Webhook Notifications

Data Infrastructure Insights

NetApp
February 03, 2026

Table of Contents

Webhook Notifications

Workload Security notifications using webhooks

Webhooks allow users to send critical or warning alert notifications to various applications using a customized webhook channel.

Many commercial applications support webhooks as a standard input interface, for example: Slack, PagerDuty, Teams, and Discord. By supporting a generic, customizable webhook channel, Workload Security can support many of these delivery channels. Information about configuring the webhooks can be found on the respective application's websites. For example, Slack provides this useful guide.

You can create multiple webhook channels, each channel targeted for a different purpose, separate applications, different recipients, etc.

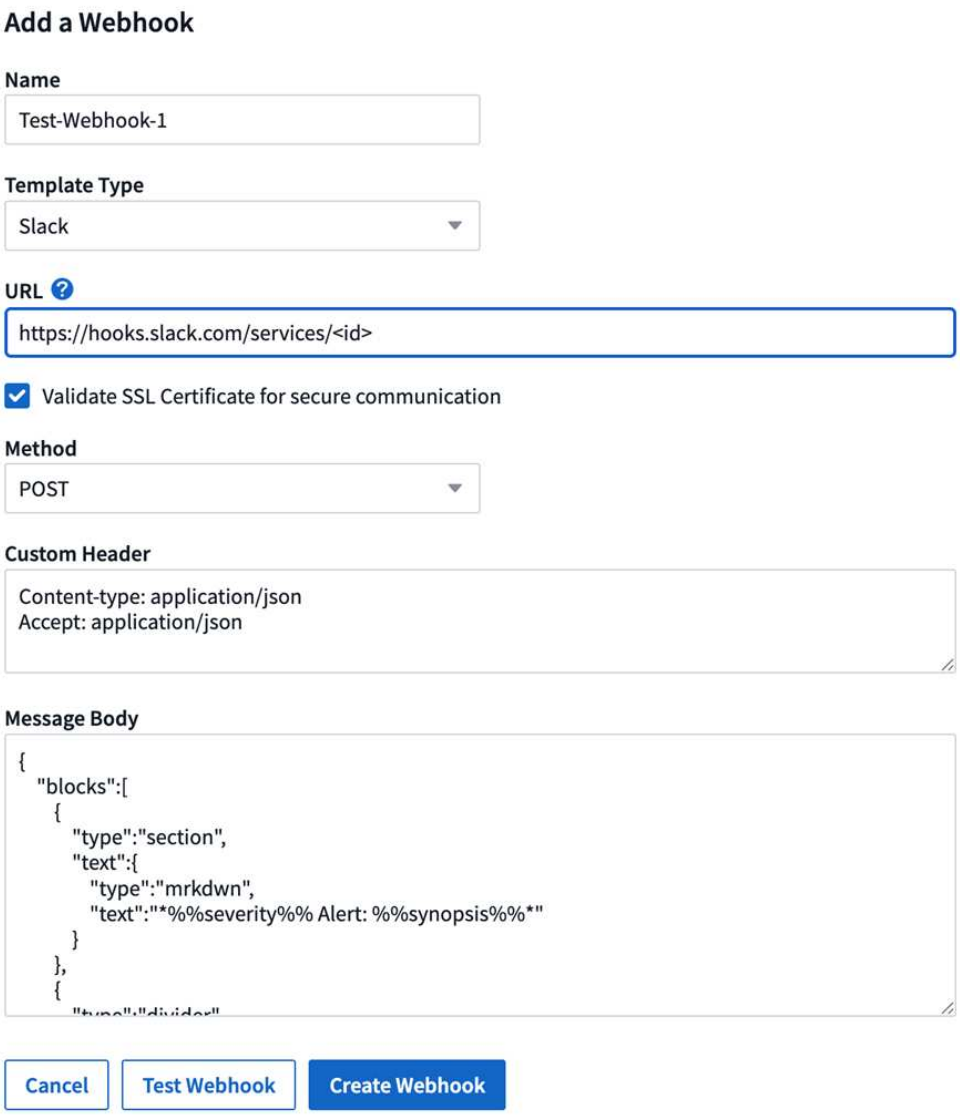
The webhook channel instance is comprised of the following elements

Name	Description
URL	Webhook target URL, including the http:// or https:// prefix along with the url params
Method	GET/POST - Default is POST
Custom Header	Specify any custom headers here
Message Body	Put the body of your message here
Default Alert Parameters	Lists the default parameters for the webhook
Custom Parameters and Secrets	Custom parameters and secrets allows you to add unique parameters and secure elements such as passwords

Creating a webhook

To create a Workload Security Webhook, go to Admin > Notifications and select "Workload Security Webhooks" tab. The following image shows a sample slack webhook creation screen.

Note: User must be a Workload Security *Admin* in order to create and manage Workload Security Webhooks.



- Enter appropriate information for each of the fields, and click "Save".
- You can also click the "Test Webhook" button to test the connection. Note that this will send the "Message Body" (without substitutions) to the defined URL according to the selected Method.
- SWS webhooks comprise a number of default parameters. Additionally, you can create your own custom parameters or secrets.

Parameters: What are they and how to use them?

