

# Penetration Testing: Introduction and Reconnaissance

CSA - September 2022





# **Introduction to Penetration Testing**





# Penetration Testing Overview

- What?
  - Good guys hack you to tell you how they did it so you know what to improve
- Who?
  - Either 3rd party hired or in-house red team
- How?
  - Black box - no insight into systems, white box - full visibility into environment
- When?
  - Some laws/compliance frameworks require organizations have it done annually or even more often
- Why?
  - You need a fresh set of eyes to pull out vulnerabilities that you hadn't noticed before

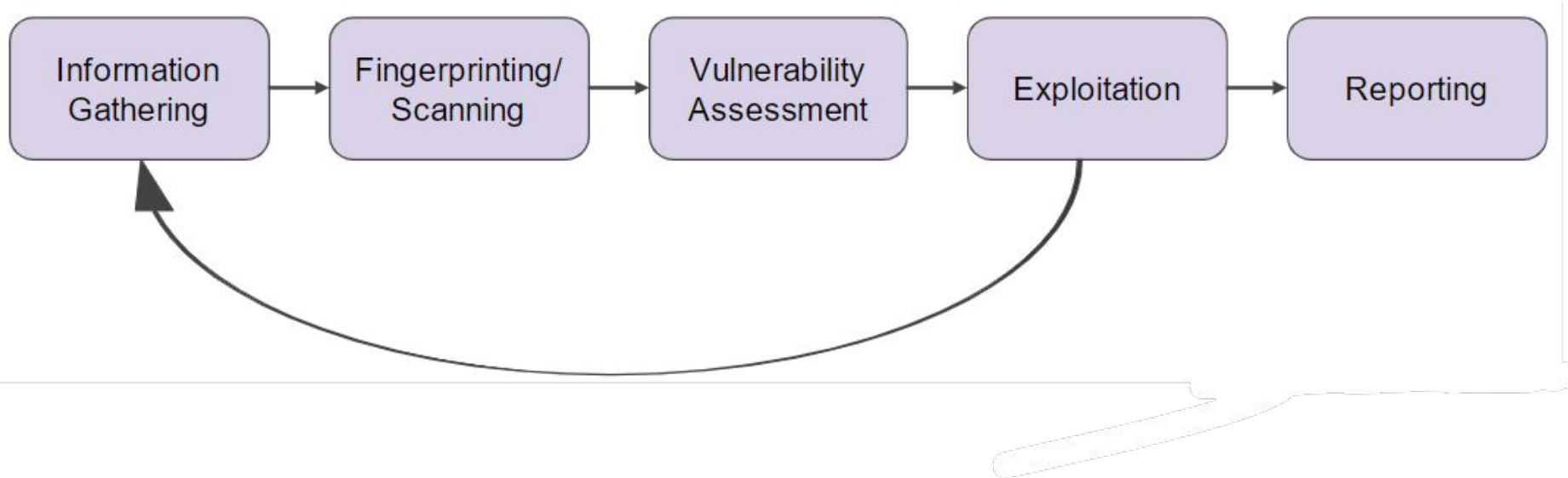


# Steps of Penetration Testing





# Penetration Testing Steps



*There is a step 0, but we won't focus on it here*



# Information Gathering

- What specific assets are in scope?
  - If given an IP range, which are active?
  - If given a wildcard subdomain scope (like \*.byu.edu), what are subdomains?
- OSINT
  - How is the company set up?
  - Do they have a dedicated IT department?
  - What do their emails look like?
  - What do they do/what service(s)/product(s) do they provide?
- This helps you enumerate your attack surface and give first insight into the security of the company



# Fingerprinting/Scanning

- What Operating System does it use?
  - Not just “Windows” or “Linux”, but exact OS and kernel/release versions
- What ports are open? What’s running on those ports?
- What type of web/SSH/FTP/SMTP/etc server does it use?
  - Again, exact versions are much more helpful because it can point you to specific CVEs to look for
- **Custom? Say no more**
  - It’s typically much easier to hack custom applications than widely-used ones
- Passive vs Active



# Vulnerability Assessment

- Once you know what services are running, what attacks can you perform?
  - If web, think SQLi/XSS/file uploads/etc.
  - If network, think null sessions/anonymous access
- Are there any known CVEs for the known version numbers?
- Can you brute force credentials?
- What are the default credentials?
- Run vulnerability scanners, which can automate a lot of this process for you
  - All results from these scanners need triaging, and only about half of the results are actionable
  - Nessus, Nikto, OpenVas are some examples
- Create a comprehensive list of ***possible vulnerabilities*** based on what you've found - exploitation is next phase. Don't rush it!





