ORACLE
Cloud Infrastructure

Oracle Cloud Infrastructureで安全にグラフィカル・アプリケーションを実行

ORACLE WHITE PAPER | 2018年8月

ORACLE

免責事項

下記事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメント（確約）するものではないため、購買決定を行う際の判断材料になさらないで下さい。オラクルの製品に関して記載されている機能の開発、リリース、および時期については、弊社の裁量により決定されます。

改訂履歴

本ホワイトペーパーは、初版の公開後、次の改訂がありました。

日付	改訂内容
2018年8月20日	初版発行

Oracle Cloud Infrastructureホワイトペーパーの最新版は、
https://cloud.oracle.com/iaas/technical-resourcesでご覧いただけます。

目次

はじめに

Linuxのアプリケーションおよびインストーラの一部には、単純なコマンドラインではなくグラフィカル・インタフェースが必要になります。インストールやアプリケーションが複雑、データ入力に必要、あるいはユーザー・エクスペリエンスの改善のため、といった理由からです。デフォルトでは、Oracle Cloud InfrastructureのLinuxインスタンスはSSHセッションを介してコマンドラインを入力することによって使用されます。グラフィカル・アプリケーションを実行するには、VNCをインストールし、セキュリティ・リストでTCPポートを開き、実行を開始するというのが通常のプロセスです。ただし、この方法で実行すると、特にインターネットを介したインスタンスへの接続上で直接実行する場合には、VNCにセキュリティ上の問題が発生することがあります。

これを解決するには、インスタンスへのセキュア接続を確立したうえで、その接続上でアプリケーションを実行する必要があります。Oracle Cloud Infrastructure内で確立されたセキュリティ・スタンスを保存する必要もあるので、それにはユーザー・アカウントで特定のパスワードを使用せずに通信と認証を非同期で暗号化します。作成するソリューションでは、アプリケーションを簡単に実行できるようにしつつ、セキュリティを確保する必要があります。

しかし、ここで問題になるのが、グラフィカル・アプリケーションをただ実行すればいいわけではないということです。「スタンドアロン」方式（特定のウィンドウ・マネージャを必要としない）で実行できるアプリケーションもありますが、完全なデスクトップ環境が必要なアプリケーションもあります。そこで、2つの問題の解決を考えます。

- **完全なデスクトップ体験を必要としない独立のアプリケーションを実行する**：この方法をデフォルトにする必要があります。公開される部分が最も少なく、インストールするソフトウェアも限られるからです。この方法の場合、グラフィカル・アプリケーションには特定の実行可能ファイルが必要ですが、デスクトップ・リソースは必要ありません。
- **完全なデスクトップ環境を実現する**：この方法では大量のソフトウェアをインストールする必要があり、十分なセキュリティのためには特定の構成も必要です。アプリケーションに、GNOMEやKDEといった完全なデスクトップ環境が必要な場合に限定するようにしてください。

幸い、この2つの方法は二者択一というわけではありませんが、グラフィカル・アプリケーションを安全に実行するためにクライアントとインスタンスの両方を設定する手順は異なります。

解決策

この項では、完全なデスクトップ体験を必要としない独立のアプリケーションを実行する場合と、完全なデスクトップ環境を実現する場合の両方のシナリオについて説明します。

シナリオ1：独立のアプリケーションを実行する

シナリオ1では、デスクトップ環境を必要とせずに独立のアプリケーションを実行します。これを行うには、1つ目のグラフィカル環境、X11を使用します。

X11は、不当に低く評価されることもあります。SSHトンネルと組み合せれば安全に実行できます。ここでもそのように設定してみます。SSHトンネル上でX11を使用して独立のアプリケーションを実行するメリットは、設定が最小限で済み、コンピュート・インスタンスのバックグラウンドでデスクトップ環境を実行する必要がないということです。必要に応じてアプリケーションを実行し、終了するだけです。

