CS 3710 Introduction to Cybersecurity
Term: Spring 2025

Lab Exercise 2 – Reconnaissance and Network Scanning Lab

Due Date: January 31, 2025 11:59pm
Points Possible: 7 points

Name:

By submitting this assignment you are digitally signing the honor code, "On my honor, I pledge that I have neither given nor received help on this assignment."

Generative AI assistance is NOT permitted on this assignment.

1. Overview

This lab exercise will provide some hands-on experience with reconnaissance, network scanning, and service enumeration.

2. Resources required

This exercise requires a Kali Linux VM running in the Virginia Cyber Range.

3. Initial Setup

From your Virginia Cyber Range course, select the **Cyber Basics** environment. Click "Start My Environment" to start your environment and once it is ready click "Join My Environment" to open your Linux desktop.

4. Tasks

Task 1: Whois lookups

For this portion of the exercise, you can use a web browser on your laptop or desktop computer, or you can log in to your Cyber Basics environment in the Virginia Cyber Range.

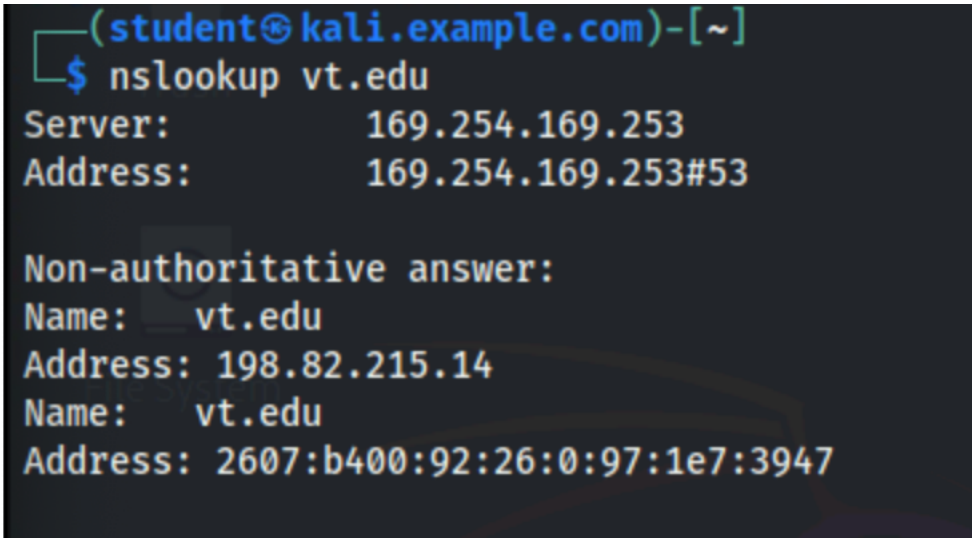
WHOIS is a tool for querying databases containing domain registration data to determine ownership, IP addresses, and other information. A reverse whois lookup can be used to find domains that are registered by a particular individual or organization. ICANN is the authoritative source for WHOIS information, however due to the General Data Protection Regulation (GDPR) a lot of its information is now restricted. Other sources of WHOIS information include https://pk.godaddy.com/whois, and https://whois.domaintools.com/.

Question #1: Do a whois lookup on the domain **jmu.edu**. To whom is the domain registered? What is the administrative contact name, address, email, and phone number? (.5 point)
The contact is Denis Little at Massanutten Hall 265 MSC 5733 Harrisonburg, VA 22807, USA.
The phone is 1 540-568-1676 and the email is littldr@jmu.edu

Task 2: nslookup and dig

Nslookup is a Linux and Windows tool for querying the distributed database that makes up the domain name system (DNS). This database translates host names (such as www.virginiacyberrange.org) to IP addresses (3.167.69.13). This translation is necessary because your computer must have the IP address of systems, such as web servers, that it communicates with, but humans are not good at remembering strings of numbers so we remember hostnames instead. DNS converts hostnames to the proper IP address so your web browser can find that web page. This DNS lookup usually happens in the background so users don't realize it is happening. You can use the nslookup tool to do this mapping from the command line.

Question #2: Use **`nslookup`** to find the IP address for vt.edu. What is the IPv4 address? Provide a screen shot and explain where you found the answer. There are multiple IP addresses in the output so you may need to research how to interpret the results. (.5 point)
The IPV4 is 198.82.215.14



Dig is another, and generally more powerful, tool for DNS database queries. However, dig is only available on Linux and Unix systems.

