Profundización Nmap (flags, scripting)

En el contexto de una auditoría de ciberseguridad, uno de los pasos fundamentales es el **reconocimiento y mapeo de los sistemas y servicios expuestos**. Para ello, el escaneo de puertos y servicios es esencial, ya que permite identificar vectores de ataque potenciales, vulnerabilidades conocidas y configuraciones inseguras que pueden comprometer la seguridad de la organización.

Nmap es una de las herramientas más potentes y ampliamente utilizadas para realizar escaneos de red y auditorías de seguridad. Gracias a su flexibilidad, permite desde un simple ping a un host hasta la ejecución de scripts avanzados para detectar vulnerabilidades específicas, pasando por la identificación de versiones de servicios y sistemas operativos.

¿Por qué usar Nmap en auditorías?

- **Detección de superficie de ataque inicial**: Nmap permite descubrir rápidamente puertos abiertos, servicios expuestos y sistemas activos, lo que facilita enfocar los siguientes pasos de la auditoría.
- **Clasificación de servicios y versiones**: con funciones como -sV, es posible identificar versiones concretas de servicios, lo que permite correlacionarlas con bases de datos de vulnerabilidades (CVE).
- **Automatización con scripts NSE**: Nmap Script Engine (NSE) ofrece una gran variedad de scripts para detectar configuraciones inseguras, certificados débiles, vulnerabilidades conocidas, etc., lo que mejora la profundidad del análisis inicial.
- **Adaptación a diferentes escenarios**: mediante flags avanzados, Nmap se adapta tanto a auditorías rápidas como a entornos sensibles donde se requiere sigilo para evitar disparar alarmas.
- **Aumento de la precisión en la auditoría**: un escaneo bien configurado permite reducir falsos positivos, comprender mejor la arquitectura de la red y entregar un informe detallado con evidencias sólidas.

Importancia en la fase de auditoría técnica

En una auditoría de seguridad, la fase técnica comienza generalmente con un **reconocimiento activo**, donde se usan herramientas como Nmap para:

- Confirmar la información de activos proporcionada por la organización (o descubrir activos desconocidos).

- Identificar servicios que no cumplen las políticas de seguridad establecidas (por ejemplo, servicios inseguros como Telnet o FTP sin cifrado).
- Establecer una línea base para futuras auditorías (comparando resultados en el tiempo).
- Clasificar hallazgos por criticidad, ayudando a priorizar correcciones.

¿Qué es una flag en Nmap?

En Nmap, una **flag** es un parámetro u opción que se utiliza en la línea de comandos para modificar el comportamiento del escaneo. Cada flag permite activar o configurar funcionalidades específicas, como el tipo de escaneo, la velocidad, la detección de sistemas operativos, la salida del informe, etc.

Por ejemplo:

-sS: flag para realizar un escaneo TCP SYN sigiloso.

-p: flag para especificar qué puertos escanear (ej.: -p 22,80,443). -A: flag para activar detección avanzada (OS, versiones, scripts). -D: flag para añadir direcciones IP señuelo y evadir detección.

En resumen: las flags son como “interruptores” que permiten personalizar qué hace Nmap en cada escaneo.

¿Qué es el scripting en Nmap?

El **scripting** en Nmap se refiere al uso del *Nmap Scripting Engine* (NSE), un sistema que permite ejecutar scripts automatizados durante un escaneo para realizar tareas adicionales, como:

- Detectar vulnerabilidades conocidas.
- Realizar ataques específicos de prueba (p.ej., intentos de autenticación con credenciales débiles).
- Obtener información detallada de servicios (p.ej., banners, certificados SSL).
- Verificar configuraciones inseguras (p.ej., FTP anónimo, SNMP público).

Los scripts NSE se organizan por categorías como:

- default: scripts ejecutados con -sC (escaneo estándar).
- vuln: scripts para detectar vulnerabilidades.
- safe: scripts que no alteran el servicio objetivo.

- intrusive: scripts más agresivos que pueden generar alertas.

Ejemplo de uso de scripting:

nmap --script vuln <IP-objetivo>

En resumen: el scripting es la capacidad de Nmap para automatizar comprobaciones avanzadas mediante scripts predefinidos (o personalizados), y así ampliar su utilidad mucho más allá de un simple escaneo de puertos.

Cómo usar este módulo de ejercicios

En esta guía práctica encontrarás 10 ejercicios diseñados para que puedas practicar las opciones avanzadas de Nmap (flags y scripting NSE) en escenarios realistas. Cada ejercicio te ayudará a:

- Profundizar en el manejo de Nmap.
- Mejorar tus habilidades para detectar y clasificar vulnerabilidades iniciales.
- Aumentar la precisión de tus auditorías técnicas.

Los ejercicios se centran en la combinación de escaneos rápidos y detallados, detección de sistemas operativos, uso de scripts NSE, evasión básica y generación de evidencias para tus informes.

