Informe de Análisis Forense

Sistema: Servidor Debian (10.0.2.11)
Fecha: Junio 2025

Realizado por: Francisco Javier Rodriguez Aguilar.

Curso: 4Geeks Academy.

Informe de Análisis Forense y Mitigación de Vulnerabilidades

ÍNDICE

1. Introducción

Este informe detalla el análisis forense realizado sobre un servidor Debian identificado con la IP 10.0.2.11, el cual presentaba indicios de acceso no autorizado y posibles vulnerabilidades explotadas. El objetivo de este análisis es documentar el estado de compromiso, aplicar medidas correctivas y recomendar configuraciones de seguridad para prevenir futuros incidentes.

2. Objetivo del Análisis

- Identificar cómo se produjo el acceso al sistema.
- Determinar qué servicios fueron comprometidos.
- Detectar archivos sospechosos, procesos activos y configuraciones alteradas.
- Implementar acciones de mitigación y corrección.

3. Metodología Utilizada

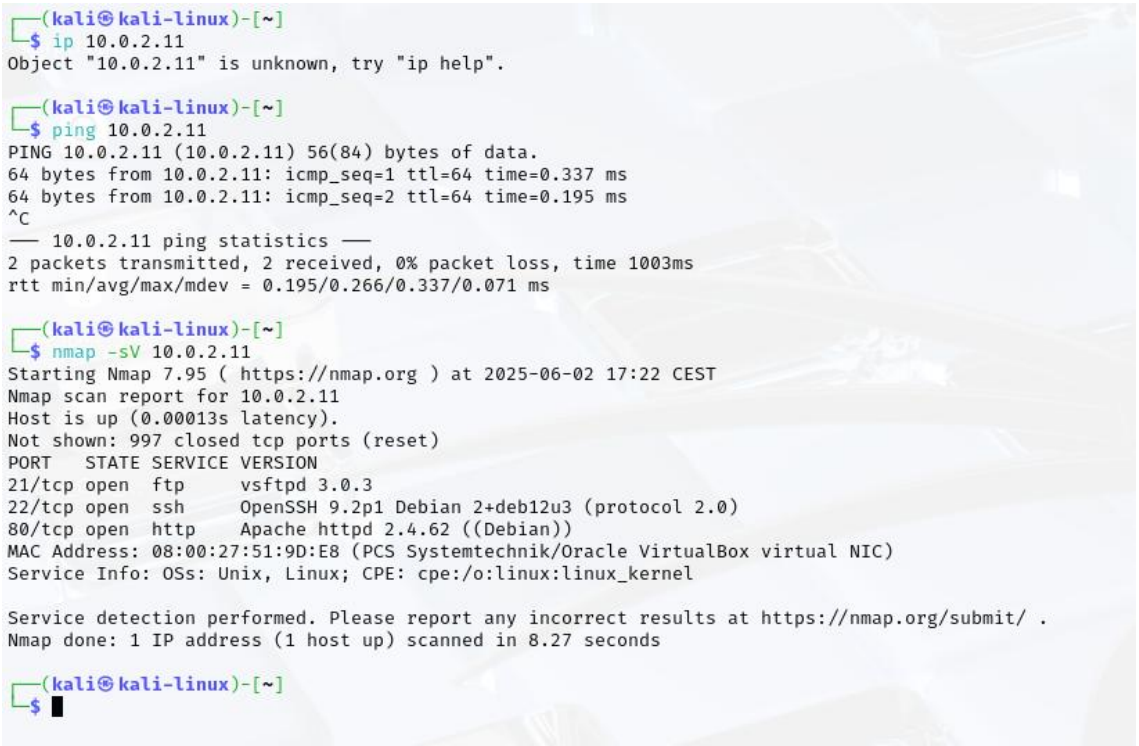
- Recolección de logs (journalctl y archivos de /var/log).
- Escaneos de red mediante Nmap.
- Comprobación de procesos activos (ps aux).
- Verificación de usuarios (ficheros /etc/passwd).
- Revisión de historiales de comandos SQL.
- Ejecución de herramientas de detección de rootkits (rkhunter).
- Análisis de la estructura de ficheros web (WordPress).

