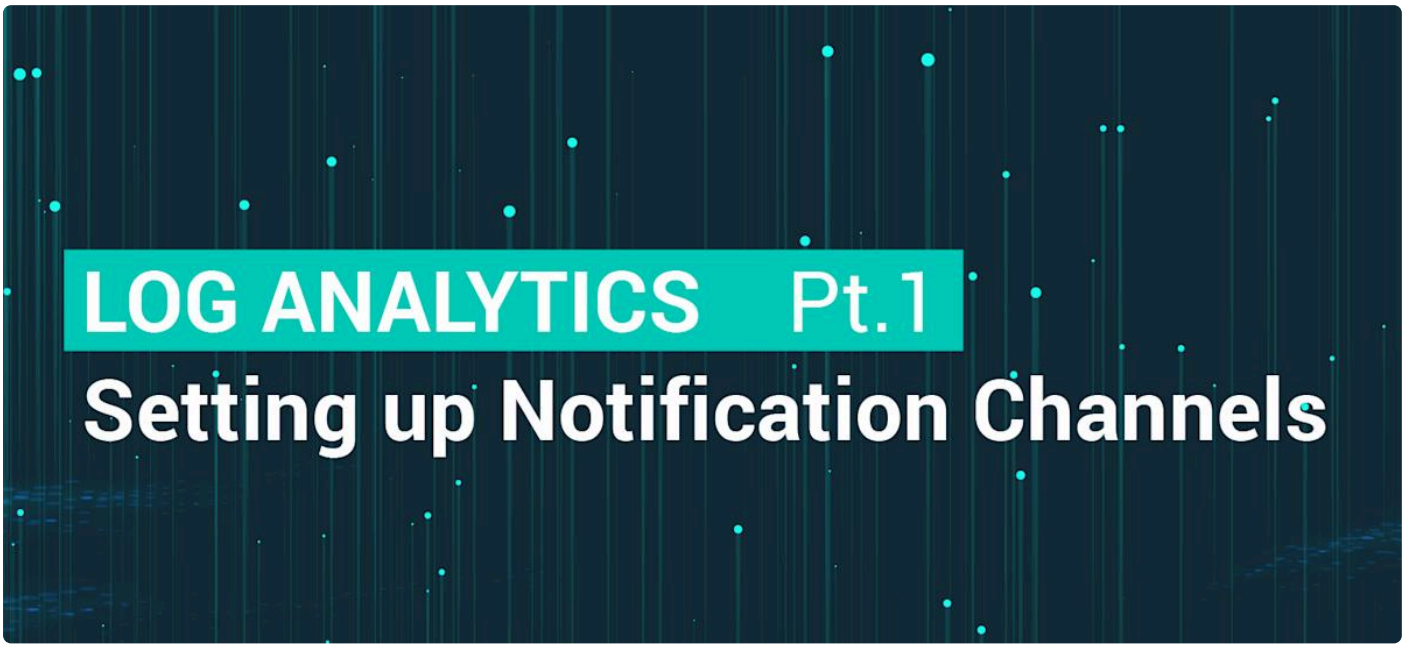
Log Monitoring with Signals Pt.1: Setting up Notification Channels

Jochen Kressin · 2024-07-18



Search Guard Signals is a powerful monitoring and alerting solution designed for Elasticsearch environments. It enables users to proactively monitor their data by setting up notifications for specific events or anomalies detected. With a range of supported notification channels like Email, Slack, and PagerDuty, Search Guard Alerting ensures timely awareness and response to critical system events.

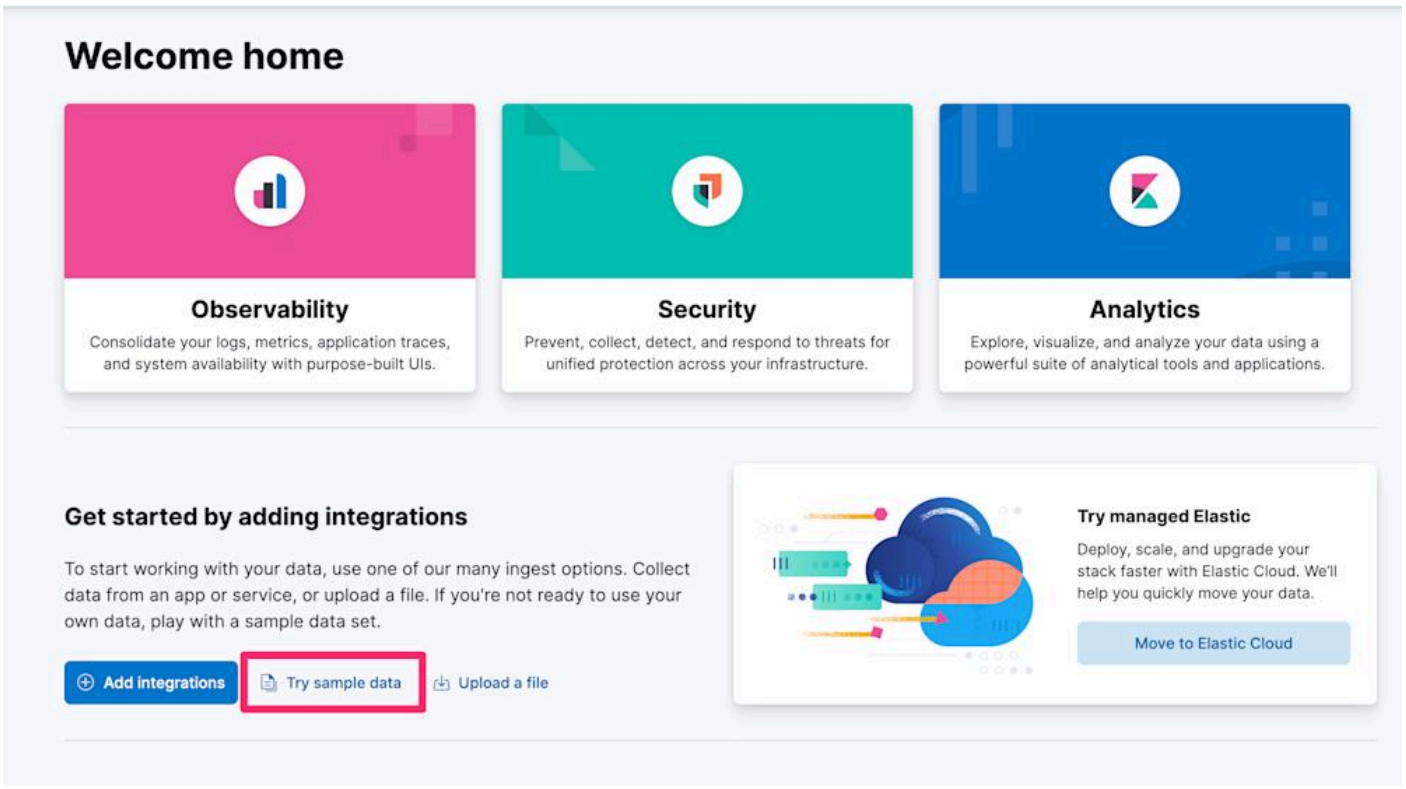
In this article series, we will show you how to use Signals to monitor ingested log data for anomalies and send notifications on various channels if such an anomaly is detected.

