Universidad de los Andes
Facultad de Ingeniería

Departamento de Ingeniería de Sistemas y Computación
Infraestructura de Comunicaciones

ABET Engineering Accreditation Commission

Asist. De laboratorio: Nathalia Quiroga
n.quiroga@uniandes.edu.co

GUÍA 1 – TUTORIAL WIRESHARK

1. OBJETIVO (S)

Aprender el funcionamiento básico de un analizador de protocolos, específicamente Wireshark, y comprender los resultados generados por éste, obteniendo así un mayor conocimiento sobre algunos protocolos usados comúnmente en una red.

2. ANALIZADOR DE PROTOCOLOS WIRESHARK

El análisis de paquetes o análisis de protocolos se describe como el proceso de captura y la interpretación de datos en tiempo real a medida que los paquetes circulan a través de una red, esto con el fin de entender mejor lo que está sucediendo en dicha red. Análisis de paquetes se realiza normalmente por un analizador de protocolos, una herramienta que se utiliza para capturar los datos de la red que viajan a través del medio físico (Ejemplo un cable de cobre).

Análisis de paquetes puede ayudarnos a entender las características de la red, saber quién está en una red, determinar quién o qué está utilizando el ancho de banda disponible, identificar las horas pico de uso de red, identificar a posibles ataques o actividad maliciosa, y encontrar aplicaciones no seguras.

Hay varios tipos de programas de detección de paquetes, incluyendo libres y comerciales. Cada programa está diseñado con diferentes objetivos. Algunos de los programas de análisis de paquetes más populares son tcpdump (un programa de línea de comandos), OmniPeek y Wireshark.

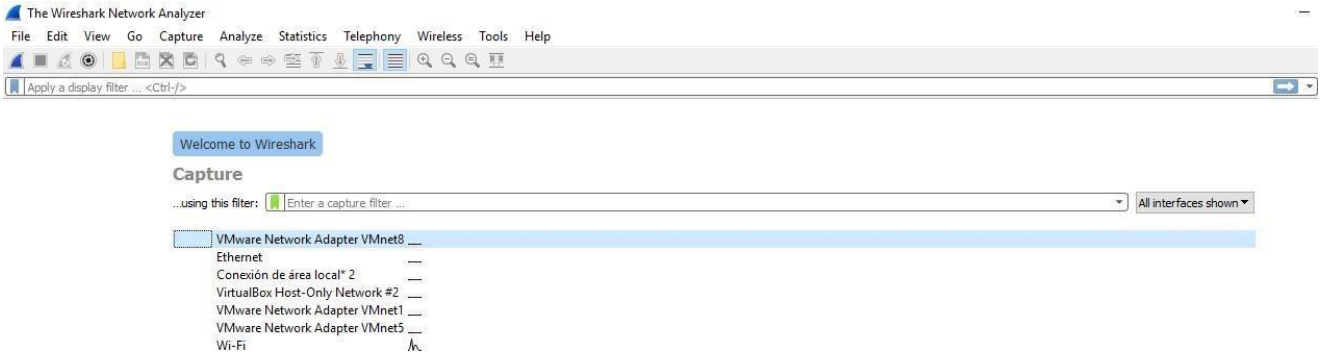
Wireshark es un analizador de paquetes de red (sniffer). Un analizador de paquetes de red intenta capturar paquetes en la red e intenta visualizar los datos de esos paquetes tan detalladamente como sea posible. Se puede pensar en un analizador de paquetes de red como un dispositivo de medida usado para examinar que está pasando al interior de un cable de red.

Wireshark tiene todas las características estándares que se pueden esperar en un analizador de protocolos; su licencia es de código abierto y puede ser ejecutado sobre plataformas como Unix, Linux y Windows. Wireshark sobresale en el número de protocolos que soporta (más de 850). Estos protocolos van desde los más comunes como IP y DHCP hasta protocolos propietarios más avanzados como AppleTalk.

