Note: A more recent version of IBM API Connect is available.
For details, see the IBM API Connect 10.0.2 and later product documentation.

Tutorial: Validate a JSON Web Token (JWT)

Last Updated: 2021-07-14

This tutorial shows you how to define and implement a REST API definition that validates a JSON Web Token (JWT).

About this tutorial

In this tutorial, you complete the following lessons:

1. Validate a JWT
2. Testing the REST API

Note: The Sandbox catalog must be configured to use either a DataPower® Gateway (v5 compatible) or a DataPower API Gateway or both. See Creating and configuring Catalogs.

Before You Begin

You must also do the following steps.

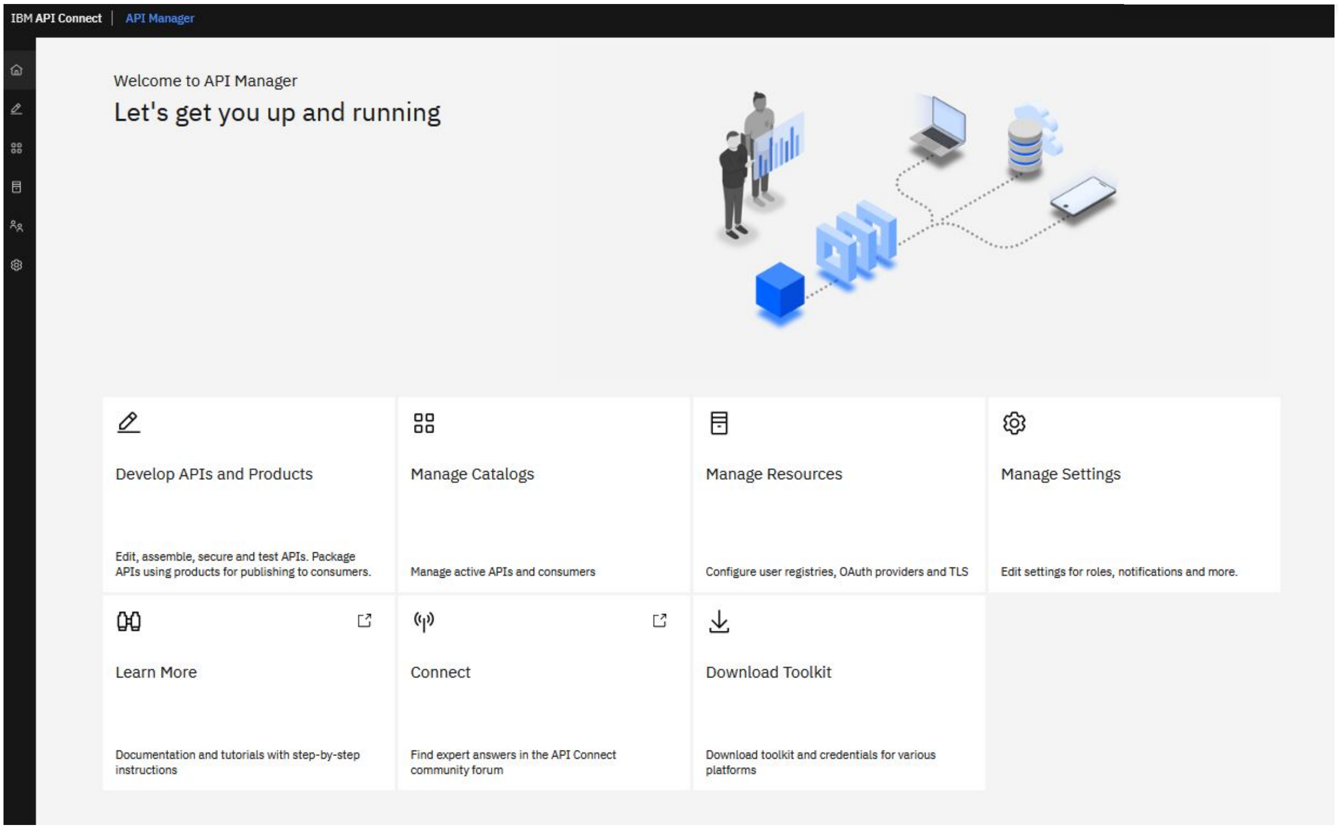
- Complete the Tutorial: Generate a JSON Web Token (JWT) tutorial. This tutorial generates a JSON Web Token that can be validated by this tutorial. You need this JWT to test this validation API.

Validate a JWT

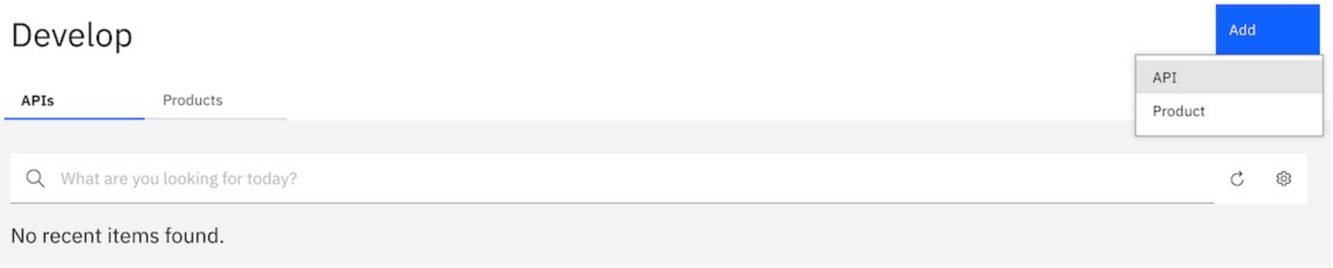
Create a REST API to validate a JSON Web Token (JWT).

To add and define this REST API, complete the following steps:

