# Lab 2: SimSpace

Student Name: Raymond Ng

IS 3513 Summer 2023

Instructor Name: Dr. William Pugh

June 26, 2023

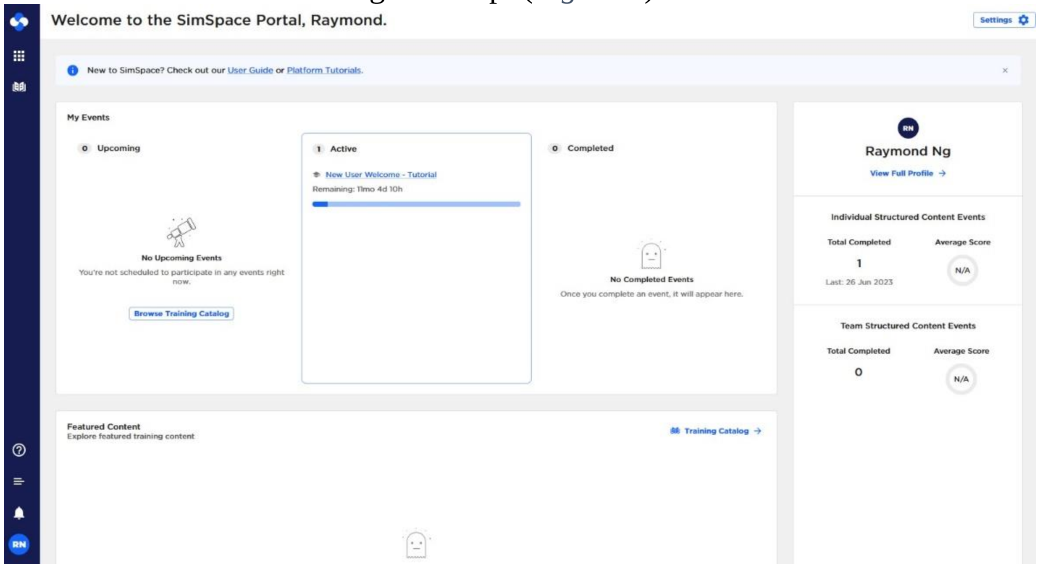
## Introduction

This lab aims to introduce and familiarize the user with the use of SimSpace. Additionally, the lab will explore the use of the ***nmap*** utility for network discovery and security auditing via virtual machine (VM). Further, the user will experiment and demonstrate the use of applications useful for a security practitioner via VM.

## Procedure

### Step 1

To begin the lab, I created a user account via SimSpace Portal. Below is a screenshot of a SimSpace Portal after a successful login attempt (*Figure 1*).



New to SimSpace? Check out our User Guide or Platform Tutorials.

You're not scheduled to participate in any events right now.

Once you complete an event, it will appear here.

*Figure 1: SimSpace Portal.*

### Step 2

I accessed the User Guide by clicking in the lower left corner of Sim Space Portal, where my initials were located (*Figure 2*):

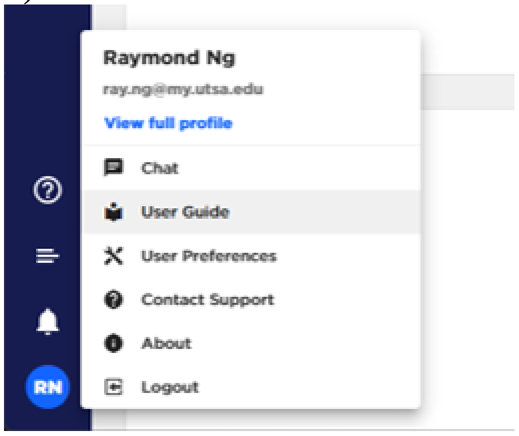
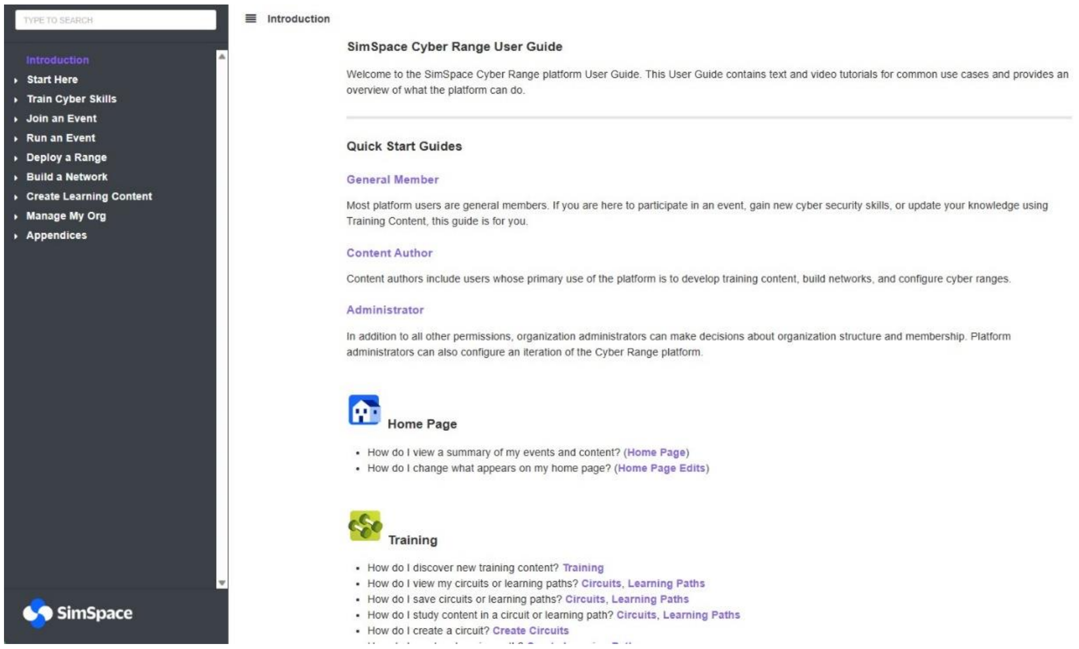


*Figure 2: Accessing SimSpace User Guide.*

After accessing the *User Guide*, I reviewed the content of the *SimSpace Cyber Range User Guide* (*Figure 3*).



SimSpace Cyber Range User Guide

Welcome to the SimSpace Cyber Range platform User Guide. This User Guide contains text and video tutorials for common use cases and provides an overview of what the platform can do.

Quick Start Guides

General Member

Most platform users are general members. If you are here to participate in an event, gain new cyber security skills, or update your knowledge using Training Content, this guide is for you.

Content Author

Content authors include users whose primary use of the platform is to develop training content, build networks, and configure cyber ranges.

Administrator

In addition to all other permissions, organization administrators can make decisions about organization structure and membership. Platform administrators can also configure an iteration of the Cyber Range platform.

Home Page

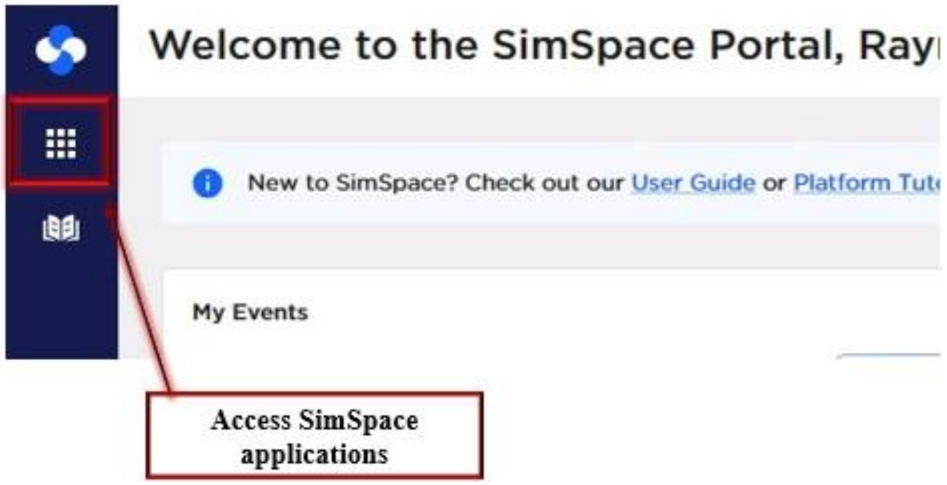
- How do I view a summary of my events and content? (Home Page)
- How do I change what appears on my home page? (Home Page Edits)

Training

- How do I discover new training content? Training
- How do I view my circuits or learning paths? Circuits, Learning Paths
- How do I save circuits or learning paths? Circuits, Learning Paths
- How do I study content in a circuit or learning path? Circuits, Learning Paths
- How do I create a circuit? Create Circuits

*Figure 3: SimSpace Cyber Range User Guide.*

<u>Step 3</u>
To access the 'Events' application, I clicked on the application's icon underneath the SimSpace logo in the upper left corner (*Figure 4*).



Welcome to the SimSpace Portal, Ray
New to SimSpace? Check out our User Guide or Platform Tut
My Events
Access SimSpace applications

*Figure 4:Accessing SimSpace applications.*

Next, I clicked 'Events' (*Figure 5*), which took me to the Structured Content Event Outlook (*Figure 6*).

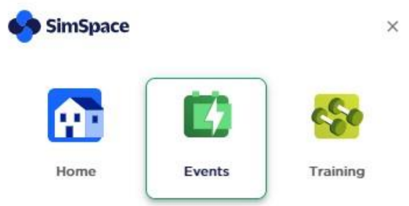
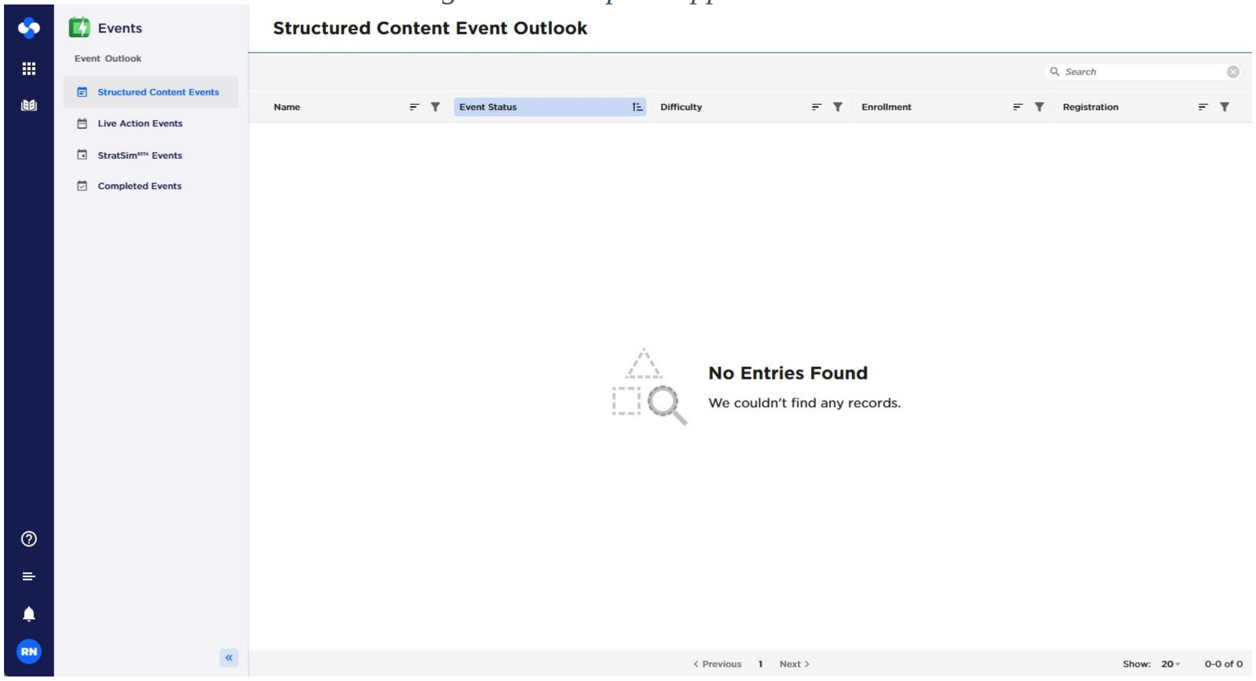


*Figure 5: SimSpace applications.*



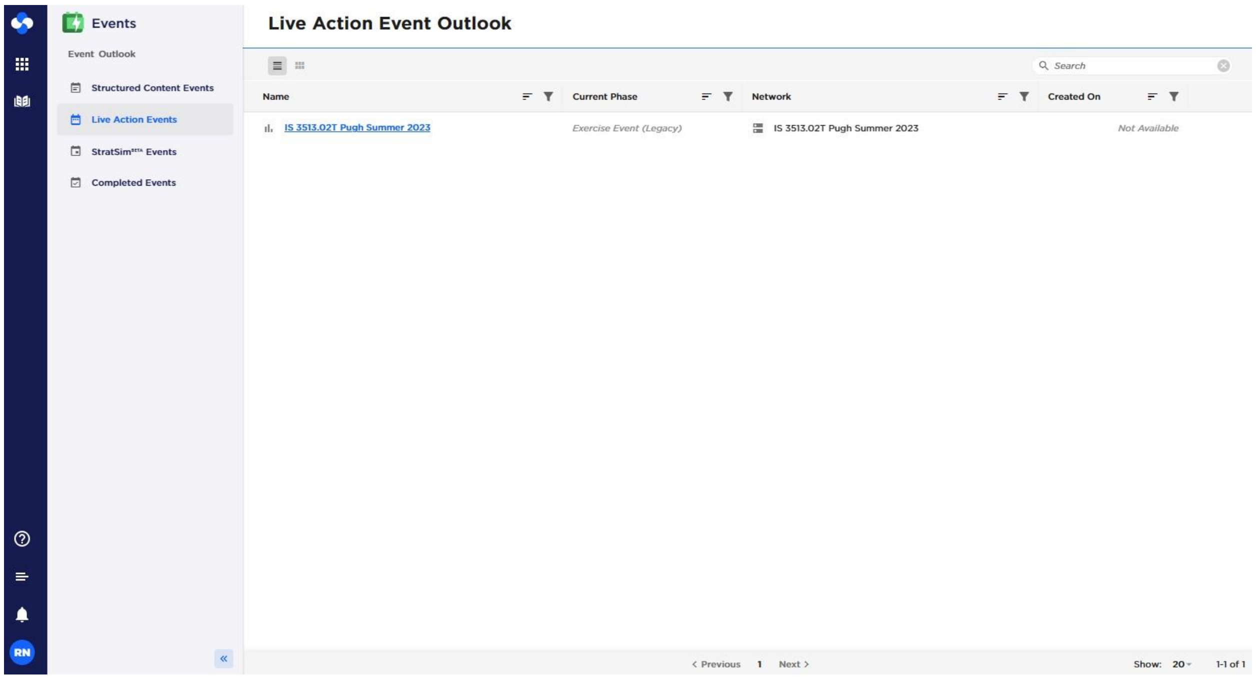
Structured Content Event Outlook

No Entries Found

We couldn't find any records.

