

Configure AppDynamics Alerts to Webex for Live Incident Response

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Background Information](#)

[Configure](#)

[Why Integrate AppDynamics with Webex?](#)

[Configurations](#)

[Step 1: Create a Webex Webhook](#)

[Step 2: Set Up the HTTP Request Template in AppDynamics](#)

[Step 3: Link the HTTP Request Template to a Policy](#)

[Verify](#)

[Troubleshoot](#)

[Conclusion](#)

[Need Further Assistance?](#)

[Related Information](#)

Introduction

This document describes how to configure AppDynamics Controller to send alert notifications directly to Webex spaces.

Prerequisites

- Administrative access to AppDynamics Controller (SaaS or on-premises)
- Webex account with permission to create Webhook
- Familiarity with AppDynamics alerting and HTTP request templates
- Basic understanding of REST APIs

Requirements

- AppDynamics Controller version 21.x or later
- Webex App account and access to the [Webex AppHub](#)
- Network connectivity between AppDynamics Controller and Webex cloud endpoints
- Configured notification policies in AppDynamics

Components Used

- AppDynamics Controller
- Webex App
- HTTP Request Template

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

In the fast-paced environments of today, minimizing Mean Time to Detect (MTTD) and Mean Time to Resolve (MTTR) is critical. This guide walks you through the steps to integrate the AppDynamics Controller with Webex in order to automatically push alert notifications to your team collaborative spaces — enabling real-time visibility and accelerating incident response.

Configure

Why Integrate AppDynamics with Webex?

- **Instant Collaboration:** Receive AppDynamics alerts in Webex spaces for immediate team awareness and action.
- **Enhanced Incident Response:** Enable IT and DevOps teams to triage and resolve issues collaboratively in real time.
- **Unified Workflows:** Manage incidents and discussions in a single platform, reducing context switching.

Configurations

Step 1: Create a Webex Webhook

- Create Webex Space:

Create a space

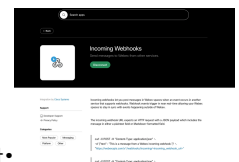
Start a group conversation with others.

✕

☐ Make this space public
 Anyone in your organization can find and join a public space.

Close
Create

- Login to [Webex AppHub](https://webex.com) and Navigate to **Incoming Webhooks** and select **Connect**:



- Provide Webhook Name and select Webex Space to send alerts:
- Click **Add** and copy webhook URL to your notepad:

Integration by Cisco Systems

Support

Developer Support

Privacy Policy

Categories

Most Popular

Platform

Other

Incoming webhooks let you post messages in Webex spaces when an event occurs in another service that supports webhooks. Webhook events trigger in near real-time allowing your Webex spaces to stay in sync with events happening outside of Webex.

The incoming webhook URL expects an HTTP request with a JSON payload which includes the message in either a plaintext field or Markdown-formatted field.

```
curl -X POST -H "Content-Type: application/json" \
  -d '{"text": "This is a message from a Webex incoming webhook."}' \
  "https://webexapi.com/v1/webhooks/incoming/incoming_webhook_url"
```

```
curl -X POST -H "Content-Type: application/json" \
  -d '{"markdown": "This is a formatted message from a Webex incoming webhook."}' \
  "https://webexapi.com/v1/webhooks/incoming/incoming_webhook_url"
```

Webhook name:

Select a space:

Add

Incoming Webhooks

AppD-Alerts

Room: AppDynamics Alerts



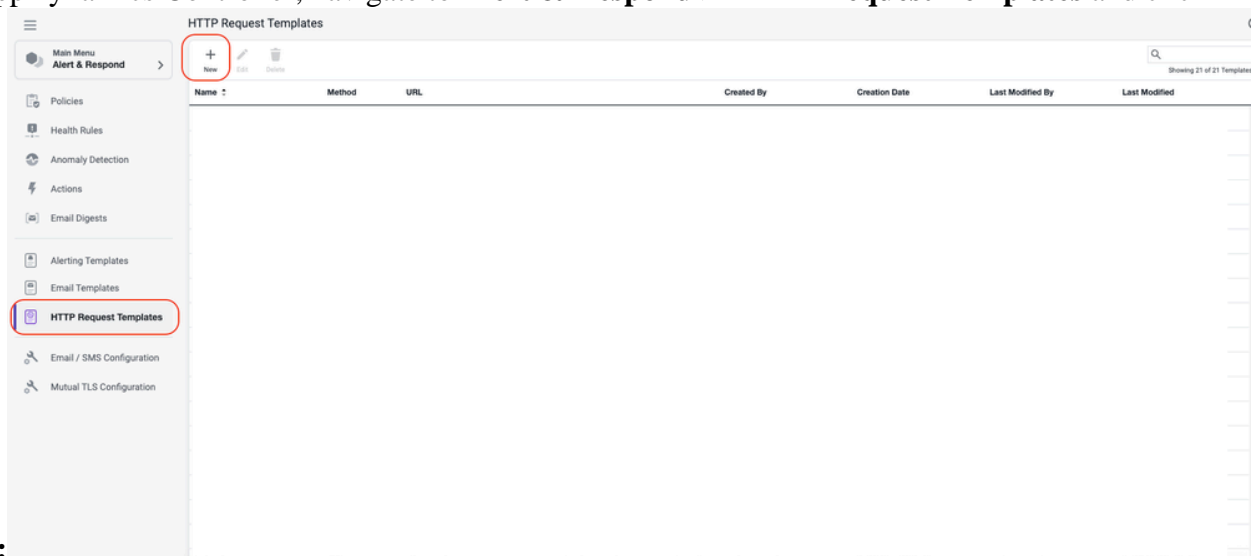
Webhook URL

`https://webexapis.com/v1/webhooks/incoming/Y2lzY'`



Step 2: Set Up the HTTP Request Template in AppDynamics

1. In AppDynamics Controller, navigate to **Alert & Respond > HTTP Request Templates** and click



New:

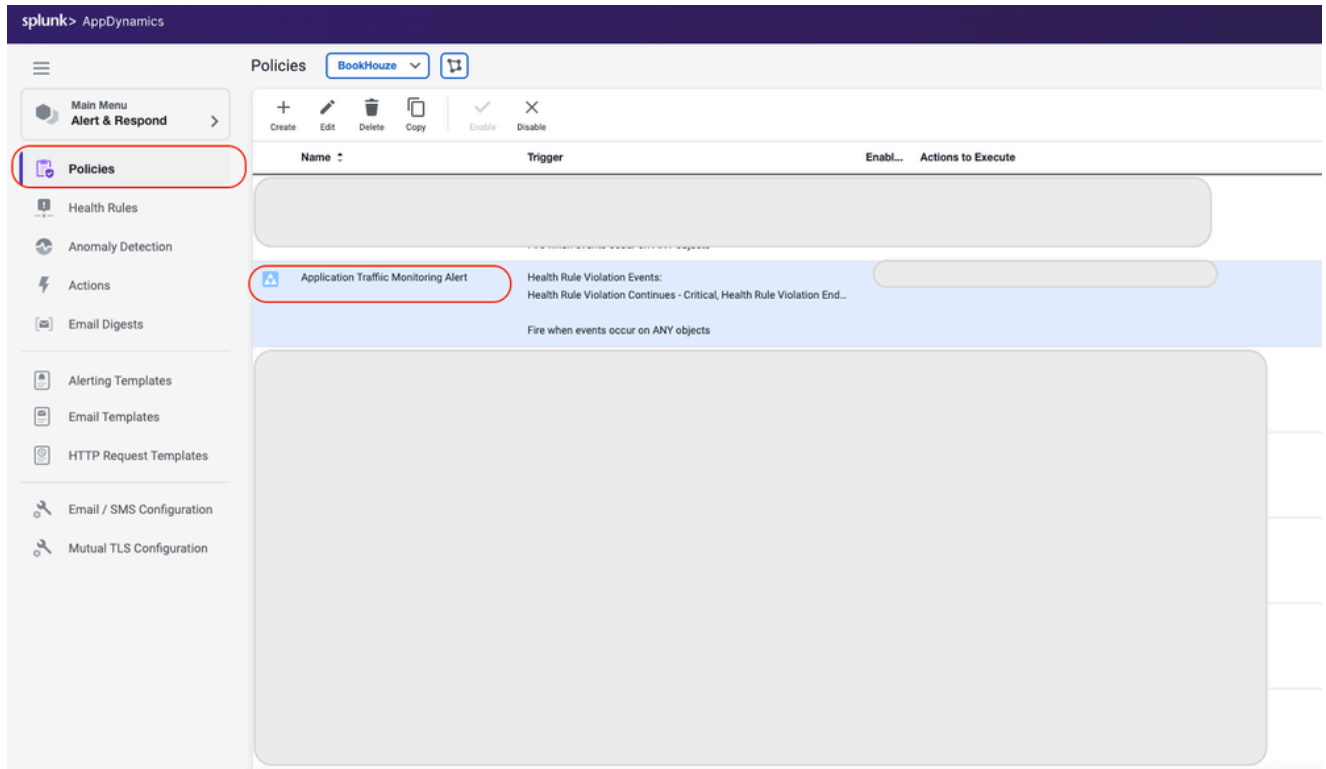
2. Give name to this HTTP Request Template and choose the POST method:
3. Enter copied Webex Webhook endpoint for messages from step 1:

4. Enter your payload. A sample payload is provided. Update based on your business needs:

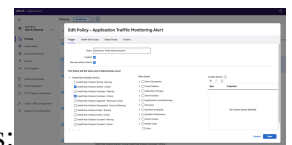
```
{
  "markdown": "AppDynamics Alert: ${latestEvent.displayName}\nSeverity: ${latestEvent.severity}\nTime: "
}
```

Step 3: Link the HTTP Request Template to a Policy

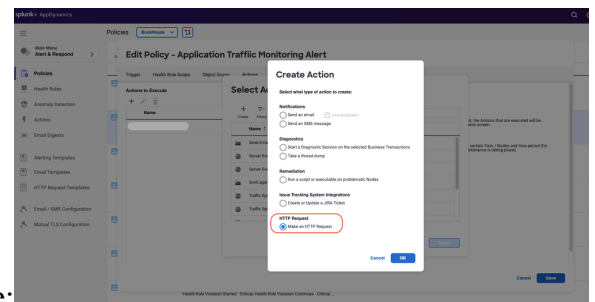
1. Go to **Alert & Respond > Policies** in AppDynamics:



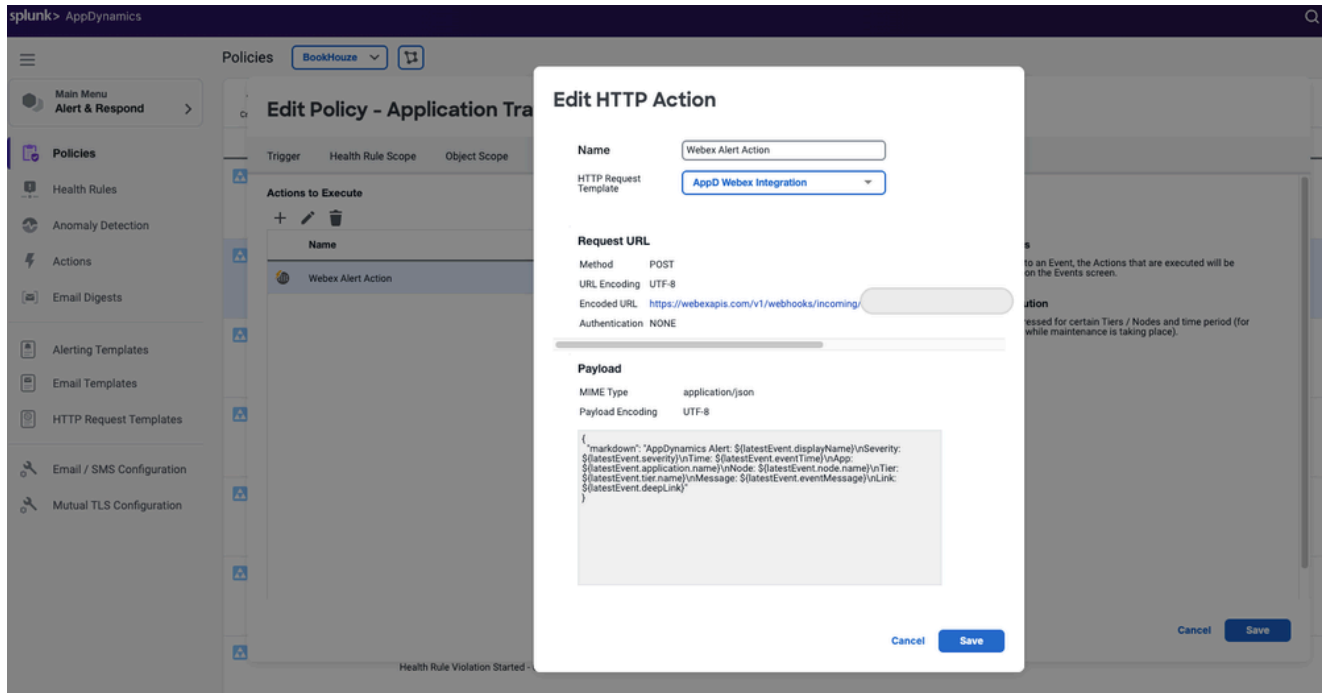
2. Create or edit a policy that triggers on your desired health rules or custom events:



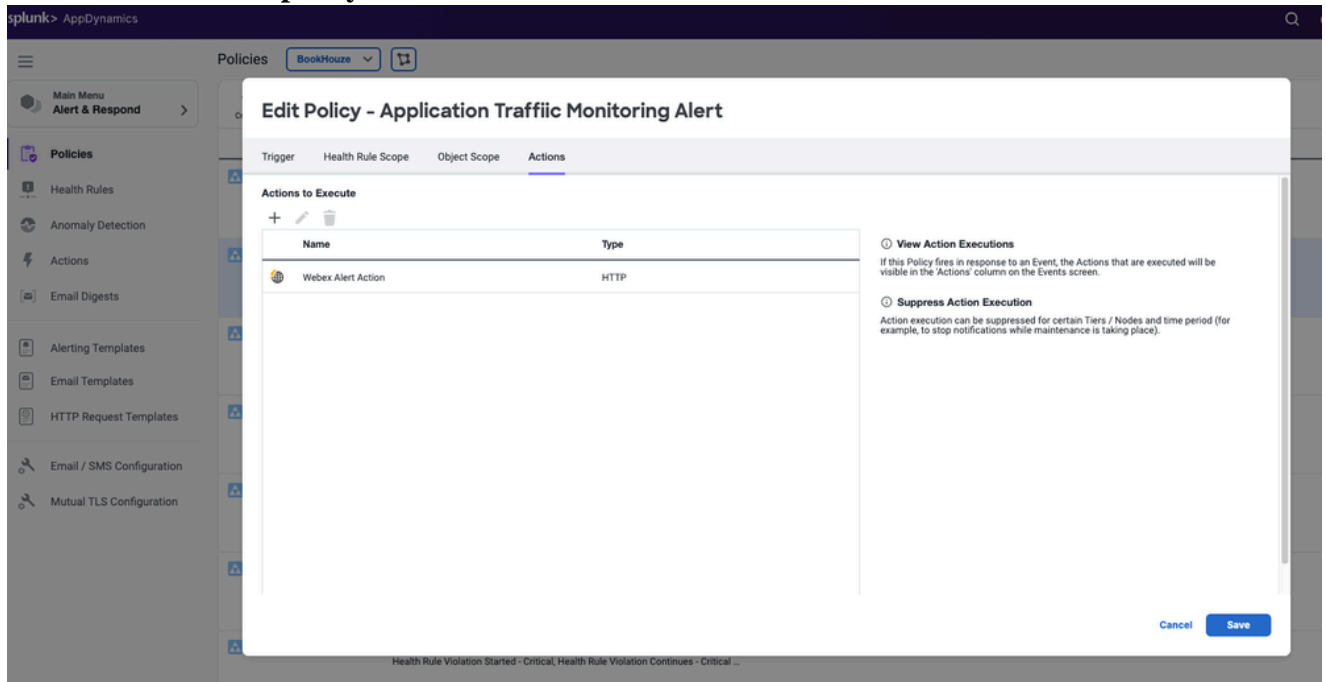
3. Add an **Action** to use your new HTTP Request Template:



The HTTP Request Template name populates automatically, as created in Step 2:



4. Save and enable the policy:



Verify

- Trigger a test alert in AppDynamics:

- Check the designated Webex space for the alert message:

- Confirm that the alert includes all relevant details and links back to AppDynamics for further investigation.

Troubleshoot

Issue	Troubleshooting Steps
No Message in Webex	<ul style="list-style-type: none"> - Verify the bot token and permissions - Check AppDynamics HTTP request logs for errors - Ensure the space ID and API endpoint are correct
Authentication Errors	<ul style="list-style-type: none"> - Confirm the Webhook URL is correct
Message Formatting Issues	<ul style="list-style-type: none"> - Validate your JSON payload - Confirm AppDynamics variable names are correct

Conclusion

By integrating AppDynamics with Webex, you empower your IT and DevOps teams to respond to alerts more quickly, reduce time to resolution, and increase service reliability. This low-code integration streamlines incident collaboration right where your teams already work.

Need Further Assistance?

If you have a question or are experiencing issues, please reach out to [AppDynamics Support](#) and include details such as error messages, configuration information, or relevant logs to help expedite troubleshooting.

Related Information

- [AppDynamics Documentation](#)
- [HTTP Request Template](#)