1. Log in to API Manager.
2. In the Welcome page, click the **Develop APIs and Products** tile.

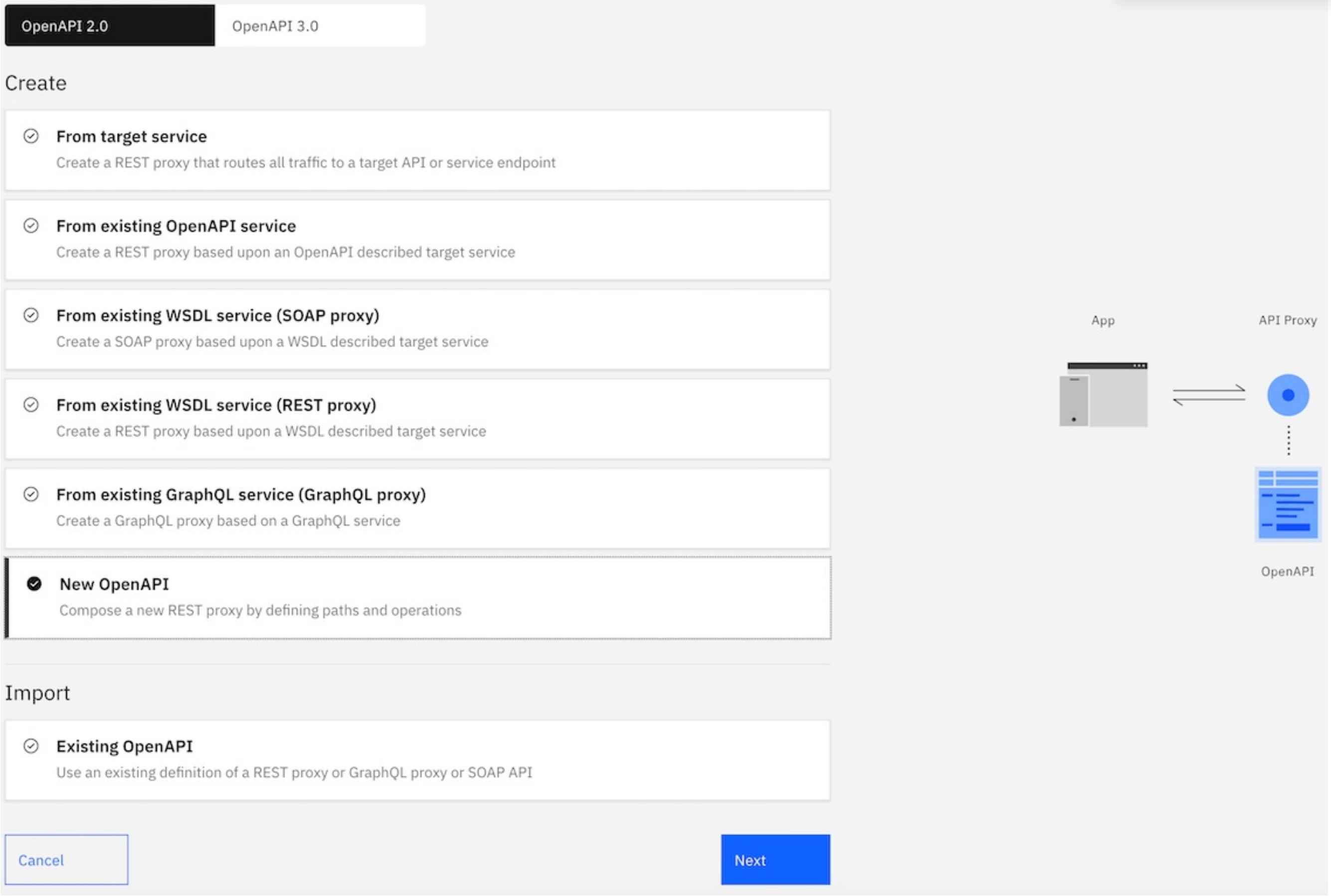

IBM API Connect | API Manager
Welcome to API Manager
Let's get you up and running
Develop APIs and Products
Edit, assemble, secure and test APIs. Package APIs using products for publishing to consumers.
Manage Catalogs
Manage active APIs and consumers
Manage Resources
Configure user registries, OAuth providers and TLS
Manage Settings
Edit settings for roles, notifications and more.
Learn More
Documentation and tutorials with step-by-step instructions
Connect
Find expert answers in the API Connect community forum
Download Toolkit
Download toolkit and credentials for various platforms

3. Click **Add > API**.



4. Ensure that **OpenAPI 2.0** is selected.
5. Select **New OpenAPI**. Click **Next**.

Cookie Preferences



Create

○ **From target service**
Create a REST proxy that routes all traffic to a target API or service endpoint

○ **From existing OpenAPI service**
Create a REST proxy based upon an OpenAPI described target service

○ **From existing WSDL service (SOAP proxy)**
Create a SOAP proxy based upon a WSDL described target service

○ **From existing WSDL service (REST proxy)**
Create a REST proxy based upon a WSDL described target service

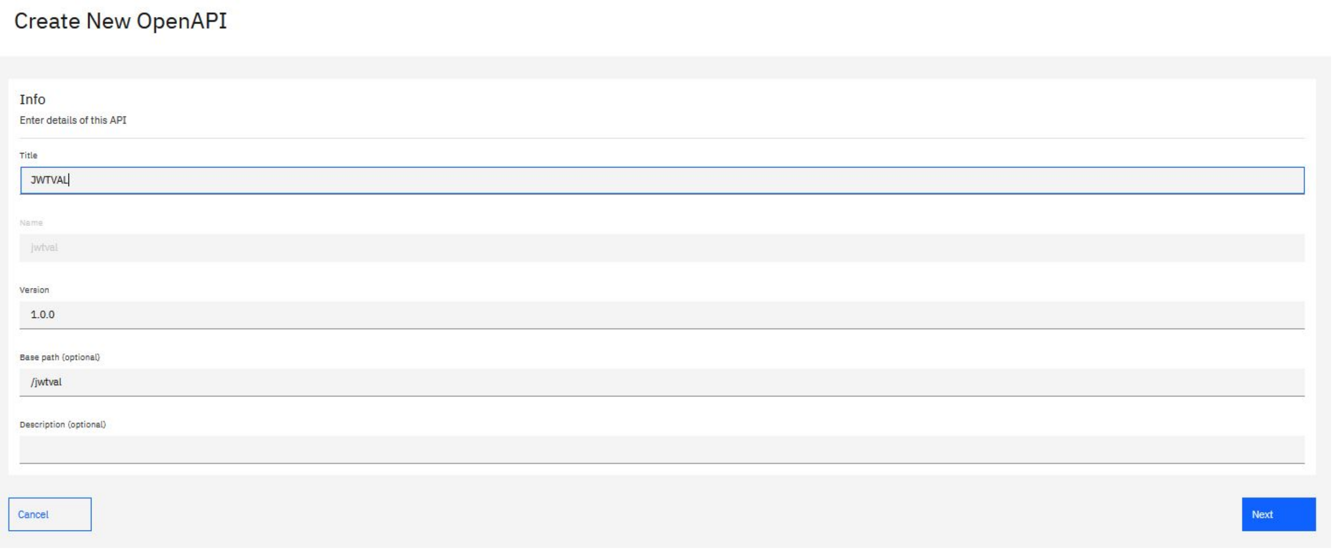
○ **From existing GraphQL service (GraphQL proxy)**
Create a GraphQL proxy based on a GraphQL service

● **New OpenAPI**
Compose a new REST proxy by defining paths and operations

Import

○ **Existing OpenAPI**
Use an existing definition of a REST proxy or GraphQL proxy or SOAP API

6. Enter the appropriate information to create a REST API definition.
   a. In the **Title** field, enter `JWTVAL`.
   b. The **Name** and **Base Path** fields auto-populate with the terms `jwtval` and `/jwtval` respectively.
   c. The **Version** field auto-populates with `1.0.0`.



