Hands-On Lab:
Managing API Threat Protection

Objectives

In this hands-on lab, you will configure a second API Gateway instance acting as a threat protection layer in front of the internal API Gateway. You will define a global denial of service threat protection policy which will be imposed by the Threat Protection API Gateway before the request reaches the internal API Gateway.

Steps

1) Open the **Windows Services** panel and double-check that the following services, needed for two API Gateway instances and the native services, are up and running. If a service is not running, start the service.

   a) **Software AG Integration Server 10.11 (default)**

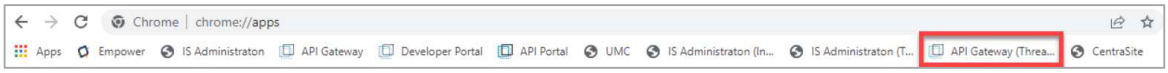
   b) **Software AG Threat Protection Integration Server 10.11**

   c) **Software AG Runtime 10.11**

   Note: You can monitor the progress of the Threat Protection Integration Server startup sequence in **Baretail** by opening the following logfile:

   C:\SoftwareAGThreatProtection\IntegrationServer\instances\default\logs\server.log

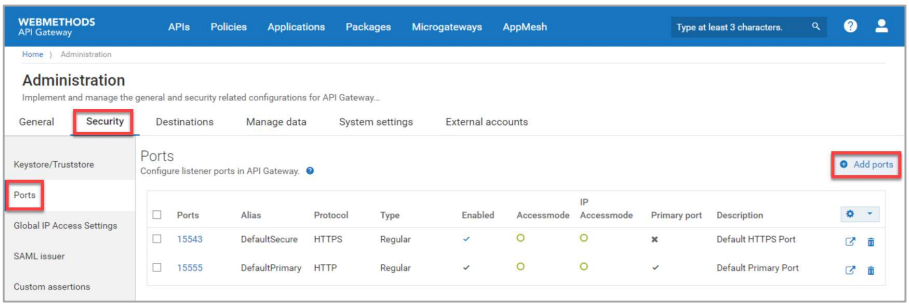
2) Open **Google Chrome**. Connect to the Threat Protection API Gateway UI by clicking the **API Gateway (Threat Protection)** bookmark. Login as **Administrator | manage.**



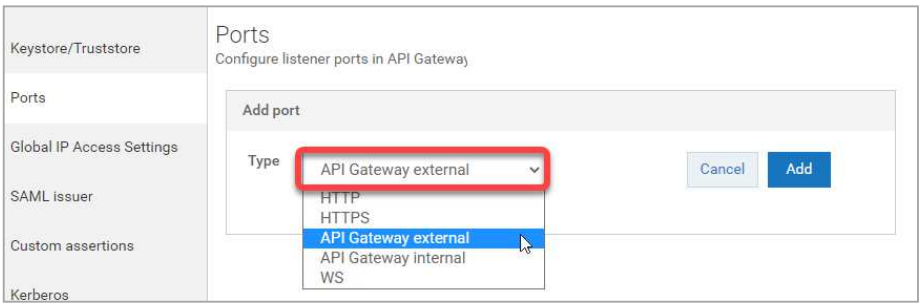
3) Configure the connectivity to the (internal) API Gateway for the Threat Protection API Gateway:

   a) In the User Menu of the Threat Protection API Gateway UI, navigate to **Administration** > **Security** > **Ports**.

   b) Click the **+ Add ports** button to create the External Port which is exposed to the API consumer.



c) Select type **API Gateway external**.



Click **Add**.

d) Provide the following properties:

i) API Gateway external listener configuration

(1) External port: **8888**

(2) Protocol: **HTTP**

(3) Alias: **ExtPortAlias**

(4) Bind Address: < ***leave empty*** >

(5) Description: **Threat Protection API Gateway external HTTP Listener Port 8888**

