**CSCE 629 Lab 6**

**Winter 2019**

**Network Attacks**

**Assigned: Lesson 17, 31 Jan**

**Due: Lesson 25, 14 Feb, 1400**

You will work with your partner and submit one solution. I encourage you to switch roles (i.e., Blackhat and target) throughout the assignment. You must be root for most of these commands to work; use the **su** command if necessary.

**Task 1. ARP Cache Poisoning**

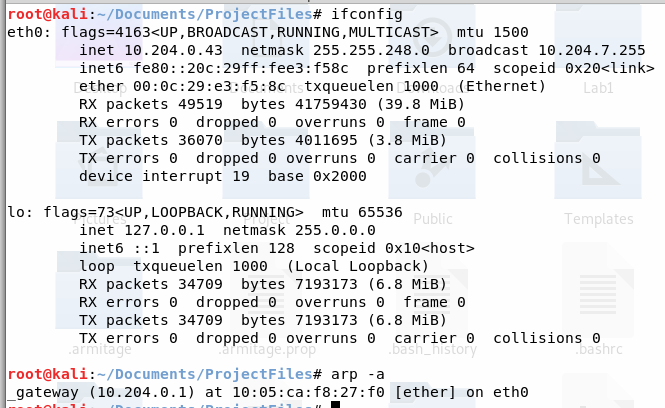
This assignment requires you to poison the ARP cache on the target’s machine in order to sniff his/her traffic through a switch. You will be implementing the attack shown in the following figure, which should help visualize the attack and keep track of all the moving parts.



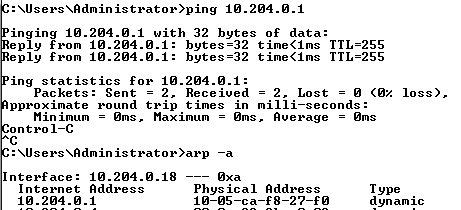
Record the IP and MAC address of the gateway:

**Gateway IP: 10.204.0.1 Gateway MAC: 10-05-CA-F8-27-F0**

* The gateway information was found using the “ifconfig” command followed by “arp –a”, as shown below, from the BH computer:

****

* The target verified the default gateway MAC using

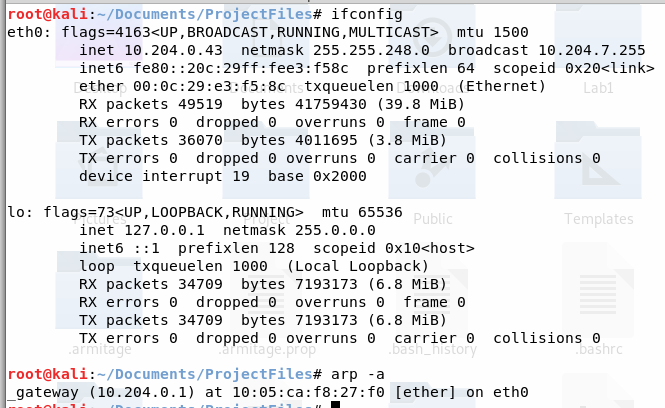
****

Blackhat (BH):

* Log into Kali.
* Record your IP and MAC addresses here:

**BH IP : 10.204.0.43 BH MAC: 00-0C-29-E3-F5-8C**

* The below information was found using the following command:

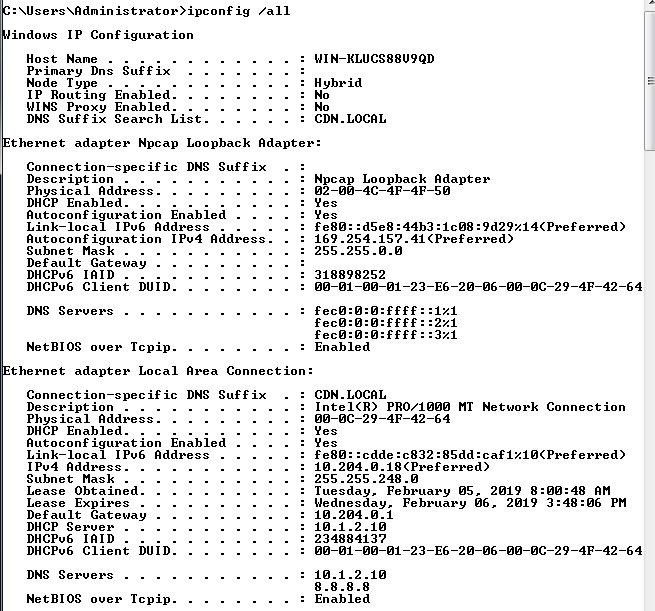
****

Target:

* Log into Windows
* Record your IP and MAC addresses here:

**Target IP: 10.204.0.18 Target MAC 00-0C-29-4F-42-64**

* The IP and MAC were found using the command on the next page, with results outlined in red: “ipconfig /all”



BH:

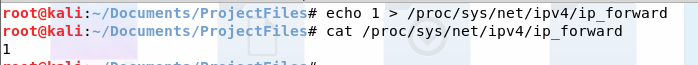
* Turn on IP forwarding on your machine by executing:

**echo 1 > /proc/sys/net/ipv4/ip\_forward**

* Verify IP forwarding was actually turned on by executing:

**cat /proc/sys/net/ipv4/ip\_forward** 🡨 should display a 1

The above commands produced the following output:

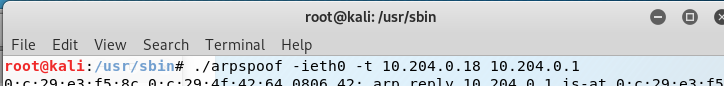


* Start sniffing your Ethernet interface with Wireshark. Remember you can filter the packets displayed by using eth.dst==00:02:29:4F:42:64
* Find your copy of arpspoof and execute the tool against your partner’s computer (target). You will need your partner’s IP address. I assume your machine is using eth0; if not, change “eth0” to the appropriate interface.