Create New OpenAPI

Info
Enter details of this API

Title: JWTVAL

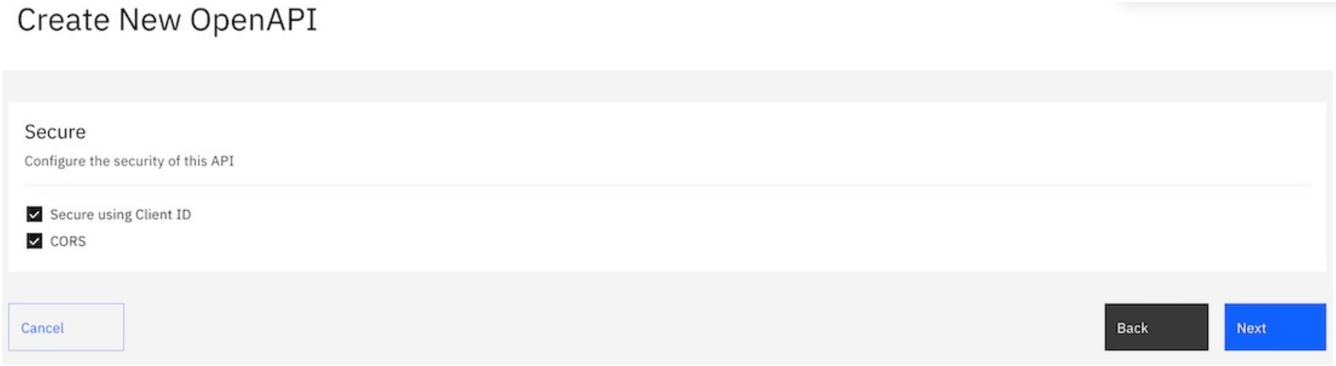
Name: jwtval

Version: 1.0.0

Base path (optional): /jwtval

Description (optional):

