

演習 2 - セキュリティー定義の付加されたAPIの作成

この演習では、セキュリティー定義が付加されたAPIを作成します。

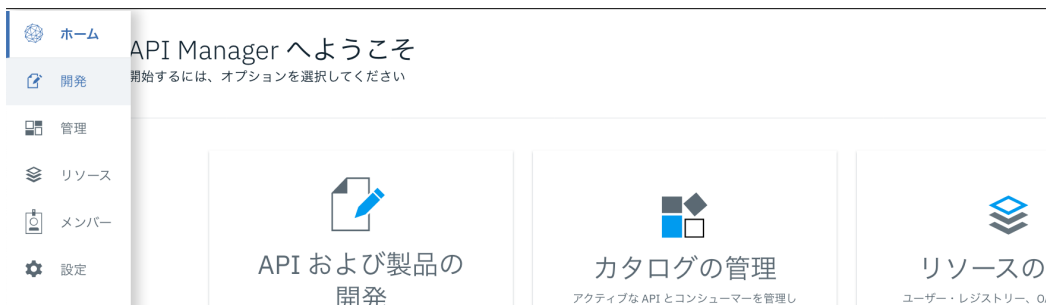
演習 2 - 目的

この演習では、以下の内容を理解できます。

- APIへのセキュリティー定義の付加方法
- セキュリティー定義の付加されたAPIのテスト方法
- 公開されたAPIのURLの確認方法

2.1 - API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. 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。



3. 演習1と同じ手順で、異なる名前のAPIを作成します。 **開発** 画面で、 **追加** メニューから **API** を選択します。



4. ターゲット・サービスから を選択し、 次へ をクリックします。

←

API の追加

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

```
graph LR; App[App] --> APIProxy[API Proxy]; APIProxy --> TargetEndpoint[Target Endpoint]; OpenAPI[OpenAPI] -.-> APIProxy;
```

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

5. 作成するAPIの情報を以下のように入力し、 次へ をクリックします。指定するターゲット・サービスURLは、 branch APIと同じです。

項目	入力値	備考
タイトル	branch-key	APIの名前
バージョン	1.0.0	デフォルトで 1.0.0 が

		入力されます
基本パス	/branch-key	デフォルトでAPIのタイトルと同じ名前が入力されます
ターゲット・サービス URL	https://apictutorials.mybluemix.net/branches	このAPIが呼び出すターゲットサービスを入力します

情報

この API の詳細を入力します

タイトル

branch-key

名前

branch-key

バージョン

1.0.0

基本パス (オプション)

/branch-key

説明 (オプション)

ターゲット・サービス URL

プロキシ処理するターゲット・サービスの URL を入力します

ターゲット・サービス URL

キャンセル

次へ



ターゲット・サービスURL に入力するURLは、店舗情報をGET要求で取得するAPIです。ブラウザの別のタブを開き、ターゲット・サービスURL に入力したURLをコピー&ペーストして、APIのレスポンスが返ることを確認してください。

6. キー単位でAPI呼び出しを制限する にチェックを入れ、 5 / 1 分 の設定に変更して 次へ をクリックします。

セキュア

この API のセキュリティを構成します

☒ クライアント ID を使用した保護

☒ キー単位で API 呼び出しを制限する

5 / 1 分

☒ CORS

[戻る](#)[キャンセル](#)[次へ](#)

7. APIが作成されるので、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

✓

セキュリティが適用されました

✓

レート制限が追加されました

API の編集

8. 左のメニューから セキュリティ定義 を選択します。デフォルトで clientID が定義されています。 clientID をクリックして詳細を表示します。

API のセットアップ

セキュリティ定義

セキュリティ

バス

定義

プロパティ

ターゲット・サービス

カテゴリ

アクティビティ・ログ

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](#)

追加

名前	タイプ	場所	
clientID	apiKey	ヘッダー	⋮

9. `clientID` セキュリティ定義の詳細が確認できます。この定義では、クライアント・アプリケーションごとに発行する `APIキー` を、HTTPヘッダーに、`X-IBM-Client-Id` という名前で受け付けることが定義されています。ヘッダ一名は変更することが可能です。ここでは、このセキュリティ定義をそのまま利用します。

API セキュリティ定義

名前

clientID

説明 (オプション)

タイプ

☒ API キー ☐ 基本 ☐ OAuth2

場所

ヘッダー

鍵タイプ (オプション)

1 つ選択してください...

