

日々是Oracle APEX

Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

2024年3月8日 金曜日

Oracle Autonomous Database Free Container ImageをOracle Cloudのコンピュート・インスタンスで動作させる

Oracle Autonomous Database Free Container ImageをOracle Cloudのコンピュート・インスタンスで動作させてみました。

Oracle APEXのインストールが英語リソースのみであったり、現時点でのコンテナ・イメージにAPEX 23.1（ひとつ前のバージョン）がインストールされている、といったことでAPEXの開発環境としては今ひとつです。

この環境の使い所という点では、データベースを壊しかねないような検証、例えば、Database VaultやEdition Base Redefinition、Liquibaseを使ったスキーマ更新などの検証に使えるのではないかと考えています。Always FreeのAutonomous Databaseは2環境までという制限がありますし、有償のAutonomous Databaseは最小構成のAPEXサービスでも、発生する費用は桁が違います。

また、コンテナを動かせることができるのであれば、他のクラウド、例えばAWS、Microsoft AzureやGoogle Cloudでも変わらず実行できるでしょう。

以下より作業手順を記述します。

Oracle Autonomous Database Free Container ImageはOracle Container Registryよりダウンロードできます。

<https://container-registry.oracle.com/ords/ocr/ba/database/adb-free>

コンテナ・イメージとしては同じと思われますが、GitHub上でも紹介されています。

<https://github.com/oracle/adb-free/pkgs/container/adb-free>

最初にOracle Cloudでコンピュート・インスタンスを作成します。



名前はadb-freeとします。



イメージはOracle Linux 8を選択し、Shapeは最小限必要な2 core OCPU、8 GB memoryを設定します。



パブリックIPv4アドレスの自動割当てにチェックを入れます。



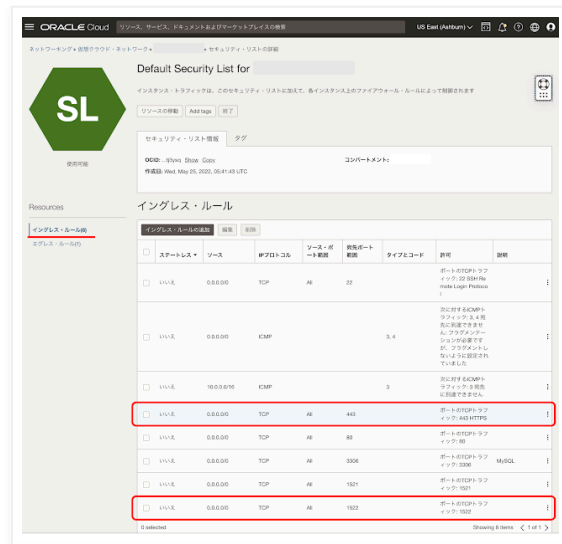
キー・ペアを自動で生成を選択し、秘密キーの保存をクリックします。すでに手元に公開キーと秘密キーのペアを所有している場合は、公開キーをアップロードするか貼り付けることができます。

この設定でコンピュータ・インスタンスを作成します。



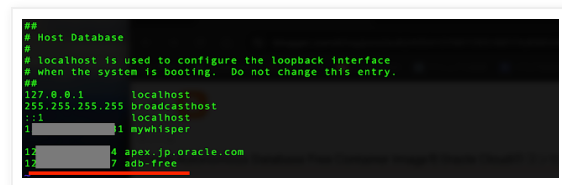
コンピュータ・インスタンスが作成されます。パブリックIPv4アドレスをコピーしておきます。

ネットワークのイングレス・ルールとして、宛先ポート443とポート1522（とポート22）に通過許可を与えておきます。



クラウド側の準備は以上です。

データベースやWebサーバーに接続する際にIPアドレスを毎回指定することを省くため、手元のPCのhostsファイルにIPアドレスのエントリを追加しておきます。ホスト名はadb-freeとします。



