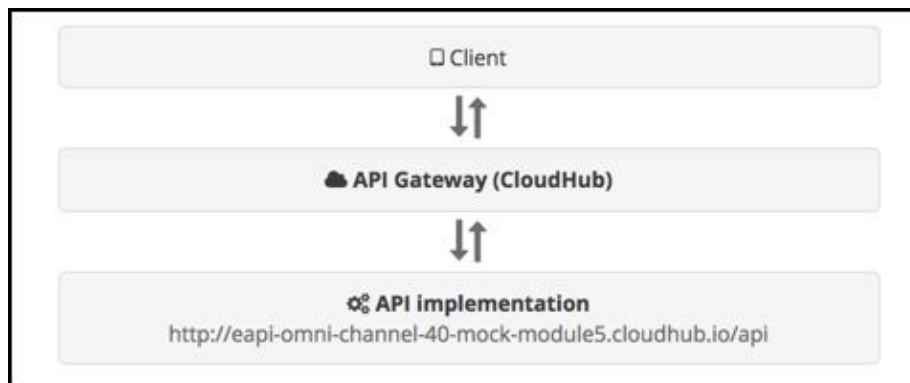


Módulo 5: API Manager

A gestão de API é essencial para uma arquitetura baseada em API. Através do **Anypoint API Manager** podemos aplicar a governança diretamente a uma API ou através de um proxy gateway intermediário. É possível executar as políticas **on-premise**, em uma nuvem privada ou no **CloudHub**. Neste laboratório, vamos configurar um API Gateway para proteger uma aplicação.



Ao invés de fornecer acesso direto a URL <http://workshop-omni-channel-mock-service-v40.cloudhub.io/api>, os clientes acessam a API através do API Gateway, que atua como um elemento intermediário para controlar os acessos.

Etapas do laboratório

[Etapa 1: Configurar o proxy da API](#)

[Etapa 2: Testar o proxy da API](#)

[Etapa 3: Aplicar a política de Rate Limiting](#)

[Etapa 4: Testar o proxy da API com as políticas](#)

[Etapa 5: Remova a política de Rate Limiting](#)

[Etapa 6 \(Opcional\): Criar SLA layers](#)

[Etapa 7 \(Opcional\): Aplicar Rate Limiting com SLA](#)

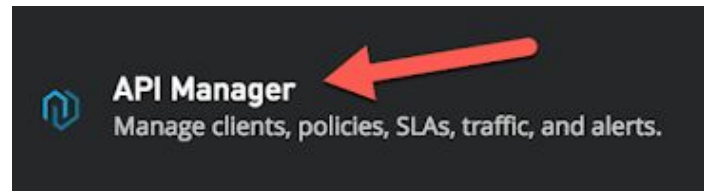
[Etapa 8: Solicitar acesso à API](#)

[Etapa 9: Testar uma API protegida](#)

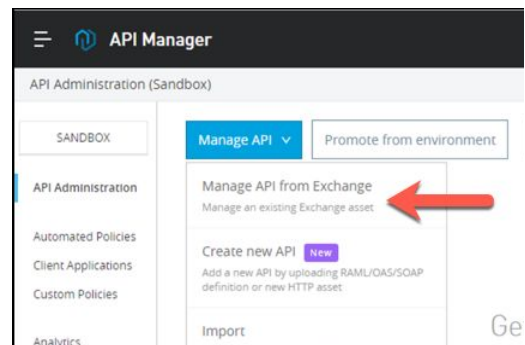
Etapa 1: Criar um gateway de API

Para este laboratório, vamos configurar um proxy para a implementação da nossa API.

