



SECURITY+ LAB SERIES

Lab 1: Network Devices and Technologies – Capturing Network Traffic

Document Version: **2015-09-24**



This work by the National Information Security and Geospatial Technologies Consortium (NISGTC), and except where otherwise noted, is licensed under the [Creative Commons Attribution 3.0 Unported License](https://creativecommons.org/licenses/by/3.0/).

Development was funded by the Department of Labor (DOL) Trade Adjustment Assistance Community College and Career Training (TAACCT) Grant No. TC-22525-11-60-A-48; The National Information Security, Geospatial Technologies Consortium (NISGTC) is an entity of Collin College of Texas, Bellevue College of Washington, Bunker Hill Community College of Massachusetts, Del Mar College of Texas, Moraine Valley Community College of Illinois, Rio Salado College of Arizona, and Salt Lake Community College of Utah.

This workforce solution was funded by a grant awarded by the U.S. Department of Labor's Employment and Training Administration. The solution was created by the grantee and does not necessarily reflect the official position of the U.S. Department of Labor. The Department of Labor makes no guarantees, warranties or assurances of any kind, express or implied, with respect to such information, including any information on linked sites, and including, but not limited to accuracy of the information or its completeness, timeliness, usefulness, adequacy, continued availability or ownership.

Contents

Introduction	3
Lab Topology	4
Lab Settings	5
Pre-Lab Setup	6
1 Using tcpdump to Capture and Analyze Network Traffic	7
1.1 Using tcpdump to Capture ICMP Traffic	7
1.2 Using tcpdump to Capture ARP Traffic	11
1.3 Using arpspoof to Spoof Network Traffic.....	13
1.4 Using arpwatrch to Mitigate Spoofed Network Traffic	16
2 Using Wireshark to Capture & Analyze Network Traffic	17
2.1 Using Wireshark to Capture FTP Traffic	17
2.2 Using Wireshark to Capture SFTP Traffic	20
3 Capturing and Analyzing HTTP Traffic.....	23
3.1 Using dumpcap to Capture HTTP Traffic.....	23
3.2 Using Network Miner to Capture HTTP Traffic	24



Introduction

The material in this lab aligns to the following learning objectives:

- **Objective 1.1:** Implement security configuration parameters on network devices and other technologies
- **Objective 1.4:** Given a scenario, implement common protocols and services

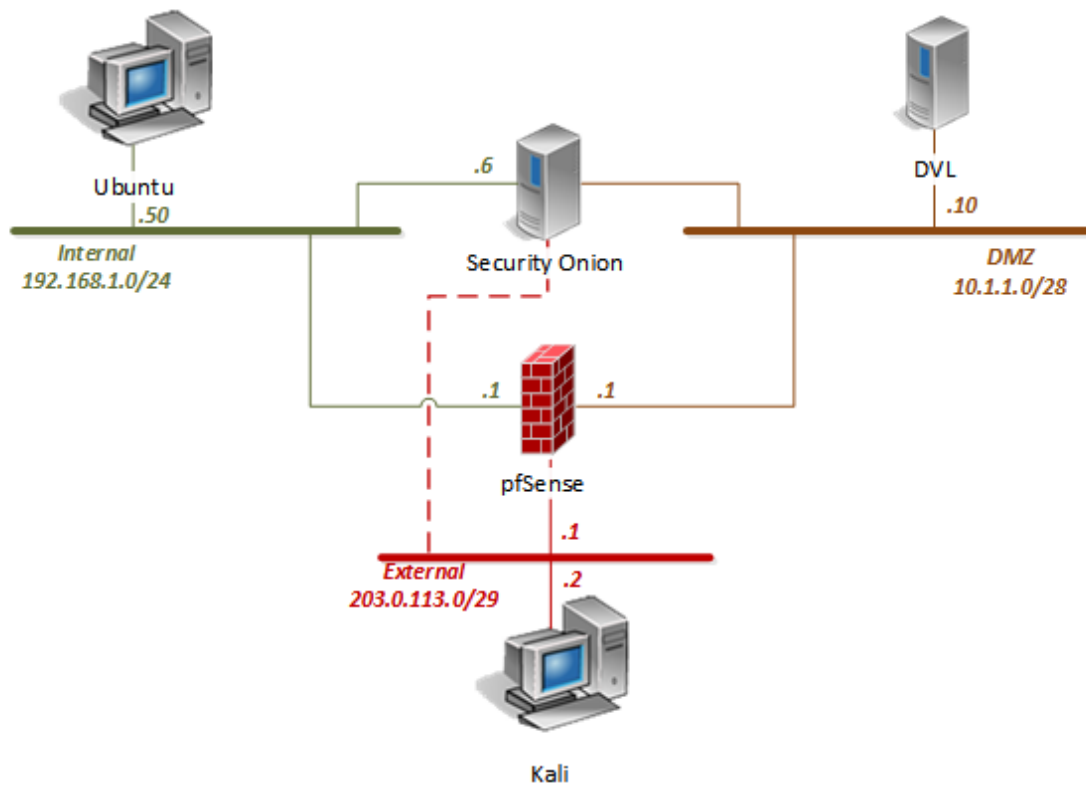
More information about individual objectives and their sections can be found in CompTIA document SY0-401, which is available from the CompTIA website.