ダウンロードした秘密キーのファイルを使い、ユーザーopcで作成したコンピュート・インスタンスに接続します。

```
ssh -i ssh-key-2024-03-08.key opc@adb-free
```

```
% ssh -i ssh-key-2024-03-08.key opc@adb-free
The authenticity of host 'adb-free (***.***.***.***)' can't be established.
ED25519 key fingerprint is SHA256:junz4X3Ux9VJi2GKfjcKTXpwOVRZxVHPIDQgcYdw85Q.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'adb-free' (ED25519) to the list of known hosts.
Activate the web console with: systemctl enable --now cockpit.socket
```

```
[opc@adb-free ~]$
```

podmanをインストールします。手順は以下の資料を参照しています。

https://docs.oracle.com/ja/learn/intro_podman/index.html#introduction

```
sudo dnf module install container-tools:ol8
```

ホストにOracle Linux 9を選んでいる場合のコマンドは以下です。

```
sudo dnf install container-tools
```

```
[opc@adb-free ~]$ sudo dnf module install container-tools:ol8
Failed to set locale, defaulting to C.UTF-8
Ksplice for Oracle Linux 8 (x86_64) 60 MB/s | 5.5 MB 00:00
```

```

MySQL 8.0 for Oracle Linux 8 (x86_64)          41 MB/s | 3.6 MB      00:00
MySQL 8.0 Tools Community for Oracle Linux 8 (x  11 MB/s | 634 kB      00:00
MySQL 8.0 Connectors Community for Oracle Linux 558 kB/s | 30 kB       00:00
Oracle Software for OCI users on Oracle Linux 8 125 MB/s | 118 MB     00:00
Oracle Linux 8 BaseOS Latest (x86_64)          120 MB/s | 71 MB     00:00
Oracle Linux 8 Application Stream (x86_64)       115 MB/s | 55 MB     00:00
Oracle Linux 8 Addons (x86_64)                  73 MB/s | 9.4 MB    00:00
Latest Unbreakable Enterprise Kernel Release 7  109 MB/s | 28 MB     00:00
Dependencies resolved.

```

```

=====
Package
Arch    Version
=====
Installing group/module packages:
aardvark-dns
x86_64  2:1.7.0-1.module+el8.9.0+90147+a4870853  ol8_appstream 1.0 M
buildah
x86_64  1:1.31.3-3.0.1.module+el8.9.0+90147+a4870853 ol8_appstream 8.8 M
cockpit-podman
noarch  75-1.module+el8.9.0+90147+a4870853      ol8_appstream 738 k
common x86_64 3:2.1.8-1.module+el8.9.0+90147+a4870853  ol8_appstream 56 k
container-selinux
noarch  2:2.221.0-1.module+el8.9.0+90147+a4870853 ol8_appstream 68 k
containernetworking-plugins

```

[中略]

```

python3-pytoml-0.1.14-5.git7dea353.el8.noarch
python3-pyxdg-0.25-16.el8.noarch
runc-1:1.1.12-1.module+el8.9.0+90147+a4870853.x86_64
shadow-utils-subid-2:4.6-19.el8.x86_64
skopeo-2:1.13.3-3.0.1.module+el8.9.0+90147+a4870853.x86_64
slirp4netns-1.2.1-1.module+el8.9.0+90147+a4870853.x86_64
udica-0.2.6-20.module+el8.9.0+90147+a4870853.noarch
yajl-2.1.0-12.el8.x86_64

```

```

Complete!
[opc@adb-free ~]$

```

firewalldの設定を行ないます。

ポート1522、8443に接続許可を与え、ポート443への接続を8443へ転送します。

```

sudo firewall-cmd --add-port=1522/tcp
sudo firewall-cmd --add-port=8443/tcp
sudo firewall-cmd --add-forward-port=port=443:proto=tcp:toport=8443
sudo firewall-cmd --runtime-to-permanent
sudo firewall-cmd --list-all

```

```

[opc@adb-free ~]$ sudo firewall-cmd --add-port=1522/tcp
success
[opc@adb-free ~]$ sudo firewall-cmd --add-port=8443/tcp
success
[opc@adb-free ~]$ sudo firewall-cmd --add-forward-
port=port=443:proto=tcp:toport=8443
success
[opc@adb-free ~]$ sudo firewall-cmd --runtime-to-permanent
success
[opc@adb-free ~]$ sudo firewall-cmd --list-all
public (active)
target: default
icmp-block-inversion: no

