+🚀 Step 1: Setup (Creating Users & Assigning Incorrect Permissions)

1️⃣ Create users (`user1` & `user2`)

Explanation:

- `sudo useradd -m -s /bin/bash user1`
  - `sudo` → Runs the command with root privileges.
  - `useradd` → Creates a new user.
  - `-m` → Creates a home directory for the user.
  - `-s /bin/bash` → Sets the default shell for the user to Bash.
  - `user1` → The name of the new user.
- `sudo passwd user1` → Sets a password for `user1`. The system will prompt for a new password.
- The same process is repeated for `user2`.

2️⃣ Give full access to `/etc/shadow` (Misconfiguration)

Explanation:

- `chmod` → Changes file permissions.
- `777` → Gives full read (`r`), write (`w`), and execute (`x`) permissions to **everyone** (owner, group, and others).
- `/etc/shadow` → This file stores hashed passwords for system accounts. Making it world-readable is a major security risk.

3️⃣ Verify incorrect permissions

Explanation:

`ls -l` → Lists files in long format, displaying permissions, owner, group, and other details.

The output should show `-rwxrwxrwx`, meaning all users have full access to `/etc/shadow`.

🚀 Step 2: Exploitation (Accessing Sensitive System Files)

4️⃣ Switch to `user1` (Low-privileged user)

Explanation:

- `su - user1` → Switches to `user1` and loads its environment.
- Normally, `user1` should not have access to sensitive system files, but due to the misconfiguration, it can now exploit the vulnerability.

5️⃣ Try reading `/etc/shadow` (Exploit)

```
—$ su - user1
Password:
┌──(user1㉿kali)-[~]
└─$ cat /etc/shadow
root:*:20057:0:99999:7:::
daemon:*:20057:0:99999:7:::
bin:*:20057:0:99999:7:::
sys:*:20057:0:99999:7:::
sync:*:20057:0:99999:7:::
games:*:20057:0:99999:7:::
man:*:20057:0:99999:7:::
lp:*:20057:0:99999:7:::
mail:*:20057:0:99999:7:::
news:*:20057:0:99999:7:::
uucp:*:20057:0:99999:7:::
proxy:*:20057:0:99999:7:::
www-data:*:20057:0:99999:7:::
backup:*:20057:0:99999:7:::
list:*:20057:0:99999:7:::
irc:*:20057:0:99999:7:::
_apt:*:20057:0:99999:7:::
nobody:*:20057:0:99999:7:::
systemd-network:!*:20057::::::
dhcpcd:!:20057::::::
systemd-timesync:!*:20057::::::
messagebus:!:20057::::::
tss:!:20057::::::
strongswan:!:20057::::::
tcpdump:!:20057::::::
sshd:!:20057::::::
dnsmasq:!:20057::::::
avahi:!:20057::::::
nm-openvpn:!:20057::::::
speech-dispatcher:!:20057::::::
usbmux:!:20057::::::
pulse:!:20057::::::
nm-openconnect:!:20057::::::
lightdm:!:20057::::::
saned:!:20057::::::
polkitd:!*:20057::::::
rtkit:!:20057::::::
colord:!:20057::::::
_galera:!:20057::::::
mysql:!:20057::::::
stunnel4:!*:20057::::::
_rpc:!:20057::::::
geoclue:!:20057::::::
Debian-snmp:!:20057::::::
sslh:!:20057::::::
ntpsec:!:20057::::::
cups-pk-helper:!:20057::::::
redsocks:!:20057::::::
_gophish:!:20057::::::
iodine:!:20057::::::
miredo:!:20057::::::
```

```
_gvm:!:20057::::::
kali:$y$j9T$ufXTBpN1QpgwlgqRFmb/B0$/.y0ybAF4iNQXniErsDWf9QSl2HZH7LnBeRHB4ZiQa9:20057:0:99999:7:::
user2:$y$j9T$7iKGysBFevMloFzK0aIBA1$dScpsdOEPHyVq5h1EEIiKGs27d6iydZAw4sJ6.rME11:20164:0:99999:7:::
user1:$y$j9T$oqyXxnZZlMqO5HptaRWNg/$E6g16H5vT7uxe1KjCJWHIibnhyvJn9VLDJlmEjj3p88:20164:0:99999:7:::
```

Explanation:

- `cat /etc/shadow` → Attempts to display the contents of the shadow file.
- If permissions are misconfigured (`777`), `user1` will successfully read hashed passwords, which is a major security issue.

🚀 Step 3: Mitigation (Fixing the Security Issue)

6️⃣ Exit `user1` and switch back to Kali user

Explanation:

- `exit` → Logs out of `user1` and returns to the previous user session.

7️⃣ Restore correct file permissions

Explanation:

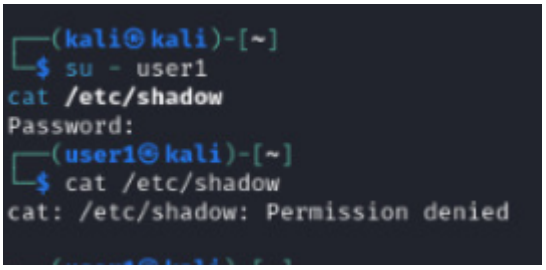
- `chmod 640 /etc/shadow`
  - `640` → Sets permissions to **read & write (rw-) for root**, **read-only (r--) for the shadow group**, and **no access (---) for others**.
  - This prevents unauthorized users from reading or modifying the file.
- `chown root:shadow /etc/shadow`
  - `chown` → Changes file ownership.
  - `root:shadow` → Sets the owner as `root` and the group as `shadow`.
  - This ensures that only root and the shadow group can access the file.

8️⃣ Verify permissions are fixed

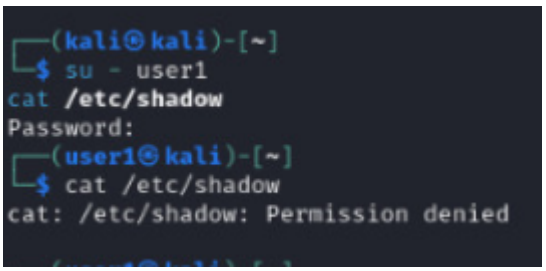
Explanation:

- Checks if `/etc/shadow` now has the correct permissions (`-rw-r-----`).

9️⃣ Ensure `user1` is now denied access



Expected Output:



Explanation:

- Since permissions were fixed, `user1` is now correctly denied access.