Recomendación para auditores

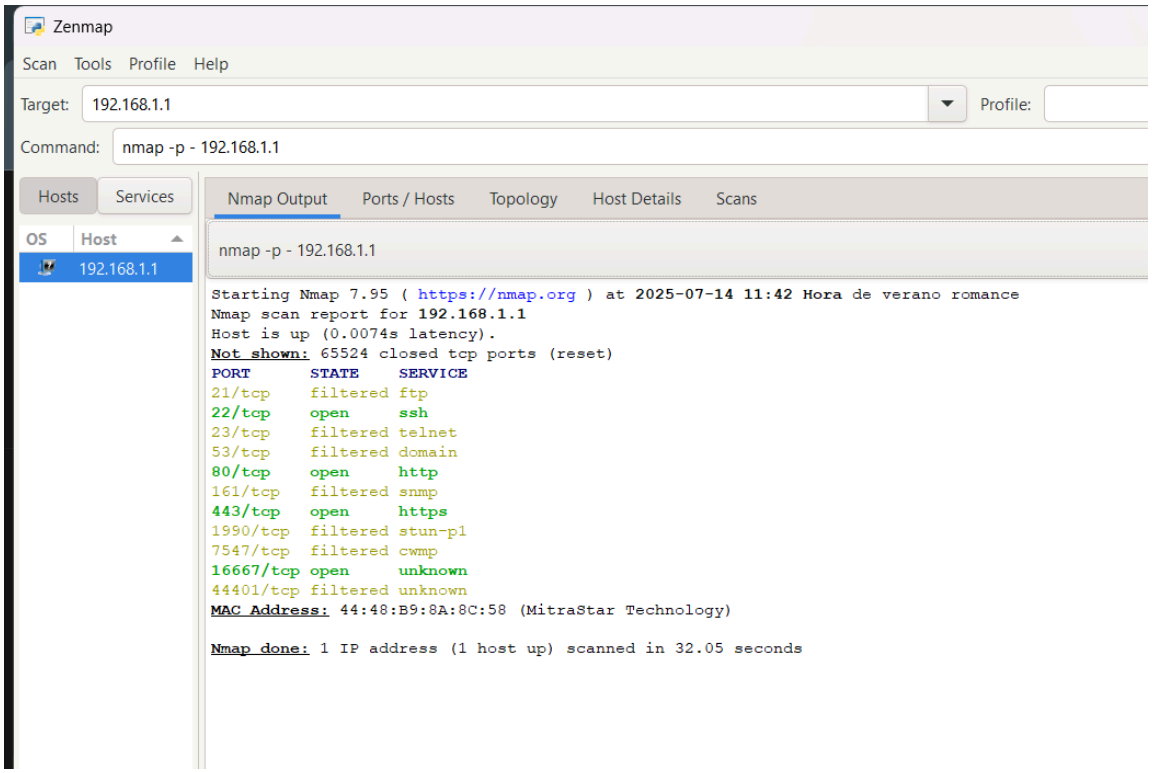
Antes de ejecutar escaneos en un entorno corporativo, asegúrate de contar con **autorización formal y documentada** para evitar violaciones legales o conflictos con el equipo de operaciones. Realiza siempre tus pruebas dentro del alcance acordado en la auditoría.

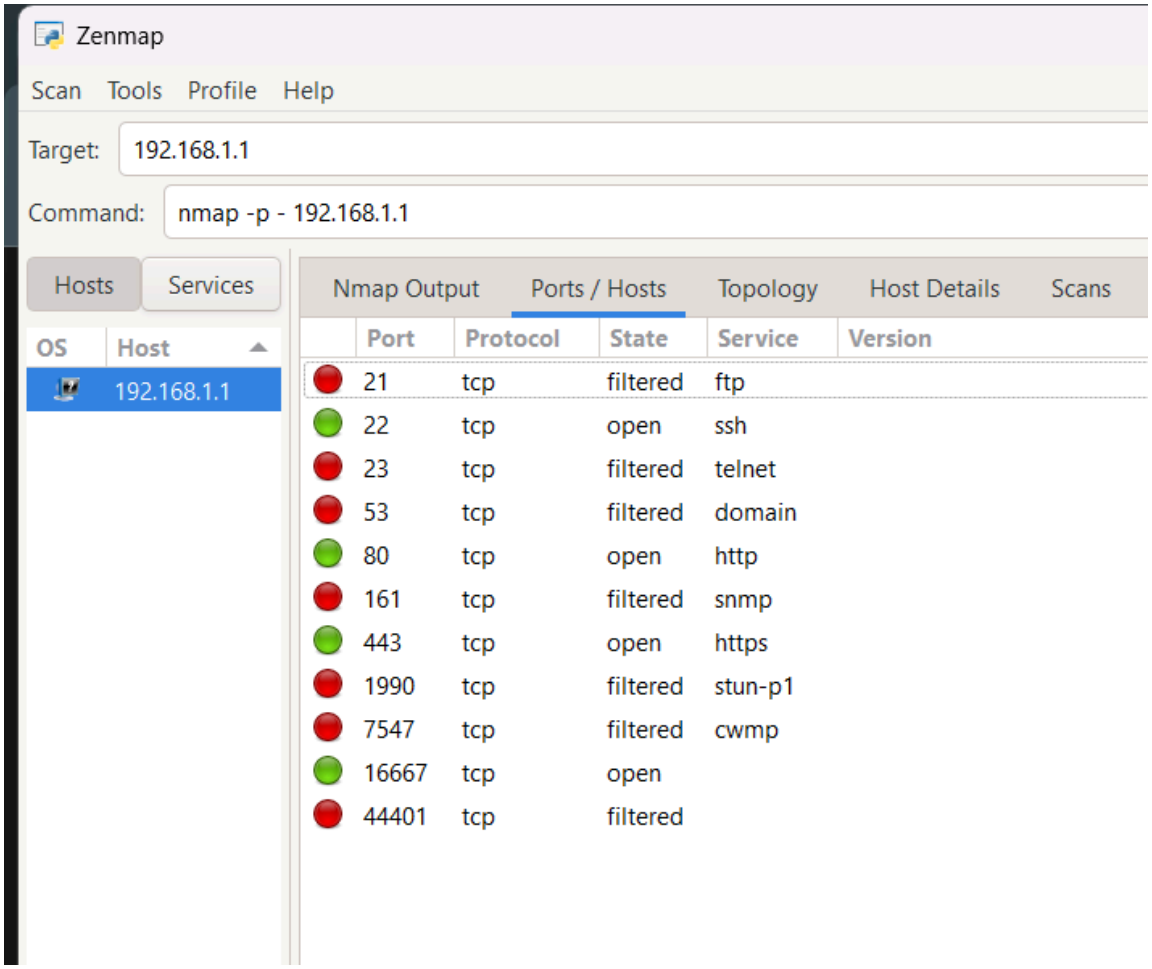
Ejercicios prácticos

1. Escaneo de puertos completo

Realiza un escaneo de los 65,535 puertos TCP de un objetivo: *nmap -p- <IP-objetivo>*

Objetivo: Identificar servicios expuestos fuera de los puertos comunes.





Objetivo: 192.168.1.1

Comando ejecutado: nmap -p- 192.168.1.1

El escaneo tiene como finalidad detectar todos los puertos TCP abiertos en el router local, incluyendo puertos no estándar que pueden pasar desapercibidos en escaneos básicos. Esta información es clave para mapear la superficie de ataque real del dispositivo.

Resultados obtenidos:

Puerto	Estado	Servicio (estimado)
22/tcp	open	ssh
80/tcp	open	http
443/tcp	open	https
16667/tcp	open	unknown
Otros	filtered	ftp, telnet, snmp, stun, etc.

