

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6月14日 火曜日

Okta Customer Identityを使ってAPEXアプリをSAMLで認証する

Okta Customer Identityのトライアル環境を使って、Oracle APEXのアプリケーションをSAMLで認証させてみました。

Okta Customer Identityのトライアル環境の取得については、Oktaからの情報を参照してください。

Oracle APEX側の環境は以下の2種類で実施します。

- Oracle APEX 22.1 + ORDS 22.1 + Oracle Database XE 21.3の環境。
- Oracle APEX 21.2 + Customer Managed ORDS 21.4.3 + Autonomous Databaseの環境。

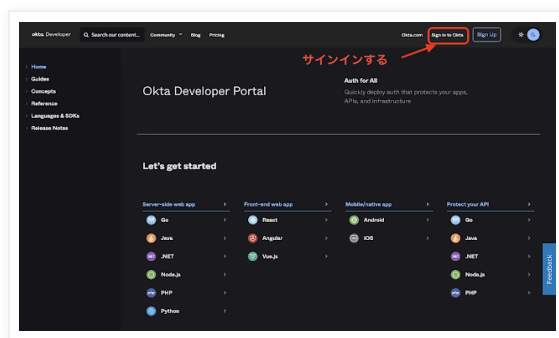
ただ、どちらの環境でも手順はほぼ同じなので、違う点だけを説明に加えます。

以下、作業手順になります。

アプリケーションの作成

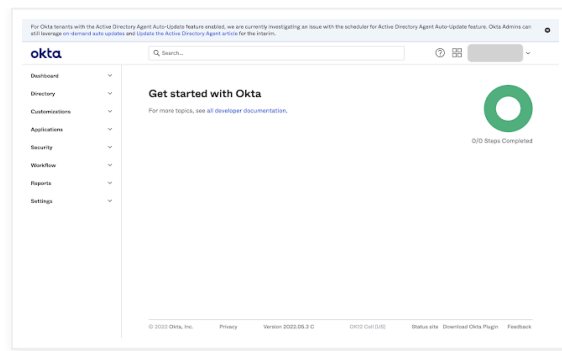
トライアル・アカウントを取得したのち、Okta Developer Portalにアクセスします。

<https://developer.okta.com>



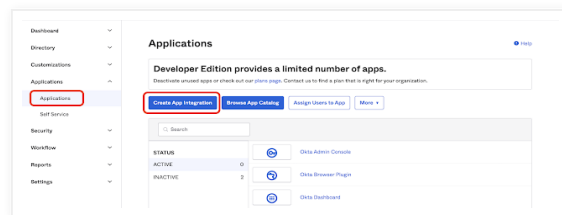
取得済みのトライアル・アカウントでサインインします。

Get startedの画面が開きます。



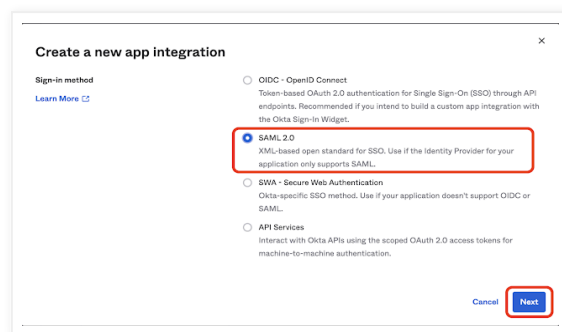
左のナビゲーション・メニューの**Applications**以下の**Applications**を開き、ApplicationとしてOracle APEXのインスタンスを追加します。

Create App Integrationをクリックします。



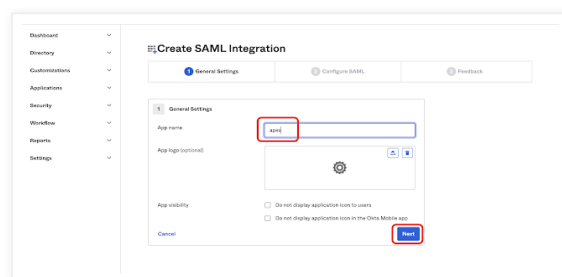
Create a new app integrationのダイアログが開くので、**SAML 2.0**を選択します。

Nextをクリックします。



App nameは**apex**とします。SAML認証を検証することが目的なので、それ以外は特に設定しません。

Nextをクリックします。



SAML Settingsの画面が開きます。

GeneralのSingle Sign on URLとAudience URI(SP Entity ID)として、Oracle APEX側のSAMLコールバックのURLを設定します。APEXインスタンスの設定としては、APEX属性の発行者になります。