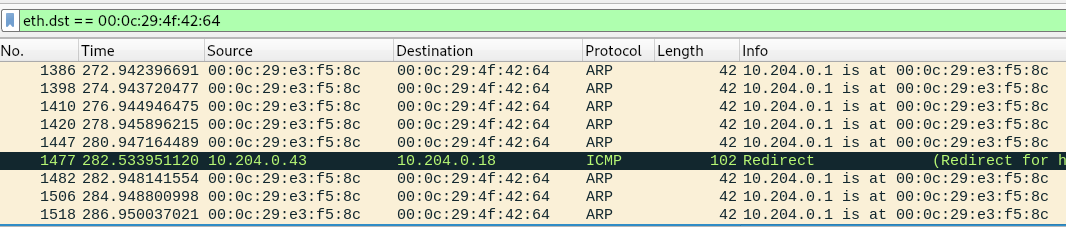
**cd /**<<location of arpspoof>> (perhaps in /usr/sbin/)

**./arpspoof -ieth0 –t<<targetIP>> <<gatewayIP>>**

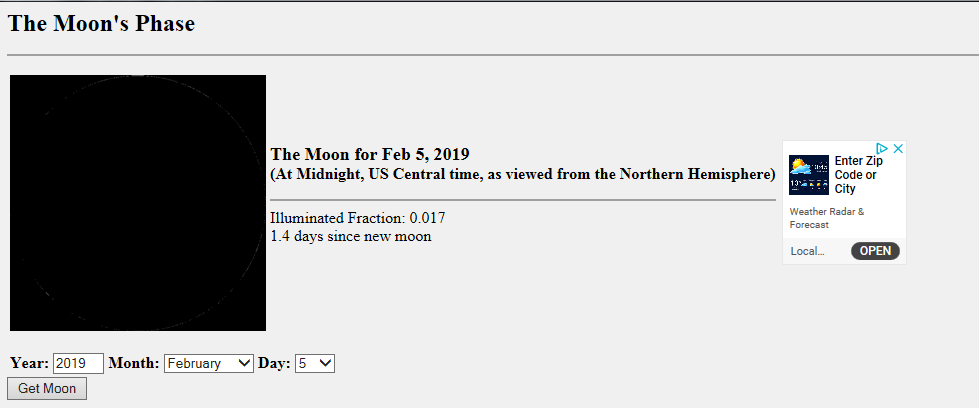
* The command executed is shown below, executed from /usr/sbin:

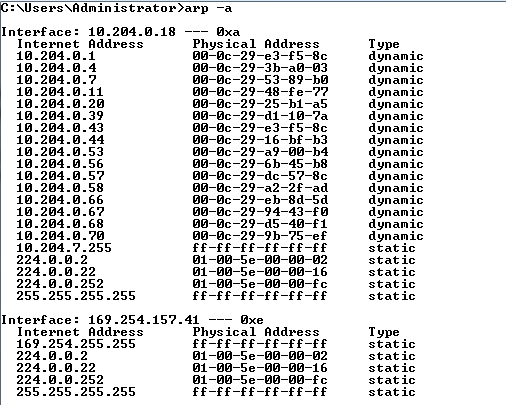
****

* **Do you see the ARP responses being sent to the target? How often are they sent?**
  + Yes, they are being sent roughly every two seconds. The ARP response shows the BH telling the Target that 10.204.0.1 (the default gateway) is at the BH’s MAC address.



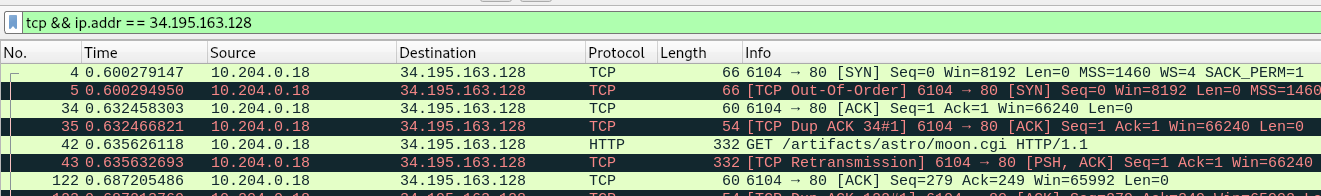
Target:

* Open a web browser and surf to www.briancasey.org/artifacts/astro/moon.cgi.
* Your webpage should be displayed.
* **List the contents of your ARP table again. What is the MAC address displayed for the gateway? Is this the address you found for the gateway above?**
  + The ARP table displayed below shows the MAC address of 00-0C-29-E3-F5-8C (the Black Hat’s). No, it is not the same MAC address as the actual default gateway.

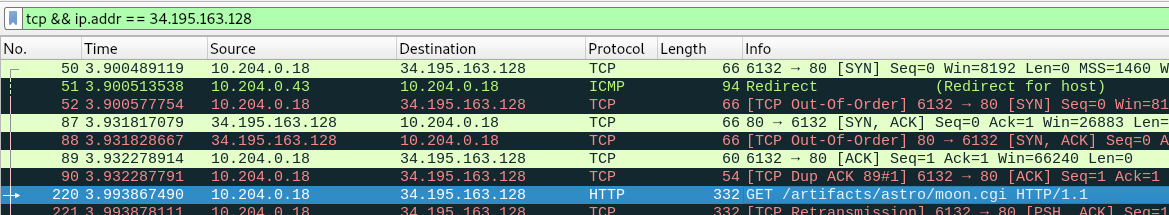


BH:

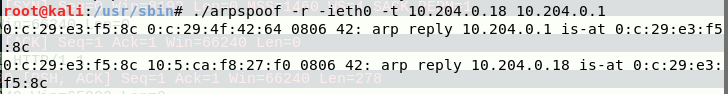
* **Did you see all traffic from/to the target in Wireshark? Provide a screenshot of your Wireshark screen including the beginning of the TCP 3-way handshake.**
  + We saw all traffic **from** the target in Wireshark. However, we did not see all traffic **to** the target. As shown below, there are no packets from 34.195.163.128 to the target computer

****

* **If you do not see all traffic, explain why and then modify the attack setup to see all traffic. Once again, provide a screenshot of your Wireshark screen including the beginning of the TCP 3-way handshake.**
  + The reason we did not see **all** traffic is that the only the target machine has been affected with an incorrect value of the default gateway MAC address. Thus, any traffic from the target to the outside world will go through the BH; however, any traffic from the outside world to the target will reach the target without going through the BH’s machine.  
    **Note:** after terminating the arpspoof session, the program “fixed” the target’s machine and sent it the proper MAC address for the default gateway.
  + The below screenshot shows the Wireshark capture of the TCP 3-way handshake.