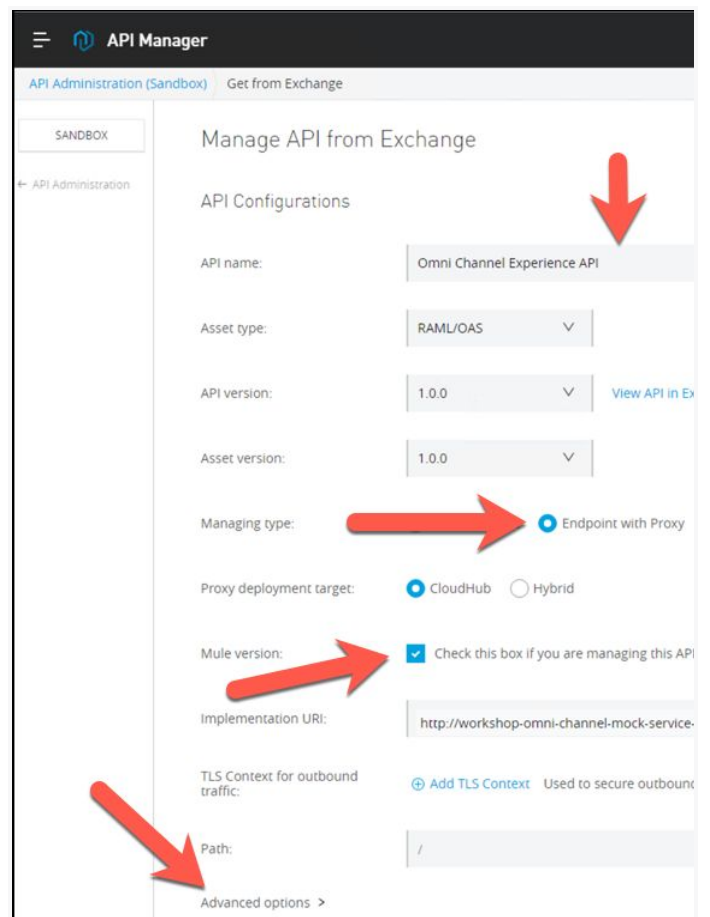
1. Clique no ícone do **API Manager** para gerenciar as API disponíveis
2. Selecione **SANDBOX** como o ambiente



3. Agora vamos configurar um proxy para esta API, clicando em **Manage API** e selecione **Manage API from Exchange**



4. Configure a API com as seguintes informações:
 - a. API name: **Omni Channel Experience API**
 - b. API version: selecione a versão **1.0.0**
 - c. Asset version: selecione a versão **1.0.0**
 - d. Managing type: **Endpoint with proxy**
 - e. Proxy deployment target: **CloudHub**
 - f. Mule version: **Check this box if you are managing this Mule 4 or above**
 - g. Implementation URI: <http://workshop-omni-channel-mock-service-v40.cloudhub.io/api>
 - h. Path: **/**



Se quiser, podemos preencher informações mais customizadas.

5. Selecione **Opções avançadas**
6. API instance label: **<username>**
7. Pressione **Save**
8. Depois de salvar a configuração, a seção de implantação do proxy será apresentada

API instance label: ① Recommended if you have multiple managed instances, use the same API

Port: ①

Response timeout: ① (Optional)

☐ Reference user domain ①

9. Configure com as seguintes informações:
 - a. Runtime version: selecione **4.x.x**
 - b. Proxy application name: **<username>-mythical-omni-channel-api-proxy**. Essa propriedade definirá o URL e o nome da aplicação Mule
10. Pressione o botão **Deploy**

Deployment Configuration ▾

Runtime version: ▾

Proxy application name: ① cloudhub.io

☐ Update application if exists

11. O processo de deployment inicia e são executados em etapas.
12. Ao final, clique em **Close**

1 Deploying proxy 2 Starting application 3 Deploy successful

Your proxy application was successfully deployed to CloudHub. [Click here](#) to view the application in CloudHub.

13. Depois de implantado, na parte superior da página, você verá o **API Status**. A cor **verde (Active)** indica que sua API foi implantada com sucesso.

