Kube2SONiC

Prometheus Setup Guide

Oct 2023

Revision History

Revision No.	Description	Editor	Date
1.0	Document Creation	Saba Akram	Oct 06, 2023

Table of Contents

Introduction

Normal Text will be in Calibri and Font Size 12

K8s Cluster

Node	OS version
Master	Ubuntu 22.0
Worker	SONiC

The kubernetes cluster should be deployed with at least one master and one worker node.

Verify Installation

Check and verify all nodes are in Ready state

```
Python
kubectl get nodes
```

Check and verify all the k8s pods are up and running

```
Python
kubectl get pods -n kube-system
```

Node Exporter Setup

To get the kubernetes node exporter configurations clone the following repo

git clone https://github.com/sabakram/kube2sonic.git

 # Create a Namespace

```
Python
kubectl create namespace monitoring
```

 # Deploy the daemon set

```
Python
kubectl create -f daemonset.yaml
```

 # Verify the daemon set is running

```
Python
kubectl get daemonset -n monitoring
```

 # Create the service.

```
Python
kubectl create -f service.yaml
```

```
Python
kubectl get endpoints -n monitoring
```

Prometheus Setup

To get the kubernetes prometheus configurations clone the following repo

<link to be added >

 # Create a Namespace if doesn't exist already

```
Python
kubectl create namespace monitoring
```

 # Create the cluster role binding

```
Python
kubectl create -f clusterRole.yaml
```

 # Edit the config-map before applying

```
Python
kubectl get ep -n monitoring
```

```
Python
vi config-map.yaml
```

Add the node-exporter endpoint to the manifest file

```
Python
kubectl create -f config-map.yaml
```

 # Create the prometheus Deployment

```
Python
kubectl create -f prometheus-deployment.yaml
```

 # Verify that the deployment is up and running

```
Python
kubectl get deployments  -n monitoring
```

 # Create the prometheus service to expose the endpoints

```
Python
kubectl create -f prometheus-service.yaml -n monitoring
```

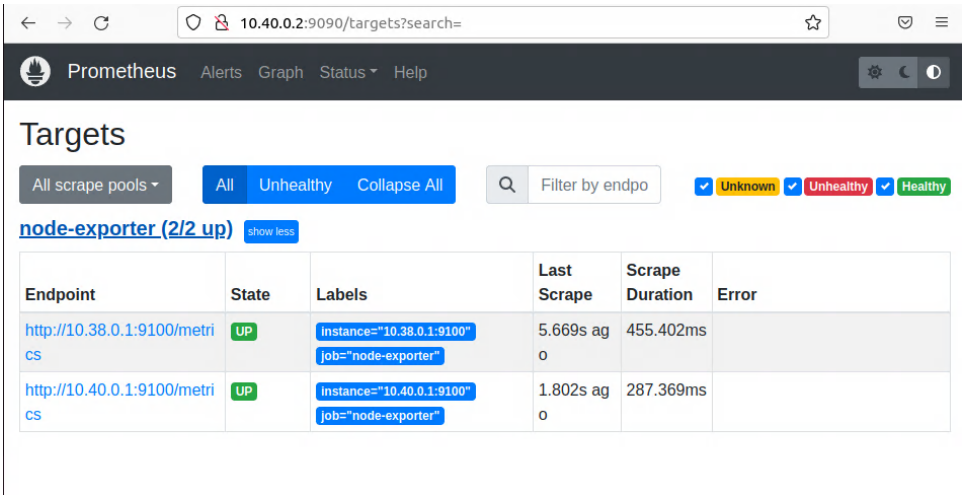
 # Get the endpoints and navigate to web browser

```
Python
kubectl get ep -n monitoring
```

 # Paste the endpoints to the web browser

http://<endpoint>:port

Navigate to the Status -> targets you will be able to view the endpoints connected to prometheus



https://github.com/adamdunstan/sonic-nos-vm-lab/tree/main