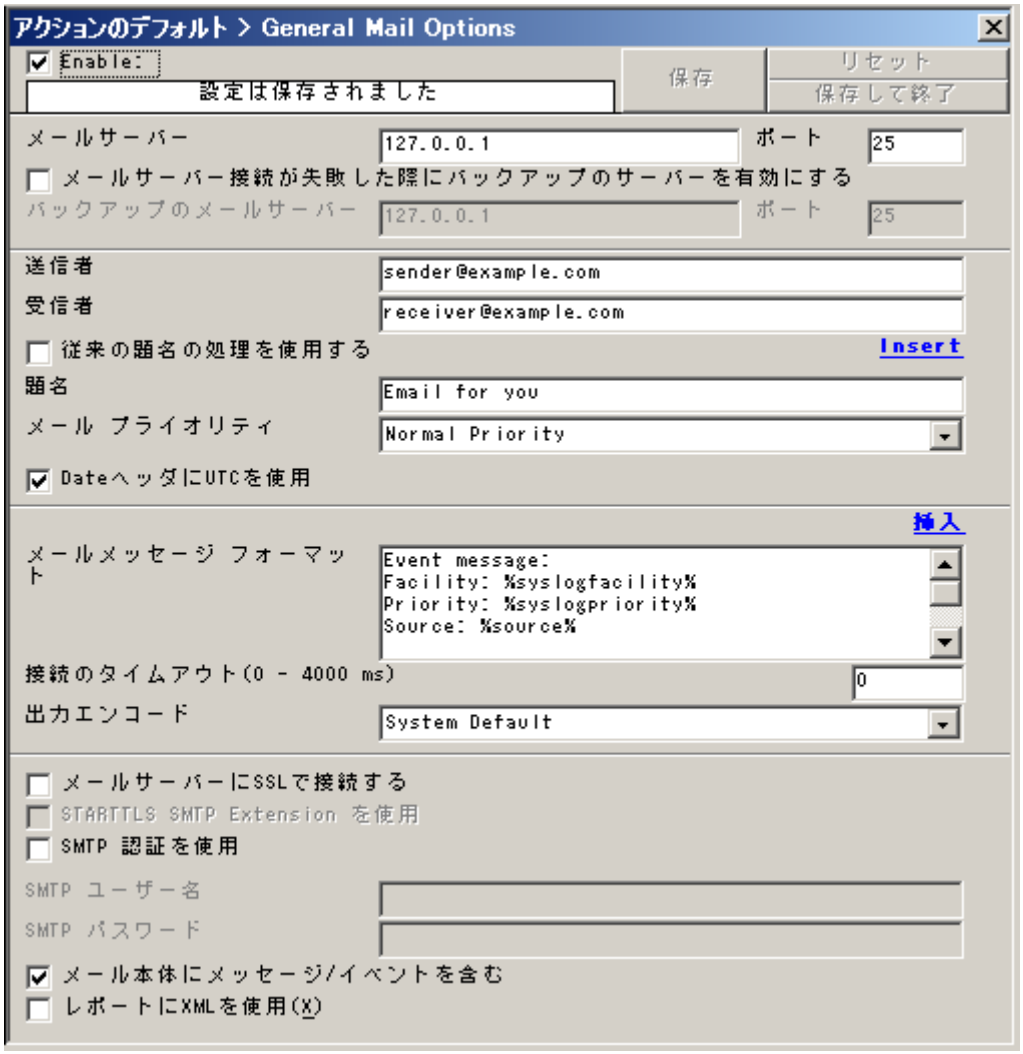
Windows のイベントログに書き込まれるメッセージを設定できます。
「挿入」をクリックし、%msg% （例） などの置換文字を追加することで、イベントビューアに書き込まれるイベントログのメッセージをカスタマイズすることが可能となります。
ここで挿入可能なイベントのプロパティにつきましては、下記 URL の「WinSyslog プロパティリスト」をご参照下さい。
http://www.jtc-i.co.jp/support/documents/tips/winsyslog_propertylist.pdf

5.5.6 E メール送信

WinSyslog で受信したメッセージをメール送信する際の設定を行います。
メッセージを電子メールで送信するには、このオプションで正しく設定を行う必要があります。

▲新クライアント「メール送信」-メールサーバー オプション

▲新クライアント「メール送信」-メールフォーマット オプション



▲旧クライアント「メール送信」

＜メールサーバー オプション＞

■ メールサーバー

メッセージの転送の際に使用するメールサーバー名か IP アドレスを指定します。

ここでは、受信者にメール配送ができるサーバーを設定するようにしてください。

WinSyslog は、標準の SMTP メールサーバーとの接続を前提としています。

■ ポート番号

メール送信に使用するポート番号を指定します。デフォルトは 25 です。

■ メインのメールサーバーに接続できない時、下記のサーバーを使用（バックアップ メールサーバー）

この機能を有効にすると、メインのメールサーバーへの接続に失敗した場合に、バックアップとして設定された別のサーバーへの接続されるようになります。

バックアップのメールサーバーは、そのセッションの間使用されます。
どれくらいのメールがバックアップサーバで処理されるかは、セッションがクローズされるまでに実行された E メールアクションの数に依存します。
このセッションがクローズされると、その後は、再びメインのサーバーへの接続が試みられます。
（ここでも接続が確立できない場合には、再度バックアップのサーバーが使用されます）

このバックアップのサーバーへの接続にも失敗した場合に、エラーが作成されます。

■ SMTP 認証を使用

ここは、サーバーで SMTP 認証が必要な場合に有効にします。
多くのサーバー管理者は、SPAM 対策のために 認証されたユーザー以外の接続は許可していません。
サーバーが匿名の投稿(anonymous posting)を許可しないよう再設定されると、それにより既存のアカウントが使用できなくなる可能性もあります。

サーバーが SMTP 認証を必要（またはサポート）する場合は、この設定を有効にして、下のテキストボックスにユーザーID とパスワードを入力して下さい。

メールサーバーが認証をサポートしていない場合は、ここは有効にしないで置いて下さい。

ここ設定は、認証に対応している場合は、有効することをお勧めします。
たとえ、現在のサーバーの設定が認証されていない接続を許可していても、（SPAM の問題が拡大すると）将来的には状況が変わってゆく可能性が十分にあります。
もしも、すでに認証を使用している場合は、そのようなサーバーの設定変更は何も影響を与えません。ですが、認証を使用していない場合は、メールサービスが止まってしまう恐れがあります。

■ セッションタイムアウト（接続のタイムアウト）

このオプションは、どんどん入って来る多数のメッセージを一つの E メールのメッセージとして統合するべきかを制御します。
指示されたタイムアウトに達するまでは、サーバーの SMTP セッションは開かれたままになっています。ここでは、秒単位でなく**ミリ秒単位**で値を入力してください。

新しいイベントが設定されたタイムアウトに達するまでに受信された場合は、前のイベントと同じ E メールメッセージに含まれます。それからタイムアウトは再起動されます。
このように、接続のタイムアウトの時間内に受信したイベントはいずれも一つのメールにまとめられます。
これは、メッセージの大量なバーストが予想され、それらのメッセージが少数の E メールにまとめられるべきである場合に最も適切です。さもなければ、管理者のメールボックスは多数のメールにより、すぐにオーバーフローしてしまいます。

接続のタイムアウトでは 0 から 4000 までの値を入力できます。
4000 より大きな値は、SMTP サーバ・パフォーマンスに影響を及ぼしてしまうなど、予測できない結果につながる可能性もあるので、サポートしていません。
接続のタイムアウトにおいて 0 を設定した場合は、各イベントが個別のメッセージとして送信されます。

■ メールサーバーに SSL で接続する

SMTP over SSL が利用可能となりました。
この機能を利用する場合には、465 番ポートを使用してください。
また、この機能を有効にした状態で SSL 非対応の SMTP サーバーへメール送信を行った場 合には、アクションは失敗してしまいます。(メールは届きません)

■ STARTTLS SMTP Extension を使用

STARTTLS SMTP 拡張機能(暗号化)を有効にします。

■ Date ヘッダに UTC を使用

WinSyslog が通知するメールのヘッダの基準時刻を設定します。
デフォルトでは、ここは有効になっておりますので、Date ヘッダには UTC タイムの時刻が入ります。
UTC タイムに対応していないメールソフトをご利用の場合には、この機能を無効にしてください。

＜メールフォーマット オプション＞

■ メール送信元(送信者)

メッセージの送信者の E メールアドレスを指定します。
SMTP サーバーが受信できる、有効なアドレスを指定して下さい。

■ メール送信先(受信者)

受信者の電子メールアドレスを指定します。
複数の宛先へメール送信したい場合には、このフィールドにすべての電子メールアドレスを入力して下さい。それぞれのアドレスは、スペース、セミコロンまたはコンマにて区切って下さい。

■ メールタイトルにレガシーの変数を使用(従来の題名の処理を有効にする)

ここでは、メールタイトルの処理の方法を指定します。
ここを有効にした場合には、以下の置換文字列を題名に組み込むことができます:
(無効の場合には、%source% 等のプロパティが利用できます。)

%s	メッセージを送信したソースシステムの IP アドレス、もしくはホスト名（「ホスト名の解決」の設定に左右されます）
%f	受信したメッセージのファシリティコードの数値
%p	受信したメッセージのプライオリティコードの数値
%m	メッセージ本体 注意: これは完全なメッセージのテキストなので、かなりの長さになる場合もあります。 従って、255 文字を超えた場合には以下が切り捨てられます。このような場合、%m 以降の情報が省略されてしまいます。 このような理由から、私達は%m という置換文字列を題名の最後で使用することをお勧めします。
%%	ひとつの%文字列へ変換されます。

上図のような置換文字列の設定がされている場合、以下のような題名が受信できます。
題名のテキストボックスに「Event from %s:%m」と設定した場合には、下記のような題名でメールが送信されます。
（これは、“172.16.0.1”から“This is a test” という内容のメッセージを受信したときの題名です）:

Syslog from 172.16.0.1: This is a test

上記の置換文字列とは別に、プロパティを組み込み、修正するという方法もあります。
例えば、以下のような題名を指定することが可能です。
「Mesg : '%msg:1:15'From:%fromhost%"」と設定し、「This is a lengthy test message」というメッセージを 172.16.0.1 から受信すると、下記のような題名になります :

Mesg : 'This is a lengt' From:"172.16.0.1"

メッセージは、%msg:1:15 にて文字列の 1 文字目から 15 文字目までと指定されていますので、16 字目以降（「hy」の 2 文字）は切り捨てられます。

■ メールタイトル（題名）

送信メールの題名を指定します。この題名は、各メッセージの送信に使用されます。
イベントの詳細を表示するためにプロパティを組み込む事も可能です。
この機能は、題名でメッセージの内容が判断できるので、携帯電話などのモバイル機で受信する際に特に役立

ちます。置換文字列の変換後、題名の最大値は 255 文字です。
それ以上の文字は切り捨てられます。ですが、メールシステムの制限により 255 文字以下で題名を切り捨てる可能性もあるという点に注意してください。
そのため、モバイル機で受信する場合には、題名は 80 文字以下になるように設定することをお奨めします。

メール本体には完全な情報（ソースシステム、ファシリティ、プライオリティ、メッセージ・テキスト）が含まれています。メッセージ本体のサイズには制限がないので、常に完全なメッセージを受信できます。

「挿入」をクリックし、%msg% （例） などの置換文字を追加することで、題名に書き込まれるイベントのメッセージをカスタマイズすることが可能となります。
ここで挿入可能なプロパティにつきましては、下記 URL の「WinSyslog プロパティリスト」をご参照下さい。
http://www.jtc-i.co.jp/support/documents/tips/winsyslog_propertylist.pdf

受け取ったひとつのメッセージに対して、一通のメールを送信します。
E メール送信は、重要な通知やアクションにたいして意味を持ちます。（重要なイベントの発生時に携帯メールに送信して、知らせるなど･･･）
E メールの報告を提供する事自体には、あまり重要な意味はありません。

■ メールの優先度（ メールプライオリティ）
ここでは、送信するメールの優先度を設定することができます。「low」、「normal」、「high」のいずれかを選択できます。

■ メール本文（メールメッセージ フォーマット）
ここでは、メールメッセージ本体のフォーマットを設定します。
「挿入」をクリックし、イベントのプロパティを組み込むことができます。
注意： このオプションは、「メール本体にメッセージ/イベントを含む」を有効にした場合のみ、使用することが可能です。

■ 出力エンコード
ここでは、メール送信を行う際に使用する文字コードを設定します。

■ メール本文にメッセージ/イベントを含む
syslog メッセージをメッセージ本文に 出力するかどうかを設定します。
ここが無効ならば、メール本文に Syslog メッセージは含まれません。

