Proof of Concept (PoC) Report

Task 2: Securing SSH Access & Mitigating Brute-Force Attacks

1. Overview

This Proof of Concept (PoC) highlights security vulnerabilities arising from improperly configured SSH settings, including root access and password-based authentication. The objective is to demonstrate the risks of such configurations through a brute-force attack and implement security enhancements to mitigate them.

2. Key Steps

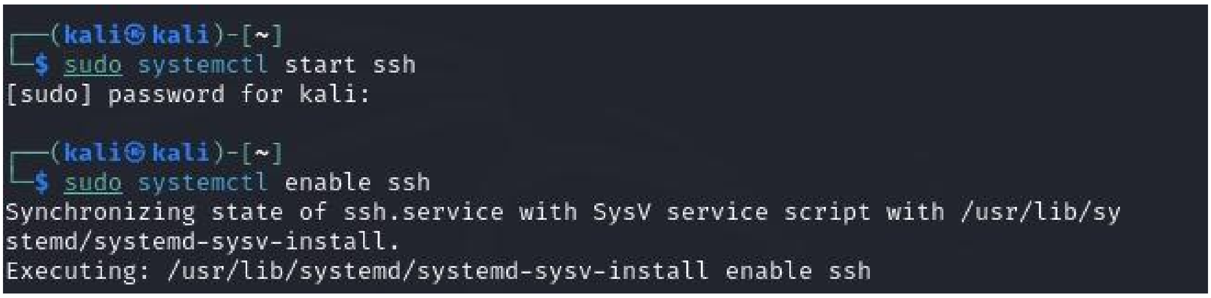
Configuration:

- Enable SSH and configure it to permit root login and password authentication.
  Exploitation:
- Execute a brute-force attack using tools like Hydra to exploit weak SSH settings.
  Mitigation:
- Restrict root login, enforce key-based authentication, and implement fail2ban to prevent brute-force attacks.

3. Environment Setup

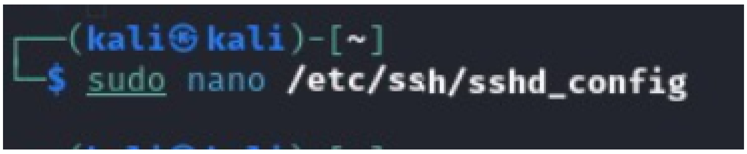
3.1 Activating SSH

SSH was activated and set to initiate at system startup with the following commands:



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

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

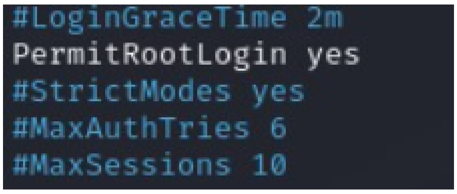
3.2 Enabling Root Login & Password-Based Authentication



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

The SSH settings in were modified to permit root login and authentication via passwords.

Configuration Modifications:



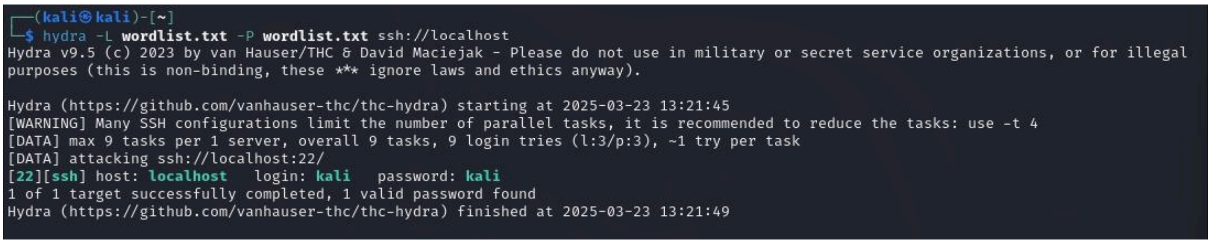
```
#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

4. Exploitation Phase

4.1 Conducting a Brute-Force Attack

A brute-force attack was executed using Hydra to exploit the weak authentication setup

Command Used:



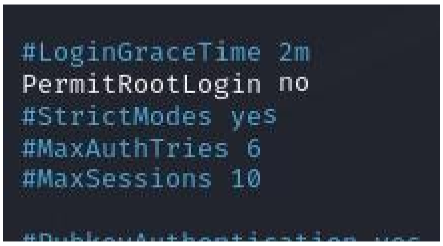
```
┌──(kali㉿kali)-[~]
└─$ hydra -L wordlist.txt -P wordlist.txt ssh://localhost
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal
purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-23 13:21:45
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 9 tasks per 1 server, overall 9 tasks, 9 login tries (l:3/p:3), ~1 try per task
[DATA] attacking ssh://localhost:22/
[22][ssh] host: localhost   login: kali   password: kali
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-03-23 13:21:49
```

5. Security Enhancements

5.1 Restricting Root Login & Password Authentication

The SSH configuration was adjusted to disable root login and enforce key-based authentication.

Configuration Changes:

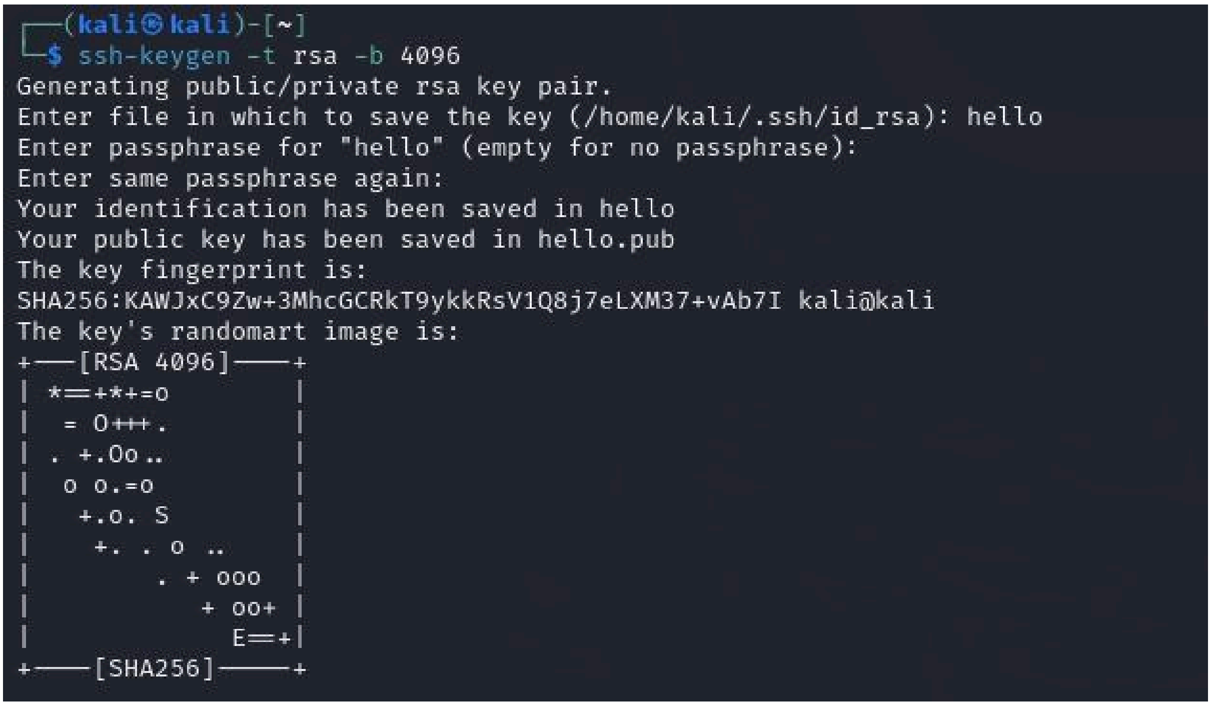


