

## LABORATORIO DIDATTICO 7

### Tracciamento del percorso seguito dai pacchetti IP

Per tracciare il percorso seguito dai pacchetti IP, percorso inteso come router attraversati per raggiungere la destinazione, è possibile utilizzare il comando **tracert** <nome host (o indirizzo IP)> in ambiente Windows o **tracert** <nome host (o indirizzo IP)> in ambiente IOS Cisco e Linux.

L'impiego dei nomi host richiede che la risoluzione dei nomi in indirizzi IP sia attiva e che quindi sia correttamente configurato nei PC e negli apparati impiegati l'indirizzo IP di almeno un server DNS.

#### 1 Comando tracert in ambiente Windows

Il comando *tracert* (*trace route*) consente di rilevare (*trace*) gli instradamenti (*route*) che un pacchetto IP subisce nell'andare da una sorgente verso una destinazione.

Spesso l'effettuazione di un *tracert* consiste nell'invio in successione di sequenze di *Echo request* (di default sono tre alla volta) verso un certo indirizzo IP, con la particolarità che le *Echo request* sono inserite in pacchetti IP il cui TTL (*Time To Live*) parte dal valore *TTL=1* e, dopo ogni sequenza, viene incrementato di 1, fino a che non si raggiunge la destinazione o fino a quando si raggiunge il numero massimo di salti (*hop*) ammessi (di default sono 30).

Quando un pacchetto IP giunge a un router, quest'ultimo ne decrementa il valore del TTL di 1, prima di inoltrarlo verso il router successivo (*next hop*). Se però il valore del TTL del pacchetto in arrivo è 1, decrementandolo di 1 esso diviene *TTL = 0* (*Time To Live* scaduto) per cui il router scarta il pacchetto IP e invia un messaggio ICMP alla sorgente per comunicare che il pacchetto in transito è stato scartato, in quanto ha superato il tempo di vita (*Time To Live*) ammesso (*Time to live exceeded, in-transit*), come indicato nelle FIGURE 3 e 4.

Di conseguenza tramite il comando *tracert* è possibile conoscere quanti e quali router si attraversano per raggiungere una certa destinazione. Va però notato che, per motivi di sicurezza, i router (nonché i siti Web) possono essere configurati per non rispondere a *ping* o *tracert*, così da non essere individuati facilmente. Inoltre, di default, il comando *tracert* effettua la risoluzione inversa dell'indirizzo IP in nome di host, per mostrare oltre ai punti di passaggio e all'intervallo di tempo che intercorre tra le richieste di *Echo request* e le *Echo replay* (*Round Trip Time* o RTT), anche il nome degli host (router) attraversati per raggiungere la destinazione.

La risoluzione degli indirizzi in nomi host, che richiede tempo e può fallire (i router possono non avere nomi associati), può essere disabilitata con l'opzione *-d*.

Digitando solo *tracert* si ottiene la sintassi del comando

Si noti che la risposta comprende:

- il numero di router attraversati;
- il loro nome (se la risoluzione dell'indirizzo in nome ha avuto esito positivo);
- il ritardo fra andata e ritorno (*RTT*, *Round Trip Time*) di ciascuna *Echo request* con la relativa risposta, per cui di default sono tre valori di RTT;
- se la risposta a una *Echo request* non giunge entro un tempo prestabilito viene visualizzato un \* (si veda la FIGURA 5).

In FIGURA 1 si è tracciato tramite il comando **tracert www.garr.it** il percorso seguito dai pacchetti IP emessi da un PC per raggiungere il server che ospita il sito *www.garr.it*.

Il PC è connesso a una rete fisica corrispondente allo scenario del LABORATORIO DIDATTICO 6.

Il PC è collegato alla subnet 1, ha indirizzo IP 192.168.1.20 e gateway predefinito con indirizzo IP 192.168.1.1, costituito dall'interfaccia FastEthernet 0/0 del router A.

