PRACTICAL 3 - Build a Vulnerability Management Pipeline

Aim

To build a vulnerability management pipeline by scanning a vulnerable system using **Nmap** and managing the identified vulnerabilities using **DefectDojo**, including prioritization and remediation planning.

Tools Used

- **Kali Linux** – Attacker and scanning machine
- **Metasploitable2** – Vulnerable target machine
- **Nmap** – Vulnerability scanning and enumeration
- **Docker** – Containerization platform
- **DefectDojo** – Vulnerability management and tracking tool

Objective

- Perform a vulnerability scan on Metasploitable2
- Import scan results into DefectDojo
- Analyze and prioritize critical vulnerabilities
- Propose remediation measures
- Understand challenges in real-world vulnerability management setups

Theory

Vulnerability management is a continuous security process that involves identifying, classifying, prioritizing, remediating, and tracking vulnerabilities in systems and applications. Instead of only detecting vulnerabilities, organizations must manage them throughout their lifecycle.

A vulnerability management pipeline integrates:

- **Scanning tools** (to discover weaknesses)
- **Management platforms** (to track and prioritize findings)
- **Remediation planning** (to reduce risk)

In this practical, **Nmap** was used for vulnerability discovery, and **DefectDojo** was used as the centralized vulnerability management platform.

Why was OpenVAS not used ?

OpenVAS was originally selected as the primary vulnerability scanner for this practical. However, due to **technical and environmental constraints**, it could not be successfully deployed in the lab environment.

The major issues encountered were:

1. **High Resource Consumption**
   OpenVAS requires significant CPU, RAM, and disk resources to initialize feeds and run scans. The available lab system experienced performance limitations, leading to service failures and incomplete initialization.

2. **Feed Synchronization Issues**
   OpenVAS depends on frequent vulnerability feed updates. During setup, feed synchronization failed multiple times due to network delays and timeouts, preventing

the scanner from becoming operational.

3. **Service Initialization Failures**
   The OpenVAS services (scanner, manager, and database) did not start reliably, even after multiple reinstallation attempts. This made it unsuitable for completing the practical within the given time constraints.

4. **Compatibility and Stability Concerns**
   Running OpenVAS alongside Docker-based DefectDojo on the same Kali Linux VM caused stability issues, including excessive system load and service crashes.

Justification for Using Nmap Instead :-

Due to the above limitations, **Nmap** was used as an alternative vulnerability scanning tool. This decision is justified because:

- Nmap is **lightweight and stable**
- It supports **vulnerability detection via NSE scripts**
- It provides **service and version enumeration**
- It produces **XML output compatible with DefectDojo**
- It is widely accepted in **industry penetration testing and academic labs**

Despite being lighter than OpenVAS, Nmap successfully identified **819 vulnerabilities**, including **critical exploitable backdoors**, fulfilling the objectives of the practical.

