ETHICAL HACKING
LAB SERIES

Lab 13: Testing Firewall Rules with Firewalking

Material in this Lab Aligns to the Following Certification Domains/Objectives
Certified Ethical Hacking (CEH) Domain
16: Evading IDS, Firewalls and Honeypots

Document Version: 2016-03-09

Contents

Introduction

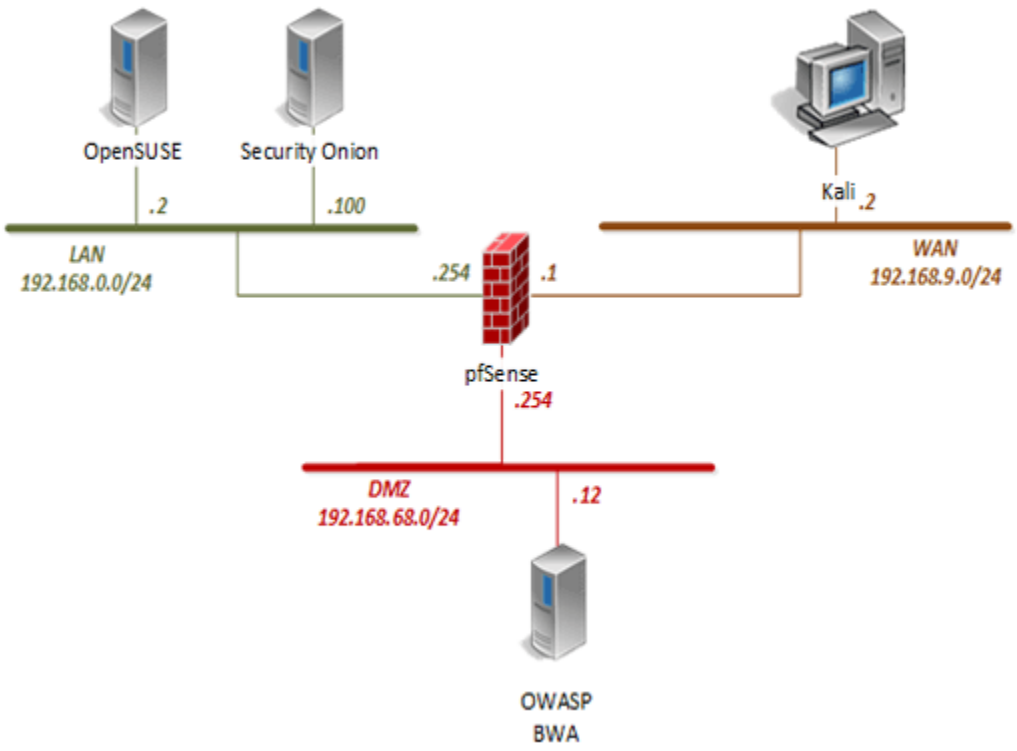
Firewall rules or ACLs are fundamental in controlling ingress and egress traffic in a network. In this lab, we use a method of testing whether those rules are properly configured.

Objective

In this lab, you will be conducting ethical hacking practices using various tools. You will be performing the following tasks:

1. Navigating to the pfSense Dashboard
2. Scan for Firewall Rules with Firewalk
3. Configuring ACL Rules
4. Test Configured Firewall Rules with Firewalk

