PENETRATION TESTING REPORT

Client: 4Geeks Academy

Project: Penetration Test & Vulnerability Assessment

Conducted by: Mohannad Darwish

Date: 14 July, 2025

Executive Summary

This report documents the findings and remediations of a penetration test conducted on a Debian-based virtual machine provided as part of a cybersecurity training program. The assessment included vulnerability scans, manual inspections, and exploitation techniques to uncover both exploited and potential system vulnerabilities.

Two vulnerabilities were identified as exploited by an attacker (confirmed via logs and forensic data), and several others were found that presented serious risks. All findings are documented here alongside recommendations and evidence of remediation.

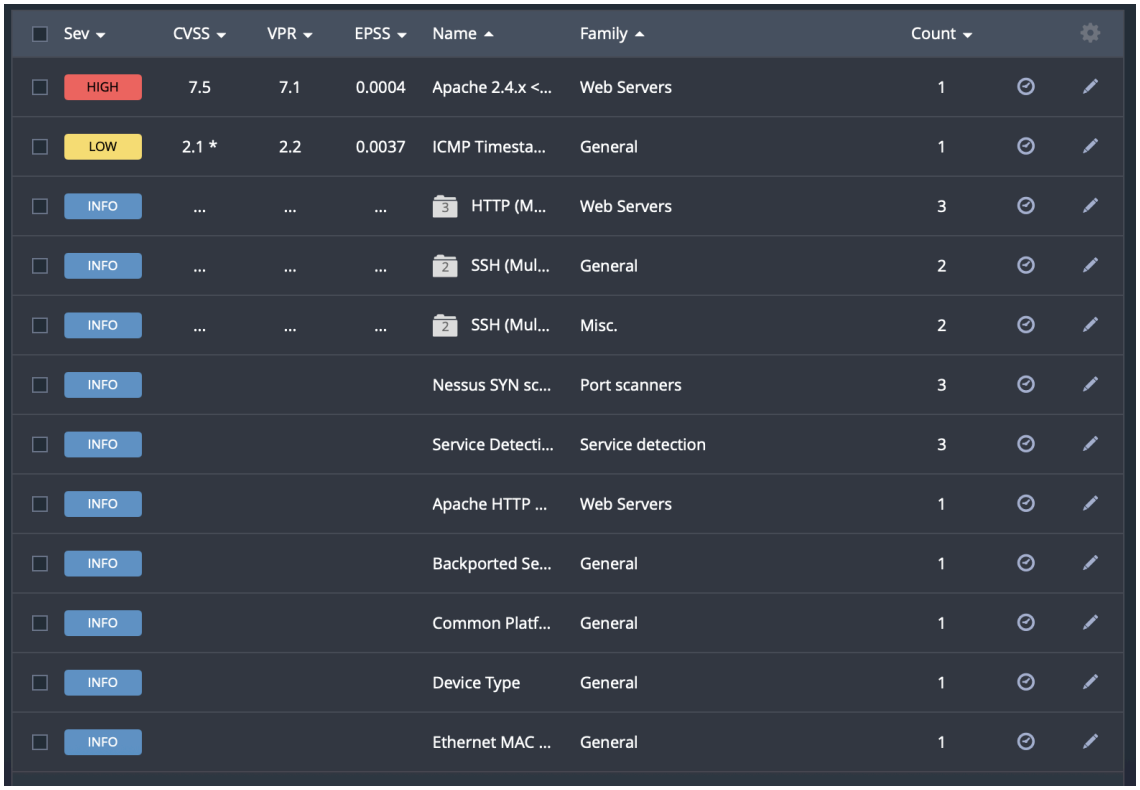
Methodology

The test was performed in accordance with industry best practices and standards such as OWASP, NIST SP 800-115, and ISO/IEC 27001. The assessment involved:

- **Information Gathering**
- **Vulnerability Scanning**
- **Manual Verification**
- **Forensic Log Analysis**
- **Exploitation (controlled and contained)**
- **Remediation Validation**

Scope

- **Target IP**: 192.168.64.4/24
- **System**: Debian-based VM with WordPress and MySQL services
- **Tools Used**: Nmap, Nessus, journalctl, WP-CLI, MariaDB CLI, Apache, FTP, SSH



