HOMEWORK 1

1)

NOME LAN	PREFISSO	INDIRIZZO BROADCAST	# INDIRIZZI INUTILIZZATI
LAN A	198.10.99.0/24	198.10.99.255/24	54
LAN B	198.10.98.64/26	198.10.98.127/26	12
LAN C	198.10.98.128/25	198.10.98.255/25	16
LAN G	198.10.98.32/27	198.10.98.63/27	10
LAN D	198.10.96.0/24	198.10.96.255/24	125
LAN E	198.10.97.0/27	198.10.97.31/27	10
LAN F	198.10.97.32/28	198.10.97.47/28	4
LAN H	198.10.98.0/30	198.10.98.3/30	0
LAN I	198.10.97.48/30	198.10.97.51/30	0
LAN L	198.10.98.4/30	198.10.98.7/30	0

Dato il blocco 198.10.96.0/22, ho 2 bit di subnetting e quindi 2^2 blocchi di classe C. Essendo già assegnato un indirizzo del blocco .98 alla sottorete collegata alla eth0 di r1, do il blocco .98 al ramo sinistro di r1,insieme al .99,essendo necessari 2 blocchi di classe C ed essendo il .99 l'unico blocco aggregabile al .98. Assegno gli indirizzi a partire dalle lan più grandi.

LAN A richiede 8 bit di host_id e quindi un intero blocco di classe C; gli assegno quindi il .99, poiché del .98 ho già utilizzato un indirizzo per la LAN H.

LAN C vuole 7 bit di host_id, quindi gli assegno gli indirizzi della metà superiore del blocco .98 sempre a causa dell'indirizzo 198.10.98.0 già assegnato.

LAN B ha 6 bit di host_id e gli continuo ad assegnare indirizzi della parte superiore del blocco rimasto; lo stesso con la LAN G,con 5 bit di host_id.
Assegno poi uno degli indirizzi rimasti alla LAN L per cui, non avendo dispositivi collegati all'infuori dei 2 router, basta una maschera di rete /30.

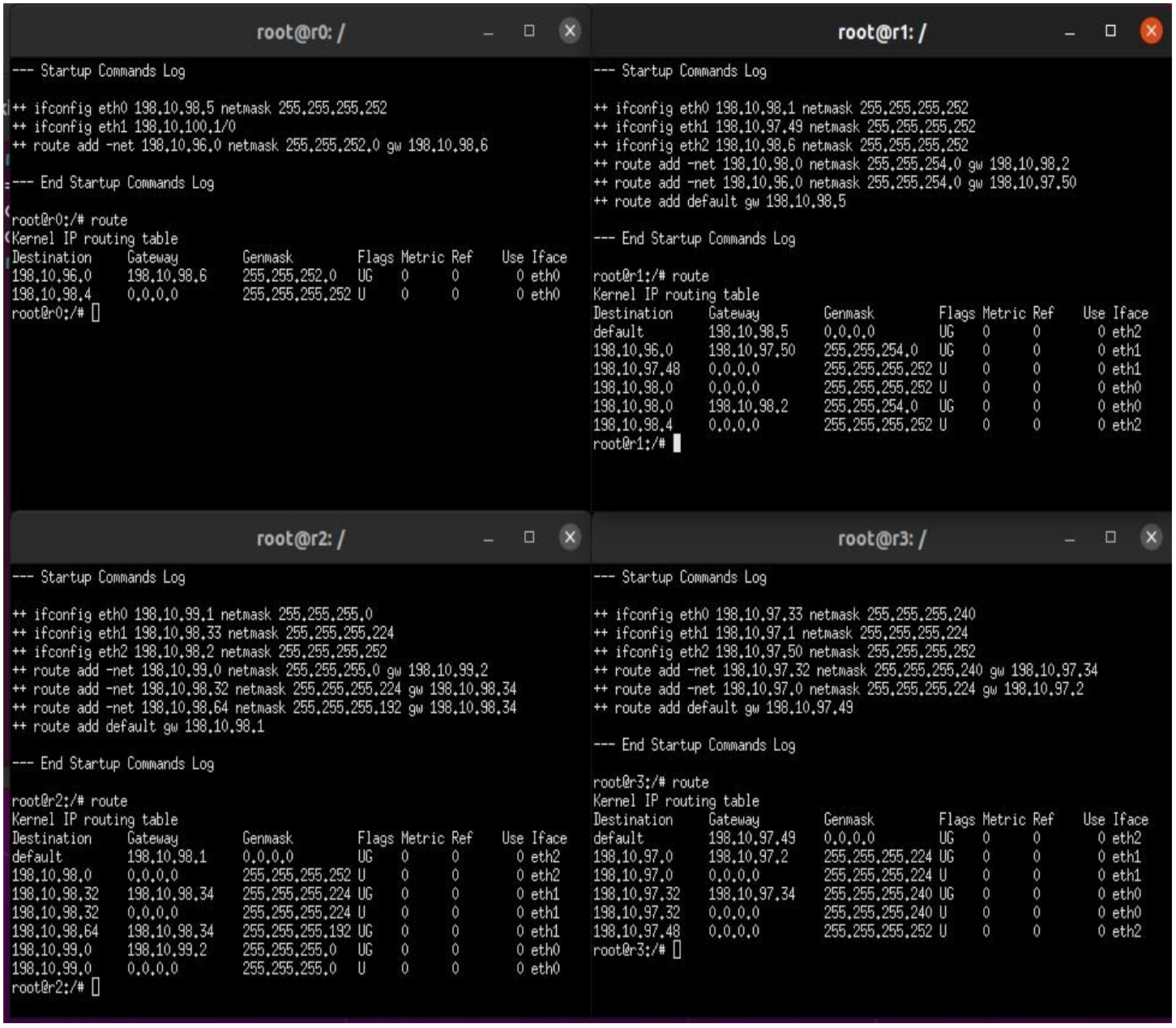
Per quanto riguarda il ramo destro collegato al router r1(quello collegato tramite eth1), ho ancora a disposizione il blocco .96 ed il .97.
Il blocco .96 lo assegno interamente alla LAN D, avendo questo 129 host per cui 2^7 indirizzi non sono sufficienti.
LAN E richiede 5 bit di host_id e quindi gli assegno i primi 32 indirizzi del blocco .97,mentre i 16 successivi li do alla LAN F avendo questa 4 bit di host_id.

