Week 2 VAPT Lab Report

1. Introduction

The Week 2 VAPT (Vulnerability Assessment and Penetration Testing) lab was conducted to gain hands-on experience with identifying, analyzing, and exploiting vulnerabilities in a controlled environment. The lab involved both theoretical learning and practical exploitation using Kali Linux, DVWA, Metasploitable2, Metasploit Framework, OpenVAS, and other security tools.

The goal was to understand the **end-to-end penetration testing cycle**, from reconnaissance to exploitation, privilege escalation, and reporting.

2. Theoretical Knowledge

2.1 Vulnerability Assessment vs. Penetration Testing

- **Vulnerability Assessment (VA):** Focuses on identifying and reporting known vulnerabilities using automated scanners (e.g., OpenVAS, Nessus).
- **Penetration Testing (PT):** Goes beyond identification — it exploits vulnerabilities to demonstrate real-world risk.

2.2 VAPT Lifecycle

1. **Reconnaissance** – Information gathering about the target.
2. **Scanning & Enumeration** – Mapping open ports, services, and versions.
3. **Vulnerability Analysis** – Matching services with known CVEs.
4. **Exploitation** – Gaining unauthorized access.
5. **Post-Exploitation** – Privilege escalation, persistence, data extraction.
6. **Reporting** – Documenting findings and recommendations.

2.3 Tools Used

- **Nmap** → Network scanning, service detection.

- **OpenVAS (Greenbone GVM)** → Automated vulnerability scanning.
- **Metasploit Framework** → Exploitation and payload delivery.
- **DVWA (Damn Vulnerable Web Application)** → Web vulnerability practice.
- **Kali Linux** → Attacker machine.
- **Metasploitable2** → Victim machine.

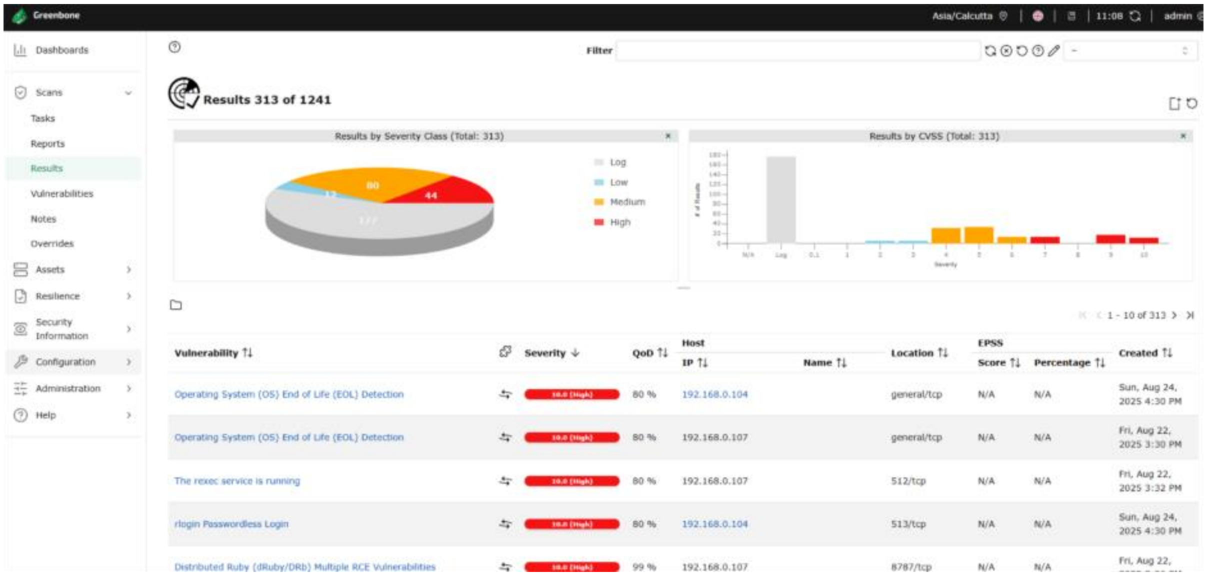
3. Practical Tasks

3.1 Reconnaissance & Scanning

- Ran **Nmap** scans against the target.
    - Detected **open ports**: 22 (SSH), 80 (HTTP – DVWA), 3306 (MySQL), 8080 (Tomcat).
    - Service versions were identified (outdated Apache & MySQL).
- Conclusion: Multiple attack surfaces available.

