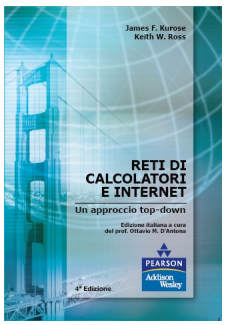
Laboratorio Wireshark: ICMP



Versione 2.1 italiano

Traduzione italiana di G. Amato, 2011.
Modifiche e adattamenti per il CLEII di G. Amato, 2011.

In questo laboratorio esploreremo vari aspetti del protocollo ICMP:

- messaggi ICMP generati dal programma Ping;
- messaggi ICMP generati dal programma Traceroute;
- formato e contenuti di un messaggio ICMP.

Prima di iniziare il laboratorio, siete incoraggiati a rivedere il materiale riguardante ICMP nel libro di testo (Sezione 4.4.3 nella 4ª edizione). Presentiamo questo laboratorio nel contesto del sistema operativo Unix/Linux. Comunque, è facile adattare il laboratorio ad un ambiente Microsoft Windows.

1. ICMP e Ping

Iniziamo la nostra avventura con ICMP catturando i pacchetti generati dal programma Ping. Forse ricorderete che il programma Ping è un semplice strumento che consente a chiunque (per esempio, all'amministratore di rete) di verificare se un sistema terminale è vivo o no. Il programma Ping sul sistema terminale sorgente invia un pacchetto verso l'indirizzo IP destinazione; se la destinazione è viva, il suo sistema operativo risponde inviando all'indietro un pacchetto. Come forse avrete già indovinato (dato che questo laboratorio riguarda ICMP), entrambi questi pacchetti sono pacchetti ICMP.

Fate quanto segue[1]:

- Iniziate l'avventura aprendo la linea di comando (shell) di Unix/Linux;
- Eseguite Wireshark, e iniziate la cattura dei pacchetti
- Dal finestra della shell, dare il comando

  ```
  ping -c 10 gaia.cs.umass.edu
  ```

  L'opzione "`-c 10`" indica che devono essere inviati 10 pacchetti ping.
- Quando il programma ping termina, interrompere la cattura dei pacchetti.

Alla fine dell'esperimento, la vostra finestra della shell dovrebbe assomigliare a quella di Figura 1. In questo esempio, il programma ping sorgente è in Massachusetts (USA) e la destinazione è ad Hong Kong. Da questa finestra vediamo che il programma ping sorgente ha inviato 10 pacchetti e ricevuto 10 risposte. Notare anche che, per ogni risposta, il ping sorgente calcola il round-trip time (RTT) che, per questi pacchetti, è in media di 375 ms.

---

1 Se non siete in grado di eseguire Wireshark dal vivo su di un computer, potete scaricare il file http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip ed estrarre il file *ICMP-ethereal-trace-1*. Le tracce in questo file zip sono state raccolte da Wireshark in esecuzione su uno dei computer degli autori, eseguendo i passi indicati in questo laboratorio. Una volta scaricata la traccia, la potete caricare ed esaminare in Wireshark usando il menù a discesa *File*, scegliendo *Open*, e selezionando il file *ICMP-ethereal-trace-1*. Potete quindi usare questa traccia per rispondere alle domande che seguono.