7. Click **Next**.
8. Make no changes on the **Secure** screen. Click **Next**.



Create New OpenAPI

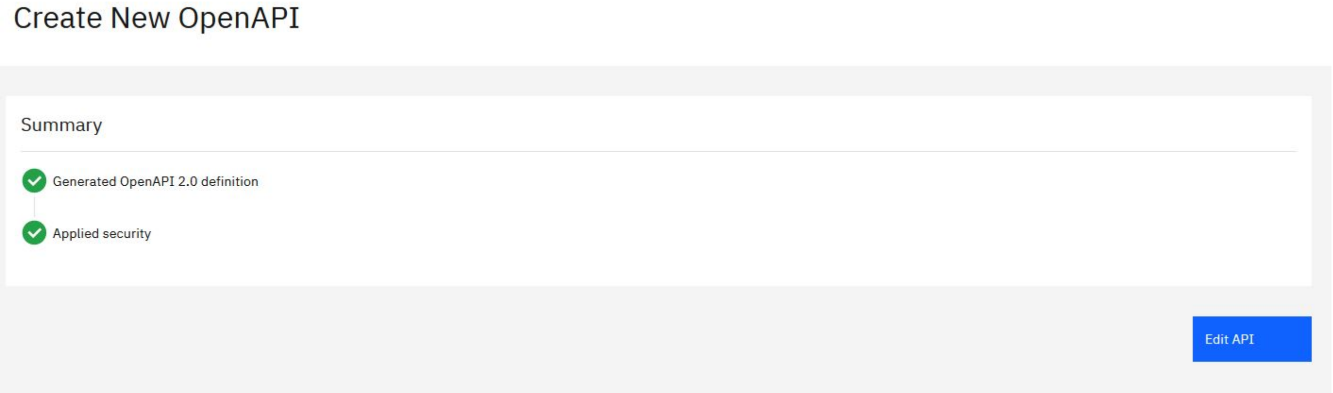
Secure

Configure the security of this API

☑ Secure using Client ID

☑ CORS

9. You see the progress as the new API gets created. When it is done, you see a Summary. Click **Edit API**.



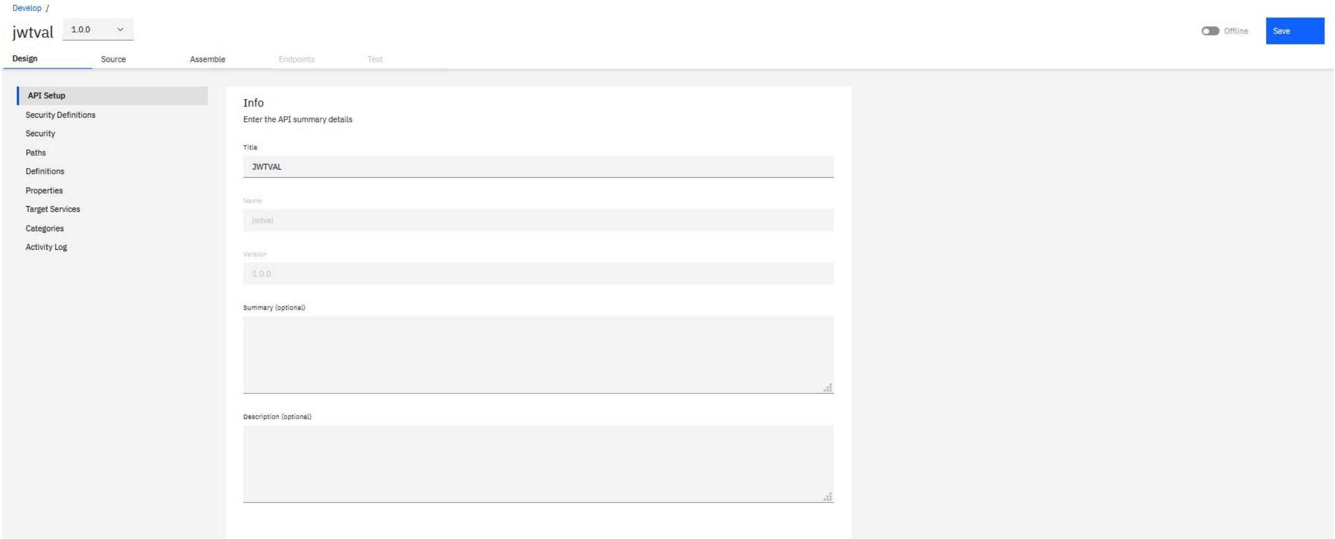
Create New OpenAPI

Summary

✓ Generated OpenAPI 2.0 definition

✓ Applied security

10. In the side bar of the Design page, select **Paths** to display the **Paths** panel.



Info

Enter the API summary details

Title: JWTVAL

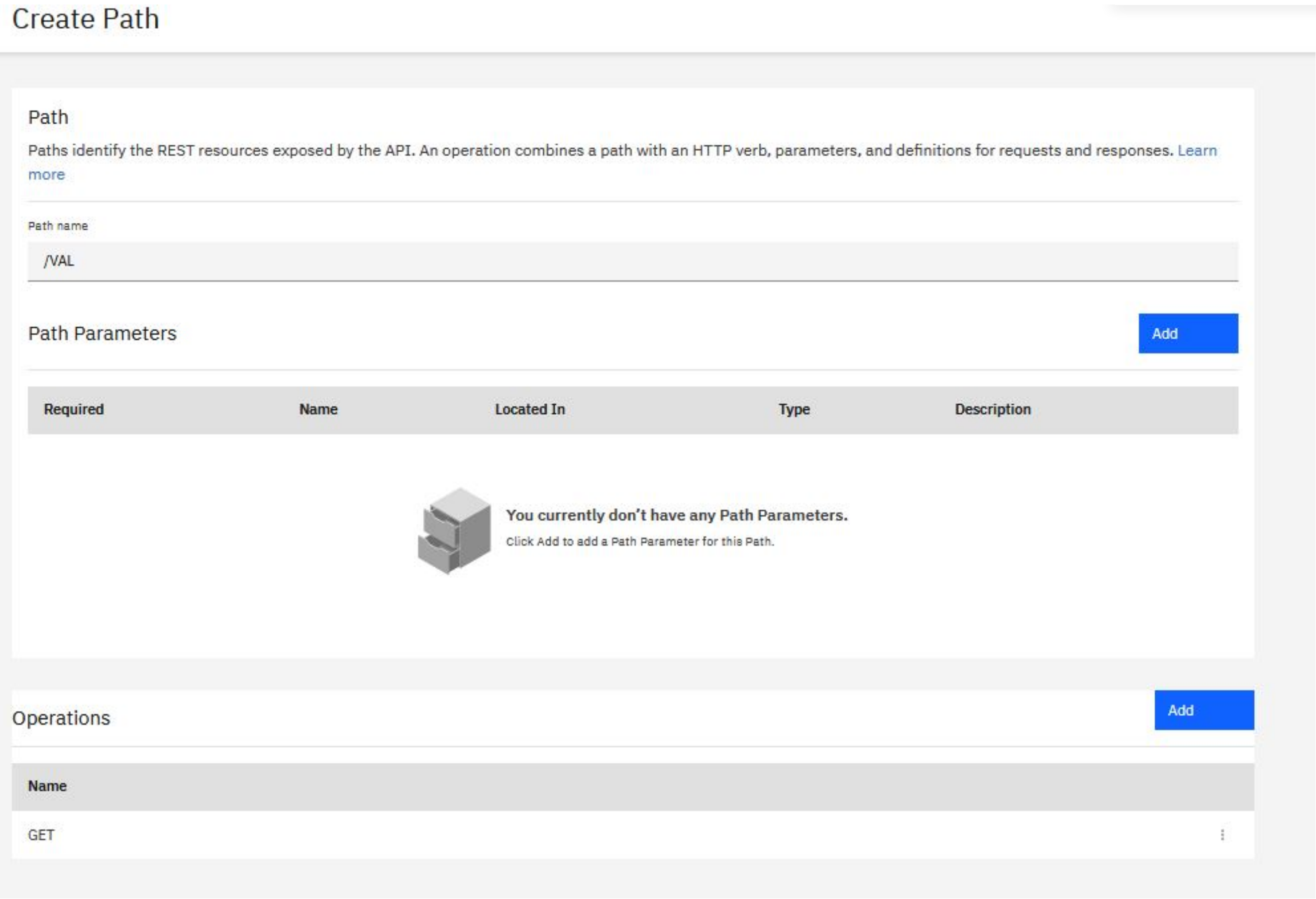
Name: jwtval

Version: 1.0.0

Summary (optional)

Description (optional)

11. Click **Add**.
12. In the **Path name** field, enter **/val**.
13. In the **Operations** section, click **Add**.
14. Select **GET** and click **Add**.



Create Path

Path

Paths identify the REST resources exposed by the API. An operation combines a path with an HTTP verb, parameters, and definitions for requests and responses. Learn more

Path name

/VAL

Path Parameters

Required	Name	Located In	Type	Description

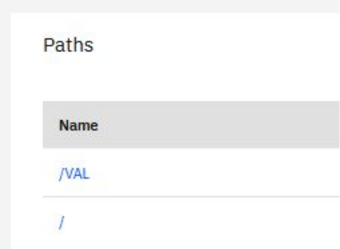
You currently don't have any Path Parameters.

Click Add to add a Path Parameter for this Path.

Operations

Name
GET

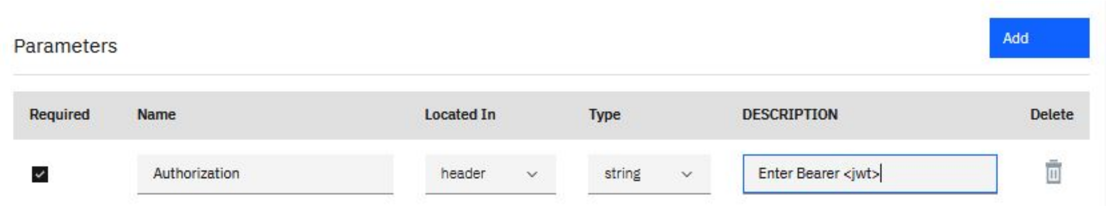
15. Click **Save**.
16. Click **/val** in the list of available paths.



Paths

Name
/VAL
/

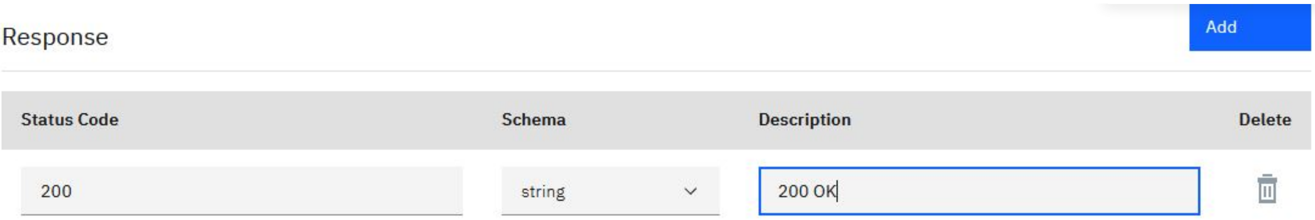
17. Click **GET** in the list of **Operations**.
18. Scroll down. In the **Parameters** section, click **Add**.
    a. Select **REQUIRED**.
    b. Enter `Authorization` in the **NAME** field.
    c. Select `header` in the **LOCATED IN** field.
    d. Select `string` in the **TYPE** field.
    e. Enter `Enter Bearer <jwt>` in the **DESCRIPTION** field.



