

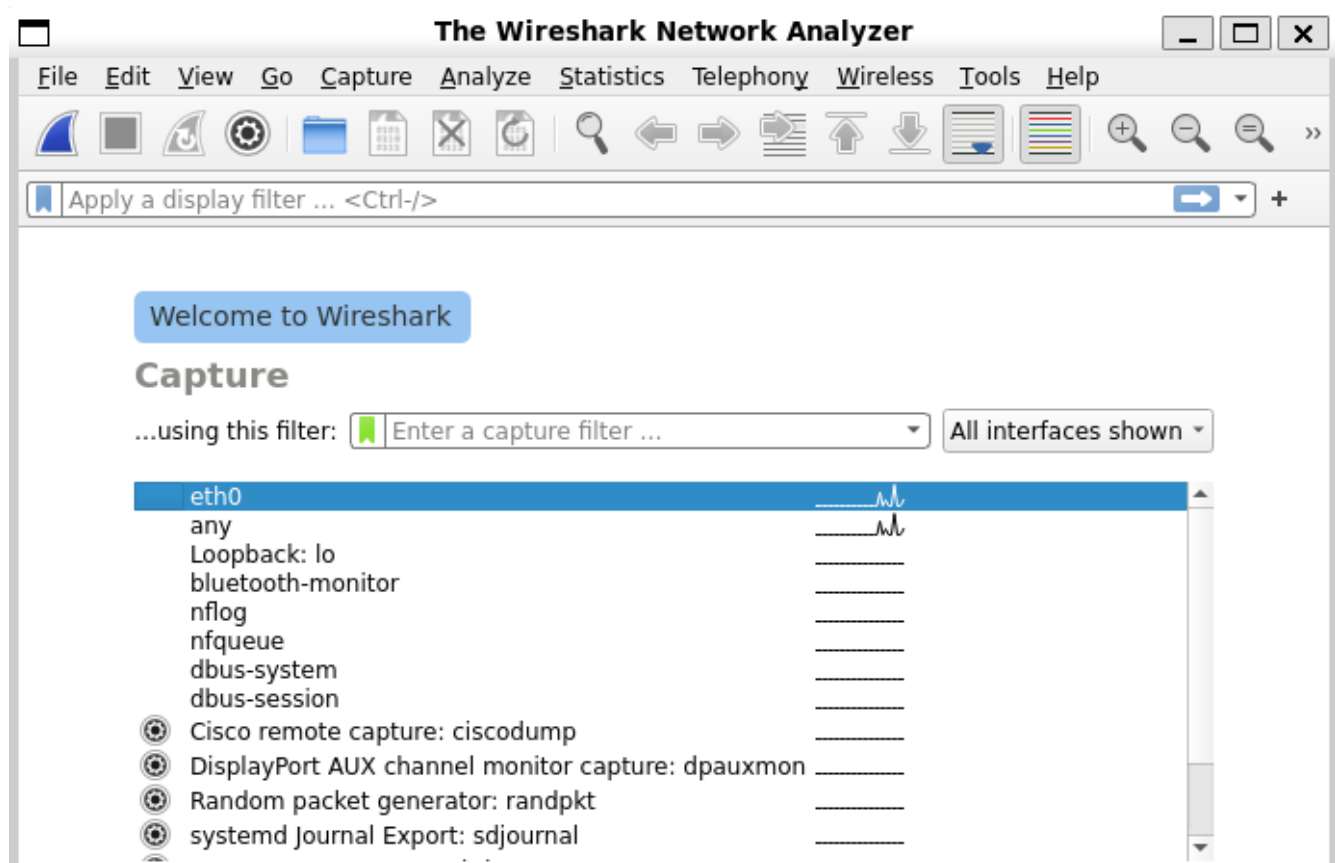
## Avvertenze

Prestare attenzione alla sicurezza nella installazione di Wireshark. Il consiglio è di installarlo e utilizzarlo all'interno di un ambiente virtualizzato.

