

演習 6 - OAuthセキュリティの実装

この演習では、OAuthセキュリティを実装してテストする方法を確認します。

演習 6 - 目的

この演習では、以下の内容を理解できます。

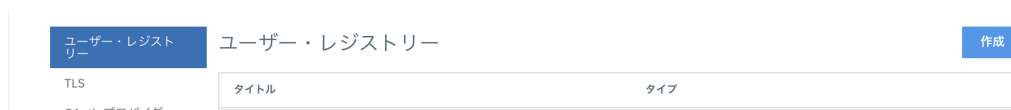
- ネイティブOAuthプロバイダーの定義方法
- OAuthで保護されるAPIのセキュリティ定義方法
- OAuthで保護されたAPIのテスト方法

6.1 - ネイティブOAuthプロバイダーの作成

1. API Managerにログインしていない場合には、ログインします。
2. まず、Native OAuthプロバイダーでの認証を行うURLを **認証ユーザー・レジストリー** として設定します。左のメニューから **リソース** を選択します。



3. **ユーザー・レジストリー** の画面で、**作成** をクリックします。



4. **認証 URL ユーザー・レジストリー** をクリックします。



ユーザー・レジストリーの作成

ユーザー・レジストリー・タイプの選択

**認証 URL ユーザー・レジストリー**
ユーザー認証の構成方法: 認証 URL

**LDAP ユーザー・レジストリー**
ユーザー認証の構成方法: LDAP プロバイダー

**ローカル・ユーザー・レジストリー**
ユーザー認証の構成方法: API Connect のローカル・ユーザー・レジストリー

5. 以下を入力して **保存** をクリックして保存します。

項目	入力値	備考
タイトル	AuthURL	認証URL名前
Display Name	AuthURL	表示名
URL	https://httpbin.org/basic-auth/user/pass	認証サービスのURL
TLSクライアント・プロファイル	TLSプロファイルなし	

認証 URL ユーザー・レジストリー

タイトル

AuthURL

名前

authurl

Display Name

AuthURL

要約 (オプション)

URL

<https://httpbin.org/basic-auth/user/pass>

TLS クライアント・プロファイル (オプション)

TLS プロファイルなし

☐ 大/小文字の区別

認証 URL ユーザー・レジストリーについて

認証 URL は、カスタム ID プロバイダーに対してユーザーを認証するためのシンプルな仕組みです。

[詳細情報](#)

キャンセル

保存



この認証サービスのURLは、Basic認証を行う外部サービスです。

- 次に、OAuthプロバイダーを設定します。OAuth プロバイダー を選択し、追加 をクリックして ネイティブ OAuth プロバイダー を選択します。

ユーザー・レジストリー

TLS

OAuth プロバイダー

OAuth プロバイダー

タイトル

タイプ

追加

ネイティブ OAuth プロバイダー

サード・パーティー OAuth プロバイダー

項目が見つかりません

- タイトル に oauthprovider と入力して 次へ をクリックします。

ネイティブ OAuth プロバイダー

タイトル

oauthprovider

名前

oauthprovider

説明 (オプション)

基本パス (オプション)

ゲートウェイ・タイプ

この OAuth プロバイダーに対するゲートウェイ・タイプを選択してください

☐ DataPower Gateway (v5 互換)

☒ DataPower API Gateway

キャンセル

次へ

- サポートされている権限付与タイプ フィールドで、リソース所有者 - パスワード にチェックを入れ、アクセス・コード のチェックを外し、次へ をクリックします。

構成

許可パス

/oauth2/authorize

トークン・パス

/oauth2/token

サポートされている権限付与タイプ

☐ 暗黙
☐ アプリケーション
☐ アクセス・コード
☒ リソース所有者 - パスワード
☐ リソース所有者 - JWT

サポートされるクライアント・タイプ

☒ 機密
☐ 公開

戻る キャンセル 次へ

9. スcope セクションで、上部の 名前 フィールドに details と入力し、説明 フィールドに Branch details と入力し、次へ をクリックします。

スコープ

この OAuth プロバイダーのスコープを追加します。

追加

名前	説明	削除
details	Branch details	

戻る キャンセル 次へ

10. リソース所有者パスワード付与 に AuthURL が選択されていることを確認して次へ をクリックします。

リソース所有者パスワード付与

認証

アプリケーション・ユーザーの認証に使用:

