

# 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。

2022年11月28日月曜日

## ロード・バランサを構成しORDSの可用性を確保する(2) - ロード・バランサの作成

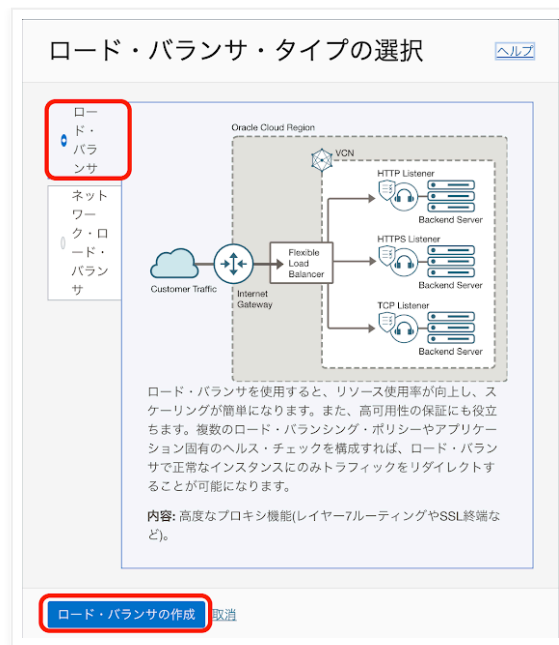
Customer Managed ORDSが実行されている2つのコンピュート・インスタンスのフロントとなるロード・バランサを作成します。バックエンドはHTTP接続をポート8080で待ち受けます。ロード・バランサはHTTPS接続を標準のポート443で待ち受けます。サーバー証明書として、OCIの証明書サービスで作成したものを使用します。

### ロード・バランサの作成

OCIのコンソールよりネットワーキングのロード・バランサを開きます。ロード・バランサの作成をクリックします。



ロード・バランサ・タイプの選択として、ロード・バランサを選択します。ロード・バランサの作成をクリックします。



ロード・バランサ名は任意です。今回は**lb-testserv**としました。**可視性タイプの選択はパブリック、パブリックIPアドレスの割当てはエフェメラルIPアドレス**を選択します。Always Free枠のロード・バランサのシェイプの**最小帯域幅**および**最大帯域幅**は、ともに**10Mbps**です。

**ネットワーキングの選択**に含まれる**仮想クラウド・ネットワーク**および**サブネット**として、あらかじめ作成済みのネットワークを指定します。

次へ進みます。



**ロード・バランシング・ポリシーの指定**として**重み付けラウンド・ロビン**を選択します。

**バックエンドの追加**をクリックしてバックエンドを追加します。

**バックエンドの選択**

ロード・バランサは、バックエンド・セット内のバックエンド・サーバーにトラフィックを分散します。バックエンド・セットは、ロード・バランシング・ポリシー、ヘルス・チェック・ポリシー、およびバックエンド・サーバーのリスト(コンピュータ・インスタンス)によって定義される論理エンティティです。

ロード・バランシング・ポリシーの指定

<b>重み付けラウンド・ロビン</b> このポリシーによって、受信トラフィックは、バックエンド・セット・リストの各サーバーに順番に分散されます。	<b>IPハッシュ</b> このポリシーにより、特定のクライアントからのリクエストが常に同じバックエンド・サーバーに転送されることが保証されます。	<b>最少接続</b> このポリシーによって、受信リクエスト・トラフィックは、アクティブ接続が最も少ないバックエンド・サーバーにルーティングされます。
---	--	--

バックエンド・サーバーの選択 オプション

バックエンド・サーバーが選択されていません。 **「バックエンドの追加」** をクリックして、使用可能なコンピュータ・インスタンスのリストからリソースを選択してください。一度に1つのコンパートメントからインスタンスを選択できます。1つのコンパートメントからインスタンスを追加した後で、 **「他のバックエンドの追加」** を選択して別のコンパートメントからインスタンスを追加できます。ロード・バランサの作成後にバックエンド・サーバーを追加することもできます。

**バックエンドの追加**

画面右にドロワーが開きます。ロード・バランサのバックエンドとなるコンピュータ・インスタンス（今回の例では**CMRODS1**と**CMORDS2**）を選択します。

選択したバックエンドの追加をクリックし、ドロワーを閉じます。

**バックエンドの追加**

バックエンド・サーバーのセットに含めるコンピュータ・インスタンスを指定します。

APEXのインスタンス [\(コンパートメントの変更\)](#)

<input checked="" type="checkbox"/>	名前	IPアドレス	OCID	可用性ドメイン
<input checked="" type="checkbox"/>	CMORDS1	10.0.0.186	...2jokuq <a href="#">表示</a> <a href="#">コピー</a>	XTPI:US-ASHBURN-AD-1
<input checked="" type="checkbox"/>	CMORDS2	10.0.0.103	...gdaxqq <a href="#">表示</a> <a href="#">コピー</a>	XTPI:US-ASHBURN-AD-1