****

* We added –r to the arpspoof command to also receive replies.

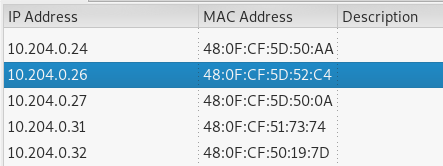
****

**Task 2. Ettercap - ARP Cache Poisoning et al.**

Blackhat:

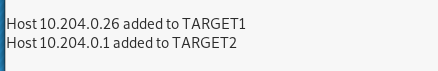
* Open Ettercap
* Click Sniff 🡪 Unified. Select your NIC in the popup box.
* Click Hosts 🡪 Scan for hosts
* Click Hosts 🡪 Hosts List
* In the hosts list, highlight the target host (your partner’s machine) and click the button “Add to Target1”

**Target Host:**



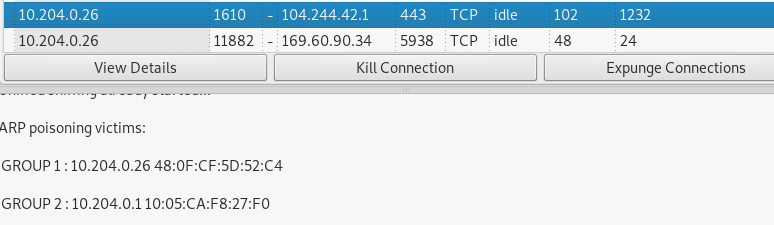
* Highlight the gateway IP and click “Add to Target2”

**Both targets added:**



* Click View 🡪 Connections (or press Shift+C)
* Click Start 🡪 Start sniffing
* In the Connections tab, notice you only see connections made to your IP or broadcast or multicast addresses.
* Now start the ARP poisoning: Mitm 🡪 ARP poisoning… 🡪 Sniff Remote Connections

**ARP Poison started:**



* Scroll down to the victim’s IP address to see the connections.
* You may wish to filter on just active TCP connections by unchecking all but “TCP” and “Active” in the “Filter” row at the top of the host list. You can also filter on the target’s IP address by entering the IP address in the Host filter field.
* Open Wireshark and sniff the Ethernet interface.

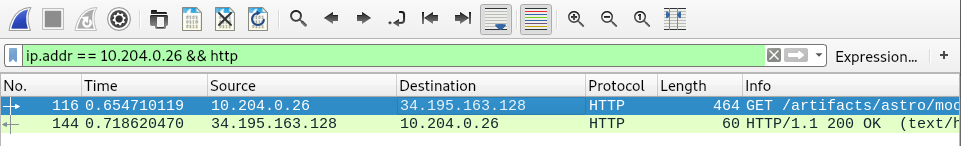
Target:

* Surf to our favorite website:   
   **www.briancasey.org/artifacts/astro/moon.cgi**

Blackhat:

* In Wireshark, stop sniffing.
  + Find the initial GET request to the webserver and use Follow 🡪 TCP Stream to display the traffic.