AuthURL

戻る キャンセル 次へ



前の手順で作成した、**認証URLユーザー・レジストリー** をOAuthプロバイダーの認証に利用するように設定しています。

11. サマリーページが表示されるので、下までスクロールして、**終了** をクリックします。

許可エンドポイント

資格情報の収集に使用する条件

基本認証

アプリケーション・ユーザーの認証に使用:

authurl

アプリケーション・ユーザーに次を使用して権限を与える

認証済み

リソース所有者パスワード付与

アプリケーション・ユーザーの認証に使用:

authurl

スコープ

名前	説明
details	Branch details

戻る

キャンセル

終了

12. **デバッグ応答ヘッダーの有効化** にチェックを入れて、**保存** をクリックします。

情報

構成

スコープ

ユーザー・セキュリティ

トークン

トークン管理

イントロスペクション

メタデータ

OpenID Connect

API エディター

ネイティブ OAuth プロバイダー

タイトル

oauthprovider

名前

oauthprovider

説明 (オプション)

ゲートウェイのバージョン

6000

基本パス (オプション)

/oauthprovider

☒ デバッグ応答ヘッダーの有効化

キャンセル

保存

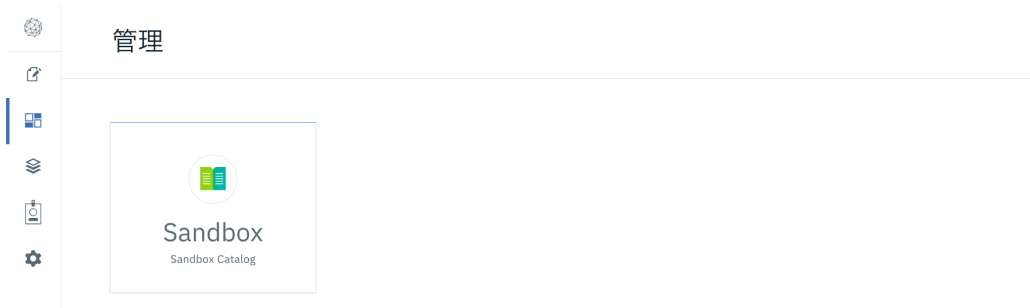
ネイティブ OAuth プロバイダーについて

ネイティブ OAuth プロバイダー・オブジェクトは、OAuth トークンの生成や検証などの OAuth 処理操作のための設定を提供します。OAuth プロバイダー・オブジェクトは、API を保護するための OAuth セキュリティ定義により参照できます。ネイティブ OAuth プロバイダーが使用される場合、OAuth 操作は API Connect によりネイティブに実行されます。すべての OAuth プロバイダー・オブジェクトにはバッキング API があります。ここでの構成により、API の Swagger 文書が自動的に更新されます。「API エディター」ページにナビゲートして、Swagger 文書を編集できます。公開された API で OAuth プロバイダー・オブジェクトを参照すると、バッキング API が自動的にゲートウェイで使用可能になります。

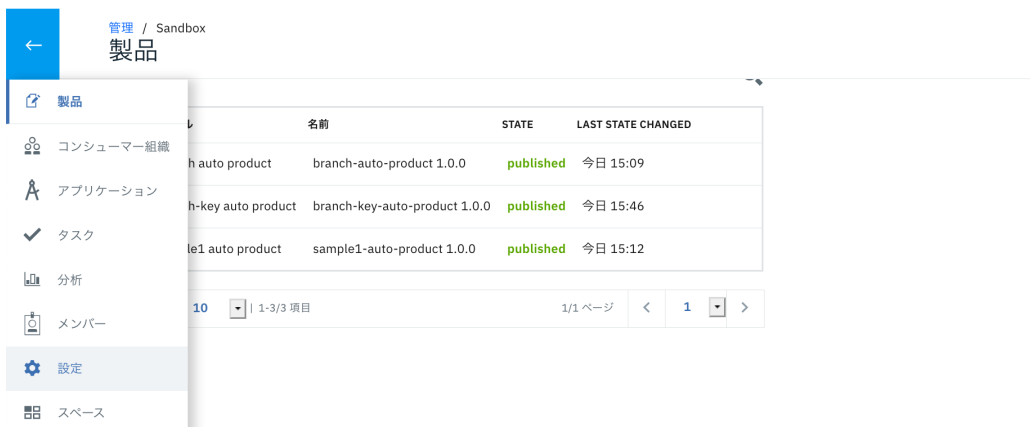
13. この認証URLとOAuthプロバイダーを利用するに、カタログで有効化します。API Managerの左のメニューから **管理** を選択します。



