Basic Pentesting - THM

```
Author: Shantanu Kakade
```

The **Basic Pentesting** machine is a beginner-friendly lab that focuses on learning the fundamentals of penetration testing through enumeration and exploitation. In this walkthrough, I am doing the steps used to identify vulnerabilities and gain access to the target in a safe, controlled environment.

Machine Information

Platform — — TryHackMe — https://tryhackme.com/room/basicpentestingjt
Operating System — — — — KALI Linux
Skills Practiced — — — Enumeration, SMB Enumeration, Password Attacks, SSH Access
Tools Used — — — -Nmap, Enum4Linux, Hydra, John the Ripper, SSH

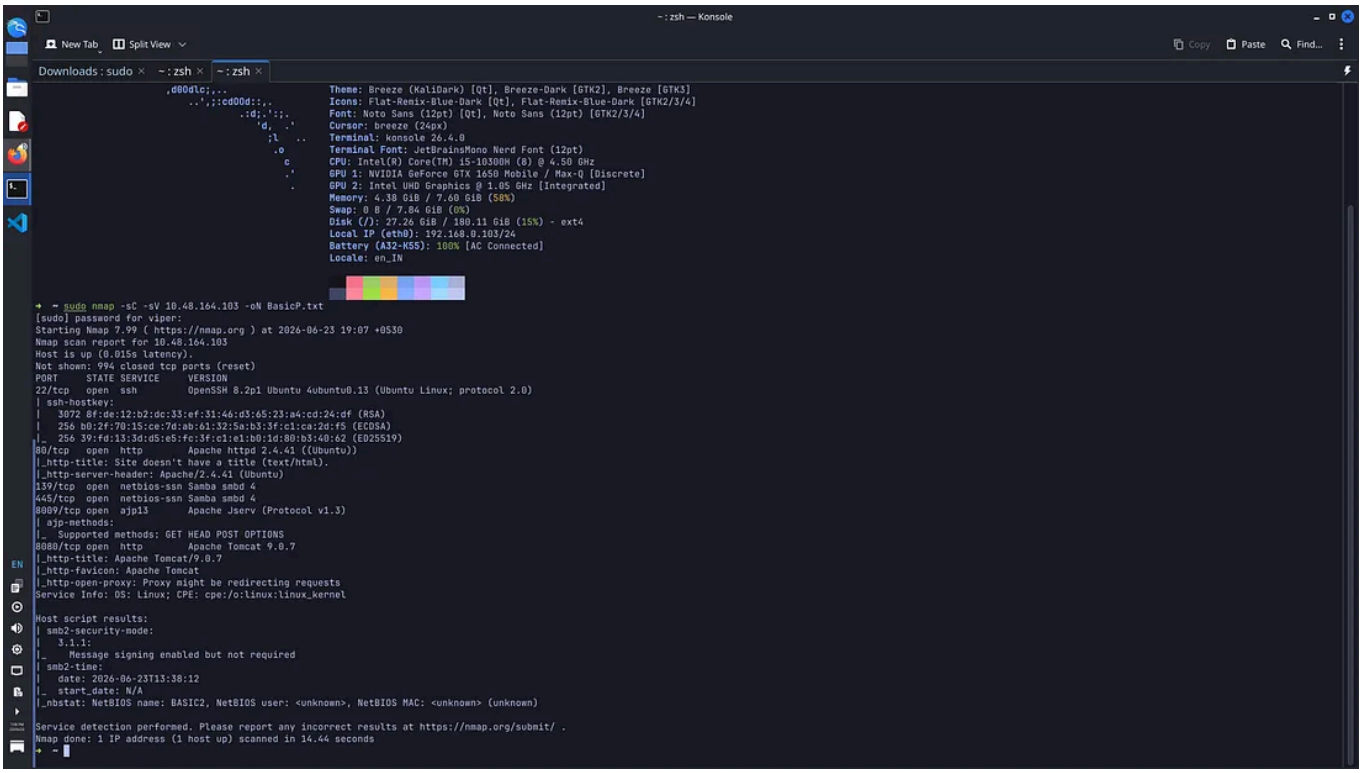
Tools Used

Nmap
Enum4Linux
Hydra
John the Ripper
SSH

Enumeration

I started by running an Nmap scan to find open ports and services on the target machine. The scan found SSH, HTTP, SMB, and Apache Tomcat. Since SMB often provides useful information, I decided to check it first.

```
sudo nmap -sC -sV 10.48.164.103 -oN BasicP.txt
```



