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

Yamit Arvey Narváez Ramírez

Asesor

Eduvin Trigos Sánchez

Universidad Nacional Abierta y a Distancia UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería – ECBTI

Especialización en seguridad informática

2025

Dedicatoria

A Dios, a mi familia que me ha acompañado en este hermoso proceso, a mis compañeros de trabajo que me han apoyado y a todos los profesionales que optan por especializarse para cumplir metas académicas y personales quienes han asumido este desafío de dar lo mejor de si, que este trabajo de grado sea un recordatorio del esfuerzo y dedicación que se ha realizado para hacer posible el sueño de ser especialistas.

Agradecimientos

Aprovecho este espacio para agradecer a mi familia y a todas esas personas que han contribuido en la realización y formación como profesional y ahora como futuro especialista en seguridad informática.

A mi familia, a mi señora madre Clara Yolima Ramírez, a mi esposa Stefania Mosquera Gutiérrez, por su apoyo contante porque a pesar de muchas dificultades siempre me han apoyado a lograr cumplir con las metas tanto a nivel académico como profesional.

A mis compañeros de trabajo que también han sido mis amigos y compañeros de estudio, Nelson Enrique Reyes Rincón y Sebastián Ospina Urrea, con quienes hace cuatro años iniciamos esta aventura académica y a pesar de estar en diferentes ciudades hemos recibido apoyo mutuo superando las dificultades que se han presentado en este proceso en el cual hemos experimentado un crecimiento personal y profesional en esta maravillosa experiencia que ha sido verdaderamente enriquecedora.

Expreso mi agradecimiento a todas las personas que, de alguna manera, aportaron a este trabajo y a mi desarrollo académico. A cada uno de ustedes, les doy las gracias por confiar en mí y por formar parte de este éxito.

Resumen

Este trabajo académico presenta un análisis integral de las capacidades técnicas, tácticas y de respuesta que deben desarrollar los equipos Red Team y Blue Team en el contexto de la ciberseguridad moderna. A partir de la experiencia adquirida en el Seminario Especializado de la UNAD, se sintetizan las cuatro etapas del curso: fundamentos estratégicos, ética profesional y marco normativo, ejecución de pruebas ofensivas y respuesta ante incidentes de seguridad. El documento aborda metodologías ampliamente reconocidas como el pentesting, la defensa en profundidad y los principios del modelo Zero Trust, incorporando herramientas clave como Metasploit, Nmap, OpenVAS y sistemas SIEM para la detección, análisis y correlación de eventos de seguridad. Asimismo, se analizan las implicaciones legales y éticas derivadas de la Ley 1273 de 2009 y los lineamientos del Consejo Profesional Nacional de Ingeniería (COPNIA), garantizando que las prácticas ofensivas y defensivas se desarrollen bajo parámetros responsables y legales. El objetivo principal es fortalecer la postura de seguridad organizacional mediante la integración de tácticas ofensivas controladas y mecanismos defensivos proactivos, contribuyendo a la mitigación de riesgos y al fortalecimiento de la resiliencia frente a ciberataques. Este informe se elabora bajo los lineamientos de la norma APA 7.ª edición, con rigor académico y soporte bibliográfico actualizado, para su incorporación en el repositorio institucional.

Palabras clave: Blue Team, ciberseguridad, pentesting, Red Team.

vulnerabilidades.

Abstract

This academic paper provides a comprehensive analysis of the technical, tactical, and response capabilities required by Red Team and Blue Team in modern cybersecurity environments. Based on the experience gained during the UNAD Specialized Seminar, the report synthesizes four key stages: strategic foundations, professional ethics and regulatory framework, offensive testing execution, and incident response. The study explores recognized methodologies such as penetration testing, defense in depth, and Zero Trust principles, incorporating essential tools like Metasploit, Nmap, OpenVAS, and SIEM systems for event detection and correlation. Furthermore, it examines legal and ethical implications under Colombian Law 1273 of 2009 and COPNIA guidelines, ensuring that offensive and defensive practices are conducted responsibly. The main objective is to strengthen organizational security posture through the integration of controlled offensive tactics and proactive defensive mechanisms, contributing to risk mitigation and resilience against cyberattacks. This report adheres to APA 7th edition standards, ensuring academic rigor and updated bibliographic support for inclusion in the institutional repository.

Keywords: Blue Team, cybersecurity, penetration testing, Red Team, vulnerabilities.

Tabla de Contenido

Lista de Figuras

Lista de Tablas

Lista de Apéndices

Glosario

Amenaza digital: situación o evento con potencial de afectar la confidencialidad, integridad o disponibilidad de los activos digitales de una organización, representando un riesgo que puede comprometer servicios, datos o procesos tecnológicos (Enterprise, n.d.).

Análisis forense digital: disciplina dedicada a la recolección, preservación y análisis de evidencia digital tras un incidente, garantizando su integridad para investigaciones técnicas o legales (Casey, 2022).

Antivirus: aplicación informática diseñada para detectar, bloquear y eliminar software malicioso, protegiendo sistemas contra virus, troyanos, spyware u otras amenazas (Pathak, 2024).

Ataque: acción intrusiva cuyo propósito es comprometer sistemas de información mediante acceso no autorizado, alteración o eliminación de recursos digitales, normalmente utilizando software malicioso (IBM, 2022).

Ataque lateral (Lateral Movement): técnica utilizada por un adversario para desplazarse dentro de la red después de comprometer un sistema inicial, aprovechando credenciales o vulnerabilidades internas (MITRE ATT&CK®, 2023).

Autenticación multifactor (MFA): mecanismo que requiere dos o más factores de verificación para validar la identidad del usuario, combinando elementos como contraseñas y códigos generados desde un dispositivo confiable (Services, 2023).

Blue Team: equipo responsable de la defensa cibernética, encargado del monitoreo, análisis y respuesta ante incidentes, así como del fortalecimiento de controles y la continuidad operativa (Ciberso, 2024; Howard & Prince, 2021).

Botnets: conjunto de dispositivos comprometidos mediante malware y controlados remotamente por un atacante sin el consentimiento de sus propietarios (Fisher, 2013).

Ciberataque: acciones maliciosas realizadas mediante medios digitales con el fin de vulnerar la disponibilidad, integridad o confidencialidad de sistemas, redes o información (IBM, 2022).

CIS Benchmarks: estándares internacionales que describen configuraciones seguras para minimizar riesgos y mejorar la postura de seguridad en sistemas y aplicaciones (Center for Internet Security, 2022).

Confianza cero (Zero Trust): modelo de seguridad basado en la premisa de no confiar en ningún usuario o dispositivo por defecto, exigiendo verificación continua para cada solicitud de acceso (IBM, 2024).

Control de accesos: conjunto de políticas y mecanismos que determinan quién puede acceder a qué recursos dentro de una organización, basado en roles y privilegios (Microsoft, 2021).

Cultura de ciberseguridad: conjunto de hábitos, valores y prácticas promovidas en una organización para fomentar un uso seguro y responsable de los sistemas tecnológicos (Grupo, 2024).

Evaluación de seguridad: procedimiento orientado a identificar vulnerabilidades o deficiencias en los controles de seguridad de una infraestructura tecnológica (Point, 2021).

Explotación (Exploit): acción de aprovechar una vulnerabilidad para ejecutar código o realizar actividades no autorizadas, tanto en pruebas de penetración como en ataques reales (Andress, 2023).

Gestión del conocimiento: práctica orientada a documentar experiencias y lecciones aprendidas en seguridad informática para mejorar la toma de decisiones y fortalecer las capacidades del personal (Vorecol, 2024).

Hardening: proceso de fortalecimiento de sistemas mediante la reducción de servicios innecesarios, aplicación de parches y adopción de configuraciones seguras (Keller & Brooks, 2022).

Incidente de seguridad: evento inesperado que afecta la disponibilidad, integridad o confidencialidad de la información o de los sistemas digitales (Hackmetrix, 2024).

Malware: software malicioso diseñado para causar daño, interrumpir operaciones o comprometer sistemas, incluyendo virus, troyanos, gusanos, keyloggers, botnets y ransomware (Pathak, 2024).

Metasploit Framework: herramienta modular utilizada para automatizar la explotación de vulnerabilidades, ejecutar payloads y validar controles de seguridad en entornos ofensivos (Raj & Walia, 2020).

Nmap: software de código abierto utilizado para el reconocimiento de redes y detección de servicios, empleado para identificar puertos abiertos o sistemas operativos (Lyon, 2009).

OpenVAS: escáner de vulnerabilidades que evalúa configuraciones inseguras y genera reportes orientados a la gestión del riesgo (Greenbone Networks, 2023).

Pentesting: prueba de penetración orientada a identificar vulnerabilidades mediante ataques controlados, evaluando el nivel de exposición de una organización (Fernández, 2022).

Phishing: técnica de ingeniería social que busca obtener credenciales o datos sensibles mediante engaños, comúnmente a través de correos electrónicos fraudulentos o suplantaciones de identidad (Pathak, 2024).

Pivoting: estrategia ofensiva en la cual un sistema comprometido se utiliza como punto de acceso para desplazarse dentro de la red interna (MITRE ATT&CK®, 2023).

Red Team: equipo encargado de simular ataques reales para evaluar la resiliencia de la infraestructura tecnológica y detectar fallas en los mecanismos defensivos (Kotwani et al., 2023; Ciberseguridad, 2021).

Respuesta ante incidentes: conjunto de acciones implementadas para contener, mitigar y recuperar la operación tras un incidente de seguridad (Microsoft, 2024).

Segmentación de red: estrategia que divide una red en segmentos independientes con el fin de limitar el movimiento de atacantes y proteger activos críticos (Zscaler, 2024).

Seguridad desde el diseño: metodología que incorpora controles de seguridad en las etapas iniciales de diseño y desarrollo de sistemas para prevenir vulnerabilidades estructurales (Wiz, 2024).

Seguridad de la información: práctica enfocada en proteger datos y sistemas frente a accesos no autorizados, alteraciones o pérdidas, garantizando la confidencialidad, integridad y disponibilidad (Innova, 2015).

Seguridad informática: conjunto de medidas destinadas a proteger redes, sistemas y datos frente a ataques o accesos no autorizados (Universidad en Internet, 2021).

SIEM (Security Information and Event Management): plataforma que centraliza registros, correlaciona eventos y genera alertas en tiempo real para apoyar la detección y respuesta ante incidentes (Chindrus & Caruntu, 2023).

Simulacro de ciberseguridad: actividad controlada que reproduce un ciberataque con el fin de evaluar la preparación y capacidad de respuesta de una organización (CERT, 2024).

Vulnerabilidad: debilidad en un sistema, proceso o software que puede ser explotada por atacantes para comprometer la seguridad (Orsys-Lemag, 2024).

Introducción

En el marco del Seminario Especializado *Equipos Estratégicos en Ciberseguridad: Red Team & Blue Team* de la Universidad Nacional Abierta y a Distancia (UNAD), tuve la oportunidad de abordar la ciberseguridad desde una perspectiva integral, que combina la ofensiva controlada y la defensa activa. Este trabajo final sintetiza los aprendizajes obtenidos en las cuatro etapas del curso, articulando conceptos teóricos, prácticas técnicas y reflexiones éticas que fortalecen mi perfil profesional en seguridad informática.

En la Etapa 1, comprendí los fundamentos estratégicos que definen la interacción entre los equipos Red Team y Blue Team, explorando metodologías como el *penetration testing* y herramientas esenciales como Metasploit, Nmap y OpenVAS, en entornos virtualizados que simulan escenarios reales de ataque y defensa (Lyon, 2009; Raj & Walia, 2020). Posteriormente, en la Etapa 2, reflexioné sobre la importancia de la ética profesional y el marco normativo colombiano, reconociendo que la práctica de pruebas ofensivas debe realizarse bajo parámetros legales y principios de transparencia, en concordancia con la Ley 1273 de 2009 y el Código de Ética del COPNIA (Congreso de la República, 2009; COPNIA, 2015).

La Etapa 3 representó un desafío técnico significativo, al ejecutar un ejercicio Red Team que consistió en explotar la vulnerabilidad CVE-2014-6287 en el servicio HttpFileServer (HFS) 2.3 mediante el módulo *rejetto_hfs_exec* de Metasploit, lo que evidenció la criticidad de las configuraciones inseguras y la necesidad de aplicar controles preventivos (Kotwani et al., 2023). Finalmente, en la Etapa 4, asumí el rol de Blue Team para implementar medidas de contención y hardening, aplicando estándares como CIS Benchmarks, segmentación de red y monitoreo con sistemas SIEM, reforzando la postura defensiva frente a ataques simulados (Center for Internet Security, 2022; Chindrus & Caruntu, 2023).

Este documento integra dichas experiencias con un enfoque académico y profesional, sustentado en literatura especializada y normas internacionales, para fortalecer las capacidades técnicas, tácticas y de respuesta en ciberseguridad. El análisis se desarrolla bajo los lineamientos de la norma APA 7ª edición, garantizando rigor metodológico y validez académica.

Justificación

La creciente sofisticación de los ciberataques y la expansión de las superficies de exposición en entornos digitales exigen profesionales capaces de comprender tanto las tácticas ofensivas como las defensivas. Este trabajo se justifica porque articula aprendizajes prácticos y teóricos adquiridos en las cuatro etapas del curso, consolidando competencias críticas para la protección de infraestructuras tecnológicas en escenarios reales.

La simulación de ataques controlados por parte del Red Team y la implementación de estrategias defensivas por el Blue Team no solo fortalecen la resiliencia organizacional, sino que también promueven la adopción de buenas prácticas alineadas con estándares internacionales como ISO/IEC 27001, CIS Benchmarks y marcos como MITRE ATT&CK (NIST, 2020; MITRE, 2023). Además, el análisis ético y normativo garantiza que las acciones se desarrollen bajo principios de legalidad y responsabilidad profesional, en concordancia con la Ley 1273 de 2009 y el Código de Ética del COPNIA (Congreso de la República, 2009; COPNIA, 2015).

Este trabajo, elaborado bajo los lineamientos de la norma APA 7ª edición, constituye un aporte académico que puede servir como referencia para futuras investigaciones y prácticas en ciberseguridad, contribuyendo a la formación de especialistas que respondan con eficacia ante los desafíos del panorama digital contemporáneo.

Objetivos

Objetivo General

Analizar las capacidades técnicas, tácticas y de respuesta que desarrollé durante el Seminario Especializado *Equipos Estratégicos en Ciberseguridad: Red Team & Blue Team*, integrando fundamentos teóricos, prácticas ofensivas y defensivas, y principios éticos y normativos, con el fin de fortalecer la postura de seguridad en entornos tecnológicos.

Objetivos Específicos

Describir los fundamentos estratégicos y metodológicos que orientan las operaciones Red Team y Blue Team, incluyendo herramientas como Metasploit, Nmap y OpenVAS, aplicadas en entornos controlados.

Evaluar las implicaciones éticas y legales en la ejecución de pruebas ofensivas y defensivas, conforme a la Ley 1273 de 2009 y el Código de Ética del COPNIA, para garantizar prácticas responsables.

Documentar el proceso de explotación controlada realizado en la Etapa 3, identificando vulnerabilidades críticas y su impacto en la infraestructura tecnológica.

Proponer medidas de hardening y estrategias de contención aplicadas en la Etapa 4, sustentadas en estándares internacionales como CIS Benchmarks y buenas prácticas de ciberseguridad.

Etapa 1. Fundamentos Estratégicos de Operaciones Red Team y Blue Team

Roles y responsabilidades red team vs blue team

En esta primera etapa del seminario, comprendí que los equipos Red Team y Blue Team cumplen funciones complementarias dentro de la estrategia global de ciberseguridad. El Red Team adopta un enfoque ofensivo, simulando ataques reales para identificar vulnerabilidades y evaluar la resiliencia de la infraestructura tecnológica. Su objetivo no es causar daño, sino exponer debilidades antes de que puedan ser explotadas por actores maliciosos (Kotwani et al., 2023). Por otro lado, el Blue Team se centra en la defensa activa y reactiva, implementando controles preventivos, monitoreo continuo y respuesta ante incidentes. Este equipo utiliza herramientas como sistemas SIEM para correlacionar eventos y generar alertas en tiempo real, fortaleciendo la postura defensiva (Chindrus & Caruntu, 2023).

Durante el desarrollo práctico, confirmé que la interacción entre ambos equipos no debe entenderse como una competencia, sino como un ejercicio colaborativo orientado a la mejora continua. Mientras el Red Team revela brechas, el Blue Team aprende a mitigarlas, aplicando principios como defensa en profundidad y segmentación de red (Howard & Prince, 2021).

Metodologías de pentesting y reglas de compromiso

El pentesting o prueba de penetración es una metodología estructurada que permite evaluar la seguridad de sistemas mediante simulación de ataques controlados. Según NIST (2020), este proceso se compone de fases claramente definidas:

- **Planificación y alcance:** Definición de activos, objetivos y limitaciones.
- **Reconocimiento:** Recolección de información sobre el objetivo.
- **Escaneo y enumeración:** Identificación de puertos, servicios y vulnerabilidades.
- **Explotación:** Validación de vulnerabilidades mediante ejecución controlada.
- **Post-explotación:** Evaluación del impacto y persistencia.

- **Informe:** Documentación técnica y recomendaciones.

En el laboratorio, apliqué estas fases respetando las reglas de compromiso, que establecen límites éticos y técnicos para evitar daños en sistemas productivos. Estas reglas son esenciales para garantizar que las pruebas se realicen bajo autorización explícita y con trazabilidad completa (Andress, 2023).

