**Investiga los detalles del paquete**

Hasta ahora, viste cómo los analizadores de protocolo de red (rastreadores de paquetes) interceptan las comunicaciones de red. También, aprendiste cómo puedes analizar las capturas de paquetes (pcaps) para obtener información sobre la actividad que ocurre en una red. Como analista de seguridad, utilizarás tus habilidades de análisis de paquetes para inspeccionarlos e identificar actividades sospechosas durante las investigaciones.

En esta lectura, volverás a examinar los encabezados IPv4 e IPv6. Luego, explorarás cómo puedes usar Wireshark para investigar los detalles de los archivos de captura de paquetes.

**Protocolo de Internet (IP)**

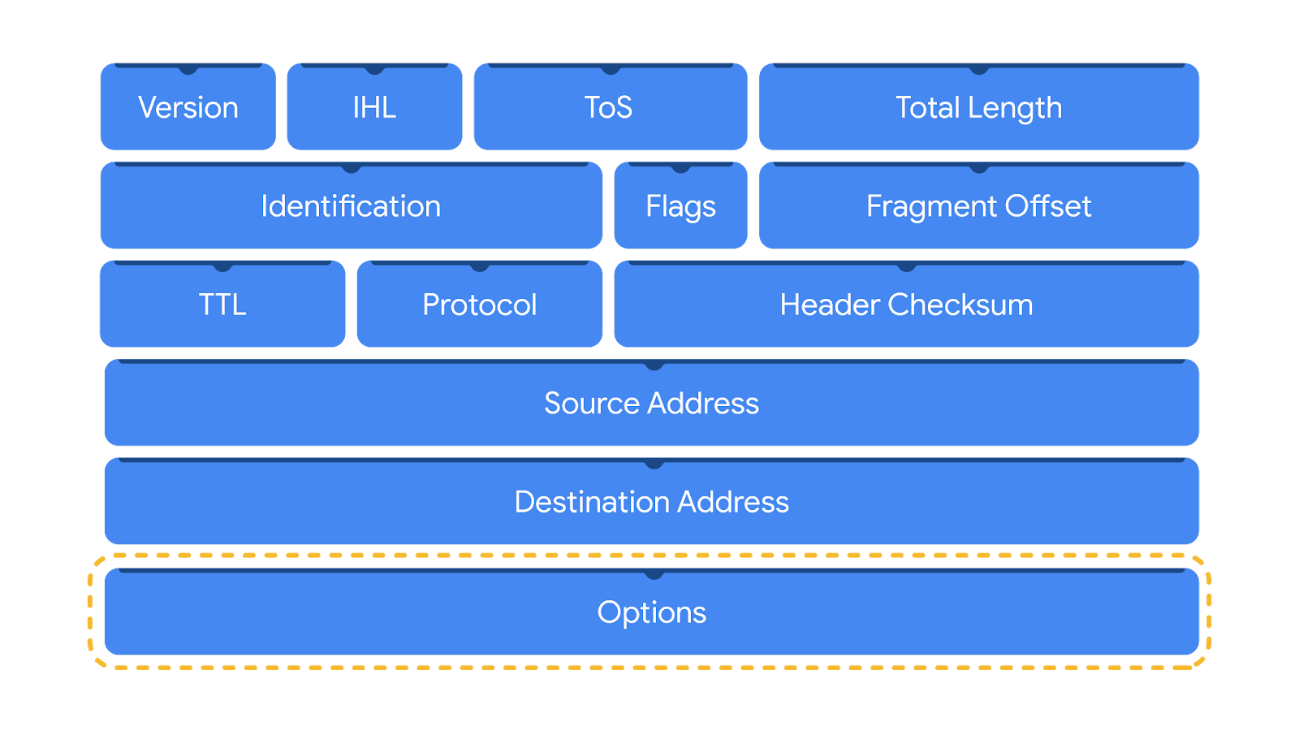
Los paquetes constituyen la base del intercambio de datos en una red, lo que significa que la detección comienza a nivel de paquetes. El **Protocolo de Internet (IP)** incluye un conjunto de estándares utilizados para enrutar y direccionar paquetes de datos mientras viajan de un dispositivo a otro en una red. El IP opera como la base para todas las comunicaciones a través de Internet.