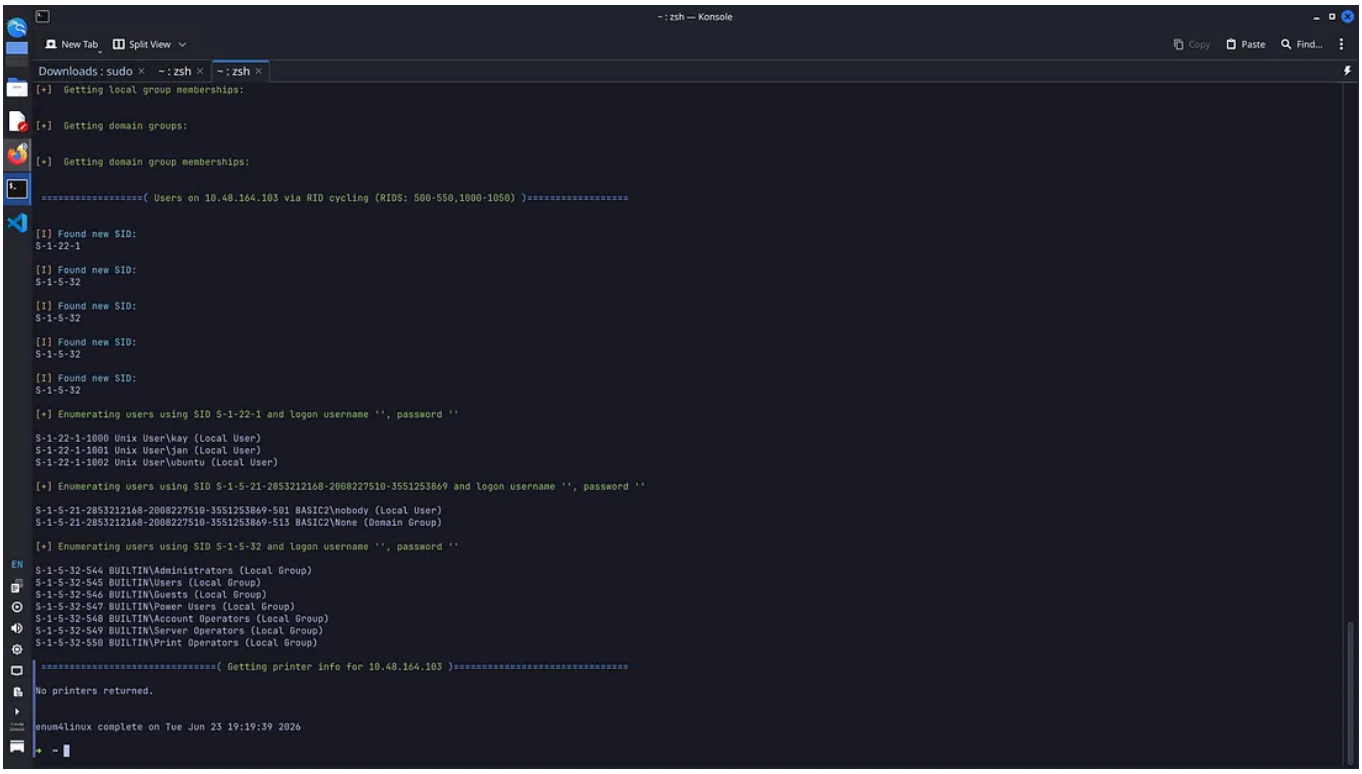

```
22 (SSH): OpenSSH 8.2p1
80 (HTTP): Apache 2.4.41
139/445 (SMB): Samba file sharing service
8009 (AJP13): Apache JServ Protocol
8080 (HTTP): Apache Tomcat 9.0.7
```

From the scan results, SMB looked like the most interesting service because it often reveals usernames, shared folders, and other useful information. So, I moved on to SMB enumeration.

