Now we are ready to get a real Kubernetes cluster up and running! In this lesson, we will bootstrap the cluster on the Kube master node. Then, we will join each of the two worker nodes to the cluster, forming an actual multi-node Kubernetes cluster.

Here are the commands used in this lesson:

- On the Kube master node, initialize the cluster:

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

That command may take a few minutes to complete.

- When it is done, set up the local kubeconfig:

```
mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

- Verify that the cluster is responsive and that Kubectl is working:

```
kubectl version
```

You should get `Server Version` as well as `Client Version`. It should look something like this:

```
Client Version: version.Info{Major:"1", Minor:"12", GitVersion:"v1.12.2", GitCommit:"17c77c7898218073f14c8d57
Server Version: version.Info{Major:"1", Minor:"12", GitVersion:"v1.12.2", GitCommit:"17c77c7898218073f14c8d57
```

- The `kubeadm init` command should output a `kubeadm join` command containing a token and hash. Copy that command and run it with `sudo` on both worker nodes. It should look something like this:

```
sudo kubeadm join $some_ip:6443 --token $some_token --discovery-token-ca-cert-hash $some_hash
```

- Verify that all nodes have successfully joined the cluster:

```
kubectl get nodes
```

You should see all three of your nodes listed. It should look something like this:

```
NAME                    STATUS     ROLES    AGE     VERSION
wboyd1c.mylabserver.com   NotReady   master   5m17s   v1.12.2
wboyd2c.mylabserver.com   NotReady   <none>   53s     v1.12.2
wboyd3c.mylabserver.com   NotReady   <none>   31s     v1.12.2
```

Note: The nodes are expected to have a STATUS of `NotReady` at this point.