Si nota come vengono attraversati prima il router A (192.168.1.1), il router B (192.168.3.2), il router ADSL (192.168.4.254), tutti aventi indirizzi IP privati, e poi i router dei Provider Internet fino a giungere al server che ospita il sito indicato, il quale risponde con un messaggio ICMP di *Echo replay* per ogni *Echo request* ricevuta.

Va notato che, per motivi di sicurezza, alcuni router sono configurati per non rispondere ai messaggi ICMP inviati quando si effettua un *ping* o un *tracert*; in questo caso nell'output del comando *tracert* compaiono degli \*.

```

C:\> Prompt dei comandi

I:\Documents and Settings\onelio>tracert www.garr.it
Rilevazione instradamento verso www.garr.it [193.206.158.2] Nome risolto in indirizzo IP
su un massimo di 30 punti di passaggio:
  1  <1 ms    <1 ms    <1 ms    192.168.1.1 gateway predefinito per il PC
  2  11 ms    11 ms    11 ms    192.168.3.2
  3  11 ms    11 ms    11 ms    192.168.4.254 default gateway verso Internet
  4  20 ms    *          *          151.6.139.72
  5  19 ms    18 ms    18 ms    151.6.45.97
  6  75 ms    20 ms    18 ms    151.6.6.198 Indirizzi IP pubblici dei router su Internet
  7  20 ms    20 ms    20 ms    151.6.2.82
  8  32 ms    32 ms    32 ms    garr.mix-it.net [217.29.66.39]
  9  40 ms    39 ms    38 ms    r-mi2-rx2-mi2.mi2.garr.net [90.147.80.78]
 10  36 ms    31 ms    30 ms    rx2-mi2-rx2-rm2.rm2.garr.net [90.147.80.65]
 11  31 ms    31 ms    30 ms    rx2-rm2-ru-dir-l1.rm2.garr.net [193.206.138.210]
 12  34 ms    34 ms    34 ms    lx1.dir.garr.it [193.206.158.2]

Rilevazione completata.
I:\Documents and Settings\onelio>_

```

FIGURA 1 Esempio di tracciamento con il comando tracert.

## 1.1 Comando pathping

Simile al comando *tracert* è il comando *pathping*, il quale fornisce statistiche più dettagliate.

La sintassi del comando *pathping* si ottiene digitando semplicemente *pathping*. In FIGURA 2 è mostrato un esempio di output del comando *pathping* (*pathping www.garr.it*)

```

C:\Users\onelio>pathping www.garr.it

Traccia route verso www.garr.it [193.206.158.2]
su un massimo di 30 punti di passaggio:
 0 PC-ASUS [192.168.4.20]
 1 192.168.4.254
 2 * 151.6.139.72
 3 151.6.45.97
 4 151.6.6.206
 5 151.6.2.18
 6 garr.mix-it.net [217.29.66.39]
 7 r-mi2-r-rm2.rm2.garr.net [90.147.80.9]
 8 rx2-mi2-rx2-rm2.rm2.garr.net [90.147.80.65]
 9 rx2-rm2-ru-dir-l1.rm2.garr.net [193.206.138.210]
10 lx1.dir.garr.it [193.206.158.2]

Statistiche di calcolo per 250 secondi...
Hop RTT Da orig. a qui questo nodo/collegamento Indir.
0 0/ 100 = 0% 0/ 100 = 0% PC-ASUS [192.168.4.20]
1 1ms 0/ 100 = 0% 0/ 100 = 0% 192.168.4.254
2 17ms 0/ 100 = 0% 0/ 100 = 0% 151.6.139.72
3 9ms 0/ 100 = 0% 0/ 100 = 0% 151.6.45.97
4 12ms 0/ 100 = 0% 0/ 100 = 0% 151.6.6.206
5 10ms 0/ 100 = 0% 0/ 100 = 0% 151.6.2.18
6 22ms 0/ 100 = 0% 0/ 100 = 0% garr.mix-it.net [217.29.66.39]
7 22ms 0/ 100 = 0% 0/ 100 = 0% r-mi2-r-rm2.rm2.garr.net [90.147.80.9]
8 23ms 0/ 100 = 0% 0/ 100 = 0% rx2-mi2-rx2-rm2.rm2.garr.net [90.147.80.65]
9 21ms 0/ 100 = 0% 0/ 100 = 0% rx2-rm2-ru-dir-l1.rm2.garr.net [193.206.138.210]
10 20ms 1/ 100 = 1% 0/ 100 = 0% lx1.dir.garr.it [193.206.158.2]

Traccia completata.