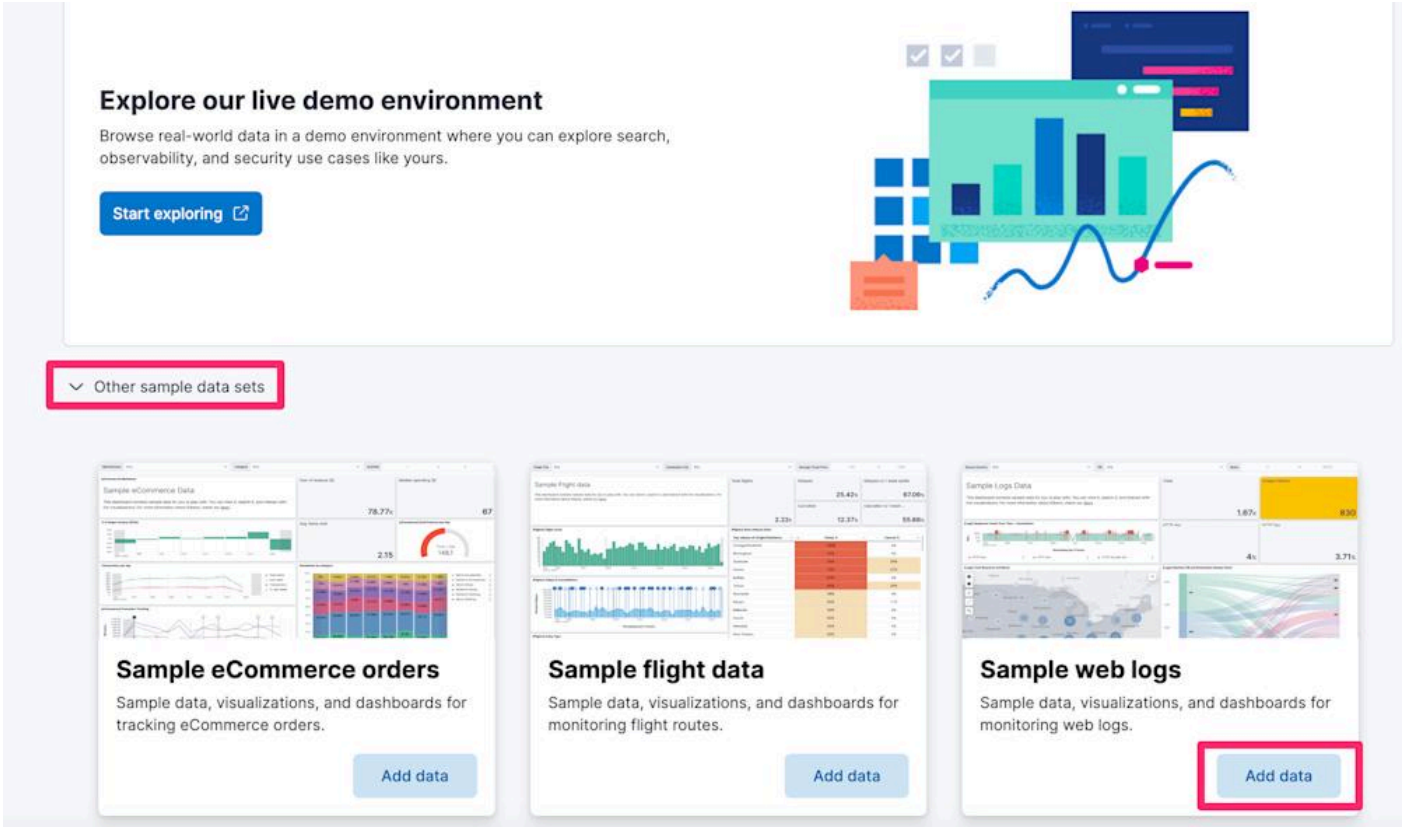
Our first objective is to set up email notifications when errors spike in the logs. Additionally, we'll configure a Slack channel to receive these notifications in parallel. Later, we'll use *Severity Levels* to determine whether to send notifications to Slack or Email based on the severity of the detected errors.

This article will walk you through preparing sample log data and setting up notification channels. In the next part, we'll create a simple watch that monitors the log files index for errors.

