

**Universidad de los Andes**  
**Departamento de Ingeniería de Sistemas**



**Laboratorio #2: Análisis De Protocolos De La Capa  
De Aplicación**

**ISIS3204 - Infraestructura de Comunicaciones**

**Grupo 3:**

**Juan Esteban Quiroga - 202013216**  
**Juan Manuel Rodriguez - 202013372**  
**Andres Felipe Ortiz - 201727662**

**2025-10**

# Contents

<b>Introducción</b>	<b>3</b>
<b>8.1 Prueba ping</b>	<b>3</b>
8.1.1 Prueba de conectividad al servidor DNS	3
8.1.2 Prueba de conectividad al servidor FTP	4
<b>8.2 Análisis de tráfico del Servicio DNS</b>	<b>5</b>
8.2.1 Prueba de conectividad al Servidor Web (IP)	5
8.2.2 Prueba de conectividad al Servidor Web (URL)	6
<b>8.3 Análisis de tráfico del Servicio FTP</b>	<b>6</b>
8.3.1 Conexión al servidor FTP	6
8.3.2 Descarga de archivo (Download)	7
8.3.3 Carga de archivo (Upload)	7
<b>8.4 Análisis de tráfico del Servicio Web</b>	<b>8</b>
8.4.1 Acceso al servidor web mediante HTTP	8
<b>8.5 Análisis del protocolo HTTPS realizando navegación en el sitio de YouTube</b>	<b>9</b>
8.5.1 Navegación en YouTube	9
8.5.2 Navegación en otros sitios HTTPS	9
8.5.2.1 <a href="https://www.elspectador.com">https://www.elspectador.com</a>	9
8.5.2.2 <a href="https://www.eltiempo.com">https://www.eltiempo.com</a>	9
8.5.2.3 <a href="https://www.uniandes.edu.co">https://www.uniandes.edu.co</a>	9
8.5.2.4 <a href="https://www.bancolombia.com">https://www.bancolombia.com</a>	9
<b>8.6 Análisis del protocolo VoIP</b>	<b>9</b>
8.6.1 Establecimiento de la llamada	9
<b>8.7 Análisis del protocolo RTMP</b>	<b>9</b>
8.7.1 Inicio de la transmisión	9
<b>9.1 Topología</b>	<b>9</b>

# Introducción

En este laboratorio configuramos y probamos algunos servicios de red: DNS, HTTP/HTTPS, FTP, VoIP y RTMP, usando una topología a pequeña escala. Con Wireshark analizamos la conectividad de estos servicios en dicha red. Este laboratorio nos ayudó a entender la interacción de protocolos importantes que soportan la comunicación en redes de computadores. En la figura 1 mostramos las IPs estáticas que usamos para este laboratorio. (Es importante que estas IPs no sean dinamicas por DHCP ya que esto asegura direccionamiento constante y facilidad de configuración stable en la red.) La asignación de direcciones inició en 172.20.10.5 porque el gateway de la red es el hotspot del iPhone en 172.20.10.1, dejando libres las direcciones previas para posibles equipos de infraestructura y evitando conflictos.

✓	VM (Lab 2 role)	Static IP	Notes
	BaseServer	172.20.10.4/28	Baseline
	server1-web	172.20.10.5/28	Web server
	server2-ftp	172.20.10.6/28	FTP server
	server3-voip	172.20.10.7/28	VoIP server
	server4-dns	172.20.10.8/28	DNS server
	server5-rtmp	172.20.10.9/28	RTMP server

Figure 1: Configuración de IPs estáticas

## 8.1 Prueba ping

En esta sección se verifica la conexión básica entre el cliente y los servidores DNS y FTP mediante pings con el protocolo ICMP, asegurando la conectividad antes de analizar los demás protocolos.

### 8.1.1 Prueba de conectividad al servidor DNS

Desde el cliente se enviaron pings (echo requests ICMP ) al servidor DNS utilizando su dirección IP (172.20.10.8). El tráfico generado se capturó y se guardó en el archivo [Ping\\_DNS\\_IP.pcap](#). El archivo fue abierto en Wireshark y se aplicó el filtro icmp para observar únicamente los paquetes de ping. Se registraron la dirección IP de origen, dirección IP de destino, dirección MAC de origen y dirección MAC de destino en las tramas capturadas.