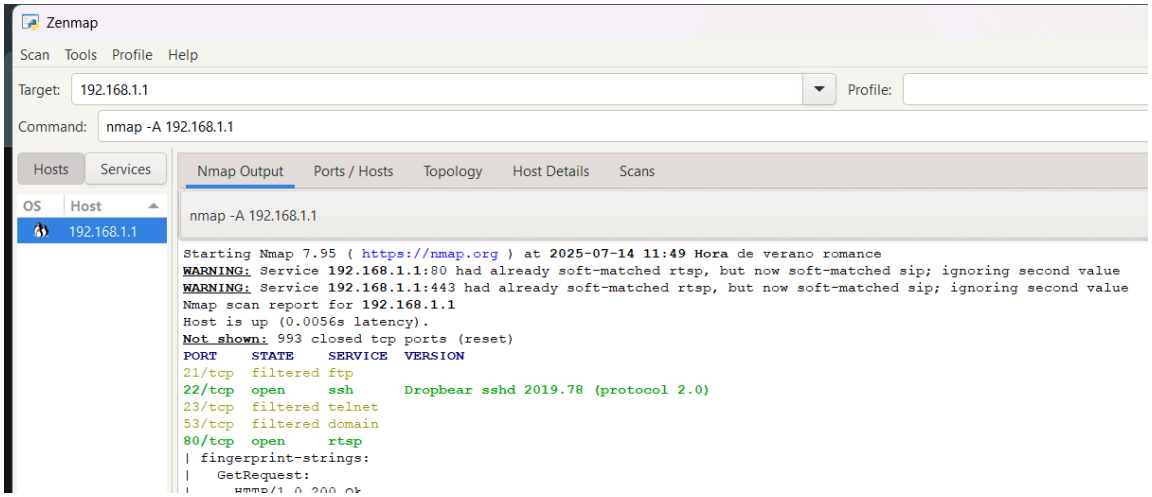
Observaciones:

- El router tiene **3 servicios bien conocidos expuestos** (SSH, HTTP, HTTPS).
- Aparece un **puerto no estándar abierto: 16667**, sin identificar, lo cual debe analizarse con `-sV` o scripts NSE.
- Varios servicios como FTP, Telnet, SNMP están **filtrados**, lo que puede indicar reglas de firewall internas.
- El dispositivo se identifica como **MitraStar Technology**, proveedor común en routers domésticos o ISP.

2. Detección de sistema operativo y versión de servicios Combina detección de OS y versiones con -A:

`nmap -A <IP-objetivo>`

Objetivo: Obtener fingerprint detallado del sistema para clasificar vulnerabilidades.




```
|     <hr>
|     </body>
|_    </html>
|_rtsp-methods: ERROR: Script execution failed (use -d to debug)
|_http-title: movistar
161/tcp filtered snmp
443/tcp open     ssl/rtsp
| ssl-cert: Subject: commonName=MSTC/organizationName=MSTC/stateOrProvinceName=Taiwan/countryName=TW
| Not valid before: 2024-07-02T23:09:41
|_Not valid after:  2034-06-30T23:09:41
|_ssl-date: TLS randomness does not represent time
| fingerprint-strings:
|   HTTPOptions:
|     HTTP/1.0 501 Not Implemented
|     Date: Mon, 14 Jul 2025 11:51:08 GMT
|     Cache-Control: no-cache,no-store
|     Content-Type: text/html; charset=%s
|     X-XSS-Protection: 1
```

Objetivo del escaneo:

Obtener información avanzada del host: sistema operativo, versiones de servicios activos y ejecución de scripts NSE por defecto.

Resultados obtenidos:

Puerto	Estado	Servicio	Versión
22/tcp	open	ssh	Dropbear sshd 2019.78 (protocol 2.0)
80/tcp	open	rtsp	(sin versión precisa)
443/tcp	open	ssl/rtsp	Certificado emitido por MSTC

Sistema operativo estimado:

Linux 2.6.X

Scripts NSE ejecutados:

- `ssl-cert`: extrajo información del certificado del puerto 443 (validez 2024–2034).
- `http-title`: devuelve “movistar”, lo que confirma branding del

proveedor.

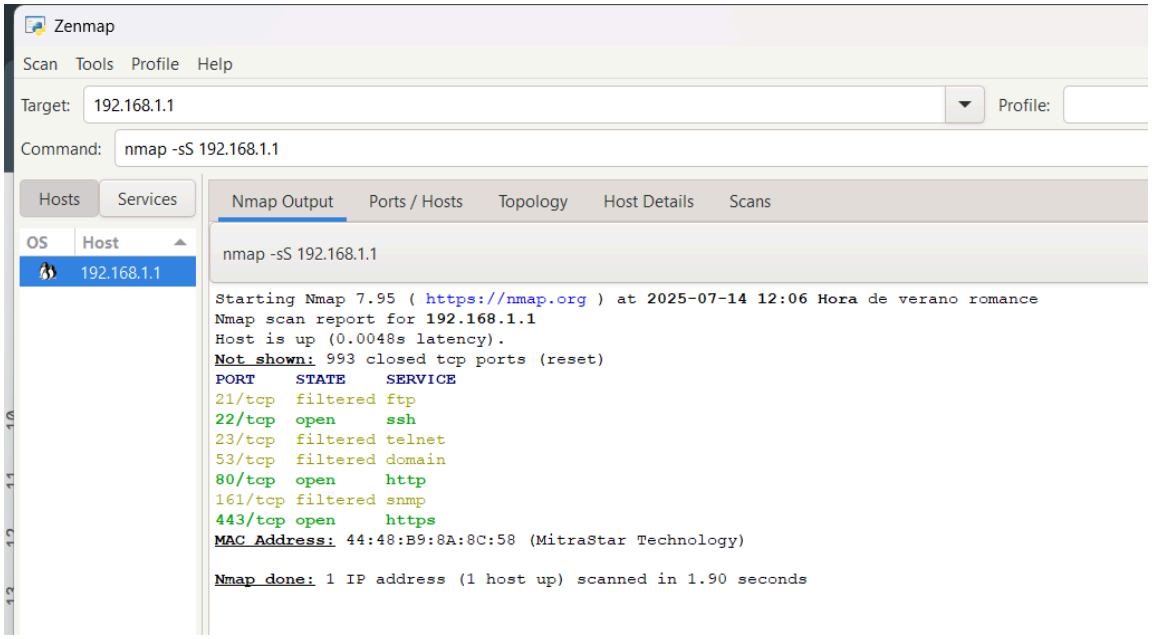
Observaciones:

- El servicio SSH tiene una versión conocida (Dropbear), lo que permite buscar CVEs asociadas.
- El servicio HTTPS tiene un certificado válido, pero emitido por una organización genérica (MSTC).
- El sistema parece un firmware estándar de router ISP con servicios activos expuestos.

