

Scanning and Enumeration

Network Scanning - Discovering systems on the network (can be hosts, switches, servers, routers, firewalls and so on) and looking at what ports are open as well as applications/services and their respective versions that may be running.

In general network scanning have three main objectives:

1. Scanning for live devices, OS, IPs in use.
 - **Server at 192.168.60.30**
2. Looking for Ports open/closed.
 - **The server 192.168.60.30 have TCP port 23 (Telnet) running**
3. Search for vulnerabilities on services scanned.
 - **The Telnet service is cleartext and have many vulnerabilities published**

Connectionless Communication - UDP packets are sent without creating a connection. Examples are TFTP, DNS (lookups only) and DHCP

Connection-Oriented Communication - TCP packets require a connection due to the size of the data being transmitted and to ensure deliverability

Scanning Methodology

- **Check for live systems** - Ping or other type of way to determine live hosts
- **Check for open ports** - Once you know live host IPs, scan them for listening ports
- **Scan beyond IDS** - If needed, use methods to scan beyond the detection systems; evade IDS using proxies, spoofing, fragmented packets and so on
- **Perform banner grabbing** - Grab from servers as well as perform OS fingerprinting (versions of the running services)
- **Scan for vulnerabilities** - Use tools to look at the vulnerabilities of open systems
- **Draw network diagrams** - Shows logical and physical pathways into networks
- **Use proxies** - Obscures efforts to keep you hidden
- **Pentest Report** - Document everything that you find

Identifying Targets

- The easiest way to scan for live systems is through ICMP.

- It has its shortcomings and is sometimes blocked on hosts that are actually live.
- **Message Types and Returns**
 - Payload of an ICMP message can be anything; RFC never set what it was supposed to be. Allows for covert channels
 - **Ping sweep** - easiest method to identify multiple hosts on subnet. *You can automate ping sweep with scripting language like Bash Script (Linux) or PowerShell (Windows) or use softwares like Advanced IP Scanner, Angry IP Scanner, Nmap, etc.*
 - **ICMP Echo scanning** - sending an ICMP Echo Request to the network IP address
 - An ICMP return of type 3 with a code of 13 indicates a poorly configured firewall
 - **Ping scanning tools**
 - **Nmap**
 - `nmap -sn 192.168.1.0/24`
 - This command uses `-sn` flag (ping scan). This will perform a ping sweep on 256 IP addresses on this subnet in seconds, showing which hosts are up.
 - **hping3**
 - `hping -1 10.0.0.x --rand-dest -I eth0`
 - `-1` --> ICMP mode
 - `--rand-dest` --> random destination address mode
 - `-I <interface>` --> network interface name
 - **Angry IP Scanner**
 - **Solar-Winds Engineer Toolkit**
 - **Advanced IP Scanner**
 - **Pinkie**
 - Nmap virtually always does a ping sweep with scans unless you turn it off
- **Important ICMP codes**

ICMP Message Type	Description and Codes
0: Echo Reply	Answer to a Type 8 Echo Request
3: Destination Unreachable	Error message followed by these codes: 0 - Destination network unreachable 1 - Destination host unreachable 6 - Network unknown 7 - Host unknown 9 - Network administratively prohibited

ICMP Message Type	Description and Codes
	10 - Host administratively prohibited 13 - Communication administratively prohibited
4: Source Quench	A congestion control message
5: Redirect	Sent when there are two or more gateways available for the sender to use. Followed by these codes: 0 - Redirect datagram for the network 1 - Redirect datagram for the host
8: Echo Request	A ping message, requesting an echo reply
11: Time Exceeded	Packet took too long to be routed (code 0 is TTL expired)

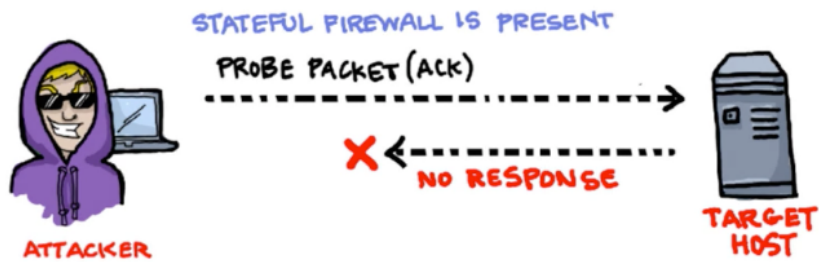
Port Discovery - Basic Concepts

Knocking the door:



- The hacker above sends a SYN packet to port 80 on the server.
 - If server returns **SYN-ACK packet** = the port is **open**
 - If server returns **RST (reset) packet** = the port is **closed**

Checking if Stateful Firewall is present:



- The hacker above sends an **ACK segment/packet** on the first interaction (*without three-way handshake*).
 - If server returns **no response** means that might have a stateful firewall handling proper sessions
 - If server returns **RST packet** means that have no stateful firewall

⚠ This can be easily achieved by using nmap only.

⚠ Keep in mind the TCP Flags & TCP Three-way handshake before use **nmap** !

- 👉 TCP Flags:

Flag	Name	Function
SYN	Synchronize	Set during initial communication. Negotiating of parameters and sequence numbers
ACK	Acknowledgment	Set as an acknowledgement to the SYN flag. Always set after initial SYN
RST	Reset	Forces the termination of a connection (in both directions)
FIN	Finish	Ordered close to communications
PSH	Push	Forces the delivery of data without concern for buffering
URG	Urgent	Data inside is being sent out of band. Example is cancelling a message

- 👉 The TCP Three-way handshake: ([explained in chapter 0 - Introduction](#))