```

```
interfaces: ens3
sources:
services: dhcpv6-client ssh
ports: 1522/tcp 8443/tcp
protocols:
forward: no
masquerade: no
forward-ports:
    port=443:proto=tcp:toport=8443:toaddr=
source-ports:
icmp-blocks:
rich rules:
[opc@adb-free ~]$
```

今回はワークロード・タイプに**ATP**を指定して、Oracle Autonomous Database Free Container Imageからコンテナ**adb-free**を作成します。

あらかじめ環境変数**MYPASS**に**パスワード**を設定しておきます。

```
export MYPASS=<パスワード>
```

以下のコマンドを実行します。

```
podman run -d \
-p 1521:1522 \
-p 1522:1522 \
-p 8443:8443 \
-p 27017:27017 \
-e WORKLOAD_TYPE='ATP' \
-e WALLET_PASSWORD=$MYPASS \
-e ADMIN_PASSWORD=$MYPASS \
--cap-add SYS_ADMIN \
--device /dev/fuse \
--name adb-free \
container-registry.oracle.com/database/adb-free:latest
```

初回実行時はコンテナ・イメージのダウンロードが発生するため、少し時間がかかります。

```
[opc@adb-free ~]$ podman run -d \
> -p 1521:1522 \
> -p 1522:1522 \
> -p 8443:8443 \
> -p 27017:27017 \
> -e WORKLOAD_TYPE='ATP' \
> -e WALLET_PASSWORD=$MYPASS \
> -e ADMIN_PASSWORD=$MYPASS \
> --cap-add SYS_ADMIN \
> --device /dev/fuse \
> --name adb-free \
> container-registry.oracle.com/database/adb-free:latest
Trying to pull container-registry.oracle.com/database/adb-free:latest...
Getting image source signatures
Copying blob 66cb99a7273e done
Copying blob 054518c1d5d2 done
Copying blob 0e3c46fdb0f9 done
Copying config 0cadb9d2c5 done
Writing manifest to image destination
28942510705f16df404a037f1e122c23c4b337a5493dc53aa7524b4278050a1e
```

```
[opc@adb-free ~]$
```

コンテナが作成されると（-dオプションが指定されているため）、プロンプトが返されます。

環境変数MYPASSからパスワードの設定を除きます。

```
unset MYPASS
```

```
[opc@adb-free ~]$ unset MYPASS
[opc@adb-free ~]$
```

コンテナの稼働状況を確認します。

```
podman ps
```

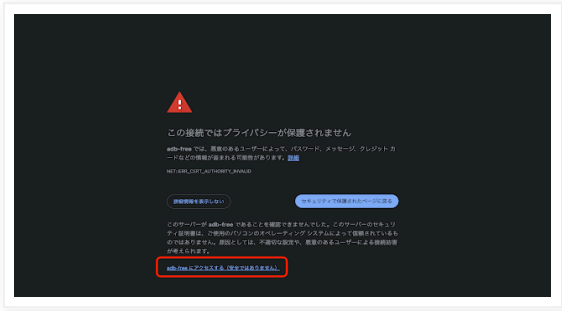
```
[opc@adb-free ~]$ podman ps
CONTAINER ID   IMAGE                                     COMMAND
CREATED       STATUS                                PORTS                                NAMES
28942510705f   container-registry.oracle.com/database/adb-free:latest  Up 3 minutes (healthy)  0.0.0.0:1521->1522/tcp, 0.0.0.0:1522->1522/tcp, 0.0.0.0:8443->8443/tcp, 0.0.0.0:27017->27017/tcp  adb-free
[opc@adb-free ~]$
```