Parameters

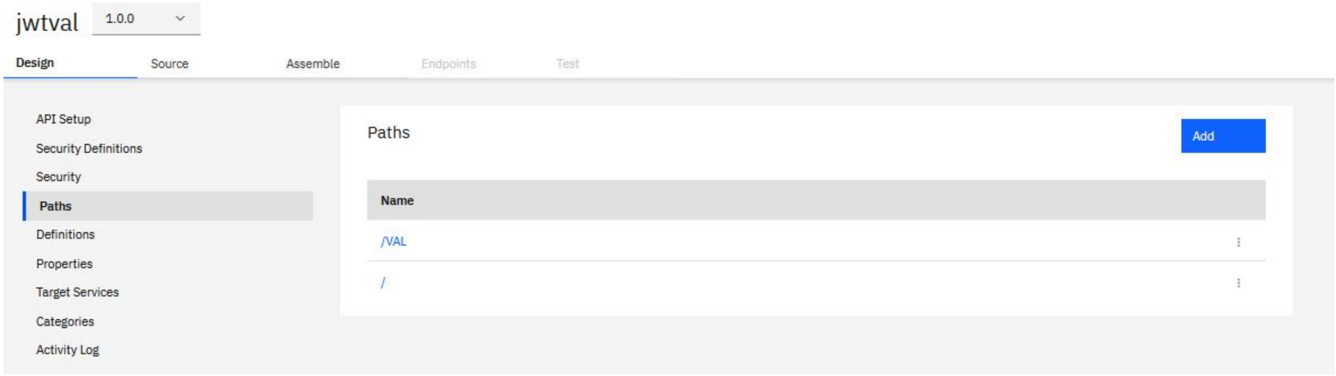
Required	Name	Located In	Type	DESCRIPTION	Delete
☑	Authorization	header	string	Enter Bearer <jwt>	🗑

19. In the **Response** section, change the description of the pre-supplied `200` status code to `200 OK`.



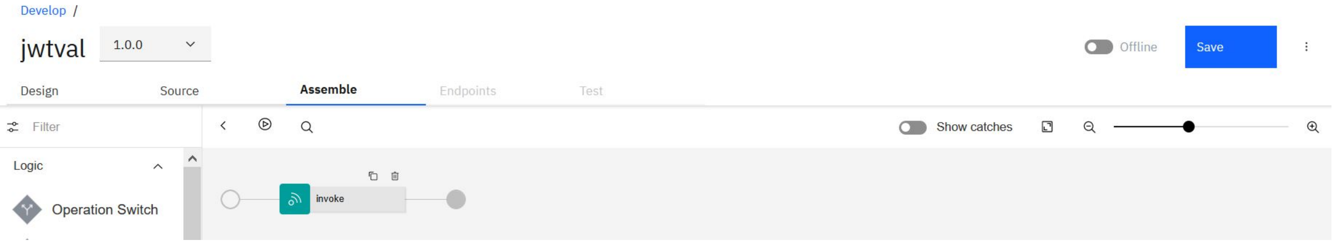
Status Code	Schema	Description	Delete
200	string	200 OK	

20. Click **Save**.
21. Click **Assemble**.

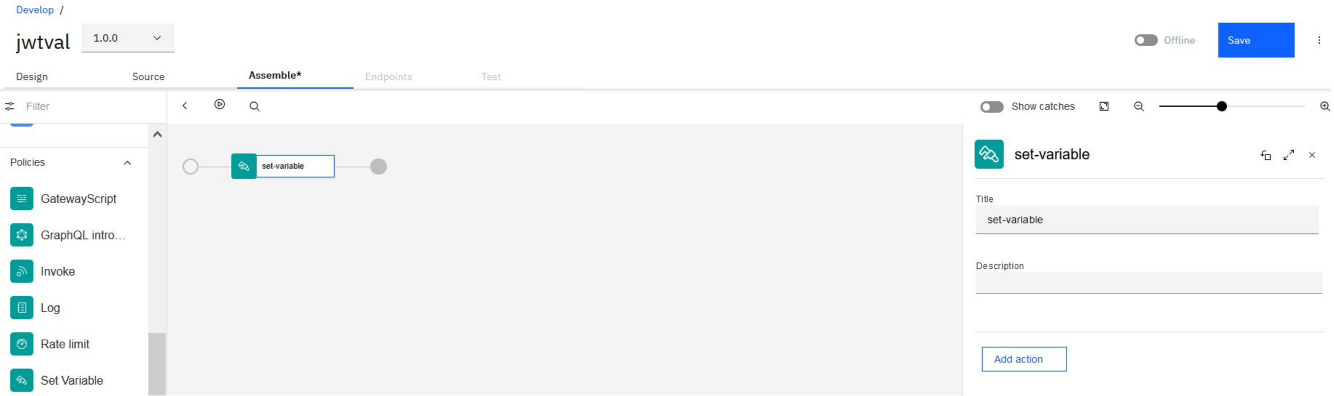


Name
/VAL
/

22. Hover the mouse over the existing **Proxy** or **Invoke** action and click the trash can icon to delete it.



23. Drag the **Set Variable** action onto the processing flow line. A configuration panel automatically opens.

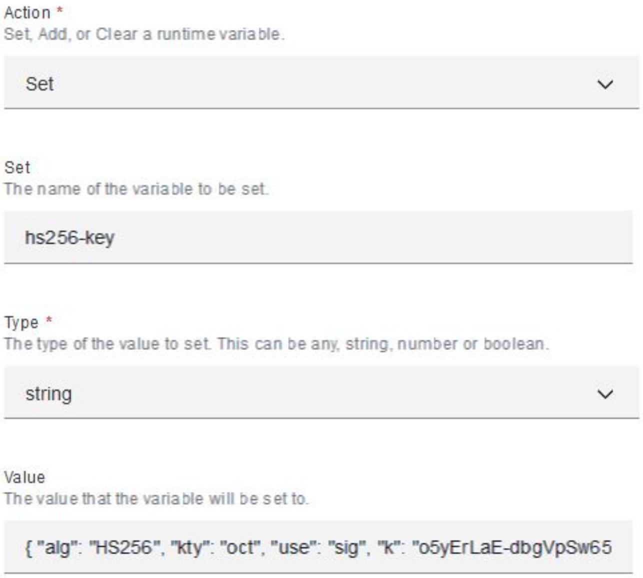


Title: set-variable

Description:

24. Click **+ Action** field.
25. Enter `hs256-key` in the **Set** field.
26. Select `string` in the **Type** field.
27. Enter a JWK in the **Value** field. Here is an example. `{ "alg": "HS256", "kty": "oct", "use": "sig", "k": "o5yErLaE-dbgVpSw65Rq57OA9dHyaF66Q_Et5azPa-XUjbyP0w9iRWhR4kru09aFfQLXeIODIN4uhjElYKXt8n76jt0Pjkd2pqk4t9abRF6tnL19GV4pflfL6uvVKkP4weOh39tqHt4TmkBgF2P-gFhgssZpjwq6l82fz3dUhQ2nkzoLA_CnyDGLZLd7SZ1yv73uzfE2Ot813zmig8KTMEMWVcWSDvy61F06vs_6LURcq_IEEevUiubBxG5S2a yHt95Siz0QUb0MNlT_X8F76wH7_A37GpKKJGqeaiNWmHkgWdE8QWDQ", "kid": "hs256-key" }`

Cookie Preferences



Action *
Set, Add, or Clear a runtime variable.

