

curso “Experto en Ciberseguridad de Redes y Sistemas”

Paso previo: Instalación de Máquina Virtual Kali.

TEMAS

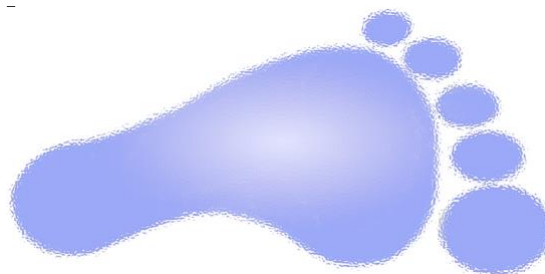
1. Fingerprinting
2. Footprinting
3. IDSs - Snort
4. Nessus
5. Firewalls
6. SGSI
7. Antiphishing
8. MITRE
9. Red Team y Blue Team
10. Principales procesos de Seguridad



Tema 1. Footprinting

Desarrollaremos los siguientes sub-temas:

Desarrollo



2.1 Presentación

El **footprinting** es la continuación natural del tema anterior. Se trata de un proceso más avanzado de reconocimiento en ciberseguridad, cuyo objetivo es recopilar la mayor cantidad posible de información sobre un objetivo antes de interactuar directamente con sus sistemas. A diferencia del *fingerprinting* —más centrado en identificar características técnicas concretas de sistemas ya detectados—, el footprinting se enfoca en construir una visión global del entorno.

Concepto general

El footprinting consiste en reunir datos relevantes sobre una organización, red o sistema sin necesidad de intrusión directa. Esta información puede incluir:

- Dominios y subdominios registrados
- Direcciones IP públicas
- Infraestructura de red visible
- Puertos y servicios
- Tecnologías utilizadas (servidores, dispositivos, proveedores)

- Información pública de empleados o estructura organizativa

Enfoque y evolución

Con el crecimiento de Internet y la digitalización de empresas, el footprinting ha cobrado una importancia clave. Hoy en día, una gran cantidad de información sensible puede obtenerse sin necesidad de escaneos activos, simplemente analizando la huella digital que una organización deja en la red.

Tipos de footprinting

Se suele distinguir entre dos enfoques principales:

- **Footprinting pasivo:**
No interactúa directamente con el objetivo. Se basa en información pública, minimizando la posibilidad de ser detectado.
- **Footprinting activo:**
Implica cierto grado de interacción con los sistemas del objetivo (por ejemplo, consultas DNS específicas), lo que puede dejar rastro.

Importancia en ciberseguridad

El footprinting es una etapa crítica porque sienta las bases para fases posteriores como la enumeración y el análisis de vulnerabilidades. Cuanta más información se recopile en esta fase, más preciso y eficaz será el resto del proceso.

Además, desde el punto de vista defensivo, entender el footprinting permite a las organizaciones evaluar qué información están exponiendo públicamente y reducir su superficie de ataque.

2.2 “nmap”, primeros pasos

Nmap (Network Mapper) es una herramienta de código abierto ampliamente utilizada en el ámbito de la ciberseguridad y la administración de redes. Su función principal es la exploración y el análisis de redes, permitiendo descubrir dispositivos conectados, identificar puertos abiertos, servicios activos y posibles vulnerabilidades.

Gracias a su flexibilidad y potencia, Nmap es empleado tanto por administradores de sistemas para tareas de mantenimiento y auditoría como por profesionales de seguridad para evaluar la exposición de una red frente a posibles amenazas. Además, ofrece múltiples opciones de escaneo y generación de informes, lo que lo convierte en una herramienta esencial para el análisis de infraestructuras de red.

En este punto desarrollaremos varias opciones del comando “nmap” para investigar diferentes tipos de configuración, dentro de una red doméstica. La mejor forma de practicar y ejecutar cada una de ellas es siguiendo el video que se presenta a continuación (**Obligatorio**).

