

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18日月曜日

podmanとYAML形式の構成ファイルを使って色々なCloudの仮想マシンにOracle APEXを実装する

Oracle Container RegistryにはOracle Database 23c FreeとOracle REST Data Servicesのコンテナ・イメージが登録されています。Oracle REST Data Servicesのコンテナ・イメージにはOracle APEXの全言語版が含まれていて、初回起動時にデータベースにOracle APEXとOracle REST Data Servicesを構成するようにコンテナ・イメージが作られています（ただし、データベースにインストールされるのは英語のみ）。

コンテナ・イメージからOracle APEXが動作する環境を作成するには、Oracle Database 23c FreeとOracle REST Data Servicesの2つのコンテナが必要です。podmanにはYAML形式の構成ファイルを読み込み、podとそれに含まれるコンテナを作成する機能があります。この機能を使って、色々なCloudの仮想マシンにOracle APEXを実装してみます。

podmanに与えるYAMLファイルを生成するスクリプトは以下です。

```
#!/bin/sh

# Usage: install_apex.sh password [directory]
#
# password for sys password to install APEX into the database.
#
APEX_VERSION=23.2.0
#LANGUAGES=""
LANGUAGES="JAPANESE"
PASSWORD=${1}
DIR=${2:-`pwd`}
YAML=apex.yaml
SECRET_YAML=secret.yaml

# create directory for /opt/oracle/variables
mkdir -p ords_secrets
chmod 777 ords_secrets
# create ords_secrets/conn_string.txt
echo "CONN_STRING=sys/${PASSWORD}@localhost:1521/freepdb1" > ${DIR}/ords_secrets/conn_string.tx

# create directory for /etc/ords/config
mkdir -p ords_config
chmod 777 ords_config
```

```

# store sys password as kubernetes secret
encpassword=`echo -n ${PASSWORD} | base64`
cat <<EOF > ${SECRET_YAML}
apiVersion: v1
data:
  password: ${encpassword}
kind: Secret
metadata:
  creationTimestamp: null
  name: oracle_params
EOF
podman kube play ${SECRET_YAML}
rm ${SECRET_YAML}

# create language resource installer.
if [ -n "${LANGUAGES}" ]; then
  custdir=""
  mkdir -p ords_custom
  cat <<EOF > ords_custom/01_install_language.sh
#!/bin/sh
cd /opt/oracle/apex/${APEX_VERSION}
export PATH=/opt/oracle/sqlcl/bin:${PATH}
sql sys/${PASSWORD}@localhost/freepdb1 as sysdba @load_trans ${LANGUAGES} 2>&1 >> /tmp/install_
EOF
  chmod 777 ords_custom/01_install_language.sh
else
  custdir="# "
fi

# Create Kube YAML
cat <<EOF > ${YAML}
apiVersion: v1
kind: Pod
metadata:
  labels:
    app: apex
    name: apex
spec:
  containers:
  - env:
    - name: ORACLE_PWD
      valueFrom:
        secretKeyRef:
          name: oracle_params
          key: password
    image: container-registry.oracle.com/database/free:latest

```

```

name: db
ports:
- containerPort: 1521
  hostPort: 1521
- containerPort: 8181
  hostPort: 8181
- containerPort: 8443
  hostPort: 8443
securityContext:
  privileged: true
- image: container-registry.oracle.com/database/ords:latest
  name: ords
  securityContext:
    privileged: true
  depends_on:
    - db
  volumeMounts:
    - mountPath: /opt/oracle/variables
      name: home-oracle-ords_secrets-host-0
    - mountPath: /etc/ords/config
      name: home-oracle-ords_config-host-1
${custdir}    - mountPath: /ords-entrypoint.d
${custdir}    name: home-oracle-ords_custom-host-2
volumes:
- hostPath:
    path: ${DIR}/ords_secrets
    type: Directory
    name: home-oracle-ords_secrets-host-0
- hostPath:
    path: ${DIR}/ords_config
    type: Directory
    name: home-oracle-ords_config-host-1
${custdir} - hostPath:
${custdir}    path: ${DIR}/ords_custom
${custdir}    type: Directory
${custdir}    name: home-oracle-ords_custom-host-2
EOF

# apex.yaml is generated, create podman pod
# podman kube play apex.yaml
echo 'apex.yaml is generated. please run'
echo 'podman kube play apex.yaml'
echo 'To monitor the progress'
echo 'podman logs apex-db'
echo 'podman exec -it apex-ords tail -f /tmp/install_container.log'
# end