*Figure 6: Structured Content Event Outlook.*

Next, I accessed the 'Live Action Events' and observed only one option/range, **IS 3513.02T Pugh Summer 2023**, provided by Dr. Pugh (*Figure 7*).



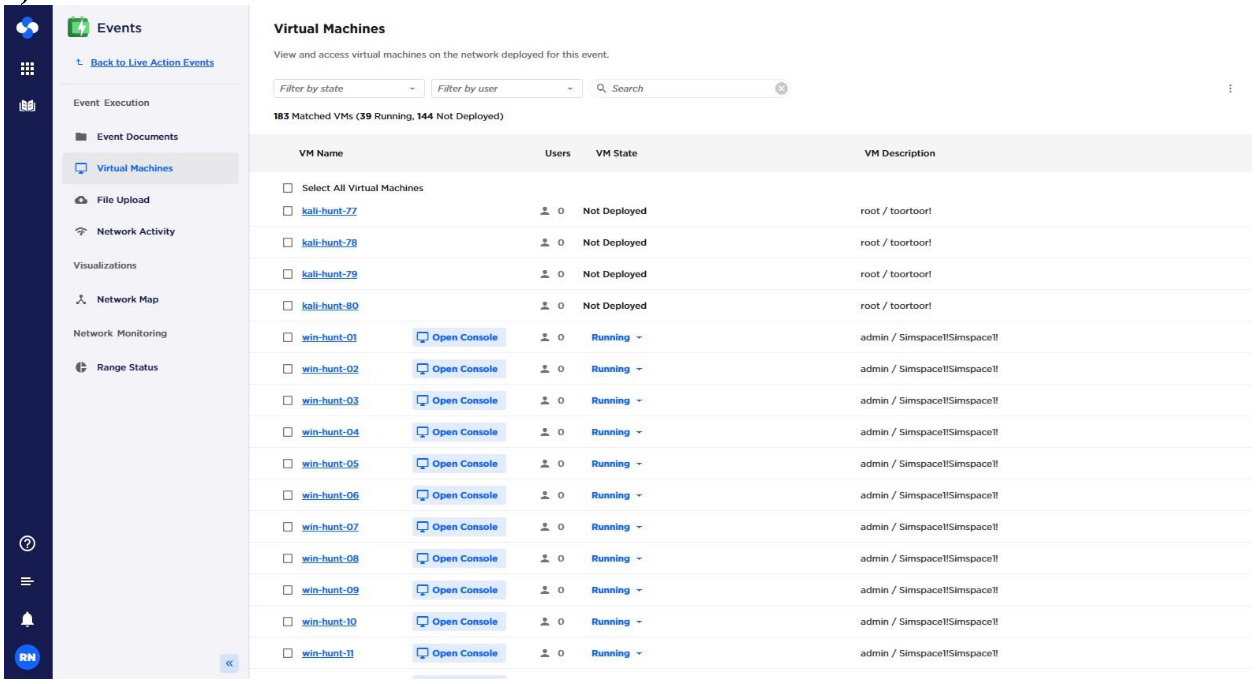
Live Action Event Outlook

| Name | Current Phase | Network | Created On |
|---|---|---|---|
| IS 3513.02T Pugh Summer 2023 | Exercise Event (Legacy) | IS 3513.02T Pugh Summer 2023 | Not Available |

Previous 1 Next   Show: 20   1-1 of 1

*Figure 7: Accessing 'Live Action Events' via SimSpace.*

Next, I clicked **IS 3513.02T Pugh Summer 2023**, from the previous screen (*Figure 7*), and explored the 'Virtual Machines' and 'Network Map' options/ranges. Additionally, I observed the username/passwords associated with each VM, underneath the 'VM Description' section (*Figure 8*).



Virtual Machines

View and access virtual machines on the network deployed for this event.

183 Matched VMs (39 Running, 144 Not Deployed)

| VM Name | | Users | VM State | VM Description |
|---|---|---|---|---|
| kali-hunt-77 | | 0 | Not Deployed | root / toortoor! |
| kali-hunt-78 | | 0 | Not Deployed | root / toortoor! |
| kali-hunt-79 | | 0 | Not Deployed | root / toortoor! |
| kali-hunt-80 | | 0 | Not Deployed | root / toortoor! |
| win-hunt-01 | Open Console | 0 | Running | admin / Simspace1!Simspace1! |
| win-hunt-02 | Open Console | 0 | Running | admin / Simspace1!Simspace1! |
| win-hunt-03 | Open Console | 0 | Running | admin / Simspace1!Simspace1! |
| win-hunt-04 | Open Console | 0 | Running | admin / Simspace1!Simspace1! |
| win-hunt-05 | Open Console | 0 | Running | admin / Simspace1!Simspace1! |
| win-hunt-06 | Open Console | 0 | Running | admin / Simspace1!Simspace1! |
| win-hunt-07 | Open Console | 0 | Running | admin / Simspace1!Simspace1! |
| win-hunt-08 | Open Console | 0 | Running | admin / Simspace1!Simspace1! |
| win-hunt-09 | Open Console | 0 | Running | admin / Simspace1!Simspace1! |
| win-hunt-10 | Open Console | 0 | Running | admin / Simspace1!Simspace1! |
| win-hunt-11 | Open Console | 0 | Running | admin / Simspace1!Simspace1! |

*Figure 8: Observing options/ranges in 'Live Action Events' via SimSpace.*

Step 4

Per the lab instructions, I was directed to access one of the win-hunt-xx VMs. I observed no Users in the win-hunt-01 VM so I opened the console (*Figure 9*).

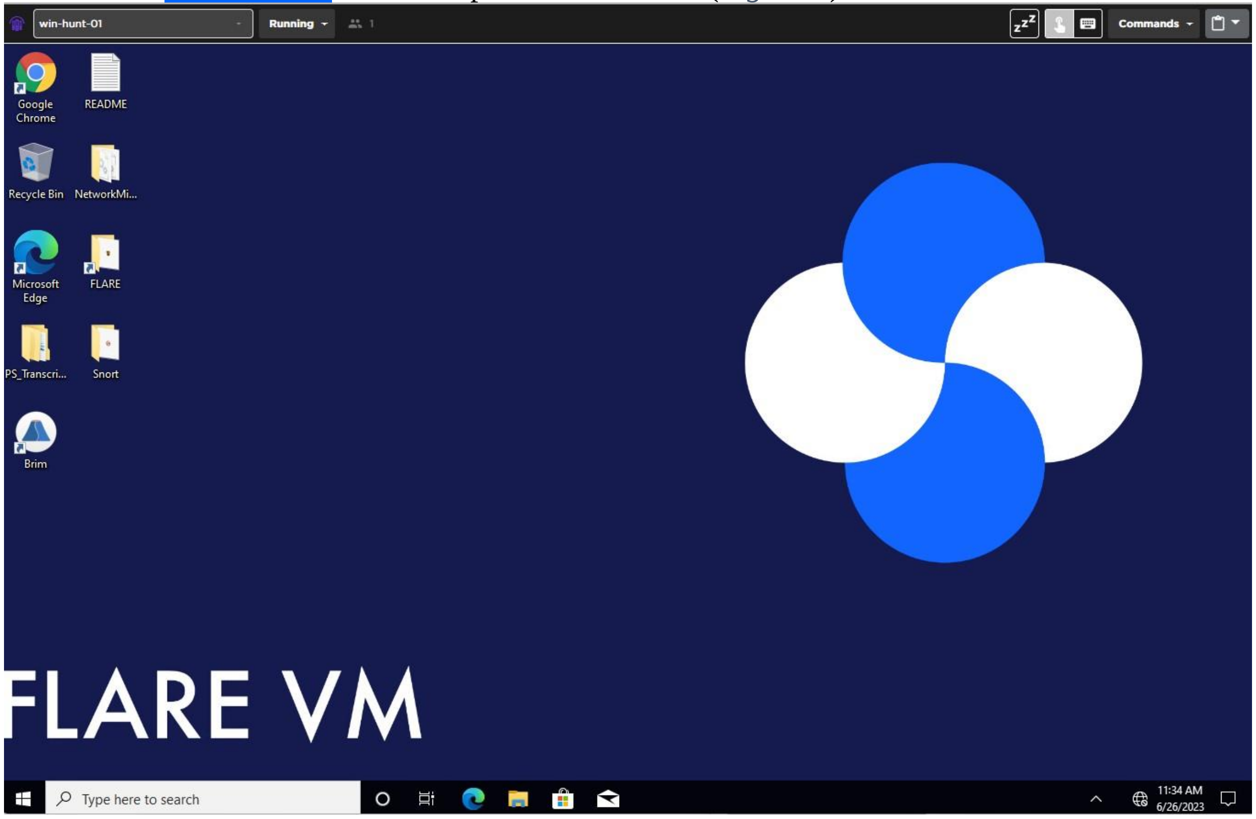
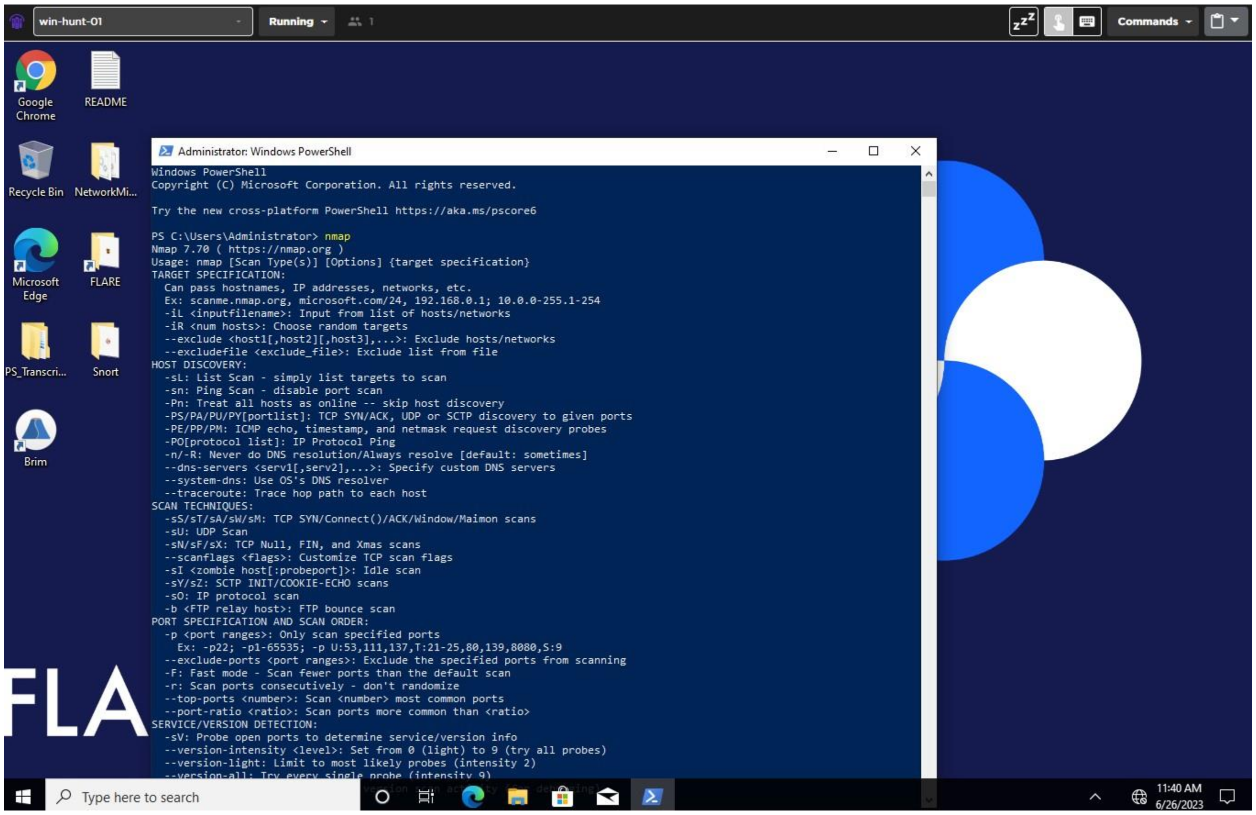