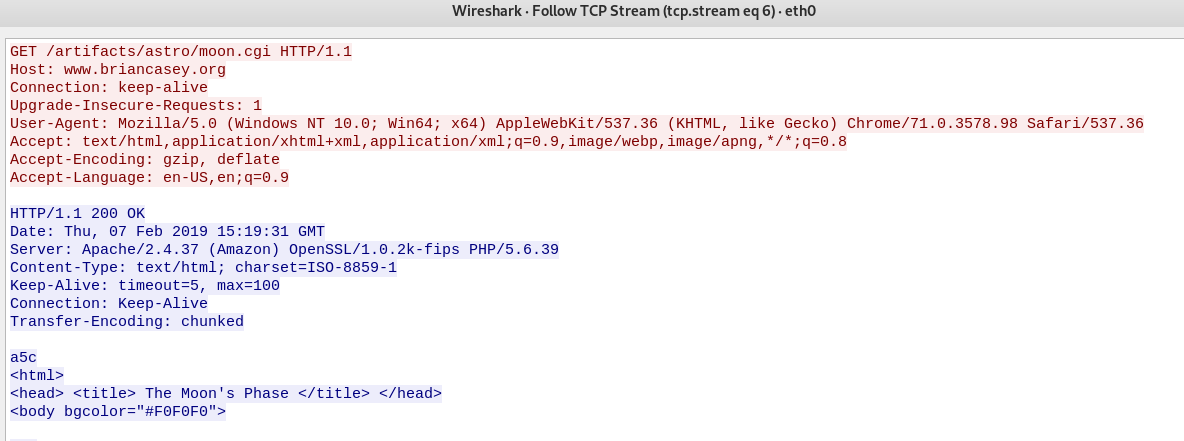
**Initial GET Request:**

****

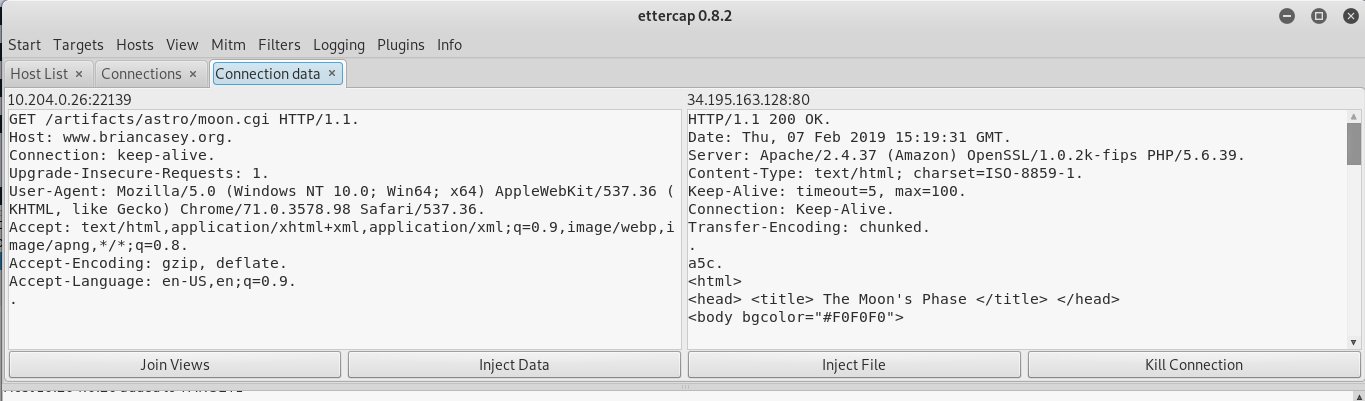
* In Ettercap, verify you see the connections on the connections screen. Double-click on the connection to see the connection data. It should match the Wireshark data.
* **Provide a screenshot of this HTTP traffic in Ettercap and Wireshark.**

Below are the two screenshots showing the matching HTTP traffic from Wireshark and Ettercap

**Wireshark Data:**



**Ettercap Data:**



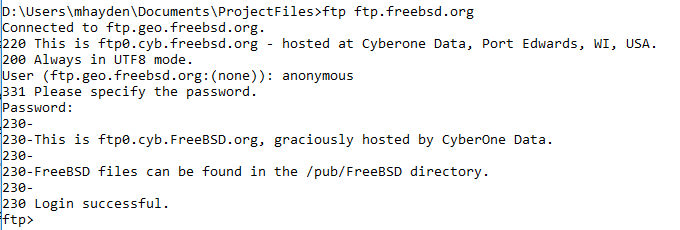
Target:

* Now open a command window and enter at the prompt:

**ftp** [**ftp.freebsd.org**](ftp://ftp.freebsd.org)

* Enter **anonymous** as the user and whatever you like as the password.

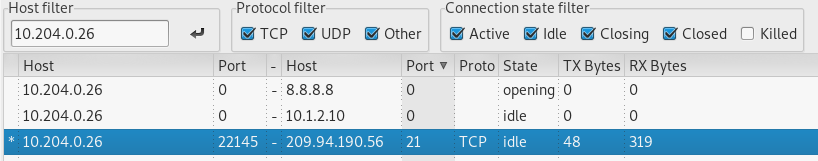
The below screenshot shows the successful opening of the FTP session using **anonymous** as the user:

****

Blackhat:

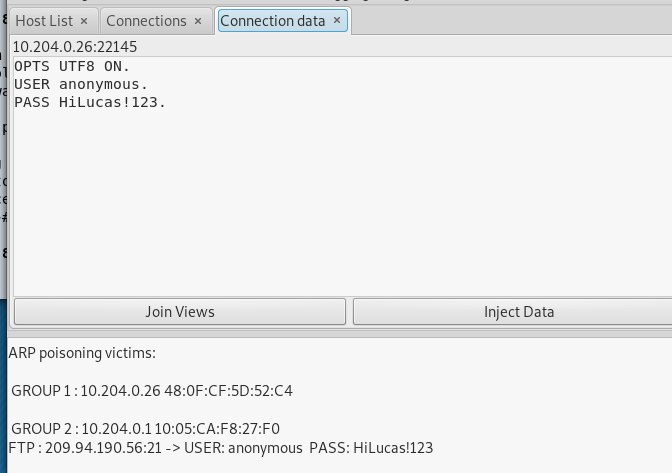
* **Did you see the connection pop up on the connections tab?**

Yes, the BH saw the connection pop up in the connections tab.



* **Double click on this connection to see the connection data. Depending on how long you wait, you may have to check the Idle, Closing, or Closed boxes to see this FTP connection.**
* **Did you also see the username and password displayed in the bottom window?**

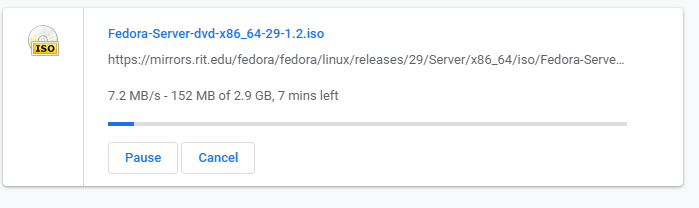
The below screenshot shows the connection data, as well as the username/password also displayed in the bottom window.

****

Target:

* You will attempt to download the latest version of Fedora. Surf to   
   **https://getfedora.org/en/server/download/**  
  and click the Download button. This will start the download process.

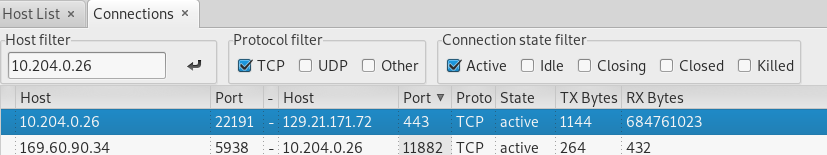
The below screenshot shows the download starting on the Target browser:



Blackhat:

* Watch the connection being made in the connections window.
* After a few seconds, attempt to kill the connection by right-clicking on the connection and selecting **Kill Connection**. Note: you may not be able to actually kill the connection, but the target should notice significant network latency.

The BH killed the connection by right clicking the highlighted connection and selecting “Kill Connection”.