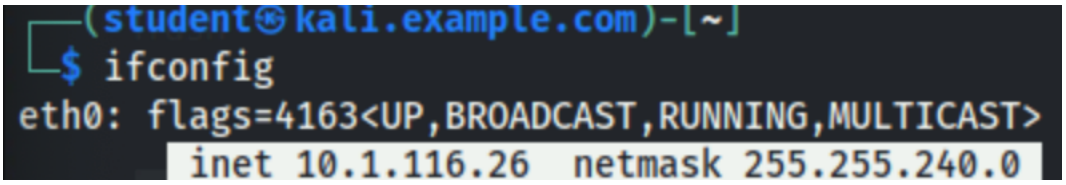
Question #3: Examine the Linux 'man page' for the dig utility to find more information about dig. What does the '**`-x`**' command-line option do in dig? (.5 point) the -x performs a reverse lookup where an ipv4 and ipv6 address can be entered and the respective domain can be found

Question #4: Use dig to conduct a reverse lookup of the IP address 134.126.20.33. What is the hostname or hostnames correspond with that IP address? (.5 point)w3.cs.jmu.edu and cs.jmu.edu

Task 3: Network scanning using nmap

Your Kali Linux virtual machine in the Virginia Cyber Range is connected to a small network subnet with other systems. Your first step in this exercise is to understand your network neighborhood.

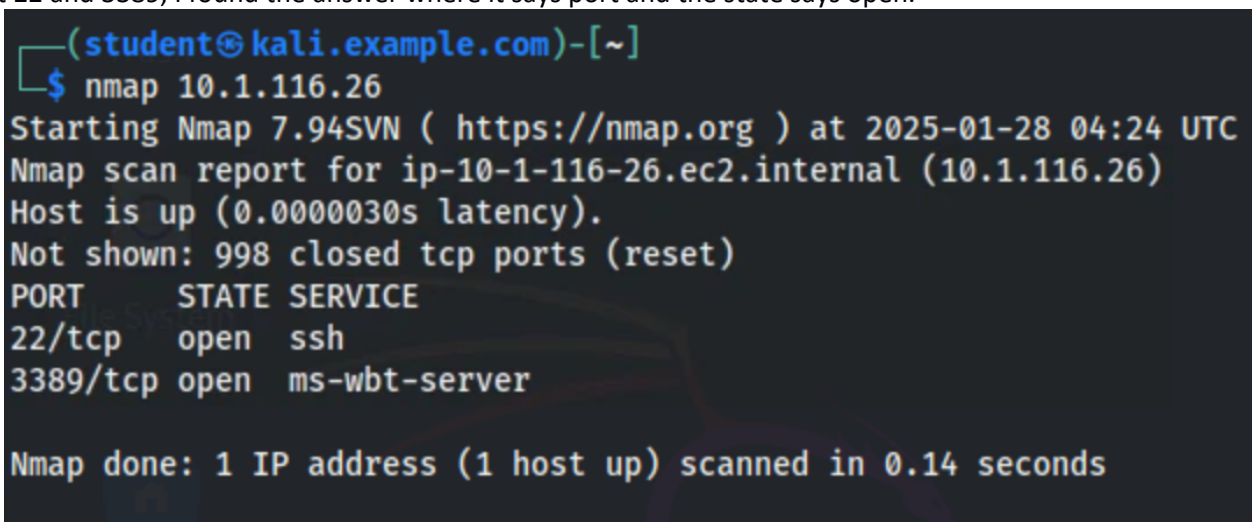
Question #5: What is your IPv4 address and netmask? (.5 point)
Inet: 10.1.116.26 nemask: 255.255.240.0



There are different ways to accomplish host discovery on a network. For this exercise we will use Nmap (https://nmap.org/book/man.html), a widely used tool for network exploration and port scanning. Nmap can be used to scan a single hostname or IP address or range of addresses. You can learn more about Nmap through the man page (**man nmap**) or simply type **nmap** with nothing else and hit enter to see a summary of command options and usage. To scan a single host you would use the following command:

```
$ nmap <options> <hostname or IP address>
```

Question #6: Run an nmap scan against your own IP address. What ports are open? Provide a screenshot and explain or show where you found your answer. (.5 point)
Port 22 and 3389, I found the answer where it says port and the state says open.



Ping scan. Let's see what other systems are on the network by using Nmap's ping scan. Nmap has a ping scan option that simply sends a ping packet to each IP address and listens for replies to identify active hosts. For this scan you will scan your network using CIDR notation which looks like the following: **`your_IP_address/CIDR`**

You will replace **`your_IP_address`** with your actual IP that you identified in Question #5. The second part is to replace the **`CIDR`** with the actual CIDR notation for your network. Use your Google skills to find the CIDR notation of your network based on your netmask found in Question #5 and replace the word **`CIDR`** with it to scan the entire network where your system lives. Don't forget to give nmap the **ping scan only** option!