このオプションは携帯などのモバイル機器で（WML 対応の場合は特に）、非常に役に立つ機能です。これらのデバイスは、たいてい表示するデータの量に制限があります。

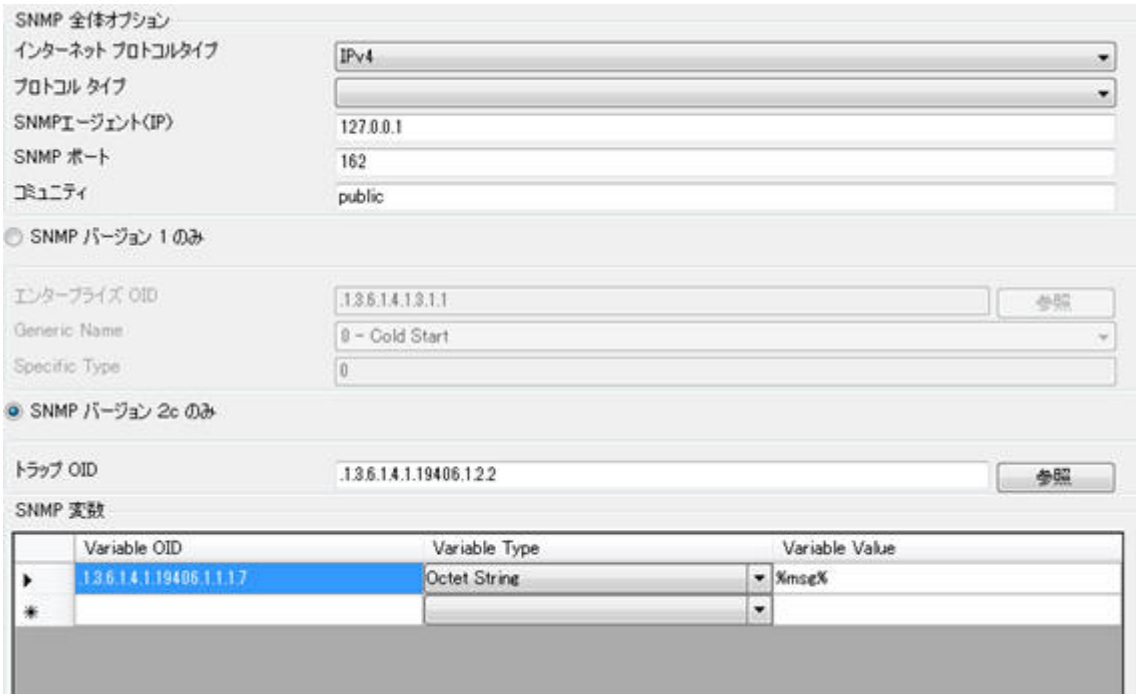
メッセージ自体は表示しないものもあります。
したがって、このオプションはメッセージ本体を送信したときに限って意味をなします。
そのため、必要がない場合は、ここでオプションをオフにできます。
オフにした場合は、題名に適切な置換文字を使用してください。

WML に対応した機器がメッセージ本体を受信できる場合であっても、このオプションをオフにした方が良い場合もあります。WML や WAP は比較的コストがかかります。
生成されたメッセージは、長くなってしまう恐れがあります（メッセージソースに左右されますが）。
したがって、このオプションの機能をオフにすることも適当である場合もあります。

■ メッセージに XML を出力（レポートに XML を使用）
ここを有効にすると、受信したイベントは XML フォーマットでメールに含まれます。
その場合、オリジナルのタイムスタンプやファシリティ、プライオリティなどの全ての情報が含まれます。メールがメッセージを解析する自動化されたシステムに送られる場合、XML フォーマットは特に役に立ちます。
チェックしない場合は、メールにはプレーンテキスト・メッセージが含まれます。

5.5.7 SNMP トラップの送信

ここでは、SNMP トラップの送信に関する設定を行います。



▲新クライアント「SNMP トラップの送信」

▲旧クライアント「SNMP トラップの送信」

■ インターネットプロトコルタイプ(IP タイプ)

送信時の IP タイプを指定します。IPv4、IPv6 のどちらかを選択します。

■ プロトコルタイプ

ここでは、UDP または TCP を指定します。

■ SNMP バージョン(旧クライアント)

ここでは、SNMP のバージョンを指定します。

■ SNMP エージェント(エージェント受信機)

SNMP トラップを受信するサーバーを指定します。ここでは、転送先 IP アドレスを指定します。

■ SNMP ポート

ここでは、送信するポートを指定します。実際に使用する値については、サーバーリファレンスを参照下さい。

■ コミュニティ

メッセージの属する SNMP コミュニティを指定します。

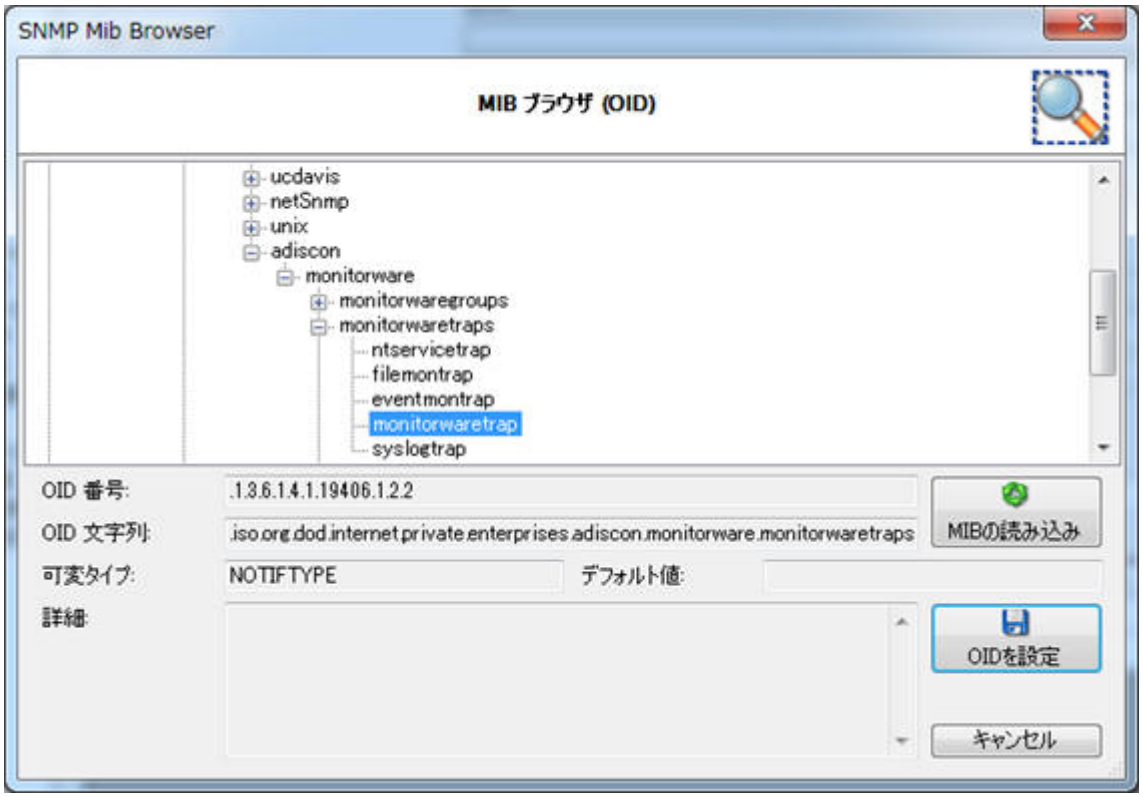
SNMP バージョン 1 のみ（SNMP　V1 特定パラメーター ）

このグループボックスでは、SNMP バージョン 1 に関するパラメーターを確認できます。

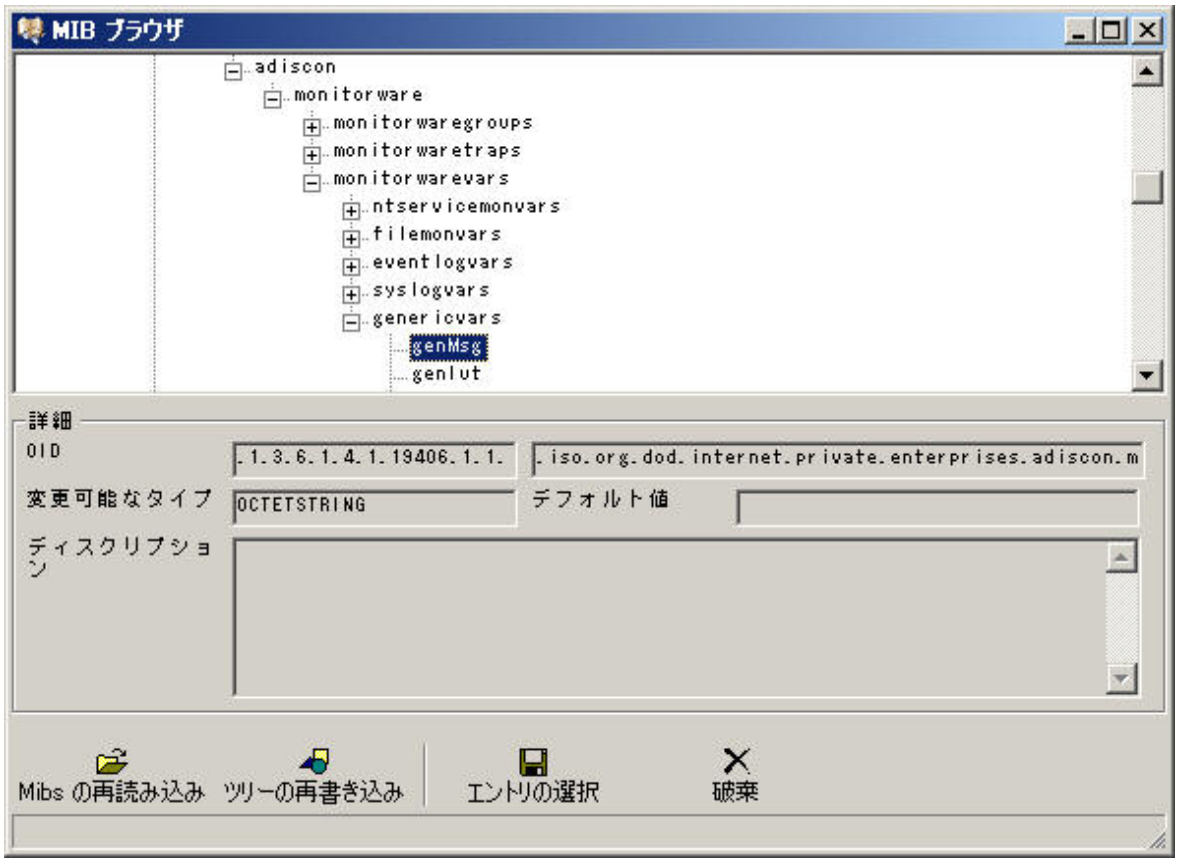
■ エンタープライズ OID

ここでは、エンタープライズ OID を指定します。

OID の選択には、「参照（ブラウザ ）」オプションを使用できます。ここをクリックすると下記画面が現れます。



▲新クライアント「MIB ブラウザ」



▲旧クライアント「MIB ブラウザ」

■ Generic Name（一般名）

coldStart(0), warmStart(1), linkDown(2), linkUp(3), authenticationFailure(4), egpNeighborLoss(5) や enterpriseSpecific(6) のようなトラップの Generic Name を指定することができます。

