Lab 5.2.2 - ARP with Wireshark

Name: ______________________________ Date: ______________ Class: ______________

In this lab you will learn how to use the Wireshark application for packet capture and network traffic analysis. First watch this video to get an overview of Wireshark and a brief demonstration of how to use the features. (https://vimeo.com/579540878) Here are some key points about Wireshark:

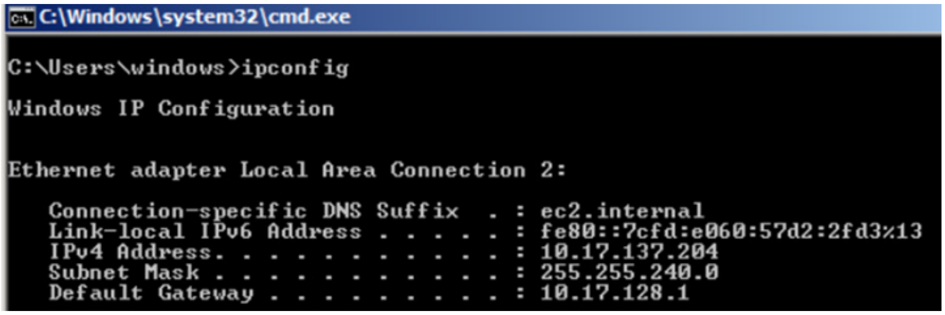
- Data goes across a network in “packets” (more details about that in the next unit).
- Wireshark is a *network* packet analyzer - aka *packet sniffer*
- Wireshark can capture a copy of the data as it goes across a LAN between devices.
- That means a user can examine all parts of the data packets telling us who it went to, what time, the actual message inside and lots more.
- **Use for good** – troubleshoot a network to see why there is too much traffic or packets are being list.
- **Use for bad** – steal plaintext passwords or files that are being transmitted.

Instruction:

- Go to CYBER.ORG **Range** (https://apps.cyber.org)
- Click on the **Range tab**, then click on **Launch Kali** and **Launch Windows7**.
- Once the status changes to booted for Windows 7, click Open.
- In the bottom left, click the **Start** button and search for **Network and Sharing Center**
- In the pop-up window, select the **Public Network** and change it to **Home**
- Open a command prompt on your Windows computer

To open the command prompt in Windows: Click the Start button in the lower left, then type “cmd” and click Enter

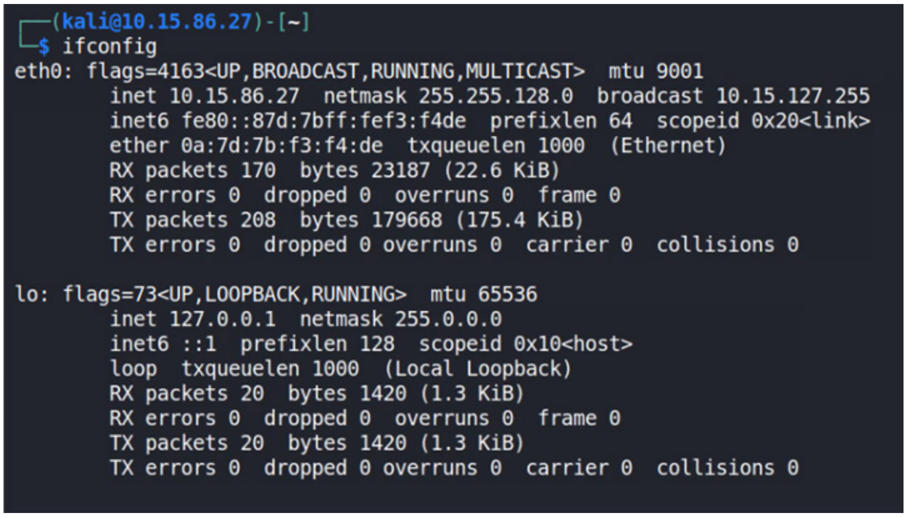
- Type **`ipconfig`**
- In the results, look for “Ethernet adapter Local Area Connection 2”



```
C:\Windows\system32\cmd.exe

C:\Users\windows>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection 2:

   Connection-specific DNS Suffix  . : ec2.internal
   Link-local IPv6 Address . . . . . : fe80::7cfd:e060:57d2:2fd3%13
   IPv4 Address. . . . . . . . . . . : 10.17.137.204
   Subnet Mask . . . . . . . . . . . : 255.255.240.0
   Default Gateway . . . . . . . . . : 10.17.128.1
```

What is the IPv4 address of this computer?

