Setting Up Aerospike Monitoring with Datadog

This guide outlines the steps required to set up Aerospike monitoring with Datadog using the OpenTelemetry Collector (otel-collector) and Datadog Agent. It covers the installation and configuration of both OpenTelemetry Collector and Datadog Agent to ensure metrics from Aerospike are correctly sent to Datadog for monitoring.

Prerequisites:

- **Aerospike Cluster**: Ensure that your Aerospike cluster is running, and you have access to the Aerospike server.
- **Datadog Account**: You must have a Datadog account to collect metrics.
- **Docker (optional)**: you can run the OpenTelemetry Collector and Datadog Agent in Docker containers.

Aerospike Monitoring with Datadog via OpenTelemetry Collector (otel-collector)

Step 1: Clone the Aerospike Monitoring Repository

First, clone the Aerospike monitoring repository and navigate to the otel directory:

- git clone https://github.com/aerospike/aerospike-monitoring.git
- cd examples/otel

Step 2: Configure the docker-compose.yml and otel-collector

datadog-docker-compose.yml

- The docker-compose file contains the services such as aerospike-prometheus-exporter, otel-collector. These services are connected to a shared network (aerospike_otel_nw) with specific environment configurations and volume mounts.

datadog-otel-collector-config.yml

- This file is the configuration for the otel-collector. It sets up receivers, processors, exporters, and service pipelines for handling traces.
- **Important**: Update the datadog-api-site and datadog-api-key in this configuration file to match your Datadog account details.

```
datadog:
  api:
    site: datadoghq.com
    key: <DATADOG-APP-KEY>
```

Step 3: Run the Docker Compose Command

Once everything is configured, start the services using Docker Compose:

- docker-compose -f datadog-docker-compose.yml up -d

Integrating Datadog Agent with Aerospike Prometheus Exporter

Step1: Navigate to the Datadog Agent Configuration Directory

Go to the Datadog Agent's configuration directory for Aerospike integration.

- cd /etc/datadog-agent/conf.d/aerospike

Step 2: Copy the Example Configuration File

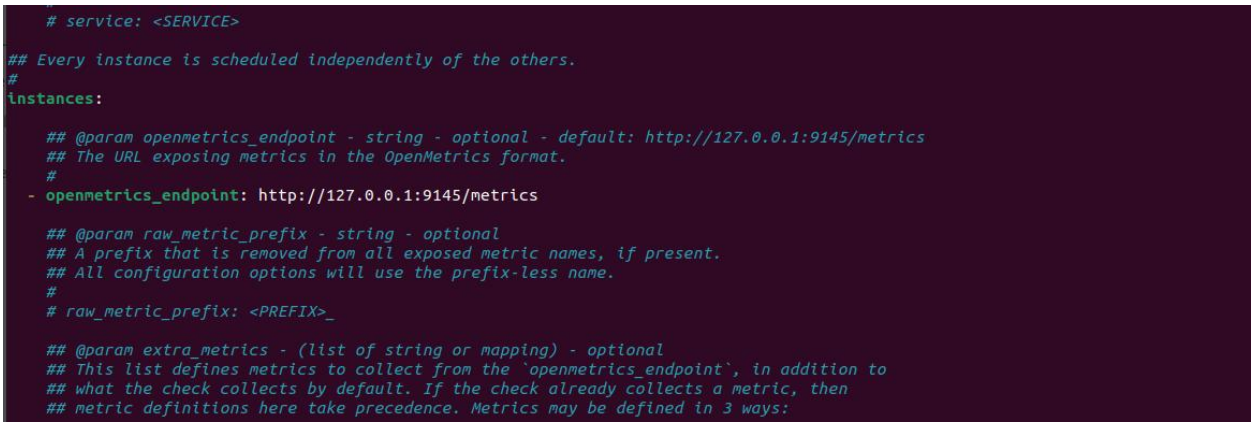
Copy the example configuration file to conf.yaml.

- cp conf.yaml.example conf.yaml

Step 3: Update the OpenMetrics Endpoint

Edit the conf.yaml file and update the openmetrics_endpoint with the IP address or container name of your Aerospike Prometheus exporter. This configuration tells the Datadog Agent where to scrape metrics from the Prometheus exporter.