Sev	CVSS	VPR	EPSS	Name	Family	Count
HIGH	7.5	7.1	0.0004	Apache 2.4.x <...	Web Servers	1
LOW	2.1 *	2.2	0.0037	ICMP Timesta...	General	1
INFO	...	...	...	HTTP (M...	Web Servers	3
INFO	...	...	...	SSH (Mul...	General	2
INFO	...	...	...	SSH (Mul...	Misc.	2
INFO				Nessus SYN sc...	Port scanners	3
INFO				Service Detecti...	Service detection	3
INFO				Apache HTTP ...	Web Servers	1
INFO				Backported Se...	General	1
INFO				Common Platf...	General	1
INFO				Device Type	General	1
INFO				Ethernet MAC ...	General	1

INFO				FTP Server Det...	Service detection	1
INFO				mDNS Detecti...	Service detection	1
INFO				Nessus Scan I...	Settings	1
INFO				OpenSSH Dete...	Misc.	1
INFO				OS Fingerprint...	General	1
INFO				OS Identificati...	General	1
INFO				OS Security Pa...	Settings	1
INFO				Patch Report	General	1
INFO				SSH Server Ty...	Service detection	1
INFO				Target Credent...	Settings	1
INFO				TCP/IP Timest...	General	1
INFO				Traceroute Inf...	General	1
INFO				vsftpd Detecti...	FTP	1

Detailed Findings and Recommendations

1: Insecure SSH Configuration

Vulnerability:

- The system allowed root login via password authentication over SSH.
- Apache logs show the attacker successfully gained access as root on **October 8** from IP 192.168.0.134.

```
debian@debian:/$ sudo journalctl _COMM=sshd | grep "Accepted"
Oct 08 17:40:59 debian sshd[1650]: Accepted password for root from 192.168.0.134 port 45623 ssh2
```

Exploit:

Logs clearly show root login occurred using a password from an unfamiliar IP. This confirms an attacker gained access via SSH due to insecure configuration.

Security Issues Identified:

- PermitRootLogin was set to yes
- PasswordAuthentication was enabled
- MaxAuthTries was left at the default value of 6 (too high)

```
debian@debian:/$ grep -E "PermitRootLogin|PasswordAuthentication|MaxAuthTries" /etc/ssh/sshd_config
PermitRootLogin yes
#MaxAuthTries 6
PasswordAuthentication yes
# PasswordAuthentication.  Depending on your PAM configuration,
# the setting of "PermitRootLogin prohibit-password".
# PAM authentication, then enable this but set PasswordAuthentication
```

Actions Taken to Block Exploit:

- Generated SSH key pair on local machine

```
→  ~ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/Users/main/.ssh/id_rsa):
Created directory '/Users/main/.ssh'.
Enter passphrase for "/Users/main/.ssh/id_rsa" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /Users/main/.ssh/id_rsa
Your public key has been saved in /Users/main/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:jU5CNX9SOFQi6QFJI1F9m6O8fCGNDldvBxTuoGW2CaY main@MacBookPro.attlocal.net
The key's randomart image is:
+---[RSA 4096]----+
|    o+=++ooo+.   |
|     ..o++++.    |
|      ..ooB=o    |
|     . o.O=B.    |
|      E.S=+o..   |
|      .+* o o .  |
|       =.o o .   |
|        + .      |
|         .       |
+----[SHA256]-----+
```

- Enabled public key authentication for SSH access

```
→  ~ ssh debian@192.168.64.4
Enter passphrase for key '/Users/main/.ssh/id_rsa':
Linux debian 6.1.0-25-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.106-3 (2024-08-26) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jul 14 02:27:54 2025 from 192.168.64.4
debian@debian:~$ exit
logout
Connection to 192.168.64.4 closed.
→  ~
```

- Disabled PasswordAuthentication
- Disabled PermitRootLogin
- Reduced MaxAuthTries from 6 to 3

```
PermitRootLogin no
MaxAuthTries 3
PasswordAuthentication no
```

Prevention Recommendations:

- Always disable password-based login for root.
- Require SSH keys and passphrases.
- Limit authentication attempts to deter brute-force attacks.