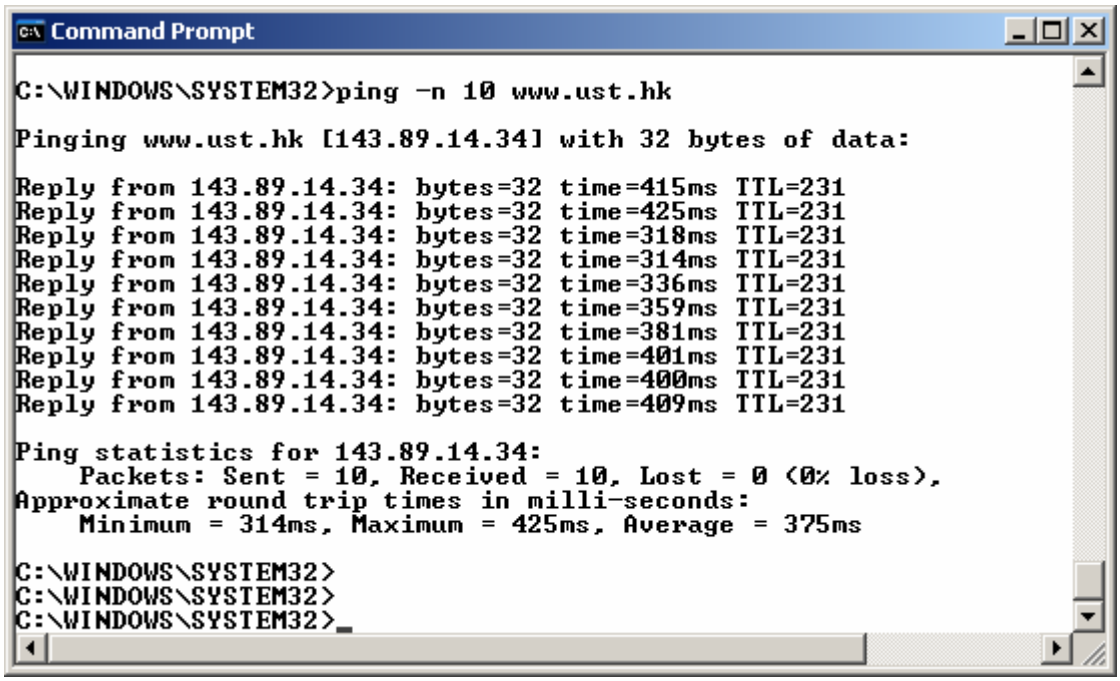


Figura 1: Finestra dei comandi dopo aver eseguito il comando Ping.

La Figura 2 è uno screenshot dell'output di Wireshark, dopo aver immesso "icmp" come filtro per la visualizzazione. Notate che l'elenco dei pacchetti mostra 20 pacchetti: le 10 richieste Ping inviate dalla sorgente e le 10 risposte Ping ricevute dalla sorgente. Inoltre notate che l'indirizzo IP sorgente è un indirizzo privato (dietro un NAT) della forma 192.168/12; l'indirizzo IP destinazione è quello del server Web all'HKUST. Ora concentriamoci sul primo pacchetto (inviato dal client); in Figura 2 la finestra dei dettagli fornisce varie informazioni. Vediamo che il datagramma IP dentro il pacchetto ha numero di protocollo 01, che è il numero di protocollo per ICMP. Questo significa che il payload del datagramma IP è un pacchetto ICMP.

La Figura 3 si focalizza sullo stesso pacchetto ICMP ma espandendo, nella finestra dei dettagli, le informazioni sul protocollo ICMP. Osservate che questo pacchetto ICMP ha Type 8 e Code 0 – si tratta quindi di un cosiddetto pacchetto ICMP "echo request". (Vedi Figure 4.23 nel libro di testo.) Notate anche che questo pacchetto ICMP contiene una checksum, un identificatore e un numero di sequenza.

Cosa consegnare:

Dovreste consegnare uno screenshot della finestra della shell simile alla Figura 1 di cui sopra. Quando possibile, nel rispondere alle domande, dovreste consegnare una stampa dei pacchetti all'interno della traccia che avete usato per rispondere. Annotate la stampa per spiegare la risposta. Per stampare un pacchetto usate *File → Print*, scegliete *Selected packet only*, scegliete *Packet summary line*, e selezionate la minima quantità di dettagli che è necessaria per rispondere alle domande.

Dovreste rispondere alle seguenti domande:

1. Qual è l'indirizzo IP del vostro host? Qual è l'indirizzo IP dell'host di destinazione?
2. Perché un pacchetto ICMP non ha numeri di porta sorgente e destinazione?
3. Esaminare una delle richieste ping inviate dal vostro host. Quali sono i valori dei campi type e code del pacchetto ICMP? Da quanti byte sono formati i campi checksum, numero di sequenza e identificatore?