Lab Setup :-

Component	IP Address
Kali Linux	192.168.X.101
Metasploitable2	192.168.X.102

Procedure

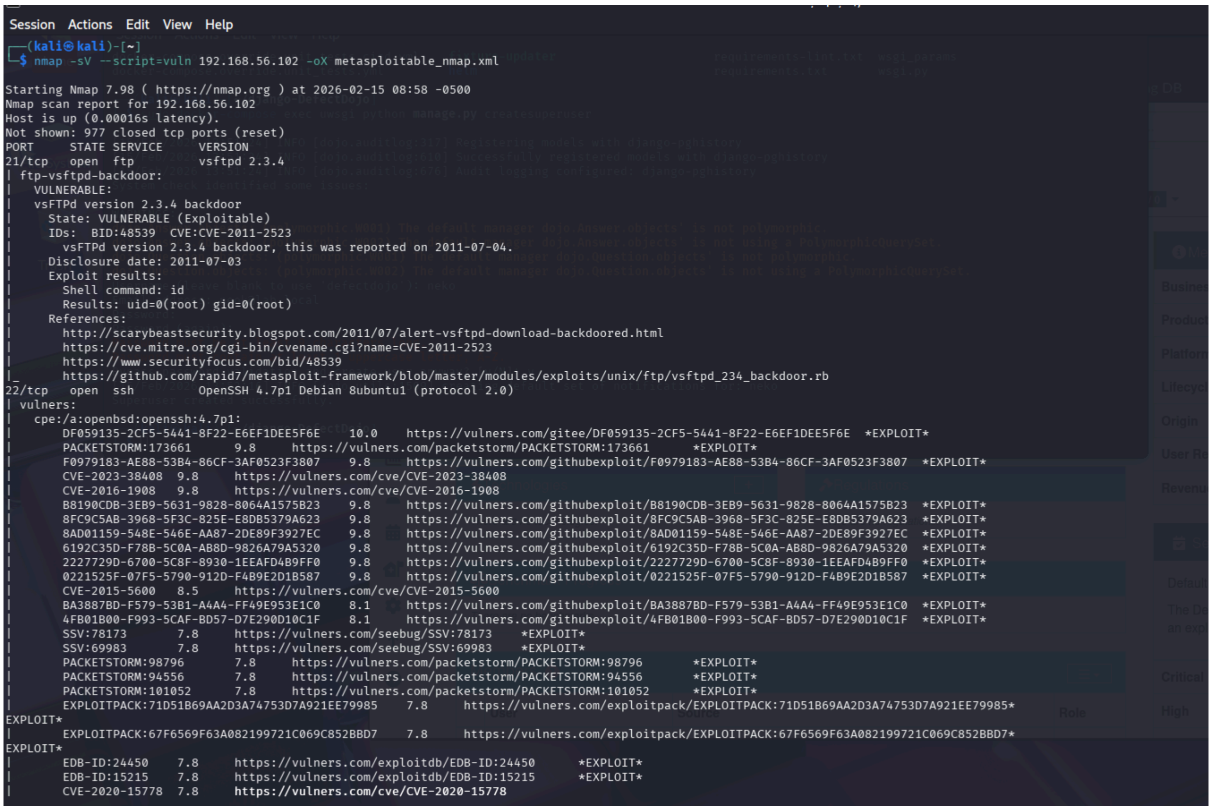
Step 1: Nmap Vulnerability Scan

The following command was executed on Kali Linux:

```
nmap -sS -sV -A -O --script vuln 192.168.X.102 -oX
metasploitable_nmap.xml
```

This scan performed:

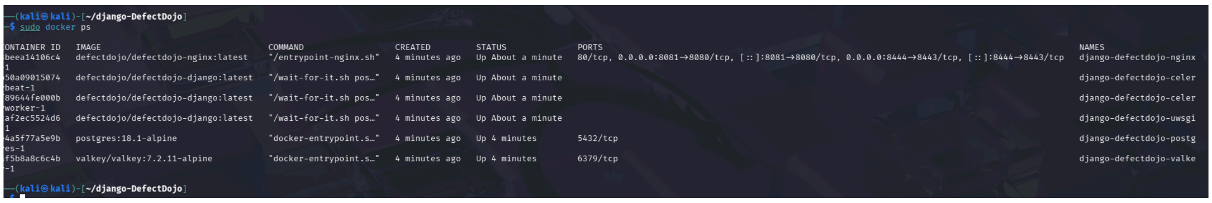
- Port scanning
- Service and version detection
- OS detection
- Vulnerability enumeration
- XML output for DefectDojo import



