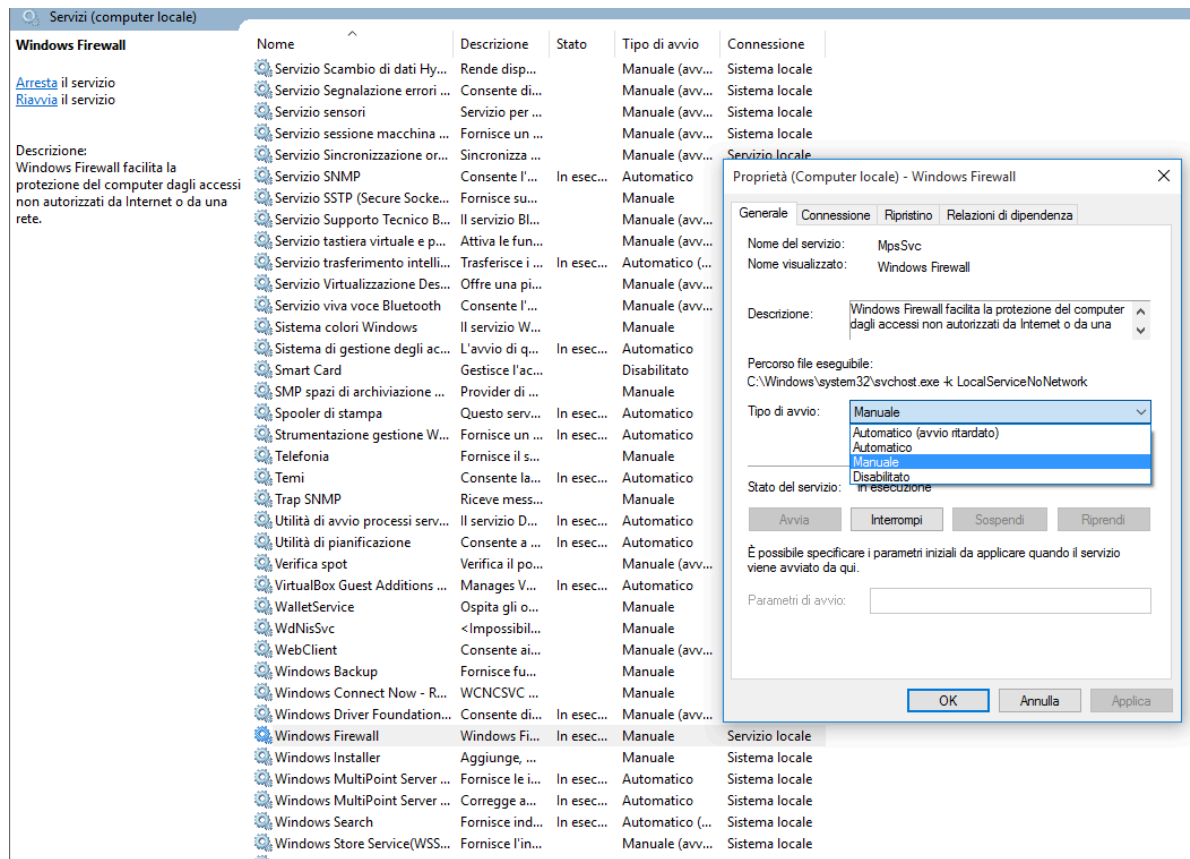


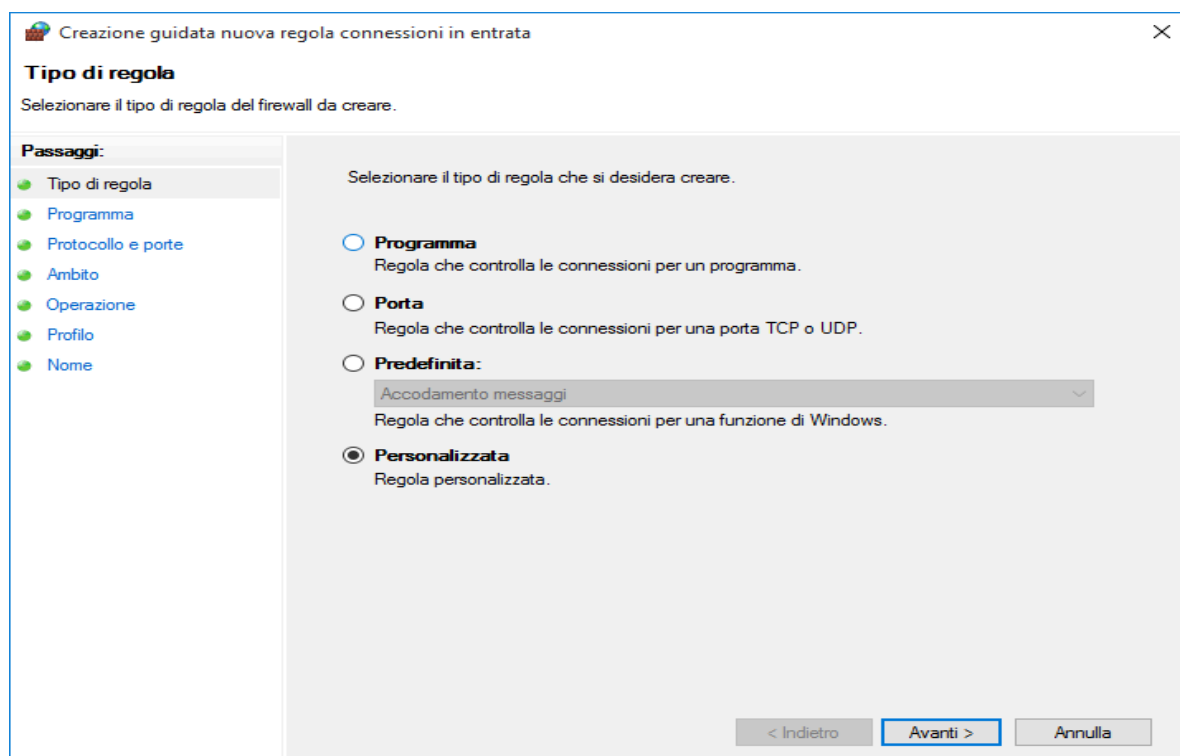
W3D4 - ESERCIZIO POLICY & PACKET CAPTURE

1)

Ricordiamo dall'es. W2D4 che la macchina Kali, e Metasploitable, non riuscivano a comunicare con la macchina Windows, questo perché il Firewall non lo permetteva, quindi per far ciò bisogna innanzitutto andare su Servizi e modificare da Automatico a Manuale il servizio Windows Firewall (come in figura sotto)



dopodiché avviare “Windows Firewall con Sicurezza avanzata” e andando su “regole conn. in entrata”(inbound rules) bisogna creare una nuova regola.



come tipo di regola si seleziona personalizzata -> per tutti i programmi -> si seleziona il tipo di protocollo ICMPv4 (ricordo che ping è basato su ICMP) -> qualsiasi indirizzo IP -> consenti la connessione -> selezioni tutti i tipi di applicazione della regola -> e infine inserire un nome della regola in questo caso ho scelto "ping_ok".