Question #7: How many IP addresses did you discover on the network? Provide a screenshot and explain or show where you found your answer. (.5 point) 4096. At the bottom it says 4096 IP addresses.

```
└─$ nmap -sn 10.1.116.26/20
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-28 04:29 UTC
Nmap scan report for ip-10-1-112-1.ec2.internal (10.1.112.1)
Host is up (0.000099s latency).
MAC Address: 0A:FC:9E:A0:86:AC (Unknown)
Nmap scan report for ip-10-1-112-2.ec2.internal (10.1.112.2)
Host is up (0.000078s latency).
MAC Address: 0A:FC:9E:A0:86:AC (Unknown)
Nmap scan report for ip-10-1-120-46.ec2.internal (10.1.120.46)
Host is up (0.00037s latency).
MAC Address: 0A:FF:CF:C2:DC:6B (Unknown)
Nmap scan report for ip-10-1-121-238.ec2.internal (10.1.121.238)
Host is up (0.00011s latency).
MAC Address: 0A:FF:D9:DF:B5:93 (Unknown)
Nmap scan report for ip-10-1-123-48.ec2.internal (10.1.123.48)
Host is up (0.00013s latency).
MAC Address: 0A:FF:C2:74:B3:FD (Unknown)
Nmap scan report for ip-10-1-116-26.ec2.internal (10.1.116.26)
Host is up.
Nmap done: 4096 IP addresses (6 hosts up) scanned in 9.65 seconds
```

Port scan. By default, **`nmap`** will conduct a port scan of the target address(es), trying to connect to ports 1 – 1000 for each IP address scanned and report which ports it finds open, or "listening". Now that we have identified potential target systems we will scan them to identify open networking ports. Use **`nmap`** with the **--open** option to port scan the hosts again to see what targets have open ports.

Question #8: List each IP address and the port numbers and services open on each system. (.5 point); ftp

1. 10.1.112.2; 53; domain
2. 10.1.120.46 ; 21; ftp
3. 10.1.121.238; 80 ; http

4. 10.1.123.48; 22,80,139,445; ssh, http netbios-ssn, microsoft-ds respectively
5. 10.1.116.26; 22,33889;ssh and ms-wbt-server respectively

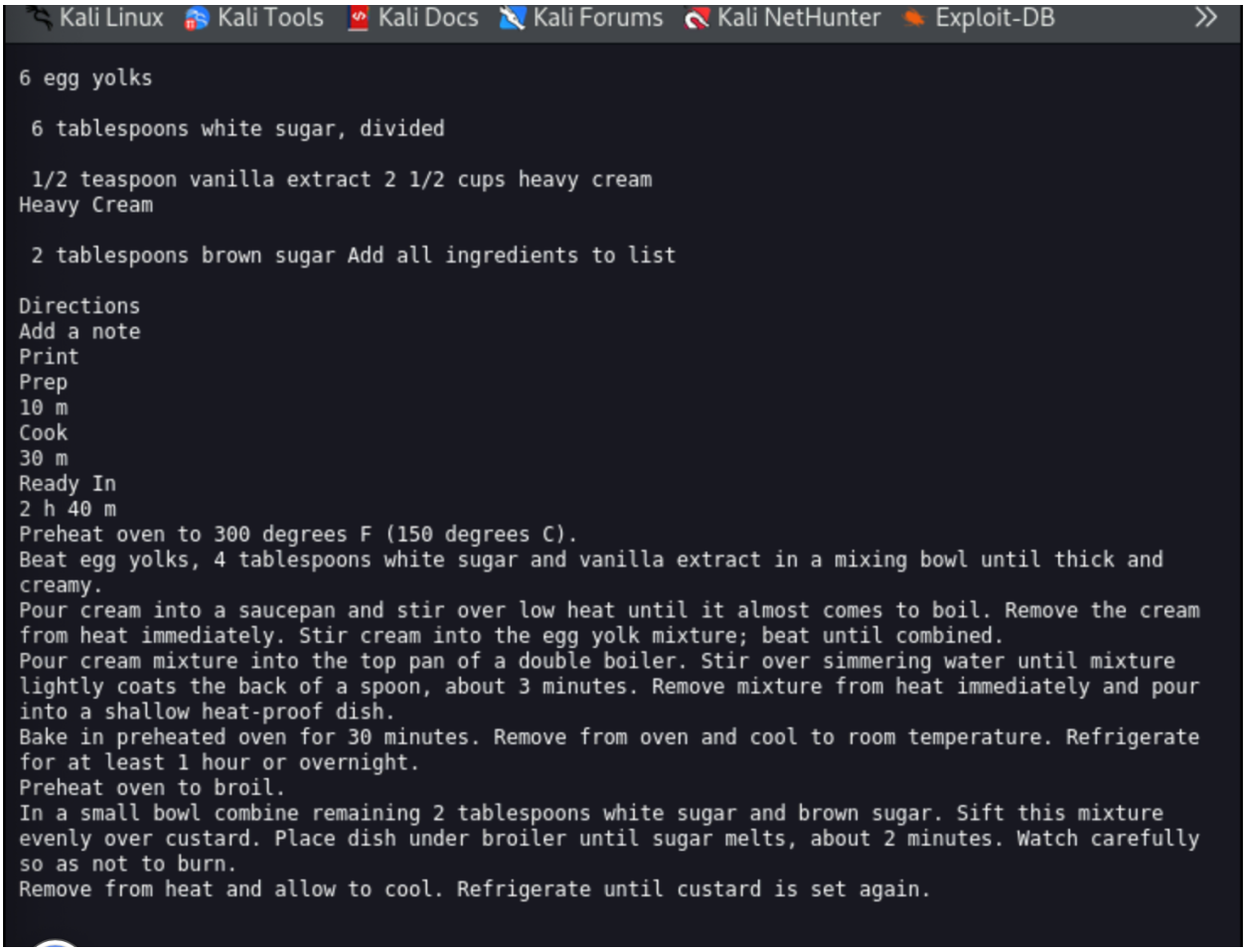
Question #9: Which systems (IPs) are possibly running a web server? Explain along with a screenshot. (.5 point)