Time	172.20.10.2	172.20.10.8	Comment
0.000000		Echo (ping) request id=0x0fb1, seq=1/256, ttl=64	ICMP: Echo (ping) request id=0x0fb1, seq=1/256, ttl=64
0.000307		Echo (ping) reply id=0x0fb1, seq=1/256, ttl=64	ICMP: Echo (ping) reply id=0x0fb1, seq=1/256, ttl=64
0.930467		Echo (ping) request id=0x0fb1, seq=2/512, ttl=64	ICMP: Echo (ping) request id=0x0fb1, seq=2/512, ttl=64
0.930934		Echo (ping) reply id=0x0fb1, seq=2/512, ttl=64	ICMP: Echo (ping) reply id=0x0fb1, seq=2/512, ttl=64
2.537523		Echo (ping) request id=0x0fb1, seq=3/768, ttl=64	ICMP: Echo (ping) request id=0x0fb1, seq=3/768, ttl=64
2.538032		Echo (ping) reply id=0x0fb1, seq=3/768, ttl=64	ICMP: Echo (ping) reply id=0x0fb1, seq=3/768, ttl=64
3.067381		Echo (ping) request id=0x0fb1, seq=4/1024, ttl=64	ICMP: Echo (ping) request id=0x0fb1, seq=4/1024, ttl=64
3.067859		Echo (ping) reply id=0x0fb1, seq=4/1024, ttl=64	ICMP: Echo (ping) reply id=0x0fb1, seq=4/1024, ttl=64
5.824435		Echo (ping) request id=0x0fb1, seq=6/1536, ttl=64	ICMP: Echo (ping) request id=0x0fb1, seq=6/1536, ttl=64
5.824706		Echo (ping) reply id=0x0fb1, seq=6/1536, ttl=64	ICMP: Echo (ping) reply id=0x0fb1, seq=6/1536, ttl=64
7.576445		Echo (ping) request id=0x0fb1, seq=8/2048, ttl=64	ICMP: Echo (ping) request id=0x0fb1, seq=8/2048, ttl=64
7.576909		Echo (ping) reply id=0x0fb1, seq=8/2048, ttl=64	ICMP: Echo (ping) reply id=0x0fb1, seq=8/2048, ttl=64
8.544390		Echo (ping) request id=0x0fb1, seq=9/2304, ttl=64	ICMP: Echo (ping) request id=0x0fb1, seq=9/2304, ttl=64
8.544830		Echo (ping) reply id=0x0fb1, seq=9/2304, ttl=64	ICMP: Echo (ping) reply id=0x0fb1, seq=9/2304, ttl=64

Figure 2: Flujo de packets en DNS ping

```

> Frame 18: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
> Ethernet II, Src: Apple_16:40:75 (28:cf:e9:16:40:75), Dst: VMware_7b:b3:c5 (00:0c:29:7b:b3:c5)
> Internet Protocol Version 4, Src: 172.20.10.2 (172.20.10.2), Dst: 172.20.10.8 (172.20.10.8)
> Internet Control Message Protocol

```

Figure 3: Evidencia de IPs y MACs origen/destino

#### 8.1 Prueba ping (DNS)

Direccionamiento IP			Direccionamiento MAC		
Tipo de Paquete	Origen	Destino	Tipo de Paquete	Origen	Destino
Request	172.20.10.2	172.20.10.8	Request	28:CF:E9:16:40:75	00:0C:29:7B:B3:C5
Reply	172.20.10.8	172.20.10.2	Reply	00:0C:29:7B:B3:C5	28:CF:E9:16:40:75

Figure 4: Tabla de IPs y MACs origen/destino

### 8.1.2 Prueba de conectividad al servidor FTP

Desde el cliente se enviaron pings al servidor FTP utilizando su dirección IP. El tráfico generado se capturó y se guardó en el archivo [Ping\\_FTP\\_IP.pcap](#) El archivo fue analizado en Wireshark con el filtro icmp. Se identificaron las direcciones IP y MAC correspondientes a los paquetes de solicitud y respuesta.