Herramientas esenciales (Metasploit, Nmap, OpenVAS)

Las herramientas utilizadas en esta etapa fueron seleccionadas por su relevancia en entornos profesionales:

- **Metasploit Framework:** Plataforma modular para explotación y post-explotación, que permite automatizar ataques controlados y validar la efectividad de los controles defensivos (Raj & Walia, 2020).
- **Nmap:** Herramienta de escaneo que facilita la identificación de hosts, puertos abiertos y servicios activos, siendo clave en la fase de reconocimiento (Lyon, 2009).
- **OpenVAS:** Escáner de vulnerabilidades que genera reportes detallados sobre configuraciones inseguras y riesgos potenciales (Greenbone Networks, 2023).

Estas herramientas fueron implementadas en un entorno virtualizado, garantizando aislamiento y seguridad durante las pruebas.

Ejemplo práctico de configuración de laboratorio.

En el laboratorio, configuré un entorno compuesto por una máquina anfitriona con VirtualBox y dos máquinas virtuales: Windows 7 y Parrot OS. El direccionamiento IP se estableció en modo puente para permitir la comunicación entre los sistemas. Esta configuración replicó un escenario corporativo básico, donde el Red Team ejecutó pruebas ofensivas y el Blue Team aplicó medidas defensivas.

Marco legal colombiano que regula los delitos informáticos.

En Colombia, el marco legal sobre delitos informáticos y protección de datos personales está conformado por diversas normas que buscan salvaguardar la información digital, la privacidad de los ciudadanos y la integridad de los sistemas informáticos.

La **Ley 1273 de 2009** es la base principal, ya que introdujo un nuevo título en el Código Penal para proteger la información y los datos. Esta ley tipifica delitos como el acceso no autorizado, la interceptación de comunicaciones, la alteración o destrucción de datos y el fraude informático, lo que implica que cualquier prueba de penetración o simulación de ataque sin autorización formal podría considerarse una conducta delictiva. Para los *equipos Red Team,* esto significa que las pruebas ofensivas deben realizarse con aprobación escrita y bajo límites claramente definidos para no infringir la ley.

Por otra parte, la **Ley 1581 de 2012** establece los principios y obligaciones para el tratamiento de datos personales, garantizando derechos como el acceso, actualización, rectificación y eliminación de la información. A esta norma la complementa el ***Decreto 1377 de 2013****,* que precisa los procedimientos para obtener autorizaciones, conservar registros y aplicar medidas de seguridad. En la práctica, los *equipos Blue Team,* encargados de la defensa y monitoreo, deben aplicar controles que aseguren el cumplimiento de esta normativa, especialmente en la gestión de incidentes y la manipulación de información sensible (Congreso de la República de Colombia, 2012; Presidencia de la República, 2013).

Además, la **Ley 1266 de 2008**, conocida como Ley de Hábeas Data Financiero, regula el manejo de datos crediticios y financieros, y la **Ley 1621 de 2013** define las funciones y límites de las actividades de inteligencia del Estado, con el fin de proteger los derechos fundamentales frente al uso de información. Estas leyes se sustentan en el *artículo 15 de la Constitución*

Política, que reconoce el derecho a la intimidad y a la protección de los datos personales. En conjunto, estas disposiciones obligan a los profesionales de la ciberseguridad a actuar bajo principios de confidencialidad, legalidad y respeto por la privacidad, tanto en actividades de defensa (Blue Team) como de ataque controlado (Red Team).

Expertos en ciberseguridad y los equipos nacionales **ColCERT** y **CSIRT-PONAL** recomiendan que cualquier ejercicio de prueba o simulación se realice con reglas claras de actuación (*rules of engagement*), bitácoras documentadas, consentimiento informado y planes de contingencia. Estas buenas prácticas no solo evitan vulnerar el marco legal colombiano, sino que fortalecen la confianza y la madurez de las operaciones de ciberseguridad dentro de las organizaciones (ColCERT, 2024; CSIRT-PONAL, 2025).

Etapas del pentesting:

Planeación y alcance (Pre-engagement / Planning)

Esto básicamente quiere decir que antes de tocar nada, acuerdas con la organización *qué* vas a probar, *hasta dónde*, *cuándo* y *quiénes* participan. Se firma el permiso (rules of engagement) y se definen límites, horas, activos incluidos, y cómo reportar hallazgos críticos. Por qué importa: evita problemas legales y hace que las pruebas sean útiles (no destruyas sistemas productivos).

Reconocimiento (Reconnaissance / Information Gathering)

Qué es: Recolectar información pública y abierta sobre el objetivo: dominios, subdominios, contactos, tecnologías detectables, e IPs. Se hace sin atacar activamente (footprinting pasivo) y con técnicas activas si está permitido.

Herramienta ejemplo / cómo usarla: *Nmap* — se usa para descubrir hosts y puertos: nmap -sS -Pn -p- target.com (escaneo de puertos TCP). Complementa con *whois*, *dnsenum*, *subfinder* o

theHarvester para recolectar correos y subdominios.

Cómo puede ayudar: suministra el mapa de superficie de ataque (qué servicios exponerán vectores de entrada).

MITRE (relación): Recon corresponde a tácticas de *Reconnaissance* y técnicas como recopilación de *Domain* y *DNS* (ATT&CK reconoce recogida de información del objetivo).

Escaneo y descubrimiento (Scanning / Vulnerability Discovery)

Qué es: Con la información del paso anterior, pruebas cada servicio para encontrar versiones, configuraciones débiles o vulnerabilidades conocidas. Aquí combinamos escaneo de puertos con escáneres de vulnerabilidades.

Herramienta ejemplo / cómo usarla: *Nessus* o *OpenVAS* — instálalos en tu entorno de testing y lanza un escaneo dirigido a las IPs en scope; revisa el informe y prioriza por CVSS / criticidad. Para web, complementa con *Nikto* o *OWASP ZAP* para encontrar inyecciones y malas configuraciones.

Nota práctica: filtra falsos positivos — no todo lo marcado es explotable; valida manualmente lo más crítico.

MITRE (relación): técnicas de *Discovery* (p. ej. enumeración de servicios y software). NIST SP 800-115 describe estas actividades como parte de la fase de información y análisis.

Análisis de vulnerabilidades (Vulnerability Analysis / Prioritization)

Qué es: Interpretar resultados del escaneo, correlacionar vulnerabilidades con contexto (si un servicio es accesible desde Internet vs. dentro de LAN), y decidir cuáles probar primero.

Herramienta ejemplo / cómo usarla: *Burp Suite* (para aplicaciones web): revisa items marcados, reproduce peticiones problemáticas y clasifícalas (alta/ media/baja). En redes, usa *Nmap* + *searchsploit* para buscar exploits públicos.

Consejo práctico: prioriza exploits que permiten acceso inicial o control (RCE, auth bypass) antes que issues de menor impacto (info disclosure).

MITRE (relación): mapea a *Initial Access* / *Privilege Escalation* dependiendo del hallazgo.

Explotación (Exploitation / Gaining Access)

Qué es: Intentar explotar las vulnerabilidades priorizadas para verificar si son reales y medir el impacto (obtener una shell, robar credenciales, etc.). Esta es la parte ofensiva propiamente dicha.

Herramienta ejemplo / cómo usarla: *Metasploit Framework* — ejemplo práctico: cargar un módulo exploit apropiado (use exploit/unix/ftp/...), configurar RHOST, RPORT y payload, ejecutar y ver si se obtiene acceso. Siempre dentro del scope y con permiso.

Precaución: evita acciones destructivas y controla la explotación (no ejecutar exploits que borren datos).

MITRE (relación): técnicas como *Exploit Public-Facing Application* (T1190) o *Command and Scripting Interpreter* (ejecución remota).

Post-explotación y escalamiento (Post-exploitation / Privilege Escalation / Lateral Movement).

Qué es: Una vez dentro, verificas hasta dónde puede llegar un atacante: escalar privilegios, recopilar credenciales, moverse lateralmente y buscar datos críticos. También consiste en mantener acceso si fuera necesario (por ejemplo, para pruebas más largas), siempre documentado.

Herramienta ejemplo / cómo usarla: *BloodHound* (para entornos Active Directory) — te ayuda a mapear relaciones entre cuentas y encontrar rutas de escalada; *Responder* o *Mimikatz* (en pruebas autorizadas y controladas) para demostrar extracción de credenciales.

Uso práctico: extrae credenciales con Mimikatz en una máquina de prueba y luego usa esas

credenciales para moverte a otros hosts; registra todo para que el Blue Team pueda reproducir.

MITRE (relación): técnicas de *Privilege Escalation* (T1068), *Credential Access* (T1555), *Lateral Movement* (T1021).

Mantenimiento de acceso / pruebas de persistencia (Persistence)

Qué es: Opcional en pruebas controladas: demostrar si un atacante puede permanecer tras reinicios o parches mediante backdoors, cuentas ocultas o cambios de configuración. Siempre debe quedar documentado y removible.

Herramienta ejemplo / cómo usarla: *Metasploit* (persistence modules) o técnicas administrativas (crear servicio, tarea programada).

Advertencia ética: esto sólo se hace si está expresamente autorizado y con un plan para eliminar artefactos.

MITRE (relación): técnica *Persistence* (T1543, T1546).

Exfiltración / impact assessment

Qué es: Simular extracción de datos (sin realmente robar datos sensibles en entornos productivos) para demostrar riesgo: qué tipo de información estaría en riesgo y cómo lo haría un atacante.

Herramienta ejemplo / cómo usarla: *Netcat* para simular canales de exfiltración (o técnicas de exfiltración documentadas en MITRE). En pruebas, usa *fichas dummy* o datos ficticios.

Buenas prácticas: nunca transferir datos reales sin autorización; usar muestras sintéticas.

MITRE (relación): *Exfiltration* (p. ej. T1041).

Limpieza y remediación (Cleanup / Remediation)

Qué es: Eliminar herramientas, shells, cuentas y cualquier rastro que se haya dejado; ayudar al cliente a parchear y cerrar brechas. Es el paso que demuestra profesionalismo.

Cómo hacerlo (ejemplo): revocar cuentas creadas, borrar binarios subidos, restaurar

configuraciones y dejar reportes de acciones y comandos ejecutados.

MITRE (relación): contramedidas y mitigaciones aplicadas a las técnicas detectadas.

Reporte y seguimiento (Reporting / Post-test activities / Re-scan)

Qué es: Entregar un informe técnico + ejecutivo que explique hallazgos, evidencia, riesgo (impacto y probabilidad) y recomendaciones claras para mitigación y priorización. Idealmente se hace un re-scan tras las correcciones.

Contenido mínimo del reporte: resumen ejecutivo, lista priorizada de vulnerabilidades (con CVE y evidencia), paso a paso de explotación (solo en la versión técnica), y plan de remediación.

Herramienta ejemplo / cómo usarla: *Dradis* o plantillas basadas en NIST/SP800-115 para organizar evidencia; usar capturas, logs y PoC controladas.

MITRE (relación): usa ATT&CK para recomendar detecciones específicas y reglas de detección en EDR/IDS.

Herramientas de ciberseguridad:

Metasploit: Metasploit Framework es una plataforma modular de código abierto, desarrollada principalmente en Ruby, diseñada para la creación, prueba y ejecución de exploits y payloads durante evaluaciones de seguridad (penetration testing) y actividades de investigación de vulnerabilidades. Su diseño modular —con módulos para exploits, payloads, auxiliares y post-explotación— facilita la automatización de escenarios de ataque controlados y la replicación de técnicas usadas por actores malintencionados, con el fin de validar controles y mejorar la postura defensiva de una organización.

Arquitectura y componentes principales: La arquitectura de Metasploit se organiza alrededor de varios tipos de módulos: (a) exploits, que aprovechan vulnerabilidades concretas; (b) payloads, que definen el código que se ejecutará en el objetivo (por ejemplo, Meterpreter);

(c) auxiliary, que ofrece funciones como escaneo y fuzzing; y (d) post modules, orientados a tareas de recolección, escalamiento de privilegios y persistencia tras una intrusión controlada. Esta separación promueve reutilización y extensibilidad, permitiendo a investigadores y equipos Red/Blue combinar y adaptar módulos según la campaña de prueba.

Meterpreter y el manejo de payloads: meterpreter es un payload avanzado y dinámicamente extensible que opera en memoria, proporcionando una sesión interactiva con el sistema comprometido y una API potente para ejecutar comandos, cargar extensiones y mover procesos (pivoting). Meterpreter permite operaciones de post-explotación como recolección de información del sistema, volcado de hashes y migración de procesos, lo que lo convierte en una herramienta central dentro de los escenarios de explotación realistas. Además, Metasploit mantiene una amplia colección de payloads (reverse_tcp, reverse_https, staged/unstaged y nuevos enfoques como *fetch payloads*), lo que permite adaptar la conexión y la evasión a las condiciones operativas del objetivo.

Flujo de trabajo en pruebas de intrusión (Red Team) — síntesis práctica: Un flujo típico incluye: (1) reconocimiento y escaneo (módulos *auxiliary*); (2) selección del exploit y del payload adecuados a la versión/servicio detectado; (3) configuración de parámetros (RHOSTS, LHOST, puerto, opciones específicas del módulo); y (4) ejecución y manejo de sesiones (handlers que reciben conexiones entrantes). Metasploit facilita además la persistencia de sesiones y la ejecución de módulos de post-explotación para recopilar evidencia y medir impacto. Estas capacidades permiten replicar ataques reales en un entorno controlado y documentar hallazgos con reproducibilidad.

Aplicación para Blue Team y valor pedagógico: aunque Metasploit es conocido por su uso en Red Team, tiene un rol pedagógico y operativo para equipos defensivos. Los Blue Teams usan Metasploit en laboratorios controlados para: (a) validar reglas de detección (SIEM,

IDS/IPS, EDR); (b) generar telemetría que permita ajustar firmas y correlaciones; y (c) formar a analistas en técnicas de post-explotación que deben reconocer y mitigar. Por ello, el conocimiento profundo de Metasploit mejora las capacidades de detección y respuesta. Además, la comunidad y la documentación permiten a educadores construir ejercicios prácticos (p. ej., máquinas vulnerables como Metasploitable) para enseñanza.

Fortalezas, limitaciones y consideraciones éticas-legales: entre sus fortalezas destacan la modularidad, la comunidad activa y la documentación oficial; entre las limitaciones, el riesgo de uso indebido y la posibilidad de que resultados en laboratorio no reflejen todas las variables de producción (falsas conclusiones si no se diseñan escenarios realistas). Éticamente, Metasploit debe utilizarse únicamente con **autorización explícita** (contrato o permiso por escrito) y en entornos aislados o de laboratorio; su uso no autorizado es ilícito y contrario a la ética profesional. En trabajos académicos y de grado, es imprescindible documentar el entorno de pruebas (IPs privadas, snapshots, alcance autorizado) para garantizar trazabilidad y cumplimiento legal.

Nmap: (Network Mapper) es una herramienta libre y de código abierto para el descubrimiento de hosts y la auditoría de seguridad en redes. Fue creada por Gordon “Fyodor” Lyon y se utiliza ampliamente para identificar hosts activos, puertos abiertos, servicios y versiones, sistemas operativos y características de red. Nmap combina técnicas de sondeo (scanning) de bajo y alto nivel, un motor de detección de servicios y sistema operativo, y un rico ecosistema de scripts (NSE — Nmap Scripting Engine) que permiten automatizar tareas desde detección simple hasta comprobaciones avanzadas de vulnerabilidades y configuraciones erróneas. (Lyon, 2009; Nmap Project, s. f.).

Arquitectura y componentes principales: la funcionalidad central de Nmap se puede desglosar en tres componentes principales: (1) motor de escaneo —encargado de generar y enviar paquetes con distintos tipos de sondas (SYN, ACK, UDP, ICMP, etc.) y de analizar respuestas—; (2) detección de servicios/so —módulos que correlacionan respuestas con bases de firmas para identificar servicios, versiones y sistemas operativos; y (3) Nmap Scripting Engine (NSE) —un framework que permite extender Nmap con scripts en Lua para tareas como enumeración, explotación ligera, comprobaciones de configuración y recolección de información avanzada. Junto con la utilidad de línea de comandos nmap, el proyecto ofrece ndiff (comparación de escaneos), ncat (herramienta de red) y zenmap (GUI), formando un conjunto integral para auditorías. (Lyon, 2009; Nmap Project, s. f.).

Tipos de escaneo y opciones relevantes: Nmap soporta numerosos tipos de escaneo y opciones que deben ser seleccionadas según el objetivo y las restricciones del entorno:

- **-sS** (SYN scan): escaneo “half-open” rápido y sigiloso.
- **-sT** (TCP connect): cuando no se dispone de privilegios para raw sockets.
- **-sU** (UDP scan): para detectar servicios UDP.
- **-sV**: detección de versión de servicio.
- **-O**: detección de sistema operativo.
- **-A**: modo agresivo (combina -sV, -O, scripts y traceroute).
- **--script**: ejecución de scripts NSE (ej.: --script vuln o --script http-enum).
- **-T0..-T5**: control de timing (velocidad vs. ruido/detección).
- **-Pn**: desactivar ping previo (útil si ICMP bloqueado).

  La elección de estas opciones define precisión, tiempo de ejecución y nivel de “ruido” que genera el escaneo en la red objetivo.

