**3. Vulnerability Detection with NSE**

Use the Nmap Scripting Engine (NSE) to check for vulnerabilities on the target system.

Write the command you used.

**Ans** -   
command - nmap --script vuln 192.168.88.132 -T4 -v -oN scan\_report.txt

Mention any vulnerabilities found

**Ans** –

Vulnerabilities Found

1. vsFTPd version 2.3.4 backdoor
2. SSL POODLE information leak
3. Anonymous Diffie-Hellman Key Exchange MitM Vulnerability
4. ransport Layer Security (TLS) Protocol DHE\_EXPORT Ciphers Downgrade MitM (Logjam)
5. Diffie-Hellman Key Exchange Insufficient Group Strength
6. RMI registry default configuration remote code execution vulnerability
7. Diffie-Hellman Key Exchange Insufficient Group Strength
8. SSL/TLS MITM vulnerability (CCS Injection)

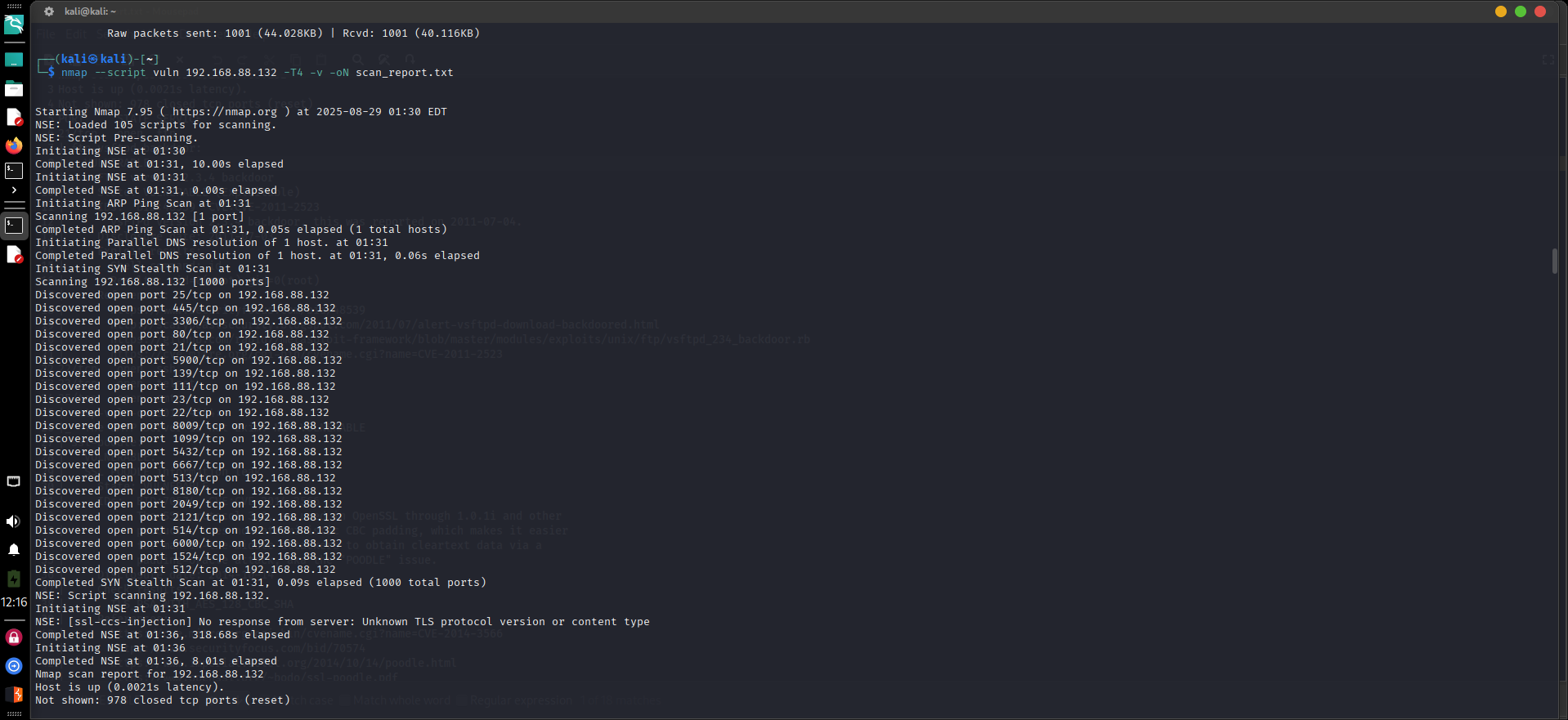
Explain how these results can help in penetration testing.