El IP se encarga de garantizar que los paquetes lleguen a sus destinos. Existen dos versiones de IP en uso en la actualidad: IPv4 e IPv6. Ambas versiones utilizan encabezados diferentes para estructurar la información de los paquetes.

**IPv4**

IPv4 es la versión más utilizada de IP. En el encabezado hay 13 campos:

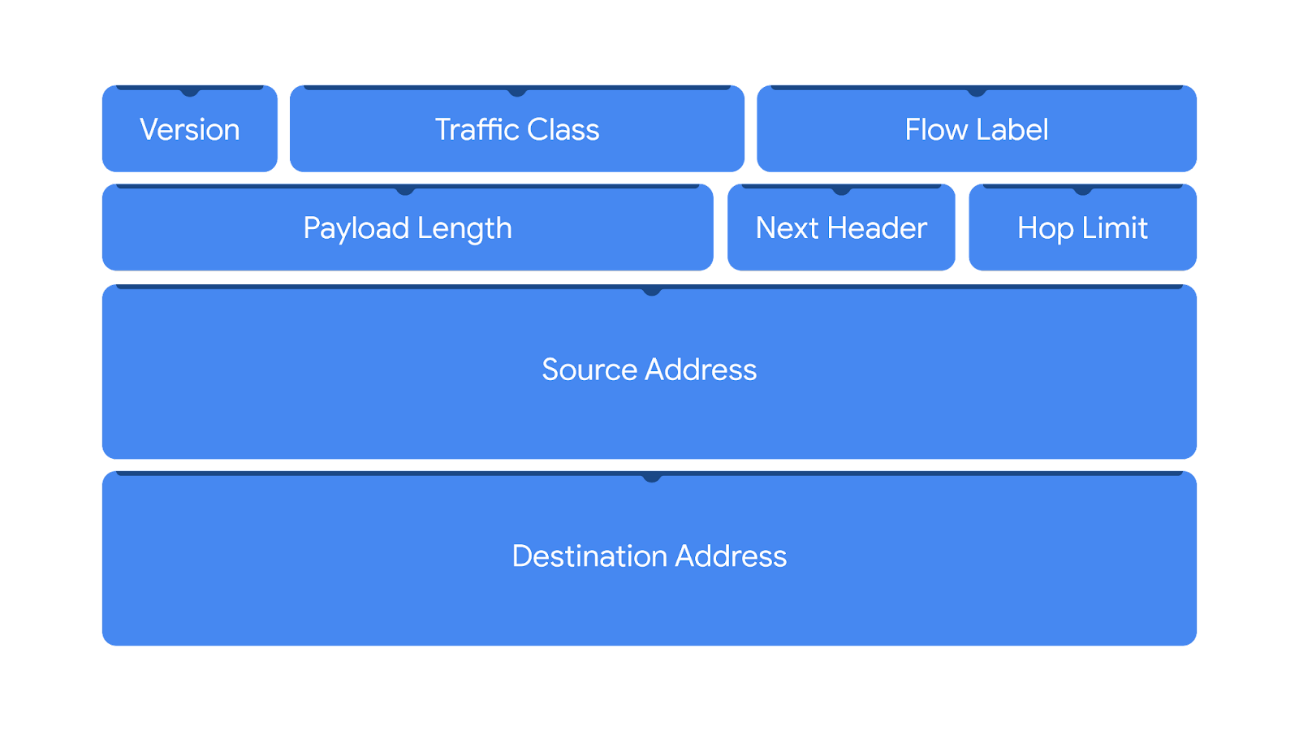
* **Versión**: Indica la versión de IP. Para un encabezado IPv4, se utiliza IPv4.
* **Longitud del encabezado de Internet (IHL)**: Especifica la longitud del encabezado IPv4, incluidas las opciones.
* **Tipo de servicio (ToS)**: Proporciona información sobre la prioridad del paquete para la entrega.
* **Longitud total**: Especifica la longitud total de todo el paquete IP, incluidos el encabezado y los datos.
* **Identificación**: Especifica un identificador único para fragmentos de un paquete IP original para que puedan volver a ensamblarse una vez que lleguen a su destino. Esto es así porque los paquetes que son demasiado grandes para enviar se fragmentan en piezas más pequeñas.
* **Indicadores**: Proporciona información sobre la fragmentación de paquetes, incluso si se fragmentó el paquete original y si hay más fragmentos en tránsito.
* **Desplazamiento de fragmentos**: Se utiliza para identificar la secuencia correcta de los fragmentos.
* **Tiempo de vida (TTL)**: Limita el tiempo que un paquete puede circular en una red, lo que evita que los routers envíen los paquetes de forma indefinida.
* **Protocolo**: Especifica el protocolo utilizado para el área de datos del paquete.
* **Suma de comprobación de cabecera**: Especifica un valor de suma de comprobación que se utiliza para comprobar el error del encabezado.
* **Dirección de origen**: Especifica la dirección de origen del emisor.
* **Dirección de destino**: Especifica la dirección de destino del receptor.
* **Opciones**: Es opcional y se puede utilizar para aplicar opciones de seguridad a un paquete.