API Manager

API Administration (Sandbox) Omni Channel Experience Af

SANDBOX

Omni Channel Exper

API Status: ● Active Asset Vers

Implementation URL: <http://worksh>

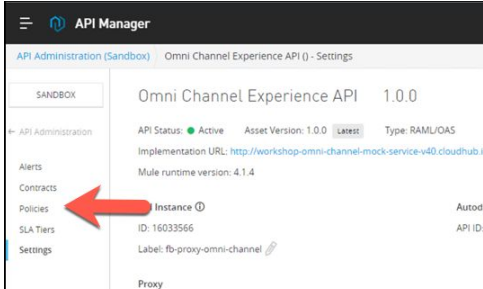
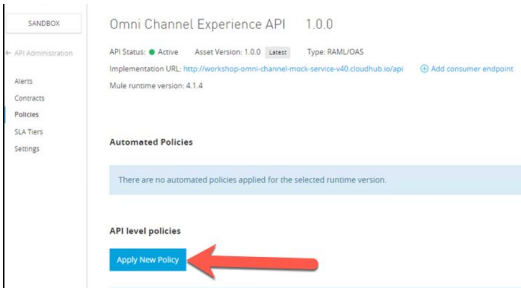
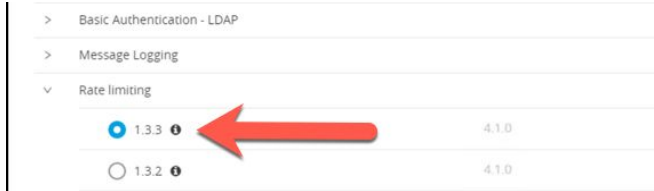
Etapa 2: Testar o gateway de API

Seu gateway agora está acessível através do **CloudHub**.

<div><div>1. Vá para o API Manager</div><div>2. No lado esquerdo, clique em Settings.</div><div>3. Copie a Proxy URL com o endereço do gateway</div></div>	<div><div><div>SANDBOX</div><div>API Administration</div><div>Alerts</div><div>Client Applications</div><div>Policies</div><div>SLA Tiers</div><div>Settings</div></div><div><div>Omni Channel Experience API 1.0.0</div><div>API Status: Active Asset Version: 1.0.0 Type: RAML/OAS</div><div>Implementation URL: http://workshop-omni-channel-mock-service-v40.cl</div><div>API Instance ID: 6785648 Label: jgw-proxy-omni-channel</div><div>Autodiscovery: API Name: groupId:ea35fd7f-d2l API Version: 1.0.0:6785648</div><div>Proxy</div><div>Proxy Application: jgw-mythical-omni-channel-api-http-proxy</div><div>Proxy URL: jgw-mythical-omni-channel-api-http-proxy.cloudhub.io</div></div></div>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Etapa 3: Aplicar a política de Rate Limiting

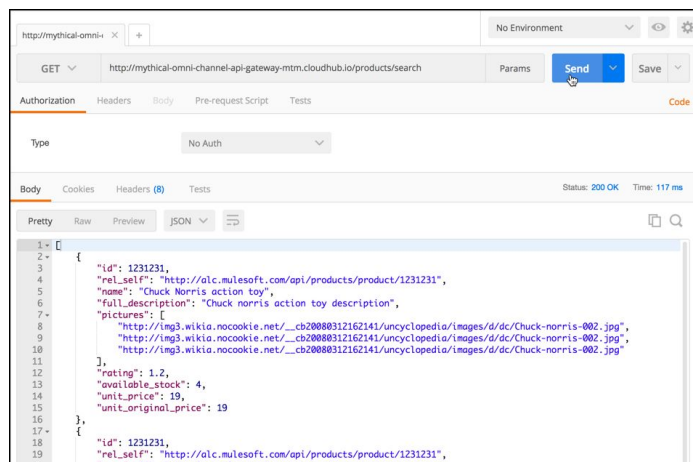
Agora que o proxy para a nossa API está configurado e sendo executado corretamente vamos adicionar uma política de **rate limiting** a nossa API.

<div>1. Vá para a página de administração da Omni Channel Experience API</div> <div>2. No lado esquerdo, clique na guia Policies.</div>							
<div>3. Clique no botão Apply New Policy</div>							
<div>4. Clique na política de Rate Limiting, selecione a última versão e Configure Policy</div>							
<div>5. Digite um número máximo de solicitações de 3 por 1 minuto:</div> <div><div><div># of Reqs: 3</div><div>Time Period: 1</div><div>Time Unit: Minute</div></div></div> <div>6. Clique em Apply</div>	<div><div>Apply Rate limiting policy</div><p>Specifies the maximum value for the number of messages processed per time period, and rejects any messages beyond the maximum. Applies rate limiting to all API calls, regardless of the source.</p><p>Identifier</p><p>For each identifier value, the set of Limits defined in the policy will be enforced independently. I.e.: # [attributes.queryParams["identifier"]].</p><div></div><p>Limits</p><p>Pairs of maximum quota allowed and time window.</p><table><tr><th># of Reqs *</th><th>Time Period *</th><th>Time Unit *</th></tr><tr><td>3</td><td>1</td><td>Minute</td></tr></table><p>Add Limit</p><p><input checked="" type="checkbox"/> Clusterizable</p><p>When using a clustered runtime with this flag enabled, configuration will be shared among all nodes.</p><p><input type="checkbox"/> Expose Headers</p><p>Defines if headers should be exposed in the response to the client. These headers are: x-ratelimit-remaining, x-ratelimit-limit and x-ratelimit-reset.</p><p>Method & Resource conditions</p><p><input checked="" type="radio"/> Apply configurations to all API methods & resources</p><p><input type="radio"/> Apply configurations to specific methods & resources</p><div><div>Cancel</div><div>Apply</div></div></div>	# of Reqs *	Time Period *	Time Unit *	3	1	Minute
# of Reqs *	Time Period *	Time Unit *					
3	1	Minute					

