Firewall/IPTables - Lab Guide

This lab was developed by Sparta Global for Cybersecurity courses. It was built based on the original lab that was developed for the Labtainer framework by the Naval Postgraduate School, Center for Cybersecurity and Cyber Operations under National Science Foundation Award No. 1438893. This work is in the public domain, and cannot be copyrighted.

1 Overview

Iptables is a command line software-based firewall in Linux. It uses policy chains to allow and to block traffic.

IPTables is used as a Firewall and can perform NAT and PAT operations.

In this lab, we focus on IPTables configuration to allow and deny access from/to IP addresses and/or services.

2 Lab Environment

This lab runs in the Labtainer framework, available at http://my.nps.edu/web/c3o/labtainers. That site includes links to a pre-built virtual machine that has Labtainers installed, however Labtainers can be run on any Linux host that supports Docker containers.

From your labtainer-student (/labtainer/labtainer-student) directory start the lab using:

```
labtainer sparta-firewall
```

A link to this lab guide will be displayed.

3 Network Configuration

IP addresses and routing are configured on all devices.

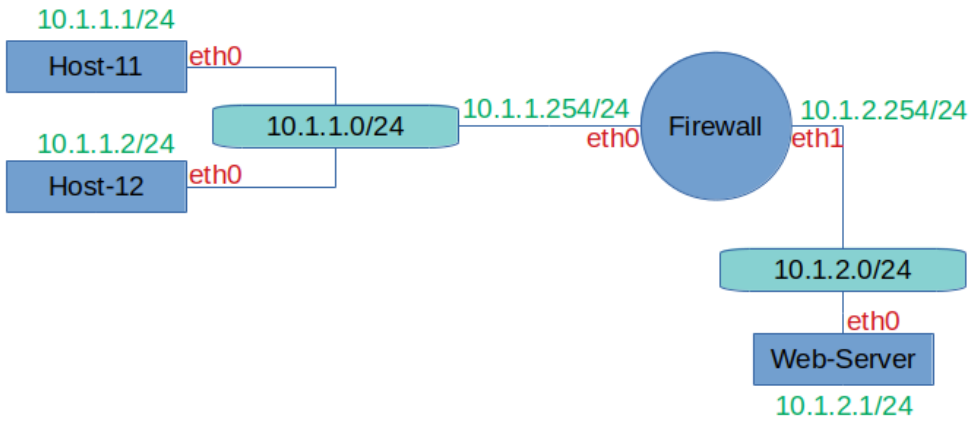

Figure 1: Network topology for routing-basics lab

4 Credentials

- **Host-11**:
  - **Username:** user-11
  - **Password:** user-11
- **Host-12**:
  - **Username:** user-12
  - **Password:** user-12

  o
- **Web-Server**:
  - **Username:** web-admin
  - **Password:** web-admin
- **Firewall**:
  - **Username:** admin
  - **Password:** admin

5 Lab Tasks

5.1 Testing the Initial Configuration

Lets check what we can/can't do in this network.

- On Host-11 (Ping Host-11 -> Web-Server)

  ```
  ping 10.1.2.1
  ```

  What is the result ?
- On Host-11 (Send HTTP request Host-11 -> Web-Server to get a web page)

  ```
  wget http://10.1.2.1/index.html
  ```

  What is the result ?
- On Host-11 (SSH Access Host-11 -> Router1 to access the server itself)

  ```
  ssh web-admin@10.1.2.1
  ```

Note: You will be prompted to accept adding the SSH fingerprint
The authenticity of host '10.1.2.1 (10.1.2.1)' can't be established. ECDSA key fingerprint is SHA256:Z2/CHKW/UJw7
Are you sure you want to continue connecting (yes/no)? You should type 'yes' and hit 'Enter'. In a real environment, you need to check whether the fingerprint is correct or not in order to prevent some types of attacks against SSH protocol (this subject will be discussed later in Cybersecurity module).

Note: Be patient, it takes long time to start.

What is the result ?

- On Host-12 (Ping Host-12 -> Router1)

```
ping 10.1.1.254
```

What is the result ?

- On Host-12 (Ping Host-12 -> Web-Server)

```
ping 10.1.2.1
```

What is the result ?

- On Host-12 (Send HTTP request Host-12 -> Web-Server to get a web page)

```
wget http://10.1.2.1/index.html
```

What is the result ?

- On Host-12 (SSH Access Host-12 -> Router1 to access the server itself)

```
ssh web-admin@10.1.2.1
```

Note: Accept the SSH fingerprint

What is the result ?

- On Host-12 (Ping Host-12 -> Router1)

```
ping 10.1.1.254
```

What is the result ?

5.2 Configuring the Firewall

Now, lets configure the firewall to allow only:

- Host-11 -> Web-Server: SSH Protocol (TCP 22)
- Host-12 -> Web-Server: HTTP Protocol (TCP 80)

The default rules in iptables allow all type of traffic to pass.

- We will start by deleting (flushing) all rules.

```
sudo iptables -F
sudo iptables -t nat -F
sudo iptables -X
```

- Drop all packets by default

```
sudo iptables -P FORWARD DROP
sudo iptables -P INPUT   DROP
sudo iptables -P OUTPUT  DROP
```

- Allow (Operation: ACCEPT) forwarding traffic on firewall that matches the rules:
  - Source IP: 10.1.1.1
  - Destination IP: 10.1.2.1
  - Protocol: TCP
  - Port: 22 (SSH)

```
sudo iptables -A FORWARD -p tcp --dport 22 -s 10.1.1.1 -d 10.1.2.1 -j ACCEPT
```

- On Host-11 (SSH Access Host-11 -> Router1 to access the server itself)

```
ssh web-admin@10.1.2.1
```

  What is the result ? What is the reason in your opinion ?

- Allow (Operation: ACCEPT) forwarding traffic on firewall that matches the rules:
  - Source IP: 10.1.1.2
  - Destination IP: 10.1.2.1
  - Protocol: TCP
  - Port: 80 (HTTP)

```
sudo iptables -A FORWARD -p tcp --dport 80 -s 10.1.1.2 -d 10.1.2.1 -j ACCEPT
```

- Test the connectivity: On Host-12 (Send HTTP request Host-12 -> Web-Server to get a web page)

```
wget http://10.1.2.1/index.html
```

  What is the result ? What is the reason in your opinion ?

- allow the traffic that it is a part of a connection to pass through the firewall

```
sudo iptables  -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
```

Now, refer to Section 5.1 and redo all the tests again. Report to your instructor the tests which they worked and the ones that didn't work.

6 Submission

After finishing the lab, go to the terminal on your Linux system that was used to start the lab and type:

```
stoplab sparta-firewall
```

When you stop the lab, the system will display a path to the zipped lab results on your Linux system. Provide that file to your instructor, e.g., via the Sakai site.