Corso di Laurea in Ingegneria delle Telecomunicazioni

Corso di Reti di Calcolatori

Docente: Simon Pietro Romano
spromano@unina.it

ICMP – ARP – RARP – DHCP - NAT

ICMP (Internet Control Message Protocol)

- Funzionalità:
  - Verificare lo stato della rete
    - echo request, echo reply
  - Riportare anomalie
    - destination unreachable
    - time exceeded
    - parameter problem
  - Scoprire la netmask
    - mask request
    - address mask reply
  - Migliorare il routing
    - redirect

ICMP

Valore	Tipo di Messaggio
0	Echo Reply
3	Destination Unreachable
4	Source Quence
5	Redirect
8	Echo Request
11	Time Exceeded for a Datagram
12	Parameter Problem on a Datagram
13	Timestamp Request
14	Timestamp Reply
15	Information Request (obsolete)
16	Information Reply (obsolete)
17	Address Mask Request
18	Address Mask Reply

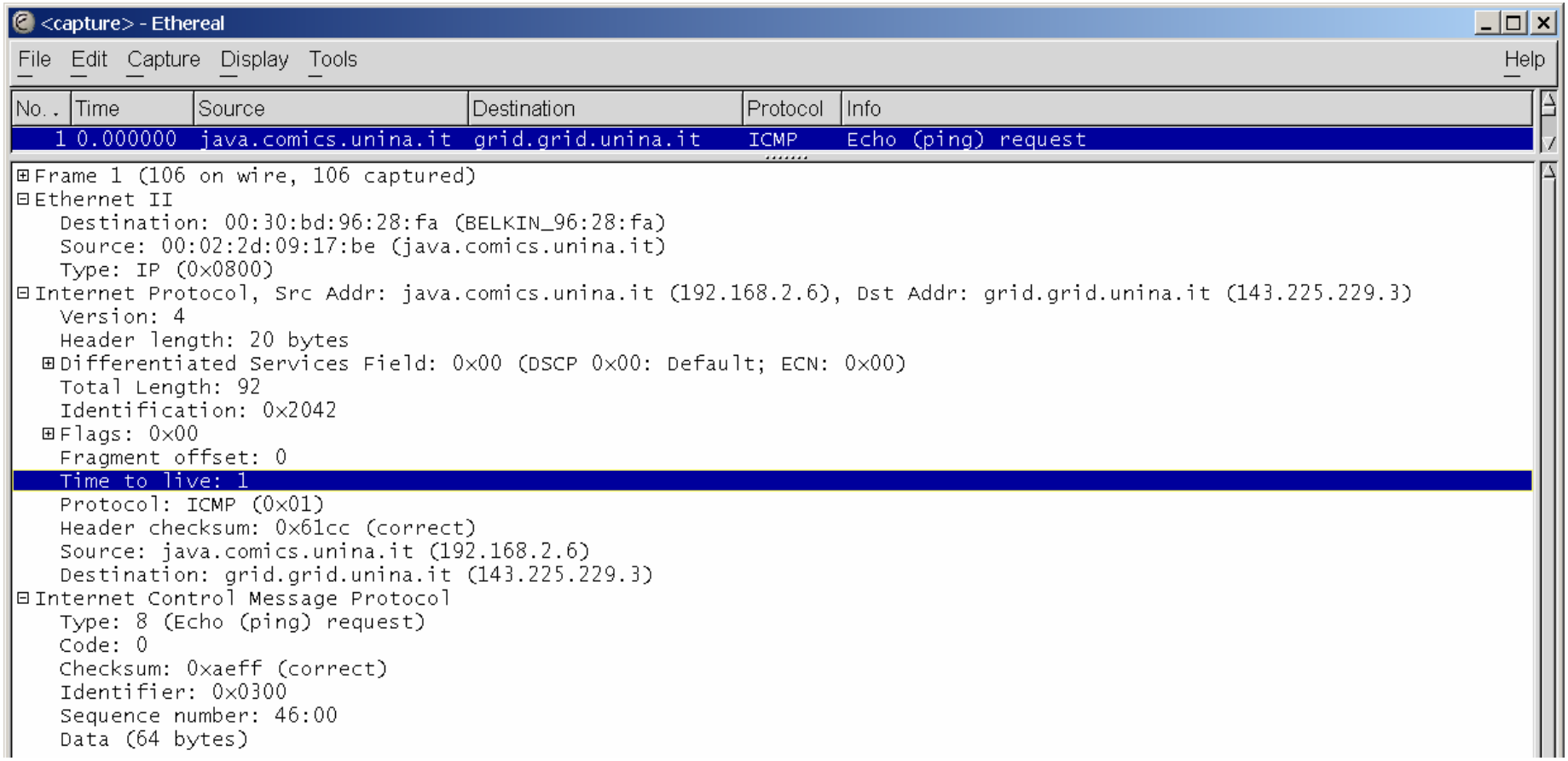
ICMP

- Applicazioni:
  - Ping
    - Utilizzato per verificare la connettività a livello rete tra due host, A e B
      - l'host A invia un pacchetto "echo request"
      - alla ricezione di tale messaggio, l'host B risponde con un pacchetto "echo reply"
  - Traceroute
    - Utilizzato per scoprire il percorso seguito per raggiungere una certa destinazione
    - Viene inviata una serie di pacchetti con TTL via via crescente, a partire da 1:
      - il router che, decrementando il TTL, lo azzera invierà indietro un messaggio "time exceeded"
        - » in questo modo si riesce a determinare il percorso fino alla destinazione