*Figure 9: Opened* win-hunt-01 *VM via SimSpace.*

Step 5
Per the lab's instructions, I opened Windows PowerShell via Hunt image (win-hunt-01) by searching for PowerShell in the VM. Next, I executed ***nmap*** via PowerShell. (*Figure 10*)

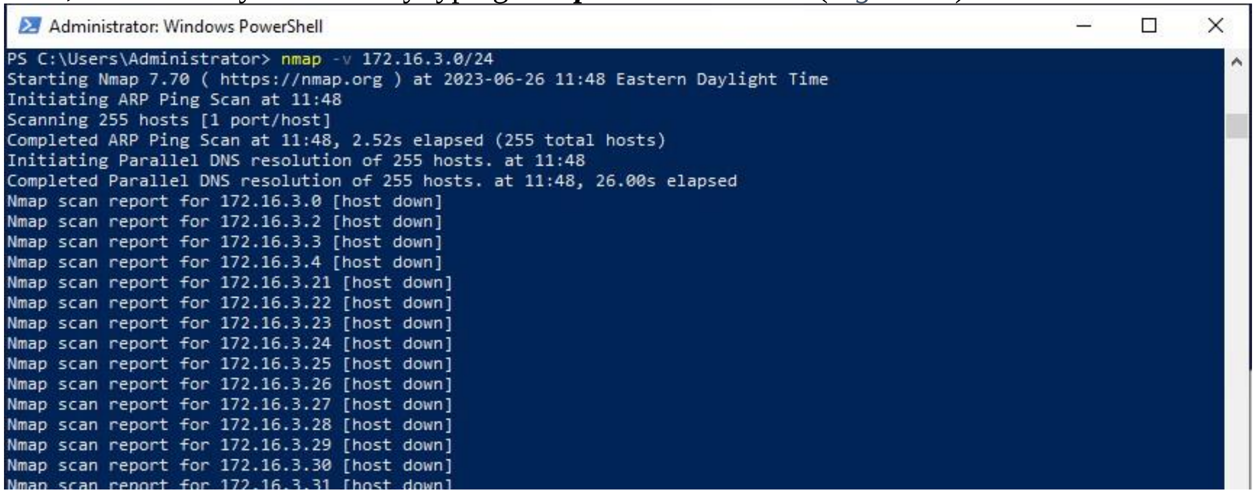


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Administrator> nmap
Nmap 7.70 ( https://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
  -PS/PA/PU/PY[portlist]: TCP SYN/ACK, UDP or SCTP discovery to given ports
  -PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
  -PO[protocol list]: IP Protocol Ping
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2],...>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
  --traceroute: Trace hop path to each host
SCAN TECHNIQUES:
  -sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans
  -sU: UDP Scan
  -sN/sF/sX: TCP Null, FIN, and Xmas scans
  --scanflags <flags>: Customize TCP scan flags
  -sI <zombie host[:probeport]>: Idle scan
  -sY/sZ: SCTP INIT/COOKIE-ECHO scans
  -sO: IP protocol scan
  -b <FTP relay host>: FTP bounce scan
PORT SPECIFICATION AND SCAN ORDER:
  -p <port ranges>: Only scan specified ports
    Ex: -p22; -p1-65535; -p U:53,111,137,T:21-25,80,139,8080,S:9
  --exclude-ports <port ranges>: Exclude the specified ports from scanning
  -F: Fast mode - Scan fewer ports than the default scan
  -r: Scan ports consecutively - don't randomize
  --top-ports <number>: Scan <number> most common ports
  --port-ratio <ratio>: Scan ports more common than <ratio>
SERVICE/VERSION DETECTION:
  -sV: Probe open ports to determine service/version info
  --version-intensity <level>: Set from 0 (light) to 9 (try all probes)
  --version-light: Limit to most likely probes (intensity 2)
  --version-all: Try every single probe (intensity 9)
```

*Figure 10: Opened PowerShell in VM and executed nmap via PowerShell.*

<u>Step 6</u>
After typing in nmap via PowerShell, I observed and analyzed the output. The output seemed to depict the capabilities of ***nmap***, including all the switches that can be used with ***nmap*** (*Figure 10*)

<u>Step 7</u>
Next, I initiated my first scan by typing ***nmap -v 172.16.3.0/24*** (*Figure 11*).



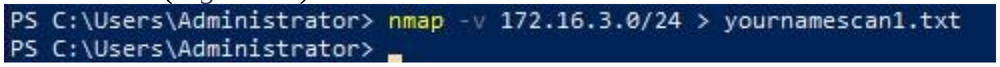
```
PS C:\Users\Administrator> nmap -v 172.16.3.0/24
Starting Nmap 7.70 ( https://nmap.org ) at 2023-06-26 11:48 Eastern Daylight Time
Initiating ARP Ping Scan at 11:48
Scanning 255 hosts [1 port/host]
Completed ARP Ping Scan at 11:48, 2.52s elapsed (255 total hosts)
Initiating Parallel DNS resolution of 255 hosts. at 11:48
Completed Parallel DNS resolution of 255 hosts. at 11:48, 26.00s elapsed
Nmap scan report for 172.16.3.0 [host down]
Nmap scan report for 172.16.3.2 [host down]
Nmap scan report for 172.16.3.3 [host down]
Nmap scan report for 172.16.3.4 [host down]
Nmap scan report for 172.16.3.21 [host down]
Nmap scan report for 172.16.3.22 [host down]
Nmap scan report for 172.16.3.23 [host down]
Nmap scan report for 172.16.3.24 [host down]
Nmap scan report for 172.16.3.25 [host down]
Nmap scan report for 172.16.3.26 [host down]
Nmap scan report for 172.16.3.27 [host down]
Nmap scan report for 172.16.3.28 [host down]
Nmap scan report for 172.16.3.29 [host down]
Nmap scan report for 172.16.3.30 [host down]
Nmap scan report for 172.16.3.31 [host down]
```

*Figure 11: Executing first scan using nmap via PowerShell.*

***-v*** is a switch/option used for verbosity, providing more detailed information about the scan it is performing. This can be useful for understanding exactly what ***nmap*** is doing, what stages of the scan it's at, and whether or not it's experiencing any difficulties (NMAP.org, n.d.).

Step 8
After executing the Linux command from Step 7, there was an overwhelming amount of output that was produced so I proceeded to Step 9 where it directed me to "redirect" the output to a file.

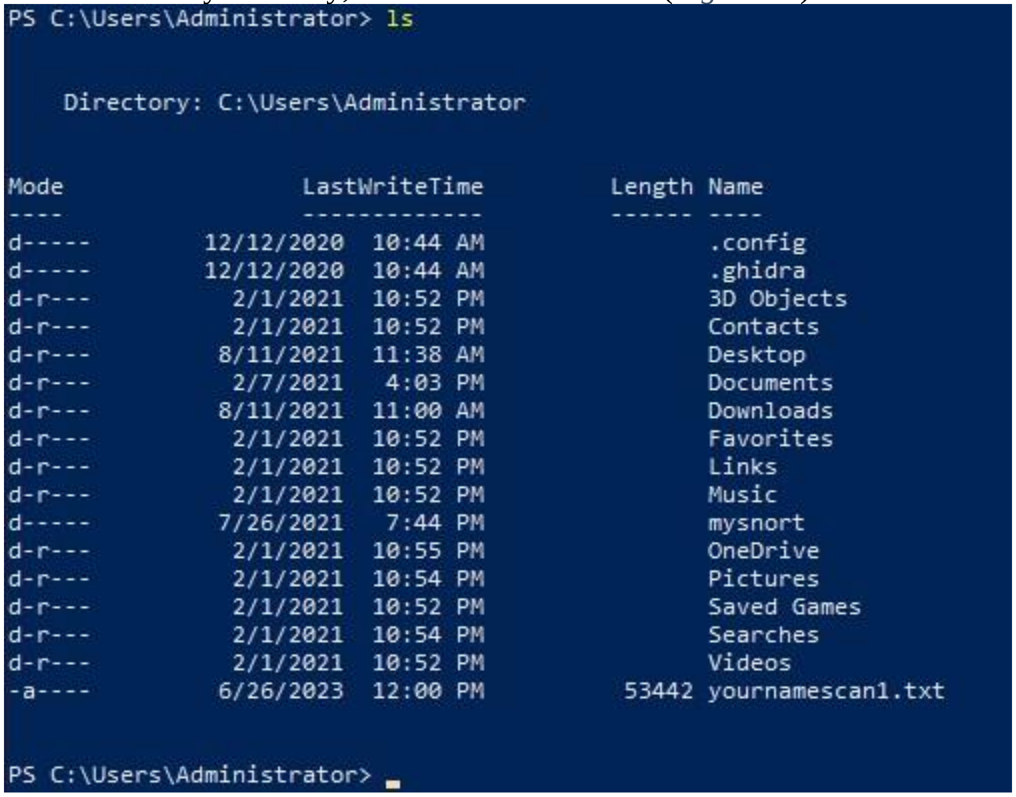
Step 9
Here, I executed another scan but redirected the output to a file by typing ***nmap -v 172.16.3.0/24 > yournamescan1.txt*** (*Figure 12*).



```
PS C:\Users\Administrator> nmap -v 172.16.3.0/24 > yournamescan1.txt
PS C:\Users\Administrator>
```

*Figure 12: Executed next scan and redirected output to a file.*

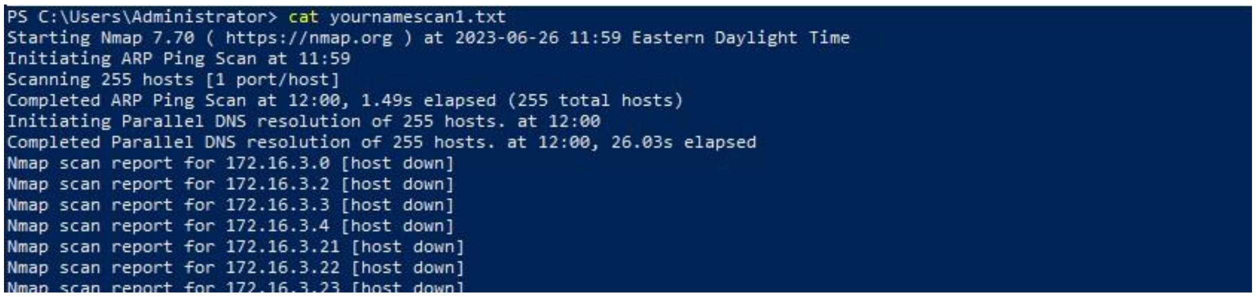
Step 10
To observe the files in my directory, I executed ***ls*** command (*Figure 13*).



```
PS C:\Users\Administrator> ls


    Directory: C:\Users\Administrator


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        12/12/2020  10:44 AM                .config
d-----        12/12/2020  10:44 AM                .ghidra
d-r---          2/1/2021  10:52 PM                3D Objects
d-r---          2/1/2021  10:52 PM                Contacts
d-r---         8/11/2021  11:38 AM                Desktop
d-r---          2/7/2021   4:03 PM                Documents
d-r---         8/11/2021  11:00 AM                Downloads
d-r---          2/1/2021  10:52 PM                Favorites
d-r---          2/1/2021  10:52 PM                Links
d-r---          2/1/2021  10:52 PM                Music
d-----         7/26/2021   7:44 PM                mysnort
d-r---          2/1/2021  10:55 PM                OneDrive
d-r---          2/1/2021  10:54 PM                Pictures
d-r---          2/1/2021  10:52 PM                Saved Games
d-r---          2/1/2021  10:54 PM                Searches
d-r---          2/1/2021  10:52 PM                Videos
-a----         6/26/2023  12:00 PM          53442 yournamescan1.txt


PS C:\Users\Administrator>
```

*Figure 13: Observing files in my directory.*

Step 11
To observe my output type I executed ***cat yournamescan1.txt*** in PowerShell (*Figure 14*). The output was same/similar as the output produced in Step 7.

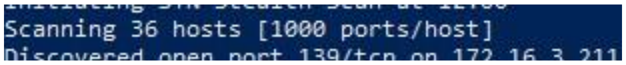


```
PS C:\Users\Administrator> cat yournamescan1.txt
Starting Nmap 7.70 ( https://nmap.org ) at 2023-06-26 11:59 Eastern Daylight Time
Initiating ARP Ping Scan at 11:59
Scanning 255 hosts [1 port/host]
Completed ARP Ping Scan at 12:00, 1.49s elapsed (255 total hosts)
Initiating Parallel DNS resolution of 255 hosts. at 12:00
Completed Parallel DNS resolution of 255 hosts. at 12:00, 26.03s elapsed
Nmap scan report for 172.16.3.0 [host down]
Nmap scan report for 172.16.3.2 [host down]
Nmap scan report for 172.16.3.3 [host down]
Nmap scan report for 172.16.3.4 [host down]
Nmap scan report for 172.16.3.21 [host down]
Nmap scan report for 172.16.3.22 [host down]
Nmap scan report for 172.16.3.23 [host down]
```

*Figure 14: Observing contents of yournamescan1.txt.*

<u>Step 12</u>
From reviewing the output in Step 11, I know that when I executed the nmap command to scan for 172.16.3.0/24 I was scanning the network 172.16.3.0 with a netmask of 255.255.255.0. The /24 is a classless inter-domain routing (CIDR) notation; therefore, I was scanning IPs from 172.16.3.0 through 172.16.3.255. Looking at *Figure 14*, 255 total hosts, or 255 total addresses were scanned.

<u>Step 13</u>
Further reviewing/analyzing output after scanning 172.16.3.0/24 I could determine that 36 hosts were up because the ***nmap*** output later depicts it scanned 36 hosts (*Figure 15*) because the others were down.