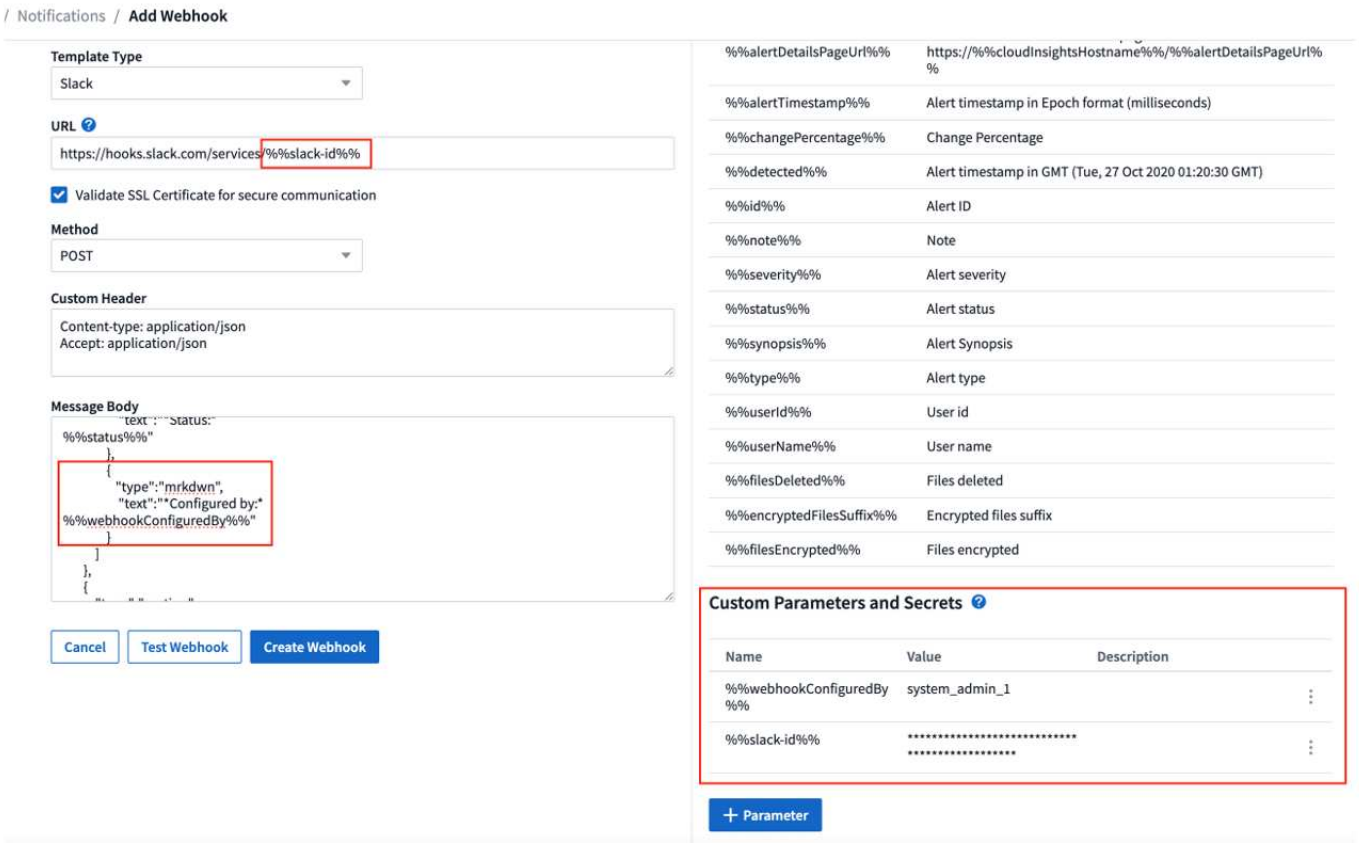
Alert Parameters are dynamic values populated per alert. For example, the *%%severity%%* parameter will be replaced with the severity type of the alert.

Note that substitutions are not performed when clicking the "Test Webhook" button; the test sends a payload that shows the parameter's placeholders (*%%<param-name>%%*) but does not replace them with data.

Custom Parameters and Secrets

In this section you can add any custom parameters and/or secrets you wish. A custom parameter or secret can be in the URL or message body. Secrets allow user to configure a secure custom parameter like password, apiKey etc.

The following sample image shows how custom parameters are used in webhook creation.



Workload Security Webhooks List Page

On the Webhooks list page, displayed are the Name, Created By, Created On, Status, Secure, and Last Reported fields.
Note: The value of 'status' column will keep changing based on the result of last webhook trigger result. The following are examples of status results.

Status	Description
OK	Successfully sent notification.
403	Forbidden.
404	URL not found.

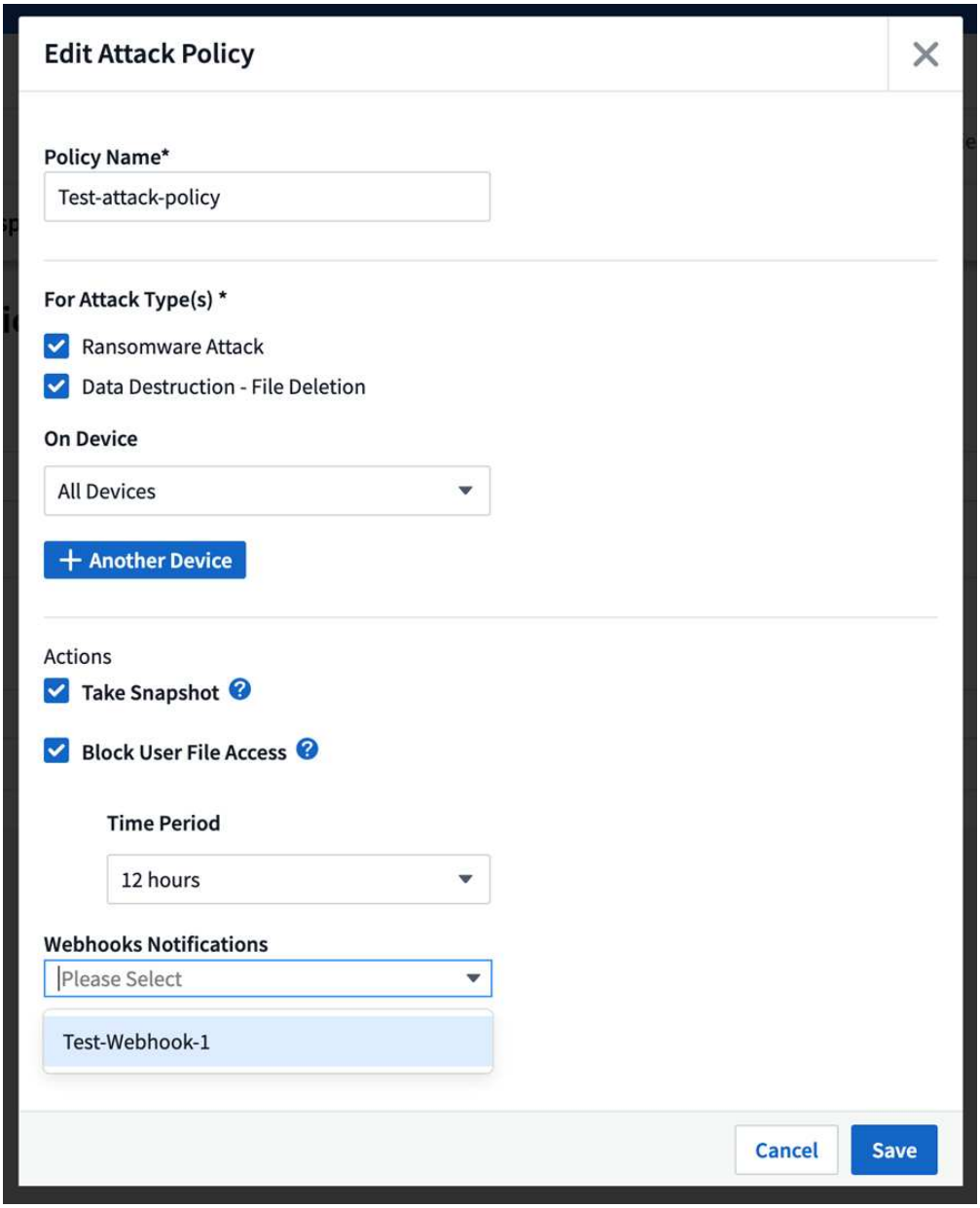
400	Bad Request. You might see this status if there is any error in the message body, for example: • Badly formatted json. • Providing invalid value for reserved keys. For example, PagerDuty accepts only critical/warning/error/info for “Severity”. Any other result may yield a 400 status. • Application specific validation errors. For example, Slack allows a maximum of 10 fields inside a section. Including more than 10 may result in a 400 status.
410	Resource is no longer available

“Last Reported” column indicates the time when the webhook was last triggered.