- Switch to the Kali machine by selecting Machines in the top right corner of your webpage
- Open a command prompt on your Kali.
- Type **`ifconfig`**



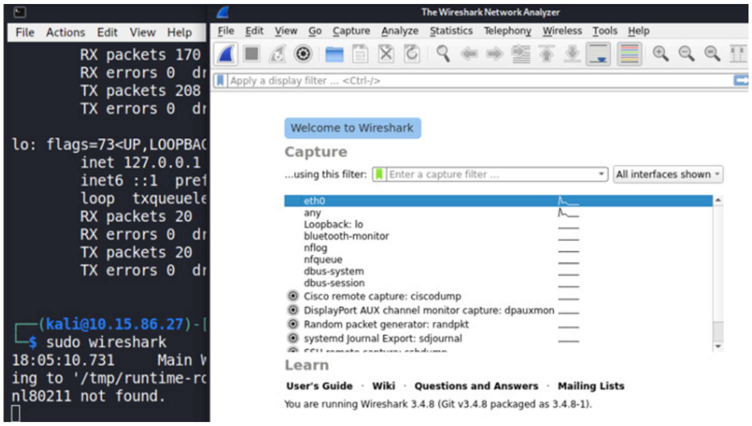
```
┌──(kali@10.15.86.27)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9001
        inet 10.15.86.27  netmask 255.255.128.0  broadcast 10.15.127.255
        inet6 fe80::87d:7bff:fef3:f4de  prefixlen 64  scopeid 0x20<link>
        ether 0a:7d:7b:f3:f4:de  txqueuelen 1000  (Ethernet)
        RX packets 170  bytes 23187 (22.6 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 208  bytes 179668 (175.4 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 20  bytes 1420 (1.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 20  bytes 1420 (1.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

- In the results, scroll down to find "ether xx:xx:xx:xx:xx:xx"

  What is the MAC address of this computer? (The xx:xx:xx:xx:xx:xx)

  What is the IPv4 address of this computer?

- Type **`sudo wireshark`**
- On the main screen, click the blue fin to start. Wireshark will now start capturing any packets that travel to or from the Kali Linux machine



File Actions Edit View Help
The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Welcome to Wireshark

```
        RX packets 170
        RX errors 0  dr
        TX packets 208
        TX errors 0  dr

lo: flags=73<UP,LOOPBAC
        inet 127.0.0.1
        inet6 ::1  pref
        loop  txqueuele
        RX packets 20
        RX errors 0  dr
        TX packets 20
        TX errors 0  dr


