**Prometheus + Grafana + GitHub ARC Metrics Integration**

**Objective**

Deploy Prometheus and Grafana using Terraform and Helm, and enable metrics scraping from the GitHub ARC (Actions Runner Controller) deployed in EKS.

**Cluster Observations**

* ARC controller pods are running in the runner-controller namespace.
* No Kubernetes Service currently exposes the ARC controller’s /metrics endpoint.
* Prometheus requires a stable DNS target (svc.cluster.local) to scrape metrics reliably.

**Required Kubernetes Service**

The GitHub ARC controller is deployed in the runner-controller namespace. The pod name is **github-controller-ghs-rs-controller**, and it is currently running. To enable Prometheus to scrape metrics from this controller, we need to expose it via a Kubernetes Service.

service:

**apiVersion: v1**

**kind: Service**

**metadata:**

**name: arc-controller**

**namespace: runner-controller**

**spec:**

**selector:**

**app: < Replace with actual pod label>**

**ports:**

**- protocol: TCP**

**port: 8080**

**targetPort: 8080**

**Prometheus Scrape Configuration:**

Create a file named arc-scrape-config.yaml with the following content:

**prometheus:**

**prometheusSpec:**

**additionalScrapeConfigs:**

**- job\_name: "github-arc"**

**static\_configs:**

**- targets:**

**- "arc-controller.runner-controller.svc.cluster.local:8080"(Once we create the service for arc-controller)**

**We have to replace service name and namespace if different.**

**Terraform Code to Deploy Prometheus Stack:**

**terraform {**

**required\_version = ">= 1.3.0"**

**required\_providers {**

**aws = {**

**source = "hashicorp/aws"**

**version = "~> 5.0"**

**}**

**kubernetes = {**

**source = "hashicorp/kubernetes"**

**version = "~> 2.0"**

**}**

**helm = {**

**source = "hashicorp/helm"**

**version = "~> 2.0"**

**}**

**}**

**}**

**provider "aws" {**

**region = var.aws\_region**

**}**

**data "aws\_eks\_cluster" "eks" {**

**name = var.eks\_cluster\_name**

**}**

**data "aws\_eks\_cluster\_auth" "eks" {**

**name = var.eks\_cluster\_name**

**}**

**provider "kubernetes" {**

**host = data.aws\_eks\_cluster.eks.endpoint**

**cluster\_ca\_certificate = base64decode(data.aws\_eks\_cluster.eks.certificate\_authority[0].data)**

**token = data.aws\_eks\_cluster\_auth.eks.token**

**}**

**provider "helm" {**

**kubernetes {**

**host = data.aws\_eks\_cluster.eks.endpoint**

**cluster\_ca\_certificate = base64decode(data.aws\_eks\_cluster.eks.certificate\_authority[0].data)**

**token = data.aws\_eks\_cluster\_auth.eks.token**

**}**

**}**

**resource "kubernetes\_namespace" "monitoring" {**

**metadata {**

**name = "monitoring"**

**labels = {**

**"pod-security.kubernetes.io/enforce" = "baseline"**

**"pod-security.kubernetes.io/audit" = "baseline"**

**"pod-security.kubernetes.io/warn" = "baseline"**

**}**

**}**

**}**

**resource "helm\_release" "prometheus\_stack" {**

**name = "kube-prometheus-stack"**

**namespace = kubernetes\_namespace.monitoring.metadata[0].name**

**repository = "https://prometheus-community.github.io/helm-charts"**

**chart = "kube-prometheus-stack"**

**version = "56.6.2"**

**create\_namespace = false**

**values = [**

**file("${path.module}/arc-scrape-config.yaml")**

**]**

**depends\_on = [kubernetes\_namespace.monitoring]**

**}**