Target:

* **Did you notice the connection being dropped or interrupted?** **How do you know the connection was dropped/interrupted?**

Yes, the target noticed the connection being dropped. We know the connection was dropped/interrupted because the browser cancelled the download due to a “Network Error”. The below screenshot shows the download immediately after the BH killed the connection.



* Stop the download.

Blackhat:

* Stop the ARP poisoning: Mitm 🡪 Stop Mitm attack(s)
* This allows Ettercap to Re-ARP the victim so connections to the Internet will still work.
* Close Ettercap

The below screenshot shows the BH Re-ARPing the victim:

**D:\Users\mhayden\Documents\ProjectFiles\Lab6\Screenshots\T2_BH_re-arp.png**

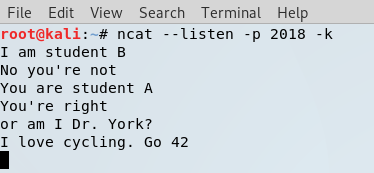
**Task 3. Ncat**

This assignment requires you to use ncat in several different roles to learn this tool’s utility.

3.1 Ncat Conversation (This is not the chat option.)

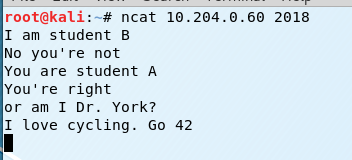
Student A:

* Start an ncat listener.



Student B:

* Connect to the listener using an ncat client.

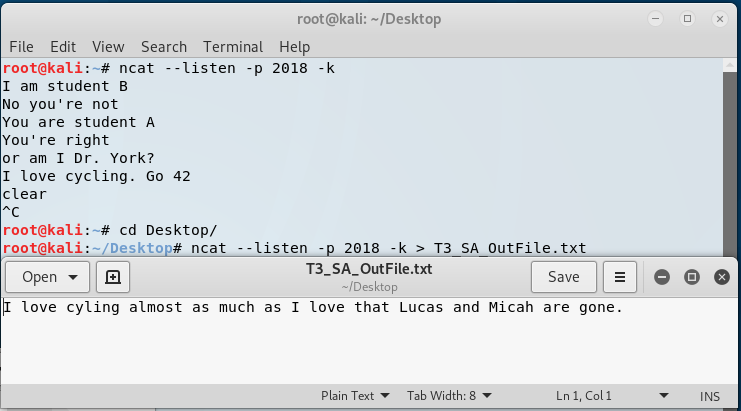


Type messages back and forth to each other. **Provide a screenshot of both ncat windows.**

3.2 Push a file

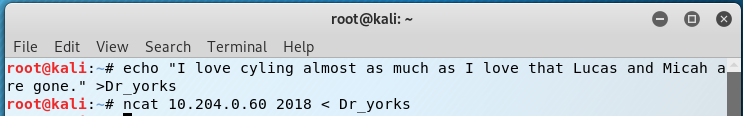
Student A:

* Start an ncat listener and redirect your output (>) to a file.



Student B:

* Create a short file (e.g., **echo This is a short file. > filename1**).
* Connect to the listener using an ncat client and redirect the file (<) into the session.



Student A:

* Verify the file was transferred to your machine.

**Provide a screenshot of both ncat windows.**

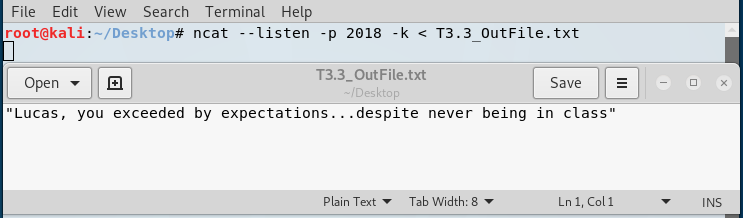
The Student A screenshot shows the initialization of the NCAT listener, as well as the output file after Student B pushed the file.

The Student B screenshot shows the generation of the Dr\_yorks file, as well as pushing it to the ncat listener.

3.3 Pull a file

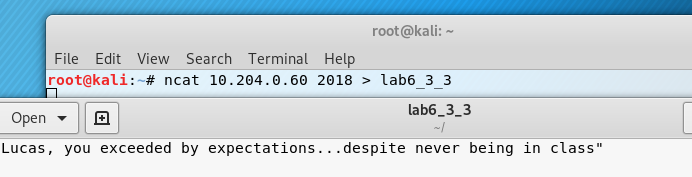
Student A:

* Create a different short file (e.g., **echo This is another short file. > filename2**).
* Start an ncat listener and direct your short file (<) as the input.



Student B:

* Connect to the listener using an ncat client and redirect the output (>) into a file.



Student B:

* Verify the file was transferred to your machine.

**Provide a screenshot of both ncat windows.**

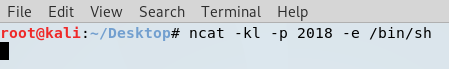
The Student A screenshot shows the contents of the T3.3 output file used for the ncat listener.

The Student B screenshot shows the ncat output directed into the “lab6\_3\_3” file, as well as its contents after the file transfer.

3.4 Passive backdoor command shell

Student A:

* Start an ncat listener so anyone connecting to your machine on the port you specify will be provided a shell.



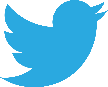
Student B:

* Connect to the listener using an ncat client.
* **Did you receive a command shell from Student A’s listener?**
* Type **hostname** and into the remote shell. **Did the shell return the host name of Student A’s computer?**

**Provide a screenshot of both ncat windows.**

The below screenshot shows Student B connecting to the listener, and using it to execute the **hostname** command. The shell returned the name “kali”.