```

FIGURA 2 Esempio di output del comando *pathping*

## 1.2 Analisi dei messaggi ICMP con Wireshark

Nelle FIGURE 3 e 4 si mostra l'analisi del traffico che deriva dall'effettuazione di un *tracert* da un PC avente indirizzo IP 10.0.0.46 verso il server che ospita il sito *www.garr.it*, effettuata sempre con il comando **tracert www.garr.it**.

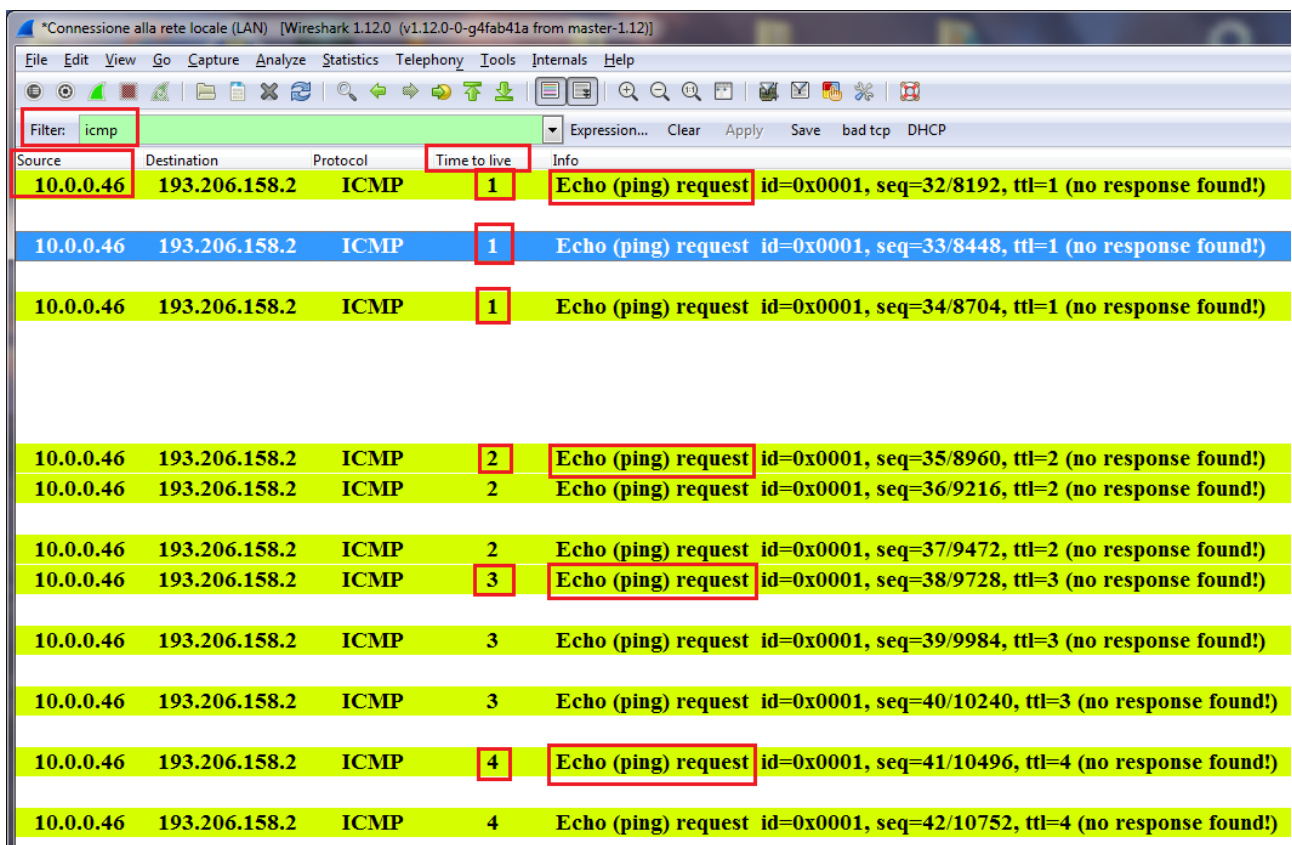
L'analisi è effettuata con l'analizzatore di protocollo *Wireshark* inserendo il *display filter icmp*.