Uso para Blue Team y valor defensivo: para defensores, Nmap es una herramienta valiosa para validar el inventario de activos, detectar puertos y servicios no autorizados, verificar el endurecimiento de sistemas y testear reglas de segmentación. Los Blue Teams ejecutan escaneos controlados (y comparan con inventarios) para detectar desviaciones, y utilizan Nmap para simular técnicas de reconocimiento y así ajustar alertas en SIEM y firmas IDS/IPS. Además, la familia de scripts NSE permite generar escenarios que prueban detección de misconfiguraciones (p. ej., servicios con versiones vulnerables) y evaluar la eficacia de controles de red. (Nmap Project, s. f.).

Fortalezas y limitaciones: fortalezas: Nmap es altamente flexible, eficiente para inventario y descubrimiento, y extensible mediante NSE; su amplia adopción y documentación lo hacen un estándar en auditorías de red. Limitaciones: los escaneos activos pueden ser detectados por soluciones defensivas (EDR, IDS/IPS) y, en entornos productivos, pueden provocar impactos si se realizan sin precauciones (ej.: escaneos UDP intensivos). Además, la detección de versiones y OS no siempre es concluyente; los resultados dependen del entorno, firewalls y evasiones. Es imprescindible interpretar hallazgos dentro del contexto y corroborarlos con pruebas adicionales. (Lyon, 2009).

Consideraciones éticas y legales: al igual que con otras herramientas de seguridad, Nmap solo debe utilizarse con autorización explícita. Realizar escaneos no autorizados puede constituir acceso ilícito o preparación para delitos informáticos según la jurisdicción. Para trabajos académicos y pruebas de grado se recomienda documentar alcance (IPs, ventanas horarias), usar entornos de laboratorio o redes aisladas, y conservar registros que demuestren autorización. Además, minimizar impacto (timing, limitación de puertos escaneados) reduce riesgos operativos.

OpenVas: OpenVAS (Open Vulnerability Assessment Scanner) es el motor de escaneo de vulnerabilidades de código abierto que forma parte del conjunto Greenbone Vulnerability Management (GVM). Provee capacidades de escaneo tanto no autenticado como autenticado, ejecución de pruebas de vulnerabilidad (Vulnerability Tests — VTs o NVTs), generación de reportes y gestión de resultados mediante componentes que orquestan y programan tareas de escaneo. En los últimos años Greenbone ha reordenado su portafolio y el término OpenVAS aparece como el nombre del scanner dentro del marco más amplio GVM / Greenbone.

Arquitectura y componentes principales

La solución se compone de varios elementos que trabajan en conjunto:

OpenVAS Scanner (ospd-openvas / openvas-scanner): el motor que ejecuta los tests (VTs/NVTs) contra los objetivos.

GVMD (Greenbone Vulnerability Manager Daemon): el servicio que gestiona tareas, usuarios, políticas y consolida resultados; expone la API GMP para orquestación.

GSA (Greenbone Security Assistant / gsad): interfaz web para crear tareas, ver reportes y gestionar la plataforma.

Feeds: el scanner utiliza feeds de vulnerabilidades (Greenbone Community Feed gratuito y Greenbone/Enterprise Feed comercial) que se actualizan regularmente para mantener la librería de pruebas.

Esta separación permite: planificación y programación en gvmd, ejecución escalable de escaneos por openvas-scanner y visualización/descarga de reportes desde GSA. Para integraciones y automatización, se usan gvm-cli, la API GMP o protocolos OSP/GMP según el despliegue.

Consideraciones éticas y legales

Como cualquier herramienta de evaluación, **solo** usar con autorización explícita. Registrar alcance, responsables y evidencia del permiso, y ejecutar primero en entornos de laboratorio o durante ventanas autorizadas para evitar impactos. Documentar configuraciones, versiones de feed y políticas aplicadas para reproducibilidad y auditoría.

Servicios en línea:

ExploitDB: Exploit Database (Exploit-DB) es un archivo público y CVE-compatible que recopila exploits, pruebas de concepto (PoC), shellcode, papers de seguridad y la Google Hacking Database (GHDB) para la comunidad de seguridad. Mantenido por Offensive Security, su objetivo es proporcionar a pentesters, investigadores y educadores una colección accesible y actualizable de recursos accionables que faciliten el análisis y la replicación de vulnerabilidades.

Arquitectura y componentes principales

Exploit-DB ofrece dos formas principales de acceso: (1) interfaz web (exploit-db.com) para búsquedas avanzadas por CVE, plataforma, autor, tipo de exploit y filtros; y (2) repositorio descargable y herramienta CLI llamada SearchSploit, que permite realizar búsquedas locales sin conexión y copiar exploits o PoC a un directorio de trabajo. El repositorio también incluye la GHDB (colección de dorks) y una sección de papers académicos y técnicos. Además, el proyecto mantiene estadísticas y metadatos (fechas, autores, tags) que facilitan la priorización y clasificación de hallazgos.

Flujo de trabajo (Red Team / pentesting)

En un proceso típico de evaluación, Exploit-DB se usa como fuente secundaria de exploits y PoC tras la identificación de una vulnerabilidad (por ejemplo, tras correlacionar un CVE con un servicio/version detectada por Nmap). El flujo práctico es: (1) identificar CVE o firma de

servicio; (2) buscar exploit/Poc en Exploit-DB (web o searchsploit); (3) revisar el código y requisitos del exploit (entorno, dependencias, versión exacta); (4) adaptar el exploit a un laboratorio controlado; y (5) ejecutar pruebas en entornos autorizados, documentando resultados y artefactos para reporte. SearchSploit permite además llevar una copia local del fichero de exploits para auditorías en redes aisladas.

Uso para Blue Team y valor defensivo

Para defensores, Exploit-DB es valioso por dos motivos: primero, proporciona PoC concretos que permiten validar si una detección (SIEM, EDR) estaría realmente cubierta ante el exploit real; segundo, ayuda a priorizar remediación cuando existe un exploit público disponible que facilita la explotación automática. Los Blue Teams suelen descargar PoC relevantes, reproducirlos en entornos de laboratorio y ajustar reglas de detección y playbooks de respuesta con base en la telemetría generada. También la GHDB ofrece dorks que orientan sobre exposición accidental de información sensible en buscadores.

Fortalezas y limitaciones

Fortalezas:

Accesibilidad y amplitud: colección extensa de exploits y PoC aportados por la comunidad y filtrados por metadatos.

Modo offline: SearchSploit permite trabajar en entornos aislados y mantener copia local del repositorio.

Limitaciones / riesgos:

Calidad variable: los exploits y PoC provienen de múltiples autores; algunos requieren ajustes, otros pueden estar desactualizados o no funcionar fuera del contexto exacto (sistema operativo, versión, configuración). Es imprescindible revisar el código y entender dependencias antes de ejecutar.

Facilidad para el abuso: al ser pública y accionable, la base de datos también es consultada por actores maliciosos; por tanto, su uso debe reforzarse con buenas prácticas legales y éticas.

Consideraciones éticas y legales

Exploit-DB es una herramienta de referencia; **su uso debe acatar autorización explícita**. Ejecutar exploits o PoC sin permiso puede constituir acceso ilícito o daños a sistemas, y conlleva responsabilidad legal. En contextos académicos, documenta siempre: (a) alcance autorizado; (b) entornos de laboratorio utilizados; (c) versiones exactas y cambios aplicados a PoC; y (d) evidencia de permisos. Para pruebas en producción se recomienda replicar primero en clones o snapshots y coordinar ventanas de mantenimiento.

CVE (Common Vulnerabilities and Exposures)

Es un diccionario público de identificadores únicos (CVE IDs) asignados a vulnerabilidades de seguridad divulgadas públicamente. Su propósito es proporcionar nombres comunes estandarizados para vulnerabilidades, de modo que distintas herramientas, bases de datos y equipos de seguridad puedan referirse a la misma falla sin ambigüedad. El programa CVE es mantenido como iniciativa comunitaria por MITRE bajo la supervisión de agencias gubernamentales y la comunidad de seguridad.

Arquitectura, actores y proceso de asignación

Los componentes y actores principales del ecosistema CVE son:

CVE List / CVE Program (MITRE): la lista central y el programa que coordina el sistema; mantiene políticas, la asignación de bloques y la infraestructura pública para la búsqueda de CVE.

CVE Numbering Authorities (CNAs): organizaciones autorizadas para asignar CVE IDs dentro de un ámbito determinado (vendors, proyectos de código abierto, CERTs,

proveedores de bug bounty). Los CNAs agilizan la asignación de IDs y publican los registros asociados.

NVD (National Vulnerability Database) y otras bases de datos secundarias: consumen la lista CVE y enriquecen cada registro con metadatos adicionales (CVSS, configuraciones afectadas, referencias, explotación conocida), proporcionando contexto útil para priorización y remediación. CVE y NVD son programas distintos pero complementarios.

El proceso típico para que una vulnerabilidad obtenga un CVE incluye identificación (investigador/vendedor), solicitud de CVE (directa a un CNA o a MITRE cuando no existe CNA aplicable), validación y publicación del registro con su identificador único (por ejemplo, CVE-2025-12345). Las CNAs siguen reglas operativas definidas por el programa para mantener consistencia.

Etapa 2 – Ética Profesional y Marco Normativo

Principios éticos en ciberseguridad

En esta segunda etapa del seminario, reflexioné sobre la importancia de la ética profesional en el ejercicio de la ciberseguridad. Comprendí que las pruebas ofensivas, como las realizadas por el Red Team, deben ejecutarse bajo principios de transparencia, responsabilidad y respeto por la legalidad. El hacking ético no se limita a la aplicación técnica, sino que implica un compromiso moral con la protección de la información y la privacidad de los usuarios (Howard & Prince, 2021).

Los principios éticos que guían esta práctica incluyen la honestidad, la confidencialidad y la responsabilidad social, asegurando que las acciones no generen daños colaterales ni vulneren derechos fundamentales. Tal como señala el Código de Ética del COPNIA, el profesional debe actuar con rectitud moral y abstenerse de participar en actividades ilícitas o contrarias al interés público (COPNIA, 2015).

Ley 1273 de 2009 y delitos informáticos

Durante el análisis normativo, identifiqué que la Ley 1273 de 2009 constituye el marco legal fundamental para la protección de la información en Colombia. Esta ley tipifica conductas como el acceso abusivo a sistemas informáticos, la interceptación de datos y la violación de información personal, estableciendo sanciones penales para quienes incurran en estas prácticas (Congreso de la República, 2009).

En el contexto del seminario, comprendí que cualquier actividad de pentesting sin autorización podría considerarse un delito según esta normativa. Por ello, la planificación de pruebas ofensivas debe incluir acuerdos formales y trazabilidad completa, garantizando que las acciones se realicen bajo consentimiento informado y con fines legítimos (Andress, 2023).

Código de Ética COPNIA aplicado a Red/Blue Team

El Código de Ética del COPNIA refuerza la obligación del profesional de actuar con independencia moral, rechazando presiones económicas o políticas que lo lleven a vulnerar la ley. En las actividades del Red Team y Blue Team, este principio se traduce en la necesidad de priorizar la seguridad y el bienestar social sobre cualquier interés particular (COPNIA, 2015).

Aplicar este código implica documentar cada acción, respetar la confidencialidad de la información y garantizar que las pruebas no se conviertan en mecanismos de espionaje o sabotaje. Esta reflexión fue clave para comprender que la ciberseguridad no es solo un desafío técnico, sino también un compromiso ético con la sociedad digital.

Análisis crítico del caso SecureNova Labs.

En el estudio del caso SecureNova Labs, evidencié cláusulas contractuales que promovían la omisión de denuncias frente a actividades ilícitas, lo cual contradice los principios éticos y la normativa vigente. Aceptar condiciones que impidan reportar delitos informáticos no solo vulnera la Ley 1273, sino que compromete la responsabilidad penal del profesional. Este análisis reafirmó mi convicción de que la ética debe prevalecer sobre cualquier incentivo económico, incluso cuando se ofrecen contratos atractivos o beneficios permanentes (Guarnizo Portela, 2024).

Etapa 3 – Ejecución práctica del red team

Explotación controlada (CVE-2014-6287)

En esta tercera etapa del seminario, asumí el rol de Red Team para ejecutar una actividad práctica orientada a validar vulnerabilidades en un entorno controlado. El objetivo fue comprometer un sistema Windows 7 mediante la explotación de la vulnerabilidad CVE-2014-6287, presente en el servicio HttpFileServer (HFS) versión 2.3. Esta falla permite la ejecución remota de comandos sin autenticación, lo que representa un riesgo crítico en entornos corporativos (Kotwani et al., 2023).

Para llevar a cabo la prueba, configuré un laboratorio virtual utilizando Oracle VirtualBox, con dos máquinas virtuales: una con Windows 7 como objetivo y otra con Parrot OS para ejecutar las herramientas ofensivas. Esta arquitectura replicó un escenario realista, garantizando aislamiento y seguridad durante la práctica.

Uso del módulo Metasploit rejetto_hfs_exec

La explotación se realizó mediante el framework Metasploit, una herramienta ampliamente utilizada en pruebas de penetración por su capacidad modular y automatización (Raj & Walia, 2020). Seleccioné el módulo exploit/windows/http/rejetto_hfs_exec, configurando los parámetros RHOST (IP del objetivo) y RPORT (puerto 80) para establecer la conexión. Posteriormente, ejecuté el payload windows/meterpreter/reverse_tcp, que me permitió obtener una sesión remota en el sistema comprometido.

Este procedimiento evidenció la importancia de aplicar parches y controles preventivos, ya que una vulnerabilidad sin corregir puede otorgar acceso total al atacante. Documenté cada comando ejecutado y los resultados obtenidos para garantizar trazabilidad y reproducibilidad, siguiendo las buenas prácticas recomendadas por NIST (2020).

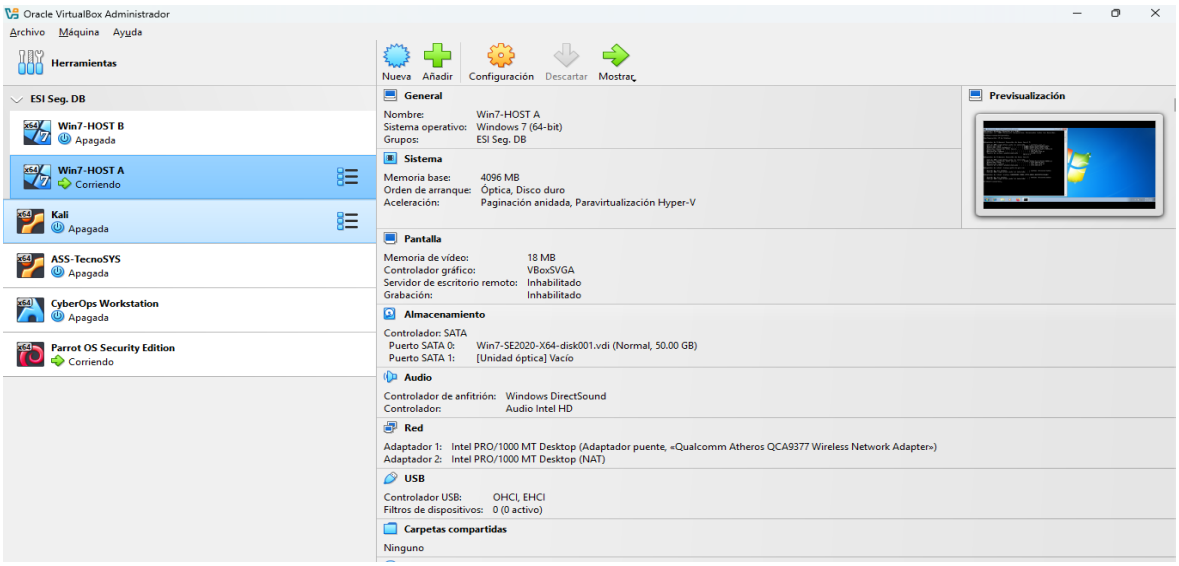
Evidencias del ataque y resultados obtenidos

Durante la explotación, capturé pantallas que demuestran el éxito del ataque y la interacción con la sesión Meterpreter. Estas evidencias son fundamentales para el análisis posterior y la elaboración del informe técnico, a continuación, procedo con la demostración del proceso realizado teniendo en cuenta las maquinas virtualizadas suministradas por el tutor.

Preparación del escenario

Figura 1

Entorno VirtualBox con las máquinas virtuales instaladas



Fuente. Autoría Propia

Nota: Interfaz de Oracle VirtualBox mostrando la configuración de la máquina virtual Win7-HOST A. Se observa el sistema operativo Windows 7 (64 bits), memoria asignada de 4096 MB, disco virtual de 50 GB y adaptadores de red configurados.