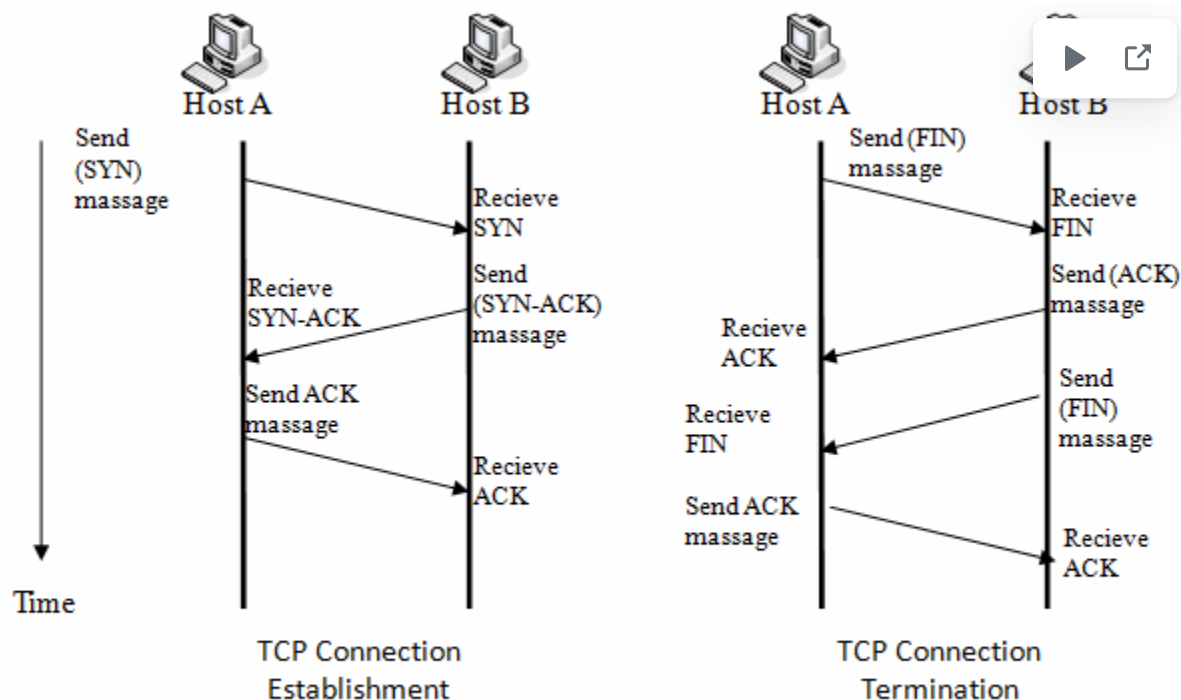


Figure 2.1. TCP session establishment and termination

Nmap

⚠ The CEH exam will definitely cover Nmap questions, about switches and how to perform a specific type of scan.

⚡ It is highly recommended to try out and explore the nmap in your own virtual environment; I made a couple [practical labs\[1\]](#) [\[2\]](#) [\[3\]](#) to help you understand the functionality of nmap.

Nmap ("Network Mapper") is a free and open source (license) utility for network discovery and security auditing. Many systems and network administrators also find it useful for tasks such as network inventory, managing service upgrade schedules, and monitoring host or service uptime. 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.** [\[+\]](#)

Nmap Scan Types:

Stealth Scan

Half-open scan or SYN scan - only SYN packets sent. Responses same as full.

- Useful for hiding efforts and evading firewalls
 - `nmap -sS <target IP>`
-

Full connect

TCP connect or full open scan. The first two steps (SYN and SYN/ACK) are exactly the same as with a SYN scan. Then, instead of aborting the half-open connection with a RST packet, nmap acknowledges the SYN/ACK with its own ACK packet, completing the connection.

- Full connection and then tears down with RST.
 - Easiest to detect, but most reliable
 - `nmap -sT <target IP>`
-


TCP ACK scan / flag probe - multiple methods

- TTL version - if TTL of RST packet < 64, port is open
 - Window version - if the Window on the RST packet is anything other than 0, port open
 - Can be used to check filtering. If ACK is sent and no response, stateful firewall present.
 - `nmap -sA <target IP>` (ACK scan)
 - `nmap -sW <target IP>` (Window scan)
-

NULL, FIN and Xmas Scan

 Uses FIN, URG or PSF flag.

- Open gives no response. Closed gives RST/ACK
- `nmap -sN <target IP>` (Null scan)
- `nmap -sF <target IP>` (FIN scan)
- Xmas Scan - Sets the FIN, PSF, and URG flags, lighting the packet up like a Christmas tree.
 - Responses are same as Inverse TCP scan
 - Do not work against Windows machines
 - `nmap -sX <target IP>`

 The key advantage to these scan types (NULL, FIN or Xmas scan) is that they can sneak through certain non-stateful firewalls and packet filtering routers.

IDLE Scan

uses a third party to check if a port is open

- Looks at the IPID to see if there is a response
 - Only works if third party isn't transmitting data
 - Sends a request to the third party to check IPID id; then sends a spoofed packet to the target with a return of the third party; sends a request to the third party again to check if IPID increased.
 - IPID increase of 1 indicates port closed
 - IPID increase of 2 indicates port open
 - IPID increase of anything greater indicates the third party was not idle
 - `nmap -sI <zombie host> <target IP>`
-

Spoofing

- Decoy:
 - `nmap -Pn -D <spoofed IP> <target>`
 - This will perform a spoofed ping scan.
- Source Address Spoofing:
 - `nmap -e <network interface> -S <IP source> <target>`
 - Example --> `nmap -e eth0 -S 10.0.0.140 10.0.0.165`
- MAC Address Spoofing:
 - `nmap --spoof-mac <MAC|Vendor> <target>`
 - Example --> `nmap --spoof-mac Cis 10.0.0.140`



Decoys will send spoofed IP address along with your IP address.

Firewall Evasion

- Multiple Decoy IP addresses:
 - This command is used to scan multiple decoy IP addresses. Nmap will send multiple packets with different IP addresses, along with your attacker's IP address.
 - `nmap -D RND:<number> <target>`
 - Example --> `nmap -D RND:10 192.168.62.4`
- IP Fragmentation:
 - Used to scan tiny fragment packets
 - `nmap -f <target>`
- Maximum Transmission Unit:

- This command is used to transmit smaller packets instead of sending one complete packet at a time.
 - `nmap -mtu 8 <target>`
 - Maximum Transmission Unit (-mtu) and 8 bytes of packets.
-

Timing & Performance

- **Paranoid**
 - Paranoid (0) Intrusion Detection System evasion
 - `nmap <target> -T0`
 - **Sneaky**
 - Sneaky (1) Intrusion Detection System evasion
 - `nmap <target> -T1`
 - **Polite**
 - Polite (2) slows down the scan to use less bandwidth and use less target machine resources
 - `nmap <target> -T2`
 - **Normal**
 - Normal (3) which is default speed
 - `nmap <target> -T3`
 - **Aggressive**
 - Aggressive (4) speeds scans; assumes you are on a reasonably fast and reliable network
 - `nmap <target> -T4`
 - **Insane**
 - Insane (5) speeds scan; assumes you are on an extraordinarily fast network
 - `nmap <target> -T5`
-

UDP Scan

Most popular services runs over the TCP, but there are many common services that also uses UDP: DNS (53), SMTP (25), DHCP (67), NTP (123), NetBIOS-ssn (137), etc.