■ Specific Type（特定タイプ）

トラップの追加コードを定義できます。こちらも同様に整数値です。

SNMP 変数

SNMP トラップに送信する変数。トラップコードをご存知の場合、それらを入力することも可能です。
そうでない場合は、SNMP MIB ブラウザをご利用下さい。

■ Variable OID（変更可能な OID）

SNMP トラップの OID。利用できる OID については、MIB ブラウザのリストをご利用下さい。

■ Variable Type（変更可能なタイプ）

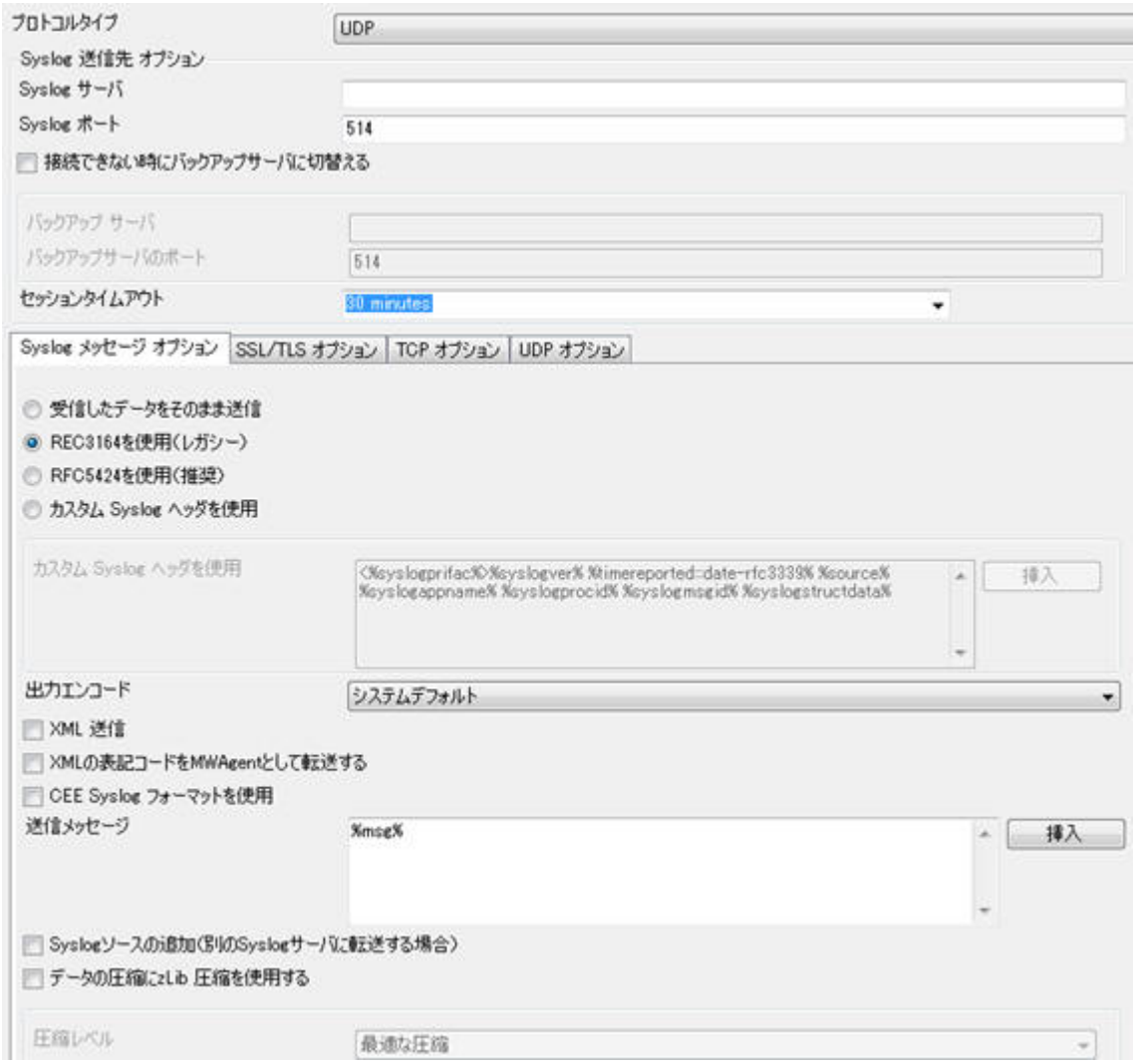
OCTETSTRING や INTEGER などのタイプ。
このタイプによっては、変数を正確にフォーマットする必要があります。（IPADDR など ）

■ Variable Value（変更可能な値）

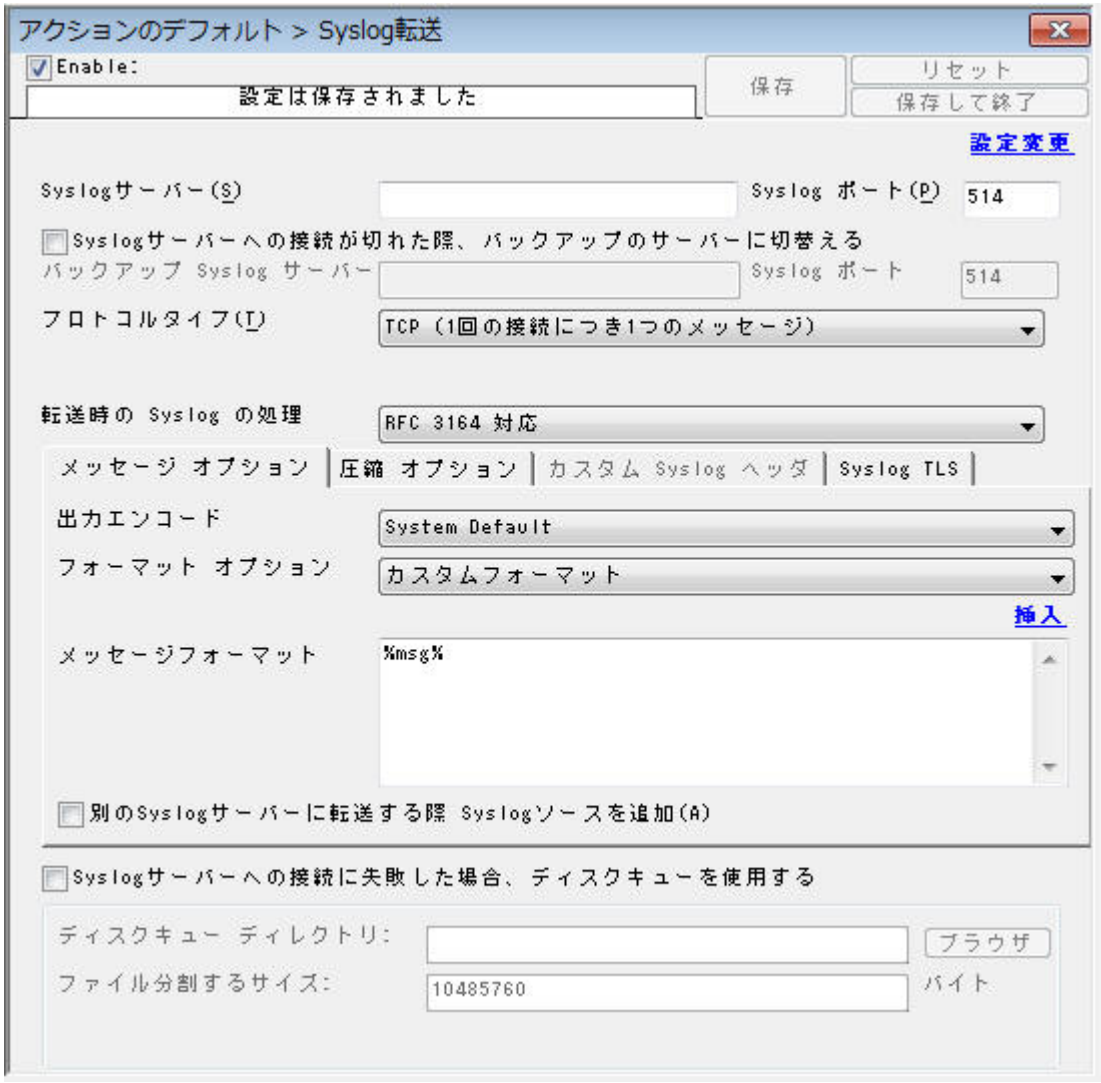
タイプによってフォーマットされる必要があります。

5.5.8 Syslog 転送

ここでは、受信したメッセージを別の Syslog サーバーへ転送するための設定を行います。



▲新クライアント「Syslog 転送」



▲旧クライアント「Syslog 転送」 TCP を選択した際の画面

■ プロトコルタイプ

syslog メッセージの転送方法としては、UDP、TCP または RFC 3195 RAW が使用可能です。

デフォルトは、UDP です。
UDP はほとんどすべてのサーバーで使用できますが、ネットワークエラーなどによりメッセージが消失してしまう可能性があります。その性質を理解された上、ご利用ください。

TCP と RFC 3195 をベースにしたメッセージは、UDP よりも確実に送信されます。
RFC3195 は、特殊な通信モードです。このモードは必要でない限り利用しないことをお勧めします。

TCP には、**「TCP(1 回の接続につき 1 つのメッセージ)」**、**「TCP(持続して接続)」**、**「TCP(オクテットベースのフレーミング)」** の 3 つのモードがあります。

「TCP（1 回の接続につき 1 つのメッセージ）」は、2006 年以前の Adiscon サーバーのための互換モードです。（ほかのベンダーでも要求されるモードかもしれません）

「TCP（持続して接続）」は、1 度の接続で複数のメッセージを送信するモードです。（メッセージを送信し終わるまでポートは開いた状態になります）高いパフォーマンス性が期待できますが、Syslog メッセージ内に制御文字などが含まれる場合には、問題が起こる可能性があります。

「TCP（オクテットベースのフレーミング）」は、やがて公開される（未確定ですが）IETF 標準のアルゴリズムを実装しています。このモードも継続的に接続されます。このモードでは、制御文字が含まれるメッセージも処理されます。しかし、このモードに対応したレシーバーは、現在ところ非常に数少ない状況です。そのため、このモードは、最新の Adiscon 製品間の通信でご利用になることをお勧め致します。

「TCP（持続して接続）」、「TCP（オクテットベースのフレーミング）」を選択した場合には、セッションタイムアウトの設定が行えます。デフォルト（30分）の場合、メッセージが送信されないまま 30 分経過すると、接続が切られるようになります。もしも、処理するメッセージが少ない場合には、これより低い値を設定しても結構です。

■ Syslog サーバー

Syslog メッセージを送信する相手先システムの名前または IP アドレスを指定します。
IPv4 と IPv6 のどちらにも対応しております。

■ Syslog ポート

syslog 転送する際のポート番号を指定します。
確信がない場合には、一般的に使用されるデフォルトポート：514 のままにして置いてください。
別のポートは、例えばセキュリティへの配慮が必要な場合などに使用されます。

■ 接続できない時にバックアップサーバに切替える

（Syslog サーバーへの接続が切れた際、バックアップのサーバーに切替える ）
ここを有効にすると、Syslog サーバーへの接続が確立できない時に、設定したバックアップのサーバーの IP アドレス、ポート番号のサーバーに自動的に切替わります。
このオプションは TCP 通信の場合のみ使用可能なので、ご注意ください。

＜Syslog メッセージオプション＞

（転送時の Syslog の処理）

Syslog メッセージの処理方法を下記 4 つのオプションから選択します。

- **RFC3164(従来の処理)**
- **RFC5424（推奨）**

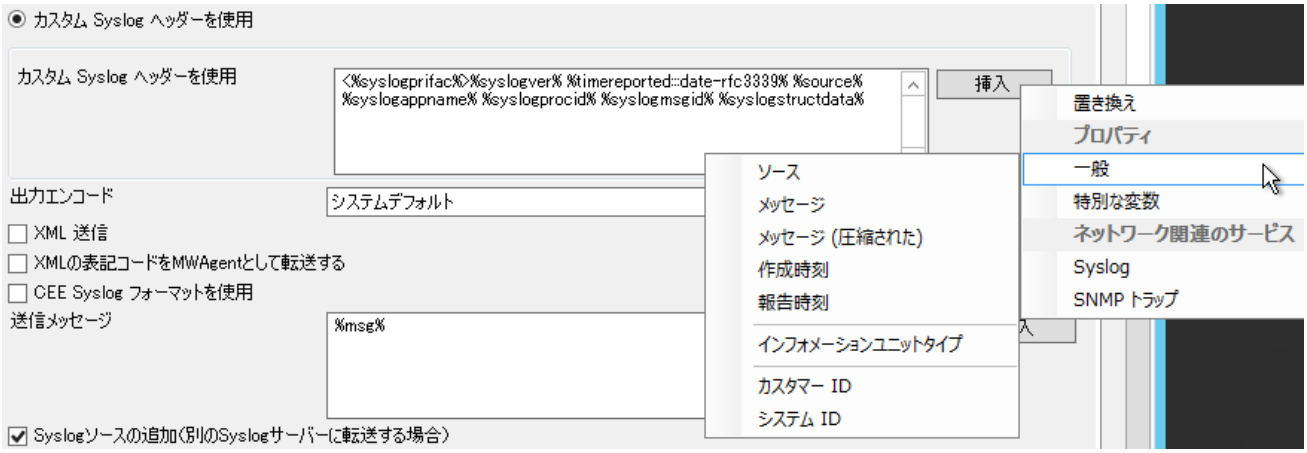
・受信したメッセージを処理せずそのまま送信（加工しない）

・カスタム Syslog ヘッダーを使用（ヘッダの内容をカスタマイズ）

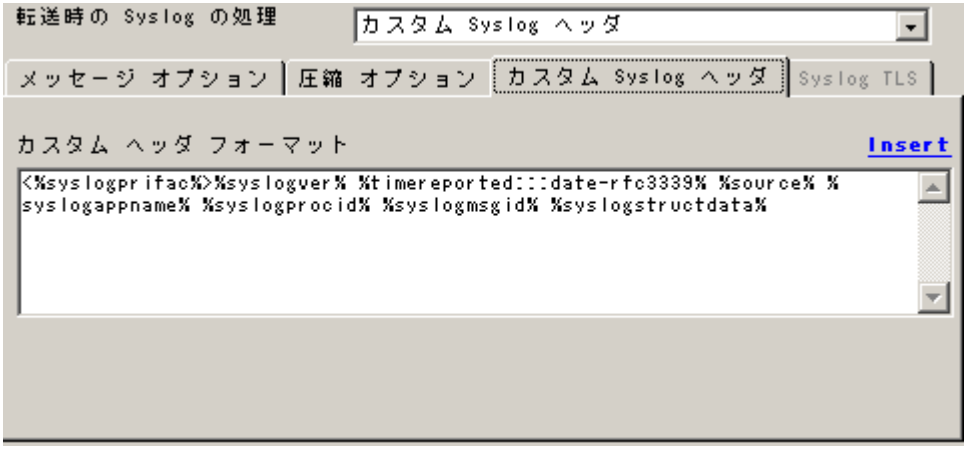
カスタム Syslog ヘッダの使用を有効にすると、ヘッダ部分をカスタマイズすることができます。
デフォルトで設定されているのは、RFC5424 のヘッダです。