2: Weak User Credentials + Sudo Privileges

Vulnerability:

- The default user debian had an extremely weak password (123456) and full sudo privileges.
- Journal logs show failed attempts by the user not in sudoers, followed by successful root-level commands.

```
debian@debian:~$ sudo -l
[sudo] password for debian:
Matching Defaults entries for debian on debian:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User debian may run the following commands on debian:
    (ALL : ALL) ALL
```

Exploit:

- Initial logs show debian was *not* in the sudoers file.
- A short time later, commands were successfully executed as root.
- This implies that the attacker likely brute-forced debian's weak password, then escalated privileges.

```
Jul 31 16:09:19 debian sudo[1543]:   debian : user NOT in sudoers ; TTY=pts/0 ; PWD=/home/debian ; USER=root ; COMMAND=/usr/bin/systemctl stop speech-dispatcher
Jul 31 16:14:30 debian sudo[1602]: pam_unix(sudo:auth): authentication failure; logname= uid=1000 euid=0 tty=/dev/pts/0 ruser=debian rhost=  user=debian
Jul 31 16:14:37 debian sudo[1602]:   debian : user NOT in sudoers ; TTY=pts/0 ; PWD=/home/debian ; USER=root ; COMMAND=usermood -aG root debian
Jul 31 16:16:44 debian sudo[1657]:   debian : user NOT in sudoers ; TTY=pts/0 ; PWD=/home/debian ; USER=root ; COMMAND=usermood -aG sudo debian
Jul 31 16:19:16 debian sudo[1684]:   debian : user NOT in sudoers ; TTY=pts/0 ; PWD=/home/debian ; USER=root ; COMMAND=/usr/sbin/visudo
Jul 31 16:22:10 debian sudo[1720]:     root : TTY=pts/0 ; PWD=/home/debian ; USER=root ; COMMAND=/usr/sbin/visudo
Jul 31 16:22:10 debian sudo[1720]: pam_unix(sudo:session): session opened for user root(uid=0) by (uid=0)
Jul 31 16:25:03 debian sudo[1720]: pam_unix(sudo:session): session closed for user root
Jul 31 16:26:39 debian sudo[1872]:     root : TTY=pts/0 ; PWD=/home/debian ; USER=root ; COMMAND=/usr/bin/systemctl stop speech-dispatcher
Jul 31 16:26:39 debian sudo[1872]: pam_unix(sudo:session): session opened for user root(uid=0) by (uid=0)
Jul 31 16:26:39 debian sudo[1872]: pam_unix(sudo:session): session closed for user root
Jul 31 16:27:33 debian sudo[1937]:     root : TTY=pts/0 ; PWD=/home/debian ; USER=root ; COMMAND=/usr/bin/systemctl disable speech-dispatcher
Jul 31 16:27:33 debian sudo[1937]: pam_unix(sudo:session): session opened for user root(uid=0) by (uid=0)
Jul 31 16:27:35 debian sudo[1937]: pam_unix(sudo:session): session closed for user root
```

Security Issues Identified:

- Weak user password: 123456
- Full root access available through that account

Actions Taken to Block Exploit:

- Password was updated to a stronger one: 4G.Finaldeb

- Logs were reviewed to assess potential escalation

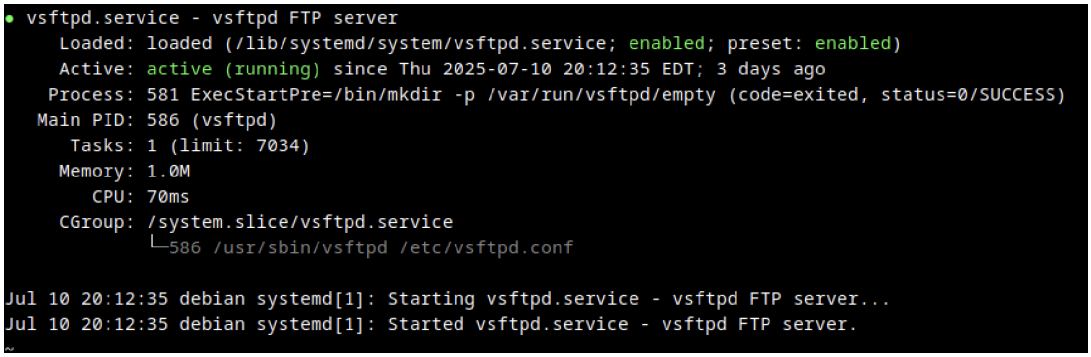
Prevention Recommendations:

- Enforce strong password policies and two-factor authentication.
- Review and restrict sudo privileges regularly.
- Monitor and audit privilege escalation activity.

3: FTP Server Misconfiguration

Vulnerability:

The FTP server was active and misconfigured in multiple ways, including anonymous access, unrestricted write permissions, and lack of filesystem isolation.