Time	172.20.10.2	172.20.10.6	Comment
0.000000		Echo (ping) request id=0x0fc4, seq=1/256...	ICMP: Echo (ping) request id=0x0fc4, seq=1/256.
0.000477		Echo (ping) reply id=0x0fc4, seq=1/256, tt...	ICMP: Echo (ping) reply id=0x0fc4, seq=1/256, .
1.052587		Echo (ping) request id=0x0fc4, seq=2/512...	ICMP: Echo (ping) request id=0x0fc4, seq=2/512.
1.053065		Echo (ping) reply id=0x0fc4, seq=2/512, tt...	ICMP: Echo (ping) reply id=0x0fc4, seq=2/512, ..
12.008779		Echo (ping) request id=0x0fc4, seq=11/281...	ICMP: Echo (ping) request id=0x0fc4, seq=11/28..
12.008781		Echo (ping) request id=0x0fc4, seq=12/307...	ICMP: Echo (ping) request id=0x0fc4, seq=12/30.
12.009323		Echo (ping) reply id=0x0fc4, seq=11/2816...	ICMP: Echo (ping) reply id=0x0fc4, seq=11/281...
12.009368		Echo (ping) reply id=0x0fc4, seq=12/3072...	ICMP: Echo (ping) reply id=0x0fc4, seq=12/307..
14.659701		Echo (ping) request id=0x0fc4, seq=15/384...	ICMP: Echo (ping) request id=0x0fc4, seq=15/38.
14.660001		Echo (ping) reply id=0x0fc4, seq=15/3840...	ICMP: Echo (ping) reply id=0x0fc4, seq=15/384.
26.021105		Echo (ping) request id=0x0fc4, seq=26/665...	ICMP: Echo (ping) request id=0x0fc4, seq=26/6...
26.021603		Echo (ping) reply id=0x0fc4, seq=26/6656...	ICMP: Echo (ping) reply id=0x0fc4, seq=26/66...
26.950529		Echo (ping) request id=0x0fc4, seq=27/691...	ICMP: Echo (ping) request id=0x0fc4, seq=27/6...
26.951204		Echo (ping) reply id=0x0fc4, seq=27/6912...	ICMP: Echo (ping) reply id=0x0fc4, seq=27/691.

Figure 5: Flujo de packets en FTP ping

```

> Frame 52: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
> Ethernet II, Src: Apple_16:40:75 (28:cf:e9:16:40:75), Dst: VMware_30:c7:64 (00:50:56:30:c7:64)
> Internet Protocol Version 4, Src: 172.20.10.2 (172.20.10.2), Dst: 172.20.10.6 (172.20.10.6)
> Internet Control Message Protocol

```

Figure 6: Evidencia de IPs y MACs origen/destino

#### 8.1 Prueba ping (FTP)

Direccionamiento IP			Direccionamiento MAC		
Tipo de Paquete	Origen	Destino	Tipo de Paquete	Origen	Destino
Request	172.20.10.2	172.20.10.6	Request	28:CF:E9:16:40:75	00:50:56:30:C7:64
Reply	172.20.10.6	172.20.10.2	Reply	00:50:56:30:C7:64	28:CF:E9:16:40:75

Figure 7: Tabla de IPs y MACs origen/destino

⚠ Los controladores de Wi-Fi de macOS no permiten que software de terceros como VMware inyecte direcciones MAC arbitrarias en la tarjeta inalámbrica. Dentro de la máquina virtual, el sistema operativo invitado cree que tiene una tarjeta de red con su propia dirección MAC, pero cuando el paquete sale de la VM y llega a la tarjeta Wi-Fi del MacBook, VMware Fusion la reemplaza por la MAC de la Wi-Fi del MacBook. **[36:08:7F:70:CA:40]**

Para evitar esta limitación, ejecuté el siguiente comando en cada VM e importé los resultados en Wireshark:

```
# En el servidor DNS
sudo tcpdump -i ens33 -nn -e -2 capture-dns-vm.pcap

# En el servidor FTP
sudo tcpdump -i ens33 -nn -e -2 capture-ftp-vm.pcap
```

Por eso existen 2 archivos **.pcap** adicionales para **Ping\_DNS(VM)** y **Ping\_FTP(VM)** ya a que en los archivos sin el sufijo (VM) aparece la misma MAC del MacBook **[36:08:7F:70:CA:40]** como destino en ambas pruebas!