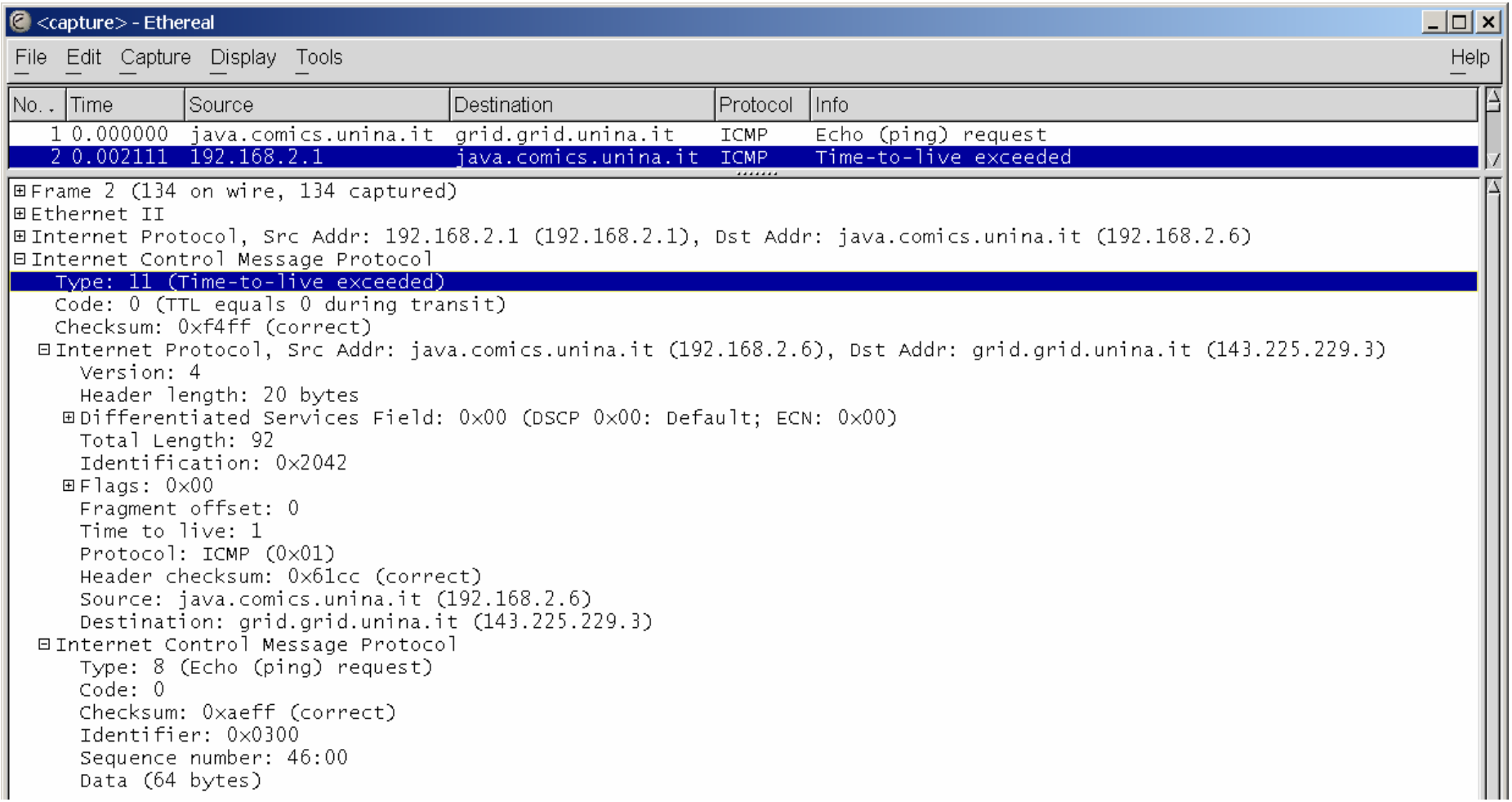
Esempio di traceroute (1/9)

```
C:\Documents and Settings\spromano>tracert 143.225.229.3

Rilevazione instradamento verso grid.grid.unina.it [143.225.229.3]
su un massimo di 30 punti di passaggio:

  1     2 ms     2 ms     2 ms  192.168.2.1
  2     4 ms     8 ms     9 ms  217.9.64.193
  3    11 ms    19 ms    19 ms  192.55.101.129
  4     3 ms     3 ms     3 ms  grid.grid.unina.it [143.225.229.3]

Rilevazione completata.
```

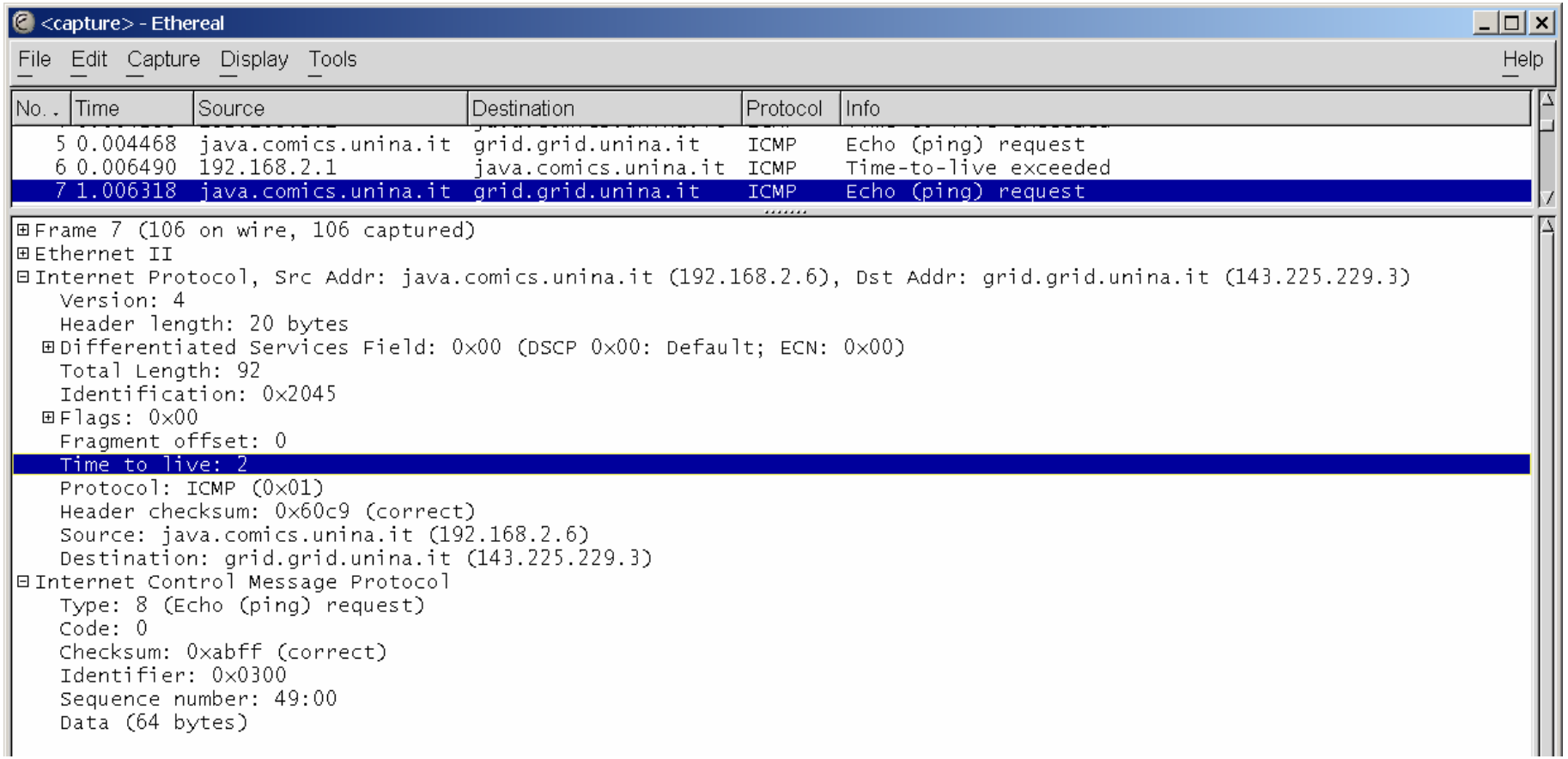
Esempio di traceroute (2/9)



