IPTABLES

1 Overview

This Labtainer exercise illustrates the use of iptables on a firewall to limit network access to a server from a client, as illustrated in figure 1

When properly configured, the firewall will only allow selected traffic from the client to the server.

1.1 Background

Limiting the types of network traffic sent to a server can help to protect the server from unauthorized access. For example, if the server contains an unsecured service available through its network interface, exploitation of that service is more difficult if something blocks traffic destined for that service.

A variety of different techniques and products exist for the purpose of limiting IP network traffic between computers. In this lab, you will limit IP traffic through use of Linux iptables. The student is expected to have separately learned about the use of iptables to selectively block network traffic. The firewall component includes an example firewall setting script that you can reference. The manpage for iptables can be viewed on the firewall component using:

```
man iptables
man iptables-extensions
```

Students are expected to have a basic familiarity with the Linux command line, and the ability to edit files and run simple shell scripts. Some experience with Wireshark is presumed, e.g., performance of the wireshark-intro lab.

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 directory start the lab using:

```
labtainer iptables2
```

A link to this lab manual will be displayed.

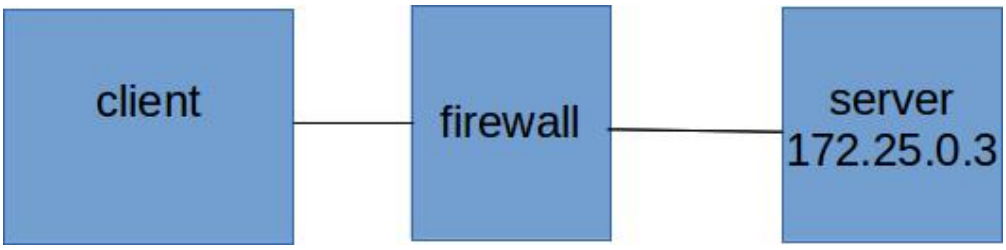

Figure 1: Network topology for the iptables lab

3 Lab Tasks

3.1 Explore

The Wireshark utility is installed on the firewall. Use it to view network traffic through the firewall, and to debug your firewall rules. Start it from the firewall terminal:

```
wireshark &
```

Then select the eth0 interface.

On the client terminal use the nmap utility to list (some of the) open ports on the server:

```
nmap server
```

Use wget to confirm that the server response to HTTP requests:

```
wget server &
```

Confirm an ssh service if offered – you need not login when prompted, just use `ctrl C` to exit once you get a response from the server.

```
ssh server
```

Finally, confirm that telnet is offered (again, no need to login):

```
telnet server
```

Observe the traffic in wireshark, making note the source IP addresses and the destination ports used by the clients when connecting to the server

3.2 Use iptables to limit traffic

The iptables utility is installed on the "firewall" component. Use it to prevent the firewall from forwarding any traffic to the server other than SSH and HTTP.

You may reference and experiment with the example firewall script that is on the firewall component in the home directory. To run the `example_fw.sh` script, use:

```
sudo ./example_fw.sh
```

View the content of the script to understand what it does. Consider putting your iptables commands in a script so it is easy to test and reconfigure the iptables if you restart the lab.

Note the last line in the `example_fw.sh` script directs iptables to log dropped packets. You can view these from one of the firewall terminal tabs via:

```
tail -f /var/log/iptables.log
```

After modifying your iptables configuration, use the applications on the client to demonstrate that the firewall only allows the desired traffic. Watch the traffic in wireshark to see that the TCP handshake fails when attempting to connect to filtered ports.

Use nmap to confirm the proper configuration:

```
nmap server
```

3.3 Open new service port

The client computer includes a `wizbang` program that you must now allow to send traffic to the server. Run the program from the client, and observe which port it attempts to use within wireshark:

```
./wizbang
```

Then alter your iptables to allow this service. After adjusting your iptables, confirm that you can run the `wizbang` program successfully. Also, again use nmap to confirm the proper configuration

```
nmap server
```

4 Submission

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

```
stoplab
```

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.