**Descripción general de tcpdump**

Como analista de seguridad, utilizarás analizadores de protocolo de red para protegerte contra cualquier intrusión. Ya aprendiste los siguientes términos relacionados con el monitoreo y análisis de la red:

* Un **analizador de protocolos de red (rastreador de paquetes)** es una herramienta diseñada para capturar y analizar el tráfico de datos dentro de una red.
* La **captura, o rastreo, de paquetes** es la práctica de capturar e inspeccionar paquetes de datos a través de una red.

En esta lectura, aprenderás más sobre tcpdump, un analizador de protocolos de red que se puede utilizar para capturar y ver comunicaciones de red.

**¿Qué es tcpdump?**

**Tcpdump** es un analizador de protocolos de red con línea de comandos. Una **interfaz de línea de comandos (CLI)** es una interfaz de usuario basada en texto que utiliza comandos para interactuar con la computadora.

Se utiliza para capturar el tráfico de red, que puede guardarse en un **pcap**, es decir, un archivo que contiene paquetes de datos interceptados desde una interfaz o red. Al archivo pcap se puede acceder, o también puede ser analizado o compartido en otra ocasión. Los analistas utilizan tcpdump por una variedad de razones, desde la resolución de problemas de red hasta la identificación de actividades maliciosas. Tcpdump viene preinstalado en muchas distribuciones de Linux y también se puede instalar en otros sistemas operativos basados en Unix, como macOS®.

**Nota**: Es habitual que el tráfico de red esté cifrado, lo que significa que los datos están codificados o no se pueden leer. Inspeccionar los paquetes de red puede requerir descifrar los datos mediante las claves privadas apropiadas.

**Cómo capturar paquetes con tcpdump**

Anteriormente, aprendiste que un **usuario root (raíz) o superusuario de Linux**  tiene privilegios importantes para modificar el sistema. También, viste que el comando **sudo** otorga temporalmente permisos importantes a usuarios específicos en Linux. Al igual que muchas otras herramientas de rastreo de paquetes, deberás tener privilegios a nivel de administrador para capturar el tráfico de red mediante tcpdump. Esto significa que tendrás que iniciar sesión como usuario root o tener la capacidad de usar el comando sudo. Aquí se muestra un desglose de la sintaxis tcpdump para capturar paquetes:

**sudo tcpdump [-i interface] [option(s)] [expression(s)]**

* El comando **sudo tcpdump** comienza a ejecutar tcpdump con permisos importantes como sudo.
* El parámetro **-i** especifica la interfaz de red para capturar el tráfico de red. Debes especificar una interfaz de red desde la que capturar para comenzar a capturar paquetes. Por ejemplo, si especificas **-i any**, detectarás el tráfico de todas las interfaces de red del sistema.
* Las **option(s)** son facultativas y te brindan la capacidad de alterar la ejecución del comando. Las **expression(s)** son una forma de filtrar aún más los paquetes de tráfico de red para que puedas aislarlo. En la siguiente sección, aprenderás más sobre las **option(s)** y **expression(s)**.

**Nota**: Antes de que puedas comenzar a capturar tráfico de red, debes identificar qué interfaz de red deseas usar para capturar paquetes. Puedes usar el indicador **-D** para enumerar las interfaces de red disponibles en un sistema.

**Opciones**

Con tcpdump, puedes aplicar opciones, también conocidas como indicadores, al final de los comandos para filtrar el tráfico de red. Las opciones cortas se abrevian y se representan con un guión y un solo carácter como **-i**. Las opciones largas, en tanto, se explican con un guión doble como **--interface**. Tcpdump tiene más de 50 opciones que puedes explorar a través de [la página man](https://www.tcpdump.org/manpages/tcpdump.1.html). Aquí, examinarás un par de opciones claves de tcpdump, que incluyen cómo escribir y leer archivos de captura de paquetes.

**Nota**: Las opciones distinguen entre mayúsculas y minúsculas. Por ejemplo, una **-w** minúscula es una opción separada con un uso diferente que la opción con una **-W** mayúscula.