- `nmap -sU <target>`

You also can specify which UDP port:

- `nmap -sU -p U:53, 123 <target>`

Also you can fire up both TCP and UDP scan with port specification:

- `nmap -sU -sS -p U:53,123 T:80,443 <target>`

List of Switches

Switch	Description
-sA	ACK scan
-sF	FIN scan
-sI	IDLE scan
-sL	DNS scan (list scan)
-sN	NULL scan
-sO	Protocol scan (tests which IP protocols respond)
-sP or -sn	Ping scan
-sR	RPC scan
-sS	SYN scan
-sT	TCP connect scan
-sW	Window scan
-sX	XMAS scan
-A	OS detection, version detection, script scanning and traceroute
-sV	Determine only service/version info
-PI	ICMP ping
-Pn	No ping
-Po	No ping
-PS	SYN ping
-PT	TCP ping
-oN	Normal output
-oX	XML output
-n	Never do DNS resolution/Always resolve
-f	--mtu : fragment packets (optionally w/given MTU)
-D	IP address Decoy: <decoy1,decoy2[,ME],...>: Cloak a scan with decoys

Switch	Description
<code>-T0</code> through <code>-T2</code>	Serial scans. T0 is slowest
<code>-T3</code> through <code>-T5</code>	Parallel scans. T3 is slowest
<code>-F</code>	Fast mode - Scan fewer ports than the default scan

Notes:

- Nmap runs by default at a T3 level (3 - Normal).
- Nmap runs by default TCP scans.
- Nmap ping the target first before the port scan by default, but if the target have a firewall, maybe the scan will be blocked. **To avoid this, you can use `-Pn` to disable ping.**
- If you're in LAN and you need to disable ARP ping, use:
 - `--disable-arp-ping`
- You can add a input from external lists of hosts/networks:
 - `-iL hosts-example.txt`
- **Fingerprinting** - another word for port sweeping and enumeration

+ More Useful Information about Nmap: +

Switch	Example	Description
<code>-p</code>	<code>nmap 192.168.1.1 -p 21</code>	Port scan for port x
<code>-p</code>	<code>nmap 192.168.1.1 -p 21-100</code>	Port range
<code>-p</code>	<code>nmap 192.168.1.1 -p U:53,T:21-25,80</code>	Port scan multiple TCP and UDP ports
<code>-p-</code>	<code>nmap 192.168.1.1 -p-</code>	Port scan all ports
<code>-p</code>	<code>nmap 192.168.1.1 -p http,https</code>	Port scan from service name
<code>-F</code>	<code>nmap 192.168.1.1 -F</code>	Fast port scan (100 ports)
<code>--top-ports</code>	<code>nmap 192.168.1.1 --top-ports 2000</code>	Port scan the top x ports

Switch	Example	Description
-p-65535	nmap 192.168.1.1 -p-65535	Leaving off initial port in range makes the scan start at port 1
-p0-	nmap 192.168.1.1 -p0-	Leaving off end port in range makes the scan go through to port 65535

2. Service and Version Detection

Switch	Example	Description
-sV	nmap 192.168.1.1 -sV	Attempts to determine the version of the service running on port
-sV --version-intensity	nmap 192.168.1.1 -sV --version-intensity 8	Intensity level 0 to 9. Higher number increases possibility of correctness
-sV --version-light	nmap 192.168.1.1 -sV --version-light	Enable light mode. Lower possibility of correctness. Faster
-sV --version-all	nmap 192.168.1.1 -sV --version-all	Enable intensity level 9. Higher possibility of correctness. Slower
-A	nmap 192.168.1.1 -A	Enables OS detection, version detection, script scanning, and traceroute

3. OS Detection

Switch	Example	Description
-O	nmap 192.168.1.1 -O	Remote OS detection using TCP/IP stack fingerprinting

Switch	Example	Description
-O --osscan-limit	nmap 192.168.1.1 -O --osscan-limit	If at least one open and one closed TCP port are not found it will not try OS detection against host
-O --osscan-guess	nmap 192.168.1.1 -O --osscan-guess	Makes Nmap guess more aggressively
-O --max-os-tries	nmap 192.168.1.1 -O --max-os-tries 1	Set the maximum number x of OS detection tries against a target
-A	nmap 192.168.1.1 -A	Enables OS detection, version detection, script scanning, and traceroute

4. Timing and Performance

Switch	Example input	Description
--host-timeout <time>	1s; 4m; 2h	Give up on target after this long
--min-rtt-timeout/max-rtt-timeout/initial-rtt-timeout <time>	1s; 4m; 2h	Specifies probe round trip time
--min-hostgroup/max-hostgroup <size><size>	50; 1024	Parallel host scan group sizes
--min-parallelism/max-parallelism <numprobes>	10; 1	Probe parallelization
--scan-delay/--max-scan-delay <time>	20ms; 2s; 4m; 5h	Adjust delay between probes

Switch	Example input	Description
<code>--max-retries <tries></code>	3	Specify the maximum number of port scan probe retransmissions
<code>--min-rate <number></code>	100	Send packets no slower than <numberr> per second
<code>--max-rate <number></code>	100	Send packets no faster than <number> per second

5. NSE Scripts

NSE stands for Nmap Scripting Engine, and it's basically a digital library of Nmap scripts that helps to enhance the default Nmap features and report the results in a traditional Nmap output.

One of the best things about NSE is its ability to let users write and share their own scripts, so you're not limited to relying on the Nmap default NSE scripts. [\[+\]](#)

Switch	Example	Description
<code>-sC</code>	<code>nmap 192.168.1.1 -sC</code>	Scan with default NSE scripts. Considered useful for discovery and safe
<code>--script default</code>	<code>nmap 192.168.1.1 --script default</code>	Scan with default NSE scripts. Considered useful for discovery and safe
<code>--script</code>	<code>nmap 192.168.1.1 --script=banner</code>	Scan with a single script. Example banner
<code>--script</code>	<code>nmap 192.168.1.1 --script=http*</code>	Scan with a wildcard. Example http

Switch	Example	Description
--script	nmap 192.168.1.1 --script=http,banner	Scan with two scripts. Example http and banner
--script	nmap 192.168.1.1 --script "not intrusive"	Scan default, but remove intrusive scripts
--script-args	nmap --script snmp-sysdescr --script-args snmpcommunity=admin 192.168.1.1	NSE script with arguments

Useful NSE Script Examples

Command	Description
nmap -Pn --script=http-sitemap-generator scanme.nmap.org	http site map generator
nmap -n -Pn -p 80 --open -sV -vvv --script banner,http-title -iR 1000	Fast search for random web servers
nmap -Pn --script=dns-brute domain.com	Brute forces DNS hostnames guessing subdomains
nmap -n -Pn -vv -O -sV --script smb-enum*,smb-ls,smb-mbenum,smb-os-discovery,smb-s*,smb-vuln*,smbv2* -vv 192.168.1.1	Safe SMB scripts to run
nmap --script whois* domain.com	Whois query
nmap -p80 --script http-unsafe-output-escaping scanme.nmap.org	Detect cross site scripting vulnerabilities
nmap -p80 --script http-sql-injection scanme.nmap.org	Check for SQL injections

- Source: <https://www.stationx.net/nmap-cheat-sheet/>

hping

⚡ Check the hping3 [practical lab](#)

Hping3 is a scriptable program that uses the Tcl language, whereby packets can be received and sent via a binary or string representation describing the packets.