[](https://twitter.com/DrEvilM4I)**Task 4. Proxy Chains**

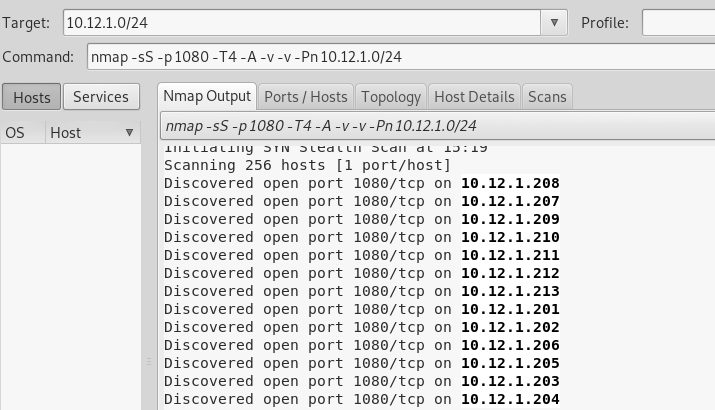
**Do not share your answers to this question with other teams.**

Using techniques you’ve seen in class, you will attempt to learn the contents of a flag file on a computer owned by Mullins Movies, Music, and Machines Inc. (M4I) using proxy chains. Your reconnaissance indicates there is a SOCKS4 proxy in the IP range 10.12.1.0/24. There is also a target in the IP range 10.12.1.0/24 that is running a SSH server on port 22222. Only use the proxy and target assigned to your team; you may not be able to learn every computers name, so use the IP address to determine your machines. You may not exploit these machines or change file/folder permissions.

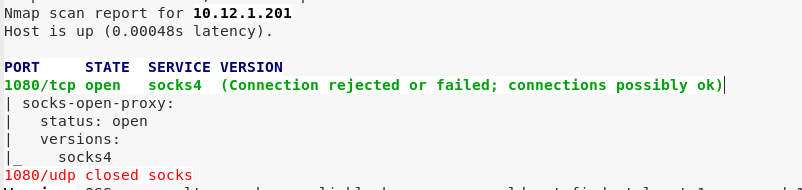
The following questions will help guide your exploitation. Besides describing how you would complete a task, provide the exact commands you used as well as screenshots of your results for each step. There are multiple identical copies of the target machine with different IP addresses.

a. How did you learn the IP address of your team’s SOCKS4 proxy server?

* We conducted an nmap scan of the subnet 10.12.1.0/24 using the below nmap command, outlined in red. Port 1080 was the port scanned because that is the port which SOCKS4 utilizes.

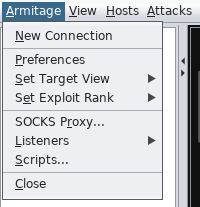


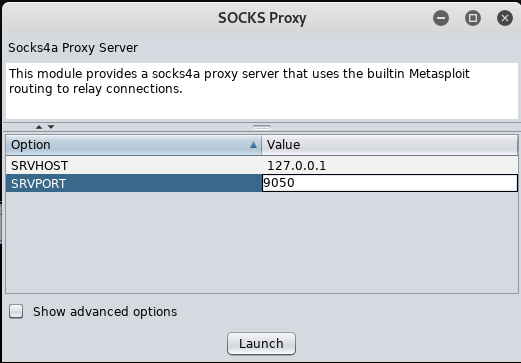
* Because we are Team 1, we know 10.12.1.201 (the lowest IP) is our team’s proxy:



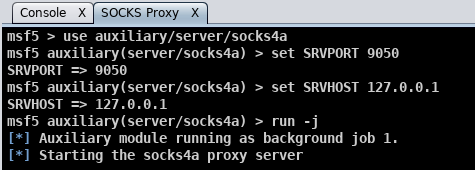
b. How did you learn the IP address of your team’s target?

* We established a local proxy on IP 127.0.0.1 and port 9050 using Armitage. In the Armitage window select Armitage🡪SOCKS Proxy using the below options:

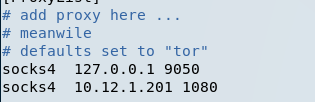




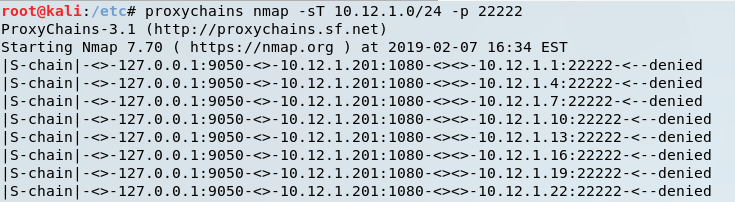
* Below is the SOCKS Proxy output in the Armitage tab:



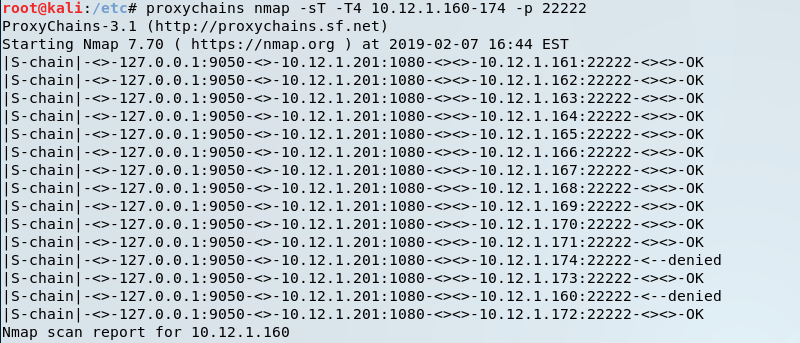
* We then updated the /etc/proxychains.conf file with the below additions. This sets the proxychain to go through the local proxy and then the target proxy at 10.12.1.201 with ports 9050 and 1080, respectively.



* Once the proxy chain was established, we ran an nmap scan of the subnet 10.12.1.0/24 looking for an open port 22222, using the proxy chain. This command is shown below:

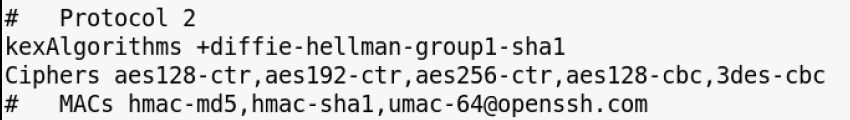


* We saw a group of open ports on IPs in the range 10.12.1.160-170, but they were not sequential, so we ran another nmap scan on a smaller list of IPs to verify the proper target IP:

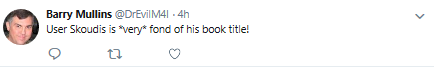


c. How did you establish a connection to the target? Explain all steps and how you may have changed configuration files. Draw a network diagram illustrating how your connection was made. In other words, your diagram should include intermediate proxies as well as the target. Do not forget about your localhost. Your diagram cannot be hand drawn—PowerPoint or Visio work well.

