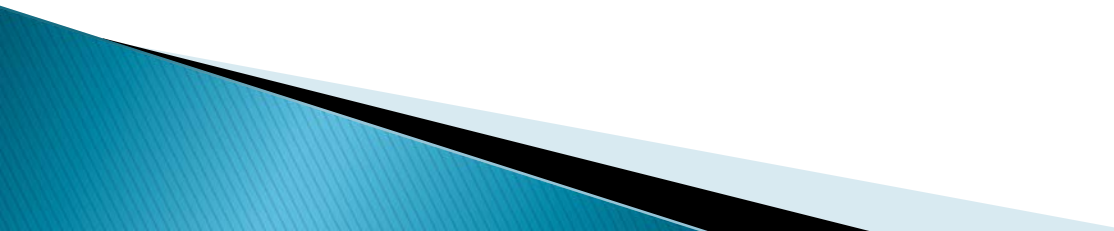


Clase de laboratorio de Redes y Sistemas Distribuidos

Analizadores de protocolos

Contenido

- ▶ Analizadores de protocolos
 - ▶ Wireshark:
 - Elección del interfaz y la captura
 - Filtrado
 - Análisis de un paquete
 - Otras opciones de visualización
 - Estadísticas
 - Exportación
- 

Analizadores de protocolos

- ▶ Un analizador de protocolos es un programa informático o una pieza de hardware que puede interceptar y registrar tráfico que pasa por una red o parte de una red.
 - Analizador de red, analizador de paquetes o *sniffer*
- ▶ Gracias a que el medio de transmisión es compartido por varios ordenadores, estos programas pueden capturar información que no va destinada a ellos.
 - Modo promiscuo
 - El uso de este modo depende mucho del sistema operativo usado, driver, interfaz de red, ...

Analizadores de protocolos


- ▶ Estos programas tienen muchísimas aplicaciones, algunas de ellas delictivas.
 - Análisis de fallos para descubrir problemas en la red
 - Análisis de cuellos de botella
 - Monitorización de utilización del ancho de banda
 - Monitorización de la información en movimiento
 - Depuración de protocolos de red
 - Depuración de aplicaciones cliente-servidor
 - Detección de intrusos en la red
 - Espiar a los usuarios de la red y recolectar información confidencial como contraseñas
 - ...

Analizadores de protocolos

- ▶ Algunos analizadores de protocolos:
 - **Etercap**: Permite detectar ataques man-in-the-middle.
 - **Kismet**: Permite detectar intrusiones en redes inalámbricas.
 - **TCPDUMP**: Herramienta de línea de comandos, que permite capturar en tiempo real el tráfico que pasa por la red.
 - **Wireshark**: Es un analizador de protocolos de uso académico con la funcionalidad estándar de un analizador de protocolos.
 - ...



Wireshark

- ▶ Analizador de protocolos gratuito.
 - ▶ Disponible para la mayoría de los sistemas operativos actuales.
 - ▶ Descarga: <http://www.wireshark.org/>
 - ▶ Durante esta clase (y las próximas) aprenderemos las principales funcionalidades ofrecidas por esta herramienta.
- 

Wireshark: Inicio

The screenshot shows the Wireshark Network Analyzer interface. The 'Capture Interfaces' dialog box is open, displaying a list of available network interfaces. The 'Ethernet' interface is selected, and the 'Promiscuous' checkbox is checked. The 'Capture filter for selected interfaces' field is empty. The 'Start' button is highlighted.

Interface	Traffic	Link-layer Header	Promi:	Snaplen	Buffer (B)	Monitor	Capture Filter
> Ethernet		Ethernet	<input checked="" type="checkbox"/>	default	2	—	
> VirtualBox Host-Only Network #2		Ethernet	<input checked="" type="checkbox"/>	default	2	—	
> VMware Network Adapter VMnet4		Ethernet	<input checked="" type="checkbox"/>	default	2	—	
> VMware Network Adapter VMnet3		Ethernet	<input checked="" type="checkbox"/>	default	2	—	
> Ethernet 2		Ethernet	<input checked="" type="checkbox"/>	default	2	—	
> VMware Network Adapter VMnet2		Ethernet	<input checked="" type="checkbox"/>	default	2	—	
> Ethernet 3		Ethernet	<input checked="" type="checkbox"/>	default	2	—	
> Npcap Loopback Adapter		BSD loopback	<input checked="" type="checkbox"/>	default	2	—	
USBPCap1		USBPCap	<input type="checkbox"/>	—	—	—	
USBPCap2		USBPCap	<input type="checkbox"/>	—	—	—	
USBPCap3		USBPCap	<input type="checkbox"/>	—	—	—	

Enable promiscuous mode on all interfaces ☒

Capture filter for selected interfaces:

Manage Interfaces... Compile BPFs

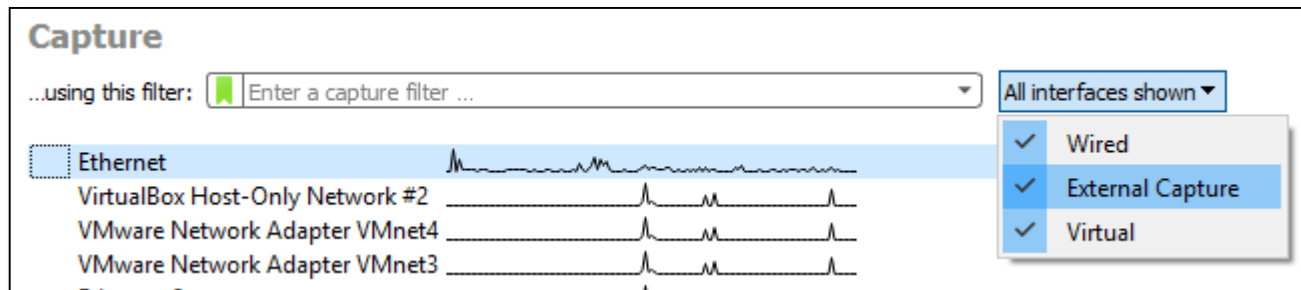
Start Cerrar Ayuda

Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists
You are running Wireshark 2.4.3 (v2.4.3-0-g368ba1ee37). You receive automatic updates.

Ready to load or capture No Packets Profile: Default

Wireshark: Elección de interfaz

- ▶ ¿Qué interfaz elegir?
 - Oculte los no deseados



- Información del SO (ifconfig, ipconfig, ...):

```
C:\Windows\System32>ipconfig

Configuración IP de Windows


Adaptador de Ethernet Ethernet:

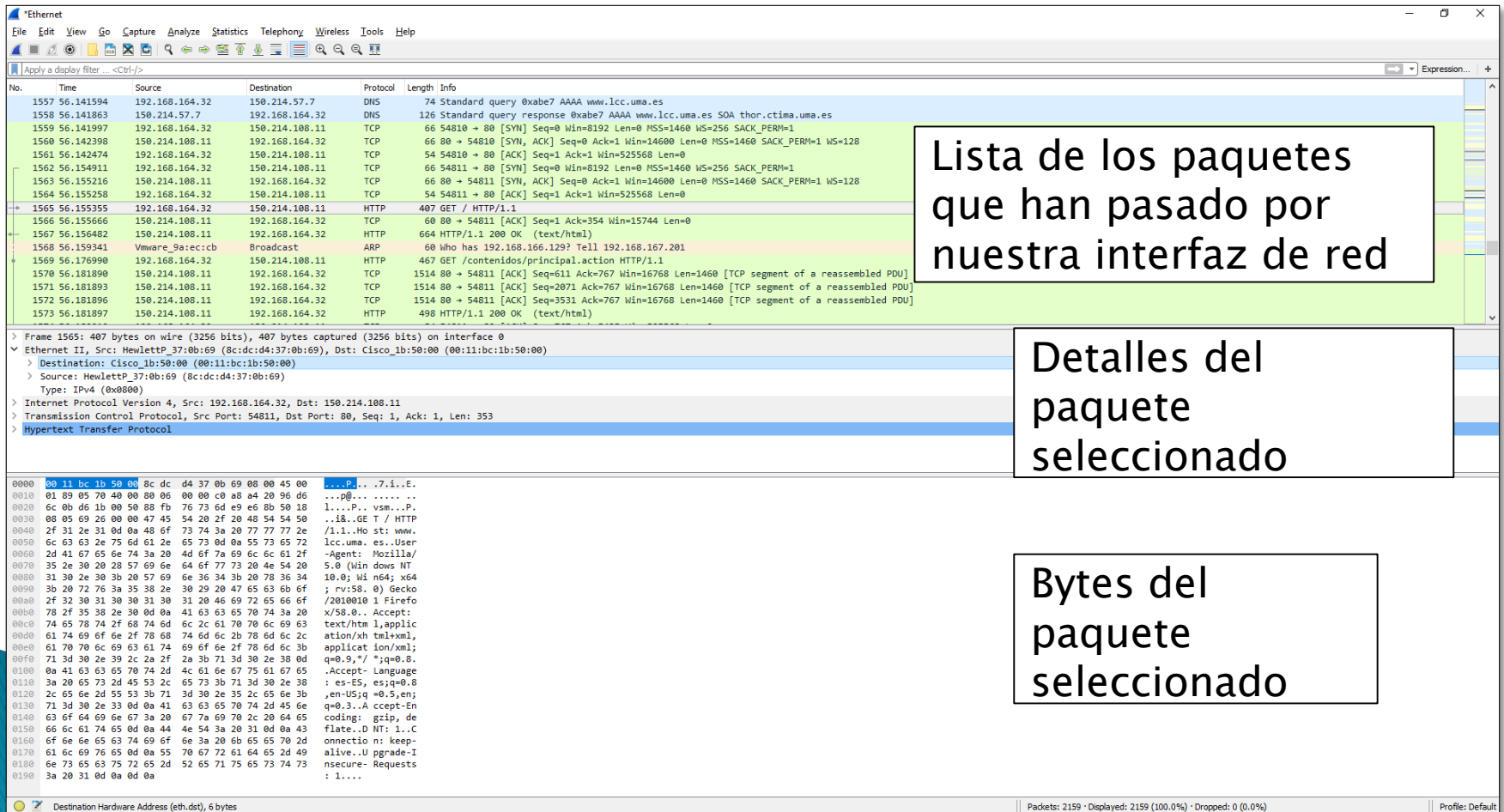
    Sufijo DNS específico para la conexión. . . :
    Dirección IPv4. . . . . : 192.168.164.32
    Máscara de subred . . . . . : 255.255.252.0
    Puerta de enlace predeterminada . . . . . : 192.168.167.254

Adaptador de Ethernet VirtualBox Host-Only Network #2:

    Sufijo DNS específico para la conexión. . . :
```