- Another powerful ping sweep and port scanning tool
- Also can craft UDP/TCP packets
- You can make a TCP flood
- hping3 -1 IP address

Switch	Description
-1	Sets ICMP mode
-2	Sets UDP mode
-8	Sets scan mode. Expects port range without -p flag
-9	Listen mode. Expects signature (e.g. HTTP) and interface (-I eth0)
--flood	Sends packets as fast as possible without showing incoming replies
-Q	Collects sequence numbers generated by the host
-p	Sets port number
-F	Sets the FIN flag
-S	Sets the SYN flag
-R	Sets the RST flag
-P	Sets the PSH flag
-A	Sets the ACK flag
-U	Sets the URG flag
-X	Sets the XMAS scan flags

Evasion Concepts

- To evade IDS, sometimes you need to change the way you scan
- One method is to fragment packets (nmap -f switch)

- **OS Fingerprinting**
 - **Active** - sending crafted packets to the target
 - **Passive** - sniffing network traffic for things such as TTL windows, DF flags and ToS fields
- **Spoofing** - can only be used when you don't expect a response back to your machine
- **Source routing** - specifies the path a packet should take on the network; most systems don't allow this anymore
- **IP Address Decoy** - sends packets from your IP as well as multiple other decoys to confuse the IDS/Firewall as to where the attack is really coming from.
 - `nmap -D RND:10 x.x.x.x`
 - `nmap -D decoyIP1,decoyIP2.....,sourceIP,.... [target]`

⚡ Check the IP Address Decoy [practical lab](#) using nmap

- **Proxy** - hides true identity by filtering through another computer. Also can be used for other purposes such as content blocking evasion, etc.
 - **Proxy chains** - chaining multiple proxies together
 - Proxy Switcher
 - Proxy Workbench
 - ProxyChains
- **Tor** - a specific type of proxy that uses multiple hops to a destination; endpoints are peer computers
- **Anonymizers** - hides identity on HTTP traffic (port 80)

Banner Grabbing

Banner grabbing can be used to get information about OS or specific server info (such as web server, mail server, etc.)

- **Active** - sending specially crafted packets and comparing responses to determine OS
- **Passive** - reading error messages, sniffing traffic or looking at page extensions
- Easy way to banner grab is connect via **telnet** on port (e.g. 80 for web server)
- **Netcat** tool
 - "Swiss army knife" of TCP/IP hacking
 - Provides all sorts of control over a remote shell on a target
 - Connects via `nc -e <IP address> <Port>`
 - From attack machine `nc -l -p 5555` opens a listening port on 5555
 - Can connect over TCP or UDP, from any port
 - Offers DNS forwarding, port mapping and forwarding and proxying
 - **Netcat** can be used to banner grab:
 - `nc <IP address or FQDN> <port number>`

- **Example of Banner grabbing on netcat - extracting request HTTP header**
 - i. `nc` command with `target IP` address and `port 80`
 - ii. Issue the `GET / HTTP/1.0` (this GET request will send to the web server).
 - iii. **The server responded with some interesting information:**

```
nc 192.168.63.143 80
```

```
GET / HTTP/1.0
```

```
HTTP/1.1 200 OK
```

```
Date: Sun, 12 Aug 2018 13:36:59 GMT
```

```
Server: Apache/2.2.8 (Ubuntu) DAV/2
```

```
X-Powered-By: PHP/5.2.4-2ubuntu5.10
```

```
Content-Length: 891
```

```
Connection: close
```

```
Content-Type: text/html
```

```
<html><head><title>Metasploitable2 - Linux</title></head><body>
```

```
<pre>
```

```

_ _ _ _ _ | _ _ _ _ _ | _ _ _ _ _ | _ _ _ _ _ |
| ' _ \ / _ \ / _ \ / _ \ | ' _ \ / _ \ / _ \ |
| | | | | _ / | | ( _ \ \ | | ( _ \ | | | | | _ / _ /
| _ | _ | _ \ \ \ \ \ _ _ / _ _ / _ _ \ \ \ \ \ _ _ / _ _ \
| _ |

```

```
Warning: Never expose this VM to an untrusted network!
```

```
Contact: msfdev[at]metasploit.com
```

```
Login with msfadmin/msfadmin to get started
```

```
</pre>
```

```
<ul>
```

```
<li><a href="/twiki/">TWiki</a></li>
```

Vulnerabilities

Vulnerability Categories:

- **Misconfiguration** - improperly configuring a service or application
- **Default installation** - failure to change settings in an application that come by default
- **Buffer overflow** - code execution flaw
- **Missing patches** - systems that have not been patched

- **Design flaws** - flaws inherent to system design such as encryption and data validation
- **Operating System Flaws** - flaws specific to each OS
- **Default passwords** - leaving default passwords that come with system/application

Vulnerability Assessment - Scans and tests for vulnerabilities but does not intentionally exploit them.

- Find the vulnerabilities so we can categorize them (OS, Misconfigurations, patch management, third-party, etc)

Vulnerability Management Life-cycle

*The Vulnerability Management Life Cycle is intended to allow organizations to **identify system security weaknesses; prioritize assets; assess, report, and remediate the weaknesses; and verify that they have been eliminated.***



1. **Discover:** Inventory all assets across the network and identify host details including operating system and open services to identify vulnerabilities. Develop a network baseline. Identify security vulnerabilities on a regular automated schedule.
2. **Prioritize Assets:** Categorize assets into groups or business units, and assign a business value to asset groups based on their criticality to your business operation.
3. **Assess:** Determine a baseline risk profile so you can eliminate risks based on asset criticality, vulnerability threat, and asset classification.
4. **Report:** Measure the level of business risk associated with your assets according to your security policies. Document a security plan, monitor suspicious activity, and describe known vulnerabilities.
5. **Remediate:** Prioritize and fix vulnerabilities in order according to business risk. Establish controls and demonstrate progress.
6. **Verify:** Verify that threats have been eliminated through follow-up audits.

Vulnerability Scanning

Can be complex or simple tools run against a target to determine vulnerabilities.

