Initial Scans

1. Four Different Types of Initial Scans in Penetration Testing

Before any detailed network or system attack, pentesters typically start by scanning the network. This helps identify live hosts, open ports, and services. The four main types of scans are:

A. Ping Discovery

- **Purpose**: Identifies live hosts by sending ICMP echo requests (pings) to see if the target system is reachable.
- **Command**:This sends pings to the network and reports which hosts respond.

```
nmp -sn [target IP/subnet]
```

B. TCP Discovery

- **Purpose**: Uses TCP SYN or TCP connect methods to discover open ports on the target.
- **Commands**:
  - **TCP SYN scan**:

    ```
    nmap -sS [target IP/subnet]
    ```

    This performs a "half-open" scan by sending a SYN packet. If the host responds with a SYN-ACK, the port is open.
  - **TCP Connect scan**:

    ```
    nmap -sT [target IP/subnet]
    ```

This completes the full TCP connection (SYN, SYN-ACK, ACK) and is less stealthy than the SYN scan but useful if SYN scan is blocked.

C. Reverse DNS Discovery

- **Purpose**: Attempts to resolve IP addresses to domain names using DNS, which helps identify potential targets by their hostnames.
- **Commands**:
  - **Reverse DNS Lookup**:

    ```
    nmap -sL [target IP/subnet]
    ```

    This lists the IP addresses and hostnames without actually scanning the target.
  - **Output Results in All Formats**:

    ```
    nmap -oA [output file prefix] [target IP/subnet]
    ```

    This command saves the scan results in three formats: `.nmap`, `.xml`, and `.gnmap`. These formats are useful for later parsing and automation.

D. UDP Discovery

- **Purpose**: Scans for open UDP ports, which is often slower than TCP due to the connectionless nature of UDP.
- **Command**: A UDP scan tries to discover services that use UDP rather than TCP, such as DNS (UDP port 53), SNMP (UDP port 161), and others.

  ```
  nmap -sU [target IP/subnet]
  ```

---

2. Optimizing Scan Speed & Accuracy

A. Host Grouping for Real-Time Results

- **Command**:

```
map [scan options] --min-hostgroup 1024 [target IP/subnet]
```

- This option speeds up the scan by grouping hosts in chunks (1024 at a time). However, this can slow down the overall scan completion time as it tries to give results in real time.

B. Adjusting Timing for Aggressiveness

- **Aggressive Timing**:

```
nmap [scan options] -T4 [target IP/subnet]
```

  - The `T4` option makes the scan faster but potentially less accurate. Aggressive scans are more likely to miss open ports (false negatives) but typically won't show ports as open when they are closed (no false positives). The timing template ranges from `T0` (slowest and stealthiest) to `T5` (fastest and least stealthy).

C. Scanning Specific Top Ports

- **Command**:

```
nmap [scan options] --top-ports [number] [target IP/subne
t]
```

  - This option tells nmap to scan only the most common `N` ports (e.g., `-top-ports 1000`). When combined with faster timing options like `T4`, this reduces scan time.

---

3. Outputting Results for Grepping and Post-Processing

When scanning large networks, it is helpful to output results in a format that can be easily searched or processed later, like greppable (`.gnmap`) format. This allows the pentester to filter out specific data, such as live hosts or open ports.

A. Parsing Ping Scan Results

- **Command**:

```
cat ping.gnmap | grep Up | cut -d ' ' -f2 > live.txt
```

  - This command filters the output of a ping scan (`ping.gnmap`), identifies hosts that are up, and extracts their IP addresses, saving them into a file called `live.txt`.

B. Parsing TCP Discovery Results

- **Command**:

```
grep 'open/' [tcp.gnmap] > open_ports.txt
```

  - This command searches through a TCP discovery scan result file (`tcp.gnmap`) and finds lines that indicate open ports, saving the results to `open_ports.txt`.

C. Combining Results from Multiple Scans

- **Command**:

```
cat liveping.txt livetcp.txt | sort -uV > live.txt
```

  - This combines results from multiple scans (e.g., ping and TCP discovery) into a single, sorted list of unique IP addresses (`live.txt`).

---

4. Random Target Selection from Results

- **Command**:

```
shuf -n 200 live.txt > targets.txt
```

  - This command selects 200 random hosts from the live hosts list (`live.txt`) and writes them to `targets.txt`. This can be useful for large networks where it is impractical to scan every host, allowing pentesters to target a random sample for further testing.