```

Oracle Cloud、Microsoft Azure、AWS、Google Cloudの仮想マシン上にOracle APEXの環境を作成します。

仮想マシンは少なくともスレッドが2つ（オラクルであれば 10CPU、その他は 2vCPU）、メモリは最低限 8GB は必要です。4GB だとスワップが発生しパフォーマンスが悪化します。

データベースのイメージにORDSをインストールする（同居させる - こちらの記事）と、メモリ要件を下げる可能性があります。

Oracle Cloud

仮想マシンは**イメージ**として**Oracle Linux 9**、**Shape**は **10CPU**、メモリ**8GB**を選択して作成しました。



仮想マシンのIPアドレスは、手元のPCにホスト名**oci-apex**で参照できるように登録しておきます。接続に使用する秘密キーのファイルは**oci-apex.key**として保存しています。

Oracle Cloudに作成した仮想マシンに接続します。

```
ssh -i oci-apex.key opc@oci-apex
```

podmanのインストールとfirewalldの設定を行います。

以下のコマンドを実行します。podmanを含むcontainer-toolsをインストールしたのち、firewalldにポート**8181**、**8443**、**1521**への接続許可とポート80への接続を8181へ、ポート443への接続を8443へ転送する設定を行なっています。今回はHTTPで構成するため、HTTPS関連の設定は実際は不要です。その後にグループ**oinstall**をGID:**54321**、ユーザー**oracle**をUID:**54321**で作成します。

```
#!/bin/sh
# Preparation script to setup Oracle APEX using container images
# on Oracle Container Registry.
#
# Install podman
# firewalld is ignored if already installed.
sudo dnf -y install container-tools firewalld
sudo systemctl enable firewalld
```

```
sudo systemctl start firewallld
# Setup firewallld to accept sqlnet, http and https connection.
sudo firewall-cmd --add-port=8181/tcp
sudo firewall-cmd --add-port=8443/tcp
sudo firewall-cmd --add-port=1521/tcp
sudo firewall-cmd --add-forward-port=port=443:proto=tcp:toport=8443
sudo firewall-cmd --add-forward-port=port=80:proto=tcp:toport=8181
sudo firewall-cmd --runtime-to-permanent
sudo firewall-cmd --list-all
# Create group oinstall and user oracle.
sudo groupadd -g 54321 oinstall
sudo useradd -u 54321 -g 54321 oracle
sudo loginctl enable-linger 54321
# end of preparation.
```

prepare-oracle-apex.sh hosted with ❤ by GitHub

[view raw](#)

作成したユーザー**oracle**で作業を行います。

sudo su - oracle

```
[opc@apex ~]$ sudo su - oracle
[oracle@apex ~]$
```

コンテナ・イメージが無ければpod作成時にダウンロードされますが、ポッドの作成に時間がかかります。作業の進捗が分かりにくくなるため、あらかじめコンテナ・イメージをダウンロードしておくことをお勧めします。

```
podman pull container-registry.oracle.com/database/free:latest
podman pull container-registry.oracle.com/database/ords:latest
```

記事に先頭に示したスクリプト**install_apex.sh**を仮想マシン上に配置します。

引数にデータベースのSYSのパスワードを指定します。

sh install_apex.sh パスワード

```
[oracle@apex ~]$ sh install_apex.sh パスワード
apex.yaml is generated. please run
podman kube play apex.yaml
To monitor the progress
podman logs apex-db
podman exec -it apex-ords tail -f /tmp/install_container.log
[oracle@apex ~]$
```

