Uso de herramientas para la auditoría de sistemas

Esta sección explora las diversas herramientas y utilidades que se emplean en auditorías de sistemas de información. Desde analizadores de red y escáneres de vulnerabilidades hasta herramientas avanzadas para la detección de ataques, cada herramienta es presentada con sus características y aplicabilidad en el contexto de la seguridad informática. El objetivo es optimizar la capacidad de detectar y gestionar posibles brechas y amenazas en los sistemas evaluados.

1. Herramientas del sistema operativo tipo Ping, Traceroute, etc.

En el mundo de la auditoría de seguridad informática, es esencial dominar las herramientas básicas que ofrecen los sistemas operativos para diagnosticar y analizar redes. Entre estas herramientas destacan Ping y Traceroute, que nos permiten comprobar la conectividad y analizar rutas de comunicación entre dispositivos.

Ping

El comando Ping es una herramienta sencilla pero poderosa para verificar si un dispositivo en la red está accesible. Se utiliza para verificar la conectividad entre el dispositivo del usuario y otro dispositivo o servidor en la red. Envía paquetes ICMP Echo Request y espera respuestas ICMP Echo Reply. Imaginemos que estamos en una oficina en Madrid y queremos comprobar la conexión con un servidor en Barcelona. Al utilizar ping seguido de la dirección IP o nombre de dominio, como ping servidorbarcelona.ejemplo.es, enviamos paquetes ICMP al destino y esperamos respuestas.

Si recibimos respuestas con tiempos de respuesta bajos, sabemos que la comunicación es estable. Por ejemplo, una respuesta de 20 ms indica una conexión rápida entre ambas ciudades. Sin embargo, si no hay respuesta o los tiempos son muy altos, podríamos estar ante problemas de red que requieren atención.

Posibles resultados al usar ping:

Respuesta exitosa de todos los paquetes:

```
$ ping www.ejemplo.com
PING www.ejemplo.com (93.184.216.34): 56 data bytes
64 bytes from 93.184.216.34: icmp_seq=0 ttl=56 time=10.123 ms
64 bytes from 93.184.216.34: icmp_seq=1 ttl=56 time=10.456 ms
64 bytes from 93.184.216.34: icmp_seq=2 ttl=56 time=10.789 ms
64 bytes from 93.184.216.34: icmp_seq=3 ttl=56 time=11.012 ms

--- www.ejemplo.com ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 10.123/10.595/11.012 ms
```

- Resultado: Todos los paquetes enviados reciben una respuesta del destino.
- Interpretación: Existe conectividad entre el origen y el destino. El tiempo de respuesta (latencia) es un indicador de la rapidez de la conexión.

Perdida de algunos paquetes (Packet Loss):

```
$ ping www.ejemplo.com
PING www.ejemplo.com (93.184.216.34): 56 data bytes
64 bytes from 93.184.216.34: icmp_seq=0 ttl=56 time=10.123 ms
Request timeout for icmp_seq 1
64 bytes from 93.184.216.34: icmp_seq=2 ttl=56 time=10.789 ms
Request timeout for icmp_seq 3

--- www.ejemplo.com ping statistics ---
4 packets transmitted, 2 packets received, 50% packet loss
round-trip min/avg/max = 10.123/10.456/10.789 ms
```

- Resultado: Solo algunos paquetes reciben respuesta, mientras que otros se pierden.
- Interpretación: Puede indicar problemas intermitentes en la red, congestión, o errores en los dispositivos intermedios. Es necesario investigar posibles fallos en la conexión o interferencias.

Tiempo de respuesta elevado:

```
$ ping www.servidor-remoto.com
PING www.servidor-remoto.com (198.51.100.25): 56 data bytes
64 bytes from 198.51.100.25: icmp_seq=0 ttl=45 time=250.456 ms
64 bytes from 198.51.100.25: icmp_seq=1 ttl=45 time=252.789 ms
64 bytes from 198.51.100.25: icmp_seq=2 ttl=45 time=249.123 ms
64 bytes from 198.51.100.25: icmp_seq=3 ttl=45 time=251.012 ms

--- www.servidor-remoto.com ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 249.123/250.845/252.789 ms
```

- Resultado: Los tiempos de respuesta (ms) son más altos de lo normal.
- Interpretación: Indica latencia alta, posiblemente debido a congestión de red, rutas ineficientes o problemas en el ancho de banda disponible.

No hay respuesta (Request timed out):

```
$ ping www.sitio-inaccesible.com
PING www.sitio-inaccesible.com (203.0.113.50): 56 data bytes
Request timeout for icmp_seq 0
Request timeout for icmp_seq 1
Request timeout for icmp_seq 2
Request timeout for icmp_seq 3

--- www.sitio-inaccesible.com ping statistics ---
4 packets transmitted, 0 packets received, 100% packet loss
```

- Resultado: Ninguno de los paquetes enviados recibe respuesta.
- Interpretación: El destino no es alcanzable. Puede deberse a que el dispositivo está apagado, la dirección IP es incorrecta, hay problemas en la ruta de red, o el tráfico ICMP está siendo bloqueado por un firewall.

Mensaje "Destination Host Unreachable":

```
$ ping 192.0.2.1
PING 192.0.2.1 (192.0.2.1): 56 data bytes
From 192.0.2.254 icmp_seq=1 Destination Host Unreachable
From 192.0.2.254 icmp_seq=2 Destination Host Unreachable
From 192.0.2.254 icmp_seq=3 Destination Host Unreachable
From 192.0.2.254 icmp_seq=4 Destination Host Unreachable

--- 192.0.2.1 ping statistics ---
4 packets transmitted, 0 packets received, +4 errors, 100% packet loss
```

- Resultado: El dispositivo intermedio (como un router) indica que no puede alcanzar el host de destino.
- Interpretación: No hay ruta hacia el destino desde el dispositivo que genera el mensaje. Puede ser un problema de configuración de red o fallos en dispositivos intermedios.

Mensaje "Unknown Host":

```
$ ping www.dominio-inexistente.xyz
ping: cannot resolve www.dominio-inexistente.xyz: Unknown host
```

- Resultado: El nombre de dominio o dirección IP no puede ser resuelto.
- Interpretación: Puede haber un error al escribir el nombre de dominio, problemas con el servidor DNS, o el dominio no existe.

Mensaje "General Failure":

```
C:\>ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes of data:
General failure.
General failure.
General failure.
General failure.

Ping statistics for 10.0.0.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

- Resultado: El ping falla sin enviar paquetes.
- Interpretación: Problemas locales en el dispositivo del usuario, como configuración de red incorrecta, adaptador de red deshabilitado, o políticas de seguridad que bloquean el ping.

Variabilidad en los tiempos de respuesta:

```
$ ping www.red-inestable.com
PING www.red-inestable.com (203.0.113.75): 56 data bytes
64 bytes from 203.0.113.75: icmp_seq=0 ttl=50 time=20.123 ms
64 bytes from 203.0.113.75: icmp_seq=1 ttl=50 time=150.456 ms
64 bytes from 203.0.113.75: icmp_seq=2 ttl=50 time=30.789 ms
64 bytes from 203.0.113.75: icmp_seq=3 ttl=50 time=200.012 ms

--- www.red-inestable.com ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 20.123/100.345/200.012 ms
```

- Resultado: Los tiempos de respuesta fluctúan significativamente entre paquetes.
- Interpretación: Puede indicar inestabilidad en la red, fluctuaciones en el ancho de banda, o congestión temporal.

¿Cómo resolver y actuar según los resultados de ping?

⇒ Verificar la dirección IP o nombre de dominio: Asegurarse de que se está utilizando la dirección correcta.
⇒ Comprobar la conectividad local: Probar ping a una dirección IP conocida (como la puerta de enlace predeterminada) para descartar problemas locales.
⇒ Revisar configuraciones de firewall: Verificar si hay políticas que bloquean el tráfico ICMP.
⇒ Contactar al administrador de red o ISP: Si el problema persiste, puede ser necesario escalar el asunto para una solución más profunda.

Ejemplo

Imaginemos que estamos en una empresa en Madrid y queremos comprobar la conectividad con el sitio web del Ministerio de Educación, Formación Profesional y Deportes en España, cuyo dominio es www.educacionfpydeportes.gob.es. Queremos asegurarnos de que nuestra red puede acceder a este recurso gubernamental sin problemas.

Abrimos la terminal o símbolo del sistema en nuestro ordenador. En Windows, podemos hacerlo buscando "Símbolo del sistema" en el menú de inicio; en macOS o Linux, buscamos "Terminal".

Escribimos el comando Ping seguido del dominio que queremos verificar. En este caso; ping www.educacionfpydeportes.gob.es.

Presionamos Enter y observamos la salida:

```
Haciendo ping a www.agenciatributaria.es [195.76.52.41] con 32 bytes de datos:
Respuesta desde 195.76.52.41: bytes=32 tiempo=30ms TTL=53
Respuesta desde 195.76.52.41: bytes=32 tiempo=29ms TTL=53
Respuesta desde 195.76.52.41: bytes=32 tiempo=31ms TTL=53
Respuesta desde 195.76.52.41: bytes=32 tiempo=30ms TTL=53

Estadísticas de ping para 195.76.52.41:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0 (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 29ms, Máximo = 31ms, Media = 30ms
```

Análisis de la salida:

- Dirección IP: El dominio se ha resuelto a la dirección IP 195.76.52.41 .
- Tiempo de respuesta: Los tiempos de 29-31 ms indican una conexión estable y rápida.
- Paquetes perdidos: 0% de pérdida indica que no hay problemas de conectividad.

Si hubiéramos obtenido "Solicitud agotada" o pérdida del 100% de los paquetes, podríamos sospechar de problemas en nuestra red o que el sitio web está inaccesible. Pero en este caso, la comunicación es satisfactoria.

Traceroute

El comando traceroute (o tracert en Windows) se utiliza para identificar la ruta que toman los paquetes desde el dispositivo del usuario hasta el destino. Muestra cada salto (hop) en el camino y el tiempo que tarda en alcanzarlo. Siguiendo el ejemplo anterior, si notamos lentitud en la conexión con el servidor de Barcelona, podemos ejecutar traceroute servidorbarcelona.ejemplo.es. Esto nos revelará cada salto intermedio que realizan los paquetes y los tiempos asociados.

Supongamos que observamos que en el salto número 5, que corresponde a un router en Zaragoza, hay una demora significativa. Esto nos indica que puede haber un problema en ese punto específico de la red, permitiéndonos informar al equipo correspondiente para su solución.

Posibles resultados al usar traceroute:

Ruta completa hasta el destino:

```
$ traceroute www.ejemplo.com
traceroute to www.ejemplo.com (93.184.216.34), 30 hops max, 60 byte packets
 1  192.168.1.1 (192.168.1.1)  1.123 ms  0.987 ms  0.912 ms
 2  10.0.0.1 (10.0.0.1)  5.456 ms  5.789 ms  6.012 ms
 3  203.0.113.1 (203.0.113.1)  10.123 ms  10.456 ms  10.789 ms
 4  198.51.100.1 (198.51.100.1)  15.123 ms  15.456 ms  15.789 ms
 5  93.184.216.34 (93.184.216.34)  20.123 ms  20.456 ms  20.789 ms
```

- Resultado: Muestra todos los saltos intermedios hasta llegar al destino final.
- Interpretación: Permite visualizar la ruta y los dispositivos por los que pasa el tráfico. Es útil para identificar dónde pueden estar ocurriendo retrasos o problemas.

Asteriscos en algunos saltos:

```
$ traceroute www.sitio-con-firewall.com
traceroute to www.sitio-con-firewall.com (203.0.113.100), 30 hops max, 60 byte packets
 1  192.168.1.1 (192.168.1.1)  1.123 ms  0.987 ms  0.912 ms
 2  10.0.0.1 (10.0.0.1)  5.456 ms  5.789 ms  6.012 ms
 3  * * *
 4  * * *
 5  203.0.113.100 (203.0.113.100)  25.123 ms  25.456 ms  25.789 ms
```

- Resultado: En lugar de tiempos de respuesta, aparecen asteriscos (*) en ciertos saltos.
- Interpretación: El dispositivo en ese salto no responde a las solicitudes ICMP Time Exceeded o está configurado para no revelar información. No necesariamente indica un problema, pero limita la visibilidad de la ruta completa.

Traceroute se detiene antes de llegar al destino:

```
$ traceroute www.sitio-inaccesible.com
traceroute to www.sitio-inaccesible.com (203.0.113.50), 30 hops max, 60 byte packets
 1  192.168.1.1 (192.168.1.1)  1.123 ms  0.987 ms  0.912 ms
 2  10.0.0.1 (10.0.0.1)  5.456 ms  5.789 ms  6.012 ms
 3  203.0.113.1 (203.0.113.1)  10.123 ms  10.456 ms  10.789 ms
 4  * * *
 5  * * *
 6  * * *
 (continúa hasta que se alcanza el máximo de saltos)
```

- Resultado: El comando no muestra más saltos después de cierto punto y no alcanza el destino.
- Interpretación: Puede haber un corte en la ruta, dispositivos que bloquean el tráfico, o el destino no es alcanzable. Es necesario investigar dónde ocurre la interrupción.

Saltos con tiempos de respuesta elevados:

```
$ traceroute www.servidor-remoto.com
traceroute to www.servidor-remoto.com (198.51.100.25), 30 hops max, 60 byte packets
 1  192.168.1.1 (192.168.1.1)  1.123 ms  0.987 ms  0.912 ms
 2  10.0.0.1 (10.0.0.1)  5.456 ms  5.789 ms  6.012 ms
 3  203.0.113.1 (203.0.113.1)  10.123 ms  10.456 ms  10.789 ms
 4  198.51.100.1 (198.51.100.1)  150.123 ms  151.456 ms  152.789 ms
 5  198.51.100.25 (198.51.100.25)  155.123 ms  156.456 ms  157.789 ms
```

- Resultado: Algunos saltos muestran tiempos de respuesta mucho más altos que otros.
- Interpretación: Indica posibles cuellos de botella o congestión en esos puntos de la red. Es útil para identificar dónde se producen retrasos.

Mensajes de error como "Request timed out" o "Destination Unreachable":