Tabla 1

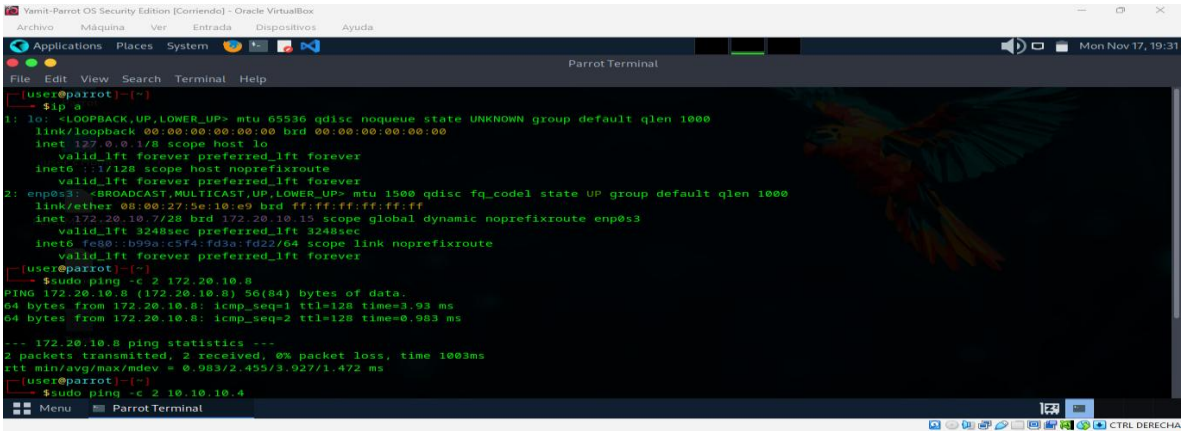
Descripción hardware máquina virtual Kali Linux

Nombre	Parrot OS Security Edition
Sistema Operativo	Debian 12 Bookworm (64-bit)
RAM	524 MB
Procesadores	1 CPU
Orden de arranque	Disco duro
Virtualización	VT-x/AMD-V, PAE/NX, Paravirtualización KVM, Paginación anidada
Memoria de video	128 MB
Controlador gráfico	VMSVGA
Aceleración 3D	Habilitado
Disco virtual	Parrot-security-6.3.amd64-disk001.vdi (64 GB)
Unidad óptica	Vacío
Audio	Controlador ICH AC97
Red	Intel PRO/1000 MT Desktop (Adaptador puente, Qualcomm Atheros QCA9377)
USB	Controlador OHCI, EHCI

Nota. Esta tabla muestras las condiciones del hardware máquina Windows

Figura 2

Confirmación direccionamiento IP en el SO Parrot

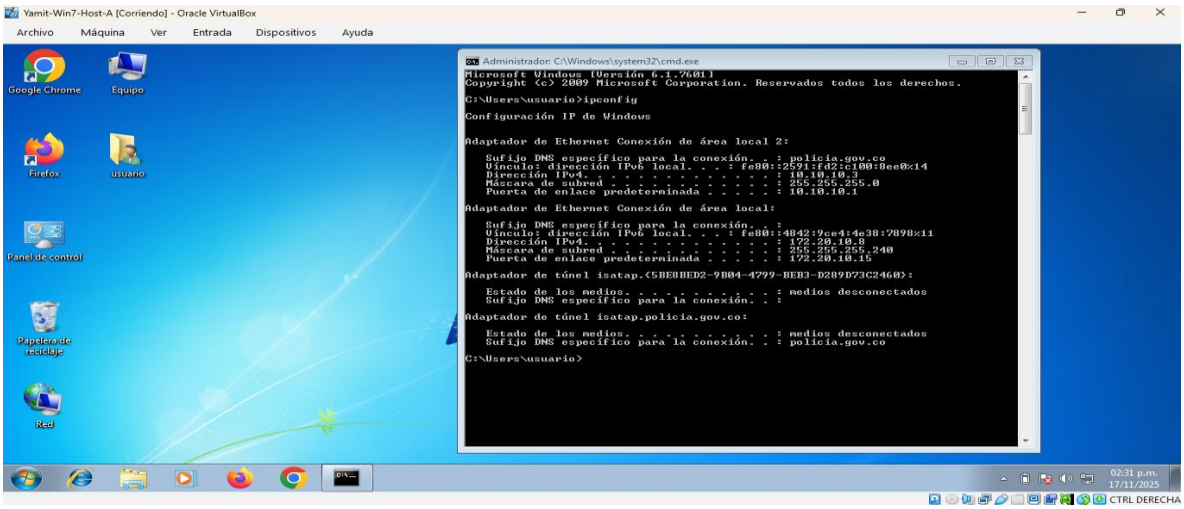


Fuente. Autoría Propia

Observamos la IP 172.20.10.7 asignada al equipo atacante Parrot OS mediante adaptador puente, confirmando su presencia en el mismo segmento de red que Host-A, lo cual permite iniciar el reconocimiento activo.

Figura 3

Visualización de Adaptadores de Red y Configuración IP en Windows 7 Host A

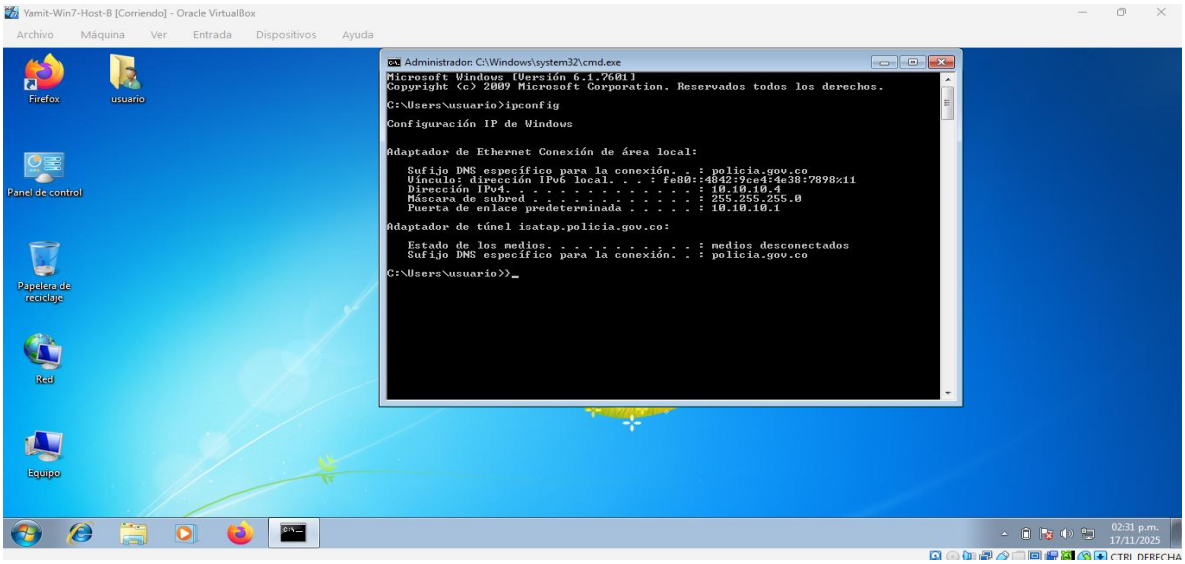


Fuente. Autoría Propia

La imagen muestra la ejecución del comando ipconfig en la consola de Windows 7 para obtener la configuración de red. Se observan adaptadores Ethernet y túneles, con detalles como sufijos DNS, direcciones IPv4, máscara de subred y estado de conexión. Esta información permite identificar parámetros de red activos y medios desconectados.

Figura 4

Consulta de Configuración IP en Windows 7 host B



Fuente. Autoría Propia

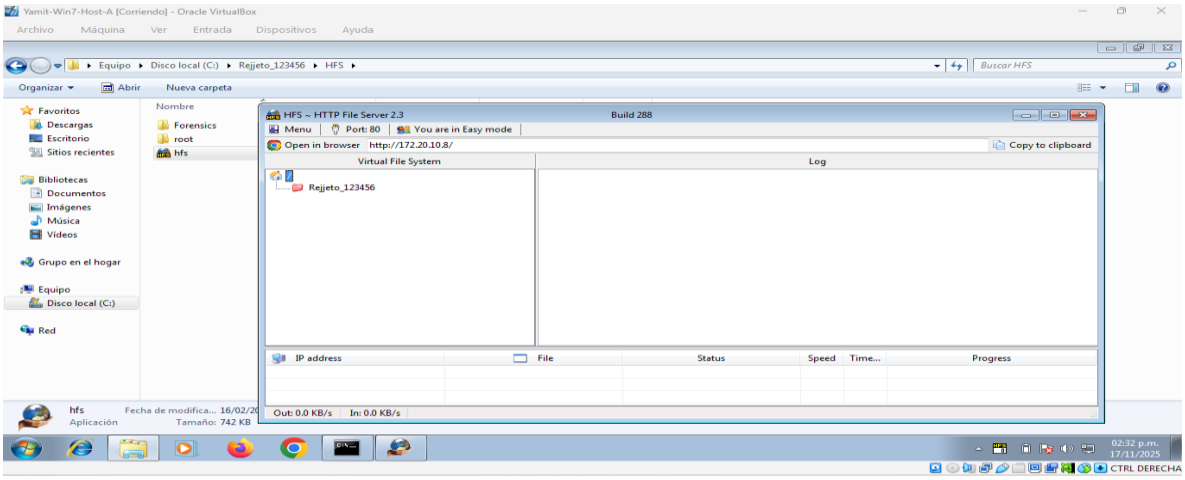
La imagen muestra la ejecución del comando ipconfig en la consola de Windows 7 para visualizar la configuración de red. Se observa un adaptador Ethernet con detalles como sufijo DNS, dirección IPv4, máscara de subred y puerta de enlace predeterminada, además del estado de adaptadores de túnel que aparecen desconectados.

Verificación REJETTO host

1) Instalar y configurar HFS (Rejetto) en Host-A (puerto 80)

Figura 5

Interfaz de Servidor HTTP en Windows 7 con HFS instalación Host-A



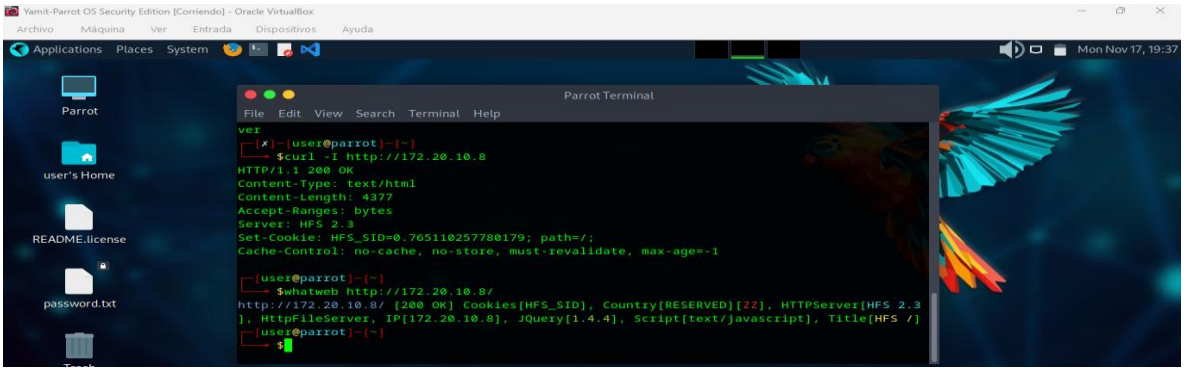
Fuente. Autoría Propia

La imagen muestra la ejecución de la aplicación HFS (HTTP File Server) en Windows 7, configurada para compartir archivos mediante el protocolo HTTP. Se observa la interfaz principal con el sistema de archivos virtual, donde aparece la carpeta Rojito_123456, y la dirección de acceso remoto http://172.20.18.8/ en el puerto 80.

2) Reconocimiento y enumeración (fase de pentesting)

Figura 6

Descubrimiento de servicios en Host-A (desde Kali)

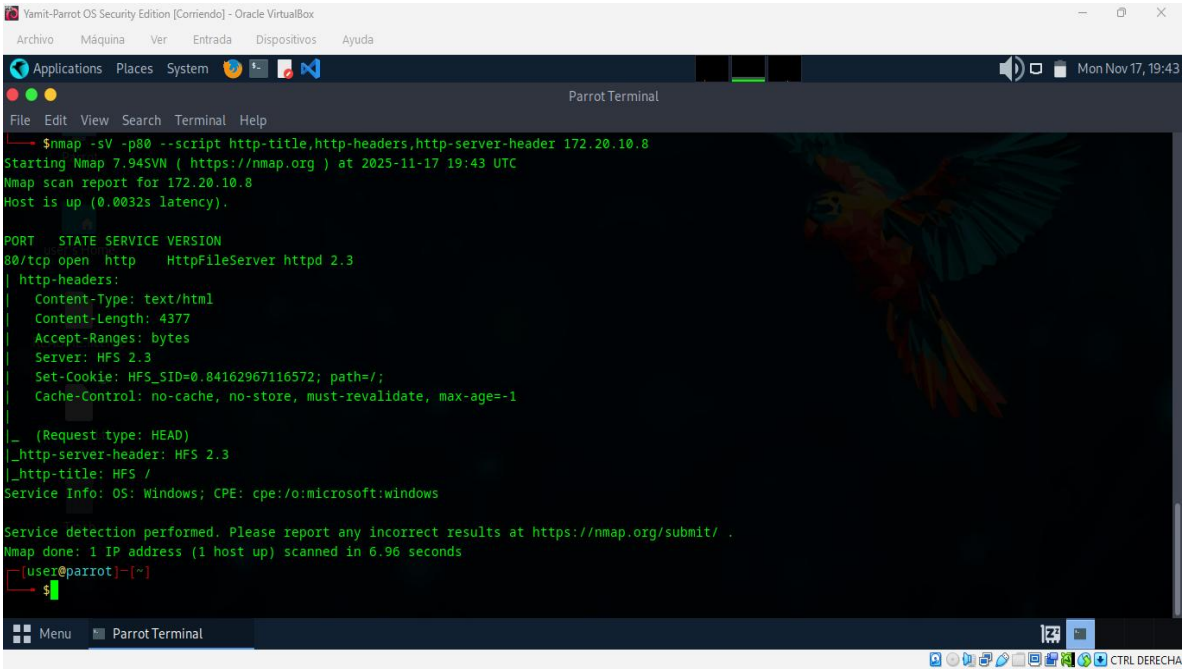


Fuente. Autoría Propia

La imagen muestra la ejecución de comandos en la terminal de Parrot OS para analizar el servicio HTTP alojado en la dirección http://172.20.10.8. Se observa la respuesta del servidor con código 200 OK, encabezados como Content-Type: text/html, y la identificación del servidor HFS 2.3 mediante la herramienta whatweb, que también detecta tecnologías como JQuery y scripts en JavaScript.

Figura 7

Escaneo HTTP con Nmap en Parrot OS para verificar puertos abiertos

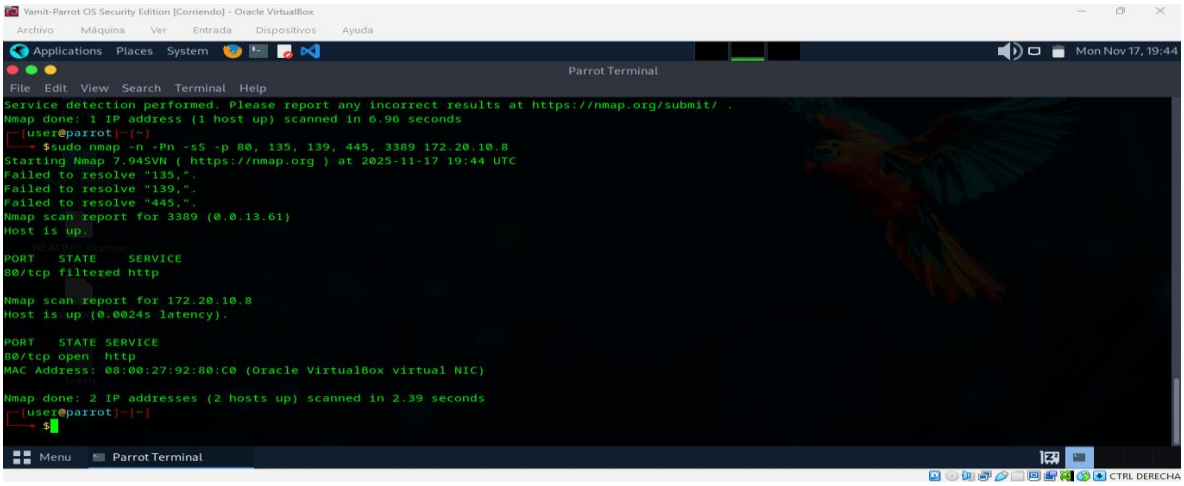


Fuente. Autoría Propia

Se muestra un análisis realizado con Nmap al host 172.20.10.8, donde se identifica el puerto 80 abierto y el servicio HttpFileServer (HFS) 2.3 sobre Windows. El resultado incluye encabezados HTTP y el título del servidor, información clave para la enumeración y detección de vulnerabilidades.

Figura 8

Escaneo de puertos con Nmap en Parrot OS

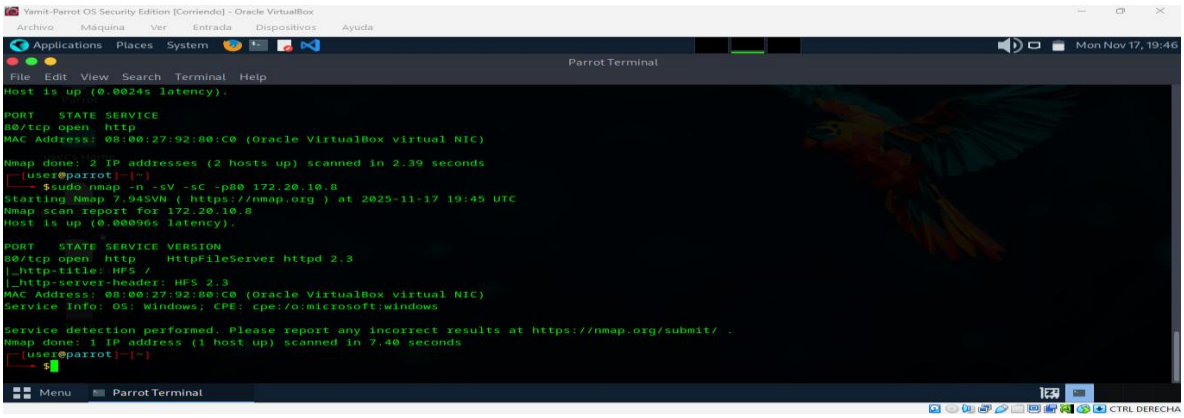


Fuente. Autoría Propia

La imagen muestra un análisis con Nmap para identificar puertos abiertos en el host 172.20.10.8. El resultado indica que el puerto 3389 está filtrado y el puerto 80 abierto, asociado a una interfaz de red virtual de Oracle VirtualBox. Esta información es relevante para la enumeración de servicios y la evaluación de posibles vectores de ataque.