```
Scanning 36 hosts [1000 ports/host]
Discovered open port 139/tcp on 172.16.3.211
```

*Figure 15: NMAP scanning hosts that were up.*

<u>Step 14</u>
After reviewing the Network Map (*Figure 16*) via SimSpace further validated the number of hosts that I observed were up from <u>Step 13</u>. From the Network Map, I could see that exactly 36 hosts were up.

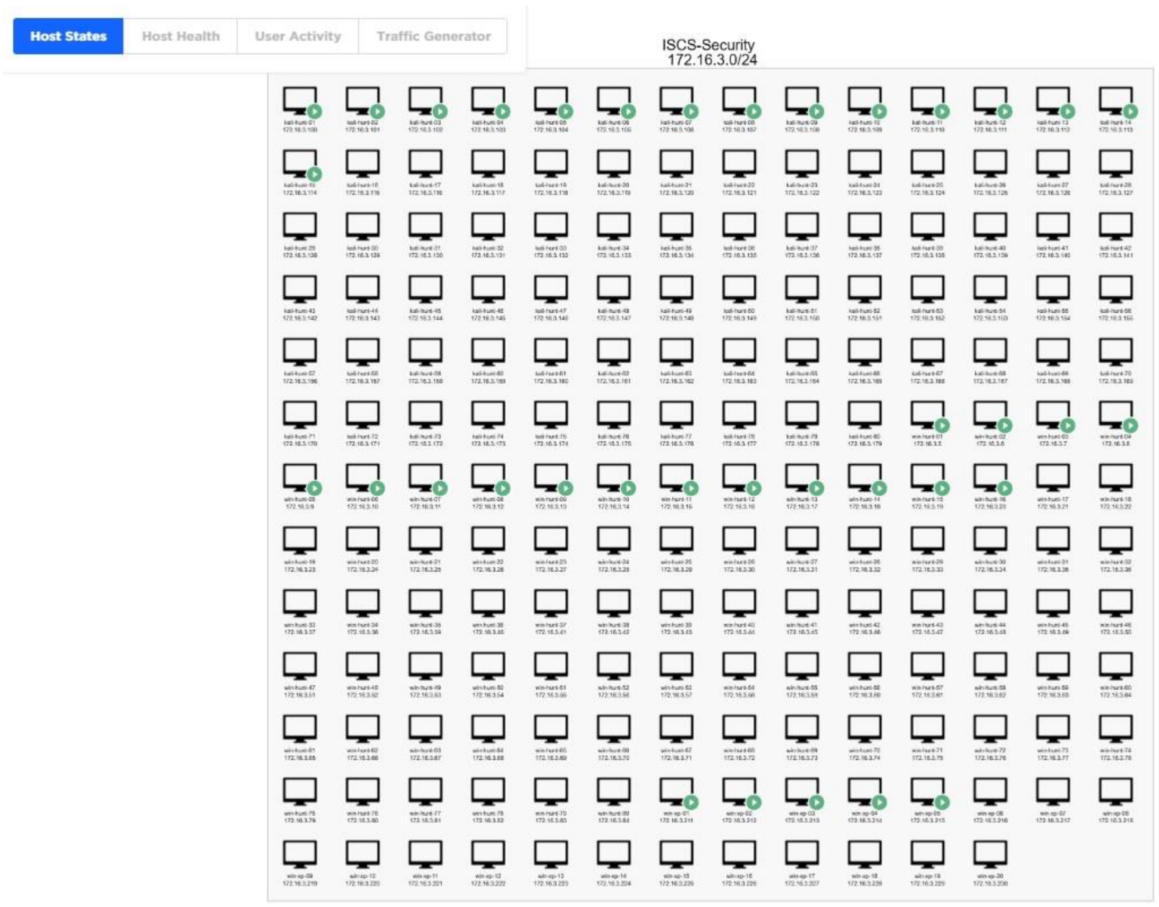
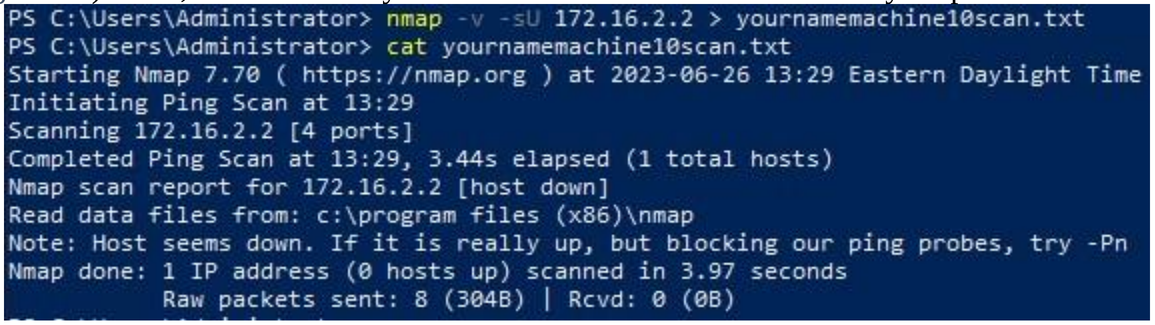


*Figure 16: Network Map of 172.16.3.0/24 subnet via SimSpace*

Step 15
Per the lab instructions, I executed **nmap -v -sU 172.16.2.2 > yournamemachine10scan.txt** (*Figure 17*). Next, I executed cat yournamemachine10scan.txt to view my output.



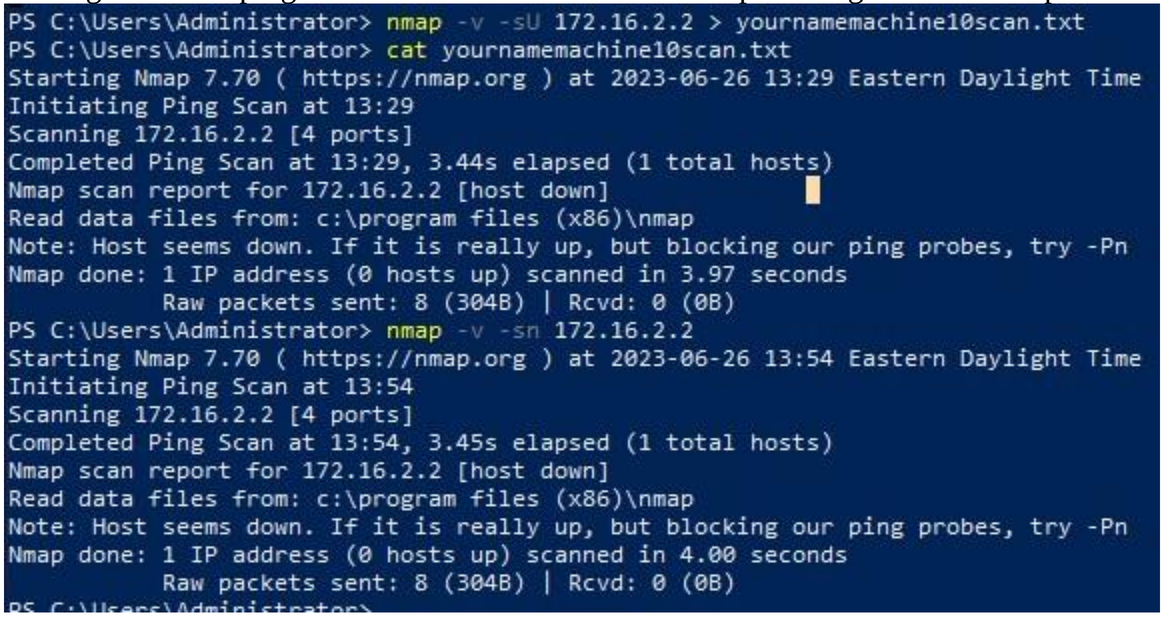
```
PS C:\Users\Administrator> nmap -v -sU 172.16.2.2 > yournamemachine10scan.txt
PS C:\Users\Administrator> cat yournamemachine10scan.txt
Starting Nmap 7.70 ( https://nmap.org ) at 2023-06-26 13:29 Eastern Daylight Time
Initiating Ping Scan at 13:29
Scanning 172.16.2.2 [4 ports]
Completed Ping Scan at 13:29, 3.44s elapsed (1 total hosts)
Nmap scan report for 172.16.2.2 [host down]
Read data files from: c:\program files (x86)\nmap
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.97 seconds
           Raw packets sent: 8 (304B) | Rcvd: 0 (0B)
```

*Figure 17: NMAP scanning for open UDP ports at IP address 172.16.2.2*

In Step 7 we used ***-v*** option/switch to increase verbosity of ***nmap*** output. Here we added ***-v -sU***. ***-sU*** is an option that tells NMAP to perform a User Datagram Protocol (UDP) scan (NMAP.org, n.d.). This scan looks for open UDP ports. Therefore, I am telling ***nmap*** to perform a detailed scan for UDP ports at IP address 172.16.2.2 and give a verbose output as seen in *Figure 17*.

Looking at *Figure 17*, NMAP scanned for 4 ports at IP address 172.16.2.2; however, the host itself seemed to be down.

Further, per the lab instructions, I executed ***nmap -v -sn 172.16.2.2***. To compare the outputs. Adding ***-sn***, ping scan, I can determine which hosts on a network are online and responding to pings, but it does not scan the ports of the target hosts. Essentially, it's a quick way to identify hosts on the network without going through the longer process of port scanning. So here, I'm performing a verbose ping scan at IP address 172.16.2.2 and providing a detailed output.

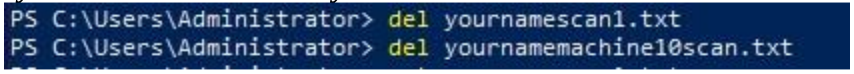


```
PS C:\Users\Administrator> nmap -v -sU 172.16.2.2 > yournamemachine10scan.txt
PS C:\Users\Administrator> cat yournamemachine10scan.txt
Starting Nmap 7.70 ( https://nmap.org ) at 2023-06-26 13:29 Eastern Daylight Time
Initiating Ping Scan at 13:29
Scanning 172.16.2.2 [4 ports]
Completed Ping Scan at 13:29, 3.44s elapsed (1 total hosts)
Nmap scan report for 172.16.2.2 [host down]
Read data files from: c:\program files (x86)\nmap
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.97 seconds
           Raw packets sent: 8 (304B) | Rcvd: 0 (0B)
PS C:\Users\Administrator> nmap -v -sn 172.16.2.2
Starting Nmap 7.70 ( https://nmap.org ) at 2023-06-26 13:54 Eastern Daylight Time
Initiating Ping Scan at 13:54
Scanning 172.16.2.2 [4 ports]
Completed Ping Scan at 13:54, 3.45s elapsed (1 total hosts)
Nmap scan report for 172.16.2.2 [host down]
Read data files from: c:\program files (x86)\nmap
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 4.00 seconds
           Raw packets sent: 8 (304B) | Rcvd: 0 (0B)
```

*Figure 18: Comparing output of NMAP scan using -v -sU versus -v -sn.*

Looking at *Figure 18*, I observed no difference in output between nmap ***-v -sU*** versus ***nmap -v -sn*** scans.

<u>Step 16</u>
Lastly, I deleted ***yournamescan1.txt*** and ***yournamemachine10scan.txt*** from VM (*Figure 19*).