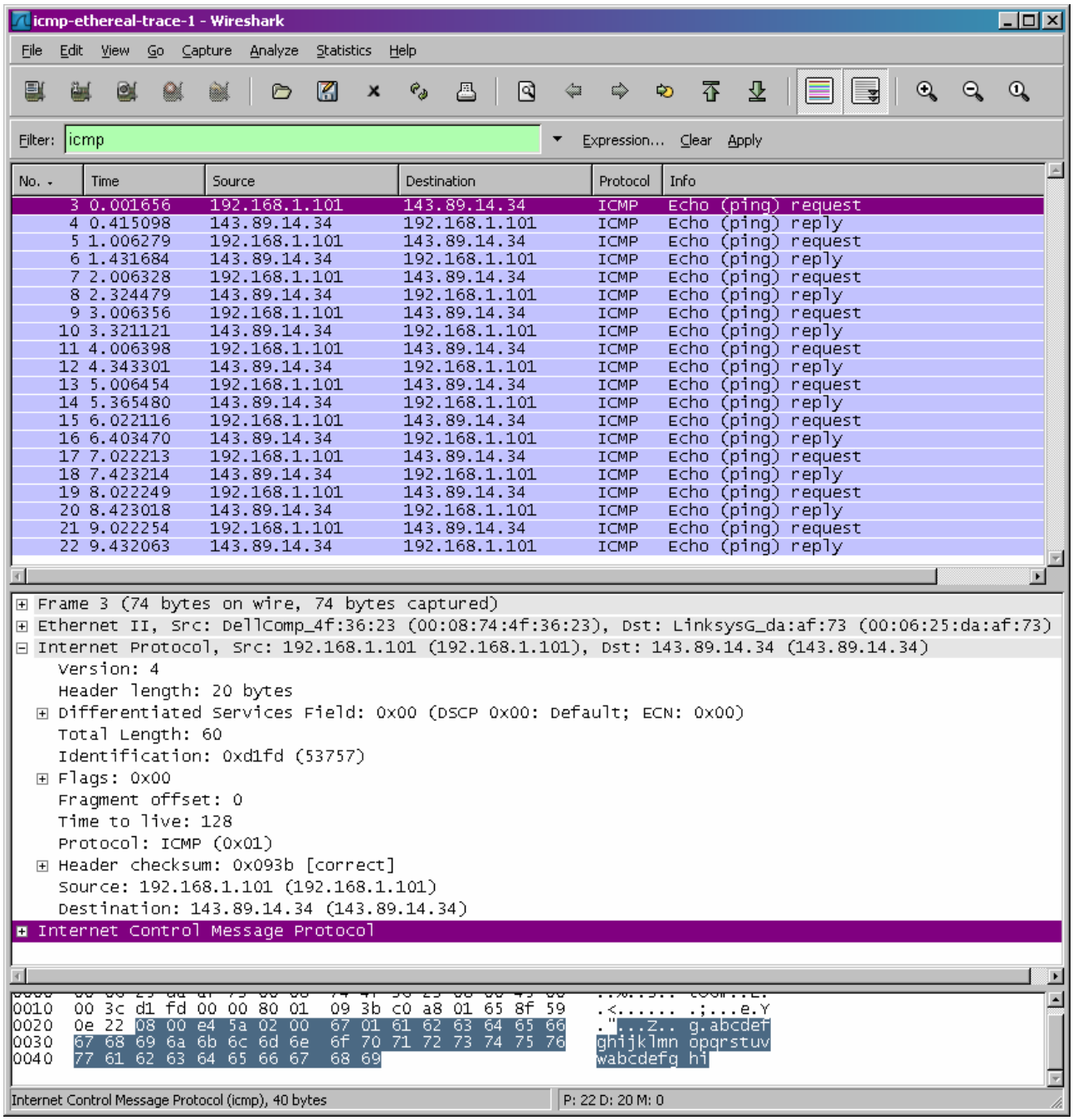


Figura 2: Output di Wireshark per il programma Ping con la sezione Internet Protocol espansa.

4. Esaminare il corrispondente pacchetto di risposta. Quali sono i valori dei campi type e code? Quali altri campi ha questo pacchetto ICMP? Da quanti byte sono formati i campi checksum, numero di sequenza e identificatore?