今回の例では、以下のURLを設定します。

`https://test.mydomain.dev/ords/xepdb1/apex_authentication.saml_callback`

Autonomous DatabaseとCustomer Managed ORDSで構成している場合は、PDBのパスはつかないので、以下のようなURLになります。

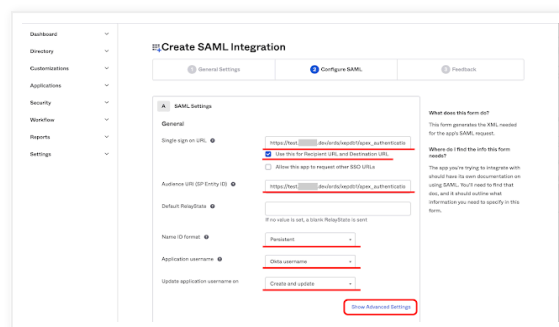
`https://test.mydomain.dev/ords/apex_authentication.saml_callback`

Use this for Recipient URL and Destination URLにチェックを入れます。チェックを外すと、それぞれのURLを個別に設定できるようになります。

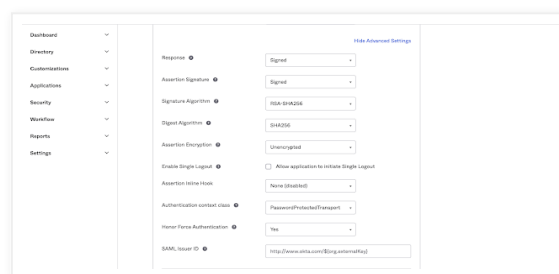
Default RelayStateは空白のまま、Name ID formatにはPersistentを選択します。これは、Oracle APEX側の名前IDフォーマットのデフォルトがPersistentなので、それに合わせています。

Application usernameはOkta username、Update application username onは作成と更新とします。

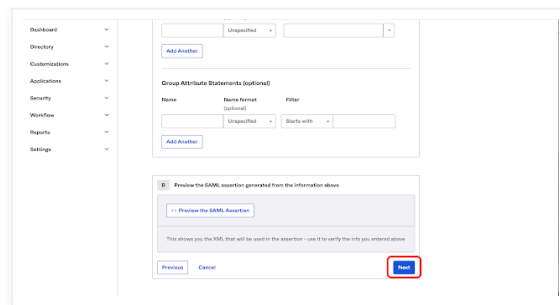
以上を設定し、Show Advanced Settingsをクリックします。



Assertion EncryptionがUnencryptedであることを確認します。ここでは、特に変更は必要ありません。

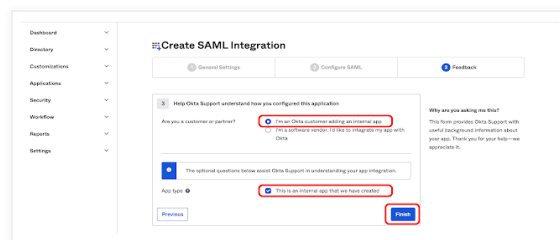


画面下までスクロールし、Nextをクリックします。

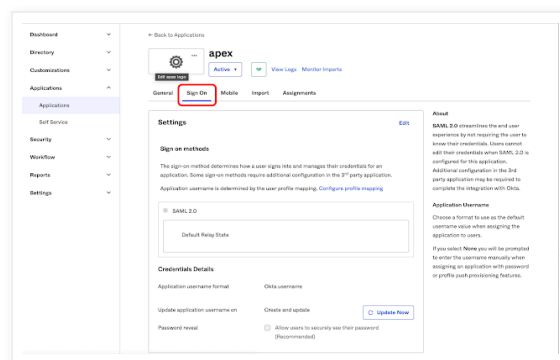


Are you a customer or partner?の質問にたいして、I'm an Okta customer adding an internal appを選択します。This is an internal app that we have createdには、チェックを入れます。

以上の設定で、Finishをクリックします。

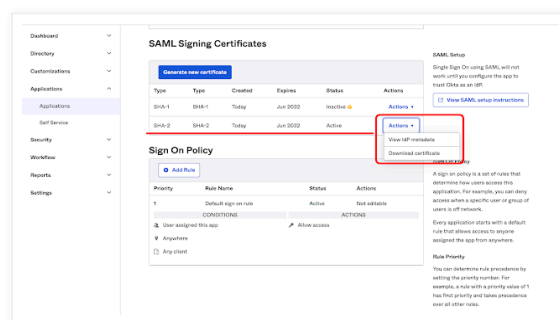


アプリケーションとしてapexが作成されます。Sign Onタブを開きます。



画面を下にスクロールし、SAML Signing Certificatesのセクションを表示します。

StatusがActiveなCertificateのActionsより、Download certificateの実行とView IdP metadataの実行を行います。