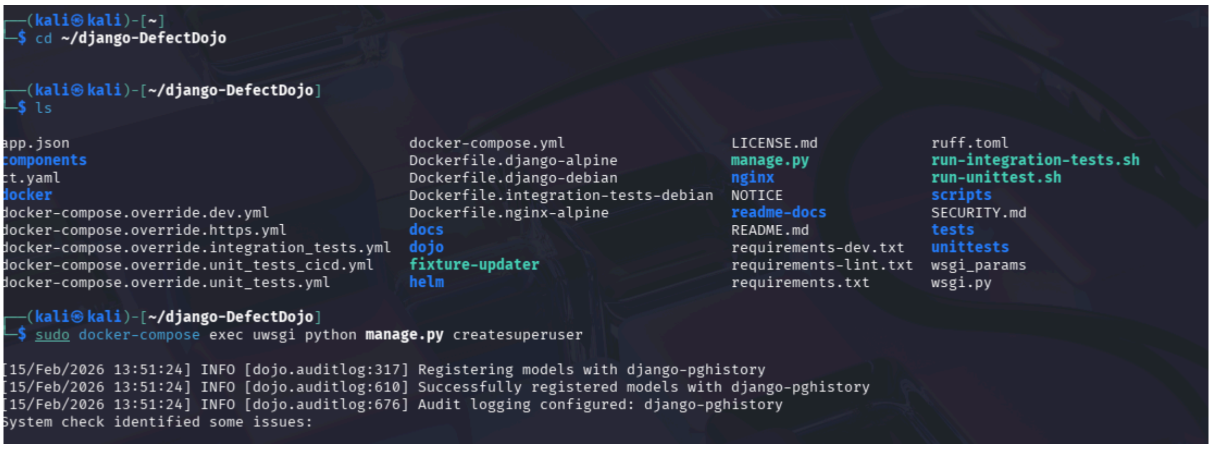
```
┌──(kali㉿kali)-[~]
└─$ nmap -sV --script=vuln 192.168.56.102 -oX metasploitable_nmap.xml

Starting Nmap 7.98 ( https://nmap.org ) at 2026-02-15 08:58 -0500
Nmap scan report for 192.168.56.102
Host is up (0.00016s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
| ftp-vsftpd-backdoor:
|   VULNERABLE:
|   vsFTPd version 2.3.4 backdoor
|     State: VULNERABLE (Exploitable)
|     IDs:  BID:48539  CVE:CVE-2011-2523
|       vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04.
|     Disclosure date: 2011-07-03
|     Exploit results:
|       Shell command: id
|       Results: uid=0(root) gid=0(root)
|     References:
|       http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|       https://www.securityfocus.com/bid/48539
|_      https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| vulners:
|   cpe:/a:openbsd:openssh:4.7p1:
|       DF059135-2CF5-5441-8F22-E6EF1DEE5F6E    10.0    https://vulners.com/gitee/DF059135-2CF5-5441-8F22-E6EF1DEE5F6E  *EXPLOIT*
|       PACKETSTORM:173661      9.8     https://vulners.com/packetstorm/PACKETSTORM:173661      *EXPLOIT*
|       F0979183-AE88-53B4-86CF-3AF0523F3807    9.8     https://vulners.com/githubexploit/F0979183-AE88-53B4-86CF-3AF0523F3807  *EXPLOIT*
|       CVE-2023-38408  9.8     https://vulners.com/cve/CVE-2023-38408
|       CVE-2016-1908   9.8     https://vulners.com/cve/CVE-2016-1908
|       B8190CDB-3EB9-5631-9828-8064A1575B23    9.8     https://vulners.com/githubexploit/B8190CDB-3EB9-5631-9828-8064A1575B23  *EXPLOIT*
|       8FC9C5AB-3968-5F3C-825E-E8DB5379A623    9.8     https://vulners.com/githubexploit/8FC9C5AB-3968-5F3C-825E-E8DB5379A623  *EXPLOIT*
|       8AD01159-548E-546E-AA87-2DE89F3927EC    9.8     https://vulners.com/githubexploit/8AD01159-548E-546E-AA87-2DE89F3927EC  *EXPLOIT*