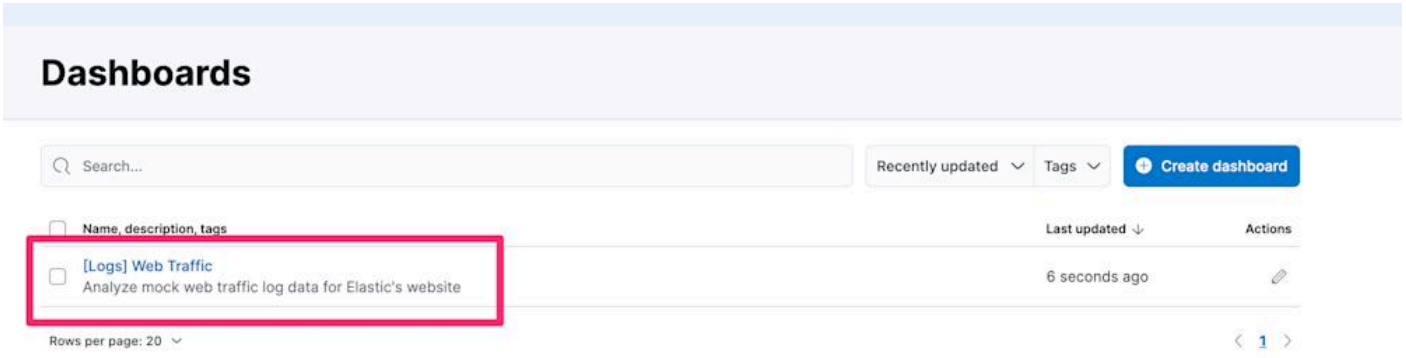
Importing the Sample Data

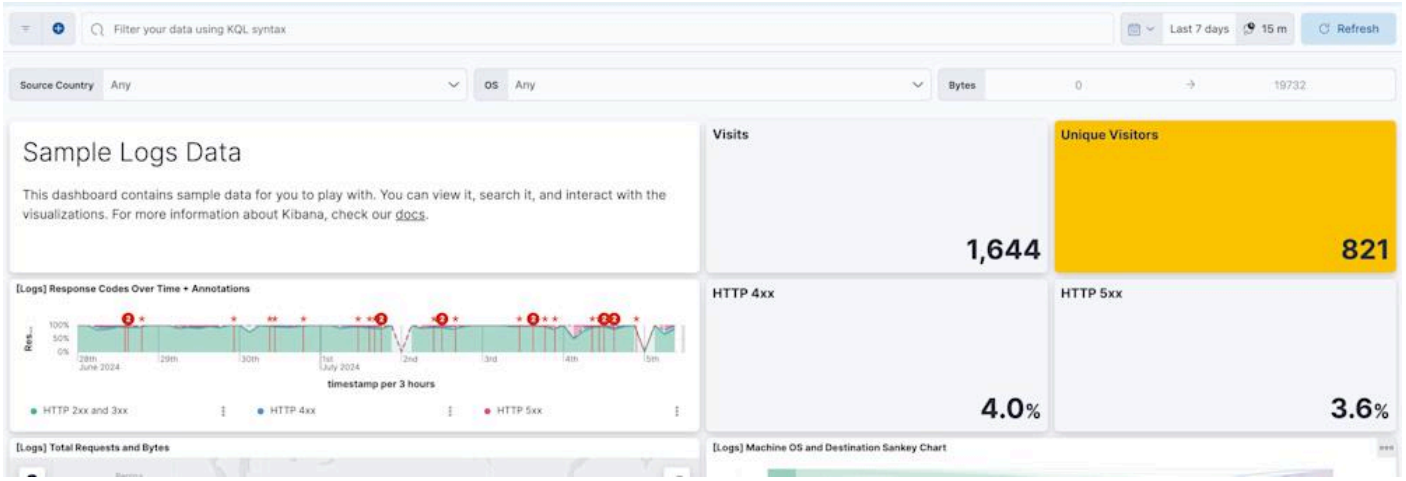
First, we'll import the Kibana sample log data. On the home screen, click on "Add sample data" and select the "Sample web logs" data:





This will import sample weblog data and set up a dashboard with some visualizations:





A sample document in our logs index looks roughly like this (some fields have been omitted for brevity):

```
{
  "agent": "Mozilla/5.0 (X11; Linux x86_64; rv:6.0a1) Gecko/20110421 Firefox/6.0a1",
  "bytes": 5297,
  "clientip": "212.46.165.124",
  "extension": "deb",
  "host": "artifacts.elastic.co",
  "index": "kibana_sample_data_logs",
  "ip": "212.46.165.124",
  "machine": {
    "ram": 5368709120,
    "os": "osx"
  },
  ...
  "memory": null,
  "referer": "http://twitter.com/success/charles-fullerton",
  "request": "/elasticsearch/elasticsearch-6.3.2.deb",
  "response": 200,
  "@timestamp": "2024-07-07T18:40:10.094Z",
  "url": "https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-6.3.2.deb",
  "utc_time": "2024-07-07T18:40:10.094Z",
}
```

The document contains a *response* field with the request's HTTP response code. We'll use this field to check for errors and the *timestamp* field to check for errors within a specific time period.