Wireshark: Ventana principal

- ▶ Tras elegir el interfaz empieza a capturar con y pare de capturar 



Lista de los paquetes que han pasado por nuestra interfaz de red

Detalles del paquete seleccionado

Bytes del paquete seleccionado

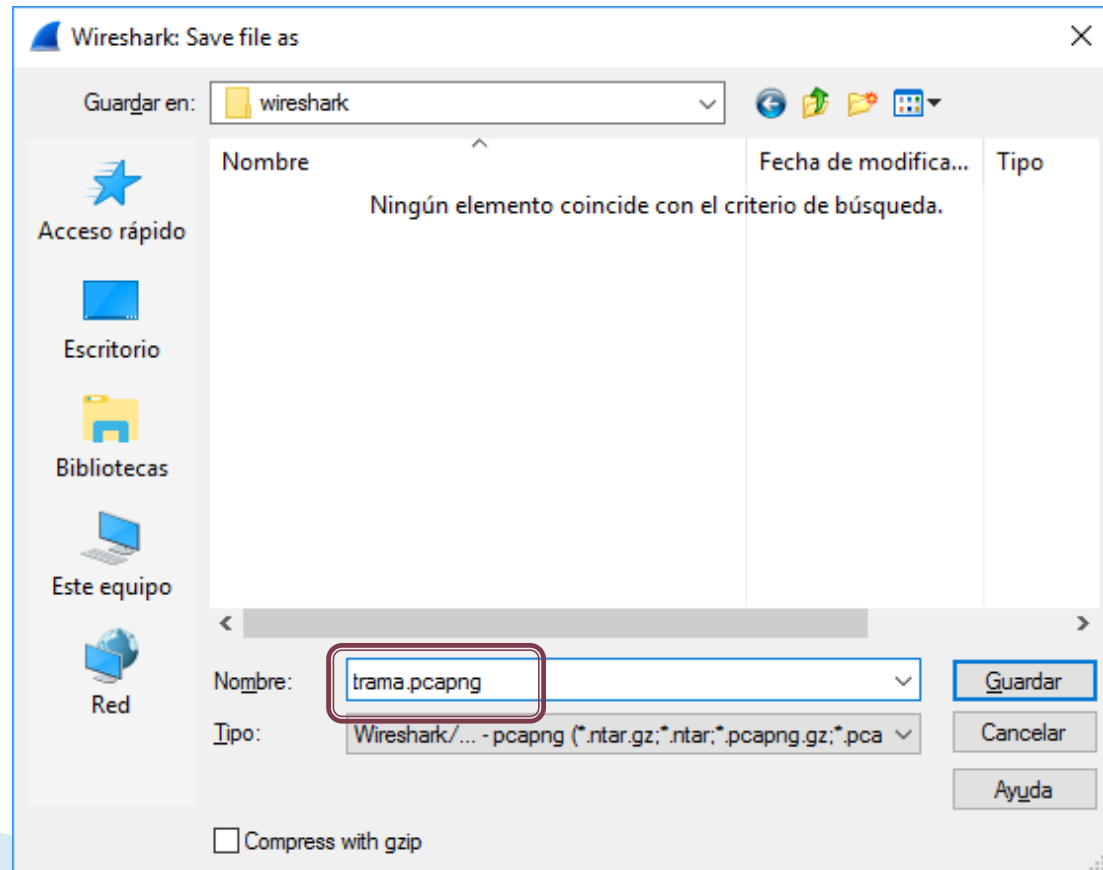
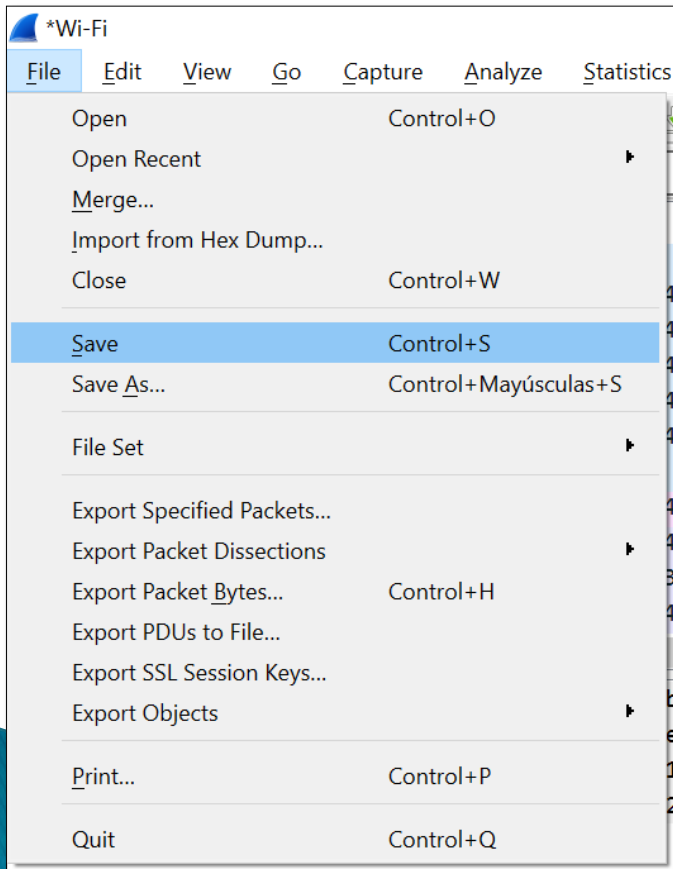
Destination Hardware Address (eth.dst), 6 bytes

Packets: 2159 · Displayed: 2159 (100.0%) · Dropped: 0 (0.0%)

Profile: Default

Wireshark: Guardado

- Una vez capturado el tráfico deseado se debe guardar (**en las prácticas hay que entregarlas**)



Wireshark: Filtrado

- ▶ Ajuste lo máximo posible el tiempo de captura al elemento a analizar:

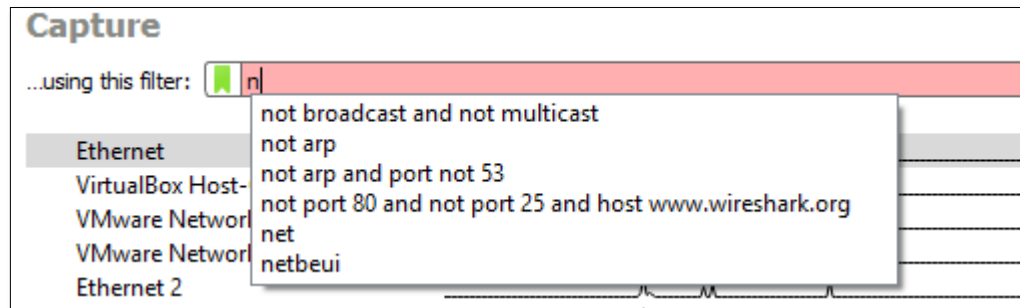
No.	Time
2159	79.194746

¡2159 paquetes en 1 min y 20 seg!

- ▶ También se puede filtrar:
 - Filtros de captura
 - Filtros de visualización

Wireshark: Filtrado

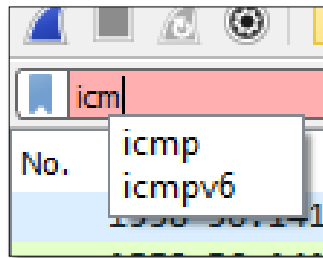
- ▶ Filtrado de captura:
 - Se indican antes de empezar la captura
 - El filtro indica los paquetes a capturar
 - Los paquetes que no cumplan la condición no los captura Wireshark



Wireshark te ofrece algunas
sugerencias de filtros habituales

Wireshark: Filtrado

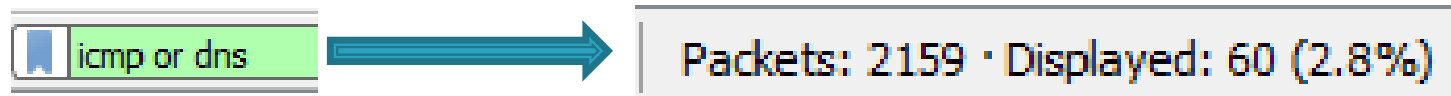
- ▶ Filtrado de visualización:
 - Se indican durante/tras la captura
 - El filtro indica los paquetes a visualizar
 - No elimina paquetes solo los oculta temporalmente
 - Los filtros no se guardan en el fichero de la traza
- ▶ Se indican en la barra superior
- ▶ El más básico es indicar el nombre de un protocolo:



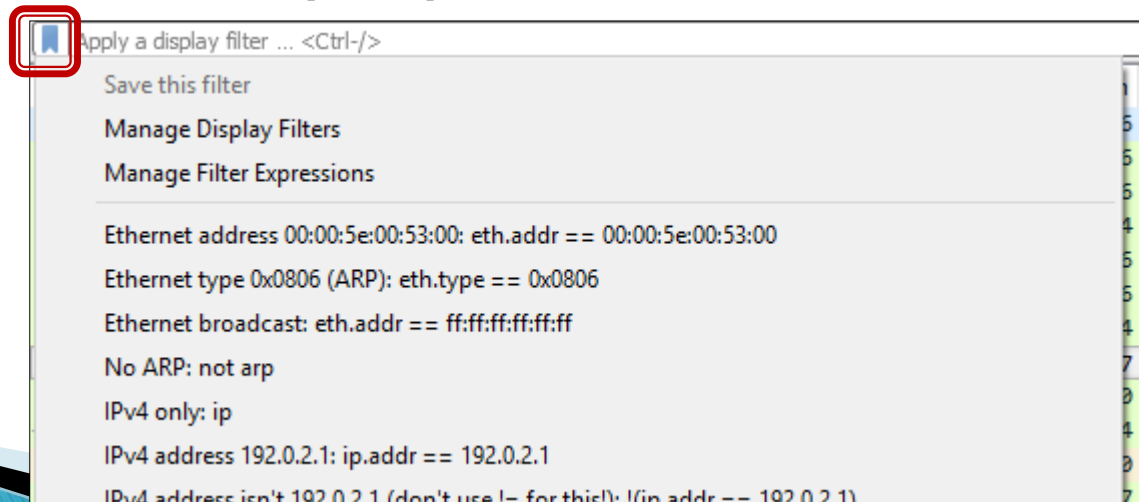
Wireshark ofrece propuestas conforme se va escribiendo

Wireshark: Filtrado

- ▶ Se pueden crear filtro más complejos con **and** y **or**

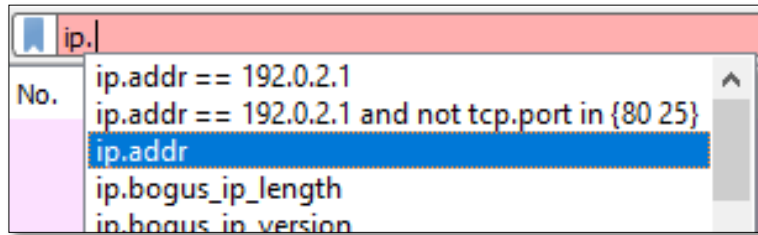


- ▶ Wireshark ofrece algunos muy usados (se pueden añadir propios):