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

1. Use tcpdump to Capture and Analyze Network Traffic
2. Use Wireshark to Capture and Analyze Network Traffic
3. Capture and Analyze HTTP Traffic



Lab 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)
Ubuntu	192.168.1.50	student	securepassword
DVL Server	10.1.1.10	root	toor
Security Onion	192.168.1.6	soadmin	mypassword
pfSense	192.168.1.1 10.1.1.1 203.0.113.1	admin	pfsense
Kali	203.0.113.2	root	toor



Pre-Lab Setup

Before continuing to Task 1, log into the following systems below as instructed.

I. Kali

1. On the login screen, select **Other**.
2. When presented with the username, type **root**. Press **Enter**.
3. When prompted for the password, type **toor**. Press **Enter**.
4. Minimize the *PC viewer* window.

II. Ubuntu

1. On the login screen, select the **student** account.
2. When prompted for the password, type **securepassword**. Press **Enter**.
3. Minimize the *PC viewer* window.

III. Security Onion

1. On the login screen, type **soadmin**. Press **Enter**.
2. When prompted for the password, type **mypassword**.
3. Minimize the *PC viewer* window.

IV. DVL

1. On the login screen, type **root**. Press **Enter**.
2. When prompted for a password, type **toor**. Press **Enter**.
3. When presented with the user prompt, type **startx**. Press **Enter**.
4. Once the desktop boots, close the *X Desktop* window.
5. Minimize the *PC viewer* window.

1 Using tcpdump to Capture and Analyze Network Traffic

1.1 Using tcpdump to Capture ICMP Traffic

1. Open the **Security Onion PC Viewer**. If closed, click on the **Security Onion** icon on the *Topology* page.



2. First, we will need to verify that the *Internal* and *External* adapters on the *Security Onion* machine are set to *promiscuous mode* so it can see all traffic. Open the *Terminal* window by double-clicking on the **Terminal Emulator** desktop icon.



3. Type the command below to view all available interfaces on the system.

```
ifconfig -a
```

```
soadmin@Security-Onion:~$ ifconfig -a
eth0      Link encap:Ethernet  HWaddr 00:50:56:9c:a8:23
          inet addr:192.168.1.6  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::250:56ff:fe9c:a823/64  Scope:Link
          UP BROADCAST RUNNING PROMISC MULTICAST  MTU:1500  Metric:1
          RX packets:759 errors:1 dropped:0 overruns:0 frame:0
          TX packets:342 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:527899 (527.8 KB)  TX bytes:26200 (26.2 KB)
          Interrupt:18 Base address:0x2000

eth1      Link encap:Ethernet  HWaddr 00:50:56:9c:3a:38
          UP BROADCAST RUNNING NOARP PROMISC MULTICAST  MTU:1500  Metric:1
          RX packets:195 errors:0 dropped:0 overruns:0 frame:0
          TX packets:9 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:18257 (18.2 KB)  TX bytes:1752 (1.7 KB)
          Interrupt:19 Base address:0x2080

eth2      Link encap:Ethernet  HWaddr 00:50:56:9c:9d:ba
          UP BROADCAST RUNNING NOARP PROMISC MULTICAST  MTU:1500  Metric:1
          RX packets:5963 errors:4 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1013728 (1.0 MB)  TX bytes:1001 (1.0 KB)
          Interrupt:16 Base address:0x2400

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128  Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:14757 errors:0 dropped:0 overruns:0 frame:0
          TX packets:14757 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:10596971 (10.5 MB)  TX bytes:10596971 (10.5 MB)
```

The *Security Onion* system has three interfaces, each assigned to a different network.

4. Issue the command below to identify which flags are currently set for each interface:

```
netstat -i
```

```
soadmin@Security-Onion:~$ netstat -i
Kernel Interface table
Iface  MTU Met  RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0    1500 0      761   1     0 0       342    0     0 0    0 BMPRU
eth1    1500 0      195   0     0 0         9    0     0 0    0 BMPORU
eth2    1500 0     5993   4     0 0         8    0     0 0    0 BMPORU
lo      65536 0     14901   0     0 0      14901   0     0 0    0 LRU
```

Notice how “*BMPRU*” is set for the *eth0* physical interface under the “*Flg*” column. Notice that “*BMPORU*” is set for both *eth1* and *eth2*. For a quick

overview: B flag is for broadcast, M flag is for multicast, P flag is for promisc mode, O flag is for no ARP (Address Resolution Protocol) requests, R flag is for running and U flag is for up. Also, notice that “*LRU*” is set for *lo*; the L flag means that the specified interface is a loopback device.

5. To familiarize yourself with the *tcpdump* utility, type the following command to view several available options for *tcpdump*:

```
tcpdump --help
```

```
soadmin@Security-Onion:~$ tcpdump --help
tcpdump: invalid option -- '-'
tcpdump version 4.2.1
libpcap version 1.1.1
Usage: tcpdump [-aAbDefhHIKlLnNOpqRStuUvX] [-B size] [-c count]
              [-C file_size] [-E algo:secret] [-F file] [-G seconds]
              [-i interface] [-M secret]
              [-r file] [-s snaplen] [-T type] [-w file]
              [-W filecount] [-y datalinktype] [-z command]
              [-Z user] [expression]
```

6. Open the **Kali PC Viewer**. If closed, click the **Kali** icon on the *Topology* page.



7. Open a new **Terminal** window.



8. Type the following command to initiate a continuous ping to the *Ubuntu* system.

```
ping 192.168.1.50
```

Proceed to the next step. The pings will continue in the background.