## 8.2 Análisis de tráfico del Servicio DNS

En esta sección se analiza el servicio DNS, que traduce nombres de dominio en direcciones IP para facilitar la comunicación en la red. Se genera y captura tráfico con Wireshark, identificando consultas y respuestas, así como el protocolo de transporte y los puertos utilizados al acceder a un servidor web por IP y por nombre de dominio.

Cuando un cliente necesita comunicarse con un dominio, primero envía al servidor DNS una consulta de tipo *A* para obtener su dirección IPv4 y, en paralelo, una consulta de tipo *AAAA* para la dirección IPv6. El servidor responde con los registros correspondientes, que el cliente almacena en caché. Con la IP resuelta, el cliente ya puede establecer la comunicación (ej. enviar un ping) directamente al servidor destino.

### 8.2.1 Prueba de conectividad al Servidor Web (IP)

Cuando se accedió al servidor escribiendo directamente su dirección IP en el ping request, la conexión se estableció de inmediato ya que no fue necesario consultar al DNS y en la captura se observó únicamente tráfico ICMP entre cliente y servidor. El tráfico generado se capturó y se guardó en el archivo **Ping\_WEB\_IP.pcap** El archivo fue analizado en Wireshark con el filtro icmp.

Time	172.20.10.2	172.20.10.5	Comment
8.761403		Echo (ping) request id=0x475c, seq=1/256, ...	ICMP: Echo (ping) request id=0x475c, seq=1/256...
8.761705		Echo (ping) reply id=0x475c, seq=1/256, tt...	ICMP: Echo (ping) reply id=0x475c, seq=1/256...
10.819604		Echo (ping) request id=0x475c, seq=3/768, ...	ICMP: Echo (ping) request id=0x475c, seq=3/768...
10.819952		Echo (ping) reply id=0x475c, seq=3/768, t...	ICMP: Echo (ping) reply id=0x475c, seq=3/768...
12.873814		Echo (ping) request id=0x475c, seq=5/1280, ...	ICMP: Echo (ping) request id=0x475c, seq=5/128...
12.874236		Echo (ping) reply id=0x475c, seq=5/1280, ...	ICMP: Echo (ping) reply id=0x475c, seq=5/128...
13.910355		Echo (ping) request id=0x475c, seq=6/1536, ...	ICMP: Echo (ping) request id=0x475c, seq=6/15...
13.910696		Echo (ping) reply id=0x475c, seq=6/1536, ...	ICMP: Echo (ping) reply id=0x475c, seq=6/153...

Figure 8: Flujo de packets en WEB IP ping

```

> Frame 4: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
> Ethernet II, Src: Apple_16:40:75 (28:cf:e9:16:40:75), Dst: 36:08:7f:70:ca:40 (36:08:7f:70:ca:40)
> Internet Protocol Version 4, Src: 172.20.10.2 (172.20.10.2), Dst: 172.20.10.5 (172.20.10.5)
> Internet Control Message Protocol

```

Figure 9: Evidencia de IPs, MACs, y puertos origen/destino

En esta prueba, no hay resolución DNS.

### 8.2.2 Prueba de conectividad al Servidor Web (URL)

Cuando se accedió al servidor utilizando su nombre de dominio en la solicitud de ping, el cliente primero realizó consultas de tipo A y AAAA al servidor DNS para obtener la dirección IP correspondiente. Una vez resuelta, se estableció la comunicación con el servidor y en la captura se observó inicialmente el tráfico DNS seguido por el intercambio ICMP entre cliente y servidor. El tráfico generado se capturó y se guardó en el archivo [Ping\\_WEB.pcap](#), el cual fue analizado en Wireshark aplicando filtros para DNS e ICMP.