```
C:\>tracert www.sitio-con-problemas.com

Traza a la dirección www.sitio-con-problemas.com [203.0.113.80]
sobre un máximo de 30 saltos:

  1     1 ms     1 ms     1 ms  192.168.1.1
  2     5 ms     5 ms     6 ms  10.0.0.1
  3     *        *        *     Tiempo de espera agotado para esta solicitud.
  4     *        *        *     Tiempo de espera agotado para esta solicitud.
  5     *        *        *     Tiempo de espera agotado para esta solicitud.
```

- Resultado: El traceroute muestra estos mensajes en uno o varios saltos.
- Interpretación: Similar a los asteriscos, puede indicar dispositivos que no responden o problemas de conectividad en esos puntos.

Variaciones en las rutas en ejecuciones sucesivas:

```
$ traceroute www.sitio-dinamico.com
traceroute to www.sitio-dinamico.com (198.51.100.200), 30 hops max, 60 byte packets
 1  192.168.1.1 (192.168.1.1)  1.123 ms  0.987 ms  0.912 ms
 2  10.0.0.1 (10.0.0.1)  5.456 ms  5.789 ms  6.012 ms
 3  203.0.113.5 (203.0.113.5)  15.123 ms  15.456 ms  15.789 ms
 4  198.51.100.200 (198.51.100.200)  20.123 ms  20.456 ms  20.789 ms

$ traceroute www.sitio-dinamico.com
traceroute to www.sitio-dinamico.com (198.51.100.200), 30 hops max, 60 byte packets
 1  192.168.1.1 (192.168.1.1)  1.123 ms  0.987 ms  0.912 ms
 2  10.0.0.1 (10.0.0.1)  5.456 ms  5.789 ms  6.012 ms
 3  203.0.113.6 (203.0.113.6)  25.123 ms  25.456 ms  25.789 ms
 4  198.51.100.200 (198.51.100.200)  30.123 ms  30.456 ms  30.789 ms
```

- Resultado: Al ejecutar traceroute varias veces, la ruta cambia.
- Interpretación: Las rutas en Internet pueden ser dinámicas. Los cambios pueden deberse a balanceo de carga, cambios en la red, o rutas alternativas por fallos en ciertos segmentos.

¿Cómo resolver y actuar según los resultados de traceroute?

⇒ Identificar el punto de falla: Si traceroute se detiene en un salto específico, se puede enfocar el diagnóstico en ese segmento de la red.
⇒ Comprobar políticas de seguridad: Algunos dispositivos están configurados para no responder a traceroute. Esto puede ser normal y no necesariamente un problema.
⇒ Comunicar a los administradores de red: Si se detectan altos tiempos de respuesta o interrupciones, informar a los responsables de la red para investigar posibles problemas.
⇒ Repetir pruebas en diferentes momentos: Las condiciones de la red pueden variar, por lo que es útil realizar varias pruebas.

Ejemplo

Supongamos ahora que experimentamos lentitud al acceder al mismo sitio web del ejemplo anterior y queremos averiguar dónde podría estar el problema en la ruta que siguen los paquetes desde nuestra ubicación hasta el servidor.

Ejecutamos el comando Traceroute. En Windows, el comando es tracert; en macOS y Linux, es traceroute.

En Windows:

- tracert www.educacionfpydeportes.gob.es

En macOS/Linux:

- traceroute www.educacionfpydeportes.gob.es

Observamos la salida (ejemplo en Windows):

```
Traza a la dirección www.agenciatributaria.es [195.76.52.41]
sobre un máximo de 30 saltos:

  1     1 ms     1 ms     1 ms  192.168.0.1
  2     6 ms     5 ms     5 ms  85.57.0.1
  3     7 ms     7 ms     6 ms  213.99.77.1
  4    16 ms    15 ms    16 ms  84.16.14.50
  5    22 ms    23 ms    22 ms  62.115.113.150
  6   150 ms   152 ms   149 ms  62.115.113.155
  7    29 ms    28 ms    30 ms  195.76.52.41