Download certificateを実行すると、okta.certというファイル名で、Oracle APEX側に登録する証明書がダウンロードされます。これは、Oracle APEX側のプロバイダ属性の署名証明書になります。

View IdP metadataより、entityIDとSingleSignOnServiceのLocationとなっているURLを確認します。entityIDはプロバイダ属性の発行者、SingleSignOnServiceのLocationはサインインURLになります。

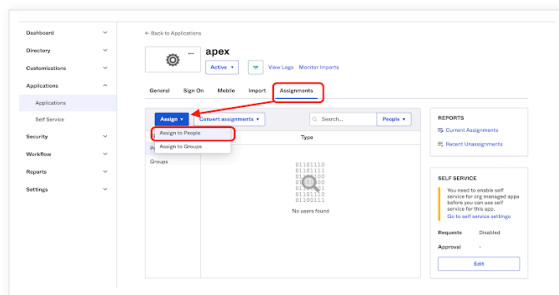


以上で、アプリケーションの作成は完了です。

Oktaでの最低限の設定

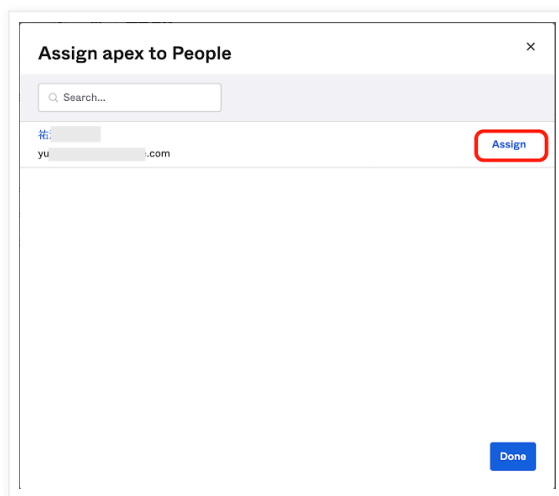
このアプリケーションにユーザーをアサインします。

Assignmentsを開き、AssignからAssign to peopleを実行します。



Assign apex to Peopleのダイアログが開きます。

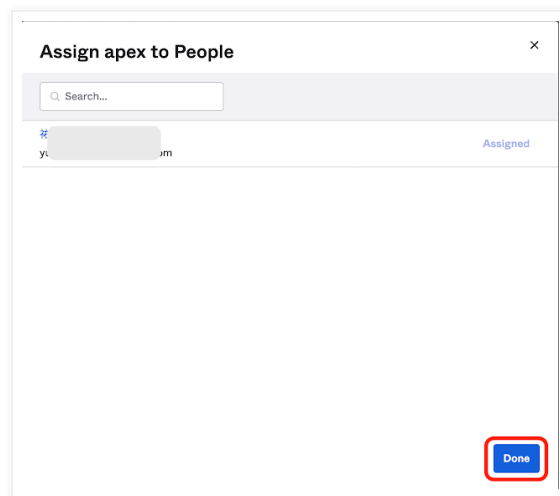
現在Oktaのツールにアクセスしているユーザーは表示されるはずなので、その人（通常は自分自身）をAssignします。



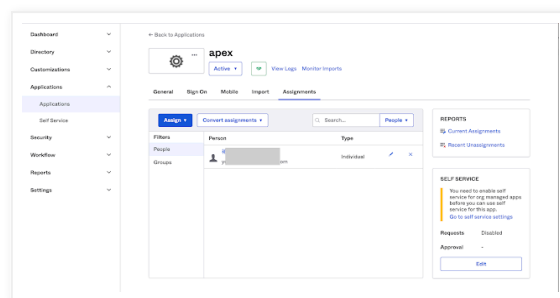
User Nameを確認し、Save and Go Backをクリックします。



元のダイアログに戻るので、**Done**を実行します。



ユーザーがアプリケーションにアサインされました。このユーザーでAPEXアプリケーションにサインインすることができます。



APEXでの認証スキームの設定

Oracle APEXのSAML認証の設定では、**内部およびワークスペース・アプリケーション用のSAML: APEX属性として証明書と秘密キー**の設定が必須となっています。**Oktaの設定では、これらの設定はAssertion EncryptionがEncryptedの場合のみ必要となっています。**