- **Types of Vuln. Assessment tools:**
 - Host-based
 - Depth-based (Fuzzer tools)
 - Application-layer tools (software, databases, etc)
 - Active scanning

- Passive scanning
- Scope tools
- Tools:
 - Industry standard is [Tenable's Nessus](#).
 - [GFI LanGuard](#).
 - [Nikto](#) - CLI; is a **web server assessment tool**. It is designed to find various default and insecure files, configurations and programs on any type of web server.
 - [OpenVAS](#) - Best competitor to Nessus and is free.
 - [wpscan](#) - CLI; Scan WordPress websites.
 - **MBSA - Microsoft Baseline Security Analyzer**.
 - **FreeScan** - Well known for testing websites and applications.
 - Qualys

CVSS and CVE

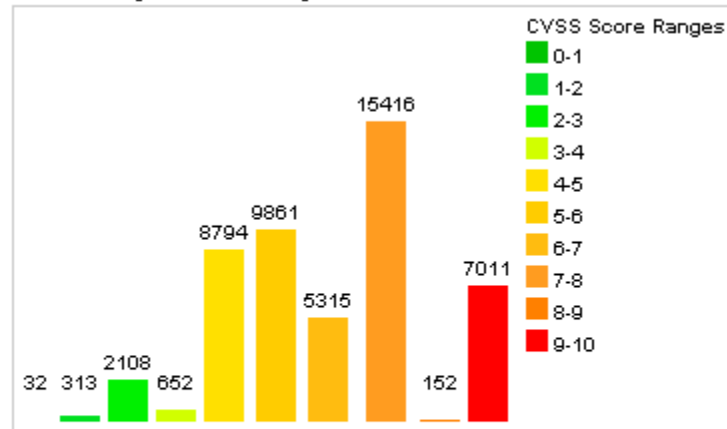
- **CVSS - Common Vulnerability Scoring System** [\[+\]](#)

- Places numerical score based on severity

Distribution of all vulnerabilities by CVSS Scores

CVSS Score	Number Of Vulnerabilities	Percentage
0-1	32	0.10
1-2	313	0.60
2-3	2108	4.20
3-4	652	1.30
4-5	8794	17.70
5-6	9861	19.90
6-7	5315	10.70
7-8	15416	31.00
8-9	152	0.30
9-10	7011	14.10
Total	49654	

Vulnerability Distribution By CVSS Scores



- **Weighted Average CVSS Score: 6.9**
 - None - white (0.0)
 - Low - green tones (0.1 - 3.9)
 - Medium - yellow/light orange (4.0 - 4.9)
 - High - orange (7.0 - 8.0)
 - Critical - red (9.0 - 10.0)

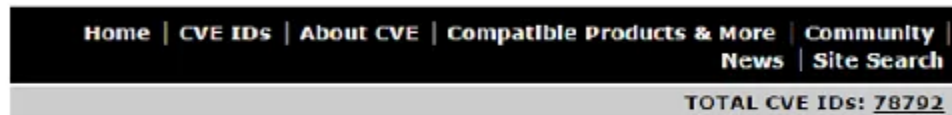
- **CVE – Common Vulnerabilities and Exposures** [\[+\]](#)

- Is a list of publicly disclosed vulnerabilities and exposures that is maintained by MITRE.



Common Vulnerabilities and Exposures

The Standard for Information Security Vulnerability Names



HOME > CVE LIST

Section Menu

CVE IDs

Coverage Goals
Reference Key/Maps
Updates & Feeds

CVE List (all existing CVE IDs)

Downloads
Search CVE List
Search Tips
View Entire CVE List (html)
NVD Advanced CVE Search
CVE ID Scoring Calculator

Request a CVE ID

CVE Numbering Authorities

CVE IDs

The [CVE List Master Copy](#) is hosted on this CVE website. The [U.S. National Vulnerability Database \(NVD\)](#), which is built upon and fed by the CVE List, provides enhanced information about CVE IDs. Learn more about the [CVE and NVD relationship](#).

What would you like to do?

Data Feeds

[Available via](#)
[Purdue](#)
University & NVD

Request a CVE ID number

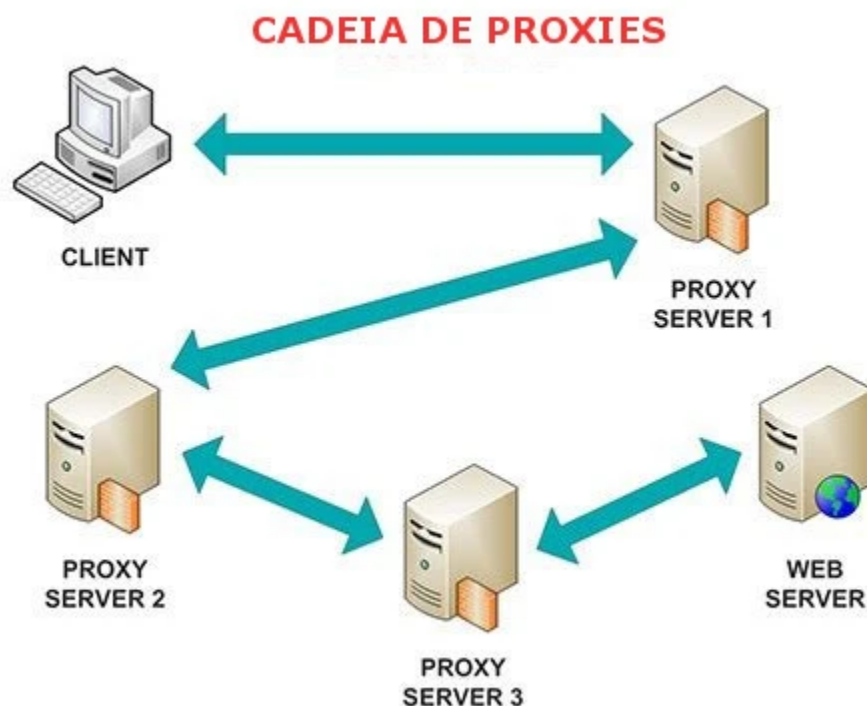
[Click for](#)
[guidelines &](#)
[more](#)

○

• NVD - National Vulnerability Database [+]

- is a database, maintained by NIST, that is fully synchronized with the MITRE CVE list; US Gov. vulnerabilities repository.

ProxyChains



ProxyChains is open-source software that is available free and most of Linux distro it is pre-installed. If you are using the latest version of Kali Linux it is pre-installed in it.

ProxyChains is a tool that redirects the TCP (Transmission Control Protocol) connection with the help of proxies like TOR, HTTP(S), and SOCKS, and it creates a proxy chain server.