(6) Backlog: < ***leave the defaults*** >

(7) Keep alive timeout: < ***leave the defaults*** >

ii) API Gateway registration listener configuration

(1) Registration port: **8889**

(2) Alias: **RegPortAlias**

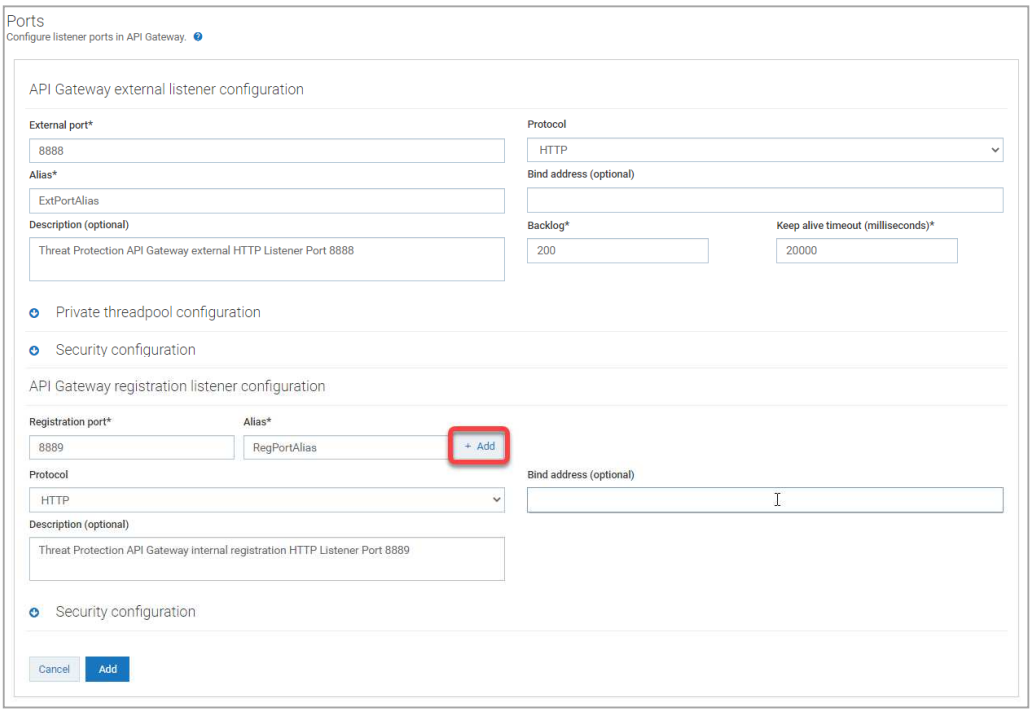
(3) Protocol: **HTTP**

(4) Bind Address: < ***leave empty*** >

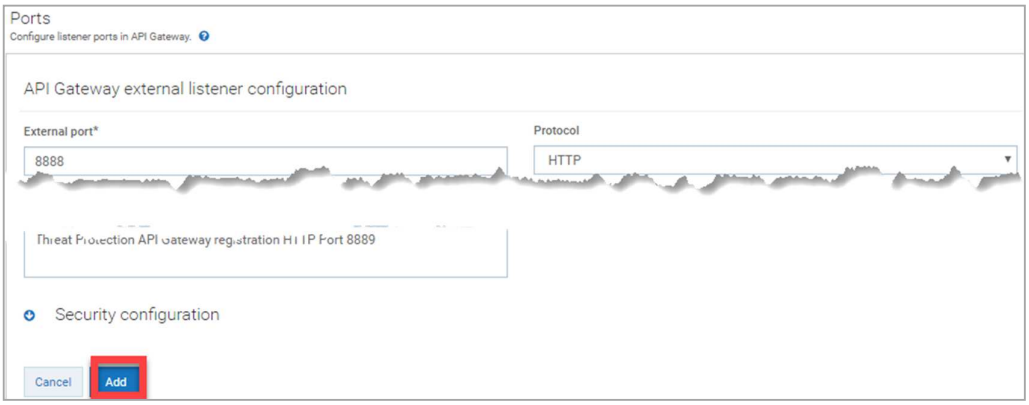
(5) Description: **Threat Protection API Gateway internal registration HTTP Listener Port 8889**

Note: The API Gateway registration listener configuration portion of this port definition configures a listener port used for an internal reverse invoke between the Threat Protection API Gateway and the internal API Gateway. We will provide the corresponding definition at the internal API Gateway in an upcoming step.

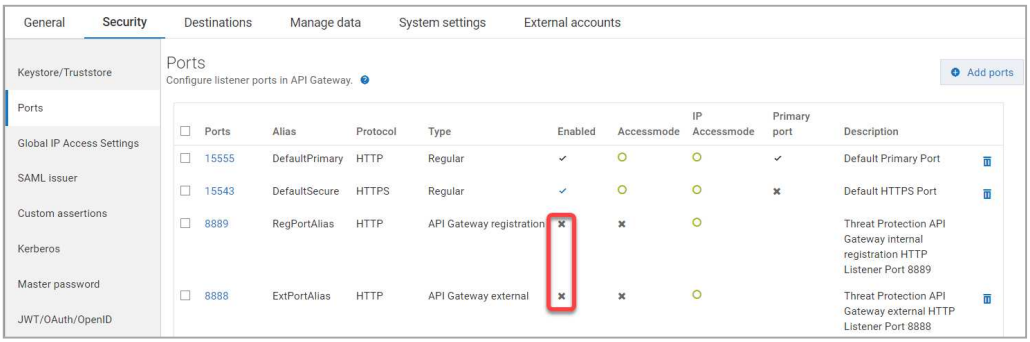
e) Click the **+Add** button next to the registration listener alias.



f) Click **Add** at the very bottom to add the external listener configuration.



g) Activate the two ports you have created in the previous step. Click the cross in the **Enabled** column of the list of ports for each. Confirm each enabling with **Yes**.



