# Install and Configure Iptables

# Iptables

- Iptables is a firewall program for **Linux**.
- It will monitor traffic from and to your server using **tables**.
- These tables contain sets of rules, called chains, that will filter incoming and outgoing data packets.
- When a packet matches a rule, it is given a target, which can be another chain or one of these special values:

➢ **ACCEPT** – will allow the packet to pass through.
➢ **DROP** – will not let the packet pass through.
➢ **RETURN** – stops the packet from traversing through a chain and tell it to go back to the previous chain.

# Contd…

- In this iptables tutorial, we are going to work with one of the default tables, called **filter**. It consists of three chains:

➢ **INPUT** – controls incoming packets to the server.

➢ **FORWARD** – filters incoming packets that will be forwarded somewhere else.

➢ **OUTPUT** – filter packets that are going out from your server.

➢ **Step 1: Installing Iptables:**

sudo apt-get update

sudo apt-get install iptables

**Step 2: Check the status of your current iptables configuration by running:**

sudo iptables -L –v

Here, the **-L** option is used to list all the rules, and **-v** is for showing the info in a more detailed format. Below is the example output:

```
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)

pkts bytes target     prot opt in out   source destination


Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)

pkts bytes target     prot opt in out   source destination


Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)

pkts bytes target     prot opt in out   source destination
```

**Step 3: Defining Chain Rules**

- Insert the **-A** option (**Append**) right after the iptables command, like so: sudo iptables –A

The combined command with other options is:

**-i** (**interface**) — the network interface whose traffic you want to filter, such as eth0, lo, ppp0, etc.

**-p** (**protocol**) — the network protocol where your filtering process takes place.

**-s** (**source**) — the address from which traffic comes from. You can add a hostname or IP address.

**–dport** (**destination port**) — the destination port number of a protocol, such as 22 (SSH), 443 (https), etc.

**-j** (**target**) — the target name (ACCEPT, DROP, RETURN). You need to insert this every time you make a new rule.

**sudo iptables -A <chain> -i <interface> -p <protocol (tcp/udp)> -s <source> --dport <port no.> -j <target>**

**Step 4: Enabling Traffic on Localhost**

```
sudo iptables -A INPUT -i lo -j ACCEPT
```

**Step 5: Enabling Connections on HTTP, SSL, and SSH Port**

```
sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

Check it using iptables –L –v

**Step 6: Filtering Packets Based on Source**

```
sudo iptables -A INPUT -s 192.168.1.3 -j ACCEPT
```

**Step 7: Reject packets from a specific IP address**

sudo iptables -A INPUT -s 192.168.1.3 -j DROP

**Step 8: Drop packets from a range of IP addresses**

sudo iptables -A INPUT -m iprange --src-range 192.168.1.100-192.168.1.200 -j DROP

**Step 9: Dropping all Other Traffic**

sudo iptables -A INPUT -j DROP

**Step 10: Deleting Rules**

sudo iptables -F

**Step 11: To delete a specific rule**

First, to see all the available rules

sudo iptables -L --line-numbers

```
Chain INPUT (policy ACCEPT)

num  target     prot opt source               destination

1    ACCEPT     all -- 192.168.0.4          anywhere
2    ACCEPT     tcp -- anywhere             anywhere tcp dpt:https
3    ACCEPT     tcp -- anywhere             anywhere tcp dpt:http
4    ACCEPT     tcp -- anywhere             anywhere tcp dpt:ssh
```

sudo iptables -D INPUT 3

# THANK YOU