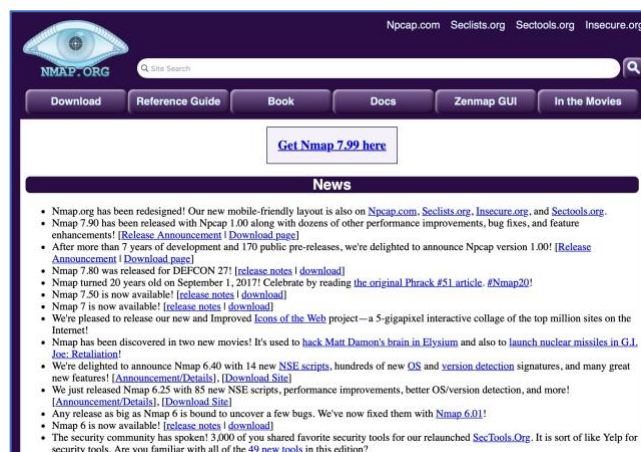
🌐 **Footprinting (Parte 1 - nmap) - Charla 88 - Aprendiendo Ciberseguridad**
<https://youtu.be/jsKNXx41mPc>



2.3 “nmap” NSE

Sigamos avanzando con el comando nmap, ahora comenzaremos a investigar lo que nos ofrece desde su página web: <https://nmap.org>

Dentro de su página web, presentaremos el acceso a toda la documentación que nos ofrece, pero más allá de esto, lo que de verdad nos interesa es el tema de “NSE”: Nmap Scripting Engine.



El NSE, es una de las funcionalidades más potentes y versátiles de nmap, diseñada para ampliar sus capacidades mediante scripts. Permite automatizar tareas complejas relacionadas con el análisis de redes, la detección de vulnerabilidades y la recopilación de información.

¿Qué es NSE?

NSE es un motor de ejecución de scripts integrado en nmap que utiliza el lenguaje **Lua**. Gracias a él, los usuarios pueden ejecutar scripts predefinidos o crear los suyos propios para adaptar los escaneos a necesidades específicas.

¿Para qué sirve?

Con NSE se pueden realizar múltiples tareas, entre ellas:

- Detección de vulnerabilidades en servicios y sistemas.
- Enumeración de servicios (usuarios, recursos compartidos, versiones, etc.).
- Obtención de información adicional sobre hosts (banners, configuraciones).
- Automatización de auditorías de seguridad.

Tipos de scripts

Los scripts de NSE se organizan en diferentes categorías según su propósito:

- safe: seguros, no intrusivos.
- intrusive: más agresivos, pueden afectar al sistema objetivo.
- vuln: orientados a detectar vulnerabilidades conocidas.
- discovery: recopilación de información.
- auth: relacionados con autenticación.

Uso básico

Para ejecutar scripts con Nmap se utiliza la opción --script. Por ejemplo:

nmap --script=vuln <objetivo>

Esto ejecuta todos los scripts de la categoría de vulnerabilidades sobre el objetivo especificado.

Ventajas principales

- Gran flexibilidad y personalización.
- Amplia biblioteca de scripts ya disponibles.
- Posibilidad de crear scripts propios.
- Integración directa con los resultados de Nmap.

En resumen, el NSE convierte a nmap en una herramienta mucho más completa, permitiendo pasar de un simple escáner de puertos a una plataforma avanzada de análisis y auditoría de seguridad.

Para avanzar de forma práctica, os invitamos a ver el video 89 **(Obligatorio)**.

 **Footprinting (parte 2 - nmap "Scripting") - Charla 89 -**
Aprendiendo Ciberseguridad paso a paso
<https://youtu.be/Taj15VD1UOs>



2.4 “nmap” NSE sobre Telnet

En el video siguiente, seguiremos avanzando con NSE, esta vez profundizando sobre la investigación de los protocolos inseguros que acabamos de ver que están presentes en el video 89.

