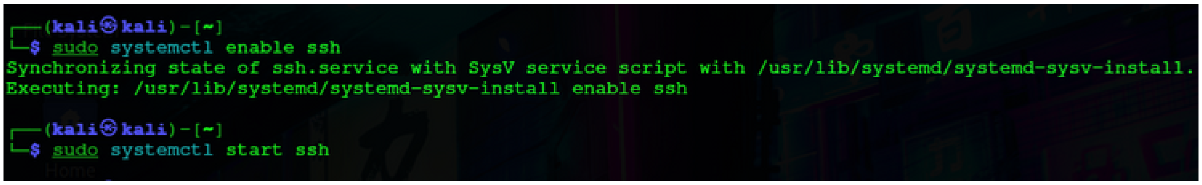
Task 2 :

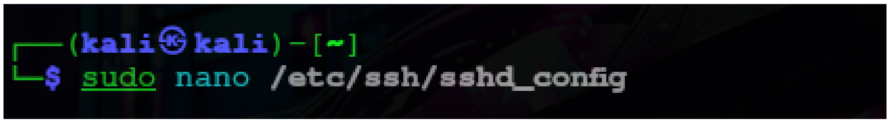
Remote Access & SSH Hardening

Setup: Enabling SSH & Weak Configuration 🔑 :



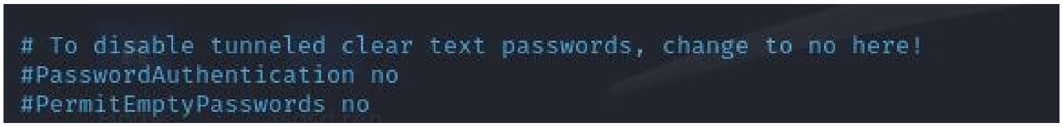
```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable ssh

┌──(kali㉿kali)-[~]
└─$ sudo systemctl start ssh
```

≣≣ To initiate the SSH service, we first enable it using `sudo systemctl enable ssh`, followed by `sudo systemctl start ssh` to ensure it is running and ready for remote access.

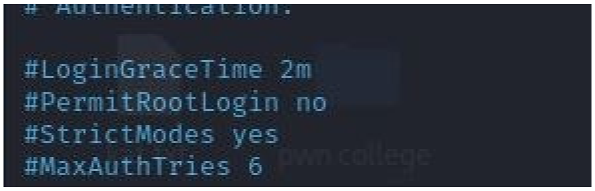


```
┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/ssh/sshd_config
```

≣≣ Next, we modify the SSH configuration to permit root login and enable password authentication by editing the `/etc/ssh/sshd_config` file.

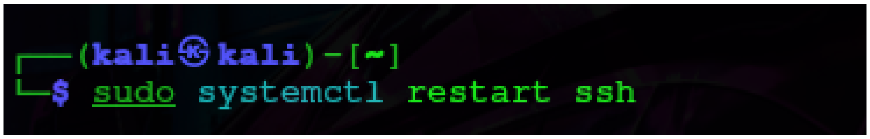


```
# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication no
#PermitEmptyPasswords no
```



```
# Authentication:

#LoginGraceTime 2m
#PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
```

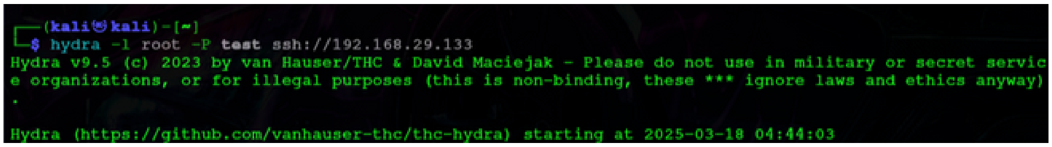
3. Update the `Per mitRootLogin` and `PasswordAuthentication` parameters to `yes.`



```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl restart ssh
```

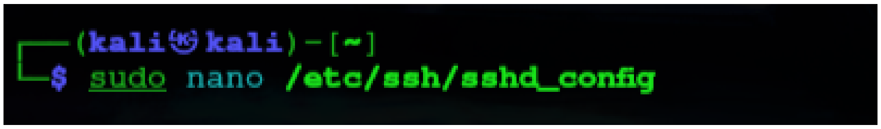
☰☰ Then we restart the ssh service.