構成ファイルが**apex.yaml**として出力されます。

podmanにこのファイルを渡し、Oracle Database 23c FreeとOracle REST Data Servicesの2つのコンテナを含むポッド**apex**を作成します。Oracle APEXとOracle REST Data Servicesの構成は、Oracle REST Data Servicesのコンテナ起動時に自動的に行われます。

```
podman kube play apex.yaml
```

```
[oracle@apex ~]$ podman kube play apex.yaml
Pod:
de1381c11f990c81c2c348a5fe11654e4fb9963a373216d54ccc3083ab515ec6
Containers:
66bac5230fe5f745e5dd1a822b3728ffb9f1cdbebbf42c944fd22b25d369e2da
a305fdf18191eb184892fc9180cb88cd420b280a2d530b72715eb26df62c70f3

[oracle@apex ~]$
```

データベースのコンテナが作成された後に、Oracle APEXのインストールが開始します。

進捗は、以下のコマンドで確認できます。

```
podman exec -it apex-ords tail -f /tmp/install_container.log
```

大体30分弱でOracle APEXの構成が終了します。

```
--application/pages/page_00206
--application/end_environment
... elapsed: 4.8 sec
```

```
...done
```

```
Adjust instance settings
```

```
PL/SQL procedure successfully completed.
```

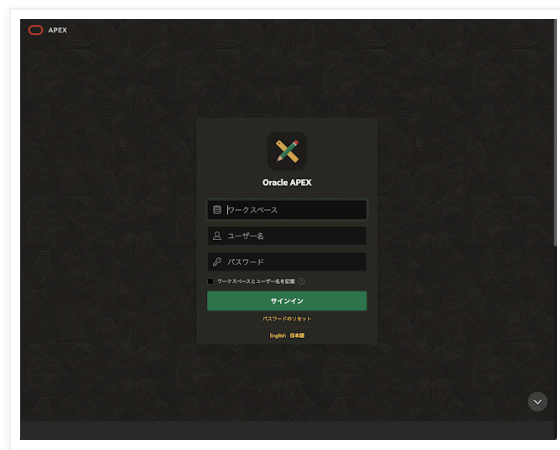
```
Disconnected from Oracle Database 23c Free Release 23.0.0.0.0 - Develop, Learn, and
Run for Free
Version 23.3.0.23.09
```

構成が終了すると、以下のURLからORDSのランディング・ページにアクセスできます。

<http://oci-apex/ords/>



Oracle APEXを開くと日本語のリソースもインストールされていることが確認できます。



Microsoft Azure

Azureの仮想マシンでは**イメージ**として**Oracle Linux 9**が選択できるので、それを選択します。OSが同じなので、Oracle Cloudと同じ手順でOracle APEXを構成できます。

サイズは**Standard_D2s_v3**（2 vcpu数、8GiBのメモリ）を選択します。

受信ポートを選択で、**HTTP (80)**, **HTTPS (443)**, **SSH (22)**を選択します。

以上の条件で仮想マシンを作成します。

仮想マシンのIPアドレスは、手元のPCにホスト名**azure-apex**で参照できるように登録しておきます。接続に使用する秘密キーのファイルは**apex_key.pem**として保存しています。

Microsoft Azureに作成した仮想マシンに接続します。

```
ssh -i apex_key.pem azureuser@azure-apex
```

これ以降の作業はOracle Cloudと同じです。

大体30分弱でOracle APEXの構成が終了しました。

構成が完了すると、以下のURLでORDSのランディング・ページに接続できます。

<http://azure-apex/ords/>

Google Cloud

Google Cloudでは**マシンの構成**に**e2-standard-2 (2vCPU、1コア、8GBメモリ)**を選択します。

マシンの構成

新登場: ストレージが最適化されたマシンシリーズ (プレビュー)
拡張ローカル SSD を備え、高密度ストレージ向けに最適化された新しい Z3 シリーズをお試しください