2. ICMP e Traceroute

Continuiamo adesso la nostra avventura all'esplorazione di ICMP catturando i pacchetti generati dal programma Traceroute. Forse ricorderete che il programma Traceroute può essere usato per scoprire il percorso che un pacchetto segue dalla sorgente alla destinazione. Traceroute è discusso nella Sezione 1.6 e Seziona 4.4 del libro di testo.

Traceroute è implementato in maniera differente su Unix/Linux e Windows. Su Unix/Linux, la

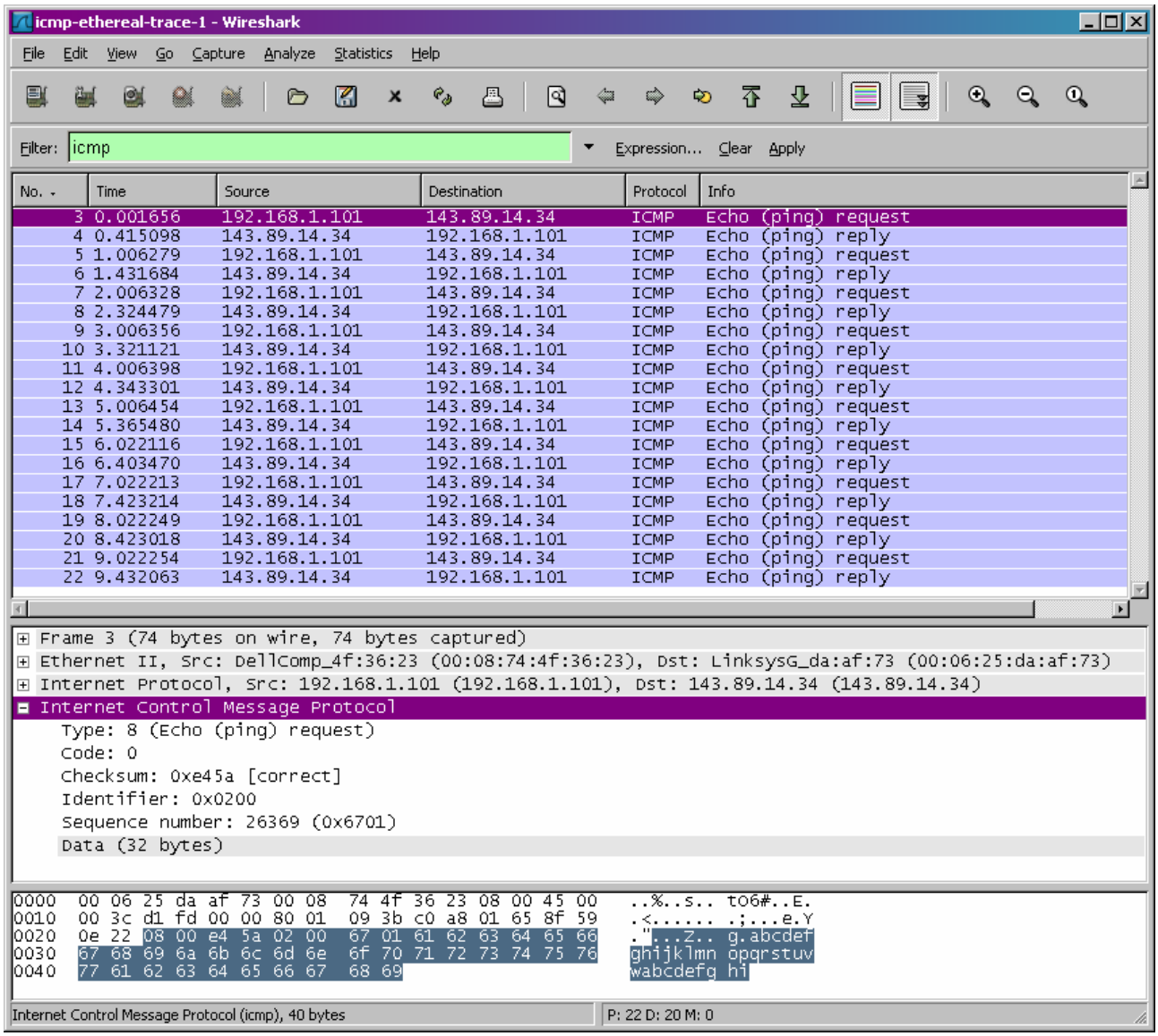


Figura 3: Output di Wireshark per il programma Ping con la sezione ICMP espansa.

