DOCKER, KUBERNETES AND HELM INSTALLATION

Cognitive Assistant for Networks (CAN) Release 5.5

JULY 6, 2021

AVANSEUS TECHNOLOGIES PVT. LTD.

REVISION HISTORY

Version	Date	Change description	Created by	Updated by	Reviewed by
V 1.0	July, 2021	Initial Release	Hemanth/Umesh	Sandeep Singh	Chiranjib

avanseus

Table of Contents

1. Introduction

This document is applicable for installation of the Docker, Kubernetes and Helm respectively.

Note:

User can skip the installation steps and directly use the AMI image. AMI image will have Docker, Kubernetes and Helm preinstalled. User can get the AMI image from Devops team.

These are the AMI images for Master Node and Worker Node:

- Avanseus_Kubernetes_Master_Templates
- Avanseus_Kubernetes_Worker_Templates

2. Prerequisites to Install Kubernetes

1 Master Node: (minimum system requirements):

- 8 GB RAM
- 2 vCPU
- OS RHEL Version 8.3 or less

2 or more Worker Nodes: (minimum system requirements)

- 16 GB RAM
- 4 vCPU
- OS RHEL Version 8.3 or less.

3. Pre-Installation Steps on both Master and Worker Nodes

First, login as a 'sudo' user. Execute the following set of commands and then, update your repository.

```
$ sudo su
# yum update -
```

Turn Off the Swap Space

```
# swapoff -a
# vi /etc/fstab
```

Comment out the line in which swap is available and save. Now, the swap has been disabled.

Disable SElinux

```
#vi /etc/sysconfig/selinux
```

And type SELINUX=disabled.

Restart your machine(s) using the below command:

```
#reboot
```

To Allow Ports in Firewall or Disable Firewall

By allowing the below ports or disabling firewall, all containers, network drivers, and pods will communicate across the kubernetes cluster properly.

Run the following command to allow ports in the firewall:

```
#firewall-cmd --permanent --add-port=6443/tcp
#firewall-cmd --permanent --add-port=2379-2380/tcp
#firewall-cmd --permanent --add-port=10250/tcp
#firewall-cmd --permanent --add-port=10251/tcp
#firewall-cmd --permanent --add-port=10252/tcp
#firewall-cmd --permanent --add-port=10255/tcp
#firewall-cmd --reload
```

Run the below command to disable the firewall (This step is not recommended for the production environment):

```
#systemctl stop firewalld
#systemctl disable firewalld
```

Note: The same has to be done for all the ports used while deploying the application.

- For Redhat, please refer:

  https://dzone.com/articles/kubernetes-installation-in-redhat-centos

- For Ubuntu, please refer:

  https://www.edureka.co/blog/install-kubernetes-on-ubuntu

Update IP tables to see bridged traffic

By updating IP Tables, the Port forwarding and Filtering process will work perfectly.

Run the below command to update the IP tables:

```
$modprobe br_netfilter
$echo '1' > /proc/sys/net/bridge/bridge-nf-call-iptables
```

4. Docker Installation

Add the Docker repository before the installation of docker.

```
$sudo yum install -y yum-utils
$sudo yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo
```

This command will download the Docker repository file to /etc/yum.repos.d/docker-ce.repo.
Update RPM index cache.

```
$ sudo yum makecache
```

Install Docker CE by running the command in your terminal.

```
$ sudo yum install docker-ce
```

If you get dependency issues with Podman, allow erasing.

```
$ sudo yum install docker-ce --allowerasing
```

Start and enable Docker Service to start at boot.

```
$ sudo systemctl enable --now docker
```

The docker service status should indicate running.

```
$ systemctl status docker
```

5. Kubernetes Installation

We have to execute all the below steps in both master and worker nodes.

Kubernetes - Run the below command to add Kubernetes repo:

```
#vi /etc/yum.repos.d/kubernetes.repo
```

Paste the below details in the vi editor:

```
[kubernetes]
name=Kubernetes
baseurl=https://packages.cloud.google.com/yum/repos/kubernetes-el7-x86_64
enabled=1
```

```
gpgcheck=1
repo_gpgcheck=1
gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg
       https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
```

To install Kubernetes components, run the following command to install the Kubenetes (kublet kubeadm kubectl). kubectl is optional in the worker node.

```
$yum install kubelet kubeadm kubectl -y
```

To start and enable Kubernetes services, run the below commands to start:

```
$systemctl start kubelet && systemctl enable kubelet
```

You have successfully installed Kubernetes on both the machines now! But still you have to configure the k8 cluster before using it.

6. Helm Installation Guide (Only in master)

Helm provides binary releases for a variety of OSes. These binary versions can be manually downloaded and installed.

1. Download your desired version
2. Unpack it (tar -zxvf helm-v3.0.0-linux-amd64.tar.gz)
3. Find the helm binary in the unpacked directory, and move it to its desired location (mv linux-amd64/helm /usr/local/bin/helm)
4. Test, use the below command:

```
$helm version
```

You have successfully installed Docker, Kubernetes and Helm. Proceed to install k8 cluster. Refer the document "Configuration of Kubernetes Cluster".

A note on Troubleshooting:

1. Internet not working while building docker image?

   https://medium.com/@faithfulanere/solved-docker-build-could-not-resolve-archive-ubuntu-com-apt-get-fails-to-install-anything-9ea4dfdcdcf2

2. Kubeadm join not working?

   https://thenewstack.io/how-to-install-a-kubernetes-cluster-on-red-hat-centos-8/

3. Command to delete k8 cluster:

   rm -rf /etc/cni /etc/kubernetes /var/lib/dockershim /var/lib/etcd /var/lib/kubelet /var/run/kubernetes ~/.kube/*

4. Installing Master node without taint on Ubuntu 18.04:

   https://raaaimund.github.io/tech/2018/10/23/create-single-node-k8s-cluster/