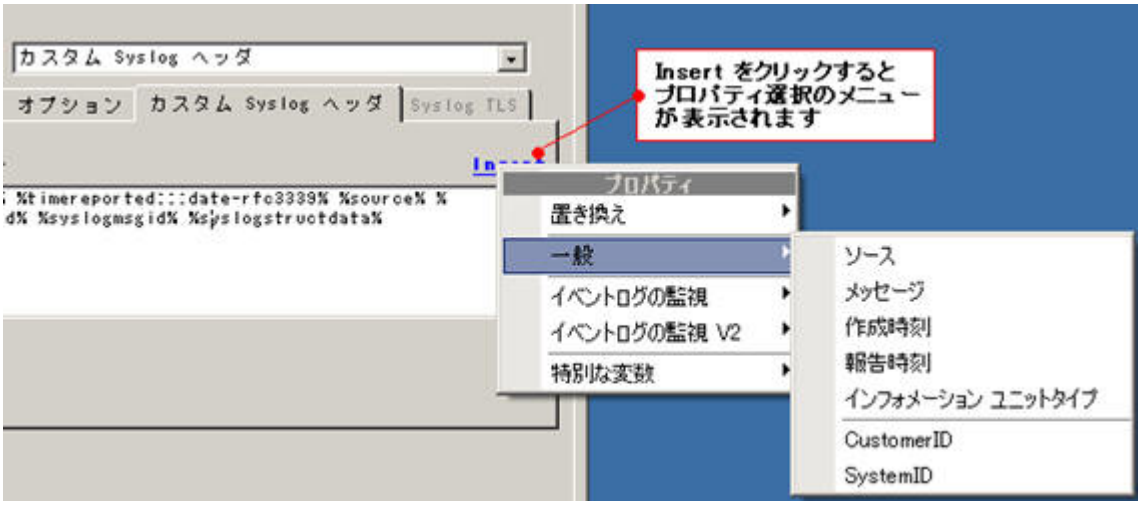
＜カスタム Syslog ヘッダーを使用＞（新クライアント）

「挿入」をクリックすると、プロパティを入力することができます。



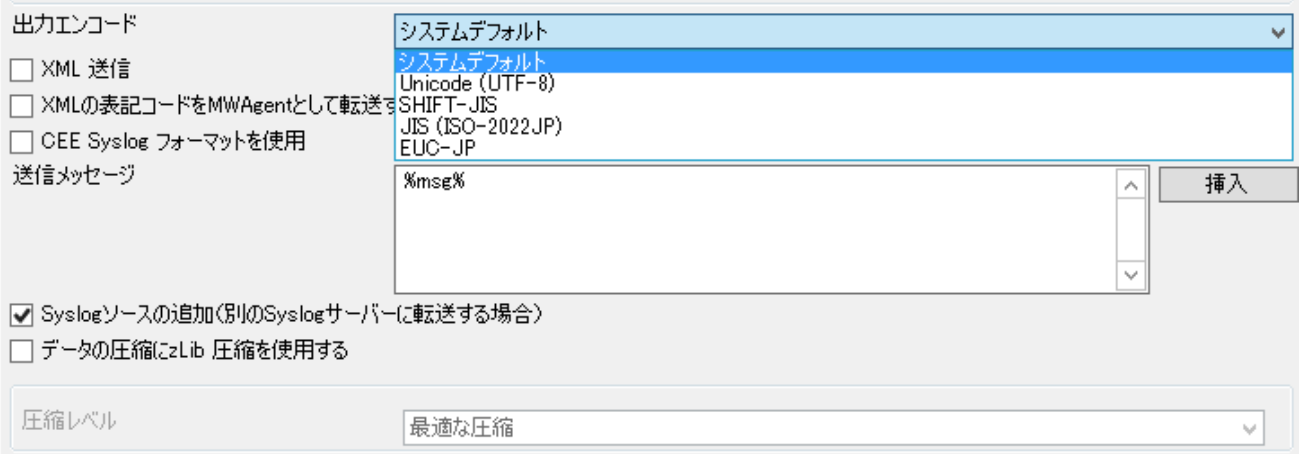
＜カスタム Syslog ヘッダ＞（旧クライアント）



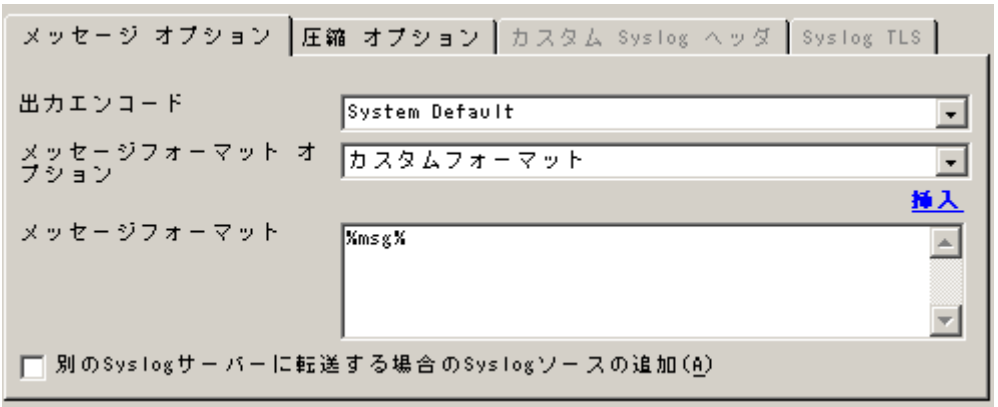


■ 出力エンコード

Syslog メッセージ転送の際に使用する文字コードを設定します。



▲新クライアント「メッセージオプション（出力エンコード）」



▲旧クライアント「メッセージオプション」

■ **XML 送信（レポートに XML を使用）**

ここを有効にすると、転送された syslog メッセージは完全な XML フォーマットされた情報記録となります。その場合、オリジナルのタイムスタンプや発信元システムなどの情報が追加されますが、見た目には読みづらくなります。ですが、この形式は、解析を容易に行うことができるようになります。

受信するシステムが XML データの解析が可能である場合は、特にこの機能は役に立ちます。

しかし、それは別な方法で転送されることができない追加の情報を含めるので、同様にマシンだけでなく人の役にも立ちます。

■ **XML の表記コードを MWAgent として転送する**

MWAgent（MonitorWare エージェント）は、イベントの特別な XML 表記をサポートしています。ここを有効にすると、転送された syslog メッセージに XML 表記が使用されます。その場合、オリジナルのタイムスタンプや発信元システムなどの情報が追加されますが、見た目には読みづらくなります。ですが、この形式は、解析を容易に行うことができるようになります。しかし、このオプションは、実験的なものであり、正式なものではありません。

■ **CEE Syslog フォーマットを使用**

ここを有効にすると、CEE フォーマットが使用されます。

■ **送信メッセージ（カスタムフォーマット）**

メッセージ・フォーマットのボックスで転送するメッセージの編集を行います。

デフォルトは、%msg% のみ設定されています。

「挿入」をクリックして、メッセージに加えたいプロパティを追加します。

■ **Syslog ソースの追加（別の Syslog サーバに転送する場合）**

（別の Syslog サーバーに転送する場合の Syslog ソースの追加）

ここをチェックすると、発信元のシステムの情報がメッセージに追加されます。

これにより、受信者が発信元を追跡する事ができます。

注意: このオプションは、RFC 3164 と互換性がありません。 インタラクティブ Syslog ビューアにメッセージを転送することを目的とする場合には、この機能を使用することをお勧めします。

■ **データの圧縮に zLib 圧縮を使用する**

ここでは、Syslog メッセージの圧縮のレベルを設定できます。

＜ Syslog メッセージの圧縮に関して ＞

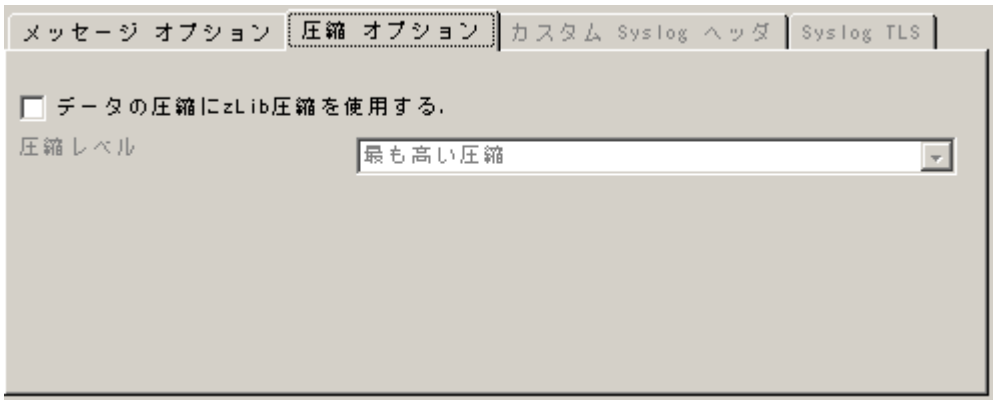
この機能は、フォーマットを十分に理解されている方（受信者）に対してのみご利用下さい。

圧縮を有効にすると、低帯域幅の環境でも帯域幅を節約することができます。ただし、どれだけ節約できるかは、メッセージによります。（全く節約できないケースもありますが、一方半分ほど節約できるケースもあります）

検証では、Windows イベントログを XML フォーマットで送信した場合に、50%ほど帯域幅を節約できました。

非常に小さなメッセージは、まったく圧縮されません。また、XML フォーマットでない場合、Syslog 送信はだいた

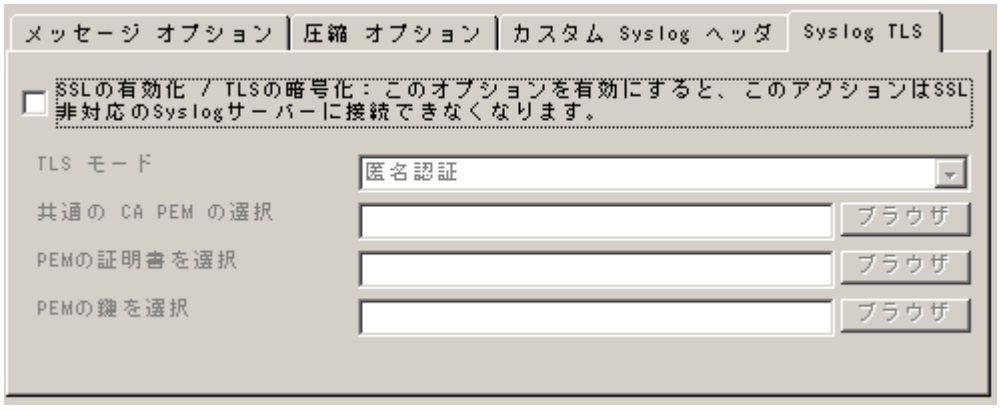
い 10～25%ほどに圧縮されます。
TCP による圧縮送信の場合には、特別なモードにする必要があります。このモード（syslog-transport-tls）は、これから公開される IETF の仕様がベースとなっています。ただ、このモードはまだ完成されたものではないので、実験的な試みとなります。その結果、今後リリースされる製品でこのモードが使用されるかどうかはわかりません。また、別のモードが今後製品に組み込まれた場合、このモードとの互換性は保証できかねます。
ただ、この機能自体はしっかりしていますが、実験的な機能であることをご理解頂いた上でご利用下さい。