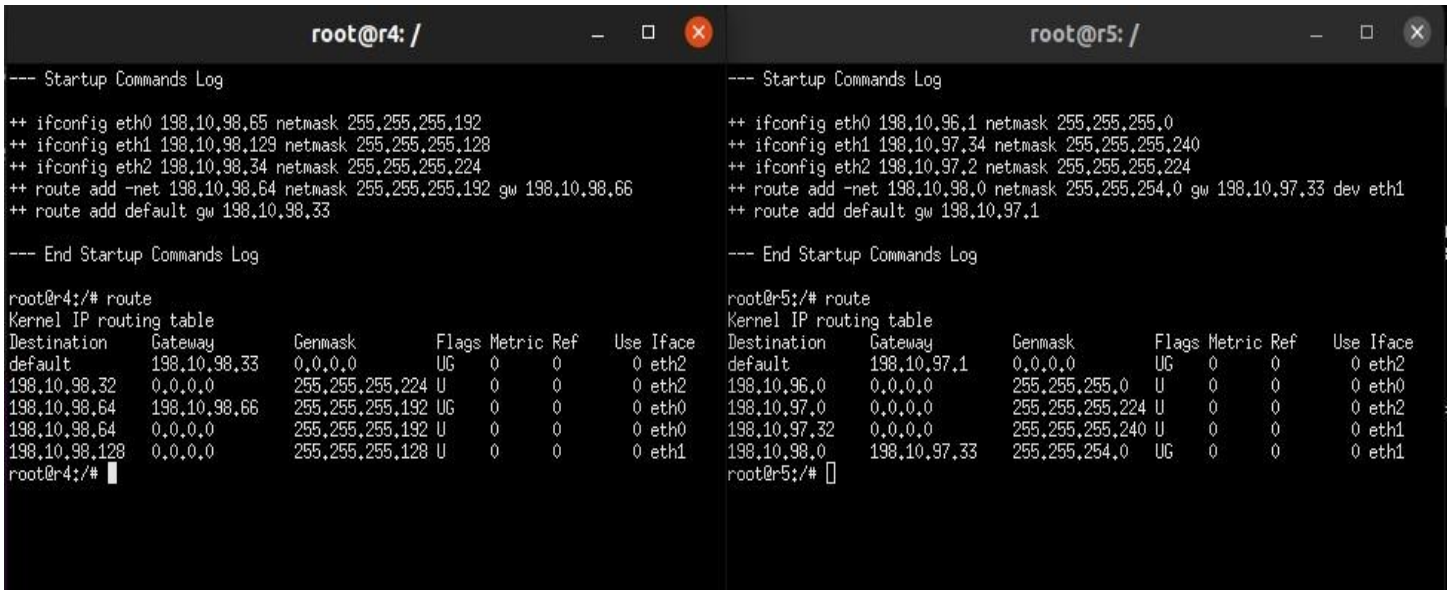
Infine, do i successivi indirizzi all'ultima lan rimasta, la LAN I, per cui bastano 4 indirizzi.

Dunque a questo punto,R1 vede tutta la parte collegata alla sua eth0 come un unico blocco con indirizzo 198.10.98.0/23, mentre quella collegata alla eth1 come il blocco 198.10.96.0/23.

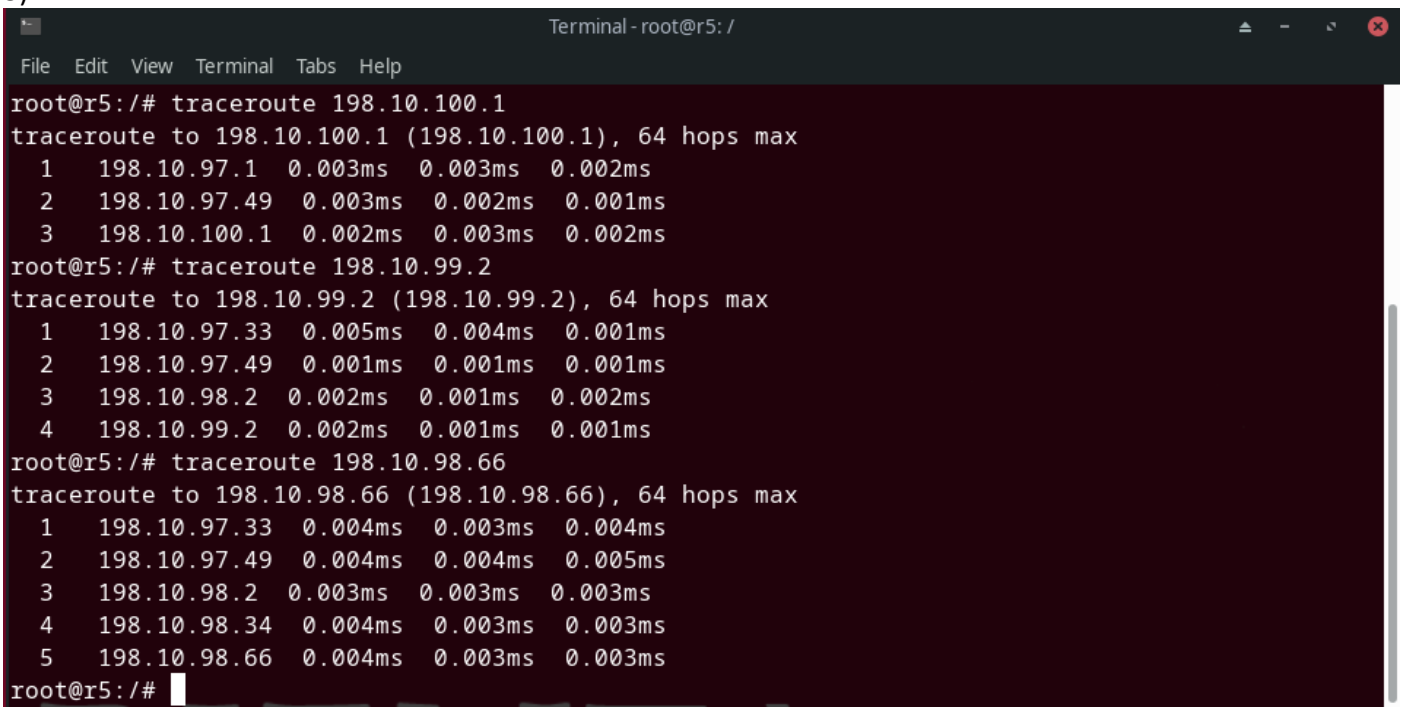
(Routing table r1 al punto successivo)