パラメーター名 (オプション)

X-IBM-Client-Id

左上のをクリックして戻ります。

10. 左のメニューから `セキュリティ` を選択します。`clientID` にチェックが入っているため、このセキュリティ定義が有効化されています。

API のセットアップ

セキュリティ定義

セキュリティ

バス

定義

プロパティ

ターゲット・サービス

カテゴリ

アクティビティ・ログ

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。[詳細情報](#)

追加

セキュリティ定義

☒ clientID
apiKey



セキュリティ定義に定義しただけでは、その定義が有効にならないことに注意してください。必ず **セキュリティ** メニューで、有効にしたい定義にチェックが入っていることを確認してください。

保存


11. 右上の **保存** ボタンをクリックして、API定義を保存します。

以上でセキュリティ定義が追加できました。

2.2 - APIのテスト

1. 演習1で行ったのと同じ手順でAPIのテストを行います。API開発画面上部から **アセンブル** をクリックして、アセンブル画面に移動します。



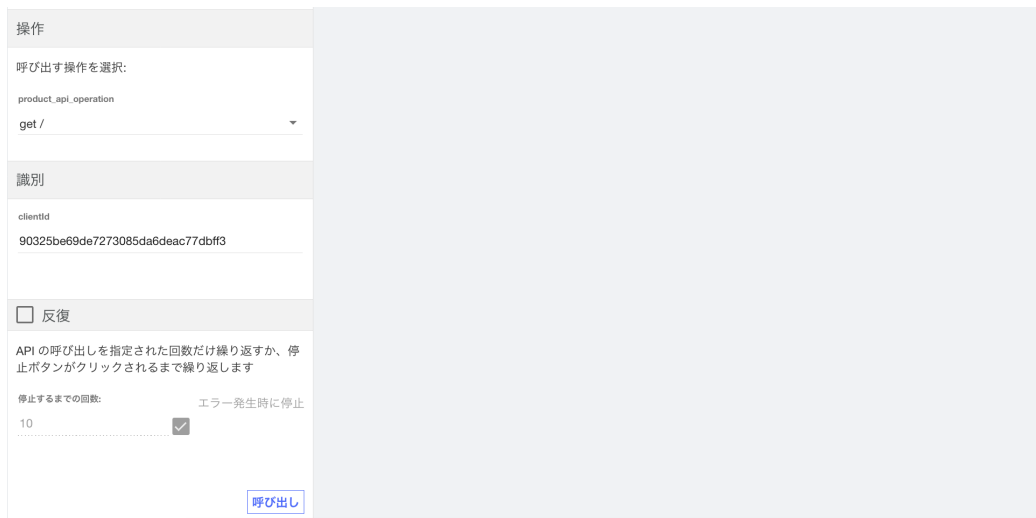
2. アセンブル画面でAPIのテストツールを利用します。画面上のボタン  をクリックしてテストツールを表示します。