Issues Identified and Fixes Applied:

- **Anonymous Access Enabled**: Anyone could connect to the FTP server without credentials.

  ➤ **Fix**: anonymous_enable was disabled in vsftpd.conf.

```
debian@debian:/$ grep anonymous_enable /etc/vsftpd.conf
anonymous_enable=NO
```

- **Global Write Access Enabled**: All users could modify, upload, or delete files.

  ➤ **Fix**: write_enable was disabled to prevent unauthorized changes.

```
debian@debian:/$ grep write_enable /etc/vsftpd.conf
write_enable=NO
```

- **No Chroot Jail**: Local users were not restricted to home directories.

  ➤ **Fix**: chroot_local_user=YES was uncommented and enforced.

```
debian@debian:/$ grep chroot_local_user= /etc/vsftpd.conf
chroot_local_user=YES
```

Prevention Recommendations:

- Never enable anonymous access unless strictly necessary.
- Restrict write permissions and isolate users within their directories.
- Continuously audit FTP settings and disable the service if unused.

4: MySQL Configuration Issues

Vulnerability:

- Multiple user accounts in MySQL were found to have weak passwords, unnecessary access, or no passwords at all.
- The general log was also disabled, making it difficult to audit account activity.

Database Structure Audited:

- Inspect the user table.

```
MariaDB [(none)]> DESC mysql.user;
+-----------------------+---------------------+------+-----+----------+-------+
| Field                 | Type                | Null | Key | Default  | Extra |
+-----------------------+---------------------+------+-----+----------+-------+
| Host                  | char(255)           | NO   |     |          |       |
| User                  | char(128)           | NO   |     |          |       |
| Password              | longtext            | YES  |     | NULL     |       |
| Select_priv           | varchar(1)          | YES  |     | NULL     |       |
| Insert_priv           | varchar(1)          | YES  |     | NULL     |       |
| Update_priv           | varchar(1)          | YES  |     | NULL     |       |
| Delete_priv           | varchar(1)          | YES  |     | NULL     |       |
| Create_priv           | varchar(1)          | YES  |     | NULL     |       |
| Drop_priv             | varchar(1)          | YES  |     | NULL     |       |
| Reload_priv           | varchar(1)          | YES  |     | NULL     |       |
| Shutdown_priv         | varchar(1)          | YES  |     | NULL     |       |
| Process_priv          | varchar(1)          | YES  |     | NULL     |       |
| File_priv             | varchar(1)          | YES  |     | NULL     |       |
| Grant_priv            | varchar(1)          | YES  |     | NULL     |       |
| References_priv       | varchar(1)          | YES  |     | NULL     |       |
| Index_priv            | varchar(1)          | YES  |     | NULL     |       |
| Alter_priv            | varchar(1)          | YES  |     | NULL     |       |
| Show_db_priv          | varchar(1)          | YES  |     | NULL     |       |
| Super_priv            | varchar(1)          | YES  |     | NULL     |       |
| Create_tmp_table_priv | varchar(1)          | YES  |     | NULL     |       |
| Lock_tables_priv      | varchar(1)          | YES  |     | NULL     |       |
| Execute_priv          | varchar(1)          | YES  |     | NULL     |       |
| Repl_slave_priv       | varchar(1)          | YES  |     | NULL     |       |
| Repl_client_priv      | varchar(1)          | YES  |     | NULL     |       |
| Create_view_priv      | varchar(1)          | YES  |     | NULL     |       |
| Show_view_priv        | varchar(1)          | YES  |     | NULL     |       |
| Create_routine_priv   | varchar(1)          | YES  |     | NULL     |       |
| Alter_routine_priv    | varchar(1)          | YES  |     | NULL     |       |
| Create_user_priv      | varchar(1)          | YES  |     | NULL     |       |
| Event_priv            | varchar(1)          | YES  |     | NULL     |       |
| Trigger_priv          | varchar(1)          | YES  |     | NULL     |       |
| Create_tablespace_priv | varchar(1)          | YES  |     | NULL     |       |
| Delete_history_priv   | varchar(1)          | YES  |     | NULL     |       |
| ssl_type              | varchar(9)          | YES  |     | NULL     |       |
| ssl_cipher            | longtext            | NO   |     |          |       |
| x509_issuer           | longtext            | NO   |     |          |       |
| x509_subject          | longtext            | NO   |     |          |       |
| max_questions         | bigint(20) unsigned | NO   |     | 0        |       |
| max_updates           | bigint(20) unsigned | NO   |     | 0        |       |
| max_connections       | bigint(20) unsigned | NO   |     | 0        |       |
| max_user_connections  | bigint(21)          | NO   |     | 0        |       |
| plugin                | longtext            | NO   |     |          |       |
| authentication_string | longtext            | NO   |     |          |       |
| password_expired      | varchar(1)          | NO   |     |          |       |
| is_role               | varchar(1)          | YES  |     | NULL     |       |
| default_role          | longtext            | NO   |     |          |       |
| max_statement_time    | decimal(12,6)       | NO   |     | 0.000000 |       |
+-----------------------+---------------------+------+-----+----------+-------+
47 rows in set (0.078 sec)
```