シナリオ1の実装に必要な全ステップを紹介します。

Oracle Cloud Infrastructureのインスタンスの設定

1. インスタンスにログインします。
2. X11にlocalhostを*使用しない*ようにSSHDを構成します。
   A. 任意のエディタで`/etc/ssh/sshd_config`を開きます。
   B. `X11UseLocalhost yes`という記述がある行（コメント・アウトされています）を検索します。
   C. この行のコメント・アウトを解除します。
   D. `yes`を`no`に変更します。
   E. ファイルを保存します。
   F. SSHDを再起動します：`sudo systemctl restart sshd`
3. xauthをインストールします：`sudo yum -y install xauth`
4. xterm（X構成の検証に使用）をインストールします：`sudo yum -y install xterm`
5. インスタンスからログアウトします。

これでインスタンスの準備ができました。ただし、グラフィカル・アプリケーションを使用するには、Xサーバーを設定する必要があります。クライアントでそれを実行する方法は、MacとPCのどちらを使用しているかによって異なります。

クライアントの設定：PCの場合

PCでクライアントを設定するには、2つのソフトウェアのどちらかを使用します。ここで使用するのは、PCユーザーに対する*ごく非公式*のアンケートで一般的だったソフトウェアです。

筆者が使用してみたXサーバーの2つのプロバイダーは次のとおりです。

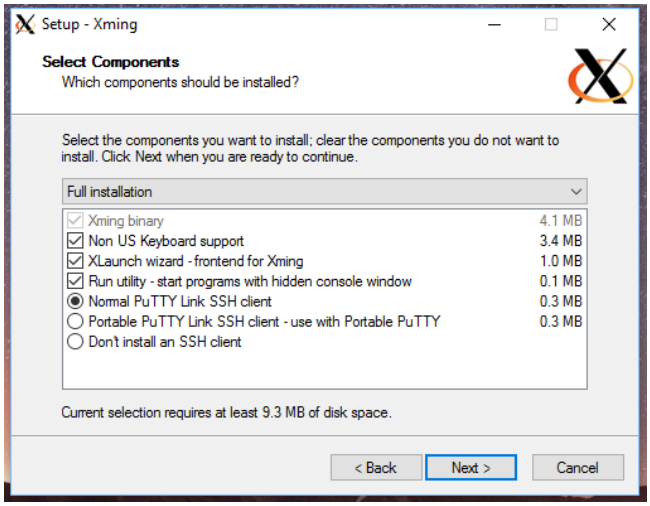
- Xming（https://sourceforge.net/projects/xming/）
- Cygwin/X（https://x.cygwin.com/）

本書ではXmingを使用します。手順の多くは、他のXサーバーでも共通なので、読み替えるのは難しくないはずです。

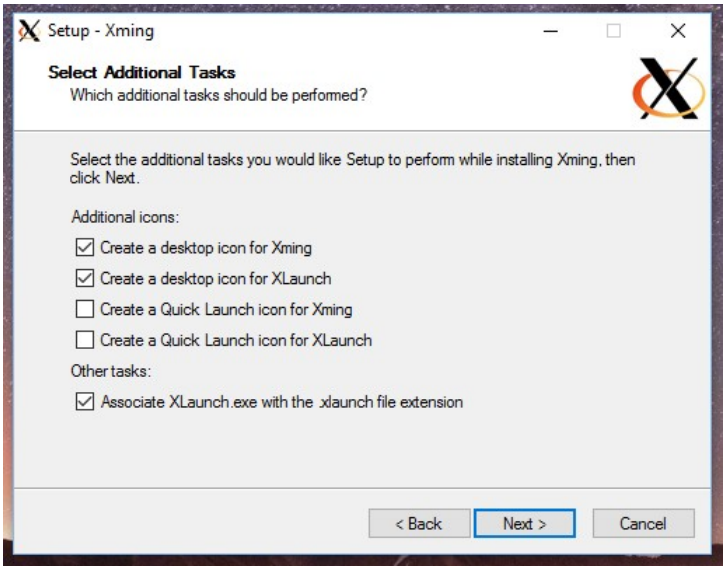
注意：インスタンスに接続して、X11トラフィック用に特定のトンネルを確立するには、引き続きSSHを使用する必要があります。SSHクライアントとしてはPuTTY（www.putty.org）が最も一般的なようですが、X11のトンネリングをサポートしていれば、どのSSHクライアントでも*使用できます*。PuTTY以外のクライアントを使用する場合、X11フォワーディングの設定については各ドキュメントを参照してください。