Pod Topology



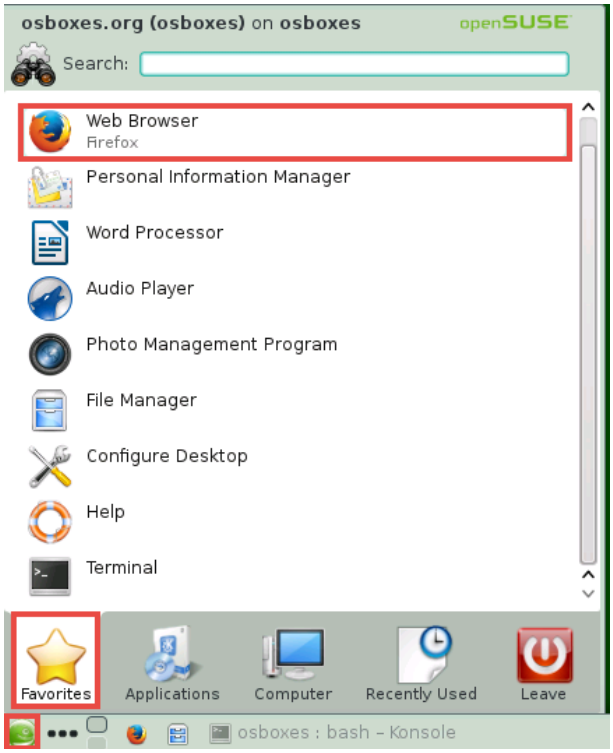
Lab Settings

The information in the table below will be needed in order to complete the lab. The task sections below provide details on the use of this information.

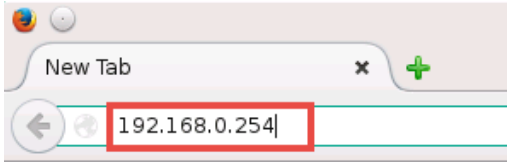
Virtual Machine	IP Address	Account (if needed)	Password (if needed)
Kali Linux	192.168.9.2	root	toor
pfSense	192.168.0.254	admin	pfsense
OWASP Broken Web App	192.168.68.12	root	owaspbwa
OpenSUSE	192.168.0.2	osboxes	osboxes.org
Security Onion	n/a	ndg	password123

1 Navigating to the pfSense Dashboard

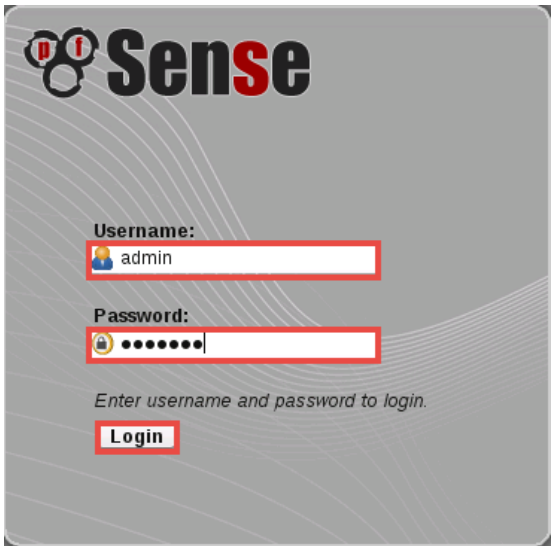
1. Click on the **OpenSUSE** graphic on the *topology page*.
2. Enter **`osboxes`** as the *username*. Click **Next**.
3. Enter **`osboxes.org`** as the *password*. Press **Enter**.
4. Open the *Firefox* browser by clicking on the **Application Launcher** and then clicking on the **Firefox Web Browser** icon.



5. While viewing the Firefox browser, type **`192.168.0.254`** into the address field and press the **Enter** key.



6. Notice the *pfSense* login page, login with **`admin`** as the *username* and **`pfsense`** as the *password*. Click **Login**.



7. Once logged into the *pfSense Dashboard*, navigate to **Firewall > Rules** using the top panel.

8. Notice when observing the *WAN* rules that all protocols are allowed to pass through.

2 Scan for Firewall Rules with Firewalk

1. Navigate to the *topology* page and click on the **Kali** VM icon.
2. Click anywhere within the *Kali* console window and press **Enter** to display the login prompt.
3. Enter `root` as the *username*. Click **Next**.
4. Enter `toor` as the *password*. Click **Sign In**.
5. Open the *Terminal* by clicking on the **Terminal** icon located on the left panel.