## Referimenti:

- [https://www.wireshark.org/docs/wsug\\_html\\_chunked/ChUseMainWindowSection.html](https://www.wireshark.org/docs/wsug_html_chunked/ChUseMainWindowSection.html)
- [https://www.wireshark.org/docs/wsdg\\_html\\_chunked/ChapterDissection.html#ChDissectWorks](https://www.wireshark.org/docs/wsdg_html_chunked/ChapterDissection.html#ChDissectWorks)
- <https://wiki.wireshark.org/DisplayFilters>
- [https://www.wireshark.org/docs/wsug\\_html\\_chunked/ChWorkBuildDisplayFilterSection.html](https://www.wireshark.org/docs/wsug_html_chunked/ChWorkBuildDisplayFilterSection.html)

Quando si avvia Wireshark, la prima cosa mostrata è una finestra per la selezione dell'interfaccia su cui effettuare la cattura dei pacchetti.



Sotto WSL2, la scheda di rete (virtuale) principale è `eth0`.

`lo` è l'interfaccia di *loopback* utilizza per la comunicazione locale.

Facendo doppio click su una delle interfacce, viene avviata la cattura dei pacchetti e viene mostrata una nuova finestra simile alla seguente.

**Capturing from eth0**

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
10	0.768419792	fe80::37c3:8675:a55...	ff02::fb	MDNS	515	Standard quer
11	0.769704360	172.21.240.1	224.0.0.251	MDNS	431	Standard quer
12	0.771111524	fe80::37c3:8675:a55...	ff02::fb	MDNS	451	Standard quer
13	17.787519312	172.21.242.109	91.189.91.157	NTP	90	NTP Version 4
14	17.898172138	91.189.91.157	172.21.242.109	NTP	90	NTP Version 4
15	24.189460893	Microsof_86:00:ea	Microsof_66:0c:b7	ARP	42	Who has 172.2
16	24.190124948	Microsof_66:0c:b7	Microsof_86:00:ea	ARP	42	172.21.240.1

Frame 1: 430 bytes on wire (3440 bits), 430 bytes captured (3440 bits) on interface eth0, i  
 Ethernet II, Src: Microsof\_66:0c:b7 (00:15:5d:66:0c:b7), Dst: IPv4mcast\_fb (01:00:5e:00:00:  
 Internet Protocol Version 4, Src: 172.21.240.1, Dst: 224.0.0.251  
 User Datagram Protocol, Src Port: 5353, Dst Port: 5353  
 Multicast Domain Name System (response)

0000 01 00 5e 00 00 fb 00 15 5d 66 0c b7 08 00 45 00 ..^.....]f...E.  
 0010 01 a0 31 17 00 00 01 11 0a 24 ac 15 f0 01 e0 00 ..1.....\$.  
 0020 00 fb 14 e9 14 e9 01 8c c1 4f 00 00 84 00 00 00 .....0.....  
 0030 00 01 00 00 00 02 06 5f 64 6f 73 76 63 04 5f 74 .....dosvc\_t  
 0040 63 70 05 6c 6f 63 61 6c 00 00 0c 00 01 00 00 00 cp:local.....  
 0050 00 00 23 0f 44 45 53 4b 54 4f 50 2d 52 45 54 55 ..#DESK TOP-RETU  
 0060 48 49 49 06 5f 64 6f 73 76 63 04 5f 74 63 70 05 HII\_dos vc\_tcp  
 0070 6c 6f 63 61 6c 00 0f 44 45 53 4b 54 4f 50 2d 52 local\_D ESKTOP-R  
 0080 45 54 55 48 49 49 06 5f 64 6f 73 76 63 04 5f 74 ETUHII\_dosvc\_t  
 0090 63 70 05 6c 6f 63 61 6c 00 00 21 00 01 00 00 00 cp:local..!....

eth0: <live capture in progress> Packets: 16 · Displayed: 16 (100.0%) Profile: Default

In alto a sinistra, troviamo i pulsanti per avviare, fermare o riavviare la cattura dei pacchetti (perdendo nell'ultimo caso tutti i pacchetti catturati).

Il campo di testo in cima permette di inserire delle espressioni, dette *filtri*, appunto per filtrare i pacchetti. Alcuni esempi:

- nome-protocollo (es. http): mostra i pacchetti contenuti un certo protocollo
- http.request.method == "POST": mostra i pacchetti contenenti richieste HTTP con il metodo POST

Il pannello immediatamente sottostante contiene la *lista dei pacchetti* dopo l'applicazione di eventuali filtri.

Il pacchetto selezionato all'interno della lista viene mostrato nel *pannello dei dettagli del pacchetto* (subito sotto) e nel *pannello dei byte del pacchetto*.