3. USO BÁSICO DE WIRESHARK

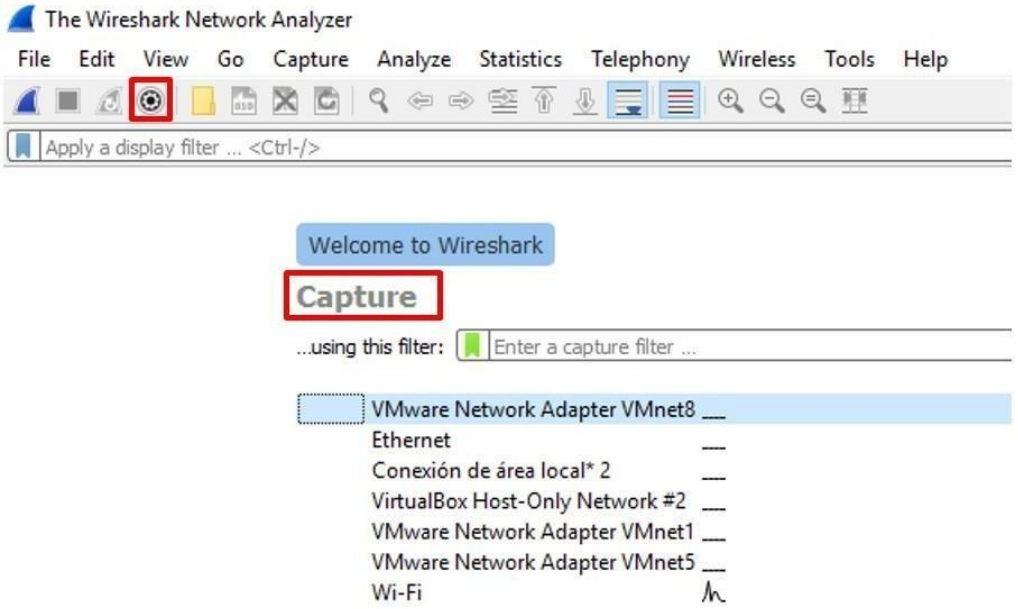
Una vez se ejecuta el programa Wireshark, se verá una ventana como la siguiente:

Figura. 1. Ventana inicial software Wireshark.



Para iniciar una captura, seleccionar la palabra en gris "Capture", en la ventana principal. También se puede acceder a esta opción a través del segundo ícono del menú, "Show the capture options".

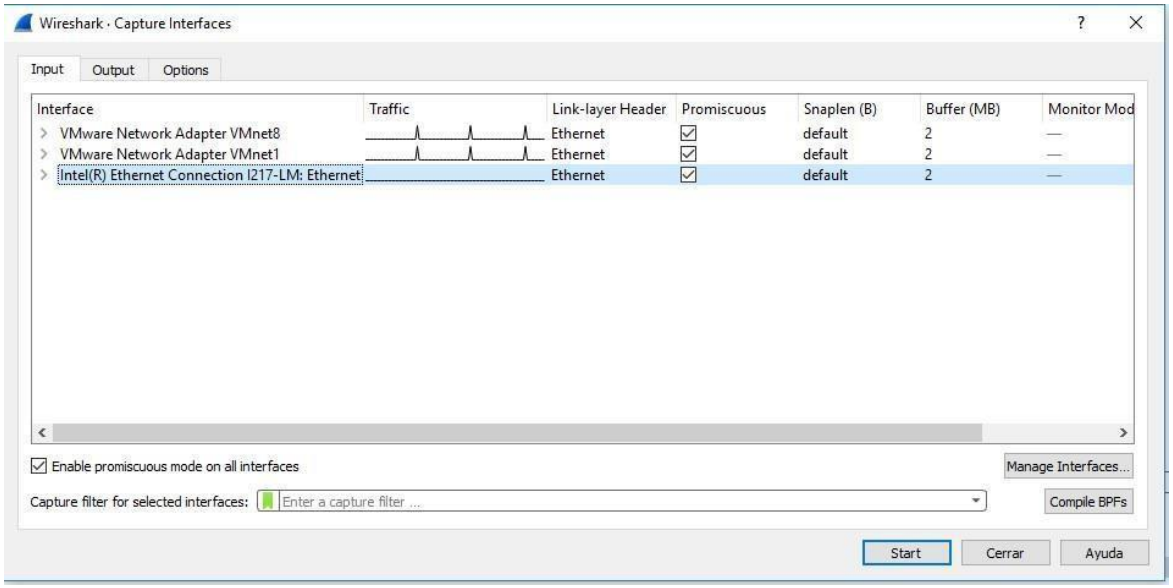
Figura. 2. Iniciar captura de tráfico.



En la ventana que aparece, realizar las siguientes acciones en las tres pestañas:

Seleccione la tarjeta de red a usar para capturar los paquetes.
Seleccione la opción para capturar paquetes en modo promiscuo.
Verifique que no haya ningún filtro, en "Capture Filter".
No remueva las selecciones en "Display Options"
Verifique que las opciones seleccionadas en "Name Resolution" involucren las direcciones MAC, las direcciones de capa de red y las direcciones de capa de transporte. Seleccionar la opción "pcap".