- http://<AEROSPIKE_PROMETHEUS_EXPORTER_IP>:9145/metrics



Importing Dashboards into Datadog UI

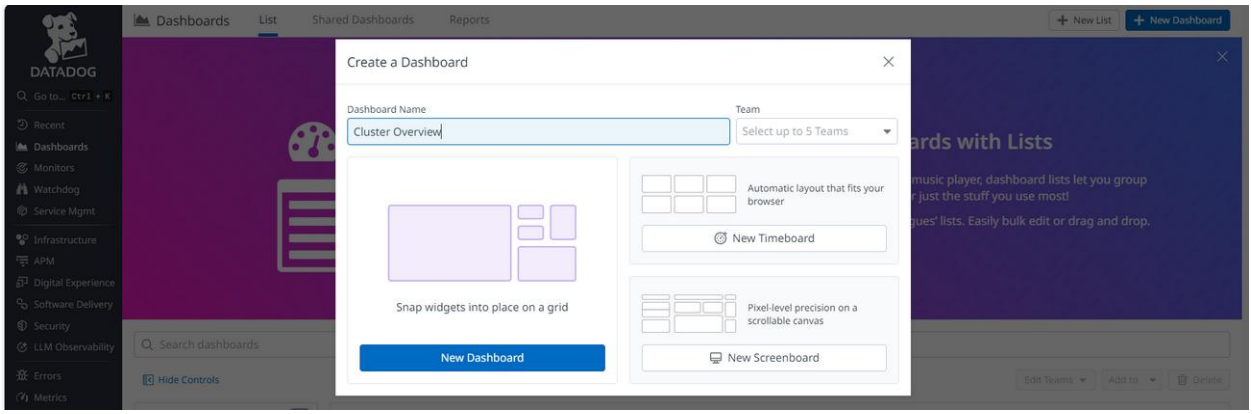
Datadog allows you to import predefined or custom dashboards into the UI. Follow the steps below to import your Aerospike monitoring dashboards into Datadog.

Step 1: Log in to Datadog

- Open your web browser and log in to your Datadog account

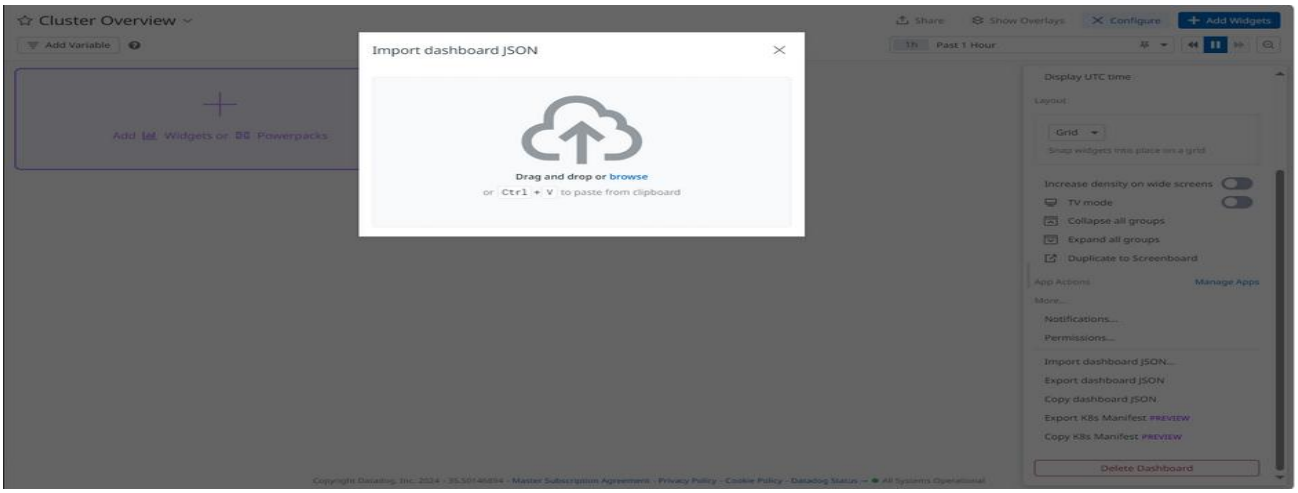
Step 2: Navigate to Dashboards

- In the Datadog UI, click on the **"Dashboards"** tab located in the left-hand sidebar.
- Click the **"New Dashboard"** button at the top-right of the page.



Step 3: Import the Dashboard JSON

- In the "New Dashboard" screen, click on the **"configure"** option on the top-right side.
- Select the import dashboard JSON option.
- Copy the contents of your dashboard JSON file and paste it or drag the JSON file into the field.
- Click **"Import"** to load the dashboard.



Creating Bulk Monitors in Datadog:

To create bulk monitoring alerts in Datadog, follow the steps below using the provided Python script. This script reads multiple monitor configurations from a JSON file and creates monitors via the Datadog API.

Prerequisites:

- **Datadog API Key** and **Application Key** are required.
- **Python** must be installed on your machine.

Steps to Create bulk monitors in Datadog:

The Script datadog_alerts_creation.py reads alert rules from a JSON file and creates corresponding monitors in Datadog using the Datadog API.

Important:

- **api_key** and **app_key**: These should be updated with your actual Datadog API and Application keys.
- **datadog_site**: Update this with the site where your Datadog account is hosted.
- **aerospike_rules.json**: This JSON file contains multiple monitors, each with its configuration like monitor name, type, query, and message.

Run the Python script with:

```
python datadog_alerts_creation.py
```