Facendo adesso la prova andando su Kali e lanciando il comando ping e l'IP di Windows (che ricordiamo è 192.168.50.102), si vedrà, come in figura, che adesso Kali riesce a comunicarsi con Windows.

```
(kali㉿kali)-[~]
$ ping -c 5 192.168.50.102
PING 192.168.50.102 (192.168.50.102) 56(84) bytes of data.
64 bytes from 192.168.50.102: icmp_seq=1 ttl=128 time=0.348 ms
64 bytes from 192.168.50.102: icmp_seq=2 ttl=128 time=0.420 ms
64 bytes from 192.168.50.102: icmp_seq=3 ttl=128 time=0.320 ms
64 bytes from 192.168.50.102: icmp_seq=4 ttl=128 time=0.289 ms
64 bytes from 192.168.50.102: icmp_seq=5 ttl=128 time=0.240 ms

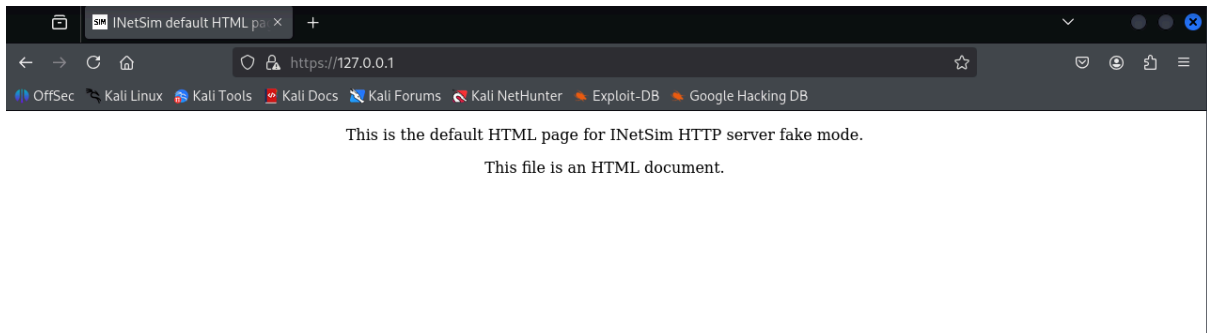
— 192.168.50.102 ping statistics —
5 packets transmitted, 5 received, 0% packet loss, time 4101ms
rtt min/avg/max/mdev = 0.240/0.323/0.420/0.060 ms
(kali㉿kali)-[~]
$
```