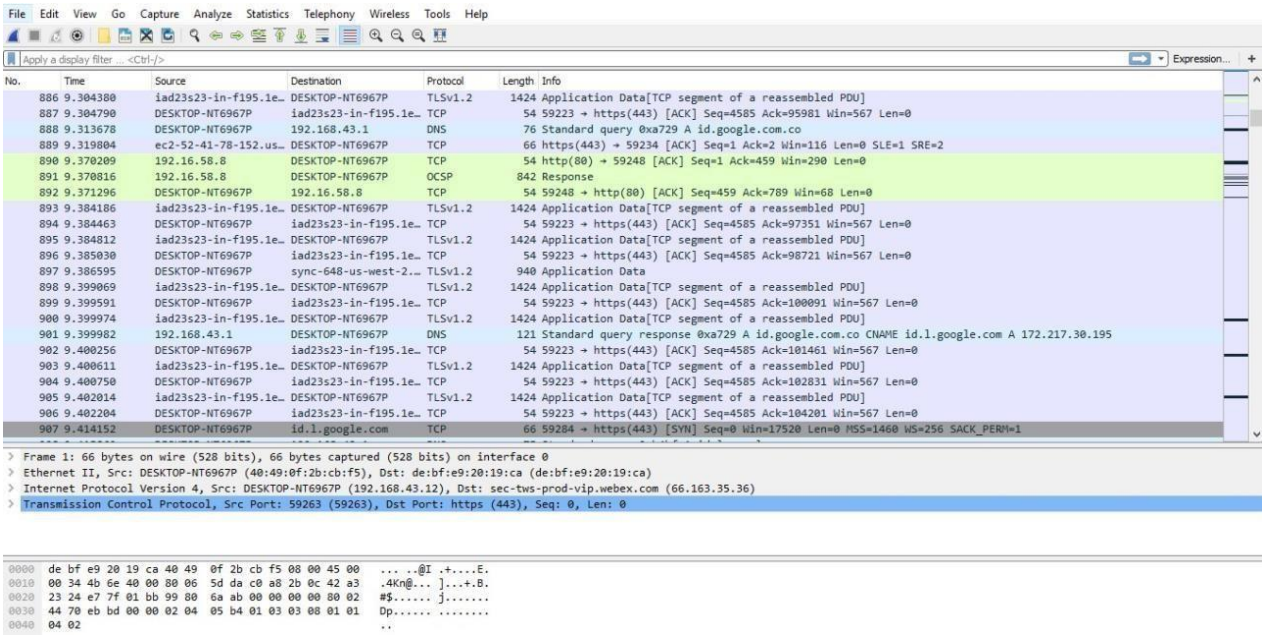
Figura. 3. Ventana de selección de interfaz para capturar trafico.



Al hacer click en "Start" se inicia la captura de tráfico que muestra el número de paquetes capturados, con sus respectivos protocolos e información.

Para terminar la captura de paquetes, hacer click en "Stop". Después de esto se podrá observar una ventana de resultados, la misma que se visualizaba previamente durante la captura.

Figura. 4. Captura de tráfico corriendo.



El filtrado de tráfico permite desplegar sólo aquellos paquetes de interés para el usuario. Para hacer esto, se usa la barra "**Filter**". Se puede escribir directamente sobre ella la condición sobre los paquetes que debe ser cumplida, o se puede usar la ventana asociada al botón "**Expression…**". Ahí se selecciona el nombre del campo, y su relación con un valor.

A continuación, se muestra el filtro de paquetes que poseen el puerto 80 TCP como origen o como destino. Para que un filtro tenga efecto, se debe hacer click en "Apply". Si se desea nuevamente mostrar todos los paquetes, se debe hacer click en "Clear".

Figura. 5. Configuración de filtro de paquete por el puesto 80.

