UNIVERSIDAD POLITÉCNICA DE SAN LUIS POTOSÍ

Universidad Politécnica de San Luis Potosí

Ingeniería en Tecnologías de la Información

ACTIVIDAD 11

Red Team Report: Pentesting de Napping 101

Informe técnico con enfoque ejecutivo y trazabilidad metodológica

Asignatura: Seguridad Informática
Docente: Mtro. Servando López Contreras
Equipo: 5
Fecha: 22 de Marzo 2026

Integrantes

Coronado Noriega Jesús Olaf	178991
De La Rosa Rodríguez Erik	177700
González Reyes Felipe de Jesús	181134
Mendoza Aguado Karina	179859
Pérez Ventura Juan Alejandro	180370
Serrano Zermeño Leonardo	177301

San Luis Potosí, S.L.P.

Índice

1. Executive Summary (Resumen ejecutivo)

Conclusión para toma de decisiones. La máquina *Napping 101* fue comprometida mediante una cadena de dos fallas conectadas: primero, un flujo de revisión administrativa que permite que contenido controlado por un usuario termine siendo procesado por una cuenta de mayor confianza; después, una automatización insegura que rompe la separación de privilegios y permite escalar hasta control total del host.
Riesgo ejecutivo. No se trata de dos hallazgos aislados. Se trata de una ruta completa desde interacción web hasta control como *root*. En un entorno real, esa cadena expone a la organización a pérdida de control operativo, alteración de archivos críticos e indisponibilidad del servicio.
Tres decisiones inmediatas.

1. Corregir propietarios y permisos de scripts ejecutados por tareas privilegiadas.
2. Rediseñar el flujo de revisión administrativa de enlaces externos.
3. Implementar controles compensatorios para que una credencial robada no equivalga a acceso operativo pleno.

1.1. Hallazgos clave

- **Modelo de confianza roto en la aplicación web.** El portal permite que un usuario no privilegiado entregue un enlace que termina siendo procesado por el administrador. Ese diseño convierte la revisión administrativa en superficie de ataque.
- **Exposición operativa amplificada por SSH.** La presencia simultánea de HTTP y SSH hace que cualquier captura de acceso útil deje de ser un problema puramente web y se convierta en una vía real de entrada al sistema.
- **Escalada de privilegios por automatización insegura.** Un artefacto ejecutado desde un contexto privilegiado puede ser alterado por un usuario con menos privilegios, habilitando control total del host.

1.2. Impacto potencial

- **Confidencialidad:** exposición de credenciales y de archivos del sistema.
- **Integridad:** modificación de scripts, automatizaciones y evidencia operativa.
- **Disponibilidad:** posibilidad de interrupción del servicio al alcanzar privilegios máximos.

2. Alcance (Scope)

2.1. Activos y límites

- **Activo evaluado:** máquina *Napping 101* en entorno de laboratorio.
- **Superficie revisada:** red, aplicación web, acceso remoto expuesto y automatizaciones locales.
- **Fuera de alcance:** infraestructura externa, denegación de servicio sostenida y acciones fuera del laboratorio.

3. Metodología aplicada (PTES / OWASP)

PTES estructuró la secuencia general del ejercicio y OWASP orientó el análisis del componente web desde superficie, autenticación, sesión y lógica de negocio. Cada fase responde una pregunta analítica concreta, ejecuta una acción verificable, utiliza evidencia específica y produce una decisión técnica.