Traza completa.
```

Análisis de la salida:

- Saltos: Cada línea representa un salto (hop) entre routers desde nuestra ubicación hasta el destino.
- Tiempos: Los tiempos en milisegundos indican la latencia en cada salto.
- Posible problema: Notamos que en el salto 6 los tiempos aumentan drásticamente a alrededor de 150 ms, comparado con los 20 ms anteriores.

Interpretación:

- Salto 1: Es nuestro router local (192.168.0.1).
- Saltos 2-5: Routers intermedios de nuestro proveedor de Internet y carriers nacionales.
- Salto 6: Se observa un aumento significativo en la latencia, lo que sugiere un posible problema en este nodo específico (62.115.113.155).
- Salto 7: El servidor destino en la Agencia Tributaria (195.76.52.41).

Posible problema identificado:

- El incremento de latencia en el salto 6 puede indicar congestión o problemas técnicos en ese router. Esto podría estar causando la lentitud al acceder al sitio web.

Nota

Algunos entornos corporativos o redes pueden restringir el uso de ping y traceroute por razones de seguridad. Ping utiliza paquetes ICMP, mientras que traceroute puede utilizar ICMP, UDP o TCP, dependiendo del sistema operativo y las opciones usadas. Además, existen versiones mejoradas de estas herramientas (como MTR - My Traceroute) que combinan funcionalidades y ofrecen análisis más detallados.

2. Herramientas de análisis de red, puertos y servicios, incluyendo Nmap, Netcat, y herramientas actualizadas como Masscan (actualización para añadir Masscan como una herramienta moderna).

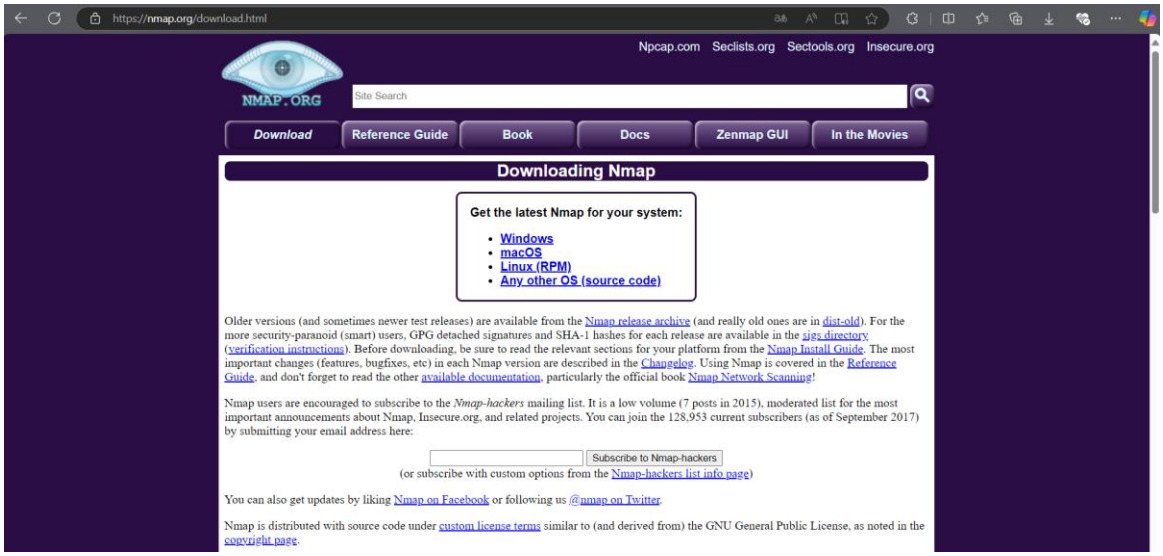
Para una auditoría más profunda, es necesario utilizar herramientas avanzadas que nos ayuden a identificar posibles vulnerabilidades en la red. Entre ellas se encuentran Nmap, Netcat y la moderna Masscan.

Nmap

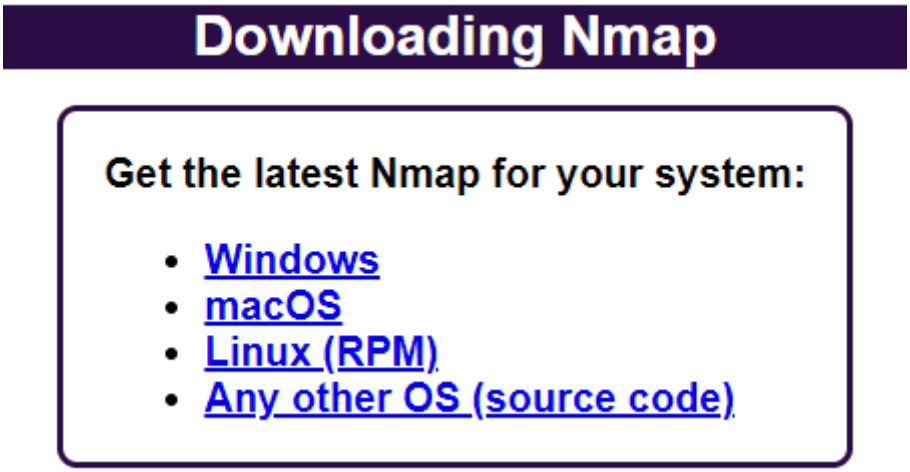
- Nmap es una herramienta de código abierto que permite escanear puertos y descubrir servicios activos en dispositivos de red. Por ejemplo, si trabajamos en una empresa en Valencia y queremos conocer los servicios que ofrece un servidor interno, podemos ejecutar nmap -sV 10.0.0.5.
- Este comando nos mostrará los puertos abiertos y las versiones de los servicios que se ejecutan en ellos. Si encontramos que el puerto 22 (SSH) está abierto y ejecutando una versión antigua del software, sabremos que es necesario actualizarlo para evitar posibles ataques.
- Además, Nmap permite realizar escaneos más avanzados, como detectar sistemas operativos o realizar scripts de detección de vulnerabilidades conocidas, lo que lo convierte en una herramienta indispensable para los auditores.

Para descargar Nmap, sigue estos pasos:

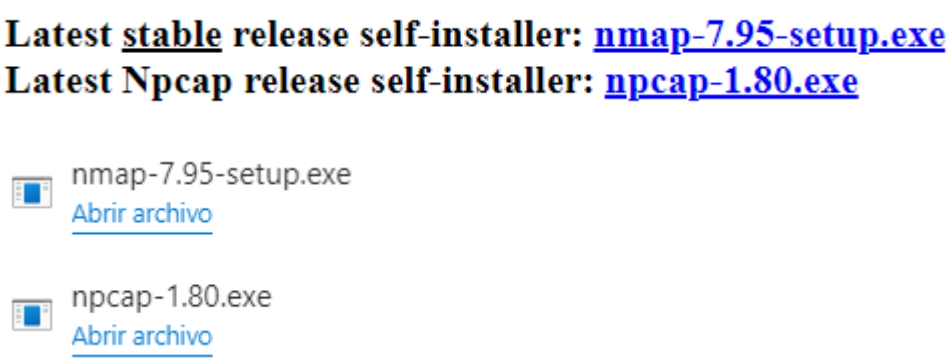
1. Visita la página oficial de descargas de Nmap:



2. Selecciona la versión adecuada para tu sistema operativo (Windows, macOS o Linux):

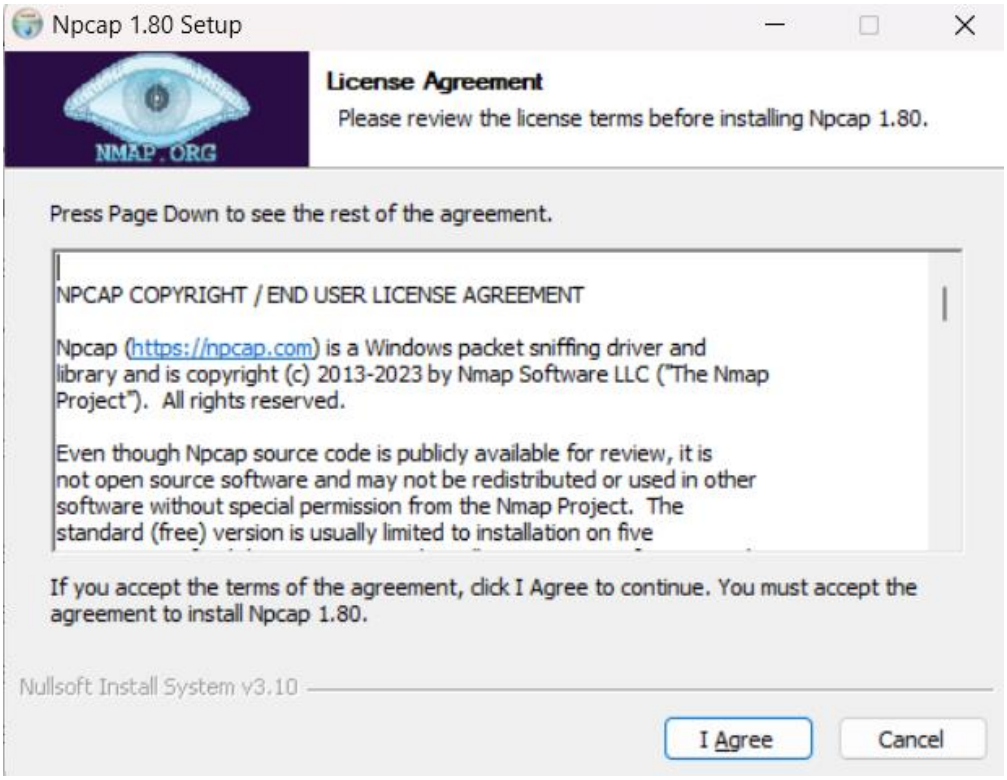


3. Sigue las instrucciones de instalación de acuerdo con tu sistema:
   - En Windows: descarga ambos archivos:

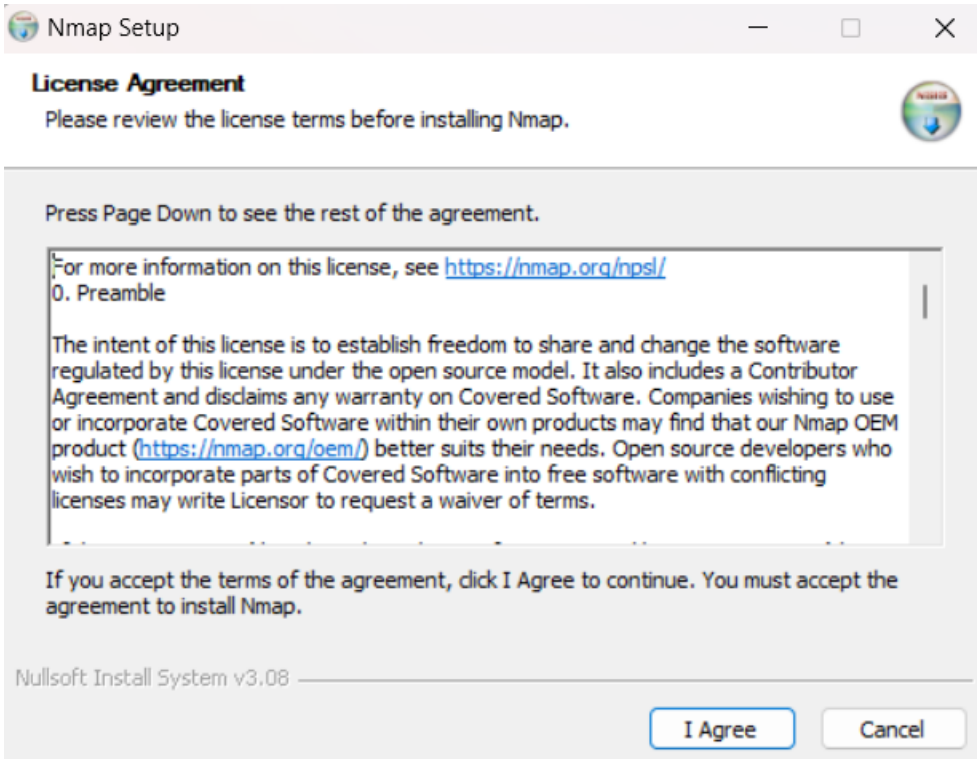


     - nmap-7.95-setup.exe: Este es el instalador principal de Nmap.
     - npcap-1.80.exe: Este es un complemento necesario para capturar tráfico en redes locales (Npcap reemplaza a WinPcap).

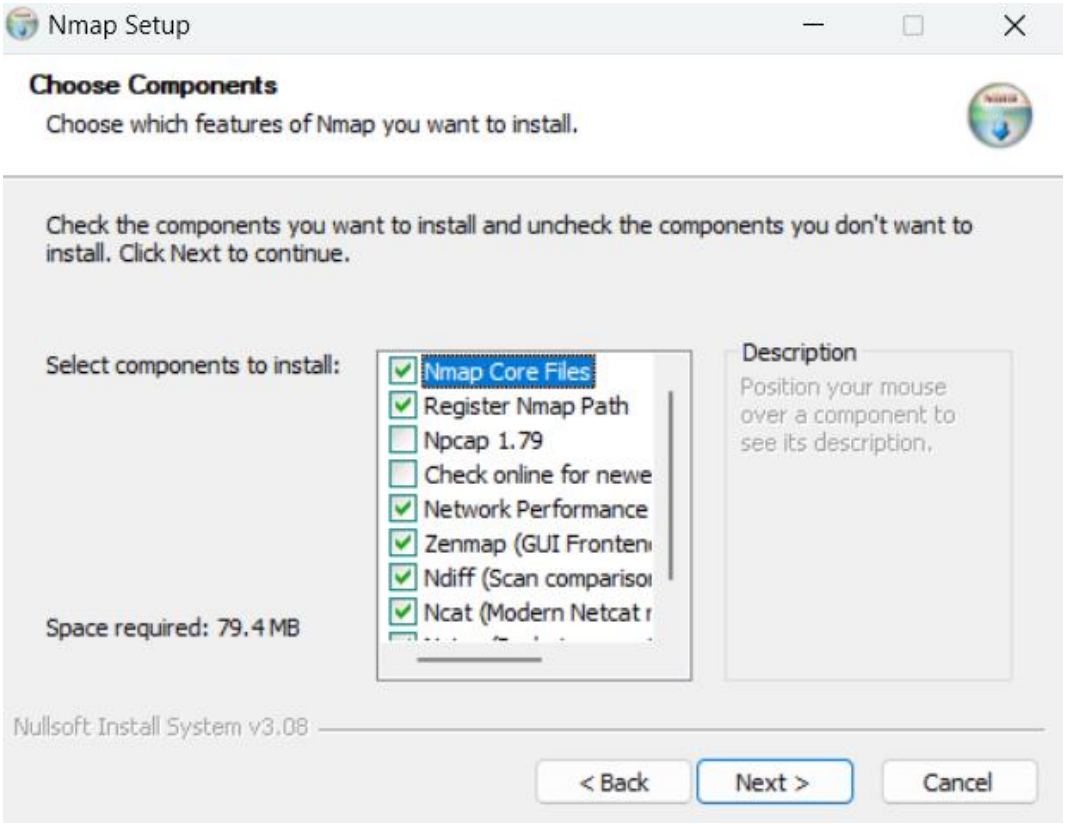
- Primero, ejecuta el instalador de Npcap y sigue las instrucciones de instalación:



- Luego, ejecuta el instalador de Nmap para completar la configuración:



- Aquí están las opciones recomendadas al instalar Nmap:



Nmap Core Files (Obligatorio): Esta es la instalación principal de Nmap.

Register Nmap Path: Permite ejecutar Nmap desde cualquier lugar en la línea de comandos.

Zenmap (GUI Frontend): Interfaz gráfica útil para principiantes.

Ncat (Modern Netcat): Herramienta para enviar y recibir datos de red, ideal para pruebas avanzadas.

Ndiff: Permite comparar resultados de escaneos anteriores.

- Opcional:
  Network Performance: Si deseas probar la velocidad de la red.

  Deja Npcap desmarcado si ya lo instalaste por separado.

- En Linux: generalmente puedes instalarlo con el comando sudo apt-get install nmap (Debian/Ubuntu) o sudo yum install nmap (RedHat/CentOS).
- En macOS: puedes usar Homebrew con brew install nmap.

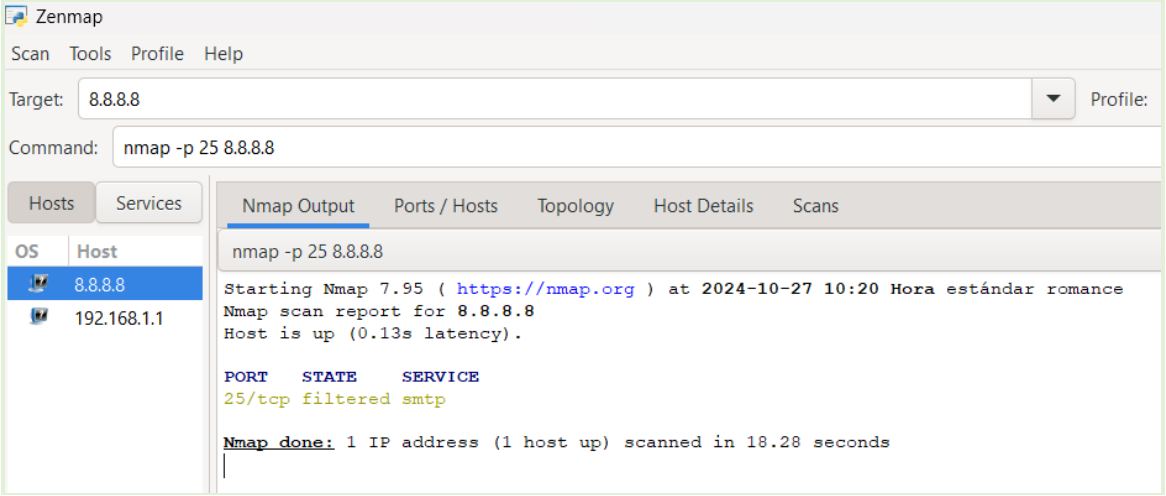
Ejemplo

Para monitorear puertos específicos con Nmap y detectar actividades inusuales:

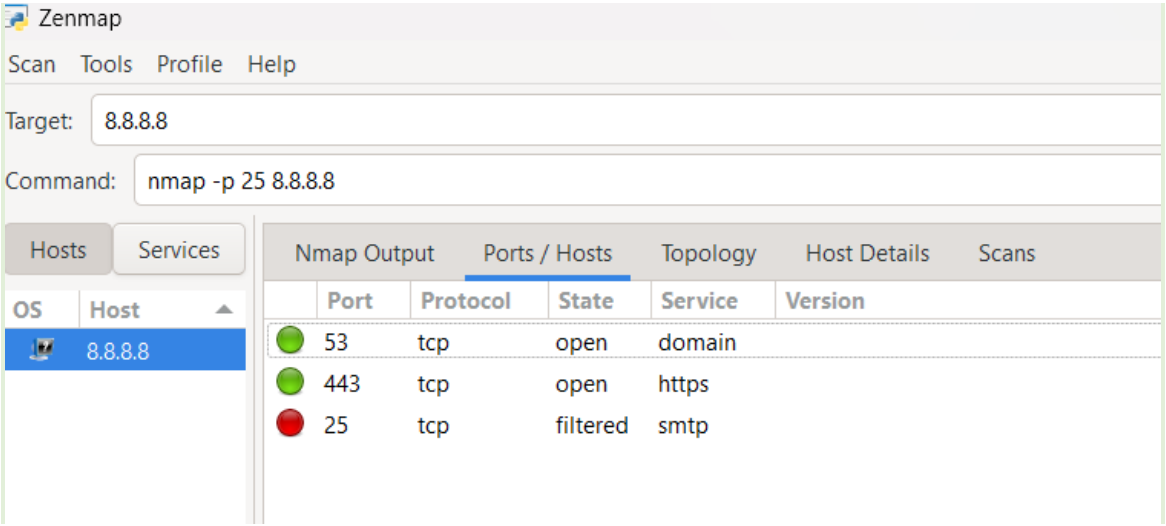
Escaneo básico de puertos específicos:

Un ejemplo de dirección IP para hacer el escaneo sería una IP de red local, como 192.168.1.1, o una IP pública, como 8.8.8.8 (que es una IP pública de Google). Para este ejemplo emplearemos esta última.

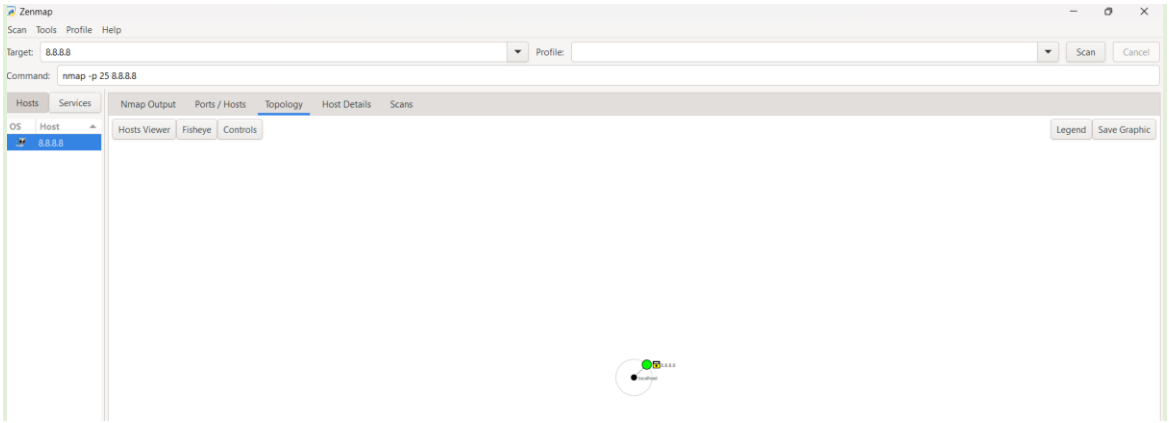
Comando: nmap -p 25 8.8.8.8



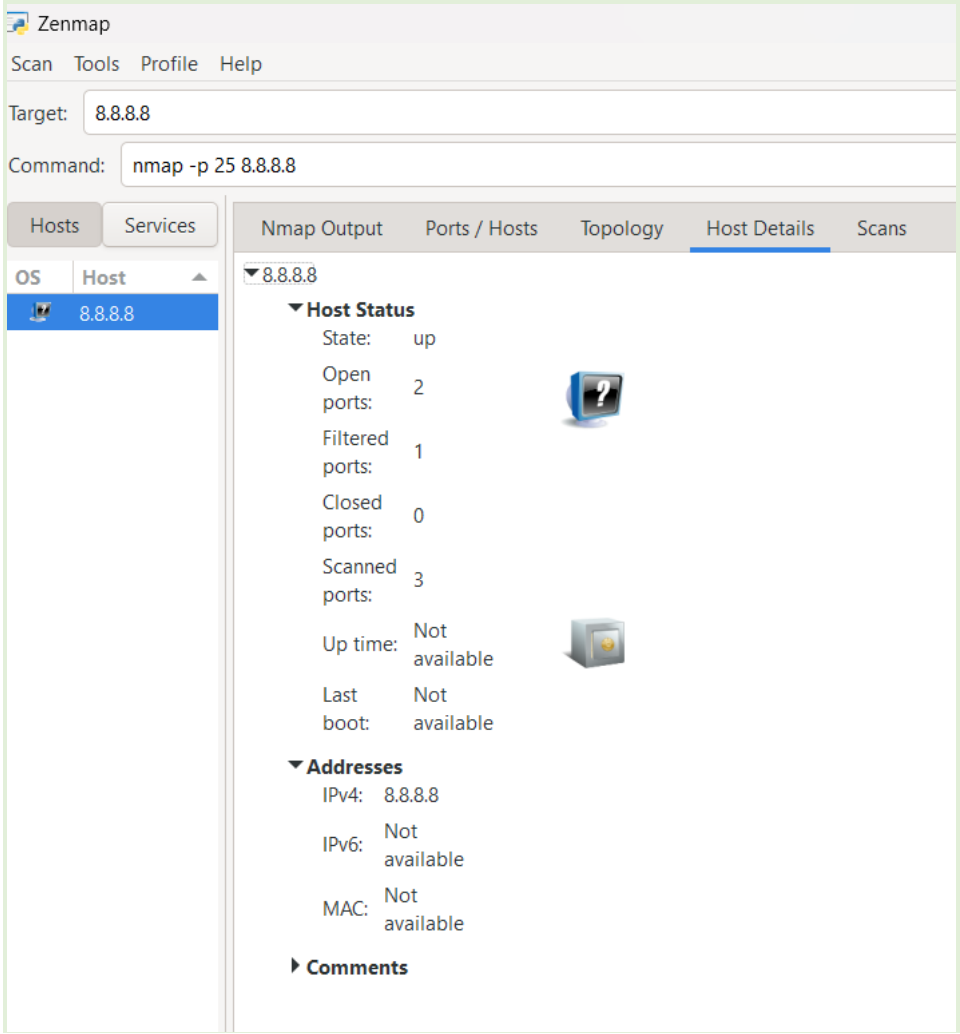
→ Ports and States: Muestra los puertos escaneados, como 25 (SMTP), marcado como filtered, lo que indica que el acceso podría estar bloqueado.



→ Puertos open como 53/tcp (domain) y 443/tcp (https) indican servicios accesibles.



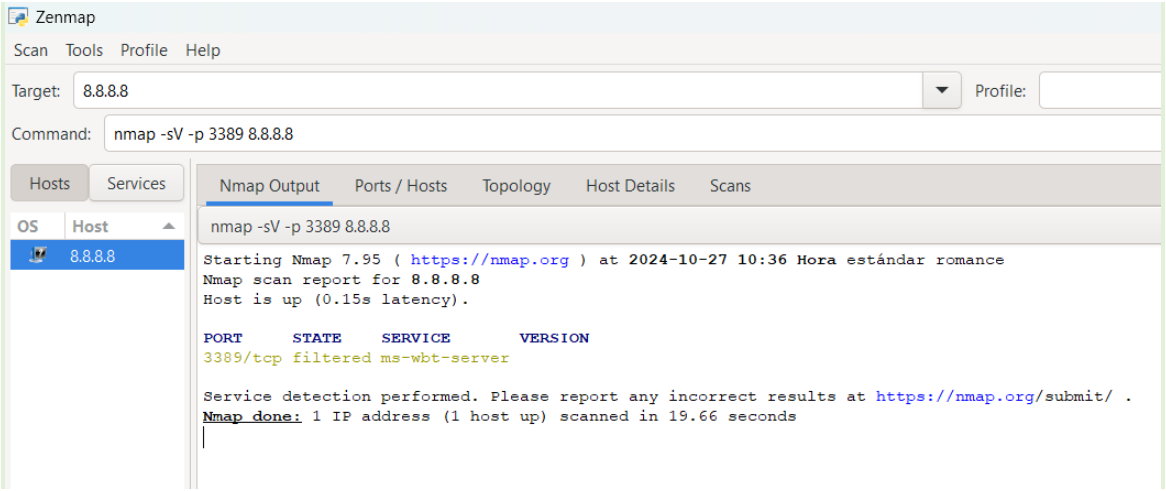
→ Topology: Visualiza la ruta de red o trazado desde tu dispositivo hacia la IP de destino, mostrando cada salto y conexión.



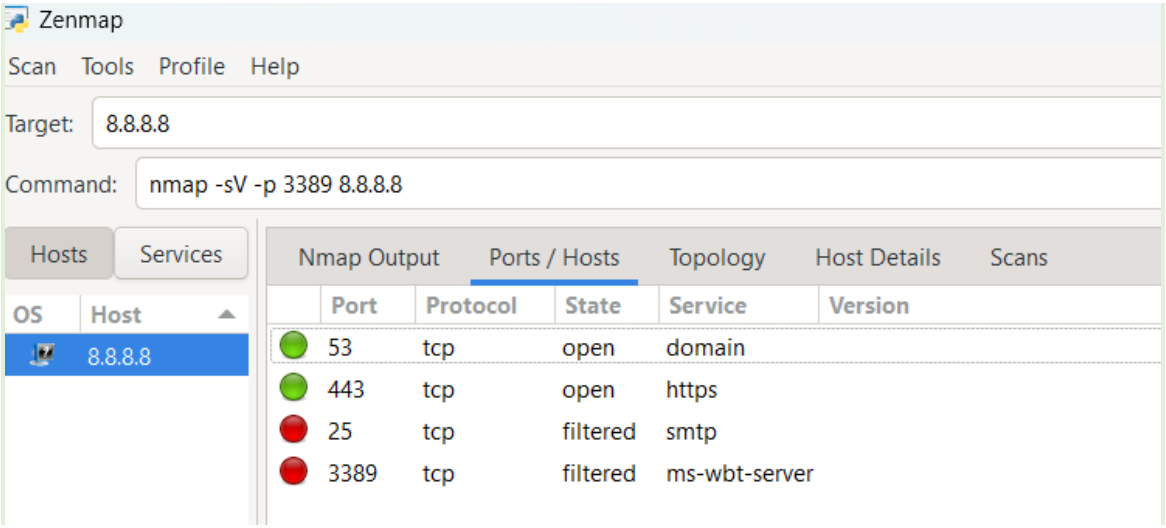
→ Host Details: Resume el estado del objetivo, incluyendo puertos abiertos, cerrados o filtrados, y detalles escaneados como dirección IP y estado del host.

Escaneo detallado de servicios:

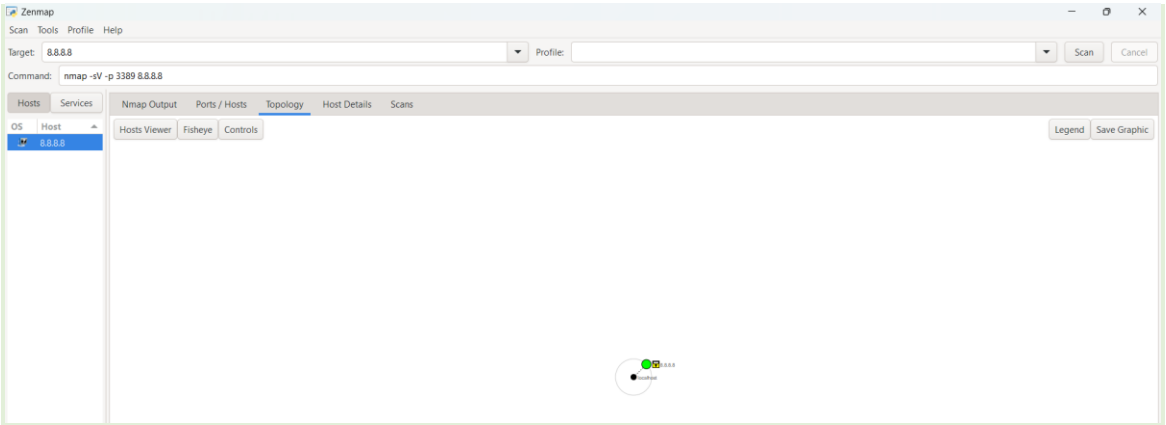
Comando: nmap -sV -p 3389 8.8.8.8 (escaneo detallado en el puerto 3389 en la IP 8.8.8.8)



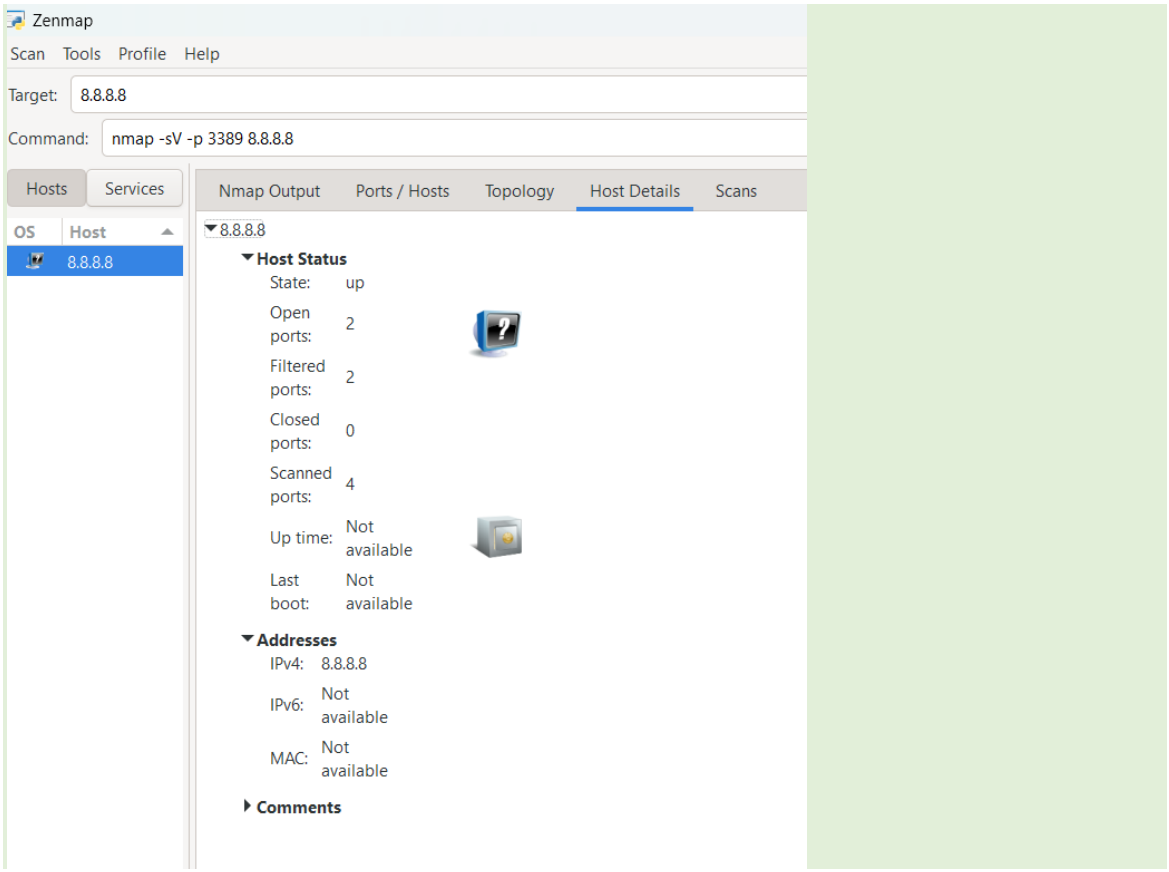
→ Nmap Output: El puerto 3389/tcp (usado para ms-wbt-server o Remote Desktop Protocol) aparece como filtered. Esto indica que los paquetes enviados al puerto están siendo bloqueados, posiblemente por un firewall o reglas de red, por lo que no se puede determinar si está abierto o cerrado.



→ Ports/Hosts: Hay otros puertos abiertos (como 53/tcp para domain y 443/tcp para https) y filtrados (25/tcp y 3389/tcp), sugiriendo que el host responde a algunos servicios, mientras que otros están restringidos.



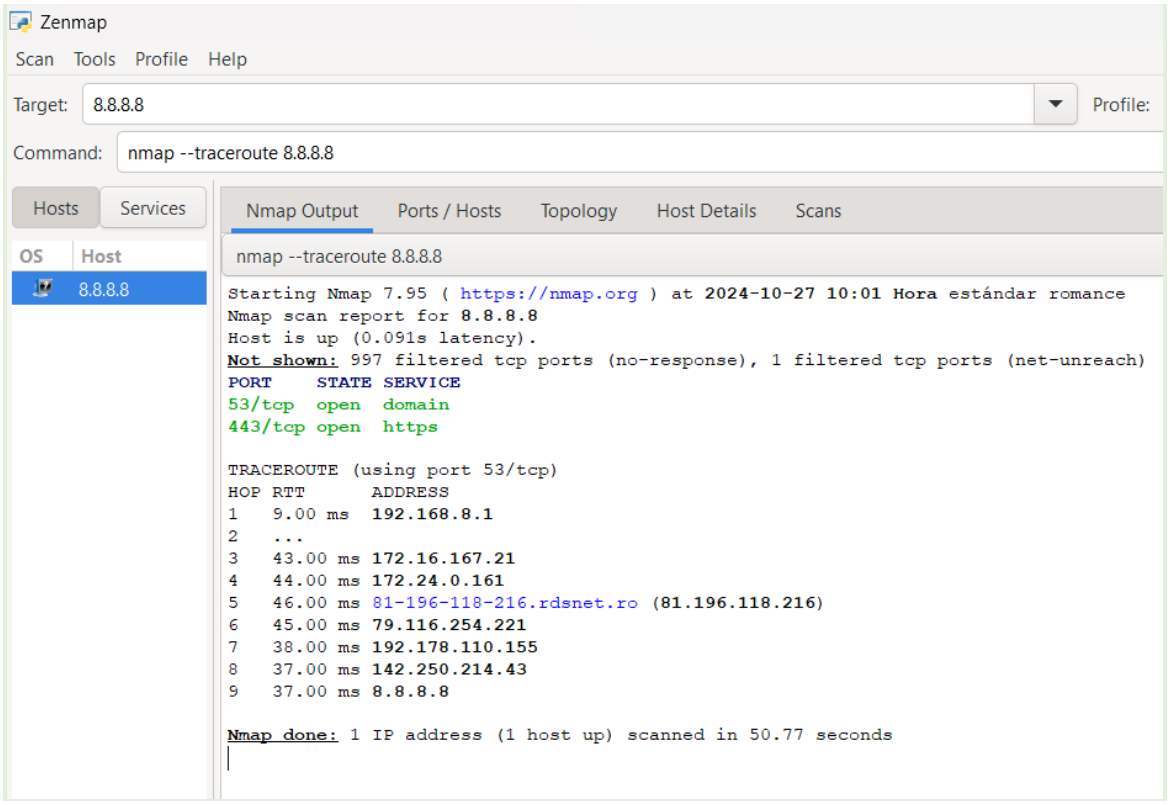
→ Topology: Muestra la ruta de la conexión hacia la IP 8.8.8.8, permitiéndote visualizar los saltos en la red.



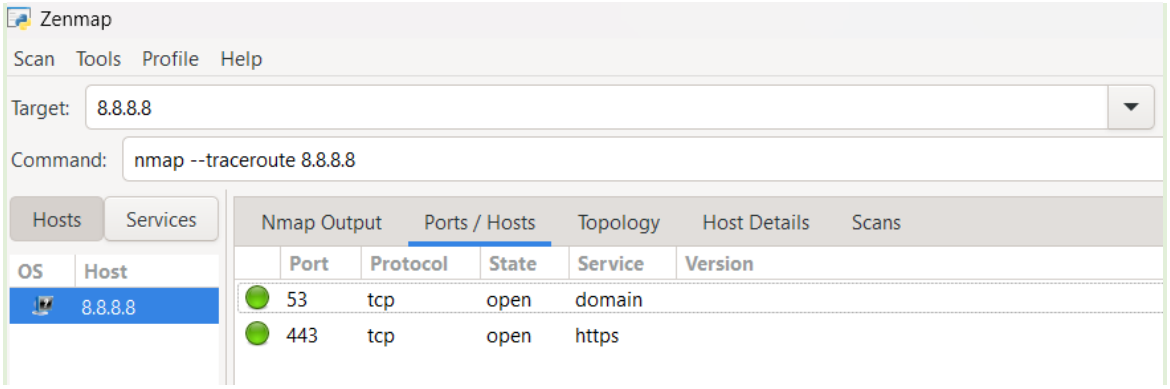
→ Host Details: Indica el estado general del host y los puertos abiertos/filtrados, lo cual ayuda en la evaluación de la configuración de seguridad y accesibilidad del sistema.

Monitoreo de rutas:

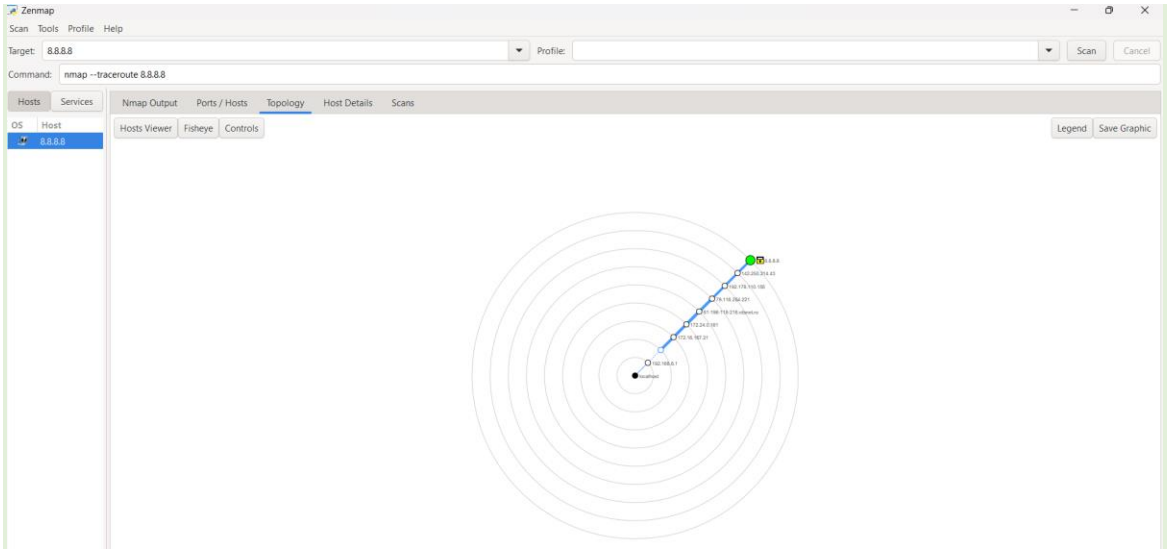
Comando: nmap --traceroute 8.8.8.8 (para ver la ruta de conexión hasta 8.8.8.8):



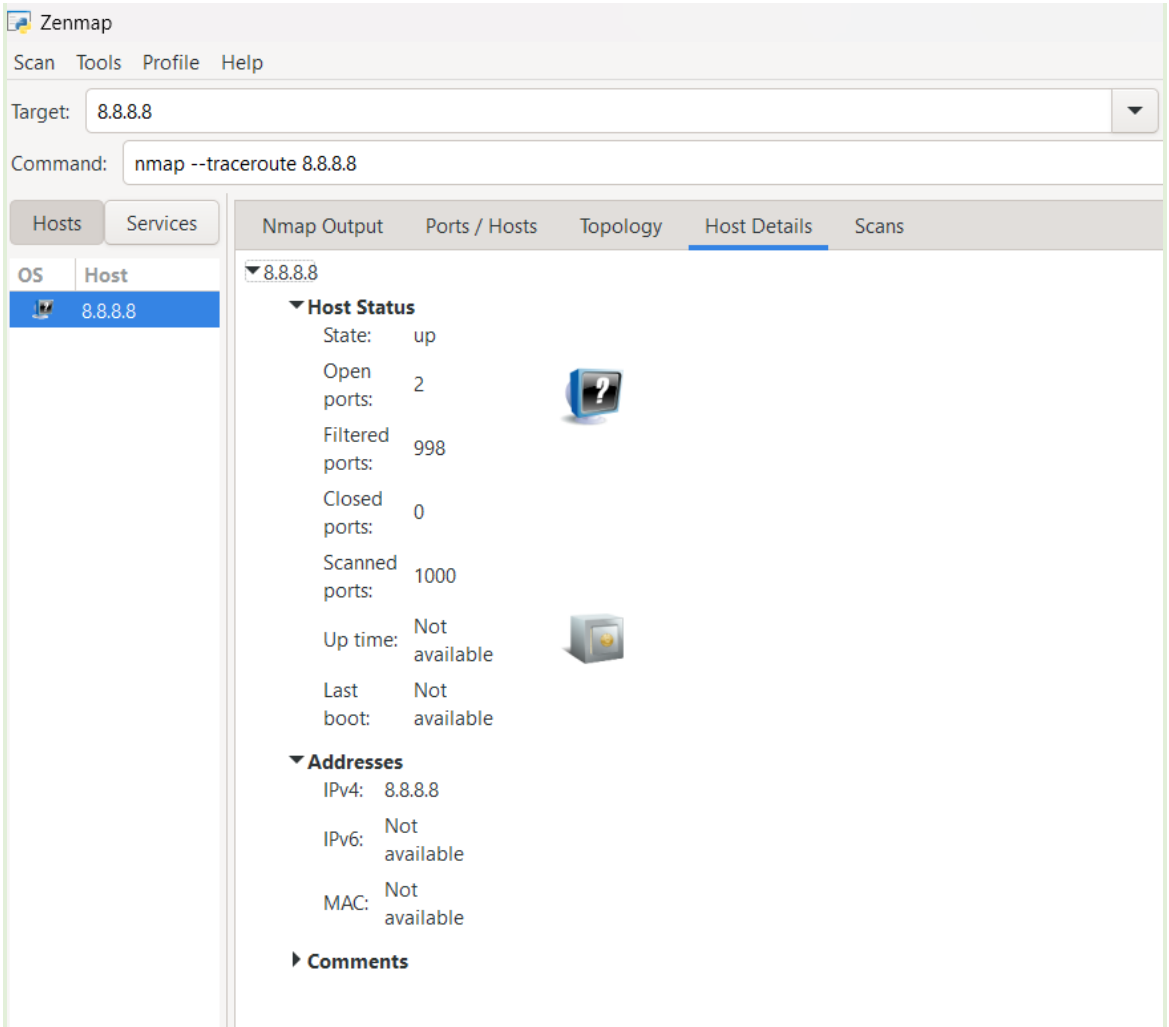
→ Puertos abiertos: Se muestra que los puertos 53 (DNS) y 443 (HTTPS) están abiertos en la IP 8.8.8.8, que es un servidor de Google.
→ Traceroute: Muestra la ruta que siguen los paquetes para llegar desde tu red hasta el servidor de destino.
→ Hops: Cada línea numerada representa un "salto" o router intermedio por el cual pasan los datos.
→ RTT: Mide el tiempo en milisegundos que tarda en llegar al destino.
→ IP Address: Indica la IP de cada router intermedio.



→ Ports / Hosts: Muestra que los puertos 53 (DNS) y 443 (HTTPS) están abiertos en la IP 8.8.8.8. Esto indica que el servidor responde en esos puertos y está configurado para esos servicios.



→ Topology: Visualiza la ruta de red o "hops" que el tráfico toma para llegar al objetivo, en este caso, 8.8.8.8. Cada círculo representa un "salto" hacia el destino, lo que ayuda a comprender el camino y posibles puntos de vulnerabilidad.



→ Host Details: Proporciona información detallada del host, como el estado (en línea), la cantidad de puertos abiertos (2), y la dirección IPv4 (8.8.8.8). No muestra datos sobre el tiempo de actividad o la dirección MAC porque el host está fuera de la red local.

Importante

Cuando usas Nmap en ciberseguridad, estás analizando la "exposición" de un dispositivo o servidor a través de los puertos que tiene abiertos o filtrados. Cada puerto representa un servicio (como envío de correos, acceso remoto, o páginas web). Por ejemplo, el puerto 25 se usa para correos, y si está abierto sin restricciones, un atacante podría intentar usarlo para enviar spam o acceder al sistema.

Analizar tu propia IP o una IP pública te ayuda a entender cómo está protegida. Por ejemplo:

Escaneo de puertos: Con el comando nmap -p 25 [IP], revisas si un puerto específico (como el de correo) está abierto. Si lo está, podrías mejorar su seguridad (cambiando configuraciones, aplicando filtros, etc.).

Detección de servicios: Al usar nmap -sV -p 3389 [IP], puedes identificar qué servicio está activo en un puerto específico, por ejemplo, el puerto 3389 para conexiones remotas. Si ves algo inesperado, como un servicio vulnerable, sabrás que necesitas tomar medidas.

Monitoreo de rutas: Con nmap --traceroute [IP], verificas cómo llegan los datos hasta el destino. Esto muestra todos los "saltos" o pasos intermedios por los que pasa la información, lo que permite ver si hay puntos vulnerables o rutas no usuales.