2)


root@r0: /
--- Startup Commands Log
++ ifconfig eth0 198.10.98.5 netmask 255.255.255.252
++ ifconfig eth1 198.10.100.1/0
++ route add -net 198.10.96.0 netmask 255.255.252.0 gw 198.10.98.6
--- End Startup Commands Log
root@r0:/# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
198.10.96.0 198.10.98.6 255.255.252.0 UG 0 0 0 eth0
198.10.98.4 0.0.0.0 255.255.255.252 U 0 0 0 eth0
root@r0:/#
root@r1: /
--- Startup Commands Log
++ ifconfig eth0 198.10.98.1 netmask 255.255.255.252
++ ifconfig eth1 198.10.97.49 netmask 255.255.255.252
++ ifconfig eth2 198.10.98.6 netmask 255.255.255.252
++ route add -net 198.10.98.0 netmask 255.255.254.0 gw 198.10.98.2
++ route add -net 198.10.96.0 netmask 255.255.254.0 gw 198.10.97.50
++ route add default gw 198.10.98.5
--- End Startup Commands Log
root@r1:/# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default 198.10.98.5 0.0.0.0 UG 0 0 0 eth2
198.10.96.0 198.10.97.50 255.255.254.0 UG 0 0 0 eth1
198.10.97.48 0.0.0.0 255.255.255.252 U 0 0 0 eth1
198.10.98.0 0.0.0.0 255.255.255.252 U 0 0 0 eth0
198.10.98.0 198.10.98.2 255.255.254.0 UG 0 0 0 eth0
198.10.98.4 0.0.0.0 255.255.255.252 U 0 0 0 eth2
root@r1:/#
root@r2: /
--- Startup Commands Log
++ ifconfig eth0 198.10.99.1 netmask 255.255.255.0
++ ifconfig eth1 198.10.98.33 netmask 255.255.255.224
++ ifconfig eth2 198.10.98.2 netmask 255.255.255.252
++ route add -net 198.10.99.0 netmask 255.255.255.0 gw 198.10.99.2
++ route add -net 198.10.98.32 netmask 255.255.255.224 gw 198.10.98.34
++ route add -net 198.10.98.64 netmask 255.255.255.192 gw 198.10.98.34
++ route add default gw 198.10.98.1
--- End Startup Commands Log
root@r2:/# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default 198.10.98.1 0.0.0.0 UG 0 0 0 eth2
198.10.98.0 0.0.0.0 255.255.255.252 U 0 0 0 eth2
198.10.98.32 198.10.98.34 255.255.255.224 UG 0 0 0 eth1
198.10.98.32 0.0.0.0 255.255.255.224 U 0 0 0 eth1
198.10.98.64 198.10.98.34 255.255.255.192 UG 0 0 0 eth1
198.10.99.0 198.10.99.2 255.255.255.0 UG 0 0 0 eth0
198.10.99.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
root@r2:/#
root@r3: /
--- Startup Commands Log
++ ifconfig eth0 198.10.97.33 netmask 255.255.255.240
++ ifconfig eth1 198.10.97.1 netmask 255.255.255.224
++ ifconfig eth2 198.10.97.50 netmask 255.255.255.252
++ route add -net 198.10.97.32 netmask 255.255.255.240 gw 198.10.97.34
++ route add -net 198.10.97.0 netmask 255.255.255.224 gw 198.10.97.2
++ route add default gw 198.10.97.49
--- End Startup Commands Log
root@r3:/# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default 198.10.97.49 0.0.0.0 UG 0 0 0 eth2
198.10.97.0 198.10.97.2 255.255.255.224 UG 0 0 0 eth1
198.10.97.0 0.0.0.0 255.255.255.224 U 0 0 0 eth1
198.10.97.32 198.10.97.34 255.255.255.240 UG 0 0 0 eth0
198.10.97.32 0.0.0.0 255.255.255.240 U 0 0 0 eth0
198.10.97.48 0.0.0.0 255.255.255.252 U 0 0 0 eth2
root@r3:/#