▲旧クライアント「圧縮 オプション」

＜SSL/TLS オプション＞

▲新クライアント「SSL/TLS オプション」



▲旧クライアント「Syslog TLS」

■ **SSL/TLS を使用（SSL に対応していないサーバーへはアクセスできなくなります）**

（SSL の有効化／TLS の暗号化：）

ここを有効にすると、SSL に対応していないサーバーへはアクセスできなくなります。

暗号化に使用される方法は、RFC5424 （Transport Layer Security (TLS) Transport Mapping for Syslog） に互換性があります。

■ **TLS モード**

次のモードから選択できます；

・匿名認証

デフォルトでは、このモードになっています。選択するとデフォルトの証明書が使用されます。

・証明書を使用

証明書を指定することができます。OpenSSL などにより作成した証明書が必要となります。

■ **共通の CA PEM の選択**

ここでは、CA（Certificate Authority；認証局）を指定します。

Syslog を送信する側・受信する側で同じ CA を使用しなければなりません。

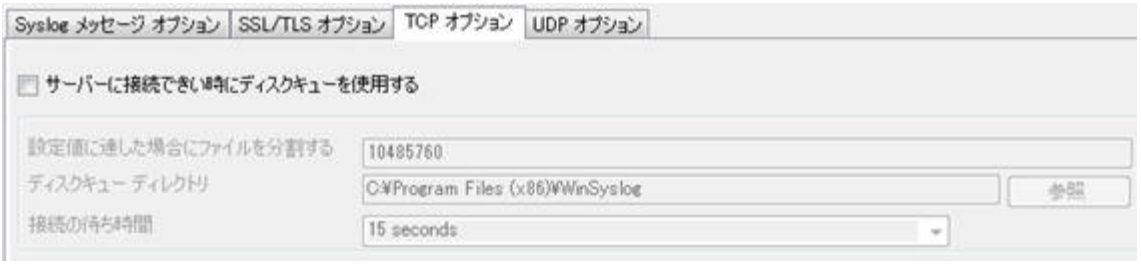
■ **PEM の証明書を選択**

クライアントの証明書（PEM フォーマット）を選択します。

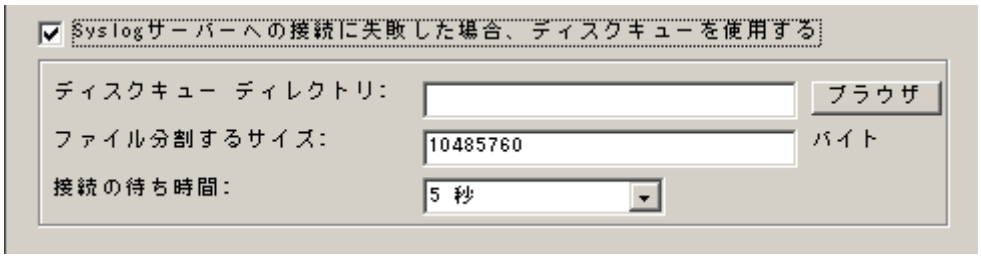
■ **PEM の鍵を選択**

クライアント証明書の鍵（キーファイル）を選択します。

＜TCP オプション＞



▲新クライアント「TCP オプション」



▲旧クライアント「ディスクキューのオプション」

■ サーバーに接続できない時にディスクキューを使用する(TCP 選択時のみ)

(Syslog サーバーへの接続に失敗した場合、ディスクキューを使用する)

この機能を利用すると、リモートの Syslog サーバーへの接続が失敗した際に、Syslog メッセージがローカルのファイルにキャッシュされるようになります。

保存先のフォルダは変更できます。Syslog 転送アクションを複数設定している場合には、そのアクション毎に GUID により個別のファイル名が生成されます。接続に失敗した Syslog サーバーへの接続が確立されると、自動的にキャッシュされたメッセージが送信されるようになります。

Syslog メッセージをキャッシュに入れている間に WinSyslog サービスを起動した場合、その起動中に Syslog サーバーが復旧しても確認ができません。(再度、アクションが実行される際に Syslog サーバーとの接続が確認され、接続されている場合にはキャッシュされたメッセージが送信されます。)

キャッシュのサイズは、特に制限はありません(ディスクのサイズに依存します)。

デフォルトにより、ファイルは 10MB ごとに分割されます。

(この値は、変更することが可能です。最大 2GB まで設定可能です)

<UDP オプション>

▲新クライアント「UDP オプション」

■ UDP で IP スプーフィングを使用

ここを有効にすると、IP スプーフィングが可能になります。

ただし、この機能は UDP で IPv4でのみ使用できるものとなっております。

また、Windows Server 2003、2008 以前のシステム、および Windows XP、Vista、7 以降のシステムではご利用になれません。(OS の制限によるものなので、詳細は Microsoft の資料等をご確認ください。)

さらに、ローカルのネットワークでしか使用できない可能性があります。

(IP アドレスがスプーフィングされた通信は、ルーターやゲートウェイを通過できない可能性があります。)

■ **IP またはプロパティを修正**

静的 IP アドレス、またはプロパティを指定することができます。

プロパティを設定した場合、プロパティの内容から IP アドレスの名前解決が実行されます。

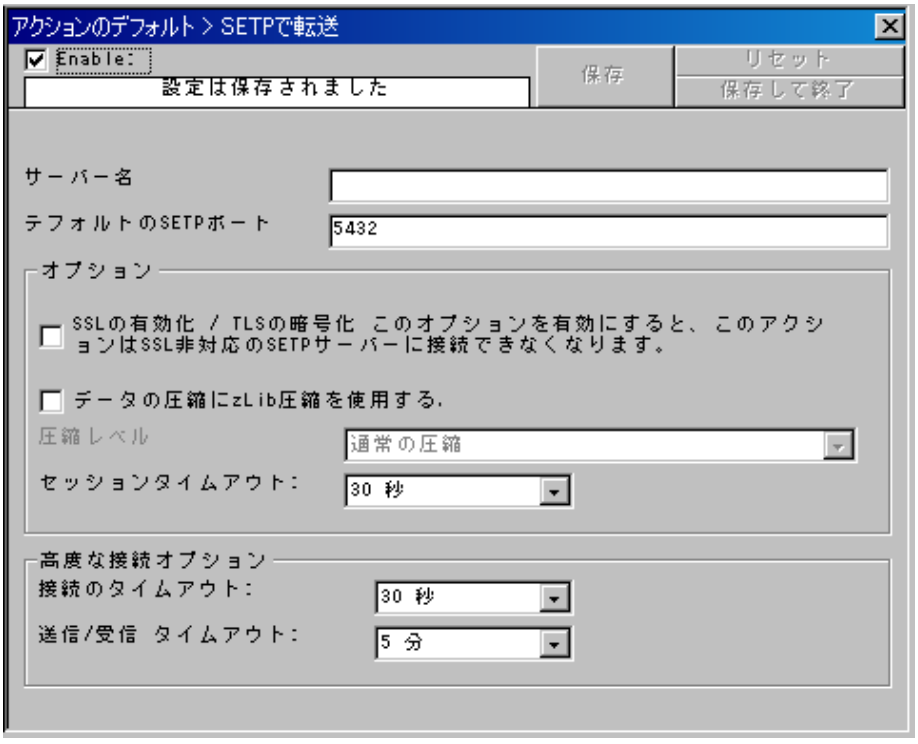
例えば、デフォルトである %source% の場合、ソース名から IP アドレスへの名前解決が行われますが、もしも、名前解決ができない場合にはデフォルトのローカル IP アドレスが使用されます。

5.5.9 SETP で転送（ベーシックエディションは対応していません）

このダイアログは、転送に関するオプションを設定します。

「SETP で転送」のアクションでは、メッセージを SETP サーバー（WinSyslog エンタープライズで作成可能なサービス）に転送する事ができます。

▲新クライアント「SETP 送信」



▲旧クライアント「SETP 送信」

■ **サーバー名**

WinSyslog は、ここで設定する名前によって SETP サーバーを認識します。

■ **SETP ポート番号（デフォルトの SETP ポート）**

SETP サーバーは、 このポートで入ってくる要求を待っています。デフォルト値は 5432 です。

注意：ここで設定される SETP ポートは、サーバ（WinSyslog エンタープライズ）で設定されるポートに合わせなければなりません。それらが合わない場合は、SETP セッションを開始する事はできません。ルールエンジンは、NT イベントログにこれを記録するでしょう。

■ **SSL/TLS を使用（SSL の有効化/TLS の暗号化）**

ここを有効にすることで、このアクションは、SSL、TLS SETP サーバーに接続することができるようになります。
ただし、ここを有効にすると、このアクションは SSL 非対応の SETP サーバーには接続できなくなります。

■ **データの圧縮に zLib 圧縮を使用する**

ここを有効にすると、zLib 圧縮が使用できます。この場合、STEP の受信側が zLib 圧縮をサポートしている必要があります。zLib 圧縮に対応していない場合には、これは機能しません。

■ **圧縮レベル**

高いレベルの圧縮が結果的に良いですが、その場合にはパフォーマンスが低下します。

■ **セッションタイムアウト**

SETP サーバーへのセッションがオープン状態である場合の最大の待ち時間を指定します。

＜高度な接続オプション＞

■ **接続のタイムアウト**

接続されるまで、または接続が切られるまでにかかる最大の待ち時間を設定します。

■ **送信/受信 タイムアウト**

データの送受信時に、ここで設定したタイムアウトが適用されます。

＜ SETP について ＞

SETP は、“Simple Event Transfer Protocol” の略です。
SETP は、MonitorWare エージェントのために Adiscon 社により開発されたプロトコルです。
Adiscon 製品（MonitorWare、WinSyslog、EventReporter など）間の通信をより確実にするよう設計されています。
特に、MonitorWare コンソールへデータを転送する際に、このプロトコルを使用した通信が役立ちます。
＊MonitorWare エージェント、MonitorWare コンソールとも弊社では扱っておりません＊

EventReporter プロフェッショナルでは**「SETP 転送」**アクションが設定できます。
WinSyslog エンタープライズでは、**「SETP 転送」**アクション、および**「SETP サーバー」**サービスを設定することが可能です。これにより、SETP を利用したメッセージ送受信が実行できます。
SETP 通信では、クライントとサーバーの間で同期通信がなされます。
SETP では、イベントは、イベントログの生成元のシステムと同じ状態で転送することができます。
また、SETP は TCP をベースにした通信プロトコルですので、より確実な通信を行えます。
（SETP は、先に送信したメッセージが受信・処理に成功してから、次のメッセージを送信します。）

5.5.10 RELP 送信（WinSyslog 10.1 で追加されたアクション）

このアクションは、Syslog 転送アクションと似ていますが、新しいプロトコル(RELP；The Reliable Event Logging Protocol)を使用するという点が異なります。
＊受信側も RELP プロトコルに対応していなければなりません。
RELP 対応のサーバーと使用することで、今まで以上に信頼できる通信を行うことが可能になります。

ただし、RELP はネットワーク経路の信頼性を高めたプロトコルですので、サービスのダウンなどローカルで発生した問題からメッセージを守ることはできませんので、ディスクキャッシュ・キューのオプションを併用するようにしてください。

■ RELP サーバー名
RELP メッセージの送信先のサーバー名、または IP アドレスを入力します。

■ RELP ポート
RELP サーバーと通信する際に使用するポート番号を指定します。
不明な場合には、デフォルト（20514）のまま使用してください。
別のポートは、例えばセキュリティへの配慮が必要な場合などに使用されます。
ポート番号の代わりにサービス名を指定することもできます。
その場合、サービス名は、ソケットサービス データベースから検索されます。