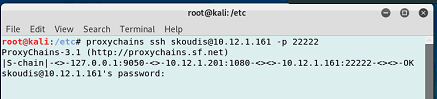
Hint: If you use SSH on Kali, your ssh connection will likely fail due to handshaking problems. Edit **/etc/ssh/ssh\_config** and uncomment the **Ciphers** line and add the following as shown below: **kexAlgorithms +diffie-hellman-group1-sha1**.



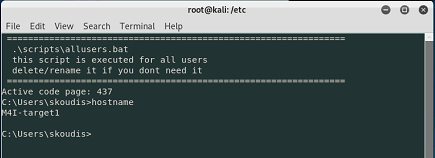
* Edited the /etc/ssh/ssh\_config file using the above changes given in the Hint.
* To ssh into the target, we used the user “skoudis” and password “counterhackreloaded” based on the following reconnaissance shown below:



* Using the established local proxy and the proxy setup described in part B (simply don’t close the local proxy in Armitage, or reopen it using the steps outlined in part B), connect using ssh using the below command:

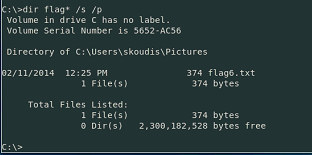


* The following ssh window demonstrated a successful connection on the host M4I-target1:



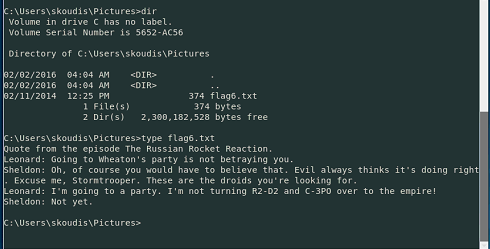
d. How did you search a computer for a specific file name?

* From the C: directory of the target computer, we executed “dir flag\* /s /p” to search for all files with flag. The result found flag6.txt in the C:\Users\skoudis\Pictures directory.



e. How did you display the contents of a file?

* We navigated to the C:\Users\skoudis\Pictures directory, and executed “type flag6.txt”, which output the file’s contents into the command shell.



f. If you’ve made it this far, you can now answer the burning question… what is the message in the file?

* The message is shown in the screenshot above, outlined in blue.

**Task 5. Attacking Wireless**

**Do not share your answers to this question with other teams.**

This assignment requires you to attack two wireless networks. Provide exact instructions and tools used with configuration information.

You are only required to use the Alfa card.

BONUS: For up to 15 bonus points use the AirPcap card to find the WEP (5.1) and WPA (5.2) credentials. If you use the AirPcap card, you must show all steps required to successfully load drivers.

One team member needs to stop by Dr. Mullins’ office to sign out an Alfa card and optionally an AirPcap adapter.

Do not leave your wireless adapters plugged in your computer while not actively using them. You may DoS our network. Trust me, it happens.

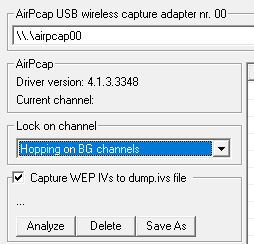
Scoping: You discovered that IP addresses 150-250 are assigned using DHCP, so you should not scan, attempt to attack, or interact with these machines.

5.1 WEP Network

**Provide all steps required to learn the WEP key. Include a screenshot of the tool displaying the SSID and cracked WEP key.**

5.1B WEP Network using AirPCAP card

* Open Cain and navigate to the Wireless tab.
* Select the “AirPcap” device from the devices list, and run a “passive scan” using the below settings:

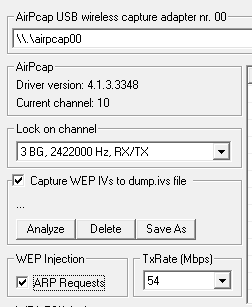


* Stop the scan after finding the M4I1 and M4I2 networks, shown below. Outlined in red are the channels for both networks – channel 3 for M4I1 and channel 9 for M4I2.

D:\Users\mhayden\Documents\ProjectFiles\Lab6\Screenshots\T5_AirPcap_FoundM4I1.PNG

D:\Users\mhayden\Documents\ProjectFiles\Lab6\Screenshots\T5_AirPcap_FoundM4I2.PNG

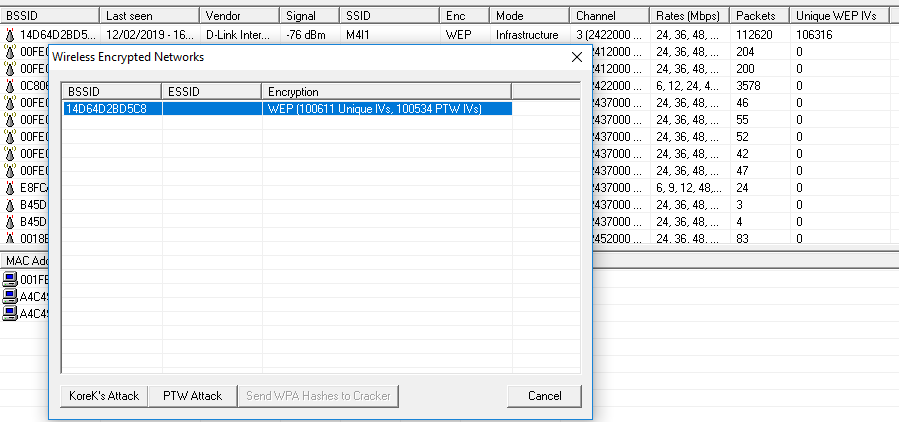
* Using the following settings, run a scan of channel 3, collecting the WEP IVs for M4I1.