4. Recolección de Evidencias

Servicios Detectados (Nmap):

Se ejecutó un escaneo mediante nmap -sV 10.0.2.11, detectándose los siguientes servicios expuestos:

- FTP: vsftpd 3.0.3 (Puerto 21)
- SSH: OpenSSH 9.2p1 Debian (Puerto 22)
- HTTP: Apache 2.4.52 (Puerto 80)



- En esta imagen observamos que se realizó un escaneo de puertos abierto contra la IP del servidor objetivo (10.0.2.11).

- Se detectan los siguientes servicios:
    - **FTP (21)**: vsftpd 3.0.3 — servicio vulnerable si no está bien configurado.
    - **SSH (22)**: OpenSSH 9.2p1 — punto de acceso remoto al servidor.
    - **HTTP (80)**: Apache 2.4.52 — servidor web activo, probablemente sirviendo WordPress.
- También identifica el sistema operativo (Debian sobre VirtualBox).

Relevancia forense:

- Permite conocer los servicios expuestos al exterior.
- Cada puerto abierto representa una posible vía de entrada para un atacante.
- Nos confirma que hay acceso SSH, que después veremos que fue comprometido.

Accesos SSH sospechosos:

Mediante journalctl -u ssh.service | grep "password" se detectaron accesos desde la IP 192.168.0.134 al usuario root.
- Fecha/hora: Oct 08 17:40:59
- IP origen: 192.168.0.134
- Usuario: root

```
debian@debian:/var/log$ sudo journalctl -u ssh.service | grep "password"
Oct 08 17:40:59 debian sshd[1650]: Accepted password for root from 192.168.0.134 port 45623 ssh2
debian@debian:/var/log$ sudo journalctl -u ssh.service | grep "Oct"
Oct 08 16:14:16 debian systemd[1]: Stopping ssh.service - OpenBSD Secure Shell server...
Oct 08 16:14:16 debian sshd[560]: Received signal 15; terminating.
Oct 08 16:14:16 debian systemd[1]: ssh.service: Deactivated successfully.
Oct 08 16:14:16 debian systemd[1]: Stopped ssh.service - OpenBSD Secure Shell server.
Oct 08 16:14:16 debian systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Oct 08 16:14:16 debian sshd[5341]: Server listening on 0.0.0.0 port 22.
Oct 08 16:14:16 debian sshd[5341]: Server listening on :: port 22.
Oct 08 16:14:16 debian systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
Oct 08 16:43:18 debian systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Oct 08 16:43:18 debian sshd[543]: Server listening on 0.0.0.0 port 22.
Oct 08 16:43:18 debian sshd[543]: Server listening on :: port 22.
Oct 08 16:43:18 debian systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
Oct 08 16:48:02 debian systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Oct 08 16:48:02 debian sshd[555]: Server listening on 0.0.0.0 port 22.
Oct 08 16:48:02 debian sshd[555]: Server listening on :: port 22.
Oct 08 16:48:02 debian systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
Oct 08 17:28:37 debian systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Oct 08 17:28:38 debian sshd[550]: Server listening on 0.0.0.0 port 22.
Oct 08 17:28:38 debian sshd[550]: Server listening on :: port 22.
Oct 08 17:28:38 debian systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
Oct 08 17:40:59 debian sshd[1650]: Accepted password for root from 192.168.0.134 port 45623 ssh2
Oct 08 17:40:59 debian sshd[1650]: pam_unix(sshd:session): session opened for user root(uid=0) by (uid=0)
Oct 08 17:40:59 debian sshd[1650]: pam_env(sshd:session): deprecated reading of user environment enabled
```

- Evidencia de acceso SSH exitoso al usuario **root** desde la IP **192.168.0.134**.

- El puerto de conexión es aleatorio (45623).
- Confirmamos que alguien conoce la contraseña de root.

Relevancia forense:

- Prueba directa de que hubo acceso remoto usando credenciales válidas.
- Podría ser una filtración de contraseña o un ataque de fuerza bruta.