healthyであれば、コンテナの起動が完了しています。

手元のブラウザからコンテナadb-freeに接続します。

```
https://adb-free/ords/
```

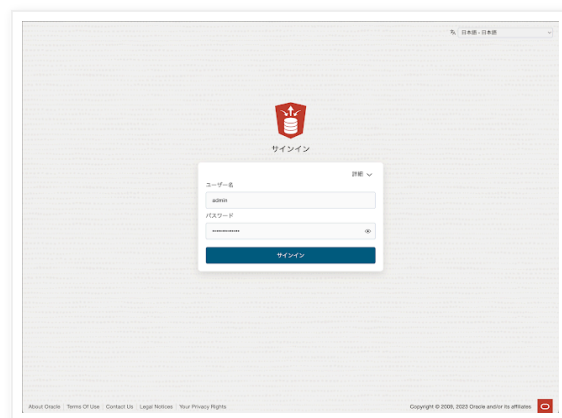
自己証明証明書を使ってHTTPSにしているため、証明書の検証に関するエラーが発生します。詳細情報を表示させ、エラーを無視して接続します。



ORDSのランディング・ページが開きます。



SQL Developer Webへ接続してみます。ユーザー名は**ADMIN**、パスワードはコンテナ**adb-free**作成時に**ADMIN_PASSWORD**として指定した値（今回の手順では環境変数**MYPASS**に設定した値）です。

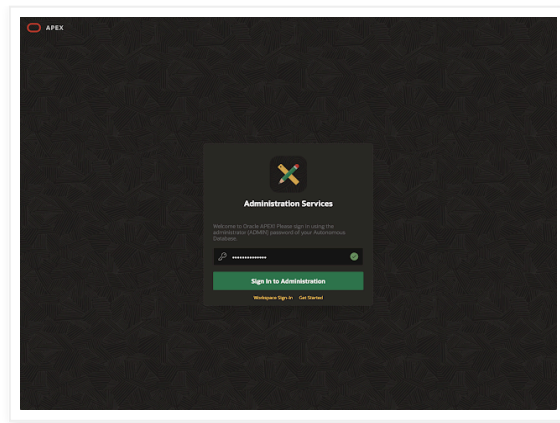


SQL Developer Webの画面が開きます。

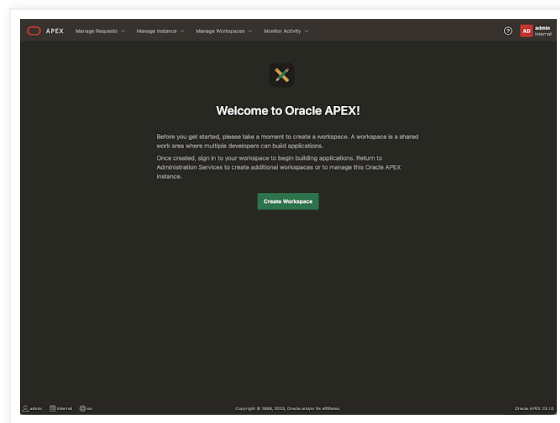


APEXに接続すると、管理者パスワードの入力画面が表示されます。

こちらのパスワードもコンテナ**adb-free**作成時に**ADMIN_PASSWORD**として指定した値です。



管理サービスにサインインするとワークスペースが存在しないため、Welcome to Oracle APEX!の画面が表示されます。これらの動作は、Autonomous Databaseとまったく同じです。



ブラウザからの操作はできるようになりました。

これから手元のPCで実行しているOracle SQL Developer Extension for VS Codeから、このデータベースに接続します。

コンテナadb-freeに含まれているウォレットを、ユーザーopcのホーム・ディレクトリ以下にコピーします。

```
podman cp adb-free:/u01/app/oracle/wallets/tls_wallet tls_wallet
```

ウォレット内のtnsnames.oraに記載されているlocalhostをadb-freeに置き換えます。

```
sed -i 's/localhost/adb-free/g' tls_wallet/tnsnames.ora
```

```
[opc@adb-free ~]$ sed -i 's/localhost/adb-free/g' tls_wallet/tnsnames.ora
```