■ セッションタイムアウト
RELP サーバーへのセッションがオープン状態である場合の最大の待ち時間を指定します。

■ 送信/受信 タイムアウト

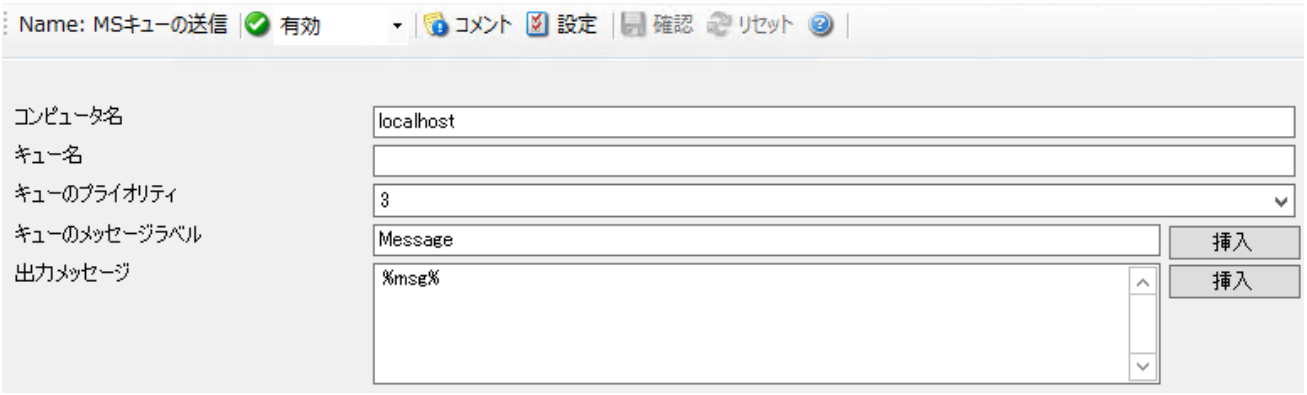
ここには、リモートサーバーの応答の最大の待ち時間を入力します。
応答がないまま設定した時間が経過すると、接続は切られ、(ルールの設定により)再試行されます。
このオプションは、なんらかの原因により、リモートのシステムとの接続が切れ、そのことを送信側が検知できない場合などに役立ちます。(例えば、ファイアーウォールの設定などにより接続できなかった場合など・・・)

■ メッセージ・フォーマット

ここでは、送信したいメッセージを入力します。デフォルトは、%msg% となっています。
ここで挿入可能なイベントのプロパティにつきましては、下記 URL の「WinSyslog プロパティリスト」をご参照下さい。
http://www.jtc-i.co.jp/support/documents/tips/winsyslog_propertylist.pdf

5.5.11 MS キューの送信

このアクションを利用すれば、メッセージキュー(MSMQ)にメッセージを送信することができます。
(MSMQ サーバーが必要です。)



▲新クライアント「MS キューの送信」

▲旧クライアント「MS キューの送信」

■ コンピューター名/IP

ここでは、問い合わせを行う MS キューの IP アドレス、またはコンピューター名を指定します。

IP アドレスは、IPv4、IPv6 のどちらにも対応しております。(IPv4、IPv6 のどちらで名前解決されたホスト名でも指定することができます)

■ キュー名

ここでは、書き込みたいキュー名を指定します。

■ キュープライオリティ

ここでは、プライオリティを設定します。

■ キューメッセージ ラベル

ここでは、キューのラベルを指定します。

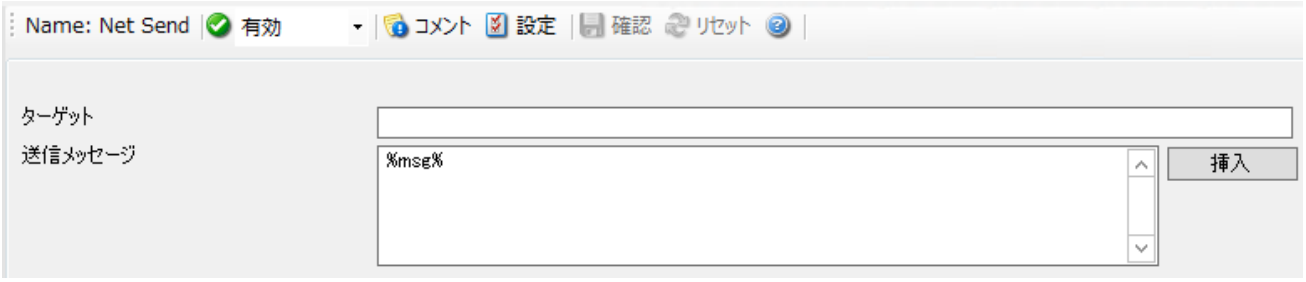
■ キュー メッセージ 本文

ここで入力したテキストがキューのメッセージとして送信されます。

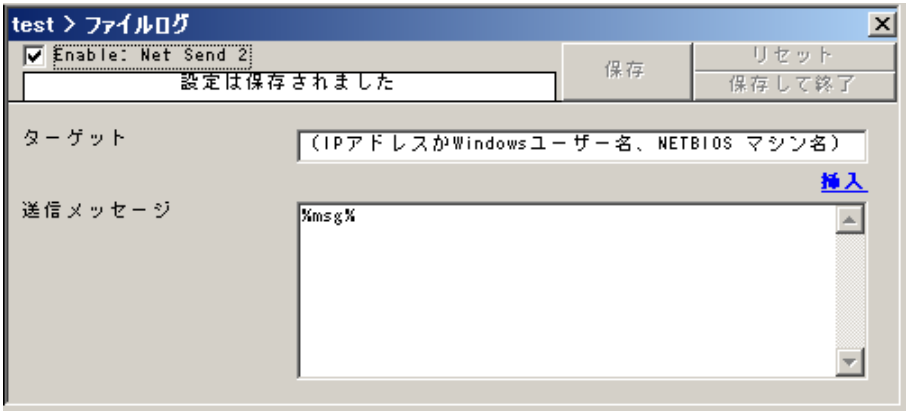
5.5.12 Net Send

ここでは、Net Send オプションを設定します。

“Net Send”アクションを使用すると、Windows の “net send” 機能を使い短い警告メッセージを送信することができます。これらのメッセージはベストエフォートをベースにして送信されます。
受信者にメッセージが届くと、それらは受信者のマシンのポップアップ メッセージボックス内に現れます。
届かない場合は、単にそれらは破棄されます。バッファリングは行われません。
従って、ルールエンジンはメッセージが届けられたかのチェックを行いません。
「net send」でのレポート送信の問題によってアクションにエラーが発生したというフラグを立てることは決してありません。



▲新クライアント「NetSend」



▲旧クライアント「Net Send」

■ ターゲット

ここでは、受信者として設定を行いたいWindows ユーザー名、NETBIOS マシン名、またはIPアドレス(10.1.1.1 のような形式で)を指定します。

■ **送信メッセージ**

設定したターゲットに対して送信するメッセージを指定します。 ここでは、メッセージの内容をプロパティで設定することも可能です。(「挿入」をクリックして、メッセージに加えたいプロパティを追加します。)

すると、以下のようなメッセージウィンドウが現れます:

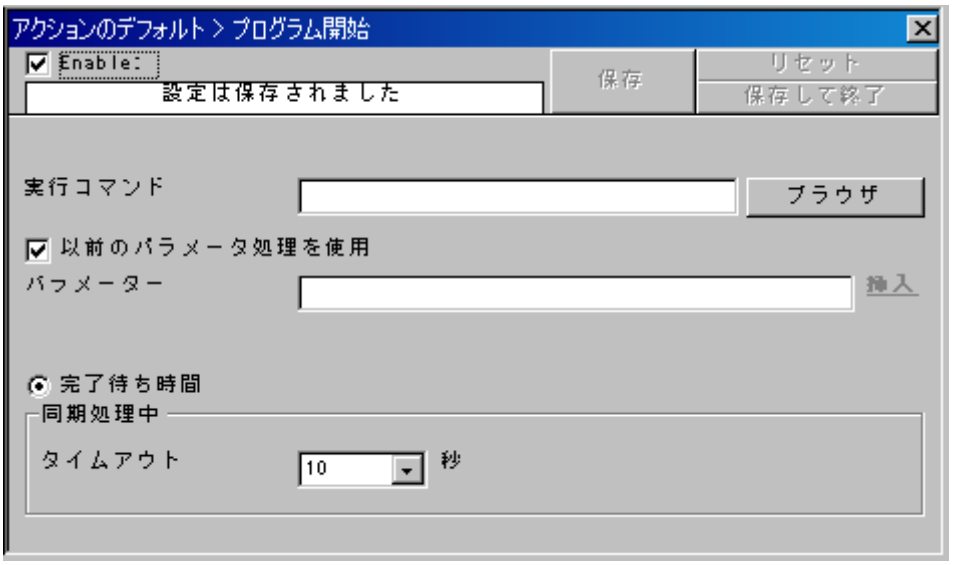
5.5.13 プログラム開始

ここでは、プログラム開始オプションを設定します。

「プログラム開始」のアクションにより、外部のプログラムを起動することができます。

exeファイルやバッチファイル(BAT)、VB スクリプト(.vbs)などの有効な Windows の実行可能プログラムであればどんなものでも起動できます。

▲新クライアント「プログラム開始」



▲旧クライアント「プログラム開始」

■ **実行コマンド**

ここでは、実行させたい実際のプログラムファイルを指定します。

有効な実行可能ファイルであればどのようなものでも指定可能です。システムの検索でファイル名を見つけられる場合、関連するファイル名を指定できます。

■ **レガシーのパラメーターを使用（以前のパラメータ処理を使用）**

ここを有効にすると、旧パラメーターの処理が使用されます。ここが無効になっている場合には、プロパティの処理が使用されます。

■ **コマンドのパラメーター（パラメーター）**

これらのパラメーターは、実行されるプログラムに渡されます。

そして、それらは、コマンド・ライン・パラメータとして渡されます。それらには特定のフォーマットはありません－それらの解析はスクリプトによって左右されます。

パラメーターは、イベントの詳細でカスタマイズするために、置換文字列を設定することも可能です。これにより、イベントデータがスクリプトに渡されます。以下の置換文字列を使用できます：

%d	日付と時間（ローカルタイム）
%s	メッセージを送信したソースシステムの IP アドレス、もしくはホスト名 （「ホスト名の解決」の設定に左右されます）
%f	受信したメッセージのファシリティコードの数値
%p	受信したメッセージのプライオリティコードの数値
%m	メッセージ本体
%%	ひとつの%文字列へ変換されます。

例として、「e1”%s””%m”」と設定し、172.16.0.1 から「This is a test.」と受信した場合は、スクリプトは 3 つのパラメーターで開始されます。

1 つ目のパラメーターは「e1」で、スクリプトに何らかの意味を持つと仮定します。2 つ目は、「%s」なので IP アドレス（172.16.0.1）に変換され、3 つ目は、「%m」なので メッセージ本体（「This is a test.」）になります。

注意： 二つの引用符（“）で置換文字列を挟むことによって、1つのパラメーターとして処理されます。引用符が抜けてしまうと、メッセージなどはバラバラになってしまいます。

置換文字列を使用する場合には、この引用符を入力することを忘れないで下さい。

■ **同期のタイムアウト（タイムアウト）**

プログラムが実行されるとき、サービスは次のアクションを行うまで、そのプログラムの終了を待ちます。この処理は、正しい順番ですべてのアクションを確実に実行するために必要になります。

外部のプログラムは、限られた時間内にのみ動作すべきです。
もし、何らかの理由でそれが妨害された場合は、それ以降の処理は実行されないでしょう。
したがって、タイムアウトの設定は必ず行わなければなりません。設定したタイムアウトの時間を過ぎてもプログラムが終了しない場合は、ルールエンジンはそれをキャンセルします。さらにアクションが失敗したというフラグが立てられ、それ以降の処理が続けられます。

重要: タイムアウトの値は最高で 30 秒まで設定はできますが、外部プログラムの実行時は、5 秒以下に制限することをお勧めします。
そうしないと、全体的なパフォーマンスに多大な影響を及ぼしてしまう可能性があります。
平均の実行時が 5 秒である場合、デフォルトの設定値 10 秒という値は、システム活動が激しいときでも、そのプログラムの終了を待つことができることを保証します。

パフォーマンスを考慮して、「プログラム開始」のアクションは、頻繁に適用されないルールに対してのみ使用することをお勧めします。

5.5.14 サウンド再生

＊Windows Vista 以降のシステムでは利用できません＊
このダイアログは、WinSyslog でメッセージが処理された際にサウンドファイルを再生させるための設定を行います。
注意: Microsoft 社により実施された仕様変更により、サービスとデスクトップとの対話（サウンドカードへのアクセスも含む）が Windows Vista 以降の OS で実行できなくなったため、それらの OS 上では「サウンド再生」のアクションは利用できません。

