

Walkthrough 5-5: Add client ID enforcement to an API specification

In this walkthrough, you add client ID enforcement to the API specification. You will:

- Modify an API specification to require client id and client secret headers with requests.
- Update a managed API to use a new version of an API specification.
- Call a governed API with client credentials from API portals.

Note: If you do not complete this exercise for Fundamentals, the REST connector that is created for the API and that you use later in the course will not have `client_id` authentication.

```
##RAML 1.0
version: v1
title: American Flights API

types:
  AmericanFlight: !include

traits:
  client-id-required:
    headers:
      client_id:
        type: string
      client_secret:
        type: string

/flights:
  is: [client-id-required]
  get:
```

GET

Mocking Service

https://mocksvc-proxy.anypoint.mulesoft.com/exc

Parameters

Headers

☒ `</>`

client_id*

client_secret*

+ Add custom header

Send

Copy the traits required to add authentication to the API specification

1. Return to the browser tab with the Settings page for American Flights API (v1) in API Manager.
2. In the left-side navigation, select Policies.
3. Click the RAML snippet link for the rate limiting – SLA based policy.

API Manager

API Administration (Sandbox) American Flights API (v1) - Policies

Consumer endpoint: http://training4-american-api-mule.cloudhub.io/

View configuration details >

View Analytics Dashboard >

SANDBOX

API Administration

Alerts

Client Applications

Policies

SLA Tiers

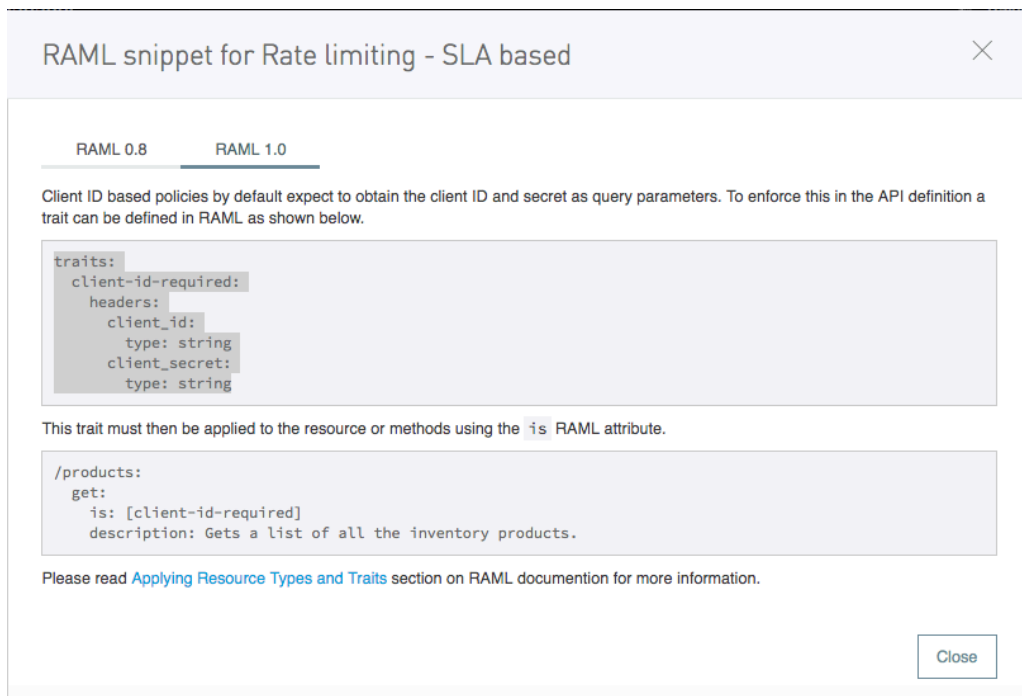
Settings

Apply New Policy

Edit policy order

Name	Category	Fulfills	Requires
> Rate limiting - SLA based ⓘ	Quality of service	SLA Rate Limiting, Client ID required	RAML snippet

4. In the RAML snippet for Rate limiting – SLA based dialog box, select RAML 1.0.
5. Copy the value for the traits.



6. Click Close.

Add authentication headers to the API specification

7. Return to the browser tab with your API in Design Center.
8. Go to a new line after the types declaration and paste the traits code you copied.