```
#LoginGraceTime 2m
PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes
```

5.2 Enabling Key-Based Authentication

SSH key-based authentication was set up by generating a key pair and adding the public key to the authorized keys file.

Commands Used:

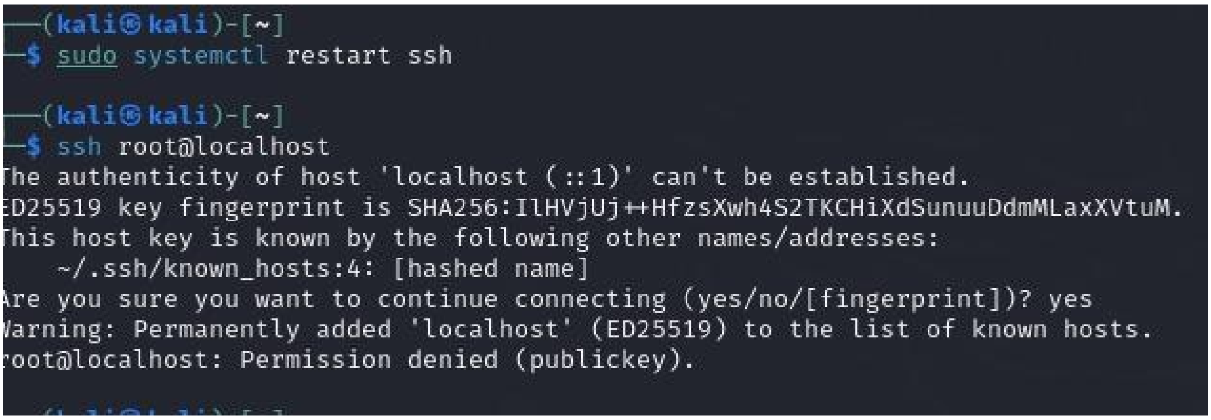


```
┌──(kali㉿kali)-[~]
└─$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/kali/.ssh/id_rsa): hello
Enter passphrase for "hello" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in hello
Your public key has been saved in hello.pub
The key fingerprint is:
SHA256:KAWJxC9Zw+3MhcGCRkT9ykkRsV1Q8j7eLXM37+vAb7I kali@kali
The key's randomart image is:
+---[RSA 4096]----+
| *==+*+=o        |
|  = O+++.        |
| . +.Oo..        |
|  o o.=o         |
|   +.o. S        |
|    +. . o ..    |
|        . + ooo  |
|           + oo+ |
|             E==+|
+----[SHA256]-----+
```

5.3 Validation of Security Measures

Attempts to log in as root or use password authentication were blocked following the implementation of security measures.

Verification Command:



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

┌──(kali㉿kali)-[~]
└─$ ssh root@localhost
The authenticity of host 'localhost (::1)' can't be established.
ED25519 key fingerprint is SHA256:IlHVjUj++HfzsXwh4S2TKCHiXdSunuuDdmMLaxXVtuM.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:4: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.
root@localhost: Permission denied (publickey).
```

6. Summary

This PoC effectively demonstrated how weak SSH configurations can be exploited and the importance of hardening SSH settings. By disabling root login, enforcing key-based authentication, and deploying measures against brute-force attacks, the security posture of SSH was significantly improved.

.