SMB Enumeration

Since SMB was running on the target, I decided to enumerate it to look for shared folders, usernames, and other useful information that could help gain access to the machine.

```
enum4linux -a 10.48.164.103
```



The scan successfully listed several local user accounts on the target machine. Among them, jan and kay stood out as valid usernames that could be used in the next phase of the assessment.

Discovered Users

```
jan
kay
ubuntu
```

Since I now had valid usernames, the next step was to look for weak passwords that could allow access to the system.

Password Brute Force

After finding the usernames through SMB enumeration, I used Hydra to test a wordlist against the SSH service.

The scan successfully listed several local user accounts on the target machine. Among them, jan and kay stood out as valid usernames that could be used in the next phase of the assessment.

Discovered Users

jan
kay

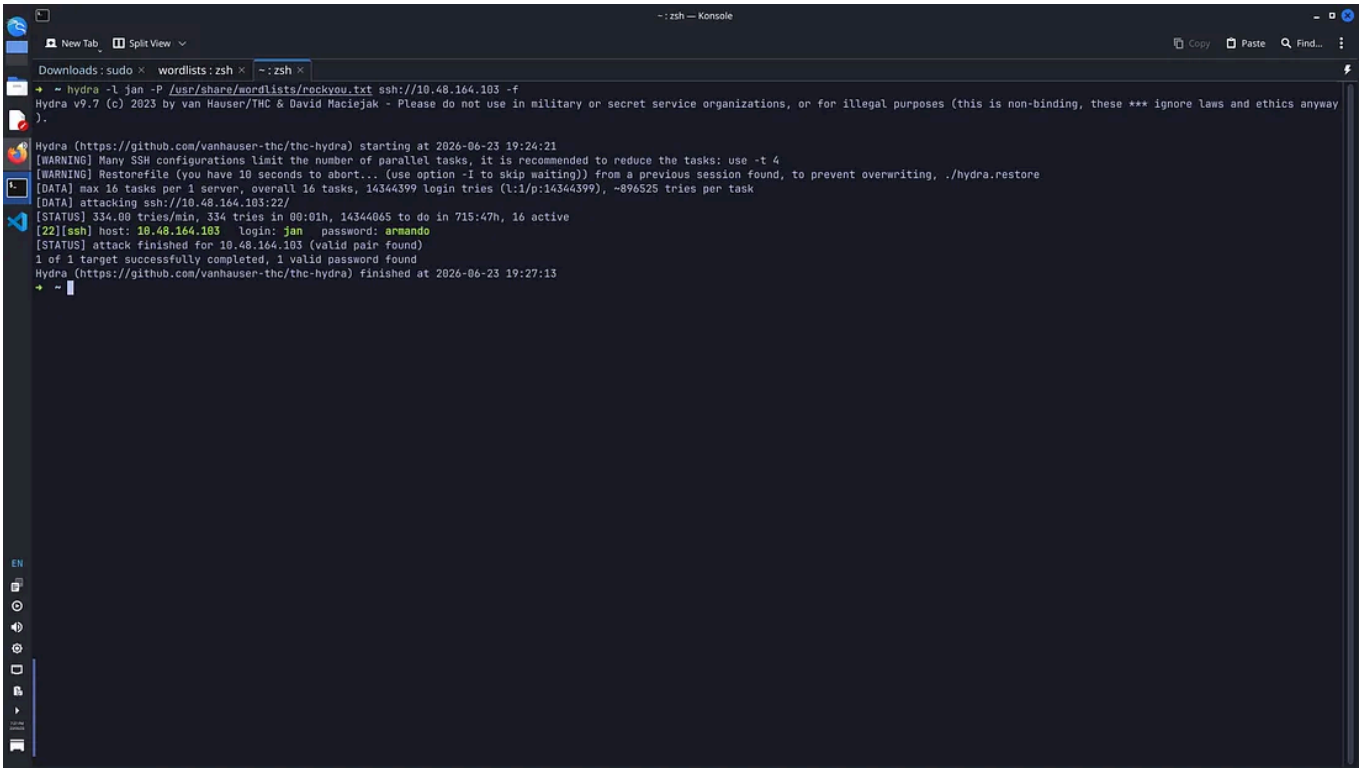
ubuntu

Since I now had valid usernames, the next step was to look for weak passwords that could allow access to the system.