From the webhooks listing page users can also Edit/Duplicate/Delete webhooks.

Configure Webhook notification in alert policy

To add a webhook notification to an alert policy, go to -Workload Security > Policies- and select an existing policy or add a new policy. In the *Actions* section > *Webhook Notifications* dropdown, select the required webhooks.



Webhook notifications are tied to policies. When the attack (RW/DD/WARN) happens, the action configured (Take snapshot / user blocking) will be taken and then the associated webhook notification will be triggered.

Note: Email notifications are independent of policies, they will be triggered as usual.

- If a policy is paused, webhook notifications will not be triggered.
- Multiple webhooks can be attached to a single policy but it is recommended to attach no more than 5 webhooks to a policy.

Workload Security Webhook Examples

Webhooks for Slack

Webhooks for PagerDuty
Webhooks for Teams
Webhooks for Discord

Workload Security Webhook Example for Discord

Webhooks allow users to send alert notifications to various applications using a customized webhook channel. This page provides an example for setting up webhooks for Discord.

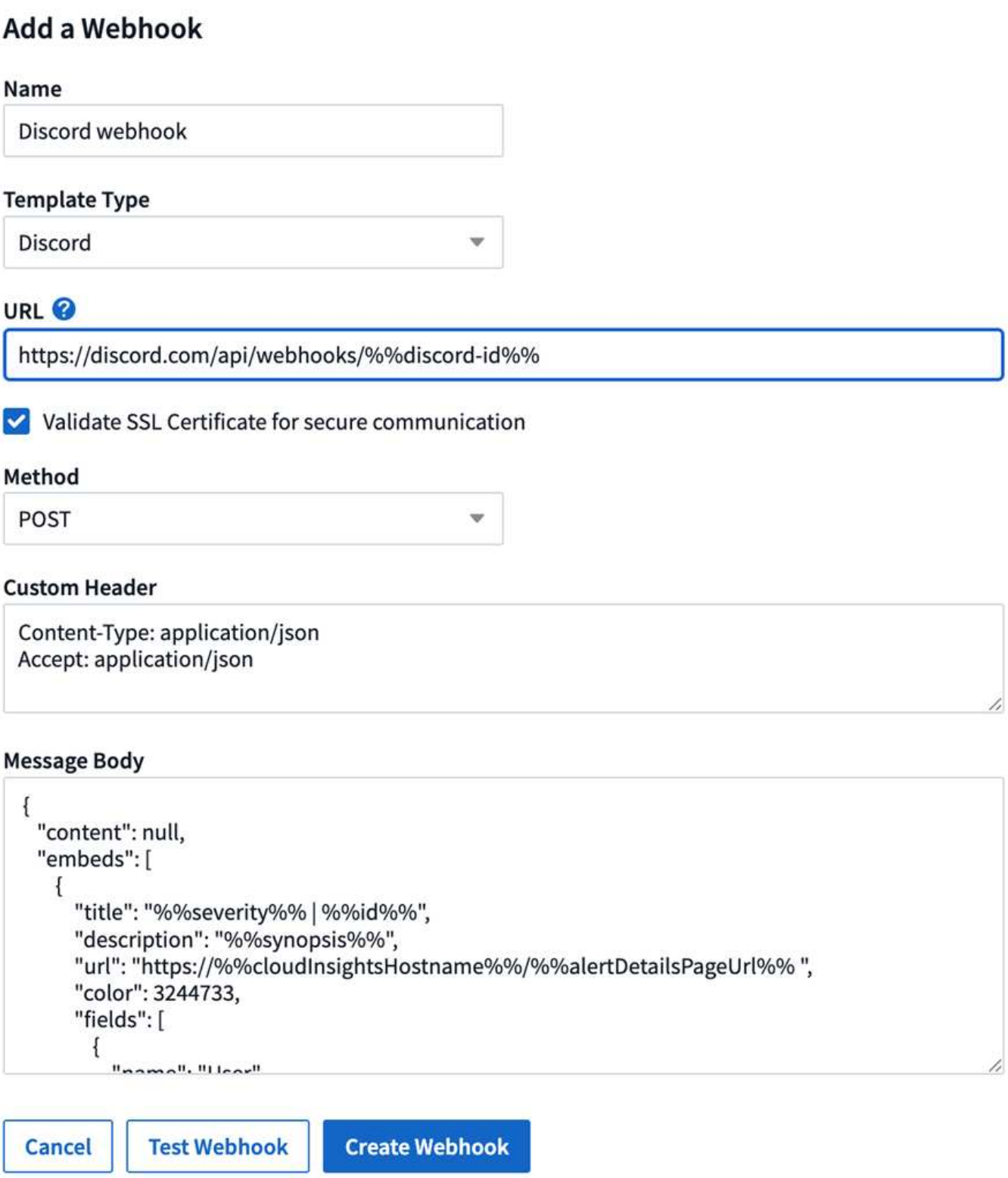
This page refers to third-party instructions, which care subject to change. Refer to the Discord documentation for the most up-to-date information.

Discord Setup:

- In Discord, select the Server, under Text Channels, select Edit Channel (gear icon)
- Select **Integrations > View Webhooks** and click **New Webhook**
- Copy the Webhook URL. You will need to paste this into the Workload Security webhook configuration.

Create Workload Security Webhook:

1. Navigate to Admin > Notifications and select the *Workload Security Webhooks* tab. Click '+ Webhook' to create a new webhook.
2. Give the webhook a meaningful Name.
3. In the *Template Type* drop-down, select **Discord**.
4. Paste the Discord URL from above into the *URL* field.



In order to test the webhook, temporarily replace the URL value in the message body with any valid URL (such as *https://netapp.com*) then click the *Test Webhook* button. Discord requires that a valid URL be supplied in order for Test Webhook functionality to work.