Figura 9

Escaneo de puertos con Nmap en Parrot OS al servicio HttpFileServer (HFS) 2.3



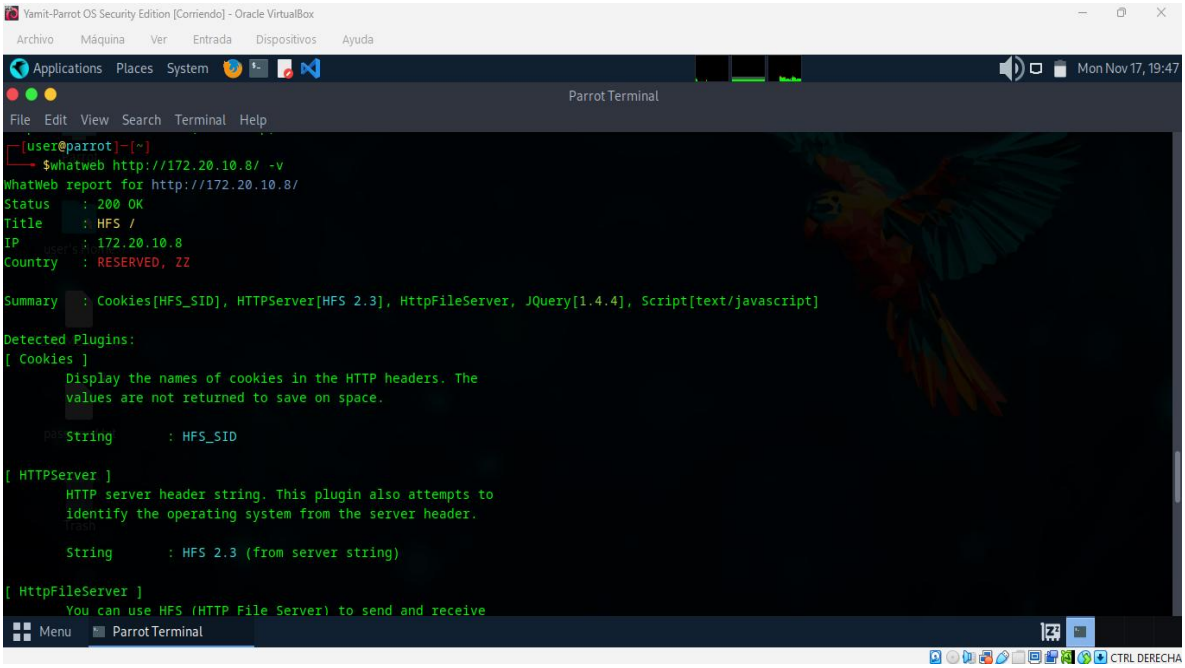
Fuente. Autoría Propia

La imagen muestra un escaneo con Nmap al host 172.20.10.8, identificando el puerto 80 abierto con el servicio HttpFileServer (HFS) 2.3 sobre Windows. El análisis incluye encabezados HTTP y confirma la interfaz de red virtual de Oracle VirtualBox, información esencial para la evaluación de vulnerabilidades.

Detección de vulnerabilidades

Figura 10

Ejecución del Comando whatweb

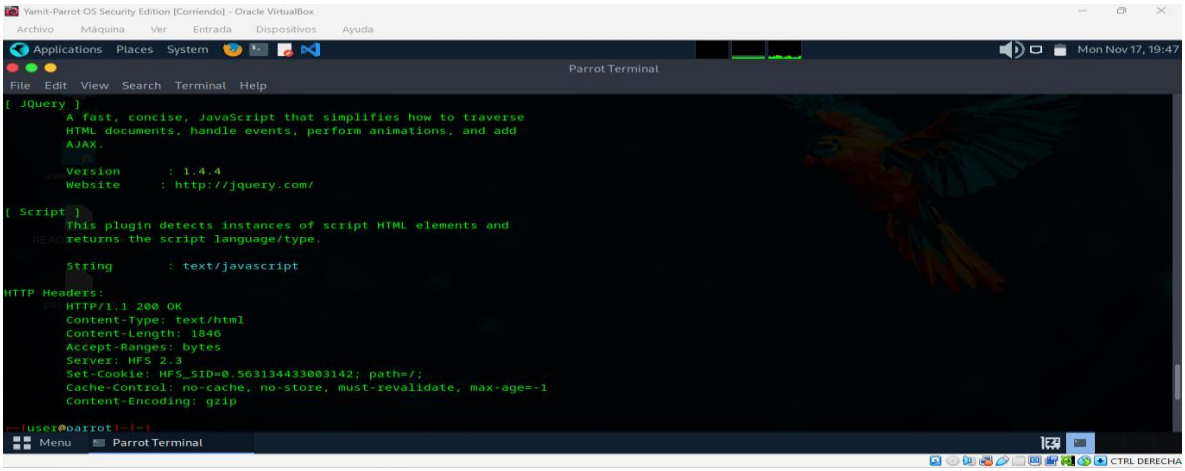


Fuente. Autoría Propia

La imagen muestra el uso de la herramienta WhatWeb para analizar el host 172.20.10.8. El resultado indica que el servidor utiliza HttpFileServer (HFS) 2.3, con tecnologías como jQuery 1.4.4 y scripts en JavaScript. También se detectan cookies y encabezados HTTP, información útil para la identificación de componentes y posibles vulnerabilidades.

Figura 11

Resultados whatweb

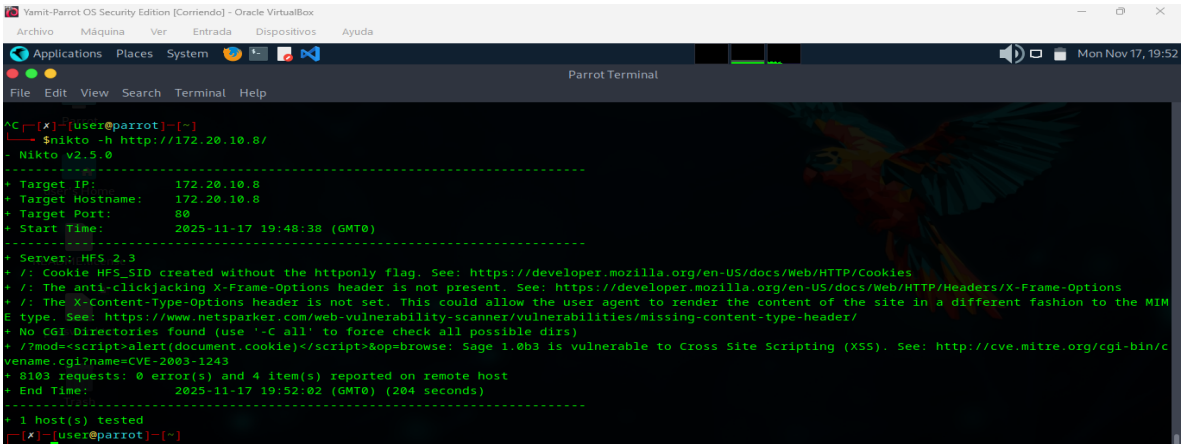


Fuente. Autoría Propia

La imagen evidencia el uso de WhatWeb para identificar tecnologías en el host 172.20.10.8. Se detecta jQuery 1.4.4, scripts en JavaScript y el servidor HttpFileServer (HFS) 2.3, junto con encabezados HTTP como Content-Type y Cache-Control. Esta información permite conocer componentes del entorno web y evaluar posibles vulnerabilidades asociadas.

Figura 12

Análisis de vulnerabilidades con Nikto en Parrot OS



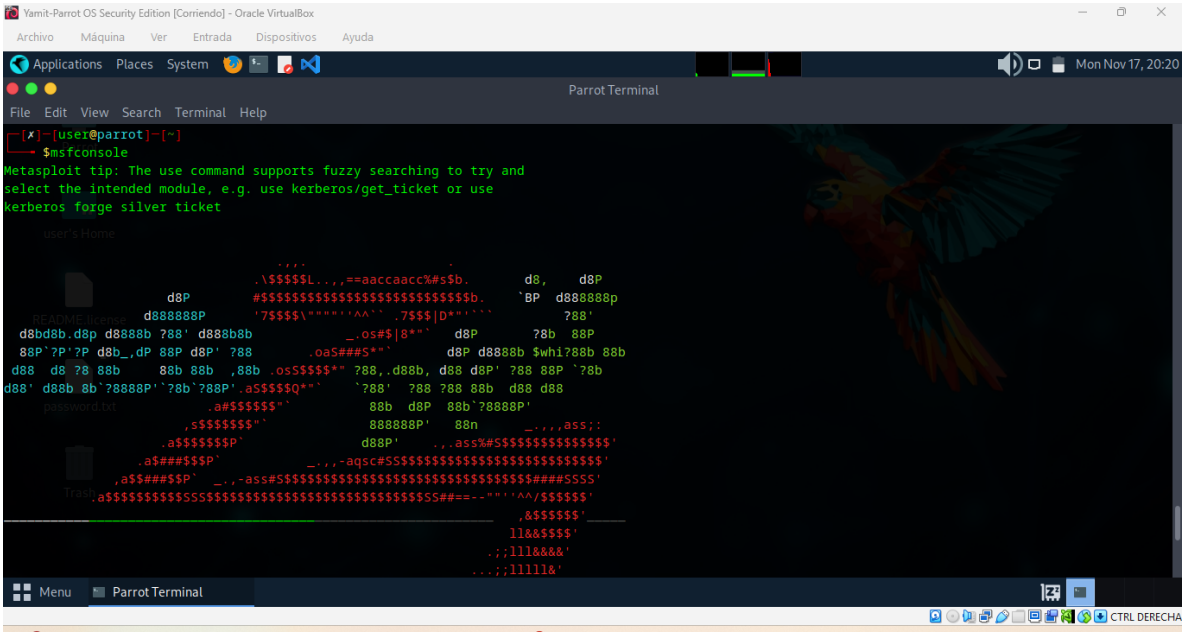
Fuente. Autoría Propia

La imagen muestra el uso de Nikto v2.5 para evaluar el host 172.20.10.8. El escaneo identifica el servidor HttpFileServer (HFS) 2.3 y reporta configuraciones inseguras, como ausencia de cabeceras X-Frame-Options y X-Content-Type-Options, además de una posible vulnerabilidad de Cross-Site Scripting (XSS). Esta información es clave para la gestión de riesgos y la mitigación de ataques web.

Explotación y post-explotación

Figura 13

Inicio de Metasploit Framework en Parrot OS

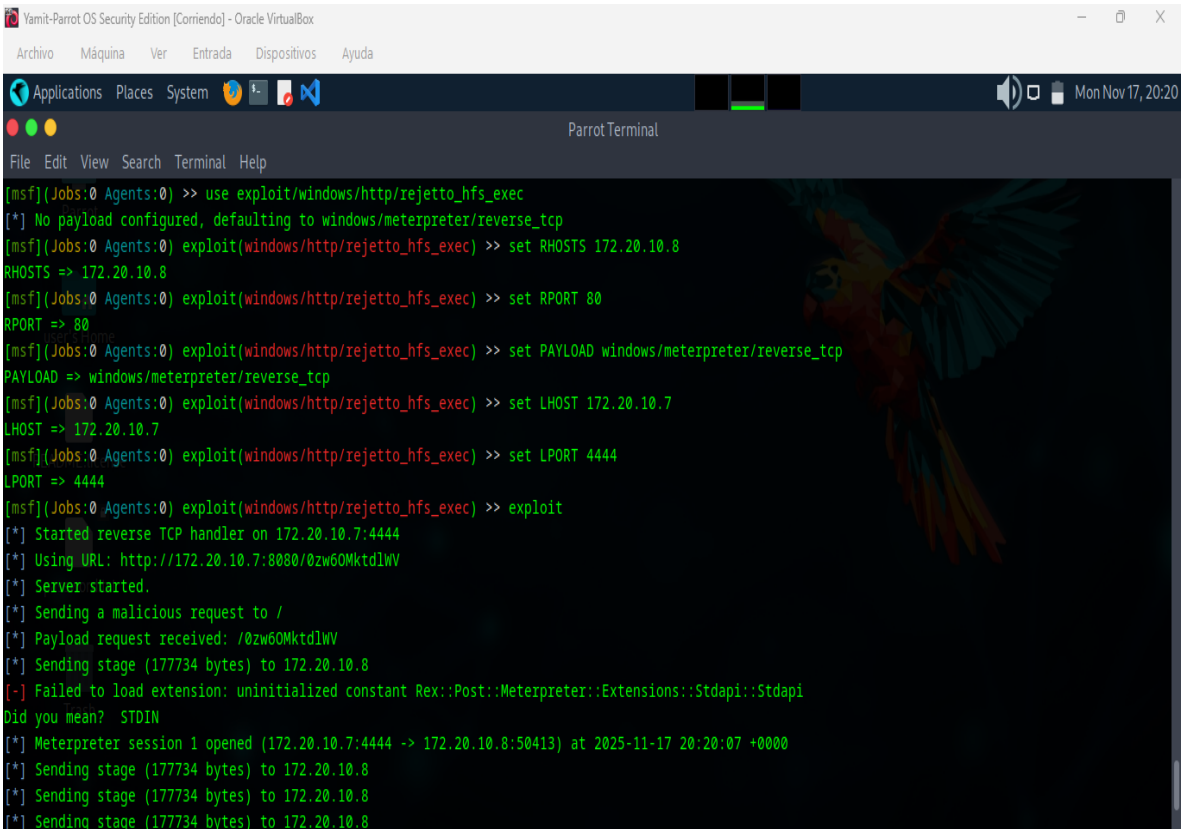


Fuente. Autoría Propia

La imagen muestra la ejecución del comando msfconsole para iniciar Metasploit Framework, una plataforma utilizada en pruebas de penetración para explotación de vulnerabilidades y desarrollo de payloads. Este entorno permite cargar módulos avanzados para ataques dirigidos.

Figura 14

Nuevo intento explotación meterpreter

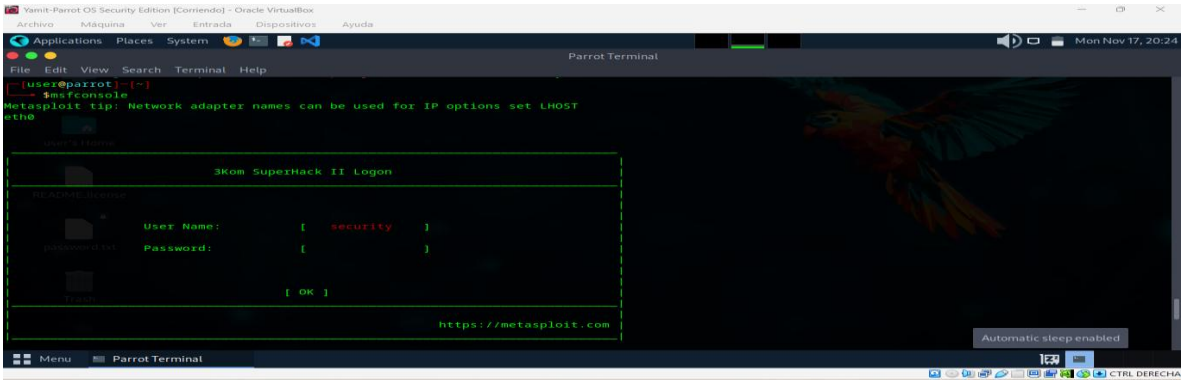


Fuente. Autoría Propia

Durante la explotación del servicio HFS, la sesión obtenida con Meterpreter no cargó la extensión stdapi, lo que restringió el uso de comandos avanzados. Esta limitación suele presentarse en exploits que emplean procesos temporales o payloads en modo staged. Para continuar con la fase de post-explotación, se utilizó el comando shell para ejecutar instrucciones del sistema operativo Windows y recopilar evidencia.