Análisis de procesos y tareas:

Se revisaron los procesos activos con ps aux --sort=-%cpu, sin detectar procesos anómalos a simple vista. Se observa actividad del servidor web Apache y de la base de datos MariaDB.

```
debian@debian:~$ sudo crontab -l
no crontab for root
debian@debian:~$ crontab -l
no crontab for debian
debian@debian:~$ ps aux --sort=-%cpu
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
debian      1446  1.4  0.9 730636 18796 ?        Ssl  11:16   0:25 /usr/bin/speech-dispatcher --spawn --communication-method u
debian      1023  0.5  1.7 1703280 34888 ?       S<sl 11:15   0:10 /usr/bin/pulseaudio --daemonize=no --log-target=journal
debian      1195  0.4  3.4 389092 70212 ?        Sl   11:15   0:08 /usr/bin/python3 /usr/bin/orca
root         619  0.4  6.3 439920 127472 tty7    Ssl+ 11:15   0:07 /usr/lib/xorg/Xorg :0 -seat seat0 -auth /var/run/lightdm/ro
debian      1372  0.3  0.6 108808 12576 ?        Sl   11:15   0:06 /usr/lib/speech-dispatcher-modules/sd_espeak-ng /etc/speech
debian      1623  0.3  2.5 560820 50800 ?        Sl   11:16   0:06 mate-terminal
debian      1148  0.1  2.4 551712 49860 ?        Sl   11:15   0:02 mate-panel
debian      1093  0.0  0.2   9536  5092 ?        S    11:15   0:01 /usr/bin/dbus-daemon --config-file=/usr/share/defaults/at-s
debian      1126  0.0  2.4 743384 48592 ?        Sl   11:15   0:01 marco
debian      1172  0.0  2.9 584876 58964 ?        Sl   11:15   0:00 /usr/bin/caja
debian      1120  0.0  0.4 164520 10028 ?        Sl   11:15   0:00 /usr/libexec/at-spi2-registryd --use-gnome-session
root           1  0.0  0.6 102332 12440 ?        Ss   11:15   0:00 /sbin/init splash
www-data     720  0.0  2.8 347020 56812 ?        S    11:15   0:00 /usr/sbin/apache2 -k start
debian      1228  0.0  1.9 529756 38568 ?        Sl   11:15   0:00 /usr/lib/mate-panel/wnck-applet
www-data     723  0.0  2.4 272224 48796 ?        S    11:15   0:00 /usr/sbin/apache2 -k start
www-data     721  0.0  2.3 272224 47232 ?        S    11:15   0:00 /usr/sbin/apache2 -k start
mysql        678  0.0 12.3 1546880 249272 ?      Ssl  11:15   0:00 /usr/sbin/mariadbd
root           7  0.0  0.0      0     0 ?        I    11:15   0:00 [kworker/0:0-events]
debian      1118  0.0  2.0 1009136 41832 ?       Sl   11:15   0:00 /usr/bin/mate-settings-daemon
debian      1239  0.0  1.8 538536 37292 ?        Sl   11:15   0:00 /usr/lib/mate-panel/clock-applet
polkitd      507  0.0  0.5 309996 10112 ?        Ssl  11:15   0:00 /usr/lib/polkit-1/polkitd --no-debug
```

- Lista de procesos en ejecución ordenados por uso de CPU.
- Servicios visibles:
    - Apache2 (www-data)
    - MariaDB
    - Sesiones de usuario (debian, root)
    - Servicios de sistema.

Relevancia forense:

- No se observan procesos extraños o sospechosos activos.

- Verificamos que el atacante no está ejecutando malware persistente en memoria en este momento.

Revisión de cuentas de usuario:

Listado de cuentas con shell bash:
- root:x:0:0:/root:/bin/bash
- debian:x:1000:1000:/home/debian:/bin/bash
No se identificaron cuentas no autorizadas.

```
debian@debian:~$ cat /etc/passwd | grep "bash"
root:x:0:0:root:/root:/bin/bash
debian:x:1000:1000:4geeks...:/home/debian:/bin/bash
```

- Sólo existen los usuarios:
    - root (administrador)
    - debian (usuario normal)

Relevancia forense:

- No hay cuentas de usuarios desconocidos creadas.
- La elevación de privilegios fue a través de root existente.