つまりOktaを使ったSAML認証では不要な情報ですが、Oracle APEXでは入力が必要になっているため、登録するための秘密キーと証明書をopensslを使って生成します。

最初にRSA暗号で使うキーペアを生成します。ファイル名は**private.pem**とします。

openssl genrsa -out private.pem 2048

```
% openssl genrsa -out private.pem 2048
Generating RSA private key, 2048 bit long modulus
.....+
++
```

```
.....+++
e is 65537 (0x10001)
%
```

自己署名証明書を生成するためのCSR（証明書署名要求）を作成します。

```
openssl req -new -key private.pem -out test.csr
```

```
% openssl req -new -key private.pem -out test.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) []:
State or Province Name (full name) []:
Locality Name (eg, city) []:
Organization Name (eg, company) []:
Organizational Unit Name (eg, section) []:
Common Name (eg, fully qualified host name) []: test.mydomain.dev
Email Address []:
```

```
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
%
```

自己署名証明書を生成します。バージョン3の証明書を生成するため、以下の1行を記述したファイル**v3.ext**を作成しておきます。

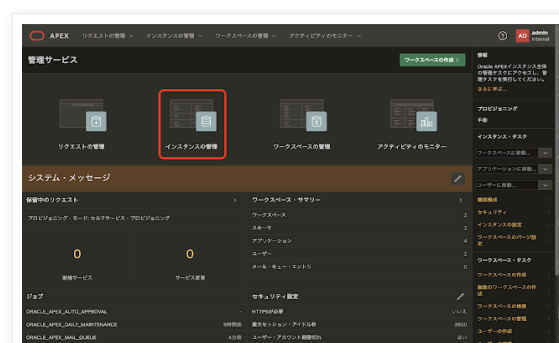
```
keyUsage = digitalSignature,keyEncipherment
```

```
openssl x509 -req -days 3650 -signkey private.pem -in test.csr -sha256 -extfile v3.ext -out
cert-test.pem
```

```
% openssl x509 -req -days 3650 -signkey private.pem -in test.csr -sha256 -extfile
v3.ext -out cert-test.pem
Signature ok
subject=/CN=test.mydomain.dev
Getting Private key
%
```

以上で、証明書と秘密キーの準備は完了です。

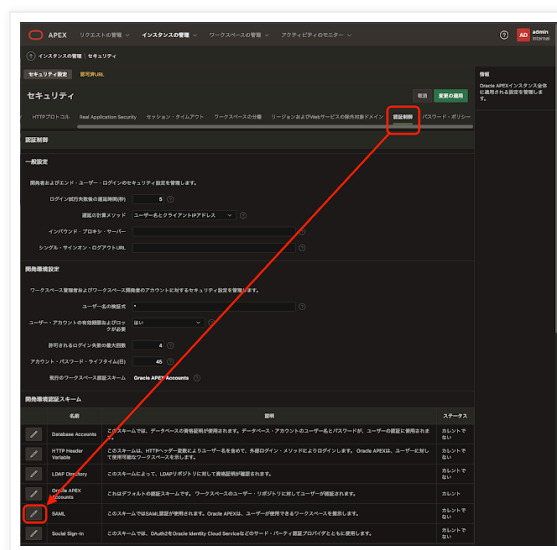
APEXの**管理サービス**に接続し、**インスタンスの管理**を開きます。



インスタンスの設定のセキュリティを開きます。



認証制御タブを選択し、開発環境認証スキームに含まれるSAMLを開きます。



内部およびワークスペース・アプリケーション用のSAML: APEX属性のアプリケーションのSAMLの有効化をONにします。

APEX属性のユーザー名属性、名前IDフォーマット、発行者は未指定のままにします。OktaにApplicationを作成する際に、APEXのデフォルトに合わせた設定を行なっています。

証明書には先ほどopensslを使って作成した証明書（手順通りであればcert-test.pem）を貼り付けます。秘密キーも貼り付けます（手順通りであればprivate.pem）。

内部およびワークスペース・アプリケーション用のSAML: アイデンティティ・プロバイダ属性の発行者はIdP metadataのentityID、サインインURLにはSigleSignInServiceのLocationを入力します。