2)

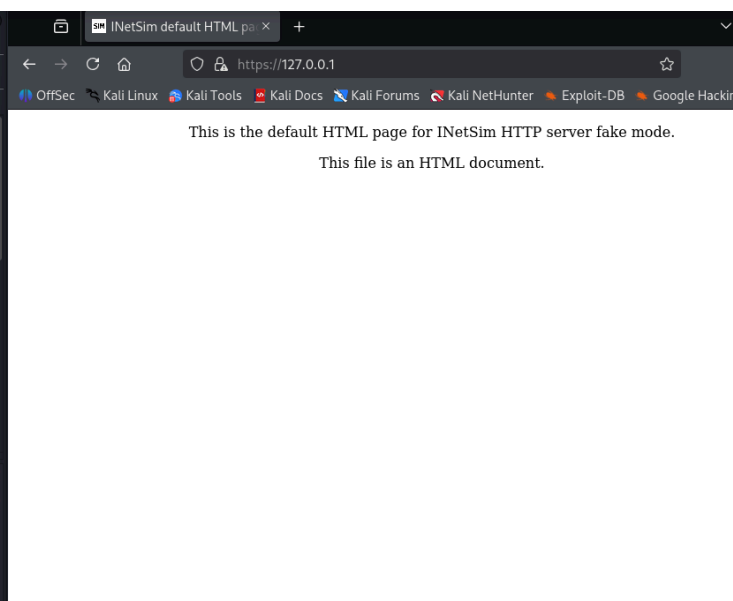
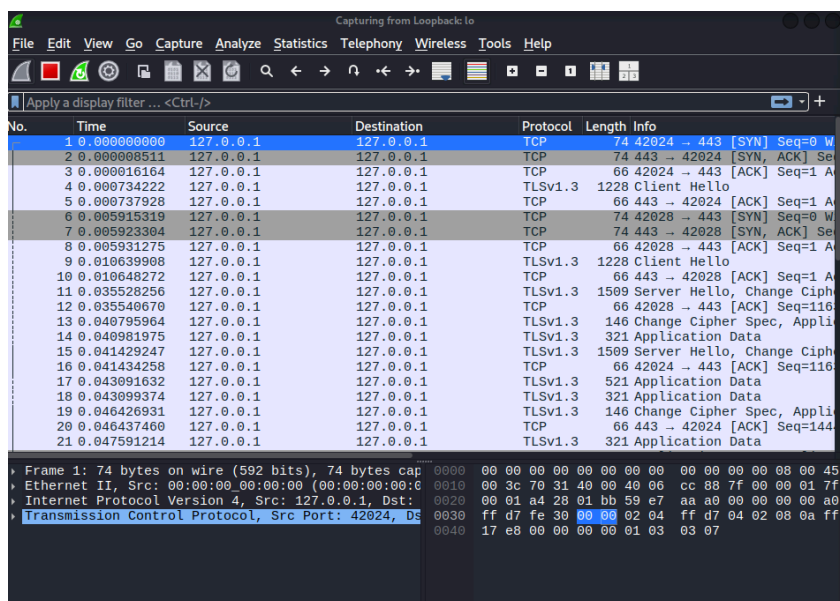
Utilizziamo InetSim per simulare alcuni servizi internet, in questo caso HTTPS con il comando `sudo inetsim`, avvieremo InetSim

```
(kali㉿kali)-[~]
$ sudo inetsim
INetSim 1.3.2 (2020-05-19) by Matthias Eckert & Thomas Hungenberg
Using log directory: /var/log/inetsim/
Using data directory: /var/lib/inetsim/
Using report directory: /var/log/inetsim/report/
Using configuration file: /etc/inetsim/inetsim.conf
Parsing configuration file.
Configuration file parsed successfully.
== INetSim main process started (PID 136978) ==
Session ID: 136978
Listening on: 127.0.0.1
Real Date/Time: 2025-07-17 18:37:44
Fake Date/Time: 2025-07-17 18:37:44 (Delta: 0 seconds)
Forking services ...
* https_443_tcp - started (PID 136988)
done.
Simulation running.
```