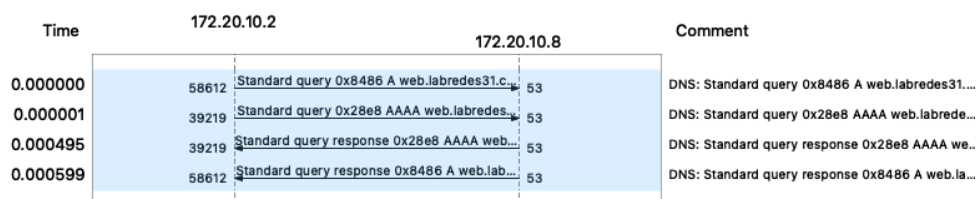


Figure 10: Flujo de packets en WEB Domain ping

```

> Frame 1: 89 bytes on wire (712 bits), 89 bytes captured (712 bits)
> Ethernet II, Src: Apple_16:40:75 (28:cf:e9:16:40:75), Dst: 36:08:7f:70:ca:40 (36:08:7f:70:ca:40)
> Internet Protocol Version 4, Src: 172.20.10.2 (172.20.10.2), Dst: 172.20.10.8 (172.20.10.8)
> User Datagram Protocol, Src Port: 58612 (58612), Dst Port: domain (53)
> Domain Name System (query)

```

Figure 11: Evidencia de IPs, MACs, y puertos origen/destino

#### 8.2 Prueba ping (WEB-IPv4)

Direccionamiento IP		
Tipo de Paquete	Origen	Destino
Request	172.20.10.2	172.20.10.5
Reply	172.20.10.5	172.20.10.2

Direccionamiento MAC		
Tipo de Paquete	Origen	Destino
Request	28:CF:E9:16:40:75	36:08:7F:70:CA:40
Reply	36:08:7F:70:CA:40	28:CF:E9:16:40:75

Puerto		
Tipo de Paquete	Origen	Destino
Request	58612	53
Reply	53	58612

Figure 12: Tabla de IPs, MACs, y puertos origen/destino

## 8.3 Análisis de tráfico del Servicio FTP

En esta sección se verifica el correcto funcionamiento del servicio FTP realizando una sesión autenticada desde el cliente para descargar y subir un archivo, capturando cada fase en los archivos [FTP\\_download.pcap](#) y [FTP\\_upload.pcap](#). Usando Wireshark, se filtra el tráfico FTP para examinar los intercambios de control y de datos, y posteriormente se documentan los detalles de la capa de aplicación, el protocolo de transporte utilizado, y los puertos involucrados.

### 8.3.1 Conexión al servidor FTP

El servidor responde primero con el mensaje de bienvenida. El cliente intenta establecer una sesión segura con AUTH TLS/SSL, pero el servidor lo rechaza con código **530** y solicita autenticación clásica (Esto se debe a que inicialmente configuré el servidor sde forma segura con vsftpd pero deshabilité la seguridad para poder observar bien el protocolo FTP). Luego el cliente envía **USER hermione** y **PASS test123**, a lo que el servidor responde con **230 Login successful**, confirmando el acceso. Luego, se ejecutan otros

comandos donde el servidor lista sus funcionalidades soportadas (PASV, EPSV, MDTM, etc.). Todo este intercambio ocurre sobre TCP puerto 21 en el canal de control.

Time	ftp.labredes31.com	172.20.10.2	Comment
28.547238	21	Response: 220 Welcome!!Thhis is the Unian...	FTP: Response: 220 Welcome!!Thhis is the Unian...
28.565064	21	Request: AUTH TLS	FTP: Request: AUTH TLS
28.565591	21	Response: 530 Please login with USER and ...	FTP: Response: 530 Please login with USER and ..
28.589602	21	Request: AUTH SSL	FTP: Request: AUTH SSL
28.590154	21	Response: 530 Please login with USER and ...	FTP: Response: 530 Please login with USER and ..
35.812933	21	Request: USER hermione	FTP: Request: USER hermione
35.813294	21	Response: 331 Please specify the password.	FTP: Response: 331 Please specify the password.
35.846624	21	Request: PASS test123	FTP: Request: PASS test123
35.899021	21	Response: 230 Login successful.	FTP: Response: 230 Login successful.
35.911268	21	Request: SYST	FTP: Request: SYST
35.911653	21	Response: 215 UNIX Type: L8	FTP: Response: 215 UNIX Type: L8
35.921806	21	Request: FEAT	FTP: Request: FEAT
35.922207	21	Response: 211-Features:	FTP: Response: 211-Features:
35.922248	21	Response: EPRT	FTP: Response: EPRT
35.922357	21	Response: EPSV	FTP: Response: EPSV
35.922439	21	Response: MDTM	FTP: Response: MDTM
35.922507	21	Response: PASV	FTP: Response: PASV
35.922575	21	Response: REST STREAM	FTP: Response: REST STREAM
35.922649	21	Response: SIZE	FTP: Response: SIZE
35.922739	21	Response: TVFS	FTP: Response: TVFS
35.922824	21	Response: 211 End	FTP: Response: 211 End
35.962356	21	[TCP Fast Retransmission] Response: 211-F...	FTP: [TCP Fast Retransmission] Response: 211-F...