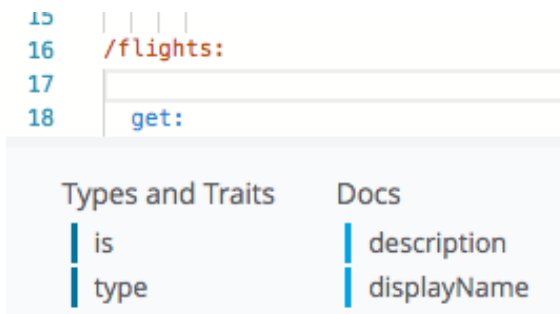
```

1  #%RAML 1.0
2  version: v1
3  title: American Flights API
4
5  types:
6    AmericanFlight: !include exchange_modules,
7
8  traits:
9    client-id-required:
10     headers:
11       client_id:
12         type: string
13       client_secret:
14         type: string
15
16  /flights:

```

9. Go to a new line after the /flights resource declaration and indent.

10. In the shelf, select is.



11. Add empty array brackets.

```
16 | /flights:  
17 |   is: []  
18 |   get:
```

12. Make sure the cursor is inside the brackets and in the shelf, select client-id-required.

```
16 | /flights:  
17 |   is: [client-id-required]  
18 |   get:
```

13. Repeat this process so the trait is applied to all methods of the {ID} resource as well.

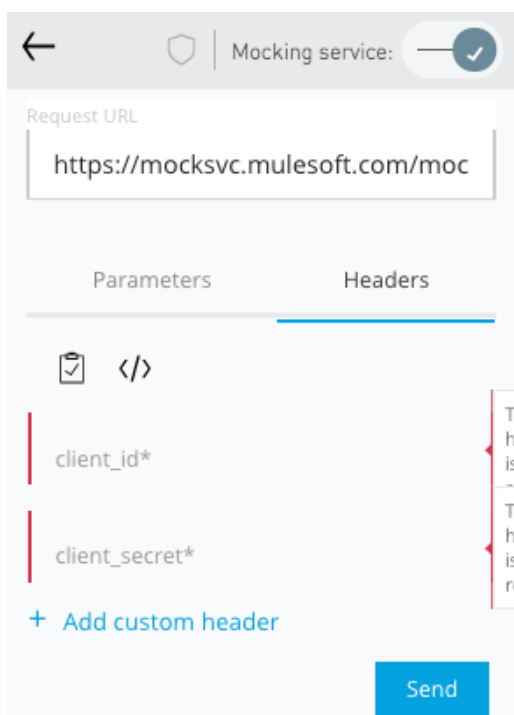
```
44 | /{ID}:  
45 |   is: [client-id-required]  
46 |   get:
```

Test the API in the API console in Design Center

14. In the API console, turn on the mocking service.

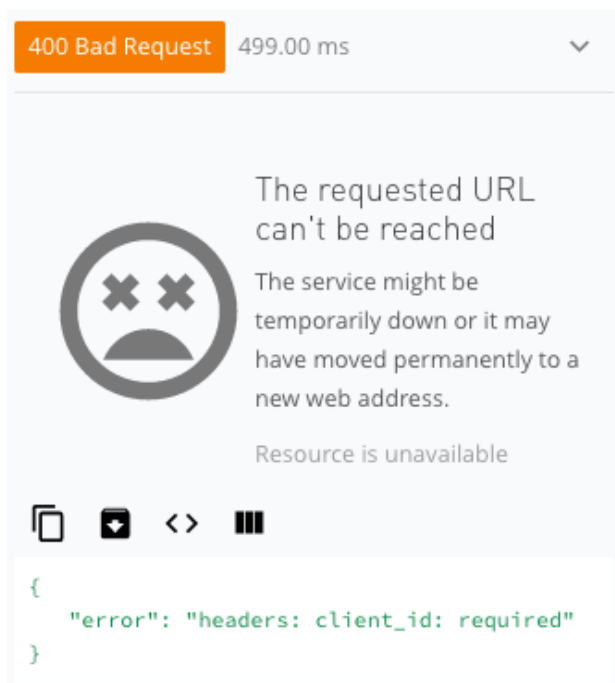
15. Select one of the resources and click Try it.

16. Select the Headers tab; you should now see fields to enter `client_id` and `client_secret`.



The screenshot shows the MuleSoft Mocking Service interface. At the top, there is a back arrow, a shield icon, and a toggle for "Mocking service:" which is turned on. Below this is a "Request URL" field containing `https://mocksvc.mulesoft.com/moc`. There are two tabs: "Parameters" and "Headers", with "Headers" being the active tab. Under the "Headers" tab, there are two input fields: `client_id*` and `client_secret*`. To the right of these fields, there is a vertical scrollbar. Below the input fields is a blue button labeled "Send".