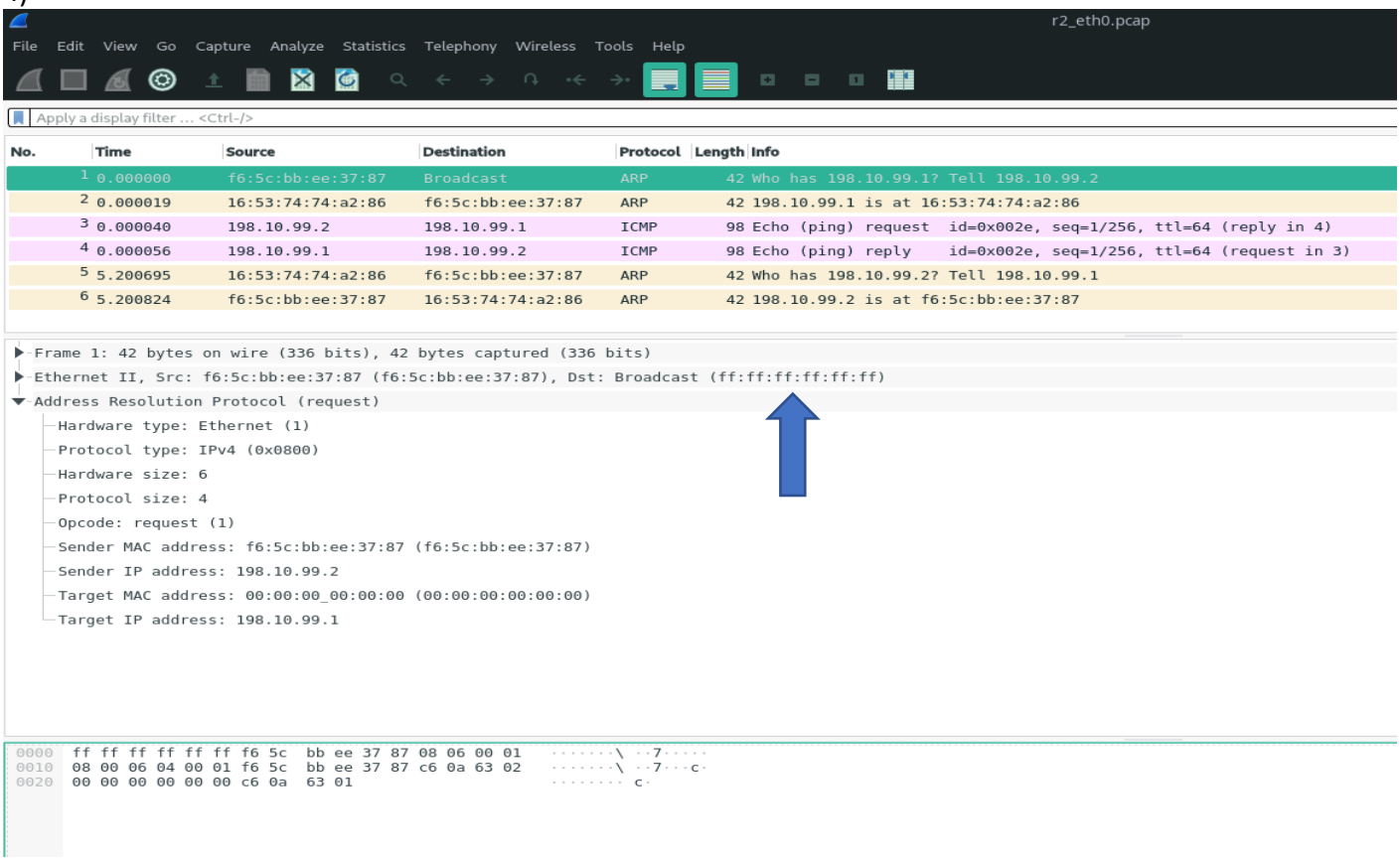

root@r4: /
--- Startup Commands Log
++ ifconfig eth0 198.10.98.65 netmask 255.255.255.192
++ ifconfig eth1 198.10.98.129 netmask 255.255.255.128
++ ifconfig eth2 198.10.98.34 netmask 255.255.255.224
++ route add -net 198.10.98.64 netmask 255.255.255.192 gw 198.10.98.66
++ route add default gw 198.10.98.33
--- End Startup Commands Log
root@r4:/# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default 198.10.98.33 0.0.0.0 UG 0 0 0 eth2
198.10.98.32 0.0.0.0 255.255.255.224 U 0 0 0 eth2
198.10.98.64 198.10.98.66 255.255.255.192 UG 0 0 0 eth0
198.10.98.64 0.0.0.0 255.255.255.192 U 0 0 0 eth0
198.10.98.128 0.0.0.0 255.255.255.128 U 0 0 0 eth1
root@r4:/#
root@r5: /
--- Startup Commands Log
++ ifconfig eth0 198.10.96.1 netmask 255.255.255.0
++ ifconfig eth1 198.10.97.34 netmask 255.255.255.240
++ ifconfig eth2 198.10.97.2 netmask 255.255.255.224
++ route add -net 198.10.98.0 netmask 255.255.254.0 gw 198.10.97.33 dev eth1
++ route add default gw 198.10.97.1
--- End Startup Commands Log
root@r5:/# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default 198.10.97.1 0.0.0.0 UG 0 0 0 eth2
198.10.96.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
198.10.97.0 0.0.0.0 255.255.255.224 U 0 0 0 eth2
198.10.97.32 0.0.0.0 255.255.255.240 U 0 0 0 eth1
198.10.98.0 198.10.97.33 255.255.254.0 UG 0 0 0 eth1
root@r5:/#

3)