**Nota**: Las opciones de tcpdump que se escriben mediante opciones cortas se pueden escribir con o sin un espacio entre la opción y su valor. Por ejemplo, **sudo tcpdump -i any -c 3** y **sudo tcpdump -iany -c3** son comandos equivalentes.

**-w**

Con el indicador **-w**, puedes escribir o guardar los paquetes de red rastreados en un archivo de captura de paquetes en lugar de solo imprimirlos en la terminal. Esto resulta de mucha utilidad  porque puedes consultar este archivo guardado para su posterior análisis. En este comando, tcpdump captura el tráfico de todas las interfaces de red y lo guarda en un archivo de captura de paquetes llamado **packetcapture.pcap**:

**sudo tcpdump -i any -w packetcapture.pcap**

**-r**

Con el indicador **-r**, puedes leer un archivo de captura de paquetes especificando el nombre del archivo como parámetro. Aquí te mostramos un ejemplo de un comando tcpdump que lee un archivo llamado **packetcapture.pcap**:

**sudo tcpdump -r packetcapture.pcap**

**-v**

Como ya viste, los paquetes contienen mucha información. De forma predeterminada, tcpdump no imprimirá toda la información de un paquete. Esta opción te permite controlar la cantidad de información de paquetes que deseas que tcpdump imprima.

Existen tres niveles de verbosidad que puedes usar según la cantidad de información del paquete que deseas que tcpdump imprima. Los niveles son **-v**, **-vv**, y **-vvv**. El nivel de verbosidad aumenta con cada v agregada. La opción más verbosa (abundancia de detalles) puede resultar útil si buscas información de paquetes, como los detalles de los campos de encabezado IP. A continuación, se muestra un ejemplo de un comando tcpdump que lee el archivo **packetcapture.pcap**:

**sudo tcpdump -r packetcapture.pcap -v**

**-c**

La opción **-c** significa conteo. Esta opción te permite controlar cuántos paquetes capturará tcpdump. Por ejemplo, especificar **-c 1** imprimirá un solo paquete, mientras que **-c 10** imprimirá 10. Este ejemplo le indica a tcpdump que solo capture los primeros tres paquetes que rastrea desde **any** (cualquier) interfaz de red:

**sudo tcpdump -i any -c 3**

**-n**

De forma predeterminada, tcpdump realizará la resolución de nombres. Esto significa que tcpdump convierte automáticamente las direcciones IP en nombres. También resolverá puertos a servicios comúnmente asociados que usan estos puertos. Esto puede ser problemático porque tcpdump no siempre es preciso en la resolución de nombres. Por ejemplo, tcpdump puede capturar tráfico desde el puerto 80 y traducir automáticamente el puerto 80 a HTTP en la salida. Aun así, esto es engañoso porque el puerto 80 no siempre va a utilizar HTTP, sino que podría estar utilizando un protocolo diferente.

Además, la resolución de nombres utiliza lo que se conoce como una búsqueda inversa de DNS, que es una consulta que busca el nombre de dominio asociado con una dirección IP. Si realizas una búsqueda inversa de DNS en el sistema de un atacante, puede que se les avise que los estás investigando a través de sus registros de DNS.

El uso del indicador **-n** deshabilita este mapeo automático de números a nombres y se considera la mejor práctica al rastrear o analizar el tráfico. El uso de **-n** no resolverá los nombres de host, mientras que **-nn** no resolverá *ni* los nombres de host ni los puertos. Aquí podrás ver un ejemplo de un comando tcpdump que lee el archivo **packetcapture.pcap** con verbosidad (abundancia de detalles) y deshabilita la resolución de nombres:

**sudo tcpdump -r packetcapture.pcap -v -n**

**Consejo profesional:** Puedes combinar las opciones. Por ejemplo, **-v** y **-n** se pueden combinar como **-vn**. Pero, si una opción acepta un parámetro justo después de él como **-c 1** o **-r capture.pcap**, entonces no podrás combinarlas.

**Expresiones**

El uso de expresiones de filtro en comandos tcpdump también es opcional, pero puede resultar útil saber cómo y cuándo usar expresiones de filtro durante el análisis de paquetes. Existen muchas maneras de utilizar las expresiones de filtro.