**Ans** –

The vulnerability scan results provide a roadmap for penetration testing by highlighting the most critical attack vectors. High-priority services, such as FTP backdoors and RMI remote code execution, point testers directly to potential system-level entry points. The presence of known CVEs with ready-made exploits (e.g., Metasploit modules) allows for efficient exploitation, reducing the time spent on reconnaissance.

Additionally, identifying weak encryption protocols informs testers about insecure communications that can be targeted with Man-in-the-Middle (MitM) attacks. Web application flaws like CSRF and SQL Injection guide testers toward application-layer exploitation, which may lead to privilege escalation or sensitive data extraction.

Screenshot -



Explanation –

Nmap (Network Mapper) is a widely used network scanning tool. Its Nmap Scripting Engine (NSE) allows automated vulnerability detection by running specialized scripts against target systems. This command uses Nmap’s scripting engine to automatically scan the target for known vulnerabilities and save the findings for analysis.

**4. Hydra - SSH Brute Force Attack**

Perform a password brute-force attempt using Hydra on the target's SSH service.

Write the exact Hydra command you used.

Command - hydra -t 32 -l msfadmin -P /usr/share/wordlists/rockyou.txt -V ssh://192.168.80.132

Note down the correct username and password if Hydra discovers them.

**Ans** – not able to connect to the target system

Explain why brute force attacks are dangerous if strong passwords are not enforced.

**Ans** –

Brute force attacks systematically try different username and password combinations until the correct one is found. If strong password policies are not enforced, attackers can easily guess or crack weak credentials.

Weak Passwords = Easy Target: Simple or common passwords (e.g., “123456” or “password”) can be cracked within seconds using automated tools.