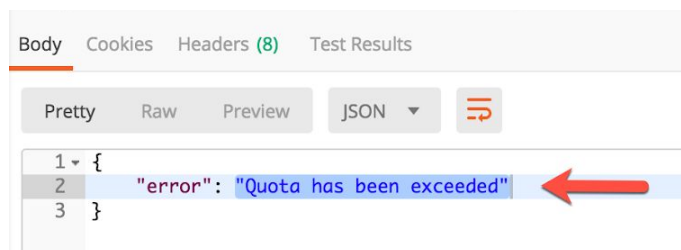
Etapa 4: Testar o gateway de API com políticas

Teste a API usando seu navegador ou Postman (mesma URL definida nos passos anteriores).

1. Acesse sua URL de proxy adicionando **/products/search** no final. Por exemplo:
<http://<proxy-url>.cloudhub.io/products/search>
2. Teste sua API realizando 3 requisições no intervalo de 1 minuto (recarregue o navegador ou pressione o botão **Send** do Postman)



3. Na terceira chamada, você receberá uma mensagem indicando que a **cota foi excedida** (*Quota has been exceeded*). Isso demonstra que sua política de **Rate Limiting** foi aplicada corretamente.



Etapa 5: Remover a política de Rate Limiting

Remova a política de Rate Limiting após os testes clicando no botão Remover conforme figura abaixo.

The screenshot shows the API Manager interface for the 'Omni Channel Experience API' (version 1.0.0). The left sidebar contains navigation links: API Administration (Sandbox), Alerts, Contracts, Policies, SLA Tiers, and Settings. The main content area displays the API details, including its status (Active), version (1.0.0), and implementation URL. Below this, there are sections for 'Automated Policies' (none applied) and 'API level policies'. A table lists the 'Rate limiting' policy, which is of the 'Quality of service' category and fulfills 'Baseline Rate Limiting'. The table has columns for Name, Category, Fulfills, and Requires. Below the table, there is a sub-table with columns for Order, Method, and Resource URI. The first row shows order 1 for 'All API Methods' on 'All API Resources'. To the right of the table, there is an 'Actions' dropdown menu with options: View Detail, Edit, Disable, and Remove. A red arrow points to the 'Rate limiting' policy name, and another red arrow points to the 'Remove' button in the actions menu.

API Manager

API Administration (Sandbox) | Omni Channel Experience API (1.0.0) - Policies

SANDBOX

Omni Channel Experience API 1.0.0

API Status: Active Asset Version: 1.0.0 Latest Type: RAML/OAS

Implementation URL: <http://workshop-omni-channel-mock-service-v40.cloudhub.io/api> Add consumer endpoint

Mule runtime version: 4.1.4

Actions

Manage CloudHub Proxy >

View API in Exchange >

View configuration details >

View Analytics Dashboard >

API level policies

Apply New Policy

Edit policy order

Name	Category	Fulfills	Requires
Rate limiting	Quality of service	Baseline Rate Limiting	

Order	Method	Resource URI
1	All API Methods	All API Resources

View Detail Actions

Edit

Disable

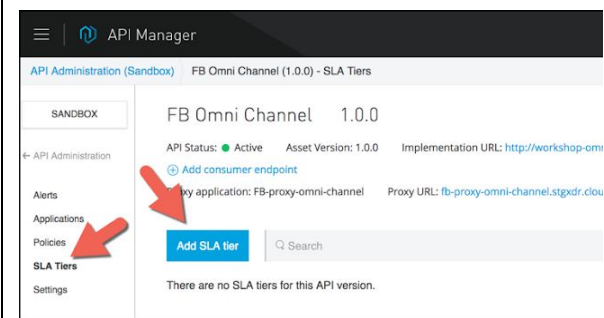
Remove

Etapa 6 (Opcional): Criar SLA layers

Podemos definir diferentes SLA para nossas APIs.