La FIGURA 3 evidenzia<sup>1</sup> che:

- dal PC 10.0.0.46 (*Source*) vengono emessi volta per volta tre messaggi (PDU) ICMP di tipo *Echo request*, corrispondenti a un *ping*;
- le PDU ICMP sono trasportate da pacchetti IP aventi TTL (*Time To Live*) che, partendo da 1, viene via via incrementato.

La FIGURA 4 evidenzia<sup>2</sup> che:

- i router intermedi rispondono al PC 10.0.0.46 (*Destination*) inviando dei messaggi (PDU) ICMP di tipo *Time-To-Live exceeded (in transit)*, per indicare che hanno ricevuto dei pacchetti IP in transito con TTL=1, i quali avendo esaurito il TTL sono stati scartati;
- il server che ospita la destinazione (il sito *www.garr.it*), avente indirizzo IP 193.206.158.2, risponde con un messaggio ICMP di tipo *Echo replay* ad ogni messaggio ICMP di tipo *Echo request* che riceve.



Source	Destination	Protocol	Time to live	Info
10.0.0.46	193.206.158.2	ICMP	1	Echo (ping) request id=0x0001, seq=32/8192, ttl=1 (no response found!)
10.0.0.46	193.206.158.2	ICMP	1	Echo (ping) request id=0x0001, seq=33/8448, ttl=1 (no response found!)
10.0.0.46	193.206.158.2	ICMP	1	Echo (ping) request id=0x0001, seq=34/8704, ttl=1 (no response found!)
10.0.0.46	193.206.158.2	ICMP	2	Echo (ping) request id=0x0001, seq=35/8960, ttl=2 (no response found!)
10.0.0.46	193.206.158.2	ICMP	2	Echo (ping) request id=0x0001, seq=36/9216, ttl=2 (no response found!)
10.0.0.46	193.206.158.2	ICMP	2	Echo (ping) request id=0x0001, seq=37/9472, ttl=2 (no response found!)
10.0.0.46	193.206.158.2	ICMP	3	Echo (ping) request id=0x0001, seq=38/9728, ttl=3 (no response found!)
10.0.0.46	193.206.158.2	ICMP	3	Echo (ping) request id=0x0001, seq=39/9984, ttl=3 (no response found!)
10.0.0.46	193.206.158.2	ICMP	3	Echo (ping) request id=0x0001, seq=40/10240, ttl=3 (no response found!)
10.0.0.46	193.206.158.2	ICMP	4	Echo (ping) request id=0x0001, seq=41/10496, ttl=4 (no response found!)
10.0.0.46	193.206.158.2	ICMP	4	Echo (ping) request id=0x0001, seq=42/10752, ttl=4 (no response found!)

FIGURA 3 Successione di Echo request ICMP incapsulate in pacchetti IP aventi Time To Live (TTL) crescente

<sup>1</sup> Le risposte sono state nascoste impiegando una Coloring Rule (Menu View) con il seguente filtro **ip.dst == 10.0.0.46** e usando il bianco sia come Background color sia come Foreground color. I valori di TTL sono stati visualizzati andando nella finestra Packet Details, selezionando il campo TTL dell'header IP, cliccando col tasto destro del mouse e selezionando **Apply as Column**

<sup>2</sup> Le risposte sono state nascoste impiegando una Coloring Rule (Menu View) con il seguente filtro **ip.src == 10.0.0.46** e usando il bianco sia come Background color sia come Foreground color.