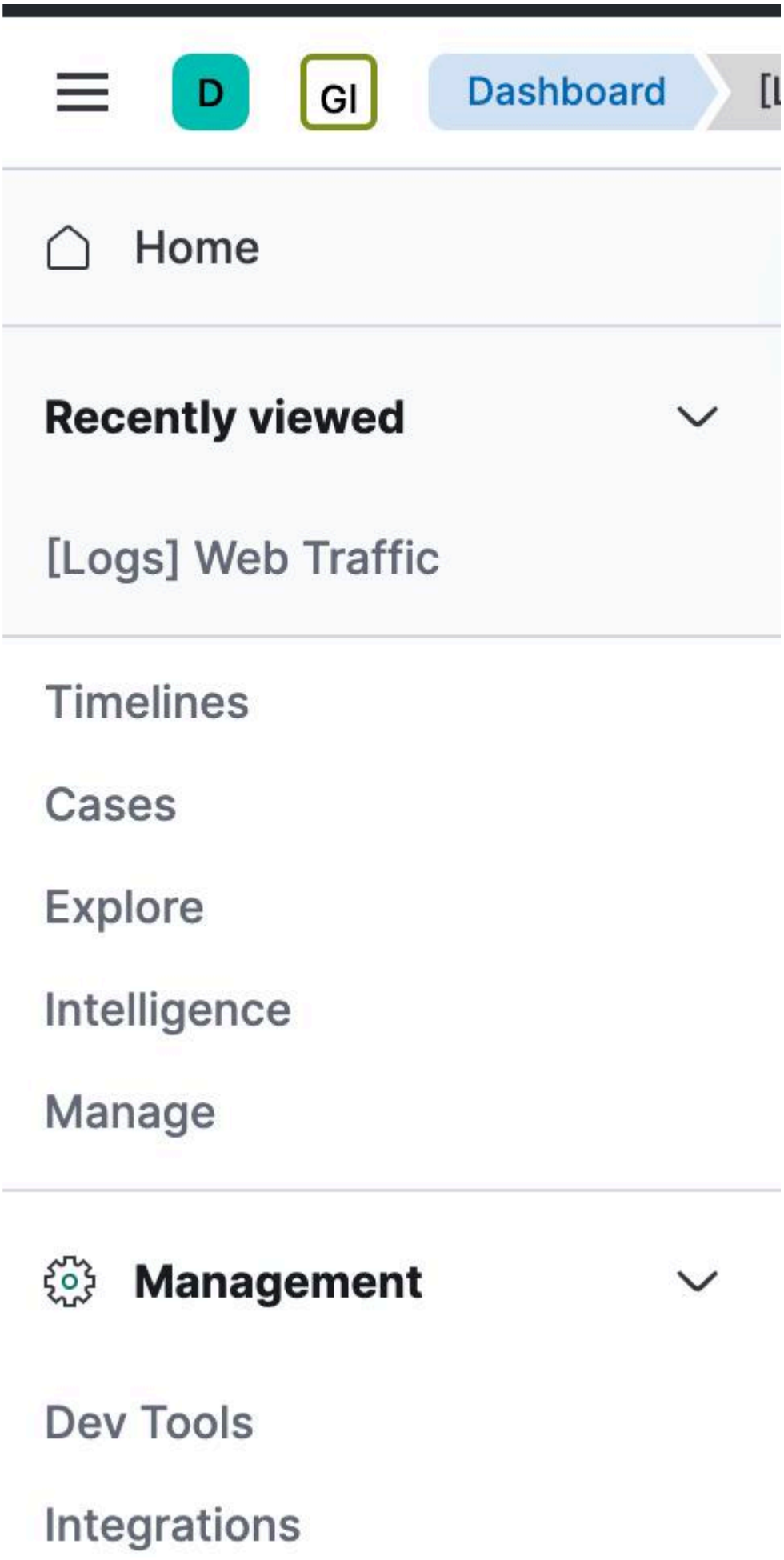
Adding an Email Notification Channel

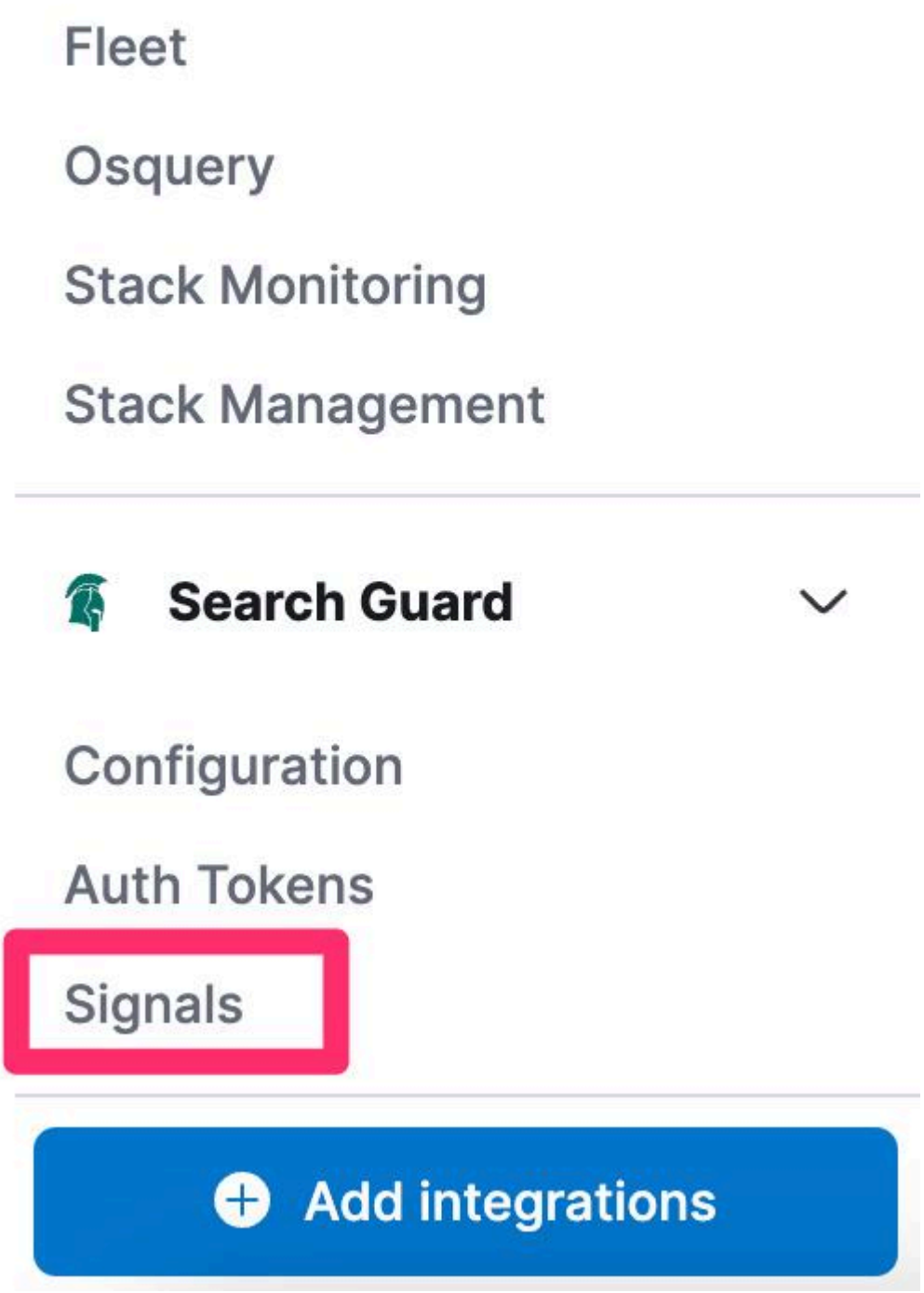
Signals supports several notification channels:

- Email
- Slack
- PagerDuty
- JIRA
- Webhooks
- Index (to write data back to an Elasticsearch index)

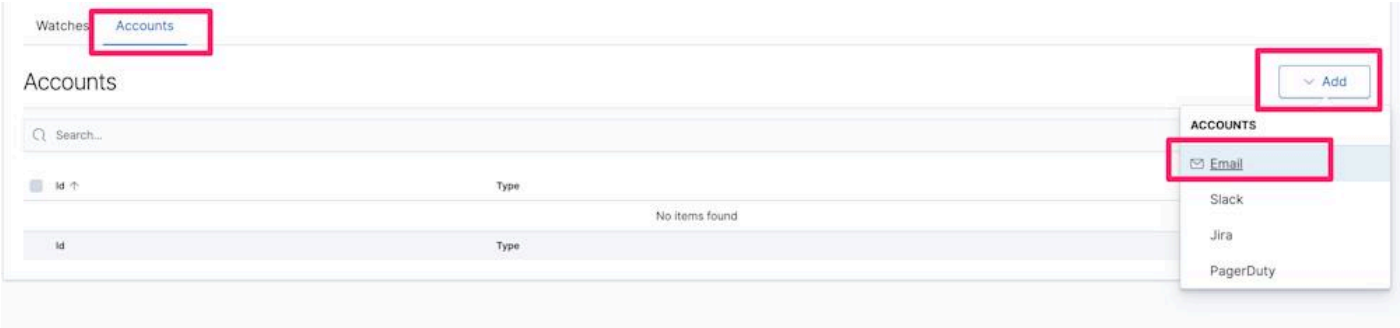
In this example, we'll set up an email notification for