10.1.121.238 and 10.1.123.48 because they are running http and port 80 is open.

```
Nmap scan report for ip-10-1-121-238.ec2.internal (10.1.121.238)
Host is up (0.00036s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE
80/tcp open  http
MAC Address: 0A:FF:D9:DF:B5:93 (Unknown)

Nmap scan report for ip-10-1-123-48.ec2.internal (10.1.123.48)
Host is up (0.00030s latency).
Not shown: 996 closed tcp ports (reset)
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 0A:FF:C2:74:B3:FD (Unknown)
```

Question #10: Use this web server information to discover a secret family recipe. Include a screenshot. (.5 point)



Question #11: **Version detection**. Now we need to look a little more to find out specifics about the open services you detected. Run an Nmap scan against each target that will perform version detection and show service versions. (there is more than one option that can do this) List all service versions that you find for each IP address. Not all ports will have a specific numerical version, but it will list more information about the service being used, so include that too. Make sure to use the **--open** command line option to filter out systems with closed ports. (1 point)

- 10.1.112.2 → (unknown banner: EC2 DNS)
- 10.1.120.46 → csftpd 2.0.8 or later
- 10.1.121.238 → Golang net/http server (Go-IPFS json-rpc or InfluxDB API)
- 10.1.123.48 → 22:OpenSSH 8.2p1 Ubuntu 4ubuntu0.8 (Ubuntu Linux; protocol 2.0); 139: Samba smbd 3.X – 4.X(workgroup: MYGROUP);445: Samba smbd 3.X – 4.X(workgroup: MYGROUP
- 10.1.116.26 → 22: OpenSSH 9.7p1 Debian 5(protocol 2.0); 3389: xrdp

Question #12: Pulling it all together. Scanning is the first step to identify active targets, which we did in Question #7 and then to identify open ports and services, which we did in Question #8. By performing version detection like we did in Question #11 we can start to identify potential vulnerabilities. One of the targets you scanned has a File Transfer Protocol (FTP) server running, which is a vulnerable way to

transfer files. The **`nmap -A`** scan can give you some really valuable information for logging into that FTP server. Exploit the anonymous FTP login and retrieve a file from the server and paste its contents here. (1 point)
Welcome to Cyber Range FTP Server

By submitting this assignment you are digitally signing the honor code, "I pledge that I have neither given nor received help on this assignment".

END OF EXERCI

References

- http://viewdns.info/
- https://nmap.org/book/man.html
- https://en.wikipedia.org/wiki/Port_(computer_networking)
- https://en.wikipedia.org/wiki/Classless_Inter-Domain_Routing