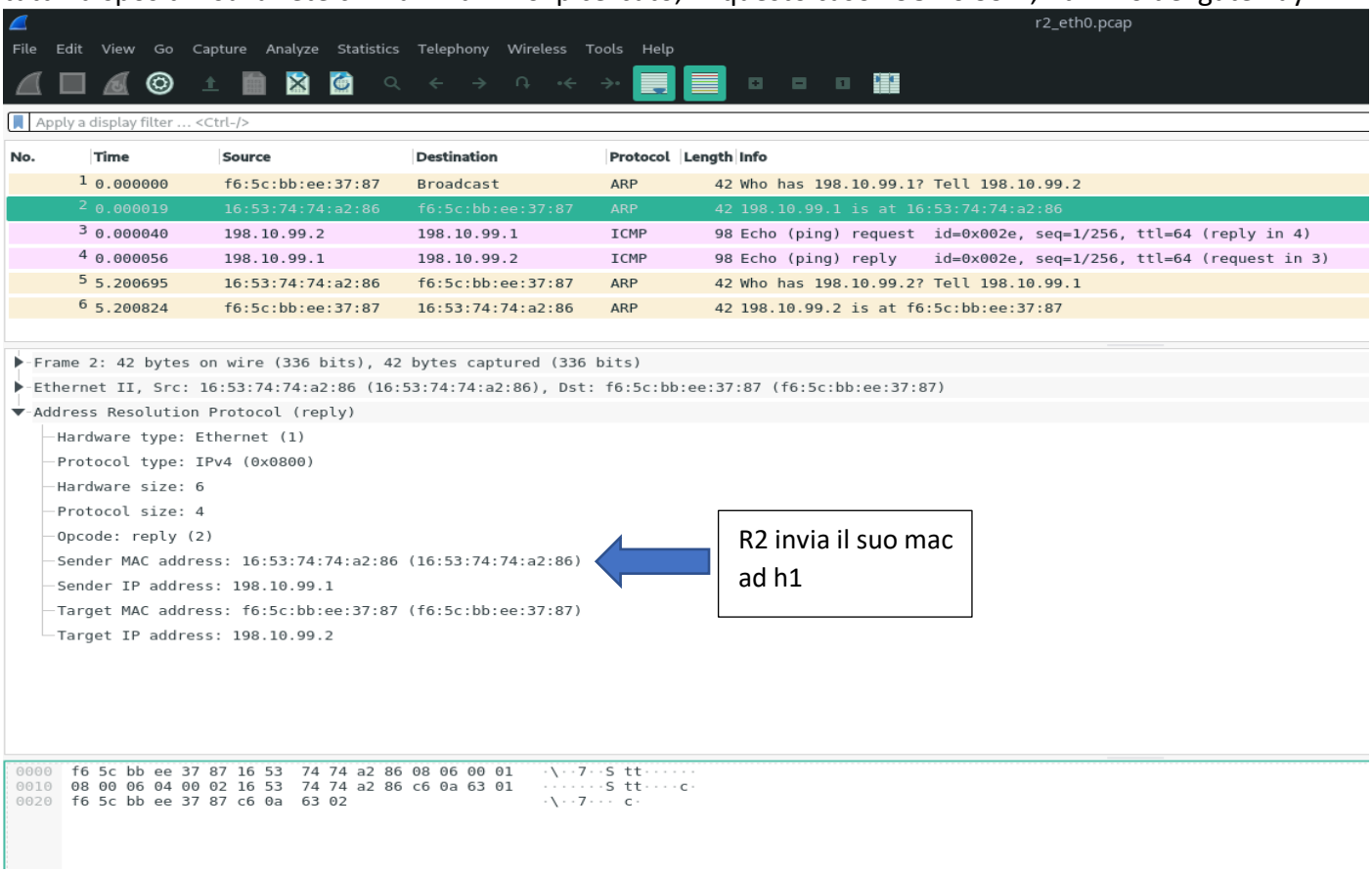


A partire da R5, il percorso verso i due host h1 e h2 passa per la Lan F (198.10.97.33 è l'indirizzo della scheda eth0 di r3 collegata alla Lan F), mentre il traffico verso internet passa per la Lan E (198.10.97.1)

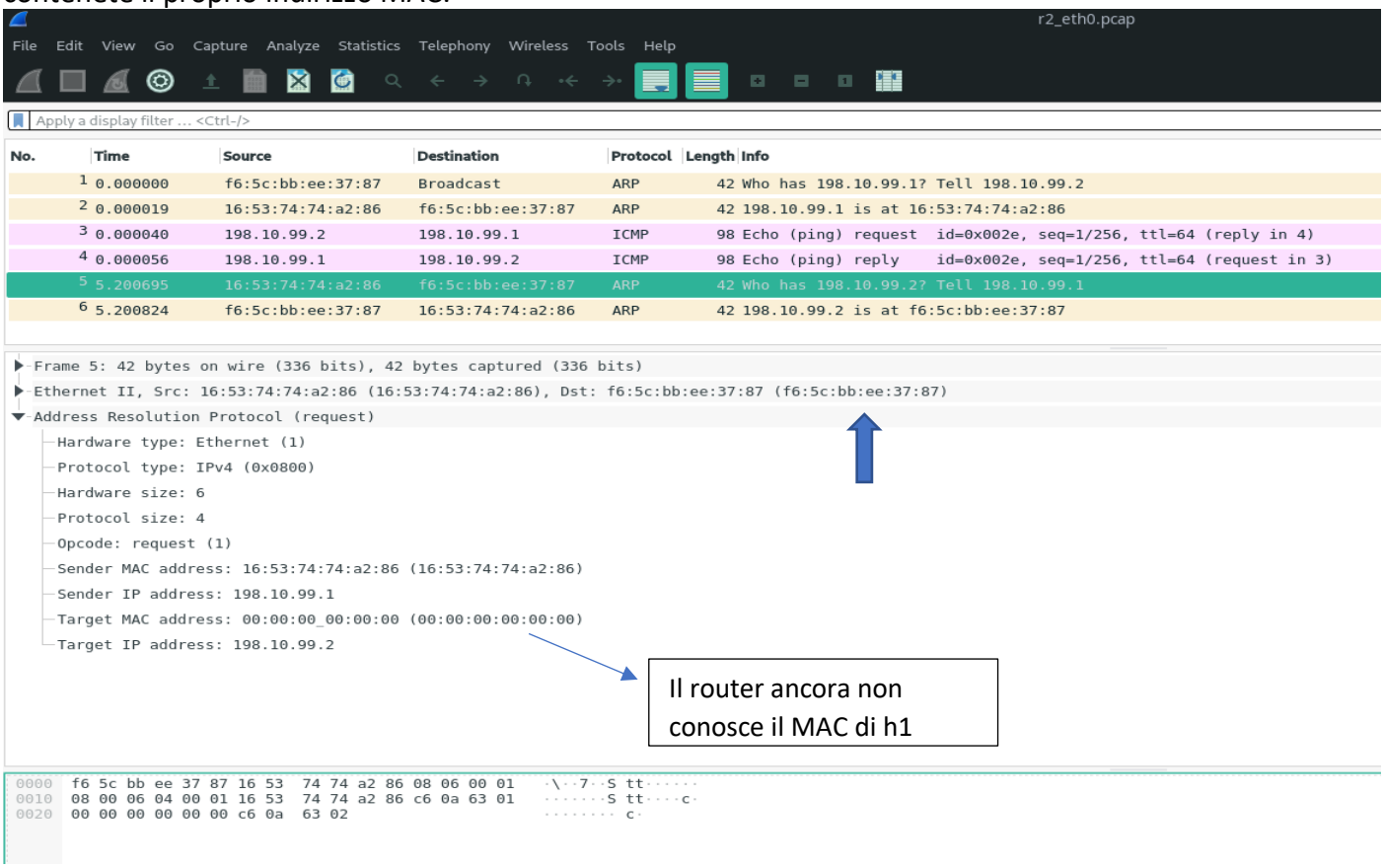
4)



H1 manda una arp request in broadcast(indirizzo MAC ff:ff:ff:ff:ff:ff, come segnalato in foto),e chiede a tutti i dispositivi sulla rete chi ha l'indirizzo ip cercato, in questo caso 198.10.99.1,indirizzo del gateway.