9. Switch back to the **Security Onion** system. Run **tcpdump** on the *internal network*, by typing the command below:

```
sudo tcpdump -i eth0 icmp
```

10. Notice the output that tcpdump provides: *HH:MM:SS.mmmmmm IP src > dst: ptype, id, seq, len*. Also, take note that for each echo request, there is a reply.

```
soadmin@Security-Onion:~$ sudo tcpdump -i eth0 icmp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
15:51:23.049370 IP 203.0.113.2 > 192.168.1.50: ICMP echo request, id 12639, seq 30, length 64
15:51:23.049490 IP 192.168.1.50 > 203.0.113.2: ICMP echo reply, id 12639, seq 30, length 64
```

HH:MM:SS.mmmmmm	Timestamp in hours, minutes, seconds and microseconds
IP	Internet Protocol
src > dst	Source and destination IP addresses
ptype	Packet type
id, seq, len	IP headers; identification, protocol (1=ICMP), total length

11. Press **CTRL+C** to stop *tcpdump* from running and discontinue the network capture.
12. From an administrator's standpoint, we may want to save the output from a *tcpdump capture* and save it automatically into a compatible file to view later with a program such as *Wireshark*. Initiate the command below to capture traffic on the **192.168.1.0/24** network and sending it to a file.

```
sudo tcpdump icmp -i eth0 -s 0 -w netcapture1.pcap -C 100
```

13. If prompted for a password, enter **mypassword**.

```
soadmin@Security-Onion:~$ sudo tcpdump icmp -i eth0 -s 0 -w netcapture1.pcap -C 100
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
```

The following table lists details of the options used with the *tcpdump* command:

icmp	Captures only ICMP packets (works for tcp, udp and icmp)
-i eth0	Use interface zero
-s 0	Disables default packet size, date and time format
-w	Write to a capture file, instead of displaying to the screen
-C	Split the captures into files of this size

- Wait for about 1-2 minutes and then press **CTRL+C** to stop *tcpdump* from running and discontinue the network capture.

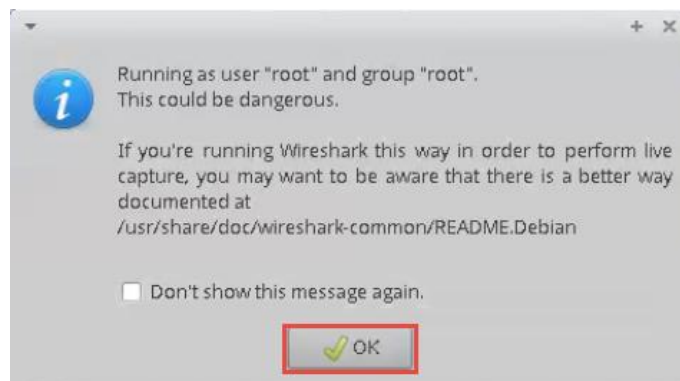
```
soadmin@Security-Onion:~$ sudo tcpdump icmp -i eth0 -s 0 -w netcapture1.pcap -C 100
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
^C143 packets captured
143 packets received by filter
0 packets dropped by kernel
```



- To view the captured file, type the command below in the *Security Onion Terminal*.

```
sudo wireshark netcapture1.pcap
```

- When prompted with a warning message, click **OK** to continue. Notice the traffic listed that takes place on the 192.168.1.0/24 network.



- Close **Wireshark**.

- Switch to the **Kali** machine and press **CTRL+C** to stop the continuous pings.

```
64 bytes from 192.168.1.50: icmp_req=242 ttl=63 time=0.458 ms
64 bytes from 192.168.1.50: icmp_req=243 ttl=63 time=0.471 ms
^C
--- 192.168.1.50 ping statistics ---
243 packets transmitted, 243 received, 0% packet loss, time 241999ms
rtt min/avg/max/mdev = 0.396/0.504/0.749/0.057 ms
```

1.2 Using tcpdump to Capture ARP Traffic

- Change focus to the **Security Onion** system and open a new **Terminal** window.

- Issue the *ARP* command below and take note of the results.

```
arp -n
```

```
soadmin@Security-Onion:~$ arp -n
Address                  Hwtype  Hwaddress          Flags Mask          Iface
192.168.1.1              ether   00:50:56:9c:3f:57  C                  eth0
```

- Type the command below to capture *ARP* packets.

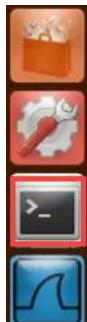
```
sudo tcpdump -i eth0 -nn -e arp
```

```
soadmin@Security-Onion:~$ sudo tcpdump -i eth0 -nn -e arp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
```

- Open the **Ubuntu PC Viewer**. If closed, click on the **Ubuntu** icon on the *Topology* page.



- Open a new **Terminal** window.