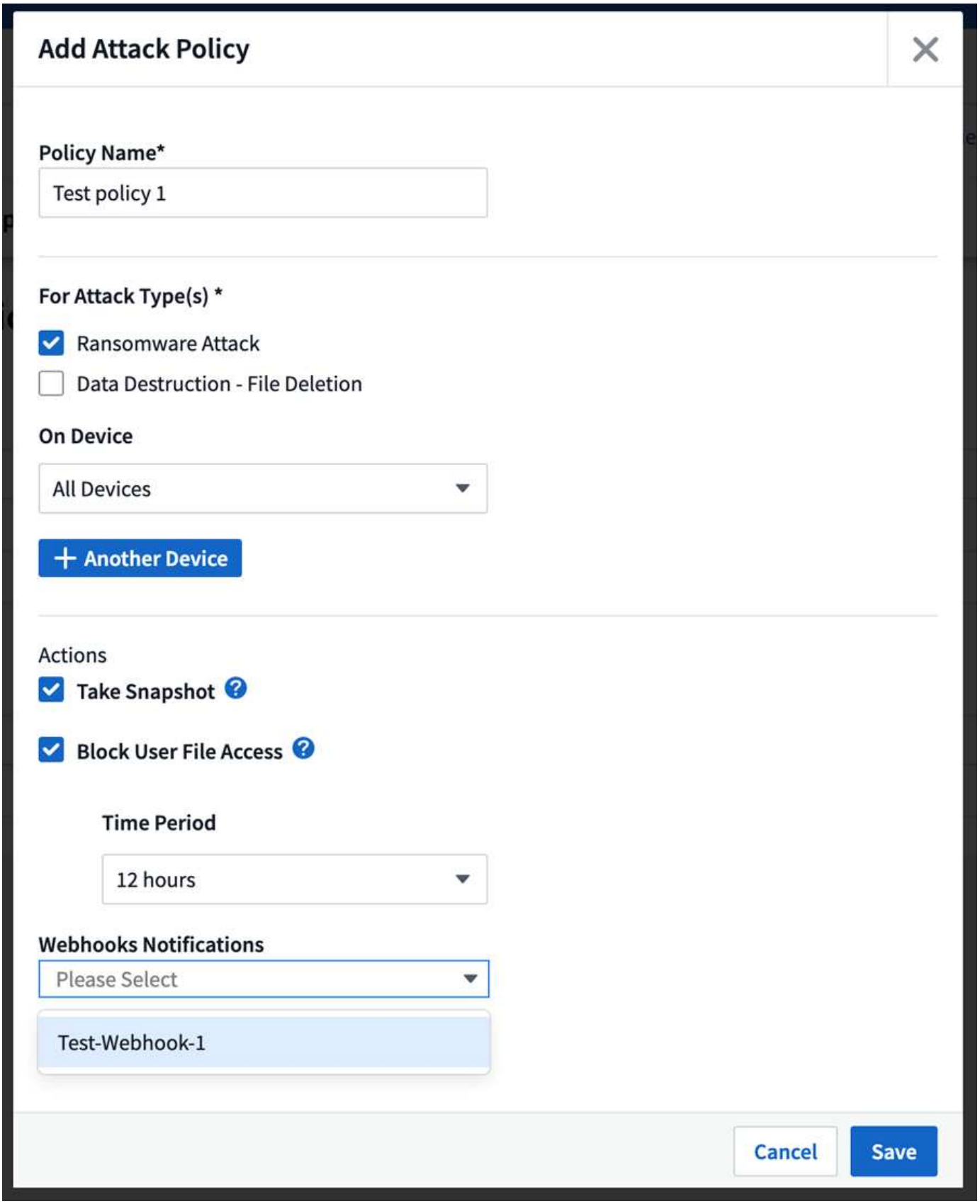
Be sure to set the message body back once the test completes.

Notifications via Webhook

To notify on events via webhook, navigate to *Workload Security > Policies*. Click on *+Attack Policy* or *+Warning Policy*.

- Enter a meaningful policy name.
- Select the required Attack Type(s), Devices to which policy should be attached, and required Actions.
- Under the *Webhooks Notifications* dropdown, select the required Discord webhooks and save.

Note: Webhooks can also be attached to existing policies by editing them.



Workload Security Webhook Example for PagerDuty

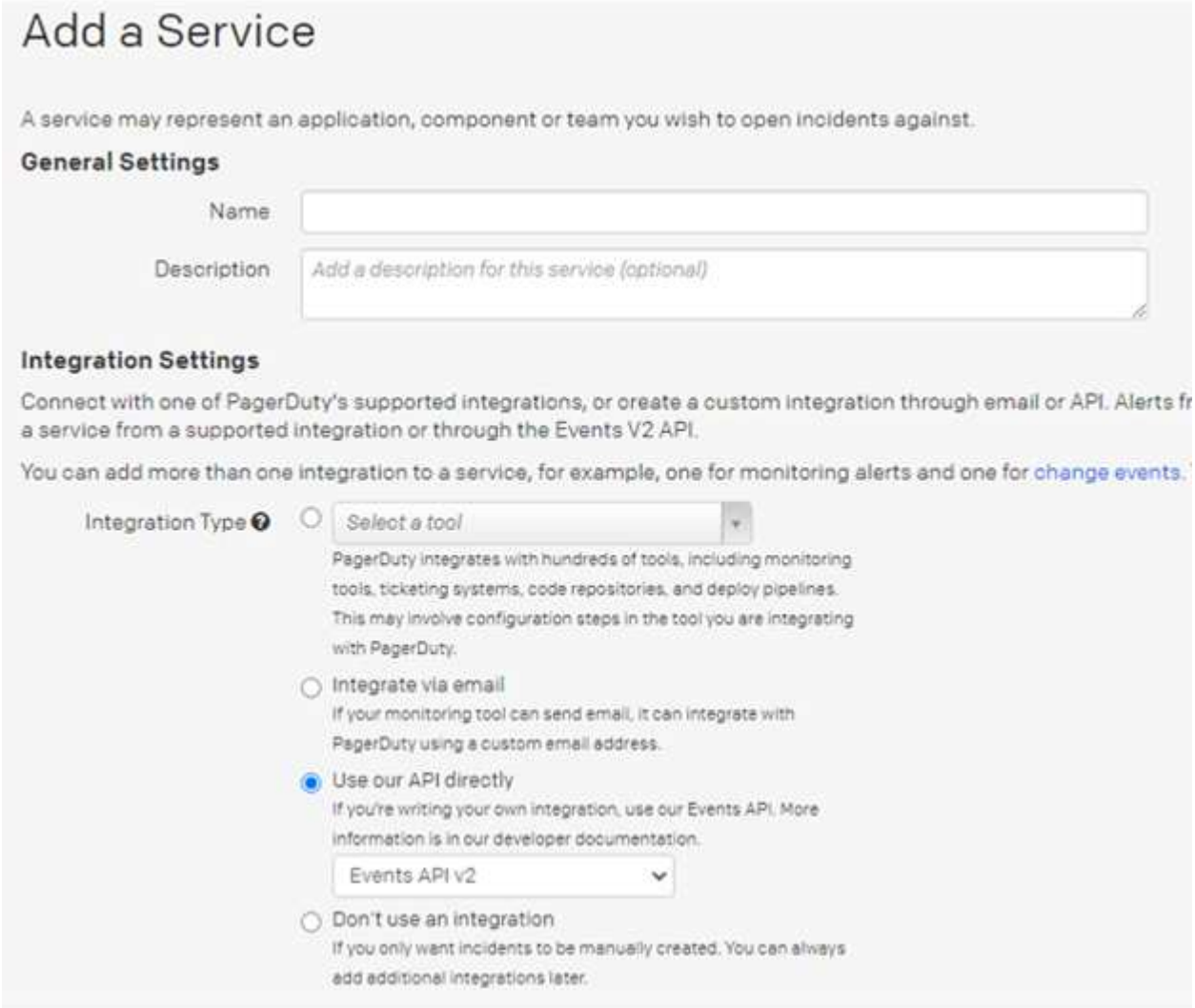
Webhooks allow users to send alert notifications to various applications using a

