**Module 3: Scanning Networks**

**OSI Model**

| **Layer** | **Name** | **Example protocols** |
| --- | --- | --- |
| 7 | Application layer | HTTP, SNMP |
| 6 | Presentation layer | MIME, ASCII |
| 5 | Session layer | SOCKS, NetBIOS |
| 4 | Transport layer | TCP, UDP |
| 3 | Network layer | IP, ICMP |
| 2 | Data link layer | MAC, ARP |
| 1 | Physical layer | ethernet, Wi-Fi |

**TCP/IP Model**

| **Layer** | **Name** | **Example protocols** |
| --- | --- | --- |
| 4 | Application layer | HTTP, SNMP |
| 3 | Transport layer | TCP, UDP |
| 2 | Internet layer | IP, ICMP |
| 1 | Link layer | ARP, MAC |

**TCP Flags**

* **SYN:** Initiates a connection between two hosts to facilitate communication
* **ACK:** Acknowledge the receipt of a packet
* **URG:** Indicates that the data contained in the packet is urgent and should process it immediately
* **PSH:** Instructs the sending system to send all buffered data immediately
* **FIN:** Tells te remote system about the end of the communication. In essence, this gracefully closes the connection
* **RST:** Reset a connection

**TCP Session Establishment**

* **Three-way Handshake**
* **Step1**: Bill to Sheela: SYN
* **Step2:** Sheela to Bill: SYN+ACK
* **Step3:** Bill to Sheela: ACK

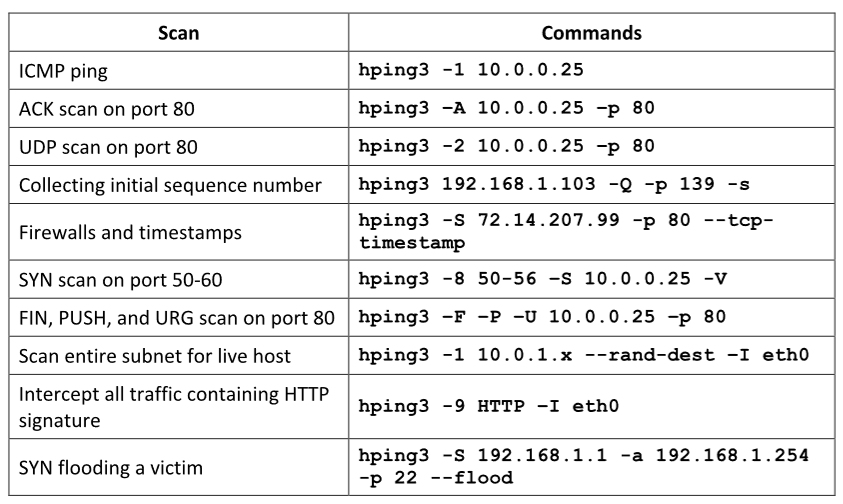
**TCP Session Termination**

* Bill is the **client**, Sheela is the **server**
* Bill to Sheela: FIN
* Sheela to Bill: ACK
* Bill to Sheela: FIN
* Sheela to Bill: ACK

**Scanning Tools**

* **Nmap:** Inventory a network. Extract info such as live hosts, open ports, services, types of packet filters/firewalls, OS and its version…
* **Hping2/Hping3:** Command line network scanning and packet crafting tool for the TCP/IP protocol. Can be used for network security auditing, firewall testing, manual path MTU discovery, remote OS fingerprinting…
* **Metasploit:** An open-source project that provides the infrastructure, content, and tools to perform penetration tests and extensive security auditing.
* **NetScanTools Pro:** Assist attackers in automatically or manually listing IPv4/v6 addresses, hostnames, domain names, and URLs

**Hping command**



**Host Discovery Techniques**

**ARP Ping Scan:**

* ARP request proble → ARP response
* NMAP: **-sn -PR** (-sn disables port scan)
* Efficient and accurate than other host discovery techniques
* Automatically handles ARP requests, retransmission, and timeout as its own direction.
* Useful for system discovery, where we may need to scan large address spaces
* Display response time or latency

**UDP Ping Scan:**

* UDP ping → UDP response
* NMAP: **-sn -PU**
* Behind firewalls with strict TCP filtering, leaving the UDP traffic forgotten

