

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。

2023年12月19日 火曜日

ARNを使ったクリデンシャルを作成してAPEXのワークスペースからAWS S3のバケットにアクセスする

Oracle Autonomous Database (Always Freeを使います) に、AWS IAMで作成したロールのARNs (Amazon Resource Names) を使ったクリデンシャルを作成して、AWS S3にアクセスしてみます。

S3のバケットは、以下の記事に書かれている手順で作成したmyoraclefreebucketを使います。ポリシーも同じ記事に手順があるポリシーOracleFreeS3Policyを使います。

Oracle Database 23c FreeにDBMS_CLOUDパッケージを入れてAmazon S3にアクセスする

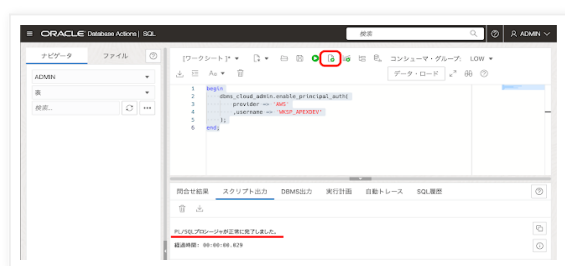
Autonomous DatabaseにユーザーADMINでアクセスし、SQL Developer WebのSQLワークシートを開きます。



APEXのワークスペースはAPEXDEV、ワークスペース・スキーマはWKSP_APEXDEVとして作成済みとします。

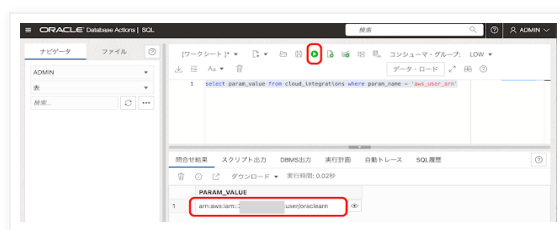
ワークスペース・スキーマWKSP_APEXDEVで、AWSのARNを使った認証を有効にします。

```
begin
  dbms_cloud_admin.enable_principal_auth(
    provider => 'AWS'
    ,username => 'WKSP_APEXDEV'
  );
end;
```



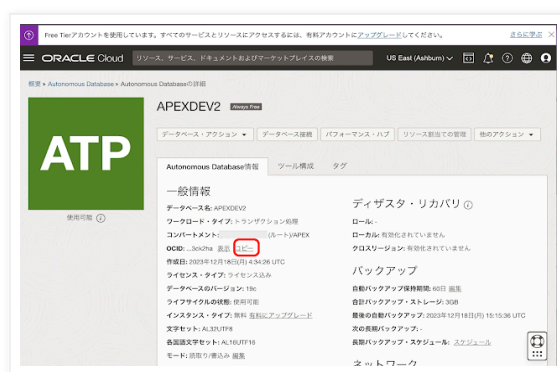
ビュー**CLOUD_INTEGRATIONS**を検索し、AWSのIAMで作成するロールに**アカウントID**として設定する値を取得します。

```
select param_value from cloud_integrations where param_name = 'aws_user_arn'
```

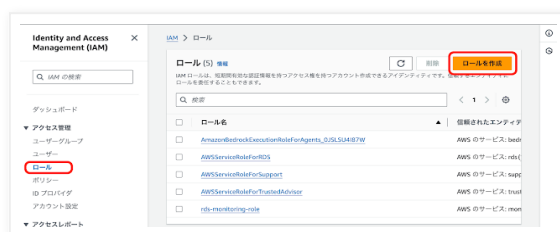


APEXからの利用なので、AWSのIAMのロールの**外部ID**に**データベースのOCID**を設定することになります。未確認ですが、**テナントのOCID**や**コンパートメントのOCID**を**外部ID**とすることもできるようです。