3. **APIのアクティブ化** をクリックし、APIをカタログ上にデプロイします。



4. 操作・メニューで呼び出す操作として `get /` を選択し、`呼び出し` をクリックしてAPIを呼び出します。

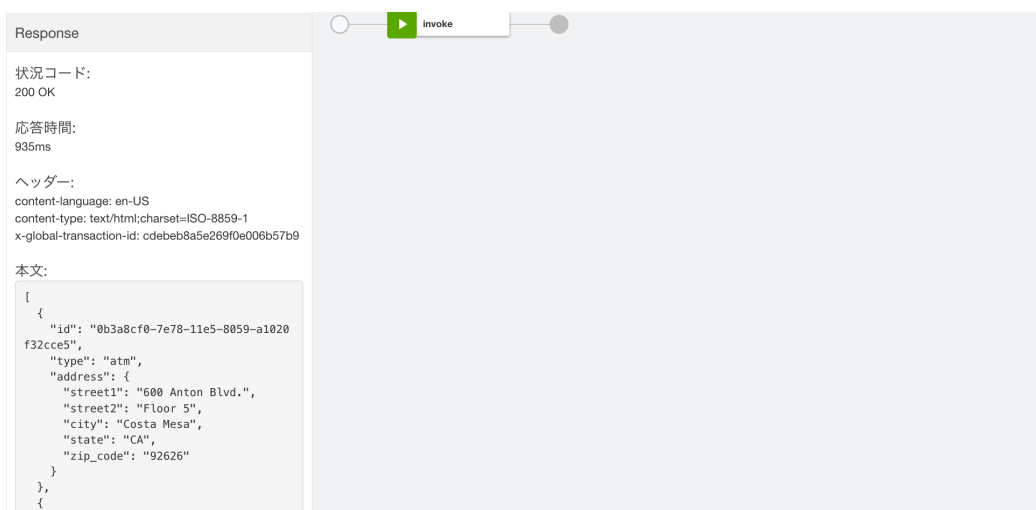


The screenshot shows a configuration panel for an API client. It has three main sections: '操作' (Operation), '識別' (Identification), and a checkbox for '反復' (Repeat). In the '操作' section, '呼び出す操作を選択:' (Select operation to call) is set to 'product_api_operation' and 'get /' is selected in the dropdown. The '識別' section shows 'clientId' as '90325be69de7273085da6deac77dbff3'. The '反復' section has a checkbox labeled 'API の呼び出しを指定された回数だけ繰り返すか、停止ボタンがクリックされるまで繰り返します' (Repeat the API call the specified number of times or until the stop button is clicked). The '停止するまでの回数:' (Number of times to repeat) is set to '10', and the 'エラー発生時に停止' (Stop on error) checkbox is checked. A blue '呼び出し' (Call) button is at the bottom right.



テストツールに、`clientID` が表示されていることに注目してください。このAPIの呼び出しにはAPIキーが必要となったため、テストツールが自動的にテスト用のAPIキーを発行し、このAPIの呼び出しができるように利用登録までが完了しています。利用登録については、後続の演習で説明があります。


5. API応答が表示されます。状況コードが `200 OK` と表示されており、応答本文にJSON形式の応答が表示されていることを確認します。



The screenshot shows the 'Response' tab of the API client. It displays the status code '200 OK', the response time '935ms', and the headers: 'content-language: en-US', 'content-type: text/html; charset=ISO-8859-1', and 'x-global-transaction-id: cdebe8a5e269f0e006b57b9'. The body is a JSON object:

```
{
  "id": "0b3a8cf0-7e78-11e5-8059-a1020f32cce5",
  "type": "atm",
  "address": {
    "street1": "600 Anton Blvd.",
    "street2": "Floor 5",
    "city": "Costa Mesa",
    "state": "CA",
    "zip_code": "92626"
  }
}
```

