Exercise: Install and Configure Firewall for a Kubernetes Cluster (Ubuntu 24.04)

📝 Objective

You will learn to:

- Install and enable the UFW firewall
- Allow only essential ports for Kubernetes
- Block all unnecessary traffic
- Set up rules for a secure single control-plane and worker node setup

---

🛠️ Prerequisites

- Two Ubuntu 24.04 servers:
- `control-plane` (master) node
- `worker` node
- Root/sudo access
- Kubernetes not yet initialized (optional, this can be done before or after firewall setup)

🧩 Part 1: Install and Enable UFW

On **both nodes**:

```
sudo apt update
sudo apt install -y ufw
sudo ufw default deny incoming
sudo ufw default allow outgoing
```

🧩 Part 2: Configure Firewall Rules for Kubernetes

🚩 On the **Control-Plane Node**

These ports must be **allowed**:

Port	Protocol	Purpose
6443	TCP	Kubernetes API server
2379-2380	TCP	etcd (cluster store)
10250	TCP	Kubelet API
10259	TCP	kube-scheduler
10257	TCP	kube-controller-manager

```
# Allow SSH
sudo ufw allow 22/tcp

# Allow Kubernetes Control Plane ports
sudo ufw allow 6443/tcp
sudo ufw allow 2379:2380/tcp
sudo ufw allow 10250/tcp
sudo ufw allow 10257/tcp
sudo ufw allow 10259/tcp
```

> ✅ Optional: Restrict access to only internal IP range (e.g. `192.168.0.0/24`):

```
sudo ufw allow from 192.168.0.0/24 to any port 6443 proto tcp
```

🚩 On the **Worker Node**

These ports must be **allowed**:

Port	Protocol	Purpose
10250	TCP	Kubelet API
30000–32767	TCP	NodePort Services (optional)
6783	TCP/UDP	(if using Weave Net)

```
# Allow SSH
sudo ufw allow 22/tcp

# Allow Kubernetes ports
sudo ufw allow 10250/tcp
sudo ufw allow 30000:32767/tcp
```

> If you're using **container network plugins** (CNI) like Flannel, Calico, or Cilium, open their ports too:

Example (Flannel):

```
sudo ufw allow 8285/udp
sudo ufw allow 8472/udp
```

Example (Calico):

```
sudo ufw allow 179/tcp
sudo ufw allow 4789/udp
```

Example (Cilium with kube-proxy):

```
# VXLAN overlay (default)
sudo ufw allow 8472/udp

# Health checks (agent <-> agent)
sudo ufw allow 4240/tcp

# Cilium agent API (optional debug)
sudo ufw allow 4244/tcp
```

Example (Cilium without kube-proxy):

```
# VXLAN overlay (or Geneve if configured)
sudo ufw allow 8472/udp

# Health checks between nodes
sudo ufw allow 4240/tcp

# Cilium agent API (optional debug)
sudo ufw allow 4244/tcp

# Cilium HostPort/ClusterIP load-balancing (BPF-based)
sudo ufw allow 6081/udp     # if using Geneve encapsulation instead of VXLAN
sudo ufw allow 179/tcp      # if using BGP (optional)
```

🚀 Part 3: Enable UFW

On **both nodes**:

```
sudo ufw enable
```

> Confirm with `y` when asked.

🔍 Part 4: Verify Rules

```
sudo ufw status numbered
```

🔐 Part 5: Optional - Restrict Access Further

- You can restrict API server access to only certain IPs:

```
sudo ufw delete allow 6443/tcp
sudo ufw allow from 192.168.0.10 to any port 6443 proto tcp
```

🧪 Part 6: Testing

- Try accessing blocked ports (e.g. 8080) with `telnet` or `nc`:

```
nc -zv <target-ip> 8080
```

- Try accessing allowed ports (e.g. 6443):

```
nc -zv <control-plane-ip> 6443
```

🧾 Summary of Required Kubernetes Ports

Component	Control Plane	Worker Node
SSH	22	22
Kubernetes API server	6443	✗
etcd	2379–2380	✗
Kubelet API	10250	10250
kube-scheduler	10259	✗
kube-controller-manager	10257	✗
NodePort services	Optional	30000–32767

August 22, 2025