1. Em **API Manager** selecione **SLA Tiers** na barra de ferramentas à esquerda.

2. Clique em **Add SLA Tier**



3. Você vai configurar três SLA:

Nome	Aprovação	Requisições	Período de
Teste	Automático	1	1 minuto
Ouro	Manual	10	1 minuto
Platinum	Manual	100	1 minuto

4. Preencha os campos conforme a tabela acima e imagem de exemplo ao lado (Veja os valores na tabela).

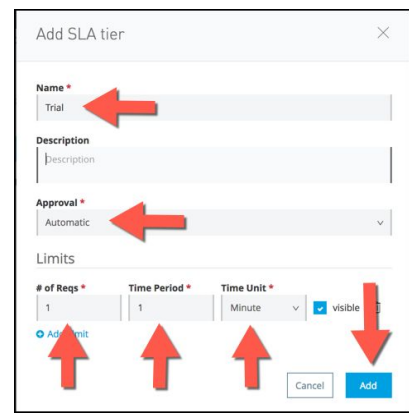
a. Name

b. Approval

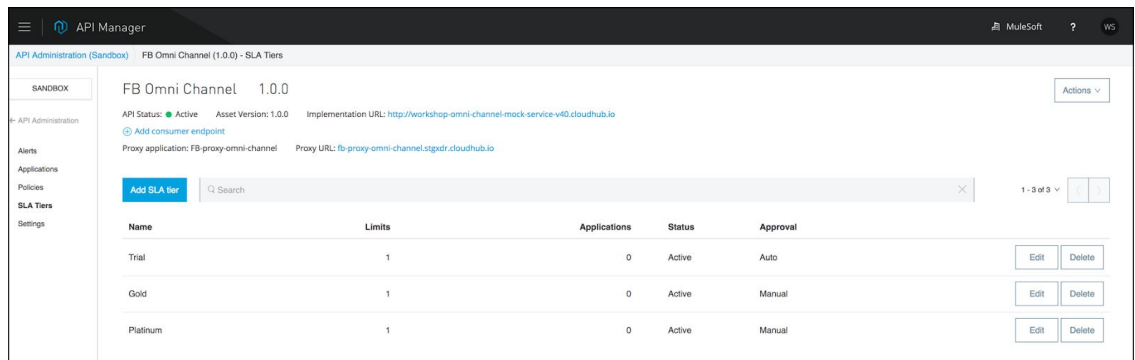
c. # of Reqs

d. Time Period

e. Time Unit

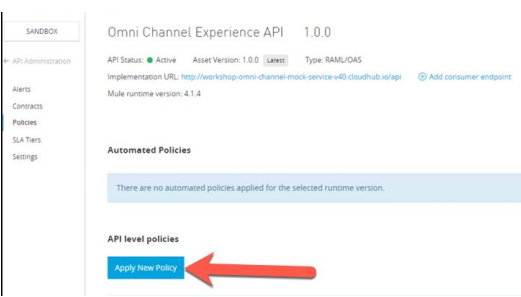
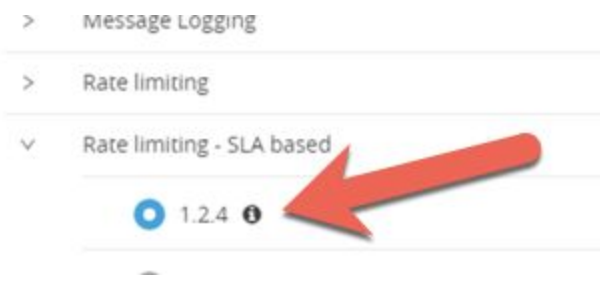
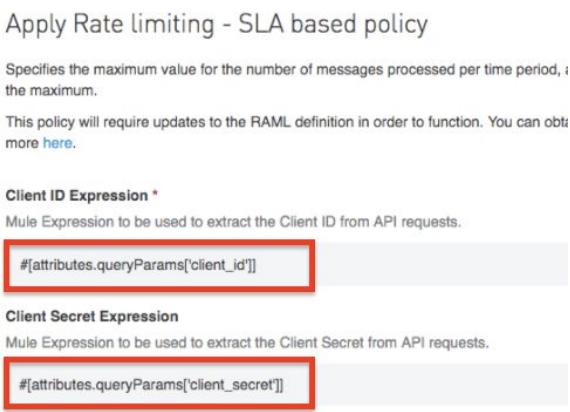
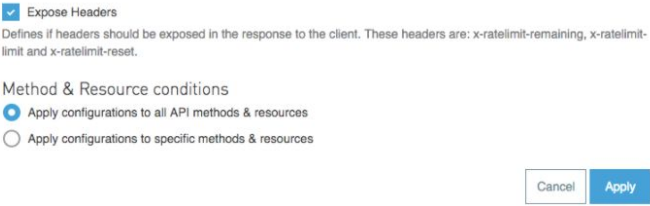