Il pannello dei dettagli del pacchetto è popolato per mezzo di vari *dissector*, ciascuno decodifica la parte di uno specifico protocollo, affidando la decodifica del protocollo incapsulato al relativo dissector.

Il pannello dei dettagli è in realtà un albero, in quando gli elementi possono essere espansi, per esempio per visualizzare i dettagli della intestazione di un certo livello. L'elemento selezionato viene inoltre selezionato nel pannello dei byte del pacchetto.

Nel seguito si assumerà di aver catturato i pacchetti relativi al seguente comando (mostrato durante la esercitazione HTTP):

```
curl --form "titolo=dante" --form "file1=@bruti.txt" --form  
"file2=@bruti.txt;type=text/plain" --form "text=<bruti.txt"  
http://httpbin.org/post
```

Usando il filtro `http`, si possono facilmente trovare la richiesta e la risposta.

The image shows the Wireshark network protocol analyzer interface. The title bar indicates the interface is `*eth0`. The menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, and Help. The toolbar contains various icons for packet capture and analysis. The packet list pane shows two captured packets, with packet 88 selected. The packet details pane shows the structure of the selected packet, including Ethernet II, Internet Protocol Version 4, Transmission Control Protocol, and Hypertext Transfer Protocol. The packet bytes pane shows the raw data and its ASCII representation.

No.	Time	Source	Destination	Protocol	Length	Info
88	374.467893523	172.21.242.109	3.224.7.64	HTTP	349	POST /post HTTP/1.1
93	374.581692751	3.224.7.64	172.21.242.109	HTTP/J...	802	HTTP/1.1 200 OK

Frame 88: 349 bytes on wire (2792 bits), 349 bytes captured (2792 bits) on interface eth0, Ethernet II, Src: Microsof\_86:00:ea (00:15:5d:86:00:ea), Dst: Microsof\_66:0c:b7 (00:15:5d:60:0c:b7), Internet Protocol Version 4, Src: 172.21.242.109, Dst: 3.224.7.64, Transmission Control Protocol, Src Port: 44274, Dst Port: 80, Seq: 641, Ack: 1, Len: 283, [3 Reassembled TCP Segments (923 bytes): #86(187), #87(453), #88(283)]

Hypertext Transfer Protocol

MIME Multipart Media Encapsulation, Type: multipart/form-data, Boundary: "-----"

0000 00 15 5d 66 0c b7 00 15 5d 86 00 ea 08 00 45 00 ..]f.... ]....E.  
0010 01 4f f5 7b 40 00 40 06 9a 8a ac 15 f2 6d 03 e0 .0. {@.@. ....m..  
0020 07 40 ac f2 00 50 83 3c 16 0c 5a 18 cb 00 80 18 .@...P.< ..Z.....  
0030 01 f6 aa e4 00 00 01 01 08 0a 37 f7 ed 1a 28 64 ..... ..7...(d  
0040 ef f7 66 61 74 74 69 20 6e 6f 6e 20 66 6f 73 74 ..fatti non fost  
0050 65 20 61 20 76 69 76 65 72 20 63 6f 6d 65 20 62 e a vive r come b  
0060 72 75 74 69 20 6d 61 20 70 65 72 20 73 65 67 75 ruti ma per sequ  
0070 69 72 65 20 76 69 72 74 75 74 65 20 65 20 63 61 ire virt ute e ca

Frame (349 bytes) Reassembled TCP (923 bytes)

Hypertext Transfer Protocol: Protocol Packets: 370 · Displayed: 2 (0.5%) Profile: Default