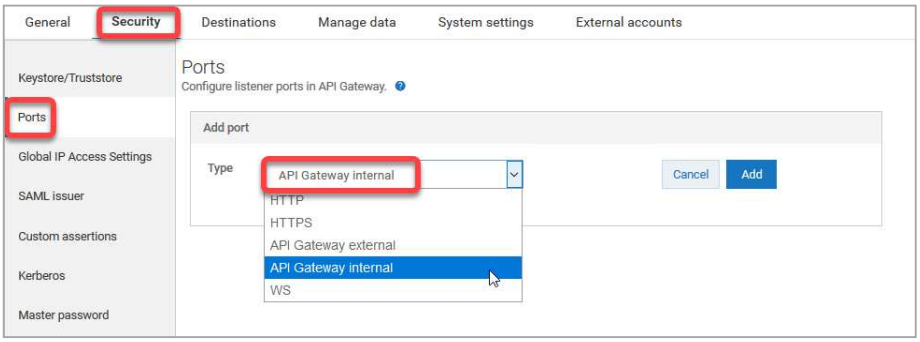
4) Configure the connectivity to the Threat Protection API Gateway for the (internal) API Gateway:

   a) Open a tab in **Mozilla Firefox** and use bookmark **API Gateway** to login to the (internal) API Gateway as **Administrator | manage**.

   b) In the User Menu, navigate to **Administration** > **Security** > **Ports**.

   c) Click the **+ Add ports** button.

   d) Select type **API Gateway internal**.



   Click **Add**.

   e) Provide the following properties:

      i) API Gateway internal listener configuration:

         (1) Protocol: **HTTP**

         (2) Description: **Internal API Gateway Port for Reverse Invoke**

         (3) Alias: **InternalReverseInvokeAlias**

         (4) Max Connections: **5**

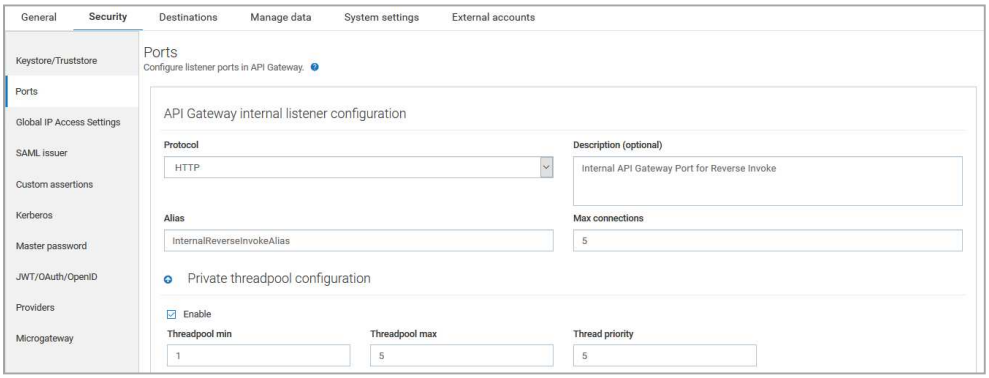
      ii) Private threadpool configuration (expand section, if collapsed):

         (1) Enable: <***checked***>

         (2) Threadpool min: **1**

         (3) Threadpool max: **5**

         (4) Thread priority: **5**



iii) API Gateway external server:

(1) Host: **localhost**

(2) Port: **8889**

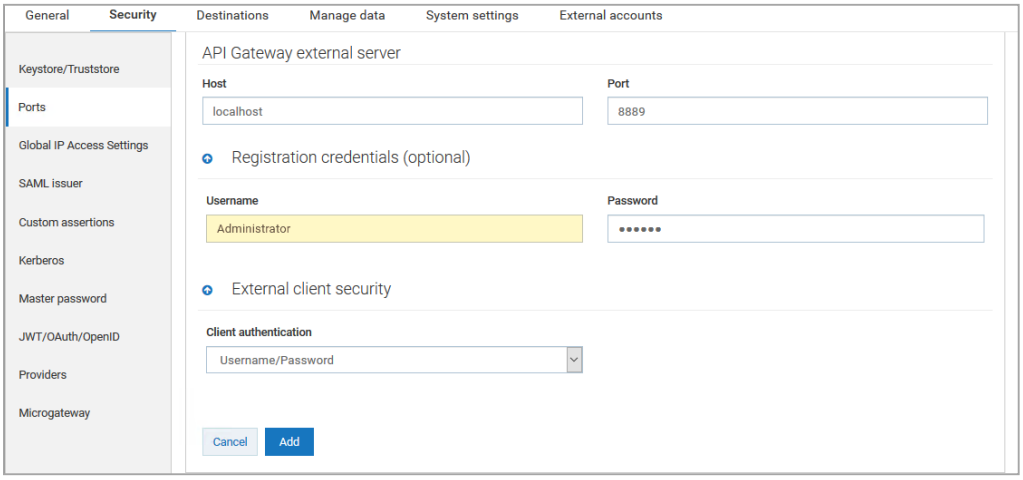
iv) Registration credentials (optional) (expand section, if collapsed):