14. **Sandbox** を選択します。



15. 左側のカタログの管理メニューから **設定** を選択します。



16. **APIユーザー・レジストリー** を選択し、**編集** をクリックします。



17. 利用可能な **APIユーザー・レジストリー** が表示されるので、**AuthURL** にチェックを入れて **保存** をクリックします。

<input checked="" type="checkbox"/>	タイトル	タイプ	要約
<input checked="" type="checkbox"/>	AuthURL	認証 URL	

キャンセル
保存

18. 次に、**OAuthプロバイダー** を選択し、**編集** をクリックします。

概要
ゲートウェイ・サービス
ライフサイクル承認
ロール
オンボーディング
API ユーザー・レジストリー
OAuth プロバイダー
API エンドポイント

OAuth プロバイダー

API Manager 用に構成される OAuth プロバイダーを管理します

編集

タイトル	タイプ
 <p>項目が見つかりません</p>	

19. 利用可能なOAuthプロバイダーが表示されるので、チェックを入れて **保存** をクリックします。

管理 / Sandbox
OAuth プロバイダーの編集

<input checked="" type="checkbox"/>	タイトル	タイプ
<input checked="" type="checkbox"/>	oauthprovider	ネイティブ

キャンセル
保存

左上の  をクリックして、管理画面に戻ります。

6.2 - OAuthセキュリティの付加されたAPIの作成

次に既存のAPIにOAuthセキュリティを追加します。

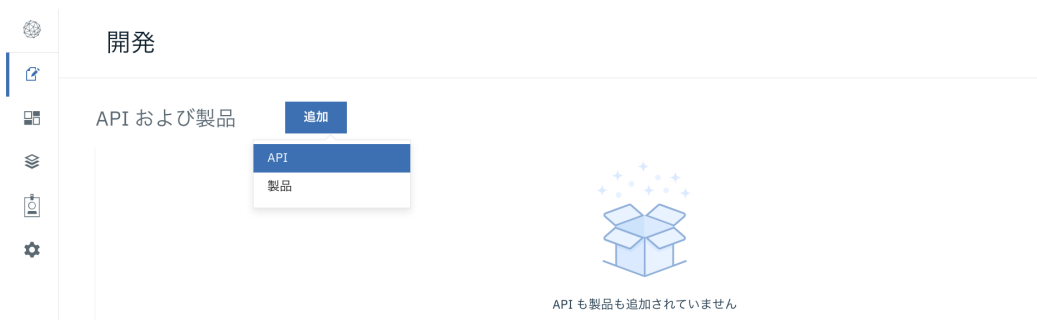
1. セキュリティを付加したAPIを作成するために、新たにAPIを作成します。演習3で利用した `findbranch.yaml` を編集して、インポートによりAPIを作成します。 `findbranch.yaml` をコピーして、ファイル名を `oauthapi.yaml` に変更して保存します。
2. `oauthapi.yaml` を開いて、3箇所の文字列 `findbranch` を `oauthapi` に変換してファイルを上書き保存します。

```
swagger: '2.0'
info:
  title: oauthapi
  x-ibm-name: oauthapi
  version: 2.0.0
schemes:
  - https
basePath: /oauthapi
security:
  - clientID: []
```

3. 演習3で行ったインポートによるAPIの作成と同じ手順でAPIを作成します。左のメニューから **開発** を選択し、開発メニューに進みます。



4. **開発** 画面で、**追加** メニューから **API** を選択します。



5. 既存のOpenAPI にチェックを入れて、**次へ** を選択します。

作成

ターゲット・サービスから

すべてのトラフィックをターゲット API またはサービス・エンドポイントにルーティングする REST プロキシを作成します

既存の OpenAPI サービスから

OpenAPI で記述されたターゲット・サービスに基づいて REST プロキシを作成します

既存の WSDL サービスから (SOAP プロキシ)

WSDL で記述されたターゲット・サービスに基づいて SOAP プロキシを作成します

既存の WSDL サービスから (REST プロキシ)