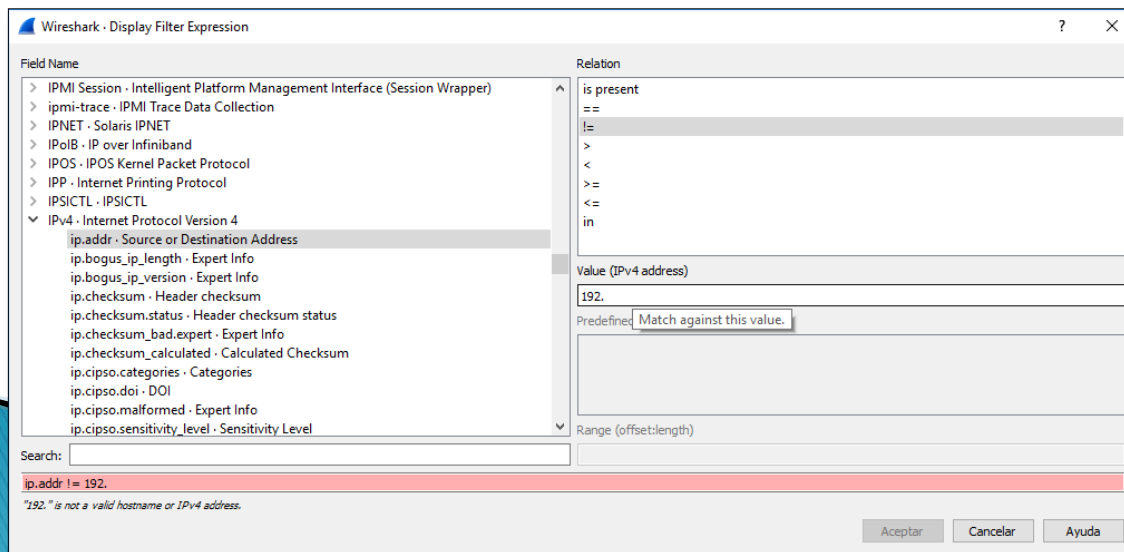
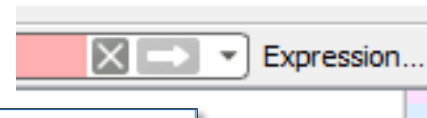
Wireshark: Filtrado

- ▶ Se pueden hacer filtros más específicos accediendo a campos de las cabeceras



Wireshark ofrece propuestas conforme se va escribiendo

- ▶ También trae un asistente:



Luego veremos otra forma de crear filtros

Wireshark: Datos de un paquete

- > Frame 46: 906 bytes on wire (7248 bits), 906 bytes captured (7248 bits) on interface 0
- > Ethernet II, Src: IntelCor_1f:99:a1 (68:07:15:1f:99:a1), Dst: Cisco_03:04:00 (00:1b:8f:03:04:00)
- > Internet Protocol Version 4, Src: 192.168.120.255, Dst: 150.214.40.97
- > Transmission Control Protocol, Src Port: 61434, Dst Port: 80, Seq: 2615667219, Ack: 429552808, Len: 852
- > Hypertext Transfer Protocol

Cab Eth

Cab IP

Cab TCP

Cab HTTP

- > Frame 46: 906 bytes on wire (7248 bits), 906 bytes captured (7248 bits) on interface 0

Toda la trama (+ resumen ofrecido por Wireshark)

- > Ethernet II, Src: IntelCor_1f:99:a1 (68:07:15:1f:99:a1), Dst: Cisco_03:04:00 (00:1b:8f:03:04:00)

Cabecera Ethernet II (Capa de Enlace)

- > Internet Protocol Version 4, Src: 192.168.120.255, Dst: 150.214.40.97

Cabecera IP (Capa de Red)

- > Transmission Control Protocol, Src Port: 61434, Dst Port: 80, Seq: 2615667219, Ack: 429552808, Len: 852

Cabecera TCP (Capa de Transporte)

- > Hypertext Transfer Protocol

Cabecera HTTP (Capa de Aplicación)

Capa Aplicación (HTTP)

Capa Transporte (TCP)

Capa Red (IP)

Capa Enlace (Ethernet II)

Wireshark: Una cabecera

Internet Protocol Version 4, Src: 192.168.120.255, Dst: 150.214.40.97

0100 = Version: 4
.... 0101 = Header Length: 20 bytes (5)

> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)

Total Length: 1500

Identification: 0x028e (654)

> Flags: 0x01 (More Fragments)

Fragment offset: 1480

Time to live: 128

Protocol: ICMP (1)

Header checksum: 0x18fb [validation disabled]
[Header checksum status: Unverified]

Source: 192.168.120.255

Destination: 150.214.40.97

[Source GeoIP: Unknown]

[Destination GeoIP: Unknown]

[Reassembled IPv4 in frame: 8](#)

Campo calculado

Campos extra ofrecidos por wireshark

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																																									
1	Versión				IHL				Tipo de servicio								Tamaño total																																																							
	Identificación												Flags				Fragmento Offset																																																							
	Tiempo de vida								Protocolo								Header checksum																																																							
																																	Dirección IP del emisor																																							
																																	Dirección IP del receptor																																							
																	Opciones																								Padding																															
																																																	Datos																							

0000	00	1b	8f	03	04	00	68	07	15	1f	99	a1	08	00	45	00
0010	05	dc	02	8e	20	b9	80	01	18	fb	c0	a8	78	ff	96	d6
0020	28	61	61	62	63	64	65	66	67	68	69	6a	6b	6c	6d	6e
0030	6f	70	71	72	73	74	75	76	77	61	62	63	64	65	66	67
0040	68	69	6a	6b	6c	6d	6e	6f	70	71	72	73	74	75	76	77

0010 0000 1011 1001
flags desplazamiento

185 (x8 = 1480)

Para las prácticas habitualmente se solicita el valor del paquete real (no el interpretado por wireshark)

Wireshark: Una cabecera

- Interpretación de la cabecera por parte de Wireshark (operaciones adicionales):

Internet Protocol Version 4, Src: 192.168.120.1

- 0100 = Version: 4
- 0101 = Header Length: 20 bytes (5)
- > Differentiated Services Field: 0x00 (DSCP: CS0)
- Total Length: 1500
- Identification: 0x028e (654)
- > Flags: 0x01 (More Fragments)
- Fragment offset: 1480
- Time to live: 128
- Protocol: ICMP (1)
- Header checksum: 0x18fb [validation disabled] [Header checksum status: Unverified]
- Source: 192.168.120.255
- Destination: 150.214.40.97
- [Source GeoIP: Unknown]
- [Destination GeoIP: Unknown]
- [Reassembled IPv4 in frame: 8](#)
- > Data (1480 bytes)

Expand Subtrees Shift+Right

Expand All Ctrl+Right

Collapse All Ctrl+Left

Apply as Column

Apply as Filter

Prepare a Filter

Conversation Filter

Colorize with Filter

Follow

Copy

Show Packet Bytes...

Export Packet Bytes... Ctrl+H

Wiki Protocol Page

Filter Field Reference

Protocol Preferences

Decode As...

Go to Linked Packet

Show Linked Packet in New Window

Botón derecho sobre el protocolo a analizar

Open Internet Protocol Version 4 preferences...

- ✓ Decode IPv4 TOS field as DiffServ field
- ✓ Reassemble fragmented IPv4 datagrams
- ✓ Show IPv4 summary in protocol tree
- Validate the IPv4 checksum if possible
- ✓ Support packet-capture from IP TSO-enabled hardware
- ✓ Enable GeoIP lookups
- Interpret Reserved flag as Security flag (RFC 3514)
- Try heuristic sub-dissectors first
- Disable IPv4...

0000 00 1b 8f 03 04 00 68 07 15 1f 99 a1 08 00 45 00h.E.

0010 05 dc 02 8e 20 b9 80 01 18 fb c0 a8 78 ff 96 d6x...

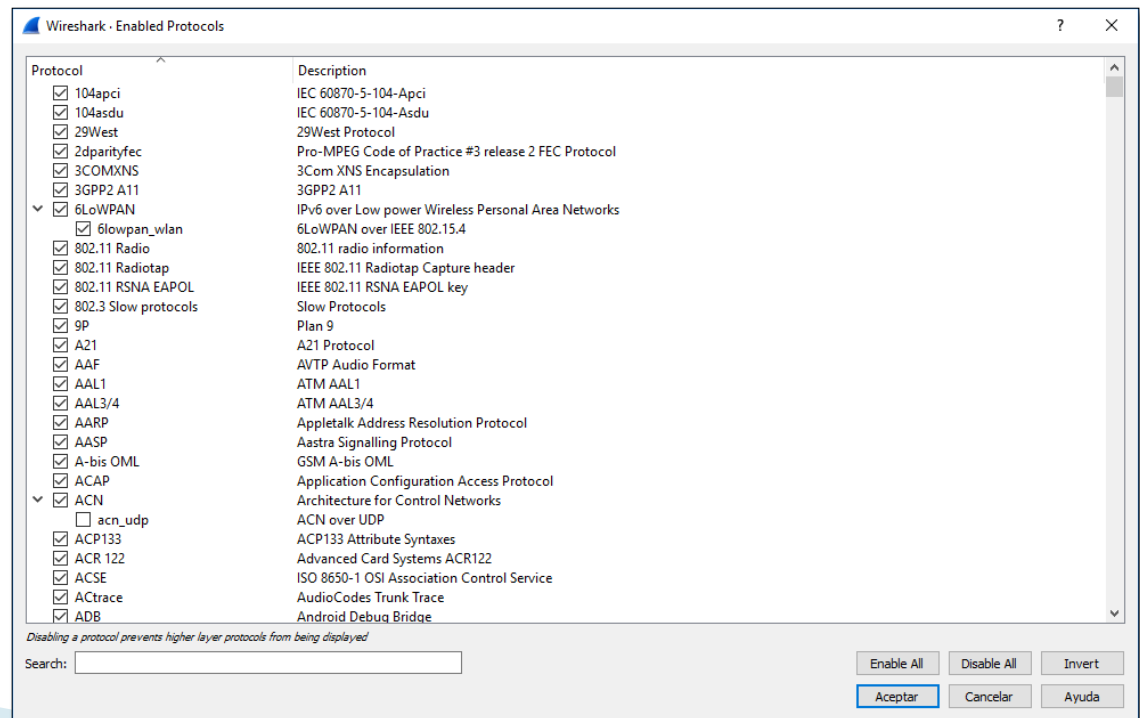
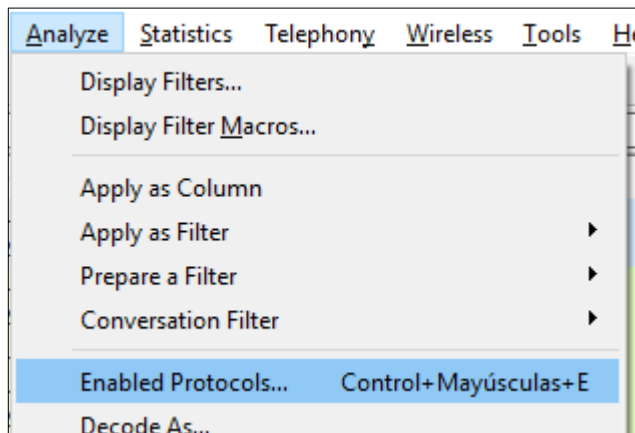
0020 28 61 61 62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e (abcdef ghijklmn

0030 6f 70 71 72 73 74 75 76 77 61 62 63 64 65 66 67 opqrstuv wabcdefg

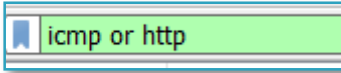
0040 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76 77 hijklmno pqrstuvw