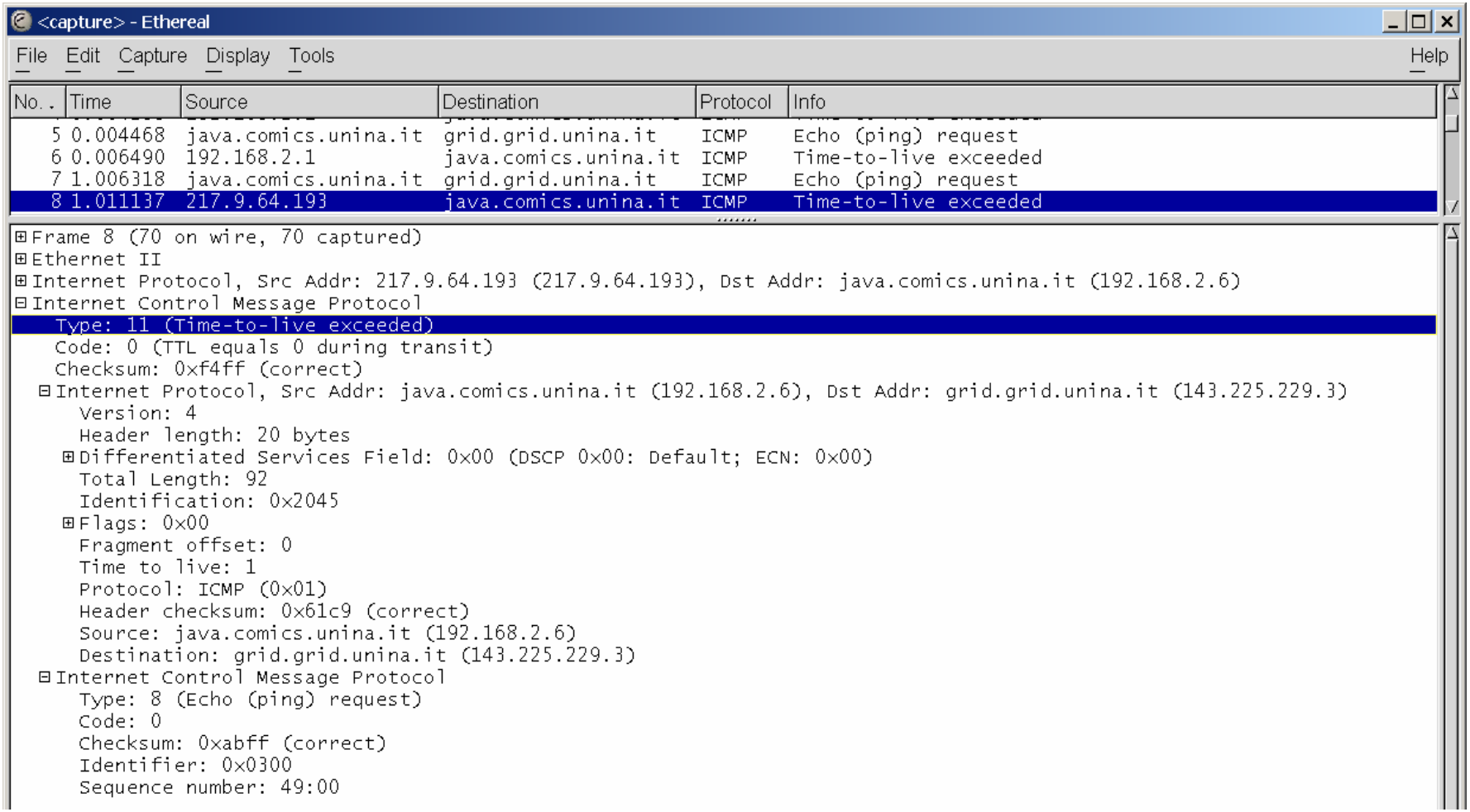
Esempio di traceroute (3/9)



