Redes de Computadores 2022-1
Hands-on 1: Introducción a Redes
Nombres: David Alsina, Laura Ortiz, Alejandra Campo

Profesor: David F. Celeita R.

Objetivos:

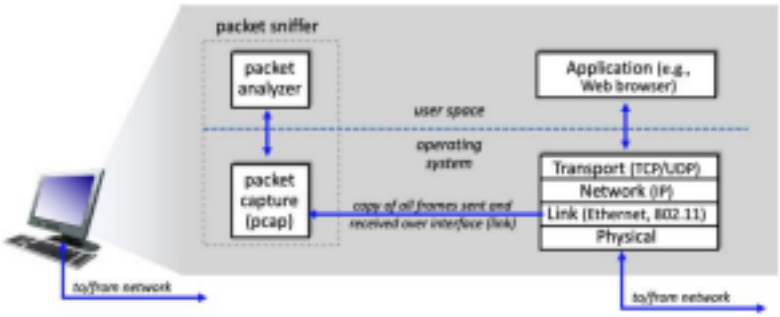
- Familiarizarse con el software para rastreo de paquetes Wireshark
- Identificar detalles de paquetes

Entrega:

- Fecha: Jueves 3 de Febrero / 2022 (E-Aulas 23:59)
- Modalidad: Por grupos
- Metodología: Observar y aprender haciendo (Hands-on)

Contexto:

La comprensión de los protocolos de red a menudo se puede profundizar en gran medida al "visualizar los protocolos en acción" y al "jugar con los protocolos": observar la secuencia de mensajes intercambiados entre dos entidades de protocolo, profundizar en los detalles de la operación del protocolo y hacer que los protocolos realicen ciertas acciones, luego observar estas acciones y sus consecuencias. Esto se puede hacer en escenarios simulados o en un entorno de red "real" como Internet. En este curso algunos talleres integrarán habilidades de programación en Python y también el uso de entornos como Wireshark, ejecutando varias aplicaciones de red en diferentes escenarios utilizando su propio PC. Observará los protocolos de red en su computadora "en acción", interactuando e intercambiando mensajes con entidades de protocolos que se ejecutan en otras partes de Internet. En este primer laboratorio, se familiarizará con Wireshark y realizará algunas capturas y observaciones de paquetes simples. La herramienta básica para observar los mensajes intercambiados entre las entidades de protocolo en ejecución se denomina rastreador de paquetes (sniffer). Como sugiere el nombre, un rastreador de paquetes captura ("snif/olfatea") los mensajes que se envían / reciben desde / por su computadora; normalmente también almacenará y/o mostrará el contenido de los diversos campos de protocolo en estos mensajes capturados. Un rastreador de paquetes en sí mismo es pasivo. Observa los mensajes enviados y recibidos por aplicaciones y protocolos que se ejecutan en su computadora, pero nunca envía paquetes por sí mismo. De manera similar, los paquetes recibidos nunca se dirigen explícitamente al rastreador de paquetes. En su lugar, un rastreador de paquetes recibe una copia de los paquetes que se envían / reciben desde / por la aplicación y los protocolos que se ejecutan en su máquina.



Estructura de un rastreador de paquetes como elemento pasivo (J.F. Kurose and K.W. Ross)