**IPv6**

Gracias a su gran espacio de direcciones, la adopción de IPv6 ha ido en aumento. Su encabezado cuenta con ocho campos:

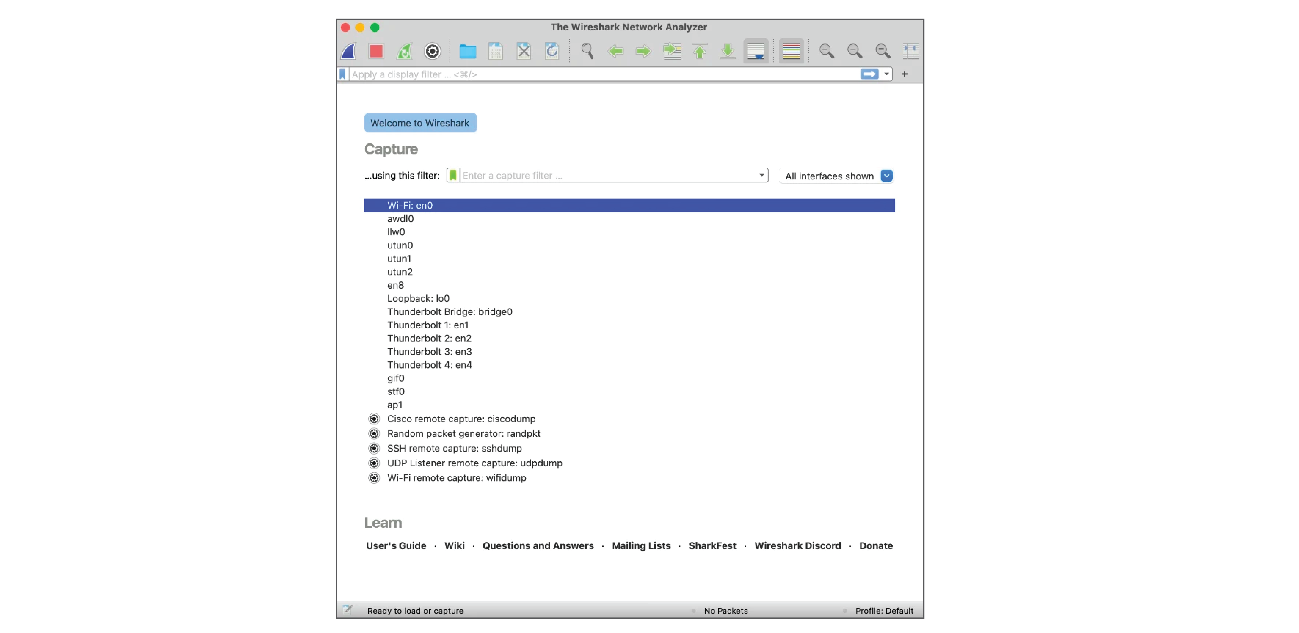
* **Versión**: Indica la versión de IP. Para un encabezado IPv6, se utiliza IPv6.
* **Clase de tráfico**: Similar al campo Tipo de servicio IPv4, proporciona información sobre la prioridad o clase del paquete para ayudar con la entrega.
* **Etiqueta de flujo**: Identifica los paquetes de un flujo. Un flujo es la secuencia de paquetes enviados desde una fuente específica.
* **Longitud de carga útil**: Especifica la longitud del área de datos del paquete.
* **Encabezado siguiente**: Indica el tipo de encabezado que sigue al encabezado IPv6, como TCP.
* **Límite de salto**: Similar al campo Tiempo de vida de IPv4, restringe el tiempo que un paquete puede viajar en una red antes de que se descarte.
* **Dirección de origen**: Especifica la dirección de origen del emisor.
* **Dirección de destino**: Especifica la dirección de destino del receptor.



