

Add policy for the application

After you deployed the application to the CloudHub, there is no limitation on access to the application. As you can see, anyone can access to the application via the application URL: <http://secondapp-instance.us-e2.cloudhub.io/api/contacts>

You can apply some policies to the application such as authentication or limit rate, etc. Since you applied the policies to the application, the end user must have necessary conditions that the application requires to access to the application.

In this article, we will try to add the authentication to the application.

Add new policy

Go to the API Manager then click on the application to go to the application detail

Manage Kubernetes-based non-Mule microservices with Anypoint Service Mesh. [Read the documentation](#) or [Watch the video](#)

API Administration (Sandbox)

SANDBOX

Manage API ▼ Promote from environment

Search ×

[All](#) [Favorites](#) [Active](#)

[Envir](#)

API Name	Version	Status	Client Applications	Create Date
▼ SecondApp				1 v
	☆ 1.0.0	● Active	0	03-

Click on the Policy menu, then click Apply new Policy button

Manage Kubernetes-based non-Mule microservices with Anypoint Service Mesh. [Read the documentation](#) or [Watch the video](#)

API Administration (Sandbox)SecondApp (1.0.0) - Policies

SANDBOX

← API Administration

Alerts

Contracts

Policies

SLA Tiers

Settings

SecondApp1.0.0

API Status: ● ActiveAsset Version: 1.0.1 Latest ⓘType: RAML/OAS

[⊕ Add consumer endpoint](#)

Mule runtime version: 4.3.0-20210304

Automated Policies

There are no automated policies applied for the selected runtime version.

API level policies

[Apply New Policy](#)

Expand Client ID enforcement, then select one version, then click Configure Policy button

Select Policy

All Categories

All Mule Versions

Policies	Min Mule Version
Client ID enforcement	
<input checked="" type="radio"/> 1.2.3 ⓘ	4.1.0
<input type="radio"/> 1.2.2 ⓘ	4.1.0
<input type="radio"/> 1.2.1 ⓘ	4.1.0
<input type="radio"/> 1.2.0 ⓘ	4.1.0
<input type="radio"/> 1.1.3 ⓘ	4.1.0
<input type="radio"/> 1.1.2 ⓘ	4.1.0
<input type="radio"/> 1.1.1 ⓘ	4.1.0

n (Sandbox) SecondApp (1.0.0) - Policies Apply Client ID enforcement policy

Origin of the Client ID and Client Secret credentials.

☐ HTTP Basic Authentication Header

☒ Custom Expression

Client ID Expression *

Mule Expression to be used to extract the Client ID from API requests

`#[attributes.headers['client_id']]`

Client Secret Expression

Mule Expression to be used to extract the Client Secret from API requests

`#[attributes.headers['client_secret']]`

Method & Resource conditions

☒ Apply configurations to all API methods & resources

☐ Apply configurations to specific methods & resources

Cancel Apply

Click Apply button. Wait a while and then try to access to the application endpoint URL <http://secondapp-instance.us-e2.cloudhub.io/api/contacts>, you can see it shows Authentication error message

```
← → ↺ ⚠ Not secure | secondapp-instance.us-e2.cloudhub.io/api/contacts
{
  "error": "Authentication denied."
}
```

Add SLA tier

The SLA tier will allow us approve the client application by manual or automation, limit number of request also.

Go to the SLA tiers menu of the application

≡

API Manager

🔔

Manage Kubernetes-based non-Mule microservices with Anypoint Service Mesh. [Read the documentation](#) or [Watch](#)

API Administration (Sandbox)

SecondApp (1.0.0) - SLA Tiers

SANDBOX

← API Administration

Alerts

Contracts

Policies

SLA Tiers

Settings

SecondApp 1.0.0

API Status: ● Active Asset Version: 1.0.1 Latest Type: RAML/OAS

[+ Add consumer endpoint](#)

Mule runtime version: 4.3.0-20210304

Add SLA tier

🔍 Search

There are no SLA tiers for this API version.