5. Suas configurações de SLA são exibidas com todas as informações que você acabou de definir. Além disso, você tem uma coluna que indica quantas aplicações estão registradas para cada SLA definido



Etapa 7 (Opcional): Aplicar Rate Limiting com SLA

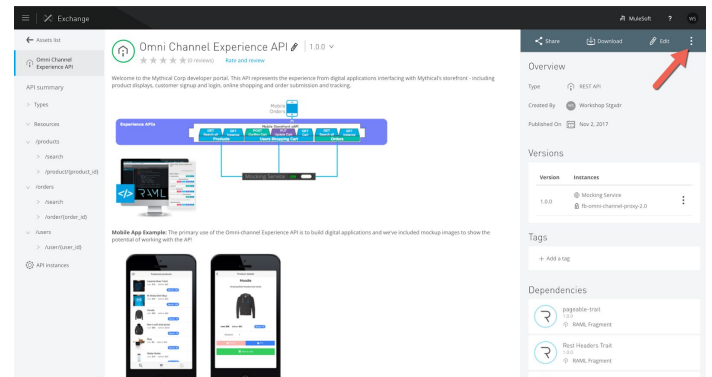
Essas políticas exigem que todas as aplicações que consomem sua API se registrem para um SLA específico. Suas credenciais deverão ser informadas em todas as chamadas para API, possibilitando identificar o consumidor e aplicar o SLA previamente contratado.

<p>Para evitar conflitos de políticas neste laboratório, verifique se não há nenhuma política aplicada a esta API.</p> <p>1. Clique em Apply New Policy.</p>	
<p>2. Selecione Rate limiting - SLA based e escolha a versão mais recente.</p>	
<p>3. Altere Client ID Expression com o valor <code># [attribute.queryParams ['client_id']]</code></p> <p>4. Altere Client Secret Expression com o valor <code># [attribute.queryParams ['client_secret']]</code></p> <p>Aviso: neste laboratório, a expressão substitui os headers são substituídos por queryParams</p>	
<p>5. Habilitar Expose Headers</p> <p>6. Clique em Apply para salvar as configurações</p>	

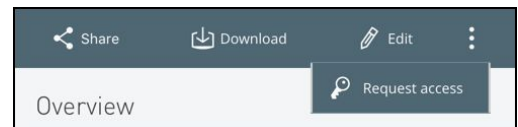
Etapa 8: Solicitar acesso à API

Nossa API foi configurada para ser protegida através de **Client Id** e **Client Secret** e com **contratos de SLA** específicos. Precisamos ir ao portal do **Anypoint Exchange** para solicitar acesso a esta API.

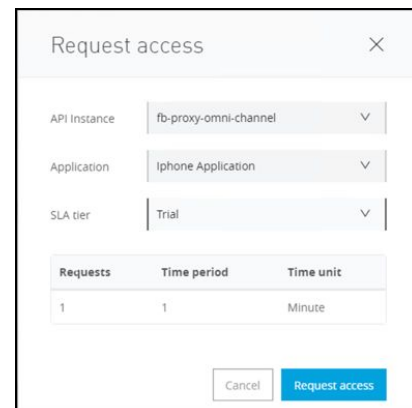
1. Vá para o **Exchange**
2. Selecione a API **Omni Channel Experience API**
3. Pressione o (...) no canto superior direito.