- Type the *ping* command below:

```
ping -c4 192.168.1.6
```



- Switch back to **Security Onion** and press **CTRL+C** to stop the capture. Notice the *ARP* output: HH:MM:SS:mmmmmm srcMAC > dstMAC: ptype, len, request/response, length.

```
soadmin@Security-Onion:~$ sudo tcpdump -i eth0 -nn -e arp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
15:56:21.147224 00:50:56:9c:59:78 > 00:50:56:9c:3f:57 ethertype ARP (0x0806) length 60 Request who-has 192.168.1.1 tell 192.168.1.50 length 46
15:56:21.147258 00:50:56:9c:3f:57 > 00:50:56:9c:59:78, ethertype ARP (0x0806), length 60: Reply 192.168.1.1 is-at 00:50:56:9c:3f:57, length 46
```

HH:MM:SS.mmmmm	Timestamp in hours, minutes, seconds and microseconds
src > dst	Source and destination MAC addresses
ptype	Packet type
len	Total packet length
request/response	Query response
Length	ARP packet length

8. Type the command shown below to display the ARP table.

```
arp -n
```

```
soadmin@Security-Onion:~$ arp -n
Address          Hwtype  Hwaddress      Flags Mask      Iface
192.168.1.1      ether   00:50:56:9c:3f:57  C           eth0
192.168.1.50     ether   00:50:56:9c:59:78  C           eth0
```

Notice the ARP entry for the IP address **192.168.1.50**.

9. Flush out the entire ARP table by typing the command below:

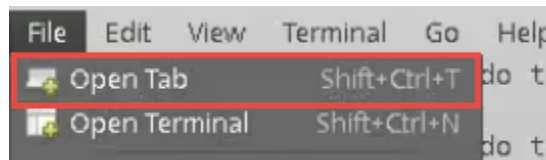
```
sudo ip -s -s neigh flush all
```

```
soadmin@Security-Onion:~$ sudo ip -s -s neigh flush all
192.168.1.1 dev eth0 lladdr 00:50:56:9c:3f:57 used 31/26/8 probes 1 STALE
192.168.1.50 dev eth0 lladdr 00:50:56:9c:59:78 used 51/47/29 probes 1 STALE

*** Round 1, deleting 2 entries ***
*** Flush is complete after 1 round ***
```

1.3 Using arpspoof to Spoof Network Traffic

1. While on the **Security Onion** system, in the *Terminal* window, select **File > Open Tab**.



- On the *first tab*, type the command below to verify the *eth0* is still in *promiscuous mode*.

```
netstat -i
```

```
soadmin@Security-Onion:~$ netstat -i
Kernel Interface table
```

Iface	MTU	Met	RX-OK	RX-ERR	RX-DRP	RX-OVR	TX-OK	TX-ERR	TX-DRP	TX-OVR	Flg
eth0	1500	0	1332	1	0	0	369	0	0	0	BMPRU
eth1	1500	0	237	0	0	0	9	0	0	0	BMPORU
eth2	1500	0	7927	4	0	0	8	0	0	0	BMPORU
lo	65536	0	48908	0	0	0	48908	0	0	0	LRU

- Once confirmed, configure the *Security Onion* system to act as a router between the *pfSense* router and the victim (in this case the *Ubuntu* system). Type the commands below to change the value from '0' to '1'. This will help by not modifying the source address of packets going through.

```
$: sudo -i
#: echo '1' > /proc/sys/net/ipv4/ip_forward
```

- If prompted for a password, type **mypassword**.
- Initiate the command below and leave it running in the background. This command will essentially spoof the host's MAC on the switch.

```
arp spoof -i eth0 -t 192.168.1.50 192.168.1.1
```

```
root@Security-Onion:~# arp spoof -i eth0 -t 192.168.1.50 192.168.1.1
0:50:56:9c:a8:23 0:50:56:9c:59:78 0806 42: arp reply 192.168.1.1 is-at 0:50:56:9c:a8:23
```

- Click on the **second tab** in the *Terminal* windows and initiate the command below to spoof the switch's MAC on the host. Leave it running in the background.

```
sudo arp spoof -i eth0 -t 192.168.1.1 192.168.1.50
```

```
soadmin@Security-Onion:~$ sudo arp spoof -i eth0 -t 192.168.1.1 192.168.1.50
0:50:56:9c:a8:23 0:50:56:9c:3f:57 0806 42: arp reply 192.168.1.50 is-at 0:50:56:9c:a8:23
```

- If prompted for a password, type **mypassword**. Press **Enter**.
- Open a **third tab** by selecting **File > Open Tab** in the *Terminal* window and type the *urlsnarf* command below. With this command, a man-in-the-middle attack can sniff the wire actively and monitor what information passes through from the victim. In this case, we are sniffing website data that the victim is entering in their web browser.

```
sudo urlsnarf -i eth0
```

```
soadmin@Security-Onion:~$ sudo urlsnarf -i eth0
urlsnarf: listening on eth0 [tcp port 80 or port 8080 or port 3128]
```


9. If prompted for a password, type **mypassword**. Press **Enter**.
10. Switch to the **Ubuntu** system and type the command below into a *Terminal* window to flush out the ARP table.

```
sudo ip -s -s neigh flush all
```

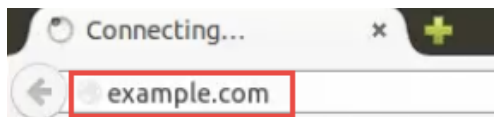
```
student@Ubuntu:~$ sudo ip -s -s neigh flush all
[sudo] password for student:
192.168.1.1 dev eth0 lladdr 00:50:56:9c:3f:57 ref 1 used 45/29/43 probes 1 REACHABLE

*** Round 1, deleting 1 entries ***
*** Flush is complete after 1 round ***
```

11. If prompted for a password, type **securepassword**. Press **Enter**.
12. Open a new **Web Browser** by clicking on the icon located on the left menu pane.