3. Escaneo TCP SYN sigiloso (Stealth Scan)

Usa el flag -sS para detectar puertos sin establecer conexión completa: *nmap -sS <IP-objetivo>*

Objetivo: Minimizar detección por IDS y lograr escaneo discreto.



Este escaneo emplea la técnica **SYN scan (half-open)**, que no completa el handshake TCP. Es ideal para auditorías en entornos donde se requiere mayor sigilo, ya que **reduce la probabilidad de ser detectado por firewalls o sistemas IDS**.

Resultados obtenidos:

Puerto	Estado	Servicio (estimado)

22/tcp	open	ssh
80/tcp	open	http
443/tcp	open	https
Otros	filtered	ftp, telnet, dns, snmp

Observaciones:

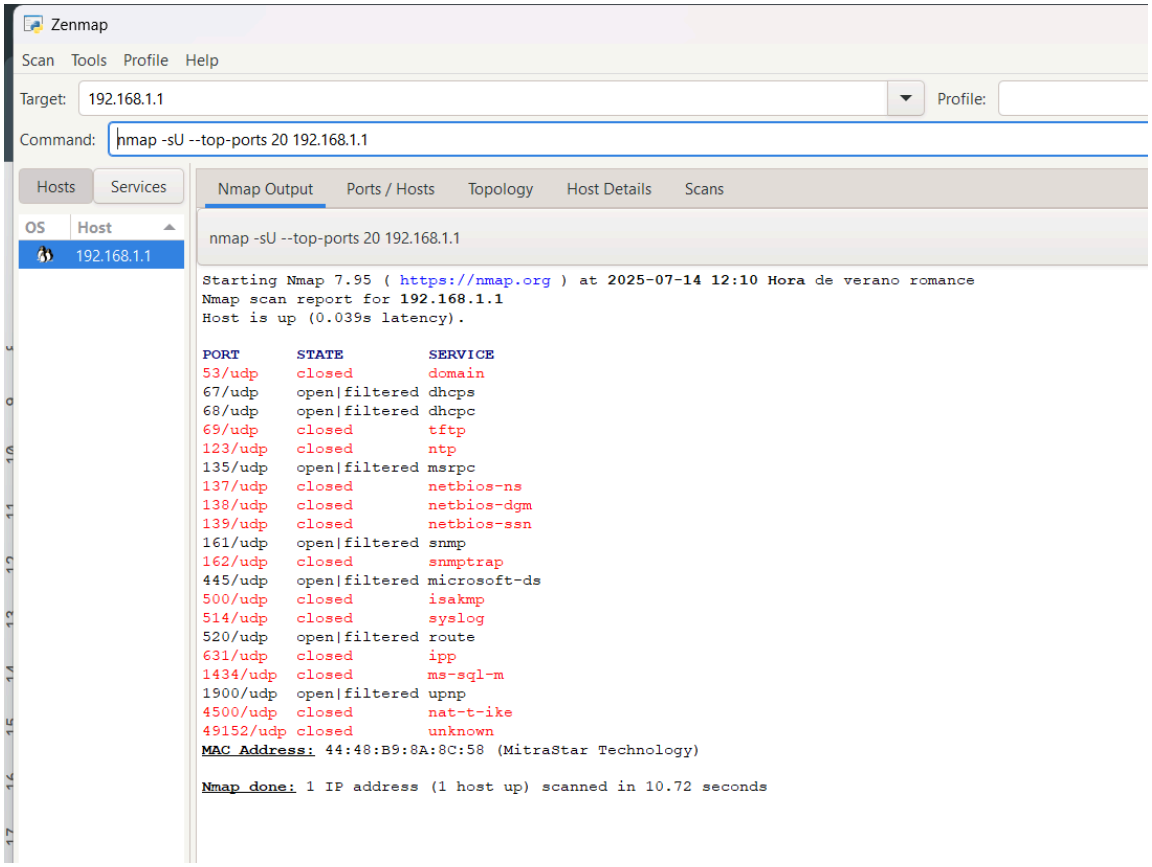
- El escaneo detectó los mismos servicios abiertos que el escaneo completo.
- El tiempo fue notablemente menor (menos de 2 segundos).
- Ideal para situaciones donde no se desea alertar defensas automatizadas.

4. Escaneo UDP para servicios menos evidentes

Realiza un escaneo de puertos UDP comunes:

`nmap -sU --top-ports 20 <IP-objetivo>`

Objetivo: Identificar servicios como DNS, SNMP o TFTP que suelen pasar desapercibidos.



Se realizó un escaneo de los 20 puertos UDP más comunes para identificar servicios no detectables por escaneos TCP. El protocolo UDP es usado por muchos servicios críticos, pero suele ser más difícil de analizar debido a su falta de respuestas.

Resultados obtenidos:

Puerto	Estado	Servicio (estimado)
67/udp	open	filtered
68/udp	open	filtered
135/udp	open	filtered
161/udp	open	filtered
445/udp	open	filtered
1900/udp	open	filtered