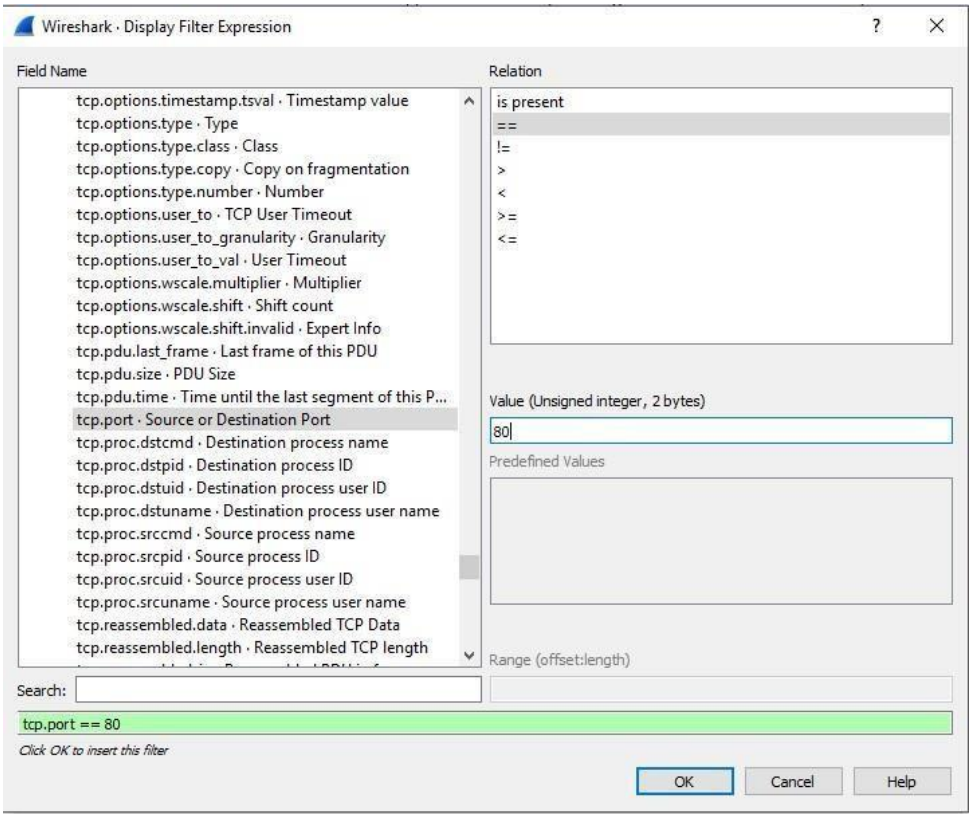
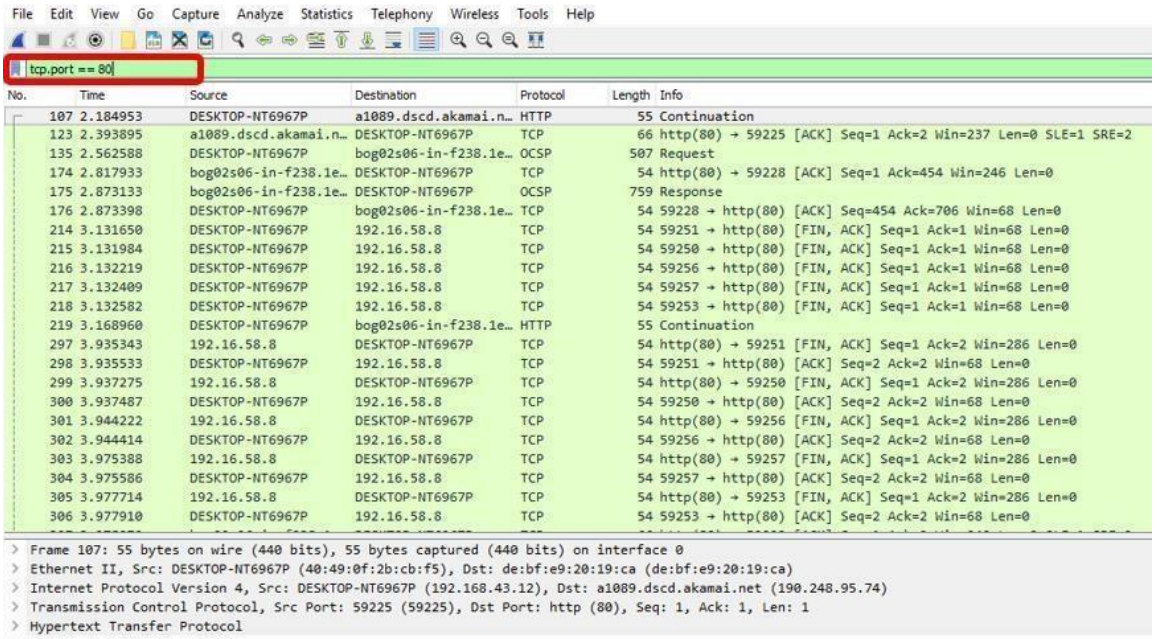