sorgente invia una serie di pacchetti UDP verso la destinazione, usando una porta destinazione casuale difficilmente utilizzata; in Windows, la sorgente manda una serie di pacchetti ICMP verso la destinazione. Per entrambi i sistemi operativi, il programma invia il primo pacchetto con TTL=1, il secondo pacchetto con TTL=2 e così via. Ricordate che un router decrementerà il valore TTL di un pacchetto. Quando un pacchetto arriva al router col valore TTL=1, il router invia un pacchetto di errore ICMP verso la sorgente.

Su Windows è possibile usare il programm nativo *tracert*. Una versione shareware molto più accattivante di Traceroute è *pingplotter* (www.pingplotter.com). Useremo *pingplotter* nel laboratorio Wireshark su IP perché fornisce funzionalità addizionali di cui avremo bisogno.

Per i sistemi Linux, il programma *traceroute* nativo è più che adeguato.

ATTENZIONE! Il firewall dell'ateneo non effettua l'inoltro di pacchetti UDP, per cui traceroute su Unix/Linux non funziona dall'aula informatica. Si può utilizzare al suo posto il progamma *mtr* (abbreviazione di *My Traceroute*) installato di default nei sistemi Ubuntu Linux, che utilizza pacchetti ICMP echo in maniera simile a *tracert* su Windows.

Fate quanto segue[2]:

2 Se non siete in grado di eseguire Wireshark dal vivo su di un computer, potete scaricare il file http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip ed estrarre il file *ICMP-ethereal-trace-2*. Le tracce in

- Iniziate aprendo una finestra con la shell del sistema operativo.
- Fate partire Wireshark e iniziate la cattura dei pacchetti.
- Dal finestra della shell, dare il comando

  ```
  traceroute gaia.cs.umass.edu
  ```

  In aula informatica, usare invece

  ```
  mtr -c 1 gaia.cs.umass.edu
  ```

  L'opzione -c 1 indica ad mtr di inviare un'unica sequenza di pacchetti ICMP e poi fermarsi. Normalmente mtr continua a inviare pacchetti ICMP fino a quando viene interrotto manualmente.
- Quando il programma traceroute termina, interrompere la cattura dei pacchetti.

