OneView: Webhooks

1 Webhooks 2

1.1 Alerts from other monitoring systems 3

1.2 DOWN/UP events 4

1.3 Logging other operation events 4

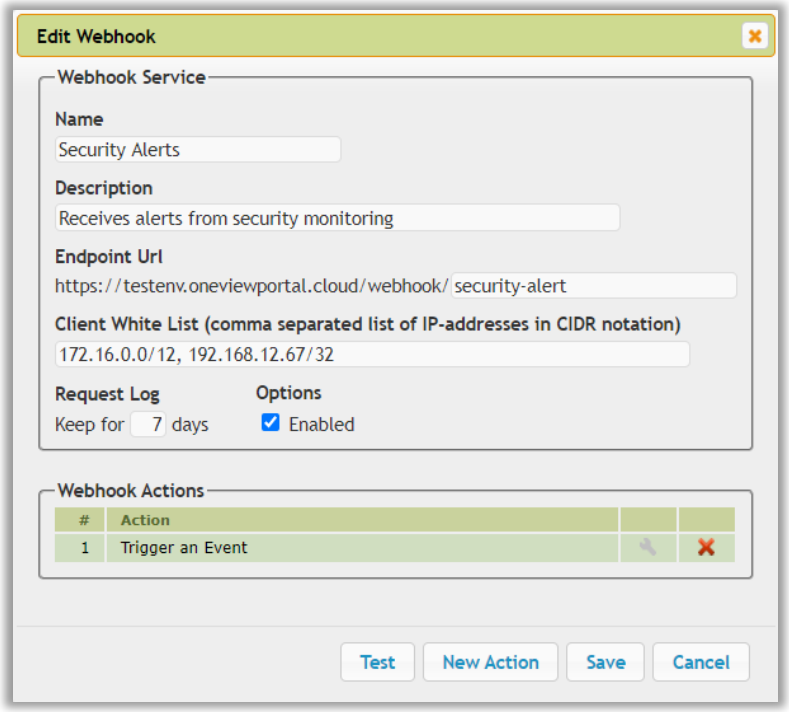
1 Webhooks

Webhooks are a simple way for other applications or monitoring tools to send events into OneView.

When creating a custom webhook in OneView, you assign it a unique endpoint URL to which an external application can send an HTTP request.

Currently a webhook does not require the client to authenticate itself, but client access can be limited by defining a white-list of explicit allowed client IP-address ranges, like your internal network range or the IP-address of the sending application.

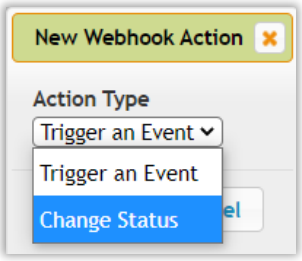
The webhook can be disabled and clients will be denied access.



A webhook request log is kept for several days for troubleshooting and documentation.

When the webhook receives the request, it can perform several actions like modifying the status of some dashboard entities in OneView.

Currently the options are to either trigger an event transaction or change status for a status transaction.

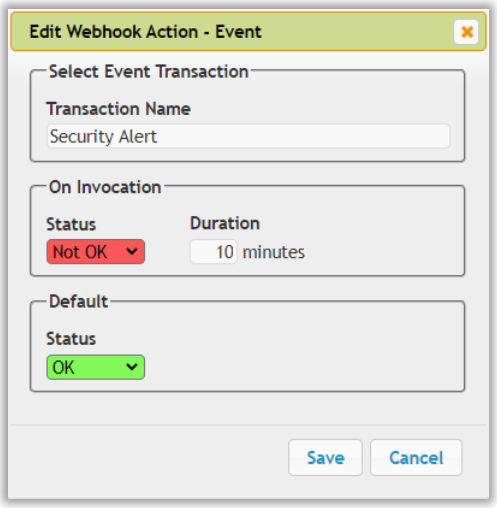


1.1 Alerts from other monitoring systems

Tools that create security/intrusion alerts based on traffic analysis or platform specific tools that create alerts about resource depletion or malfunctions may be able to trigger a webhook in OneView.

Webhooks enable you to change the status on one of your dashboard systems when an alert is raised.

The following webhook action will create an event called “Security Alert” in OneView and set event status to RED for 10 minutes and then back to GREEN if no more events are received.

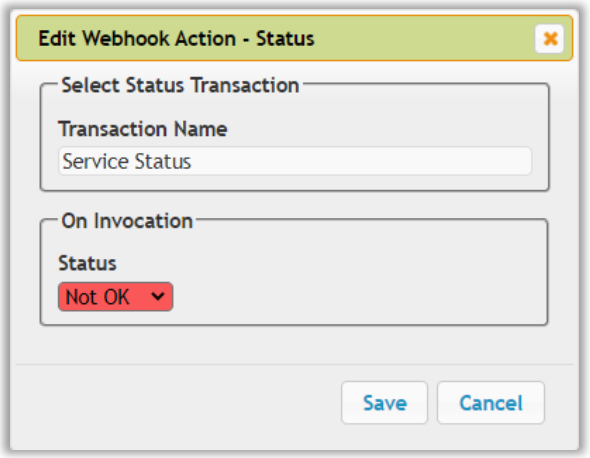


1.2 DOWN/UP events

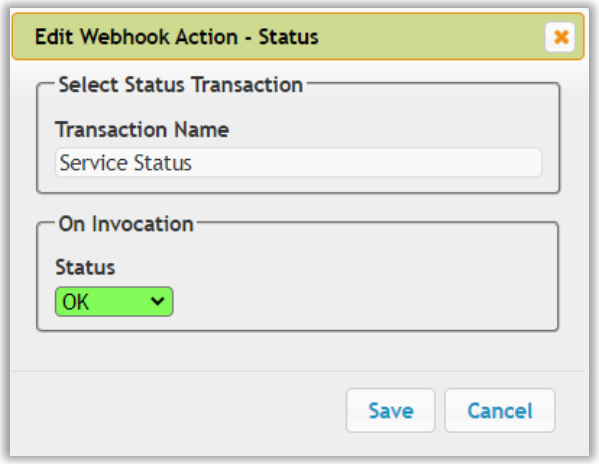
Some systems create explicit DOWN and UP events.

This can be handled by creating two webhooks (one for DOWN and one for UP) each with an action setting status for the same transaction “Service Status”.

DOWN action:



UP action:



1.3 Logging other operation events

Often management systems in operations may create events whenever they perform a task, like restarting a server, building software, or running a batch job.

It could also be events based on user activity like a failed login or checkout.

Webhooks for these kinds of events may not introduce any status changes. They just make sure that the events are logged for later analysis.