3.1. Trazabilidad metodológica

Marco	Pregunta analítica	Ejecución y evidencia	Decisión obtenida
PTES - Reconocimiento	¿Qué activo es el objetivo y qué servicios habilitan una cadena de ataque real?	Descubrimiento del host y escaneo activo. Evidencia: Figuras 1 y 2.	Se priorizó la combinación HTTP + SSH porque convierte una falla de flujo web en acceso operativo al host.
OWASP - Autenticación y sesión	¿Cómo se autentica el usuario y qué señales de hardening o debilidad muestra la aplicación?	Revisión del portal, cabeceras, cookie de sesión reportada por el escaneo y páginas funcionales. Evidencia: Figuras 2, 3 y 4.	Se descartó tratar la web como "solo una pantalla de login" y se analizó como un flujo de confianza con impacto administrativo.
OWASP - Lógica de negocio	¿Quién puede introducir contenido, quién lo procesa y bajo qué confianza?	Observación del flujo registro-login-panel-envío de enlaces para revisión administrativa. Evidencia: Figuras 3 y 4.	Se formuló la hipótesis de compromiso inicial a partir del modelo de confianza roto, no a partir de prueba ciega.
PTES - Explotación	¿La hipótesis planteada es explotable en laboratorio?	Validación controlada del compromiso inicial y captura de acceso útil. Evidencia: Figura 5.	La hipótesis pasó de plausible a demostrada. El problema deja de ser teórico.
PTES - Post-explotación y privesc	¿El acceso inicial permite impacto mayor o queda contenido?	Enumeración local, localización de automatización insegura y verificación de control total. Evidencia: Figuras 6 y 7.	Se confirmó impacto máximo: la cadena termina en control total del sistema.

3.2. Cómo se priorizó la investigación

La secuencia metodológica siguió una lógica de encadenamiento técnico: primero se identificó un activo con dos servicios relevantes; después se confirmó que el portal no solo autenticaba usuarios, sino que introducía un flujo donde un administrador revisaba enlaces enviados por terceros; por

último, se verificó si un acceso inicial derivado de ese flujo podía escalar hasta el sistema operativo. Cada transición entre fases estuvo motivada por la evidencia obtenida en la fase anterior.

3.3. Reglas de compromiso

- Actividad realizada únicamente en un laboratorio controlado.
- El impacto se validó con evidencia suficiente, evitando exfiltración innecesaria.
- Los detalles operativos sensibles se preservan solo como evidencia controlada y no como instructivo reutilizable.

4. Reconocimiento y enumeración técnica

4.1. Descubrimiento de host y mapa inicial de servicios

La fase de descubrimiento permitió aislar al objetivo dentro de una red pequeña y acotar el análisis sobre la IP que posteriormente respondió al escaneo activo. El valor de esta fase no está en "encontrar una IP", sino en evitar suposiciones: antes de tocar la aplicación había que confirmar qué activo era realmente el objetivo.

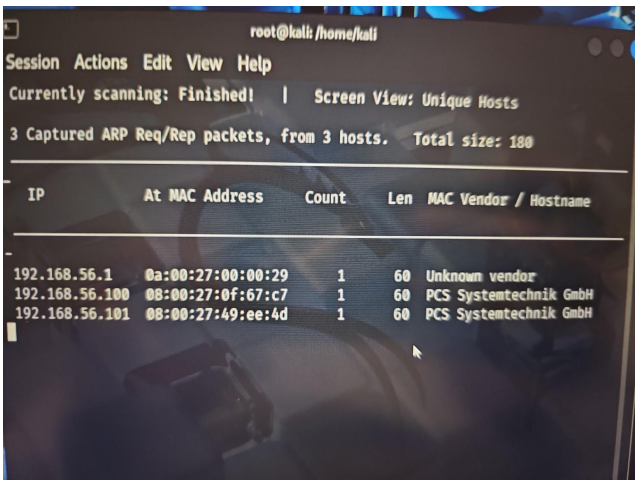

Figura 1: Descubrimiento del host dentro de la red privada.