1. Xmingをダウンロードし、インストーラを実行します。以下のステップで指定する以外は、すべてデフォルトのままにしてください。

   A. インストーラでSSHクライアントの選択を求められたら、**「Normal PuTTY Link SSH client」**を選択します。



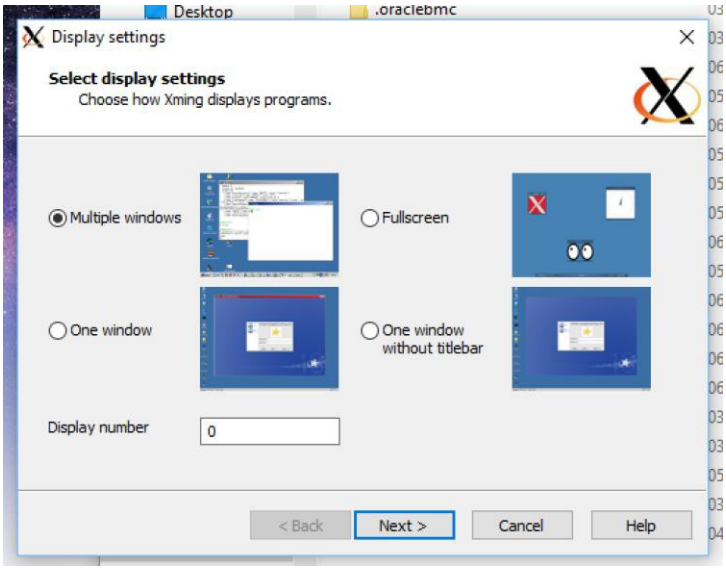
B. *（オプション）*XLaunchとXmingのデスクトップ・アイコンを作成する場合は、このオプションを選択します。**「Associate XLaunch.exe with the .launch file extension」**チェック・ボックスが選択されていることを確認してください。



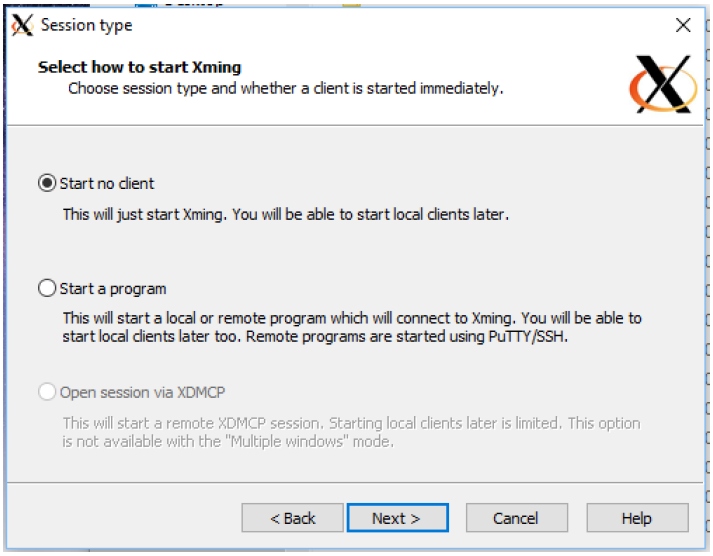
2. Xmingがインストールされたら、XLaunchアプリケーションを実行します。