Si deseas buscar tráfico de red por protocolo en específico, puedes utilizar expresiones de filtro para aislar los paquetes de red. Por ejemplo, puedes filtrar para encontrar solo tráfico IPv6 mediante la expresión de filtro **ipv6**.

También puedes usar operadores booleanos como **and** (y), **or** (o) o **not** (no) para filtrar aún más el tráfico de red para direcciones IP específicas, puertos y más. El siguiente ejemplo lee el archivo **packetcapture.pcap** y combina dos expresiones **ipv4 and port 80** con el operador booleano **and** (y):

**sudo tcpdump -r packetcapture.pcap -n 'ipv4 and port 80'**

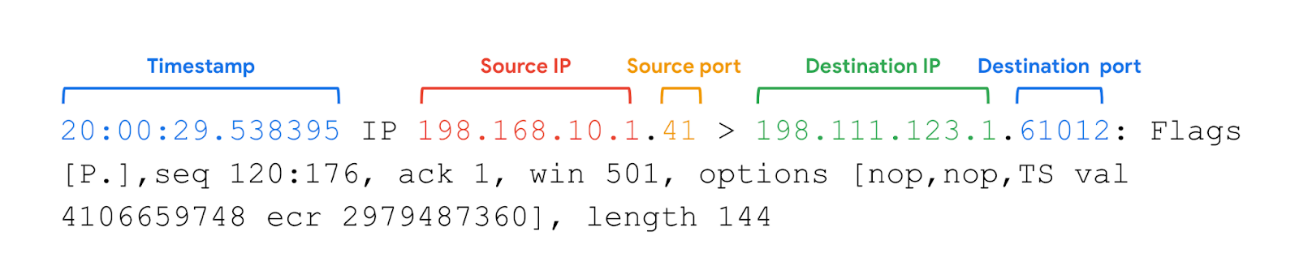
**Consejo profesional:** Puedes usar comillas simples o dobles para asegurarte de que tcpdump ejecute todas las expresiones. También puedes usar paréntesis para agrupar y priorizar diferentes expresiones. Agrupar expresiones es útil para comandos complejos o largos. Por ejemplo, el comando **ipv4 and (port 80 or port 443)** le indica a tcpdump que priorice la ejecución de los filtros encerrados entre paréntesis antes de filtrar para IPv4.

**Interpretación de la salida**

Una vez que ejecutes un comando para capturar paquetes, tcpdump imprimirá la salida del comando como los paquetes rastreados. En la salida, tcpdump imprime una línea de texto para cada paquete con cada línea, comenzando con una marca de tiempo. Aquí se muestra un ejemplo de un comando y la salida para un solo paquete TCP:

**sudo tcpdump -i any -v -c 1**

Este comando le indica a tcpdump que capture paquetes en interfaz de red **-i any**. La opción **-v** imprime el paquete con información detallada y la opción **-c 1** imprime solo un paquete. Esta es la salida de este comando:



1. **Marca de tiempo**: La salida comienza con la marca de tiempo, que empieza con horas, minutos, segundos y fracciones de segundo.
2. **IP de origen:** El origen del paquete lo proporciona su dirección IP de origen.
3. **Puerto de origen:** Este número de puerto es donde se originó el paquete.
4. **IP de destino:** La dirección IP de destino es donde se transmite el paquete.
5. **Puerto de destino:** Este número de puerto es donde se transmite el paquete.

La salida restante contiene detalles de la conexión TCP e incluye indicadores y número de secuencia. La información de **options** es información de paquetes adicionales que proporcionó la opción **-v**.

**Conclusiones clave**

En seguridad, es probable que termines utilizando herramientas de análisis de protocolos de red como tcpdump. Es importante disponer de los conocimientos para capturar, filtrar e interpretar paquetes de red en la línea de comandos.

**Recursos para obtener más información**

* Puedes acceder a más información con los [tutoriales y guías](https://www.tcpdump.org/) de tcpdump, que incluyen recursos educativos adicionales.
* Para conocer más sobre el uso de expresiones para filtrar el tráfico, lee el [tutorial de tcpdump a cargo de Daniel Miessler](https://danielmiessler.com/study/tcpdump/).

Marcar como completo

Me gusta

No me gusta

Informar de un problema