Click Add SLA tier

Add SLA tier

Name *

Second App SLA

Description

Description

Approval *

Automatic

Limits

# of Reqs *	Time Period *	Time Unit *	
1000	1	Second	<input checked="" type="checkbox"/> visible

[+ Add Limit](#)

Cancel

Add

Input the Name, add the limit of the request if you want, then click Add button.

Microservices-based non-Mule microservices with Anypoint Service Mesh. [Read the documentation](#) or [Watch the webinar](#)

Sandbox) SecondApp (1.0.0) - SLA Tiers

SecondApp 1.0.0

API Status: ● Active Asset Version: 1.0.1 Latest ⓘ Type: RAML/OAS

[+ Add consumer endpoint](#)

Mule runtime version: 4.3.0-20210304

[Add SLA tier](#)

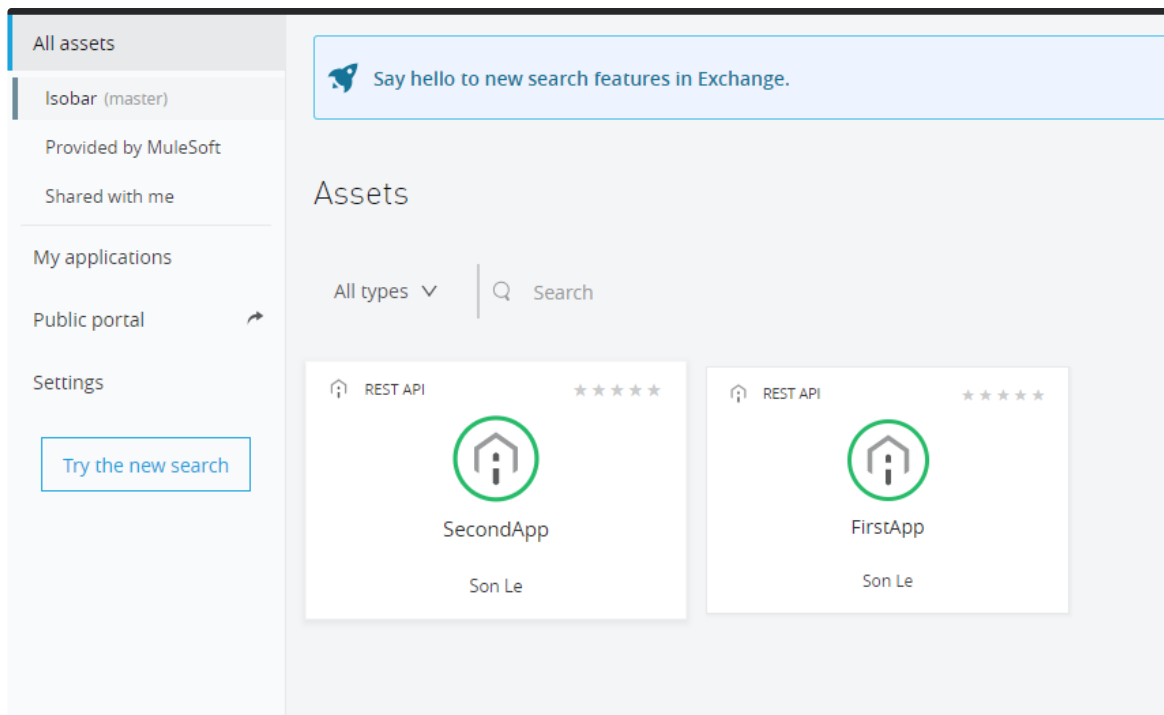
Name	Limits	Applications	Status	Approval
Second App SLA	1	0	Active	Auto