Los campos de encabezado contienen información valiosa para investigaciones y herramientas como Wireshark que ayudan a mostrarlos en un formato legible para las personas.

**Wireshark**

**Wireshark** es un analizador de protocolos de red de código abierto. Utiliza una interfaz gráfica de usuario (GUI), lo que facilita la visualización de las comunicaciones de red con fines de análisis de paquetes. Wireshark tiene muchas características para explorar que no se alcanzan a cubrir en este curso. Te enfocarás en cómo usar el filtrado básico para aislar los paquetes de red de modo que puedas encontrar lo que necesitas.



**Filtros de visualización**

Los filtros de visualización de Wireshark te permiten aplicar filtros a  archivos de captura de paquetes. Esto es útil cuando estás inspeccionando capturas de paquetes con grandes volúmenes de información. Los filtros de visualización te ayudarán a encontrar información específica que sea relevante para tu investigación. Puedes filtrar paquetes en función de información como protocolos, direcciones IP, puertos y cualquier otra propiedad encontrada en un paquete. Aquí te centrarás en la sintaxis de filtrado de visualización y en el filtrado de protocolos, direcciones IP y puertos.

**Operadores de comparación**

Puedes utilizar diferentes operadores de comparación para localizar campos y valores específicos en los encabezados. Los operadores de comparación pueden expresarse utilizando abreviaturas o símbolos. Por ejemplo, este filtro que utiliza el símbolo igual **==** en este filtro **ip.src == 8.8.8.8** es idéntico al uso de la abreviatura **eq** en este filtro **ip.src eq 8.8.8.8**.

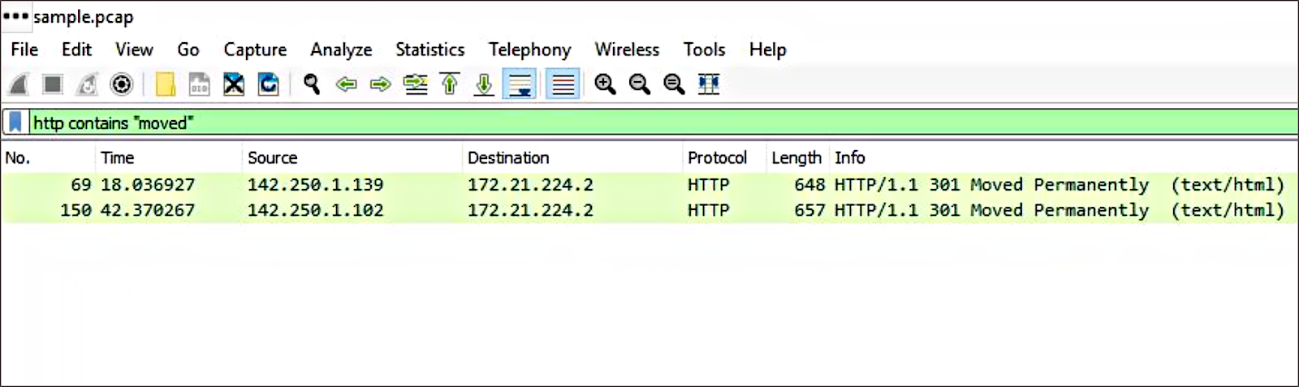
Esta tabla resume los diferentes tipos de operadores de comparación que puedes utilizar para el filtrado de visualización.