WSDL で記述されたターゲット・サービスに基づいて REST プロキシを作成します

新規 OpenAPI

パスと操作を定義して新しい REST プロキシを作成します

インポート

既存の OpenAPI

REST プロキシの既存の定義を使用します

キャンセル

次へ

6. 参照 ボタンをクリックして、作成した `oauthapi.yaml` ファイルを指定し、`次へ` をクリックします。

ファイルからインポート

インポート元となる API 定義ファイルを選択します

findbranch.yaml

X

YAML が正常に検証されました

X

キャンセル

次へ

既存の OpenAPI サービスからのインポート

REST プロキシの既存の定義を使用します

[詳細情報](#)

7. APIのアクティブ化 にチェックを入れて、 `次へ` をクリックします。

API のアクティブ化

この API は、次のオプションが有効な場合に呼び出すことができます。

☒ API のアクティブ化

戻る

キャンセル

次へ

既存の OpenAPI サービスからのインポート

REST プロキシの既存の定義を使用します

[詳細情報](#)

8. API定義がインポートされ、要約情報が表示されます。 **APIの編集** をクリックします。

要約

✓ OpenAPI 2.0 定義が生成されました

✓ API はオンラインです

API の基本 URL

API 内のすべての操作の URL はこの値で始まります。

https://example.com/api/v1/

API サブスクリプション

クライアント ID

クライアント・シークレット

既存の OpenAPI サービスからのインポート

REST プロキシの既存の定義を使用します

[詳細情報](#)

API の編集

9. **セキュリティ定義** をクリックして、 **追加** をクリックします。

API のセットアップ

セキュリティ定義

セキュリティ

パス

定義

プロパティ

セキュリティ定義

セキュリティ定義により、API キー検証、アプリケーション・ユーザー認証、OAuth を含む、API エンドポイントへのクライアント・アクセスが制御されます。 [詳細情報](#)

名前	タイプ	場所
clientID	apiKey	ヘッダー

追加

10. 以下のように入力し、 **保存** をクリックします。

項目	入力値
名前	oauth
タイプ	OAuth2
OAuth プロバイダー	oauthprovider
フロー	Resource Owner

API セキュリティー定義

名前
oauth

説明 (オプション)

タイプ
☐ API キー
☐ 基本
☒ OAuth2

OAuth プロバイダー
oauthprovider

フロー
Resource owner

トークン URL
https://\$(catalog.url)/oauthprovider/oauth2/token

拡張スコープ検査
スコープ

追加

名前	説明	削除
details	Branch details	

- 次に **セキュリティ** をクリックし、定義した **oauth** にチェックを入れ、スコープ **details** にチェックを入れて、**保存** をクリックします。

API のセットアップ

セキュリティ定義

セキュリティ

パス

定義

プロパティ

ターゲット・サービス

セキュリティ

ここで選択されたセキュリティ定義は API 全体に適用されますが、個別の操作についてオーバーライドできます。 [詳細](#)

追加

セキュリティ定義

<input checked="" type="checkbox"/> oauth	<input checked="" type="checkbox"/> details
<input checked="" type="checkbox"/> oauth2	
<input checked="" type="checkbox"/> clientID	
<input type="checkbox"/> apiKey	