Now, we need to get the client ID and client secret for the authentication

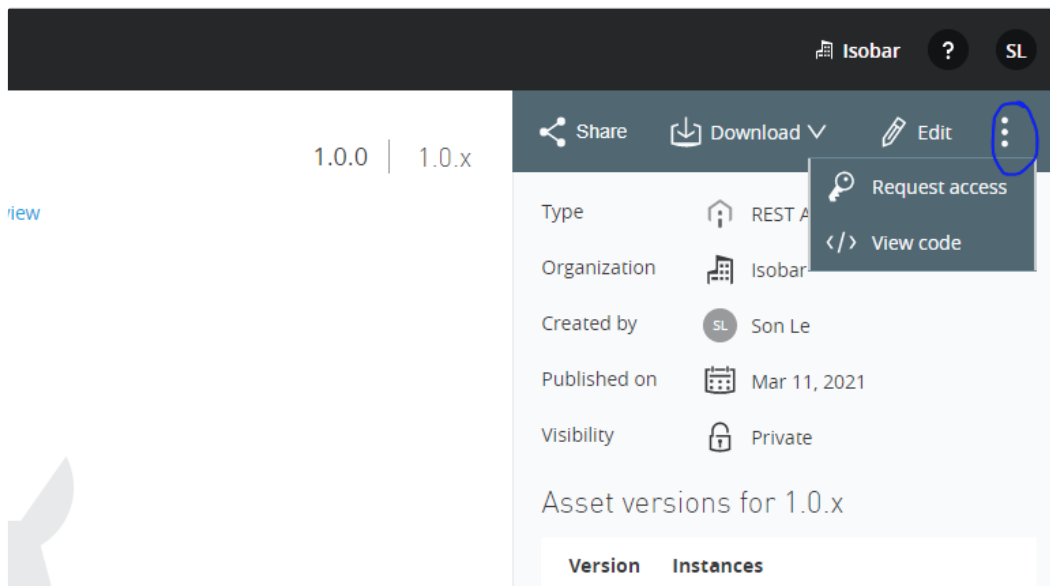
Go to the Exchange

The image shows the Anypoint Platform navigation menu on the left and a 'Recently Visited' overlay on the right. The navigation menu includes: Design Center, Exchange (highlighted), Management Center, Access Management, API Manager, Runtime Manager, Visualizer, and Monitoring. The 'Recently Visited' overlay lists: SecondApp (REST API, 21 minutes ago), SecondApp - 1.0.0 - 16684597 (API Instance, 3 hours ago), SecondApp (API specification, 1 day ago), FirstApp (API specification, 1 week ago), and FirstApp (REST API, 1 week ago). A message at the bottom states: 'This page currently doesn't contain a description. Click Edit to add text, images, videos, code blocks, etc...'

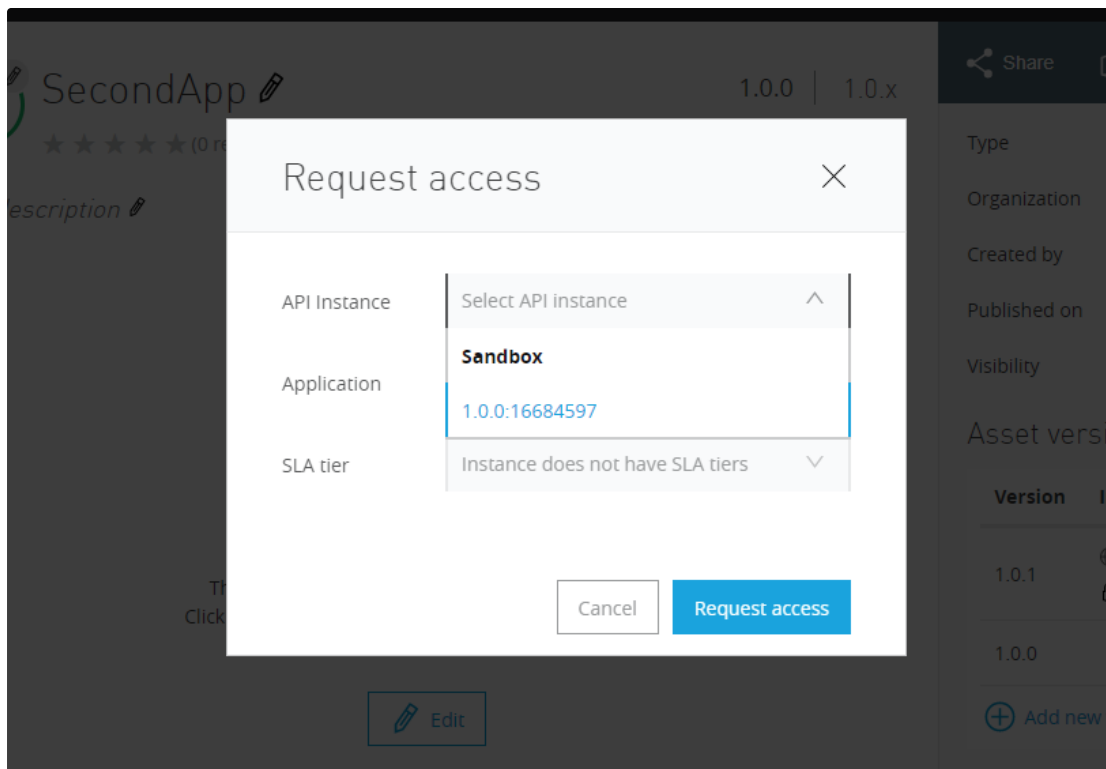
Select the API that we want to create client application



