set up a **Kubernetes cluster (1 master + 2 workers)** on **RHEL 8.6** using **Docker as container runtime**.

💻 Lab Setup

- **OS**: RHEL 8.6 (x86_64)
- **Nodes**:
  - 1 Master (control-plane)
  - 2 Worker nodes
- **Runtime**: Docker
- **Kubernetes Install Tool**: kubeadm

🔷 Step 1: Pre-requisites (on all nodes)

```
# Update system
sudo dnf update -y

# Disable SELinux
sudo setenforce 0
sudo sed -i 's/^SELINUX=enforcing$/SELINUX=permissive/' /etc/selinux/config

# Disable swap
sudo swapoff -a
sudo sed -i '/swap/d' /etc/fstab

# Load required kernel modules
cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf
br_netfilter
overlay
EOF

sudo modprobe br_netfilter
sudo modprobe overlay

# Configure sysctl params
cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-iptables  = 1
net.bridge.bridge-nf-call-ip6tables = 1
net.ipv4.ip_forward                 = 1
EOF

sudo sysctl --system
```

🔷 Step 2: Install Docker (on all nodes)

```
# Add Docker repo
sudo dnf config-manager --add-repo=https://download.docker.com/linux/centos/docker-ce.repo

# Install Docker
sudo dnf install -y docker-ce docker-ce-cli containerd.io

# Enable and start Docker
sudo systemctl enable --now docker
```

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

sudo systemctl restart docker
```

🔷 Step 3: Install Kubernetes Components (on all nodes)

```
# Add Kubernetes repo
cat <<EOF | sudo tee /etc/yum.repos.d/kubernetes.repo
[kubernetes]
name=Kubernetes
baseurl=https://pkgs.k8s.io/core:/stable:/v1.30/rpm/
enabled=1
gpgcheck=0
EOF

# Install kubelet, kubeadm, kubectl
sudo dnf install -y kubelet kubeadm kubectl

# Enable kubelet
sudo systemctl enable --now kubelet
```

🔷 Step 4: Initialize Kubernetes Control Plane (on Master only)

```
# Initialize cluster (replace with your master node's IP)
sudo kubeadm init --apiserver-advertise-address=<MASTER_IP> --pod-network-
cidr=192.168.0.0/16
```

👉 Output will give you a kubeadm join command. Copy it.

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

🔷 Step 5: Install Pod Network (on Master only)

For example, **Calico**:

```
kubectl apply -f https://docs.projectcalico.org/manifests/calico.yaml
```

🔷 Step 6: Join Worker Nodes

On each worker node, run the join command copied from Step 4. Example:

```
sudo kubeadm join <MASTER_IP>:6443 --token <TOKEN> --discovery-token-ca-cert-hash
sha256:<HASH>
```

🔷 Step 7: Verify Cluster

On Master:
kubectl get nodes
kubectl get pods -A
You should see **1 master + 2 workers** in Ready state.

✅ Now you have a **Kubernetes cluster (1 Master + 2 Workers) on RHEL 8.6 using Docker**.