13. In the address bar, type **example.com**. Press **Enter**.



14. After a response is given, switch back to the **Security Onion** system. View the third tab in the Terminal window and observe the output from the **urlsnarf** command.

```
soadmin@Security-Onion:~$ sudo urlsnarf -i eth0
urlsnarf: listening on eth0 [tcp port 80 or port 8080 or port 3128]
192.168.1.50 - - [13/Apr/2015:17:14:57 +0000] "GET http://start.ubuntu.com/12.04/Google/?sourceid=hp HTTP/1.1" - - "Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:31.0) Gecko/20100101 Firefox/31.0"
192.168.1.50 - - [13/Apr/2015:17:14:57 +0000] "GET http://start.ubuntu.com/12.04/sprite.png HTTP/1.1" - - "Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:31.0) Gecko/20100101 Firefox/31.0"
192.168.1.50 - - [13/Apr/2015:17:15:00 +0000] "GET http://example.com/ HTTP/1.1" - - "Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:31.0) Gecko/20100101 Firefox/31.0"
192.168.1.50 - - [13/Apr/2015:17:15:00 +0000] "GET http://example.com/securityonion_logo.jpg HTTP/1.1" - - "http://example.com/" "Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:31.0) Gecko/20100101 Firefox/31.0"
```

Notice the “GET” entries and how easily it is to spoof ARP entries when on the same medium.

15. Press **CTRL+C** to stop the *urlsnarf* process.

1.4 Using arpwatrch to Mitigate Spoofed Network Traffic

1. Switch back to the **Ubuntu** system. Open a **Terminal** window and type the command below to initiate *arpwatch*, a tool that actively seeks any MAC address changes on the system's interface.

```
sudo arpwatch -i eth0
```

If prompted for a password, type **mypassword** and press Enter.



2. View the output by typing the command below.

```
tail -f /var/log/syslog
```

```
student@Ubuntu:~$ tail -f /var/log/syslog
Apr 13 13:12:29 Ubuntu arpwatch: flip flop 192.168.1.1 00:50:56:9c:3f:57 (00:50:56:9c:a8:23) eth0
Apr 13 13:12:34 Ubuntu postfix/sendmail[3560]: fatal: open /etc/postfix/main.cf: No such file or directory
Apr 13 13:12:34 Ubuntu arpwatch: reaper: pid 3560, exit status 75
Apr 13 13:14:24 Ubuntu arpwatch: flip flop 192.168.1.1 00:50:56:9c:a8:23 (00:50:56:9c:3f:57) eth0
Apr 13 13:14:29 Ubuntu postfix/sendmail[3569]: fatal: open /etc/postfix/main.cf: No such file or directory
Apr 13 13:14:29 Ubuntu arpwatch: reaper: pid 3569, exit status 75
Apr 13 13:15:54 Ubuntu arpwatch: chdir(/var/lib/arpwatch): Permission denied
Apr 13 13:15:54 Ubuntu arpwatch: (using current working directory)
Apr 13 13:15:54 Ubuntu arpwatch: pcap open eth0: eth0: You don't have permission to capture on that device (socket: Operation not permitted)
Apr 13 13:15:54 Ubuntu kernel: [ 7626.791285] arpwatch uses obsolete (PF_INET,SOCK_PACKET)
```

If you do not see the “flip flop” occur in the syslog right away, you may have to wait 1-2 minutes before you see the event happen in real time.

Notice the entries from *arpwatch*. This helps mitigate the ARP spoofing attack by informing the user when a MAC change has occurred.

3. Press **CTRL+C** to stop the process.
4. Switch back to the **Security Onion** system. Navigate to the **second tab** in the *Terminal* window and press **CTRL+C** to stop the running process. Do the same for the **first tab**.
5. Close the **Terminal** window. If prompted to close all tabs, click **Close all tabs**.

2 Using Wireshark to Capture & Analyze Network Traffic

2.1 Using Wireshark to Capture FTP Traffic

1. On the *Security Onion* system, open a **Terminal** window. Confirm that both *eth0* and *eth1* are up.

```
ifconfig -a
```