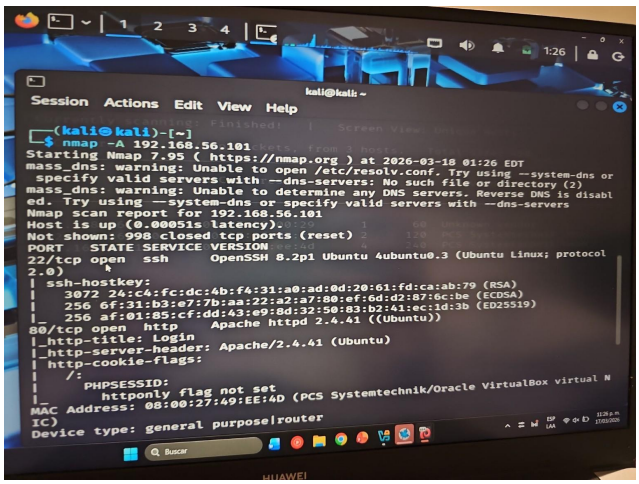

Figura 2: Escaneo activo con identificación de servicios expuestos.

A partir del escaneo se observaron dos puertos abiertos con relevancia inmediata:

- **22/tcp - SSH (OpenSSH 8.2p1).** No se trató como vulnerabilidad por sí misma. Se interpretó como **amplificador de impacto**: cualquier acceso útil obtenido en la capa web podía convertirse en entrada real al sistema.
- **80/tcp - HTTP (Apache 2.4.41).** El título del sitio reportó *Login* y el mismo escaneo mostró una cookie `PHPSESSID` sin la bandera `HttpOnly`. Esa observación no resolvía la máquina por sí sola, pero sí apuntaba a un endurecimiento web incompleto.

También hubo dos señales secundarias que ayudaron a priorizar sin sobreafirmar:

- la existencia de `/server-status` con respuesta 403 sugiere un artefacto administrativo de Apache expuesto, aunque controlado;
- la combinación "portal de login" + "SSH abierto" hacía razonable buscar un encadenamiento entre aplicación y sistema operativo.

4.2. Enumeración web y lectura del flujo funcional

La enumeración no se limitó a un listado de rutas. El objetivo fue reconstruir qué hace la aplicación y dónde deposita confianza. El barrido de recursos mostró páginas coherentes con un portal funcional: `index.php`, `login.php`, `register.php`, `logout.php`, `welcome.php` y archivos relacionados con configuración; además, `/server-status` apareció restringido con 403. Eso confirmó que había un flujo de autenticación real y no un señuelo aislado.

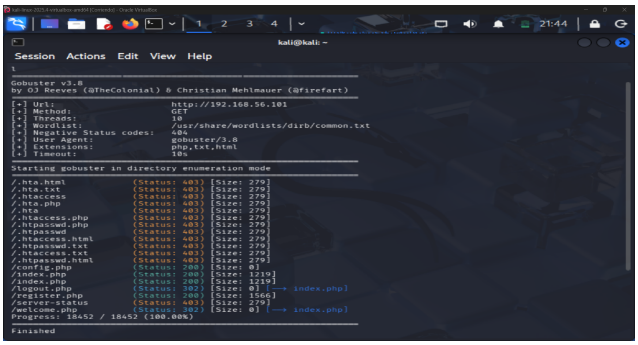

Figura 3: Enumeración de rutas y recursos relevantes del portal.

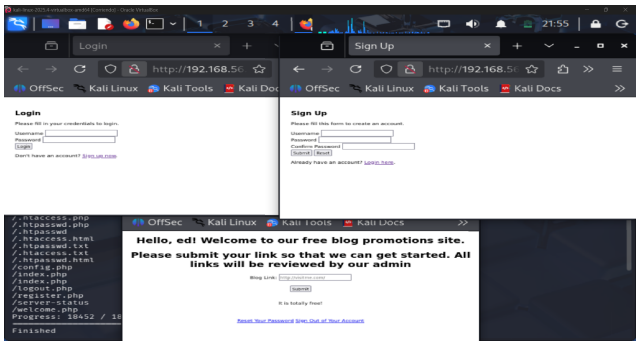

Figura 4: Flujo funcional observado: registro, autenticación y envío de enlaces para revisión.