Exploitation: Brute-Forcing SSH🛠️:



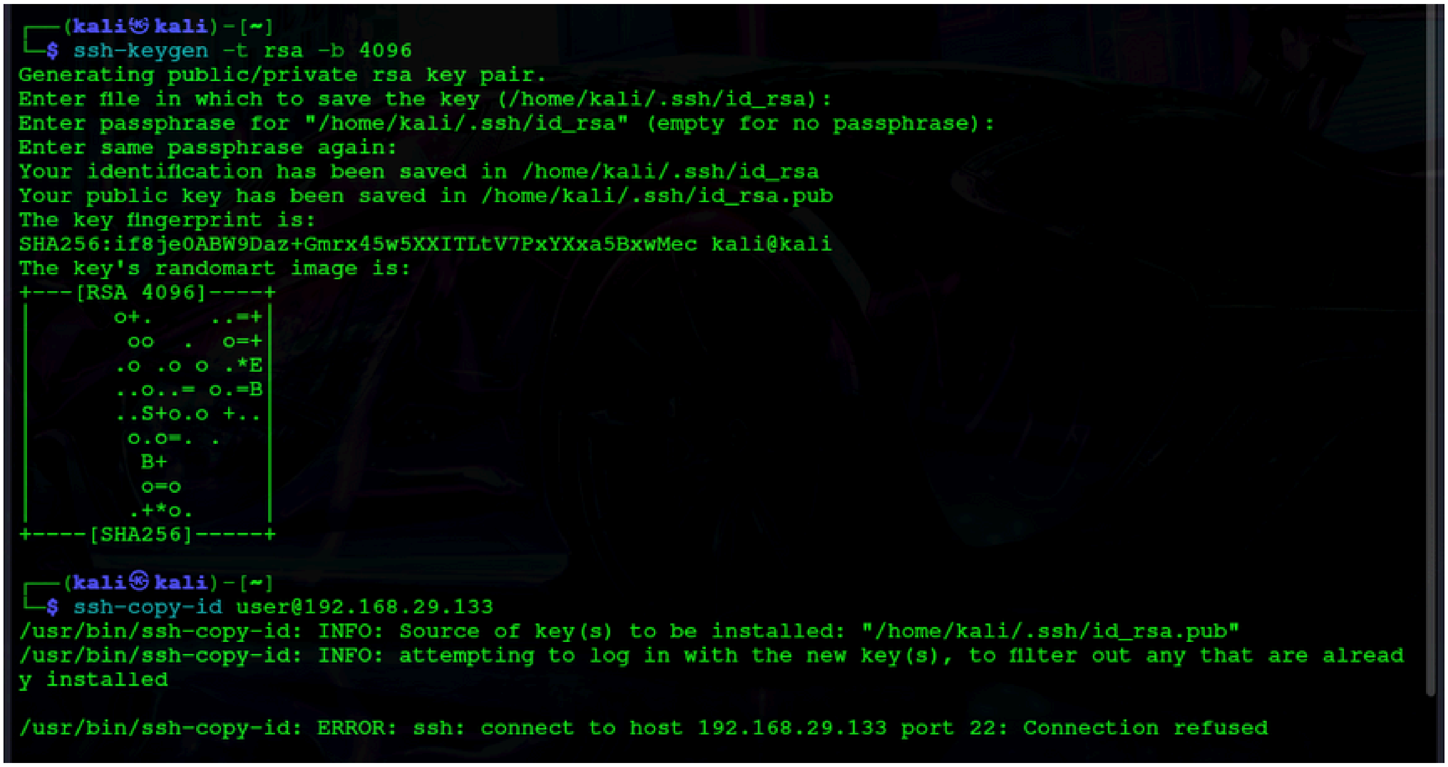
```
┌──(kali㉿kali)-[~]
└─$ hydra -l root -P test ssh://192.168.29.133
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret servic
e organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway)
.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-18 04:44:03
```

☰☰ We use **Hydra** to perform a brute-force SSH root login using a custom-generated wordlist, targeting our own machine's IP address. This allows us to test authentication security and assess password strength.



```
┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/ssh/sshd_config
```

☰☰ To enhance security, root login and password authentication are disabled by setting `PermitRootLogin no` and `PasswordAuthentication no` in the SSH configuration file, followed by restarting the SSH service to apply the changes.