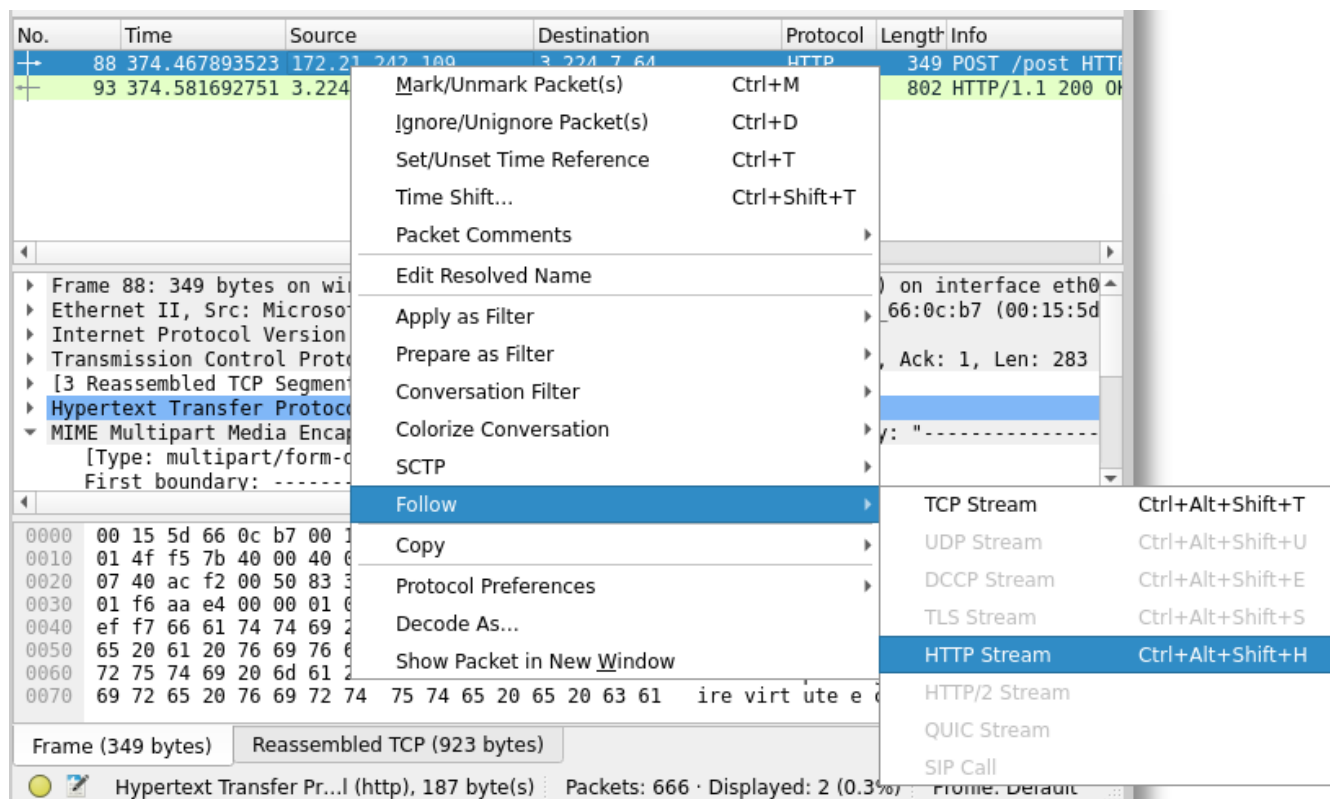
Nel pannello dei dettagli, osservare:

- il protocollo Ethernet (a livello di collegamento)
- il protocollo IP (a livello di rete), verificando che l'indirizzo sorgente e destinazione siano quello della macchina locale (trovare con `ip addr`) e quello di `httpbin.org` (trovare con `nslookup` oppure `dig`)
- il protocollo TCP (a livello di trasporto), verificano che la porta di destinazione è la 80
- la richiesta HTTP è stata ottenuta ri assemblando più segmenti, nell'esempio 3, il cui payload complessivo è stato analizzato per ottenere la richiesta HTTP
- la richiesta HTTP è stata scomposta nella intestazione e nel corpo. Guardando la intestazione, osservare tutti i campi di intestazione, incluso quello che specifica il tipo del corpo della richiesta come `multipart/form-data` e definisce il delimitatore (boundary). Osservare anche fatto Wireshark ha generato l'URL completa ottenuta combinando path e host, nonché ci fornisce un link alla risposta.