ProxyChains Features:

- Support **SOCKS5**, **SOCKS4**, and **HTTP/HTTPS CONNECT** proxy servers.
- Proxychains can be mixed up with a different proxy types in a list
- Proxychains also supports any kinds of chaining option methods, like: random, which takes a random proxy in the list stored in a configuration file, or chaining proxies in the exact order list, different proxies are separated by a new line in a file. There is also a dynamic option, that lets Proxychains go through the live only proxies, it will exclude the dead or unreachable proxies, the dynamic option often called smart option.
- Proxychains can be used with servers, like squid, sendmail, etc.
- Proxychains is capable to do DNS resolving through proxy.
- Proxychains can handle any TCP client application, ie., nmap, telnet.

Enumeration Concepts

Enumeration is the process of extracting **user names**, **machine names**, **network resources**, **shares**, **and services** from a system, and its conducted in an intranet environment.

- Get user names using email IDs
- Get information using default passwords
- Get user names using SNMP
- Brute force AD
- Get user groups from Windows
- Get information using DNS zone transfers
- NetBios, LDAP, NTP, DNS

In this phase, the attacker creates an active connection to the system and performs directed queries to gain more information about the target. The gathered information is used to identify the vulnerabilities or weak points in system security and tries to exploit in the System gaining phase.

- Defined as listing the items that are found within a specific target
- Always is active in nature
- Direct access
- Gain more information

SNMP Enumeration

⚡ Check the SNMP Enumeration [practical lab](#)

SNMP enumeration is the process of enumerating the users accounts and devices on a SNMP enabled computer.

- SNMP service comes with two passwords, which are used to configure and access the SNMP agent from the management station (MIB):
 - i. **Read community string**
 - ii. **Read/Write community string**
- These strings (`passwords`) come with a **default value**, which is same for all the systems.
- **They become easy entry points for attackers if left unchanged by administrator.**

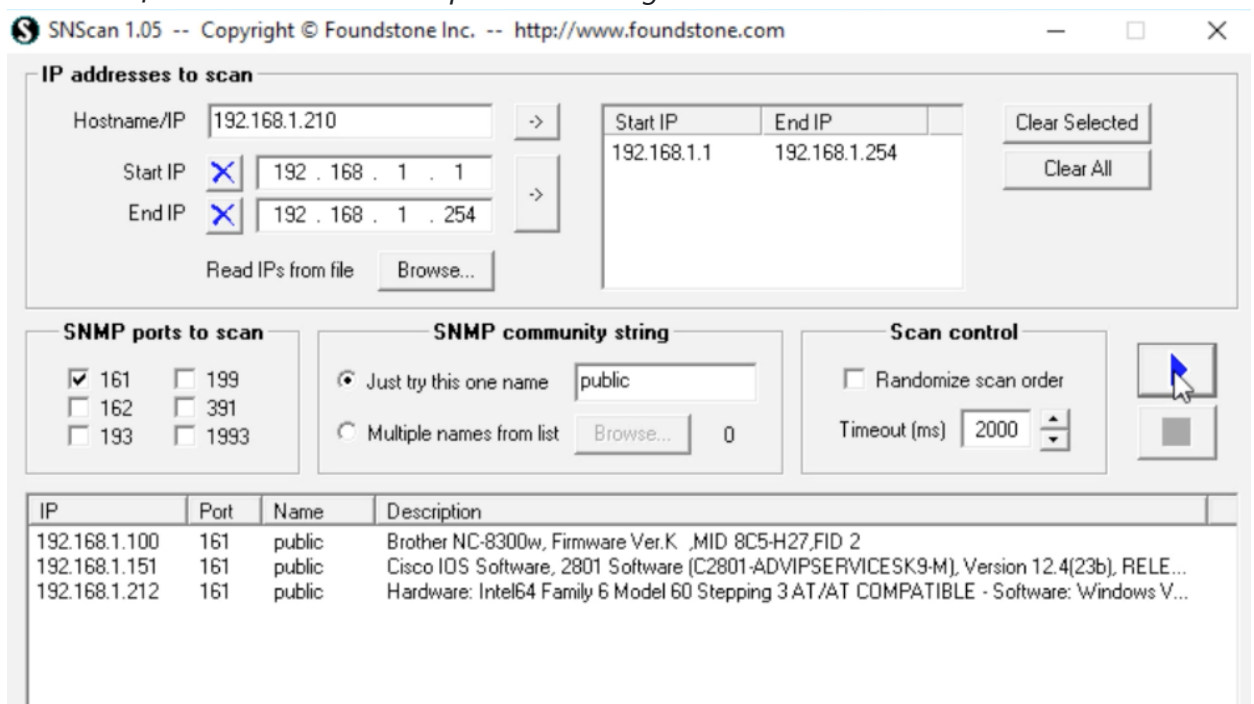
Attackers enumerate SNMP to extract information about network resources such as hosts, routers, devices, shares(...) Network information such as ARP tables, routing tables, device specific information and traffic statistics.

- **Runs on Port 161 UDP**
- **Management Information Base (MIB)** - database that stores information
- **Object Identifiers (OID)** - identifiers for information stored in MIB
- **SNMP GET** - gets information about the system
- **SNMP SET** - sets information about the system
- **Types of objects**
 - **Scalar** - single object
 - **Tabular** - multiple related objects that can be grouped together
- SNMP uses community strings which function as passwords
- There is a read-only and a read-write version
- Default read-only string is **public** and default read-write is **private**
- These are sent in cleartext unless using SNMP v3
- **CLI Tools**
 - `snmp-check` --> SNMP device enumerator comes pre-installed on Kali Linux machine; `snmp-check` supports a huge type of enumerations:
 - contact and user accounts
 - devices
 - domain
 - hardware and storage informations

- hostname
- IIS statistics
- listening UDP ports and TCP connections
- motd (banner)
- network interfaces and network services
- routing information
- etc
- Metasploit module `snmp_enum`
 - ⚡ MSF `snmp_enum` practical lab
- snmpwalk
- GUI Tools
 - Engineer's Toolset
 - SNMPScanner
 - OpUtils 5
 - SNScan

Example of SNScan:

- *Note: the first scanned item is a printer running SNMP.*



Windows System Basics

- Everything runs within context of an account
- **Security Context** - user identity and authentication information
- **Security Identifier (SID)** - identifies a user, group or computer account
- **Resource Identifier (RID)** - portion of the SID identifying a specific user, group or computer
- The end of the SID indicates the user number
 - Example SID: S-1-5-21-3874928736-367528774-1298337465-**500**
 - **Administrator Account** - SID of 500
 - Command to get SID of local user:
 - `wmic useraccount where name='username' get sid`
 - **Regular Accounts** - start with a SID of 1000
 - **Linux Systems** used user IDs (UID) and group IDs (GID). Found in /etc/passwd
- **SAM Database** - file where all local passwords are stored (encrypted)
 - Stored in C:\Windows\System32\Config
- **Linux Enumeration Commands in PowerShell or CmdPrompt**
 - `finger` - info on user and host machine
 - `rpcinfo` and `rpcclient` - info on RPC in the environment
 - `showmount` - displays all shared directories on the machine
- **Look for share resources (NetBIOS):**
 - `net view \\sysName`
- **Windows SysInternals** is a website and suite that offers technical resources and utilities to manage, diagnose, troubleshoot, and monitor.
 - <https://docs.microsoft.com/en-us/sysinternals/downloads/>
 - Lots of resources for enumerating, windows administration tools, etc.