6. In the new *Terminal* window, type the command below to get familiarized with the *firewalk* command options available. Press **Enter**.

```
firewalk
```

7. *Firewalk* has the ability to use a technique similar to traceroute to try to determine *ACL* rules on the firewall. First, try an *Nmap* scan against the firewall. Type the command below followed by pressing the **Enter** key.

```
nmap 192.168.9.1
```

```
root@Kali2:~# nmap 192.168.9.1

Starting Nmap 6.49BETA5 ( https://nmap.org ) at 2015-12-22 11:04 CST
Nmap scan report for 192.168.9.1
Host is up (0.00023s latency).
Not shown: 998 filtered ports
PORT   STATE SERVICE
53/tcp open  domain
80/tcp open  http
MAC Address: 00:50:56:9A:AD:1D (VMware)

Nmap done: 1 IP address (1 host up) scanned in 17.55 seconds
```


8. Notice there are two open ports reported by *Nmap*: ports 53 and 80. Next, try to determine the *ACL* rules. Enter the command below using *firewalk* to test if there is a rule for port 23.

```
firewalk -n -p TCP -S 23 -d 23 192.168.9.1 192.168.68.12
```

Command Break-Down:
-n: don't resolve name
-p: protocol
-S: scan ports
-d: destination

```
root@Kali2:~# firewalk -n -p TCP -S 23 -d 23 192.168.9.1 192.168.68.12
Firewalk 5.0 [gateway ACL scanner]
Firewalk state initialization completed successfully.
TCP-based scan.
Ramping phase source port: 53, destination port: 23
Hotfoot through 192.168.9.1 using 192.168.68.12 as a metric.
Ramping Phase:
 1 (TTL  1): expired [192.168.9.1]
Binding host reached.
Scan bound at 2 hops.
Scanning Phase:
port  23: A! open (port not listen) [192.168.68.12]

Scan completed successfully.

Total packets sent:               2
Total packet errors:              0
Total packets caught              2
Total packets caught of interest  2
Total ports scanned               1
Total ports open:                 1
Total ports unknown:              0
```

Notice port 23 reports *ACL* is open but not listening. This means that even though the port is closed, the rule is not in place for port 23. Looking back at the rules on the firewall, they are set to any protocol and any port concluding that there are no rules.

9. Enter the same *firewalk* command but this time check the *ACL* rules, if any, for port 25.

```
firewalk -n -p TCP -S 25 -d 25 192.168.9.1 192.168.68.12
```

Notice a similar output is given when comparing with the port 23 *firewalk* results.