customized webhook channel. This page provides an example for setting up webhooks for PagerDuty.

> This page refers to third-party instructions, which are subject to change. Refer to the PagerDuty documentation for the most up-to-date information.

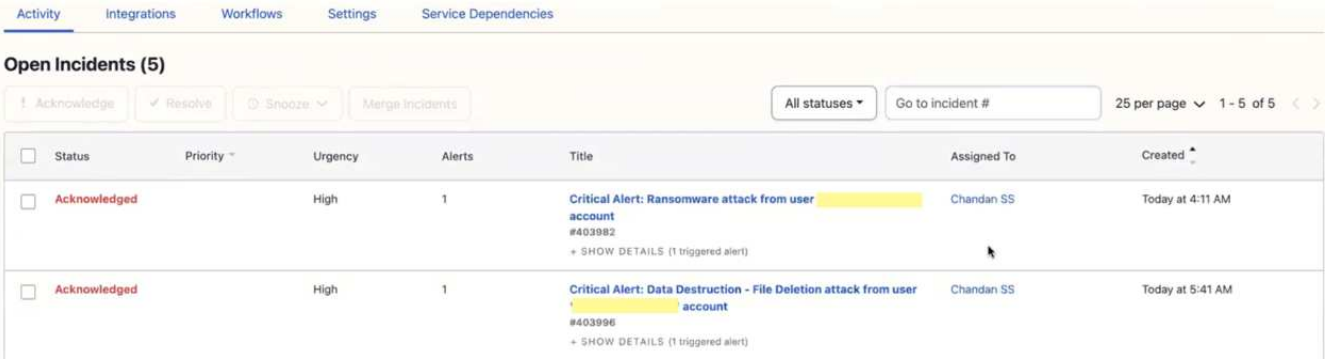
PagerDuty Setup:

1. In PagerDuty, navigate to **Services > Service Directory** and click on the **+New Service** button.
2. Enter a *Name* and select *Use our API directly*. Select *Add Service*.



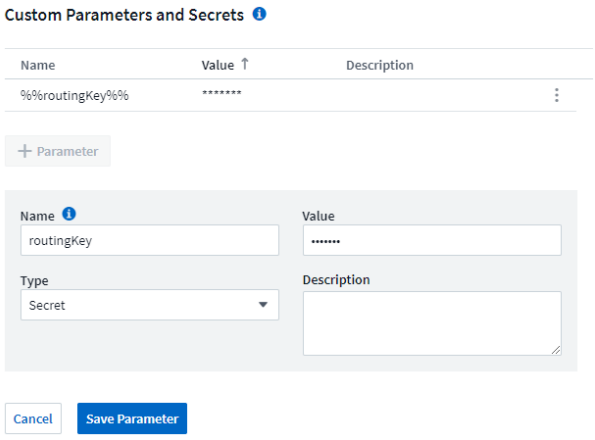
3. Select the *Integrations* tab to see the **Integration Key**. You will need this key when you create the Workload Security webhook below.

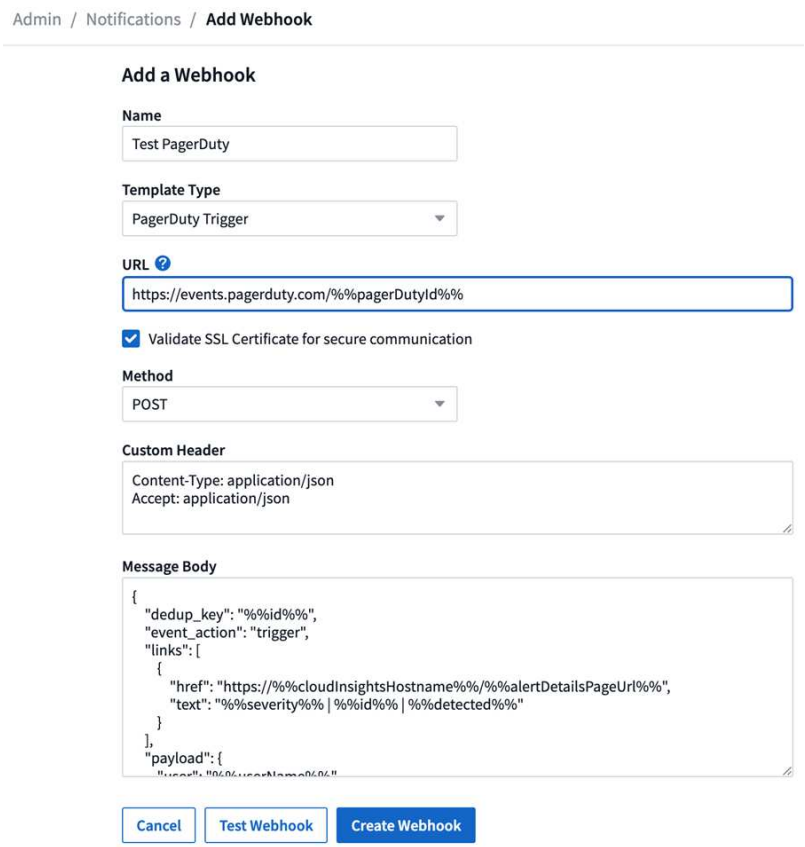
1. Go to **Incidents** or **Services** to view Alerts.