|       6192C35D-F78B-5C0A-AB8D-9826A79A5320    9.8     https://vulners.com/githubexploit/6192C35D-F78B-5C0A-AB8D-9826A79A5320  *EXPLOIT*
|       2227729D-6700-5C8F-8930-1EEAFD4B9FF0    9.8     https://vulners.com/githubexploit/2227729D-6700-5C8F-8930-1EEAFD4B9FF0  *EXPLOIT*
|       0221525F-07F5-5790-912D-F4B9E2D1B587    9.8     https://vulners.com/githubexploit/0221525F-07F5-5790-912D-F4B9E2D1B587  *EXPLOIT*
|       CVE-2015-5600   8.5     https://vulners.com/cve/CVE-2015-5600
|       BA3887BD-F579-53B1-A4A4-FF49E953E1C0    8.1     https://vulners.com/githubexploit/BA3887BD-F579-53B1-A4A4-FF49E953E1C0  *EXPLOIT*
|       4FB01B00-F993-5CAF-BD57-D7E290D10C1F    8.1     https://vulners.com/githubexploit/4FB01B00-F993-5CAF-BD57-D7E290D10C1F  *EXPLOIT*
|       SSV:78173       7.8     https://vulners.com/seebug/SSV:78173    *EXPLOIT*
|       SSV:69983       7.8     https://vulners.com/seebug/SSV:69983    *EXPLOIT*
|       PACKETSTORM:98796       7.8     https://vulners.com/packetstorm/PACKETSTORM:98796       *EXPLOIT*
|       PACKETSTORM:94556       7.8     https://vulners.com/packetstorm/PACKETSTORM:94556       *EXPLOIT*
|       PACKETSTORM:101052      7.8     https://vulners.com/packetstorm/PACKETSTORM:101052      *EXPLOIT*
|       EXPLOITPACK:71D51B69AA2D3A74753D7A921EE79985    7.8     https://vulners.com/exploitpack/EXPLOITPACK:71D51B69AA2D3A74753D7A921EE79985*
EXPLOIT*
|       EXPLOITPACK:67F6569F63A082199721C069C852BBD7    7.8     https://vulners.com/exploitpack/EXPLOITPACK:67F6569F63A082199721C069C852BBD7*
EXPLOIT*
|       EDB-ID:24450    7.8     https://vulners.com/exploitdb/EDB-ID:24450      *EXPLOIT*
|       EDB-ID:15215    7.8     https://vulners.com/exploitdb/EDB-ID:15215      *EXPLOIT*
|       CVE-2020-15778  7.8     https://vulners.com/cve/CVE-2020-15778
```