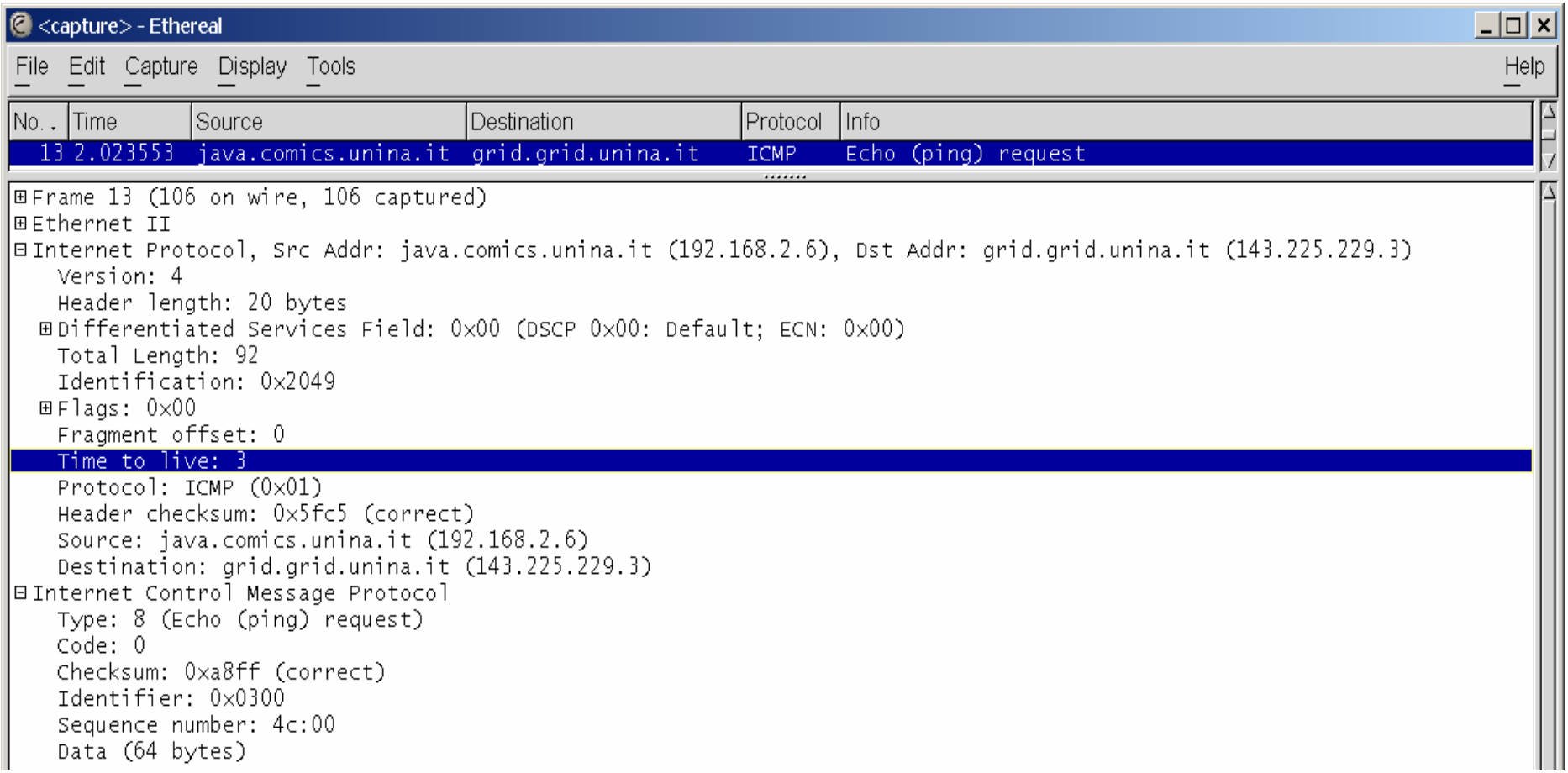
Esempio di traceroute (4/9)



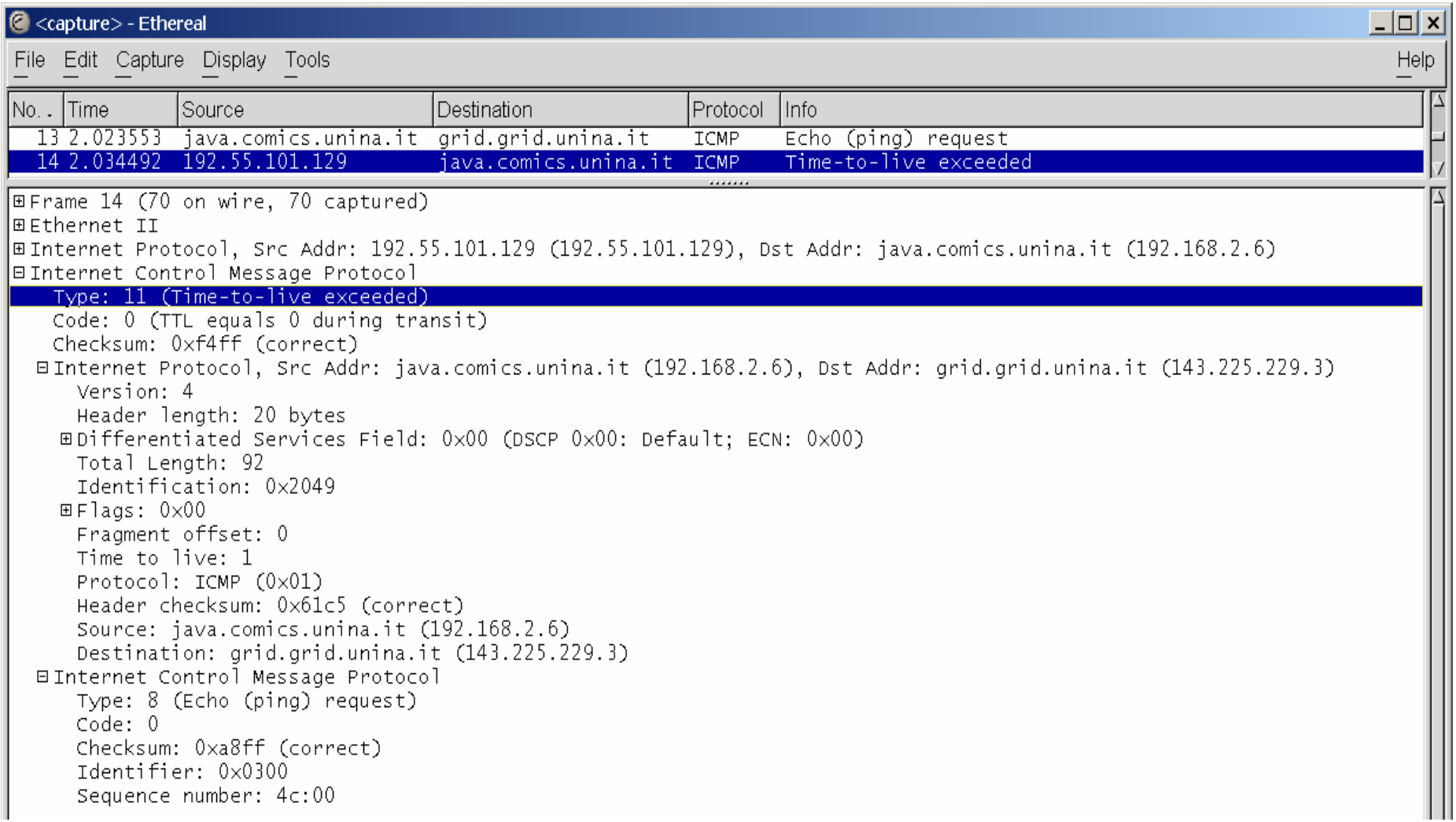
Esempio di traceroute (5/9)