Create Workload Security PagerDuty Webhook:

- Navigate to Admin > Notifications and select the *Workload Security Webhooks* tab. Select '+ Webhook' to create a new webhook.
- Give the webhook a meaningful name.
- In the *Template Type* dropdown, select *PagerDuty Trigger*.
- Create a custom parameter secret named *routingKey* and set the value to the PagerDuty *Integration Key* created above.

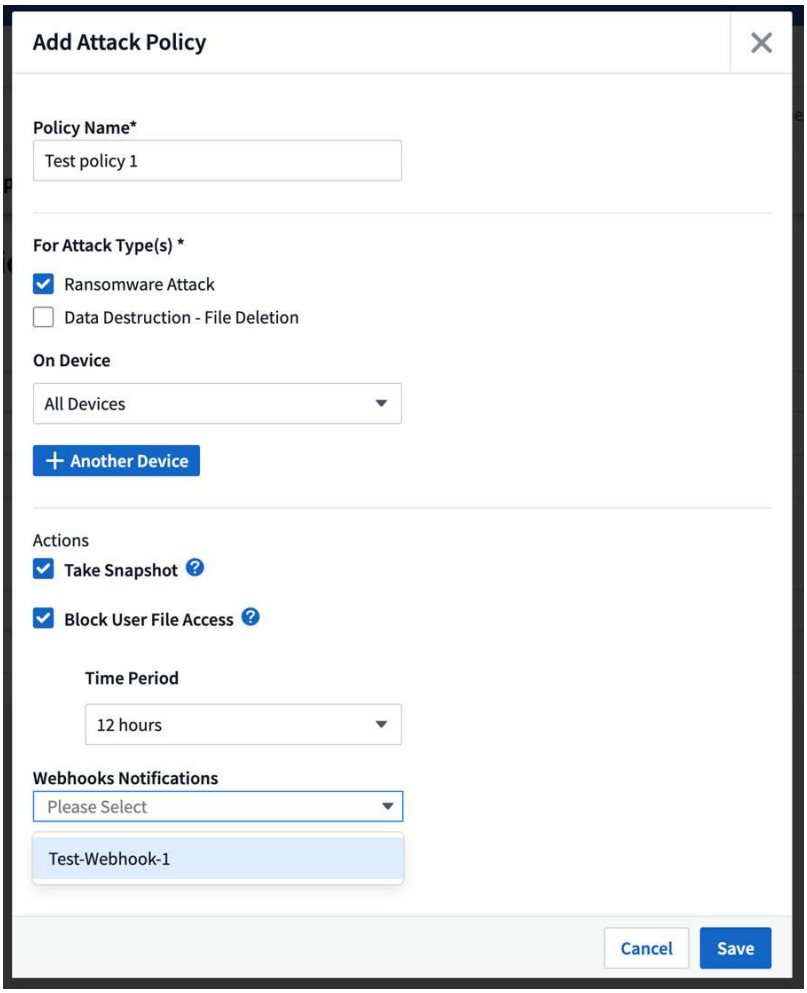




Notifications via Webhook

- To notify on events via webhook, navigate to *Workload Security* > *Policies*. Select *+Attack Policy* or *+Warning Policy*.
- Enter a meaningful policy name.
- Select required Attack Type(s), Devices to which the policy should be attached, and the required Actions.
- Under *Webhooks Notifications* dropdown, select the required PagerDuty webhooks. Save the policy.

Note: Webhooks can also be attached to existing policies by editing them.



Workload Security Webhook Example for Slack

Webhooks allow users to send alert notifications to various applications using a customized webhook channel. This page provides an example for setting up webhooks for Slack.

This page refers to third-party instructions, which are subject to change. Refer to the Slack documentation for the most up-to-date information.

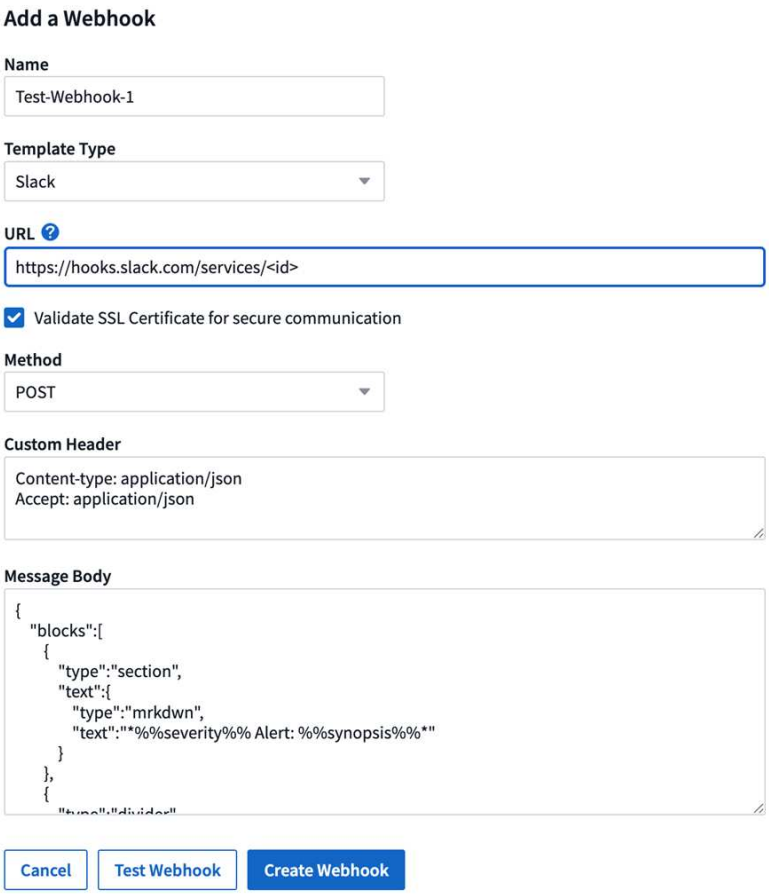
Slack Example

- Go to https://api.slack.com/apps and Create a new App. Give it a meaningful name and select a Workspace.

- Go to Incoming Webhooks, click on *Activate Incoming Webhooks*, select *Add New Webhook*, and select the Channel on which to Post.
- Copy the Webhook URL. This URL will be given when creating a Workload Security webhook.