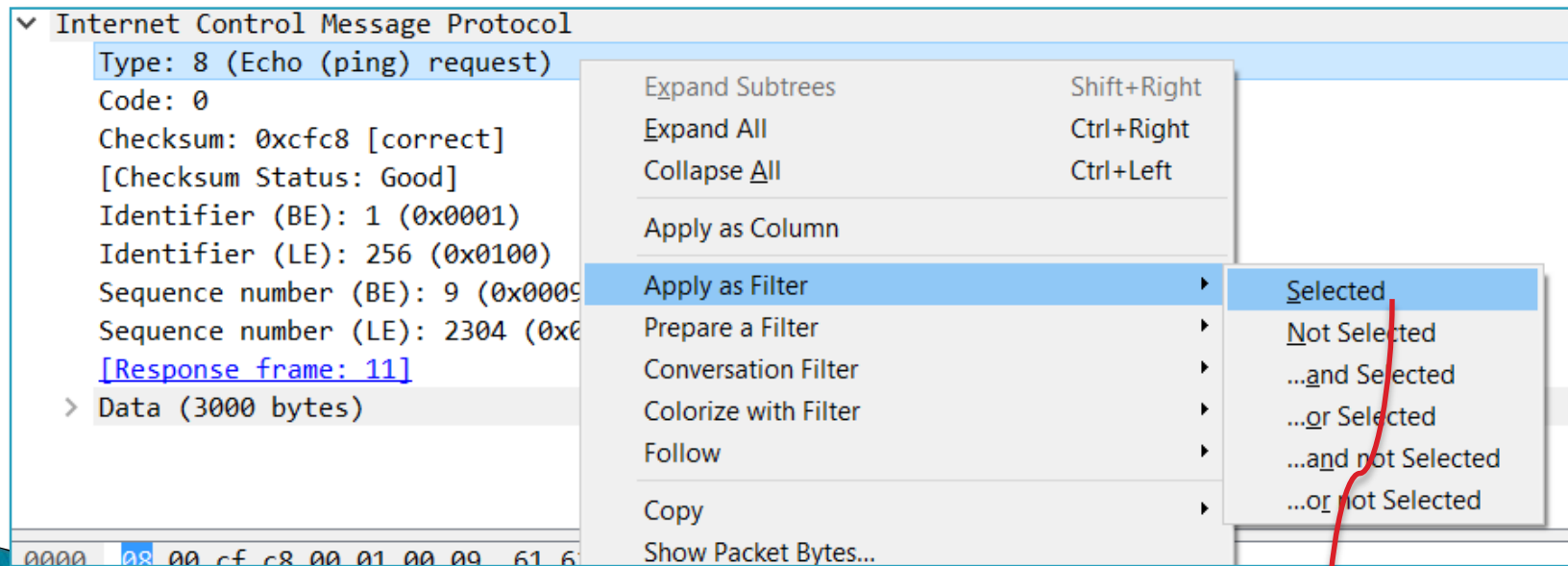
Wireshark: Una cabecera

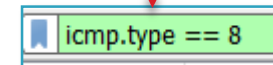
- ▶ Analyze -> Enabled protocols
 - Permite seleccionar qué protocolos queremos que se sean analizados (interpretar los datos como campos de cabecera, ...):



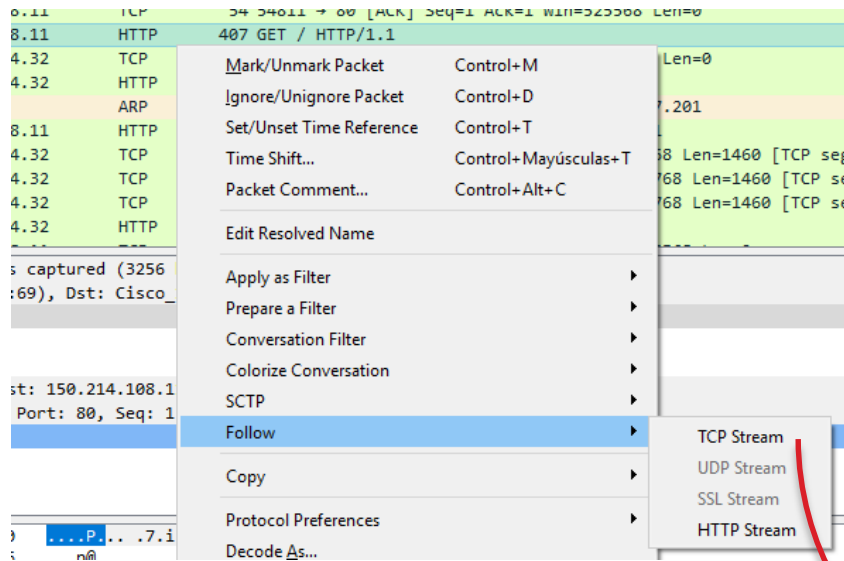
Wireshark: Filtros sobre campos

- ▶ Filtros por protocolo: 
- ▶ Filtros campos de protocolos:
 - En la barra
 - En la cabecera

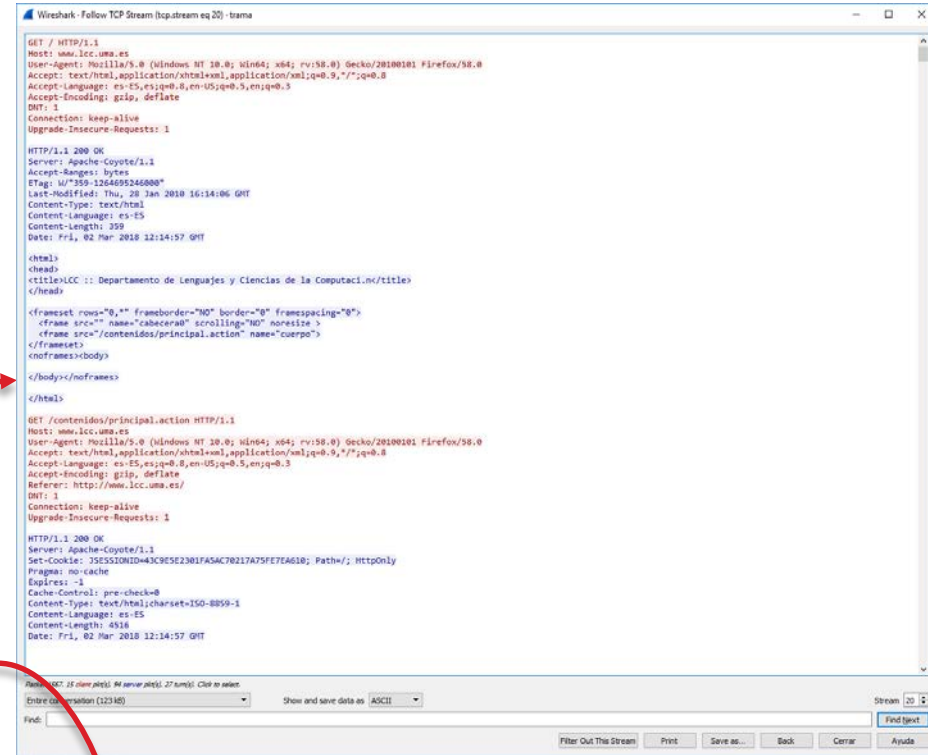




Wireshark: Filtros sobre flujos



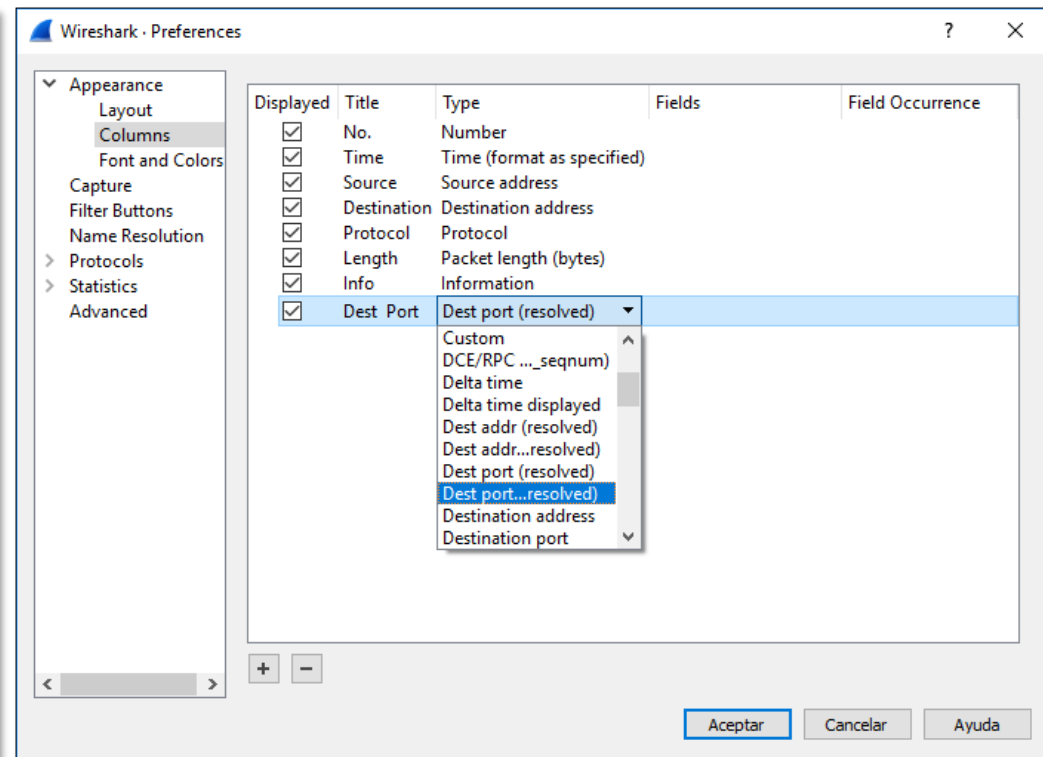
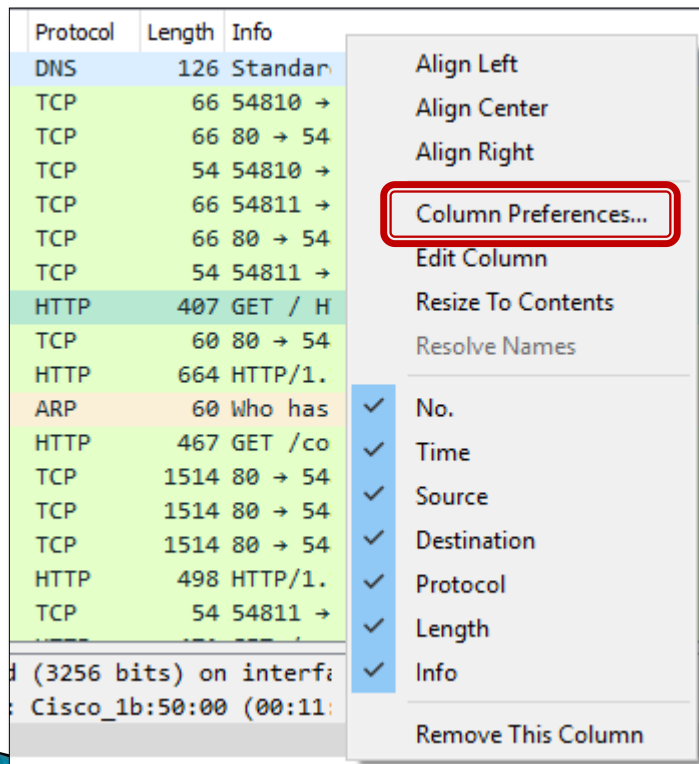
Botón derecho sobre el paquete
a analizar (debe ser TCP)



tcp.stream eq 20

Wireshark: Campos visualizados

- Podemos personalizar los campos que aparecen en la parte superior de la ventana principal:

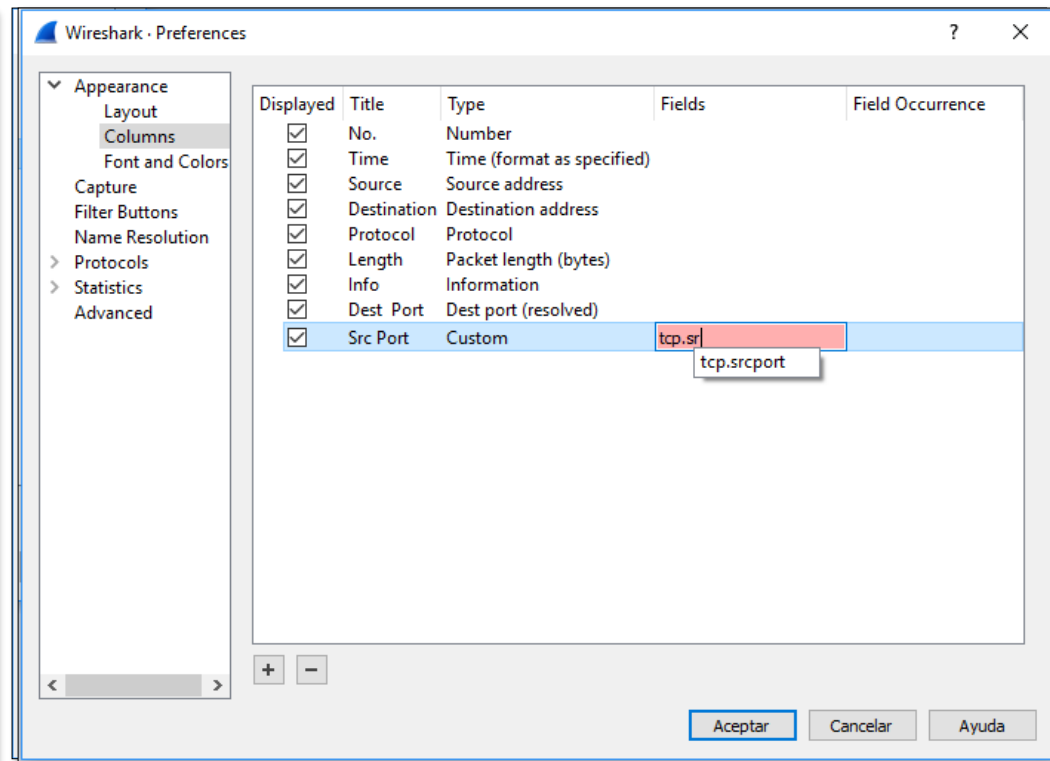
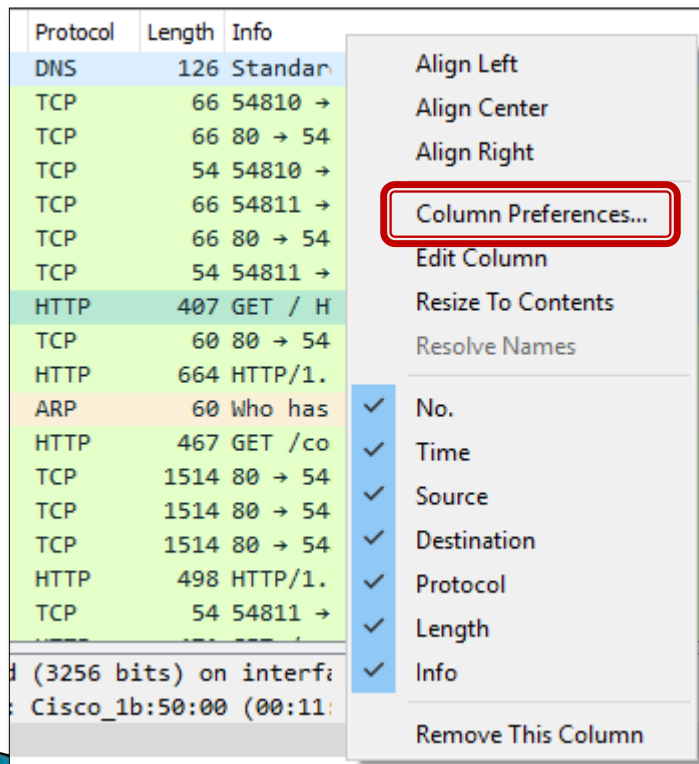


Botón derecho sobre el paquete
los títulos superiores

Usando los tipos existentes

Wireshark: Campos visualizados

- Podemos personalizar los campos que aparecen en la parte superior de la ventana principal:



Botón derecho sobre el paquete
los títulos superiores

Usando campos de los
paquetes

Wireshark: Campos visualizados

- Podemos personalizar los campos que aparecen en la parte superior de la ventana principal:

The screenshot shows the Wireshark interface with the packet list pane on the left and the packet details pane on the right. The packet list pane has columns for No., Time, Source, Destination, Protocol, Length, and Info. The packet details pane shows the selected packet (No. 1565) and its details (HTTP GET / HTTP/1.1). A red box highlights the 'Dest Port' and 'Src Port' columns in the packet list pane. A context menu is open over the packet list pane, showing options to 'Length', 'Info', and 'Remove This Column'. The 'Wireshark - Preferences' dialog box is also visible in the background.

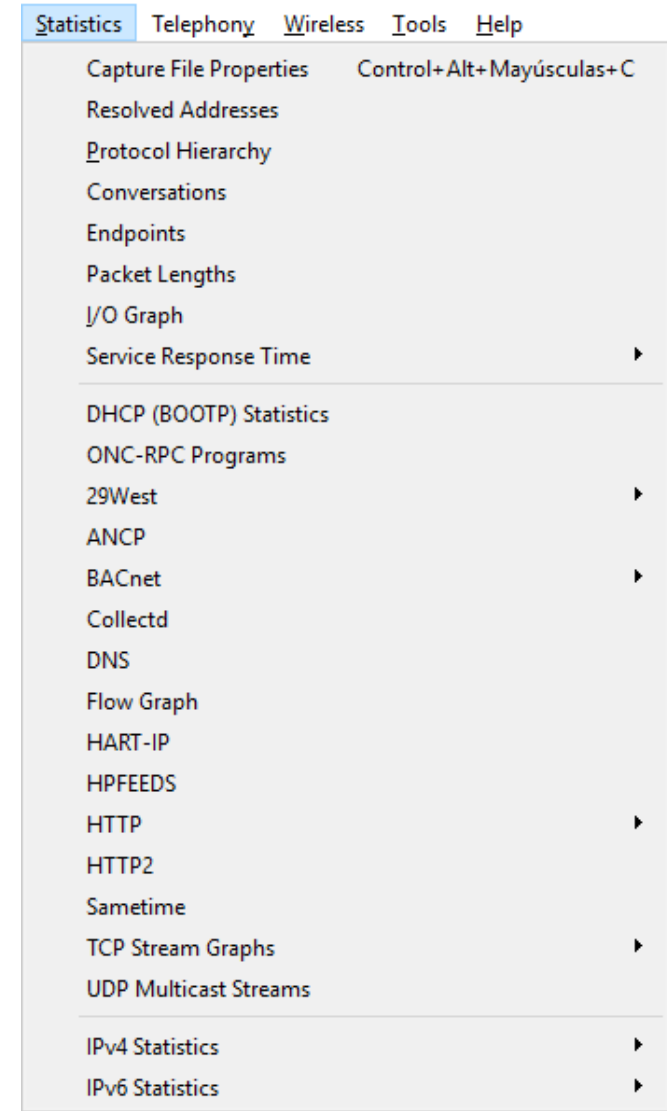
No.	Time	Source	Destination	Protocol	Length	Info	Dest Port	Src Port
1558	56.141863	150.214.57.7	192.168.164.32	DNS	126	Standard query response 0xabe7 AAAA www.lcc.uma.es SOA thor.ctima.uma.es	58123	
1559	56.141997	192.168.164.32	150.214.108.11	TCP	66	54810 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1	80	54810
1560	56.142398	150.214.108.11	192.168.164.32	TCP	66	80 → 54810 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1460 SACK_PERM=1	54810	80
1561	56.142474	192.168.164.32	150.214.108.11	TCP	54	54810 → 80 [ACK] Seq=1 Ack=1 Win=525568 Len=0	80	54810
1562	56.154911	192.168.164.32	150.214.108.11	TCP	66	54811 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1	80	54811
1563	56.155216	150.214.108.11	192.168.164.32	TCP	66	80 → 54811 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1460 SACK_PERM=1	54811	80
1564	56.155258	192.168.164.32	150.214.108.11	TCP	54	54811 → 80 [ACK] Seq=1 Ack=1 Win=525568 Len=0	80	54811
1565	56.155355	192.168.164.32	150.214.108.11	HTTP	407	GET / HTTP/1.1	80	54811
1566	56.155666	150.214.108.11	192.168.164.32	TCP	60	80 → 54811 [ACK] Seq=1 Ack=354 Win=15744 Len=0	54811	80
1567	56.156482	150.214.108.11	192.168.164.32	HTTP	664	HTTP/1.1 200 OK (text/html)	54811	80
1568	56.159341	Vmware_9a:ec:cb	Broadcast	ARP	60	Who has 192.168.166.129? Tell 192.168.167.201		
1569	56.176990	192.168.164.32	150.214.108.11	HTTP	467	GET /contenidos/principal.action HTTP/1.1	80	54811
1570	56.181890	150.214.108.11	192.168.164.32	TCP	1514	80 → 54811 [ACK] Seq=611 Ack=767 Win=16768 Len=1460 [TCP segment of	54811	80
1571	56.181893	150.214.108.11	192.168.164.32	TCP	1514	80 → 54811 [ACK] Seq=2071 Ack=767 Win=16768 Len=1460 [TCP segment of	54811	80
1572	56.181896	150.214.108.11	192.168.164.32	TCP	1514	80 → 54811 [ACK] Seq=3531 Ack=767 Win=16768 Len=1460 [TCP segment of	54811	80
1573	56.181897	150.214.108.11	192.168.164.32	HTTP	498	HTTP/1.1 200 OK (text/html)	54811	80
1574	56.182012	192.168.164.32	150.214.108.11	TCP	54	54811 → 80 [ACK] Seq=767 Ack=5435 Win=525568 Len=0	80	54811

Botón derecho sobre el paquete
los títulos superiores

Usando campos de los
paquetes

Wireshark: Estadísticas y visualizaciones adicionales

- ▶ Wireshark nos ofrece diversos análisis, estadísticas y visualizaciones de la captura
- ▶ Menú Statistics:
 - Generales
 - Sobre protocolos concretos
- ▶ En las próximas transparencias mostramos algunos ejemplos (pero pruebe el resto de opciones)