Password Brute Force

After finding the usernames through SMB enumeration, I used Hydra to test a wordlist against the SSH service.

```
hydra -l jan -P /usr/share/wordlists/rockyou.txt ssh://10.48.164.103 -f
```



Hydra successfully found the SSH credentials for the jan user.

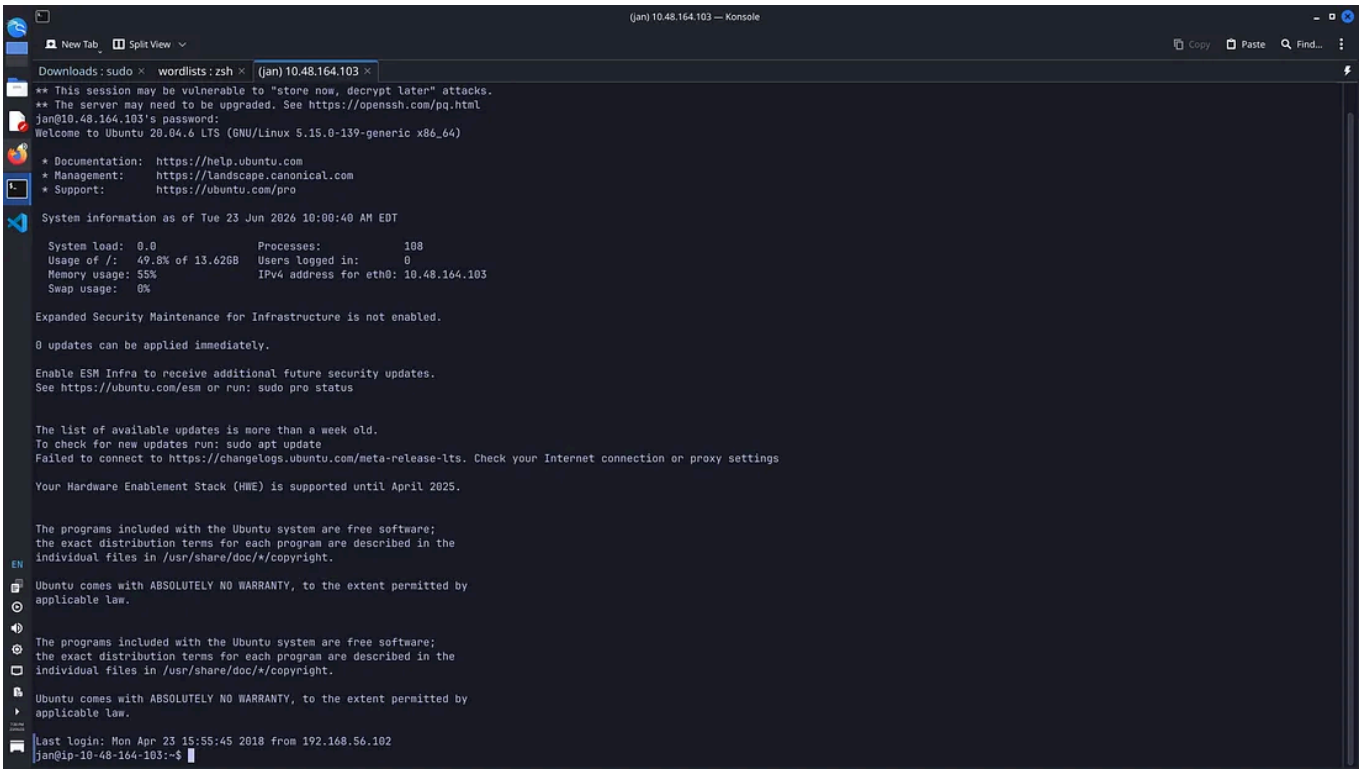
Username: jan
Password: armando

With these credentials, I was able to access the target machine using SSH.