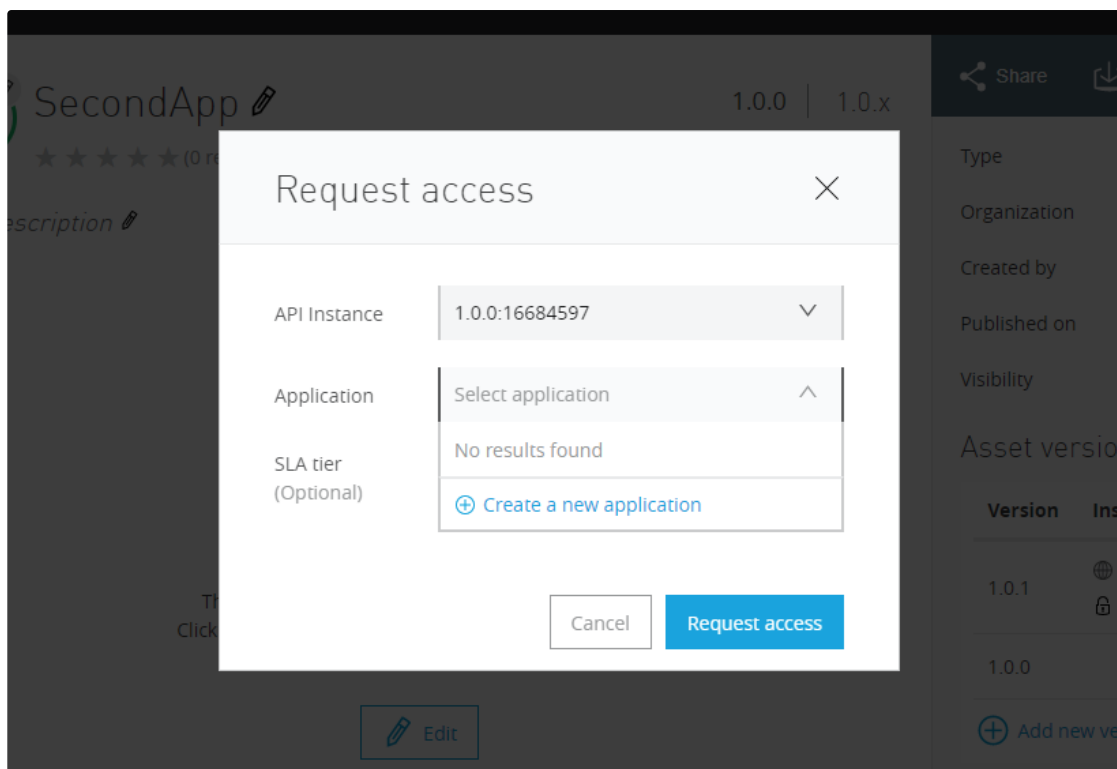
In the right corner, click 3 dots dropdown button, then select Request Access



Select the API instance



On the Application field, select create new application



Create new application

Application Name: Second App Application

Description (Optional): Write a description here...

