Prerequisites for Setting Up CI/CD with GitHub Actions and AWS EKS

Overview

This document outlines the prerequisites needed to set up a CI/CD pipeline using GitHub Actions to deploy applications to an Amazon Elastic Kubernetes Service (EKS) cluster. It covers the necessary installations and configurations for AWS CLI, `eksctl`, Docker, and GitHub.

---

1. AWS Account and CLI

Create an AWS Account

If you do not already have an AWS account, create one at AWS Console.

Install AWS CLI

To interact with AWS services, install the AWS Command Line Interface (CLI):

For Linux & Mac

```
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o
"awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install
```

For Windows

Download and install from AWS CLI Installation Guide.

Verify AWS CLI Installation

```
aws --version
```

Configure AWS CLI

Run the following command and enter your AWS credentials:

```
aws configure
```

You will need to provide:

- AWS Access Key ID
- AWS Secret Access Key
- Default AWS region (e.g., `us-east-1`)
- Output format (default: `json`)

---

2. Install `eksctl`

`eksctl` is a command-line tool for creating and managing EKS clusters.

For Linux

```
curl -sL
"https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_Linux_amd
64.tar.gz" | tar xz -C /tmp
sudo mv /tmp/eksctl /usr/local/bin
```

For macOS

```
brew tap weaveworks/tap
brew install weaveworks/tap/eksctl
```

For Windows

Download and install from eksctl releases.

Verify `eksctl` Installation

```
eksctl version
```

---

3. Install Docker

Docker is required to build container images for deployment.

For Linux

```
sudo apt update && sudo apt install -y docker.io
sudo systemctl start docker
sudo systemctl enable docker
```

For macOS & Windows

Download and install Docker Desktop.

Verify Docker Installation

```
docker --version
```

To allow running Docker commands without `sudo`:

```
sudo usermod -aG docker $USER
newgrp docker
```

---

4. GitHub Account & Repository

Ensure you have a GitHub account at GitHub.

Create a New GitHub Repository

1. Navigate to GitHub.
2. Click **New Repository**.
3. Provide a repository name and choose **Private** or **Public**.
4. Click **Create Repository**.

Configure AWS Secrets in GitHub

1. Go to **Settings** in your GitHub repository.
2. Navigate to **Secrets and Variables** > **Actions**.
3. Click **New repository secret** and add:
    - `AWS_ACCESS_KEY_ID`
    - `AWS_SECRET_ACCESS_KEY`

---

5. Install `kubectl`

`kubectl` is used to interact with Kubernetes clusters.

For Linux

```
curl -LO "https://dl.k8s.io/release/$(curl -L -s
https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"
sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
```

For macOS

```
brew install kubectl
```

For Windows

Install via Kubernetes CLI.

Verify `kubectl` Installation

```
kubectl version --client
```

Next Steps

Once all prerequisites are installed, proceed with:

1. **Creating an EKS Cluster**

   ```
   eksctl create cluster --name demo-cluster --region us-east-1 --node-type
   t3.micro --nodes 2
   ```

2. **Configuring `kubectl` to Use the EKS Cluster**

   ```
   aws eks update-kubeconfig --name demo-cluster --region us-east-1
   ```

Now, you are ready to set up your CI/CD pipeline using GitHub Actions and deploy applications to AWS EKS.

Happy Coding! 🚀