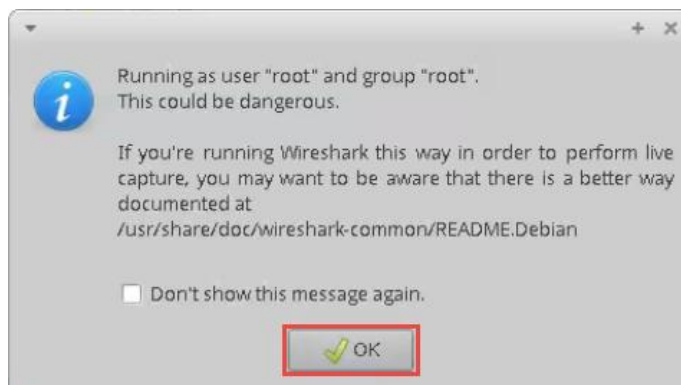
```
soadmin@Security-Onion:~$ ifconfig -a
eth0      Link encap:Ethernet  HWaddr 00:50:56:9c:a8:23
          inet addr:192.168.1.6  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::250:56ff:fe9c:a823/64 Scope:Link
          UP BROADCAST RUNNING PROMISC MULTICAST  MTU:1500  Metric:1
          RX packets:3472 errors:1 dropped:0 overruns:0 frame:0
          TX packets:6033 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:980709 (980.7 KB)  TX bytes:633165 (633.1 KB)
          Interrupt:18 Base address:0x2000

eth1      Link encap:Ethernet  HWaddr 00:50:56:9c:3a:38
          UP BROADCAST RUNNING NOARP PROMISC MULTICAST  MTU:1500  Metric:1
          RX packets:545 errors:0 dropped:0 overruns:0 frame:0
          TX packets:9 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:47377 (47.3 KB)  TX bytes:1752 (1.7 KB)
          Interrupt:19 Base address:0x2080
```

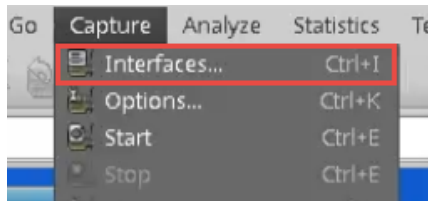
2. If either interface is down, bring them back up by using the **ifconfig eth0 up** command.
3. Type the command below to run *Wireshark* as root.

```
sudo wireshark
```

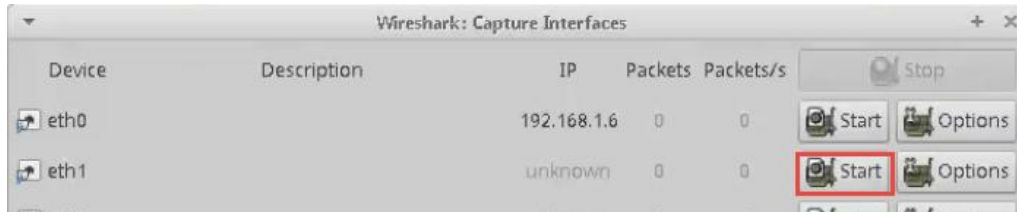
- a. If prompted for password, type **mypassword**. Press **Enter**.
- b. If prompted with a message stating that running Wireshark can be dangerous to run while in root, select **OK** to proceed.



4. Start capturing traffic by clicking the top menu option **Capture > Interfaces**.



5. In the new *Capture Interfaces* window, click the **Start** button for the *eth1* network device.



10. Open the **DVL PC Viewer**. If closed, click on the **DVL** icon on the *Topology* page.



6. Open a new **Terminal** window by clicking on the icon located on the bottom menu pane.

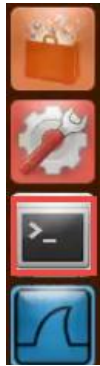


7. Type **proftpd** followed by pressing **Enter** to initiate the FTP server.

```
bt ~ # proftpd
- IPv6 getaddrinfo 'bt.example.net' error: Name or service not known
bt ~ #
```

Ignore the bt.example.net error.

8. Switch to the **Ubuntu** system and open a **Terminal** window.

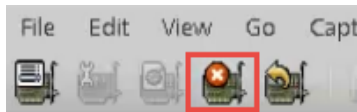


9. Type the command below to connect to the FTP server located on the *DVL Server*.

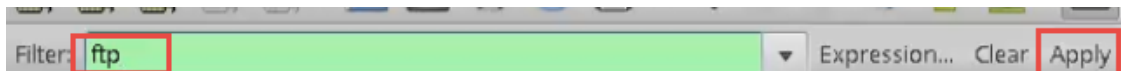
```
ftp 10.1.1.10
```

```
student@Ubuntu:~$ ftp 10.1.1.10
Connected to 10.1.1.10.
220 ProFTPD 1.3.0 Server (ProFTPD Default Installation) [::ffff:10.1.1.10]
Name (10.1.1.10:student): ftp
331 Password required for ftp.
Password:
230 User ftp logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
```

11. When prompted for a user name and password, enter: **ftp** as the user and **ftp** again as the password.
12. Once connected to the FTP service, switch back to the **Security Onion** system and click the **Stop Capture** button in the *Wireshark* interface.



13. Type **ftp** in the filter pane and click **Apply**.





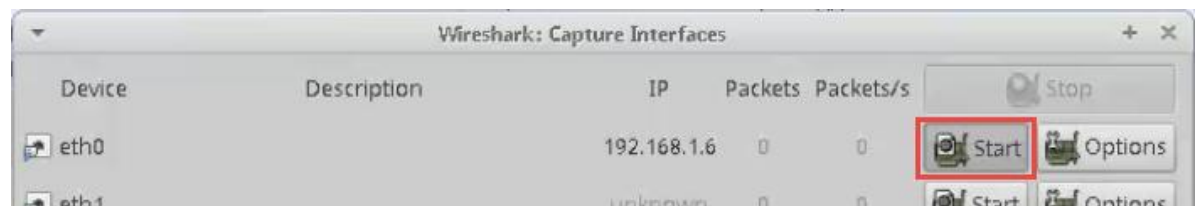
- Now that the packet focus is on FTP only, locate the initial request. Here, you can see the username *ftp* and the password of *ftp* in clear text.