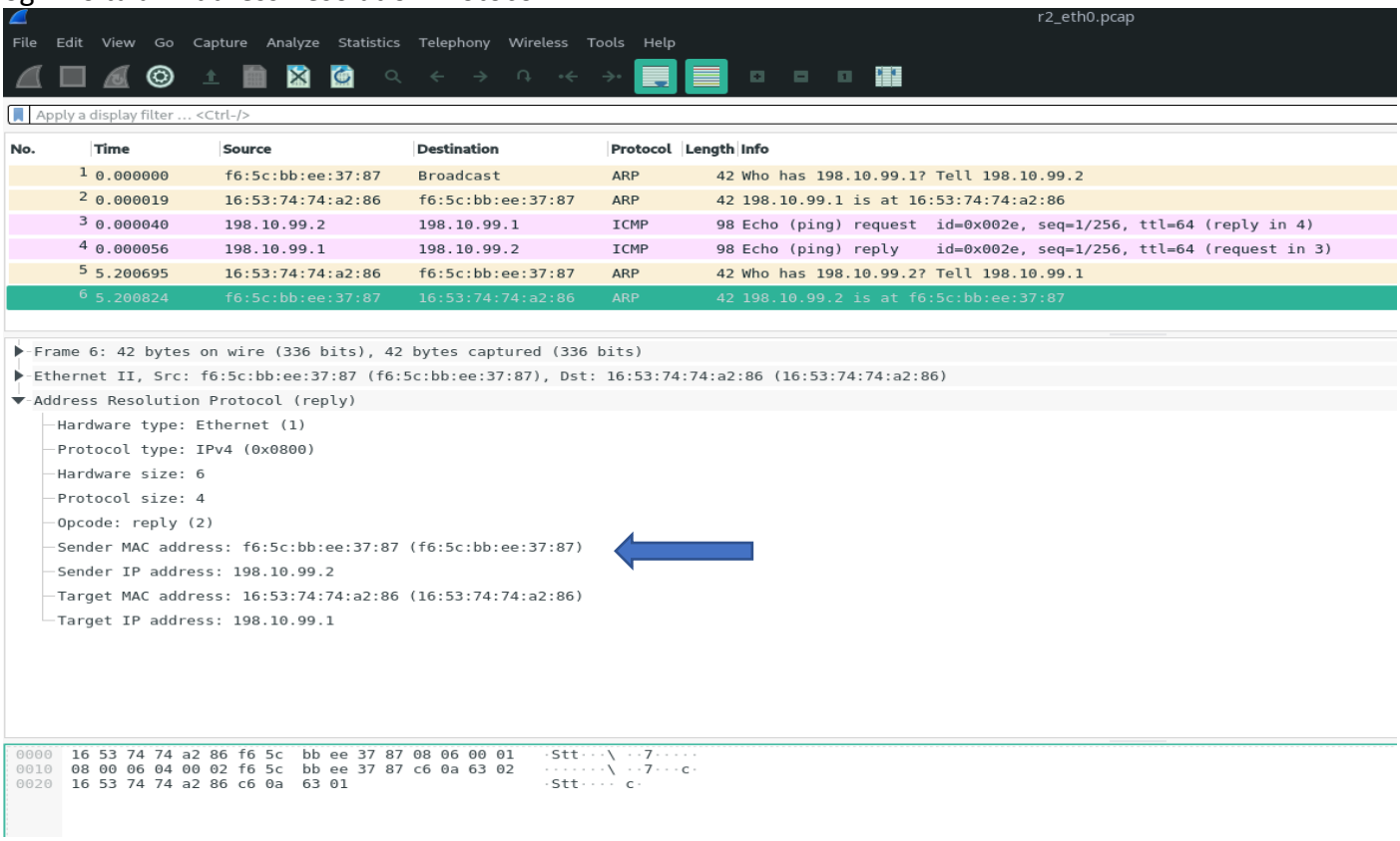


A questa arp request, risponde solo il dispositivo che ha l'indirizzo ip segnalato, con una arp reply in unicast contenete il proprio indirizzo MAC.