NetBIOS Enumeration

- NetBIOS provides name servicing, connectionless communication and some Session layer stuff
- The browser service in Windows designed to host information about all machines within domain or TCP/IP network segment
- NetBIOS name is a **16-character ASCII string** used to identify devices

Enumerating NetBIOS:

- You can use `nmap` or `zenmap` to check which OS the target is using, and which ports are open:
 - `nmap -O <target>`

- If there's any **UDP port 137** or **TCP port 138/139** open, we can assume that the target is running some type of NetBIOS service.

- On Windows is `nbtstat` command:

`nbtstat` displays protocol statistics and current TCP/IP connections using NetBIOS over TCP/IP.

- `nbtstat` gives your own info
- `nbtstat -a` list the remote machine's name table given its **name**
- `nbtstat -A` - list the remote machine's name table given its **IP address**
- `nbtstat -n` gives local table
- `nbtstat -c` gives cache information

```

C:\>nbtstat -A 172.16.212.133

Local Area Connection 2:
Node IpAddress: [172.16.212.128] Scope Id: []

    NetBIOS Remote Machine Name Table

    Name                Type             Status
    -----
    METASPLOITABLE <00>    UNIQUE          Registered
    METASPLOITABLE <03>    UNIQUE          Registered
    METASPLOITABLE <20>    UNIQUE          Registered
    _MSBROWSE_ <01>    GROUP           Registered
    WORKGROUP <00>    GROUP           Registered
    WORKGROUP <1D>    UNIQUE          Registered
    WORKGROUP <1E>    GROUP           Registered

    MAC Address = 00-00-00-00-00-00

C:\>_

```

Code	Type	Meaning
<1B>	UNIQUE	Domain master browser
<1C>	UNIQUE	Domain controller
<1D>	GROUP	Master browser for subnet
<00>	UNIQUE	Hostname
<00>	GROUP	Domain name
<03>	UNIQUE	Service running on system
<20>	UNIQUE	Server service running

- NetBIOS name resolution doesn't work on IPv6
- Other Tools for NetBIOS enumeration:
 - SuperScan

- Hyena
- NetBIOS Enumerator (is a nbtstat with GUI)
- NSAuditor

Linux System Basics

- **Enum4linux** is a tool for enumerating information from Windows and Samba systems:

- `enum4linux -u CEH -p Pa55w0rd -U 10.0.2.23`
 - `-u` Username, `-p` Password, `-U` users information
- ⚡ [enum4linux practical lab](#)
- Key features:
 - RID cycling (*When RestrictAnonymous is set to 1 on Windows 2000*)
 - User listing (*When RestrictAnonymous is set to 0 on Windows 2000*)
 - Listing of group membership information
 - Share enumeration
 - Detecting if host is in a workgroup or a domain
 - Identifying the remote operating system
 - Password policy retrieval (using polenum)

- **finger** --> who is currently logged in, when and where.

```

Login      Name      Tty      Idle   Login Time   Office      Office Phone
kali       Kali      tty7     10:09   Sep 1 14:14 (:0)

```

- **w** --> Show who is logged on and what they are doing.

```

00:27:15 up 9:32, 1 user, load average: 0.06, 0.09, 0.09
USER      TTY      FROM      LOGIN@   IDLE   JCPU   PCPU WHAT
kali      tty7     :0         14:16    10:11m 30.26s 2.09s xfce4-session

```



Linux architecture and commands will be cover later on next module.

LDAP Enumeration

- Runs on TCP ports 389 and 636 (over SSL)
- Connects on 389 to a Directory System Agent (DSA)
- Returns information such as valid user names, domain information, addresses, telephone numbers, system data, organization structure and other items
- To identify if the target system is using LDAP services you can use **nmap** with `-sT` flag for TCP connect/Full scan and `-O` flag for OS detection.

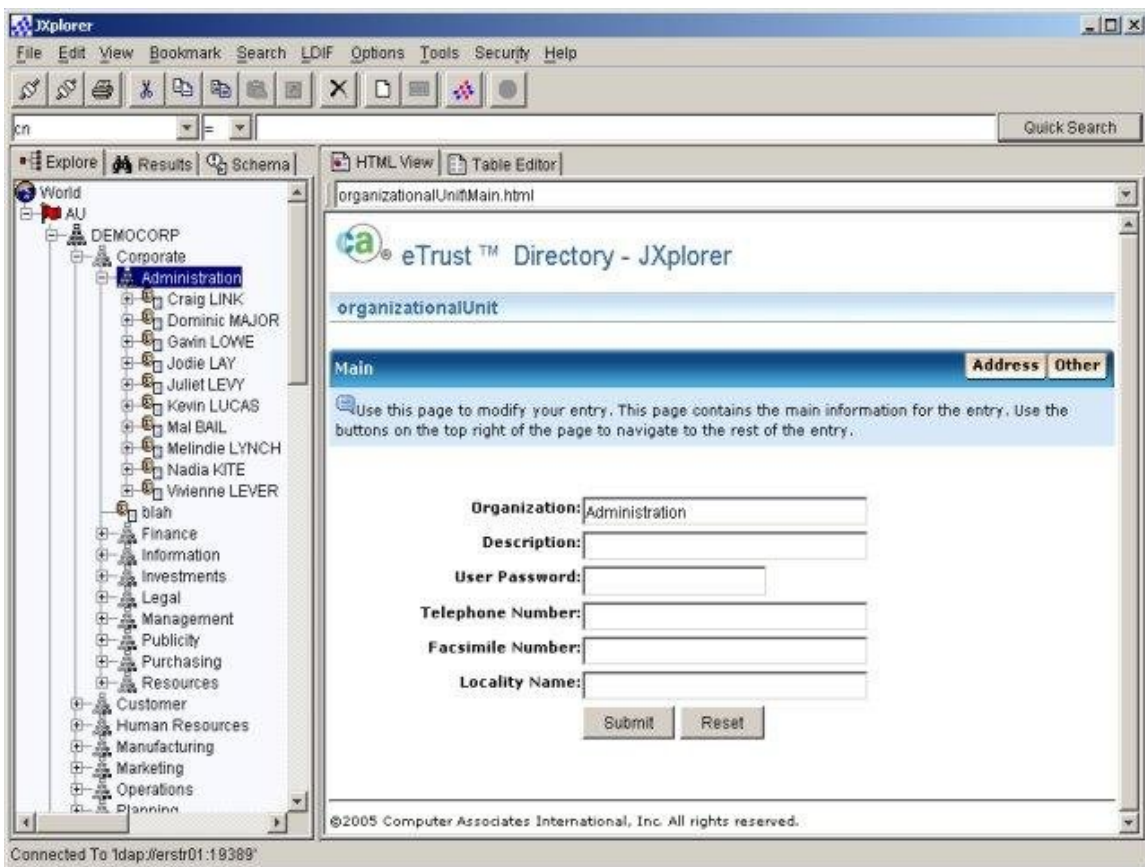
```
sudo nmap -sT -O <target IP address>
```

```
PORT      STATE SERVICE
53/tcp    open  domain
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap <-----
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl <-----
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
49154/tcp open  unknown
49155/tcp open  unknown
49157/tcp open  unknown
49158/tcp open  unknown
49159/tcp open  unknown
MAC Address: 00:00:11:33:77:44
Running: Microsoft Windows 2012
OS CPE: cpe:/o:microsoft:windows_server_2012:r2
OS details: Microsoft Windows Server 2012 or Windows Server 2012 R2
Network Distance: 1 hop
```