以上でAPIへのOAuthセキュリティ定義の追加が完了しました。

6.3 - 製品プランの作成と公開

- 新規に製品を作成して作成したAPIを公開します。左のメニューから **開発** を選択し、開発メニューに進みます。

API Manager へようこそ
開始するには、オプションを選択してください

ホーム

開発

管理

リソース

メンバー

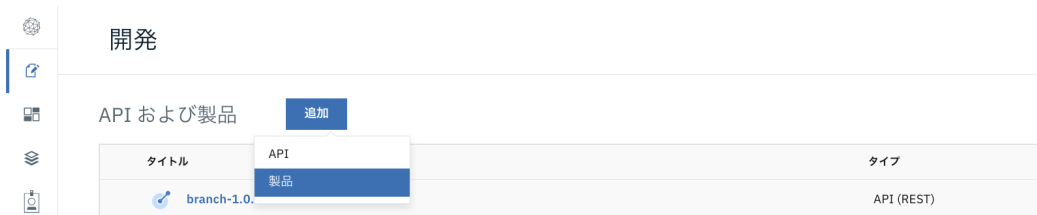
設定

API および製品の開発

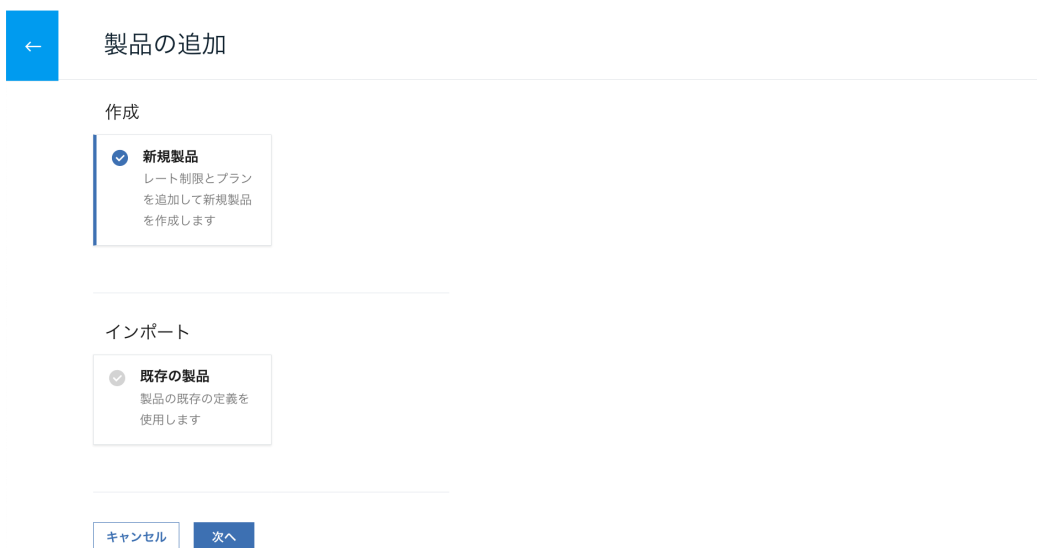
カタログの管理
アクティブな API とコンシューマーを管理し

リソースの
ユーザー・レジストリー、O

2. 開発画面で、追加メニューから製品を選択します。



3. 新規製品を選択し、次へをクリックします。



4. タイトルにoauth-test-productと入力し、次へをクリックします。



5. この製品に追加するAPIを選択します。ここでは、oauthapiを選択して、次へをクリックします。

API

この製品に追加する API を選択します

<input type="checkbox"/>	タイトル	バージョン	説明
<input type="checkbox"/>	branch	1.0.0	
<input type="checkbox"/>	branch-key	1.0.0	
<input type="checkbox"/>	FindBranch	2.0.0	
<input checked="" type="checkbox"/>	oauthapi	2.0.0	

項目/ページ 10 ▼ | < >

戻る

キャンセル

次へ

新規製品の作成

レート制限とプランを追加して新規製品を作成します

[詳細情報](#)

6. **プラン** はデフォルトのまま **次へ** をクリックします。



新規製品の作成

プラン

レート制限とプランを追加して新規製品を作成します

追加

デフォルトのプラン

タイトル

デフォルトのプラン

説明 (オプション)

デフォルトのプラン

レート制限

100 / 1 時間

戻る

キャンセル

次へ

新規製品の作成

レート制限とプランを追加して新規製品を作成します

[詳細情報](#)

7. 次の画面もデフォルトのまま **次へ** をクリックします。

公開

この製品の公開を有効にします

☐ 製品の公開

可視性

この製品を表示可能にする組織またはグループを選択します

☒ 公開

☐ 認証済み

☐ カスタム

サブスクリプション可能性

この製品をサブスクリプションする組織またはグループを選択します

☒ 認証済み

☐ カスタム

戻る

キャンセル

次へ

新規製品の作成

レート制限とプランを追加して新規作成

[詳細情報](#)

8. 製品の枠が作成されます。 **製品の編集** をクリックして製品の詳細画面を表示

し、 **右上の保存** ボタン **保存** をクリックして保存します。

←

新規製品の作成

要約

☒ 新規製品が作成されました

☒ API が追加されました

☒ レート制限が追加されました

[製品の編集](#)

新規製品の作成

レート制限とプランを追加して新規作成

[詳細情報](#)