El uso del **NSE** cobra especial relevancia cuando se analizan protocolos antiguos o inseguros como **Telnet**, ampliamente conocido por sus debilidades en entornos modernos.

Telnet como protocolo inseguro

Telnet es un protocolo de comunicación utilizado para acceder de forma remota a otros sistemas a través de la red. Sin embargo, presenta un problema crítico: **transmite la información en texto plano**, incluyendo credenciales como nombres de usuario y contraseñas. Esto lo hace vulnerable a ataques de interceptación (sniffing) y a accesos no autorizados.

NSE aplicado a Telnet

Gracias a NSE, es posible detectar y analizar servicios Telnet activos en una red de forma automatizada. Algunos usos habituales incluyen:

- **Identificación de servidores Telnet abiertos** en el puerto 23.
- **Enumeración de banners**, que puede revelar información del sistema o versión del servicio.
- **Pruebas de autenticación básica**, para detectar configuraciones débiles.
- **Detección de vulnerabilidades conocidas** asociadas a implementaciones de Telnet.

Por ejemplo, nmap permite ejecutar scripts específicos relacionados con Telnet:

nmap -p 23 --script telnet-* <objetivo>

Importancia en auditorías de seguridad

El análisis de Telnet mediante NSE es clave porque:

- Permite **identificar servicios obsoletos** que deberían sustituirse por alternativas seguras como SSH.
- Ayuda a detectar **malas configuraciones** o accesos innecesarios.
- Facilita la **evaluación del riesgo real** en una red.

En definitiva, combinar NSE con el análisis de protocolos inseguros como Telnet permite a los profesionales de seguridad no solo detectar problemas, sino también priorizar su mitigación dentro de una estrategia de protección más amplia.

Todo esto lo desarrollamos en el video 91 **(Obligatorio)**.

 **Footprinting (parte 3 - nmap Scripting continuación) - Charla 91** - Aprendiendo Ciberseguridad paso a paso
<https://youtu.be/s7h9x6q7pvk>



2.5 “nmap” script y bash

En este video, continuaremos empleando NSE scripting, y a su vez, desarrollaremos algún programa muy básico en bash para combinar ambas técnicas.

La **programación en bash** consiste en escribir scripts utilizando el intérprete de comandos **bash** (Bourne Again Shell), habitual en sistemas GNU/Linux y macOS. Permite automatizar tareas del sistema, gestionar archivos, ejecutar comandos en secuencia y crear flujos de trabajo eficientes sin necesidad de lenguajes más complejos.

Un script bash es, en esencia, un archivo de texto con una serie de comandos que el sistema ejecuta de forma ordenada. Además, bash ofrece estructuras básicas de programación como variables, condicionales (if), bucles (for, while) y funciones, lo que facilita la creación de scripts más avanzados.

Se utiliza principalmente para:

- Automatizar tareas repetitivas (copias de seguridad, instalaciones, mantenimiento).
- Administrar sistemas y servidores.
- Procesar archivos y datos desde la línea de comandos.

En resumen, bash es una herramienta fundamental para administradores de sistemas y profesionales técnicos, ya que permite ahorrar tiempo y mejorar la eficiencia mediante la automatización.

Todo esto lo desarrollamos en el video 92 **(Obligatorio)**.

 **Footprinting (parte 4 - 2x1 Scripts con nmap y bash) - Charla 92**
- Aprendiendo Ciberseguridad paso a paso
<https://youtu.be/EEZVEIUuSUo>