2件を選択済 2アイテムを表示中 < 1 / 1 >

**選択したバックエンドの追加** 取消

バックエンド・サーバーが追加されます。このとき、**ポート**が80になっていたら**8080**へ変更します。ORDSは、ポート番号は8080でHTTP接続を待ち受けるように構成されています。

バックエンド・サーバーの選択 オプション

**他のバックエンドの追加**

名前	IPアドレス	OCID	可用性ドメイン	コンパートメント	ポート
CMORDS1	10.0.0.186	...2jokuq <a href="#">表示</a> <a href="#">コピー</a>	XTPI:US-ASHBURN-AD-1	APEX	<b>8080</b>
CMORDS2	10.0.0.103	...gdaxqq <a href="#">表示</a> <a href="#">コピー</a>	XTPI:US-ASHBURN-AD-1	APEX	<b>8080</b>

2アイテムを表示中

ヘルス・チェック・ポリシーの指定に含まれる**ポート**は8080、**ステータス・コード**は302、**URLパス(URI)**は/ordsとします。バックエンドの通信はHTTPなので、**SSLの使用にチェックは入れません**。

次へ進みます。

ヘルス・チェック・ポリシーの指定

ヘルス・チェックは、バックエンド・サーバーの可用性を確認するためのテストです。ヘルス・チェックはリクエストの場合も、接続の試行の場合もあります。指定された時間間隔に基づき、ロード・バランサによってヘルス・チェック・ポリシーが適用され、バックエンド・サーバーが継続的にモニターされます。

プロトコル: HTTP

ポート: R080

時間とリトリ オプション: 10000

タイムアウト(ミリ秒) オプション: 3000

再試行回数 オプション: 3

ステータス・コード オプション: 302

URL(パスのみ): localhost

レスポンス本文の正規表現 オプション

☐ SSLの使用

送信

リスナー名は任意です。今回はlsnr-testservとしました。トラフィックのタイプはHTTPS、モニターするポートはデフォルトの443のままとします。

SSL証明書の証明書リソースとして証明書サービス管理対象証明書を選択します。証明書として、先ほど作成したTestServを選択します。

次へ進みます。

ロード・バランサの作成

リスナーの構成

リスナー名: lsnr-testserv

リスナーで処理するトラフィックのタイプを指定します

タイプ: HTTPS (selected), HTTP, HTTP/2, TCP

リスナーでモニタリングするポートを指定します

ポート: 443

SSL証明書

証明書リソース: 証明書サービス管理対象証明書 (selected)

TestServ (selected)

送信

ロギングの管理は今回の検証の対象ではないため、エラー・ログ、アクセス・ログともにOFFにします。

以上で送信をクリックします。

ロード・バランサの作成

ロギングの管理

アクセス・ログとエラー・ログの有効化はオプションですが、推奨されます。これらのログを確認すると、バックエンド・サーバーでの問題の診断と修正に役立ちます。ロード・バランサ・ロギングの詳細は、こちらを参照してください。

ロギングはロード・バランサ・サービスのオプションです。ロギング機能を有効化すると、標準的な制限、制約およびレートが適用されます。

エラー・ログ

無効

アクセス・ログ

無効

送信

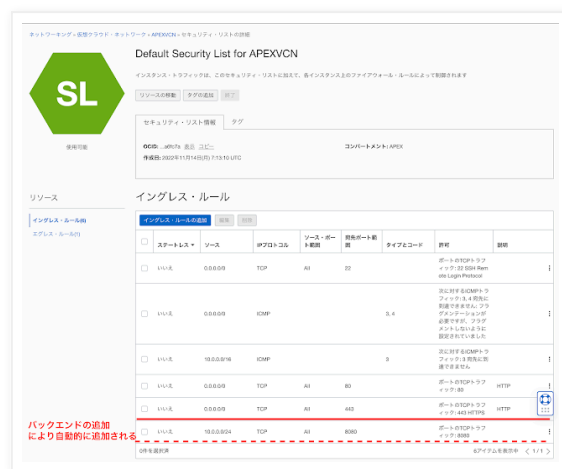
ロード・バランサが作成されます。ロード・バランサに割り当てられたIPアドレス(パブリック)を確認します。



以上でOracle APEXに接続可能になっています。

ロード・バランサへの接続を許可するため、ロード・バランサが配置されているネットワークにポート443への接続を許可するイングレス・ルールが設定済みであることを確認します。