一連の設定ページが表示されます。

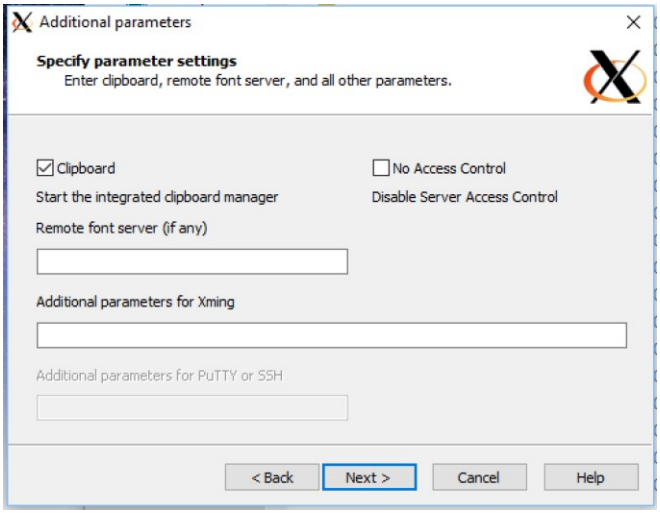
3. 表示設定については「**Multiple Windows**」オプションを選択し、「**Next**」をクリックします。



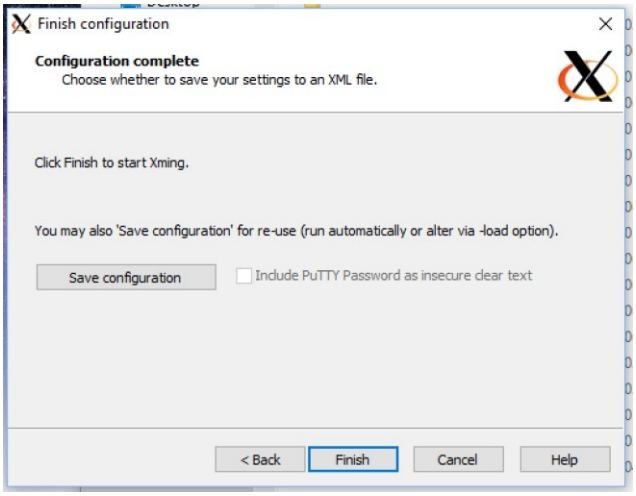
4. セッション・タイプは、「**Start no client**」オプションを選択します。



5. その他のパラメータは**「Clipboard」**チェック・ボックスを選択します。



6. **「Finish configuration」**ページで**「Save configuration」**ボタンをクリックして、この構成を**C:\Program Files (x86)\Xming**ディレクトリに保存します。最後に**「Finish」**をクリックします。

