



# Hacking The Matrix

## 0 Day



Lead - Friday  
Co-lead - Issac



# General Agenda

1. Intro
2. Attendance Sheet & Cyberlab machine sign up
3. Sign The WhiteHat Agreement!
4. Penetration Testing Methodology
5. Focus On Recon Phase
6. Introduce Common Network Reconnaissance Tools
7. Using AI/Confirming via RTFM
8. Practicing What We've Learned



# ~ \$ id

## Friday

- I make/mix music (DnB, Breakcore, Jungle)
- I'm colorblind
- I like AI
- Here's my github
  - <https://github.com/peroxidee>



## Issac

- I do cyber stuff
- I am not colorblind
- My girlfriend is in the audience
- Check out my website:  
<https://1d8.github.io>





# Sign the attendee list

Sign or we do not eat tonight ):

<https://shorturl.at/hkxz9>

As you sign up, we'll assign

Cyberlab machines to you





What is hacking?

# Penetration Testing/Pentest Methodology



# What is Reconnaissance

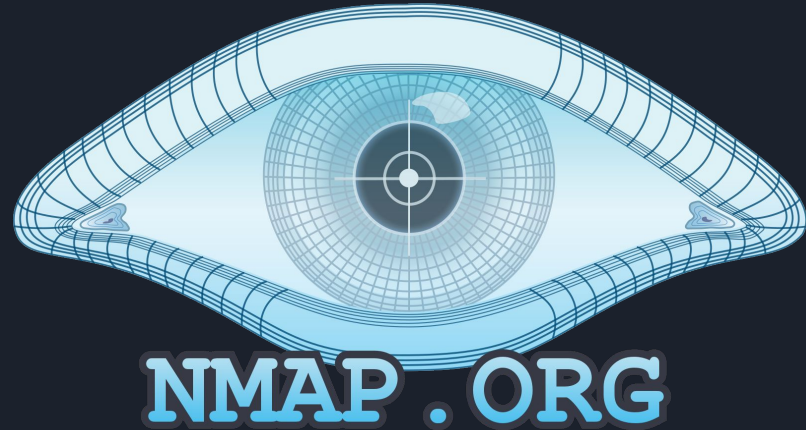


- In military terms, it's exploring an area to gain info about enemy force, their terrain, & civil activities within their area
- Military definition still applies!
  - Enemy force - Our target organization/machine
  - The Terrain - Their network
  - Civil Activities - Normal traffic on their network, helps us blend in
    - For physical pentest, extremely useful too so you don't look suspicious & can tailgate someone easily. Can also help w/your pretext


# What Tools Are Suitable For Reconnaissance?

- For Network Reconnaissance:

- Nmap
- Wireshark
- Nessus
- FFuF
- Gobuster
- CobaltStrike







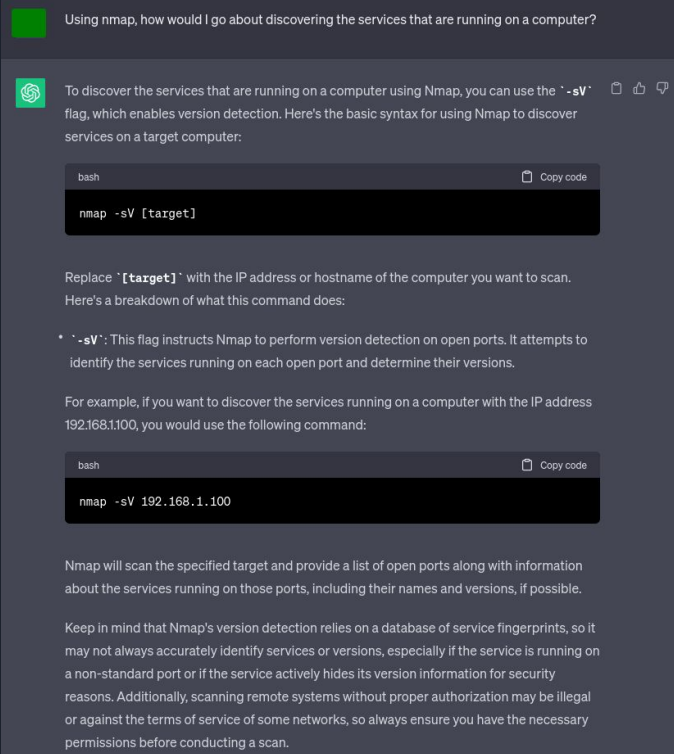
# How Could We Use AI To Help With Reconnaissance?