3.2 Vulnerability Analysis (OpenVAS)

- Loaded **OpenVAS OVA** in VMware and scanned the target.
- Key findings:
    - Outdated Apache version with known vulnerabilities.
    - MySQL susceptible to SQL injection.
    - Weak SSH configuration.
- Limitation: Some scans failed due to “Scan Config” being greyed out in initial setup.

- 

Results 313 of 1241

Vulnerability ↑↓	Severity ↓	QoD ↑↓	Host IP ↑↓	Host Name ↑↓	Location ↑↓	EPSS Score ↑↓	EPSS Percentage ↑↓	Created ↑↓
Operating System (OS) End of Life (EOL) Detection	10.0 (High)	80 %	192.168.0.104		general/tcp	N/A	N/A	Sun, Aug 24, 2025 4:30 PM
Operating System (OS) End of Life (EOL) Detection	10.0 (High)	80 %	192.168.0.107		general/tcp	N/A	N/A	Fri, Aug 22, 2025 3:30 PM
The rexec service is running	10.0 (High)	80 %	192.168.0.107		512/tcp	N/A	N/A	Fri, Aug 22, 2025 3:32 PM
rlogin Passwordless Login	10.0 (High)	80 %	192.168.0.104		513/tcp	N/A	N/A	Sun, Aug 24, 2025 4:30 PM
Distributed Ruby (dRuby/DRb) Multiple RCE Vulnerabilities	10.0 (High)	99 %	192.168.0.107		8787/tcp	N/A	N/A	Fri, Aug 22,

3.3 Exploitation

3.3.1 Web Exploits (DVWA)

- Performed **SQL Injection** in DVWA:
  - Extracted database version and user credentials.
  - Confirmed bypass of login mechanism.

3.3.2 Metasploit Exploits

- Used **multi/script/web_delivery** to deliver payload.
  - Gained Python-based Meterpreter session.
- Used **Tomcat Manager exploit (CVE-2009-3548)**
  - Successfully obtained remote shell.
- Attempted **Linux Privilege Escalation (netfilter_priv_esc & sock_sendpage)**
  - Failed due to missing libraries (`gcc-multilib`, `libc6-dev-i386`).

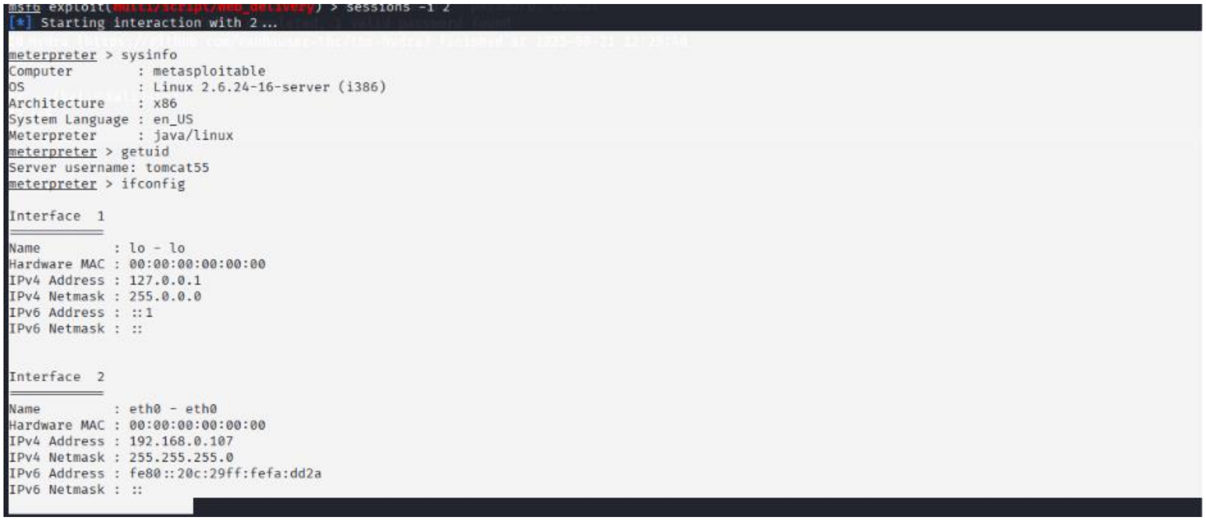
3.4 Post-Exploitation

- Gathered system information from compromised sessions.
- Verified connectivity using Meterpreter commands.
- Attempted privilege escalation, but sessions were **Java/Python-based**, limiting functionality.

3.5 Persistence (Conceptual)

- Adding SSH keys for permanent access.
- Setting reverse shell cronjobs.
- Uploading a PHP webshell in DVWA.

