



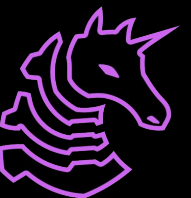
SP2025 Week 01 • 2025-01-28

Network Security & Active Recon

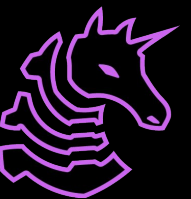
Ronan Boyarski

Table of Contents

- Review of general network security
- Active Recon
 - Port scanning
 - Enumerating services
 - Edge cases (proxies, UDP)
- Gaining Access
 - Services & Misconfigurations
 - Exploiting known vulnerabilities
 - Exploitdb, Searchsploit, GitHub, Metasploit
 - Password Brute Forcing & Password Spraying
- GOAD Live

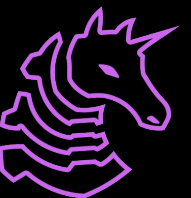


Network Security Overview



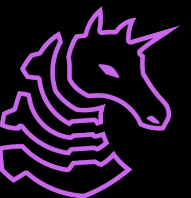
Network Security Overview

- Generally, we're concerned about the security of **services**
 - A service is a piece of software that's doing something important and remotely accessible on a target computer, like remote logins, file sharing, etc.
 - Examples include default system ones (like SMB) as well as user-configured ones (FTP, web, etc)
 - Going to have different **protocols** on different **default ports**
 - The type of service is going to have a huge impact on the kinds of things we can do with it
 - We're looking to chain **primitives** (like PWN)
 - Can we read/write files, modify users, make network connections?



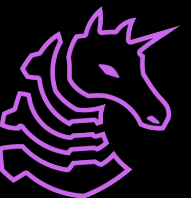
Network Security Overview

- How can we tell what a computer is running?
 - Port scanning
 - Service scanning
 - Interacting with target services
- How can we get in?
 - Known vulnerable versions
 - Misconfigurations
 - Brute force

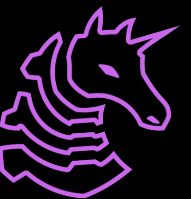


General Attack Flow

- Start with OSINT, begin mentally modeling the target
- The goal is to refine that model with active reconnaissance
- The flow is going to be scanning ports, identifying services, and then looking for vulnerabilities & misconfigurations
- Typically, we will start with **nmap** or 🦀 **rustscan** 🦀
 - rustscan is super fast but basic and not stealthy
- If you're having trouble gaining access, ensure that you've actually looked for as many ways in as possible

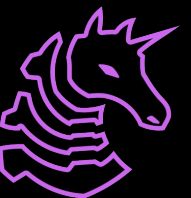


Active Recon



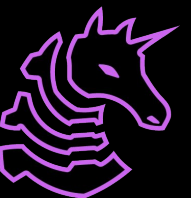
Port Scanning

- Port range: 0-65535, TCP & UDP
- `sudo nmap -Pn -F -sV -vv $IP -oN fast.txt`
- `sudo nmap -Pn -A -sV -p- -vv $IP -oN full.txt`
- `-Pn` skips the ping check
- `-A` means that nmap will run scripts and OS fingerprinting
- `-sV` will have the scan perform version checking
- `-p-` will scan every single port from 1-65535
- `-vv` will enable very verbose output
- `-oN` saves the result to a text file so you don't re-scan



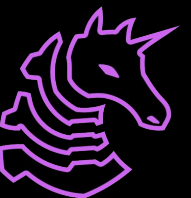
Port Scanning - Edge Cases

- Don't forget UDP services like SNMP!
- `sudo nmap -Pn -F -sU -vv $IP -oN udp.txt`
- `-sU` will have the scan check UDP ports
- `-F` will scan top 1000 ports (UDP scanning is **SLOW**)
- If you're scanning through a SOCKS proxy, you can only scan TCP ports, and should use the `-sT` flag
 - This does a TCP scan rather than SYN scan
- If you're in a network, do a very fast scan using IP range
- General workflow tip: make a directory for each target

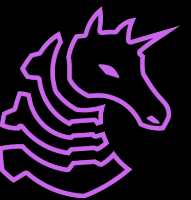


Port Scanning Alternative - Rustscan

- 🦀 **rustscan** 🦀 is a modern, insanely fast alternative to **nmap**
- Can scan all 65535 TCP ports in **as fast as 3 seconds**
- Does direct TCP connections, simpler than nmap
- Ideal for situations where the only thing that matters is speed
- Integrates with nmap for service scanning and script execution
 - Runs nmap only on discovered open ports, best of both worlds

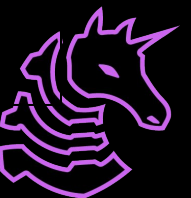


Services



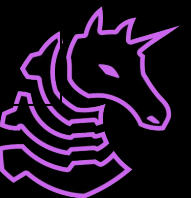
Service Scanning: SMB

- Server Message Block runs by default on all Windows computers (but can be disabled manually)
- Designed for file sharing
- Primitives
 - Authenticated file read & file write
 - **Administrator**: Start arbitrary services
- Vulnerabilities
 - A valid admin login allows you to write a file and start a service running it (this is an example of chaining primitives to get RCE)
 - **MS17-010**, old but gold for anything running unpatched Windows 7
 - Sometimes there are **anonymous** readable / writable shares