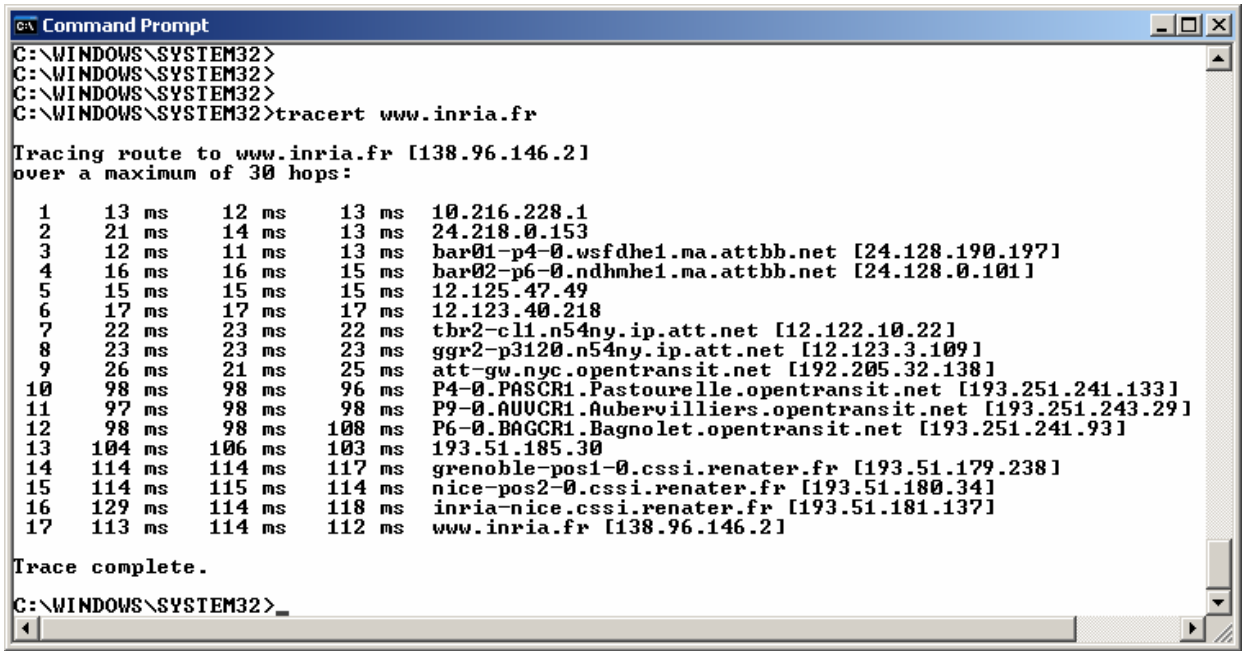


Figura 4: Finestra della shell con i risultati del comando Traceroute

Alla fine dell'esperimento, la finestra con la shell dovrebbe essere simile a quella di Figura 4. In questa figura, il programma client Traceroute è in Massachussetts (USA), la destinazione in Francia. Da questa figura potete vedere che, per ogni valore di TTL, il programma sorgente ha inviato tre pacchetti "sonda". Traceroute visualizza l'RTT per ognuno di questi pacchetti, assieme all'indirizzo IP (e possibilmente il nome) dei router che hanno restituito il messaggio ICMP TTL-exceeded.

La Figura 5 mostra la finestra Wireshark per un pacchetto ICMP restituito da un router. Notare che questo pacchetto di errore ICMP contiene molti più campi del pacchetto ICMP echo inviato da Ping.

Cosa consegnare:

Per questa parte del laboratorio, dovete consegnare uno screenshot della finestra della shell. Quando possibile, nel rispondere alle domande, dovreste consegnare una stampa dei pacchetti all'interno della traccia che avete usato per rispondere. Annotate la stampa per spiegare la risposta. Per

questo file zip sono state raccolte da Wireshark in esecuzione su uno dei computer degli autori, eseguendo i passi indicati in questo laboratorio. Una volta scaricata la traccia, la potete caricare ed esaminare in Wireshark usando il menù a discesa *File*, scegliendo *Open*, e selezionando il file *ICMP-ethereal-trace-2*. Potete quindi usare questa traccia per rispondere alle domande che seguono.