SSH Login

Using the credentials found by Hydra I connected to the target machine through SSH.

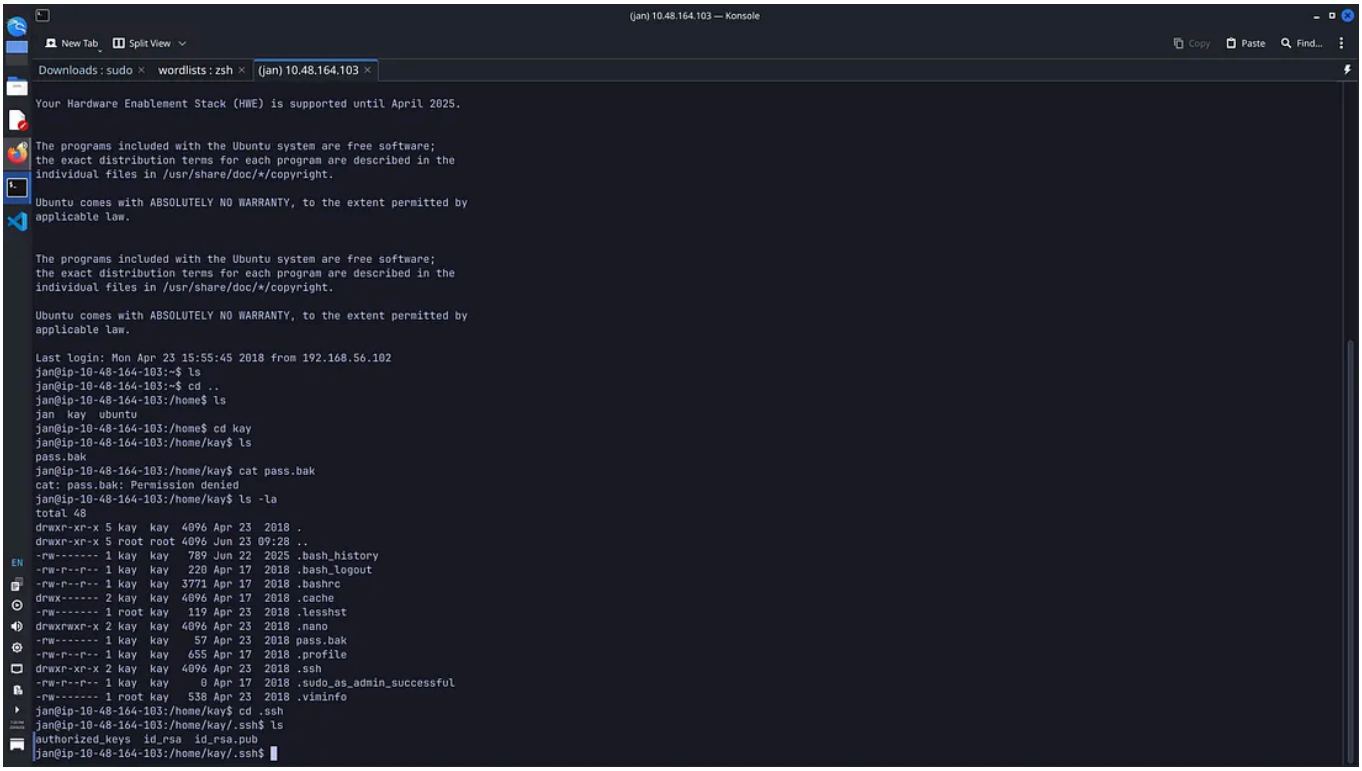
```
ssh jan@10.48.164.103
```



After logging in, I started exploring the system to find useful files and continue the enumeration.

Exploring the System

After logging in as jan, I explored the system to look for other users and any files that might contain useful information.