6. ここまでで、セキュリティー定義を付加したAPIの定義とテストが完了しまし

た。右上の 保存 ボタン  をクリックして、API定義を保存し、メニューから ホーム を選択してホームに戻ります。

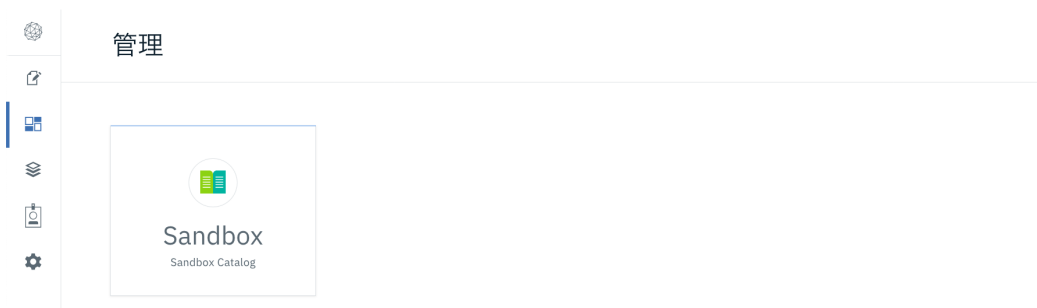


2.3 - APIの外部ツールからのテスト

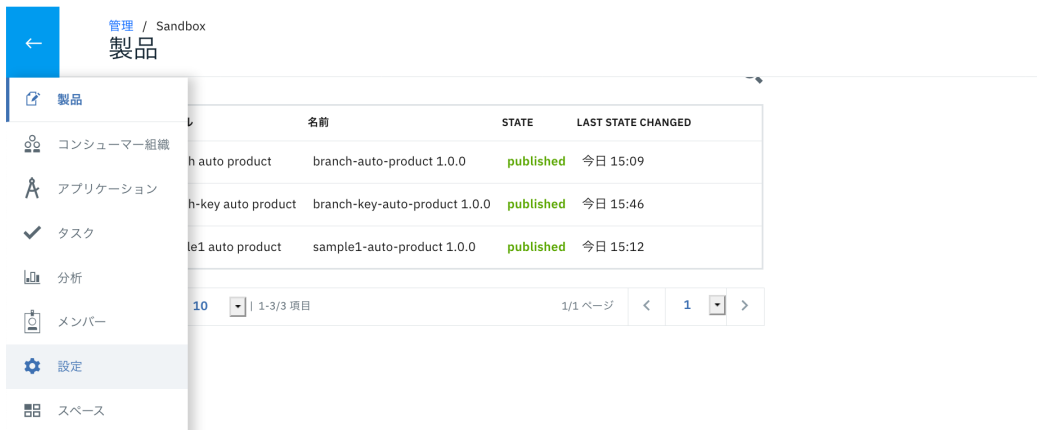
1. 演習1、演習2で作成したAPIが、どのようなURLで公開されるのかを確認します。公開されるAPIのエンドポイントは、ゲートウェイ、プロバイダー組織、カタログによって決まります。エンドポイントのURLはカタログの設定で確認できます。API Managerの左のメニューから 管理 を選択します。



2. Sandbox を選択します。



3. 左側のカタログの管理メニューから 設定 を選択します。



4. APIエンドポイントを選択すると、このカタログのエンドポイントが確認できます。外部から呼び出しをテストするために、このURLをコピーしておきます。



カタログごとの API エンドポイント のルールは、<ゲートウェイのエンドポイント>/<プロバイダー組織の名前>/<カタログの名前> となります。

5. まず、演習1で作成した、branch APIをテストしてみましょう。APIのURLは以下の形式となります。

APIのURL

<カタログごとのAPIエンドポイント>/<APIの基本パス>/<呼び出すパス>

演習1で作成した branch APIは、基本パスが /branch、パスは / となっています。したがって、前の手順でコピーしたAPIエンドポイントの後ろに、/branch/ を追加したものが、branch APIのURLとなります。

ブラウザなどの外部ツールを起動し、このURLに対してGET要求でAPIを呼び出してみましょう。パラメーター等は不要です。

branch APIのURL

<カタログごとのAPIエンドポイント>/branch/

branch APIはセキュリティー定義を付加していないため、呼び出しが成功するはずです。

6. 同様に、セキュリティー定義を付加した branch-key APIを同様の手順で呼び出してみましょう。

branch-key APIのURL

<カタログごとのAPIエンドポイント>/branch-key/

今度はエラーとなるはずです。branch-key APIには、セキュリティー定義でAPIキーが必要なため、401 Unauthorized エラーが応答として戻ります。以下の画面は、Chromeブラウザで実行した応答の例です。

以上で、演習2は終了です。

続いて、[演習 3 - インポートによるAPIの作成](#)に進んでください。