Lesson 9 Demo 1: Install Kubernetes and Pod Configuration, and Join Another Linux Node to Cluster

This lab has 5 subsections:

* Install Docker
* Install kubernetes
* Initialize kubernetes
* Configure pod network
* Add a Linux node to the kubernetes cluster

**Step 1:** Install Docker

* Run the command below in the terminal to ensure that the apt-get package is updated successfully.

**sudo apt-get update**

* Allow HTTP access to apt-get package repositories and sources by running the command mentioned below:

**sudo apt-get install apt-transport-https -y**

**Please note:** If you get a dpkg **frontend lock** error, please follow the below set of commands to free the process and install JDK again.

**lsof /var/lib/dpkg/lock-frontend  
sudo kill -9**

**sudo rm /var/lib/dpkg/lock-frontend**

**sudo dpkg --configure -a**

* Run the commands mentioned below to install Docker (Run the command even if you have installed docker while performing demos of previous lessons).

**sudo apt-get install docker.io -y**

**sudo apt-get install -y curl apt-transport-https docker.io**

* Enable Docker service to start on system boot and execute the commands mentioned below:

**sudo systemctl start docker**

**sudo systemctl enable docker**

**Step 2:** Install kubernetes

* To download and add the key to allow kubernetes installation, execute the commands mentioned below:

**sudo su**

**curl -s** [**https://packages.cloud.google.com/apt/doc/apt-key.gpg**](https://packages.cloud.google.com/apt/doc/apt-key.gpg) **| apt-key add -**

**sudo echo deb** [**http://apt.kubernetes.io/**](http://apt.kubernetes.io/) **kubernetes-xenial main > /etc/apt/sources.list.d/kubernetes.list**

* Update the apt-get package by executing the command mentioned below:

**sudo apt-get update**

* Install the kubernetes and the tools required to manage it. Run the command mentioned below in the terminal:

**sudo apt-get install -y kubelet kubeadm kubectl**

**Step 3:** Initialize kubernetes

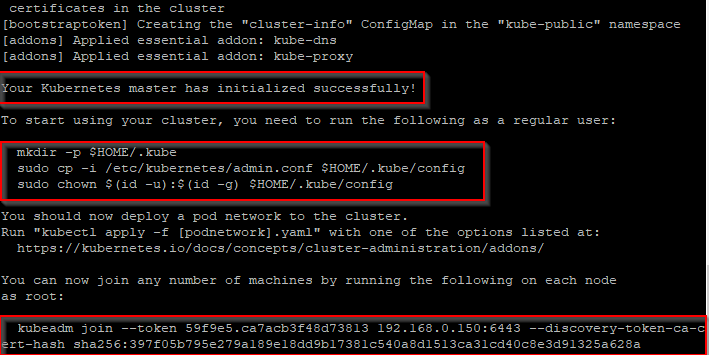
* Update the apt-get package by executing the command mentioned below:

**sudo apt-get update**

* Run the command mentioned below to initialize the kubernetes cluster to run it as master:

**sudo kubeadm init**

* After execution, you must receive the output mentioned in the screenshot:



* To complete the setup, execute the commands below:

**mkdir -p $HOME/.kube**

**sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config**

**sudo chown $(id -u):$(id -g) $HOME/.kube/config**

**Step 4:** Configure pod network

* Apply network configurations to the kubernetes cluster. It allows pod-to-pod communication, and it is a dependency for kube-dns. Execute the commands mentioned below in your terminal:

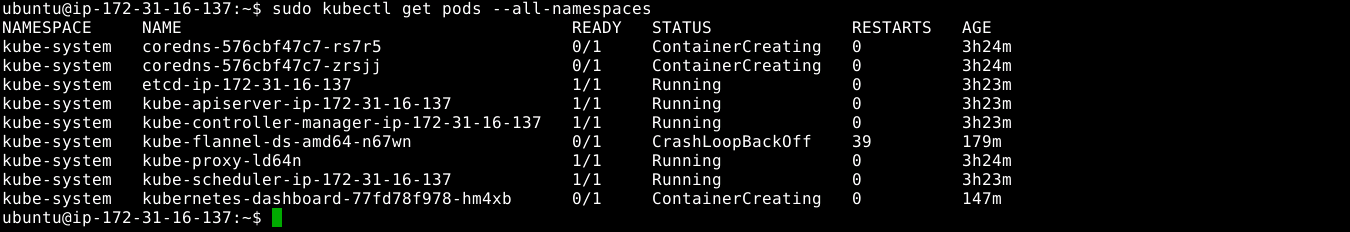
**sudo kubectl apply -f** [**https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml**](https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml)

**sudo kubectl apply -f** [**https://raw.githubusercontent.com/coreos/flannel/master/Documentation/k8s-manifests/kube-flannel-rbac.yml**](https://raw.githubusercontent.com/coreos/flannel/master/Documentation/k8s-manifests/kube-flannel-rbac.yml)

* Execute the command mentioned below to check the result of all pods:

**sudo kubectl get pods --all-namespaces**

* You should ensure that all the pods are running successfully as shown in the screenshot:



**Step 5:** Add a Linux node to the kubernetes cluster.

* Execute the command mentioned below to add a node to the existing cluster:

**sudo kubectl apply -f** [**https://raw.githubusercontent.com/kubernetes/dashboard/v2.0.0-beta1/aio/deploy/recommended.yaml**](https://raw.githubusercontent.com/kubernetes/dashboard/v2.0.0-beta1/aio/deploy/recommended.yaml)

* Execute the command mentioned below to check the result of all pods:

**sudo kubectl get pods --all-namespaces**