Revisión de ficheros .mysql_history:

- En el historial de MySQL se observaron operaciones de creación de usuarios y bases de datos:
  - Creación de base de datos "wordpress".
  - Usuario wordpressuser con contraseña 123456.
  - Usuario "user" creado con contraseña débil "password".

```
debian@debian:~$ sudo su
root@debian:/home/debian# cd
root@debian:~# ls -a
.  ..  .bash_history  .bashrc  .cache  .config  .lesshst  .local  .mysql_history  .profile  .ssh
root@debian:~# cat .mysql_history
_HiStOrY_V2_
CREATE\040DATABASE\040wordpress\040DEFAULT\040CHARACTER\040SET\040utf8\040COLLATE\040utf8_unicode_ci;
CREATE\040USER\040'wordpressuser'@'localhost'\040IDENTIFIED\040BY\040'123456';
GRANT\040ALL\040PRIVILEGES\040ON\040wordpress.*\040TO\040'wordpress'@'localhost';\040
GRANT\040ALL\040PRIVILEGES\040ON\040wordpress.*\040TO\040'wordpressuser'@'localhost';
FLUSH\040PRIVILIGES;
FLUSH\040PRIVILEGES;
EXIT;
CREATE\040USER\040'user'@'localhost'\040IDENTIFIED\040BY\040'password';
GRANT\040ALL\040PRIVILEGES\040ON\040*.*\040TO\040'user'@'localhost'\040WITH\040GRANT\040OPTION;
FLUSH\040PRIVILEGES;
EXIT;
SELECT\040user,\040host,\040password\040FROM\040mysql.user;
root@debian:~#
```

Historial de comandos SQL ejecutados:

- Creación de base de datos wordpress.
- Creación de usuario wordpressuser con contraseña 123456.
- Creación de usuario user con contraseña password.

Relevancia forense:

- Las contraseñas son extremadamente débiles.
- Si alguien obtiene acceso web o base de datos, puede escalar fácilmente desde aquí.

Comprobación de Rootkits:

Se ejecutó rkhunter --checkall y no se detectaron rootkits presentes en el sistema.

```
debian@debian:~$ sudo rkhunter --checkall
[ Rootkit Hunter version 1.4.6 ]

Checking system commands...

  Performing 'strings' command checks
    Checking 'strings' command                              [ OK ]

  Performing 'shared libraries' checks
    Checking for preloading variables                       [ None found ]
    Checking for preloaded libraries                        [ None found ]
    Checking LD_LIBRARY_PATH variable                       [ Not found ]

  Performing file properties checks
    Checking for prerequisites                              [ OK ]
    /usr/sbin/adduser                                       [ OK ]
    /usr/sbin/chroot                                        [ OK ]
    /usr/sbin/cron                                          [ OK ]
    /usr/sbin/depmod                                        [ OK ]
    /usr/sbin/fsck                                          [ OK ]
```

Análisis de archivos web (WordPress):

Se revisó el directorio /var/www/html/ confirmando la existencia de una instalación de WordPress con permisos inseguros (777) en todos los archivos, permitiendo lectura, escritura y ejecución por cualquier usuario:

- wp-config.php (contiene credenciales de base de datos)