9. 製品を公開します。左のメニューから **開発** を選択し、 **oauth-test-product** 製品の右のメニューから **公開** を選択します。



API および製品

追加

タイトル	タイプ	最終変更
 branch-1.0.0	API (REST)	昨日 23:47
 branch-key-1.0.0	API (REST)	2020/01/21
 FindBranch-2.0.0	API (REST)	昨日 12:32
 oauthapi-2.0.0	API (REST)	今日 00:15
 branch auto product-1.0.0	製品	昨日 23:47
 branch-key auto product-1.0.0	製品	2020/01/21
 FindBranch-1.0.0	製品	月曜日 16:15
 oauth-test-product-1.0.0	製品	今日 00:17
 oauthapi auto product-2.0.0	製品	今日 00:15

公開
ステージ

10. 公開先に **Sandbox** を選択して、**公開** をクリックします。

←

製品の公開

公開先

カタログ

Sandbox

☐ 特定のゲートウェイ・サービスに公開する

デフォルトでは、この製品はすべてのゲートウェイ・サービスに公開されます。このオプションを有効にすることで、特定のゲートウェイサービスにも公開できます。

キャンセル 公開

6.4 - 開発者ポータルからのアプリケーション作成と利用登録

1. 開発者ポータルにログインして、上部のメニューから **アプリケーション** をクリックします。



2. **新規アプリケーションの作成** をクリックします。



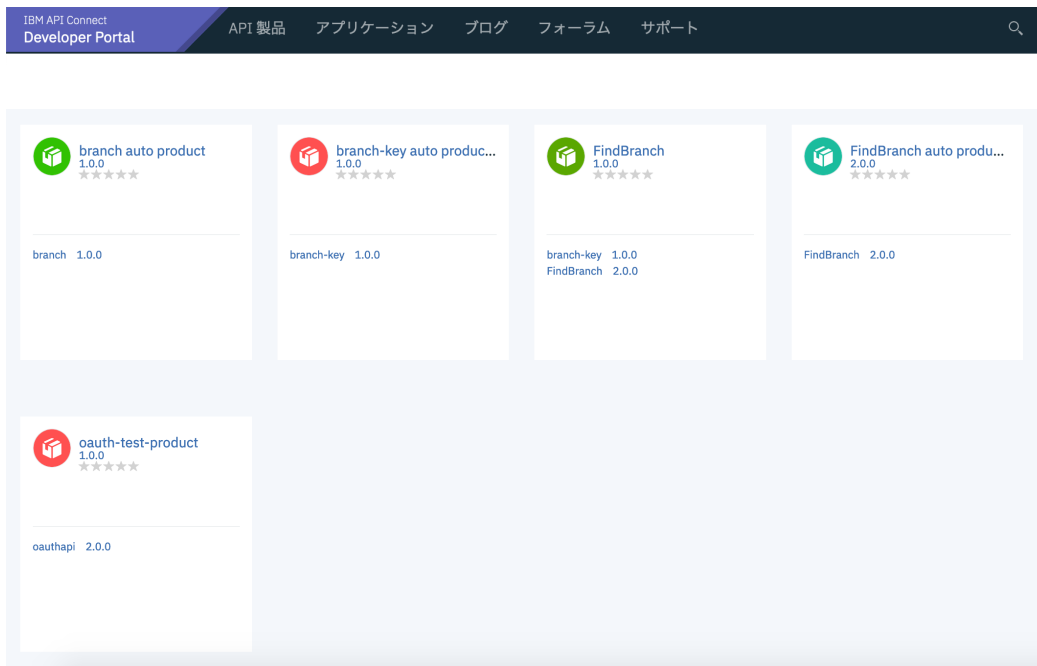
3. タイトルに **oauth-test-app** と入力し、**送信** をクリックします。



4. アプリケーションが登録されると、**APIキー** と **秘密鍵(シークレット)** が表示されます。シークレットはここで一度しか表示されないため、今後のためにコピーして保存しておいてください。後続の演習で利用します。**継続** をクリックします。