Wireshark: Estadísticas generales

► Propiedades de la captura (datos generales):

Wireshark · Capture File Properties · trama

Details

File

Name: F:\wireshark\trama.pcapng
Length: 1399 kB
Format: Wireshark/... - pcapng
Encapsulation: Ethernet

Time

First packet: 2018-03-02 13:14:00
Last packet: 2018-03-02 13:15:19
Elapsed: 00:01:19

Capture

Hardware: Intel(R) Pentium(R) CPU G3240 @ 3.10GHz (with SSE4.2)
OS: 64-bit Windows 10, build 14393
Application: Dumpcap (Wireshark) 2.4.3 (v2.4.3-0-g368ba1ee37)

Interfaces

Interface	Dropped packets	Capture filter	Link type	Packet size limit
\Device {\NPFF_{9BEE4EA5-203E-4E5A-9014-9C2EB5EA0CF3}}	0 (0 %)	none	Ethernet	262144 bytes

Statistics

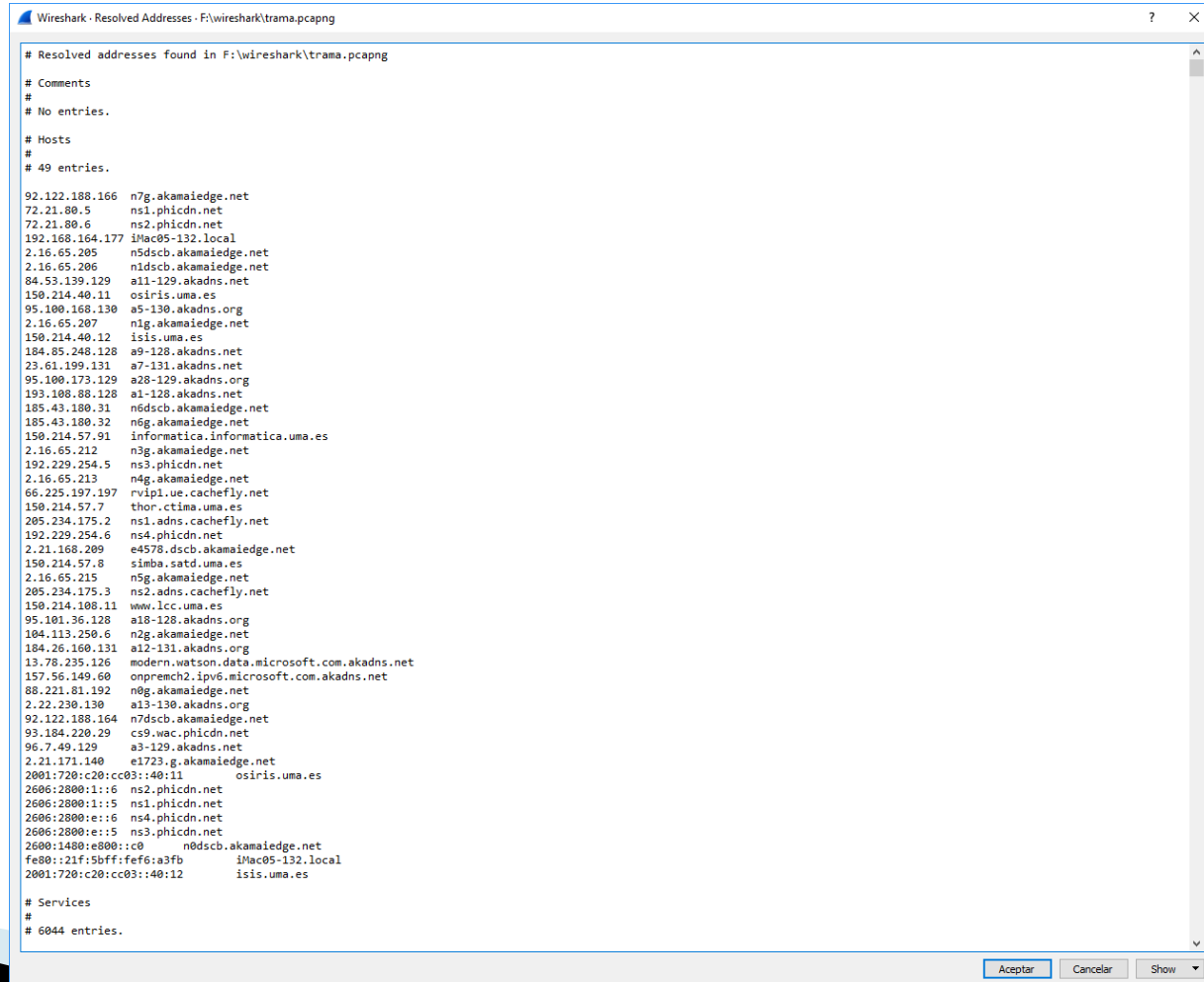
Measurement	Captured	Displayed	Marked
Packets	2159	2159 (100.0%)	—
Time span, s	79.195	79.195	—
Average pps	27.3	27.3	—

Capture file comments

Refresh Save Comments Cerrar Copy To Clipboard Ayuda

Wireshark: Estadísticas generales

- Direcciones (físicas y lógicas) que aparecen en la captura



```
Wireshark - Resolved Addresses - F:\wireshark\trama.pcapng

# Resolved addresses found in F:\wireshark\trama.pcapng
#
# Comments
#
# No entries.
#
# Hosts
#
# 49 entries.
92.122.188.166 n7g.akamaiedge.net
72.21.80.5 ns1.phicdn.net
72.21.80.6 ns2.phicdn.net
192.168.164.177 iMac05-132.local
2.16.65.205 n5dscb.akamaiedge.net
2.16.65.206 n1dscb.akamaiedge.net
84.53.139.129 a11-129.akadns.net
150.214.40.11 osiris.uma.es
95.100.168.130 a5-130.akadns.org
2.16.65.207 n1g.akamaiedge.net
150.214.40.12 isis.uma.es
184.85.248.128 a9-128.akadns.net
23.61.199.131 a7-131.akadns.net
95.100.173.129 a28-129.akadns.org
193.108.88.128 a1-128.akadns.net
185.43.180.31 n6dscb.akamaiedge.net
185.43.180.32 n6g.akamaiedge.net
150.214.57.91 informatica.informatica.uma.es
2.16.65.212 n3g.akamaiedge.net
192.229.254.5 ns3.phicdn.net
2.16.65.213 n4g.akamaiedge.net
66.225.197.197 rvip1.ue.cachefly.net
150.214.57.7 thor.ctima.uma.es
205.234.175.2 ns1.adns.cachefly.net
192.229.254.6 ns4.phicdn.net
2.21.168.209 e4578.dscb.akamaiedge.net
150.214.57.8 simba.satd.uma.es
2.16.65.215 n5g.akamaiedge.net
205.234.175.3 ns2.adns.cachefly.net
150.214.108.11 www.lcc.uma.es
95.101.36.128 a18-128.akadns.org
104.113.250.6 n2g.akamaiedge.net
184.26.160.131 a12-131.akadns.org
13.78.235.126 modern.watson.data.microsoft.com.akadns.net
157.56.149.60 onpremh2.ipv6.microsoft.com.akadns.net
88.221.81.192 n0g.akamaiedge.net
2.22.230.130 a13-130.akadns.org
92.122.188.164 n7dscb.akamaiedge.net
93.184.220.29 cs9.wac.phicdn.net
96.7.49.129 a3-129.akadns.net
2.21.171.140 e1723.g.akamaiedge.net
2001:720:c20:cc03::40:11 osiris.uma.es
2606:2800:1::6 ns2.phicdn.net
2606:2800:1::5 ns1.phicdn.net
2606:2800:e::6 ns4.phicdn.net
2606:2800:e::5 ns3.phicdn.net
2600:1480:e000::c0 n0dscb.akamaiedge.net
fe80::21f:5bff:fef6:a3fb iMac05-132.local
2001:720:c20:cc03::40:12 isis.uma.es

# Services
#
# 6044 entries.
```

Wireshark: Estadísticas generales

► Jerarquía de protocolos (con estadísticas)