ウォレットに含まれるファイルをZIPに固めます。

```
cd tls_wallet
zip -r ../wallet.zip *
```

```
[opc@adb-free ~]$ cd tls_wallet/
[opc@adb-free tls_wallet]$ zip -r ../wallet.zip *
  adding: adb_container.cert (deflated 24%)
  adding: cwallet.sso (stored 0%)
  adding: cwallet.sso.lck (stored 0%)
```



```

adding: ewallet.p12 (stored 0%)
adding: ewallet.p12.lck (stored 0%)
adding: ewallet.pem (deflated 25%)
adding: keystore.jks (stored 0%)
adding: ojdbc.properties (deflated 49%)
adding: README (stored 0%)
adding: sqlnet.ora (deflated 16%)
adding: tnsnames.ora (deflated 87%)
adding: truststore.jks (deflated 7%)
[opc@adb-free tls_wallet]$

```

コンピュータ・インスタンスでの作業は以上で終了です。

手元のPCに作成した**wallet.zip**をダウンロードします。

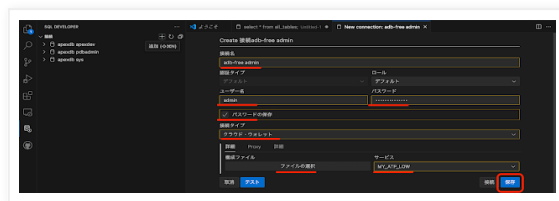
```
scp -i ssh-key-2024-03-08.key opc@adb-free:wallet.zip .
```

VS Codeを開き、Oracle SQL Developer Extension for VS Codeより接続を作成します。

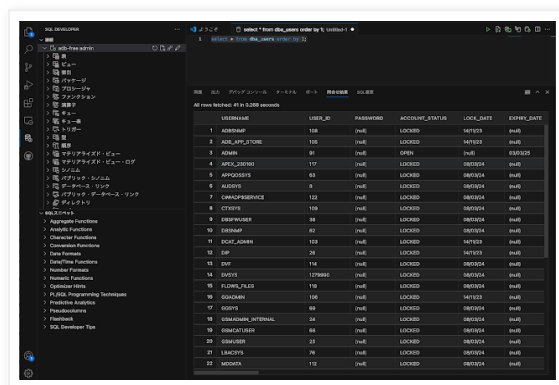
接続名は**adb-free admin**としています。ロールは**デフォルト**、ユーザー名は**admin**、パスワードはコンテナ**adb-free**作成時に**ADMIN_PASSWORD**として指定した値です。

接続タイプに**クラウド・ウォレット**を選択し、**ファイルの選択**をクリックして先ほどダウンロードした**wallet.zip**を選択します。サービスのドロップダウン・リストが有効になるので、**MY_ATP_LOW**を選択します。

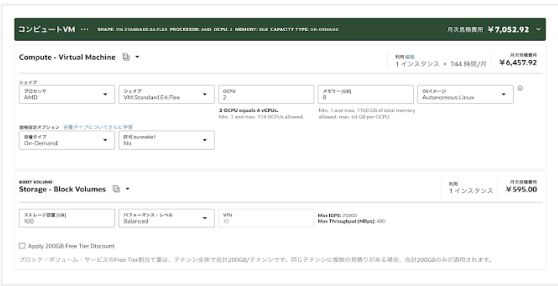
テストをクリックして設定を確認したのち、接続を**保存**します。



接続**adb-free admin**が作成されます。これからは、**SQLワークシート**や**SQLcl**より、コンテナ・イメージ**adb-free**のデータベースに接続して操作できます。



このシェイプ（2 OCPU、8GBメモリ）のコンピュータ・インスタンスを1ヶ月使用すると、だいたい7千円程度の費用がかかるようです。（2024年3月8日の時点の情報）



今回の記事は以上になります。

完

Yuji N. 時刻: 18:00

共有

ウェブ バージョンを表示

自己紹介

Yuji N.

日本オラクル株式会社に勤務していて、Oracle APEXのGroundbreaker Advocateを拝命しました。こちらの記事につきましては、免責事項の参照をお願いいたします。

詳細プロフィールを表示