Step-by-Step Guide: Using Datadog in DevOps

Table of Contents:

Introduction

Datadog is a comprehensive monitoring platform for cloud-scale applications. It enables DevOps teams to track system metrics, monitor services, analyse logs, and perform tracing.

This guide will walk you through the steps to set up Datadog in a DevOps environment and integrate it with cloud services like AWS, Kubernetes, Docker, and more.

Prerequisites

- Access to the Datadog platform sign up
- Root/admin privileges on your systems
- Familiarity with cloud platforms (AWS, GCP, Azure) or containerized environments (Docker, Kubernetes)
- A machine with internet access

Setting Up Datadog

1. Create a Datadog Account:
   - Sign up at Datadog with a free or paid plan.
   - After registration, you will be provided an API key and access to the Datadog dashboard.

2. Get the API Key:
   - Go to Integrations > APIs on the dashboard and copy the API key for agent configuration.

Agent Installation

Datadog agents are lightweight applications installed on your servers, containers, or cloud environments to collect metrics, logs, and traces.

Linux Installation

1. Run Installation Command:

```
DD_API_KEY=<your_api_key> bash -c "$(curl -L
https://s3.amazonaws.com/dd-agent/scripts/install_script.sh)"
```

2. Verify Installation:

   After installation, the agent should start automatically. Verify with:

```
sudo datadog-agent status
```

3. Configure the Agent:

   Edit the `/etc/datadog-agent/datadog.yaml` file to add custom configurations.

Windows Installation

1. Download the Agent Installer:
   - Go to the Datadog website and download the Windows installer.

2. Run the Installer:
   - Follow the installation steps, and when prompted, enter your API key.

3. Verify Installation:
   - Open a command prompt and run:

```
"C:\Program Files\Datadog\Datadog Agent\embedded\agent.exe"
status
```

Containerized Environments

Docker

1. Run Datadog as a Docker Container:

```
docker run -d --name datadog-agent \
-e DD_API_KEY=<your_api_key> \
-e DD_SITE="datadoghq.com" \
-v /var/run/docker.sock:/var/run/docker.sock:ro \
datadog/agent:latest
```

Kubernetes

1. Deploy Datadog Agent on Kubernetes:
   - Use Helm to install the agent:

```
helm repo add datadog https://helm.datadoghq.com
   helm repo update
   helm install datadog-agent --set datadog.apiKey=<your_api_key>
--set targetSystem=linux datadog/datadog
```

Integrating Datadog with Cloud Services

Datadog provides integrations with multiple cloud providers like AWS, GCP, and Azure.

AWS Integration

1. Navigate to Integrations > AWS in Datadog:
   - Click on AWS
   - Follow the steps to create a new IAM role in AWS.
   - Attach the provided policy to grant Datadog access to your AWS metrics.

2. Configure CloudWatch Metrics:
   - Enable the services and regions from which you want to pull metrics.

Kubernetes Integration

1. Edit the Agent Configuration:
   - Modify the Helm values file or use `kubectl` to configure the agent for Kubernetes cluster monitoring.

Configuring Monitoring & Alerts

1. Create a Monitor:
   - In the Datadog dashboard, navigate to Monitors.
   - Choose the type of metric (e.g., CPU, Memory, Disk) or service (e.g., HTTP status codes) to monitor.
   - Set alert conditions, such as thresholds, and configure notifications (email, Slack, etc.).

2. Set Up Custom Alerts:
   - Create custom alerts based on specific criteria relevant to your infrastructure or services.

Logging and Tracing

Datadog allows log collection and tracing for full-stack observability.

Log Collection

1. Enable Log Collection in the Agent:
   - Modify `/etc/datadog-agent/datadog.yaml` and enable the `logs_enabled` option:

```
logs_enabled: true
```

2. Configure Log Sources:
   - Define custom log collection paths or services in `/etc/datadog-agent/conf.d`.

```
APM (Application Performance Monitoring)
```

APM (Application Performance Monitoring)

1. Enable APM:
   - To enable tracing, edit the agent configuration file and ensure APM is enabled:

```
apm_config:
enabled: true
```

2. Integrate with Your Application:
   - Install the appropriate Datadog APM library for your application (e.g., Python, Java, Node.js).
   - Configure the APM client to send traces to Datadog.

Creating Dashboards

1. Create a New Dashboard:
   - In the Datadog dashboard, navigate to Dashboards > New Dashboard
   - Add widgets to visualise metrics such as system load, memory usage, and application response times.

2. Customise Widgets:
   - Add graphs, heat maps, service maps, and other visualisations to monitor performance in real-time.

Best Practices

- Tagging: Use tags to organise metrics, logs, and traces. This helps in filtering and organising data.
- Limit Alerts: Avoid alert fatigue by fine-tuning alerts to trigger only for critical issues.
- Automate: Use Terraform, Ansible, or other configuration management tools to automate Datadog agent deployment and configuration.

Conclusion

Datadog provides a powerful, all-in-one monitoring platform suitable for cloud-scale applications. By following this guide, you can effectively monitor infrastructure, applications, and services while integrating with popular DevOps tools and practices.