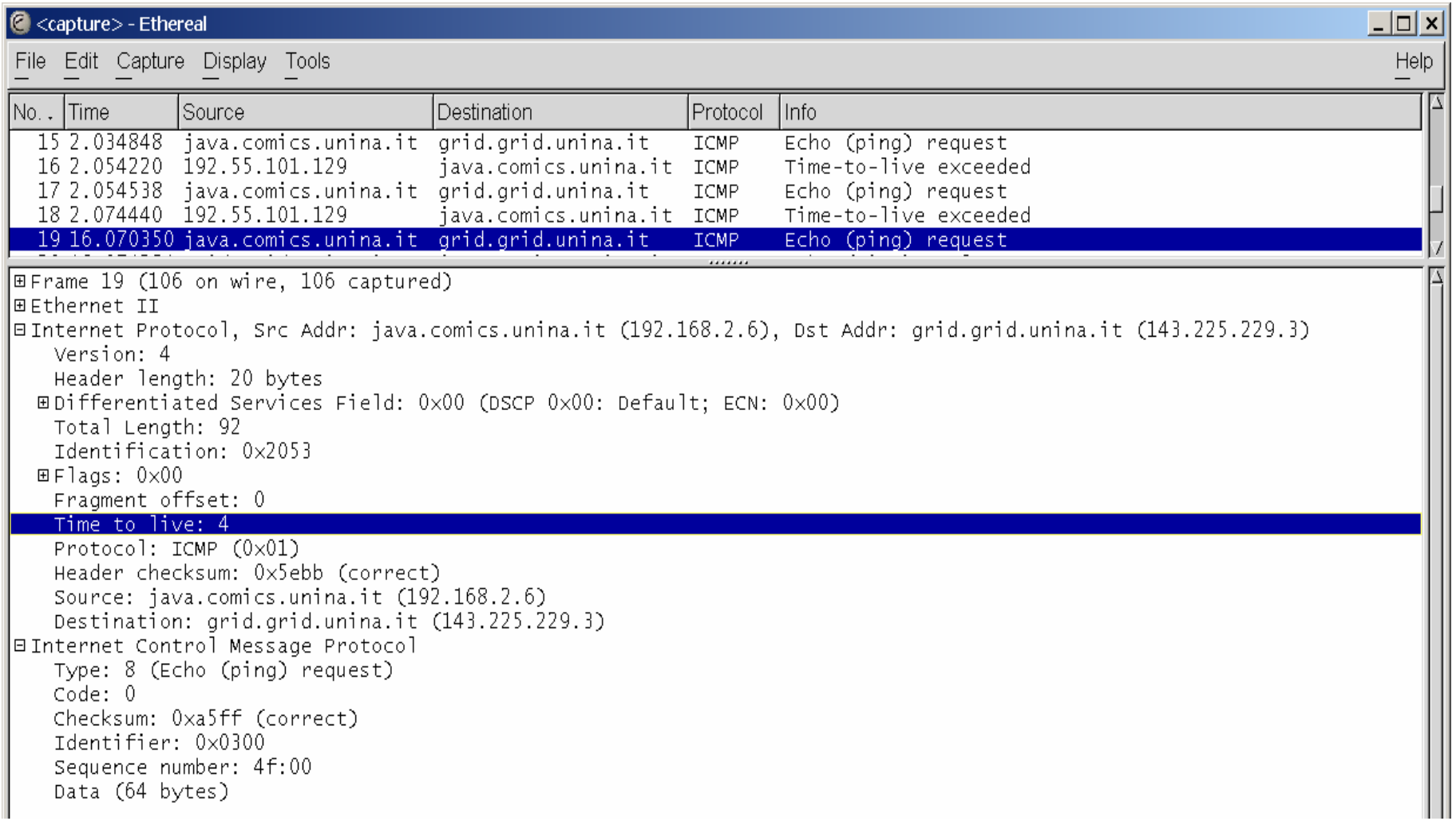
Esempio di traceroute (6/9)