Actividad 6

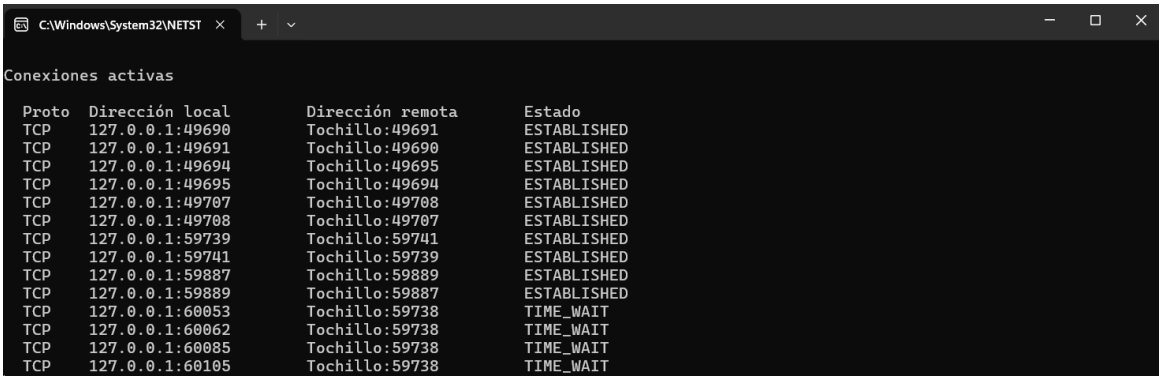
Utiliza Nmap para identificar puertos abiertos en una red:

Descarga e instala Nmap en tu sistema.
Realiza un escaneo básico de una red pública.
Identifica los puertos abiertos.

Netcat

- Conocida como la "navaja suiza" de las redes, Netcat nos permite establecer conexiones TCP o UDP para leer y escribir datos directamente. Supongamos que queremos comprobar si el puerto 80 (HTTP) de un servidor en Sevilla está abierto y aceptando conexiones. Podemos utilizar nc -vz 10.0.0.10 80, y Netcat nos informará si el puerto está abierto o cerrado.
- Además, Netcat es útil para transferir archivos entre máquinas o incluso crear servidores y clientes simples para pruebas. Por ejemplo, podemos enviar un archivo desde un equipo a otro dentro de la misma red sin necesidad de configurar servicios adicionales.

- Es una utilidad básica que viene integrada en la mayoría de los sistemas operativos. Netstat permite ver todas las conexiones activas y los puertos que están en uso en un sistema, lo que facilita la identificación de servicios abiertos:



Masscan

- En escenarios donde necesitamos escanear grandes segmentos de red en poco tiempo, Masscan es la herramienta adecuada. Diseñada para ser extremadamente rápida, Masscan puede escanear todas las direcciones IPv4 en cuestión de minutos.
- Imaginemos que en una auditoría para una organización gubernamental en España necesitamos identificar todos los puertos abiertos en un rango de IPs amplio. Con Masscan, podríamos ejecutar masscan -p80,443 192.168.0.0/16 --rate=100000, ajustando la velocidad según las necesidades y limitaciones de la red.

Importante

Es importante tener en cuenta que, debido a su potencia, Masscan debe utilizarse con precaución y siempre con autorización, ya que un escaneo a gran escala puede interpretarse como una actividad maliciosa si no se realiza en entornos controlados.

En el contexto actual, donde las infraestructuras de red son cada vez más complejas y extensas, es esencial mantenerse actualizado con las herramientas modernas. Aunque Nmap y Netcat siguen siendo relevantes, herramientas como Masscan se vuelven indispensables para auditorías a gran escala, especialmente en sectores como telecomunicaciones y organismos gubernamentales.

Actividad 7

Después de leer atentamente el artículo proporcionado sobre Masscan, realiza las siguientes actividades:

¿Qué es Masscan y cuál es su principal objetivo en el ámbito de la seguridad informática?

¿En qué se diferencia Masscan de Nmap en términos de funcionamiento y enfoque al realizar escaneos de puertos?

Enumera y describe al menos tres funcionalidades destacadas de Masscan mencionadas en el artículo.

¿Cómo permite Masscan la personalización de paquetes durante el escaneo y por qué es útil esta característica?

Describe un escenario práctico en el que Masscan sería la herramienta preferida para realizar un escaneo de red.

¿Cuáles son las consideraciones éticas y de seguridad que se deben tener en cuenta al utilizar herramientas como Masscan?

Enlace al artículo: https://keepcoding.io/blog/que-es-masscan-y-como-funciona/

3. Herramientas de análisis de vulnerabilidades tipo Nessus.

Las amenazas cibernéticas evolucionan constantemente, por este motivo, es fundamental que las organizaciones identifiquen y corrijan las vulnerabilidades de sus sistemas antes de que puedan ser explotadas. Para ello, herramientas como Nessus se han convertido en aliados indispensables en el ámbito de la auditoría de seguridad informática.

¿Qué es Nessus?

Nessus es una herramienta profesional desarrollada por Tenable Network Security que permite realizar análisis exhaustivos de vulnerabilidades en sistemas y redes. Su objetivo es detectar debilidades que puedan ser aprovechadas por atacantes, facilitando así la implementación de medidas preventivas.

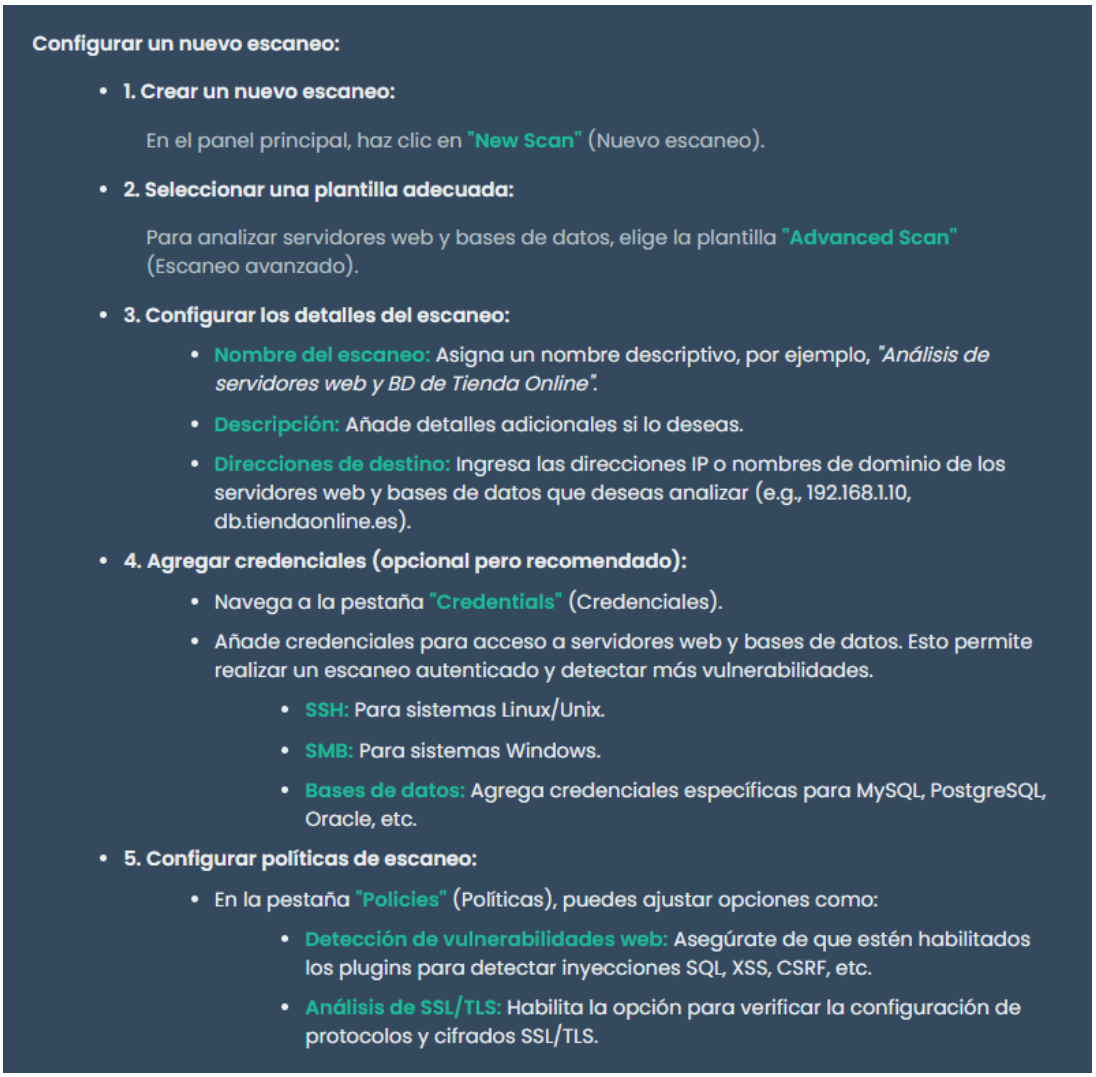
Pie de imagen: Nessus ofrece una prueba gratuita de 7 días.

Principales características de Nessus:

- Amplia cobertura de vulnerabilidades: Nessus cuenta con una base de datos actualizada que incluye más de 100.000 vulnerabilidades conocidas, lo que permite detectar desde fallos en sistemas operativos hasta configuraciones erróneas en aplicaciones.