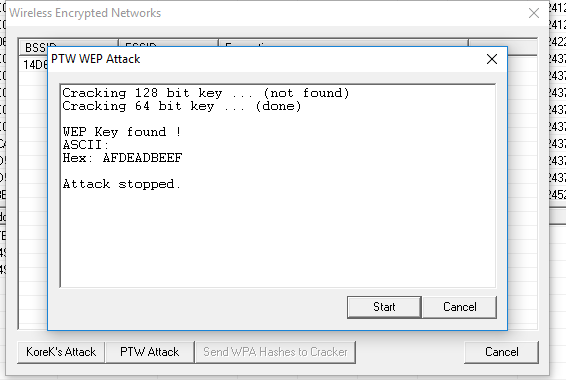
* Stop the scan once there are enough IVs for the desired network. Shown below is the output from our scan, showing 106316 IVs for M4I1.

D:\Users\mhayden\Documents\ProjectFiles\Lab6\Screenshots\T5_AirPcap_WEP_IVsFound.PNG

* Select the M4I1 network, Click Analyze 🡪 PTW Attack, using 64 bit as WEP key length.



* The output is shown below: “AFDEADBEEF”



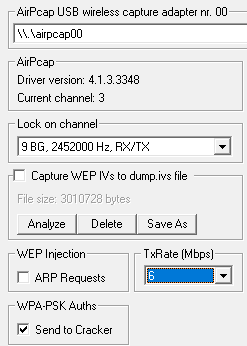
5.2 WPA Network

5.2B WPA Network with AirPcap

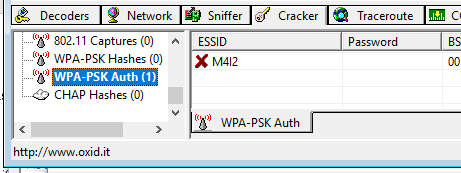
* Note that M4I2 operates on Channel 9, as shown in the screenshot below:

D:\Users\mhayden\Documents\ProjectFiles\Lab6\Screenshots\T5_AirPcap_FoundM4I2.PNG

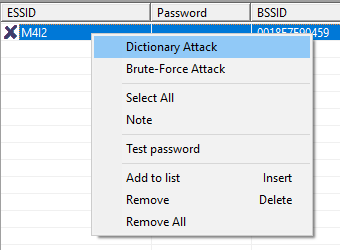
* Run a Passive Scan on Channel 9 using the settings shown below. These settings also send the WPA-PSK Authorizations to the Cracker window in Cain.

****

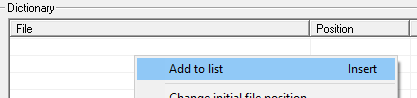
* Stop the scan once the Cracker window shows a detected handshake, as shown below.



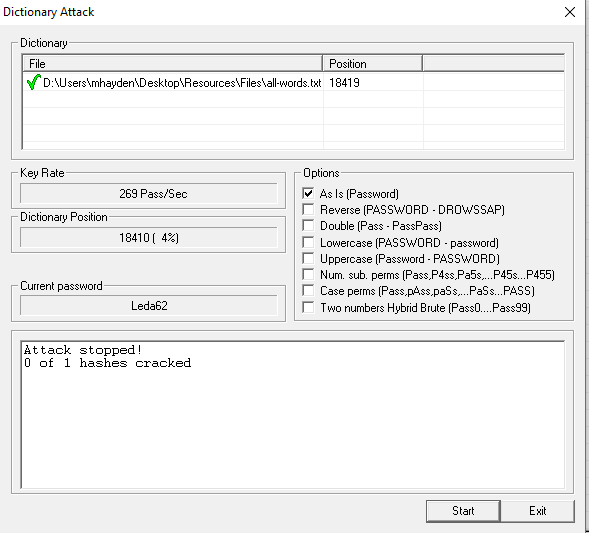
* Right click on M4I2 and select Dictionary Attack



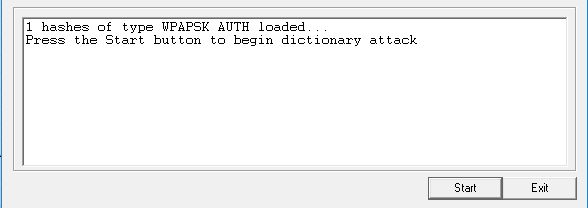
* We added the all-words.txt to the list of dictionaries by right clicking in the dictionary space, and selecting “Add to list” as shown below.



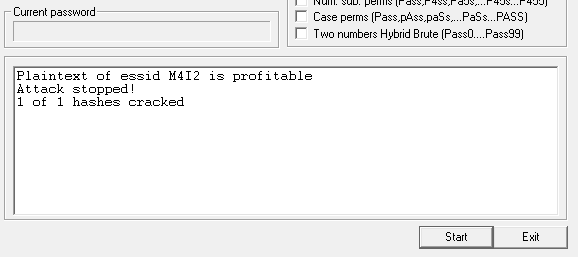
* We used the dictionary attack using the below settings which removed all permutations given the below hint stating that “the passphrase matches the dictionary word exactly”.



* Select Start to begin the Dictionary Attack:



* Use the provided password “profitable” to connect to the network:



**Provide all steps required to learn the WPA passphrase. Include a screenshot of the tool displaying the SSID and cracked WPA passphrase.** The word list used in lab 4 (all-words.txt) is still useful, and I happen to know the passphrase matches the dictionary word exactly (no permutations required).

**General Observations**

How long did it take you to complete this lab?

Was it an appropriate length lab?

What corrections and or improvements do you suggest for this lab? Please be very specific, and if you add new material, provide the exact wording and instructions you would give to future students in the new lab handout. You may cross out and edit the text of the lab on previous pages to make minor corrections/suggestions.