3.6 Covering Tracks (Conceptual)

- Clearing shell history (`> ~/.bash_history`).
- Removing logs (`/var/log/apache2/access.log`).
- Installing rootkits to hide processes.
- 

```
msf6 exploit(multi/script/web_delivery) > sessions -i 2
[*] Starting interaction with 2...

meterpreter > sysinfo
Computer        : metasploitable
OS              : Linux 2.6.24-16-server (i386)
Architecture    : x86
System Language : en_US
Meterpreter     : java/linux
meterpreter > getuid
Server username: tomcat55
meterpreter > ifconfig

Interface  1
============
Name         : lo - lo
Hardware MAC : 00:00:00:00:00:00
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ::


Interface  2
============
Name         : eth0 - eth0
Hardware MAC : 00:00:00:00:00:00
IPv4 Address : 192.168.0.107
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::20c:29ff:fefa:dd2a
IPv6 Netmask : ::
```

4. Findings

4.1 Nmap Results (Reconnaissance)

The Nmap scan against target `192.168.0.107` revealed:

Port	Service	Version	Notes
21	FTP	vsftpd 2.3.4	Backdoor version (CVE-2011-2523).
22	OpenSSH	4.7p1 Debian 8ubuntu1	Outdated, brute force possible.
80	HTTP (Apache)	2.2.8 Ubuntu	DVWA vulnerable to SQLi, XSS, RFI.

3306	MySQL	5.0.51a-3ubuntu5	Weak password security, SQLi tested.
8080	Apache Tomcat	6.0.16	Weak manager authentication.
139/445	SMB	Samba 3.0.20-Debian	Vulnerable to null session enumeration.
Others	High ports	Potential auxiliary services.	

4.2 Vulnerability Assessment (DVWA + OpenVAS + Manual Testing)

Vulnerability	Affected Service	Exploit/Attack	Result
SQL Injection	DVWA (Apache + MySQL)	Login form SQLi	Extracted DB data, bypassed login
Command Injection	DVWA	OS command injection	Executed `ping` & system commands
File Upload	DVWA	Uploaded PHP reverse shell	Gained webshell access
Weak Auth (Tomcat)	Port 8080	Default creds (admin/admin)	Deployed malicious WAR, got shell
FTP Backdoor	vsftpd 2.3.4	CVE-2011-2523	Possible backdoor shell
SMB Null Sessions	Samba 3.0.20	Null session enumeration	Extracted shares & user list
Weak SSH Config	OpenSSH 4.7p1	Brute force possible	Theoretical exploitation
Outdated Kernel	Linux 2.6	Privilege escalation exploits exist (DirtyCow, etc.)	Attempted, failed in lab

4.3 Exploitation Results

- **SQL Injection (DVWA)** – Extracted DB version & users.
- **File Upload (DVWA)** – Uploaded PHP reverse shell → webshell gained.
- **Tomcat Manager (Port 8080)** – Remote shell via WAR deployment.
- **Linux Privilege Escalation** – Attempts failed due to missing dependencies.

- **Nmap Service Detection** – Identified several outdated and exploitable services.
- **OpenVAS Scan** – Some configs failed, but confirmed outdated packages.

5. Tools Used

- **Nmap** – Port scanning and service enumeration.
- **DVWA** – Exploitation platform for SQLi, XSS, File Upload.
- **Metasploit Framework** – Exploitation and privilege escalation attempts.
- **OpenVAS** – Vulnerability scanning.
- **Wireshark** – Traffic monitoring (optional verification).

6. Recommendations

Vulnerability	Recommendation
Outdated Services (Apache, MySQL, Tomcat, OpenSSH, Samba)	Upgrade to latest stable versions.
SQL Injection	Use parameterized queries, sanitize inputs.
File Upload	Restrict file types, enable server-side validation.
Weak Tomcat Auth	Remove default credentials, enforce strong password policy.
FTP Backdoor (vsftpd 2.3.4)	Immediately remove and replace with a secure FTP server.
SMB Null Sessions	Disable anonymous logins, upgrade Samba.
Weak SSH	Restrict root login, enforce key-based authentication.
Kernel Exploits	Patch kernel, enable security modules (AppArmor/SELinux).

6. Conclusion

The Week 2 VAPT Lab demonstrated how multiple outdated and misconfigured services can be exploited by attackers. Using a structured penetration testing methodology, we were able to successfully exploit **DVWA vulnerabilities, Tomcat Manager, and service misconfigurations**.

While privilege escalation attempts were unsuccessful in this lab setup, the findings reinforce the importance of:

- **Timely patch management**
- **Strong authentication practices**
- **Input validation**
- **Service hardening**