┌──(kali@10.15.86.27)-[
└─$ sudo wireshark
18:05:10.731     Main W
ing to '/tmp/runtime-ro
nl80211 not found.
```

Capture

...using this filter: Enter a capture filter ... All interfaces shown

eth0
any
Loopback: lo
bluetooth-monitor
nflog
nfqueue
dbus-system
dbus-session
Cisco remote capture: ciscodump
DisplayPort AUX channel monitor capture: dpauxmon
Random packet generator: randpkt
systemd Journal Export: sdjournal

Learn

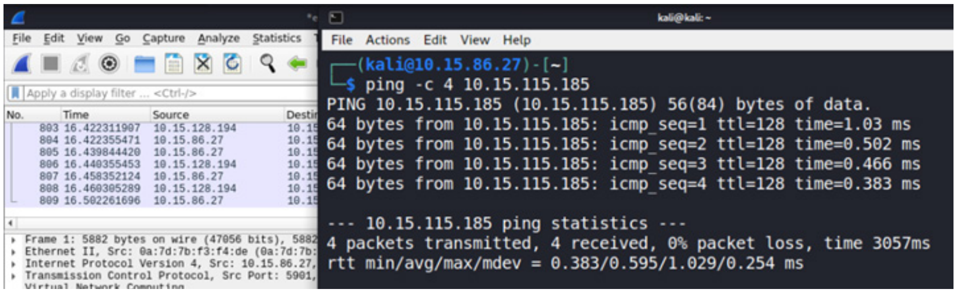
User's Guide · Wiki · Questions and Answers · Mailing Lists

You are running Wireshark 3.4.8 (Git v3.4.8 packaged as 3.4.8-1).

- Leave Wireshark open and open another Command prompt window. You will now try to contact the Windows device using the ping command. PING is a command prompt command that sends small packets to another device and asks for a reply. It is most often used to test connectivity between two devices.

  Type `ping -c 4 10.15.x.x` (**Make sure to replace the x's with the two numerals for YOUR Windows 7 machine)**

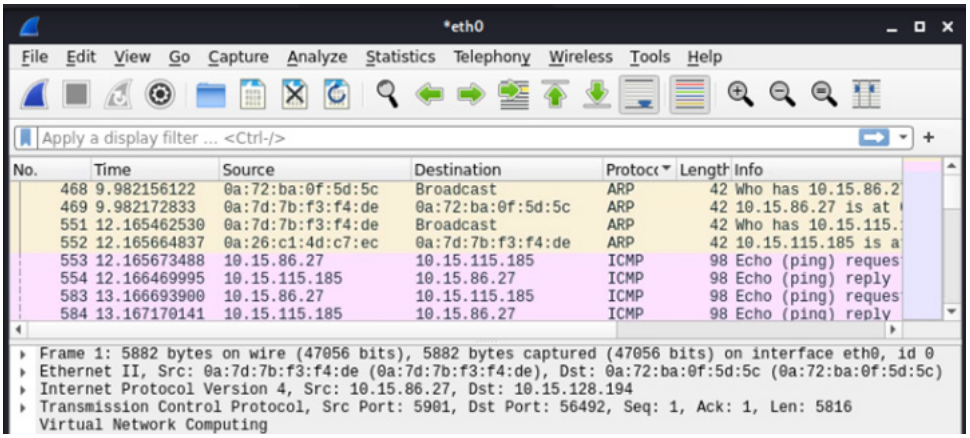
- Stop the Wireshark capture by clicking on the red box in the upper left.


File Edit View Go Capture Analyze Statistics
Apply a display filter ... <Ctrl-/>
File Actions Edit View Help
kali@kali: ~

```
No.  Time          Source          Desti
 803 16.422311907  10.15.128.194   10.15
 804 16.422355471  10.15.86.27     10.15
 805 16.439844420  10.15.86.27     10.15
 806 16.440355453  10.15.128.194   10.15
 807 16.458352124  10.15.86.27     10.15
 808 16.460305289  10.15.128.194   10.15
 809 16.502261696  10.15.86.27     10.15

▸ Frame 1: 5882 bytes on wire (47056 bits), 5882
▸ Ethernet II, Src: 0a:7d:7b:f3:f4:de (0a:7d:7b:
▸ Internet Protocol Version 4, Src: 10.15.86.27,
▸ Transmission Control Protocol, Src Port: 5901,
  Virtual Network Computing
```

```
┌──(kali@10.15.86.27)-[~]
└─$ ping -c 4 10.15.115.185
PING 10.15.115.185 (10.15.115.185) 56(84) bytes of data.
64 bytes from 10.15.115.185: icmp_seq=1 ttl=128 time=1.03 ms
64 bytes from 10.15.115.185: icmp_seq=2 ttl=128 time=0.502 ms
64 bytes from 10.15.115.185: icmp_seq=3 ttl=128 time=0.466 ms
64 bytes from 10.15.115.185: icmp_seq=4 ttl=128 time=0.383 ms