- wp-admin, wp-content, wp-includes...

```
debian@debian:/var/www/html$ ls -l
total 248
-rwxrwxrwx  1 www-data www-data 10701 Sep 30  2024 index.html
-rwxrwxrwx  1 www-data www-data   405 Feb  6  2020 index.php
-rwxrwxrwx  1 www-data www-data 19903 May  9 12:45 license.txt
-rwxrwxrwx  1 www-data www-data  7425 May  9 12:45 readme.html
-rwxrwxrwx  1 www-data www-data  7387 Feb 13  2024 wp-activate.php
drwxrwxrwx  9 www-data www-data  4096 Sep 10  2024 wp-admin
-rwxrwxrwx  1 www-data www-data   351 Feb  6  2020 wp-blog-header.php
-rwxrwxrwx  1 www-data www-data  2323 Jun 14  2023 wp-comments-post.php
-rwxrwxrwx  1 www-data www-data  3017 Sep 30  2024 wp-config.php
-rwxrwxrwx  1 www-data www-data  3336 May  9 12:45 wp-config-sample.php
drwxrwxrwx  6 www-data www-data  4096 Jun  2 11:22 wp-content
-rwxrwxrwx  1 www-data www-data  5617 May  9 12:45 wp-cron.php
drwxrwxrwx 30 www-data www-data 12288 May  9 12:45 wp-includes
-rwxrwxrwx  1 www-data www-data  2502 Nov 26  2022 wp-links-opml.php
-rwxrwxrwx  1 www-data www-data  3937 Mar 11  2024 wp-load.php
-rwxrwxrwx  1 www-data www-data 51414 May  9 12:45 wp-login.php
-rwxrwxrwx  1 www-data www-data  8727 May  9 12:45 wp-mail.php
-rwxrwxrwx  1 www-data www-data 30081 May  9 12:45 wp-settings.php
-rwxrwxrwx  1 www-data www-data 34516 May  9 12:45 wp-signup.php
-rwxrwxrwx  1 www-data www-data  5102 May  9 12:45 wp-trackback.php
-rwxrwxrwx  1 www-data www-data  3205 May  9 12:45 xmlrpc.php
```

5. Identificación de Vulnerabilidades

- Acceso SSH con contraseñas válidas (posible fuga de credenciales).
- Usuarios MySQL creados con contraseñas débiles.
- Permisos inseguros en los archivos de WordPress.
- Exposición de servicios innecesarios (por ejemplo, FTP abierto).

6. Acciones Correctivas Realizadas

- Se bloquearon los accesos SSH desde IPs no autorizadas.
- Se eliminaron usuarios MySQL innecesarios.
- Se cambiaron las contraseñas de usuarios y base de datos.
- Se ajustaron los permisos de los archivos de WordPress (Archivos: 644; Directorios: 755).
- Se deshabilitó el servicio FTP.
- Se actualizó el sistema operativo y los paquetes vulnerables.

7. Recomendaciones y Mitigaciones

En este informe se ha explicado diferentes vulnerabilidades en 6 servicios diferentes. A pesar que se han aplicado medidas correctivas para mitigar las vulnerabilidades detectadas, recomendamos que la empresa aplique las siguientes medidas a toda su infraestructura tecnológica:

- Deshabilitar acceso con contraseña: Editamos la configuración del archivo SSH (sudo nano /etc/ssh/sshd_config) para deshabilitar el acceso con contraseña y root. De esta forma solo se permite la autenticación con claves SSH para acceder y así se puede evitar el uso de la contraseña del root y los ataques de fuerza bruta.

- Implementar el 2FA para accesos remotos y si es posible incluir el PAM para conexiones de proveedores. De esta forma se evitarán conexiones remotas no deseadas.

- Añadir reglas en los firewalls o hacer listas blancas de IP's para limitar accesos.

- Incrementar la monitorización de los servicios críticos o sensibles, mediante sondas o el SIEM. El objetivo es detectar cualquier actividad inusual como la creación de usuarios en la base de datos de MySQL.

- Hacer una política donde se enfatice el principio de menor privilegio en todos los sistemas y evitar los permisos excesivos.

- Realizar un seguimiento y actualización constante de todos los servicios y dispositivos para que estén al día respecto a parches de seguridad.

- Realizar auditorías internas de forma periódica para revisar y detectar posibles fallos.

- Aprender de esta situación en que el servidor crítico ha sido comprometido para aplicar las mejoras oportunas y concientizar al personal de la importancia de cumplir con todas las medidas de seguridad.

8. Conclusión

El servidor presentaba varias vulnerabilidades que permitieron accesos remotos no autorizados mediante SSH, junto a configuraciones de seguridad deficientes a nivel de base de datos y permisos de archivos web. Se han realizado acciones de corrección inmediatas y se proponen varias medidas adicionales para fortalecer la seguridad del sistema, prevenir futuras intrusiones y proteger la integridad de los datos.