Figura 15

Ingreso a msfconsole

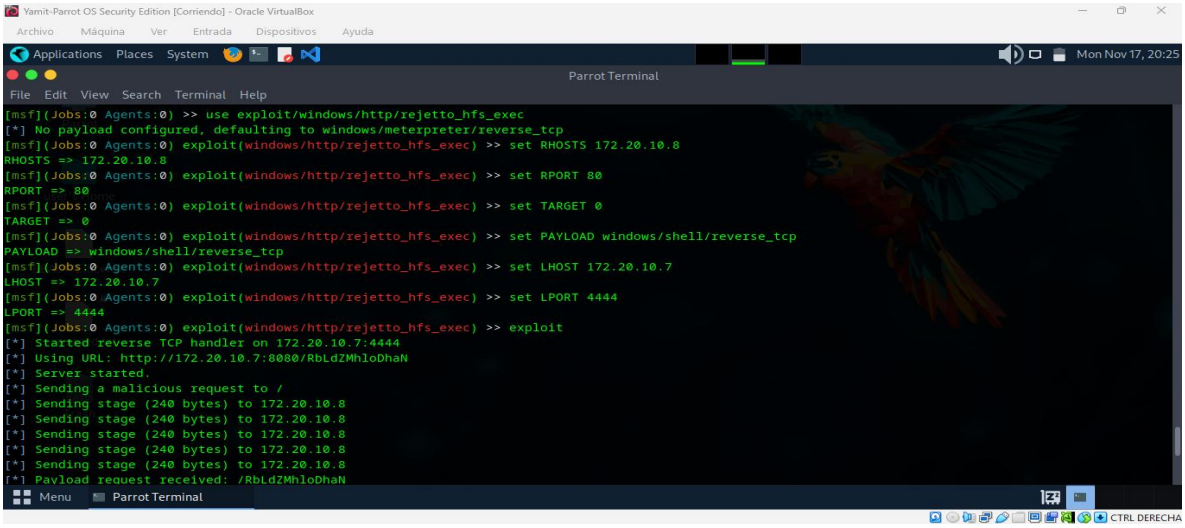


Fuente. Autoría Propia

La imagen muestra la consola de Metasploit preparada para ejecutar el exploit, con una configuración inicial limpia y estable lista para su implementación.

Figura 16

Comando acceso rejetto_hfs_exec

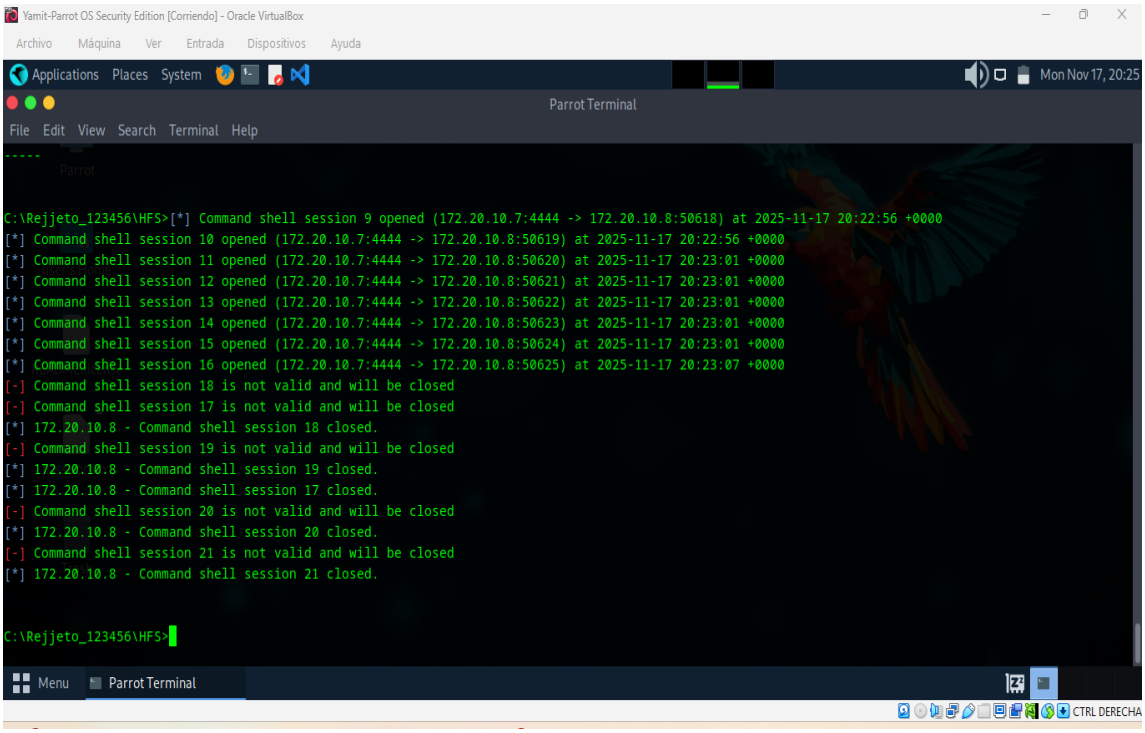


Fuente. Autoría Propia

La imagen presenta la ejecución del módulo exploit/windows/http/rejetto_hfs_exec en Metasploit Framework, configurando parámetros como RHOSTS 172.20.10.8, RPORT 80 y el payload windows/shell/reverse_tcp. El resultado indica la recepción del payload y la apertura de una conexión inversa, confirmando la explotación exitosa del servicio HFS.

Figura 17

Gestión de sesiones tras explotación en HFS

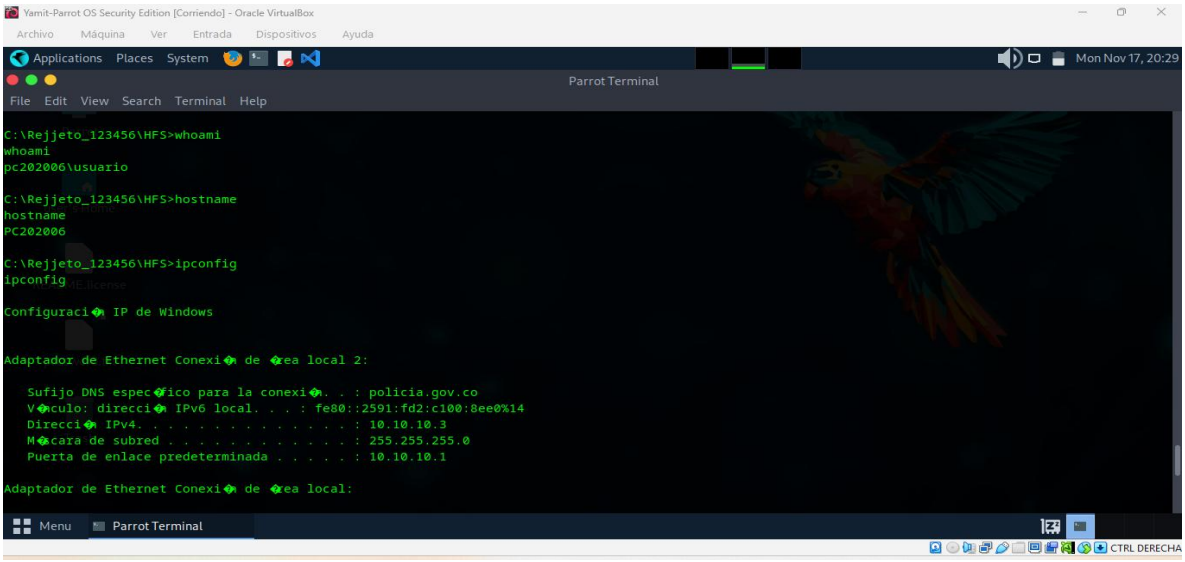


Fuente. Autoría Propia

La imagen muestra múltiples sesiones de comando abiertas mediante la explotación del servicio HttpFileServer (HFS) con Metasploit. Varias sesiones se cierran por no ser válidas, mientras otras permiten interacción con el sistema comprometido, evidenciando la fase de post-explotación.

Figura 18

Comandos de reconocimiento tras explotación en HFS

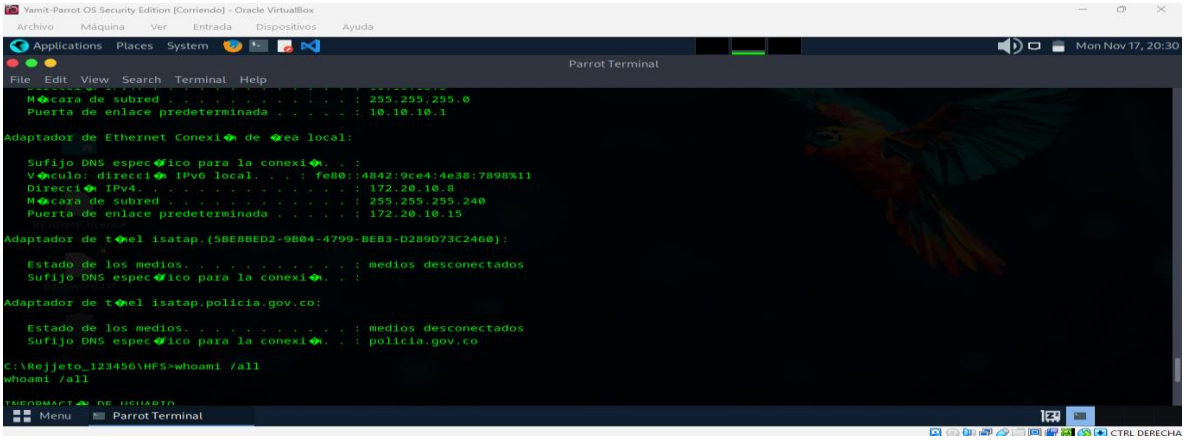


Fuente. Autoría Propia

La imagen muestra la ejecución de comandos whoami, hostname e ipconfig en una sesión obtenida tras explotar el servicio HttpFileServer (HFS). Se confirma el control del sistema Windows, revelando el usuario activo, el nombre del equipo y la configuración de red.

Figura 19

Verificación de configuración de red en el sistema comprometido

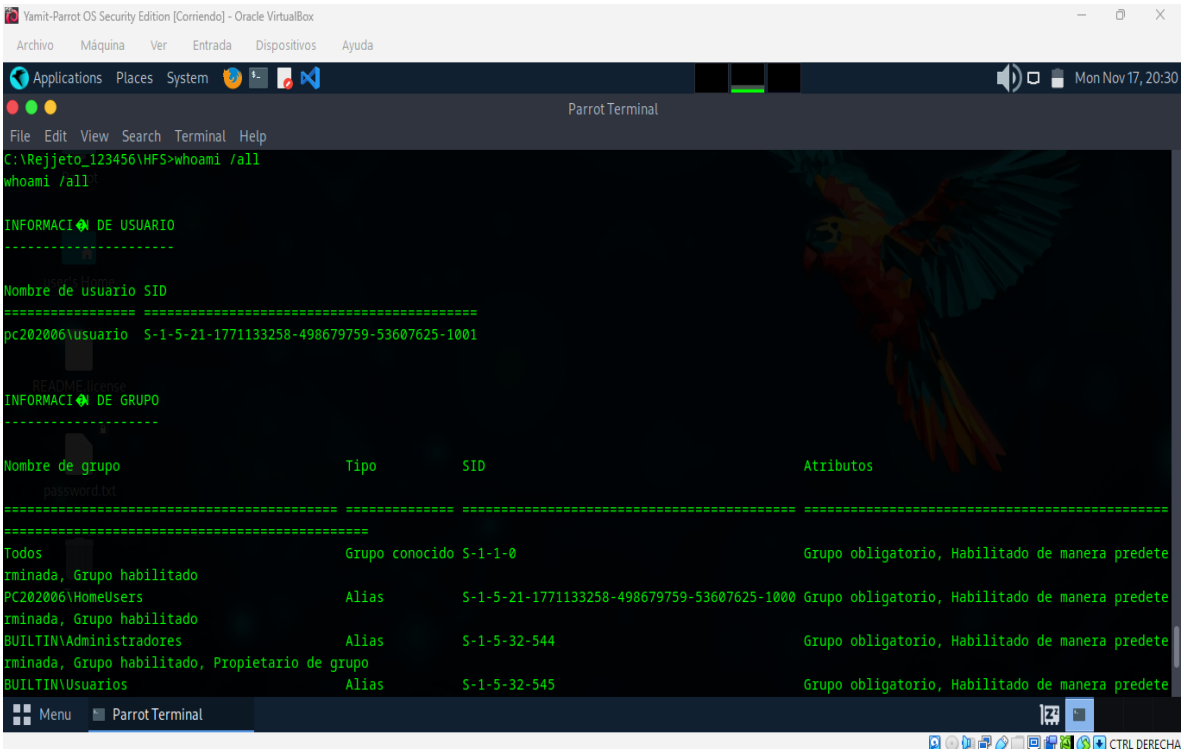


Fuente. Autoría Propia

La imagen muestra la ejecución del comando ipconfig /all en una sesión obtenida tras explotar el servicio HttpFileServer (HFS). Se revela la configuración de red del equipo Windows, incluyendo direcciones IPv4 (172.20.10.3 y 172.20.10.15), máscara de subred y sufijos DNS asociados a policia.gov.co, información clave para la fase de reconocimiento interno.

Figura 20

Enumeración de información de usuario y grupos en el sistema comprometido

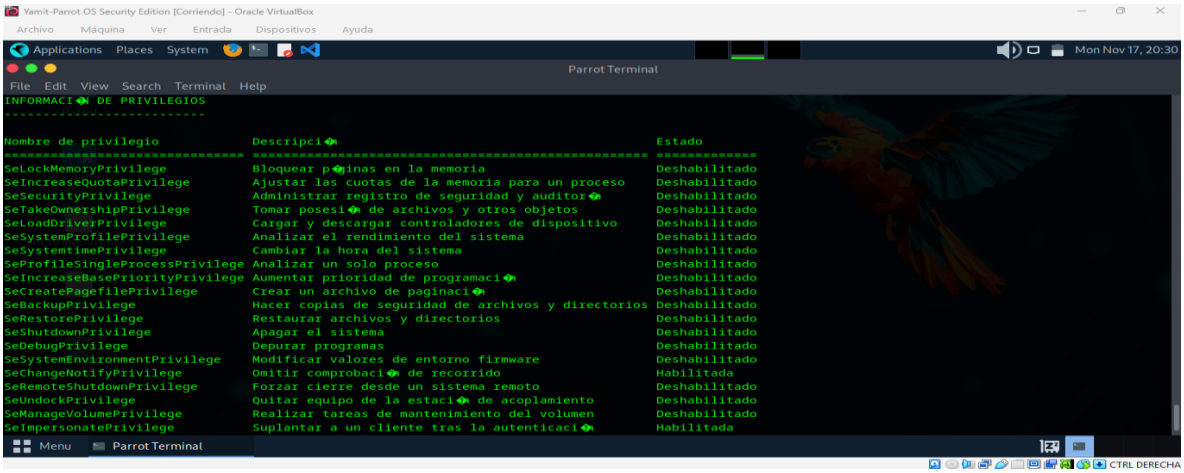


Fuente. Autoría Propia

La imagen muestra la ejecución del comando whoami /all en una sesión obtenida tras explotar el servicio HttpFileServer (HFS). Se revela el SID del usuario, el nombre del equipo y la pertenencia a grupos del sistema, información crítica para la fase de post-explotación y escalamiento de privilegios.

Figura 21

Enumeración de privilegios del usuario comprometido

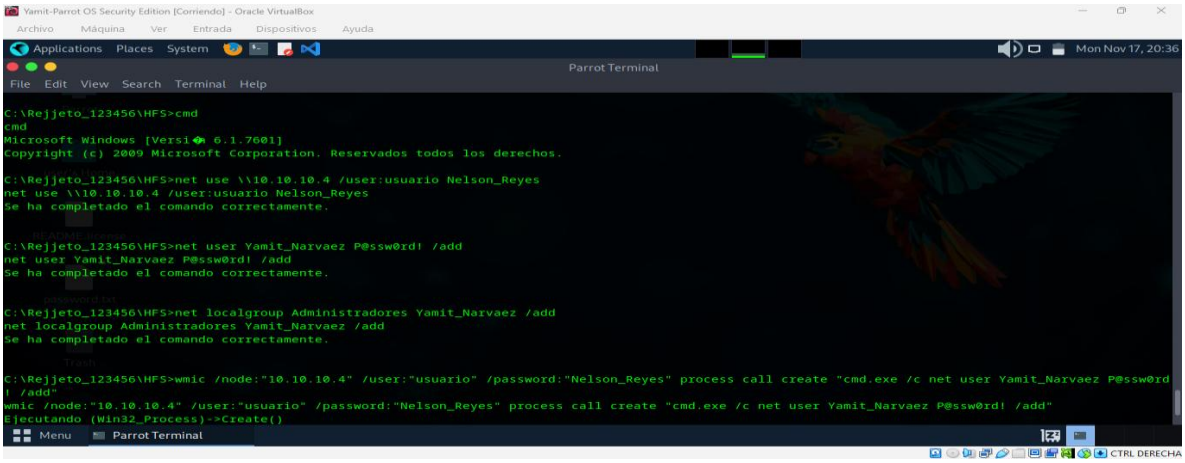


Fuente. Autoría Propia

La imagen muestra la lista de privilegios del usuario en el sistema Windows comprometido, obtenida tras la explotación del servicio HttpFileServer (HFS). Se identifican permisos como SeChangeNotifyPrivilege habilitado y otros deshabilitados, información esencial para evaluar posibilidades de escalamiento de privilegios.