Autonomous Databaseの情報ページを開き、**OCID**をコピーします。



AWSコンソールを開きIAMのロールより、**ロール**を作成します。

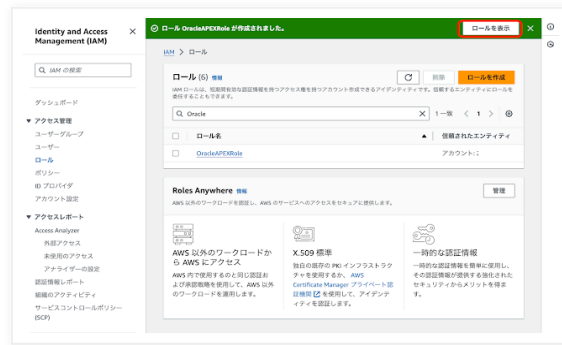


信頼されたエンティティタイプとして**AWSアカウント**を選択します。**AWSアカウント**として別の**AWSアカウント**を選択し、**アカウントID**にビュー**CLOUD_INTEGRATIONS**に**PARAM_NAME**が**aws_user_arn**の**PARAM_VALUE**として得られた値の**12桁の数値の部分**を設定します。

オプションの外部IDを要求するに**チェック**を入れ、**外部ID**として**データベースのOCID**を設定します。

次へ進みます。

ロール**OracleAPEXRole**が作成されます。ロールの**ARN**を取得するため、作成された**ロールを表示**します。



作成されたロール**OracleAPEXRole**の**ARN**を**コピー**します。

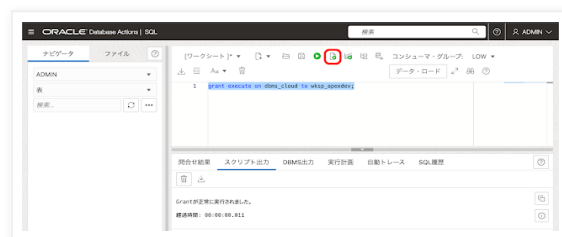


以上でAWS側の設定は完了です。

Autonomous Databaseに戻ります。

パッケージDBMS_CLOUDの実行権限を、APEXのワークスペース・スキーマWKSP_APEXDEVに与えます。デフォルトの状態では、APEXのワークスペース・スキーマはDBMS_CLOUDの実行権限を持っていません。

grant execute on dbms_cloud to wksp_apexdev;

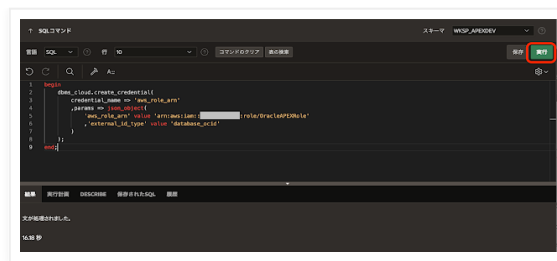


これからは、APEXの**SQLコマンド**から作業を行います。

AWSのロールのARNを使ったクリデンシャルを**aws_role_arn**という名前で作成します。

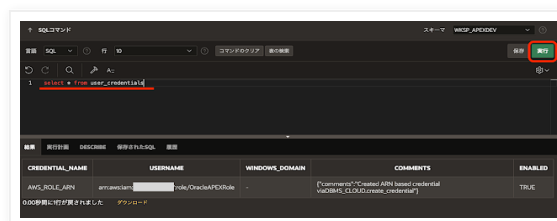
```
begin
  dbms_cloud.create_credential(
    credential_name => 'aws_role_arn'
    ,params => json_object(
      'aws_role_arn' value 'ロールOracleAPEXRoleのARN'
      ,'external_id_type' value 'database_ocid'
    )
  );
end;
```

作成が完了するまでにかなりの時間（10秒前後）かかります。



ビューUSER_CREDENTIALSを検索し、作成されたクリデンシャルを確認します。

```
select * from user_credentials
```