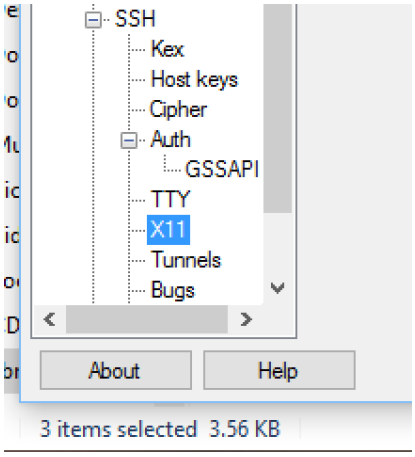


PuTTY SSHセッションの構成

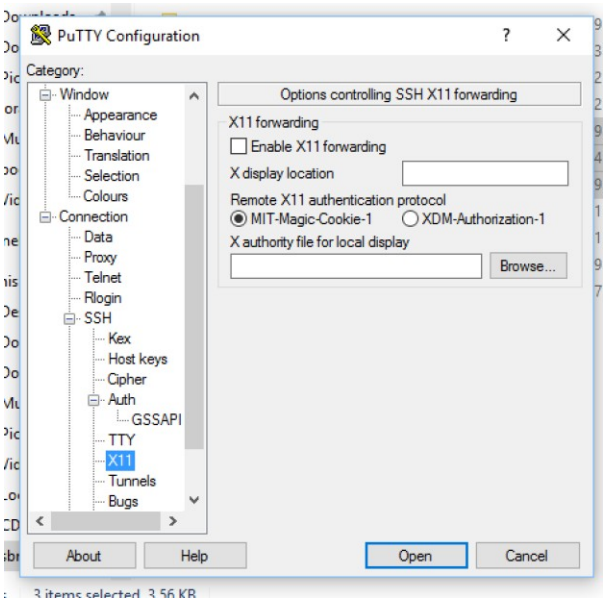
PuTTYでSSHセッションを設定している場合には、セッションを編集してセッション構成にX11フォワーディングを追加します。この構成を保存しない場合は、グラフィカル・アプリケーションを実行するたびに構成が必要になります。

1. PuTTYを開きます。

2. 左端の列で**SSH**を展開し、**「X11」**を選択します。



3. オプションのペインで、**「Enable X11 forwarding」**チェック・ボックスを選択します。



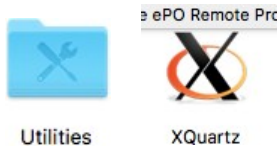
4. ここでいったん停止します。セッションは*開かない*でください。構成を保存する場合は、ここで保存します。「グラフィカル・アプリケーションの実行」の項に進んでください。

クライアントの設定：Macの場合

PCの設定の場合と同じく、Macの設定でもXサーバーをインストールします。この場合は、XQuartzを使用します。

http://www.xquartz.orgにアクセスして`.dmg`ファイルをダウンロードし、インストールします。すべてデフォルト値のままでかまいません。

XQuartzがインストールされたら、Finderで**「ユーティリティ」**フォルダを開いて**「XQuartz」**アイコンをダブルクリックします。



ターミナル・ウィンドウが開きます。SSH接続にこのセッションを使用できますが、XQuartzを起動していれば、どのターミナル・ウィンドウでも機能します。

Macでは、SSH設定は特に必要ありません。オペレーティング・システムのSSHが組み込まれています。ターミナル・ウィンドウで`ssh`と入力するだけです。

クライアントの設定が済むと、独立のアプリケーションを実行できるようになります。

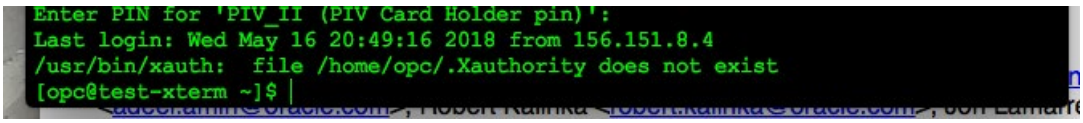
グラフィカル・アプリケーションの実行

インスタンスとクライアントの両方の設定が終わったら、アプリケーションを実行できます。一般的には、次の手順を実行します。

1. ラップトップまたはデスクトップでXサーバーを起動します。
   - Macの場合は、**「XQuartz」**アイコンをダブルクリックします。
   - PCの場合は、**「Xming」**アイコンをダブルクリックします。
2. SSHを使用してインスタンスに接続します。
   - PCの場合は、前述した設定の項で指定したとおりにPuTTYセッションが構成されていることを確認してください。
   - Macの場合は、次のようにコマンドを入力します。

     ```
     ssh -XC opc@<public_IP_address>
     ```

注意：初めてインスタンスに接続する場合は、次のような警告が表示されます。



この警告は正常で、機能に影響はないので、無視しても問題ありません。

3. 次のコマンドを実行して、DISPLAY変数が設定されていることを確認します。
   `env | grep DISPLAY.`
   次のような結果が表示されるはずです（次の図は、セッションにiTerm2を使用しているMacでの表示）。

4. `xterm &`を実行します。
   ローカル・デスクトップで新しいシェル・セッションが開き、Xサーバーが正常に稼働していることを検証します。
   xtermセッションがバックグラウンドに移り、同じセッションで他のアプリケーションを起動できるようになります。