- Analyze password hash lengths.

```
+---------------+-----------+-------------+
| User          | Host      | pswd_length |
+---------------+-----------+-------------+
| mariadb.sys   | localhost |           0 |
| root          | localhost |          41 |
| mysql         | localhost |           7 |
| wordpressuser | localhost |          41 |
| user          | localhost |          41 |
+---------------+-----------+-------------+
5 rows in set (0.030 sec)
```

Identified Issues and Fixes:

- **No Password Assigned**: User mariadb.sys had no password. The user with no password is “mariadb.sys”. This is a special system user created automatically by MariaDB during installation or upgrade. It’s not a human account, not for application use, and not meant to be logged into manually. It's used internally by MariaDB for things like Performance Schema tasks, Optimization routines, and Background tasks.

  ➤ **Fix**: Instead of assigning a password to the account, the account has been locked. It never needs to log in, and giving it login capability increases risk without benefit.

```
MariaDB [(none)]> ALTER USER 'mariadb.sys'@'localhost' ACCOUNT LOCK;
Query OK, 0 rows affected (0.033 sec)
```

- **Weak Password Detected**: User mysql had a weak hash length of 7.

  ➤ **Fix**: Password updated to a strong one.

```
MariaDB [(none)]> ALTER USER 'mysql'@'localhost' IDENTIFIED BY 'VM@pwd94new!cS';
Query OK, 0 rows affected (0.017 sec)
```

- **Shared Hashes Detected**: root and wordpressuser had identical password hashes.

```
MariaDB [(none)]> SELECT user, authentication_string FROM mysql.user WHERE user IN ('root', 'wo
rdpressuser', 'user');
+---------------+-------------------------------------------+
| User          | authentication_string                     |
+---------------+-------------------------------------------+
| root          | *6BB4837EB74329105EE4568DDA7DC67ED2CA2AD9 |
| wordpressuser | *6BB4837EB74329105EE4568DDA7DC67ED2CA2AD9 |
| user          | *2470C0C06DEE42FD1618BB99005ADCA2EC9D1E19 |
+---------------+-------------------------------------------+
3 rows in set (0.021 sec)
```

  ➤ **Fix**: Each user was given a unique, strong password.

```
MariaDB [(none)]> ALTER USER 'root'@'localhost' IDENTIFIED BY '52@0T0Jv02Q#';
Query OK, 0 rows affected (0.013 sec)

MariaDB [(none)]> ALTER USER 'wordpressuser'@'localhost' IDENTIFIED BY 'wXN0!ea85X<Z';
Query OK, 0 rows affected (0.013 sec)
```

Privilege Review:

- Checked privilege columns: super_priv, grant_priv, create_priv, file_priv - Users with full privileges: root, mysql, and user
- Users with no privileges: mariadb.sys, wordpressuser

```
MariaDB [(none)]> SELECT user, host, Super_priv, Grant_priv, Create_priv, File_priv FROM mysql.
user WHERE user IN ('root', 'mysql', 'wordpressuser', 'user', 'mariadb.sys');
+--------------+-----------+------------+------------+-------------+-----------+
| User         | Host      | Super_priv | Grant_priv | Create_priv | File_priv |
+--------------+-----------+------------+------------+-------------+-----------+
| mariadb.sys  | localhost | N          | N          | N           | N         |
| root         | localhost | Y          | Y          | Y           | Y         |
| mysql        | localhost | Y          | Y          | Y           | Y         |
| wordpressuser | localhost | N          | N          | N           | N         |
| user         | localhost | Y          | Y          | Y           | Y         |
+--------------+-----------+------------+------------+-------------+-----------+
5 rows in set (0.018 sec)
```

