**Setting up Kubernetes**

Kubernetes is a system used to control a wide number of hosts for the purpose of deploying and managing containerized applications. In our specific setup, we will be using Kubernetes in association with Docker to serve as our application container system. The fundamental idea behind Kubernetes, beyond the setup stage, is that, as the user, there is no need to know about which host an application is running on. This serves the purpose of requiring the user to only need to focus on the application and not have to worry about the specifics of the host or management.

The setup of Kubernetes relies on a single host serving as a Master that will serve as the primary controller that will manage the Kubernetes installation. Any additional hosts that are to be used for application deployment are defined as a Node. For the tutorial, the design will be to assign one single instance as the Master and the second as the Node.

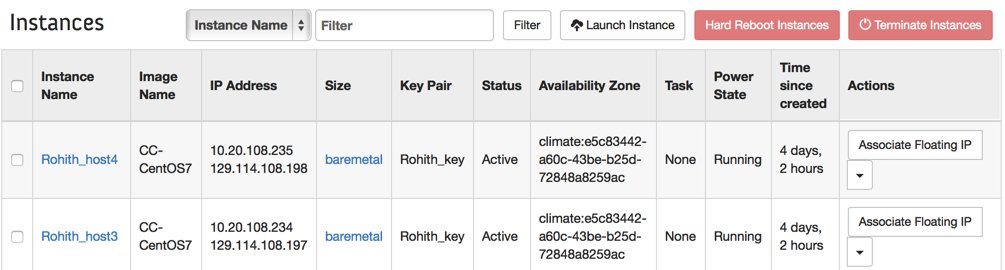
**Prerequisites:**

Following three prerequisites serves for the successful implementation of Kubernetes.

1. A Chameleon user account
2. An active project in the Chameleon Dashboard with appropriate administrative permissions.
3. Two active instances accessible by one another through a network connection.

**STEP-1**

Firstly, it is recommended to list and make a note of the local ip addresses of the two instances we wish to use for our installation.



Now, here in this tutorial host3 and host4 are appointed to be node and master respectively. So the corresponding local area ip address for the node and master would be 10.20.108.234 and 10.20.108.235

Lets go ahead and configure the Kubernetes on the master.

* Connect to the master instance using appropriate SSH aliases like as below:

**$** **ssh cc@<public\_ip\_address\_of\_the\_master\_instance>**

# In our scenario the public ip address is 129.114.108.198

* After logged in, install the **Kubernetes** and one of its dependency **etcd** using the following:

**[cc@rohith-host4 ~]$ sudo yum install kubernetes etcd -y**

This installation will setup all the appropriate configuration files into the **/etc/kubernetes/** directory that need to be modified accordingly as follows:

# To make sure whether the configuration files were setup successfully or not make use of the following command:

**[cc@rohith-host4 ~]$ sudo ls /etc/kubernetes**

Successful installation of kubernetes should list the files as follows using the above command.

****

* Now lets move on modifying the configuration files
* **Note:** For all the following configuration files **<master\_ipaddress>** should be replaced by 10.20.108.235 and **<node\_ipaddress>** is replaced by 10.20.108.234(which are local area ip addresses for master and node in our case)

1. For the file **/etc/kubernetes/apiserver** ensure the following lines are uncommented and edited to match what is show

KUBE\_API\_ADDRESS="--address=0.0.0.0"

KUBE\_API\_PORT="--port=8080"

KUBELET\_PORT="--kubelet\_port=10250"

KUBE\_ETCD\_SERVERS="--etcd\_servers=http://127.0.0.1:2379"

KUBE\_SERVICE\_ADDRESSES="--service-cluster-ip-range=10.254.0.0/16"

KUBE\_ADMISSION\_CONTROL="--admission\_control=NamespaceLifecycle,NamespaceExists,LimitRanger,ServiceAccount,ResourceQuota"

**# sudo vi /etc/kubernetes/apiserver**

In addition, add the following line to the end of the above file **/etc/kubernetes/apiserver**

KUBE\_MASTER="--master=http://<master\_ipaddress>:8080"

1. For the file **/etc/kubernetes/config**, ensure the following lines are uncommented and edited to match what is shown:

**# sudo vi /etc/kubernetes/config**

KUBE\_LOGTOSTDERR="--logtostderr=true"

KUBE\_LOG\_LEVEL="--v=0"

KUBE\_ALLOW\_PRIV="--allow\_privileged=false"

KUBE\_MASTER="--master=http:// <master\_ipaddress>:8080"

1. For the file **/etc/kubernetes/controller-manager**, add the following line to reflect what is shown:

**# sudo vi /etc/kubernetes/controller-manager**

KUBLET\_ADDRESSES="--machines=<node\_ipaddress>"

4. For the file **/etc/kubernetes/kubelet** ensure the following lines are uncommented and it looks as follows

**# sudo vi /etc/kubernetes/kublet**

KUBELET\_ADDRESS="--address=127.0.0.1"

KUBELET\_HOSTNAME="--hostname\_override=127.0.0.1"

KUBELET\_API\_SERVER="--api\_servers=http://127.0.0.1:8080"

5. For the file **/etc/kubernetes/proxy,** edit and uncomment the following line:

**# sudo vi /etc/kubernetes/proxy**

KUBE\_PROXY\_ARGS="--master=http:// <master\_ipaddress>:8080"

Now, we will go through a similar process for the Node instance. Connect to the secondary instance.

Login using the appropriate SSH aliases

* After logged in, install the **Kubernetes** using the following:

**[cc@rohith-host3 ~]$ sudo yum install kubernetes –y**

Once the installation of Kubernetes is done go ahead confirming the all the setup files on **/etc/Kubernetes,** which should like the same on Master instance.

Due to this instance being the node, the configuration files will need to be altered in a slightly different manner than previously with the master.

* **Note:** For all the following configuration files **<master\_ipaddress>** should be replaced by 10.20.108.235 and **<node\_ipaddress>** is replaced by 10.20.108.234(which are local area ip addresses for master and node in our case)

1. For the file **/etc/kubernetes/apiserver** ensure the following lines are uncommented and edited to match what is shown:

**# sudo vi /etc/kubernetes/apiserver**

KUBE\_API\_ADDRESS="--address=127.0.0.1

KUBE\_ETCD\_SERVERS="--etcd\_servers=http://<master\_ipaddress>:4001"

KUBE\_SERVICE\_ADDRESSES="--service-cluster-ip-range=10.254.0.0/16"

KUBE\_ADMISSION\_CONTROL="--admission\_control=NamespaceLifecycle,NamespaceExists,LimitRanger,SecurityContextDeny,ServiceAccount,ResourceQuota"

1. For the file **/etc/kubernetes/config**, add the following lines at the end

**# sudo vi /etc/kubernetes/config**

KUBE\_MASTER="--master=http:// <master\_ipaddress>:8080"

KUBE\_ETCD\_SERVERS="--etcd\_servers=http://<master\_ipaddress>:4001"

3. For the file **/etc/kubernetes/kublet**, ensure that following lines are added and uncommented to reflect what is shown:

**# sudo vi /etc/kubernetes/kublet**

KUBELET\_ADDRESS="--address=0.0.0.0"

KUBELET\_PORT="--port=10250"

KUBELET\_HOSTNAME="--hostname\_override=<node\_ipaddress>"

KUBELET\_API\_SERVER="--api\_servers=http://<master\_ipaddress>:8080"

4. For the file **/etc/kubernetes/proxy,** edit and uncomment the following line:

**# sudo vi /etc/kubernetes/proxy**

KUBE\_PROXY\_ARGS="--master=http:// <master\_ipaddress>:8080"

On the Master (the first host), we need to restart the service in order for the configuration changes to take effect.

[cc@rohith-host4 ~]$ **sudo systemctl restart kube-apiserver**

Additionally, we will enable each service so that is will start at boot for the server.

[cc@rohith-host4 ~]$ **for cmd in restart enable status; do sudo systemctl $cmd etcd kube-apiserver kube-scheduler kube-controller-manager; done**

The Node instance will also need to restart and enable similar services.

[cc@rohith-host3 kubernetes]$ **for cmd in restart enable status; do sudo systemctl $cmd kube-proxy kubelet docker; done**