Figure 13: Inicio de sesión FTP

### 8.3.2 Descarga de archivo (Download)

En esta parte se observa la navegación y transferencia de un archivo en FTP. Tras el mensaje **230 Login successful**, el cliente cambia al directorio **/files** con **CWD** y el servidor confirma con **250**. Luego el cliente consulta el directorio actual y solicita modo pasivo con **8.3-FTP-download-flowPASV**. El servidor responde con la dirección y puerto a usar (**227 Entering Passive Mode**). El cliente pide descargar el archivo **a.txt** con **RETR a.txt**, el servidor abre la conexión de datos (**150 Opening data connection**) y confirma la transferencia exitosa de este archivo con **226 Transfer complete**.

Time	ftp.labredes31.com	172.20.10.2	Comment
44.602779	21	Response: 230 Login successful.	FTP: Response: 230 Login successful.
44.837013	21	Request: CWD /files	FTP: Request: CWD /files
44.837590	21	Response: 250 Directory successfully chang...	FTP: Response: 250 Directory successfully chan...
45.070873	21	Request: PWD	FTP: Request: PWD
45.071307	21	Response: 257 "/files" is the current directory	FTP: Response: 257 "/files" is the current director
45.076366	21	Request: TYPE A	FTP: Request: TYPE A
45.076953	21	Response: 200 Switching to ASCII mode.	FTP: Response: 200 Switching to ASCII mode.
45.089291	21	Request: PASV	FTP: Request: PASV
45.090593	21	Response: 227 Entering Passive Mode (172,...	FTP: Response: 227 Entering Passive Mode (172,...
45.107844	21	Request: RETR a.txt	FTP: Request: RETR a.txt
45.117497	21	Response: 150 Opening BINARY mode data ..	FTP: Response: 150 Opening BINARY mode data ..
45.145126	21	Response: 226 Transfer complete.	FTP: Response: 226 Transfer complete.

Figure 14: Descarga de archivo por usuario "hermione"

### 8.3.3 Carga de archivo (Upload)

Después del inicio de sesión exitoso (**230 Login successful**), el cliente cambia al directorio **/files** con **CWD /files**, confirmado por el servidor con **250 Directory successfully changed**. Luego verifica la ubicación con



**PWD**, y el servidor responde con **257 "/files"**. El cliente ajusta el modo de transferencia a ASCII con **TYPE A**, y el servidor responde **200 Switching to ASCII mode**. Con el comando **PASV** se abre un canal de datos en modo pasivo, indicado por la respuesta **227 Entering Passive Mode**. Al final del proceso, el cliente solicita hacer upload del archivo **b.txt** con **STOR b.txt** y el servidor responde **150 Ok to send data** y, al finalizar la transferencia, confirma con **226 Transfer complete**.

En estas capturas no mostramos el hecho de que tenemos dos usuarios: "harry" y "hermione". Los archivos de "hermione" no son visibles desde POV del usuario de "harry" y vice versa. (Esto lo configuramos en `/etc/vsftpd.conf`)