- No direct signs of malicious activity from user, but due to elevated privileges, the account has been **locked**.

```
MariaDB [(none)]> ALTER USER 'user'@'localhost' ACCOUNT LOCK;
Query OK, 0 rows affected (0.008 sec)
```

- Enabled the **general log** for future monitoring.

```
MariaDB [(none)]> SHOW VARIABLES LIKE 'general_log';
+--------------+-------+
| Variable_name | Value |
+--------------+-------+
| general_log  | OFF   |
+--------------+-------+
1 row in set (0.006 sec)

MariaDB [(none)]> SET GLOBAL general_log = 'ON';
Query OK, 0 rows affected (0.018 sec)

MariaDB [(none)]> SHOW VARIABLES LIKE 'general_log';
+--------------+-------+
| Variable_name | Value |
+--------------+-------+
| general_log  | ON    |
+--------------+-------+
1 row in set (0.012 sec)
```

Prevention Recommendations: - Enforce minimum password complexity and rotation. - Lock or remove inactive accounts. - Use general logging to track user behavior and access history.

5: WordPress Administrative Account

Vulnerability: The WordPress installation had a single administrator account with predictable login credentials. WordPress users cannot change usernames — so instead, a new admin user was created, and the original account was deleted.

```
debian@debian:~$ sudo find /var/www/ -name wp-config.php 2>/dev/null
/var/www/html/wp-config.php
```

```
debian@debian:/var/www/html$ wp --info
OS:     Linux 6.1.0-25-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.106-3 (2024-08-26) x86_64
Shell:  /bin/bash
PHP binary:     /usr/bin/php8.2
PHP version:    8.2.20
php.ini used:   /etc/php/8.2/cli/php.ini
MySQL binary:   /usr/bin/mariadb
MySQL version:  mariadb  Ver 15.1 Distrib 10.11.6-MariaDB, for debian-linux-gnu (x86_64) using  EditLine
 wrapper
SQL modes:
WP-CLI root dir:        phar://wp-cli.phar/vendor/wp-cli/wp-cli
WP-CLI vendor dir:      phar://wp-cli.phar/vendor
WP_CLI phar path:       phar:///usr/local/bin/wp
WP-CLI packages dir:
WP-CLI cache dir:       /home/debian/.wp-cli/cache
WP-CLI global config:
WP-CLI project config:
WP-CLI version: 2.12.0
```

Steps Taken:

- Listed current accounts via WP-CLI

```
debian@debian:/var/www/html$ wp user list --path=/var/www/html
+----+---------------+---------------+---------------------+----------------------+--------------+
| ID | user_login    | display_name  | user_email          | user_registered      | roles        |
+----+---------------+---------------+---------------------+----------------------+--------------+
| 1  | wordpress-user | wordpress-user | rosinnicuentas@gmail | 2024-09-30 16:23:12  | administrator |
|    |               |               | .com                |                      |              |
+----+---------------+---------------+---------------------+----------------------+--------------+
```

- Created new administrator user with stronger credentials

```
debian@debian:/var/www/html$ wp user create 4GeeksMiami-BigBoss administrator@4geeksfinal.com \
> --role=administrator \
> --user_pass="FinalWP@ultra$" \
> --display_name="Admin Head" \
> --path=/var/www/html
Success: Created user 2.
```

- Deleted original user and reassigned all content
    - Username reassignment failed; content was reassigned by user ID

```
debian@debian:/var/www/html$ wp user list
+----+--------------------+---------------+--------------------+-------------------------+---------------+
| ID | user_login         | display_name  | user_email         | user_registered         | roles         |
+----+--------------------+---------------+--------------------+-------------------------+---------------+
| 2  | 4GeeksMiami-BigBos | Admin Head    | administrator@4gee | 2025-07-13 16:22:37     | administrator |
|    | s                  |               | ksfinal.com        |                         |               |
| 1  | wordpress-user     | wordpress-user | rosinnicuentas@gma | 2024-09-30 16:23:12     | administrator |
|    |                    |               | il.com             |                         |               |
+----+--------------------+---------------+--------------------+-------------------------+---------------+
debian@debian:/var/www/html$ wp user delete wordpress-user --reassign=2
Success: Removed user 1 from http://localhost.
```

```
debian@debian:/var/www/html$ wp user list
+----+--------------------+--------------+--------------------+-------------------------+---------------+
| ID | user_login         | display_name | user_email         | user_registered         | roles         |
+----+--------------------+--------------+--------------------+-------------------------+---------------+
| 2  | 4GeeksMiami-BigBoss | Admin Head   | administrator@4geek | 2025-07-13 16:22:37     | administrator |
|    |                    |              | sfinal.com         |                         |               |
+----+--------------------+--------------+--------------------+-------------------------+---------------+
```