| **Tipo de operador** | **Símbolo** | **Abreviatura** |
| --- | --- | --- |
| Igual | **==** | eq |
| No es igual | **!=** | ne |
| Mayor que | **>** | gt |
| Menor que | **<** | lt |
| Mayor que o igual a | **>=** | ge |
| Menor que o igual a | **<=** | le |

**Consejo profesional:** Puedes combinar operadores de comparación con operadores lógicos booleanos como **and** (y) y **or** (o) para crear filtros de visualización complejos. Los paréntesis también se pueden utilizar para agrupar expresiones y priorizar términos de búsqueda.

**Operador contains**

El operador **contains** se utiliza para filtrar paquetes que contienen una coincidencia exacta de una cadena de texto. Aquí, podrás ver un ejemplo de un filtro que muestra todos los flujos HTTP que coinciden con la palabra clave **"moved"** (movido).

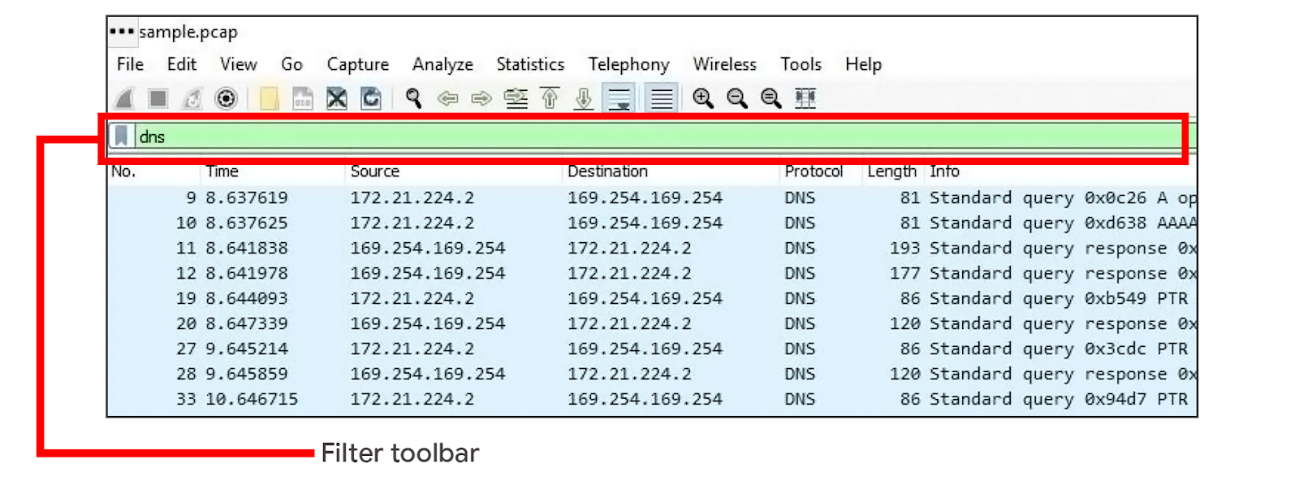


**Operador matches**

El operador **matches** se utiliza para filtrar paquetes basados en la expresión regular (regex) que se especifica. La expresión regular es una secuencia de caracteres que forma un patrón. Más adelante en este programa, explorarás más sobre las expresiones regulares.