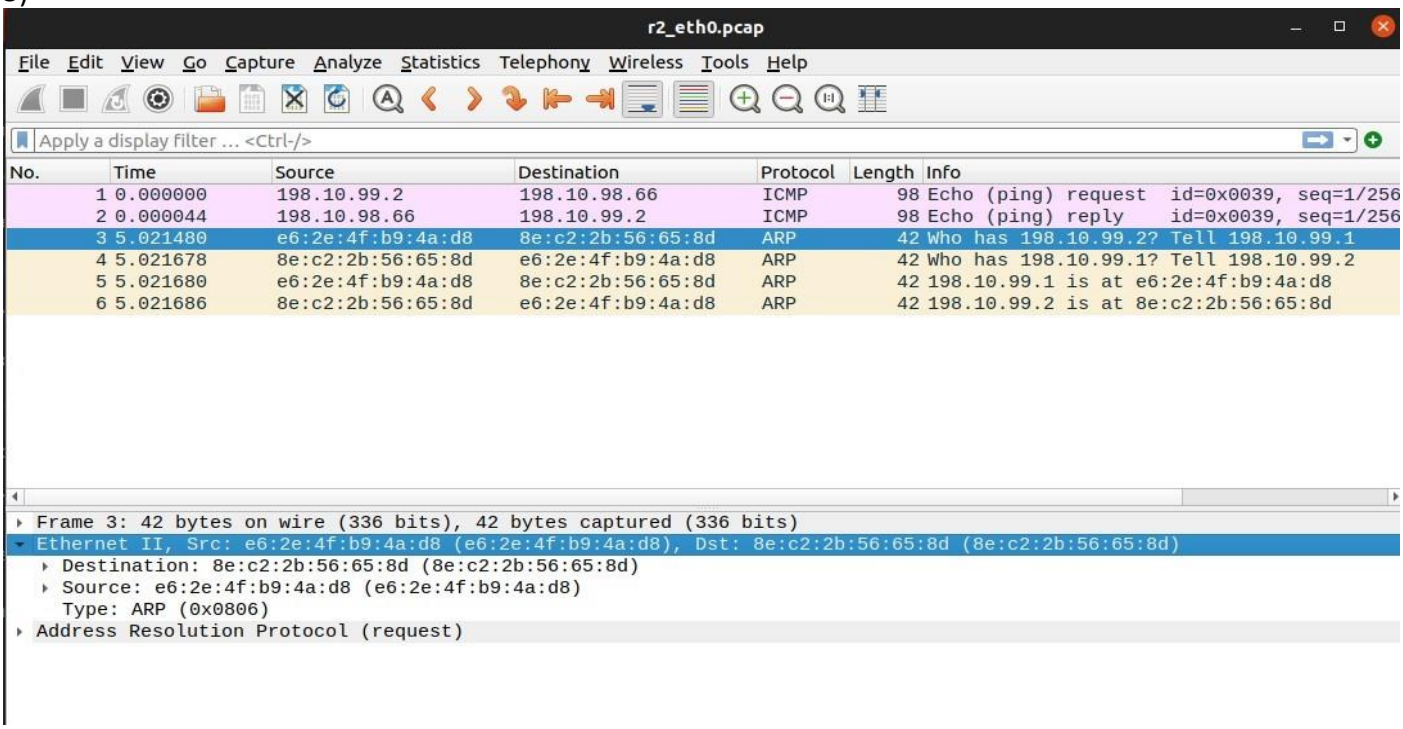
Successivamente, è il router a inviare una arp request ad h1 per conoscere il suo indirizzo MAC,stavolta direttamente in unicast poiché grazie alla precedente comunicazione,il gateway conosce già il suo indirizzo ip. Questo processo è il MAC learning, che permette al gateway di sapere anticipatamente i mac dei

dispositivi a se connessi per consegnare pacchetti provenienti dall'esterno della rete senza dover ricorrere ogni volta all'Address Resolution Protocol.

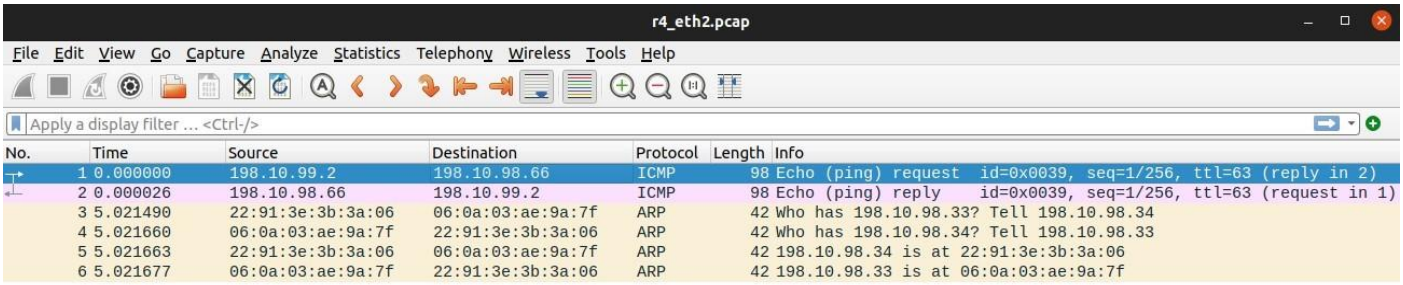


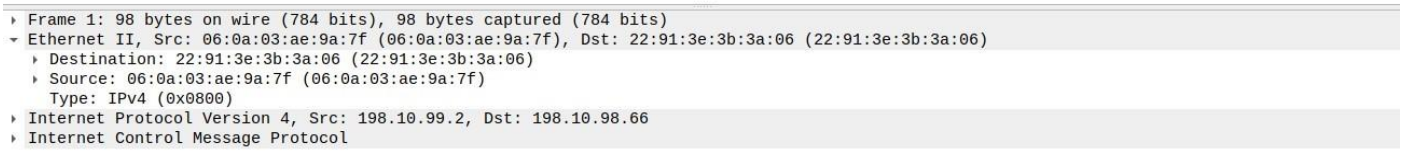
Infine, l'ultimo scambio avviene proprio con h1 che manda una arp reply al router, contenente il suo indirizzo MAC.

5)