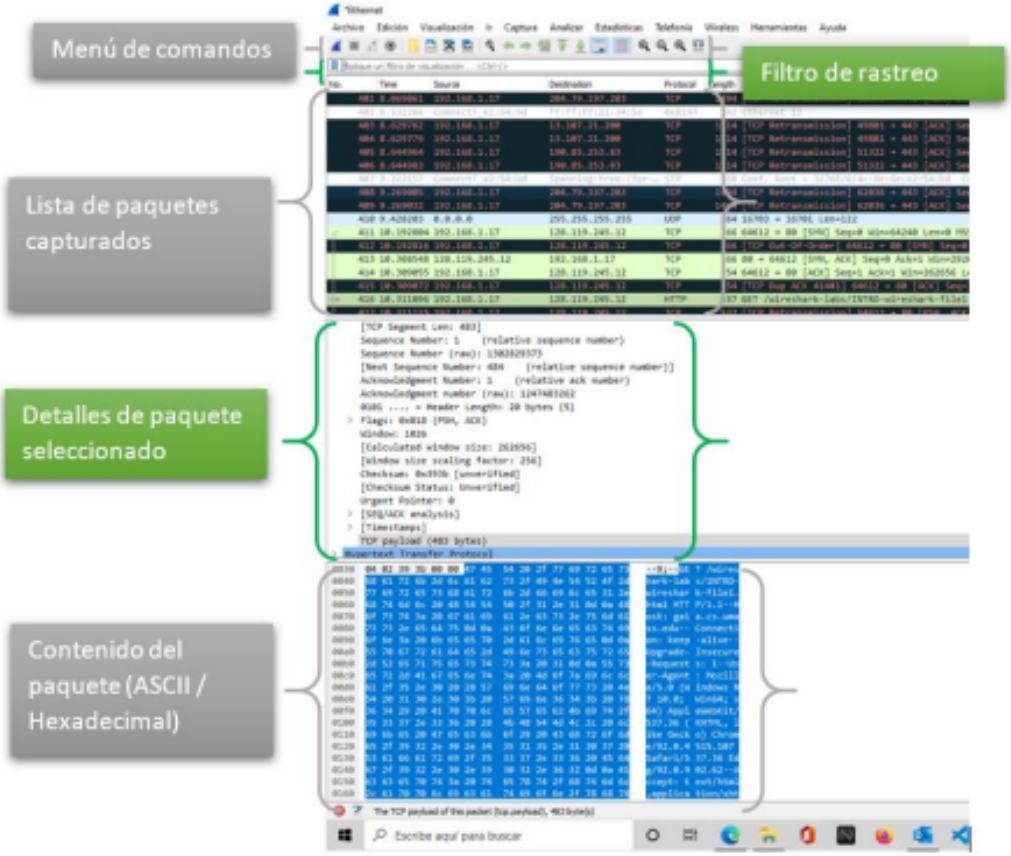
A la derecha de la figura se encuentran los protocolos (en este caso, los protocolos de Internet) y

las aplicaciones (como un navegador web o un cliente de correo electrónico) que normalmente se ejecutan en su computador. El rastreador de paquetes, que se muestra dentro del rectángulo resaltado en rojo, es una adición al software habitual de su computadora y consta de:

1. **Una biblioteca de captura de paquetes:**recibe una copia de cada trama de la capa de enlace que se envía o recibe de su computadora a través de una interfaz determinada (capa de enlace, como Ethernet o WiFi). Los mensajes intercambiados por protocolos de capa superior como HTTP, FTP, TCP, UDP, DNS o IP finalmente se encapsulan en tramas de capa de enlace que se transmiten a través de medios físicos, como un cable Ethernet o WiFi 802.11.
2. **El analizador de paquetes:** muestra el contenido de todos los campos dentro de un mensaje de protocolo. Para hacerlo, el analizador de paquetes debe "comprender" la estructura de todos los mensajes intercambiados por protocolos. El analizador de paquetes entiende el formato de las tramas Ethernet y, por lo tanto, puede identificar el datagrama IP dentro de una trama Ethernet. También comprende el formato del datagrama IP, de modo que puede extraer el segmento TCP dentro del datagrama IP. Finalmente, comprende la estructura del segmento TCP, por lo que puede extraer el mensaje HTTP contenido en el segmento TCP.

Actividad 1:

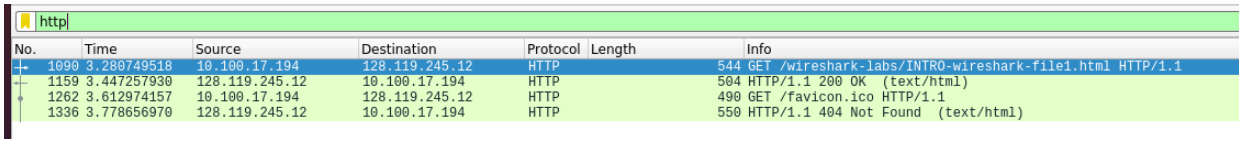
1. Abrir Wireshark
2. Seleccionar interfaz de rastreo (Ethernet o WiFi)
3. Iniciar captura y rastreo de paquetes
4. Abrir un explorador de internet el siguiente enlace:
   http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html
5. Detenga la captura de paquetes en Wireshark, una vez visualice el mensaje:
   “Congratulations! You've downloaded the first Wireshark lab file!”
6. Identificar los módulos de Wireshark (ver figura en la siguiente página)



Preguntas:

A. ¿Cuál es la IP de su computador (source/fuente)? **10.100.17.194**

Incluya impresión de pantalla de la IP registrada por Wireshark.