- Escaneos personalizados: Permite configurar los análisis según las necesidades específicas de la organización, ya sea enfocándose en ciertos rangos de IP, puertos o tipos de vulnerabilidades.
- Informes detallados: Genera reportes que clasifican las vulnerabilidades por nivel de riesgo, ofreciendo recomendaciones claras para su mitigación.
- Facilidad de uso: Posee una interfaz intuitiva que facilita su uso incluso para aquellos que no son expertos en seguridad informática.

Imaginemos una empresa en Madrid dedicada al comercio electrónico. Dada la naturaleza de su negocio, es vital proteger los datos de sus clientes y garantizar la integridad de sus transacciones. Utilizando Nessus, el equipo de seguridad puede:

⇒ Realizar un escaneo completo de sus servidores web y bases de datos para identificar vulnerabilidades como inyecciones SQL o configuraciones inseguras de SSL/TLS:



Ejecutar el escaneo:

- **1. Guardar y lanzar el escaneo:**

  Haz clic en "Save" (Guardar) para almacenar la configuración. Luego selecciona el escaneo en la lista y haz clic en "Launch" (Iniciar).

- **2. Monitorear el progreso:**

  Puedes ver el estado del escaneo en tiempo real. Nessus mostrará el porcentaje completado y el tiempo estimado restante.

Analizar los resultados:

- **1. Acceder al reporte:**

  Una vez finalizado el escaneo, haz clic en el nombre del escaneo para ver los resultados detallados.

- **2. Revisar vulnerabilidades detectadas:**

  Nessus clasificará las vulnerabilidades por nivel de severidad: Crítico, Alto, Medio, Bajo e Informativo. Haz clic en cada vulnerabilidad para obtener más información.

- **3. Identificar inyecciones SQL y configuraciones inseguras de SSL/TLS:**

  Busca vulnerabilidades relacionadas con inyecciones SQL (e.g., "SQL Injection") y problemas en SSL/TLS (como uso de protocolos obsoletos o certificados inválidos).

Pie de imagen: Guía para escaneo completo en Nessus.

⇒ Detectar sistemas desactualizados que requieren parches de seguridad, evitando así posibles brechas:

Configurar un escaneo de actualización de sistemas:

- **Crear un nuevo escaneo:**

  Haz clic en "New Scan" (Nuevo escaneo) en el panel principal.

- **Seleccionar la plantilla "Basic Network Scan":**

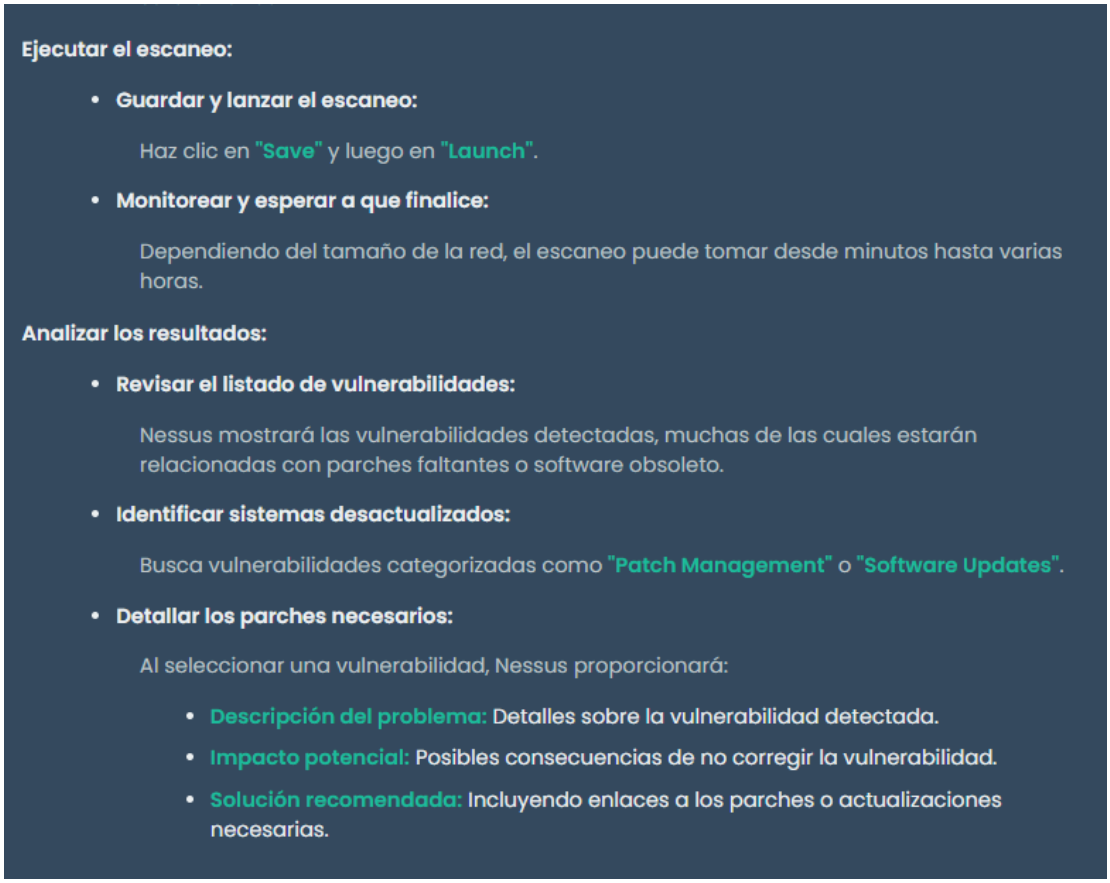
  Esta plantilla está diseñada para detectar vulnerabilidades comunes y sistemas que necesitan actualizaciones.

- **Configurar los detalles del escaneo:**
  - Nombre del escaneo: Por ejemplo, *"Detección de sistemas desactualizados"*.
  - Direcciones de destino: Ingresa el rango completo de direcciones IP de la red corporativa, por ejemplo, 192.168.1.0/24.
- **Agregar credenciales para escaneo autenticado:**

  En la pestaña "Credentials", añade credenciales administrativas para acceder a los sistemas.

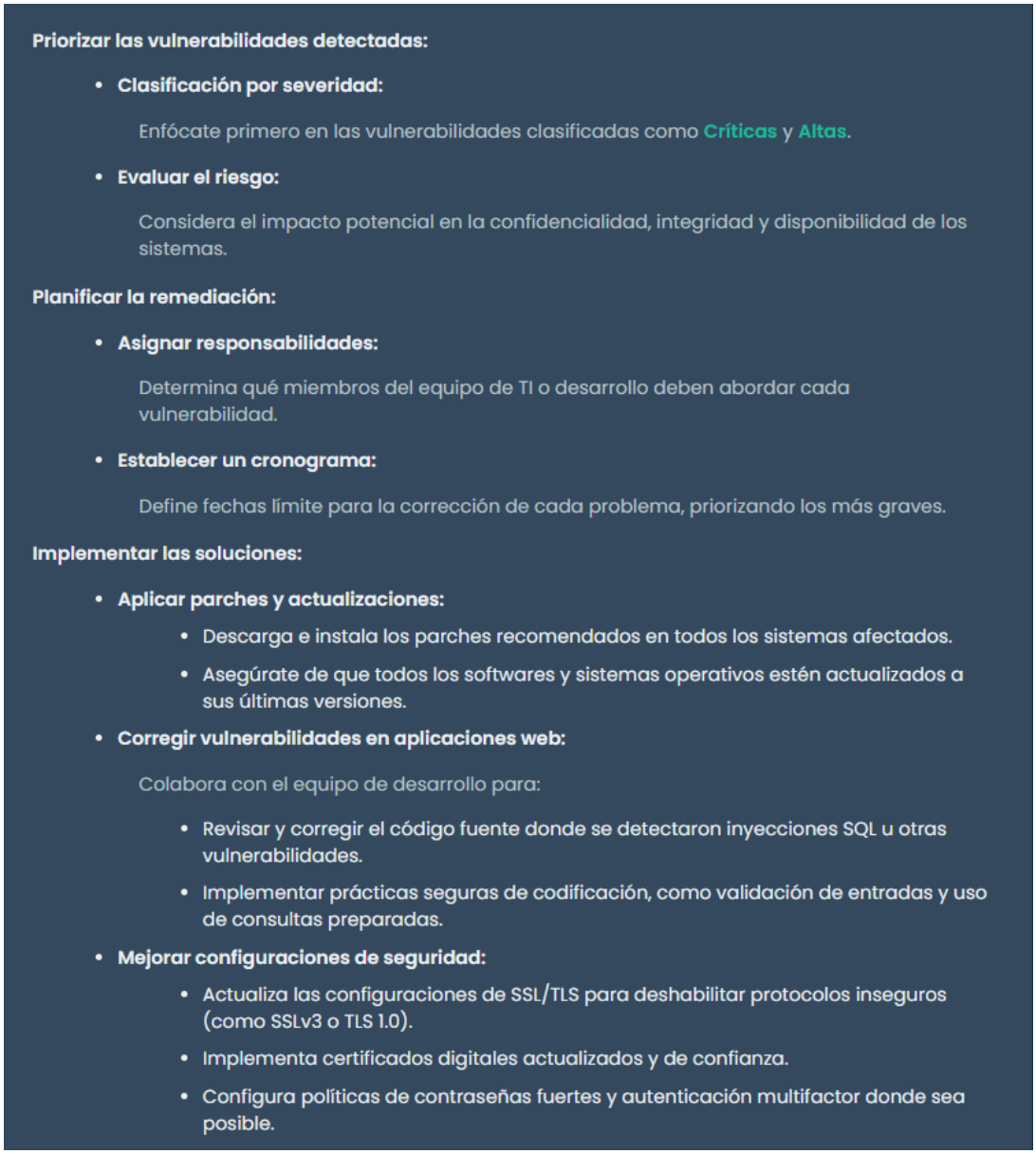
  - Para sistemas Windows: Proporciona el nombre de usuario y contraseña con privilegios de administrador.
  - Para sistemas Linux/Unix: Añade credenciales SSH (puede ser usuario y contraseña o clave privada).
- **Configurar opciones avanzadas:**

  En "Settings", puedes ajustar parámetros como limitar la velocidad del escaneo para no saturar la red.



Pie de imagen: Guía para detectar sistemas desactualizados en Nessus.

⇒ Implementar soluciones proactivas basadas en las recomendaciones proporcionadas por Nessus, fortaleciendo su postura de seguridad:



Pie de imagen: Guía para implementar soluciones en Nessus.

Aunque existen otras herramientas en el mercado, como OpenVAS o Qualys, Nessus destaca por su equilibrio entre funcionalidad y facilidad de uso. En el contexto español, donde las pymes buscan soluciones eficaces sin una gran complejidad, Nessus se presenta como una opción óptima.

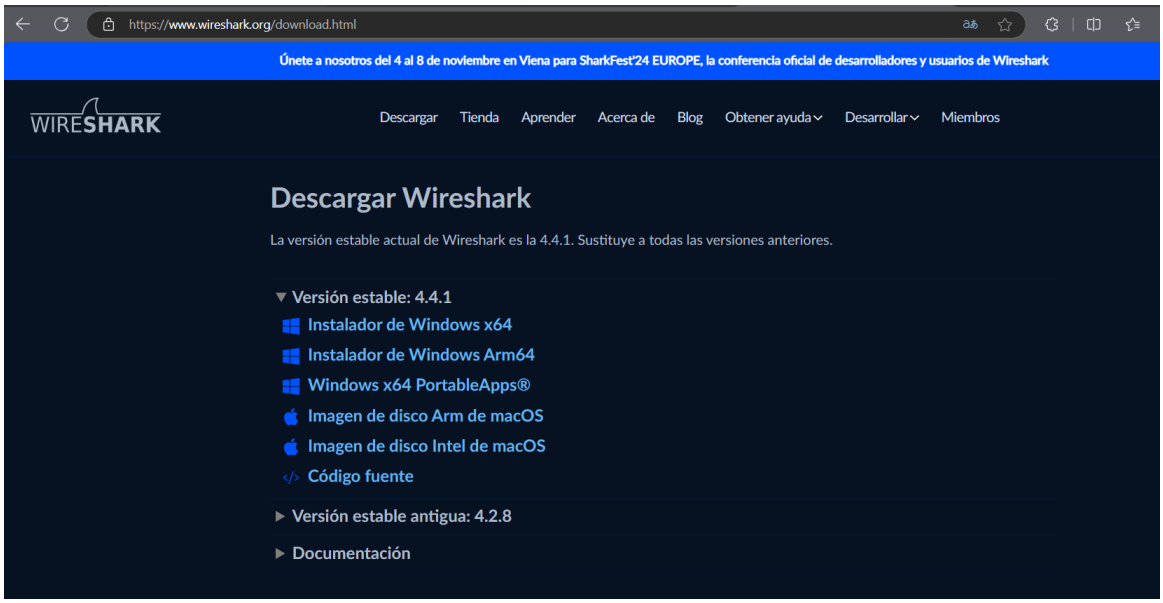
4. Analizadores de protocolos, incluyendo WireShark y alternativas actuales en la nube y análisis remoto (ampliación para reflejar el uso de servicios de análisis en entornos en la nube y SaaS).

El análisis de protocolos es esencial para comprender el comportamiento del tráfico en una red y detectar actividades sospechosas o no autorizadas. Herramientas como Wireshark han sido fundamentales en este ámbito, pero con la evolución tecnológica, también han surgido soluciones en la nube y opciones de análisis remoto.

Wireshark: el estándar en análisis de protocolos

¿Qué es Wireshark?

Wireshark es una herramienta de código abierto que permite capturar y analizar en profundidad el tráfico de red. Es utilizada tanto por profesionales de seguridad como por administradores de sistemas para diagnosticar problemas y detectar anomalías.