2.6 footprinting con Wireshark

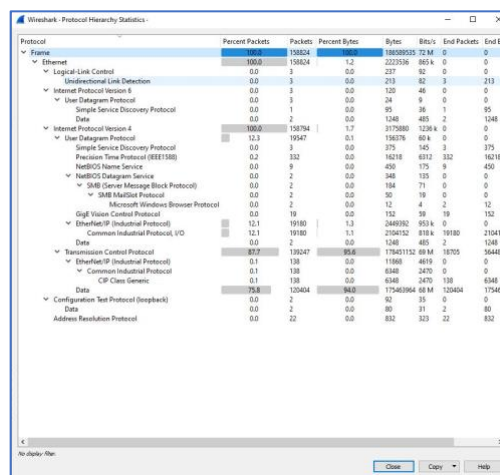
El **footprinting** con **Wireshark** va más allá de una simple observación del tráfico: permite reconstruir cómo funciona una red, qué sistemas la componen y qué servicios están activos, todo ello de forma pasiva (sin generar tráfico adicional). Esto lo convierte en una técnica muy valiosa en fases iniciales de auditoría o análisis.

Profundizando en el análisis con Wireshark

2.6.1. Correlación de estadísticas

Las distintas vistas del menú “Estadísticas” no deben analizarse de forma aislada:

- **Jerarquía de protocolos + IPv4 Statistics:** permite detectar qué protocolos dominan la red (por ejemplo, mucho SMB o DNS puede indicar entorno corporativo Windows).
- **Conversaciones + Puntos finales:** ayuda a identificar qué equipos son más activos o actúan como servidores.
- **Direcciones resueltas:** al asociar MAC con fabricantes (Cisco, HP, etc.), puedes deducir el tipo de dispositivo (router, impresora, PC...).



Esto permite construir un mapa lógico de la red sin necesidad de escanearla directamente.

2.6.2. Análisis de comportamiento del tráfico

La **gráfica de flujo (Flow Graph)** es especialmente útil para:

- Entender secuencias de comunicación (por ejemplo, autenticaciones).
- Detectar patrones repetitivos (consultas, conexiones periódicas).
- Identificar posibles anomalías o comportamientos sospechosos.

También puedes seguir flujos TCP completos (**Follow TCP Stream**) para ver el contenido de las comunicaciones, lo cual es crítico si el tráfico no está cifrado.



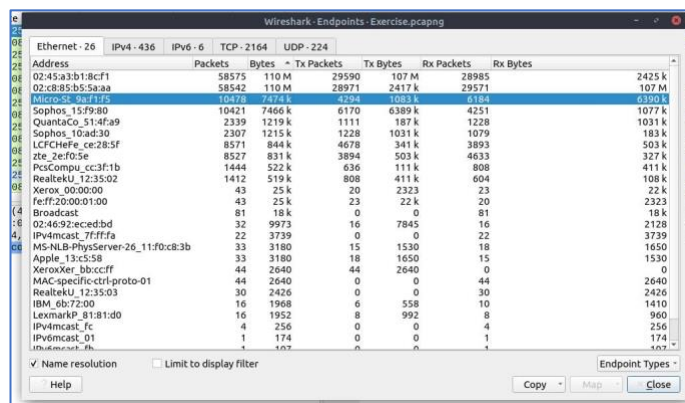
2.6.3. Importancia de NetBIOS y tráfico Windows

Los filtros como nbns.name y nbdgm son clave en redes Windows:

- Permiten **descubrir nombres de equipos y dominios** sin autenticación.
- Revelan roles dentro de la red (controladores de dominio, servidores, etc.).
- Facilitan la identificación de recursos compartidos.

Los códigos NetBIOS (como 1B, 1D, 20) aportan información muy valiosa:

- Saber quién es el **Domain Master Browser** permite ubicar el núcleo de la red.
- Detectar servidores de archivos (20) puede indicar objetivos críticos.
- Identificar estaciones de trabajo (00) ayuda a estimar el tamaño de la red.



Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes
02:45:a3:b1:8c:f1	58575	110 M	29590	107 M	28985	2425 k
02:c8:85:b5:5a:aa	58542	110 M	28971	2417 k	29571	107 M
Macosx-3114-p	10421	7466 k	6170	6389 k	4251	3303 k
Sophos-15-f9-90	2339	1219 k	1111	187 k	1228	1031 k
QuantaCo_51:4fa9	2307	1215 k	1228	1031 k	1079	183 k
Sophos_10ad:30	8571	844 k	4678	341 k	3893	503 k
LCrCheFe_cce28:5f	8527	831 k	3894	503 k	4633	327 k
zte_2e:f0:5e	1444	522 k	636	111 k	808	411 k
PcsCompu_cc3f:1b	1412	519 k	808	411 k	604	108 k
RealtekU_12:35:02	43	25 k	20	2323	23	22 k
Xerox_00:00:00	43	25 k	23	22 k	20	2323
feff:20:00:01:00	81	18 k	0	0	81	18 k
Broadcast	32	9973	16	7845	16	2128
02:4e:92:ce:cd:bd	22	3739	0	0	22	3739
IPV4mcast_7f:ff:fa	33	3180	15	1530	18	1650
MS-NLB-PhysServer-26_11:f0:c8:3b	33	3180	18	1650	15	1530
Apple_13:c5:38	44	2640	44	2640	0	0
XeroxKer_bb:cc:ff	44	2640	0	0	44	2640
MAC-specific-ctrl-prot-01	30	2426	0	0	30	2426
RealtekU_12:35:03	16	1968	6	558	10	1410
IBM_db:72:00	16	1952	8	992	8	960
LexmarkP_81:81:d0	4	256	0	0	4	256
IPV4mcast_fc	1	174	0	0	1	174
IPV6mcast_01	1	167	0	0	1	167

2.6.4. Fingerprinting pasivo

A diferencia de herramientas activas como Nmap, Wireshark permite hacer **fingerprinting pasivo**:

- Identificación de sistemas operativos por patrones de tráfico.
- Detección de servicios sin enviar paquetes.
- Análisis de TTL, tamaños de ventana TCP, etc.

Esto reduce el riesgo de ser detectado en entornos monitorizados.

2.6.5. Limitaciones y contexto actual

Aunque es muy potente, hay que tener en cuenta:

- El uso extendido de **cifrado (HTTPS, SSH, VPN)** limita la visibilidad del contenido.
- Es necesario estar en una posición adecuada de red (misma LAN o tráfico replicado).
- Puede generar grandes volúmenes de datos que requieren filtrado cuidadoso.

Conclusión

El footprinting con Wireshark es una técnica avanzada que combina análisis estadístico, protocolos y correlación de datos para obtener una visión completa de la red. Bien utilizado, permite identificar activos, servicios y relaciones entre sistemas de forma silenciosa, siendo una herramienta clave en ciberseguridad tanto para análisis defensivo como auditorías.

Todo esto lo desarrollamos en el video 93 (**Obligatorio**).

 **Footprinting (parte 5 - Aprovechando Wireshark) - Charla 93 -**
Aprendiendo Ciberseguridad paso a paso
<https://youtu.be/dlNKvXMeq3c>



2.7 footprinting, protocolos inseguros - HTTP

En el ámbito de las redes y la seguridad informática, los protocolos de comunicación desempeñan un papel fundamental en el intercambio de información entre dispositivos. Uno de los protocolos más utilizados históricamente es **HTTP** (HyperText Transfer Protocol), base de la comunicación en la web. Sin embargo, HTTP se considera un protocolo inseguro debido a la forma en que transmite los datos.

HTTP funciona mediante un modelo de petición-respuesta entre un cliente (como un navegador web) y un servidor. El problema principal radica en que la información se envía en texto plano, sin ningún tipo de cifrado. Esto implica que cualquier atacante con acceso a la red puede interceptar, leer e incluso modificar los datos transmitidos, mediante técnicas como el “sniffing” o ataques de intermediario (Man-in-the-Middle).