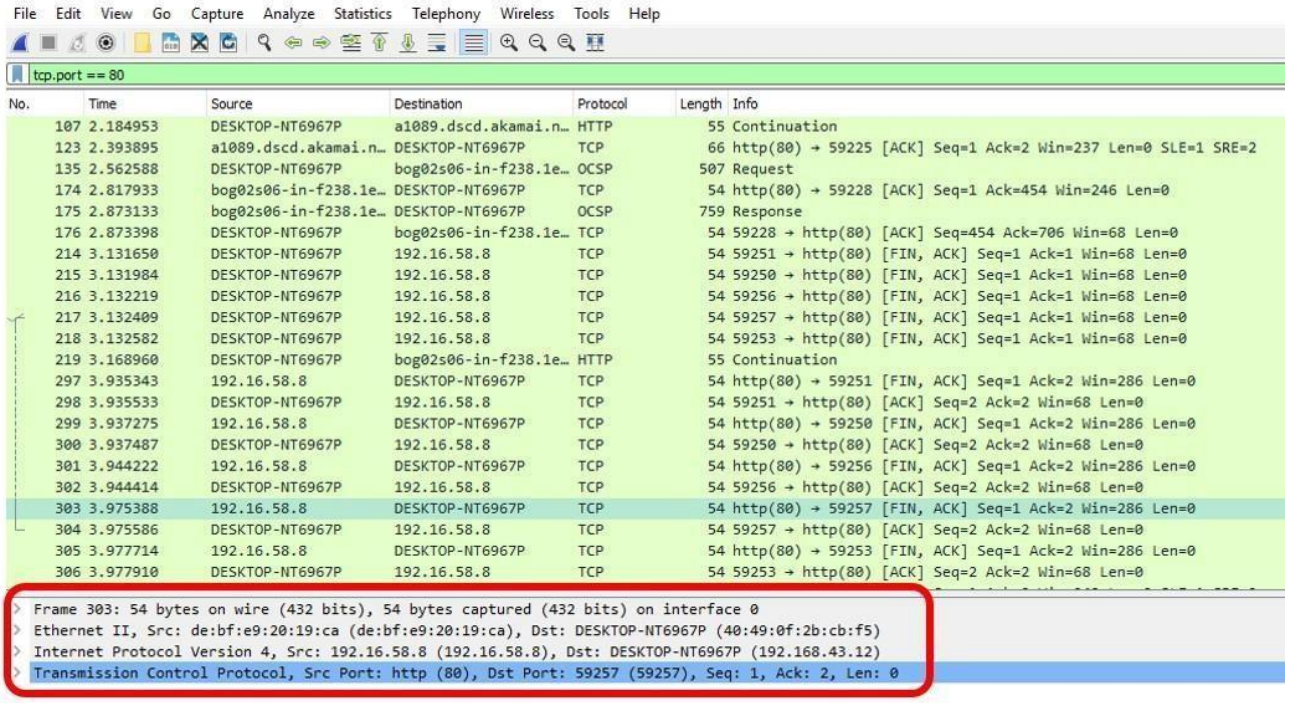