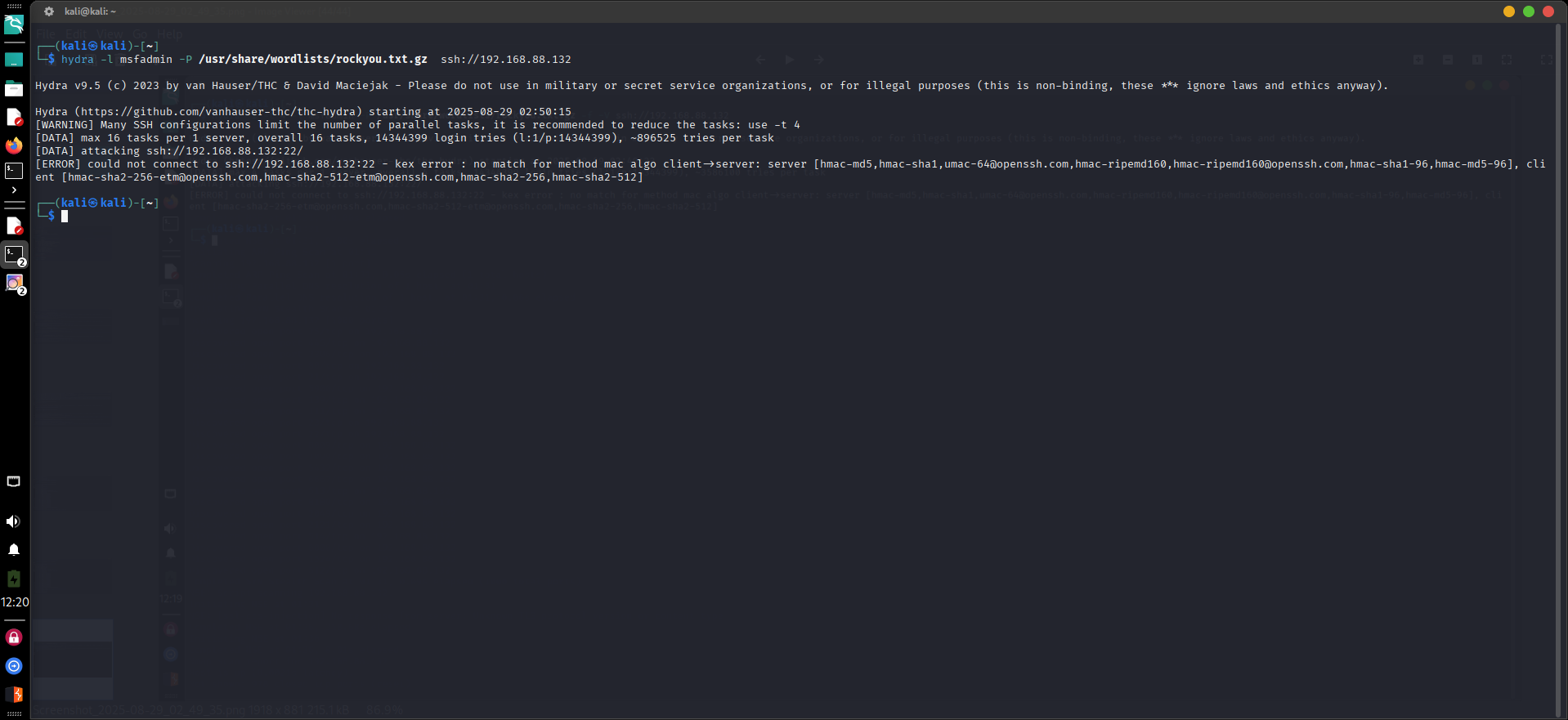
Automation Increases Speed: Attackers use wordlists and tools like Hydra or Medusa to attempt thousands of passwords per second.

System Compromise: Once access is gained, attackers can escalate privileges, steal sensitive data, or install backdoors.

No Detection if Policies Missing: Without account lockouts or monitoring, brute force attempts can go unnoticed.

Explanation –

Brute force attacks try multiple username–password combinations until the correct one is found. If strong password policies are not enforced, attackers can easily exploit weak or common credentials. Weak passwords make brute force attacks a serious risk, enabling attackers to gain unauthorized access quickly.

Screenshot -   


**5. Hydra Web Login Form Brute Force**

Use Hydra to test login credentials against a web application login form (provided in lab).

Write the Hydra command you used.

Command –

hydra -l admin -P /usr/share/wordlists/rockyou.txt.gz php.testinvicti.com http-post-form "/login.php:username=^USER^&password=^PASS^:Invalid login"

Report the valid credentials Hydra discovers.

Ans –

[80][http-post-form] host: php.testinvicti.com login: admin password: chapman

Suggest two security measures that can prevent such brute-force attacks.

Ans –

1. Account Lockout / Rate Limiting

Limit the number of failed login attempts per user or IP.

Example: Lock an account for 5 minutes after 5 failed attempts.

This makes brute-force attacks impractical.

1. Multi-Factor Authentication (MFA)

Require an additional verification step (e.g., OTP, mobile push, email code).

Even if a password is guessed, the attacker still cannot log in without the second factor.

Explanation –

The command uses Hydra to perform a brute-force attack on the web login form at php.testinvicti.com. It tries the username admin with multiple passwords from the RockYou wordlist. Hydra submits each password via the login form /login.php, and if the response does not contain Invalid login, Hydra assumes the login was successful.  
  
Screenshot –