La parte crítica de la enumeración fue identificar el flujo de negocio:

1. un usuario puede registrarse y autenticarse;
2. ese usuario llega a un panel donde envía un enlace;
3. el enlace será revisado por el administrador.

Ese punto cambia por completo la lectura del sistema. Ya no se está frente a una web cualquiera, sino frente a un proceso donde un actor de menor privilegio puede introducir contenido que terminará siendo procesado por una cuenta con mayor contexto de confianza. Esa observación fue la base de la hipótesis de ataque.

4.3. Hipótesis de ataque y justificación

La hipótesis no fue "probar cosas hasta que algo funcione". Fue esta: si el administrador revisa contenido externo enviado por terceros y la aplicación no aísla adecuadamente esa interacción, entonces existe una oportunidad razonable de obtener acceso útil aprovechando el propio flujo de confianza del sistema. El paso siguiente, por tanto, no fue seguir enumerando por inercia, sino validar si esa hipótesis podía materializarse en un compromiso inicial real.

5. Explotación documentada

5.1. Secuencia validada

La explotación se documentó como una cadena de verificación, no como una colección de capturas inconexas:

1. se utilizó una cuenta de bajo privilegio para interactuar con el flujo legítimo del portal;
2. se suministró contenido controlado dentro del flujo de revisión administrativa;
3. la revisión administrativa generó exposición de acceso útil para el atacante;
4. ese acceso se reutilizó para validar entrada real al sistema y continuar la investigación local.

La secuencia validada demuestra una cadena de explotación funcional sin exponer detalles operativos innecesarios.

5.2. Evidencia de compromiso inicial

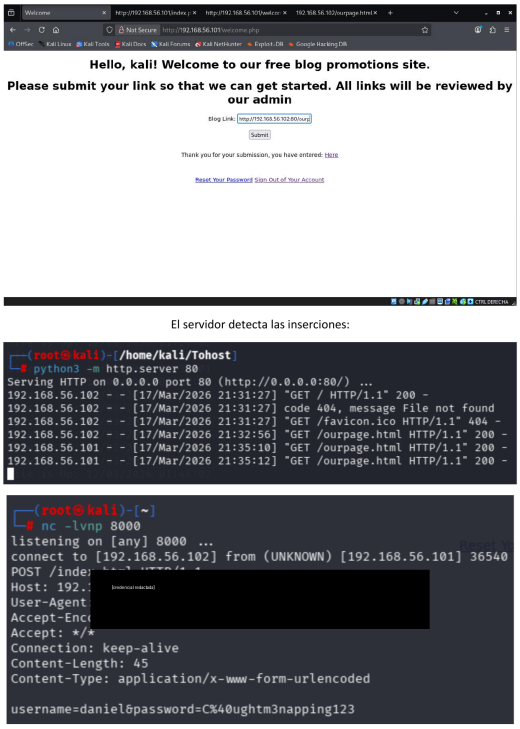

Figura 5: Compromiso inicial validado. La evidencia muestra la materialización del acceso útil obtenido a partir del flujo de revisión administrativa. La credencial visible en la captura fue redactada para mantener control de la evidencia.

Alcance de la evidencia. La captura confirma que el flujo web permitió transformar una interacción administrativa en acceso útil para el atacante.
Interpretación técnica. La evidencia sustenta el compromiso inicial validado. Los controles compensatorios sobre autenticación y protección de cuentas privilegiadas se presentan como medidas

de mitigación, no como vulnerabilidades independientes demostradas durante la práctica.

5.3. Acceso inicial y validación

Con el acceso inicial se realizó una revisión local mínima para responder una pregunta concreta: ¿el compromiso inicial queda contenido o abre la puerta a mayor impacto? La enumeración local identificó usuarios, archivos de interés y una automatización con potencial de abuso. En ese punto, la investigación pasó de acceso inicial a post-explotación.