Figura 22

Enumeración de privilegios del usuario comprometido

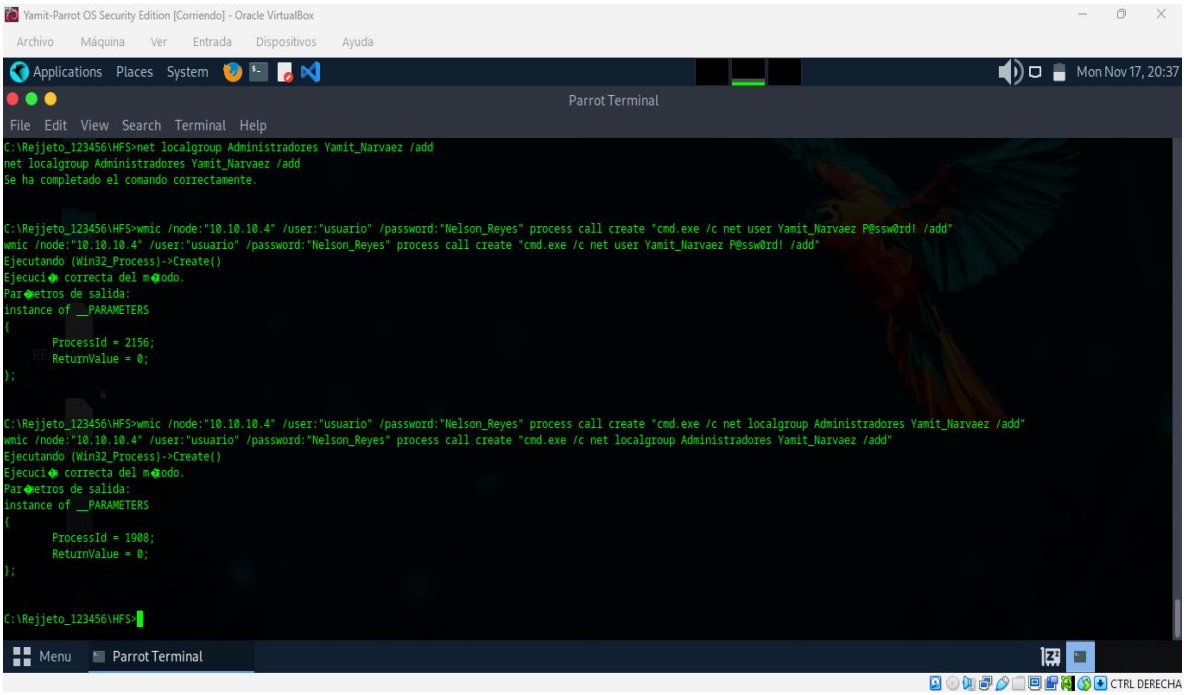


Fuente. Autoría Propia

La imagen muestra la lista de privilegios del usuario en el sistema Windows comprometido, obtenida tras la explotación del servicio HttpFileServer (HFS). Se identifican permisos como SeChangeNotifyPrivilege habilitado y otros deshabilitados, información esencial para evaluar posibilidades de escalamiento de privilegios.

Figura 23

Creación de usuario y asignación de privilegios administrativos tras explotación

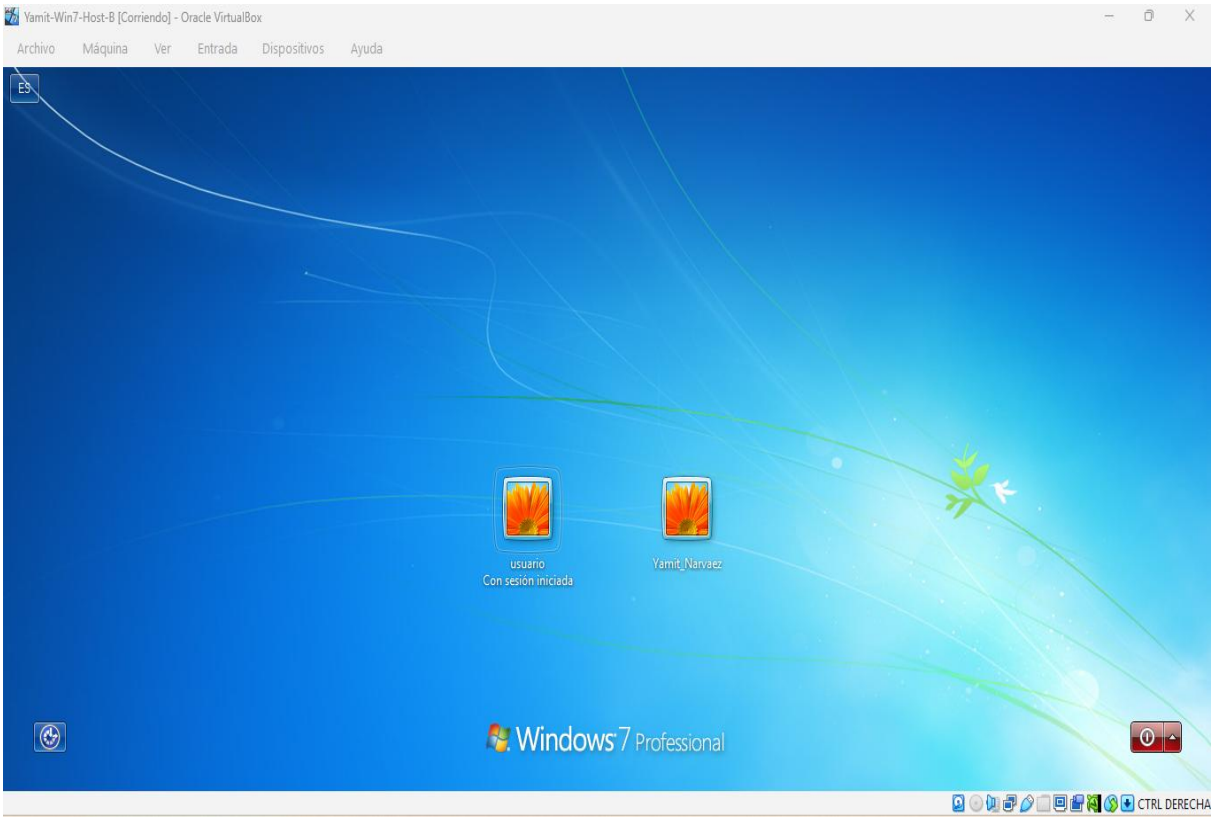


Fuente. Autoría Propia

La imagen muestra la ejecución de comandos para crear un nuevo usuario en el sistema comprometido (net user Yamit_Narvaez /add) y agregarlo al grupo de administradores (net localgroup Administradores Yamit_Narvaez /add). Esta acción se realiza desde la sesión obtenida tras explotar el servicio HttpFileServer (HFS), evidenciando la fase de escalamiento de privilegios.

Figura 24

Confirmación de acceso mediante cuenta creada tras explotación

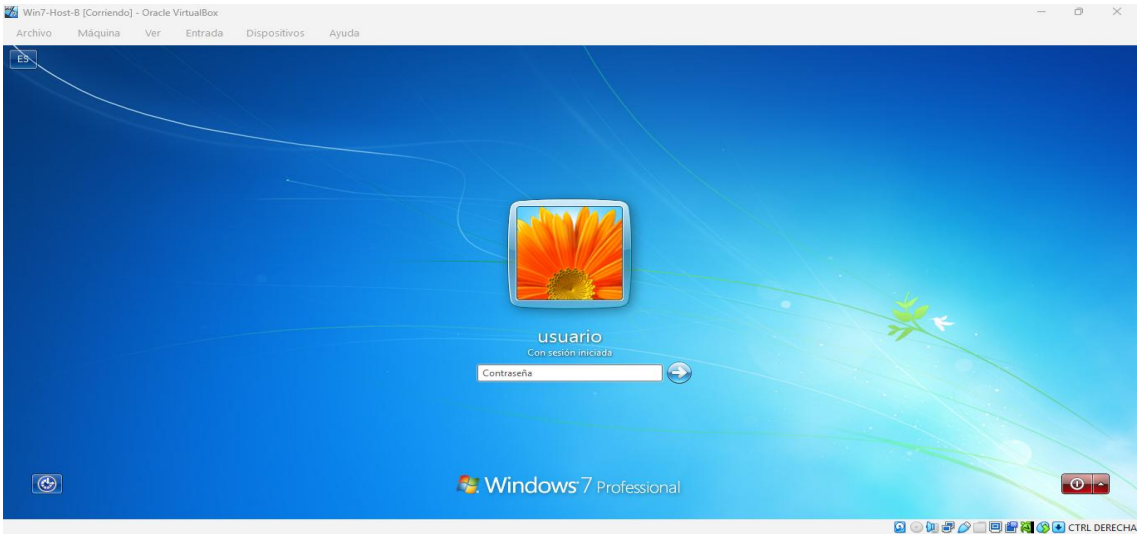


Fuente. Autoría Propia

La imagen muestra la pantalla de inicio de sesión de Windows 7 Professional, donde aparece la cuenta Yamit_Narvaez creada previamente desde la sesión obtenida tras explotar el servicio HttpFileServer (HFS). Esto confirma la persistencia y escalamiento de privilegios en el sistema comprometido.

Figura 25

Verificación en Host-A eliminación usuario creado



Fuente. Autoría Propia

La imagen confirma que no quedan reglas activas ni usuarios creados, cerrando el ejercicio de forma controlada y ética.

Respuesta y Contención ante Incidentes de Ciberseguridad

La aplicación de técnicas de hardening resulta esencial para disminuir la superficie de ataque. El uso de los **CIS Benchmarks** permite establecer configuraciones seguras de manera estandarizada (CIS Security, 2020). A esto se suma la gestión de parches, que es crucial para reducir vulnerabilidades (Scarfone & Mell, 2022). Asimismo, una adecuada administración de incidentes es determinante para fortalecer la resiliencia organizacional (Zambrano Hernández, Peña Hidalgo & Cárdenas Corral, 2024). Finalmente, la evaluación de riesgos junto con la implementación de controles técnicos constituye procesos críticos para asegurar la continuidad operativa (Centro de Respuestas a Incidentes Informáticos – CSIRT Académico UNAD, 2024).

Etapa 4 – Respuesta y contención del Blue Team

¿Qué sería lo primero que indagaría y haría si llegara a encontrarse un ataque en tiempo real? especifique su respuesta con argumentos técnicos.

Si detectara un ataque en tiempo real dentro del entorno de SecureNova Labs, lo primero que haría sería correlacionar la alerta con el vector de compromiso ya identificado en la fase anterior. En la Etapa 3 confirmé que el origen del ataque fue la explotación de la vulnerabilidad CVE-2014-6287, asociada a HttpFileServer (HFS) 2.3, un fallo que permite la ejecución remota de comandos a través de macros del template del servidor. Este vector fue aprovechado mediante el módulo rejetto_hfs_exec de Metasploit, lo que permitió al atacante obtener una shell en el sistema Windows 7 objetivo.

Debido a esto, ante un ataque activo mi primera acción sería verificar si el servicio HFS continúa operativo y si mantiene conexiones anómalas, ya que esta vulnerabilidad depende directamente de que el servicio esté disponible en el puerto HTTP. Para ello, priorizo la recolección de evidencia volátil utilizando herramientas nativas como netstat, tasklist o ipconfig, que fueron también empleadas en la fase de postexplotación del Red Team. Estas revisiones me permiten identificar si existe una sesión remota activa, conexiones sospechosas establecidas hacia direcciones externas o procesos asociados a la ejecución arbitraria de comandos.

En segundo lugar, indagaría si el atacante mantiene persistencia o privilegios elevados. Durante el ataque previo se logró crear un usuario con privilegios administrativos, lo cual implica que debo validar nuevamente el estado de cuentas locales, sesiones abiertas y privilegios activos mediante comandos como whoami /all, net user y la revisión de servicios en ejecución. Este análisis es crítico porque la explotación del CVE-2014-6287 permite ejecutar comandos sin autenticación previa, lo que facilita la instalación de mecanismos de persistencia si no se detectan a tiempo.

Una vez analizado el comportamiento de red y verificado el estado del sistema, procedería a contener el ataque sin eliminar evidencia, siguiendo las directrices de la Guía de la Etapa 4, que enfatiza actuar sobre el sistema de manera técnica y ordenada. Mi acción inmediata sería interrumpir la comunicación del atacante, ya sea bloqueando temporalmente el puerto HTTP utilizado por HFS o deshabilitando el adaptador de red, respetando la necesidad de conservar la evidencia para el análisis posterior.

Todo este proceso cumple con lo establecido en el Anexo 5 – Escenario 4, donde se especifica que el Blue Team debe actuar sobre el sistema comprometido utilizando herramientas nativas o bajo licencia GPL, coherentes con el banco de trabajo usado en la Etapa 3. Finalmente, registro cada comando ejecutado, los hallazgos obtenidos y la hora exacta en la que se realizaron.

Flujo de respuesta a incidentes:

Detección → Análisis → Contención → Recuperación

¿Teniendo en cuenta el ataque ejecutado desde el ejercicio de Red Team, qué medidas de hardenización propondría para que el ataque no se repita?

Después de analizar el ataque ejecutado en la Etapa 3, donde se explotó la vulnerabilidad CVE-2014-6287 en HttpFileServer (HFS) 2.3, considero que las medidas de hardenización deben centrarse en cerrar el vector exacto que permitió la intrusión, reforzar la superficie expuesta del sistema Windows 7 afectado y aplicar controles de defensa en profundidad, tal como sugiere el Enfoque Blue Team del Anexo 5.

En primer lugar, propondría la eliminación o actualización inmediata del servicio HFS 2.3, ya que esta versión es vulnerable a la ejecución remota de comandos sin autenticación. En el escenario original, este servicio permanecía expuesto por HTTP sin ningún tipo de control de acceso, lo cual facilitó que el Red Team pudiera ejecutar el exploit rejetto_hfs_exec y obtener una

shell remota en el sistema. Evitar que se repita requiere, por tanto, retirar completamente el software vulnerable o reemplazarlo por una solución segura y actualizada.

Adicionalmente, dado que el ataque dependió del puerto 80 abierto y accesible, aplicaría una política estricta de filtrado de puertos. Esto incluye deshabilitar todos los puertos que no sean necesarios para la operación del equipo y permitir únicamente el tráfico interno requerido. En la etapa anterior, el escaneo de servicios y el mapeo de puertos mostraron que el servicio HTTP estaba expuesto sin restricciones, lo cual no debe volver a ocurrir. El cierre de puertos innecesarios está alineado con las recomendaciones de la Guía de Respuesta y Contención, donde se resalta la importancia del control de la superficie de ataque.

Otra medida fundamental es restringir los privilegios locales y revisar los usuarios creados durante la intrusión. En la explotación del CVE-2014-6287, el atacante logró crear un usuario con privilegios administrativos, lo que amplificó el impacto del compromiso. Para evitarlo, implementaría políticas de mínimo privilegio, bloqueo de creación remota de cuentas y monitoreo continuo de cambios en el sistema. Estas acciones se sustentan en la evidencia de la fase Red Team, donde se identificaron privilegios elevados obtenidos tras la explotación.

También considero necesaria la configuración de auditorías de eventos y registros del sistema, orientadas a detectar procesos inesperados, cambios en la configuración, uso interactivo de cuentas privilegiadas y ejecución de comandos sospechosos. Esta medida permite construir un mecanismo de detección temprana coherente con los lineamientos del Escenario 4, que exige intervención reactiva y preventiva del Blue Team sobre el sistema afectado.

Finalmente, complementaría las acciones anteriores con segmentación de red y aislamiento de servicios, reduciendo las posibilidades de movimiento lateral y limitando las rutas que un atacante podría explotar. Esto es coherente con la Rúbrica de Evaluación, que establece la importancia de implementar medidas de prevención documentadas y justificadas técnicamente.

En conjunto, estas medidas de hardenización abordan tanto la vulnerabilidad concreta utilizada por el Red Team como las debilidades estructurales que hicieron posible la explotación, garantizando que el ataque no pueda repetirse bajo las mismas condiciones.

¿Describa con sus palabras las diferencias entre un equipo blueteam y un equipo de respuesta a incidentes informáticos?

Desde mi perspectiva como analista, aunque el equipo Blue Team y el equipo de respuesta a incidentes trabajan dentro del mismo proceso de defensa organizacional, sus enfoques y responsabilidades son claramente distintos. El Blue Team tiene una visión amplia y preventiva: su función principal es fortalecer la infraestructura, reducir la superficie de ataque y mantener la seguridad operativa del sistema día a día. En el Escenario 4, este rol incluye la verificación del sistema comprometido, el análisis del estado actual del equipo Windows afectado y la aplicación de medidas técnicas que impidan nuevas intrusiones.

Por otro lado, el equipo de respuesta a incidentes actúa de manera más puntual y reactiva. Su intervención ocurre cuando ya existe un incidente confirmado, y su objetivo se centra en contener, erradicar y recuperar el entorno afectado. En la Guía de la Etapa 4 se menciona explícitamente este proceso, resaltando las acciones de contención, preservación de evidencia y recuperación del sistema como tareas características del personal responsable de gestionar incidentes de seguridad.

En otras palabras, mientras que el Blue Team busca evitar que los ataques ocurran mediante monitoreo, endurecimiento y controles preventivos, el equipo de respuesta a incidentes se encarga de actuar cuando el ataque ya está en curso o ha generado un impacto, siguiendo procedimientos estructurados para minimizar daños y restaurar la operación. Según la rúbrica, ambos roles se complementan, pero la evaluación diferencia claramente entre la labor de análisis técnico proactivo del Blue Team y la ejecución de acciones de contención propias del manejo de incidentes.

En resumen, el Blue Team protege y fortalece; el equipo de respuesta interviene, controla y recupera. Aunque trabajan alineados, su alcance, objetivos y tiempos de actuación son diferentes y se articulan para asegurar una defensa integral.

¿Si dentro de un equipo blue team le indican que debe trabajar con CIS "Center For Internet Security", usted lo utilizaría para qué fin?