(1) Name: **Administrator**

(2) Password: **manage**

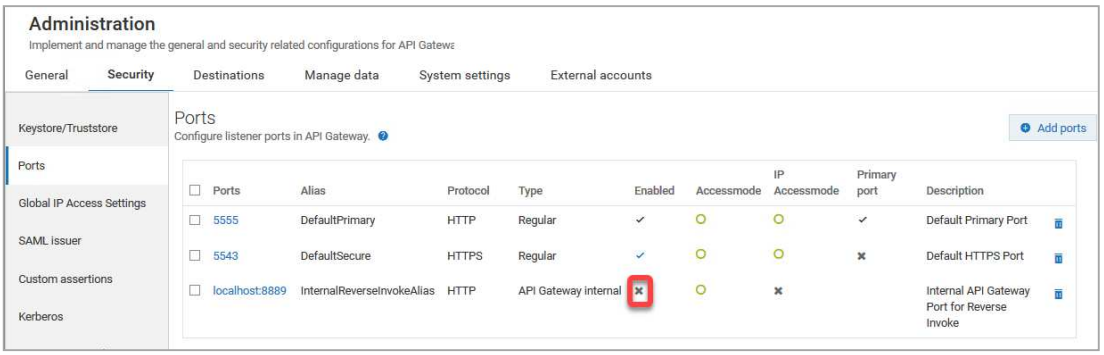
v) External Client Security (expand section, if collapsed)::

(1) Client Authentication: **Username/Password**



f) Click **Add** at the very bottom to add the port configuration.

g) Activate the internal port you have created in the previous step. To do so, click the cross in the **Enabled** column of the list of ports for port alias **InternalReverseInvokeAlias**. Confirm enabling with **Yes**.



Note: Now we can access all our APIs through the external port 8888 of the Threat Protection API Gateway.

5) Open **Postman** as a REST Client. Configure a **GET** request against the **SearchCruise** API on port **8888**.

   a) Method: **GET**

   b) URL:

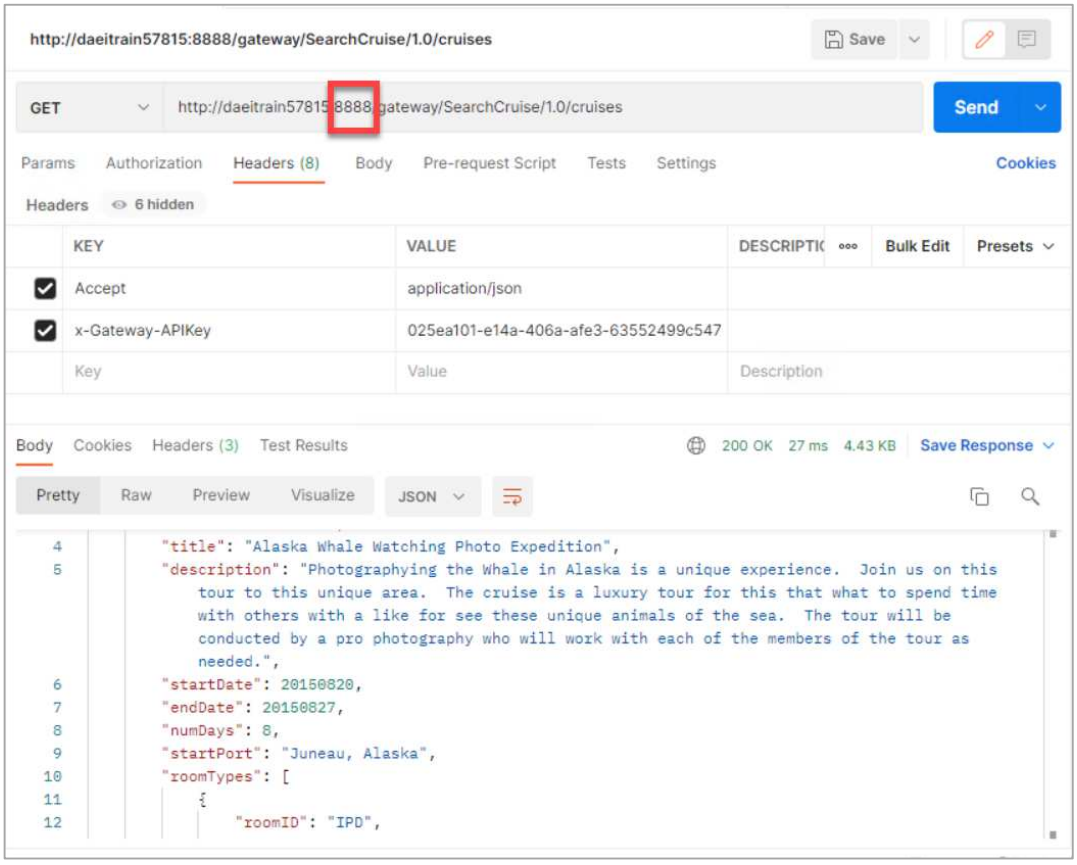
      http://<*hostname*>:8888/gateway/SearchCruise/1.0/cruises