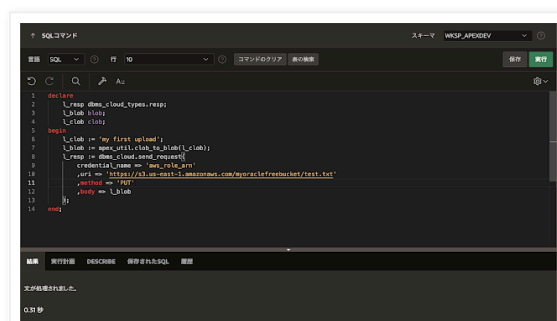


S3のバケットを操作し、クリデンシャルが正しく作成されたことを確認します。

これからの手順は、以前の記事と同じでクリデンシャルだけが異なります。

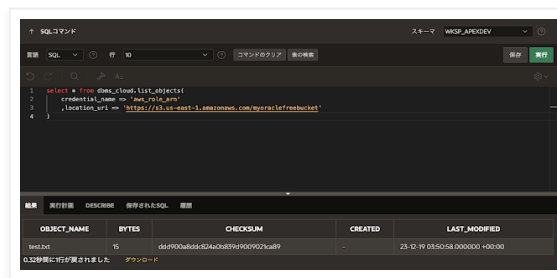
内容にmy first uploadと記載したファイルtest.txtを、S3のバケットにアップロードします。

```
declare
  _resp dbms_cloud_types.resp;
  _blob blob;
  _clob clob;
begin
  _clob := 'my first upload';
  _blob := apex_util.clob_to_blob(_clob);
  _resp := dbms_cloud.send_request(
    credential_name => 'aws_role_arn'
    ,uri => 'https://s3.us-east-1.amazonaws.com/myoraclefreebucket/test.txt'
    ,method => 'PUT'
    ,body => _blob
  );
end;
```



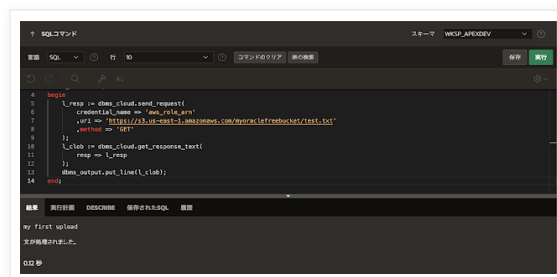
バケットの内容をリストします。test.txtがアップロードされていることが確認できます。

```
select * from dbms_cloud.list_objects(
  credential_name => 'aws_role_arn'
  ,location_uri => 'https://s3.us-east-1.amazonaws.com/myoraclefreebucket'
)
```



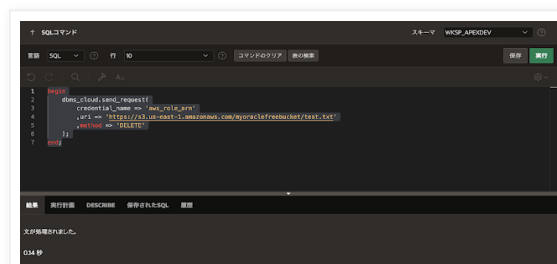
ファイル**test.txt**を取得します。ファイルの内容が**my first upload**として表示されます。

```
declare
  l_resp dbms_cloud_types.resp;
  l_clob clob;
begin
  l_resp := dbms_cloud.send_request(
    credential_name => 'aws_role_arn'
    ,uri => 'https://s3.us-east-1.amazonaws.com/myoraclefreebucket/test.txt'
    ,method => 'GET'
  );
  l_clob := dbms_cloud.get_response_text(
    resp => l_resp
  );
  dbms_output.put_line(l_clob);
end;
```

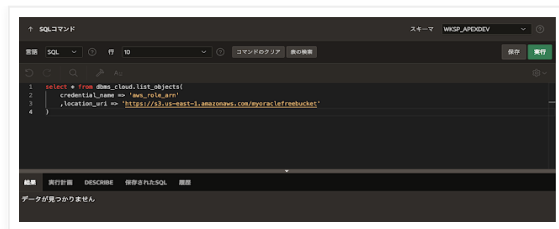


ファイル**test.txt**を削除します。

```
begin
  dbms_cloud.send_request(
    credential_name => 'aws_role_arn'
    ,uri => 'https://s3.us-east-1.amazonaws.com/myoraclefreebucket/test.txt'
    ,method => 'DELETE'
  );
end;
```



再度、バケット内のファイルをリストします。**データが見つかりません**と返されます。



今回の記事は以上になります。

Oracle APEXのアプリケーション作成の参考になれば幸いです。

完

Yuji N. 時刻: 13:01

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.