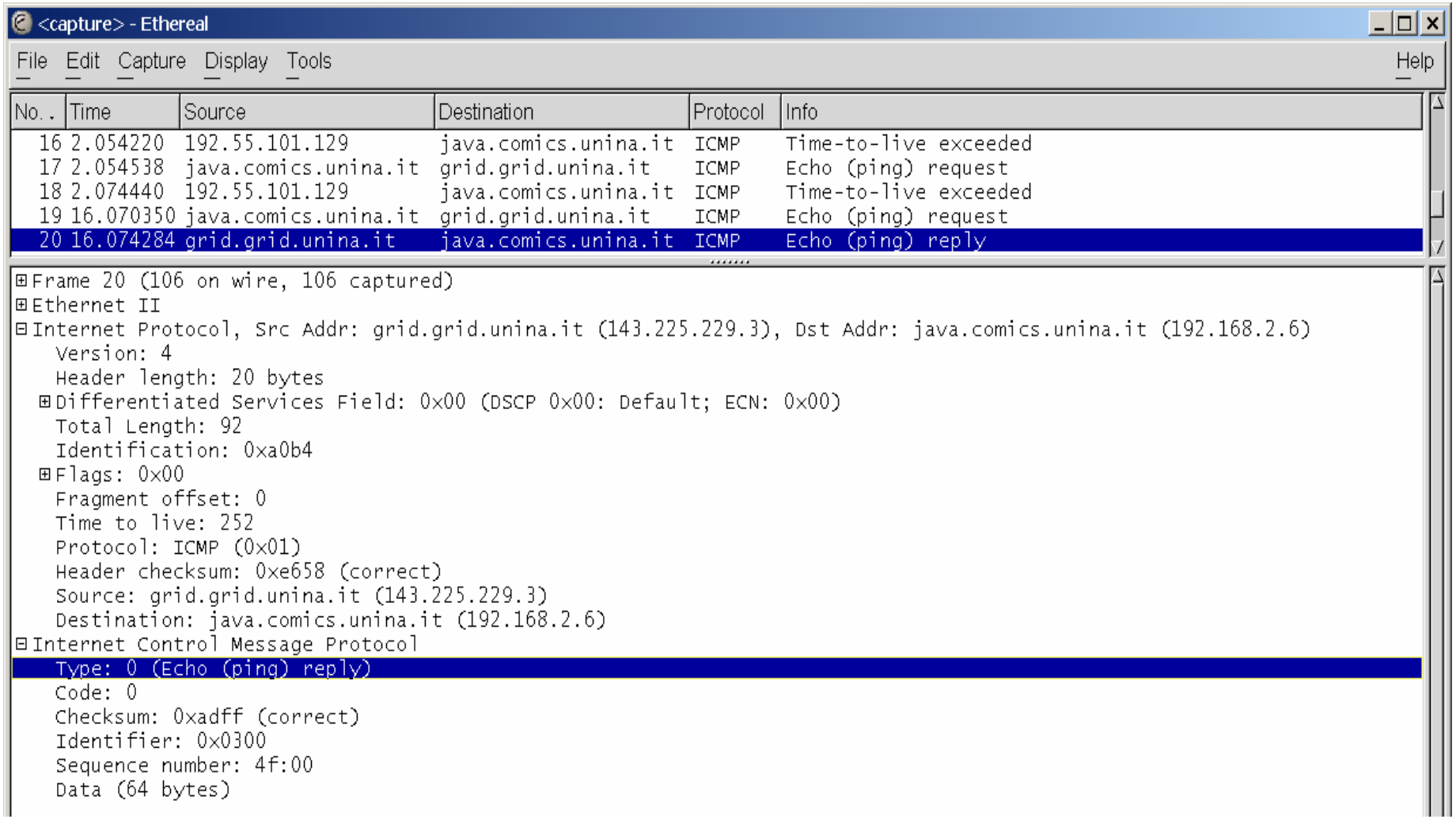
Esempio di traceroute (7/9)



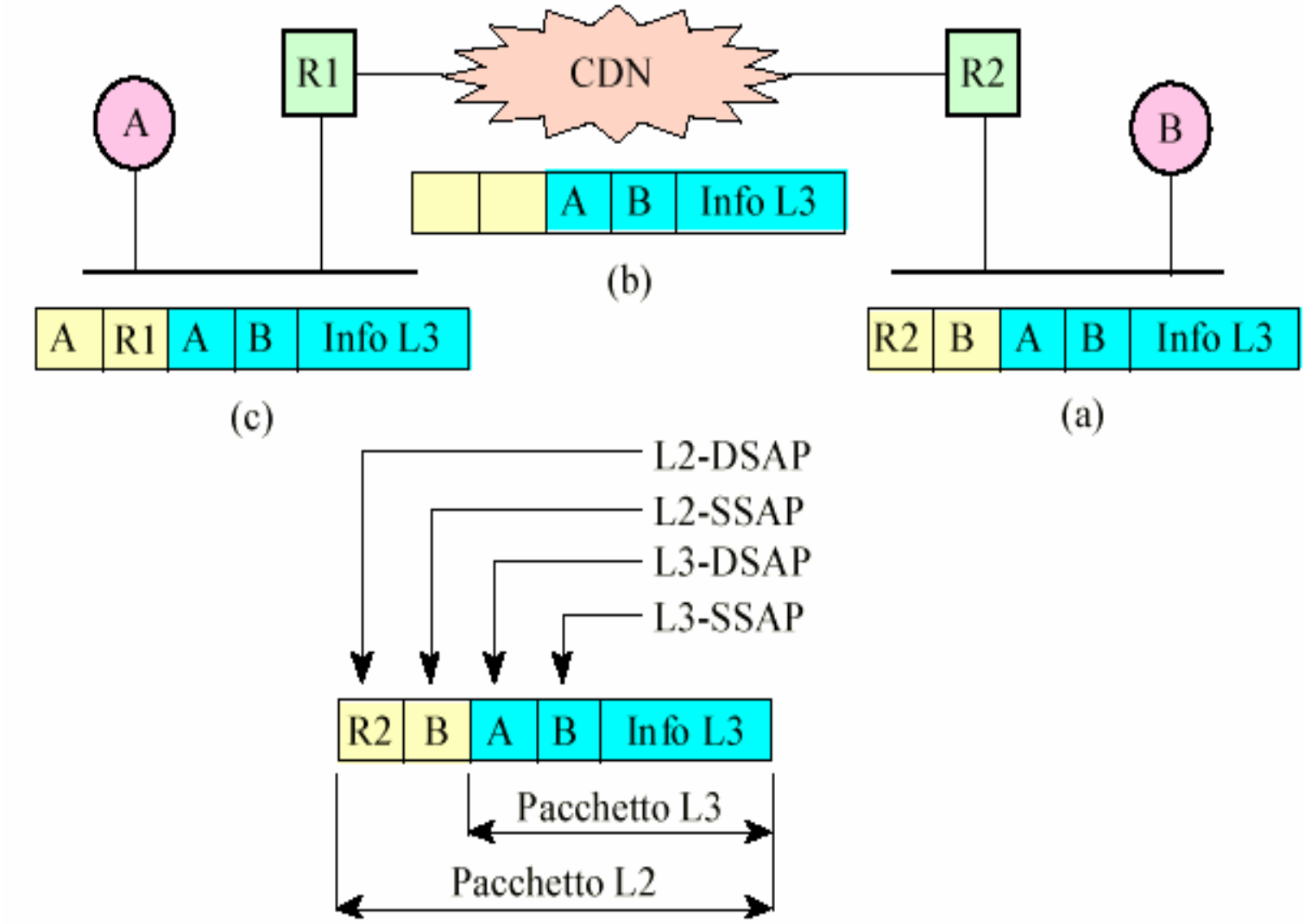
Esempio di traceroute (8/9)