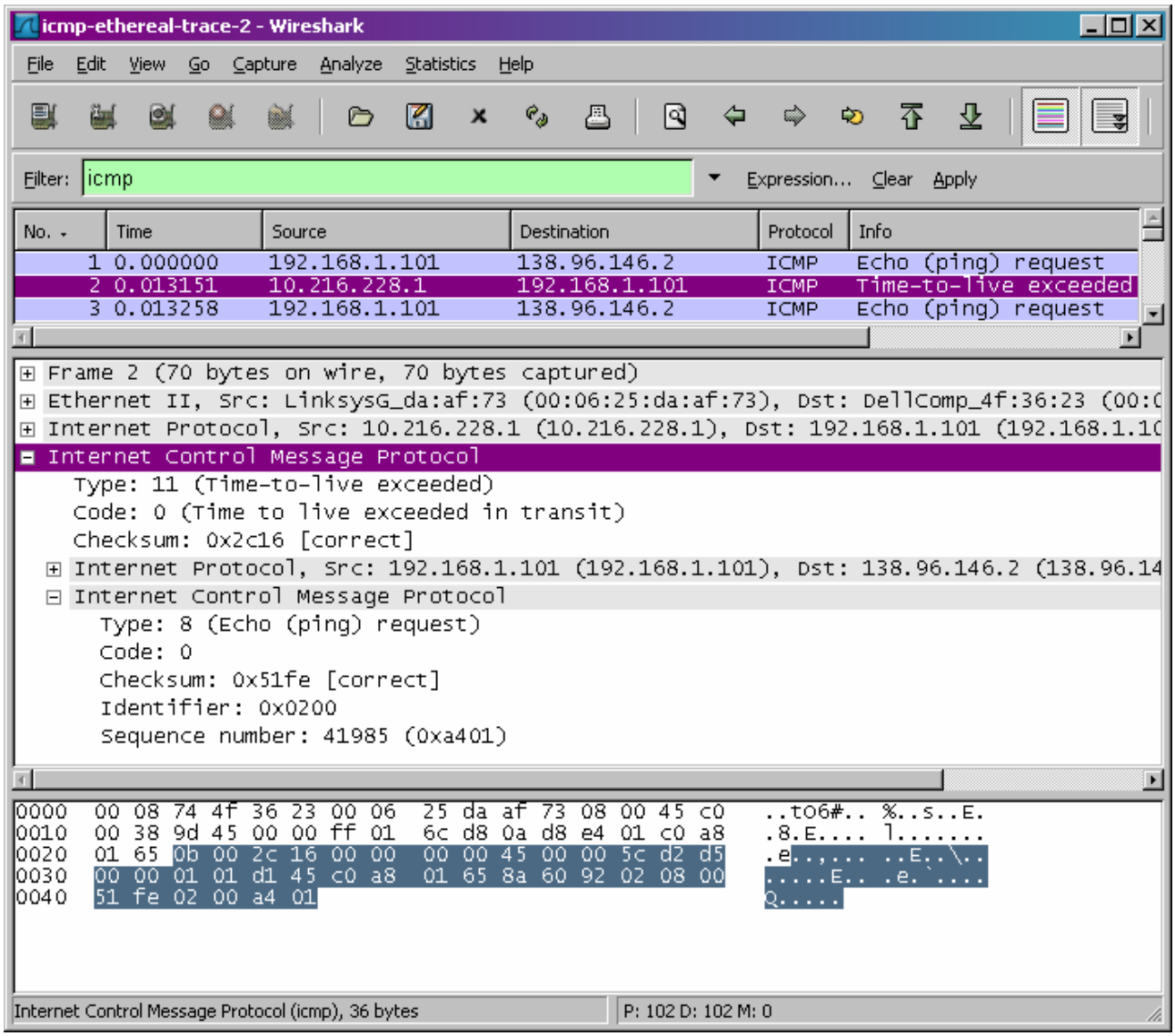


Figura 5: Finestra Wireshark con un pacchetto di errore ICMP e dettagli ICMP espansi

stampare un pacchetto usate *File → Print*, scegliete *Selected packet only*, scegliete *Packet summary line*, e selezionate la minima quantità di dettagli che è necessaria per rispondere alle domande.

Rispondete alle seguenti domande:

5. Qual è l'indirizzo IP del vostro host? Qual è l'indirizzo IP del destinatario?
6. Se ICMP avesse inviato pacchetti UDP (come su Unix/Linux col programma *traceroute* di default), il numero di protocollo nel datagramma IP sarebbe sempre 01 per i pacchetti "sonda"? Se no, quale sarebbe?
7. Esaminate il pacchetto ICMP echo nel vostro screenshot. È diverso dal pacchetto ICMP echo inviato da Ping nella prima parte di questo laboratorio. Se sì, qual'è la differenza?
8. Eaminata i pacchetti di errori ICMP nel vostro screenshot. Ha più campi del pacchetto ICMP echo? Cosa è incluso in questi campi?
9. Esaminate gli ultimi pacchetti ICMP ricevuti dal vostro host. In cosa differiscono questi pacchetti dai pacchetti di errore ICMP? Perché sono diversi?
10. Esaminando i risultati di RTT di traceroute, c'è un collegamento il cui ritardo è significativamente superiore agli altri? Riferendosi alla Figura 4, c'è un collegamento il cui ritardo è significativamente superiore agli altri? Sulla base dei nomi dei router, potete

indovinare la collocazione fisica dei due router agli estremi di questo collegamento?

Credito Extra

Per uno degli esercizi di programmazione avete creato un programma client per il ping UDP. Questo programma, a differenza del ping standard, manda pacchetti sonda UDP invece di ICMP. Usate questo programma client per inviare un pacchetto UDP con una porta destinazione inusuale a qualche sistema terminale attivo. Allo stesso tempo, usate Wireshark per catturare qualunque risposta proveniente dall'host destinazione. Fornire uno screenshot di Wireshark per la risposta e una analisi della stessa.