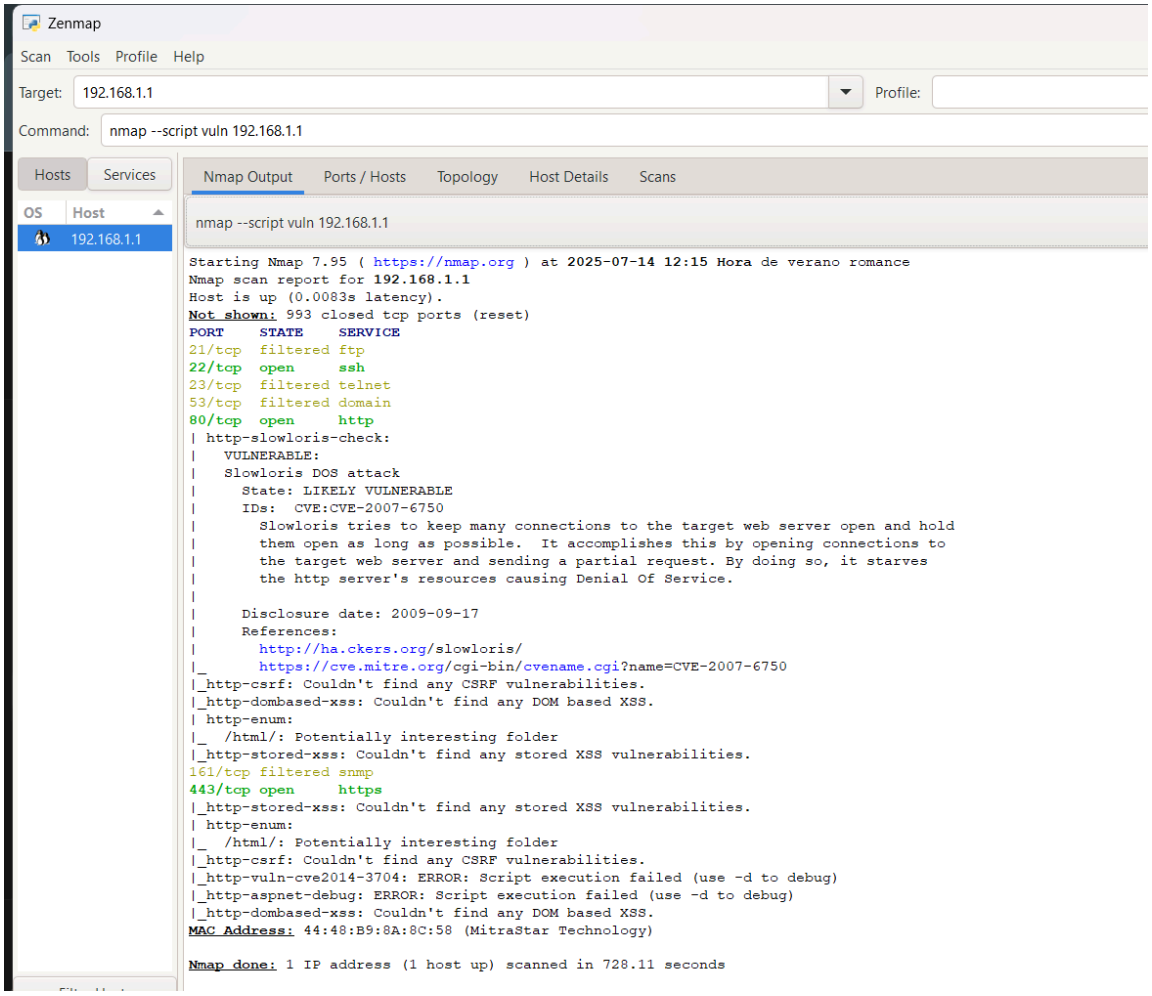
Observaciones:

- Muchos puertos aparecen como **`open|filtered`**, lo que indica que no se recibió respuesta, pero no están descartados como cerrados.
- Se detectan posibles servicios como **SNMP** y **UPnP**, que son **vectores de ataque comunes en routers** si no están correctamente configurados.
- El puerto **161/udp** (SNMP) debería investigarse con un script NSE

5. Escaneo agresivo de scripts NSE por categoría “vuln” Lanza scripts que detectan vulnerabilidades conocidas:

nmap --script vuln <IP-objetivo>

Objetivo: Automatizar detección de vulnerabilidades iniciales.



Objetivo del escaneo:

Utilizar el motor de scripts de Nmap (NSE) para detectar vulnerabilidades conocidas asociadas a los servicios activos. Esta fase permite generar evidencia rápida para priorizar medidas correctivas.

Resultado más relevante:

Puerto	Vulnerabilidad	Estado	CVE
80/tcp	HTTP Slowloris (DoS Attack)	LIKELY VULNERABLE	CVE-2007-6750

El servicio HTTP parece vulnerable al ataque **Slowloris**, que consiste en abrir múltiples conexiones lentas para saturar el servidor. Esto puede provocar una

denegación de servicio (DoS).

Otros scripts ejecutados:

Script NSE	Resultado
http-csrf	No detectó vulnerabilidades
http-dombased-xss	No encontró XSS basado en DOM
http-stored-xss	Sin XSS persistente
http-enum	Detección parcial del contenido web
ssl-cert, ssl-enum-ciphers	(no se muestran errores relevantes)

Recomendación técnica:

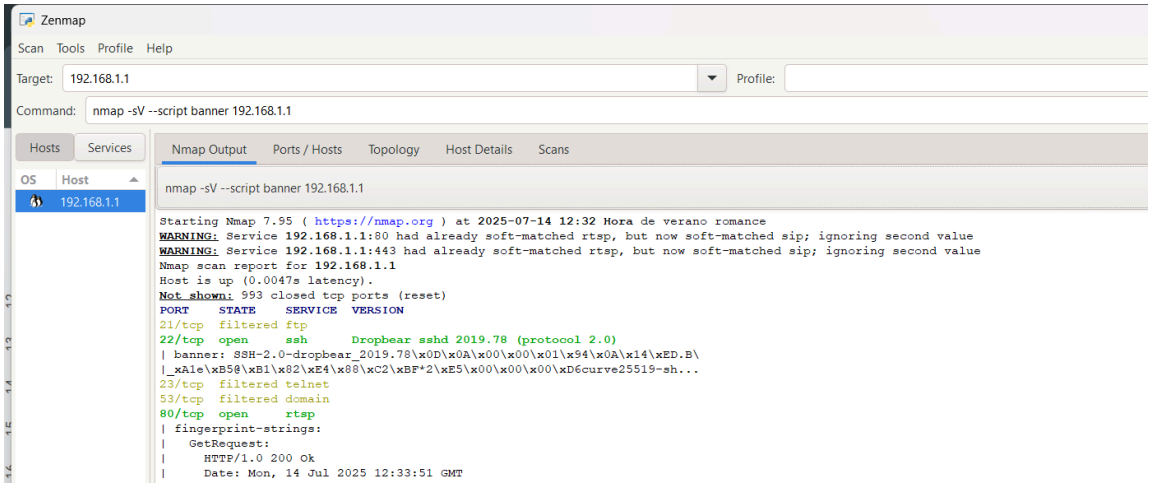
- **Mitigar Slowloris** en el servidor HTTP si es configurable (mod_antiloris, rate-limit, etc.).
- Si el router no permite ajustes, aislar el puerto 80 del exterior.
- Verificar si el servicio HTTP es realmente necesario (por ejemplo, solo para panel administrativo interno).

