

Firewall Configuration and Testing

Introduction

A firewall is a network security system that monitors and controls incoming and outgoing network traffic based on predefined security rules. It acts as a protective barrier between trusted internal networks and untrusted external networks such as the internet. Firewalls help prevent unauthorized access, cyber attacks, and data breaches.

Tools Used

- UFW (Uncomplicated Firewall) – Linux (Ubuntu / Kali)

Firewall Concepts

- **Inbound Traffic:** Data entering the system
- **Outbound Traffic:** Data leaving the system
- **Allow Rule:** Permits specific network traffic
- **Deny Rule:** Blocks unauthorized traffic
- **Port:** Communication endpoint (e.g., 22 for SSH, 80 for HTTP)
- **IP Address:** Unique identifier of a device on a network

Firewall Configuration Steps

Enable Firewall (UFW)

```
sudo ufw enable  
sudo ufw status
```

```
(kali㉿kali)-[~]  
└─$ sudo ufw enable  
[sudo] password for kali:  
Firewall is active and enabled on system startup  
  
(kali㉿kali)-[~]  
└─$ sudo ufw status  
Status: active
```

Figure 1: Enabling UFW and Checking Firewall Status

Configure Firewall Rules

Allow SSH (Port 22):

```
sudo ufw allow 22
```

Allow HTTP (Port 80):

```
sudo ufw allow 80
```

Deny FTP (Port 21):

```
sudo ufw deny 21
```

```
(kali㉿kali)-[~]
└─$ sudo ufw allow 22
[sudo] password for kali:
Rule added
Rule added (v6)

(kali㉿kali)-[~]
└─$ sudo ufw allow 80
Rule added
Rule added (v6)

(kali㉿kali)-[~]
└─$ sudo ufw deny 21
Rule added
Rule added (v6)
```

Figure 2: Configuring Allow and Deny Firewall Rules

Allow or Deny Specific Ports

Allow a range of ports:

```
sudo ufw allow 1000:2000/tcp
```

Block Telnet (Port 23):

```
sudo ufw deny 23
```

```
(kali㉿kali)-[~]  
$ sudo ufw allow 1000:2000/tcp  
Rule added  
Rule added (v6)  
  
(kali㉿kali)-[~]  
$ sudo ufw deny 23  
Rule added  
Rule added (v6)
```

Figure 3: Allowing Port Range and Blocking Telnet Port

Testing Connectivity

Test network connectivity using ping:

```
ping google.com
```

Scan open ports using Nmap:

```
nmap localhost
```

Allowed ports appear as *open* and blocked ports appear as *filtered* or *closed*.

```

(kali@kali)-[~]
$ ping google.com
PING google.com (142.250.70.46) 56(84) bytes of data.
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=1 ttl=128 time=39.9 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=2 ttl=128 time=40.5 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=3 ttl=128 time=39.4 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=4 ttl=128 time=41.3 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=5 ttl=128 time=39.1 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=6 ttl=128 time=40.8 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=7 ttl=128 time=39.0 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=8 ttl=128 time=40.3 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=9 ttl=128 time=48.1 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=10 ttl=128 time=39.4 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=11 ttl=128 time=41.5 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=12 ttl=128 time=36.7 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=13 ttl=128 time=39.4 ms
64 bytes from pnbomb-aa-in-f14.1e100.net (142.250.70.46): icmp_seq=14 ttl=128 time=41.7 ms
^C
-- google.com ping statistics --
14 packets transmitted, 14 received, 0% packet loss, time 13010ms
rtt min/avg/max/mdev = 36.727/40.510/48.075/2.433 ms

(kali@kali)-[~]
$ nmap localhost
Starting Nmap 7.95 ( https://nmap.org ) at 2026-02-01 01:21 EST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000050s latency).
Other addresses for localhost (not scanned): ::1
All 1000 scanned ports on localhost (127.0.0.1) are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Nmap done: 1 IP address (1 host up) scanned in 0.28 seconds

```

Figure 4: Testing Firewall Rules Using Ping and Nmap

Firewall Logging

Enable logging:

```
sudo ufw logging on
```

```

(kali@kali)-[~]
$ sudo ufw logging on
Logging enabled

```

Figure 5: Firewall Log Monitoring

Blocking Malicious IP Addresses

Block a specific IP address:

```
sudo ufw deny from 192.168.1.100
```

Block an entire subnet:

```
sudo ufw deny from 192.168.1.0/24
```

```

(kali@kali)-[~]
$ sudo ufw deny from 192.168.1.100
Rule added

(kali@kali)-[~]
$ sudo ufw deny from 192.168.1.0/24
Rule added

```

Figure 6: Blocking Malicious IP Address and Subnet

Documenting Firewall Rules

List all firewall rules:

```
sudo ufw status numbered
```

```

(kali@kali)-[~]
$ sudo ufw status numbered
Status: active

      To Action From
      --
[ 1] 22 ALLOW IN Anywhere
[ 2] 80 ALLOW IN Anywhere
[ 3] 21 DENY IN Anywhere
[ 4] 1000:2000/tcp ALLOW IN Anywhere
[ 5] 23 DENY IN Anywhere
[ 6] Anywhere DENY IN 192.168.1.100
[ 7] Anywhere DENY IN 192.168.1.0/24
[ 8] 22 (v6) ALLOW IN Anywhere (v6)
[ 9] 80 (v6) ALLOW IN Anywhere (v6)
[10] 21 (v6) DENY IN Anywhere (v6)
[11] 1000:2000/tcp (v6) ALLOW IN Anywhere (v6)
[12] 23 (v6) DENY IN Anywhere (v6)

```

Figure 7: Documenting Firewall Rules

Firewall documentation should include allowed ports, blocked ports, blocked IP addresses, and the purpose of each rule.

Impact of Firewall Rules

Rule	Impact
Allow SSH (22)	Enables secure remote administration
Block FTP (21)	Prevents insecure file transfers
Block malicious IP	Protects against attacks and brute-force attempts
Enable logging	Helps monitor and analyze security events

Windows Firewall Testing (Brief)

Windows Defender Firewall allows users to configure inbound and outbound rules through the Advanced Security settings. Specific ports and applications can be allowed or blocked, and connectivity can be tested using Command Prompt or web browsers.

Conclusion

Firewall configuration and testing are essential security practices for protecting systems and networks. Properly configured firewall rules, regular testing, log monitoring, and documentation significantly improve an organization's security posture.