**ICMP Ping Scan**

* **ICMP ECHO Ping:** 
  + Send ICMP ECHO requests to a host.
  + Useful for locating active devices or determining if the ICMP is passing through a firewall.
  + NMAP: **-sn -PE**
* **ICMP ECHO Ping Sweep:** 
  + Determine the live hosts from a range of IP address
  + NMAP: **-sn -PE <IP Range>**
  + Tool: Angry IP Scanner
  + Countmeasures: Configure firewalls, IDS/IPS, Evaluate type of ICMP traffic…
* **ICMP Timestamp Ping:** 
  + If the administrators block ICMP ECHO pings.
  + NMAP: **-sn -PP**
* **ICMP Address Mask Ping:** 
  + If the administrators block ICMP ECHO pings.
  + NMAP: **-sn -PM**

**TCP Ping Scan**

* **TCP SYN Ping:** 
  + Send empty TCP SYN packets, an **ACK response** means active host
  + NMAP: **-sn -PS**
* **TCP ACK Ping:** 
  + Send empty TCP ACK packets, an **RST reponse** means active host
  + NMAP: **-sn -PA**

**IP Protocol Scan:**

* Send various probe packets using different IP protocols, **any response** means active host
* NMAP: **-sn -PO**

**Common Ports and Services**

* **ftp-data:** 20/tcp, data transfer
* **ftp:** 21/tcp, ftp command
* **ssh:** 22/tcp
* **telnet:** 23/tcp
* **smtp:** 25/tcp
* **domain:** 53/dual
* **sql\*net:** 66/dual
* **tftp:** 69/dual, Trivial File Transfer
* **www-http:** 80/dual
* **kerberos:** 88/dual
* **pop3:** 110/tcp
* **nntp:** 119/dual, Usenet Network News Transfer
* **ntp:** 123/tcp, Network Time Protocol
* **netbios-ns:** 137/dual, Netbios Name Service
* **netbios-dgm:** 138/dual, Netbios Datagram Service
* **netbios-ssn:** 139/dual, Netbios Session Service
* **snmp:** 161/dual
* **snmp-trap:** 162/dual

**Port Scanning Techniques**

**TCP Scanning**

**1: Open TCP Scanning Methods**

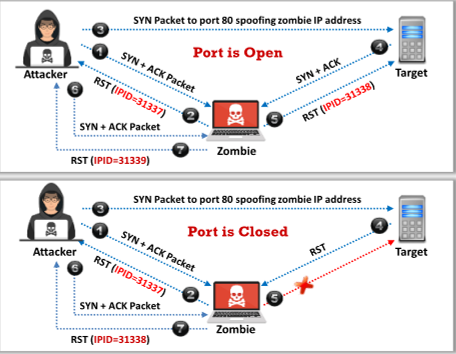
* **1.1: TCP Connect/Full Open Scan:**
  + Detect when a port is open after completing three-way handshake
  + Establish a full connection and then close it by sending RST packet
  + Do not require superuser privileges
  + **Easily detectable and filterable**
  + OPEN: **three-way handshake and end it with RST packet**
  + CLOSED: **get a RST reponse**
  + NMAP: **-sT**

**2: Stealth TCP Scanning Methods**

* **2.1: Half-open Scan:**
  + Abruptly reset the TCP connection before the three-way handshake
  + Bypass firewall rules and logging mechanisms, hide themselves
  + OPEN: **two-way handshake, end it with RST packet**
  + CLOSED: get a **RST response**
  + NMAP: **-sS**
* **2.2: Inverse TCP Flag Scan**
  + Send TCP probe packets with a **TCP flag (FIN, URG, PSH)** set or with no flags
  + **No reponse** implies open port, **RST response** means closed port
  + Avoid many IDS and logging system, highly stealthy
  + Pros: **Requiring super-user privileges**
  + Cons: **Not effective against Windows hosts**
    - **2.2.1: XMAS Scan**
      * Send a TCP frame to a remote device with **FIN, URG, and PUSH** flags set
      * Will not work against any current version of MS Windows.
      * Pros: Avoid IDS and TCP three-way handshake
      * Cons: Works on the UNIX platform only
      * NMAP: **-sX**
    - **2.2.2: FIN Scan**
      * A Fin probe with the FIN TCP flag set
      * FIN scanning works only with OS uses an RFC 793-based TCP/IP implementation
      * NMAP: **-sF**
    - **2.2.3: NULL Scan**
      * A NULL probe with no TCP flags set
      * NMAP: **-sN**
    - **2.2.4: Maimon Scan**
      * Similar to NULL, FIN, and Xmas scan, but the probe used here is FIN/ACK
      * NMAP: **-sM**
* **2.3: ACK Flag Probe Scan**
  + Send TCP probe packets set with an ACK flag, and then analyze the header information (TTL and Window field) of received RST packets.
  + Can be used to check the filtering system of a target
  + NMAP: **-sA**
  + Filtered (stateful firewall is present): **No response**
  + Not filtered: **RST response**
  + Pros: Evade most IDS
  + Cons: Slow and can exploit only older OS with vulnerable BSD-derived TCP/IP stracks
    - **2.3.1: TTL-Based Scan**
      * NMAP: **-ttl [time] [target]**
      * Open: **Less than the boundary value**
    - **2.3.2: Window Scan**
      * Open: **TCP RST with non-zero window field**
      * Closed: **TCP RST zero window field**
      * NMAP: **-sW**