Pie de imagen: Sitio web de descarga "https://www.wireshark.org/download.html".

Características destacadas:

- ► Análisis detallado: Descompone los paquetes de datos, mostrando información a nivel de bits.
- ► Amplio soporte de protocolos: Reconoce y analiza cientos de protocolos, desde los más comunes hasta los especializados.
- ► Filtros avanzados: Facilita la búsqueda y focalización en paquetes específicos dentro de grandes volúmenes de datos.

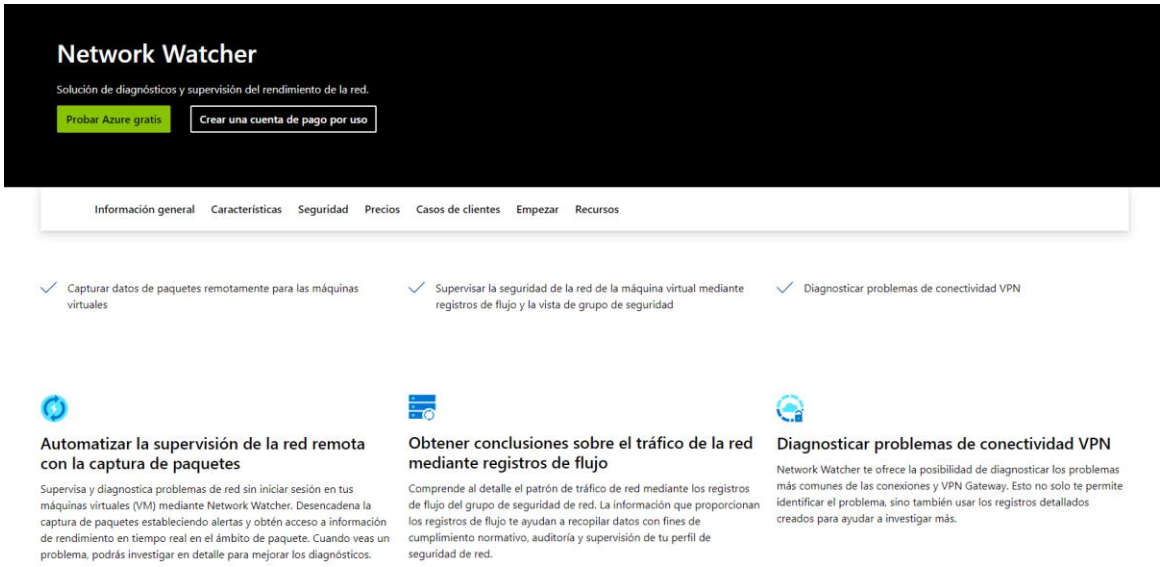
Por ejemplo, imagina que un proveedor de servicios de Internet (ISP) en Barcelona detecta que algunos de sus clientes experimentan latencias elevadas. Utilizando Wireshark, los técnicos pueden:

- ⇒ Capturar el tráfico en puntos clave de la red.
- ⇒ Identificar cuellos de botella o retransmisiones excesivas causadas por configuraciones erróneas.
- ⇒ Optimizar los parámetros de red, mejorando la experiencia de los usuarios.

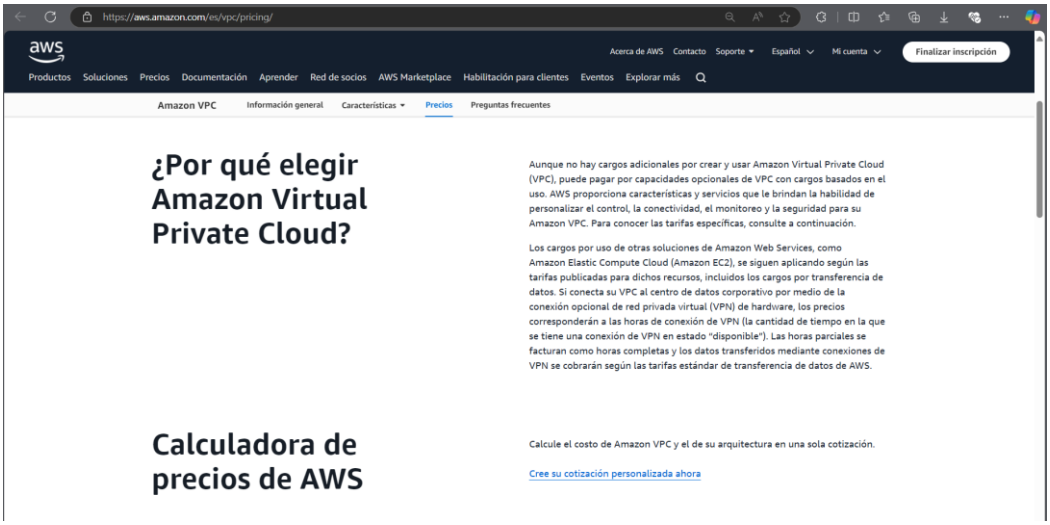
Con la creciente adopción de servicios en la nube y arquitecturas distribuidas, han surgido herramientas que permiten el análisis de protocolos sin necesidad de estar físicamente en el lugar o de instalar software localmente. Algunas de la herramientas destacadas son:

- CloudShark: Es una plataforma que permite cargar capturas de tráfico y analizarlas desde un navegador web, facilitando la colaboración entre equipos remotos:

- Azure Network Watcher: Para entornos de Microsoft Azure, ofrece capacidades de monitoreo y diagnóstico de redes en la nube:



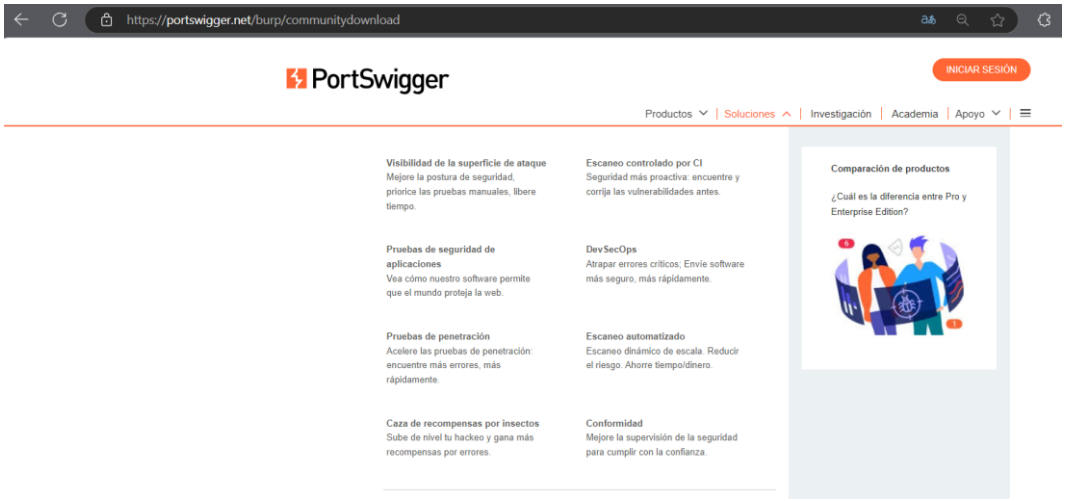
- AWS VPC Traffic Mirroring: En Amazon Web Services, permite replicar el tráfico de red de instancias específicas para su análisis en herramientas externas:



5. Analizadores de páginas web como Burp Suite, OWASP ZAP y Dirb (sustituye herramientas desactualizadas como Parosproxy por Burp Suite y OWASP ZAP, más usadas en auditorías actuales).

Las aplicaciones web son objetivos frecuentes de ataques debido a su exposición pública. Para garantizar su seguridad, es necesario analizarlas en profundidad utilizando herramientas especializadas.

¿Qué es Burp Suite?

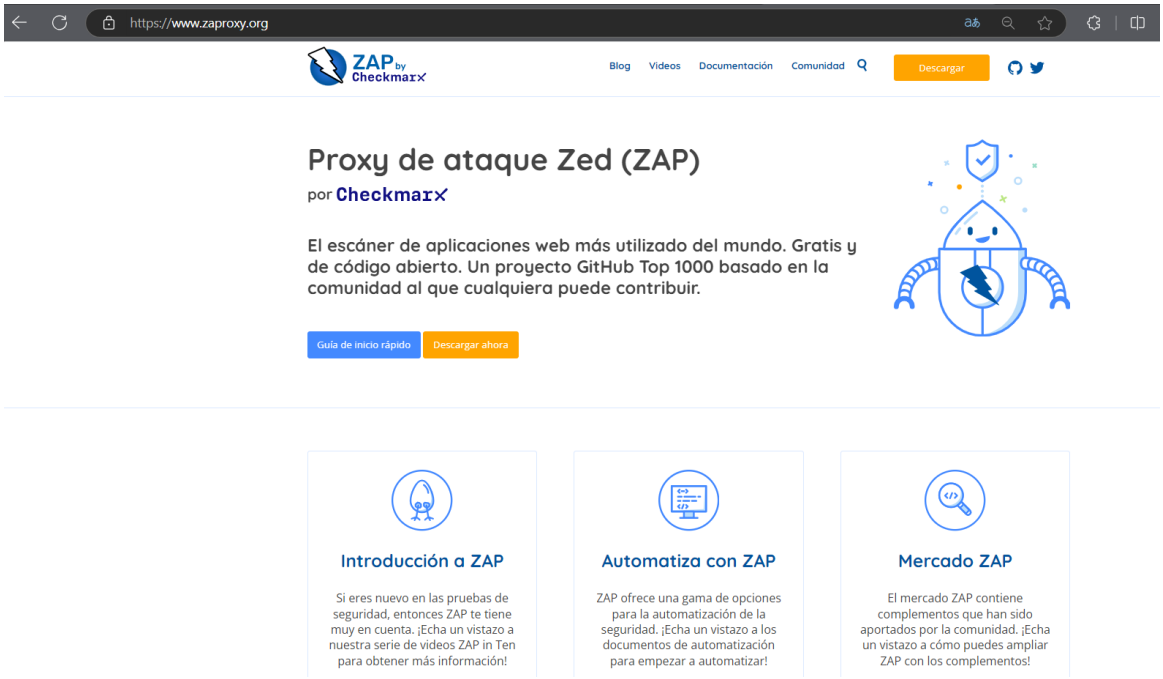


Burp Suite es una plataforma integrada para realizar pruebas de seguridad en aplicaciones web. Desarrollada por PortSwigger, es ampliamente utilizada por pentesters y auditores.

Funciones clave:

- Proxy interceptador: Permite interceptar y modificar las solicitudes y respuestas entre el navegador y el servidor.
- Escáner automático: Detecta vulnerabilidades como XSS, inyecciones SQL, SSRF, entre otras.
- Intruder: Facilita la realización de ataques personalizados, como fuerza bruta o fuzzing.
- Extensiones: Su versión profesional permite integrar plugins que amplían sus capacidades.

¿Qué es OWASP ZAP?



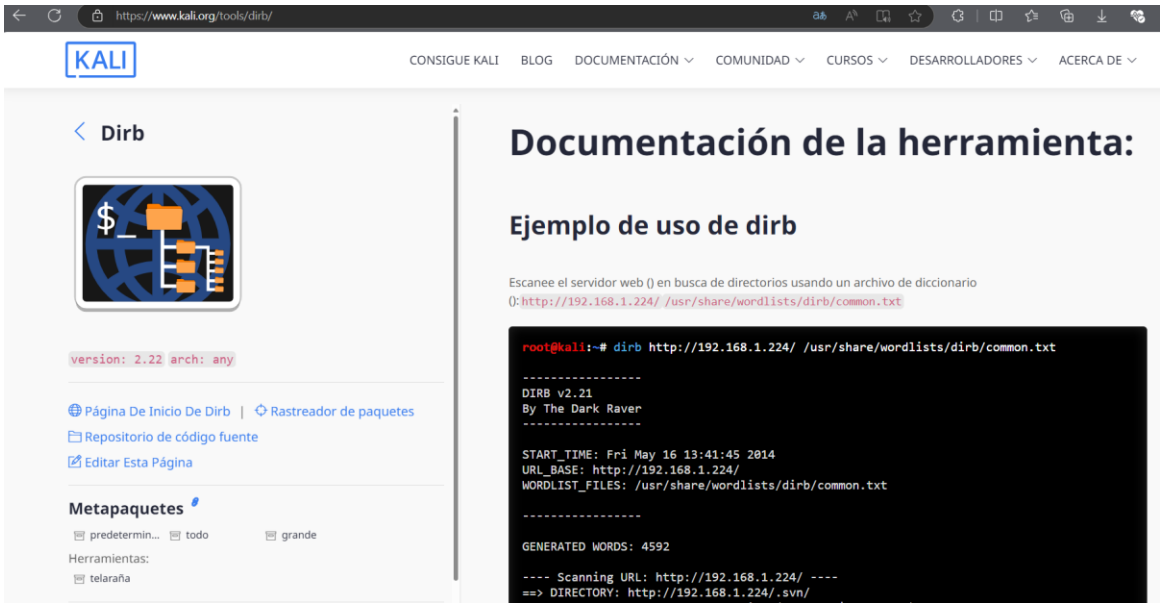
El Zed Attack Proxy (ZAP) es una herramienta gratuita y de código abierto mantenida por la comunidad OWASP. Está diseñada para encontrar vulnerabilidades en aplicaciones web durante las etapas de desarrollo y pruebas.

Pie de imagen: Sitio web de descarga.

Características principales:

- Facilidad de uso: Ideal para desarrolladores y testers que no son expertos en seguridad.
- Modo automatizado: Permite realizar escaneos rápidos sin necesidad de configuraciones complejas.
- Actualizaciones constantes: La comunidad agrega nuevas funcionalidades y mejora la detección de vulnerabilidades.

¿Qué es Dirb?



Dirb es una herramienta de línea de comandos que busca directorios y archivos ocultos en sitios web utilizando listas de palabras. Es útil para descubrir recursos no vinculados o mal configurados.

Por ejemplo, imaginemos una empresa de comercio electrónico en Bilbao que lanza una nueva plataforma web. Antes de su lanzamiento público, el equipo de seguridad:

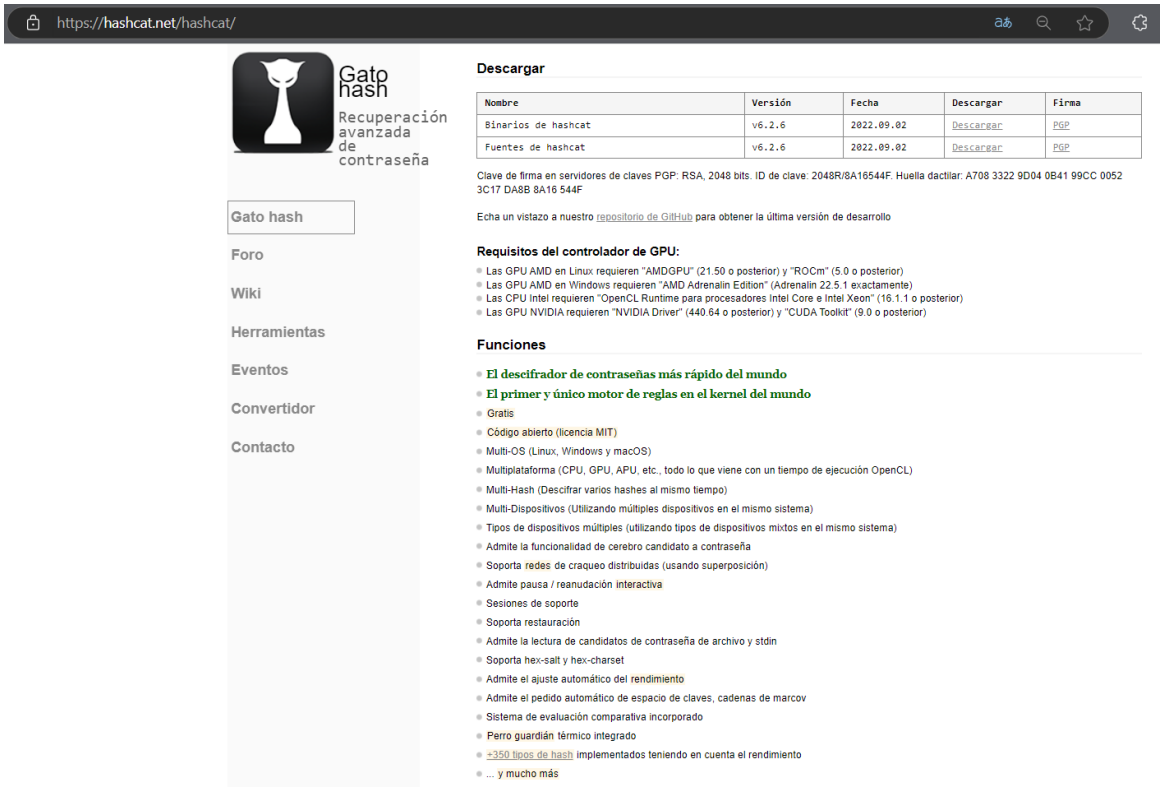
→ Utiliza Burp Suite para interceptar el tráfico y detectar posibles inyecciones SQL en formularios de búsqueda.
→ Emplea OWASP ZAP para un escaneo automático, identificando vulnerabilidades XSS en la sección de comentarios.
→ Aplica Dirb para encontrar directorios administrativos expuestos que no deberían ser accesibles públicamente.

Herramientas como Parosproxy, que fueron populares en el pasado, han sido reemplazadas por soluciones más modernas como Burp Suite y OWASP ZAP. Estas ofrecen mejores capacidades de detección, interfaces más amigables y reciben actualizaciones para enfrentar nuevas amenazas.