10. Enter the command below to use *firewalk* against a range of ports between 53-80.

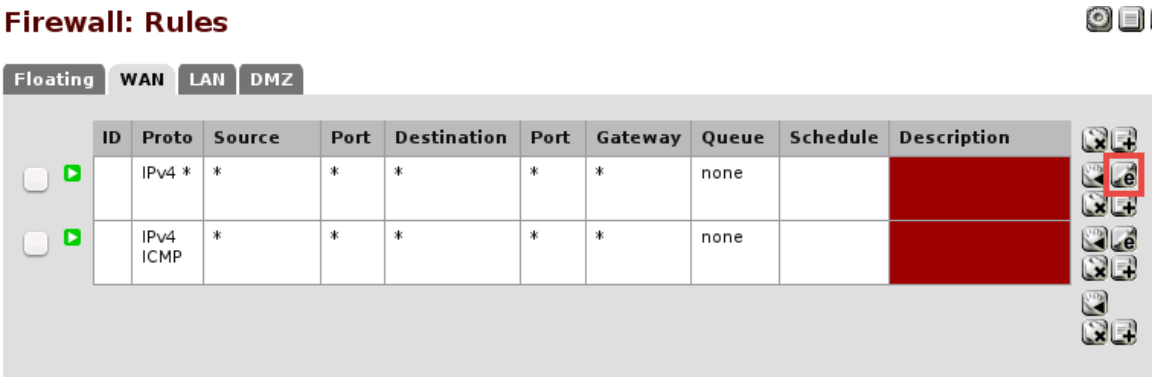
```
firewalk -S 53-80 -n -p TCP 192.168.9.1 192.168.68.12
```

```
root@Kali2:~# firewalk -n -p TCP -S 53-80 192.168.9.1 192.168.68.12
Firewalk 5.0 [gateway ACL scanner]
Firewalk state initialization completed successfully.
TCP-based scan.
Ramping phase source port: 53, destination port: 33434
Hotfoot through 192.168.9.1 using 192.168.68.12 as a metric.
Ramping Phase:
 1 (TTL  1): expired [192.168.9.1]
Binding host reached.
Scan bound at 2 hops.
Scanning Phase:
port  53: A! open (port not listen) [192.168.68.12]
port  54: A! open (port not listen) [192.168.68.12]
port  55: A! open (port not listen) [192.168.68.12]
port  56: A! open (port not listen) [192.168.68.12]
port  57: A! open (port not listen) [192.168.68.12]
port  58: A! open (port not listen) [192.168.68.12]
port  59: A! open (port not listen) [192.168.68.12]
port  60: A! open (port not listen) [192.168.68.12]
port  61: A! open (port not listen) [192.168.68.12]
port  62: A! open (port not listen) [192.168.68.12]
port  63: A! open (port not listen) [192.168.68.12]
port  64: A! open (port not listen) [192.168.68.12]
port  65: A! open (port not listen) [192.168.68.12]
port  66: A! open (port not listen) [192.168.68.12]
port  67: A! open (port not listen) [192.168.68.12]
port  68: A! open (port not listen) [192.168.68.12]
port  69: A! open (port not listen) [192.168.68.12]
port  70: A! open (port not listen) [192.168.68.12]
port  71: A! open (port not listen) [192.168.68.12]
port  72: A! open (port not listen) [192.168.68.12]
port  73: A! open (port not listen) [192.168.68.12]
port  74: A! open (port not listen) [192.168.68.12]
port  75: A! open (port not listen) [192.168.68.12]
port  76: A! open (port not listen) [192.168.68.12]
port  77: A! open (port not listen) [192.168.68.12]
port  78: A! open (port not listen) [192.168.68.12]
port  79: A! open (port not listen) [192.168.68.12]
port  80: A! open (port listen) [192.168.68.12]

Scan completed successfully.

Total packets sent:                29
Total packet errors:               0
Total packets caught               29
Total packets caught of interest   29
Total ports scanned                28
Total ports open:                  28
Total ports unknown:               0
```

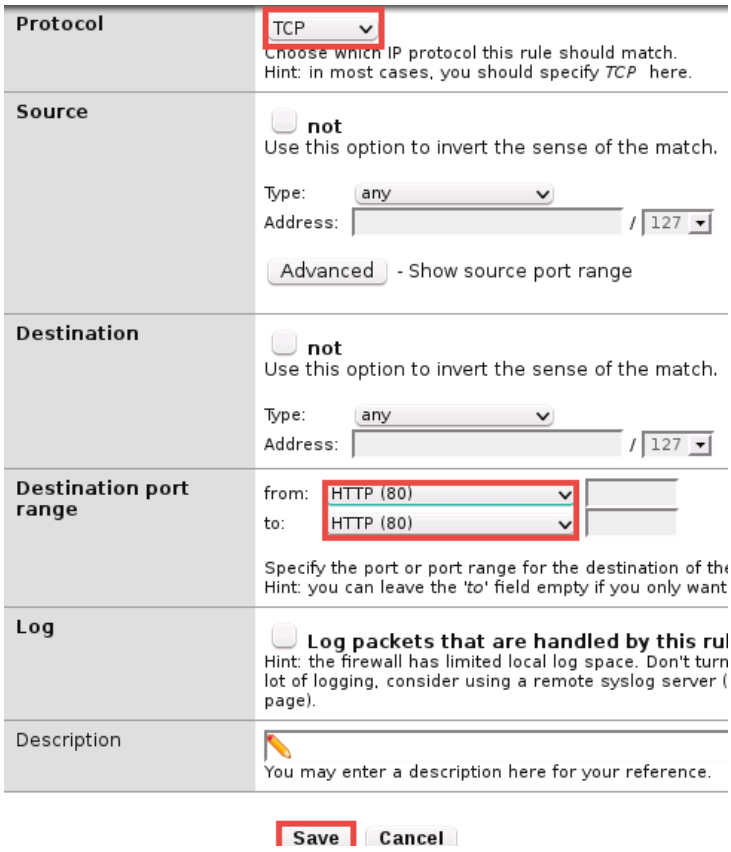
Notice no ACLs are present, even on the ports that are closed.

