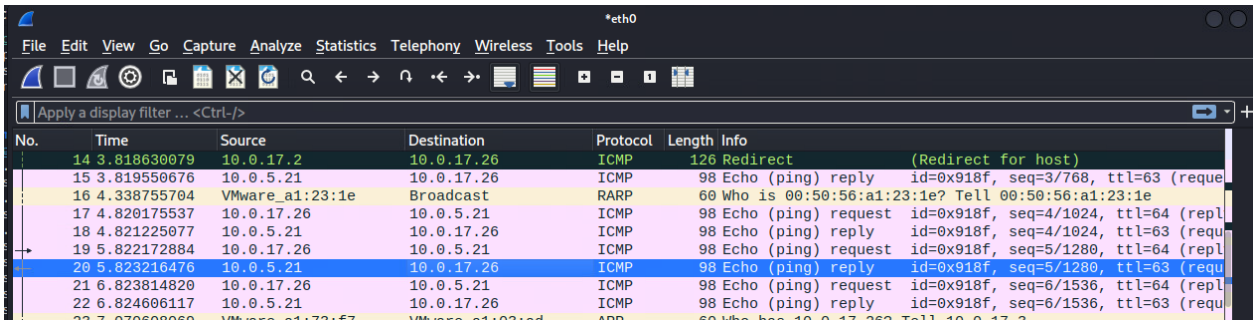
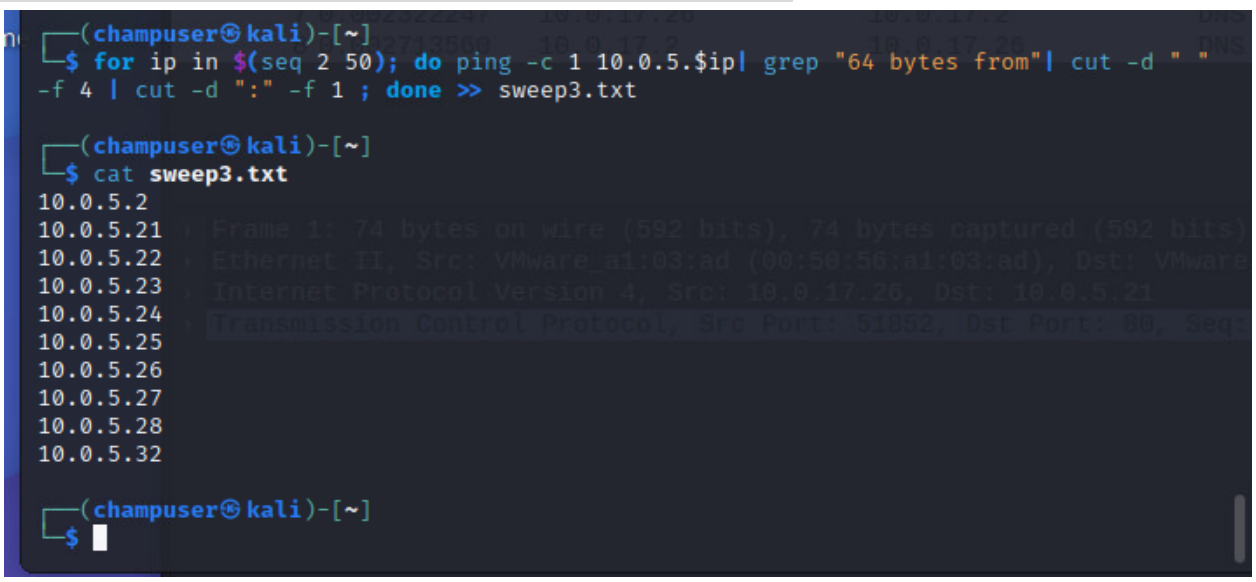
Max Gallagher Class Activity 2.1 Host Discovery
David Thomsen

Deliverable 1. Provide a screenshot similar to the one below that shows 1 outbound ping and the captured request and reply.



No.	Time	Source	Destination	Protocol	Length	Info
14	3.818630079	10.0.17.2	10.0.17.26	ICMP	126	Redirect (Redirect for host)
15	3.819550676	10.0.5.21	10.0.17.26	ICMP	98	Echo (ping) reply id=0x918f, seq=3/768, ttl=63 (reque
16	4.338755704	VMware_a1:23:1e	Broadcast	RARP	60	Who is 00:50:56:a1:23:1e? Tell 00:50:56:a1:23:1e
17	4.820175537	10.0.17.26	10.0.5.21	ICMP	98	Echo (ping) request id=0x918f, seq=4/1024, ttl=64 (repl
18	4.821225077	10.0.5.21	10.0.17.26	ICMP	98	Echo (ping) reply id=0x918f, seq=4/1024, ttl=63 (requ
19	5.822172884	10.0.17.26	10.0.5.21	ICMP	98	Echo (ping) request id=0x918f, seq=5/1280, ttl=64 (repl
20	5.823216476	10.0.5.21	10.0.17.26	ICMP	98	Echo (ping) reply id=0x918f, seq=5/1280, ttl=63 (requ
21	6.823814820	10.0.17.26	10.0.5.21	ICMP	98	Echo (ping) request id=0x918f, seq=6/1536, ttl=64 (repl
22	6.824606117	10.0.5.21	10.0.17.26	ICMP	98	Echo (ping) reply id=0x918f, seq=6/1536, ttl=63 (requ