Create Workload Security Slack Webhook

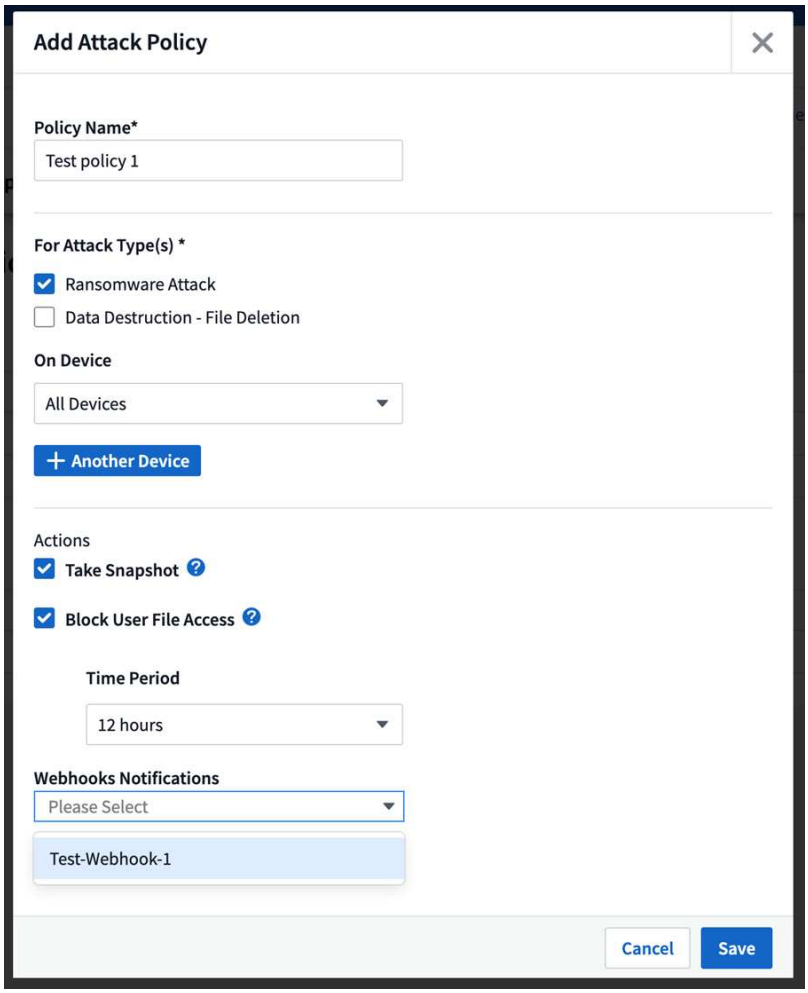
1. Navigate to Admin > Notifications and select the *Workload Security Webhooks* tab. Select + *Webhook* to create a new webhook.
2. Give the webhook a meaningful name.
3. In the *Template Type* dropdown, select *Slack*.
4. Paste the URL copied from above.



Notifications via webhook

- To notify on events via webhook, navigate to *Workload Security > Policies*. Click on *+Attack Policy* or *+Warning Policy*.
- Enter a meaningful policy name.
- Select required Attack Type(s), Devices to which the policy should be attached, and required Actions.
- Under the *Webhooks Notifications* dropdown, select the required webhooks. Save the policy.

Note: Webhooks can also be attached to existing policies by editing them.



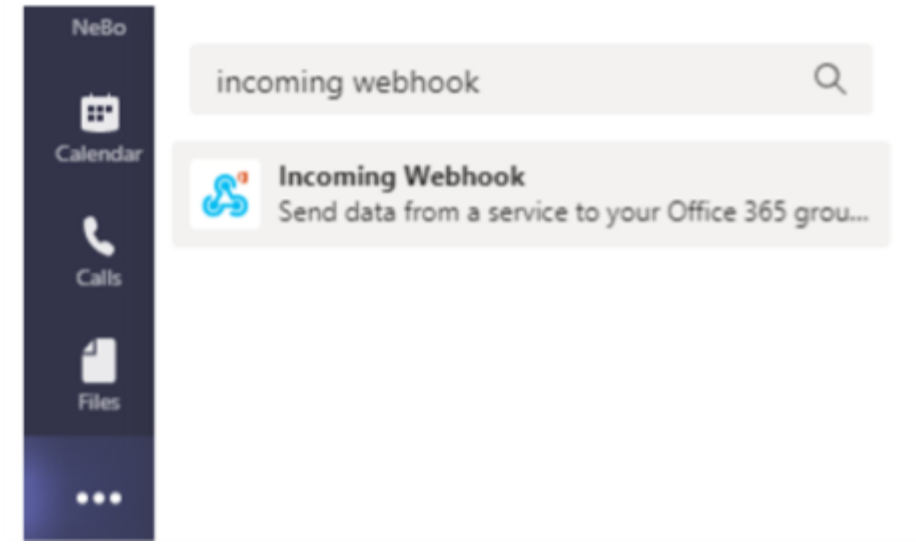
Workload Security Webhook Example for Microsoft Teams

Webhooks allow users to send alert notifications to various applications using a customized webhook channel. This page provides an example for setting up webhooks for Teams.

This page refers to third-party instructions, which are subject to change. Refer to the Teams documentation for the most up-to-date information.

Teams Setup:

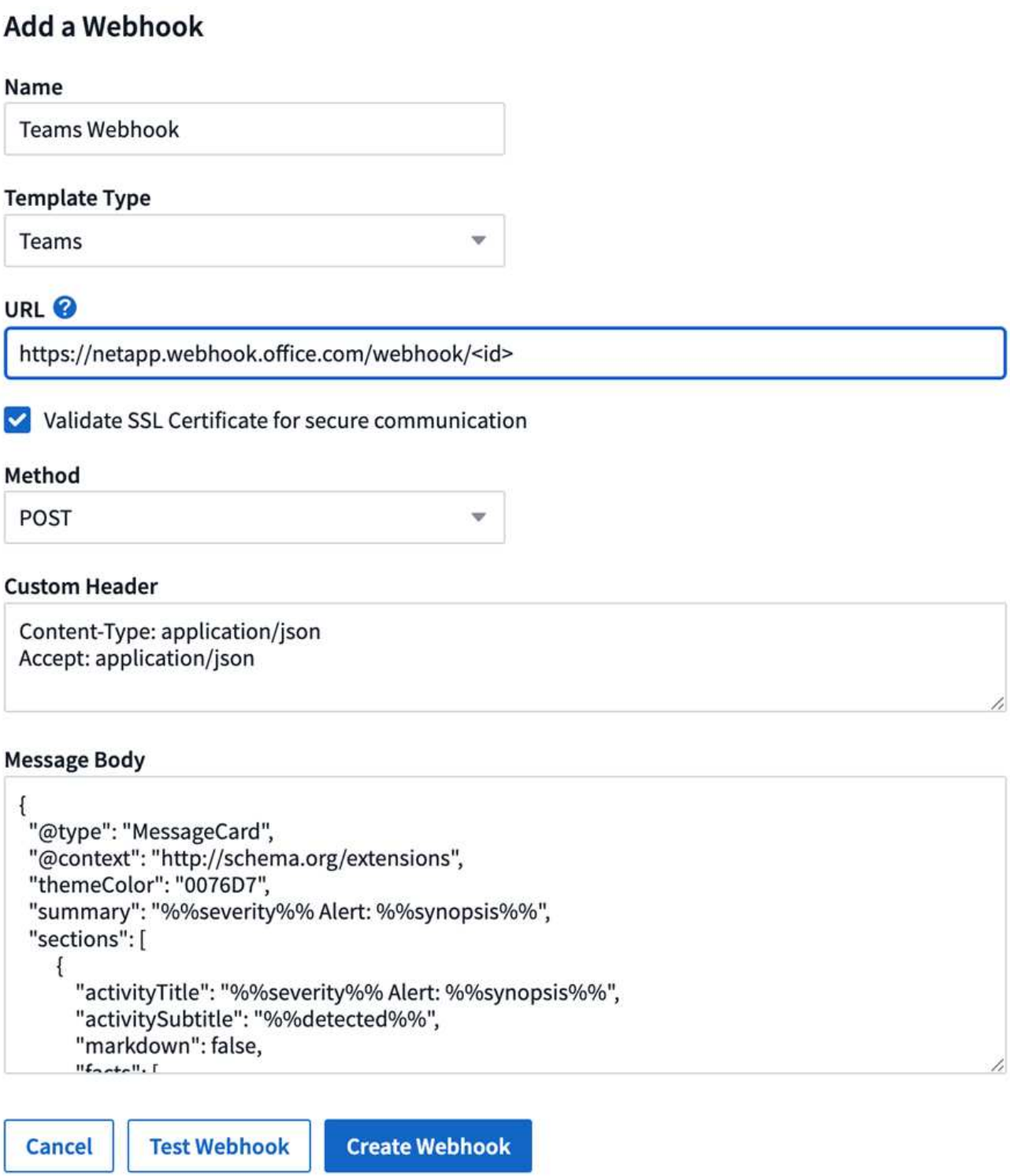
1. In Teams, select the kebab, and search for Incoming Webhook.