   c) Custom Headers:

      i) **Accept**: **application/json**

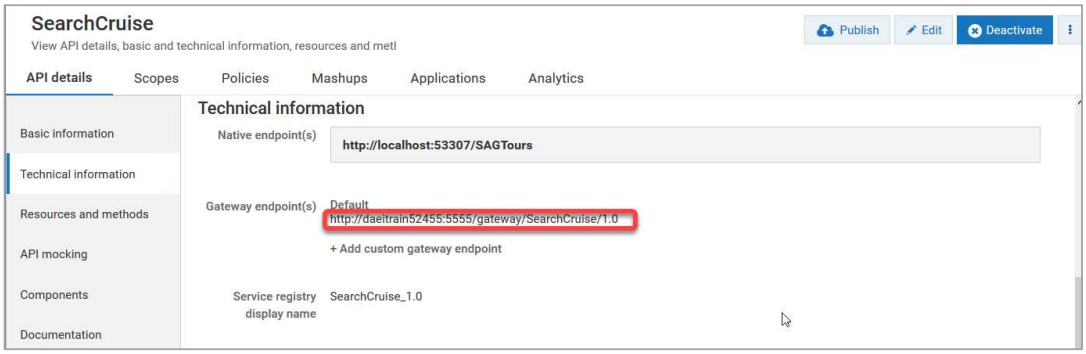
      ii) **x-Gateway-APIKey**: <***value of API key from API Gateway, see previous lab***>

6) Run the request using the **Send** button and verify the response message.



Note: The result should be the same as before when calling the API directly on the (internal) API Gateway on port 5555.

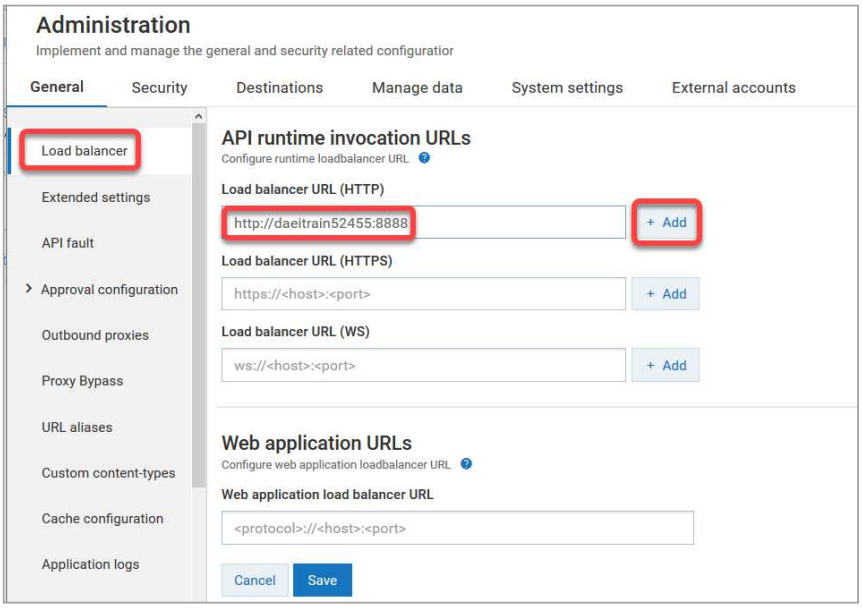
7) We can now access all our APIs through the external port of the Threat Protection API Gateway, but the Gateway endpoints of our APIs in the internal API Gateway still point at the primary port 5555 of the internal API Gateway.



To fix this, we will define a **Load balancer URL** in the internal API Gateway:

a) In a Firefox browser tab, login to the (internal) API Gateway as **Administrator | manage**. From the User Menu navigate to **Administration** > **General** > **Load balancer**.

b) On the Load balancer page provide the following property:
Load balancer URL (HTTP): **http://<*hostname*>:8888**