Wireshark · Protocol Hierarchy Statistics · trama

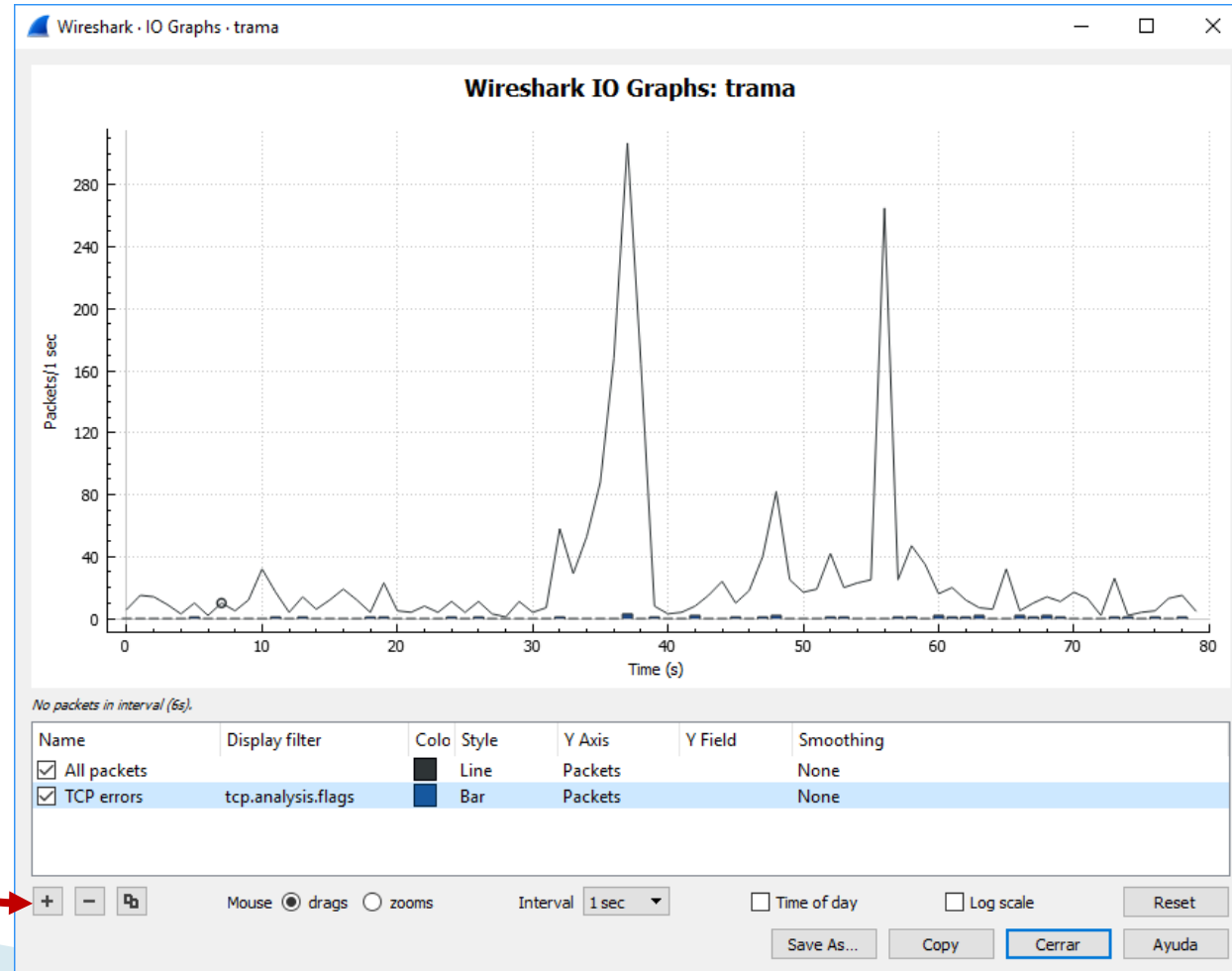
Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
▼ Frame	100.0	2159	100.0	1326644	134 k	0	0	0
▼ Ethernet	100.0	2159	2.3	30226	3053	0	0	0
▼ Logical-Link Control	1.9	41	0.2	2005	202	0	0	0
Spanning Tree Protocol	1.9	40	0.1	1400	141	40	1400	141
Cisco Discovery Protocol	0.0	1	0.0	477	48	1	477	48
▼ Internet Protocol Version 6	3.3	72	0.2	2880	290	0	0	0
▼ User Datagram Protocol	2.1	45	0.0	360	36	0	0	0
Multicast Domain Name System	0.6	13	0.3	3800	383	13	3800	383
Link-local Multicast Name Resolution	1.1	24	0.0	544	54	24	544	54
DHCPv6	0.4	8	0.1	688	69	8	688	69
Internet Control Message Protocol v6	1.3	27	0.1	688	69	27	688	69
▼ Internet Protocol Version 4	91.7	1979	3.0	39692	4009	0	0	0
▼ User Datagram Protocol	31.2	673	0.4	5384	543	0	0	0
Teredo IPv6 over UDP tunneling	0.3	6	0.0	366	36	0	0	0
Simple Service Discovery Protocol	8.2	178	2.6	34626	3497	178	34626	3497
NetBIOS Name Service	7.1	154	0.6	7946	802	154	7946	802
Multicast Domain Name System	1.3	27	0.3	4360	440	27	4360	440
Link-local Multicast Name Resolution	11.9	256	0.5	6092	615	256	6092	615
Domain Name System	1.3	28	0.4	5573	562	28	5573	562
Data	0.9	20	1.6	21640	2186	20	21640	2186
Bootstrap Protocol	0.2	4	0.1	1457	147	4	1457	147
▼ Transmission Control Protocol	57.2	1234	86.8	1151545	116 k	1044	1001494	101 k
▼ Hypertext Transfer Protocol	9.0	195	85.3	1131685	114 k	39	12879	1300
Secure Sockets Layer	5.7	123	70.4	933719	94 k	118	915662	92 k
Portable Network Graphics	0.6	14	8.0	106596	10 k	14	109636	11 k
Online Certificate Status Protocol	0.2	4	0.1	1496	151	4	2108	212
Media Type	0.0	1	0.1	1098	110	1	1098	110
Line-based text data	0.2	5	1.9	24985	2523	5	25713	2597
JPEG File Interchange Format	0.0	1	1.8	24266	2451	1	24522	2477
CompuServe GIF	0.4	8	1.3	16690	1685	8	17198	1737
Protocol Independent Multicast	0.1	3	0.0	102	10	3	102	10
Open Shortest Path First	0.4	8	0.0	448	45	8	448	45
Internet Group Management Protocol	1.3	29	0.0	232	23	29	232	23
Internet Control Message Protocol	1.5	32	0.2	2336	235	32	2336	235
▼ Configuration Test Protocol (loopback)	0.4	8	0.0	368	37	0	0	0
Data	0.4	8	0.0	320	32	8	320	32
Address Resolution Protocol	3.0	65	0.1	1820	183	65	1820	183

No display filter.