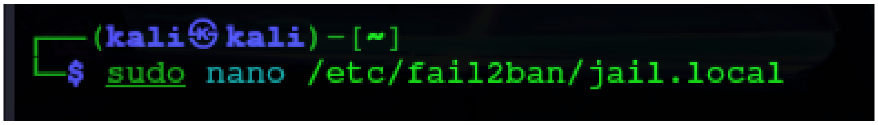


```
┌──(kali㉿kali)-[~]
└─$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/kali/.ssh/id_rsa):
Enter passphrase for "/home/kali/.ssh/id_rsa" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/kali/.ssh/id_rsa
Your public key has been saved in /home/kali/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:if8je0ABW9Daz+Gmrx45w5XXITLtV7PxYXxa5BxwMec kali@kali
The key's randomart image is:
+---[RSA 4096]----+
|      o+.    ..=+|
|       oo  .  o=+|
|      .o .o o .*E|
|      ..o..= o.=B|
|      ..S+o.o +..|
|       o.o=. .   |
|        B+       |
|        o=o      |
|       .+*o.     |
+----[SHA256]-----+

┌──(kali㉿kali)-[~]
└─$ ssh-copy-id user@192.168.29.133
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/kali/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are alread
y installed

/usr/bin/ssh-copy-id: ERROR: ssh: connect to host 192.168.29.133 port 22: Connection refused
```

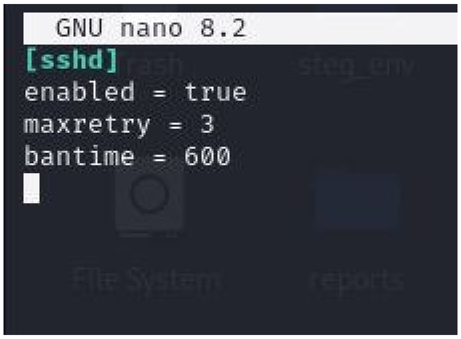
4 . To enhance authentication security, generate an SSH key pair on the client machine using ssh-keygen -t rsa -b 4096. Next, copy the public key to the server with ssh-copy-id user@<server-IP≣, and finally, restart the SSH service using sudo systemctl restart ssh to apply the changes.

Configure Fail2Ban to Prevent Brute-Force Attacks

≣≣ To enhance system security, install **Fail2Ban** by running sudo apt install fail2ban -y, which helps protect against brute-force attacks by monitoring and blocking suspicious login attempts.

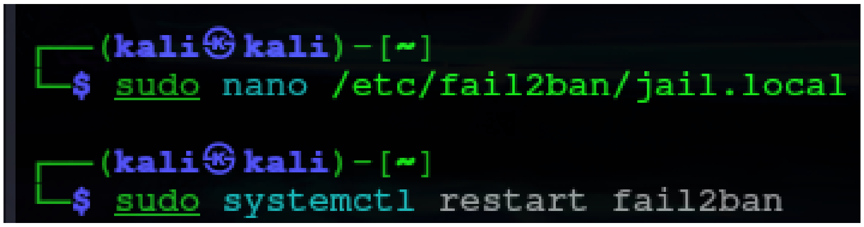


```
┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/fail2ban/jail.local
```



```
  GNU nano 8.2
[sshd]
enabled = true
maxretry = 3
bantime = 600
```

≣≣ To configure **Fail2Ban**, edit the jail configuration file using sudo nano /etc/fail2ban/jail.local, then add the following settings under [sshd]: enabled = true, maxretry ≣ 3, and bantime ≣ 600, ensuring protection against repeated failed SSH login attempts.



```
┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/fail2ban/jail.local

┌──(kali㉿kali)-[~]
└─$ sudo systemctl restart fail2ban
```

≣≣ Finally restart fail2ban to avoid ssh attacks.