```
PS C:\Users\Administrator> del yournamescan1.txt
PS C:\Users\Administrator> del yournamemachine10scan.txt
```

*Figure 19: Using del command to delete text files via PowerShell.*

<u>Step 17</u>
In this part of the lab, per the lab instructions, I was directed to log into one of the **kali-hunt-xx** VMs and pick 3 applications that could be useful for a security practitioner. I logged into **kali-hunt-01** VM and began exploring applications.

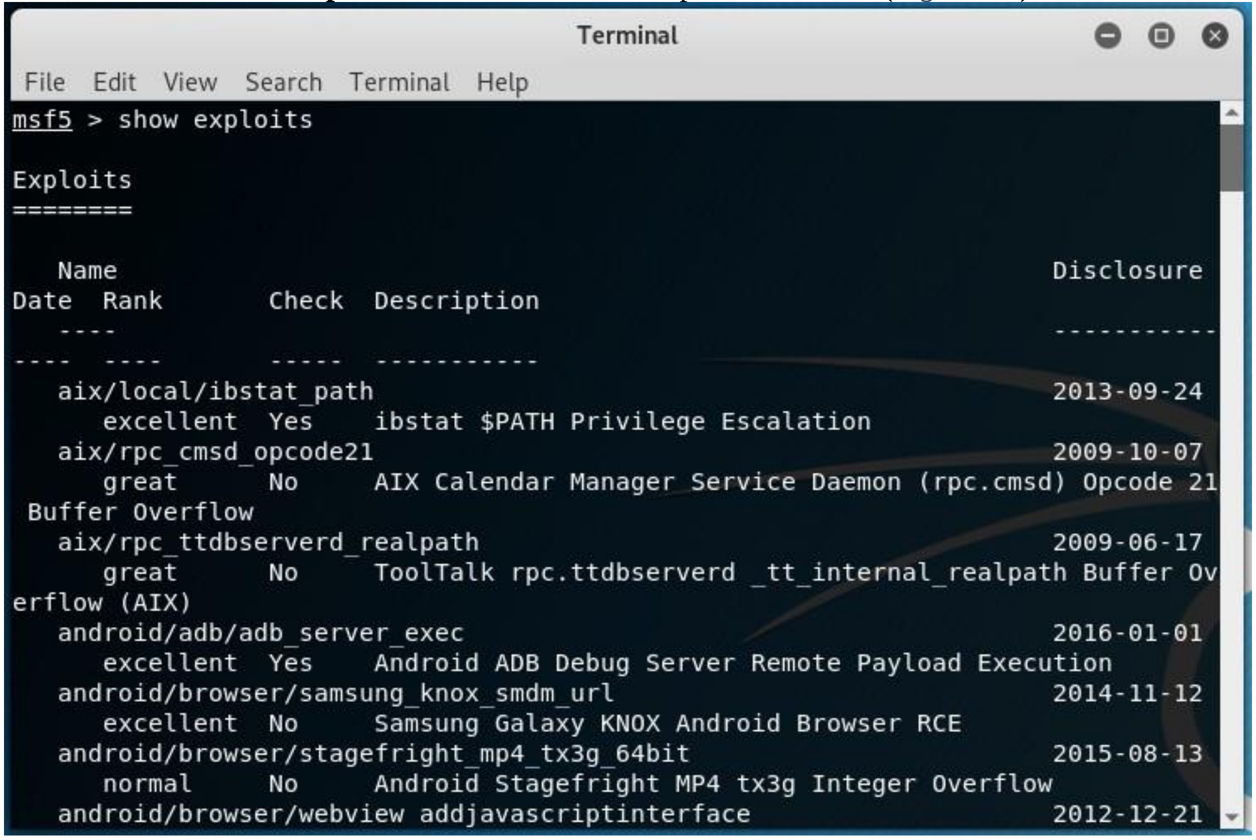
### *Metasploit*

Metasploit is a powerful and widely used penetration testing framework that is highly valuable for network cybersecurity professionals (Simplilearn, 2023). According to Okta, Metasploit comes with three components used to exploit network vulnerabilities: exploit, payloads, and auxiliaries. Exploits act as carriers, executing specific actions based on known vulnerabilities, while payloads function as viruses, deployed by the exploits. Additionally, auxiliaries are custom functions that handle tasks unrelated to system exploits, such as scanning and sniffing (Okta, 2022).

In an effort to familiarize myself with Metasploit, I followed a tutorial via YouTube provided the C4 Cyber Club (C4 Cyber Club, 2021).

Continuing on the **kali-hunt-01** VM, I opened up the Metasploit application. Per the YouTube video instructions, I needed execute a few command sequences to initialize, configure and launch a console, but upon opening up the application I noticed that it executed the aforementioned on its own.

Here, I executed ***show exploits*** and viewed all the exploits available (*Figure 20*).

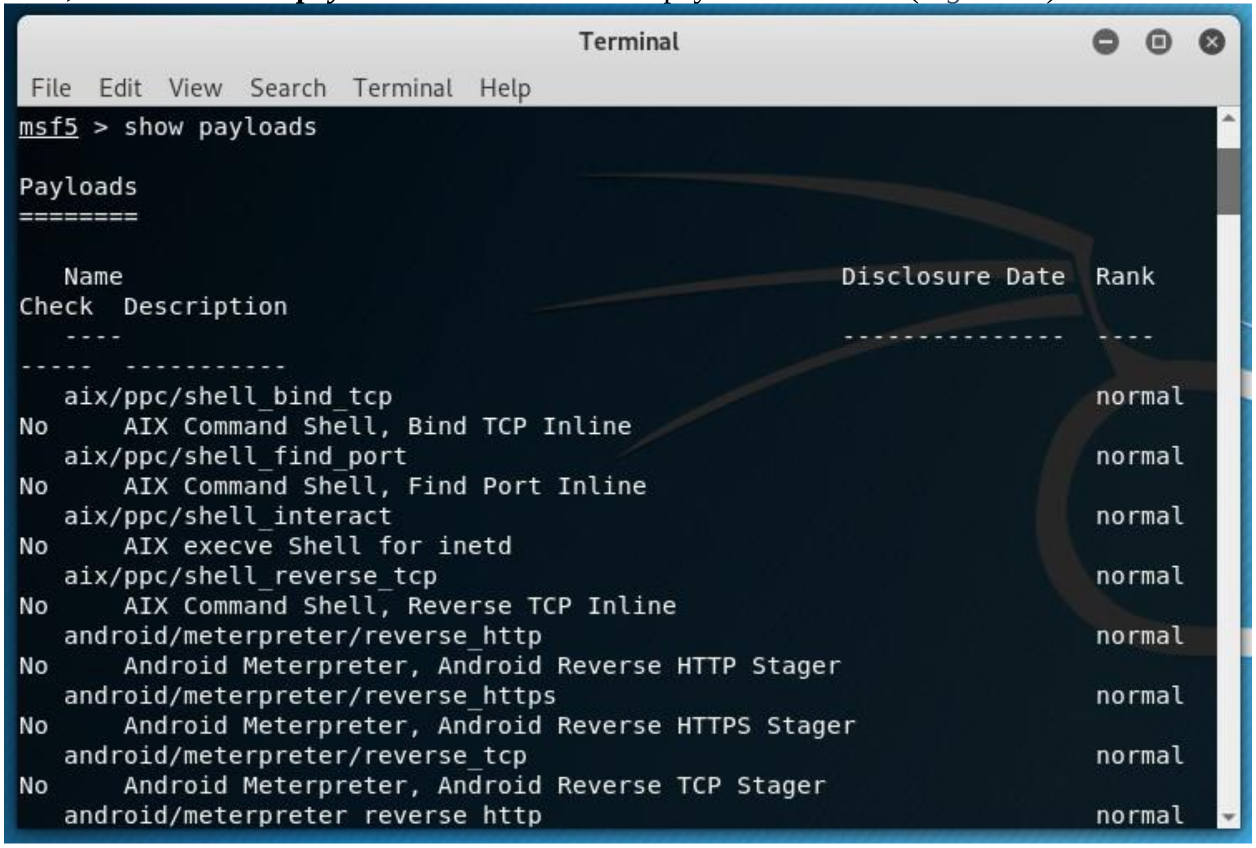


```
msf5 > show exploits

Exploits
========

   Name                                                              Disclosure
Date  Rank       Check  Description
   ----                                                              -----------
----  ----       -----  -----------
   aix/local/ibstat_path                                             2013-09-24
      excellent  Yes    ibstat $PATH Privilege Escalation
   aix/rpc_cmsd_opcode21                                             2009-10-07
      great      No     AIX Calendar Manager Service Daemon (rpc.cmsd) Opcode 21
 Buffer Overflow
   aix/rpc_ttdbserverd_realpath                                      2009-06-17
      great      No     ToolTalk rpc.ttdbserverd _tt_internal_realpath Buffer Ov
erflow (AIX)
   android/adb/adb_server_exec                                       2016-01-01
      excellent  Yes    Android ADB Debug Server Remote Payload Execution
   android/browser/samsung_knox_smdm_url                             2014-11-12
      excellent  No     Samsung Galaxy KNOX Android Browser RCE
   android/browser/stagefright_mp4_tx3g_64bit                        2015-08-13
      normal     No     Android Stagefright MP4 tx3g Integer Overflow
   android/browser/webview addjavascriptinterface                    2012-12-21
```

*Figure 20: Executing show exploits command via Metasploit.*

The ***show exploits*** command in Metasploit is an invaluable tool for security professionals. It allows them to explore and research various exploits within the framework, gain insights into vulnerabilities and target platforms, and stay updated on emerging threats. By examining the list of exploits, practitioners can identify vulnerabilities in their target systems, prioritize security efforts, and make informed decisions about exploit selection based on reliability and compatibility. They can also customize exploits to suit specific needs, leveraging the command's parameters and configurations. Additionally, the command provides detailed documentation for each exploit, offering functionality explanations and usage instructions.

Here, I executed ***show payloads*** and viewed all the payloads available (*Figure 21*).

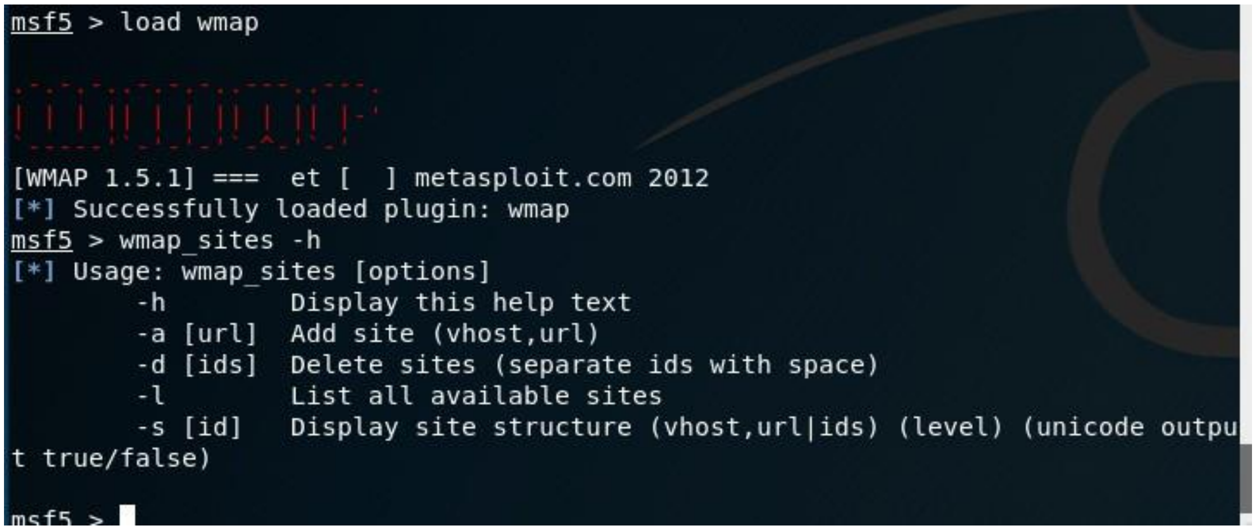


```
msf5 > show payloads

Payloads
========

   Name                                                Disclosure Date  Rank
Check  Description
   ----                                                ---------------  ----
-----  -----------
   aix/ppc/shell_bind_tcp                                                normal
No     AIX Command Shell, Bind TCP Inline
   aix/ppc/shell_find_port                                               normal
No     AIX Command Shell, Find Port Inline
   aix/ppc/shell_interact                                                normal
No     AIX execve Shell for inetd
   aix/ppc/shell_reverse_tcp                                             normal
No     AIX Command Shell, Reverse TCP Inline
   android/meterpreter/reverse_http                                      normal
No     Android Meterpreter, Android Reverse HTTP Stager
   android/meterpreter/reverse_https                                     normal
No     Android Meterpreter, Android Reverse HTTPS Stager
   android/meterpreter/reverse_tcp                                       normal
No     Android Meterpreter, Android Reverse TCP Stager
   android/meterpreter reverse http                                      normal
```

*Figure 21: Executing show payloads command via Metasploit.*

***show payloads*** in Metasploit provides security professionals with a comprehensive list of available payloads in the framework. Payloads are code or scripts that are executed on a target system after a successful exploit. Security professionals can use this command to select payloads tailored to their specific testing goals, customize payload parameters for compatibility and evasion, identify payloads for post-exploitation activities, and access detailed documentation for functionality and usage guidance. By leveraging the ***show payloads*** command, practitioners enhance the effectiveness and efficiency of security testing, assessment, and post-exploitation activities conducted with Metasploit.

Metasploit also contains vulnerability scanning modules. Per the video, I loaded ***wmap*** (*Figure 22*).