# Exploitation

- Once possible vulnerabilities are listed, choose the most likely option and pursue it!
- If exploitation is successful, you travel back to the reconnaissance phase
  - What else can you see now? What user accounts are there? Interesting files? Credentials?
- Privilege Escalation
  - Can you become admin/root? Can you pivot to other accounts?
- Network pivoting
  - What networks does the machine have on it? What increased access do you have to other systems?
- Establish persistence - what changes can you make to ensure you have access even if the machine is rebooted/web shell discovered/connection broken?



# Reporting

- **MOST IMPORTANT PART OF THE JOB**
  - If you can hack them but can't effectively convey how you did it or what to do to stop it, you are of no use to them. Effective reports are your job security.
- Include an overview of how their infrastructure withstood your pentests, major takeaways, etc.
- A large part of the pentest report afterwards includes detailing how to reproduce vulnerabilities, why the vulnerability is so important, and how to mitigate it
- Not a huge part of what we focus on because it's probably best if you learn from your company how they make the reports



# Reconnaissance





# Information Gathering

- Enumerating subdomains
  - **Depends on scope** - use tools like sublist3r, DNSDumpster, and crt.sh to find
  - theHarvester is a common choice also
  - Can brute force with a wordlist
- Enumerating IP addresses
  - Ping scan - which machines in the given IP range are up and responding?
- OSINT
  - What email format does the organization use?
  - Who are some employees? Do they have any secrets on their social media?
  - Who are the executives?
  - What is the layout of the company? Do they have a dedicated IT department?



# Passive vs Active Scanning

- Active - more effective, faster, more noisy
- Passive - slower, less effective, more stealthy
- Which do I choose?
  - How likely are you to be caught?
  - Do they have an IDS that will detect you?
  - ***Custom == More Stealthy***
- Passive techniques/tools
  - Capturing WiFi traffic w/ wifi pineapple
  - Using Wireshark/p0f to capture & analyze network traffic not intended for you
  - Google dorks



# Ping Scan

- “Ping” is a layer 3 protocol that sends an ICMP Echo packet to an IP and expects a response from machines that are running
  - Ping can be disabled on machines, but not very likely
- ARP packets can also be sent - these operate on layer 2 and try to resolve IP addresses into MAC addresses
  - If ping is disabled, ARP may work
  - ARP resolution is done by default when nmap is run with sudo
- Nmap
  - -sn -> ping/ARP only
  - -Pn -> no ping

```
$ sudo nmap -sn 192.168.126.0/24
[sudo] password for justin:
Starting Nmap 7.92 ( https://nmap.org ) at 2022-08-26 19:13 MDT
Stats: 0:00:17 elapsed; 0 hosts completed (0 up), 256 undergoing Ping Scan
Ping Scan Timing: About 8.30% done; ETC: 19:16 (0:03:08 remaining)
Nmap scan report for 192.168.126.128
Host is up (0.0013s latency).
Nmap done: 256 IP addresses (1 host up) scanned in 21.12 seconds
```

Somewhat quick

REALLY quick

```
$ fping -g 192.168.126.0/24 2> /dev/null
192.168.126.128 is alive
192.168.126.1 is unreachable
192.168.126.2 is unreachable
192.168.126.3 is unreachable
192.168.126.4 is unreachable
192.168.126.5 is unreachable
192.168.126.6 is unreachable
192.168.126.7 is unreachable
192.168.126.8 is unreachable
192.168.126.9 is unreachable
```



# Fingerprinting - Operating Systems

- Nmap has a -O option that will guess an operating system based on minutiae in the network packets
  - Use the --osscan-guess option afterwards so it will tell you which OSes are likely even if it's not 100% certain
  - Somewhat aggressive, but also effective
- Presence of running services may indicate which OS is running
  - Ports 135-139, 445, 3389 -> Windows
  - Port 22 -> Linux
  - IIS web server -> Windows
- Web servers and SSH servers may include the OS name in their banner
- Knowing exact OS versions can help later on in exploitation phase





# Port Scanning

- Understand TCP handshake (difference between SYN scan and TCP scan)
- Which port numbers
  - Nmap does top 1000 by default
  - -p used to specify port number
    - -p 80
    - -p 22,80,440-443
    - -p- (for all 65535)
- UDP ports
  - Not done by default, -sU enables this
- Fingerprint services -> -sV (grabs banners)
  - Netcat can do the same thing for some services



# Fingerprinting - Web Servers/Languages

- Many servers have their exact server version as the Server header in the HTTP request **by default**
  - However, can be removed or changed
- Can tell by file extensions or debug messages
- Not all 404/403/401 error messages are the same if you look at the HTML level
- Tools like httpprint can also do this work for you
- The structure of the website can also tell you