Set

Set
The name of the variable to be set.

hs256-key

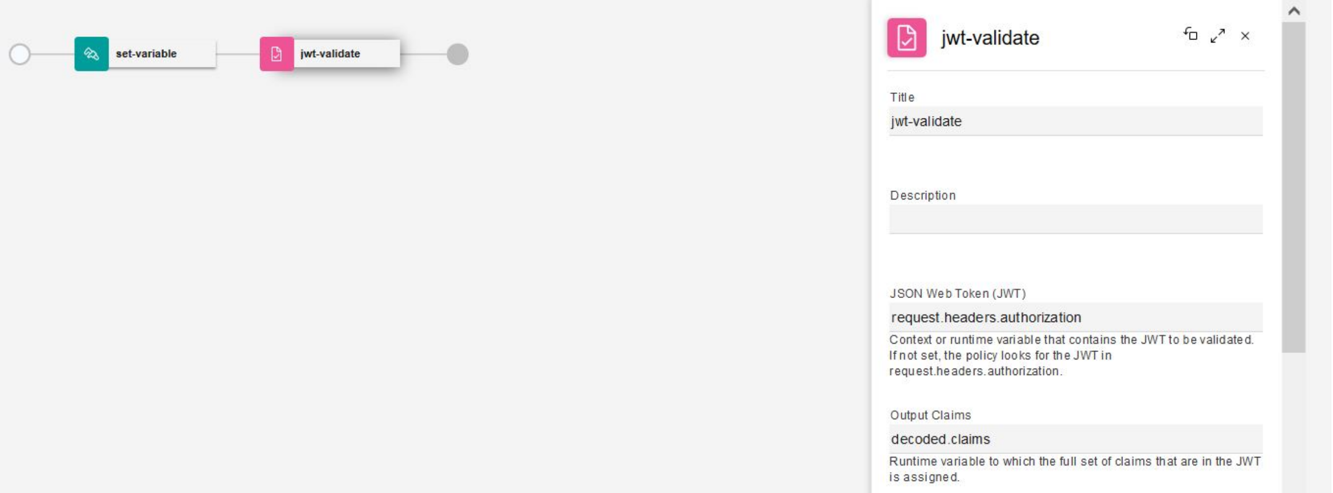
Type *
The type of the value to set. This can be any, string, number or boolean.

string

Value
The value that the variable will be set to.

{ "alg": "HS256", "kty": "oct", "use": "sig", "k": "o5yErLaE-dbgVpSw65

28. Close the property panel. Click **Save**.
29. Drag the **Validate JWT** action onto the processing flow line after the **set-variable** icon. A configuration panel automatically opens.



jwt-validate

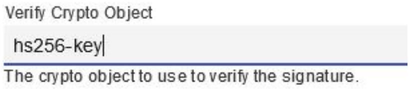
Title
jwt-validate

Description

JSON Web Token (JWT)
request.headers.authorization
Context or runtime variable that contains the JWT to be validated. If not set, the policy looks for the JWT in request.headers.authorization.

Output Claims
decoded.claims
Runtime variable to which the full set of claims that are in the JWT is assigned.

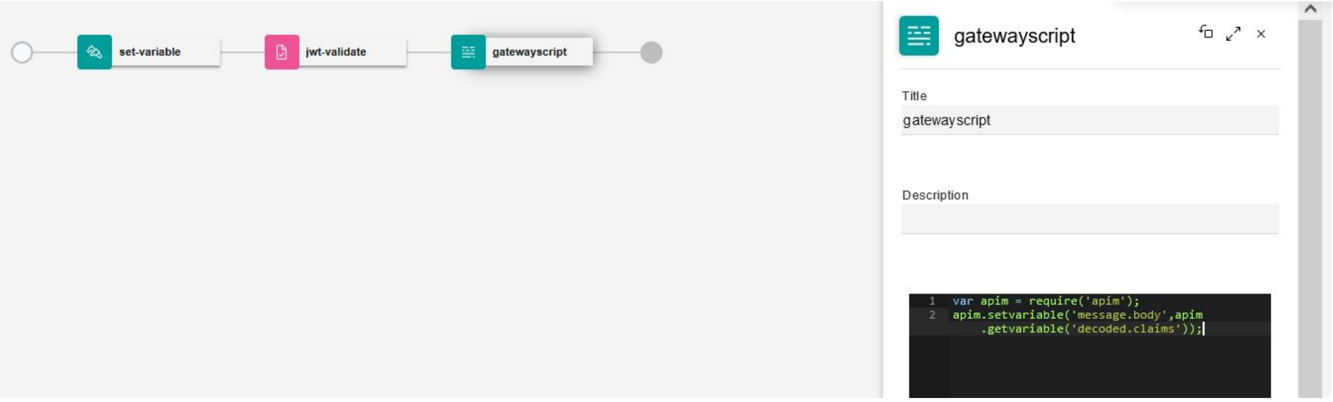
30. Enter hs256-key in the **Verify Crypto JWK variable name** field.



Verify Crypto Object
hs256-key
The crypto object to use to verify the signature.