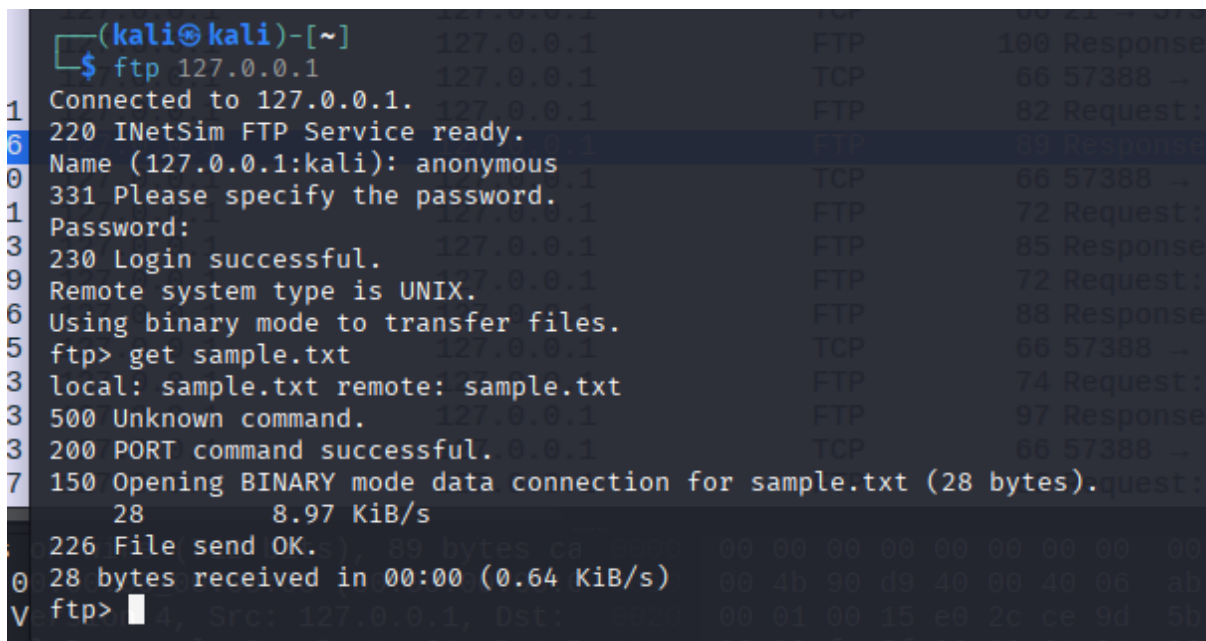
come possiamo vedere l'indirizzo è quello di default 127.0.0.1 del localhost, quindi se facciamo la prova su Firefox andando a cercare `https://127.0.0.1` ci darà questo risultato, quindi è attivo



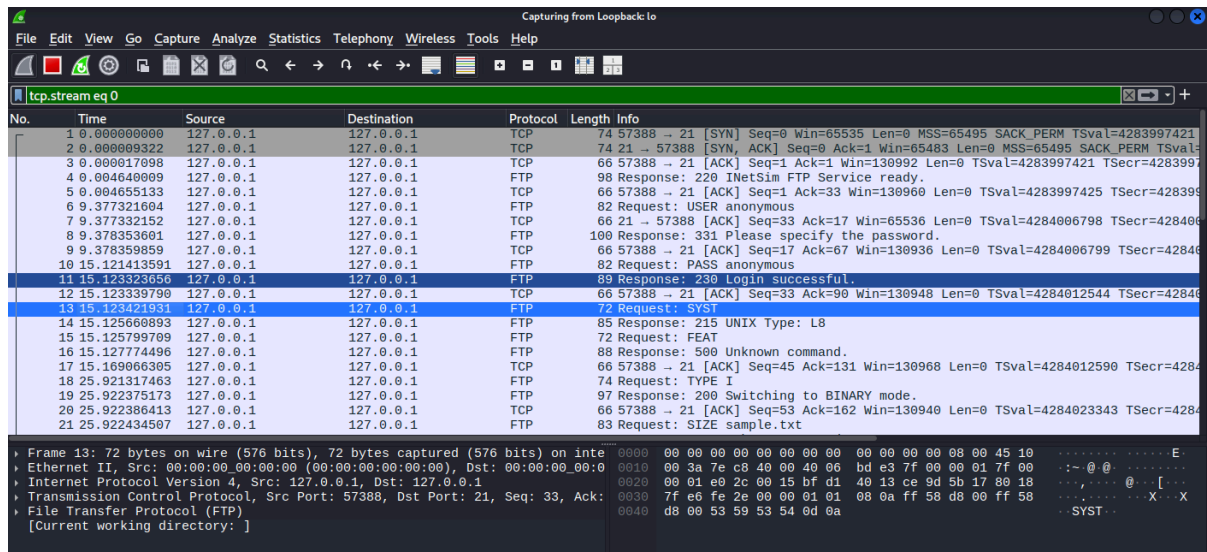
passando ora su Wireshark, che è un programma che cattura i pacchetti e li analizza, possiamo fare la prova sempre cercando https://127.0.0.1 e analizzare i pacchetti