**Barra de herramientas de filtrado**

Puedes aplicar filtros a una captura de paquetes mediante la barra de herramientas de filtrado de Wireshark. En este ejemplo, **dns** es el filtro aplicado, lo que significa que Wireshark solo mostrará paquetes que contengan el protocolo DNS.



**Consejo profesional**: Wireshark utiliza diferentes colores para representar los protocolos. Puedes personalizar los colores y crear tus propios filtros.

**Filtrar por protocolos**

El filtrado de protocolos es una de las formas más simples de utilizar los filtros de visualización. Basta solo con ingresar el nombre del protocolo para filtrar. Por ejemplo, para filtrar paquetes DNS, escribe **dns** en la barra de herramientas de filtrado. A continuación, podrás ver una lista de algunos protocolos que puedes filtrar:

* dns
* http
* ftp
* ssh
* arp
* telnet
* icmp

**Filtrar una dirección IP**

Puedes utilizar filtros de visualización para localizar paquetes con una dirección IP específica.

Por ejemplo, si deseas filtrar paquetes que contienen una dirección IP específica, utiliza **ip.addr**, seguido de un espacio, el operador de comparación **==** igual y la dirección IP. A continuación se muestra un ejemplo de un filtro de visualización que filtra la dirección IP **172.21.224.2**:

**ip.addr == 172.21.224.2**

Para filtrar paquetes que se originan desde una dirección IP de origen específica, puedes usar el filtro **ip.src**. Aquí, puedes ver un ejemplo que busca la dirección IP de origen **10.10.10.10**:

**ip.src == 10.10.10.10**

Para filtrar los paquetes entregados a una dirección IP de destino específica, puedes utilizar el filtro **ip.dst**. Aquí, puedes ver un ejemplo que busca la dirección IP de destino **4.4.4.4**:

**ip.dst == 4.4.4.4**

**Filtrar una dirección MAC**

También puedes filtrar paquetes según la **dirección de control de acceso al medio (MAC)**. Como actualización, una dirección MAC es un identificador alfanumérico único que se asigna a cada dispositivo físico en una red.

A continuación, se muestra un ejemplo:

**eth.addr == 00:70:f4:23:18:c4**

**Filtrar por puertos**

El filtrado de puertos se utiliza para filtrar paquetes en función de los números de puerto. Esto es útil cuando quieres aislar tipos específicos de tráfico. El tráfico DNS utiliza el puerto TCP o UDP 53, por lo que esto enumerará el tráfico relacionado con las consultas y respuestas de DNS únicamente.

Por ejemplo, si quieres filtrar un puerto UDP:

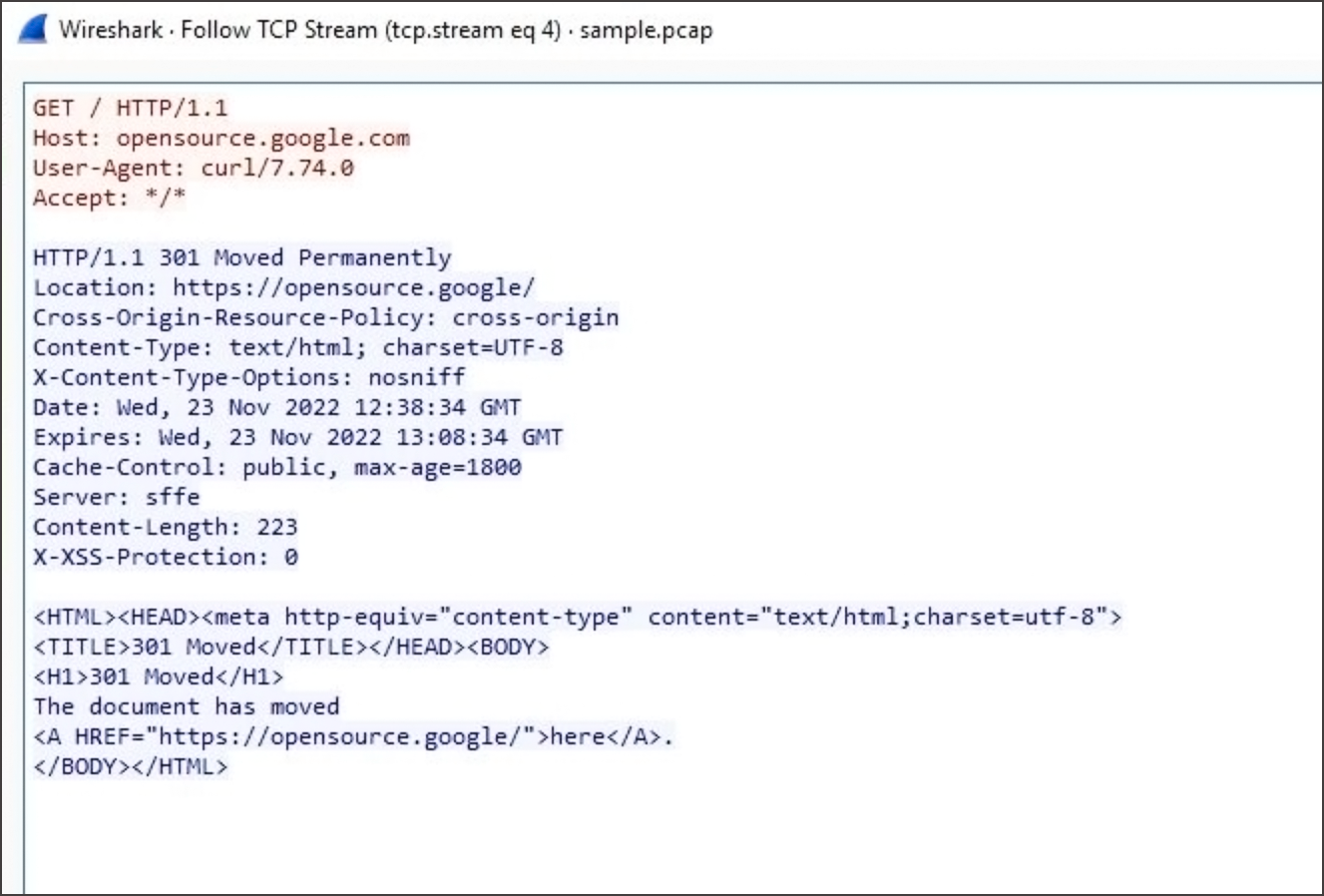
**udp.port == 53**

Del mismo modo, también puedes filtrar puertos TCP:

**tcp.port == 25**

**Seguir flujos**

Wireshark ofrece una función que permite filtrar paquetes específicos para un protocolo y ver flujos. Un flujo o conversación es el intercambio de datos entre dispositivos que utilizan un protocolo. Wireshark vuelve a ensamblar los datos que se transfirieron en el flujo en un formato legible.



Seguir un flujo de protocolo es útil cuando se trata de comprender los detalles de una conversación. Por ejemplo, puedes examinar los detalles de una conversación HTTP para ver el contenido de los mensajes intercambiados de solicitud y respuesta.

**Conclusiones clave**

En esta lectura, exploraste los filtros de visualización básicos con Wireshark. El análisis de paquetes es una habilidad fundamental que seguirás desarrollando con el tiempo en tu recorrido por el mundo de la ciberseguridad. Pon en práctica tus habilidades en la próxima actividad y explora los detalles de un archivo de captura de paquetes utilizando Wireshark.

**Recursos**

* Para obtener más información sobre las características y capacidades completas de Wireshark, revisa la [guía oficial del usuario de Wireshark](https://www.wireshark.org/docs/wsug_html/).