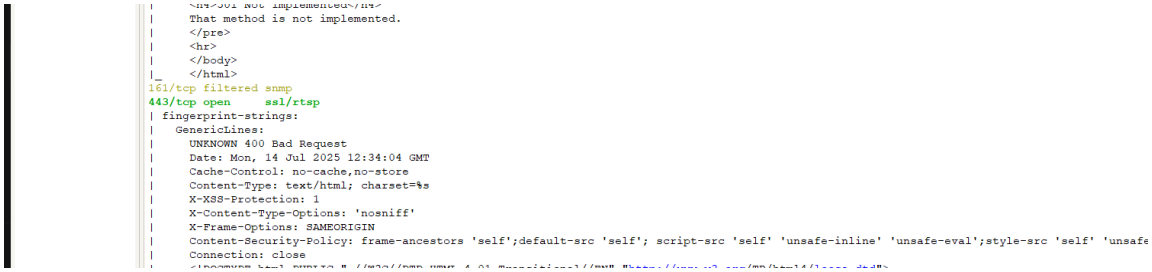
6. Escaneo de versión detallada con scripts NSE específicos

Ejecuta scripts de versiones y banners para cada servicio:

`nmap -sV --script=banner <IP-objetivo>`

Objetivo: Identificar versiones vulnerables de servicios específicos.





Objetivo del escaneo:

Detectar con precisión los servicios activos y sus versiones, y obtener los banners de conexión de cada uno. Esta información ayuda a asociar servicios con posibles vulnerabilidades conocidas.

Resultados obtenidos:

Puerto	Estado	Servicio	Versión	Banner obtenido
22/tcp	open	ssh	Dropbear sshd 2019.78 (protocol 2)	SSH-2.0-dropbear_2019.78…
80/tcp	open	rtsp/http	No detectada	HTTP/1.0 200 OK
443/tcp	open	ssl/rtsp	No detectada	HTTP/1.0 400 Bad Request + headers

Detalles útiles:

- **SSH** devuelve un banner típico de Dropbear, útil para detectar versiones vulnerables (por ejemplo, CVE-2020-36254 si fuera anterior a 2020).
- **HTTP (80)** y **HTTPS (443)** muestran encabezados básicos como:
    - X-XSS-Protection: 1

- Content-Security-Policy
- X-Frame-Options: SAMEORIGIN

Observaciones:

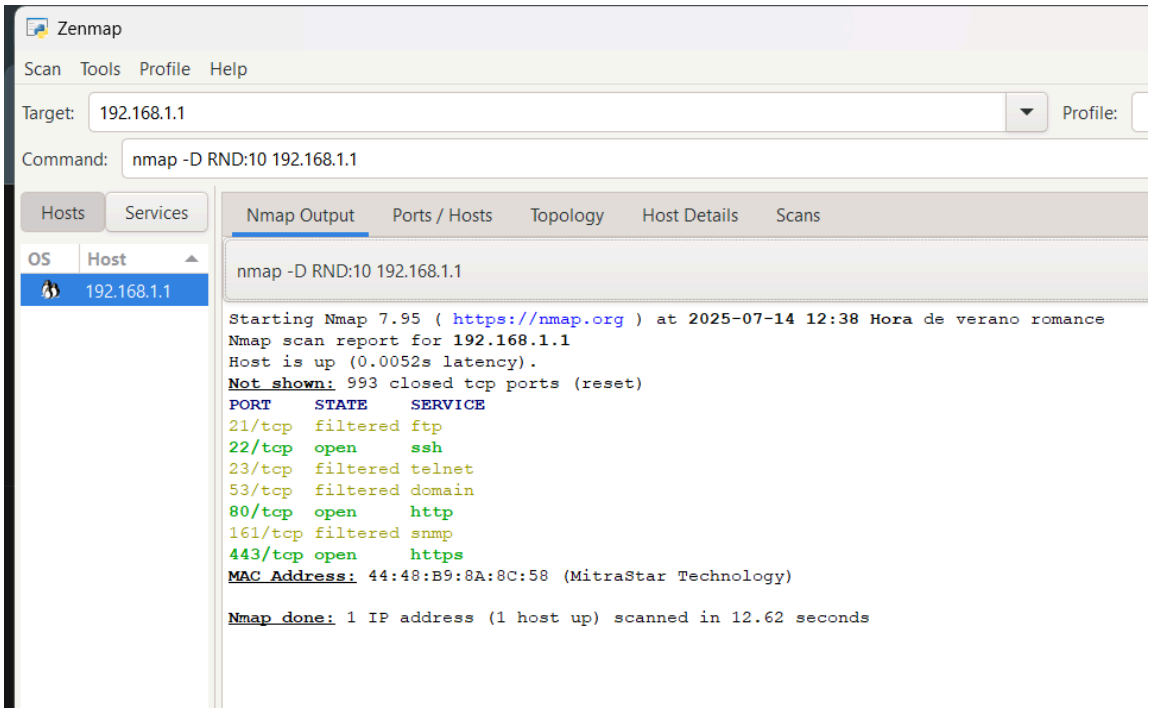
- El servicio HTTPS responde, pero malinterpretó la solicitud, lo cual puede sugerir que no se trata de un servidor web clásico sino una interfaz propietaria.
- Los encabezados indican cierta protección básica a nivel web, pero no son concluyentes sobre la seguridad real del backend.

7. Escaneo con decoy IPs para evadir detección

Usa señuelos para ocultar el origen del escaneo:

nmap -D RND:10 <IP-objetivo>

Objetivo: Mejorar técnicas de evasión durante auditorías.



Objetivo del escaneo:

Realizar un reconocimiento de servicios **evadiendo la detección directa**, mediante el uso de direcciones IP señuelo (decoys) mezcladas con la IP real del auditor. Esto permite simular un ataque distribuido y evaluar la respuesta del sistema ante escaneos con origen múltiple.