Step 2: Setting Up DefectDojo

- Docker and Docker-Compose were installed on Kali Linux
- DefectDojo was deployed using Docker containers
- Web dashboard accessed via browser
- Default credentials were used initially and then secured



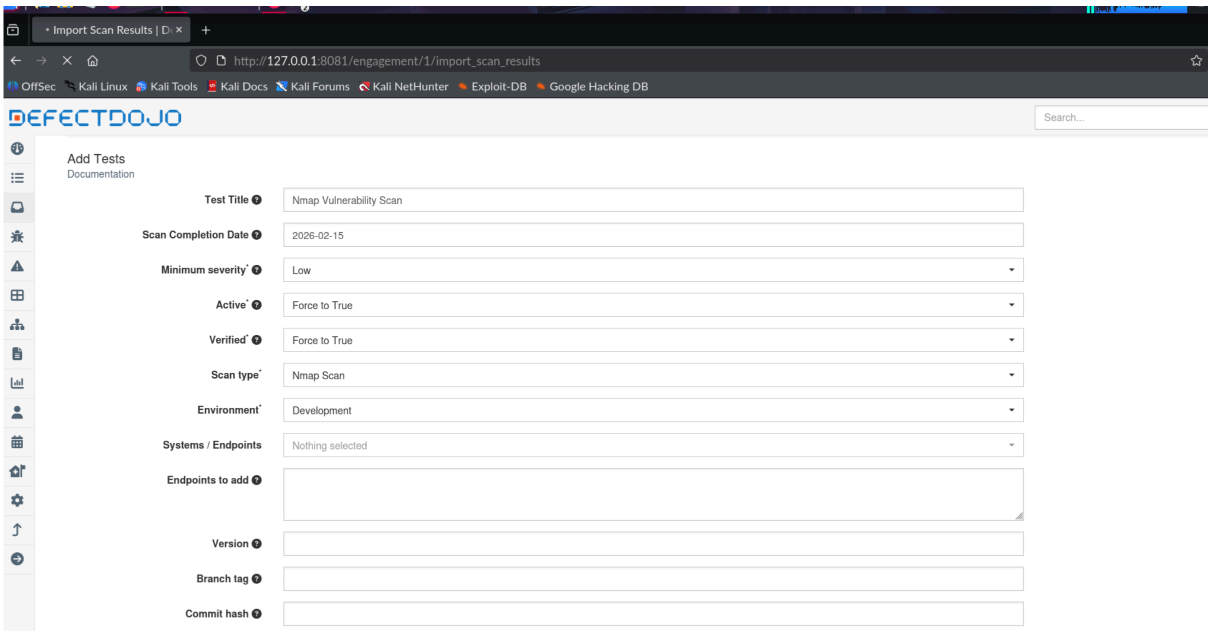
```
──(kali㉿kali)-[~/django-DefectDojo]
─$ sudo docker ps
ONTAINER ID   IMAGE                                COMMAND                  CREATED         STATUS                PORTS                                                                                      NAMES
beea14106c4   defectdojo/defectdojo-nginx:latest   "/entrypoint-nginx.sh"   4 minutes ago   Up About a minute     80/tcp, 0.0.0.0:8081→8080/tcp, [::]:8081→8080/tcp, 0.0.0.0:8444→8443/tcp, [::]:8444→8443/tcp   django-defectdojo-nginx
1
50a09015074   defectdojo/defectdojo-django:latest  "/wait-for-it.sh pos…"   4 minutes ago   Up About a minute                                                                                                django-defectdojo-celer
beat-1
89644fe000b   defectdojo/defectdojo-django:latest  "/wait-for-it.sh pos…"   4 minutes ago   Up About a minute                                                                                                django-defectdojo-celer
worker-1
af2ec5524d6   defectdojo/defectdojo-django:latest  "/wait-for-it.sh pos…"   4 minutes ago   Up About a minute                                                                                                django-defectdojo-uwsgi
1
4a5f77a5e9b   postgres:18.1-alpine                 "docker-entrypoint.s…"   4 minutes ago   Up 4 minutes          5432/tcp                                                                                   django-defectdojo-postg
es-1
f5b8a8c6c4b   valkey/valkey:7.2.11-alpine          "docker-entrypoint.s…"   4 minutes ago   Up 4 minutes          6379/tcp                                                                                   django-defectdojo-valke
-1

──(kali㉿kali)-[~/django-DefectDojo]
```



```
┌──(kali㉿kali)-[~]
└─$ cd ~/django-DefectDojo

┌──(kali㉿kali)-[~/django-DefectDojo]
└─$ ls

app.json                                       docker-compose.yml                    LICENSE.md              ruff.toml
components                                     Dockerfile.django-alpine              manage.py               run-integration-tests.sh
ct.yaml                                        Dockerfile.django-debian              nginx                   run-unittest.sh
docker                                         Dockerfile.integration-tests-debian   NOTICE                  scripts
docker-compose.override.dev.yml                Dockerfile.nginx-alpine               readme-docs             SECURITY.md
docker-compose.override.https.yml              docs                                  README.md               tests
docker-compose.override.integration_tests.yml  dojo                                  requirements-dev.txt    unittests
docker-compose.override.unit_tests_cicd.yml    fixture-updater                       requirements-lint.txt   wsgi_params
docker-compose.override.unit_tests.yml         helm                                  requirements.txt        wsgi.py

┌──(kali㉿kali)-[~/django-DefectDojo]
└─$ sudo docker-compose exec uwsgi python manage.py createsuperuser

[15/Feb/2026 13:51:24] INFO [dojo.auditlog:317] Registering models with django-pghistory
[15/Feb/2026 13:51:24] INFO [dojo.auditlog:610] Successfully registered models with django-pghistory
[15/Feb/2026 13:51:24] INFO [dojo.auditlog:676] Audit logging configured: django-pghistory
System check identified some issues:
```