No.	Time	Source	Destination	Protocol	Length	Info
24	127.567847	10.1.1.10	192.168.1.50	FTP	142	Response: 220 ProFTPD 1.3.0 Server (ProF
26	134.133804	192.168.1.50	10.1.1.10	FTP	76	Request: USER ftp
28	134.135819	10.1.1.10	192.168.1.50	FTP	98	Response: 331 Password required for ftp.
30	138.134183	192.168.1.50	10.1.1.10	FTP	76	Request: PASS ftp

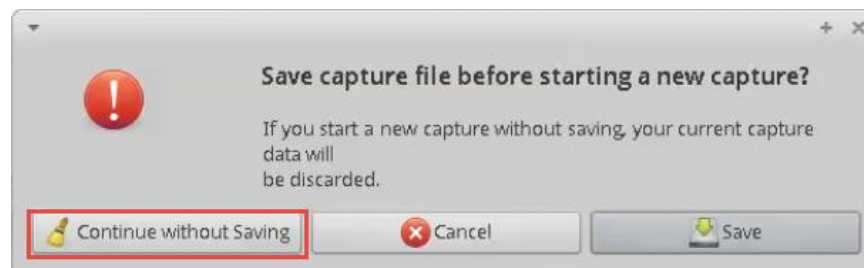
Using FTP has gone a long way; since it is no longer a secure channel to use, we will show how using VSFTP (Very Secure FTP) can be more secure in the section.

2.2 Using Wireshark to Capture SFTP Traffic

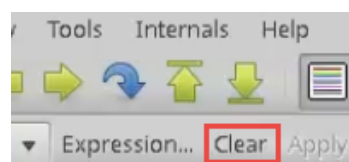
- Start a new capture by selecting **Capture > Interfaces** and then clicking the **Start** button for the **eth0** network device.



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



- Click on the **Clear** button to clear the filter pane.



- Switch to the **Ubuntu** system.

5. While on the **Terminal** window, type **exit** followed by pressing **Enter** to log out from the FTP server.

```
ftp> exit
221 Goodbye.
student@Ubuntu:~$
```

6. Type the command below to verify that the SSH service is running.

```
service ssh status
```

```
student@Ubuntu:~$ sudo service ssh start
start: Job is already running: ssh
```

7. If the SSH service is not running, type the command below followed by **Enter**:

```
sudo service ssh start
```

8. Open the **Kali PC Viewer**. If closed, click on the **Kali** icon on the *Topology* page.

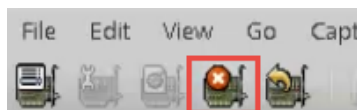


9. While logged into Kali, open the Terminal window and type the command below:

```
sftp student@192.168.1.50
```

```
root@Kali-Attacker:~# sftp student@192.168.1.50
The authenticity of host '192.168.1.50 (192.168.1.50)' can't be established.
ECDSA key fingerprint is a9:a5:61:27:5c:ad:3d:cc:64:78:1d:fa:5f:d8:47:1f.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.50' (ECDSA) to the list of known hosts.
student@192.168.1.50's password:
Connected to 192.168.1.50.
sftp>
```

- a. If prompted “Are you sure you want to continue connecting”, type **yes**. Press **Enter**.
 - b. When prompted for a password, enter **securepassword**. Press **Enter**.
10. Switch to the **Security Onion** system and stop the capture by clicking the **Stop Capture** button.



11. Locate the *Diffie-Hellman key exchange* between the client and the SFTP service.



SSHv2	1050 Server: Key Exchange Init
SSHv2	146 Client: Diffie-Hellman Key Exchange Init
SSHv2	378 Server: New Keys
SSHv2	82 Client: New Keys

12. Notice after the key exchange, the TCP packets that follow are encrypted over the medium.

TCP	66 ssh > 34336 [ACK] Seq=1336 Ack=1408 Win=31872 Len=0 TSval=92321 TSecr=2251481
TCP	114 [TCP segment of a reassembled PDU]
TCP	66 ssh > 34336 [ACK] Seq=1336 Ack=1456 Win=31872 Len=0 TSval=92321 TSecr=2251490
TCP	114 [TCP segment of a reassembled PDU]
TCP	130 [TCP segment of a reassembled PDU]

You can no longer see the username and password in clear text when compared to using FTP.

13. Close the **Wireshark** application.

3 Capturing and Analyzing HTTP Traffic

3.1 Using dumpcap to Capture HTTP Traffic

1. While on the **Security Onion** system, navigate to the **Terminal** window and type the command below:

```
sudo dumpcap -i eth0 -w /tmp/netcapture2.pcap
```

2. If prompted for a password, type **mypassword**. Press **Enter**. Leave it running in the background.
3. Switch focus to the **Ubuntu** system and open the **Web Browser**.



4. Type **192.168.1.6** into address bar. Press **Enter**.



Wait until the page finishes loading.