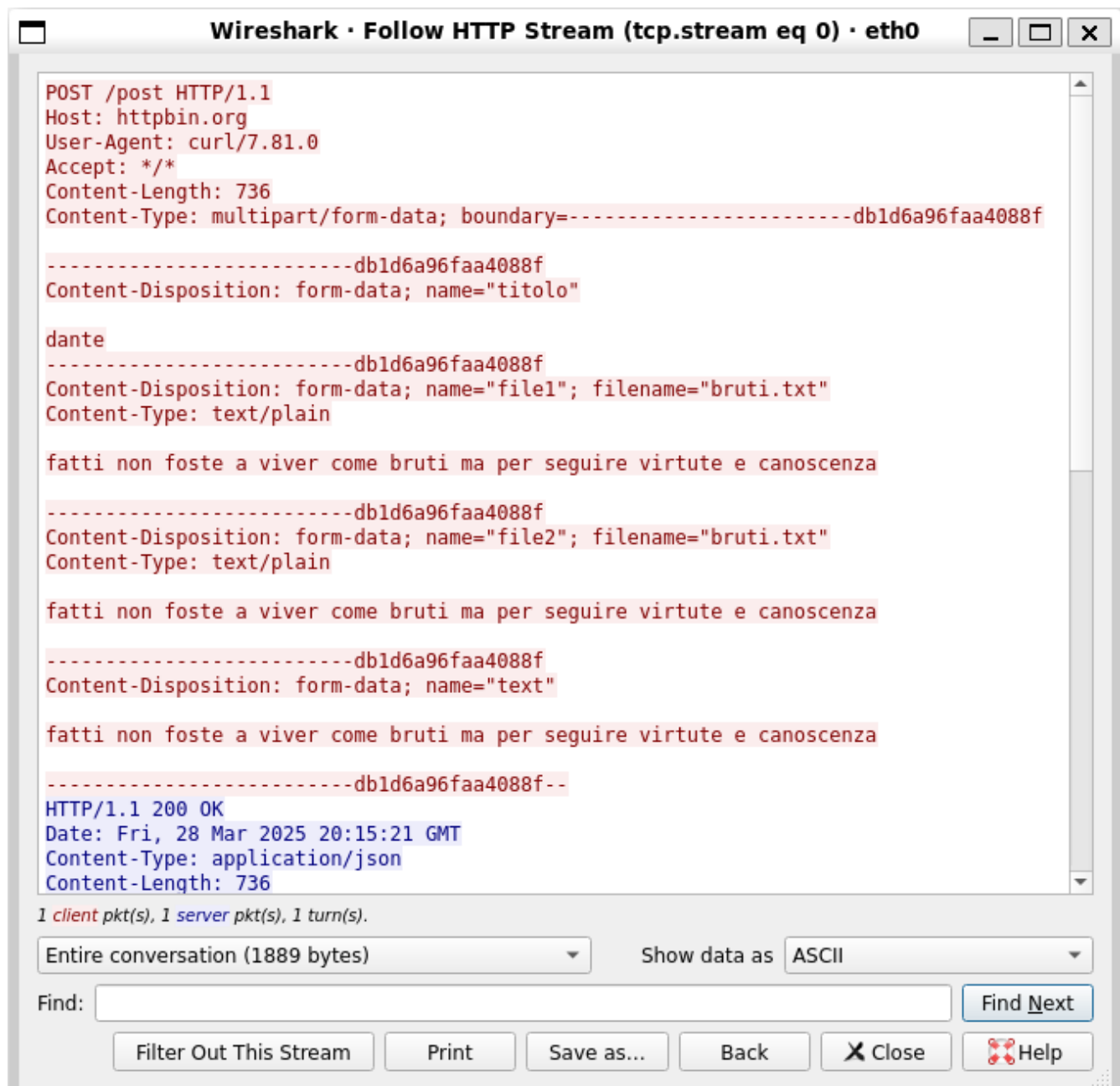
Notare che quando si opera a livello applicativo sono mostrati i byte prodotti dal ri assemblamento dei segmenti TCP.

Per la risposta valgono considerazioni analoghe, ma attenzione che mittente e destinatario sono invertiti in molte intestazioni.

Premendo col tasto destro del mouse su un pacchetto a HTTP, selezione *follow/HTTP stream*.



In questo modo è possibile vedere agevolmente la richiesta http e la relativa risposta.



Quando viene chiusa la finestra, nella finestra principale si troverà un filtro tipo `tcp.stream eq N` dove `N` è il numero del flusso TCP (a partire da 0) che è stato utilizzato da quel flusso http.

**Attenzione ai filtri:** in presenza di un filtro eventuali collegamenti tra richiesta e risposta potrebbero non essere seguiti, se il pacchetto non è stato escluso dai filtri.