署名証明書として、Oktaからダウンロードした証明書（okta.certという名前でダウンロードしています）を貼り付けます。

以上を設定し、変更の適用を行います。



マニュアルの[こちらに記載](#)があるように、ORDSでクロス・オリジン・リソース共有を行うには、明示的な許可が必要です。そのため、パラメータ**security.externalSessionTrustedOrigins**に設定を追加します。

Oktaを使用する際は、IdP metadataの**SingleSignOnService**の**Location**のURLの**ホスト部分**を**security.externalSessionTrustedOrigins**として設定します。

ORDS 22.1以降では、以下のコマンドで設定します。ordsコマンドの位置や構成ディレクトリの位置は、それぞれのインストールによって変わります。

```
/usr/local/bin/ords --config /etc/ords/config config set security.externalSessionTrustedOrigins https://dev-010101010.okta.com
```

```
[oracle@apex ~]$ /usr/local/bin/ords --config /etc/ords/config config set security.externalSessionTrustedOrigins https://dev-010101010.okta.com
```

```
ORDS: Release 22.1 Production on Tue Jun 14 07:32:49 2022
```

```
Copyright (c) 2010, 2022, Oracle.
```

```
Configuration:
/etc/ords/config/
```

```
The global setting named: security.externalSessionTrustedOrigins was set to:
https://dev-010101010.okta.com
[oracle@apex ~]$
```

ORDS 21.xまでであれば、実行するコマンドは以下になります。

```
java -jar ords.war set-property security.externalSessionTrustedOrigins https://dev-010101010.okta.com
```

```
[oracle@ords ords]$ java -jar ords.war set-property security.externalSessionTrustedOrigins https://dev-010101010.okta.com
2022-06-14T08:04:55.095Z INFO Modified:
/opt/oracle/ords/conf/ords/defaults.xml, setting:
security.externalSessionTrustedOrigins = https://dev-010101010.okta.com
[oracle@ords ords]$
```

または、構成ファイルの**defaults.xml**に以下の記述を追加します。

```
<entry key="security.externalSessionTrustedOrigins">https://dev-010101010.okta.com</entry>
```

設定変更を反映するには、ORDSを再起動する必要があります。

SAMLサインインの確認

作成したAPEXアプリケーションに接続し、SAMLによるサインインを確認します。

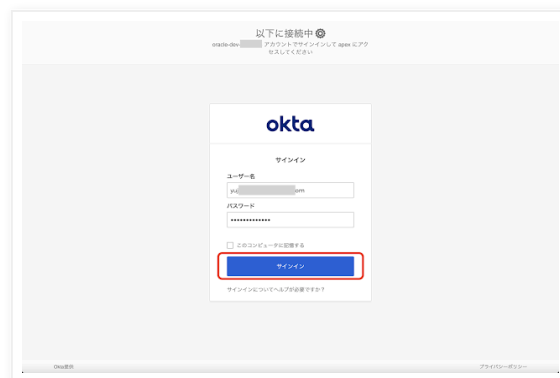
https://ホスト名/ords/PDB名/r/ワークスペース名/samltest/home

今回の例では、以下のURLにアクセスします。

https://test.mydomain.dev/ords/xepdb1/r/apexdev/samltest/home

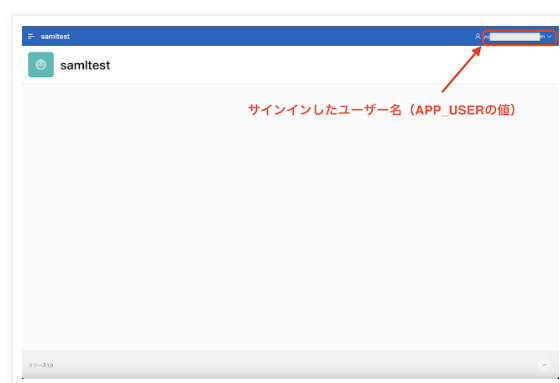
Oktaでのサインイン画面が表示されます。

ユーザー名とパスワードを入力し、サインインを実行します。



ユーザー名、パスワードが正しければ、サインインに成功し、何もないホーム・ページが表示されます。

右上にサインイン時に設定されたユーザー名が表示されます。Oktaのサインインに使用したユーザー名がAPEXのユーザー名になっています。メニュー・バーに表示されるユーザー名は小文字に変換されているので、APP_USER自体の値は大文字です。



以上で、Okta Customer Identityを使ってAPEXアプリをSAMLで認証する手順の紹介は終了です。

完

Yuji N. 時刻: 17:09

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](#)

Powered by Blogger.