3 Configuring ACL Rules

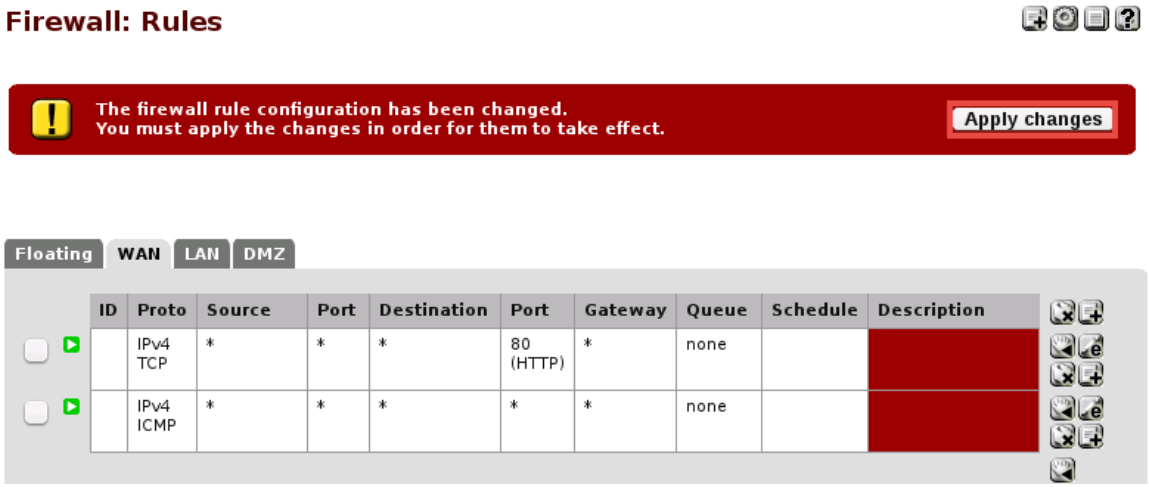
1. Navigate back to the **OpenSUSE** VM.
2. Using the *pfSense* administrative console, make sure to be viewing the **Firewall: Rules** page and click the first **Edit Rule** icon to edit the first *WAN* rule.



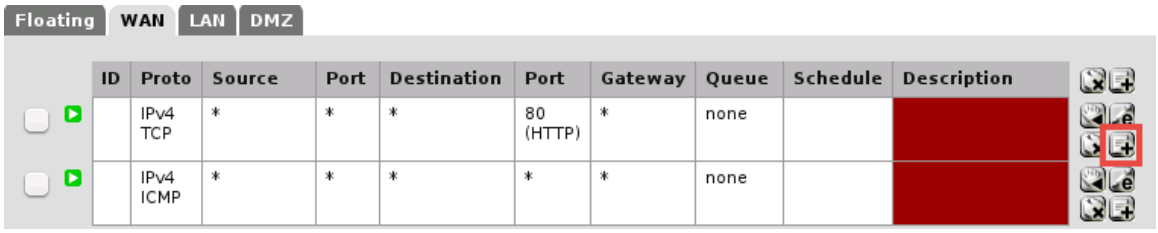
3. On the *Firewall: Rules: Edit* page, configure the following settings:
   a. Protocol: **TCP**
   b. Destination port range:
      i. from: **HTTP (80)**
      ii. to: **HTTP (80)**
   c. Everything else in **default**
   d. Click **Save**