```
msf5 > load wmap

[WMAP 1.5.1] ===  et [  ] metasploit.com 2012
[*] Successfully loaded plugin: wmap
msf5 > wmap_sites -h
[*] Usage: wmap_sites [options]
        -h        Display this help text
        -a [url]  Add site (vhost,url)
        -d [ids]  Delete sites (separate ids with space)
        -l        List all available sites
        -s [id]   Display site structure (vhost,url|ids) (level) (unicode outpu
t true/false)

msf5 >
```

*Figure 22: Loaded WMAP via Metasploit.*

***wmap*** is a module within Metasploit that focuses on web application scanning and vulnerability assessment. It provides a range of automated functionalities to aid security professionals in identifying and assessing vulnerabilities in web applications.

## Armitage

Armitage is essentially a GUI version of Metasploit, simplifying network penetration testing and post-exploitation tasks. It provides visual representations of the target network, aiding security practitioners in understanding the network's structure and identifying attack paths (Artykov, 2021). Armitage facilitates collaboration among team members, automates exploit execution, generates detailed reports, and includes a task management system for organizing and prioritizing activities. These features make Armitage valuable to cybersecurity professionals, improving efficiency, simplifying exploit execution, enabling comprehensive reporting, and aiding in task management during penetration testing engagements.

By following another YouTube tutorial by HackerSploit (HackerSploit, 2018), I opened Armitage and began observing the interfaces. (*Figure 23*).

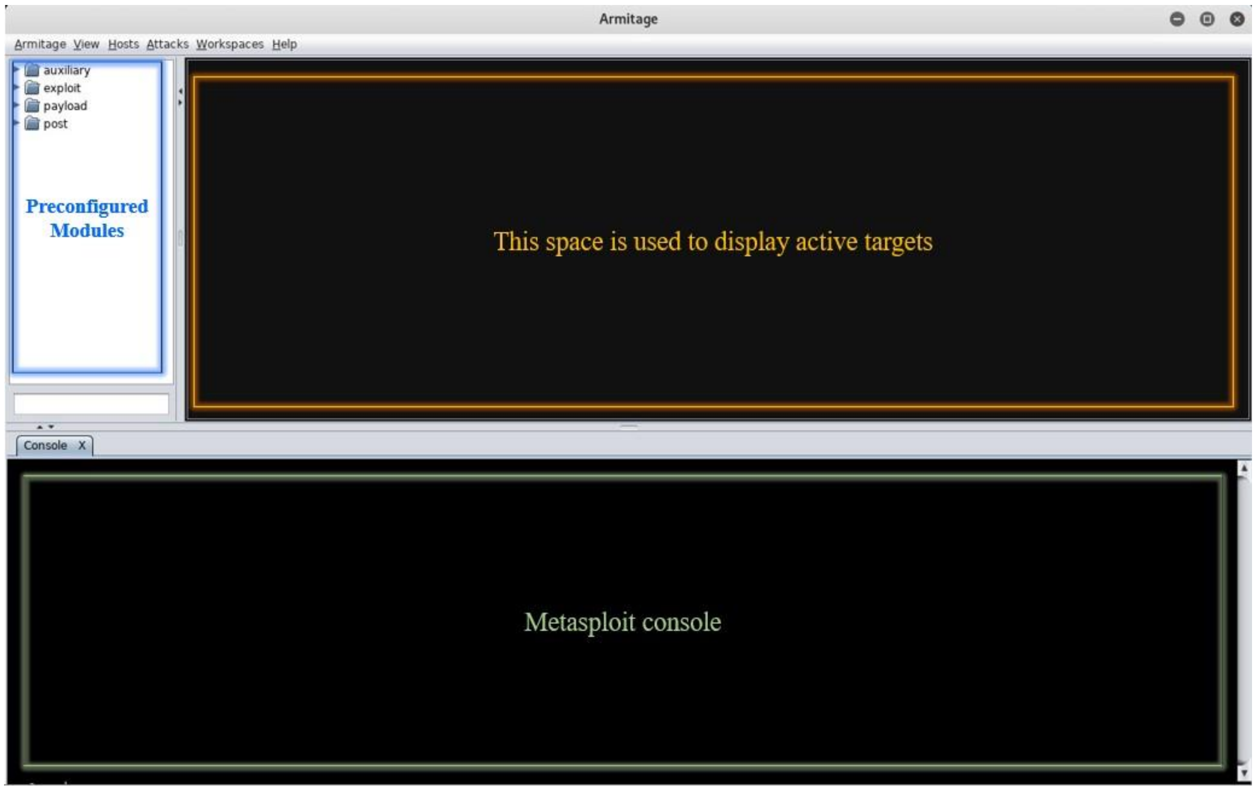

Armitage
Armitage View Hosts Attacks Workspaces Help
auxiliary
exploit
payload
post
Preconfigured Modules
This space is used to display active targets
Console X
Metasploit console

*Figure 23: Successful loading of Armitage via VM.*

Armitage has the ***nmap*** utility built into the toolbar in the upper left, under the 'Hosts' option. Per the video, I executed a 'Quick Scan (OS detect)'.

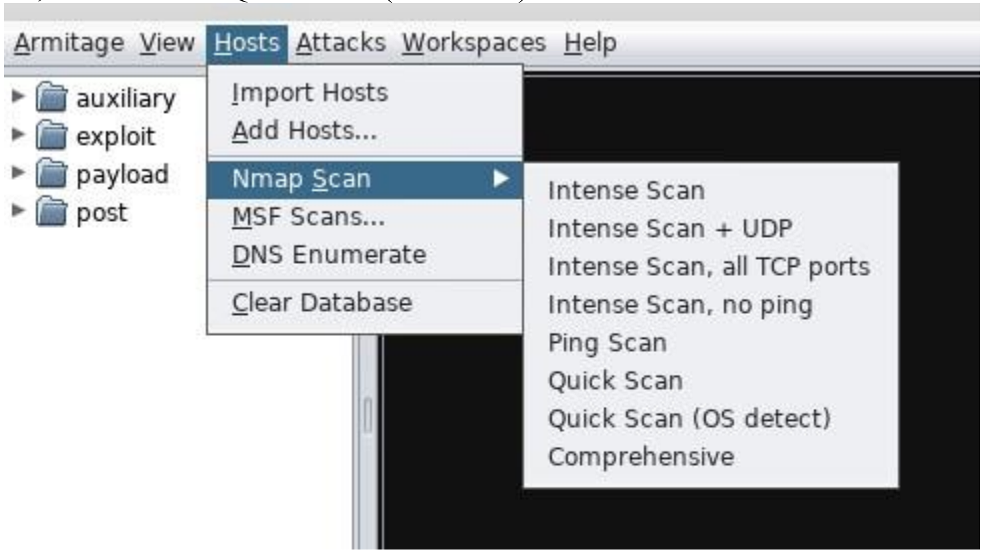
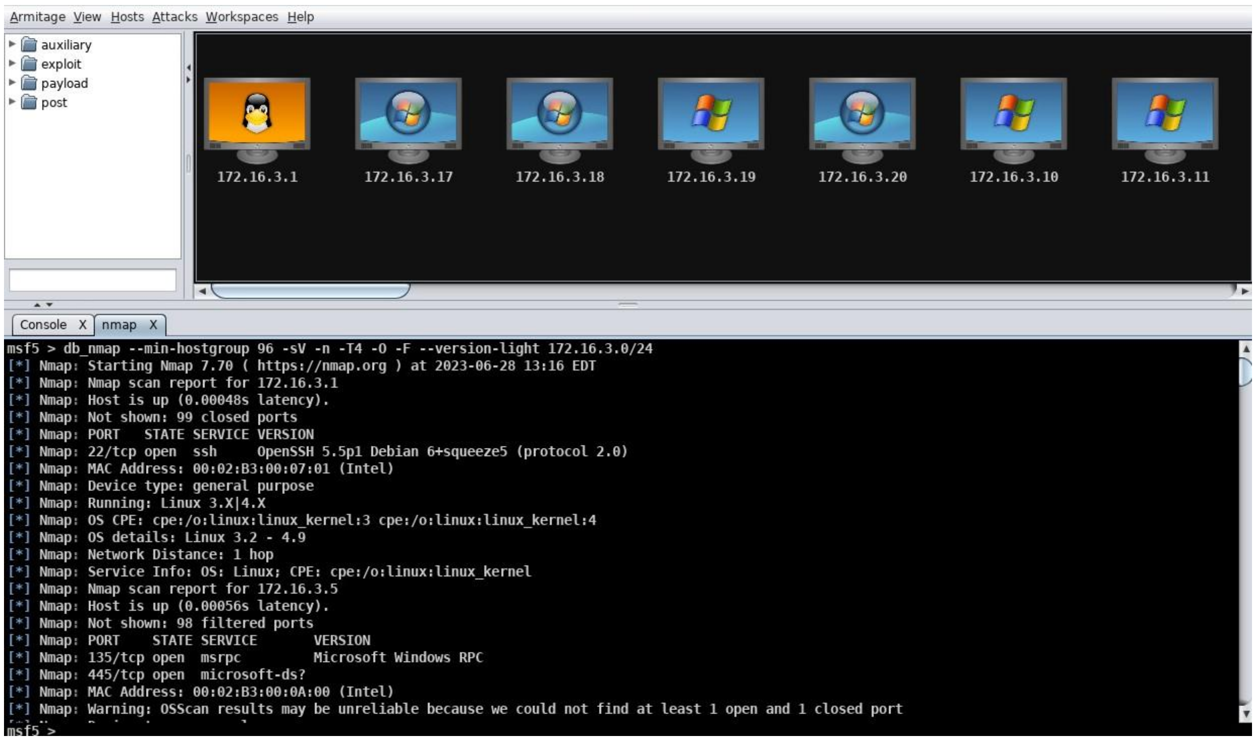


*Figure 24: Accessing Nmap Quick Scan (OS detect) via Armitage.*

Instead of using the IP address from the video I used the IP address from Step 7, 172.16.3.0/24 and executed the ***nmap*** scan via Armitage. After the scan was complete, I noticed that in the second interface—the space that depicts active targets—depicted 36 hosts (Figure 25), which further validates the 36 hosts that I found in Step 13 during the ***nmap*** scan via PowerShell.



```
msf5 > db_nmap --min-hostgroup 96 -sV -n -T4 -O -F --version-light 172.16.3.0/24
[*] Nmap: Starting Nmap 7.70 ( https://nmap.org ) at 2023-06-28 13:16 EDT
[*] Nmap: Nmap scan report for 172.16.3.1
[*] Nmap: Host is up (0.00048s latency).
[*] Nmap: Not shown: 99 closed ports
[*] Nmap: PORT   STATE SERVICE VERSION
[*] Nmap: 22/tcp open  ssh     OpenSSH 5.5p1 Debian 6+squeeze5 (protocol 2.0)
[*] Nmap: MAC Address: 00:02:B3:00:07:01 (Intel)
[*] Nmap: Device type: general purpose
[*] Nmap: Running: Linux 3.X|4.X
[*] Nmap: OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
[*] Nmap: OS details: Linux 3.2 - 4.9
[*] Nmap: Network Distance: 1 hop
[*] Nmap: Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
[*] Nmap: Nmap scan report for 172.16.3.5
[*] Nmap: Host is up (0.00056s latency).
[*] Nmap: Not shown: 98 filtered ports
[*] Nmap: PORT    STATE SERVICE       VERSION
[*] Nmap: 135/tcp open  msrpc         Microsoft Windows RPC
[*] Nmap: 445/tcp open  microsoft-ds?
[*] Nmap: MAC Address: 00:02:B3:00:0A:00 (Intel)
[*] Nmap: Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
msf5 >
```

*Figure 25: Executed Nmap scan via Armitage.*

There many ways to use Armitage to exploit vulnerabilities including the options in the preconfigured modules and/or simply by right clicking one of the active targets. The one I found most interesting was the option to find exploits automatically by going to the tool bar, and selecting ‘Attacks’, then ‘Find Attacks’(*Figure 26*).

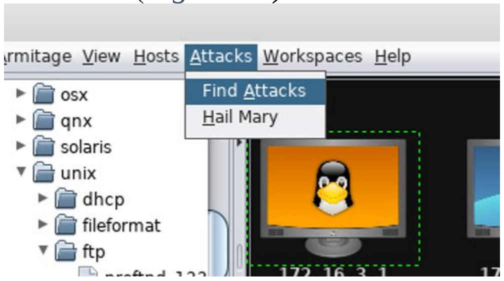
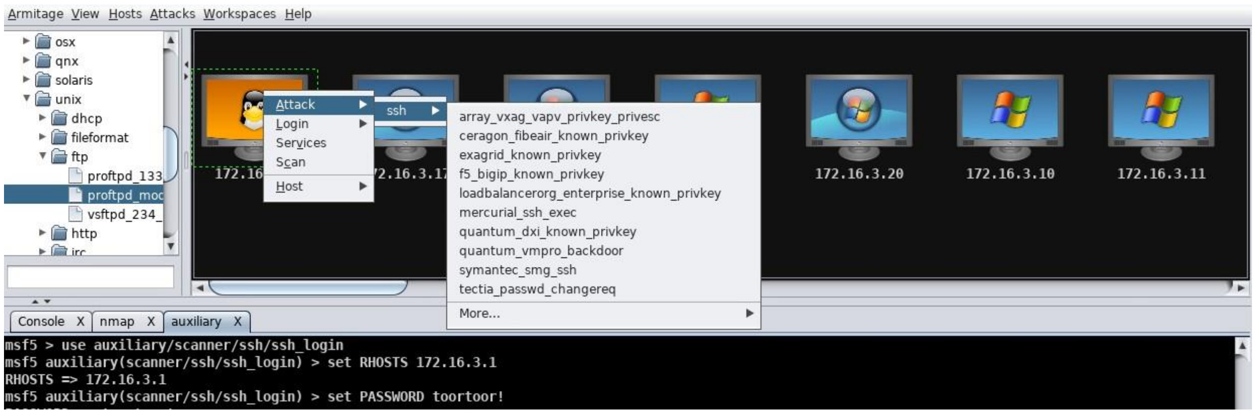


*Figure 26:Selecting 'Find Attacks' via Armitage.*

Next, Armitage begins querying for exploits. After it was complete, I right clicked on the first host and observed all the vulnerabilities/exploits that Armitage could find (*Figure 27*).