6. Escalada de privilegios

6.1. Hallazgo local relevante

La revisión local identificó un patrón crítico: una automatización ejecutada con mayor privilegio dependía de un artefacto que podía ser alterado por un usuario sin privilegios máximos. El problema no era "un archivo raro", sino una ruptura directa del principio de separación de privilegios.

Figura 6: Automatización insegura y artefacto modificable asociado a la hipótesis de escalada. El detalle operativo sensible se mantuvo redactado, pero la relación entre archivo, automatización y ejecución privilegiada permanece visible.

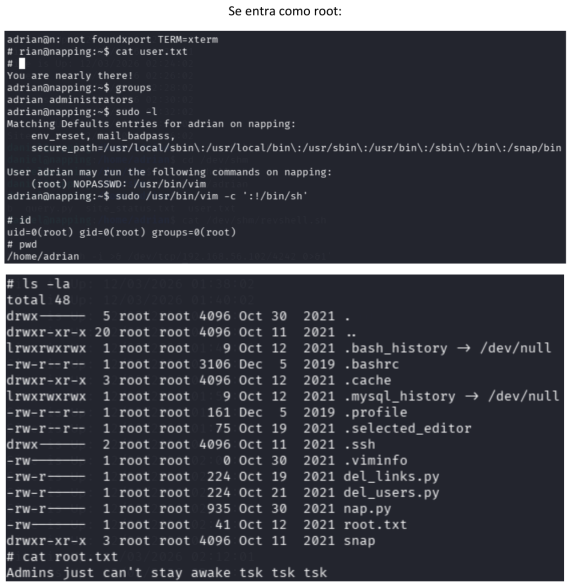

Figura 7: Validación de control total del host tras la escalada de privilegios.

6.2. Causa raíz

La causa raíz se define como una **cadena de mala práctica operativa**:

- propietarios y permisos incorrectos sobre artefactos que terminan ejecutándose en contexto privilegiado;
- ausencia de separación clara entre espacio modificable por usuario y ruta consumida por una automatización crítica;

- dependencia de confianza implícita en archivos cuya integridad no está protegida.

6.3. Por qué esto sí prueba impacto máximo

La escalada no se trató como un supuesto. Se verificó la capacidad de operar con privilegios máximos en el host. Eso cambia la evaluación del riesgo: ya no se habla de una falla puntual en la web, sino de compromiso completo del sistema.

7. Análisis de impacto: del CIA al impacto operativo

7.1. Modelo CIA por activo

Activo	Confidencialidad	Integridad	Disponibilidad
Cuenta administrativa del portal	Alta	Media	Media
Automatizaciones y scripts del host	Media	Alta	Alta
Servidor web y su contenido	Media	Alta	Alta
Archivos del sistema y cuentas locales	Alta	Alta	Alta

7.2. Traducción a impacto organizacional

Impacto operativo. La organización pierde control sobre un activo que combina portal y acceso al sistema. La explotación no solo compromete una sesión web; habilita manipulación de archivos, de automatizaciones y de cuentas locales.
Impacto de negocio. En un entorno real, la cadena observada obligaría a contención urgente, revisión de credenciales, verificación de integridad del host y posible indisponibilidad temporal del servicio.

8. Recomendaciones técnicas y ejecutivas

8.1. Prioridad inmediata (0 a 7 días)

- Corregir propietarios y permisos de cualquier script o artefacto ejecutado por tareas privilegiadas.
- Revisar `crontab`, temporizadores y automatizaciones equivalentes para eliminar dependencias sobre rutas modificables por usuarios no privilegiados.
- Rediseñar o aislar el flujo de revisión administrativa de enlaces externos.
- Implementar controles compensatorios que impidan que una credencial robada equivalga por sí sola a acceso operativo efectivo.

8.2. Prioridad de corto plazo (8 a 30 días)

- Endurecer la capa web: validación estricta de URLs, revisión de cabeceras y banderas de cookie, y saneamiento de endpoints administrativos expuestos.

- Incorporar mecanismos de integridad y revisión periódica de artefactos críticos.
- Reducir superficie operativa innecesaria y revisar principio de mínimo privilegio.