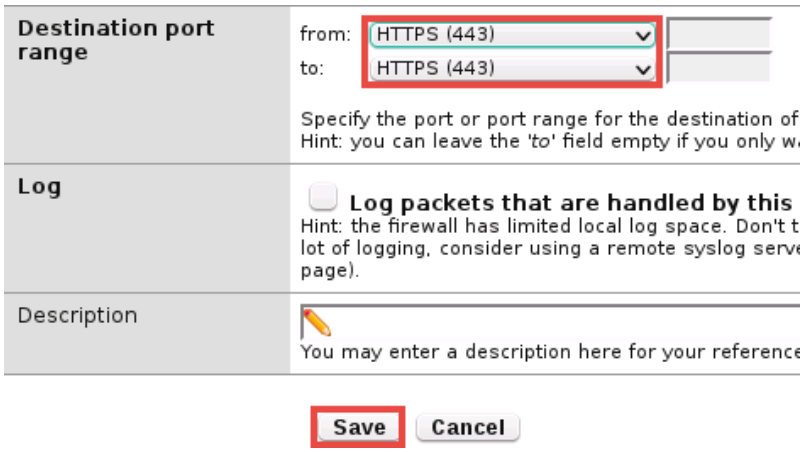
4. Once the page redirects, click the **Apply changes** button that appears at the top of the page.



5. Once applied, click the **add a new rule based on this one** (**+**) icon next to the first rule.

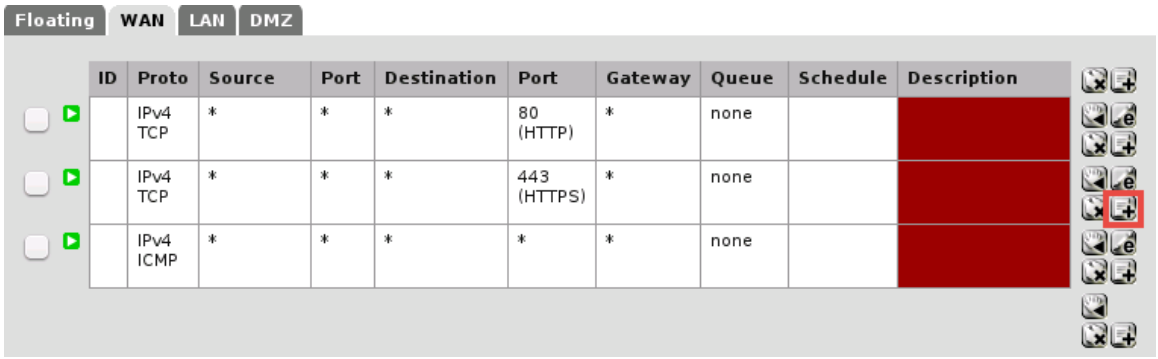


6. On the *Firewall: Rules: Edit* page, choose **HTTP (443)** as the *from:* and *to:* for Destination port range and leave the rest at their defaults. Click the Save button.

