📘 Methodology & Tooling Standard Operating Procedure

Project: Network Discovery & Service Mapping

Task Reference: DEV-353

Date: January 2026

1. 📋 Objective

The primary objective of this document is to establish a **transparent** and **reproducible** scanning methodology for the infrastructure security audit. It defines the specific toolchain, command structures, and parameter choices used to map the network attack surface without causing service disruption.

2. 🛠️ Environment & Toolchain

The audit was executed within a controlled Linux environment to ensure raw socket access and tool compatibility.

Category	Tool	Version	Purpose
Scanner	Nmap	7.95	Core engine for host discovery, port scanning, and service versioning.
OS	Ubuntu Linux	22.04 LTS	Execution environment (CLI).
Validation	Netcat (nc)	1.10	Manual TCP handshake verification for ambiguous ports.
Validation	Curl	7.81	HTTP header fetching for web service validation.
Scripting	Bash	5.0+	Automation of scan loops and output management.

---

3. ⚙️ Execution Methodology

The scanning process was divided into three distinct phases to prioritize efficiency and accuracy.

Phase 1: Host Discovery (Ping Sweep)

Goal: Identify active IP addresses within the provided subnet/list to build a "Live Host" inventory.

- **Command:** nmap -sn -iL targets.txt
- **Output:** A list of 15 active hosts was confirmed.

Phase 2: Comprehensive Service Discovery

Goal: Scan all 65,535 ports on the identified hosts to map running services and versions.

- **Command:**

Bash

```
sudo nmap -sV -p- -T4 -Pn -iL targets.txt -oN scan_results.txt
```

- **Why sudo?** Root privileges were required to perform TCP SYN scanning (stealth) and OS fingerprinting.

Phase 3: Validation (False Positive Reduction)

Goal: Manually verify high-risk findings to ensure report accuracy (DEV-356).

- **Action:** Targeted re-scans using netcat were performed on flagged ports (e.g., Redis: 9111) to confirm external reachability.

---

4. 📖 Parameter Technical Definitions

The following Nmap flags were specifically chosen to balance **speed** with **completeness**.

Flag	Parameter Name	Technical Justification
-sV	**Service Versioning**	Probes open ports to determine the exact service name and version (e.g., distinguishing "Apache 2.4" from generic "HTTP"). Essential for CVE mapping.
-p-	**All Ports**	Scans the full port range (1–65535). Standard scans only check the top 1,000 ports; this flag ensures "Shadow IT" services on non-standard ports (e.g., 8085, 9000) are not missed.
-T4	**Timing Template**	"Aggressive" timing. Reduces RTT (Round Trip Time) timeouts to speed up the scan on broadband networks while retaining congestion control.
-Pn	**No Ping**	Treats all hosts as "Online." This bypasses firewalls that are configured to block ICMP Echo requests, ensuring we don't miss stealthy hosts.
--privileged	**Raw Sockets**	Forces Nmap to use raw packet sockets instead of the OS network stack, increasing scan accuracy and allowing for OS detection.

---

5. 🔄 Reproducibility Guide

To reproduce these results exactly:

1. **Prepare Inventory:** Ensure a file named targets.txt exists with the 15 target IP addresses.
2. **Environment:** Use a Debian-based Linux distribution (Ubuntu/Kali).
3. **Permissions:** Ensure the user has sudo access.
4. **Execution:** Run the command documented in **Phase 2** exactly as written.
5. **Validation:** Compare the output hash of the new scan_results.txt with the archived log to verify consistency.