8.3. Prioridad de madurez (30 a 90 días)

- Formalizar *threat modeling* del flujo administrativo para que el diseño de la aplicación no deposite confianza ciega en contenido de terceros.
- Integrar revisiones periódicas de seguridad web y del host como práctica continua, no como corrección puntual.

9. Tabla de hallazgos

ID	Hallazgo	Severidad	Evidencia	Impacto	Recomendación
F-01	Modelo de confianza roto en el flujo de revisión administrativa de enlaces externos.	Crítica	Figuras 3, 4 y 5	Permite compromiso inicial y convierte una interacción legítima del portal en acceso útil para el atacante.	Aislar revisión de enlaces, validar URLs y agregar controles compensatorios sobre cuentas de alto valor.
F-02	Endurecimiento web incompleto: cookie de sesión sin `HttpOnly` reportada en el escaneo y presencia de endpoint administrativo restringido.	Media	Figuras 2 y 3	No resuelve la máquina por sí solo, pero incrementa la superficie de ataque y reduce resiliencia del portal ante fallas adicionales.	Revisar banderas de sesión, cabeceras, exposición de endpoints y configuración de Apache.
F-03	Automatización privilegiada depende de artefacto modificable por usuario no privilegiado.	Crítica	Figuras 6 y 7	Habilita escalada de privilegios hasta control total del host.	Corregir propietarios y permisos, separar rutas de ejecución y proteger integridad de artefactos críticos.

10. Conclusión

El problema principal de *Napping 101* no fue una suma de detalles técnicos aislados, sino una cadena coherente: una aplicación que deposita confianza donde no debe y un sistema operativo que permite que esa primera grieta se convierta en control total. La conclusión técnica es clara: mientras no se corrijan tanto el flujo de confianza del portal como la automatización insegura del host, el activo seguirá siendo comprometible con impacto alto.

UNIVERSIDAD POLITÉCNICA DE S.L.P.

Red Team Report:

Informe técnico con enfoque ejecutivo y trazabilidad metodológica orientada al impacto de negocio.

CLASIFICACIÓN

Uso Directivo / Confidencial

EQUIPO 5

Coronado N., De La Rosa R., González R., Mendoza A., Pérez V., Serrano Z.

DOCENTE

Mtro. Servando López Contreras

Alcance y Resumen Ejecutivo

El objetivo del ejercicio ofensivo fue determinar el riesgo real de la máquina **Napping 101**, evaluando no solo las fallas aisladas, sino si existe una ruta completa hacia el impacto crítico del sistema bajo un entorno controlado de laboratorio.

VEREDICTO EJECUTIVO

Comprometido

Mensaje directivo: Existe una cadena viable desde la capa web hasta el control operativo total del host (root). No se trata de vulnerabilidades teóricas, sino de una explotación encadenada probada exitosamente.

Prioridad Máxima | Contención Inmediata

ACTIVO EVALUADO

Servidor "Napping 101" (Laboratorio)

LÍMITES DE ALCANCE (ROE)

- Sin infraestructura externa
- Sin acciones de denegación de servicio (DoS)
- Sin ejecución automatizada destructiva

Framework de Ejecución: PTES + OWASP

Aseguramos una ejecución estructurada aplicando el **Penetration Testing Execution Standard (PTES)** para la lógica del ataque, apoyado con directrices **OWASP Top 10** para el análisis en profundidad de la aplicación web alojada.

PTES (FASE ESTRUCTURAL)

- **Reconocimiento Analítico:** Mapa de servicios y priorización de vectores en base a exposición (HTTP + SSH).
- **Construcción de Hipótesis:** Análisis pasivo del flujo de confianza sin alertar al objetivo.
- **Explotación Determinista:** Validación de la vulnerabilidad inicial sin usar fuerza bruta o escaneo destructivo.
- **Post-Explotación:** Verificación y consolidación de impacto hasta root.