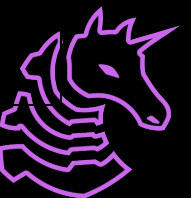
Practical: NXC quick reference

- `nxc smb <IP> -u <username> -p <password> -M <module>`
- List users, groups, and shares
 - `nxc smb <IP> -u <user> -p <pass> -users -groups -shares`
 - This **MAY** be anonymous if misconfigured, but that's uncommon
 - If you run this on a domain controller you'll get domain users & groups
- Scan for coercion vulnerabilities
 - `nxc smb <IP> -u <user> -p <pass> -M coerce_plus`
- Scan for famous Domain Controller vulnerabilities
 - `nxc smb <IP> -u <user> -p <pass> -M nopac -M zero1ogon -M smbghost`



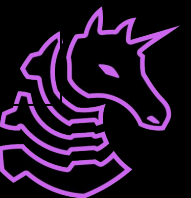
Service Scanning: Other services

- FTP: can be used to upload files or download sensitive files if left unsecured
 - Especially problematic if the folder overlaps with a web server!
- SSH: if you have a password or id_rsa, get a shell as a feature
- Telnet: like SSH, but without the secure part (yikes)
- SNMP: Simple Network Management Protocol, allows viewing all of the running processes, usernames, and software versions, including command-line arguments
- SMTP: Simple Mail Transfer Protocol (email server)
- Redis: Database, can **gain RCE as a feature**

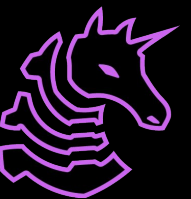


Service Scanning

- This is just the tip of the iceberg
- Get in the flow of understanding unfamiliar services quickly and think in terms of primitives
- <https://book.hacktricks.xyz/> has some good preliminary steps for interacting with and attacking unfamiliar network services
- Web and AD are their own beasts, but remember that it all comes down to chaining primitives
 - Mastering netsec, pwn, web, and AD will give you a huge range of options and lets you chain things together very creatively

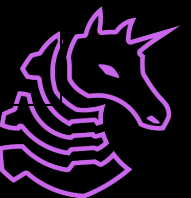


Gaining Access



Gaining Access: Exploitation

- Sometimes, when attacking vulnerable software, it's as easy as running **searchsploit** or the relevant metasploit module
- Other times, custom exploit development is necessary
 - This is where time spend doing traditional CTF is helpful
- Example workflow:
 - nmap -> port 80 is open -> feroxbuster -> find gitlab instance
 - searchsploit gitlab -> find exploit -> read & modify -> execute
- **ALWAYS** read exploit code before running it!
 - Do not let some D-tier cobalt strike "threat actor" onto our GOAD by running precompiled binaries from github



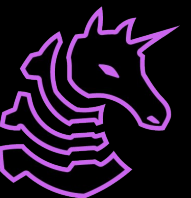
Gaining Access: Misconfiguration

- Sometimes, services are set up with really stupid permissions
- For example, an file server that lets you write to anything in a web server or a user's home directory would be a huge problem
- Example Workflow:
 - nmap -> port 21 & 80 -> unauthenticated FTP server with access to /var/www/html -> put webshell -> browse to port 80 -> get shell
- There are way too many possible misconfigurations to cover here
- **Get in the habit of thinking about what access is appropriate**



Gaining Access: Password Attack

- Lots of common software, like WordPress, or VPN logins, don't rate-limit authentication, so you can go through an obscene amount of login attempts
- **Hydra** is a fantastic general-purpose password attack tool
- Example workflow:
 - nmap -> port 443 -> feroxbuster -> /wp-admin
 - hydra -l Admin -P /usr/share/wordlists/rockyou.txt 10.10.230.209
http-post-form "/wp-login.php:log=^USER^&pwd=^PWD^:The password you entered for the username" -t 30
- Use admin login to upload PHP reverse shell (feature)

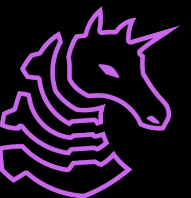


GOAD



GOAD Challenges: NetSec

- 1 network enumeration challenge (Wide Open)
- 1 web challenge (Updog)
- 1 privesc challenge (Hot Potato)
- 1 known vulnerable target (PACman)
- 1 post-exploitation challenge (SAMantha's Security System)
- All of these are designed to mimic OSCP-level challenges
 - They're simple, but not necessarily easy!
 - Defender & Real Time Protection is disabled
 - One of these days I'll make you deal with Elastic EDR + Host Hardening >:)



Next Meetings

2025-02-04 • Next Tuesday

- Linux Privilege Escalation Review - stop crutching on PEAS

