LESSON – 18

FIREWALL

Royal University of Bhutan

LEARNING OUTCOMES

- Firewall basics
- Firewall configuration

Firewall

- What is firewall?
  - is a computer network security system that restricts internet traffic in to, out of, or within a network.
  - decides which network traffic is allowed to pass through and which traffic is deemed dangerous. Essentially, it works by filtering out the good from the bad, or the trusted from the untrusted.
  - ufw / firewalld / iptables

FirewallD Introduction

- is a dynamic daemon to manage firewall with support for networks zones.
- is frontend controller for iptables used to implement persistent network traffic rules.
- is a wrapper for iptables to allow easier management of iptables rules–it is not an iptables replacement.
- a simple, stateful, zone-based firewall.

FirewallD

- Working with FirewallD has two main differences compared to directly controlling iptables:
  - FirewallD uses zones and services instead of chain and rules.
  - It manages rulesets dynamically, allowing updates without breaking existing sessions and connections.

FirewallD

- Working with FirewallD has two main differences compared to directly controlling iptables:
  - FirewallD uses zones and services instead of chain and rules.
  - It manages rulesets dynamically, allowing updates without breaking existing sessions and connections.

Installing and Managing FirewallD

- FirewallD has to be installated on Ubuntu:

To install the service and enable FirewallD

```
# sudo apt update
# sudo apt install firewalld
```

By default, the service should be started, if not running, start and enable it to start on boot:

```
#sudo systemctl enable firewalld
#sudo systemctl start firewalld
```

- To stop and disable it
  `#systemctl stop firewalld`
- `#systemctl disable firewalld`

- To reload a firewalld configuration:
- `#firewall-cmd --reload`

Configuring FirewallD

- Is configured with XML files.

- firewall-cmd should be used in very specific configuration.

- Configuration files are located in two directories
  - /usr/lib/firewalld : It holds default configurations. Avoid updating them.

  -/etc/firewalld : It holds system configuration files. These files will overwrite a default configuration

Configuration Sets

- Firewalld uses two configuration sets:

  1. Runtime: Changes are not retained on reboot or upon restarting firewalld.
  2. Permanent: Changes are not applied to a running system.

- By default, firewall-cmd commands apply to runtime configuration but using the --permanent flag will establish a persistent configuration.

Configuration Sets

- Add the rule to both the permanent and runtime sets:

```
#firewall-cmd --zone=public --add-service=http --permanent
#firewall-cmd --zone =public --add-service = http
```

- Add the rule to the permanent set and reload FirewallD

```
#firewall-cmd --zone=public --add-service=http --permanent
#firewall-cmd --reload
```

Available Firewalld Zones

1. **Drop Zone:**
   It will deny the incoming network connections are rejected with an icmp-host-prohibited messages.

2. **Block Zone:**
   Any incoming network packets are dropped, there is no reply. Only outgoing network connections are possible.

3. **Public Zone:**
   For use in public areas. You do not trust the other computers on networks to not harm your computer. Only selected incoming connections are accepted.

Available Firewalld Zones

4. Drop Zone:

Any incoming External Zone: This zone will act as router options with masquerading is enabled other connections will be dropped and will not accept, only specified connection will be allowed.

5. DMZ Zone:

If we need to allow access to some of the services to public, you can define in DMZ zone. This too have the feature of only selected incoming connections are accepted.

6. Work Zone:

Can define only internal networks i.e. private networks traffic are allowed..

Available Firewalld Zones

7. Home Zone:

we can use this zone to trust the other computers on networks to not harm your computer as every zone. This too allow only the selected incoming connections.

8. Internal Zone:

This one is similar to work zone with selected allowed connections.

9. Trusted Zone:

If we set the trusted zone all the traffic are accepted.

Managing firewalld

1. To view the default zone:

```
#firewall-cmd --get-default-zone
```

2. To change the default zone:

```
#firewall-cmd --set-default-zone=internal
```

3. To see the zones used by your network interface(s)

```
#firewall-cmd --get-active-zones
```

4. To get all configurations for a specific zone:

```
#firewall-cmd --zone=public --list-all
```

Managing firewalld

5. To get all configurations for all zones

```
#firewall-cmd –list-all-zones
```

6. Changing the Zone of an Interface

```
#firewall-cmd --zone=home --change-interface=eth0
```

7. Opening the port for your zone

```
#firewall-cmd –zone=public --add-port=5000/tcp
```

to check this:

```
#firewall-cmd --zone=public --list-ports
```

Managing firewalld

It is also possible to specify a sequential range of ports by separating the beginning and ending port in the range with a dash. For instance, if our application uses UDP ports 4990 to 4999, we could open these up on "public“ by typing:

```
#firewall-cmd –zone=public --add-port=4990-4999/udp
```

Creating Your Own Zones

```
#firewall-cmd --permanent --new-zone=publicweb

#firewall-cmd --permanent --new-zone=privateDNS
```

Reload the firewall and verify:

```
#firewall-cmd –permanent --get-zones
```

To add services:

```
#firewall-cmd --zone=publicweb --add-service=ssh
```

Working with services

FirewallD can allow traffic based on predefined rules for specific network services.

Location:

Default: /usr/lib/firewalld/services

user-created: /etc/firewalld/services

Working with services

To view the default available services:

```
#firewall-cmd --get-services
```

Example:

To enable or disable the HTTP service:

```
#firewall-cmd --zone=public --add-service=http –permanent

#firewall-cmd --zone=public --remove-service=http --permanent
```

Constructing a Ruleset with FirewallD

- As an example, here is how we would use FirewallD to assign basic rules to our Linode if we were running a web server:

1. Assign the dmz zone as the default zone to eth0.

```
#firewall-cmd --set-default-zone = dmz
#firewall-cmd --set-zone=dmz --add-interface=eth0
```

2. Add permanent service rules for HTTP and HTTPS to the dmz zone:

```
#firewall-cmd --zone=dmz --add-service=http --permanent
```

Constructing a Ruleset with FirewallD

3. Reload FirewallD so the rules take effect immediately:

```
#firewall-cmd --reload
#firewall-cmd --zone=dmz --list-all
```

- This tells us that the dmz zone is our default which applies to the eth0 interface, all network sources and ports.

- Incoming HTTP (port 80), HTTPS (port 443) and SSH (port 22) traffic is allowed.

SELF Exercises

Connect two PCS on the same network and do the following:

- *Change the SSH port to 3333*
- *Create a new zone. Name the zone as **itm***
  - *Create a rule to do the following:*
    - *Block ICMP*
    - *Allow HTTP and HTTP services*
    - *Allow custom port SSH*

SUMMARY

- In this lesson, you have learnt that:

- Installing and configuring firewalls
- Creating zones in firewalls

IP Tables

- Example of IP tables rules

```
# Log dropped packets
sudo iptables -A INPUT -j LOG --log-prefix "IPTABLES-DROP: " --log-level 4
sudo iptables -A INPUT -j DROP

# Log specific traffic
sudo iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix "SSH: "
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT

# Limit logging to avoid flooding
sudo iptables -A INPUT -m limit --limit 2/min -j LOG --log-prefix "IPTABLES: "
```