OWASP (CAPA DE APLICACIÓN)

- **Defectos de Autenticación / Sesión:** Intercepción de galletas y banderas deficientes (falta de HttpOnly).
- **Lógica de Negocio Quebrantada:** El origen del ataque; cómo la aplicación procesa enlaces bajo un falso esquema de confianza ciega hacia el administrador.

Riesgo Crítico y Descubrimiento Inicial

El reconocimiento técnico y escaneo activo (Nmap) confirmaron una superficie de ataque que combina servicios web vulnerables con vías de acceso remoto; lo que cataliza cualquier debilidad transformándola en riesgo sistémico.

DRIVERS DEL RIESGO IDENTIFICADO

- Exposición combinada: **Puerto 80 (HTTP) y 22 (SSH)** actúan como puente hacia el SO.
- Modelo de confianza roto en el flujo web.
- Deficiencias en separación de privilegios (Root).

CLASIFICACIÓN CVSS BASE ESTIMADA

9.8 (CRÍTICO)

EVIDENCIA TÉCNICA: ESCANEO DE SUPERFICIE

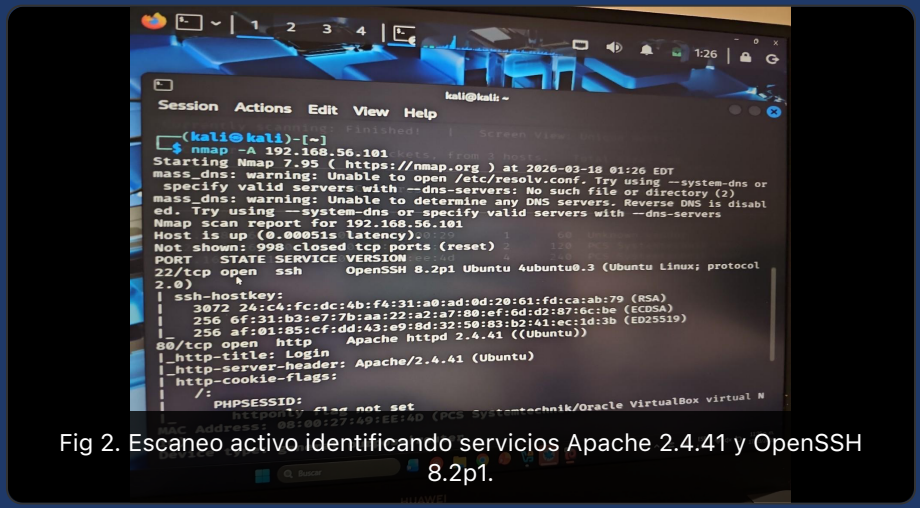

Fig 2. Escaneo activo identificando servicios Apache 2.4.41 y OpenSSH 8.2p1.

Cadena de Ataque y Explotación

El esquema ofensivo no depende de una sola falla, sino de una **cadena de ataques vinculados (Kill Chain)** que culminan en el secuestro total de las capacidades operativas del servidor.

Inyección Web
Flujo Administrativo

→

Robo Sesión/Cred
(Acceso Útil)

→

Pivote por SSH
(Acceso Operativo)

→

Abuso de Cronjob
(Escalada Privilegios)

→

Control Root
(Impacto Máximo)

EVIDENCIA: COMPROMISO INICIAL

Fig 5. Robo de sesión validado mediante inyección.

EVIDENCIA: ESCALADA A ROOT (PRIVESC)

Fig 7. Escalada exitosa obteniendo privilegios de root.

Pérdida Total del Triángulo CIA

Traducido a un escenario corporativo, el control a nivel **root** implica que el servidor está a merced del actor amenaza. El daño operacional y financiero supera el nivel de la aplicación web.

CONFIDENCIALIDAD

Exfiltración Masiva: Acceso ilimitado a bases de datos, contraseñas hash del SO (/etc/shadow), y exfiltración de propiedad intelectual alojada.