**3: Third Party and Spoofed TCP Scanning Methods**

**3.1: IDLE/IP ID Header Scan**

* Every IP packet on the Internet has a fragment identification number (IPID); an OS increases the IPID for each packet send, thus, probing an IPID gives an attacker the number of packets send after the past probe.
* 
* NMAP: **-sI**

**UDP Scanning**

**1: UDP Scanning**

* Open: **No response**
* Closed: **ICMP port unreachable message**
* NMAP: **-sU**
* Pros: Very efficiently on Windows devices
* Cons: Provide port info only.

**SCTP Scanning**

**1: SCTP INIT Scanning**

* SCTP: **Stream Control Transport Protocol,** a **reliable message-oriented transport layer protocol**, it is used as an alternative to the TCP/UDP protocol, its characteristics are similar to those of TCP and UDP
* **For-way handshake:**
* Step1: Cliend to Server: **INIT**
* Step2: Server to Client: **INIT-ACK**
* Step3: Client to Server: **COOKIE-ECHO**
* Step4: Server to Client: **COOKIE-ACK**
* Open: Attackers send an **INIT chunk** to the target host, and an **INIT+ACK chunk response** implies open port
* Closed: **ABORT Chunk** response
* Filter: **ICMP unreachable exception**
* NMAP: **-sY**

**2: SCTP COOKIE/ECHO Scanning**

* Open: Send a **COOKIE ECHO chunk** to the target host, **no response** implies open port
* Closed: **ABORT Chunk**
* Not blocked by non-stateful firewall rulesets
* Only a good IDS will be able to detect SCTP COOKIE ECHO chunk
* NMAP: **-sZ**
* Pros: Not as conspicuous as the INIT scan
* Cons: Cannot differentiate clearly between open and filtered ports, show open/filtered in both cases.

**SSDP Scanning**

**1: SSDP and List Scanning**

SSDP: **Simple Service Discovery Protocol**, a network protocol that **works in conjunction with the UPnP to detect plug and play devices**

* Vulnerabilities in UpnP may alllow attackers to launch **Buffer overflow or DoS attacks**
* Attacker may use the **UpnP SSDP M-SEARCH** info discovery tool to check if the machine is vulnerable to UPnP exploits or not.

List Scanning: Generate and print a **list of IPs/Names** without actually pinging them. **A reverse DNS resolution** is performed to identify the host names

* NMAP (List scanning): **-sL**
* Pros: Perform a good sanity check. Detect incorrectly defined IP addresses in the cmd line or in an option file.

**Ipv6 Scanning**

**1: Ipv6 Scanning**

* Harvest Ipv6 addresses from network traffic, recorded logs, or Received from: header lines in archived emails.
* NMAP: **-6**

**Service Version Discovery**

* Help attackers to obtain info about running service and their versions on a target system
* Determine the vulnerability or target system to particular exploits
* NMAP: **-sV**

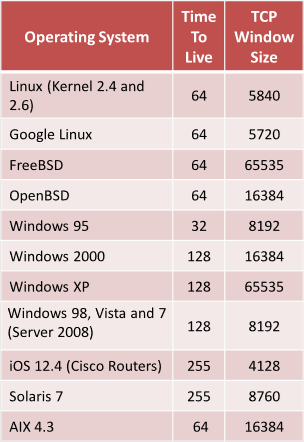
**Counter Port Scanning**

* Configure firewall and IDS rules
* Filter all ICMP messages at the firewalls and routers
* Check network configuration, firewall configuration by scanning org’s hosts
* Anti-scanning and anti-spoofing

**OS Discovery**

* **Active banner grabbing:** Specially crafted packets are send to remote OS and the responses are noted. Compare the responses with a database. Responses vary due to different TCP/IP implementation.
* **Passive banner grabbing:** From error messages, sniffing the network traffic, from page extensions, such as **.aspx→ IIS server and Windows platform**

