**Setting Up Kubernetes API Server Monitoring in Datadog (GKE)**

### 1. ****Enable API Server Metrics in GKE****

* Ensure Kubernetes API server metrics are exposed:

kubectl get --raw /metrics

* In GKE, enable monitoring.googleapis.com/kube-apiserver by modifying the monitoring configuration.

### 2. ****Install Datadog Agent in GKE****

* Deploy Datadog Agent as a DaemonSet:

helm repo add datadog https://helm.datadoghq.com

helm repo update

helm install datadog-agent datadog/datadog --set datadog.apiKey=YOUR\_API\_KEY --set datadog.site="datadoghq.com"

### 3. ****Verify Kubernetes API Monitoring****

* Check API server metrics are collected:

datadog-agent status | grep kube\_apiserver

* Ensure API response metrics are being collected:

kubectl get --raw /metrics | grep apiserver\_request\_duration\_seconds

### 4. ****Querying GKE API Server Metrics in Datadog****

* Run an API request to Datadog metrics endpoint:

curl -X GET "https://api.datadoghq.com/api/v1/metrics" \

-H "DD-API-KEY: YOUR\_API\_KEY"

### 5. ****DogStatsD Python Script for AdmissionReview HTTP Responses****

from flask import Flask, request, jsonify

from datadog import DogStatsd

import time

app = Flask(\_\_name\_\_)

statsd = DogStatsd(host="127.0.0.1", port=8125)

@app.route("/validate", methods=["POST"])

def validate():

start\_time = time.time()

try:

admission\_request = request.get\_json()

admission\_response = {"response": {"uid": admission\_request["request"]["uid"], "allowed": True}}

response\_code = 200

except Exception as e:

response\_code = 500

admission\_response = {"error": str(e)}

response\_time = time.time() - start\_time

statsd.increment("admissionreview.http\_response\_status", tags=[f"status\_code:{response\_code}"])

statsd.histogram("admissionreview.response\_time", response\_time)

return jsonify(admission\_response), response\_code

if \_\_name\_\_ == "\_\_main\_\_":

app.run(host="0.0.0.0", port=5000)

### 6. ****Deploying Python Script in Kubernetes****

* Create a Kubernetes Deployment:

apiVersion: apps/v1

kind: Deployment

metadata:

name: admissionreview-metrics

spec:

replicas: 1

selector:

matchLabels:

app: admissionreview-metrics

template:

metadata:

labels:

app: admissionreview-metrics

spec:

containers:

- name: admissionreview-metrics

image: myrepo/admission-metrics:latest

ports:

- containerPort: 5000

env:

- name: DD\_AGENT\_HOST

value: "127.0.0.1"

### 7. ****Verify Datadog Metrics****

* Check if Datadog agent is receiving metrics:

datadog-agent status | grep admissionreview

* Query logs for debugging:

sudo journalctl -u datadog-agent -f | grep "admissionreview"

### 8. ****Test API Server Response Metrics****

* Send a test AdmissionReview request:

curl -X POST http://localhost:5000/validate -H "Content-Type: application/json" -d '{

"apiVersion": "admission.k8s.io/v1",

"kind": "AdmissionReview",

"request": {"uid": "1234", "object": {}}

}'

### 9. ****Create Datadog Dashboard****

* Add **Metrics:**
  + admissionreview.http\_response\_status
  + admissionreview.response\_time
* Filter by **status\_code** and set alerts on high response times.

### 10. ****Automate with Systemd (Linux Only)****

* Create a systemd service file:

sudo nano /etc/systemd/system/admission-metrics.service

* Add:

[Unit]

Description=Kubernetes Admission Controller Metrics Service

After=network.target

[Service]

User=root

ExecStart=/usr/bin/python3 /path/to/admissionreview\_metrics.py

Restart=always

[Install]

WantedBy=multi-user.target

* Start the service:

sudo systemctl enable admission-metrics

sudo systemctl start admission-metrics