▲新クライアント「サウンド再生」

▲旧クライアント「サウンド再生」

注意：マシンに複数のサウンドカードがインストールされている場合、常に 最初にインストールされたカードのみが使用されます。

■ サウンドファイル名

再生させたいサウンドファイルのファイル名を選択します。このファイルは、wav 形式以外のフォーマット（MP3など）はサポートされていません。ローカルマシンのサウンドファイルのみを使用することをお奨めします。リモートでのサウンドファイルの使用は、基本的にはサポートしておりません。

存在しないサウンドファイルが指定されていたり、無効なフォーマットのファイルが指定されていたりする場合には、システムビープ音が発せられます。

■ ファイルの再生回数（ファイルを何回再生するか）

サウンドファイルを再生する回数を指定します。

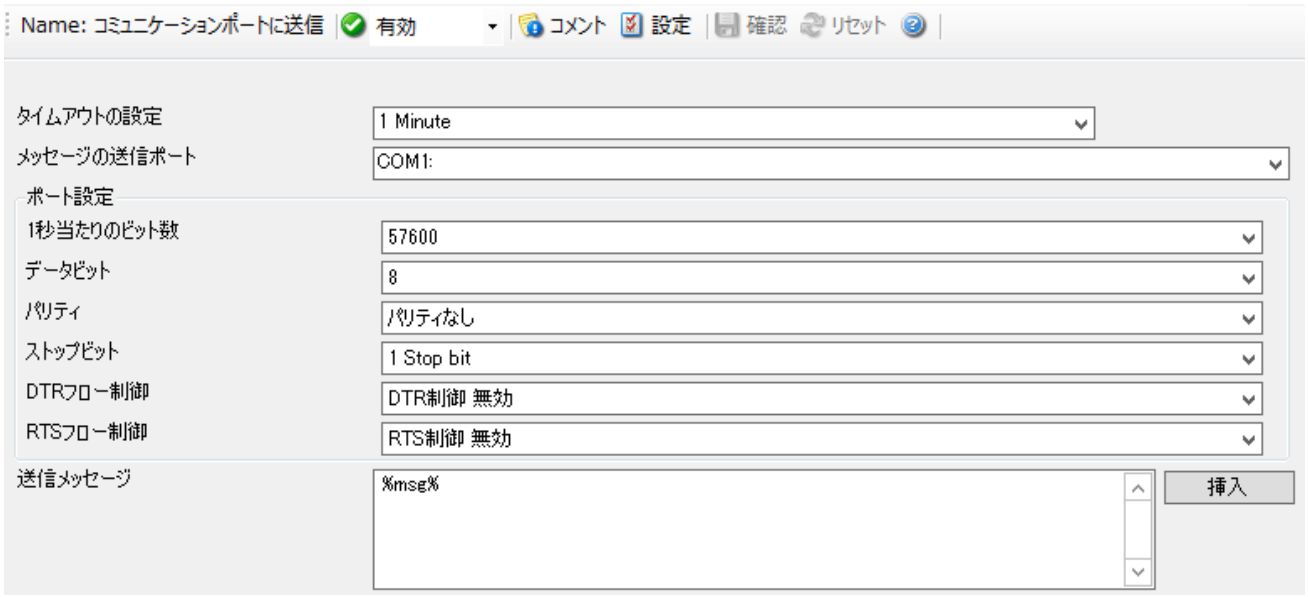
注意： 再生回数は、1 回から 101 回まで選択できますが、パフォーマンスを考慮して最低限の回数に抑えることをお奨めします。EventReporter は、サウンド再生のアクションの実行時には、他のアクションは行いませんので、その点もご注意下さい。

■ 再生の間隔（ミリ秒）

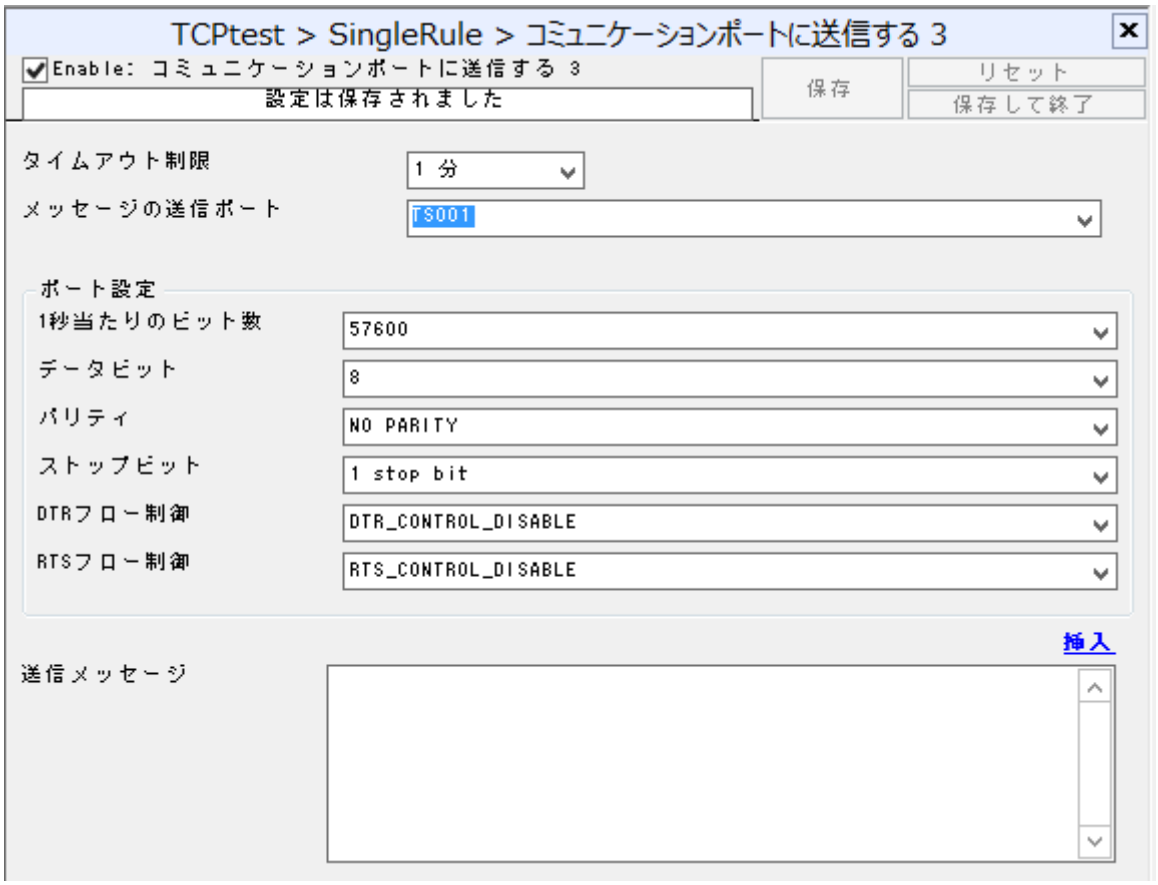
サウンドの再生回数が複数回に指定された場合の、各サウンド再生の間隔をここで設定します。
「カスタム」で値を入力する場合には、その単位は「ミリ秒」として指定して下さい。

5.5.15 コミュニケーションポートに送信

このアクションにより、コミュニケーションデバイスに文字列を送信することが可能となります。
つまり、シリアルポートでメッセージの送信を行うことができます。



▲新クライアント「コミュニケーションポートに送信」



▲旧クライアント「コミュニケーションポートに送信」

■ タイムアウトの設定（タイムアウト制限）

ここでは、 メッセージ送信の処理のタイムアウトを設定します。

ここで設定した時間内にメッセージ送信を行えなかった場合、アクションは途中で停止されます。デバイスによっては、不安定な状態になるかもしれません。

■ メッセージの送信ポート

ここでは、実装されているデバイスのポートを指定します。
一般的には、COMxポートのうちの一つです。リストボックスには、ローカルマシンで検出された全てのポートが表示されます。リモートマシンを設定している場合には、この値を別の値に合わせる必要がある場合もあります。

■ ポート設定

デバイスのポート設定に合うように設定を行って下さい。詳細は、各デバイスのマニュアルをご覧下さい。

■ 1 秒当たりのビット数

1 秒当たりのビット数は、110 から 256000 まで設定可能です。デフォルトは 57600 に設定されています。

■ データビット

データ ビットでは、送受信の 1 文字に含まれるビット数を指定します。

■ パリティ

ここでは、以下の値からパリティを設定します。

1.	Even Parity（偶数）
2.	Mark Parity（マーク）
3.	No Parity （なし）
4.	Odd Parity（奇数）
5.	Space Parity（スペース）

■ ストップビット

ここでは、以下の値からストップビットを設定します。

1.	1 stop bit
2.	1.5 stop bits
3.	2 stop bits

■ DTR フロー制御

DTR (data-terminal-ready)を設定します。以下の値から設定することができます。

1.	DTR_CONTROL_DISABLE – デバイスが開いていて、使用不能な際、DTR ラインを使用不能にします
2.	DTR_CONTROL_ENABLE – デバイスが開いていて、使用可能な際、DTR ラインを使用可能にします
3.	DTR_CONTROL_HANDSHAKE – DTR のハンドシェイクを可能にします

■ **RTS フロー制御**

RTS (request-to-send)を設定します。以下の値から設定することができます。

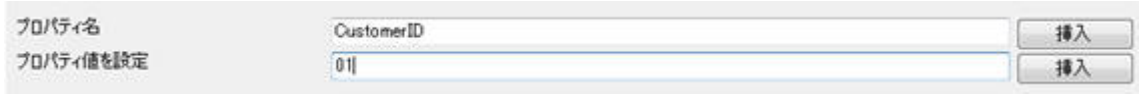
1.	RTS_CONTROL_DISABLE – デバイスが開いていて、使用不能な際、RTS ラインを使用不能にします
2.	RTS_CONTROL_ENABLE – デバイスが開いていて、使用可能な際、RTS ラインを使用可能にします
3.	RTS_CONTROL_HANDSHAKE – RTS のハンドシェイクを可能にします。ドライバーは、「タイプアヘッド(入力)」バッファが 1/2 未満の場合、RTSラインを上げて、バッファが 3/4 以上の場合は、RTS ラインを下げます。
4.	RTS_CONTROL_TOGGLE – バイトが通信に利用可能な状態では、RTS ラインが高くなり、バッファリングされた全てのバイトが送信を完了すると、RTS ラインが低くなります。

■ **送信メッセージ**

ここでは、デバイスに送信されるメッセージを送信します。
直接テキストを入力したり、現在のイベントから全てのプロパティを組み込んだりできます。
ここで挿入可能なイベントのプロパティにつきましては、下記 URL の「WinSyslog プロパティリスト」をご参照下さい。
http://www.jtc-i.co.jp/support/documents/tips/winsyslog_propertylist.pdf

5.5.16 ステータスの設定

このダイアログは、ステータスの設定オプションを設定します。
各々のインフォメーションユニットは特定のプロパティ(例えばイベント ID、プライオリティ、ファシリティなど)を持っています。そして、これらのプロパティは、いくつかの値を持っています。
イベント ID がプロパティ値 01 を持つと仮定します。既存のプロパティの中に「新たに自分で選んだプロパティを追加」したい場合には、このステータスの設定アクションでそれを行うことが可能です。
下図のように、新たにプロパティを作成し、有効な(適当な)値を割り当てることができます。
下図では、プロパティ名を「CustomerID」、プロパティに値を設定において「01」と設定しています。このアクションでプロパティを作成すると、それに対するフィルタを定義することができます。
複雑なフィルタを設定するために、製品の中には内部のステータスリストがあります。



注意：プロパティを変更すると、このアクションが実行されるとすぐに値は変更されます。
ステータスの設定のアクション実行前には変更しません。それから、以前の値は、その後は利用できないので、設定後は全てのアクションとフィルタの条件が新しい値を使用します。
従って、例えば名前の付け直しを行いたいような場合は、ステータスの設定のアクションをルールベースの先頭に持ってくるようにして下さい。

■ プロパティ名

プロパティ名を入力します。
ここで設定すると、ルールベースの内部（フィルタの条件とアクション）で使用することができるようになります。

■ プロパティに値を設定

この値は、プロパティに割り当てられます。有効なプロパティタイプ値であれば、どんな値でも入力できます。

5.5.17 ステータス変数算出

このアクションは、内部処理に関するものです。
この機能は、カウンタベースで作用するルールセットに対して必要なものです。
ここでは、ステータス変数算出の管理を行います。