INTEGRIDAD

Alteración Silenciosa: Capacidad de inyectar código malicioso permanente en el código fuente de Napping, modificar permisos y eliminar logs.

DISPONIBILIDAD

Secuestro (Ransomware): Posibilidad de cifrar la máquina en su totalidad, destruir respaldos locales o desplegar troyanos (botnets) interrumpiendo el negocio.

Impacto Operacional Directivo

La organización requerirá activación del plan de manejo de incidentes, contención forense de servidores productivos afectados, y notificación mandatoria bajo marcos de cumplimiento normativo (Compliance).

Mapeo de Probabilidad vs Impacto

El análisis demuestra que los vectores principales no requieren acceso privilegiado previo y tienen documentación pública (alta probabilidad), desencadenando en control de activo (alto impacto).

Focos Críticos

F-03: Automatización Privilegiada Insegura. (Impacto: Total | Probabilidad: Alta tras compromiso ini).

F-01: Modelo de Confianza Roto Web. (Impacto: Medio/Alto | Probabilidad: Muy Alta).

La acumulación de los vectores F-01 y F-03 ubica el riesgo global de la aplicación en el cuadrante rojo crítico definitivo.

Mitigación Definitiva (Causa Raíz)

No basta con parchar módulos; se debe rediseñar la arquitectura de procesos, corregir el principio de menor privilegio y endurecer la validación de terceros.

1

Aislamiento de Revisiones (Web)

Validar estrictamente las entradas (URLs), restringir los privilegios del bot revisor, y auditar galletas de sesión imponiendo directivas HttpOnly y Secure.

2

Corrección de Escalada de Privilegios (OS)

Auditar de inmediato las Tareas Programadas Críticas (Cronjobs). Asegurar que los scripts del sistema tengan dueños correctos (root:root) y descartar modificaciones de grupos de bajo nivel.

3

Zero Trust y Controles Compensatorios

Impedir que una credencial robada otorgue el control del host. Aplicar 2FA a servicios expuestos (SSH) e implementar detección activa en el Endpoint (EDR).

Plan de Ejecución a 90 Días

Estrategia iterativa diseñada para detener la brecha de inmediata, y evolucionar la madurez de seguridad informática en el mediano plazo.

Bloque 1: Contención 0 - 7 DÍAS

Hotfix PrivEsc: Corregir permisos en archivos leídos por Cron (chmod 755 y chown root).

Aislar revisión administrativa (Sandbox o deshabilitar temporalmente).

Rotar todas las contraseñas web, de sistema y llaves SSH comprometidas.

Bloque 2: Hardening 8 - 30 DÍAS

Código Seguro: Desinfectar la aplicación Napping garantizando validaciones de entradas (OWASP ASVS).

Actualizar Apache y parametrizar cabeceras HTTP de seguridad restrictivas.

Endurecer el acceso SSH limitando IP y rehusando passwords por defecto.

Bloque 3: Madurez 30 - 90 DÍAS

Threat Modeling: Analizar la lógica de negocio para futuros despliegues.

Establecer monitoreo continuo (SIEM) de logs críticos.

Crear auditorías recurrentes sobre el cumplimiento del principio del menor privilegio (PoLP).

Glosario Final para Gobierno Corporativo

La seguridad no se vulneró en un solo golpe maestro, sino en el eslabonamiento de debilidades de capa de aplicación, que al tener acceso sin controles compensatorios subyacentes, culminó en el control a nivel de kernel.

Solución Inmediata Obligatoria

Basta con corregir la falla de Cron (PrivEsc) para desarmar radicalmente el 80% del impacto nocivo hacia el centro de datos.

COMPROMISO A LA ACCIÓN DIRECTIVA

- Firma del plan de remediación "Bloque 1".
- Asignación de ventana de mantenimiento para rotación de credenciales.
- Monitoreo de validación por equipo Blue Team.

Fin del Engagement (Cierre Exitoso)