**Identify Target System OS**

* Look at the **TTL** and **TCP window size** in the IP header of the first packet in a TCP session
* Use packet-sniffing tools such as Wireshark and observe the TTL and windows size fields.
* 
* NMAP: **-O**
* Unicornscan: **unicornscan <target ip address>**
* NMAP Script Engine: **nmap –script smb-os-discovery.nse <target IP >**
* IPv6 use several **additional advanced probes specific to IPv6** along with **a separate OS detection engine that is specialized for IPv6**
* IPv6 Fingerprinting: **nmap -6 -O <target IP>**

**Banner Grabbing Countermeasures**

* Display **false banners** to mislead attackers
* Turn off unnecessary services
* Use **ServerMask** to disable or change banner info
* Hide file extensions to mask web technologies

**IDS/Firewall Evasion Techniques**

* **Packet Fragmentation:**

Splitting of a probe packet into several smaller packets

Not a new method but a modification of the precvious techniques

NMAP: **-f**

* **Source Routing:**

Send a packet to the intended destination with a partially or competely specified route (without firewall/IDS-configured routers)

* **Source Port Manipulation:**

Manipulate actual port numbers with common port numbers

It occurs when a firewall is configured to allow packets from well-known ports such as HTTP, DNS, FTP…

NMAP: **-g** or **–source-port**

* **IP Address Decoy:**

Generate or manually specify the IP address of decoys

The technique makes it difficult for the IDS/firewall to determine which IP address was actually scanning the network and…

NMAP: **-D RND:10** or **-D decoy1, decoy2, decoy3…**

* **IP Address Spoofing:**

Change the source IP address

The reply return to the spoofed address rather than the attacker’s

The attacker modify the address info in the IP packet header and the source address bits field

Hping3: **Hping3 <targe addresst> -a <spoofed address>**

**Detections of IP Spoofing:**

**Direct TTL Probes**: Send a packet to the host of a suspented spoofed packet, compare the TTL, **successful when the attacker is in a different subnet from that of the victim**

**IP Identifacation Number:** Send a probe, compare IPIDs. **Reliabled even if the attacker is in the same subnet.**

**TCP Flow Control Method:** Attackers will not receive SYN-ACK packets from the target, therefore attackers cannot respond to a change in the congestion window size. When received traffic continues after a window size is exhausted, the packets are most likely spoofed.

Window size field represents the maximum amount of data that the recipient can receive and the maximum amount of data that the sender can transmit without ack. The sender should stop sending data whenever the window size is set to 0.

**IP Spoofing Countermeasure**

Encryption all the network traffic such as IPsec, TLS, SSH, HTTPS

Use multiple firewalls

Do not reply on IP-based authentication

Use a random **ISN (initial sequence number)**

Ingree Filtering

Egress Filtering

* **Creating Custom Packet:**

Create custom TCP packets using various packet crafting tools like Colasoft Packet Builder, NetScanTools Pro

NMAP: **nmap <target address> --data 0xdeadbeef** (Append Custom Binary Data)

**namp <target address> --data-string “Ph34r my l33t skills”** (Regular string as payloads)

**nmap <target address> --data-string 5** (Append Random Data)

* **Randomizing Host Order:**

Scan the number of hosts in random order

NMAP: **nmap –randomize-hosts <target host>**

* **Send Bad Checksum:**

Send packets with bad or bogus TCP/UDP checksums

NAMP: **nmap –badsum <target>**

* **Proxy Server:**

An application serve as an intermediary for connecting..

hide the actual source of a scan, evade certain IDS/firewall restrictions

Mask the actual source of an attack by impersonating the fake source address of the proxy

Remotely access intranets and other website resources that restrcited

Interrupt all requests sent by a user and tranmit them to a third destination such that victims can only idenfity the proxy server address

Chain multiple proxy servers to avoid detection

**Proxy Chaining:**

**Step1**: User requests a resource from the dest

**Step2**: Proxy client at the user’s system connects to a proxy server and passes the request to proxy server

**Step3**: The proxy server strips the user’s id info and passes the request to next proxy server

**Step4:** The process is repeated by all the proxy server in the chain

**Step5**: At the end, the unencrypted request is passed to the web server

**Proxy Tools**:Proxy Switcher, CyberGhost VPN…

* **Anonymizers:**

Remove all id info from the user;s computer

Make activity on the Internet untraceable

Allow you to bypass Internet cernsor

**Why use?**

Privacy and anoymity, Protection against online attacks, Access restricted content, Bypass IDS and firewall rules.

**Censorship Circumvention Tools:** Alkasir, Tails

**Anonymizers:** Whonix, Psiphon

**Network Discovery and Mapping Tools**

* Discover a network and produces a comprehensive network diagram.
* Display in-depth connections such as **OSI Layer2 and Layer3** topology data