汎用 コンピューティング最適化 メモリ最適化 ストレージ最適化 **新着** GPU

費用と柔軟性を考慮して最適化された、一般的なワークロード向けのマシンタイプ

Series	説明	vCPUs	Memory	プラットフォーム
C3	一貫して高いパフォーマンス	4 - 176	8~1,408 GB	Intel Sapphire Rapids
C3D	一貫して高いパフォーマンス	4 - 360	8~2,880 GB	AMD Genoa
E2	低コスト、日常的なコンピューティング	0.25 - 32	1~128 GB	可用性に基づく
N2	料金とパフォーマンスのバランス	2 - 128	2~864 GB	Intel Cascade Lake
N2D	料金とパフォーマンスのバランス	2 - 224	2~896 GB	AMD EPYC
T2A	ワークロードのスケールアウト	1 - 48	4~192 GB	Ampere Altra Arm
T2D	ワークロードのスケールアウト	1 - 60	4~240 GB	AMD EPYC Milan
N1	料金とパフォーマンスのバランス	0.25 - 96	0.6~624 GB	Intel Skylake

マシンタイプ
ほとんどのワークロードに適した vCPU 数とメモリ量が事前に設定されているマシンタイプを選択します。または、ワークロードの特定のニーズに合わせてカスタムマシンを作成することもできます。詳細

プリセット カスタム

e2-standard-2 (2 vCPU, 1 コア, 8 GB メモリ)

	vCPU	Memory
	2 (1 コア)	8 GB

高度な構成

ブートディスクに**Rocky Linux 9 optimized for GCP**を選択しました。

ブートディスク

名前 apex

タイプ 新しいバランステキストディスク

サイズ 20 GB

ライセンスの種類 無料

イメージ **Rocky Linux 9 optimized for GCP**

変更

ファイアウォールのHTTPトラフィックとHTTPSトラフィックを許可するにチェックを入れます。

ファイアウォール

タグとファイアウォールルールを追加して、インターネットからの特定のネットワークトラフィックを許可します

☒ HTTP トラフィックを許可する

☒ HTTPS トラフィックを許可する

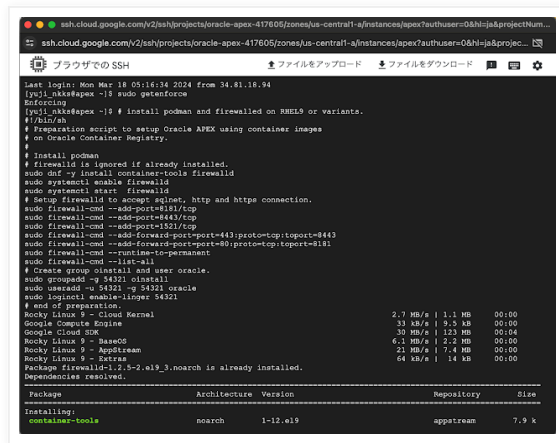
☐ ロードバランサーのヘルスチェックを許可する

以上の条件で仮想マシンを作成します。

Google Cloudでは秘密鍵の扱いが少々異なるため、Google Cloudの仮想マシンは**ブラウザウィンドウ**から操作することにしました。



Rocky Linux 9をブートイメージとして選択しているため、以降の作業はOracle Linux 9と同じです。



仮想マシンのIPアドレスは、手元のPCにホスト名**gcp-apex**で参照できるように登録しておきます。

大体30分弱でOracle APEXの構成が終了しました。

構成が完了すると、以下のURLでORDSのランディング・ページに接続できます。

<http://gcp-apex/ords/>

Amazon Lightsail

インスタンスイメージにCentOS 9 Streamを選択します。



サイズは8GBメモリ、2仮想CPUの\$40を選択します。

サイズを選択

次でソート: 月次料金 ▼

料金 (1か月あたり)	メモリ	CPU	ストレージ	転送
\$3.5	512 MB	2 仮想 CPU	20 GB SSD	1 TB
\$5	1 GB	2 仮想 CPU	40 GB SSD	2 TB
\$10	2 GB	2 仮想 CPU	60 GB SSD	5 TB
\$20	4 GB	2 仮想 CPU	80 GB SSD	4 TB
\$40	8 GB	2 仮想 CPU	160 GB SSD	5 TB
\$80	16 GB	4 仮想 CPU	320 GB SSD	6 TB
\$160	32 GB	8 仮想 CPU	640 GB SSD	7 TB

接続に使用する**SSHキーペアの変更**を行います。

作成されたインスタンスの接続に使えるSSHキーファイルを、**aws-key.pem**として保存しておきます。

SSH キーペアの変更 情報

インスタンスへの SSH に使用するキーペアを選択、作成、またはアップロードします。
[SSH キーの詳細はこちら](#)

新規作成 + 今すぐアップロード

デフォルトキー

ダウンロード

以上の条件でインスタンスを作成します。

ネットワーキングのIPv4ファイアウォールについて、HTTPはデフォルトで許可されているため、特に変更しません。プロトコルTCPのポート番号**1521**と**443**（HTTPS）は、必要に応じて追加します。

IPv4 ファイアウォール ?