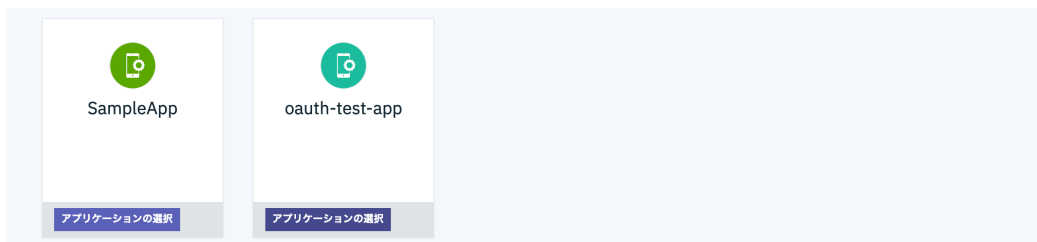
5. プランへのサブスクライブを行います。API製品 タブをクリックし、oauth-test-product 製品を選択します。



6. デフォルトのプラン プランにの サブスクライブ をクリックします。



7. 作成した oauth-test-app が表示されるので、アプリケーションの選択 をクリックします。



8. 内容を確認して 次へ をクリックします。

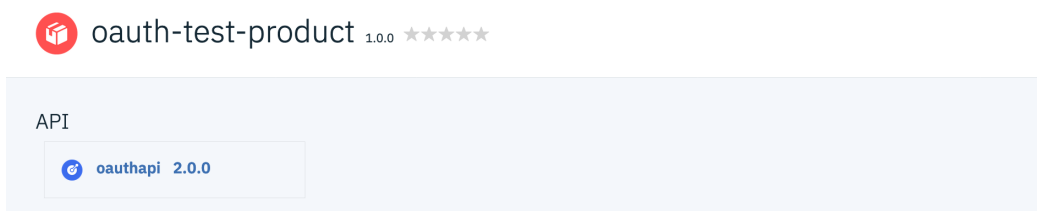


9. **完了** をクリックします。

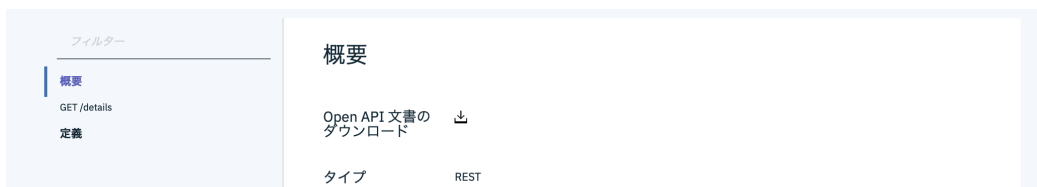


6.5 - 開発者ポータルからのAPIのテスト

1. APIをテスト実行してみましょう。 **oauthapi** APIをクリックします。



2. APIの詳細が表示されます。パスの詳細を表示するために、左のメニューから **GET /details** を選択します。



3. **試してみる** をクリックしします。

The screenshot shows the OAuth API interface. On the left, there is a sidebar with a 'フィルター' (Filter) section and a '概要' (Overview) section. The '概要' section has a sub-section 'GET /details'. On the right, the main area displays 'GET : /details' with two tabs: '詳細' (Details) and '試してみる' (Try it out). The '試してみる' tab is currently selected.

- クライアントID に `oauth-test-app` を選択し、クライアント秘密鍵 には、コピーしておいた、シークレットを入力します。

The screenshot shows the OAuth API interface. On the left, there is a sidebar with a 'セキュリティ' (Security) section. The main area displays the '識別' (Identification) section. It has two fields: 'クライアント ID' (Client ID) with a dropdown menu showing 'oauth-test-app' and 'クライアント秘密鍵' (Client Secret Key) with a text input field containing a long string of asterisks.

- ユーザー名に `user` 、パスワードに `pass` を入力し、スコープ `details` にチェックを入れて、トークンの取得 ボタンをクリックします。

The screenshot shows the OAuth API interface. On the left, there is a sidebar with a 'セキュリティ' (Security) section. The main area displays the '許可' (Authorization) section. It has four fields: 'ユーザー名' (Username) with a text input field containing 'user', 'パスワード' (Password) with a text input field containing '****', 'スコープ' (Scope) with a checkbox labeled 'details' that is checked, and 'トークン URL' (Token URL) with a text input field containing a long string of asterisks. Below these fields is a 'トークンの取' (Get Token) button.

- そのまま下にスクロールして 送信 をクリックします。応答が返ることを確認します。

以上で、演習6は終了です。