Figura. 6. Trafico filtrado por el puerto TCP 80.



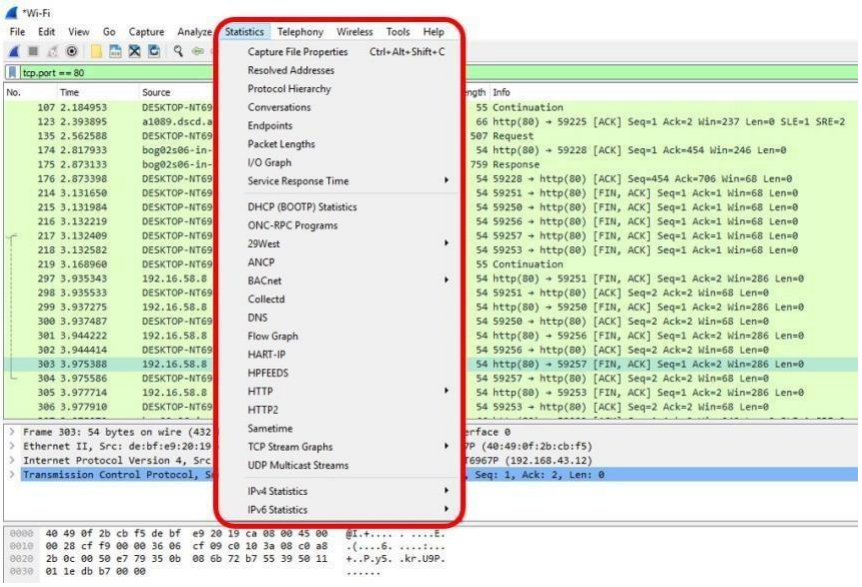
Para el caso particular del protocolo TCP, la información de cada captura es mostrada en cuatro partes: "Frame", "Ethernet", "Internet Protocol", y "Transmission Control Protocol".