unusual error rates in our logs. While everything can be configured via the Signals API, we'll use the Signals Kibana UI for simplicity.

First, select "Alerting" from the Dashboards Navigation:

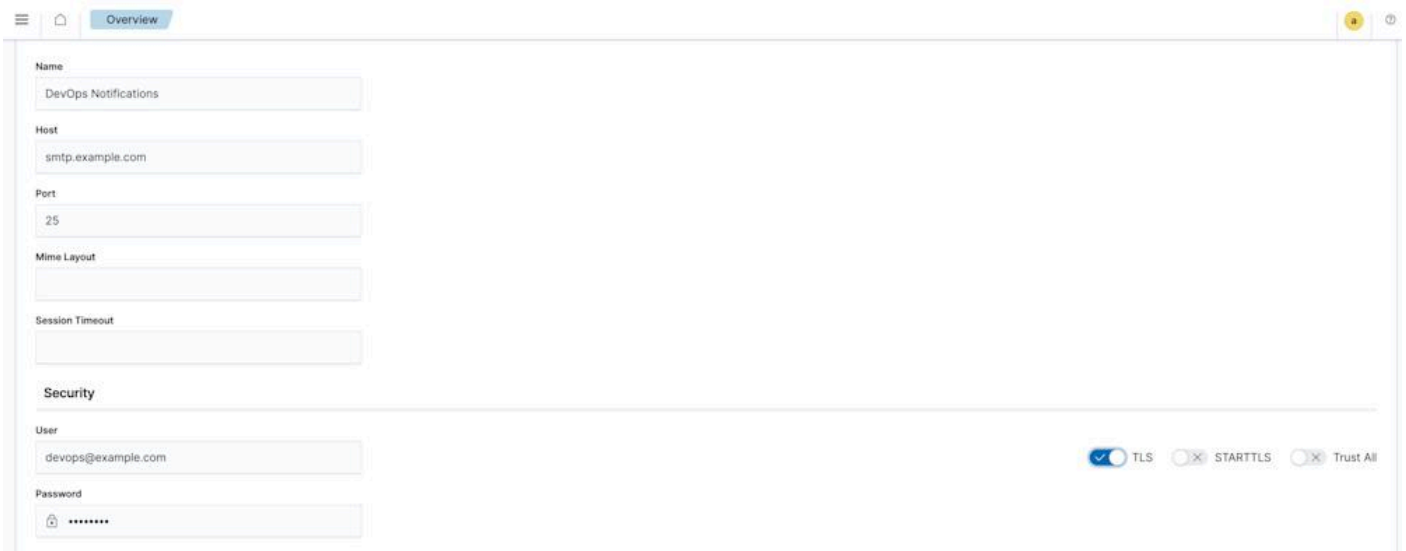

D
GI
Dashboard
Home
Recently viewed
[Logs] Web Traffic
Timelines
Cases
Explore
Intelligence
Manage
Management
Dev Tools
Integrations