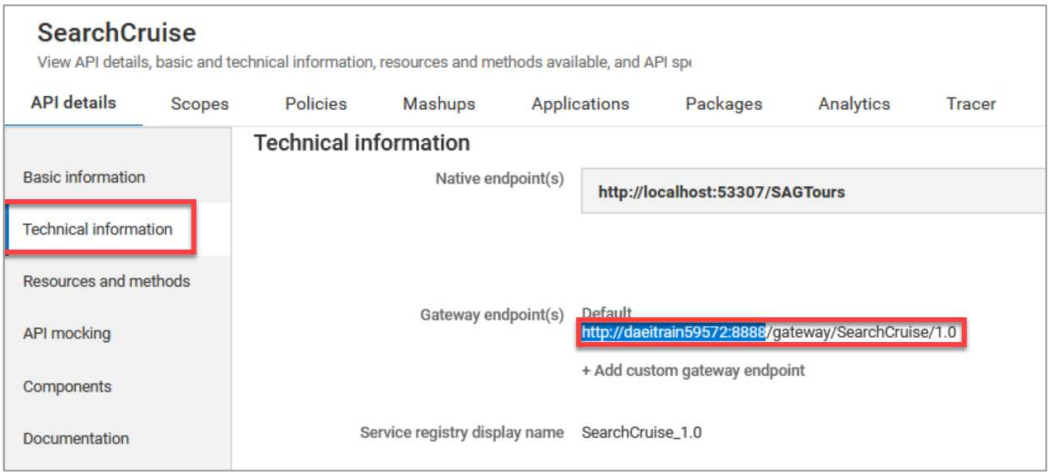


Note: The <*hostname*> depends on your environment.

Click on **+ Add** next to the Load balancer URL.

c) Click on **Save**.

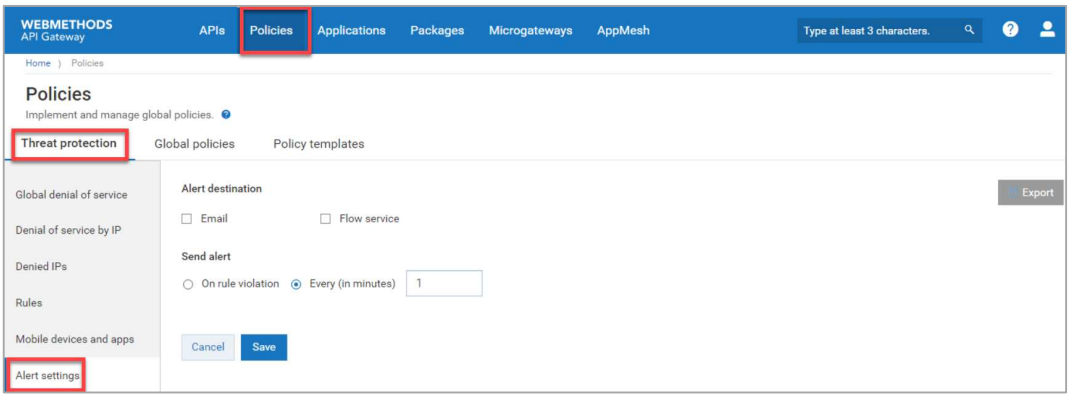
d) Navigate to the **Technical information** section of the **SearchCruise** API. You will now see the URL of the external Threat Protection API Gateway with port 8888 as Gateway endpoint.



8) Now we want to try the option that whenever a rule is violated in the Threat Protection API Gateway, an alert will be generated. We want to configure the alert setting on a global level.

a) Open a **Google Chrome** browser tab and connect to the **Threat Protection API Gateway** as **Administrator | manage**.

b) Navigate to **Policies > Threat protection > Alert settings.**



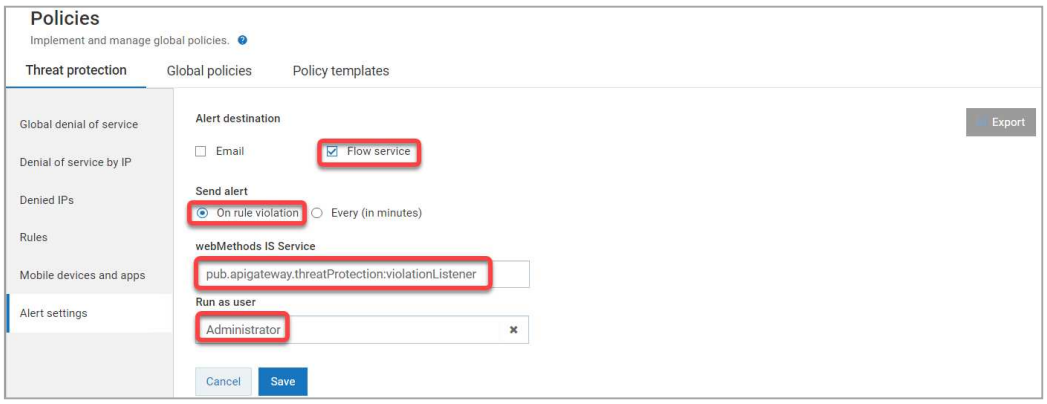
c) Provide the following properties:

i) Alert Destination: **Flow Service**

ii) Send alert: **On rule violation**

iii) webMethods Is Service: **pub.apigateway.threatProtection:violationListener**

iv) Run as user: **Administrator**



Click **Save**.