Si dentro del equipo Blue Team se me asignara el uso de las guías del Center for Internet Security (CIS), las utilizaría principalmente para establecer controles de seguridad estandarizados y respaldados por buenas prácticas internacionales, con el fin de reducir la superficie de ataque del sistema comprometido y fortalecer su configuración. En el contexto del escenario que he analizado, donde la máquina Windows 7 fue vulnerada a través del servicio HFS 2.3, los controles CIS serían una referencia fundamental para aplicar medidas de endurecimiento alineadas con los procedimientos formales de defensa que exige el curso.

Desde mi rol, emplearía los CIS Benchmarks como una hoja de ruta técnica para revisar configuraciones del sistema operativo, servicios, políticas de usuarios, auditorías y mecanismos de acceso. Esto se relaciona directamente con las responsabilidades descritas para el Blue Team en el Anexo 5, donde se indica que el equipo de defensa debe intervenir en el sistema aplicando configuraciones seguras que mitiguen futuros incidentes.

Además, las guías CIS me permiten estructurar el proceso de hardenización de manera sistemática, tal como se solicita en la Guía de Etapa 4 durante las fases de prevención y contención. Su uso facilita que las medidas propuestas no sean improvisadas, sino respaldadas por estándares claros que ayudan a evitar que vulnerabilidades como la explotada mediante el CVE-2014-6287 vuelvan a poner en riesgo la infraestructura.

También utilizaría los controles CIS para verificar el cumplimiento y documentar adecuadamente los ajustes aplicados, algo relevante considerando que la rúbrica de evaluación

exige evidencias claras, organizadas y técnicamente justificadas. Al apoyarme en estos controles, garantizo que las medidas de protección aplicadas sobre el sistema afectado cumplan con criterios de calidad, consistencia y trazabilidad.

En resumen, emplearía el CIS como una herramienta para asegurar que todas las configuraciones del sistema estén alineadas con estándares reconocidos, ayudando a prevenir reinfecciones, reducir vulnerabilidades y fortalecer globalmente la postura defensiva del entorno.

Explique y redacte las funciones y características principales de lo que es un SIEM.

Desde mi perspectiva como integrante del Blue Team, un SIEM (Security Information and Event Management) es una de las herramientas más importantes para llevar a cabo tareas de monitoreo, análisis y respuesta ante posibles incidentes. Aunque en los documentos del curso no se describe directamente este tipo de plataforma, sus funciones sí se relacionan con las responsabilidades que se atribuyen al equipo defensor en el Anexo 5 y en la Guía de la Etapa 4, especialmente en lo que respecta a la detección, registro y análisis de actividad sospechosa.

En términos prácticos, considero que un SIEM cumple varias funciones clave. La primera es la centralización de registros: en lugar de revisar manualmente los eventos de cada equipo o servicio, el SIEM recopila logs del sistema operativo, registros de red, auditorías, alertas y notificaciones en un solo punto. Esto facilita el análisis al momento de investigar un incidente como el que ocurrió en el escenario, donde el compromiso del equipo Windows 7 requería revisar conexiones, servicios ejecutándose y actividades anómalas. Un SIEM permite automatizar ese proceso y visualizar patrones que podrían pasar desapercibidos en revisiones manuales.

Otra función esencial es la correlación de eventos. El SIEM compara millones de registros y detecta relaciones que indican actividad maliciosa. Por ejemplo, si un atacante explotara nuevamente una vulnerabilidad como la CVE-2014-6287, el SIEM podría identificar conexiones inusuales al puerto HTTP, ejecución de comandos inesperados o la creación repentina de un

usuario con privilegios elevados. Esta correlación respalda el enfoque reactivo y analítico que se menciona en el Escenario 4 para el rol del Blue Team.

También utilizaría el SIEM para la generación de alertas en tiempo real. Este tipo de notificaciones es fundamental cuando se necesita actuar rápido, tal como lo exige la fase de contención descrita en la Guía de Etapa 4. El SIEM permite configurar reglas que activan alertas ante comportamientos anómalos, facilitando la respuesta antes de que el incidente se expanda.

Una característica adicional es la capacidad de auditoría e historización, que sirve tanto para reconstruir incidentes como para documentar acciones. Gracias a esto, puedo revisar qué ocurrió antes, durante y después del ataque, lo cual permite tomar decisiones informadas y justificar técnicamente las acciones ejecutadas.

En resumen, un SIEM funciona como la plataforma que me permite ver en conjunto todo el comportamiento de la red y los sistemas. Su utilidad se centra en recopilar, analizar y alertar sobre eventos relevantes, permitiendo que como Blue Team pueda actuar con rapidez, precisión y conforme a los procesos de respuesta formalmente establecidos en el curso.

Defina por lo menos 3 herramientas de contención de ataques informáticos “hardware o software”, recuerde que las herramientas de contención son diferentes a las herramientas de detección.

A partir de lo trabajado en el escenario y siguiendo las directrices de la Guía de Respuesta y Contención, considero que las herramientas de contención son aquellas que me permiten detener un ataque en curso, interrumpir la comunicación del atacante o aislar el sistema comprometido para evitar que el incidente siga avanzando. Estas herramientas no están enfocadas en detectar, sino en bloquear, limitar, aislar o interrumpir la propagación del ataque.

Firewalls (hardware o software)

Utilizaría un firewall como una herramienta de contención para bloquear puertos, interrumpir conexiones sospechosas o limitar el tráfico hacia el sistema afectado. En el escenario del curso, por ejemplo, podría bloquear específicamente el puerto HTTP utilizado por HFS 2.3 después de identificar el abuso de la vulnerabilidad CVE-2014-6287. Esta medida está directamente alineada con las acciones de contención mencionadas en la Guía de Etapa 4, donde se enfatiza controlar los servicios y comunicaciones que intervienen en un incidente.

En términos de hardware, un firewall perimetral puede detener el tráfico externo; en software, el firewall del sistema operativo permite aislar el equipo localmente.

VLANs o mecanismos de aislamiento de red (hardware de red)

Otra herramienta que considero esencial para la contención es el aislamiento por VLAN, ya sea aplicándolo en un switch administrable o mediante segmentación lógica. Si un equipo está comprometido, puedo trasladarlo a una VLAN restringida con acceso limitado o sin salida a internet. Esto evita la propagación del ataque y minimiza el impacto en otros sistemas.

Este tipo de acción está alineada con el enfoque de intervención técnica descrito en el Anexo 5, en el cual se espera que el Blue Team actúe directamente sobre la conectividad y el sistema afectado para evitar que el incidente se amplíe.

Deshabilitación o bloqueo temporal del adaptador de red (software del sistema operativo)

Una medida de contención simple pero muy efectiva es deshabilitar el adaptador de red del equipo comprometido. Esta acción corta inmediatamente la comunicación con el atacante.

En el escenario, si detecto que el sistema Windows 7 sigue respondiendo al exploit o mantiene una conexión activa con el atacante, esta herramienta de contención me permite detener el ataque sin eliminar evidencia, tal como recomienda la Guía de Etapa 4 durante la fase inicial de investigación y preservación.

Estas tres herramientas cumplen con la función concreta de contención porque:

- Interrumpen el ataque en curso.
- Evitan que el atacante siga interactuando con el sistema.
- Permiten preservar evidencia según el procedimiento formal de respuesta.
- Se ajustan al modelo de contención descrito en la Etapa 4.

Comparativa y escenarios de uso:

- Firewall: Ideal para bloquear tráfico específico en tiempo real, útil cuando se requiere acción inmediata sobre puertos o servicios comprometidos.
- VLAN: Más efectiva en entornos corporativos donde se necesita aislar equipos comprometidos sin desconectarlos totalmente, permitiendo análisis forense.
- Deshabilitación del adaptador: Opción rápida en entornos pequeños o cuando no hay acceso a configuraciones avanzadas, corta la comunicación de forma inmediata.

Riesgos residuales y su gestión:

Tras la contención inicial, pueden persistir riesgos como cuentas ocultas, configuraciones inseguras, malware latente o accesos no autorizados. Para gestionarlos, se recomienda:

• Monitoreo continuo mediante SIEM para detectar actividad anómala.

• Auditorías periódicas del sistema y revisión de logs.

• Pruebas de penetración controladas para validar la efectividad de las medidas aplicadas.

Estas acciones reducen la probabilidad de reinfección y garantizan la integridad del entorno.

Evidencias de Sustentación

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

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

Conclusiones

Al finalizar el desarrollo de las cuatro etapas del seminario, puedo concluir que la ciberseguridad efectiva requiere un enfoque integral que combine tácticas ofensivas y defensivas bajo principios éticos y normativos. La práctica realizada en la Etapa 3, donde exploté la vulnerabilidad CVE-2014-6287 mediante el uso de Metasploit, evidenció la criticidad de las configuraciones inseguras y la necesidad de aplicar controles preventivos (Raj & Walia, 2020). Esta experiencia me permitió comprender que las pruebas ofensivas no son un fin en sí mismas, sino una herramienta para fortalecer la resiliencia organizacional.

Asimismo, la Etapa 4 demostró que las estrategias defensivas, como el hardening, la segmentación de red y la implementación de CIS Benchmarks, son esenciales para reducir la superficie de ataque y garantizar la continuidad operativa (Center for Internet Security, 2022). La integración de sistemas SIEM para correlación de eventos y generación de alertas en tiempo real confirmó la importancia de la detección temprana y la respuesta proactiva ante incidentes (Chindrus & Caruntu, 2023).

Otra conclusión relevante es que la ética profesional y el cumplimiento normativo no son opcionales, sino pilares fundamentales en la práctica de la ciberseguridad. La Ley 1273 de 2009 y el Código de Ética del COPNIA establecen directrices claras que deben guiar cada acción técnica, evitando que las pruebas se conviertan en mecanismos de vulneración de derechos (Congreso de la República, 2009; COPNIA, 2015).

Finalmente, este trabajo reafirma que la colaboración entre Red Team y Blue Team no debe entenderse como una competencia, sino como un proceso complementario orientado a la mejora continua. Solo mediante la integración de conocimientos técnicos, principios éticos y estándares internacionales es posible construir entornos digitales seguros y confiables (Howard & Prince, 2021).

Recomendaciones

Con base en los resultados obtenidos y las lecciones aprendidas durante el desarrollo del seminario especializado, se recomienda fortalecer las capacidades técnicas, tácticas y de respuesta de los equipos Red Team y Blue Team mediante la implementación de un programa continuo de gestión de vulnerabilidades. Este debe contemplar procesos automatizados para la identificación y corrección oportuna de fallas de seguridad, complementados con auditorías periódicas que permitan evaluar la efectividad de los controles implementados. Herramientas como OpenVAS, así como otros escáneres de vulnerabilidades, pueden integrarse en los flujos operativos de seguridad con el fin de garantizar actualizaciones constantes y una reducción progresiva de la superficie de ataque (Greenbone Networks, 2023).

Asimismo, resulta fundamental fortalecer la capacitación del personal encargado de la respuesta a incidentes, dado que la efectividad del Blue Team no depende únicamente de las herramientas tecnológicas, sino también del conocimiento y la preparación del recurso humano. En este sentido, se recomienda promover entrenamientos regulares en análisis forense digital, uso de plataformas SIEM y aplicación de marcos de referencia como MITRE ATT&CK, con el propósito de mejorar las capacidades de detección, análisis y contención de incidentes de seguridad (MITRE, 2023).

De igual manera, se sugiere adoptar estándares internacionales y buenas prácticas reconocidas en el ámbito de la ciberseguridad, tales como los CIS Benchmarks y la norma ISO/IEC 27001, los cuales contribuyen a establecer configuraciones seguras, procesos auditables y un enfoque sistemático de gestión del riesgo. La alineación con estos marcos permite reducir vulnerabilidades estructurales y facilita el cumplimiento de requisitos normativos y regulatorios (Center for Internet Security, 2022).

Otra recomendación relevante consiste en promover la colaboración constante entre los equipos Red Team y Blue Team mediante la realización de ejercicios periódicos de simulación de ataques y defensa activa, bajo reglas de compromiso claramente definidas. Este tipo de prácticas favorece la identificación de brechas de seguridad, la validación de controles existentes y el fortalecimiento de la resiliencia organizacional frente a amenazas reales (Howard & Prince, 2021).

Finalmente, se recomienda integrar tecnologías avanzadas para el monitoreo y la correlación de eventos de seguridad, priorizando la adopción de plataformas SIEM con capacidades de análisis en tiempo real y respuesta automatizada. La complementariedad de estas soluciones con servicios de inteligencia de amenazas permite reducir los tiempos de detección y mitigación de incidentes, incrementando de manera significativa la eficacia de las estrategias defensivas (Chindrus & Caruntu, 2023). En conjunto, estas recomendaciones buscan consolidar una estrategia integral que articule tecnología, procesos y talento humano, asegurando que la ciberseguridad evolucione de forma coherente con el dinamismo de las amenazas emergentes.

Referencias Bibliográficas

Andress, J. (2023). *The basics of information security: Understanding the fundamentals of InfoSec in theory and practice* (3rd ed.). Syngress.

Casey, E. (2022). *Digital evidence and computer crime: Forensic science, computers, and the internet* (4th ed.). Academic Press.

Center for Internet Security. (2022). *CIS benchmarks*. https://www.cisecurity.org/cis-benchmarks/

CERT Coordination Center. (2024). *Cybersecurity incident response exercises and simulations*. https://www.cert.org

Chindrus, C., & Caruntu, C.-F. (2023). Securing the network: A red and blue cybersecurity competition case study. *Information, 14*(11), 587. https://doi.org/10.2478/bipie-2023-0008

ColCERT. (2024). *Buenas prácticas para la gestión de incidentes de ciberseguridad en Colombia*. https://www.colcert.gov.co

Congreso de la República de Colombia. (2009). *Ley 1273 de 2009 por la cual se modifica el Código Penal y se crea un nuevo bien jurídico tutelado: la protección de la información y de los datos*. Diario Oficial.

Congreso de la República de Colombia. (2012). *Ley 1581 de 2012 por la cual se dictan disposiciones generales para la protección de datos personales*. Diario Oficial.

COPNIA. (2015). *Código de ética profesional*. Consejo Profesional Nacional de Ingeniería.

Greenbone Networks. (2023). *OpenVAS vulnerability management*. https://www.greenbone.net

Howard, M., & Prince, S. (2021). *Cybersecurity essentials*. Packt Publishing.

Kotwani, A., Gupta, R., & Singh, P. (2023). Analysis of common vulnerabilities and exposures in legacy systems. *International Journal of Information Security*, 22(3), 451–468.

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

MITRE. (2023). *MITRE ATT&CK framework*. https://attack.mitre.org

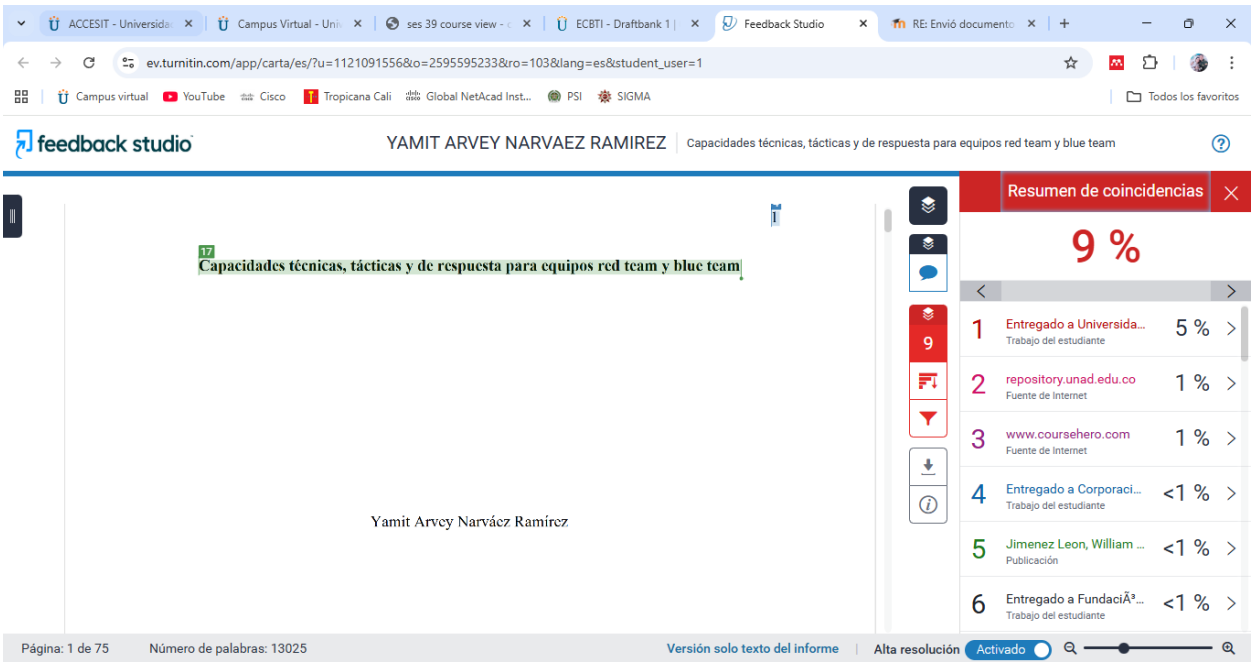
NIST. (2020). *Guide to penetration testing* (Special Publication 800-115). National Institute of Standards and Technology. https://csrc.nist.gov

Raj, A., & Walia, K. (2020). *Metasploit for penetration testing*. Packt Publishing.

Apéndices

Apéndice A

Resultado de revisión en Turnitin



Nota. Resultados del informe de similitud generado por Turnitin para el trabajo *Capacidades técnicas, tácticas y de respuesta para equipos red team y blue team*. El porcentaje total de coincidencia fue del 9 %.