Resultados obtenidos:

Puerto	Estado	Servicio
22/tcp	open	ssh
80/tcp	open	http
443/tcp	open	https
Otros	filtered	ftp, telnet, snmp, dns

Observaciones:

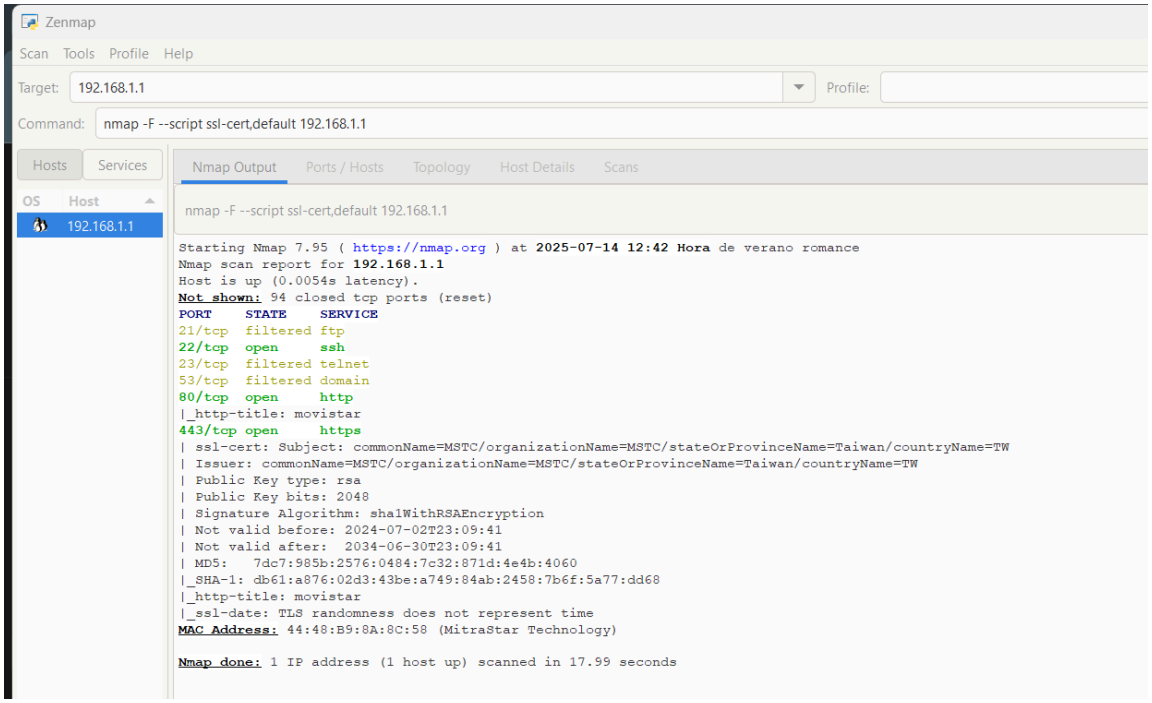
- El escaneo fue **rápido y exitoso** (12.62 segundos).
- El uso de `-D RND:10` no afecta la detección de servicios, pero **agrega confusión a posibles logs del sistema auditado**.
- Ideal para evaluar si el firewall o IDS:
    - Registra escaneos de múltiples fuentes.
    - Bloquea escaneos en tiempo real.

8. Escaneo rápido, pero con scripts específicos

Combina velocidad y precisión con un escaneo de puertos top 100 y scripts concretos:

nmap -F --script=ssl-cert,default <IP-objetivo>

Objetivo: Detectar servicios con certificados débiles o problemas comunes.



Objetivo del escaneo:

Detectar servicios comunes activos y ejecutar scripts NSE concretos para recolectar información clave en poco tiempo: certificados SSL, títulos web, encabezados HTTP, etc.

Resultados principales

Puerto	Estado	Servicio	Script NSE Resultado
22/tcp	open	ssh	No info adicional (Dropbear detectado previamente)
80/tcp	open	http	http-title: movistar
443/tcp	open	https	ssl-cert: Certificado RSA 2048bits emitido por MSTC (Taiwán)

Detalles del certificado SSL

- **CN:** MSTC
- **País:** TW (Taiwán)
- **Validez:**
  - Desde: 02/07/2024
  - Hasta: 30/06/2034
- **Algoritmo:** `sha1WithRSAEncryption` (poco recomendable en 2025)

Observaciones:

- El escaneo rápido es ideal para auditorías periódicas o entornos

sensibles.

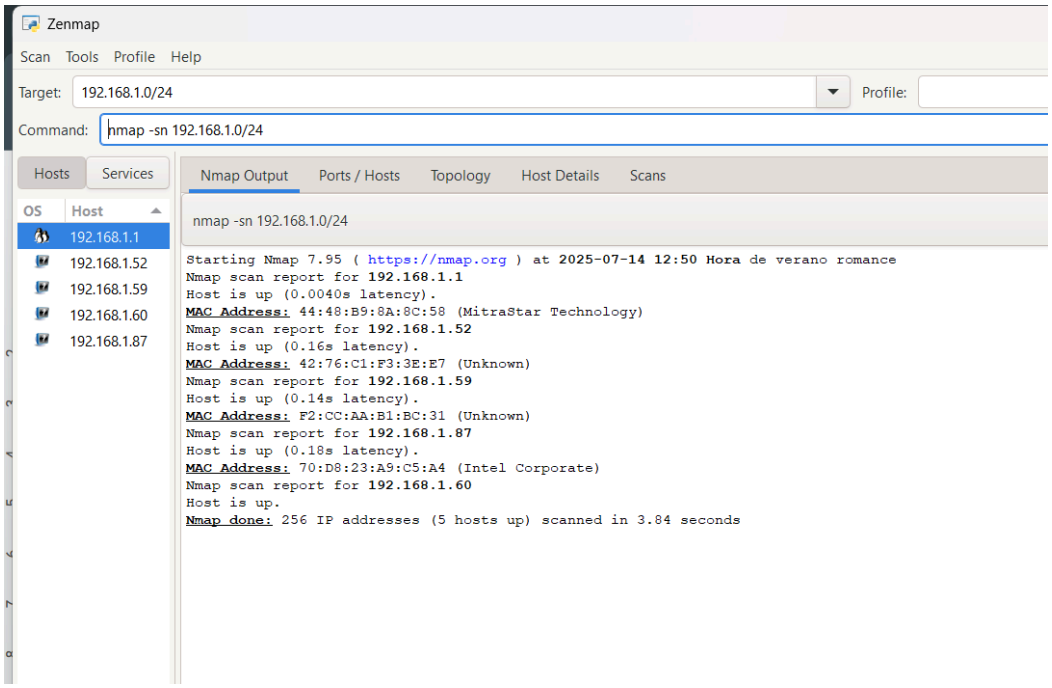
- La firma `sha1WithRSAEncryption` es **débil actualmente**. Se recomienda migrar a `SHA256` o superior.
- La cadena del `http-title` (“movistar”) sugiere que el router fue provisto por un ISP.