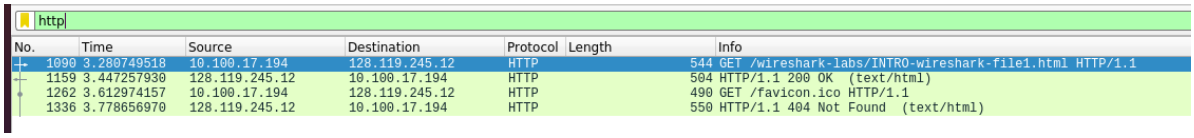
No.	Time	Source	Destination	Protocol	Length	Info
1090	3.280749518	10.100.17.194	128.119.245.12	HTTP		544 GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1
1159	3.447257930	128.119.245.12	10.100.17.194	HTTP		504 HTTP/1.1 200 OK (text/html)
1262	3.612974157	10.100.17.194	128.119.245.12	HTTP		490 GET /favicon.ico HTTP/1.1
1336	3.778656970	128.119.245.12	10.100.17.194	HTTP		550 HTTP/1.1 404 Not Found (text/html)

Incluya impresión de pantalla de la IP identificada por comandos en su sistema operativo.

```
(base) ➜  ~ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp2s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq_codel state DOWN group default qlen 1000
    link/ether 60:18:95:40:5e:3c brd ff:ff:ff:ff:ff:ff
3: wlp3s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 90:0f:0c:e6:a2:bb brd ff:ff:ff:ff:ff:ff
    inet 10.100.17.194/19 brd 10.100.31.255 scope global dynamic noprefixroute wlp3s0
       valid_lft 2005sec preferred_lft 2005sec
    inet6 fe80::efb3:fc7a:7b0:b3a8/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
4: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
    link/ether 52:54:00:37:e1:04 brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
       valid_lft forever preferred_lft forever
5: virbr0-nic: <BROADCAST,MULTICAST> mtu 1500 qdisc fq_codel master virbr0 state DOWN group default qlen 1000
    link/ether 52:54:00:37:e1:04 brd ff:ff:ff:ff:ff:ff
```

B. ¿Cuál es la IP del destinatario? **128.119.245.12**

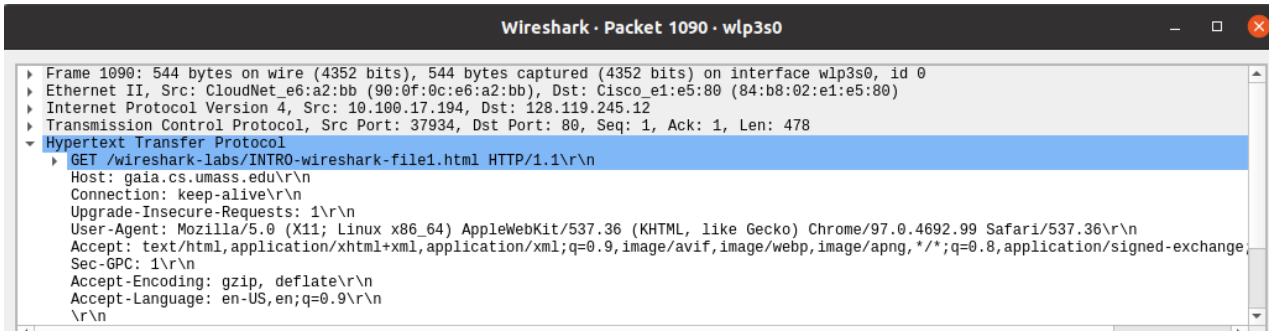
Incluya impresión de pantalla de la IP registrada por Wireshark.