17. Click Send; you should get a 400 Bad Request response with a message that a `client_id` header is required.



The screenshot shows the response from the MuleSoft Mocking Service. At the top, there is a status bar with "400 Bad Request" in an orange box, "499.00 ms", and a dropdown arrow. Below this is a large error message area. On the left is a circular icon with two 'X's and a sad face. To the right of the icon, the text reads: "The requested URL can't be reached", "The service might be temporarily down or it may have moved permanently to a new web address.", and "Resource is unavailable". At the bottom of the error message area are four icons: a document, a download arrow, a code icon, and a hamburger menu. Below the error message area is a code block containing the following JSON:

```
{
  "error": "headers: client_id: required"
}
```

18. Enter *any* values for the client_id and client_secret and click Send; you should get a 200 response with the example results.

The screenshot shows the MuleSoft API Mocking service interface. At the top, there is a back arrow, a shield icon, and a toggle switch for 'Mocking service' which is turned on. Below this is a 'Request URL' field containing 'https://mocksvc.mulesoft.com/mocks'. There are two tabs: 'Parameters' (selected) and 'Headers'. Under the 'Parameters' tab, there is a 'Query parameters' section with a checkbox for 'Show optional parameters'. Below this, there are two input fields: 'client_id*' with the value '432' and 'client_secret*' with the value '765'. A blue 'Send' button is located to the right of these fields. Below the 'Send' button, the response is shown: '200 OK' in a green box, followed by '506.59 ms' and a dropdown arrow. At the bottom, there are icons for copy, download, code, and a list icon. Below these icons, the response body is displayed as a JSON array:

```
[Array[2]
  -0: {
    "ID": 1,
    "code": "ER38sd",
    "price": 400
```

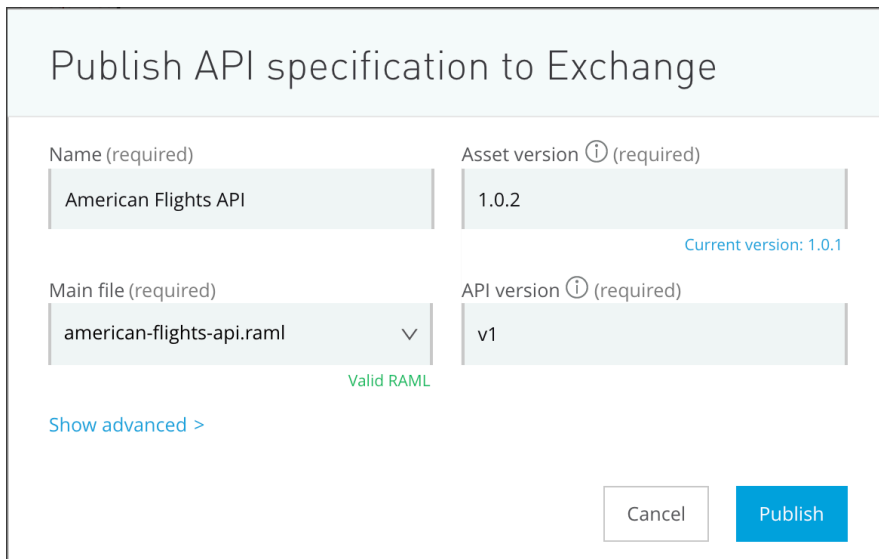
Publish the new version of the API to Exchange

19. Turn off the mocking service.

The screenshot shows the MuleSoft API Mocking service interface with the 'Mocking service' toggle switch turned off. The 'Request URL' field is visible below the toggle.