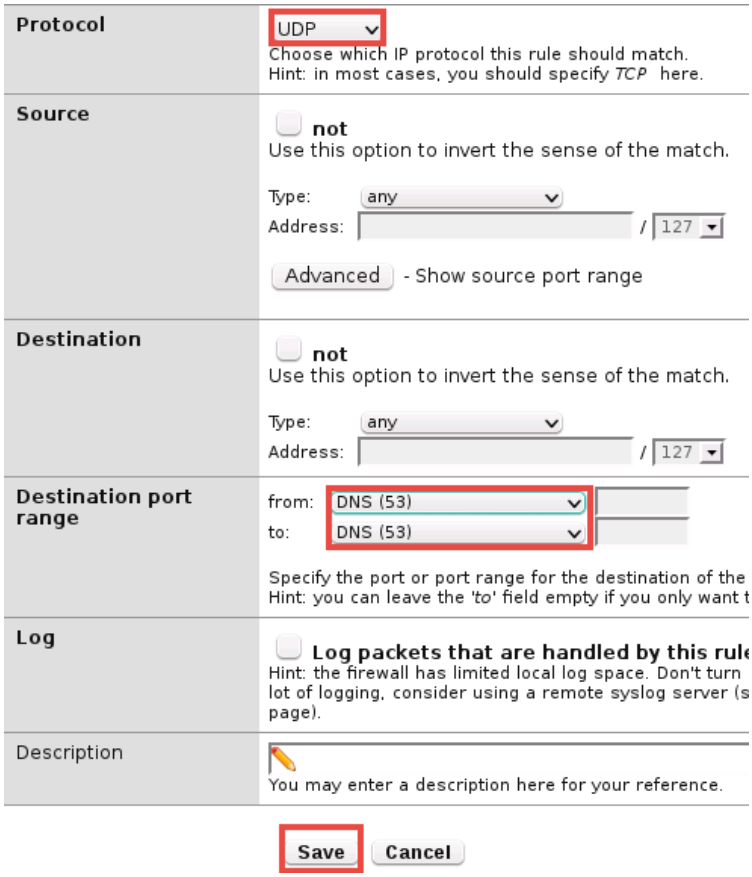


7. Once the page redirects, click the **Apply changes** button.

8. Once applied, click the **add a new rule based on this one** (+) icon next to the last rule.



9. On the *Firewall: Rules: Edit* page, configure the following settings:
   a. Protocol: **UDP**
   b. Destination port range:
      i. from: **DNS (53)**
      ii. to: **DNS (53)**
   c. Everything else in **default**
   d. Click **Save**



10. Once the page redirects, click the **Apply changes** button.

4 Test Configured Firewall Rules with Firewalk

1. Navigate back to the **Kali** VM.
2. Using the **Terminal**, enter the command below to try port 23 with *firewalk*.

```
firewalk -n -p TCP -S 23 -d 23 192.168.9.1 192.168.68.12
```

```
root@Kali2:~# firewalk -n -p TCP -S 23 -d 23 192.168.9.1 192.168.68.12
Firewalk 5.0 [gateway ACL scanner]
Firewalk state initialization completed successfully.
TCP-based scan.
Ramping phase source port: 53, destination port: 23
Hotfoot through 192.168.9.1 using 192.168.68.12 as a metric.
Ramping Phase:
 1 (TTL  1): *no response*
 2 (TTL  2): *no response*
 3 (TTL  3): *no response*
 4 (TTL  4): *no response*
 5 (TTL  5): *no response*
 6 (TTL  6): *no response*
 7 (TTL  7): *no response*
 8 (TTL  8): *no response*
```

Notice that *firewalk* now responds to the *ACL* rules in place with a no response back. The reason for no response back is that the ACL rule was set to use UDP not TCP.

3. Using the **Terminal**, enter the command below to try port 53 with *firewalk*.

```
firewalk -n -p TCP -S 53 -d 53 192.168.9.1 192.168.68.12
```

Notice similar results when compared to port 23.

4. Enter the command below to try port 53 but this time with UDP selected as an option.

```
firewalk -n -p UDP -S 53 -d 53 192.168.9.1 192.168.68.12
```

Notice a response back is now given, ICMP_UNREACH_PORT, which indicates that a rule may be in place.

5. Close the **Kali** and **OpenSUSE** PC viewers.