Prevention Recommendations:

- Never rely on predictable usernames like “admin” or “wordpressuser”
- Use WP-CLI to automate secure admin setup
- Regularly rotate admin passwords and enable two-factor authentication

---

6: Insecure Permissions on wp-config.php

Vulnerability: The file wp-config.php contained overly permissive permissions (-rwxrwxrwx), allowing any user on the system to read or modify it. This file contains sensitive data like database credentials.

```
debian@debian:/$ ls -l /var/www/html/wp-config.php
-rwxrwxrwx 1 www-data www-data 3017 Sep 30  2024 /var/www/html/wp-config.php
```

Fixes Applied:

- Set correct ownership (www-data or apache user)

```
debian@debian:/$ sudo chown www-data:www-data /var/www/html/wp-config.php
```

- Changed permissions to 640 to restrict access

```
debian@debian:/$ sudo chmod 640 /var/www/html/wp-config.php
```

- Verified permissions were applied successfully

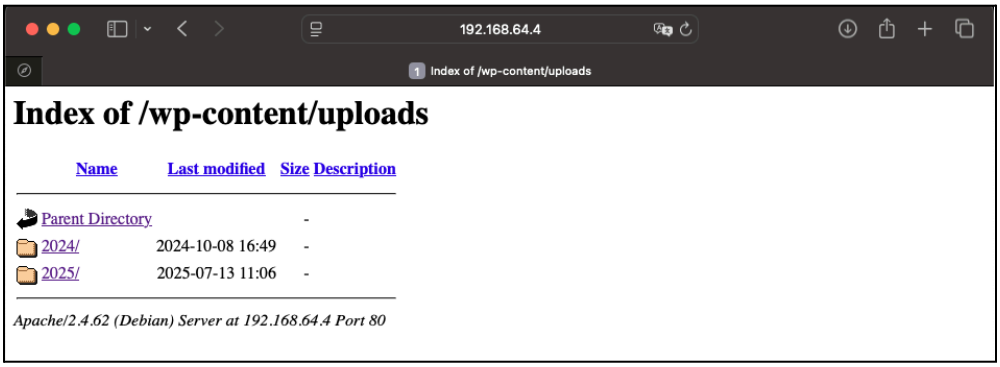
```
debian@debian:/$ ls -l /var/www/html/wp-config.php
-rw-r----- 1 www-data www-data 3017 Sep 30  2024 /var/www/html/wp-config.php
```

Prevention Recommendations:

- Audit file permissions regularly on configuration files.
- Use chmod 640 for wp-config.php and ensure ownership belongs to the web server user.

7: Listable Web Directory

Vulnerability: Apache was configured to allow directory listing. Visiting paths such as /wp-content/uploads/ displayed a file index, exposing internal directory structure.



Fixes Applied:

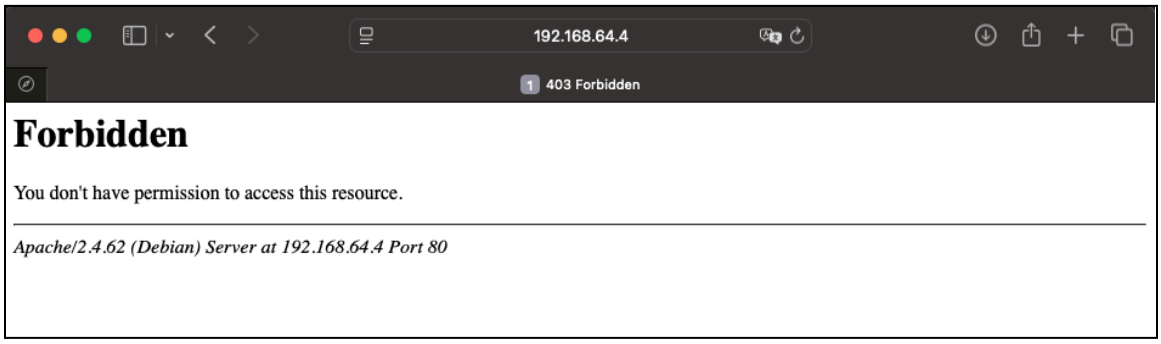
- Identified Options Indexes directive enabled in Apache config

```
debian@debian:/$ sudo grep -Ri "Options Indexes" /etc/apache2
[sudo] password for debian:
/etc/apache2/apache2.conf:      Options Indexes FollowSymLinks
/etc/apache2/apache2.conf:        Options Indexes FollowSymLinks
/etc/apache2/apache2.conf:      Options Indexes FollowSymLinks
/etc/apache2/apache2.conf:#     Options Indexes FollowSymLinks
```