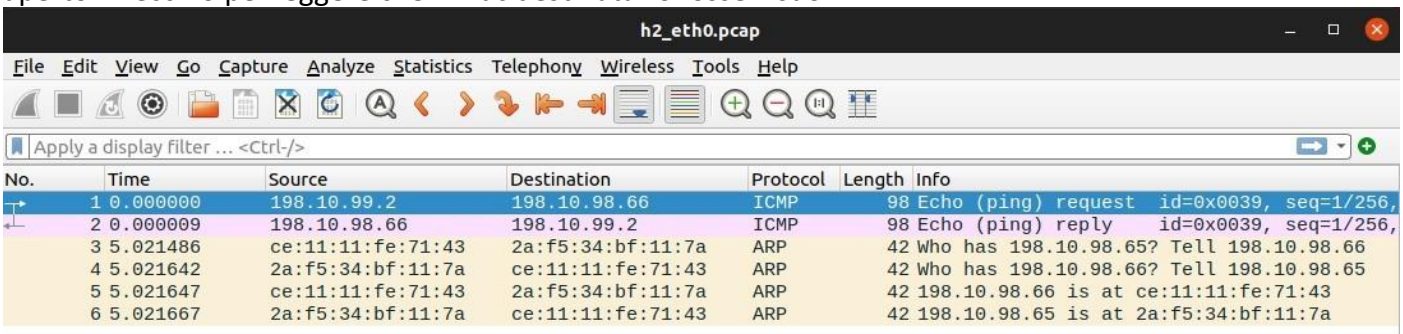


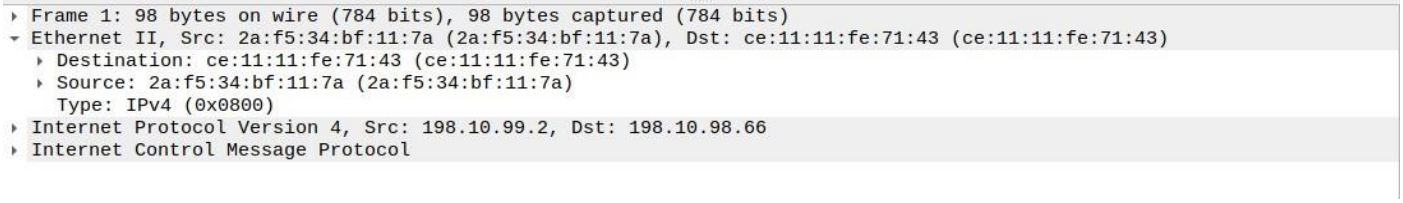
Per inviare una echo request ad h2, h1 trova l'indirizzo MAC del suo gateway tramite ARP ed invia a questo la echo request,incapsulata in un pacchetto ip, a sua volta incapsulato in un frame ethernet.
(protocollo arp come da punto 4).
Nello screen sono visibili le coordinate del frame Ethernet,composte dai mac divenuti noti dopo l'applicazione dell'arp.





R2 va quindi nella sua routing table per vedere come raggiungere la lan a cui è collegato h2,individuando l'ip del router del prossimo salto.
Per mandare a questo router r4 il frame, r2 chiede e riceve tramite arp il MAC di r4 a cui riesce quindi a rigirare il pacchetto, dopo averlo riincapsulato in un nuovo frame ethernet col MAC di r4, poichè aveva aperto il vecchio per leggere che il mac destinatario fosse il suo.

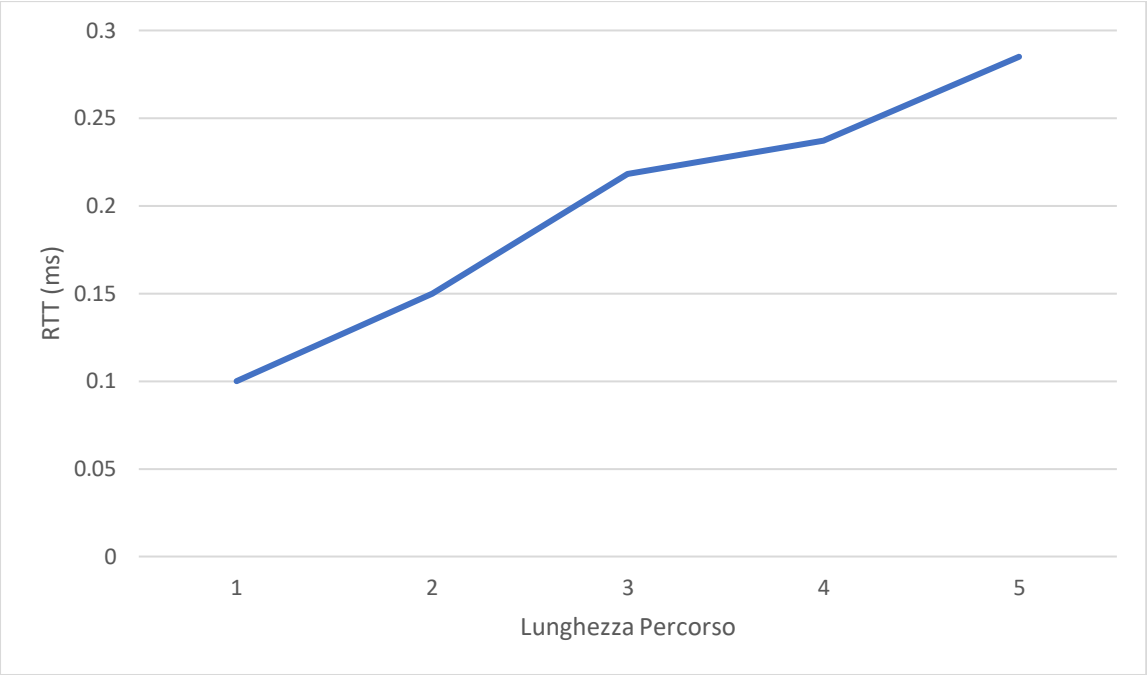




Infine,r4, che è connesso alla stessa lan di h2, chiede a questo il suo mac ancora una volta tramite arp e gli invia pacchetto ip incapsulato in un frame ethernet recante il suo MAC come destinazione.
A questo punto,h2 riuscirà ad aprire sia il frame che il pacchetto ip poiché indica il suo indirizzo e a leggere la echo request,alla quale risponderà con una echo reply.
A questo punto, inizia il processo inverso per recapitare a h1 questa echo reply.

6)
In generale, all'aumentare della lunghezza del percorso , e quindi dei dispositivi attraversati, aumenta anche il Round Trip Time; tuttavia a causa dei vari e non costanti ritardi presenti sulla rete, può capitare di rado che il RTT su un percorso più lungo sia più veloce di uno su un percorso più breve, come emerso dopo vari test.

I valori del rtt nel seguente grafico sono valori medi ottenuti da 10 ping consecutivi.



Lunghezza percorso	Rtt(ms)
1	0,1
2	0,15
3	0,218
4	0,237
5	0,285

È quindi evidente una correlazione similr-lineare tra rtt e lunghezza percorso, la quale viene in media rispettata, nonostante possano presentarsi singoli casi anomali, a sottolineare la bassa frequenza con cui accadono.