31. Close the property panel. Click **Save**.
32. Drag the **GatewayScript** action onto the processing flow line after the Validate JWT icon. A configuration panel automatically opens.
33. Enter the following code:

```
var apim = require('apim');
apim.setvariable('message.body',apim.getvariable('decoded.claims'));
```



gatewayscript

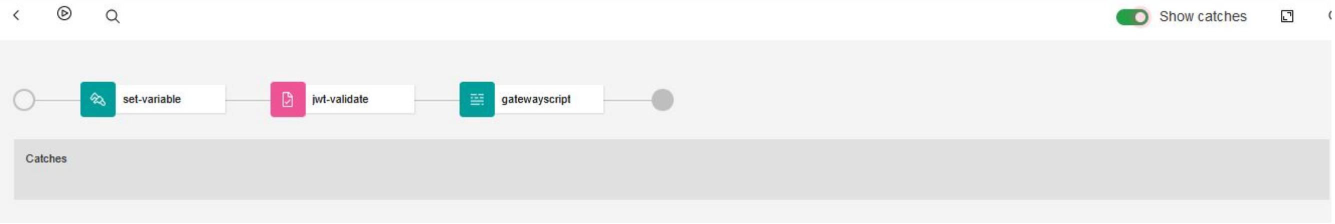
Title

gatewayscript

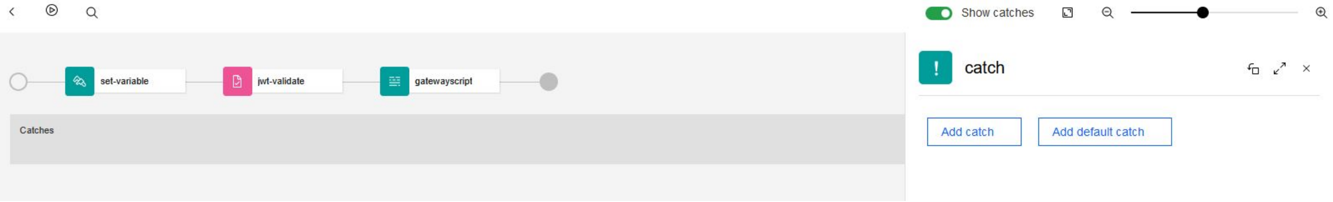
Description

```
var apim = require('apim');
apim.setvariable('message.body',apim
    .getvariable('decoded.claims'));
```

34. Close the property panel. Click **Save**.
35. Ensure that the **Show catches** option is enabled so that the **catch** area is displayed.

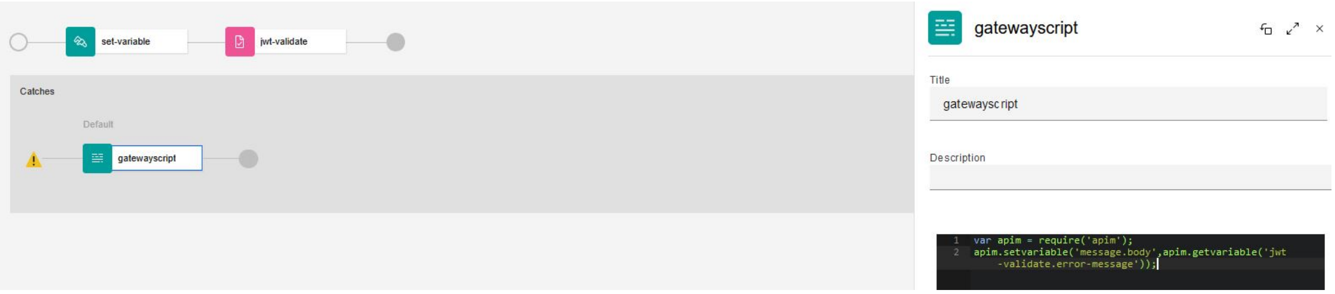


36. Click **Catches**. A property panel opens.



37. Click **+ Default**.
38. Drag the **GatewayScript** policy action onto the catch flow line.
39. Enter the following code:

```
var apim = require('apim');
apim.setvariable('message.body',apim.getvariable('jwt-validate.error-message'));
```



gatewayscript

Title

gatewayscript

Description

```
var apim = require('apim');
apim.setvariable('message.body',apim.getvariable('jwt
    -validate.error-message'));
```

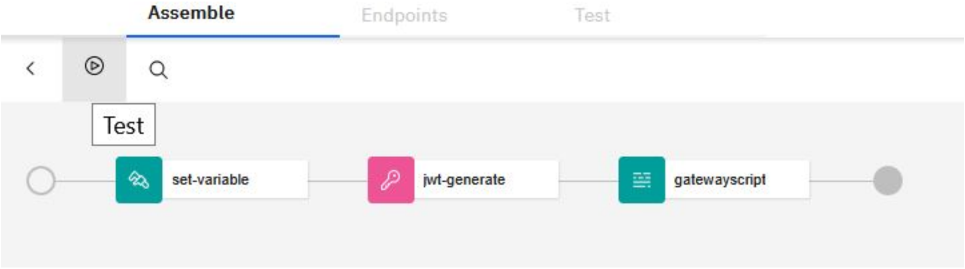
40. Close the property panel. Click **Save**.

Testing the REST API

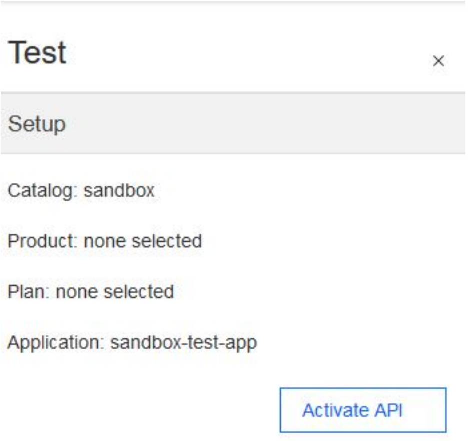
> **Note:** Due to Cross-Origin Resource Sharing (CORS) restrictions, the assembly test tool cannot be used with the Chrome or Safari browsers on the macOS Catalina platform.

To test the REST API, you need a valid JWT. You can obtain such a JWT by invoking the API created in the Tutorial: Generate a JSON Web Token (JWT). To complete testing, take the following steps:

1. Click the **Test** icon ⓟ.



2. Click **Activate API**.



Test

Setup

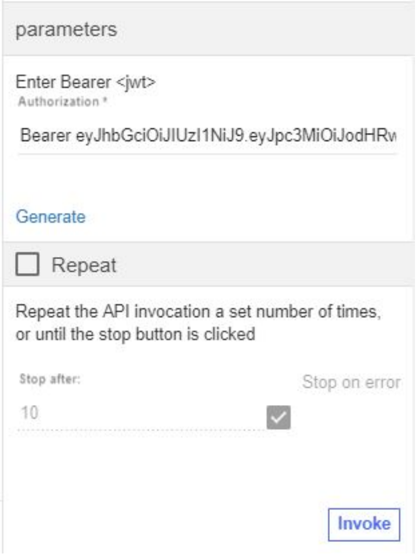
Catalog: sandbox

Product: none selected

Plan: none selected

Application: sandbox-test-app

3. Select the **get /val Operation**.
4. Enter `Bearer` followed by a space followed by a valid JWT generated with the same sign key in the **Authorization** field. Invoking the API created by the Generate JWT tutorial produces such a key.