- Disabled directory listing by removing Indexes option

```
<Directory /var/www/>
        Options -Indexes +FollowSymLinks
        AllowOverride None
        Require all granted
</Directory>
```

- Verified fix worked by visiting same URL and getting a Forbidden error



Prevention Recommendations:

- Always disable directory listing unless explicitly needed.
- Use .htaccess or Apache config files to enforce security policies.

8: Unnecessary Open Ports

Vulnerability: A port scan using nmap and Nessus revealed unnecessary services running and accessible.

Results: -

- Port 21 (FTP)
- Port 22 (SSH)
- Port 80 (HTTP)
- Port 3306 (MySQL) — confirmed *local only* due to bind-address = 127.0.0.1

```
debian@debian:/$ sudo grep bind-address /etc/mysql/mariadb.conf.d/50-server.cnf
bind-address            = 127.0.0.1
```

- Port 631 (IPP) — unnecessary, associated with printer service (CUPS)

Fixes Applied: - Disabled CUPS to close Port 631

```
debian@debian:/$ sudo systemctl stop cups
debian@debian:/$ sudo systemctl disable cups
Synchronizing state of cups.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable cups
Removed "/etc/systemd/system/sockets.target.wants/cups.socket".
Removed "/etc/systemd/system/multi-user.target.wants/cups.path".
Removed "/etc/systemd/system/multi-user.target.wants/cups.service".
Removed "/etc/systemd/system/printer.target.wants/cups.service".
```

Prevention Recommendations:

- Regularly scan for open ports and audit running services.
- Disable services not in use. - Use host-based firewalls to restrict unnecessary exposure.

9: Apache 2.4.x < 2.4.64 - Multiple Vulnerabilities

Vulnerability: Nessus flagged Apache version 2.4.62 as vulnerable. Although not running the latest 2.4.64, Debian Security Tracker confirms that version 2.4.62-1 includes patches for CVEs including SSRF and HTTP proxy fixes.

In this case, updating the system would actually be counterproductive as it could break current dependencies/extensions. Apache remains actively maintained and patched via official Debian repositories.

Forensic Review:

- Verified version using apache2 -v
- Reviewed Debian changelog: relevant CVEs (e.g., CVE-2024-42516, CVE-2024-43204) already patched
- No upgrade needed outside official package manager

Exploit Detection:

- Checked Apache access and error logs

- Searched for:
    - Suspicious HTTP methods
    - Repeated access patterns
    - Encoded payloads
    - Proxy header abuse
    - Top IP offenders

Prevention Recommendations:

- Stay updated via official distro repositories
- Regularly monitor web server logs
- Set up alerts for abnormal traffic patterns and HTTP behavior

---

Conclusion

All identified vulnerabilities were successfully resolved or mitigated. Exploited vulnerabilities were confirmed through forensic analysis and corrected. Other vulnerabilities were proactively discovered and remediated before exploitation occurred.

The system is now significantly more secure, with improved configurations across SSH, MySQL, Apache, WordPress, and system file permissions. Preventative recommendations and hardening practices have been proposed throughout.

Next Steps:

- Implement a structured patch management process
- Conduct regular audits and re-tests
- Train staff on basic system security and monitoring

END REPORT

Important Notes:

- Since my initial machine corrupted and I had to start over, I used the notes I kept from the previous machine to immediately start mirroring the same steps. Unfortunately, this means that the Nessus report on this machine was not run until much later into the process where most vulnerabilities were already addressed and remediated.
- On the other hand, the Apache vulnerability is especially unique because I was already done with the project by 7/10/25. If the file didn't corrupt, then this vulnerability wouldn't have been detected during this project.