9) Now we create a global denial of service policy in Threat Protection API Gateway:

a) Navigate to **Policies** > **Threat protection** > **Global denial of service**.

b) Provide the following properties:

i) Maximum requests: **2**

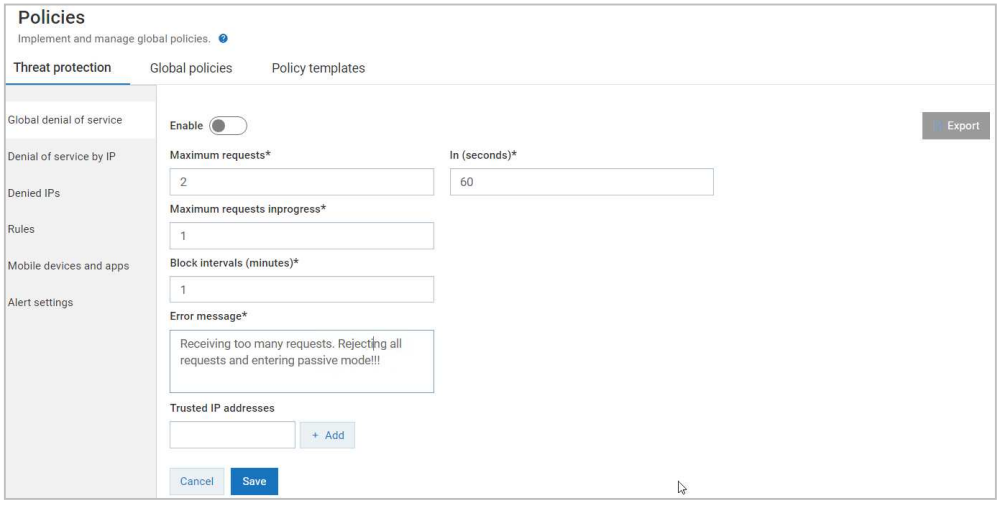
ii) In (seconds): **60**

iii) Maximum requested inprogress: **1**

iv) Block interval (minutes): **1**

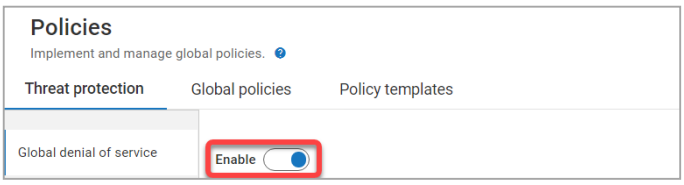
v) Error Message: **Receiving too many requests. Rejecting all requests and entering passive mode!!!**

vi) Trusted IP addresses: < ***leave empty*** >



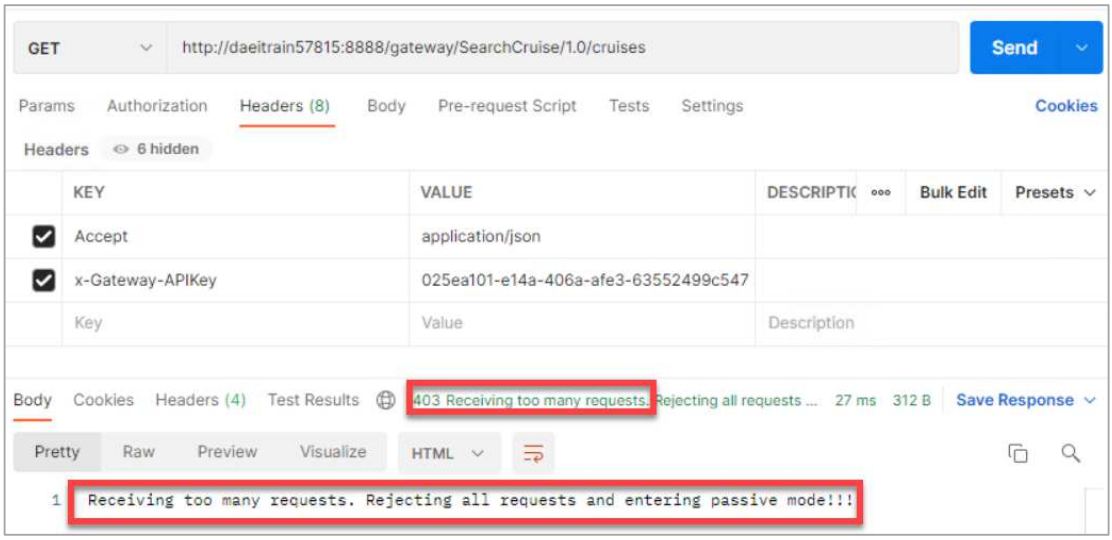
Click **Save**.

c) **Enable** the Policy.



Click **Save** again.