Figura. 7. Información en detalle de un registro de captura de trafico.



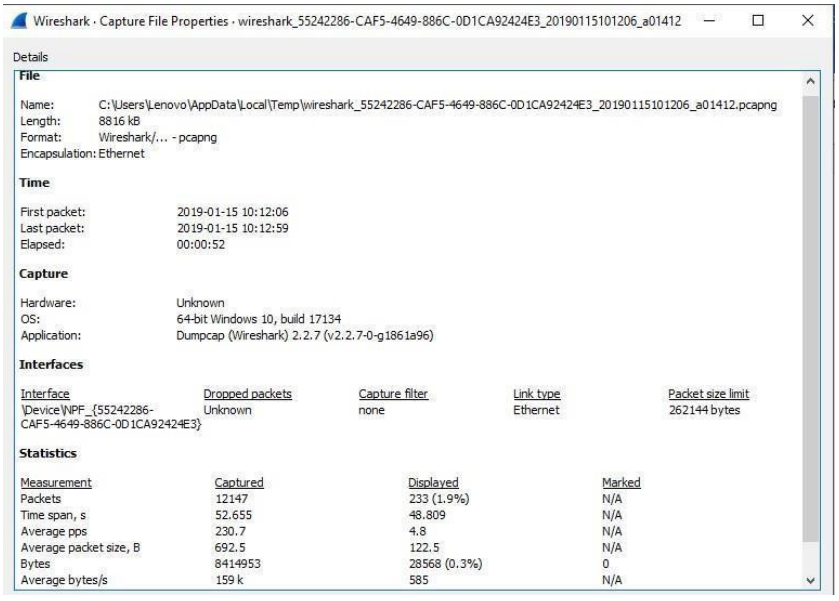
Wireshark posee un completo conjunto de herramientas que permiten obtener estadísticas. Éstas incluyen resúmenes, gráficas, jerarquías de protocolos, conversaciones, etc. Se accede a la mayoría de ellas a través del menú "Statistics".

Figura. 8. Opciones de pestaña Statistics.



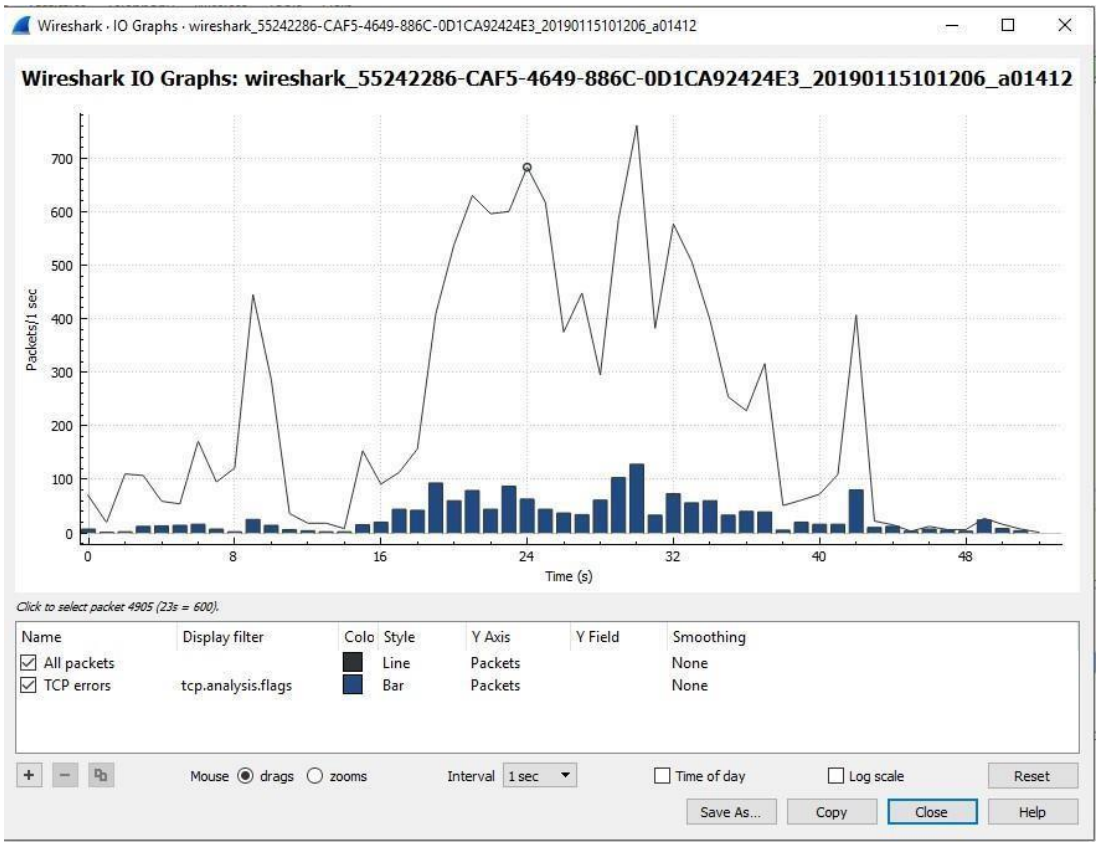
La siguiente figura muestra la opción "Capture File Properties". En ella se muestran datos de tráfico capturado y mostrado.