```
msf5 > use auxiliary/scanner/ssh/ssh_login
msf5 auxiliary(scanner/ssh/ssh_login) > set RHOSTS 172.16.3.1
RHOSTS => 172.16.3.1
msf5 auxiliary(scanner/ssh/ssh_login) > set PASSWORD toortoor!
```

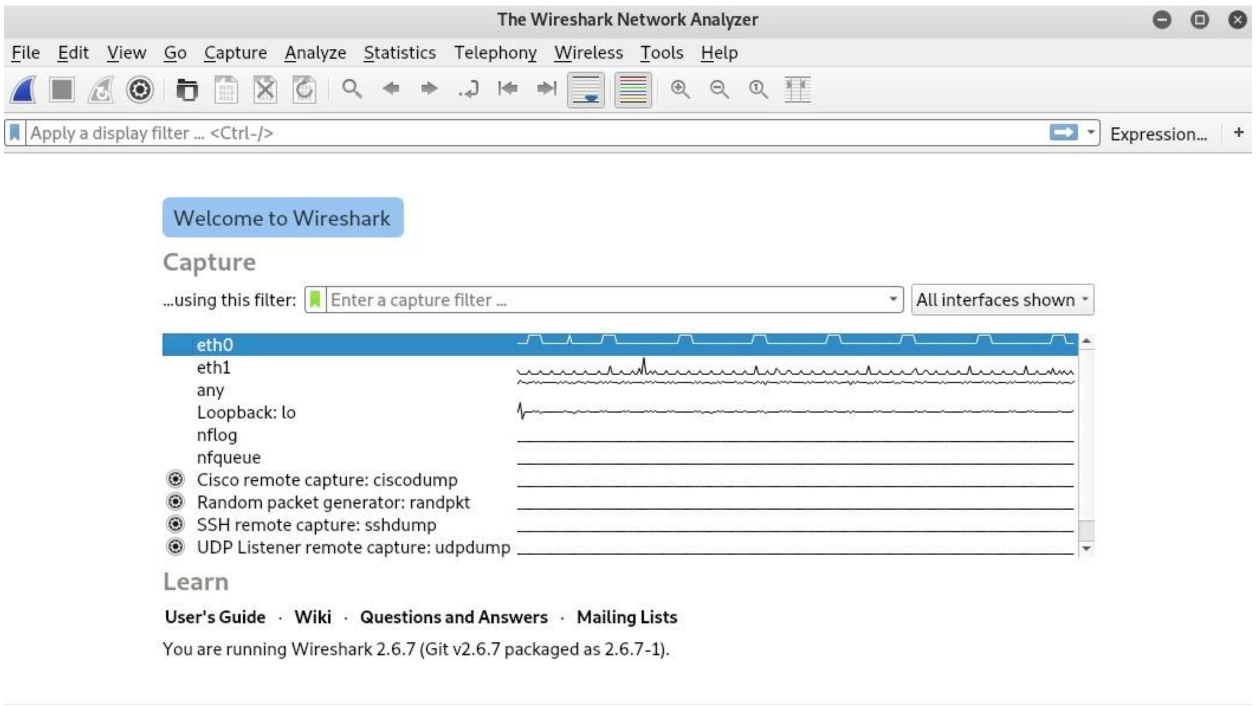
*Figure 27: Observing exploits/vulnerabilities Armitage found in the first host.*

The ‘Find Attack’ feature in Armitage is a useful tool for cybersecurity professionals as it allows for automated mapping of known exploits to discovered vulnerabilities on a target. This function significantly streamlines the exploit selection process, reducing manual effort, and potentially increasing accuracy by cross-referencing identified services and versions against a comprehensive database of known vulnerabilities and exploits. Moreover, the feature aids in efficient utilization of Armitage's capabilities, and the attacks can be sorted based on their reliability or impact, thereby assisting in prioritization.

## *Wireshark*

Wireshark is a free application that allows security professionals to capture and interactively browse the traffic running on a computer network. It's useful for security practitioners because it provides detailed insight into network activities, helping identify malicious activities, analyze packets, troubleshoot network issues, and understand network protocols in-depth. It’s features like filtering, color coding, and protocol dissectors, Wireshark aids in visualizing complex network interactions, which can be valuable for intrusion detection, network forensics, and monitoring network health. Therefore, Wireshark serves as a critical tool in a security professional's toolkit for diagnosing both routine and complex network-related issues (CompTIA, n.d.).

When opening the application for the first time, I’m welcomed by the welcome screen along with what appears to be network traffic (*Figure 28*).

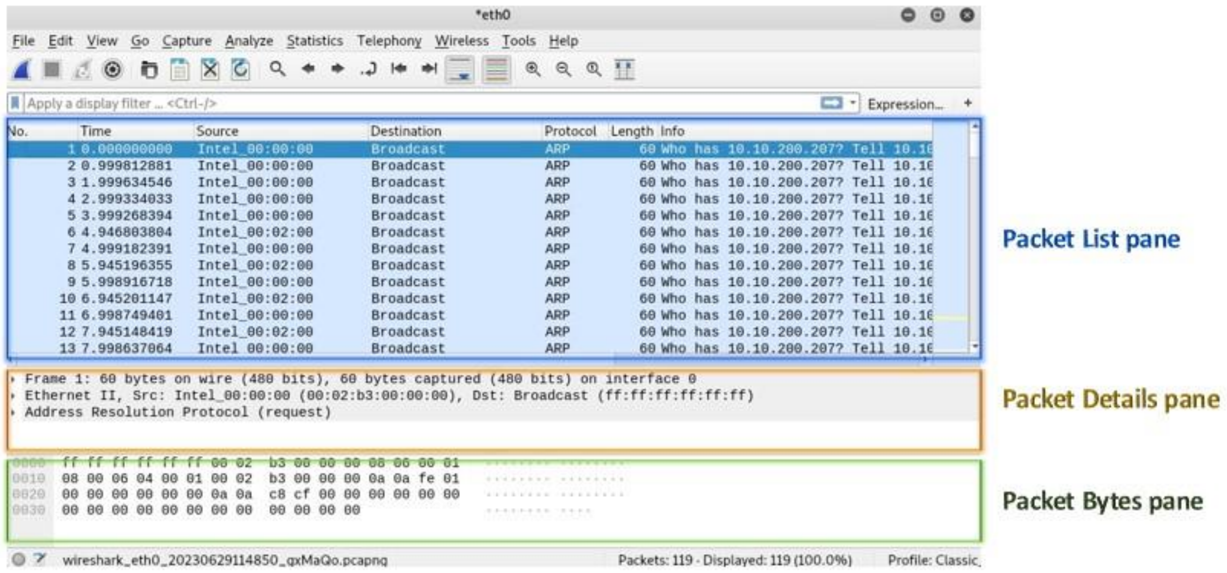

The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
Welcome to Wireshark
Capture
...using this filter:
Enter a capture filter ...
All interfaces shown
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists

```
eth0
eth1
any
Loopback: lo
nflog
nfqueue
Cisco remote capture: ciscodump
Random packet generator: randpkt
SSH remote capture: sshdump
UDP Listener remote capture: udpdump
```

You are running Wireshark 2.6.7 (Git v2.6.7 packaged as 2.6.7-1).

*Figure 28: Wireshark welcome screen.*

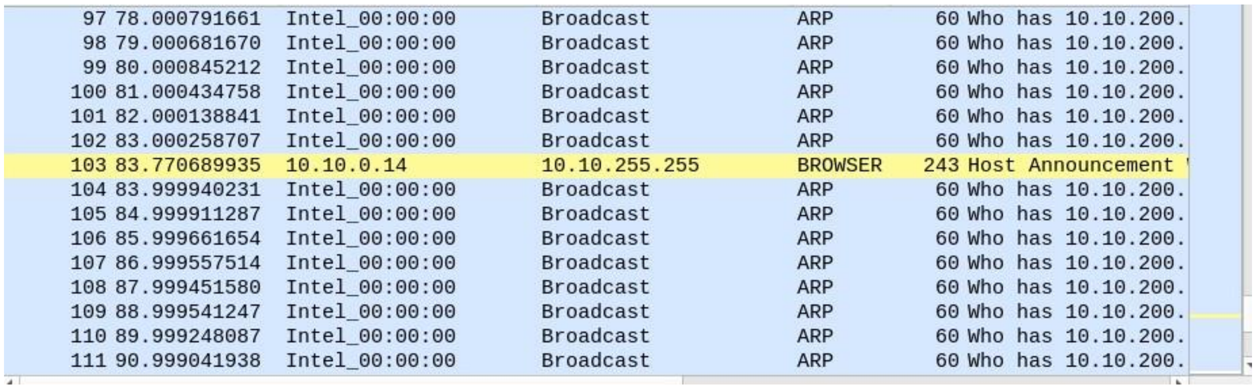
I began my first packet capture by pressing the little blue shark fin button located in the upper left corner. I ran the feature for approximately one minute in hopes of capturing some data packets (*Figure 29*).


*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
Packet List pane
Packet Details pane
Packet Bytes pane

```
No.   Time          Source          Destination     Protocol  Length Info
    1 0.000000000   Intel_00:00:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1
    2 0.999812881   Intel_00:00:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1
    3 1.999634546   Intel_00:00:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1
    4 2.999334033   Intel_00:00:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1
    5 3.999268394   Intel_00:00:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1
    6 4.946803804   Intel_00:02:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1
    7 4.999182391   Intel_00:00:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1
    8 5.945196355   Intel_00:02:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1
    9 5.998916718   Intel_00:00:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1
   10 6.945201147   Intel_00:02:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1
   11 6.998749401   Intel_00:00:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1
   12 7.945148419   Intel_00:02:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1
   13 7.998637064   Intel_00:00:00  Broadcast       ARP           60 Who has 10.10.200.207? Tell 10.1

› Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
› Ethernet II, Src: Intel_00:00:00 (00:02:b3:00:00:00), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
› Address Resolution Protocol (request)

0000  ff ff ff ff ff ff 00 02  b3 00 00 00 08 06 00 01   ........ ........
0010  08 00 06 04 00 01 00 02  b3 00 00 00 0a 0a fe 01   ........ ........
0020  00 00 00 00 00 00 0a 0a  c8 cf 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....

wireshark_eth0_20230629114850_qxMaQo.pcapng     Packets: 119 · Displayed: 119 (100.0%)     Profile: Classic
```

*Figure 29: Wireshark packet capture.*

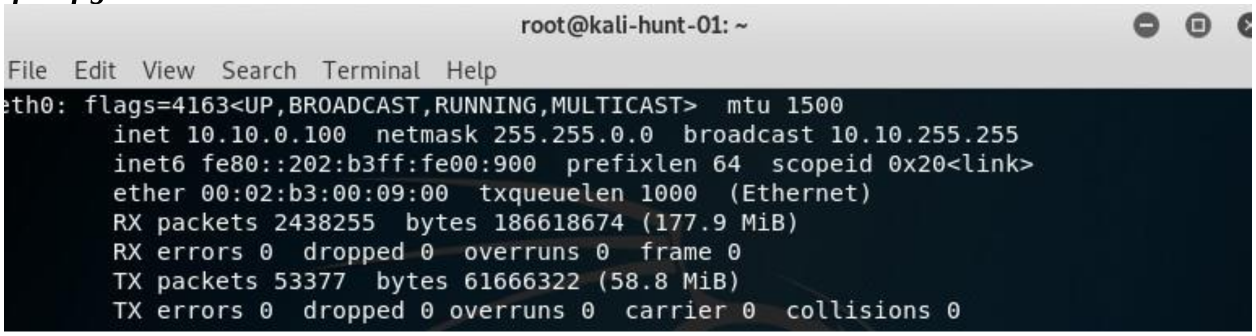
Analyzing the data packets via the packet list pane, most packets looked similar except I found one that stood out. Wireshark also highlighted the packet in yellow to delineate the difference. (*Figure 30*)