--- 10.15.115.185 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3057ms
rtt min/avg/max/mdev = 0.383/0.595/1.029/0.254 ms
```

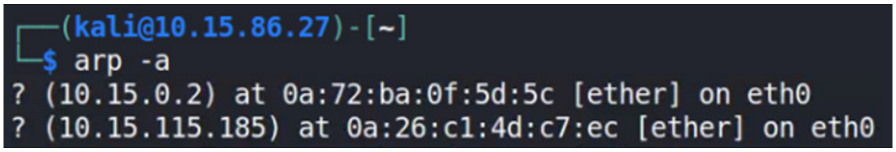
- Click the Protocol tab to sort by protocol. Scroll up to the top and you will see ARP packets. Remember, ARP (Address Resolution Protocol) is a tool used by devices on the same network to find the MAC address associated with an IP address.


*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.  Time          Source             Destination        Protoco Length Info
 468 9.982156122   0a:72:ba:0f:5d:5c  Broadcast          ARP        42 Who has 10.15.86.2
 469 9.982172833   0a:7d:7b:f3:f4:de  0a:72:ba:0f:5d:5c  ARP        42 10.15.86.27 is at
 551 12.165462530  0a:7d:7b:f3:f4:de  Broadcast          ARP        42 Who has 10.15.115.
 552 12.165664837  0a:26:c1:4d:c7:ec  0a:7d:7b:f3:f4:de  ARP        42 10.15.115.185 is a
 553 12.165673488  10.15.86.27        10.15.115.185      ICMP       98 Echo (ping) reques
 554 12.166469995  10.15.115.185      10.15.86.27        ICMP       98 Echo (ping) reply
 583 13.166693900  10.15.86.27        10.15.115.185      ICMP       98 Echo (ping) reques
 584 13.167170141  10.15.115.185      10.15.86.27        ICMP       98 Echo (ping) reply

▸ Frame 1: 5882 bytes on wire (47056 bits), 5882 bytes captured (47056 bits) on interface eth0, id 0
▸ Ethernet II, Src: 0a:7d:7b:f3:f4:de (0a:7d:7b:f3:f4:de), Dst: 0a:72:ba:0f:5d:5c (0a:72:ba:0f:5d:5c)
▸ Internet Protocol Version 4, Src: 10.15.86.27, Dst: 10.15.128.194
▸ Transmission Control Protocol, Src Port: 5901, Dst Port: 56492, Seq: 1, Ack: 1, Len: 5816
  Virtual Network Computing
```

Look in your capture results for the ARP packets. The Kali machine asks who has 10.15.x.x. Below enter the MAC address of the replying device:

- We can also find the MAC addresses of the devices we communicate with by examining the arp table. Return to the Command prompt window:
  Type **`arp -a`**



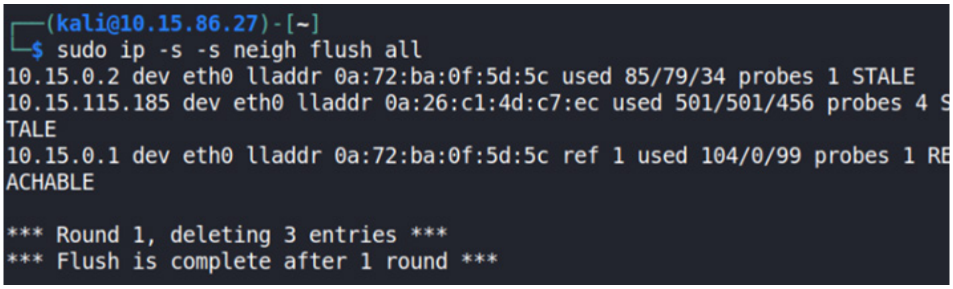
```
┌──(kali@10.15.86.27)-[~]
└─$ arp -a
? (10.15.0.2) at 0a:72:ba:0f:5d:5c [ether] on eth0
? (10.15.115.185) at 0a:26:c1:4d:c7:ec [ether] on eth0
```

- Look at your results:
  What is listed as the Physical Address for the Windows machine?

Does this match the information gathered from the Wireshark capture?

- Now clear the arp table

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



```
┌──(kali@10.15.86.27)-[~]
└─$ sudo ip -s -s neigh flush all
10.15.0.2 dev eth0 lladdr 0a:72:ba:0f:5d:5c used 85/79/34 probes 1 STALE
10.15.115.185 dev eth0 lladdr 0a:26:c1:4d:c7:ec used 501/501/456 probes 4 S
TALE
10.15.0.1 dev eth0 lladdr 0a:72:ba:0f:5d:5c ref 1 used 104/0/99 probes 1 RE
ACHABLE

*** Round 1, deleting 3 entries ***
*** Flush is complete after 1 round ***
```

- Check to see that it is cleared (the 10.15.0.1 address will still be there).
  Type **`arp - a`**
- Partner with another classmate and repeat the previous steps pinging your classmate's machines!