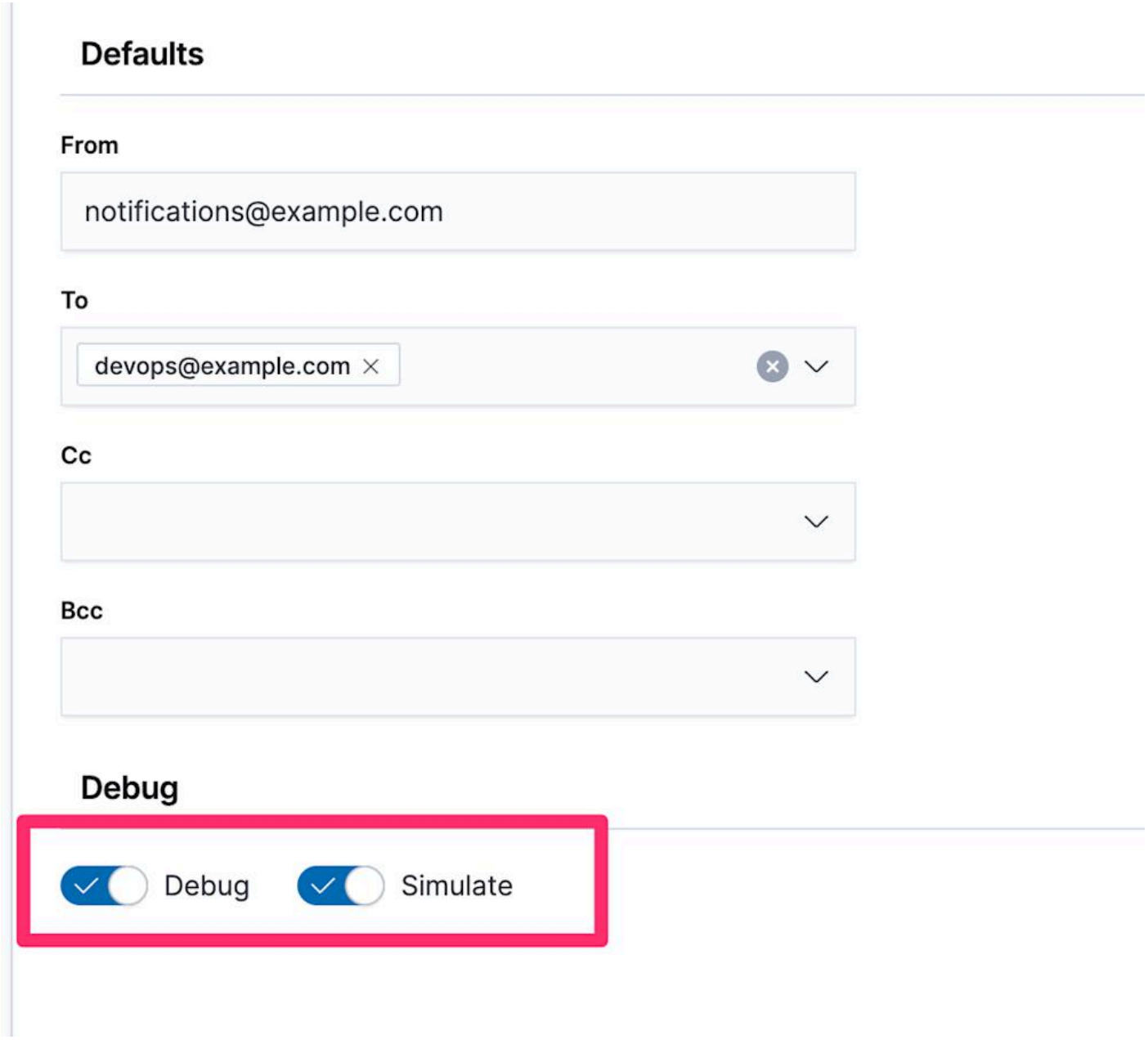
To set up an email notification channel, click on "Accounts" → "Add" → "Email".



On the next screen, configure your Email server connection details. The minimal fields required are the hostname, port, and the credentials of the Email account:



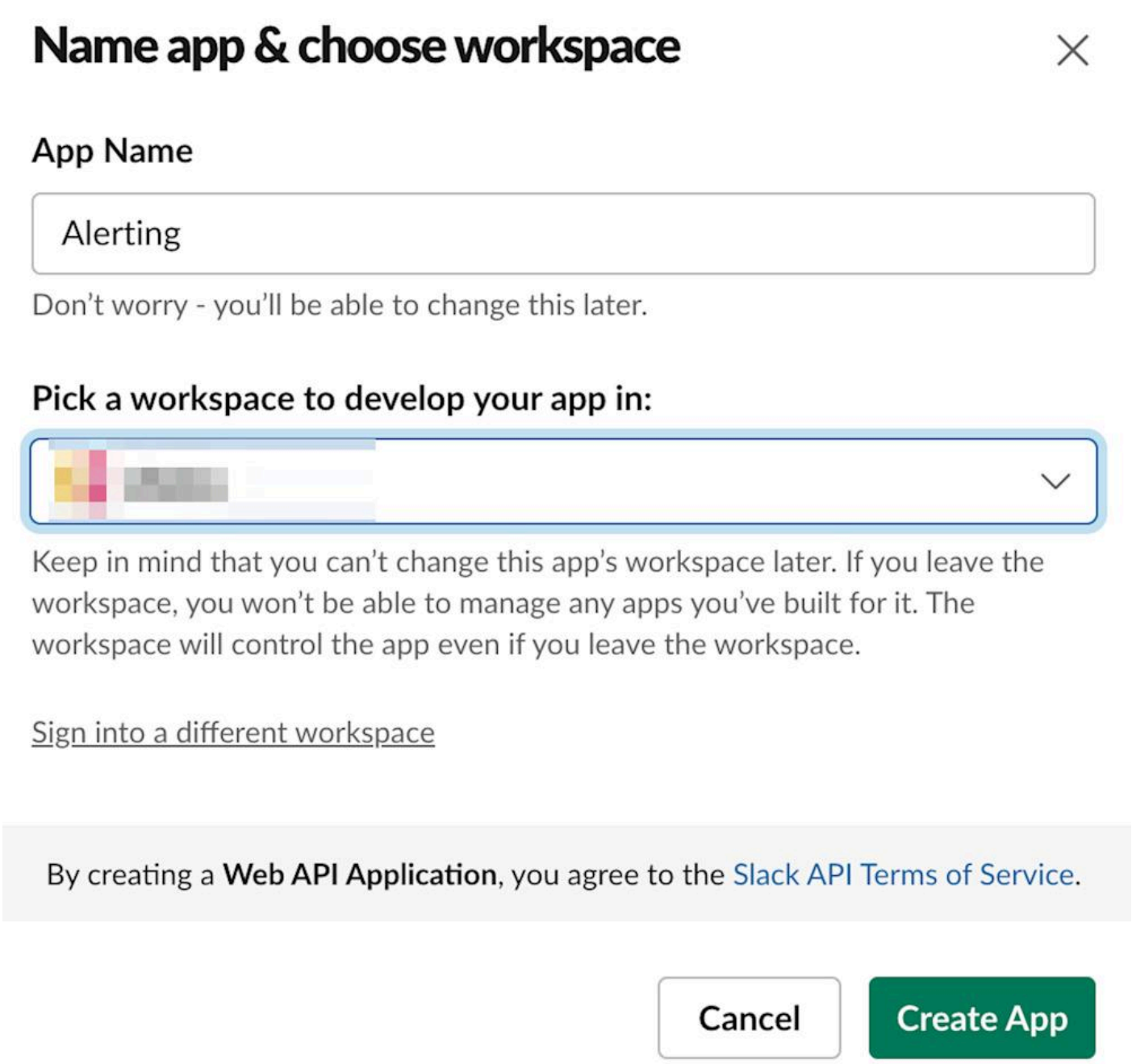
We want to verify our notifications before sending actual emails. Enable the *Simulate* mode and set *Debug* to true. *Simulate* mode provides information about the Email that would be sent without actually sending it, which is great for testing the setup. We can disable these options after setting up and testing the watch.