Figura. 9. Ventana Capture File properties.



La siguiente gráfica muestra la opción "Statistics → IO Graphs", donde se han seleccionado los parámetros de tráfico TCP para todos los paquetes.

Figura. 10. Ventana I/O Graphs para el trafico TCP.



Las siguientes gráficas se refieren a la configuración de la opción "Flow Graph" y a la muestra de resultados del mismo. Esto muestra el flujo de mensajes entre uno o más sistemas finales, en su orden cronológico.

Figura. 11. Herramienta Flow Graph.

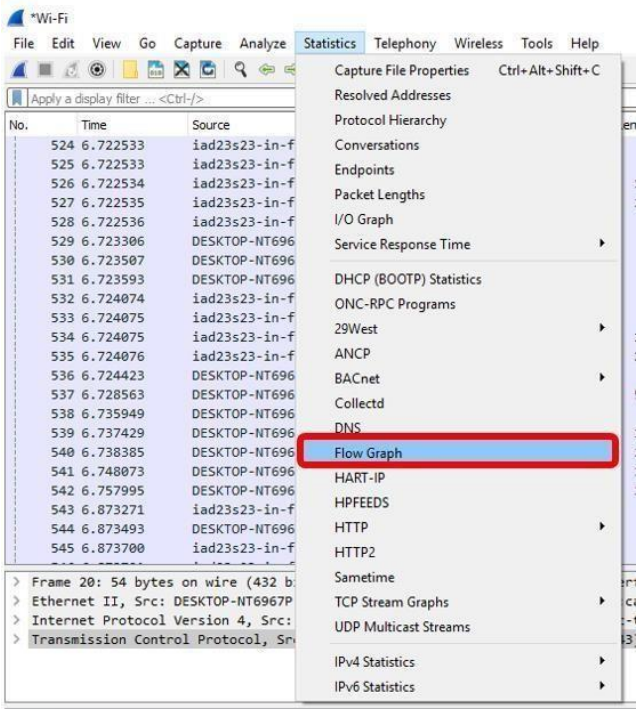
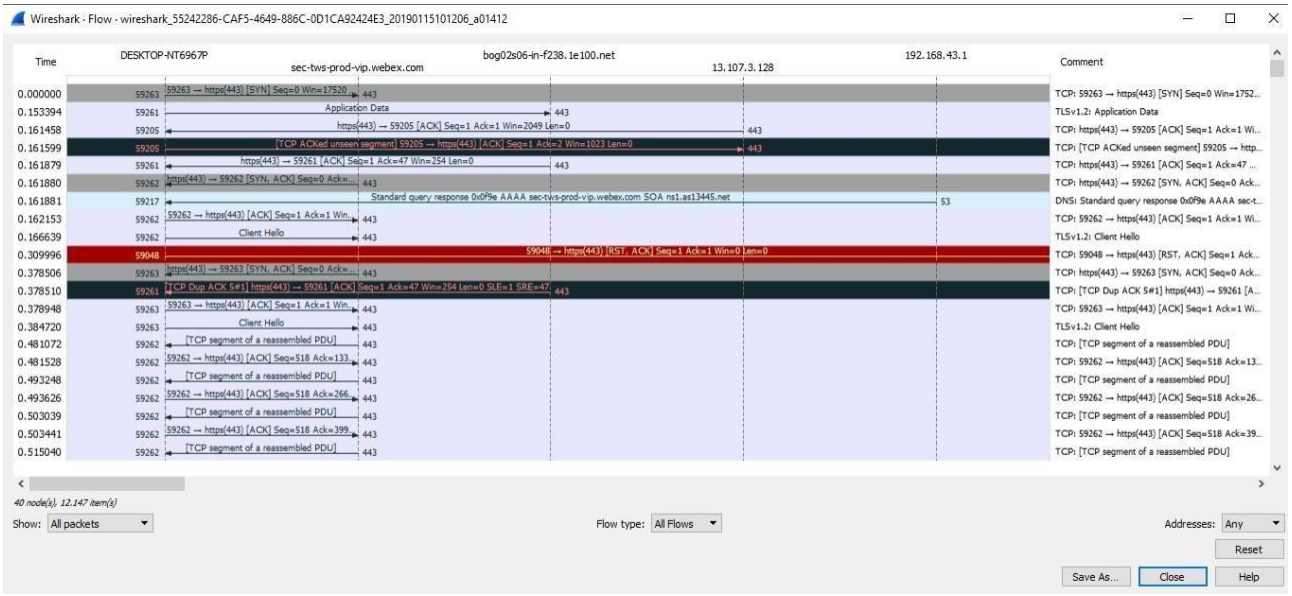
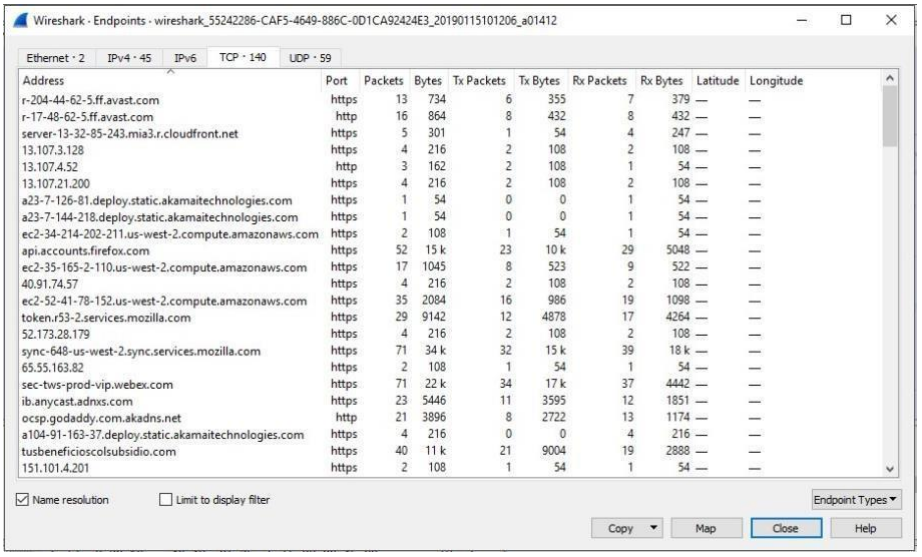


Figura. 12. Ventana de Flowgraph para una captura de trafico.



Las siguientes gráficas muestran el uso de la opción "Statistics → Endpoints". En ellas se puede observar información referente a cada uno de los endpoints en el caso de los protocolos TCP y UDP.

Figura. 13. Ventana de herramienta Statistics-Endpoints.



4. BIBLIOGRAFÍA

[1] The Wireshark Field Guide - Analyzing and Troubleshooting Network Traffic. Robert Shimonski. Syngress.

[2] Network Analysis Using Wireshark. Yoram Orzach. Packt Publishing.

[3] Computer Networking, a top-down approach. James Kurose, Keith Ross. Addison-Wesley, 6th ed.

HISTORIAL DE REVISIONES

Fecha	Autor	Observaciones
05/02/2021	**Arnold Andres Lara** a.larav@uniandes.edu.co	Ajustes de redacción y actualización del documento
11/02/2020	**Arnold Andres Lara** a.larav@uniandes.edu.co	Ajustes de redacción.
14/01/2019	**Jonatan Legro Pastrana** j.legro@uniandes.edu.co	Actualización de la documentación para la versión de Wireshark 2.6.6