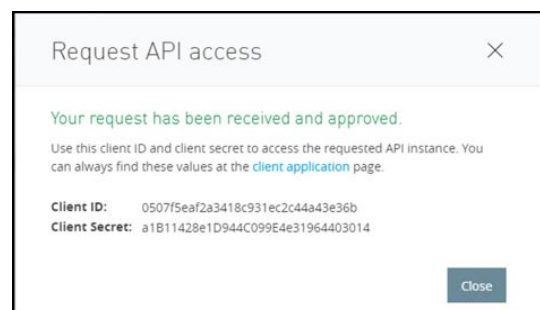
4. Selecione **Request Access**



5. Nas instâncias da API, selecione sua API.
6. Em **Application**, escolha **Create a new application**
 - o Complete a caixa de diálogo da nova aplicação (você deve criar um nome de aplicação como **Mobile Application**). Quando terminar, clique em **Create**
7. Selecionamos o **SLA Tier** que queremos utilizar: nesse exemplo inicial, escolhemos **Trial**
8. Clique em **Request Access**



9. Ao solicitar acesso será apresentado o **Client ID** e **Client Secret** associado a sua aplicação. Estes são os valores que deverão ser utilizadas em todas as chamadas a nossa API.



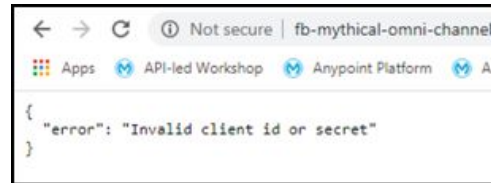
Etapa 9: Testar uma API protegida

Agora vamos testar nossa API que foi configurada nos passos anteriores.

1. Teste a API novamente usando seu navegador ou **Postman** e acesse a URL do CloudHub:

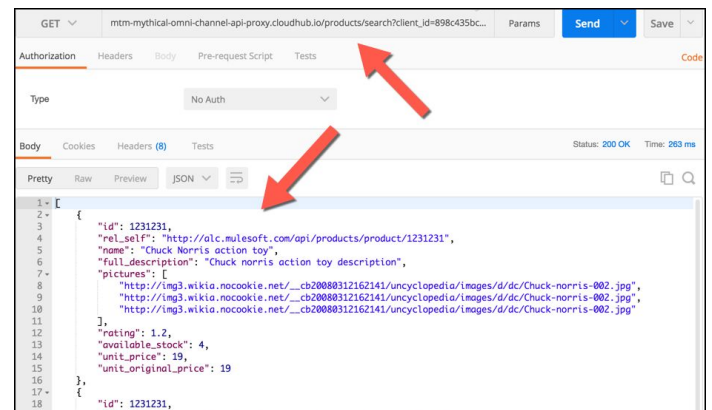
<http://<proxy>.cloudhub.io/products/search>

Como resposta devemos receber o erro: **Invalid client id or secret**

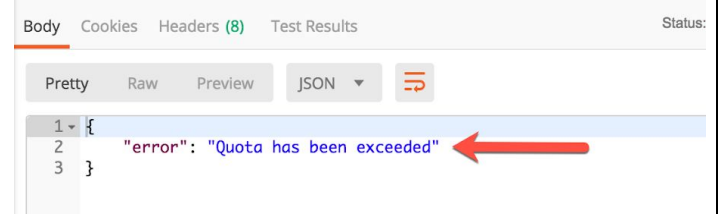


2. Adicione a URL os seguintes valores:
?client_id=<yourId>&client_secret=<yourSecret>.
Por exemplo:
http://<nome do usuário>-mythical-omni-channel-api-proxy/products/search?client_id=b466e22597b94689952bb77792cf7f8d&client_secret=53EDAFEDeA6f45dF95408596f846417F

3. Agora você deverá ser capaz de realizar as requisições a API com sucesso.



4. Execute novas requisições até verificar que o limite de acesso (baseado no SLA contratado) foi excedido.



Neste módulo, vimos os recursos de gerenciamento de APIs e aplicação de políticas que reforçam e garantem a **segurança e governança** da sua API, oferecendo um melhor **controle de como e por quem sua API será utilizada**.