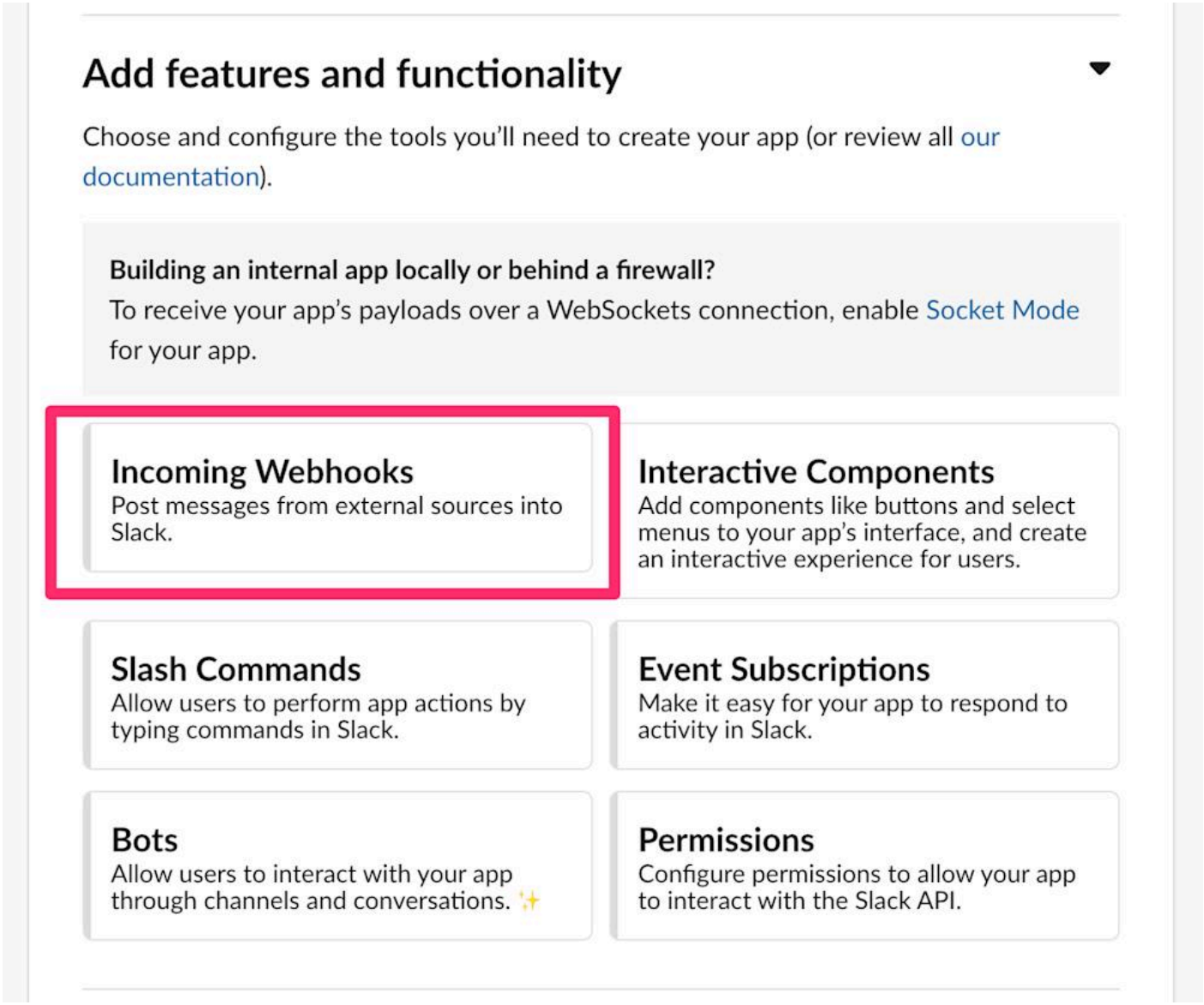
Adding a Slack Notification Channel

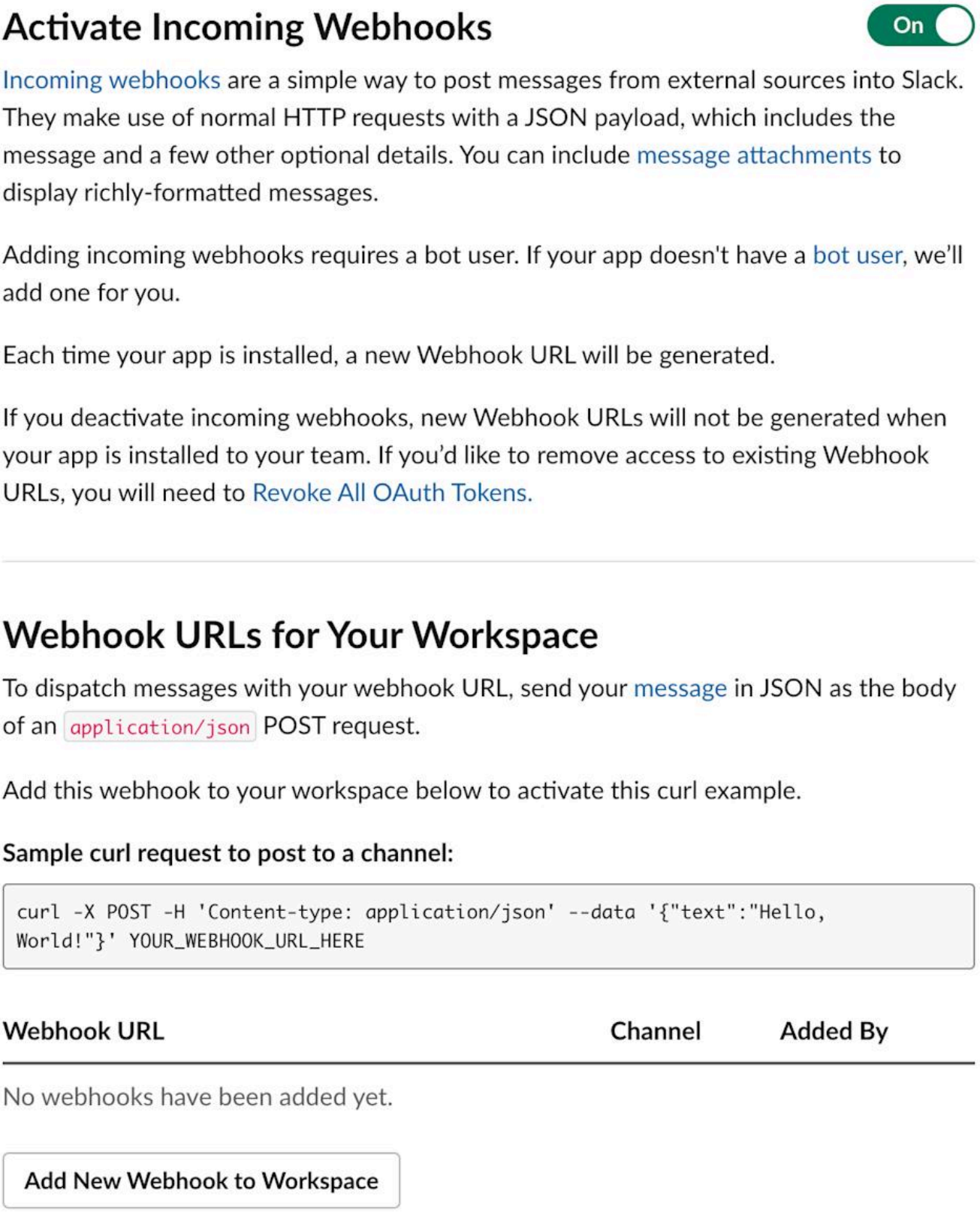
Next, we'll set up a Slack notification channel. First, add a new channel to Slack called *alerting*. To send notifications to this channel, add a Slack App. This can be done with just a few clicks.

From the Slack Admin UI, add a new App, give it a name, and define the Slack Workspace to add the App to:

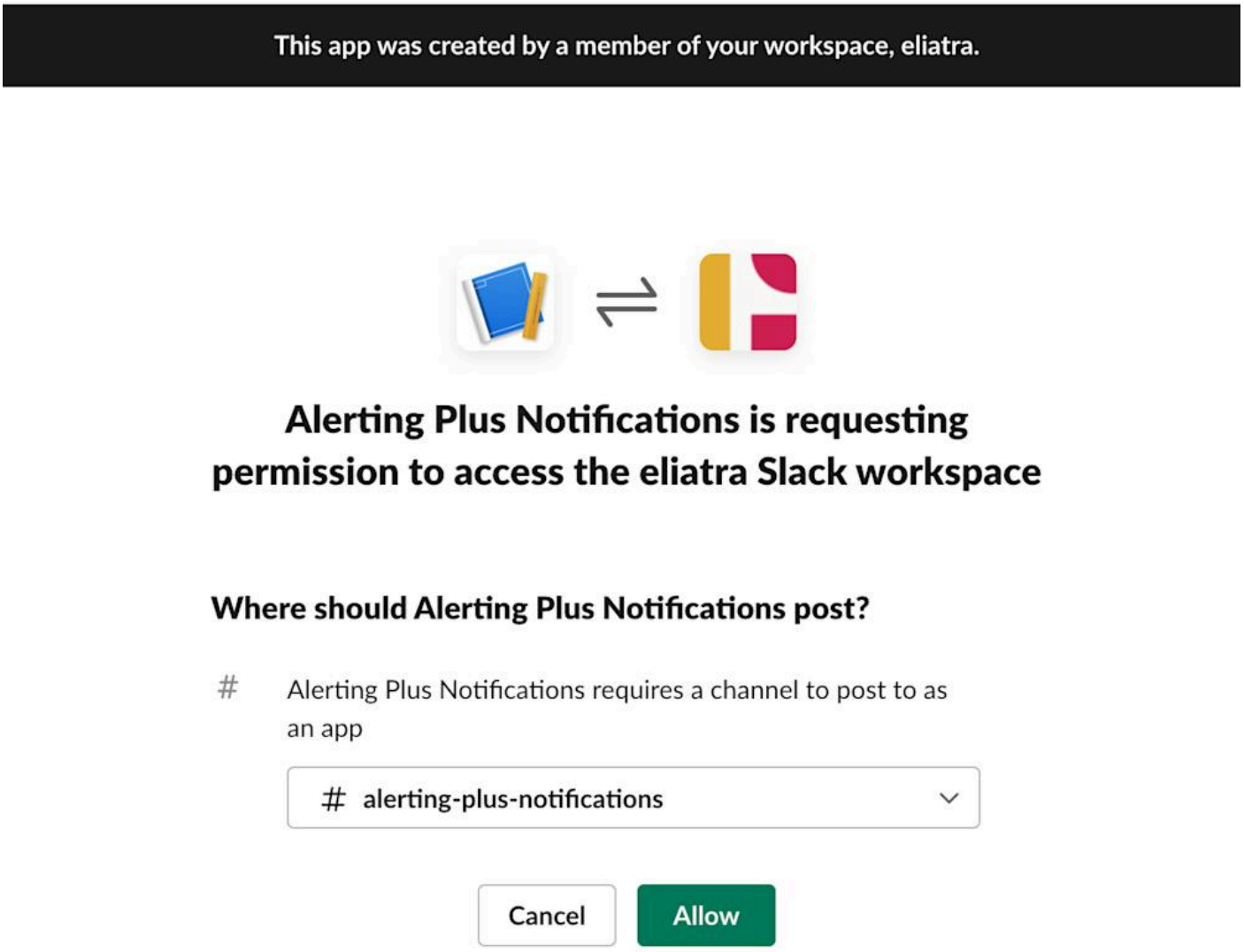


Next, choose "Add features and functionality" and select "Incoming Webhook":

