

## Kubernetes (v1.14) Installation Steps

### Requirements:

- ✓ 2 ubuntu servers
- ✓ 2 Gb RAM
- ✓ 2 core cpu (v1.14)
- ✓ connectivity between all servers same network
- ✓ unique name of hostname Ipaddress product\_uuid
- ✓ open specify ports (**TCP PORTS**)
  - ✗ 6443\* Kubernetes API server
  - ✗ 2379-2380 Etcd server client API
  - ✗ 10250 Kubelast API
  - ✗ 10251 Kube-scheduler
  - ✗ 10252 Kube-controller-manager
  - ✗ 30000-32767 nodeport services

verify the Mac address and Product\_uuid are unique for every node:

```
Mac Address Verify    - ip a or ifconfig -a
Product_uuid          - sudo cat /sys/class/dmi/id/product_uuid
```

You will install these packages on all of your machines:

- **kubeadm**: the command to bootstrap the cluster.
- **kubelet**: the component that runs on all of the machines in your cluster and does things like starting pods and containers.
- **kubectl**: the command line util to talk to your cluster

### All nodes Kubernetes Installation:

```
# apt update -y

# apt-get update && apt-get install -y apt-transport-https curl

# curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg
apt-key add -

# cat <<EOF >/etc/apt/sources.list.d/kubernetes.list
```

---

```
deb https://apt.kubernetes.io/ kubernetes-xenial main
EOF
# apt update
# apt install kubelet kubeadm kubectl -y
# apt-mark hold kubelet kubeadm kubectl
```

### Docker Installation:

```
# apt-get update && apt-get install apt-transport-https ca-
certificates curl software-properties-common

# curl -fsSL https://download.docker.com/linux/ubuntu/gpg | apt-
key add -

# add-apt-repository "deb [arch=amd64]
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"

# apt-get update && apt-get install docker-ce=18.06.2~ce-3-
0~ubuntu

# cat > /etc/docker/daemon.json <<EOF
{
  "exec-opts": ["native.cgroupdriver=systemd"],
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
EOF

# mkdir -p /etc/systemd/system/docker.service.d
# systemctl daemon-reload
# systemctl restart docker
```

### adding node ( after 23 Hours ) in kubernetes cluster:

#### Master node:

```
# kubeadm token list
21bm6b.27n14jhdgdxurz58 ( expire at 23 hours )
```

```
# kubeadm token create ( if kubeadm token list show some result.  
Don't use kubeadm token token command )
```

```
# openssl x509 -pubkey -in /etc/kubernetes/pki/ca.crt | openssl  
rsa -pubin -outform der 2>/dev/null | openssl dgst -sha256 -hex | sed  
's/^.* //'
```

**a038083c47f2594256d217a175f5c6e11679d99c148da812b5e930b7d06c315d**

adding node command:

```
# kubeadm join 10.150.15.222:6443 --token  
21bm6b.27nl4jhdgdxurz58 --discovery-token-ca-cert-hash  
sha256:a038083c47f2594256d217a175f5c6e11679d99c148da812b5e930b7d06c31  
5d
```

Master node:

```
# kubectl label node worker2 node-role.kubernetes.io/node=
```

if you want delete kubernetes token list

```
# kubeadm token list
```

**21bm6b.27nl4jhdgdxurz58**

```
# kubeadm token delete 21bm6b.27nl4jhdgdxurz58
```

Now view token list

```
# kubeadm token list
```

**Now showing empty list. It does not affect current adding node  
in kubernetes cluster.**

---

Full commands:

```
1 apt-get update && apt-get install apt-transport-https ca-
certificates curl software-properties-common

2 curl -fsSL https://download.docker.com/linux/ubuntu/gpg | apt-
key add -

3 add-apt-repository "deb [arch=amd64]
https://download.docker.com/linux/ubuntu \

4 $(lsb_release -cs) \

5 stable"

6 apt-get update && apt-get install docker-ce=18.06.2~ce~3-
0~ubuntu

7 cat > /etc/docker/daemon.json <<EOF
8 {
9     "exec-opts": ["native.cgroupdriver=systemd"],
10    "log-driver": "json-file",
11    "log-opts": {
12        "max-size": "100m"
13    },
14    "storage-driver": "overlay2"
15 }
16 EOF

17 mkdir -p /etc/systemd/system/docker.service.d
18 systemctl daemon-reload
19 systemctl restart docker

20 apt-get update && apt-get install -y apt-transport-https curl
21 curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg
| apt-key add -

22 cat <<EOF >/etc/apt/sources.list.d/kubernetes.list
```

```
23 deb https://apt.kubernetes.io/ kubernetes-xenial main
24 EOF
25 apt-get update
26 apt-get install -y kubelet kubeadm kubectl
27 apt-mark hold kubelet kubeadm kubectl
28 kubeadm version
29 kubeadm version | awk -F,
30 gawk -F: '{ print $1 }' /etc/passwd
31 sudo apt update -y
32 sudo apt upgrade
33 kubeadm init --pod-network-cidr=192.168.0.0/16
34 echo "10.150.15.223 worker1" >> /etc/hosts
35 echo "10.150.15.222 master" >> /etc/hosts
36 kubectl apply -f https://docs.projectcalico.org/v3.3/getting-
started/kubernetes/installation/hosted/rbac-kdd.yaml
37 kubectl apply -f https://docs.projectcalico.org/v3.3/getting-
started/kubernetes/installation/hosted/kubernetes-datastore/calico-
networking/1.7/calico.yaml
38 kubectl get nodes
39 kubectl label node worker1 node-role.kubernetes.io/node=
40 kubectl get nodes
41 kubapt-get install bash-completion
42 apt-get install bash-completion
43 kubectl completion bash
44 cat ~/.bashrc
45 source /usr/share/bash-completion/bash_completion
```

```
46  kubectl get pods
47  echo 'source <(kubectl completion bash)' >> ~/.bashrc
48  kubectl get pods
```