```
 97 78.000791661  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
 98 79.000681670  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
 99 80.000845212  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
100 81.000434758  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
101 82.000138841  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
102 83.000258707  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
103 83.770689935  10.10.0.14          10.10.255.255     BROWSER   243 Host Announcement
104 83.999940231  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
105 84.999911287  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
106 85.999661654  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
107 86.999557514  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
108 87.999451580  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
109 88.999541247  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
110 89.999248087  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
111 90.999041938  Intel_00:00:00      Broadcast         ARP        60 Who has 10.10.200.
```

*Figure 30: Analyzing data packets from Wireshark scan.*

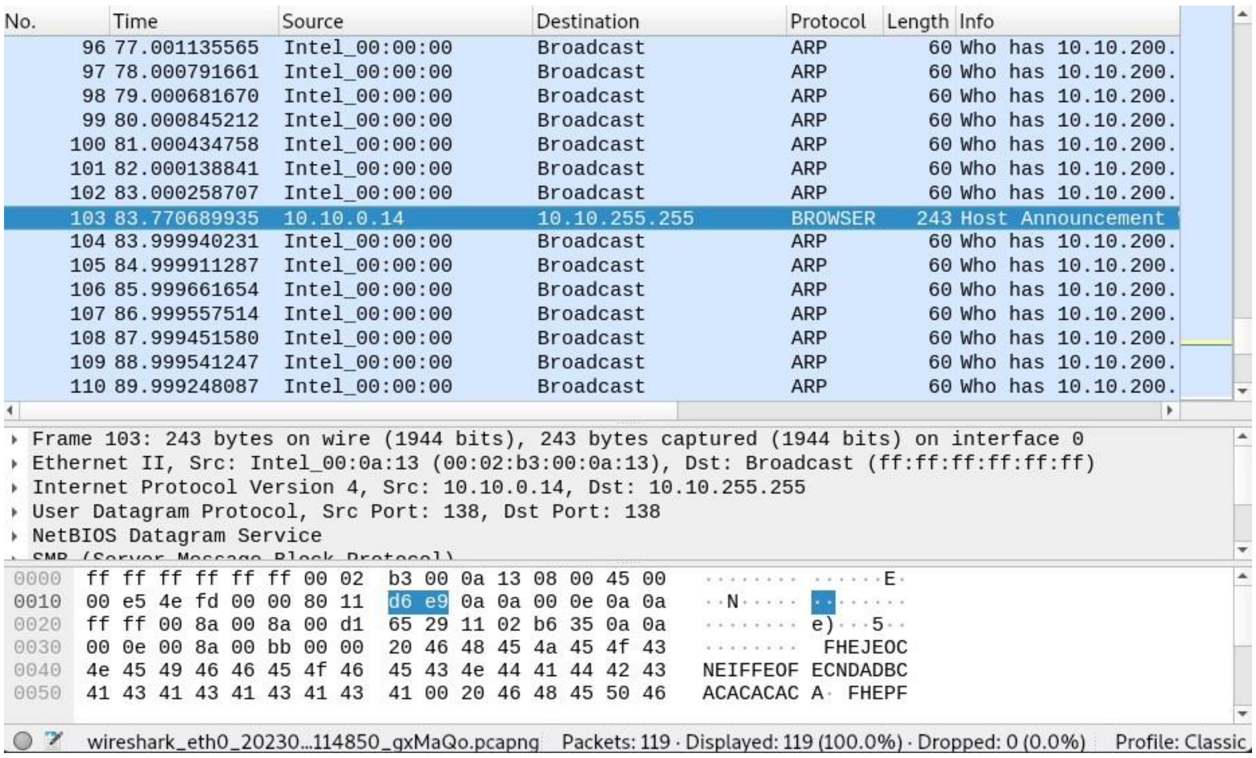
I noticed one of the IPs, 10.10.255.255, from *Figure 30*, they have appeared in my exploration of other application via **kali-hunt-01** VM. So, I opened the terminal via Kali and executed ***ifconfig***.



```
th0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.10.0.100  netmask 255.255.0.0  broadcast 10.10.255.255
        inet6 fe80::202:b3ff:fe00:900  prefixlen 64  scopeid 0x20<link>
        ether 00:02:b3:00:09:00  txqueuelen 1000  (Ethernet)
        RX packets 2438255  bytes 186618674 (177.9 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 53377  bytes 61666322 (58.8 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

*Figure 31: Executed ifconfig via command terminal.*

Later I clicked on the same data packets and analyzed the contents in the Packet Details and the Packet Bytes pane (*Figure 32*).



```
No.   Time          Source           Destination     Protocol  Length Info
   96 77.001135565  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
   97 78.000791661  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
   98 79.000681670  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
   99 80.000845212  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
  100 81.000434758  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
  101 82.000138841  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
  102 83.000258707  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
  103 83.770689935  10.10.0.14       10.10.255.255   BROWSER      243 Host Announcement
  104 83.999940231  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
  105 84.999911287  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
  106 85.999661654  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
  107 86.999557514  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
  108 87.999451580  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
  109 88.999541247  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.
  110 89.999248087  Intel_00:00:00   Broadcast       ARP           60 Who has 10.10.200.

▸ Frame 103: 243 bytes on wire (1944 bits), 243 bytes captured (1944 bits) on interface 0
▸ Ethernet II, Src: Intel_00:0a:13 (00:02:b3:00:0a:13), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▸ Internet Protocol Version 4, Src: 10.10.0.14, Dst: 10.10.255.255
▸ User Datagram Protocol, Src Port: 138, Dst Port: 138
▸ NetBIOS Datagram Service

0000  ff ff ff ff ff ff 00 02  b3 00 0a 13 08 00 45 00   ········ ······E·
0010  00 e5 4e fd 00 00 80 11  d6 e9 0a 0a 00 0e 0a 0a   ··N····· ········
0020  ff ff 00 8a 00 8a 00 d1  65 29 11 02 b6 35 0a 0a   ········ e)···5··
0030  00 0e 00 8a 00 bb 00 00  20 46 48 45 4a 45 4f 43   ········  FHEJEOC
0040  4e 45 49 46 46 45 4f 46  45 43 4e 44 41 44 42 43   NEIFFEOF ECNDADBC
0050  41 43 41 43 41 43 41 43  41 00 20 46 48 45 50 46   ACACACAC A· FHEPF

wireshark_eth0_20230…114850_gxMaQo.pcapng   Packets: 119 · Displayed: 119 (100.0%) · Dropped: 0 (0.0%)   Profile: Classic
```

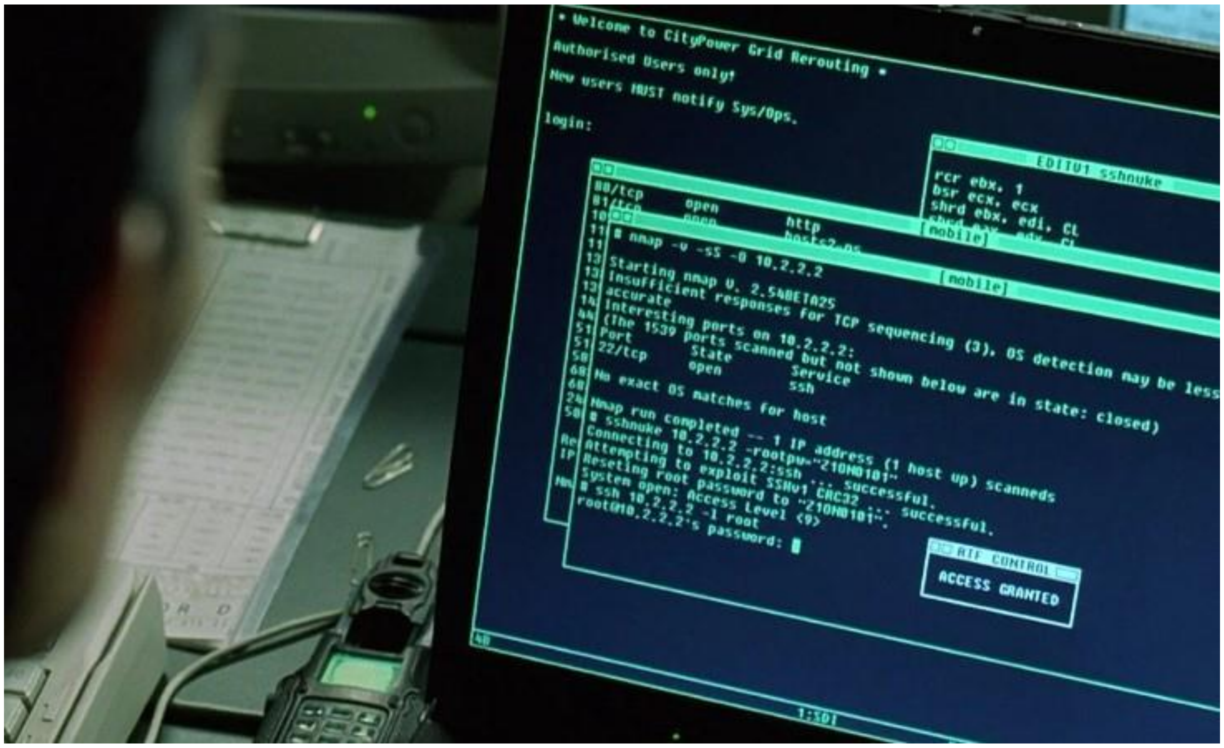
*Figure 32: Analyzing packet details and packet bytes panes.*

From previous courses, I recognized the raw bits in the Packet Bytes pane from *Figure 32*. The left section is a hexadecimal dump of data bytes in the packet and the right side is ASCII interpretation.

The ability of Wireshark to capture raw data from network traffic provides vital insights for security practitioners. This raw data assists in detecting anomalies and potential security threats like malware or DDoS attacks, as well as in understanding and identifying misuse or non-standard implementations of network protocols. Moreover, it can assist in troubleshooting network or application issues and serves as a key forensic tool, supplying critical evidence post-security incidents to prevent similar occurrences in the future. Further, this data enhances network optimization by providing a detailed view of network traffic, thus improving both security and efficiency.

## Bonus

The NMAP utility was used in one of the Matrix trilogies films, specifically The Matrix Reloaded (2003), one of the characters, Trinity, in the movie uses NMAP to change a power station’s root password to gain access (*Figure 20*).



```
* Welcome to CityPower Grid Rerouting *
Authorised Users only!
New users MUST notify Sys/Ops.

login:

# nmap -v -sS -O 10.2.2.2
Starting nmap V. 2.54BETA25
Insufficient responses for TCP sequencing (3), OS detection may be less
accurate
Interesting ports on 10.2.2.2:
(The 1539 ports scanned but not shown below are in state: closed)
Port       State       Service
22/tcp     open        ssh
No exact OS matches for host
Nmap run completed -- 1 IP address (1 host up) scanneds
# sshnuke 10.2.2.2 -rootpw="Z1ON0101"
Connecting to 10.2.2.2:ssh ... successful.
Attempting to exploit SSHv1 CRC32 ... successful.
Reseting root password to "Z1ON0101".
System open: Access Level <9>
# ssh 10.2.2.2 -l root
root@10.2.2.2's password:
```

```
rcr ebx, 1
bsr ecx, ecx
shrd ebx, edi, CL
```

*Figure 33: Scene from The Matrix Reloaded of Trinity using NMAP utility.*
*Image sourced from YouTube (YouTube, 2012).*

## Bibliography

Artykov, D. (2021, March 26). *Scan the network with Armitage*. Retrieved from Medium: https://medium.com/purple-team/scan-the-network-with-armitage-62aaae5631e9#:~:text=Armitage%20is%20a%20scriptable%20apparatus%20for%20Metasploit%20that%20visualizes%20targets,power%20and%20potential%20of%20Metasploit.

C4 Cyber Club. (2021, February 15). *Ethical Hacking Lab - NETLAB+ 09 Metasploit Framework Fundamentals and Armitage*. Retrieved from YouTube: https://www.youtube.com/watch?v=12WR7dv57G0

CompTIA. (n.d.). *What Is Wireshark and How Is It Used?* Retrieved from https://www.comptia.org/content/articles/what-is-wireshark-and-how-to-use-it#:~:text=What%20Is%20Wireshark%20Used%20For,identify%20bursts%20of%20network%20traffic.

HackerSploit. (2018, April 12). *Armitage Kali Linux Complete Tutorial*. Retrieved from YouTube: https://www.youtube.com/watch?v=JALmoY4LuT8

NMAP.org. (n.d.). *Chapter 18. Nping Reference Guide*. Retrieved from nmap.org: https://nmap.org/book/nping-man.html

NMAP.org. (n.d.). *Command-line Flags*. Retrieved from NMAP.org: https://nmap.org/book/output-formats-commandline-flags.html#:~:text=Nmap%20always%20prints%20warnings%20about,more%20warnings%20to%20be%20printed.

Okta. (2022, April 21). *Understanding the Metasploit Project and Why It's Useful*. Retrieved from https://www.okta.com/identity-101/metasploit/

Simplilearn. (2023, February 2). *What is Metasploit: Overview, Framework, and How is it Used*. Retrieved from https://www.simplilearn.com/what-is-metaspoilt-article#:~:text=Metasploit%20is%20a%20powerful%20tool,by%20security%20engineers%20across%20industries.

YouTube. (2012, November 14). *Trinity uses nmap in The Matrix Reloaded*. Retrieved from YouTube: https://www.youtube.com/watch?v=0PxTAn4g20U