Cerrar Copy Ayuda

Wireshark: Estadísticas generales

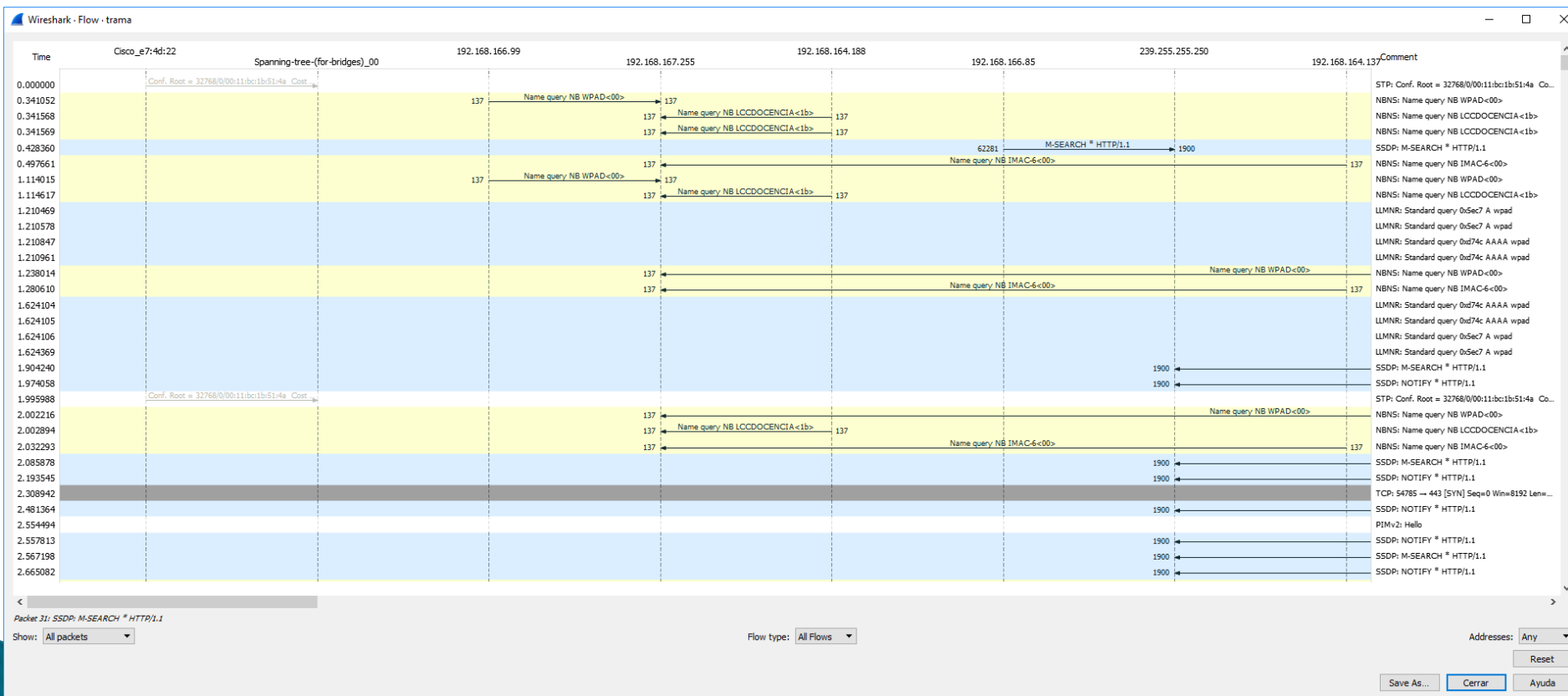
► Gráficas temporales



Se puede agregar más

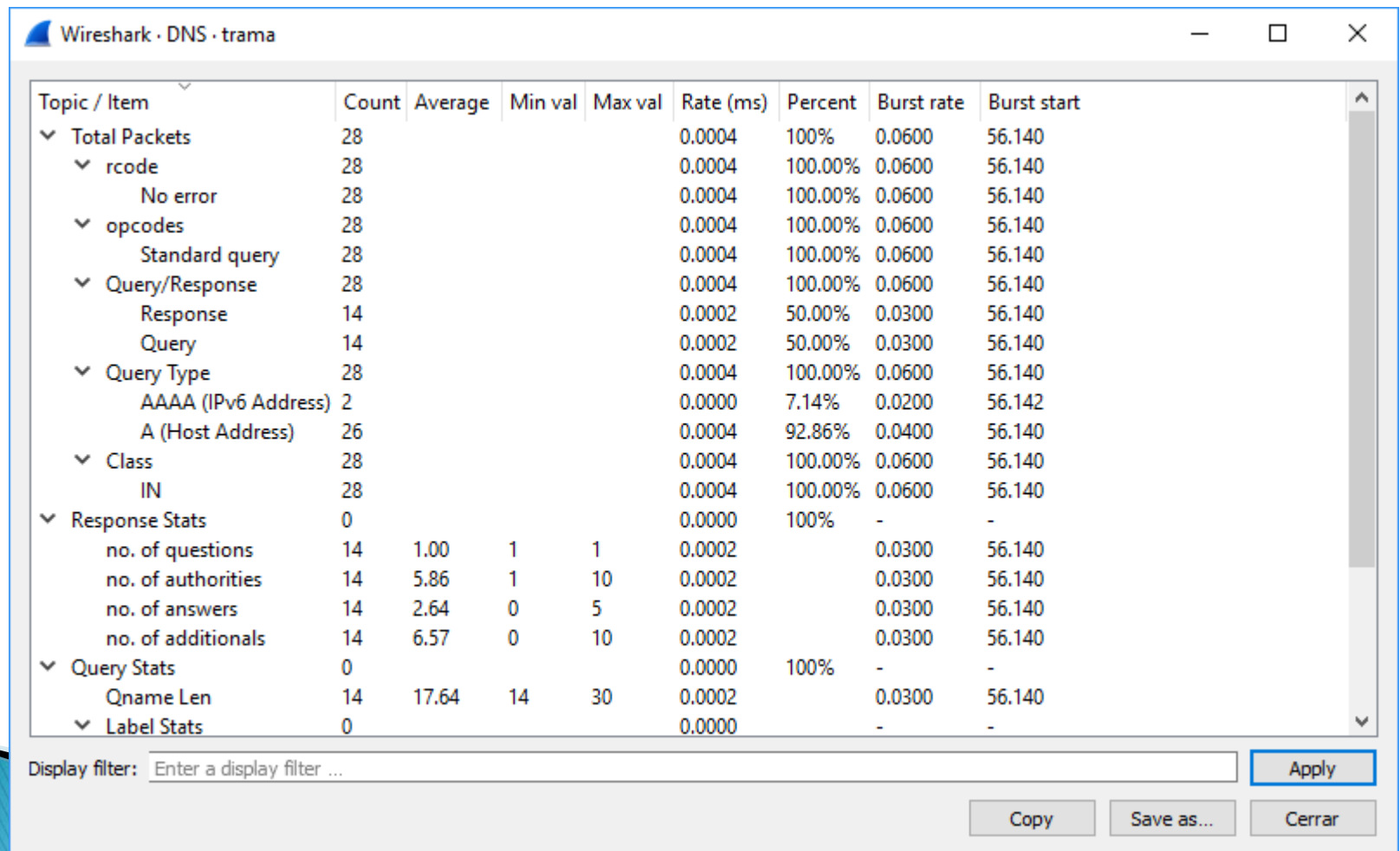
Wireshark: Estadísticas generales

► Flujo de mensajes



Wireshark: Estadísticas de Protocolos

► DNS

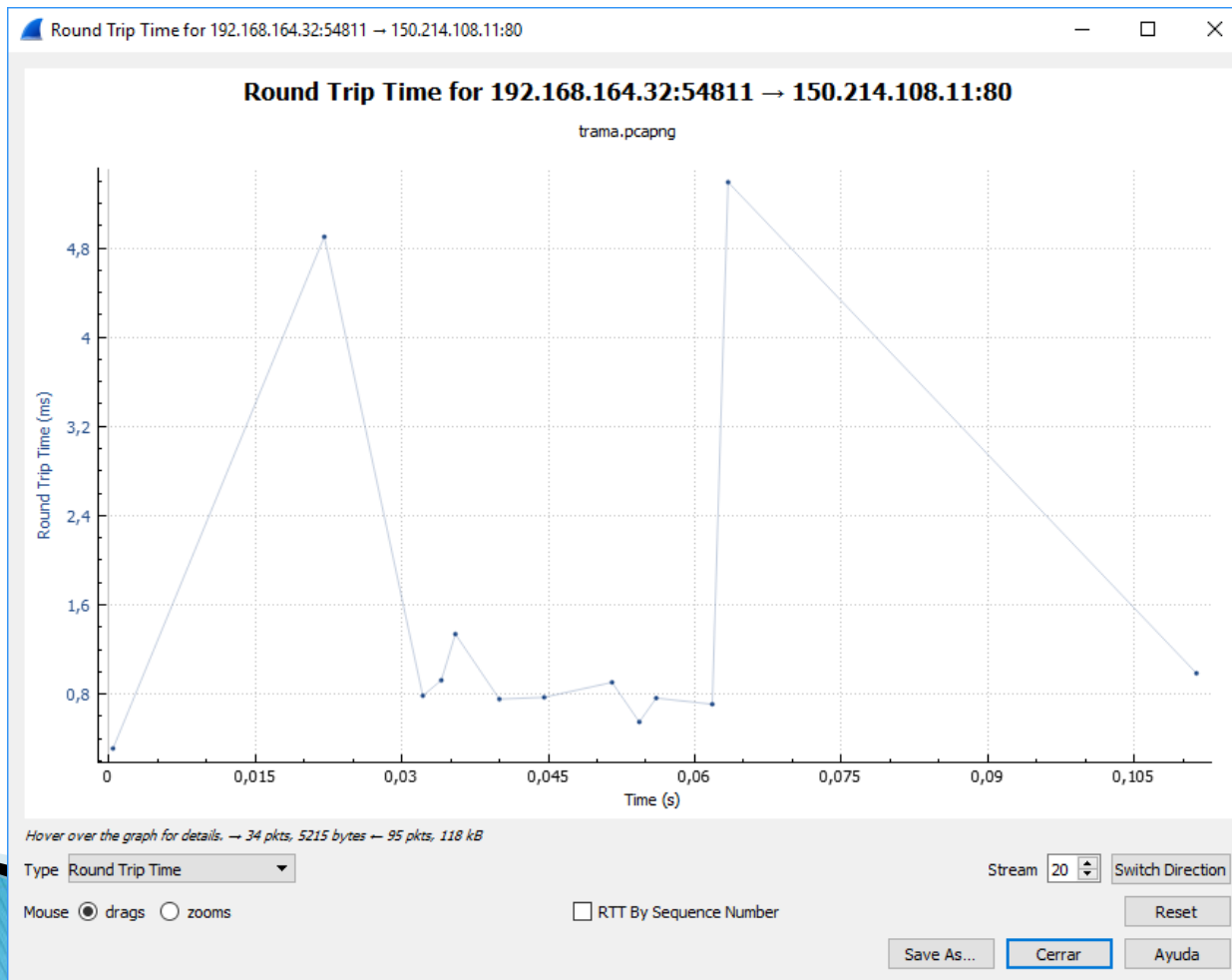
The image shows the 'Wireshark · DNS · trama' window. It contains a table with statistics for various DNS fields. The table has columns for 'Topic / Item', 'Count', 'Average', 'Min val', 'Max val', 'Rate (ms)', 'Percent', 'Burst rate', and 'Burst start'. The data is organized into a tree structure with expandable items. At the bottom, there is a 'Display filter' input field and buttons for 'Apply', 'Copy', 'Save as...', and 'Cerrar'.

Topic / Item	Count	Average	Min val	Max val	Rate (ms)	Percent	Burst rate	Burst start
▼ Total Packets	28				0.0004	100%	0.0600	56.140
▼ rcode	28				0.0004	100.00%	0.0600	56.140
No error	28				0.0004	100.00%	0.0600	56.140
▼ opcodes	28				0.0004	100.00%	0.0600	56.140
Standard query	28				0.0004	100.00%	0.0600	56.140
▼ Query/Response	28				0.0004	100.00%	0.0600	56.140
Response	14				0.0002	50.00%	0.0300	56.140
Query	14				0.0002	50.00%	0.0300	56.140
▼ Query Type	28				0.0004	100.00%	0.0600	56.140
AAAA (IPv6 Address)	2				0.0000	7.14%	0.0200	56.142
A (Host Address)	26				0.0004	92.86%	0.0400	56.140
▼ Class	28				0.0004	100.00%	0.0600	56.140
IN	28				0.0004	100.00%	0.0600	56.140
▼ Response Stats	0				0.0000	100%	-	-
no. of questions	14	1.00	1	1	0.0002		0.0300	56.140
no. of authorities	14	5.86	1	10	0.0002		0.0300	56.140
no. of answers	14	2.64	0	5	0.0002		0.0300	56.140
no. of additional	14	6.57	0	10	0.0002		0.0300	56.140
▼ Query Stats	0				0.0000	100%	-	-
Qname Len	14	17.64	14	30	0.0002		0.0300	56.140
▼ Label Stats	0				0.0000		-	-

Display filter:

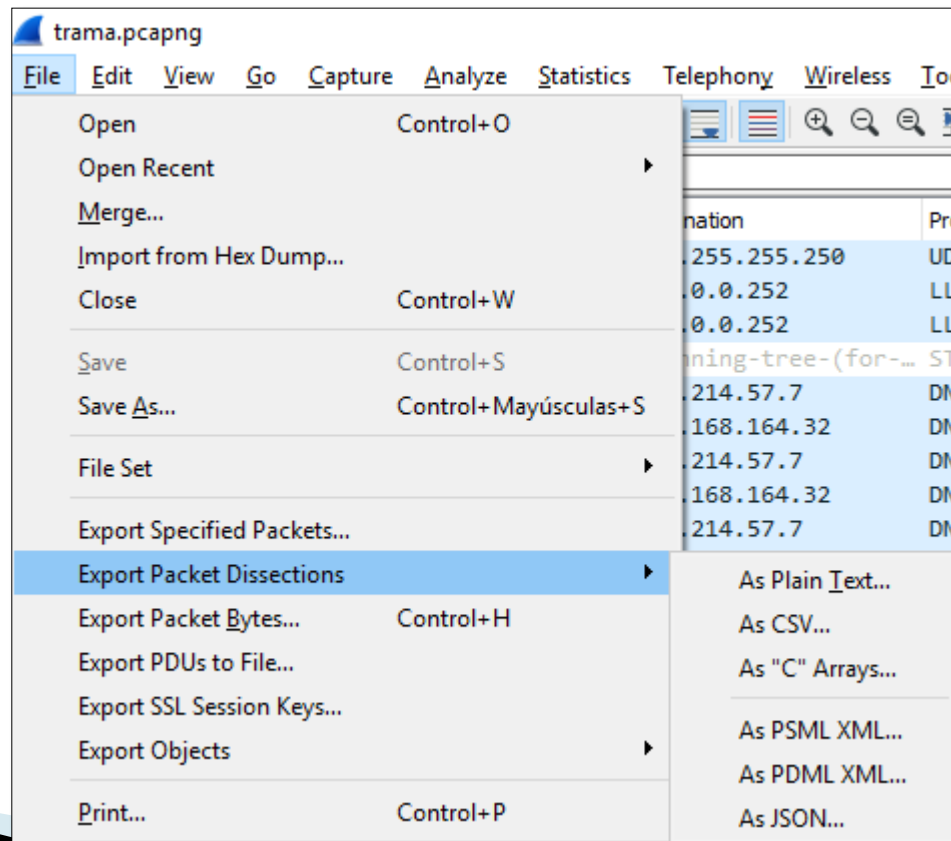
Wireshark: Estadísticas de Protocolos

- TCP (RTT = Tiempo de ida y vuelta)



Wireshark: Exportación

- ▶ Wireshark permite exportar los paquetes (ya analizados) en otros formatos:



Wireshark: Exportación

► Ejemplo en JSON



```
Archivo  Editar  Ver  Historial  Marcadores  Herramientas  Ayuda
/F:/wireshark/test.json  X  +
←  →  ↺  🏠  file:///F:/wireshark/test.json
JSON  Datos sin procesar  Cabeceras
Guardar  Copiar
}
}
},
{
  "_index": "packets-2018-03-02",
  "_type": "pcap_file",
  "_score": null,
  "_source": {
    "layers": {
      "frame": {
        "frame.interface_id": "0",
        "frame.interface_id_tree": {
          "frame.interface_name": "\\Device\\NPF_{9BEE4EA5-203E-4E5A-9014-9C2EB5EA0CF3}"
        },
        "frame.encap_type": "1",
        "frame.time": "Mar  2, 2018 13:14:56.656161000 Hora est\u00c3\u00a1ndar romance",
        "frame.offset_shift": "0.000000000",
        "frame.time_epoch": "1519992896.656161000",
        "frame.time_delta": "0.000184000",
        "frame.time_delta_displayed": "0.000184000",
        "frame.time_relative": "56.207527000",
        "frame.number": "1699",
        "frame.len": "459",
        "frame.cap_len": "459",
        "frame.marked": "0",
        "frame.ignored": "0",
        "frame.protocols": "eth:ethertype:ip:tcp:http",
        "frame.coloring_rule.name": "HTTP",
        "frame.coloring_rule.string": "http || tcp.port == 80 || http2"
      },
      "eth": {
        "eth.dst": "00:11:bc:1b:50:00",
        "eth.dst_tree": {
          "eth.dst_resolved": "Cisco_1b:50:00",
          "eth.addr": "00:11:bc:1b:50:00",
          "eth.addr_resolved": "Cisco_1b:50:00",
          "eth.lg": "0",
          "eth.ig": "0"
        },
        "eth.src": "8c:dc:d4:37:0b:69",
        "eth.src_tree": {
          "eth.src_resolved": "HewlettP_37:0b:69",
          "eth.addr": "8c:dc:d4:37:0b:69",
          "eth.addr_resolved": "HewlettP_37:0b:69",
          "eth.lg": "0",
          "eth.ig": "0"
        },
        "eth.type": "0x00008000"
      },
      "ip": {
        "ip.version": "4",
        "ip.hdr_len": "20",
        "ip.dsfield": "0x00000000",

```