- When we're running various tools, there are **FLAGS** or **SWITCHES** used in combination with the tool itself
  - These **flags** help find more info & perform additional helpful tasks
- But what do all the **flags** mean???????
- Memorizing each flag & their function is nearly impossible
  - We use ChatGPT to help
- Just cause we use AI, doesn't mean we fully trust it!
  - Cross-reference AI by RTFM
  - Man pages are a godsend, extremely useful



# Using ChatGPT To Learn Functions of Flags

- **Scenario: Discover the services running on a machine**
- How to do this ChatGPT?
- ChatGPT says **-sV**, hmmm is Mr.GPT lying to us?



Using nmap, how would I go about discovering the services that are running on a computer?

To discover the services that are running on a computer using Nmap, you can use the `-sV` flag, which enables version detection. Here's the basic syntax for using Nmap to discover services on a target computer:

```
bash
nmap -sV [target]
```

Replace `[target]` with the IP address or hostname of the computer you want to scan. Here's a breakdown of what this command does:

- \* `-sV`: This flag instructs Nmap to perform version detection on open ports. It attempts to identify the services running on each open port and determine their versions.

For example, if you want to discover the services running on a computer with the IP address 192.168.1.100, you would use the following command:

```
bash
nmap -sV 192.168.1.100
```

Nmap will scan the specified target and provide a list of open ports along with information about the services running on those ports, including their names and versions, if possible.

Keep in mind that Nmap's version detection relies on a database of service fingerprints, so it may not always accurately identify services or versions, especially if the service is running on a non-standard port or if the service actively hides its version information for security reasons. Additionally, scanning remote systems without proper authorization may be illegal or against the terms of service of some networks, so always ensure you have the necessary permissions before conducting a scan.

# Cross-Referencing By RTFM

- Access the manual of nmap by running **man nmap**

- Type **/-sv** to search for the flag -sv in the man page

```
Hacking The Matrix
~ $ man nmap

NAME
  nmap - Network exploration tool and security / port scanner

SYNOPSIS
  nmap [Scan Type...] [Options] {target specification}

DESCRIPTION
  Nmap ("Network Mapper") is an open source tool for network
  exploration and security auditing. It was designed to rapidly
  scan large networks, although it works fine against single
  hosts. Nmap uses raw IP packets in novel ways to determine
  what hosts are available on the network, what services
  (application name and version) those hosts are offering, what
  operating systems (and OS versions) they are running, what
  type of packet filters/firewalls are in use, and dozens of
  other characteristics. While Nmap is commonly used for
  security audits, many systems and network administrators find
  it useful for routine tasks such as network inventory,
  managing service upgrade schedules, and monitoring host or
  service uptime.

  The output from Nmap is a list of scanned targets, with
  supplemental information on each depending on the options
  used. Key among that information is the "interesting ports
  table". That table lists the port number and protocol,
  service name, and state. The state is either open, filtered,
  closed, or unfiltered. Open means that an application on the
```

```
Hacking The Matrix

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)
  --version-trace: Show detailed version scan activity (for debugging)
```

# ChatGPT Didn't Lie To Us



# Another Reason Flags Are Useful

- If we were to run **nmap** without using any flags, our scan is quite basic & doesn't tell us much besides what ports are open & what potential services may be running
- We aren't using the tool to its full potential when we use it w/o flags
- Flags make our job as pentesters easier

```
Hacking The Matrix

~ $ nmap 192.168.56.113
Starting Nmap 7.80 ( https://nmap.org ) at 2023-09-14 08:28 PDT
Nmap scan report for 192.168.56.113
Host is up (0.00031s latency).
Not shown: 991 filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
445/tcp    open  microsoft-ds
631/tcp    open  ipp
3000/tcp   closed ppp
3306/tcp   open  mysql
8080/tcp   open  http-proxy
8181/tcp   closed intermapper

Nmap done: 1 IP address (1 host up) scanned in 4.84 seconds
```



# Let's Get Our Hands Dirty

- Go to <https://cyberlab.csusb.edu>
- Go to your kali machine
- Perform an nmap scan on: <INSERT METASPLOITABLE IP ADDRESS HERE>

## Tasks:

1. What SMB version is running on the target?
  - a. 4.3.11
2. What OS is running on the target?
3. How many ports are closed?
4. What is the MAC address of the target?