インターネット、または特定の IPv4 アドレスまたは範囲にポートを開くためのルールを作成します。
[ファイアウォールルールの詳細](#)

+ ルールを追加

アプリケーション	プロトコル	ポートまたは範囲/コード	制限	操作
SSH	TCP	22	任意の IPv4 アドレス	✕
			Lightsail ブラウザ SSH/RDP	🔗
HTTP	TCP	80	任意の IPv4 アドレス	✕

仮想マシンのIPアドレスは、手元のPCにホスト名**aws-apex**で参照できるように登録しておきます。

Amazon Lightsailに作成した仮想マシンに接続します。

```
ssh -i aws-key.pem ec2-user@aws-apex
```

これ以降の作業はOracle Cloudと同じです。

大体40分弱でOracle APEXの構成が終了しました。

構成が完了すると、以下のURLでORDSのランディング・ページに接続できます。

<http://aws-apex/ords/>

色々なCloudの仮想マシンにOracle APEXを実装する手順の紹介は以上です。

RHEL9系列のOracle Linux 9、Rocky Linux 9、CentOS 9 Streamを選んでいるので、仮想マシンを準備した後の作業手順は基本同じです。

ポッド**apex**を作成するためにYAML形式の構成ファイルを使用しています。パスワードについてはKubernetes secretとして保存し、ポッドを作成するapex.yamlにパスワードの記述を含めないようにしています。ords_secrets/conn_string.txtおよびords_custom/01_install_language.shには、APEXおよび言語リソースをインストールするためにSYSのパスワードが記載されています。

ords_custom/01_install_language.shおよびords_secrets/conn_string.txt（このファイルはインストールが終了すると削除されます）は削除すればよいのですが、環境変数ORACLE_PWDに与えたパスワードはpodman generate kube apexを実行すると取り出すことができます。

そのため、安全のためにOracle APEXの環境構築後はsetPassword.shを呼び出して、SYS、SYSTEM、PDBADMINのパスワードの変更を変更しておくことが推奨されます。

```
podman exec apex-db ./setPassword.sh <your_password>
```

最後に、install_apex.shを実行すると以下のようなYAMLファイルが生成されています。

```
apiVersion: v1
kind: Pod
metadata:
  labels:
    app: apex
  name: apex
spec:
  containers:
  - env:
    - name: ORACLE_PWD
      valueFrom:
        secretKeyRef:
          name: oracle_params
          key: password
    image: container-registry.oracle.com/database/free:latest
    name: db
    ports:
    - containerPort: 1521
      hostPort: 1521
    - containerPort: 8181
      hostPort: 8181
    - containerPort: 8443
      hostPort: 8443
    securityContext:
      privileged: true
  - image: container-registry.oracle.com/database/ords:latest
    name: ords
    securityContext:
      privileged: true
    depends_on:
    - db
  volumeMounts:
```

```
- mountPath: /opt/oracle/variables
  name: home-oracle-ords_secrets-host-0
- mountPath: /etc/ords/config
  name: home-oracle-ords_config-host-1
- mountPath: /ords-entrypt.d
  name: home-oracle-ords_custom-host-2
volumes:
- hostPath:
    path: /home/oracle/ords_secrets
    type: Directory
  name: home-oracle-ords_secrets-host-0
- hostPath:
    path: /home/oracle/ords_config
    type: Directory
  name: home-oracle-ords_config-host-1
- hostPath:
    path: /home/oracle/ords_custom
    type: Directory
  name: home-oracle-ords_custom-host-2
```

apex.yaml hosted with ❤ by GitHub

[view raw](#)

完

Yuji N. 時刻: 17:26

共有

<

ホーム

>

[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](#)

Powered by Blogger.