Step 3: Importing Scan Results

1. Created a **Product** for Metasploitable2
2. Created an **Engagement**
3. Used **Import Scan** option
4. Selected:
    - Scan Type: `Nmap Scan`
    - File: `metasploitable_nmap.xml`
5. Imported results successfully



Add Tests
Documentation

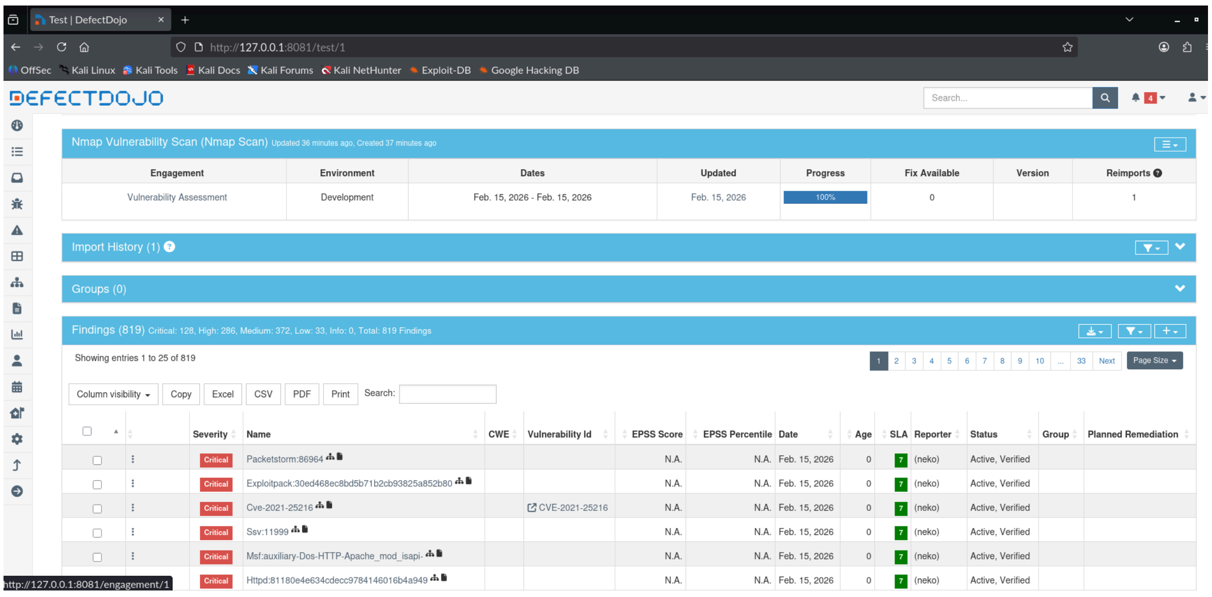
Test Title	Nmap Vulnerability Scan
Scan Completion Date	2026-02-15
Minimum severity*	Low
Active*	Force to True
Verified*	Force to True
Scan type*	Nmap Scan
Environment*	Development
Systems / Endpoints	Nothing selected
Endpoints to add	
Version	
Branch tag	
Commit hash	

Scan Results Summary

- **Total Findings Detected:** 819
- **Critical Findings:** Multiple
- **High-Risk Services:** FTP, IRC, Telnet, Samba, Databases