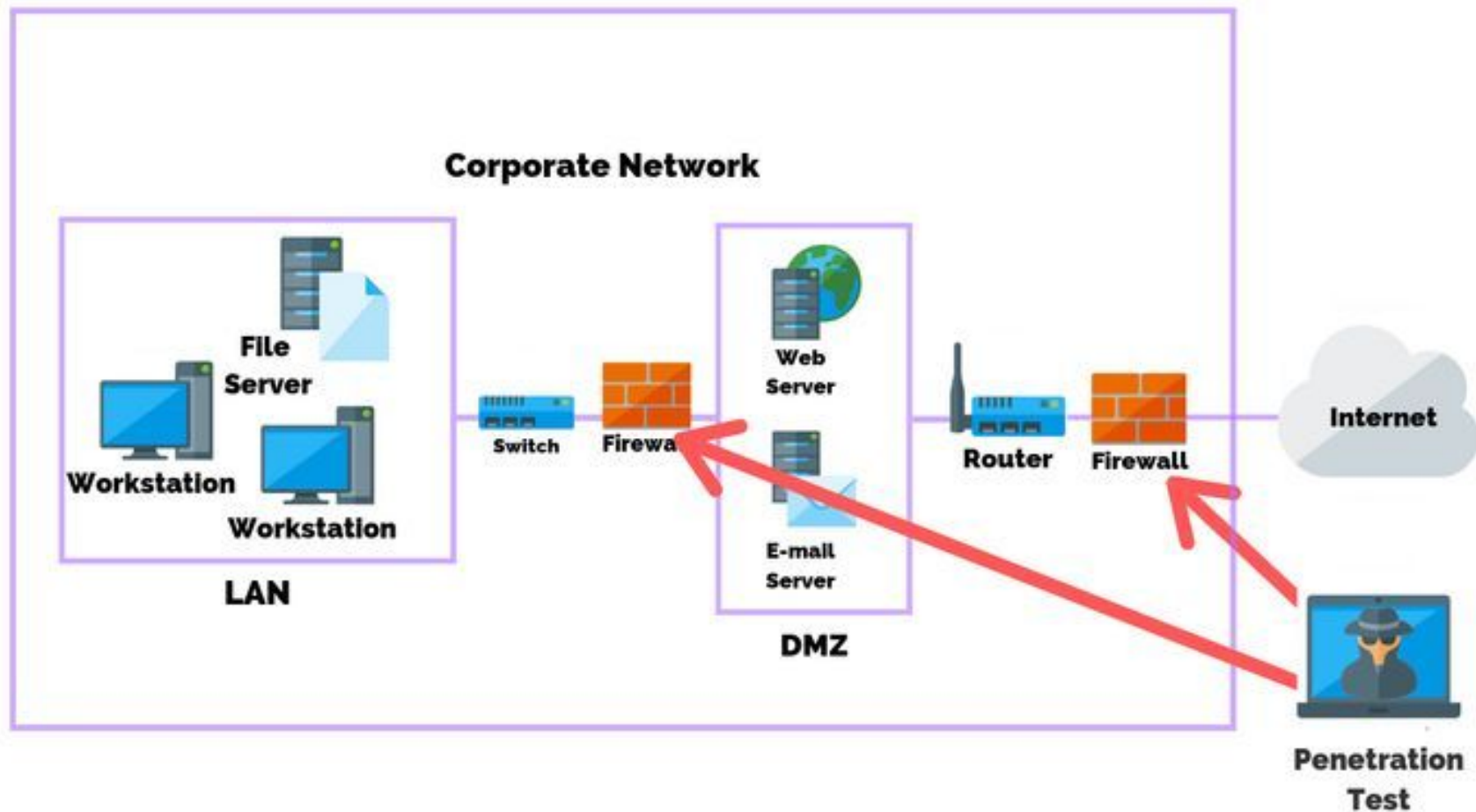
# Nmap - Miscellaneous

- Nmap scripts
- -sT vs -sS
  - -sT is a full TCP handshake, whereas -sS is only a SYN packet
- “Open” vs “closed” vs “tcpwrapped” vs “filtered”
  - Open - SYN/ACK returned, Closed - RST returned, tcpwrapped - TCP handshake complete but immediately terminated, Filtered - No response at all
- -A (all)
  - Run -O, -sV, all Nmap scripts, traceroute, etc.
- -T#
  - Range is T1 to T5 (T3 is default), with T1 as slowest (1 probe every 5 minutes or so) and T5 as the fastest (but least stealthy)



# Network Diagram

- After discovering which machines are available at what IP addresses with which OSes and services running, make a diagram!
- Perfect at this point in time (and not afterwards) because it ensures that you don't forget any assets as you get into the fun part of exploiting
- Use online tools like draw.io, LucidChart, or if you're like me... PowerPoint!





# Activity





# Recon CTF

You have until 8:30 to solve all of the problems. Go to [byu.ctfd.io](https://byu.ctfd.io), register, and start scanning!

Note - more challenges will be available in each category after the MAC challenge is solved.

Questions? Ask Kylie or Justin.

Flag format - recon{answer}