5. アプリケーションを実行します。グラフィカル・アプリケーションは、完全なコマンドラインの最後に`&`を付けてバックグラウンドで実行することをお薦めします。こうすると、複数のグラフィカル・アプリケーションを同時に実行できるからです。

おめでとうございます。必要な数だけアプリケーションを実行できますが、バックグラウンドに設定することを忘れないでください。アプリケーションを閉じるときは、デスクトップ上の場合と同じように終了します。

ヒント：アプリケーションが終了しない場合は、X11スタックに付属しているxkillツールを使用できます。このツールを使えば、クリックしたウィンドウに関係付けられているプロセスが終了します。SSHセッションでxkillを実行し、終了したいウィンドウをクリックしてください。

シナリオ2：完全なデスクトップ環境の実行

独立のアプリケーションを実行できない場合は、完全なデスクトップ環境を設定することができます。手間はかかりますが、手順は単純です。ただし、バックグラウンドで稼働するプロセスを追加するので、リソースは消費することに注意してください。

そのため、小さなインスタンスの場合には、インスタンスの動作に変化があることがあります。

Oracle Cloud Infrastructureのインスタンスの設定

1. インスタンスにログインします。
2. libEGLをインストールします：`sudo yum -y install mesa-libEGL`
3. libGLをインストールします：`sudo yum -y install mesa-libGL`
4. グラフィカル・デスクトップをインストールします：`sudo yum -y groupinstall "Server with GUI"`

   注意：このインストールには時間がかかり、ルート・ボリュームにおよそ2GBの容量が必要です。

5. TigerVNCをインストールします：`sudo yum -y install tigervnc-server`
6. `vncpasswd`を実行して、VNCにアクセスする際の`opc`のパスワードを設定します。

   注意：このパスワードはVNCによって使用されます。インスタンスのユーザー・アカウントに、あるいはインスタンスへの認証には使用されません。その機能を次のステップで無効にします。

7. アクセスするポートを指定して、テンプレートvncserver systemdスクリプトをコピーします。

   ```
   cp /lib/systemd/system/vncserver@.service
   /lib/systemd/system/vncserver@:1.service
   ```

   - ファイル名にある`:1`は、標準VNCポート（5900）からのオフセットを示します。この例では、クライアントは5901に接続するということです。
   - 他のユーザーを作成する必要がある場合は、テンプレートのコピーを複数作成し、それぞれに重複しないようにポート番号を指定します。

8. 任意のエディタで`vncserver@:1`ファイルを開きます。
9. 開いたファイルで、`<USER>`をすべて`opc`に変更します。

   複数のユーザーを構成している場合は、後続のそれぞれのファイルで`<USER>`を適切なユーザー名に変更してください。

10. `ExecStart`で始まる行を検索します。その行の一部には、`/usr/bin/vncserver %i`という文字列が含まれています。その文字列を`/usr/bin/vncserver - localhost %i`に変更します。これで、ループバック・アダプタを介してのみ接続できるようVNCに指示することになります。

インスタンス内部で見つかった以外のIPアドレスから接続する機能を無効にするので、これは重要です。基本的に、他の方法でインスタンスに接続する必要があるのは、VNCサーバーに接続できるようになる前です。他のユーザーも構成している場合には、そのユーザーについてもこのステップを繰り返します。

11. グラフィカル・デスクトップを起動します：`sudo systemctl start graphical.target`

12. VNCセッションを起動します：`sudo systemctl start vncserver@:1`

    複数ユーザーの場合は、`vncserver@:2`、`vncserver@:3`のようにポート番号を置き換えてください。

ステップ11と12が有効なのは、インスタンスを再起動するまでです。再起動後は、デスクトップもVNCサーバーも停止し、自動的には再起動しません。一般的にはこの動作で適切です。デスクトップ環境を使用するのは、何か設定するときや、構成を実行するときだけだからです。デスクトップを使用する必要がある場合は、ステップ11および12に示したコマンドを実行するだけです。