No.	Time	Source	Destination	Protocol	Length	Info
1090	3.280749518	10.100.17.194	128.119.245.12	HTTP		544 GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1
1159	3.447257930	128.119.245.12	10.100.17.194	HTTP		504 HTTP/1.1 200 OK (text/html)
1262	3.612974157	10.100.17.194	128.119.245.12	HTTP		490 GET /favicon.ico HTTP/1.1
1336	3.778656970	128.119.245.12	10.100.17.194	HTTP		550 HTTP/1.1 404 Not Found (text/html)

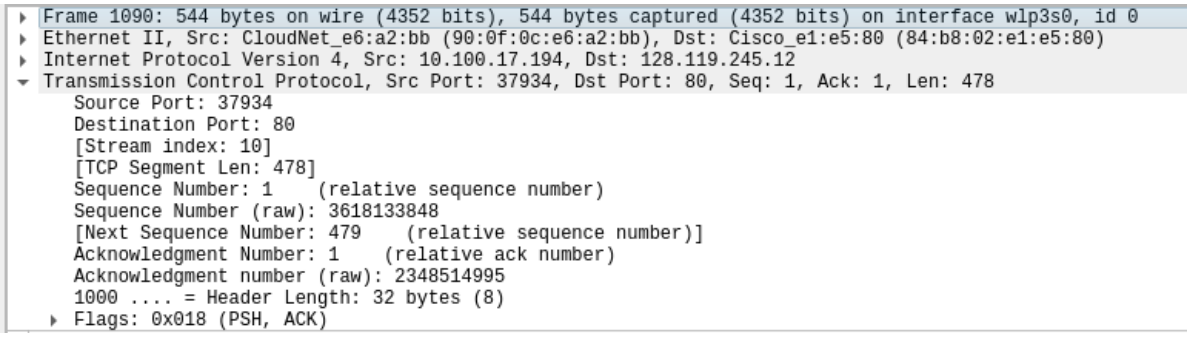
C. ¿Cuál es la longitud del paquete (solicitud HTTP GET) en bytes? **544 bytes.**

Incluya impresión de pantalla de la IP registrada por Wireshark.



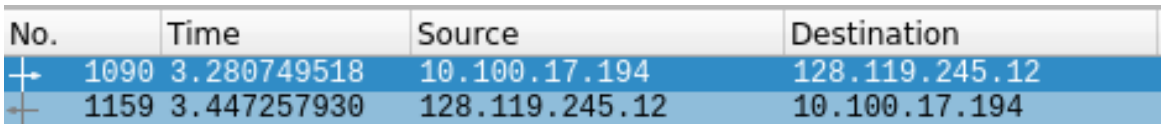
D. ¿Cuál es el puerto de conexión TCP de su computador? **37934**

Incluya impresión de pantalla de la IP registrada por Wireshark.

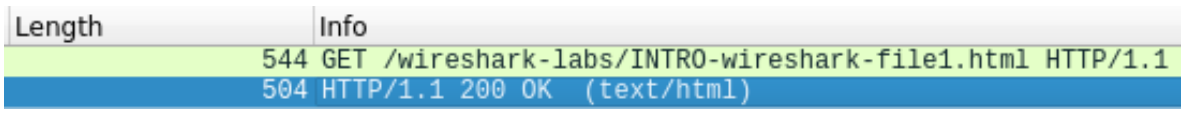


E. ¿Cuál es el puerto de conexión TCP del destinatario? **80**
Incluya impresión de pantalla de la IP registrada por Wireshark. (ver la imagen del punto D)

F. ¿Cuánto es el retardo de transmisión entre ambos hosts de acuerdo a las mediciones de tiempo de Wireshark? (3.44s - 3.28s) **0.16 segundos**
Incluya impresión de pantalla de la IP registrada por Wireshark.



G. ¿Cuál es la longitud del paquete (respuesta HTTP OK) en bytes? **504 bytes.**
Incluya impresión de pantalla de la IP registrada por Wireshark.



H. Construya una lista de todos los protocolos capturados en la actividad (i.e TCP, HTTP, UDP, entre otros. Ver la columna de protocolos en Wireshark).

- HTTP
- TCP
- DNS
- NBNS
- DB-LSP-DISC/JSON
- TLSv1.2
- UDP

Entregables: Una carpeta comprimida con los siguientes elementos

Actividad 1 (100%):

• Documento con respuestas y tomas de pantalla incluyendo los detalles del paquete HTTP GET y HTTP OK (60%)
• Archivo de Wireshark con la captura de paquetes (40%).