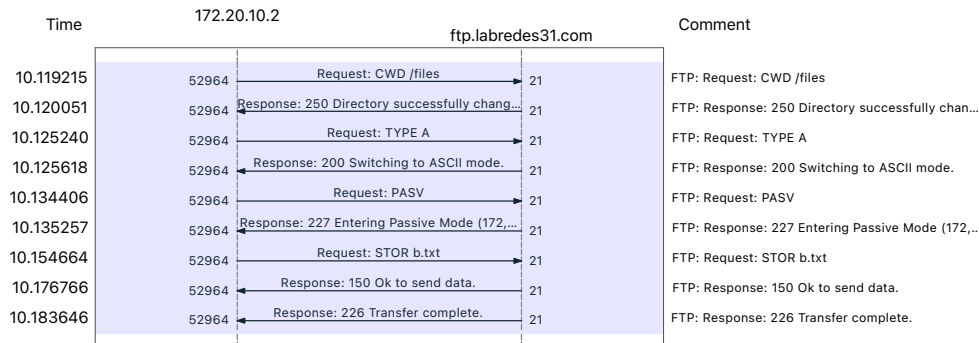


Figure 15: Descarga de archivo por usuario "hermione"

## 8.4 Análisis de tráfico del Servicio Web

En esta sección se analiza el funcionamiento del protocolo HTTP dentro de la topología configurada. Mediante capturas en Wireshark, se observan las peticiones y respuestas entre el cliente y el servidor web en texto claro, lo que permite identificar directamente los encabezados y contenidos intercambiados, así como los puertos y el protocolo de transporte utilizados en la comunicación. El tráfico generado se capturó y se guardó en el archivo `Ping_WEB_view.pcap`

### 8.4.1 Acceso al servidor web mediante HTTP

En esta captura se observa primero la resolución DNS de `web.labredes31.com`. A continuación, el cliente establece una conexión TCP con el servidor en el puerto 80, completando el three-way handshake. Luego, el cliente envía una petición **GET / HTTP/1.1** y el servidor responde con **HTTP/1.1 200 OK**, entregando la página en plaintext. Luego el cliente solicita el recurso `/favicon.ico`, que también recibe una respuesta satisfactoria **200 OK**. Finalmente, la comunicación se cierra correctamente mediante el intercambio de mensajes **FIN, ACK**, completando así el ciclo típico de una sesión HTTP.

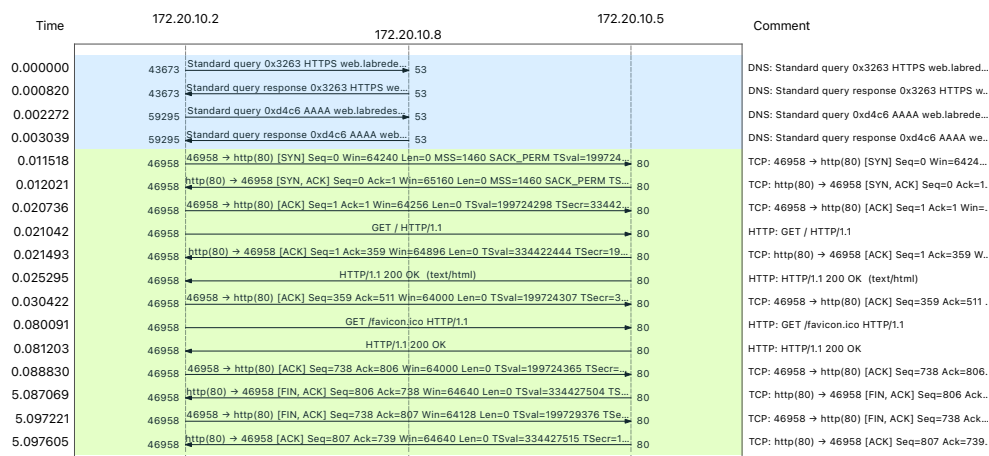


Figure 16: Flow de HTTP



## 8.5 Análisis del protocolo HTTPS realizando navegación en el sitio de YouTube

### 8.5.1 Navegación en YouTube

### 8.5.2 Navegación en otros sitios HTTPS

8.5.2.1 <https://www.elspectador.com>

8.5.2.2 <https://www.eltiempo.com>

8.5.2.3 <https://www.uniandes.edu.co>

8.5.2.4 <https://www.bancolombia.com>

## 8.6 Análisis del protocolo VoIP

### 8.6.1 Establecimiento de la llamada

## 8.7 Análisis del protocolo RTMP

### 8.7.1 Inicio de la transmisión

## 9.1 Topología

## References

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