parameters

Enter Bearer <jwt>

Authorization *

Bearer eyJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJodHRw

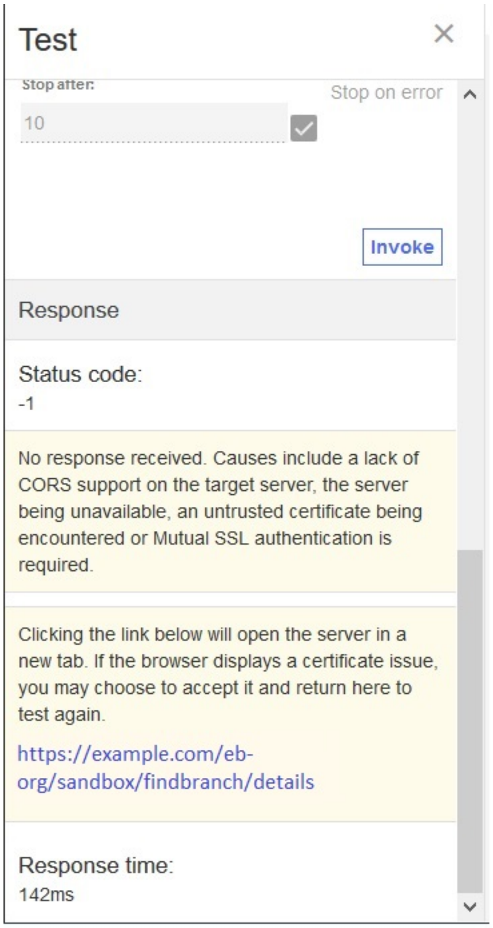
□ Repeat

Repeat the API invocation a set number of times, or until the stop button is clicked

Stop after: 10

Stop on error ☑

5. Click **Invoke**. You might encounter a yellow error box with a URL embedded in it. Click this URL to override a browser certificate error.



Status code:
-1

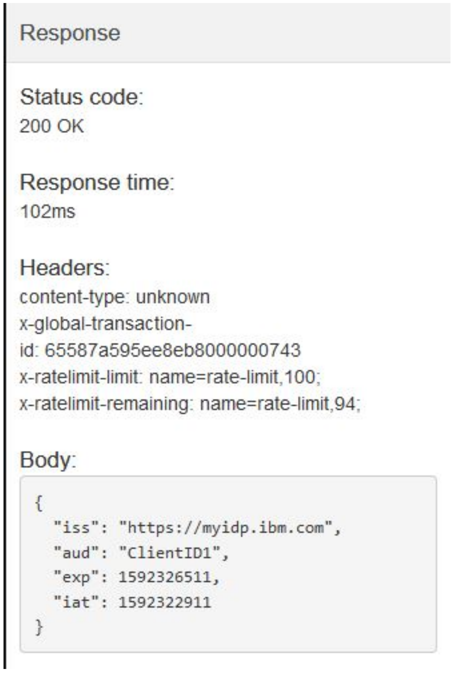
No response received. Causes include a lack of CORS support on the target server, the server being unavailable, an untrusted certificate being encountered or Mutual SSL authentication is required.

Clicking the link below will open the server in a new tab. If the browser displays a certificate issue, you may choose to accept it and return here to test again.

https://example.com/eb-org/sandbox/findbranch/details

Response time:
142ms

6. Click **Invoke** again. The response contains branch data.



Status code:
200 OK

Response time:
102ms

Headers:
content-type: unknown
x-global-transaction-id: 65587a595ee8eb8000000743
x-ratelimit-limit: name=rate-limit,100;
x-ratelimit-remaining: name=rate-limit,94;

Body:

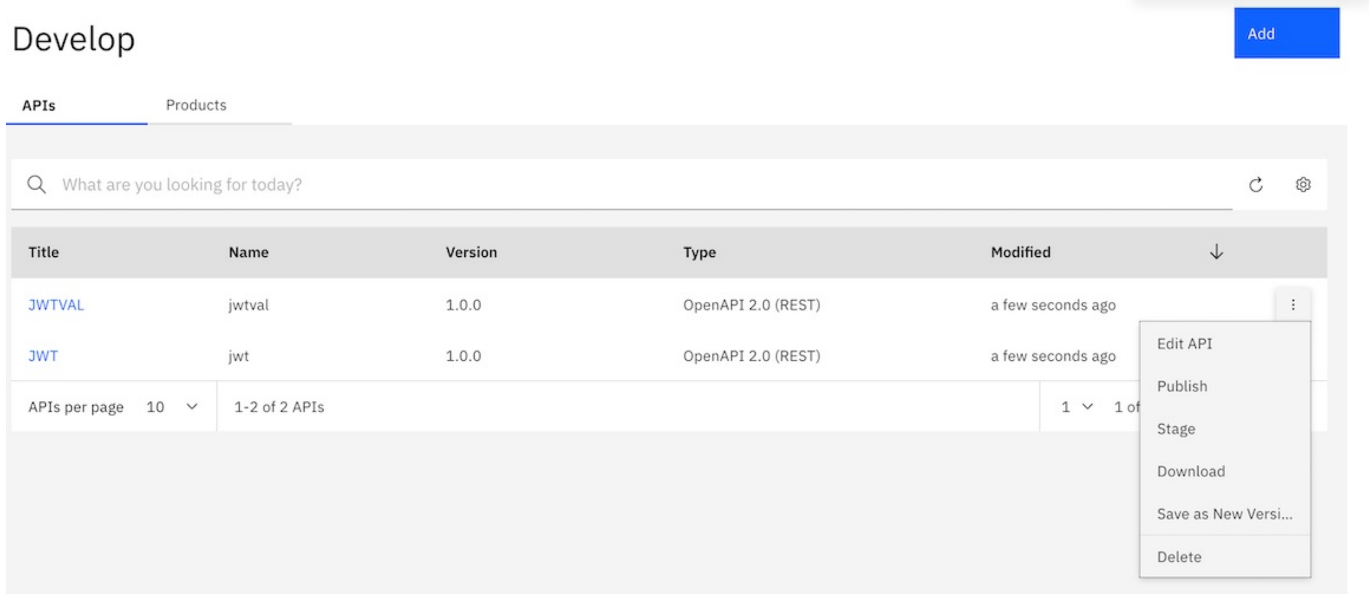
```
{
  "iss": "https://myidp.ibm.com",
  "aud": "ClientID1",
  "exp": 1592326511,
  "iat": 1592322911
}
```

Manage your API definition

Now that your new API works correctly, you can manage this API. To see your immediate options, take the following steps.

1. Click the **Develop** icon ✎ on the navigation bar.
2. Click the **Options** icon ⋮ alongside the Mapper API.

Cookie Preferences



Title	Name	Version	Type	Modified
JWTVAL	jwtval	1.0.0	OpenAPI 2.0 (REST)	a few seconds ago
JWT	jwt	1.0.0	OpenAPI 2.0 (REST)	a few seconds ago

3. Select **Download**.

What you did in this tutorial

In this tutorial, you completed the following activities:

– Created a new API definition that validates a JSON Web Token (JWT).

– Tested the new API.

Parent topic:

→ API Manager tutorials