Esempio di traceroute (9/9)



Indirizzi IP ed Indirizzi di Livello 2



Problema della risoluzione dell'indirizzo

- Due host possono comunicare direttamente solo se sono collegati alla stessa rete fisica
  - Per potersi scambiare informazioni devono conoscere i rispettivi indirizzi fisici
- Il protocollo IP consente di individuare univocamente un host tramite un indirizzo logico (indirizzo IP)
  - Tutte le applicazioni usano gli indirizzi logici ed ignorano la rete fisica. Ma per inviare un messaggio occorre necessariamente conoscere anche l'indirizzo fisico
  - Pertanto, serve un meccanismo di corrispondenza tra gli indirizzi logici e gli indirizzi fisici. Tale meccanismo è offerto dal protocollo ARP

ARP - Address Resolution Protocol

- Uno scenario tipico:
  - *A* deve spedire un datagram a *B,* host appartenente alla medesima rete logica (cioè, alla medesima rete IP)
  - *A* conosce l'indirizzo IP di *B*, ma non il suo indirizzo fisico
- Soluzione tramite ARP:
  - A manda in broadcast a tutti gli host della rete un pacchetto contenente l'indirizzo di rete di B, allo scopo di conoscere l'indirizzo fisico di B
  - B riconosce il suo indirizzo di rete e risponde ad A
  - Finalmente A conosce l'indirizzo fisico di B, quindi può spedire il datagram a B

Formato del pacchetto ARP

Hardware Type		Protocol Type
HLEN	PLEN	Operation
Sender Hardware Address		
Sender HW Address		Sender IP Address
Sender IP Address		Target HW Address
Target Hardware Address		
Target IP Address		

Incapsulamento dei pacchetti ARP

- Il protocollo ARP interagisce direttamente con il livello data link
- Il pacchetto ARP viene incapsulato in un frame e spedito in broadcast sulla rete
  - L'header del frame di livello 2 specifica che il frame contiene un pacchetto ARP

Esempio: richiesta ARP

No. .	Time	Source	Destination	Protocol	Info
1	0.000000	java.comics.unina.it	ff:ff:ff:ff:ff:ff	ARP	Who has 143.225.229.3? Tell 143.225.229.186
2	0.000239	grid.grid.unina.it	java.comics.unina.it	ARP	143.225.229.3 is at 00:90:27:d0:bb:56

```
⊞Frame 1 (42 on wire, 42 captured)
⊟Ethernet II
    Destination: ff:ff:ff:ff:ff:ff (ff:ff:ff:ff:ff:ff)
    Source: 00:08:0d:6a:a3:07 (java.comics.unina.it)
    Type: ARP (0x0806)
⊟Address Resolution Protocol (request)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (0x0001)
    Sender MAC address: 00:08:0d:6a:a3:07 (java.comics.unina.it)
    Sender IP address: java.comics.unina.it (143.225.229.186)
    Target MAC address: 00:00:00:00:00:00 (grid.grid.unina.it)
    Target IP address: grid.grid.unina.it (143.225.229.3)
```

Esempio: risposta ARP

No. .	Time	Source	Destination	Protocol	Info
1	0.000000	java.comics.unina.it	ff:ff:ff:ff:ff:ff	ARP	Who has 143.225.229.3? Tell 143.225.229.186
2	0.000239	grid.grid.unina.it	java.comics.unina.it	ARP	143.225.229.3 is at 00:90:27:d0:bb:56

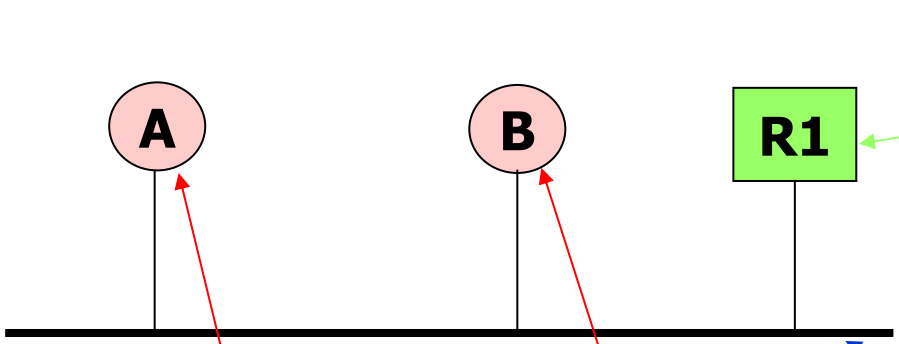
```
⊞Frame 2 (60 on wire, 60 captured)
⊟Ethernet II
    Destination: 00:08:0d:6a:a3:07 (java.comics.unina.it)
    Source: 00:90:27:d0:bb:56 (grid.grid.unina.it)
    Type: ARP (0x0806)
    Trailer: 000000000000000000000000000000000...
⊟Address Resolution Protocol (reply)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (0x0002)
    Sender MAC address: 00:90:27:d0:bb:56 (grid.grid.unina.it)
    Sender IP address: grid.grid.unina.it (143.225.229.3)
    Target MAC address: 00:08:0d:6a:a3:07 (java.comics.unina.it)
    Target IP address: java.comics.unina.it (143.225.229.186)
```

ARP: scenari tipici

- Primo caso: l'host destinazione è sulla stessa LAN (stessa subnet IP)

- Secondo caso: l'host destinazione non è sulla stessa LAN (subnet IP)

ARP: primo caso (1/3)



Default router

IP: 192.168.1.254
netmask: 255.255.255.0

Rete IP di classe C

rete: 192.168.1.0
netmask: 255.255.255.0

IP: 192.168.1.3
netmask: 255.255.255.0

IP: 192.168.1.1
netmask: 255.255.255.0

A ha intenzione di inviare un pacchetto a **B**. Prima Domanda: come fa **A** a sapere se **B** è sulla propria sottorete?