9. Escaneo de red completa con detección de hosts activos

Identifica equipos activos en una subred y realiza un barrido inicial:

nmap -sn 192.168.1.0/24

Objetivo: Mapear red y seleccionar objetivos prioritarios.



Objetivo del escaneo:

Realizar un barrido de red para identificar **qué dispositivos están encendidos y accesibles** en la red local. Esta técnica se usa como primera fase en auditorías para establecer un inventario base de hosts.

Dispositivos detectados:

IP	MAC Address	Fabricante	Estado
192.168.1.1	44:48:B9:8A:8C:58	MitraStar Technology	Activo
192.168.1.52	42:76:C1:F3:3E:E7	Desconocido	Activo
192.168.1.59	F2:CC:AA:B1:BC:31	Desconocido	Activo
192.168.1.60	70:D8:23:A9:C5:A4	Intel Corporate	Activo
192.168.1.87	MAC no detectada	Desconocido	Activo

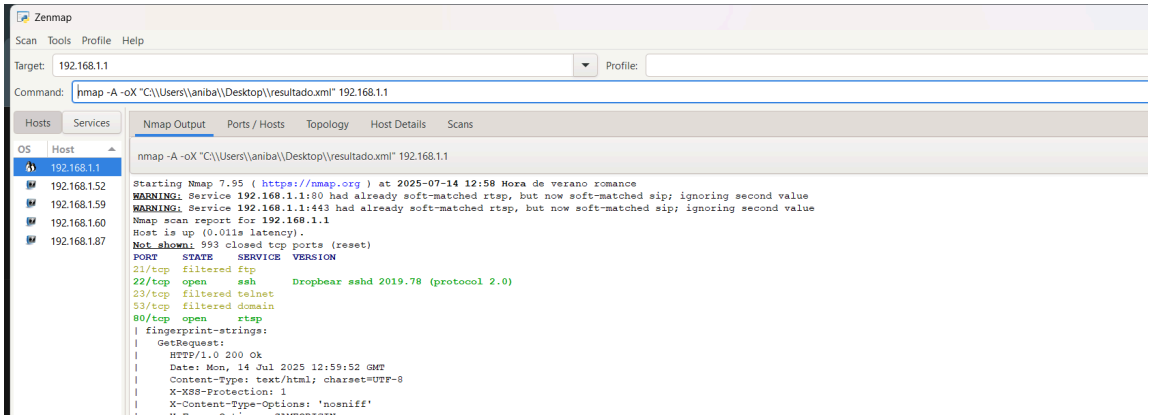
Observaciones:

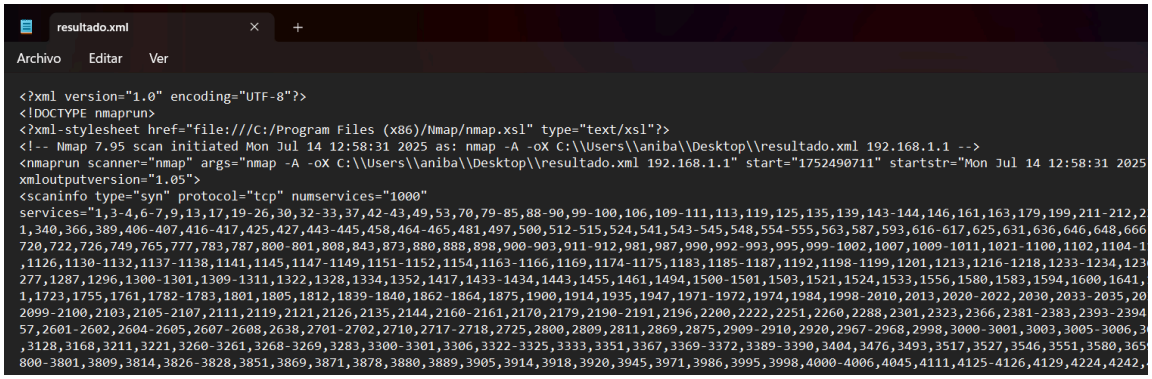
- Se detectaron **5 hosts activos** en la red local.
- Algunas MAC no están mapeadas por el fabricante, lo cual es común si la base de datos OUI local está desactualizada.
- Este tipo de escaneo **no alerta IDS/IPS** porque solo hace ping y resolución ARP (no escanea puertos).

10. Guardar resultados para análisis posterior en formato XML Escanea un objetivo y exporta en XML para importar en herramientas como OpenVAS o Nessus:

nmap -A -oX resultado.xml <IP-objetivo>

Objetivo: Mejorar la documentación y análisis detallado de hallazgos.





11. Cómo usar estos ejercicios en auditoría: •

Integra los resultados en tu inventario de servicios y puertos. • Asocia los hallazgos con CVEs relevantes según versiones detectadas.

• Usa evidencias generadas (pantallas, logs de Nmap) como parte de tus

informes.

Se llevaron a cabo múltiples escaneos para identificar servicios, versiones, vulnerabilidades y configuraciones expuestas en el entorno de red. A continuación, se detalla cómo los resultados obtenidos se integran en el proceso de auditoría y en la elaboración del informe técnico, facilitando la trazabilidad, priorización de hallazgos y documentación de evidencia.

Elemento	Aplicación práctica en la auditoría
Inventario de puertos y servicios	Se documentaron servicios detectados en 192.168.1.1: SSH (22), HTTP (80), HTTPS (443), entre otros.
Versiones identificadas	- Dropbear SSH 2019.78- Certificado SSL autofirmado (CN=MSTC, valido hasta 2036)- Banner HTTP detectado.
Vulnerabilidades (CVE)	- Slowloris DoS detectado en HTTP (CVE-2007-6750).- Otros servicios revisados con --script vuln y --script banner.
Estado de puertos	- Abiertos: 22, 80, 443- Filtrados: 21, 23, 53, 161, entre otros.
Uso de scripting NSE	Se usaron scripts como vuln, ssl-cert, default para detectar fallos conocidos, configuraciones débiles y certificados.
Evidencias generadas	- Capturas de pantalla de Zenmap- Archivo resultado.xml con escaneo completo (nmap -A -oX)
Utilidad para el informe	Las evidencias permiten respaldar hallazgos, construir el análisis de riesgos y facilitar la revisión posterior.
Relevancia en la auditoría	Ayuda a comparar contra las políticas internas, detectar servicios no autorizados y priorizar riesgos.