2. Select **Add to a Team > Select a Team > Setup a Connector**.
3. Copy the Webhook URL. You will need to paste this into the Workload Security webhook configuration.

Create Workload Security Teams Webhook:

1. Navigate to Admin > Notifications and select the *"Workload Security Webhooks* tab. Select + *Webhook* to create a new webhook.
2. Give the webhook a meaningful Name.
3. In the *Template Type* drop-down, select **Teams**.



4. Paste the URL from above into the *URL* field.

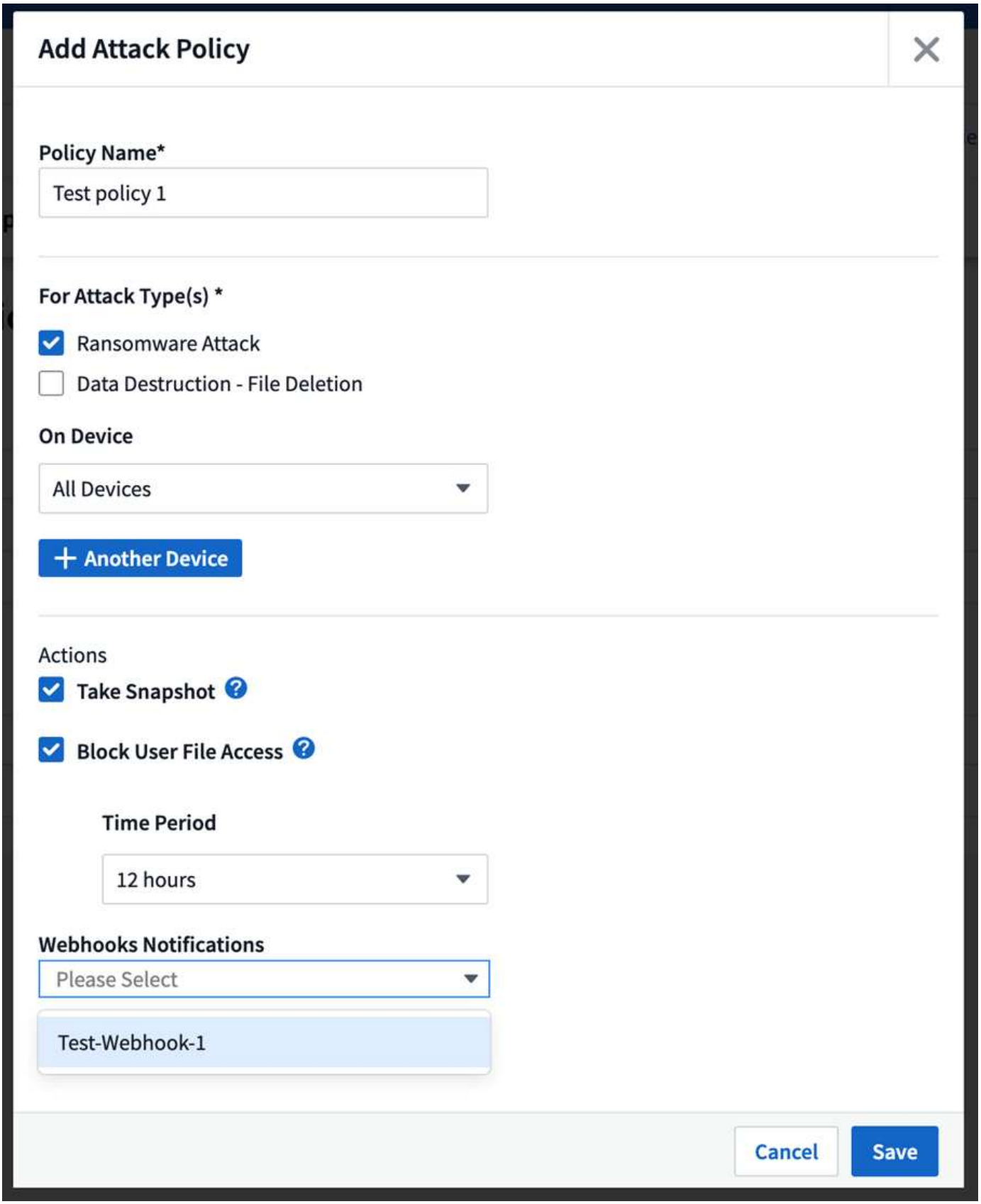
Notifications via Webhook

To notify on events via webhook, navigate to *Workload Security > Policies*. Select *+Attack Policy* or *+Warning Policy*.

- Enter a meaningful policy name.
- Select required Attack Type(s), Devices to which policy should be attached, and required Actions.

- Under the *Webhooks Notifications* dropdown, select the required Teams webhooks. Save the policy.

Note: Webhooks can also be attached to existing policies by editing them.



Copyright information

Trademark information