ただし、インスタンスを再起動するたびにデスクトップとVNCサーバーを起動したい場合は、インスタンスの再起動時に再起動するよう指示するために1回だけ次のコマンドを実行します。

```
sudo systemctl enable graphical.target
sudo systemctl enable vncserver@:1
```

クライアントの設定

前述したように、VNCサーバーはインスタンス自体の内部からの接続しか受け入れないように設定されています。その動作のためには、どうすればいいのでしょうか。その質問に答える前に、ローカル・クライアント（ラップトップまたはデスクトップ）にVNCクライアントをインストールする必要があります。

MacでもPCでも設定は同じで、OSにVNCクライアントをインストールします。この例では、TigerVNCを指定します。

http://www.tigervnc.orgに移動し、*クライアント・ソフトウェア*のみをダウンロードします。サーバーは必要ありません。

- Windowsのクライアントは、**vncviewer64*.exe**としてリストされます。
- Macのクライアントは、https://bintray.com/tigervnc/stable/tigervnc/1.8.0にある**.dmg**ファイルに含まれています。

指示に従ってソフトウェアをインストールします。

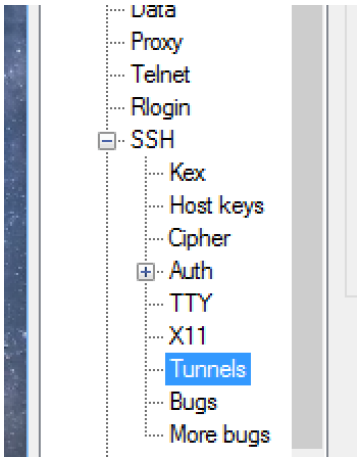
これでVNCソフトウェアの用意ができたので、SSHを使用してラップトップからインスタンスに接続します。シナリオ1と同様、SSHのためのクライアント設定は、MacとPCで異なります。