Deliverable 2. Collaborate with your teammates from Module 1 to write either a bash script or or one liner to ping ip's in the range of 10.0.5.2 - 10.0.5.50 your script should output a list of "up ip addresses" into a file called sweep.txt. Submit a screenshot similar to the redacted one below that shows either your 1 liner command or source code, followed by a cat of sweep.txt.

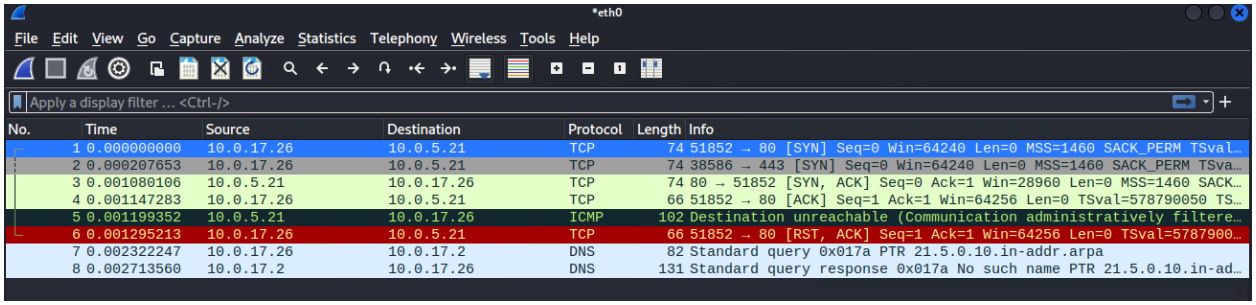


Deliverable 3. Now, do the same thing with fping. Investigate the switches that allow you to provide a range of ip addresses as well as reporting the "up" hosts. You may need to throw out error messages. Provide a screenshot similar to the one below.

```
┌──(champuser㉿kali)-[~]
└─$ fping -s -g 10.0.5.2 10.0.5.50 -r 1 | grep "alive" >>sweep2.txt
ICMP Redirect from 10.0.17.2 for ICMP Echo sent to 10.0.5.13

      49 targets
      10 alive
      39 unreachable
       0 unknown addresses

      78 timeouts (waiting for response)
      88 ICMP Echos sent
      10 ICMP Echo Replies received
       1 other ICMP received

 0.724 ms (min round trip time)
 0.947 ms (avg round trip time)
 1.18 ms (max round trip time)
        1.749 sec (elapsed real time)
```

```
┌──(champuser㉿kali)-[~]
└─$ cat sweep2.txt
10.0.5.2 is alive
10.0.5.21 is alive
10.0.5.22 is alive
10.0.5.23 is alive
10.0.5.24 is alive
10.0.5.25 is alive
10.0.5.26 is alive
10.0.5.27 is alive
10.0.5.28 is alive
10.0.5.32 is alive
```

Deliverable 4. Use nmap's -sn switch to scan 10.0.5.21, it should report that it is up. Execute nmap with this exercise. Capture traffic on eth0 using Wireshark. Provide a screenshot of your wireshark output.




```
┌──(champuser㉿kali)-[~]
└─$ nmap -sn 10.0.5.21
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-23 18:50 EST
Nmap scan report for 10.0.5.21
Host is up (0.0014s latency).
Nmap done: 1 IP address (1 host up) scanned in 0.01 seconds
```

Deliverable 5. Closely examine What destination ports and protocols were used in the use case? What observations do you have when comparing this to the ping and fping tests?

Fping uses ICMP in order to ping the other devices, where the -sn nmap scan creates a TCP connection and uses TCP as the protocol.

Deliverable 6. Write a bash one liner or script that conducts an nmap -sn scan of 10.0.5.2-50 and outputs the list of ip addresses to sweep.txt similarly to the code written for ping and fping. Take a screenshot that shows the execution and output.

```
┌──(champuser㉿kali)-[~]
└─$ nmap -sn 10.0.5.2-50 | grep for | cut -d ' ' -f 5 >sweep3.txt

┌──(champuser㉿kali)-[~]
└─$ cat sweep3.txt
10.0.5.2
10.0.5.21
10.0.5.22
10.0.5.23
10.0.5.24
10.0.5.25
10.0.5.26
10.0.5.27
10.0.5.28
10.0.5.31
10.0.5.32
```