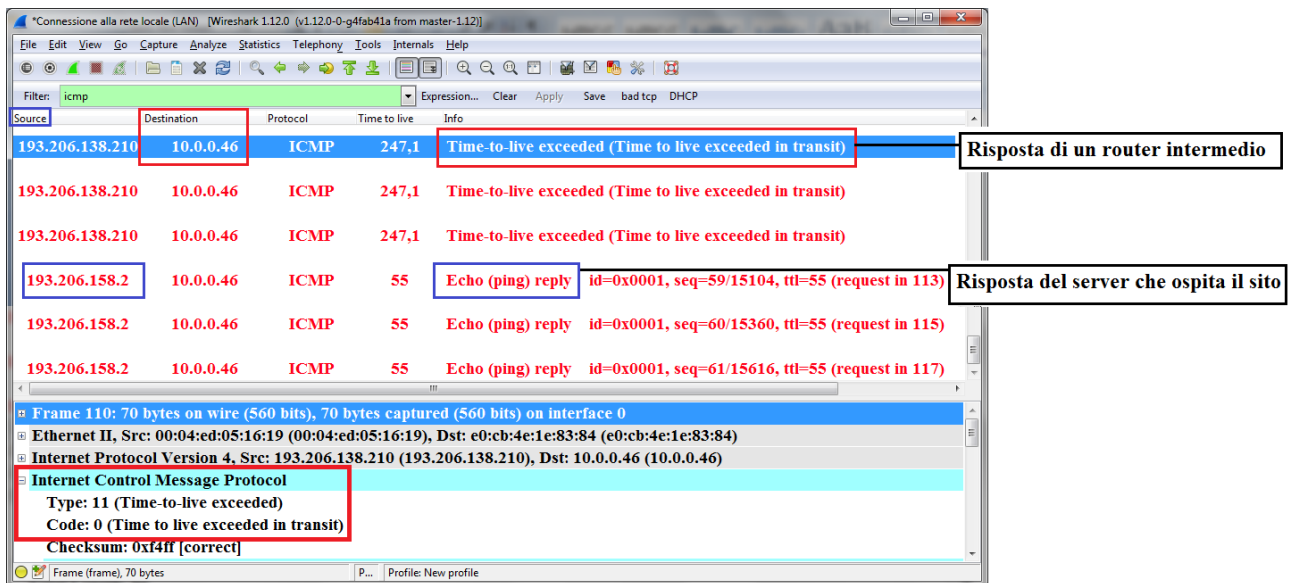


FIGURA 4 Risposte dei router intermedi indicanti che il TTL entrante era 1 per cui il pacchetto IP è stato scartato.

## 2 Comando traceroute in ambiente Linux (distribuzione Ubuntu)

In ambiente Linux è possibile effettuare il tracciamento dei percorsi con il comando **traceroute**, dato dal programma **traceroute**.

Il programma **traceroute** può essere installato da *Gestore di pacchetti* oppure con il comando (FIGURA 5) **sudo apt-get install traceroute**.

Con il comando **man traceroute** si ottiene l'aiuto fornito dalla pagina di manuale del comando stesso. Per effettuare il tracciamento del percorso in modo analogo al **tracert** è possibile aprire un terminale e digitare il comando (FIGURA 6): **sudo traceroute -I -w 20 www.garr.it**

Le opzioni inserite del comando **traceroute** hanno le seguenti funzioni:

- l'opzione **-I** indica che deve essere impiegato il protocollo ICMP in modo analogo a quanto visto per l'ambiente Windows (in ambiente Linux è possibile utilizzare anche il protocollo UDP);
- l'opzione **-w** fissa il tempo di attesa di ciascuna risposta, in secondi.

```

onelio@onelio-desktop: ~
onelio@onelio-desktop:~$ traceroute www.garr.it
Il programma "traceroute" può essere trovato nei seguenti pacchetti:
* inetutils-traceroute
* traceroute
Provare: sudo apt-get install <PACCHETTO SELEZIONATO>
onelio@onelio-desktop:~$ ^C
onelio@onelio-desktop:~$ sudo apt-get install traceroute
[sudo] password for onelio:
Lettura elenco dei pacchetti... Fatto
Generazione albero delle dipendenze
Lettura informazioni sullo stato... Fatto
I seguenti pacchetti NUOVI saranno installati:
  traceroute
0 aggiornati, 1 installati, 0 da rimuovere e 0 non aggiornati.
È necessario scaricare 51,5 kB di archivi.

```

FIGURA 5 Installazione del programma **traceroute**.

```
onelio@onelio-desktop: ~  
onelio@onelio-desktop:~$ sudo traceroute -I -w 20 www.garr.it  
traceroute to www.garr.it (193.206.158.2), 30 hops max, 60 byte packets  
 1  10.0.0.1 (10.0.0.1)  2.899 ms  2.893 ms  2.910 ms  
 2  * * *  
 3  151.6.45.105 (151.6.45.105)  15.332 ms  16.622 ms  18.862 ms  
 4  151.6.6.202 (151.6.6.202)  20.680 ms  21.780 ms  23.240 ms  
 5  151.6.2.18 (151.6.2.18)  26.303 ms  151.6.2.78 (151.6.2.78)  27.570 ms  29.12  
 2 ms  
 6  garr.mix-it.net (217.29.66.39)  44.574 ms  23.917 ms  25.690 ms  
 7  r-mi2-rx2-mi2.mi2.garr.net (90.147.80.78)  25.093 ms  28.798 ms  r-mi2-r-rm2.  
rm2.garr.net (90.147.80.9)  28.254 ms  
 8  rx2-rm2-r-rm2.rm2.garr.net (90.147.80.58)  30.200 ms  31.532 ms  rx2-mi2-rx2-  
rm2.rm2.garr.net (90.147.80.65)  36.089 ms  
 9  rx2-rm2-ru-dir-l1.rm2.garr.net (193.206.138.210)  36.009 ms  37.391 ms  39.0  
85 ms  
10  lx1.dir.garr.it (193.206.158.2)  40.347 ms  42.991 ms  20.485 ms  
onelio@onelio-desktop:~$
```

FIGURA 6 Effettuazione di un traceroute in ambiente Linux.

### 3. Comando traceroute in ambiente IOS Cisco

Per effettuare un traceroute da linea di comando su un apparato Cisco, come per esempio il Router A del LABORATORIO DIDATTICO 6, è necessario configurare l'indirizzo IP di un server DNS (nell'esempio 192.168.0.250), con il comando **ip name-server <indirizzo IP>**, e utilizzare il comando **traceroute <nome host o indirizzo IP>**, FIGURA 7.

```
Router-sede-A  
Physical Config CLI  
IOS Command Line Interface  
Router-A>enable  
Router-A#config term  
Enter configuration commands, one per line. End with CNTL/Z.  
Router-A(config)#ip name-server 192.168.0.250  
Router-A(config)#end  
Router-A#  
%SYS-5-CONFIG_I: Configured from console by console  
Router-A#copy run start  
Destination filename [startup-config]?  
Building configuration...  
[OK]  
Router-A#traceroute intranet-lab-tele  
Type escape sequence to abort.  
Tracing the route to 192.168.0.250  
 1  192.168.3.2      1 msec    0 msec    0 msec  
 2  192.168.0.250    0 msec    0 msec    1 msec  
Router-A#
```

FIGURA 7 Esempio di configurazione di un server DNS e di effettuazione di un traceroute in ambiente IOS Cisco.