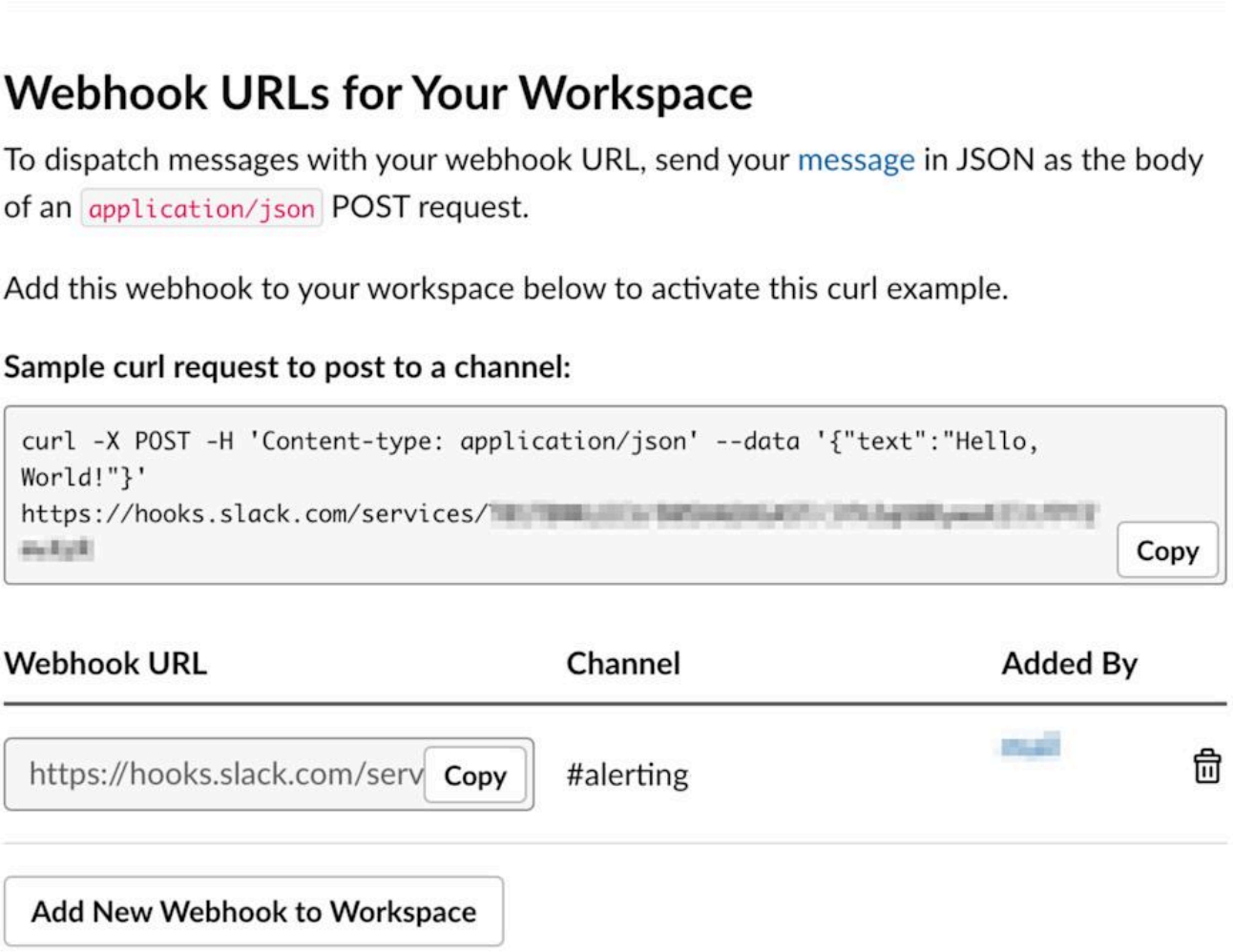

Add features and functionality
Choose and configure the tools you'll need to create your app (or review all our documentation).
Building an internal app locally or behind a firewall?
To receive your app's payloads over a WebSockets connection, enable Socket Mode for your app.
Incoming Webhooks
Post messages from external sources into Slack.
Interactive Components
Add components like buttons and select menus to your app's interface, and create an interactive experience for users.
Slash Commands
Allow users to perform app actions by typing commands in Slack.
Event Subscriptions
Make it easy for your app to respond to activity in Slack.
Bots
Allow users to interact with your app through channels and conversations.
Permissions
Configure permissions to allow your app to interact with the Slack API.



On the next screen, select the channel to post to, in this case, *alerting*.



After setting up the webhook, copy the webhook URL provided:

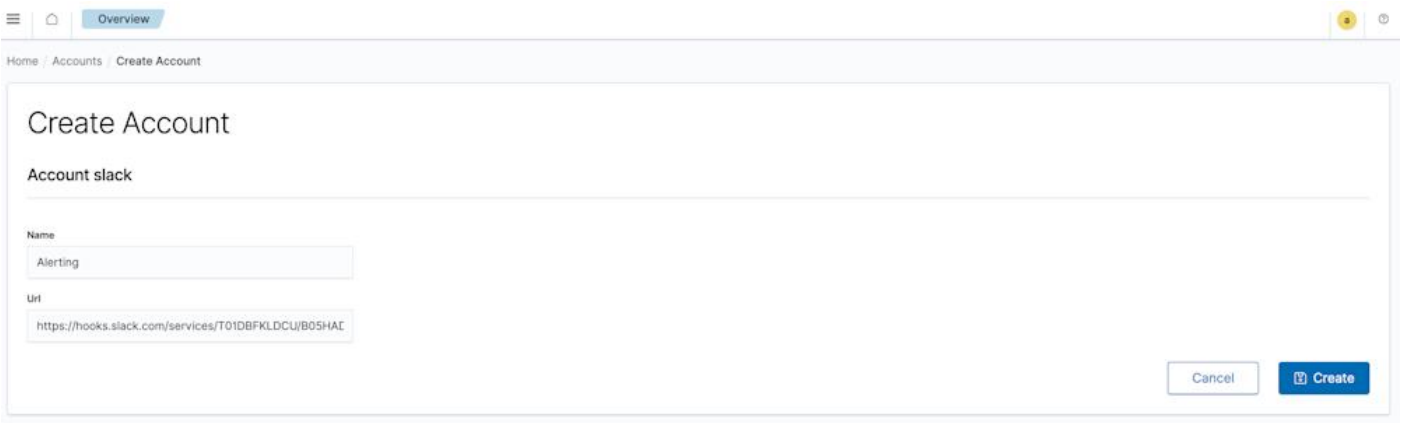


The URL will be in the format:

```
https://hooks.slack.com/services/XXXX/YYYY/ZZZZ
```

The URL will be in the format: `https://hooks.slack.com/services/XXXX/YYYY/ZZZZ`

Copy this URL, go to the Signals Accounts page, and add a new Slack channel. The required entries are the channel name and the Webhook URL:



That's it for this post. In the next article, we'll set up a watch to monitor our log data and send notifications to the channels we've created.

Articles in this series

- Log monitoring with Signals, Pt.1: Setting up Notification Channels (this article)
- Log monitoring with Signals, Pt.2: Setting up a Watch in Blocks Mode
- Log monitoring with Signals, Pt.3: Sending Notifications
- Log monitoring with Signals, Pt.4: Implementing Escalation Levels

--- Image source: Sutterstock-Vektorgrafik ID: 1007122360 Garry Killian

This article was published on the Search Guard blog: https://search-guard.com/blog/log-monitoring-signals-pt-1-notification-channels/