Esta falta de seguridad hace que HTTP sea especialmente vulnerable cuando se transmiten datos sensibles, como credenciales de acceso, información personal o datos bancarios. Como respuesta a estas debilidades, se desarrolló **HTTPS** (HTTP Secure), que incorpora cifrado mediante protocolos como **TLS/SSL** (Transport Layer Security / Secure Socket Layer) para proteger la confidencialidad e integridad de la información.

En la actualidad, el uso de HTTP sin cifrado está cada vez más desaconsejado, especialmente en entornos donde la seguridad es crítica. Comprender sus limitaciones es esencial para valorar la importancia de implementar protocolos seguros en las comunicaciones digitales.

Para comprender cómo analizar sus debilidades te presentamos el siguiente video **(Obligatorio)**.

 **Footprinting (Protocolos inseguros - HTTP) - Charla 95 -**
Aprendiendo Ciberseguridad paso a paso
<https://youtu.be/D5VKMiltqAk>



2.8 footprinting, protocolos inseguros - Telnet

Sobre el protocolo Telnet, acabamos de hablar en el punto 2.4, así que no nos extenderemos más sobre el tema. El único aspecto que sí deseamos resaltar es la importancia de emplear SSH (Secure SHell) versión 2 para el acceso a dispositivos remotos, pues tal cual definimos unos puntos más arriba y pudimos verificar, el protocolo “Telnet” JAMÁS debe emplearse, o al menos, solamente en entornos muy cerrados y para tareas triviales.



En el video que se presenta a continuación, empleando nuevamente Wireshark vamos a ver cómo se puede capturar toda la sesión Telnet y a través del “seguimiento de flujos” que nos ofrece esta herramienta, veremos toda la actividad que generó ese usuario. Por lo tanto, pasemos directamente a verlo con el siguiente video **(Obligatorio)**.

 **Footprinting (Protocolos inseguros - Telnet) - Charla 96 -**
Aprendiendo Ciberseguridad paso a paso
<https://youtu.be/utmaacFHLS8>



2.9 footprinting, protocolos inseguros - FTP

El protocolo **FTP** (File Transfer Protocol) es un estándar de red utilizado para la transferencia de archivos entre sistemas conectados a una red. Fue desarrollado en la década de 1970 y ha sido ampliamente utilizado para compartir información entre clientes y servidores.

FTP se basa en un modelo cliente-servidor. Un usuario (cliente) se conecta a un servidor FTP para subir (upload) o descargar (download) archivos. Para establecer la conexión, normalmente se requiere un nombre de usuario y contraseña, aunque también existe el acceso anónimo.

Canales de comunicación

FTP utiliza dos canales separados:

- Canal de control: se encarga de enviar comandos y respuestas.
- Canal de datos: se utiliza para transferir los archivos.

Este diseño permite gestionar múltiples transferencias, pero también añade complejidad.

FTP puede operar en dos modos de conexión:

- Modo activo: el servidor inicia la conexión de datos hacia el cliente.
- Modo pasivo: el cliente inicia ambas conexiones, siendo más compatible con firewalls.

Ventajas

- Permite transferencias de archivos grandes.
- Es compatible con múltiples sistemas operativos.
- Facilita la gestión de archivos en servidores remotos.

Inconvenientes y seguridad

FTP es considerado un protocolo inseguro porque:

- Transmite datos en texto plano (incluyendo credenciales).
- Es vulnerable a ataques de interceptación.
- No cifra la información.

Por estas razones, ha sido reemplazado en muchos casos por alternativas más seguras como **FTPS** (FTP Secure) y **SFTP** (SSH File Transfer Protocol).

Conclusión

FTP ha sido un protocolo clave en la historia de Internet para la transferencia de archivos. Sin embargo, sus limitaciones en seguridad han impulsado la adopción de soluciones más modernas y seguras. Aun así, comprender su funcionamiento sigue siendo fundamental para el estudio de redes y sistemas.

Para analizarlo de forma práctica pasemos a ver el siguiente video (**Obligatorio**).