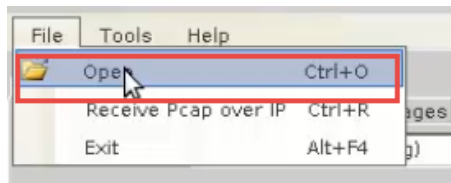
5. Switch back to the **Security Onion** system. Press **CTRL+C** to stop the running *dumpcap* process.

3.2 Using Network Miner to Capture HTTP Traffic

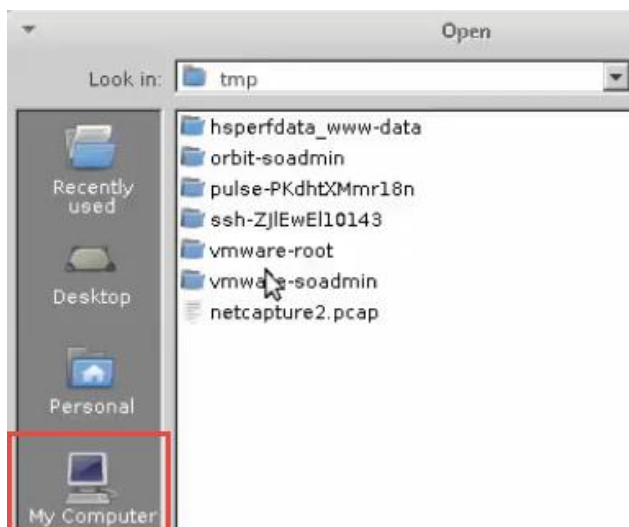
1. While in the *Terminal*, type the command below to open the program **Network Miner**:

```
sudo /opt/networkminer/networkminer
```

2. On the **Network Miner** window, navigate to **File > Open**.



3. Select the **My Computer** icon from the left menu.



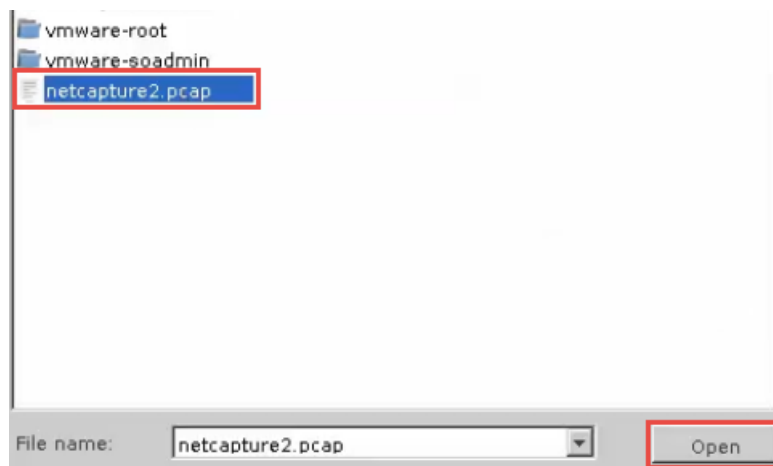
4. Double-click **HDD**.



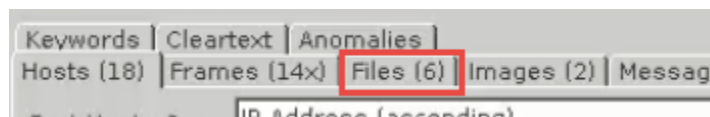
- Next, double-click the **tmp** directory.



- Select the **netcapture2.pcap** file and select **Open**.



- Click on the **Files** tab within the *Network Miner* program.



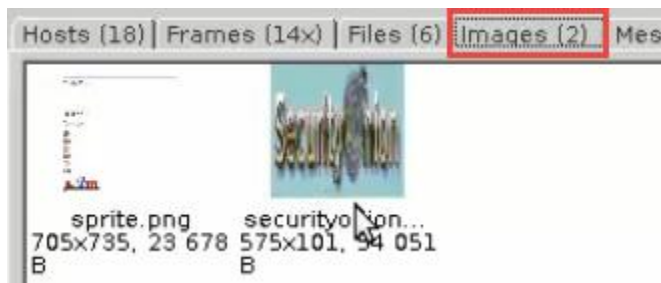
- Right-click and select **Open file** for the second html file listed that has a filename of *index.html*.

Hosts (18) Frames (14x) Files (6) Images (2) Messages Credentials Sessions (2) DNS (17) P									
Fram...	Recon...	Sour...	S. port	Destin...	D. port	Protocol	File...	Exten...	
13	/opt/n...	91.18...	TCP 80	192.1...	TCP 4...	HttpG...	index...	html	6 0
21	/opt/n...	91.18...	TCP 80	192.1...	TCP 4...	HttpG...	sprite...	png	23 6
73	/opt/n...	192.1...	TCP 80	192.1...	TCP 5...	HttpG...	index[...	html	7
77	/opt/n...	192.1...	TCP 80	19			securi...	jpg	54 0
136	/opt/n...	192.1...	TCP 80	19			favicon...	html	7

9. You should see *Security Onion's homepage*. Close the web browser window.



10. While on the **Network Miner** application, click on the **Images** tab. (Images captured from the web pages visited during the capture will be shown here.)



11. **Close** all windows.