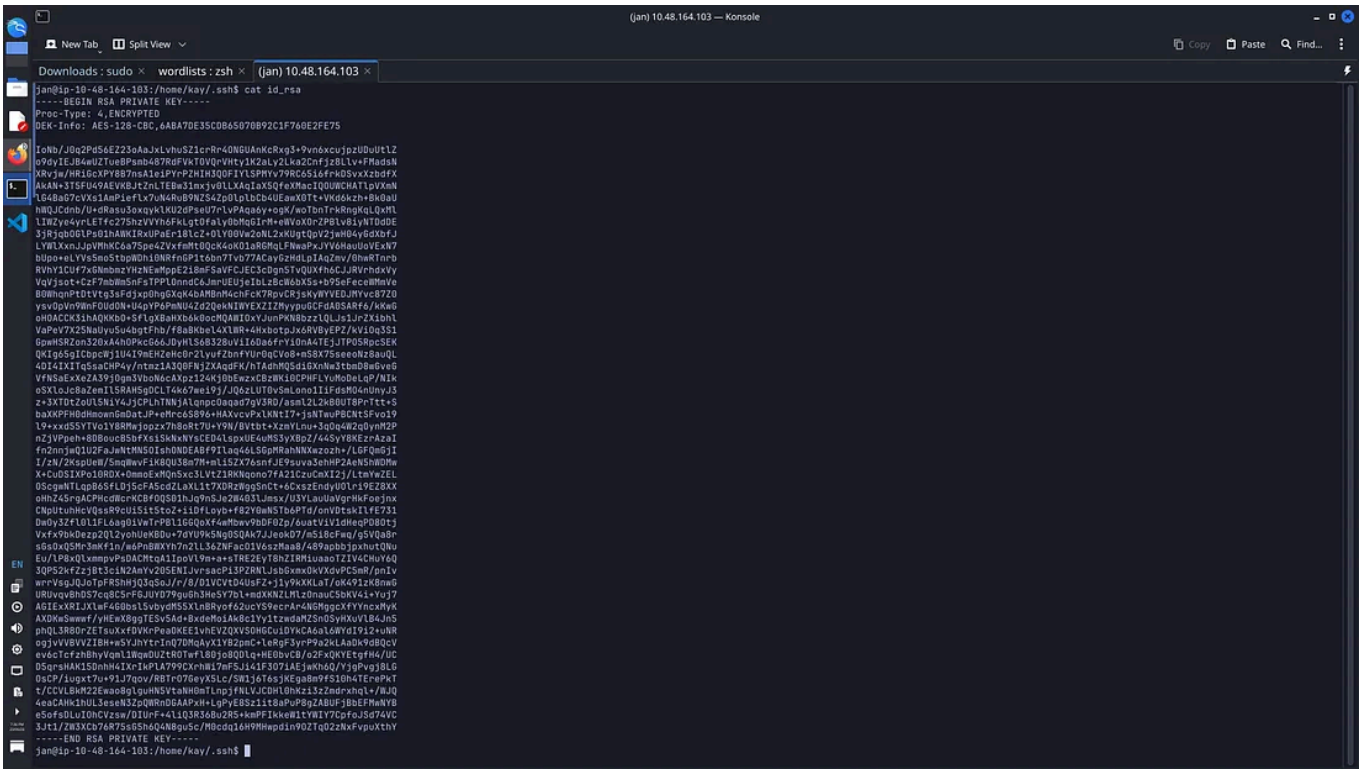


```
cd /home
ls
cd /home/kay
ls -la
```

While checking kay's home directory, I found a file named pass.bak. However, I did not have permission to read it.I also found an .ssh directory containing the following files:

```
authorized_keys
id_rsa
id_rsa.pub
```

The id_rsa file looked interesting because SSH private keys can sometimes be used to log in as another user.



The file contained an encrypted SSH private key belonging to the kay user.

This was a valuable finding because an SSH private key can be used to log in as another user. Since the key was encrypted with a passphrase, I needed to crack the passphrase before I could use it.

Saving the SSH Private Key

Since the private key was displayed on the target machine, I copied its contents and saved it as a new file on my Kali machine. This allowed me to work with the key locally.

```
nano id_rsa
```

I pasted the contents of the private key into the file, saved it, and exited the editor.After saving the key, I changed its permissions so that SSH would accept it.