6. Herramientas de fuerza bruta y descifrado de contraseñas como Hashcat y John the Ripper (sustitución de Brutus por Hashcat, más actual y con soporte activo).

La fortaleza de las contraseñas es un pilar básico en la seguridad informática. Evaluar su robustez mediante técnicas de descifrado permite a las organizaciones identificar y corregir debilidades en sus políticas de contraseñas.

¿Qué es Hashcat?



Hashcat es una herramienta avanzada para el descifrado de hashes de contraseñas. Es reconocida por su velocidad y capacidad para aprovechar la potencia de las GPU.

Características sobresalientes:

- Soporte multi-plataforma: Funciona en Windows, Linux y macOS.
- Aprovechamiento de GPU y CPU: Utiliza el poder de procesamiento paralelo de las tarjetas gráficas para acelerar el descifrado.

- ► Amplia compatibilidad de algoritmos: Soporta más de 200 tipos de hashes, incluyendo MD5, SHA-1, SHA-256, NTLM, entre otros.
- ► Modos de ataque versátiles: Diccionario, máscara, combinaciones y ataques híbridos.

¿Qué es John the Ripper?

Es una herramienta de código abierto diseñada para detectar contraseñas débiles. Es ampliamente utilizada por su efectividad y flexibilidad.

Funciones clave:

- ► Personalización de reglas: Permite definir reglas específicas para modificar y probar variaciones de palabras.
- ► Soporte para múltiples formatos: Puede descifrar contraseñas de sistemas UNIX, Windows, bases de datos y más.
- ► Escalabilidad: Puede ejecutarse en entornos distribuidos para acelerar el proceso.

Por ejemplo, imaginemos una institución educativa en Sevilla decide evaluar la seguridad de las contraseñas utilizadas por su personal y estudiantes. El equipo de seguridad:

- → Extrae hashes de contraseñas (con consentimiento y siguiendo las regulaciones de protección de datos).
- → Utiliza Hashcat con una lista de contraseñas comunes en España, incluyendo combinaciones como "Madrid2023" o "SevillaFC".
- → Identifica contraseñas débiles, notificando a los usuarios afectados y promoviendo el uso de contraseñas más seguras.

Herramientas como Brutus, que fueron populares en el pasado para ataques de fuerza bruta en servicios como FTP o HTTP, han quedado obsoletas. Hashcat y John the Ripper ofrecen mejores prestaciones, soporte activo y se actualizan frente a nuevos algoritmos y técnicas de cifrado.

Actividad 8

Investiga las herramientas Hashcat y John the Ripper y elabora una tabla comparativa que incluya sus principales características. Deberás considerar los siguientes aspectos:

Velocidad y rendimiento en el descifrado de contraseñas
Algoritmos de hash soportados
Compatibilidad con hardware (CPU, GPU)
Modos de ataque disponibles (fuerza bruta, diccionario, híbridos, etc.)
Facilidad de uso y configuración
Comunidad, soporte y actualizaciones
Casos de uso típicos y escenarios donde destaca cada herramienta

7. Prueba de autoevaluación.

¿Cuál es el objetivo de herramientas como Ping y Traceroute en la auditoría de sistemas?

a) Identificar vulnerabilidades en aplicaciones web

b) Diagnosticar conectividad y analizar rutas en redes

c) Realizar análisis de fuerza bruta

¿Cuál de las siguientes herramientas es especialmente rápida para escanear grandes segmentos de red?

a) Nmap

b) Netcat

c) Masscan

¿Qué tipo de vulnerabilidades permite detectar Nessus en sistemas y redes?

a) Problemas de conectividad

b) Vulnerabilidades conocidas y configuraciones inseguras

c) Servicios ocultos y archivos no publicados

¿Cuál es la principal ventaja de utilizar herramientas de análisis en la nube como CloudShark?

a) Proporcionan conexión directa entre dispositivos

b) Permiten realizar análisis de protocolos sin software local

c) Mejoran la velocidad de conexión

¿Qué herramienta de fuerza bruta aprovecha la potencia de las GPU para descifrar contraseñas de manera rápida?

a) John the Ripper

b) Hashcat

c) Dirb

La herramienta _____ permite escanear grandes segmentos de red rápidamente.

Para analizar tráfico de red en tiempo real, se utiliza _____.

_____ es una herramienta de fuerza bruta que utiliza el poder de GPU y CPU para descifrar contraseñas.

_____ permite descubrir archivos y directorios ocultos en aplicaciones web.

Herramientas como Burp Suite y OWASP ZAP son especialmente útiles para auditorías en aplicaciones _____.

Descripción de los aspectos sobre cortafuegos en auditorías de Sistemas Informáticos