Application URL (Optional): https://yourcompany.com/applicationURL

OAuth 2.0 redirect URIs (Optional): https://yourcompany.com/callback

Buttons: Cancel, Create

Fill the Application Name and click Create button

Request access

API Instance: 1.0.0:16684597

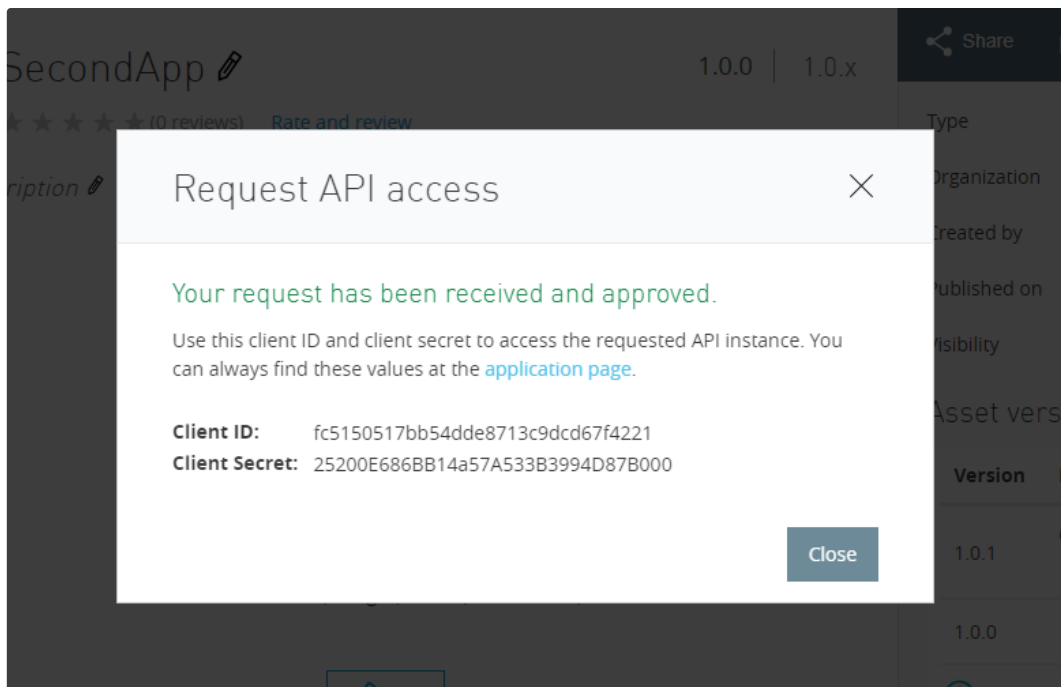
Application: Second App Application

SLA tier: Second App SLA

Requests	Time period	Time unit
1000	1	Second

Buttons: Cancel, Request access

Select the SLA tier we created above then click request access



Store the client ID and client Secret to use later.

Open Postman application, create new request, on the Headers tab, add the client_id and client_secret with the values that we create above.

Untitled Request BUILD

GET ▼ Send ▼

Params Authorization **Headers (8)** Body Pre-request Script Tests Settings

Headers 6 hidden

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	client_id	fc5150517bb54dde8713c9dcd67f4221			
<input checked="" type="checkbox"/>	client_secret	25200E686BB14a57A533B3994D87B000			
	Key	Value	Description		

Response

Click Send button, we will get the data

GET

http://secondapp-instance.us-e2.cloudhub.io/api/contacts

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

<input checked="" type="checkbox"/>	client_id	1c3130317005400e0713c30c00714221	
<input checked="" type="checkbox"/>	client_secret	25200E686BB14a57A533B3994D87B000	
	Key	Value	Description

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 809 ms Size: 5.06 KB Save F

Pretty Raw Preview Visualize

JSON



```
1 [
2   {
3     "name": "Name 1",
4     "company": "Company 1",
5     "id": "1",
6     "email": "Email 1"
7   },
8   {
9     "name": "Name 2",
10    "company": "Company 2",
11    "id": "2",
12    "email": "Email 2"
13  },
14  ]
```