20. Click the Publish to Exchange button.

21. In the Publish API specification to Exchange dialog box, note the asset version and click Publish.

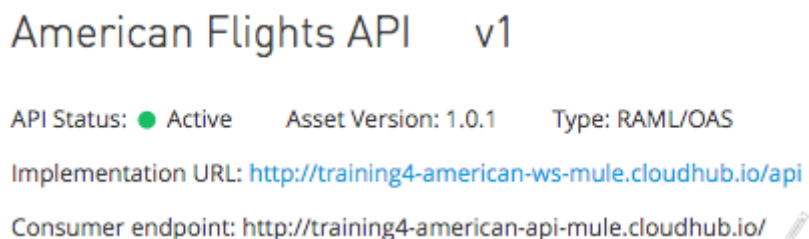


The dialog box is titled "Publish API specification to Exchange". It contains four input fields arranged in a 2x2 grid. The top-left field is "Name (required)" with the value "American Flights API". The top-right field is "Asset version ⓘ (required)" with the value "1.0.2" and a blue link "Current version: 1.0.1" below it. The bottom-left field is "Main file (required)" with a dropdown menu showing "american-flights-api.raml" and a green status "Valid RAML" below it. The bottom-right field is "API version ⓘ (required)" with the value "v1". At the bottom left is a blue link "Show advanced >". At the bottom right are two buttons: "Cancel" and "Publish".

22. After the API is published, click Done in the Publish API specification to Exchange dialog box.

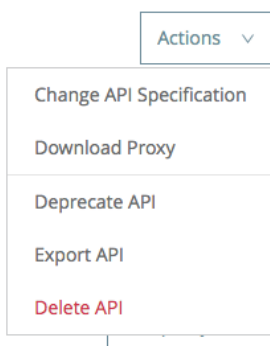
Update the managed API instance to use the new version of the API specification

23. Return to browser tab with American Flights API (v1) in API Manager.
24. Locate the asset version displayed at the top of the page; you should see 1.0.1.



The page displays the API details for "American Flights API v1". It shows the API Status as "Active" with a green dot, the Asset Version as "1.0.1", and the Type as "RAML/OAS". Below this, the Implementation URL is "http://training4-american-ws-mule.cloudhub.io/api" and the Consumer endpoint is "http://training4-american-api-mule.cloudhub.io/" with a pencil icon for editing.

25. Click the Actions button in the upper-right corner and select Change API Specification.



The image shows a dropdown menu titled "Actions" with a downward arrow. The menu contains five options: "Change API Specification", "Download Proxy", "Deprecate API", "Export API", and "Delete API" (which is highlighted in red).

26. In the Change Specification dialog box, select the latest asset version, 1.0.2.

27. Click Change.

Change Specification

Asset Version: 1.0.2

Change of specification requires redeploy to take effect

Cancel

Change

Redeploy a new proxy

28. In the left-side navigation, select Settings.

29. Scroll down to the Deployment Configuration settings; the Redeploy button should be disabled.

30. For the runtime version, select 4.x.x again; the Redeploy button should now be enabled.

Deployment Configuration

Runtime version: 4.x.x

Proxy application name: training4-american-api-mule .cloudhub.io

Redeploy

31. Click Redeploy.

32. In the Deploying to CloudHub dialog box, click the Click here link to see the logs.

33. Watch the logs and wait until the proxy application is redeployed.

Runtime Manager

SANDBOX

Applications

Dashboard

Insight

Logs

Application Data

training4-american-api-mule

Live Console

Q Search

Advanced

17:30:15.404	04/18/2018	worker-0	agw-policy-set-deployment.01	INFO
API ApiKey{id='11655540'} is now unblocked (available).				
17:31:14.756	04/18/2018	Deployment	system	SYSTEM
Application was updated successfully with zero downtime. The new version of your application has been launched and the old version has been stopped.				
17:31:14.890	04/18/2018	Deployment	system	SYSTEM
Your application is started.				

Deployments

Today

> 17:28 - Deployment

> 14:12 - Deployment

System Log

Worker-0

34. Close the browser tab.

35. Return to the browser tab with API Manager and click Close in the Deploying to CloudHub dialog box.

Test the rate limiting – SLA based policy in the API console in Exchange

36. Return to the browser tab with Exchange.

37. Return to the home page for the API (and refresh if necessary); you should see the new asset version listed.

Asset versions for v1

Version	Instances
1.0.2	<div><div>Mocking Service</div><div>Sandbox - Rate limiting - SLA based policy</div></div>
1.0.1	
1.0.0	

38. Click the GET method for the flights resource and select the Headers tab; you should see required text fields for client_id and client_secret and no longer need to add the headers manually for each request.

Note: You will test and use the authentication with the REST connector later in the Fundamentals course.

GET

Mocking Service

https://mocksvc-proxy.anypoint.mulesoft.com/exc

Parameters Headers

☒ </>

client_id*

client_secret*

+ Add custom header

Send

39. Close all Anypoint Platform browser windows and tabs.