ポート8080への接続を許可するイングレス・ルールは、ロード・バランサにバックエンドを追加したときに自動的に追加されます。



バックエンドの削除と追加を繰り返すと、同じイングレス・ルールが複数行になります。稼働に問題はありませんが、気になる場合は重複行を削除すると良いでしょう。

## 接続確認

作成した環境に接続します。/etc/hostsファイルまたはそれに準ずるファイルにパブリックIPアドレスとホスト名のペアを記述します。

IPアドレスはロード・バランサのパブリックIPアドレス、ホスト名は証明書の共通名またはSANとして設定したホスト名です。

129.\*\*\*.\*\*\*.218 testserv.\*\*\*\*\*.dev

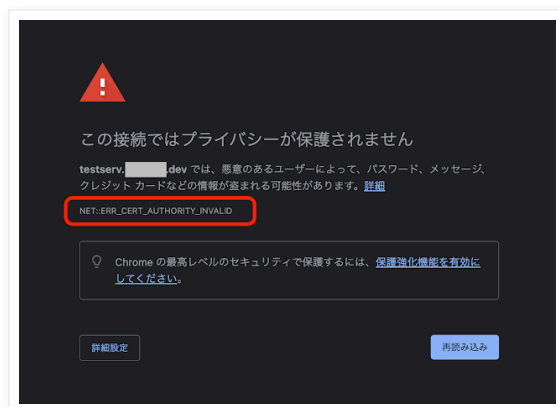
手元のPCに設定したホスト名にて、Oracle APEXの環境に接続します。

サーバー証明書の署名を行なっているCAは、プライベートCAであるため証明書に関するエラーが発生します。対応方法は、自己署名証明書を使用してHTTPS化したときと概ね同じです。

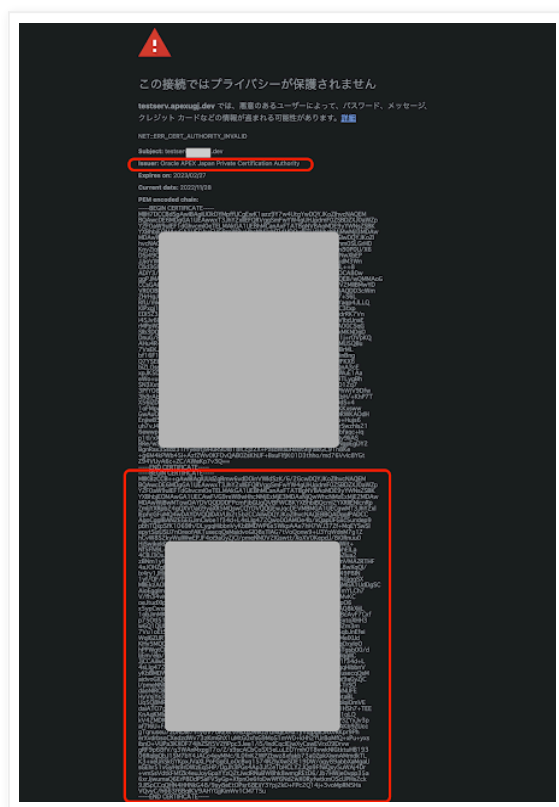
## Oracle APEXの環境作成(10) - 自己署名証明書によるHTTPS化

<http://apexugj.blogspot.com/2022/11/building-oracle-apex-environment-10-https.html>

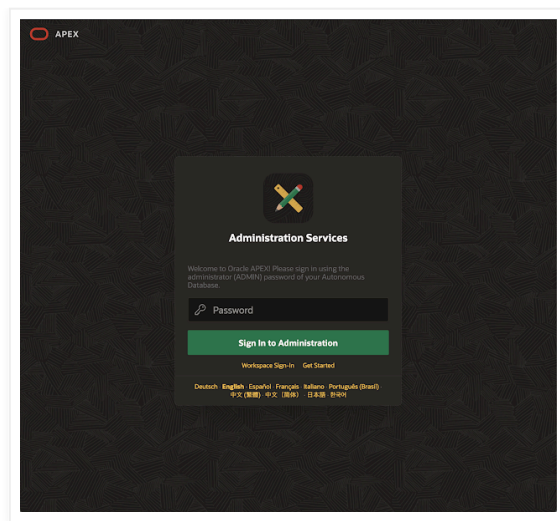
エラーが発生しているサーバー証明書を表示します。



自己署名証明のときと異なり複数の証明書が表示されます。一般に、後に表示されている証明書が上位の認証局の証明書になるため、その証明書をコピーしてPCに信頼できる証明書として登録します。



以上で、Oracle APEXの環境に接続できるようになります。



自己署名証明書の場合と異なり、一度、プライベートCAの証明書を登録すると、そのCAで署名されたサーバー証明書はすべて正しいサーバー証明書として扱われます。そのため、証明書をPCに登録する作業は、サーバーの数によらず一度で済みます。

完

Yuji N. 時刻: 16:40

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.