- Tools for Enumeration LDAP:

- Softerra
- JXplorer
- Lex
- LDAP Admin Tool

- JXplorer example:



NTP Enumeration

- Runs on UDP 123
- Querying can give you list of systems connected to the server (name and IP)
- Tools
 - NTP Server Scanner
 - AtomSync
 - Can also use Nmap and Wireshark
- Commands include `ntpttrace` , `ntpdate` , `ntpdcc` and `ntpq`

Nmap example for NTP enumeration:

- `-sU` UDP scan
- `-pU` port UDP 123 (NTP)
- `-Pn` Treat all hosts as online -- skip host discovery
- `-n` Never do DNS resolution
- The [nmap script](#) `ntp-monlist` will run against the ntp service which only runs on UDP 123

```
nmap -sU -pU:123 -Pn -n --script=ntp-monlist <target>
```

```
PORT      STATE SERVICE REASON
123/udp   open  ntp     udp-response
```

```

| ntp-monlist:
|   Target is synchronised with 127.127.38.0 (reference clock)
|   Alternative Target Interfaces:
|     10.17.4.20
|   Private Servers (0)
|   Public Servers (0)
|   Private Peers (0)
|   Public Peers (0)
|   Private Clients (2)
|     10.20.8.69      169.254.138.63
|   Public Clients (597)
|     4.79.17.248      68.70.72.194      74.247.37.194      99.190.119.152
|     ...
|     12.10.160.20      68.80.36.133      75.1.39.42      108.7.58.118
|     68.56.205.98
|     2001:1400:0:0:0:0:1 2001:16d8:dd00:38:0:0:0:2
|     2002:db5a:bccd:1:21d:e0ff:feb7:b96f 2002:b6ef:81c4:0:0:1145:59c5:3682
|   Other Associations (1)
|_    127.0.0.1 seen 1949869 times. last tx was unicast v2 mode 7

```

- As you can see on the output above, information of all clients that is using NTP services on the network shown IPv4 and IPv6 addresses.

SMTP Enumeration

- **Ports used:**
 - **SMTP: TCP 25** --> [outbound email]
 - **IMAP: TCP 143 / 993**(over SSL) --> [inbound email]
 - **POP3: TCP 110 / 995**(over SSL) --> [inbound email]
- In simple words: users typically use a program that uses SMTP for sending e-mail and either POP3 or IMAP for receiving e-mail.
- **Enumerating with nmap:**
- `-p25` port 25 (SMTP)
- `--script smtp-commands` nmap script - attempts to use EHLO and HELP to gather the Extended commands supported by an SMTP server.

```
nmap -p25 --script smtp-commands <target IP>
```

```

PORT      STATE SERVICE
25/tcp    open  smtp
| smtp-commands: WIN-J83C1DR5CV1.ceh.global Hello [10.10.10.10], TURN, SIZE 2097152, ETRN,
PIPELINING, DSN, ENHANCEDSTATUSCODES, 8bitmime, BINARYMIME, CHUNKING, VRFY, OK,
|_ This server supports the following commands: HELO EHLO STARTTLS RCPT DATA RSET MAIL QUIT

```

Nmap done: 1 IP address (1 host up) scanned in 0.86 seconds

- It is possible to connect to SMTP through **Telnet connection**, instead using port 23(Telnet) we can set the port 25(SMTP) on the telnet command:

- `telnet <target> 25`

- Case we got connected, we can use the **SMTP commands** to explore as shown below:

```
root@kali:~# telnet smtp.cox.net 25
Trying 68.6.19.8...
Connected to smtp.cox.net.
Escape character is '^]'.
220 fed1rmimp0209.cox.net cox ESMTP server ready
HELO
501 HELO requires valid address
HELO OurTest.com
250 fed1rmimp0209.cox.net hello [70.69.1.69], pleased to meet you
MAIL FROM:bob@cox.net
250 2.1.0 <bob@cox.net> sender ok
RCPT TO:john@cox.net
250 2.1.5 <john@cox.net> recipient ok
```

- Both of emails are valid to an attacker explore further attacks like brute forcing etc.

Some SMTP Commands:

Command	Description
HELO	It's the first SMTP command: it starts the conversation identifying the sender server and is generally followed by its domain name.
EHLO	An alternative command to start the conversation, underlying that the server is using the Extended SMTP protocol.
MAIL FROM	With this SMTP command the operations begin: the sender states the source email address in the "From" field and actually starts the email transfer.
RCPT TO	It identifies the recipient of the email
DATA	With the DATA command the email content begins to be transferred; it's generally followed by a 354 reply code given by the server, giving the permission to start the actual transmission.
VRFY	The server is asked to verify whether a particular email address or username actually exists.
EXPN	asks for a confirmation about the identification of a mailing list.

Other tools:

- smtp-user-enum
 - Username guessing tool primarily for use against the default Solaris SMTP service. Can use either EXPN, VRFY or RCPT TO.