 **Footprinting (Protocolos inseguros - FTP) - Charla 97 -**
Aprendiendo Ciberseguridad paso a paso
<https://youtu.be/OHt7SckbktY>



2.10 footprinting, protocolos inseguros – SNMP parte 1

El protocolo **SNMP** (Simple Network Management Protocol) es un estándar ampliamente utilizado para la monitorización y gestión de dispositivos en redes IP. Permite a los administradores supervisar el estado, rendimiento y configuración de equipos como routers, switches, servidores, impresoras y otros dispositivos conectados.

Se basa en un modelo de gestión compuesto por tres elementos principales:

- **Gestor (Manager):** sistema central que supervisa la red y toma decisiones.
- **Agente (Agent):** software que se ejecuta en los dispositivos gestionados y recopila información.
- **Dispositivos gestionados:** equipos de red que contienen los agentes y proporcionan datos al gestor.

La comunicación entre el gestor y los agentes se realiza mediante mensajes SNMP.

Base de información de gestión (MIB)

La **MIB** (Management Information Base) es una base de datos estructurada que almacena la información que puede ser consultada o modificada mediante SNMP. Está organizada jerárquicamente en forma de árbol y cada elemento se identifica mediante un **OID** (Object Identifier).

Ejemplos de información almacenada en la MIB:

- Uso de CPU
- Estado de interfaces de red
- Número de paquetes transmitidos
- Errores de red

SNMP define varias operaciones básicas para la gestión de red:

- **GET:** el gestor solicita información al agente.
- **GET-NEXT:** permite recorrer secuencialmente la MIB.
- **SET:** modifica un valor en el dispositivo.

- **TRAP:** el agente envía notificaciones al gestor de forma espontánea.
- **INFORM:** similar a TRAP, pero con confirmación de recepción.

Existen varias versiones del protocolo, cada una con mejoras en funcionalidad y seguridad:

SNMPv1: versión inicial, sencilla pero con escasa seguridad.

SNMPv2c: mejora el rendimiento, pero mantiene debilidades en seguridad.

SNMPv3: incorpora autenticación y cifrado, siendo la versión recomendada actualmente.

Las primeras versiones de SNMP utilizan “community strings” (como contraseñas en texto plano), lo que las hace vulnerables a ataques. SNMPv3 mejora este aspecto mediante:

- Autenticación de usuarios
- Cifrado de datos
- Control de acceso

Aun así, es importante configurar adecuadamente el protocolo para evitar accesos no autorizados.

Ventajas

- Permite monitorizar redes de forma centralizada.
- Es compatible con gran variedad de dispositivos.
- Facilita la detección temprana de fallos.

Inconvenientes

- Versiones antiguas inseguras.
- Puede generar tráfico adicional en la red.
- Configuración compleja en entornos grandes.

Usos habituales

SNMP se emplea en:

- Sistemas de monitorización de red
- Gestión de infraestructuras IT
- Supervisión de rendimiento
- Automatización de alertas

Conclusión

SNMP es una herramienta esencial en la administración de redes, ya que permite obtener información en tiempo real sobre el estado de los dispositivos. Aunque presenta limitaciones en sus versiones más antiguas, su evolución ha permitido mejorar la seguridad y mantener su relevancia en entornos profesionales.

En el siguiente video, comenzamos con un poco de su teoría y luego a ejercitar con este protocolo **(Obligatorio)**.

- 🌐 **Footprinting (Protocolos inseguros - SNMP parte 1) - Charla 98**
- Aprendiendo Ciberseguridad
https://youtu.be/rq_-3AIC3WQ



2.11 footprinting, protocolos inseguros – SNMP parte 2

- 🌐 **Footprinting (Protocolos inseguros - SNMP parte 2) - Charla 99 -**
Aprendiendo Ciberseguridad
<https://youtu.be/7j5WLXBhMjA>