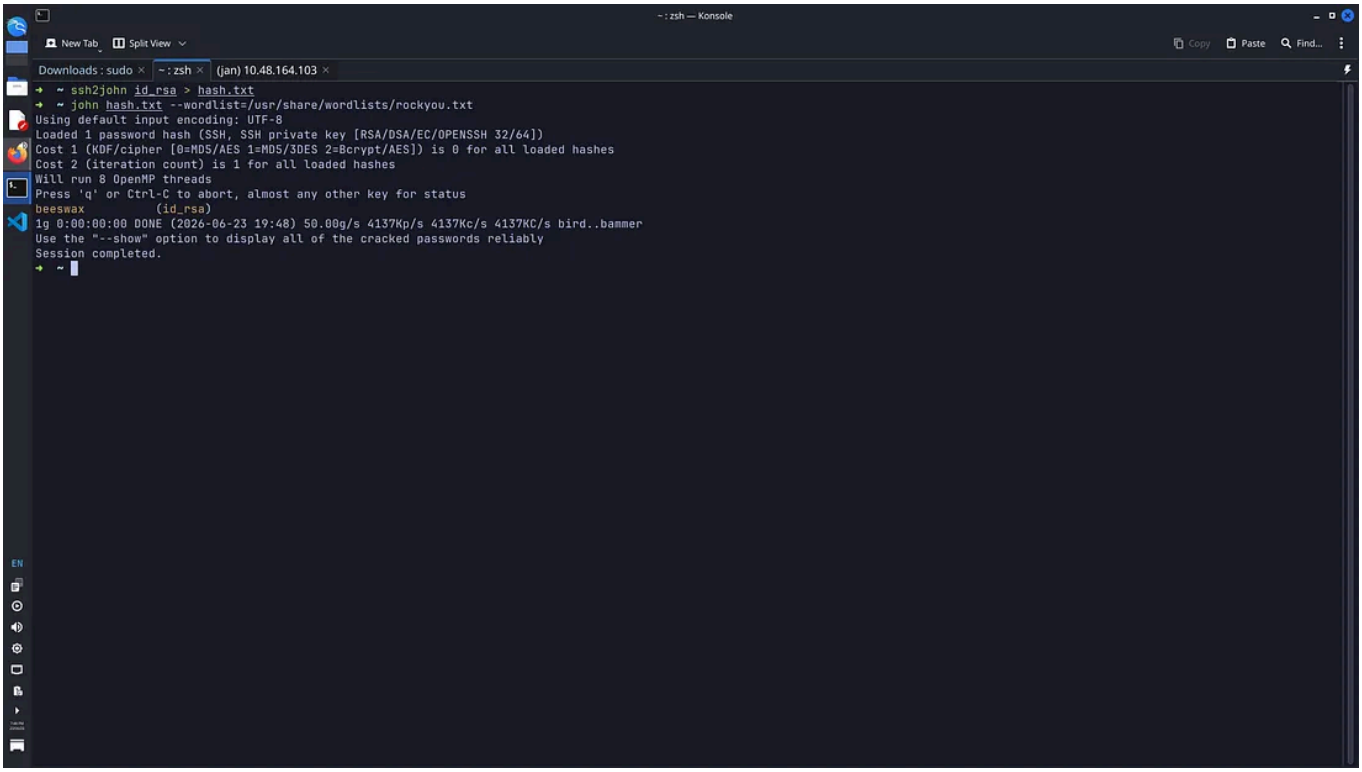
```
chmod 600 id_rsa
```

Cracking the SSH Key Passphrase

After saving the SSH private key on my Kali machine, I needed to crack its passphrase before I could use it. I first converted the key into a format that John the Ripper could understand, then used the rockyou.txt wordlist to crack the passphrase.

```
ssh2john id_rsa > hash.txt
```

```
john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt
```



John the Ripper successfully recovered the passphrase for the SSH private key.