Esercizio Facoltativo

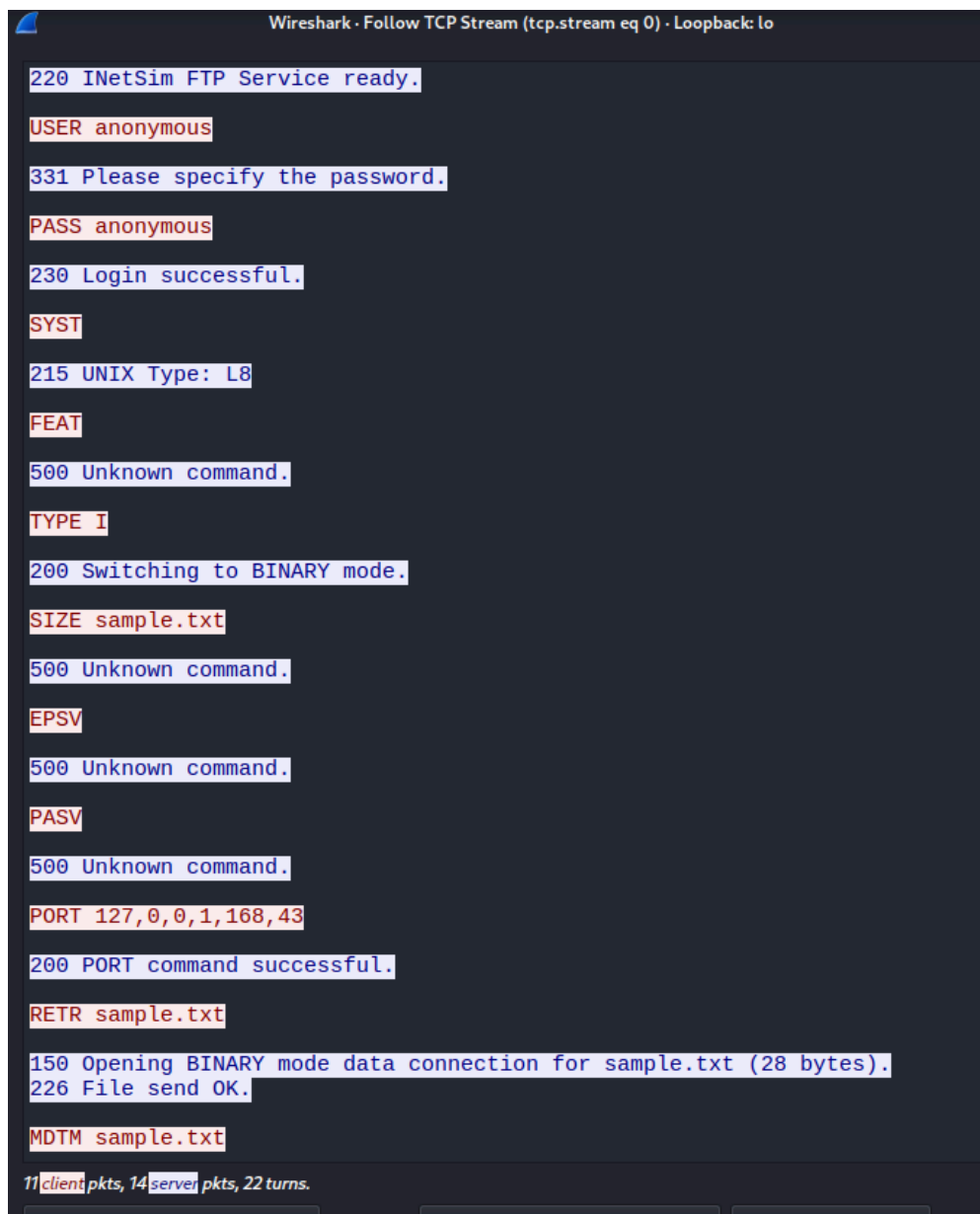


come si vede sopra, ho fatto la prova con il servizio FTP scaricando anche un file e su Wireshark fa vedere:



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	127.0.0.1	127.0.0.1	TCP	74	57388 → 21 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 SACK_PERM TSval=4283997421
2	0.000000932	127.0.0.1	127.0.0.1	TCP	74	21 → 57388 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_PERM TSval=4283997421
3	0.000017098	127.0.0.1	127.0.0.1	TCP	66	57388 → 21 [ACK] Seq=1 Ack=1 Win=130992 Len=0 TSval=4283997421 TSecr=4283997421
4	0.004640009	127.0.0.1	127.0.0.1	FTP	98	Response: 220 InetSim FTP Service ready.
5	0.004655133	127.0.0.1	127.0.0.1	TCP	66	57388 → 21 [ACK] Seq=1 Ack=33 Win=130960 Len=0 TSval=4283997425 TSecr=4283997425
6	0.377321604	127.0.0.1	127.0.0.1	FTP	82	Request: USER anonymous
7	0.377332152	127.0.0.1	127.0.0.1	TCP	66	21 → 57388 [ACK] Seq=33 Ack=17 Win=65536 Len=0 TSval=4284006798 TSecr=4284006798
8	0.378353601	127.0.0.1	127.0.0.1	FTP	100	Response: 331 Please specify the password.
9	0.378359859	127.0.0.1	127.0.0.1	TCP	66	57388 → 21 [ACK] Seq=17 Ack=67 Win=130936 Len=0 TSval=4284006799 TSecr=4284006799
10	15.121413591	127.0.0.1	127.0.0.1	FTP	82	Request: PASS anonymous
11	15.123323656	127.0.0.1	127.0.0.1	FTP	89	Response: 230 Login successful.
12	15.123339790	127.0.0.1	127.0.0.1	TCP	66	57388 → 21 [ACK] Seq=33 Ack=90 Win=130948 Len=0 TSval=4284012544 TSecr=4284012544
13	15.123421931	127.0.0.1	127.0.0.1	FTP	72	Request: SYST
14	15.125660893	127.0.0.1	127.0.0.1	FTP	85	Response: 215 UNIX Type: L8
15	15.125799709	127.0.0.1	127.0.0.1	FTP	72	Request: FEAT
16	15.127774496	127.0.0.1	127.0.0.1	FTP	88	Response: 500 Unknown command.
17	15.169906395	127.0.0.1	127.0.0.1	TCP	66	57388 → 21 [ACK] Seq=45 Ack=131 Win=130968 Len=0 TSval=4284012590 TSecr=4284012590
18	25.921317463	127.0.0.1	127.0.0.1	FTP	74	Request: TYPE I
19	25.922375173	127.0.0.1	127.0.0.1	FTP	97	Response: 200 Switching to BINARY mode.
20	25.922386413	127.0.0.1	127.0.0.1	TCP	66	57388 → 21 [ACK] Seq=53 Ack=162 Win=130940 Len=0 TSval=4284023343 TSecr=4284023343
21	25.922434507	127.0.0.1	127.0.0.1	FTP	83	Request: SIZE sample.txt

per l'appunto sul pacchetto no.11 da la risposta del login effettuato con successo, infatti se lo apriamo con tasto destro -> follow -> TCP stream, farà vedere la comunicazione tra client (rosso) e server (blu)



Client (Red)	Server (Blue)
	220 InetSim FTP Service ready.
USER anonymous	
	331 Please specify the password.
PASS anonymous	
	230 Login successful.
SYST	
	215 UNIX Type: L8
FEAT	
	500 Unknown command.
TYPE I	
	200 Switching to BINARY mode.
SIZE sample.txt	
	500 Unknown command.
EPSV	
	500 Unknown command.
PASV	
	500 Unknown command.
PORT 127,0,0,1,168,43	
	200 PORT command successful.
RETR sample.txt	
	150 Opening BINARY mode data connection for sample.txt (28 bytes).
	226 File send OK.
MDTM sample.txt	
11 client pkts, 14 server pkts, 22 turns.	