■ ステータス変数

ステータス変数名を指定します。ここでは、プロパティの置換（「挿入」をクリック）機能が使用できます。

＜オプションタイプ＞

■ 増加値（+）

オペレーション値によって、値が増加されます。

■ 減少値（+）

オペレーション値によって、値が減少されます。

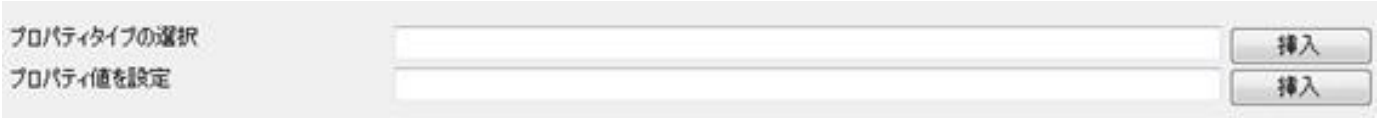
■ オペレーション値

使用するオペレーション値を設定します。

5.5.18　プロパティの設定

ここでは、プロパティの設定を管理します。
「プロパティの設定」アクションにより、入ってくるメッセージのプロパティのいくつかを修正することができます。これは、管理者が二つのデバイスに同じ名前を付け直したい場合などは特に役に立ちます。

注意：プロパティを変更すると、このアクションが実行されるとすぐに値は変更されます。
プロパティの設定のアクション実行前には変更しません。それから、以前の値は、その後は利用できないので、設定後は全てのアクションとフィルタの条件が新しい値を使用します。
従って、例えば名前の付け直しを行いたいような場合は、プロパティの設定のアクションをルールベースの先頭に持ってくるようにして下さい。



■ プロパティ タイプの選択
変更したいプロパティのタイプを選択します。
挿入（Insert）をクリックすると、プロパティのリストが表示され選択することができます。

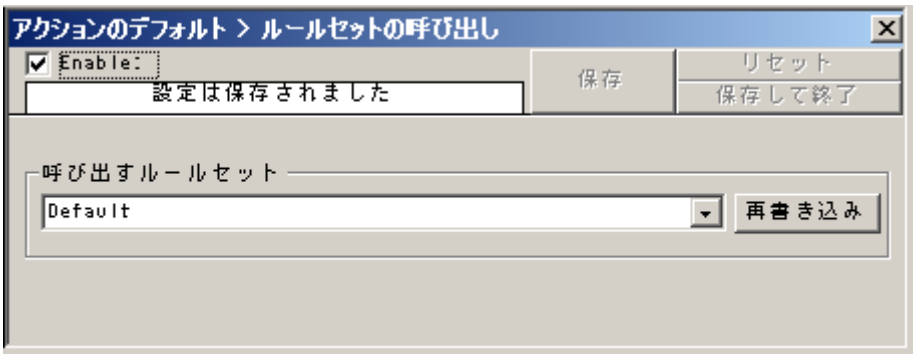
■ プロパティに値を設定
プロパティに割り当てられる新しい値を入力します。
有効なプロパティ値ならば、どんなものでも入力することが可能です。

5.5.19　ルールセットの呼び出し

このダイアログは、ルールセットの呼び出しオプションを設定します。
このアクションは、存在しているルールセットの中から、特定のルールセットを呼び出すために設定します。このアクションが実行されると、ルールエンジンは、通常の処理を止め、ここで指定したルールセットを呼び出します。（そのルールセットには、ルールが含まれいます。）そして、呼び出されたルールセット内で定義されたルールが全て実行されます。その後、通常の処理に戻ります。（処理を止めた時点に戻ります。）具体的には、下記の例をご参照下さい。
仮に、Rule1 には、アクション 1 とアクション 2 が存在するとします。
アクション 1 には、この「ルールセットの呼び出し」アクションが下図のように設定されています。Rule1 のフィルタの条件が真（True）の場合、アクション 1 が実行されます。アクション 1 は、「ルールセットの呼び出し」が設定されているので、そこで設定されているルールセット（Default）を呼び出し、そのフィルタの条件が実行されます。そこ

で真（True）となった場合には、ルールセット（Default）配下のアクションを全て実行し、アクション 2 に戻ります。もしも、呼び出したルールセットのフィルタの条件で偽（False）となった場合は、ルールセット（Default）配下のアクションは全く実行されず、アクション 2 に戻ります。

▲新クライアント「ルールセットの呼び出し」



▲旧クライアント「ルールセットの呼び出し」

■ 処理を実行するルールセット（呼び出すルールセット）

呼び出しを行うルールセットを選択します。
ルールセットが１つの場合には、このアクションは使用できません。

5.5.20 破棄

（重要でないものなど）無視したいイベントがある場合には、この破棄アクションを設定することで実行できます。
破棄のアクションが含まれるルール内のフィルタの条件で、破棄する条件（フィルタ）の設定を行ってください。
このアクションは、詳細を設定する必要がないため、アクションの追加で選択しても設定のダイアログは表示されません。

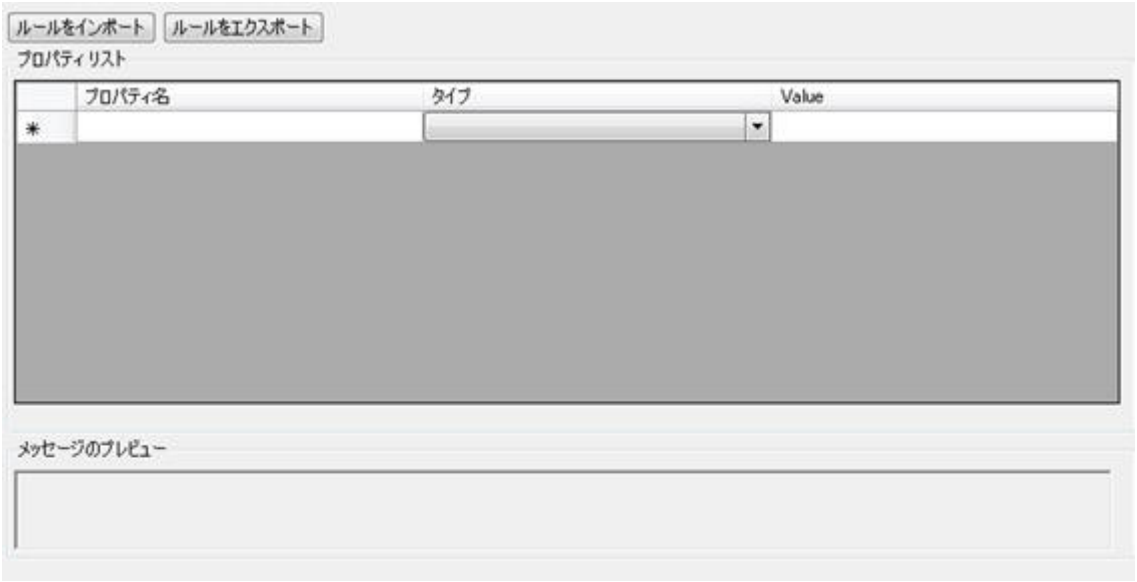
5.5.21 Post-Process イベント

Post-Process イベントアクションにより、一旦処理されたメッセージの再解析が可能となります。
このような再解析は、標準でない Syslog フォーマットを使用している場合や、メッセージから特定のプロパティを取り出したい場合に、非常に役に立ちます。

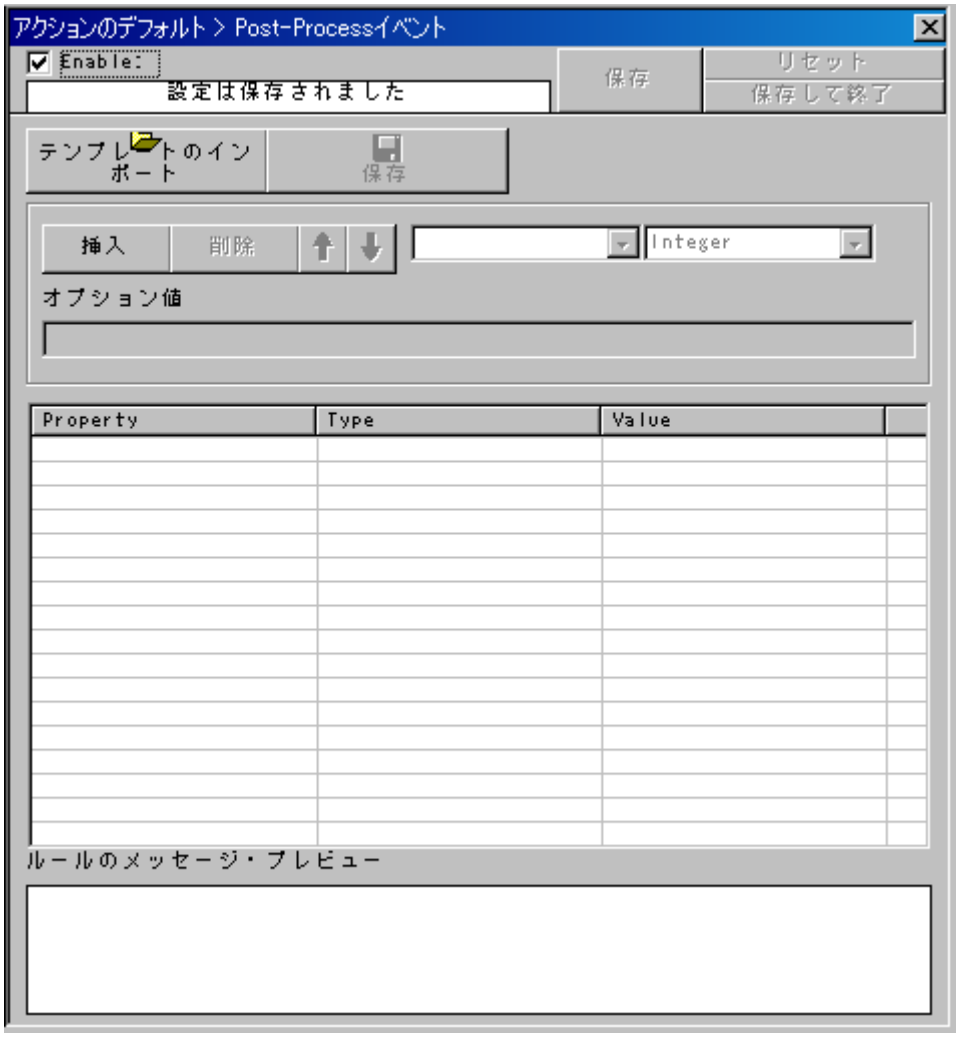
Post-Process アクションは、受信したメッセージを取り出し、解析マップにより解析を行います。解析マップには、メッセージのどの位置に、どのタイプのどのプロパティが存在するかが示されています。

メッセージが実際に解析マップに適合する場合、全てのプロパティが引き出され、イベントの一部としてセットされます。

メッセージが解析マップに適合しない場合は、最初に適合しなかったエントリで 解析は停止します。



▲新クライアント「Post Process イベント」



▲旧クライアント「Post Process イベント」

■ テンプレート

解析マップは、非常に複雑です。

解析マップとのやり取りを簡単にするために、それらは XML ファイルで処理されます。

■ 解析マップ エディタ

上図のデータグリッドにおいて、編集を行うことができます。

新クライアントでは、データグリッドをダイレクトに編集することができます。

旧クライアントでは、テキストボックスで編集を行います。

グリッドのエントリを選択すると、その値はテキストボックスで更新されます。そこで行われるどんな編集も自動的にグリッドに反映されます。「挿入」をクリックすると新たに項目が追加され、「削除」をクリックすると選択されている項目が消去されます。

■ プロパティ名（Property）

解析されるプロパティ名です。プロパティのリストボックスには、予め実装された標準とイベントプロパティが含まれています。しかし、ここには新たにプロパティを追加することができます。新たにプロパティを追加する際には、プロパティ名の頭に「u-」を付けることをお奨めします。そうすることにより、既存のプロパティと重複することを避けられます。

例えば、「MyProperty」という名のプロパティを追加したい場合には、「u-MyProperty」とするようお奨めします。

「Filler」という名のプロパティは、既に固定されています。このプロパティに振り分けられる全ての値は、破棄されてしまいます。このプロパティは、不要な充填文字などを取り除きたい場合に有効です。

■ タイプ（Type）

メッセージから解析されるタイプです。

例えば、「Integer」タイプは、メッセージから 1 つの整数を解析します。また、「Word」タイプは、次の語を解析します。（ただし、次の語がスペースの場合を除く）

■ Value

タイプの中には、値を追加する必要がある場合もあります。その際には、ここで値を追加します。

■ メッセージのプレビュー（ルールのメッセージ・プレビュー）

ここは、読み込み専用のボックスです。

設定された構文解析ルールにマッチする仮定のメッセージを表示します。

日本語マニュアル発行日　2016 年　10 月 05 日
本マニュアル原文は『WinSyslog』です
ジュピターテクノロジー株式会社　技術グループ