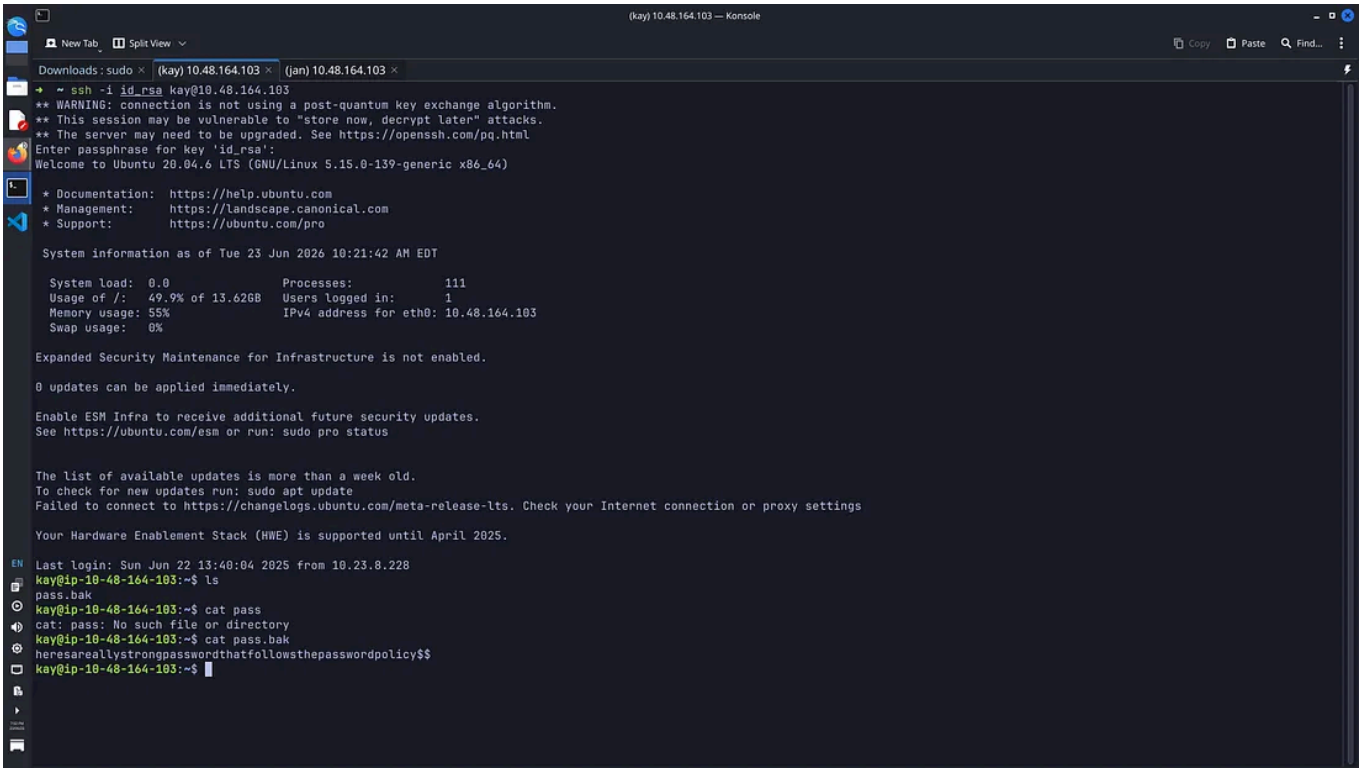
```
Passphrase: beeswax
```

With the passphrase I could use the private key to log in as the kay user

Login as Kay

Using the recovered passphrase and the SSH private key, I logged in as the kay user.

```
ssh -i id_rsa kay@10.48.164.103
```



The login was successful, and I gained access to the system as the kay user.After checking Kay's home directory, I found the pass.bak file and displayed its contents.

Conclusion

In this walkthrough, I started by scanning the target with Nmap to identify open services. Using SMB enumeration, I discovered valid usernames and then used Hydra to recover the SSH password for the jan user. After gaining SSH access, I found kay's encrypted SSH private key, cracked its passphrase with John the Ripper, and logged in as kay to retrieve the final password from pass.bak.