

# Installation Using Kubeadm for a Single-Node Kubernetes Cluster

Kubeadm is a tool provided by Kubernetes to create and manage clusters. You can use Kubeadm to set up a single-node Kubernetes cluster on a single server (master node) and run your workloads there.

## Step 1: Install Kubernetes

Components You will need to

install `kubeadm`, `kubelet`, and `kubectl`  
on your server.

For Ubuntu/Debian-based systems:

```
# Update the system
sudo apt update && sudo apt upgrade -y

# Install required packages
sudo apt install -y apt-transport-https ca-certificates curl

# Add Kubernetes APT repository
curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -
sudo apt update

# Install Kubernetes components
sudo apt install -y kubelet kubeadm kubectl
```

# Error Debugging

I got this error when I executed the last command.

```
No apt package "kubeadm", but there is a snap with that name.
```

```
Try "snap install kubeadm"
```

```
No apt package "kubectl", but there is a snap with that name.
```

```
Try "snap install kubectl"
```

```
No apt package "kubelet", but there is a snap with that name.
```

```
Try "snap install kubelet"
```

So, I do this recommend.

## 1. Install Snap (if not already installed)

If `snap` is not installed on your system, you can install it with the following commands:

```
sudo apt update  
sudo apt install snapd
```

## 2. Install kubeadm, kubectl, and kubelet using Snap

Once `snapd` is installed, you can use it to install the Kubernetes tools. Use the following commands:

```
sudo snap install kubeadm --classic  
sudo snap install kubectl --classic  
sudo snap install kubelet --classic
```

The `--classic` flag ensures that the snaps are installed in classic mode, which allows them to have access to system resources outside of the snap sandbox.

## 3. Verify the Installation

After installation, you can check if the tools are properly installed by running:

```
kubeadm version  
kubectl version --client  
kubelet --version
```

# Step 2: Disable Swap Memory (Required for Kubernetes)

Kubernetes requires that swap memory is disabled. To disable swap:

```
sudo swapoff -a
```

To permanently disable swap, comment out the swap entry in `/etc/fstab`.

# Step 3: Install `containerd`

```
sudo apt-get install -y containerd
sudo systemctl start containerd
sudo systemctl enable containerd
sudo systemctl status containerd
```

# Step 4: Initialize the Kubernetes Cluster

Run the following command to initialize the Kubernetes cluster:

```
sudo kubeadm init --pod-network-cidr=10.244.0.0/16
```

The `--pod-network-cidr` is the CIDR block for your pods and is required for setting up the network plugin (Flannel, in this case).

Once this command completes successfully, you'll see output that provides a `kubeadm join` command for adding worker nodes (though in this case, we are using a single node, so you won't need to add more nodes).

## Error Debugging

I found this following error when I executed the command:

```
W1201 05:51:43.292057 13491 checks.go:1080] [preflight] WARNING: Couldn't create the interface used for talking to the container runtime: failed to create new CRI runtime service: validate service connection: validate CRI v1 runtime API for endpoint "unix:///var/run/containerd/containerd.sock": rpc error: code = Unavailable desc = connection error: desc = "transport: Error while dialing: dial unix /var/run/containerd/containerd.sock: connect: no such file or directory"
```

```
[WARNING FileExisting-crictl]: crictl not found in system path
```

```
[WARNING Service-Kubelet]: kubelet service is not enabled, please run 'systemctl enable kubelet.service'  
error execution phase preflight: [preflight] Some fatal errors occurred:
```

```
[ERROR NumCPU]: the number of available CPUs 1 is less than the required 2
```

```
[ERROR FileExisting-contrack]: contrack not found in system path
```

```
[ERROR FileContent--proc-sys-net-ipv4-ip_forward]: /proc/sys/net/ipv4/ip_forward contents are not set to 1
```

```
[preflight] If you know what you are doing, you can make a check non-fatal with `--ignore-preflight-errors=...`
```

```
To see the stack trace of this error execute with --v=5 or higher
```

The problem was my node didn't pass the requirement. The requirement for a node is 2 CPUs. So, I upgraded my node to become 2 CPUs.

## Step 4: Ensure that the `kubelet` is Enabled and Running

```
sudo snap start kubelet
```

---

Revision #4

Created 1 December 2024 05:14:56 by Ahmad

Updated 1 December 2024 06:55:33 by Ahmad