SSHの構成

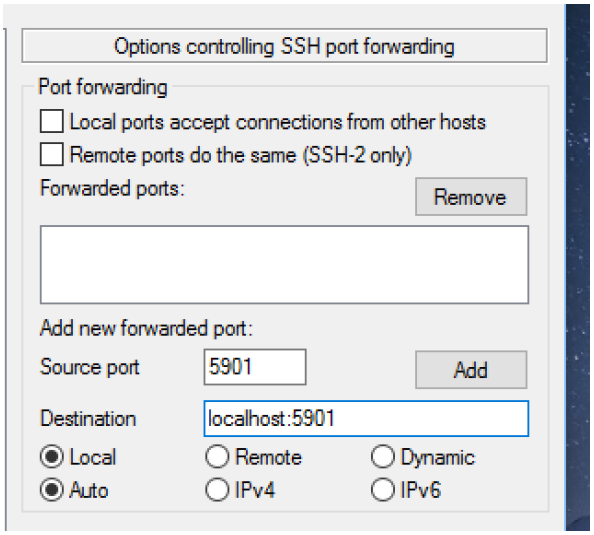
シナリオ1の場合と同様、MacではSSHの設定は特に必要ありません。

PCで、インスタンス用にPuTTYでSSHセッションを設定している場合は、セッションを編集して構成を追加する必要があります。

1. PuTTYを開きます。
2. 左端の列で**SSH**を展開し、**「Tunnels」**を選択します。

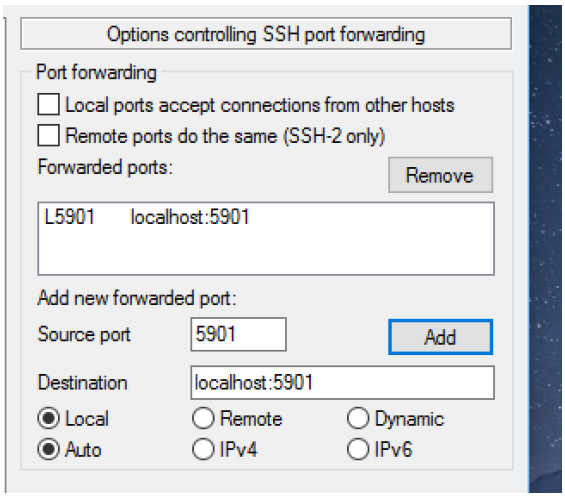


3. オプションのペインで、すでに構成してあるVNCサーバーのソース・ポートを入力します。**opc**ユーザー（**opc**のVNCサーバーとして**:1**を選択していると仮定します）の場合、このポートは通例、5901です。**「Destination」**フィールドに、**localhost:**と入力し、続いてサーバーのポート（ここでも通常は5901）を入力します。



4. **「Add」**をクリックします。

   「Forwarded ports」ボックスに、次のスクリーンショットのようなエントリが入力されます。



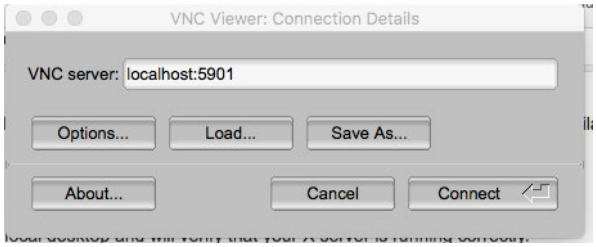
5. 構成を保存します。

クライアントの設定が済むと、完全なデスクトップ環境でアプリケーションを実行できるようになります。

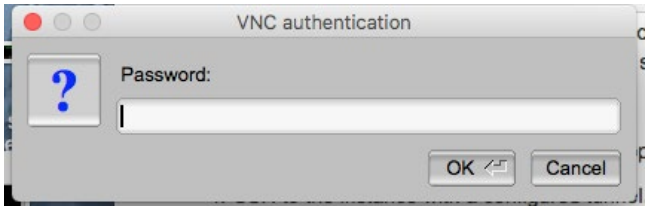
完全なデスクトップ環境の実行

これで、Oracle Cloud Infrastructureのインスタンスでデスクトップに接続できるようになりました。接続するには、いくつかのステップを実行してセキュア接続を開く必要があります。

1. SSHを使用して、構成したトンネルでインスタンスに接続します。
   - PCの場合は、設定手陣で作成したPuTTYのセッション構成を使用します。
   - Macの場合は、次のコマンドを使用します：`ssh -L 5901:localhost:5901 opc@<public_IP_address>`
2. ラップトップまたはデスクトップ上のアイコンをクリックして、TigerVNCを起動します。
3. 「**VNC server**」フィールドに、**localhost:5901**と入力します。



4. 「**Connect**」をクリックします。
5. パスワードのダイアログ・ボックスに、VNCサーバーの設定のセクションで指定したVNCパスワードを入力します。



6. これがインスタンス・デスクトップへの初めての接続の場合は、設定上の質問が表示されるので、それに答えてください。質問の答えが完了すると、デスクトップが使用できるようになります。

結論

アプリケーションには多くの要件があります。そのひとつとして、グラフィカル・モードで実行できることが求められる場合もあります。しかし、Oracle Cloud Infrastructureのセキュリティ・モードとインスタンス構成はコマンドラインでアプリケーションを実行する時点で組み込まれるため、グラフィカル・インタフェースが必要とされません。グラフィカル・アプリケーションを実行しつつ、しかも強固なセキュリティを確保するために、以上のような手順を用意しました。実行する必要がある環境のタイプ（単一のアプリケーション環境または完全なデスクトップ）は、アプリケーションごとの要件に基づいて選択できます。

ORACLE

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax：+1.650.506.7200

Integrated Cloud Applications & Platform Services

Oracle Cloud Infrastructureで安全にグラフィカル・アプリケーションを実行
2018年8月
著者：Steve B. Nelson