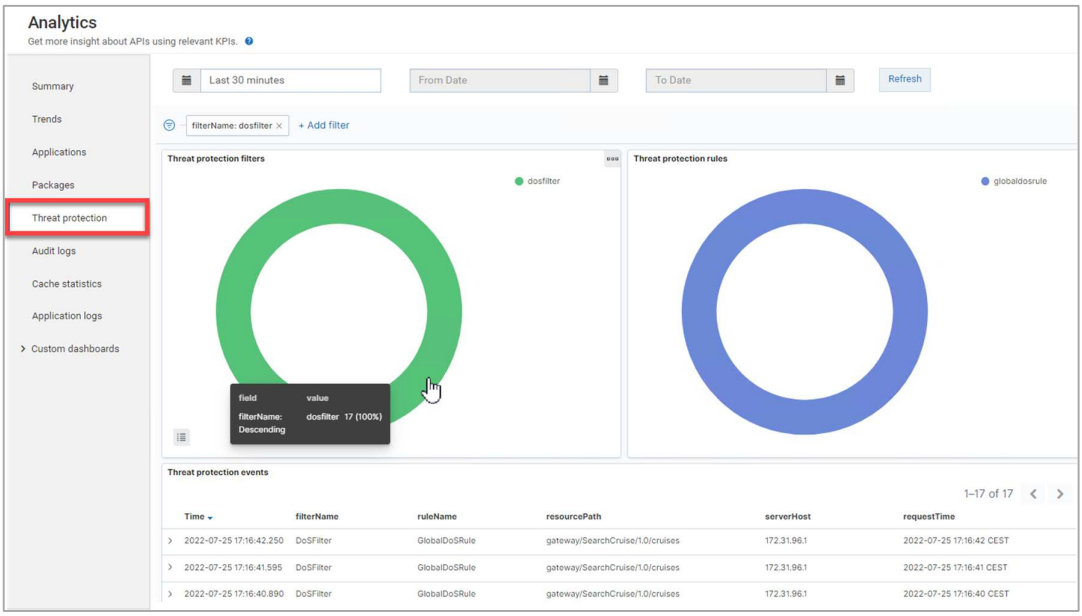
10) Switch back to the GET /cruises request in **Postman**. Rerun the same request several times. After 2 times the request should respond with the error message as defined in the Threat Protection policy.



11) We want to review the Dashboard information on Threat Protection:

a) Go back to Google Chrome where you are logged into the Threat Protection API Gateway as user **Administrator | manage**.

b) Open the User Menu and select **Analytics**. Within the left-hand menu select **Threat Protection**. Review **Threat protection filters** and **Threat protection rules**.

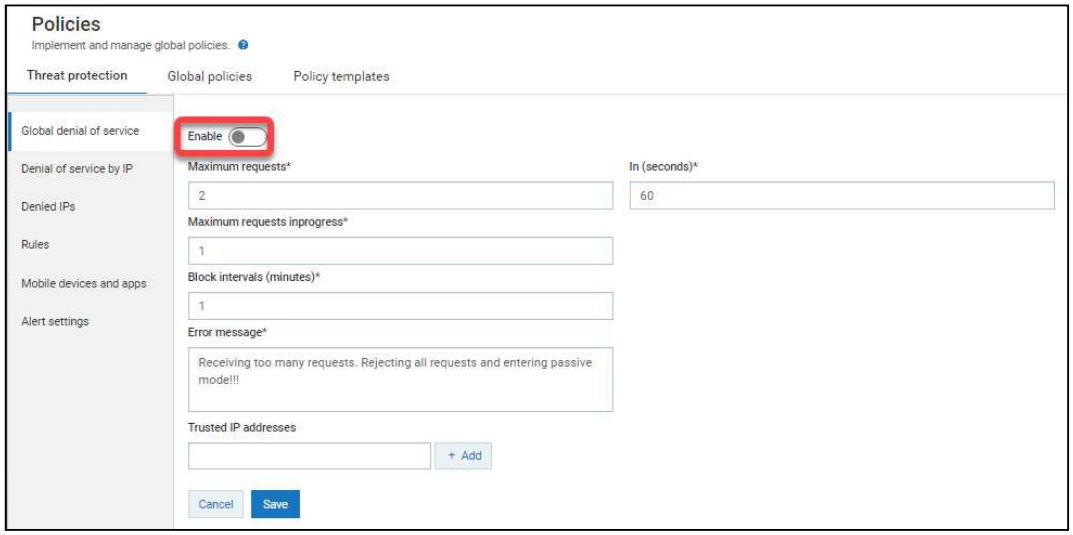


c) Click the triangle next to a threat protection event and review the details.

12) *Housekeeping*: Finally disable the Threat protection policy in the Threat Protection API Gateway:

a) Navigate to **Policies** > **Threat protection** > **Global denial of service**.

b) Disable the policy.



Click **Save**.