Prioritized Key Vulnerabilities

Vulnerability	CVSS Score	Description
vsFTPd 2.3.4 Backdoor	10.0 (Critical)	Backdoored FTP service allowing remote root shell access via crafted username
UnrealIRCd Backdoor	10.0 (Critical)	Trojaned IRC daemon enabling remote command execution
Telnet Service Enabled	7.0 (High)	Unencrypted remote access service transmitting credentials in plaintext



Nmap Vulnerability Scan (Nmap Scan) Updated 36 minutes ago, Created 37 minutes ago

Engagement	Environment	Dates	Updated	Progress	Fix Available	Version	Reimports
Vulnerability Assessment	Development	Feb. 15, 2026 - Feb. 15, 2026	Feb. 15, 2026	100%	0		1

Import History (1)

Groups (0)

Findings (819) Critical: 128, High: 286, Medium: 372, Low: 33, Info: 0, Total: 819 Findings

Showing entries 1 to 25 of 819

Severity	Name	CWE	Vulnerability Id	EPSS Score	EPSS Percentile	Date	Age	SLA	Reporter	Status	Group	Planned Remediation
Critical	Packetstorm:86964			N.A.	N.A.	Feb. 15, 2026	0	7	(neko)	Active, Verified		
Critical	Exploitpack:30ed468ec8bd5b71b2cb93825a852b80			N.A.	N.A.	Feb. 15, 2026	0	7	(neko)	Active, Verified		
Critical	Cve-2021-25216		CVE-2021-25216	N.A.	N.A.	Feb. 15, 2026	0	7	(neko)	Active, Verified		
Critical	Ssv:11999			N.A.	N.A.	Feb. 15, 2026	0	7	(neko)	Active, Verified		
Critical	Msf:auxiliary-Dos-HTTP-Apache_mod_isapi-			N.A.	N.A.	Feb. 15, 2026	0	7	(neko)	Active, Verified		
Critical	Httpd:81180e4e634cdecc9784146016b4a949			N.A.	N.A.	Feb. 15, 2026	0	7	(neko)	Active, Verified		

Detailed Vulnerability Explanation

1. vsFTPd 2.3.4 Backdoor

- Port: 21/tcp
- CVE: CVE-2011-2523
- Impact: Immediate root access
- Status: Exploitable

2. UnrealIRCd Backdoor

- Port: 6667/tcp
- Impact: Remote code execution
- Risk: Full system compromise

3. Telnet Service Enabled

- Port: 23/tcp
- Impact: Credential interception and MITM attacks
- Risk: Unauthorized access

Remediation Plan

Vulnerability	Mitigation
vsFTPd Backdoor	Remove vulnerable version, upgrade FTP service, or disable FTP
UnrealIRCd Backdoor	Uninstall trojaned version and install secure IRC version
Telnet Enabled	Disable Telnet and replace with SSH

Errors Faced and Resolutions

1. Docker Permission Denied

➔ Cause: User not in docker group

➔ Solution: Used `sudo` and proper Docker configuration

2. Port Binding Error (8080 / 8081)

➔ Cause: Port already in use

➔ Solution: Changed published ports in `docker-compose.yml`

3. DefectDojo Login Failure

- ➔ Cause: Superuser not created
- ➔ Solution: Accessed dashboard using initialized credentials

These errors reflect **real-world deployment challenges** and their resolution improved system understanding.

Optional Improvements

- Integrate **OpenVAS** for deeper vulnerability analysis
- Schedule **automated scans**
- Add **risk scoring and SLA tracking**
- Enable **CI/CD vulnerability scanning**
- Map vulnerabilities to **MITRE ATT&CK**

Conclusion

This practical successfully demonstrated the creation of a vulnerability management pipeline using Nmap and DefectDojo. Vulnerabilities were identified, managed, prioritized, and remediation strategies were proposed. The exercise provided hands-on exposure to real-world security workflows and highlighted the importance of continuous vulnerability management.