Risposta: attraverso la netmask!

ARP: primo caso (2/3)

- Ogni computer ha un indirizzo IP ed una netmask. La netmask serve ad individuare la propria sottorete IP:
  - Digitare da una shell win2000 il comando:
    - ipconfig /all
- Il computer **A** esegue una AND tra l'indirizzo IP destinazione e la propria netmask.
  - Nel caso precedente:

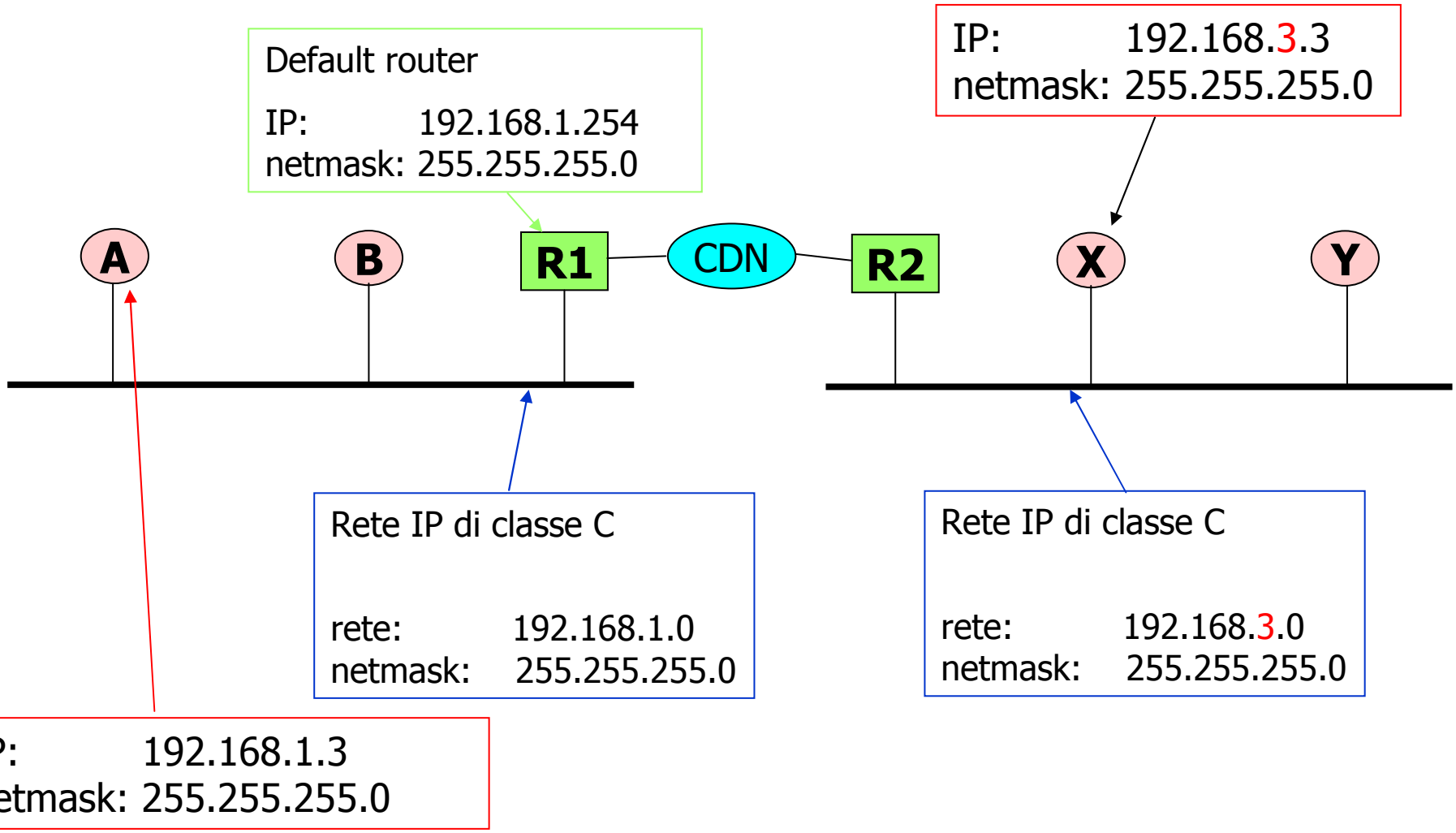
IP di B	192.168.1.2
	AND
netmask A	255.255.255.0
	=
→	192.168.1.0

E' proprio l'indirizzo della sottorete IP cui appartiene A

ARP: primo caso (3/3)

- Se il computer **B** è sulla stessa sottorete IP
  - allora mando un pacchetto *ARP request* in broadcast
    - tale pacchetto contiene, nel campo **DEST IP,** l'indirizzo IP di B

ARP: secondo caso (1/2)



ARP: secondo caso (2/2)

- Se **A** intende mandare un pacchetto a **X**, l'operazione di AND tra la netmask e l'indirizzo IP DEST fornisce un risultato differente

IP di X	192.168.3.3
	AND
netmask A	255.255.255.0
	=
→	192.168.3.0

Non è l'indirizzo della sottorete cui appartiene A → Occorre inviare il pacchetto al router.

In questo caso, pertanto, si prepara un pacchetto ARP in cui si specifica come indirizzo IP DEST proprio l'indirizzo IP del router

ARP: ricapitolando…

- Operazione di AND logico tra l'indirizzo IP della destinazione e la propria netmask:
  - Se il risultato fornisce l'indirizzo della propria subnet IP:
    - Invia una richiesta ARP per risolvere l'indirizzo della destinazione
  - …altrimenti:
    - Il pacchetto deve essere inviato al router di default:
      - Nel caso in cui l'indirizzo MAC del router non sia noto:
        - » Invia una richiesta ARP per risolvere l'indirizzo IP del router

Raffinamenti del protocollo

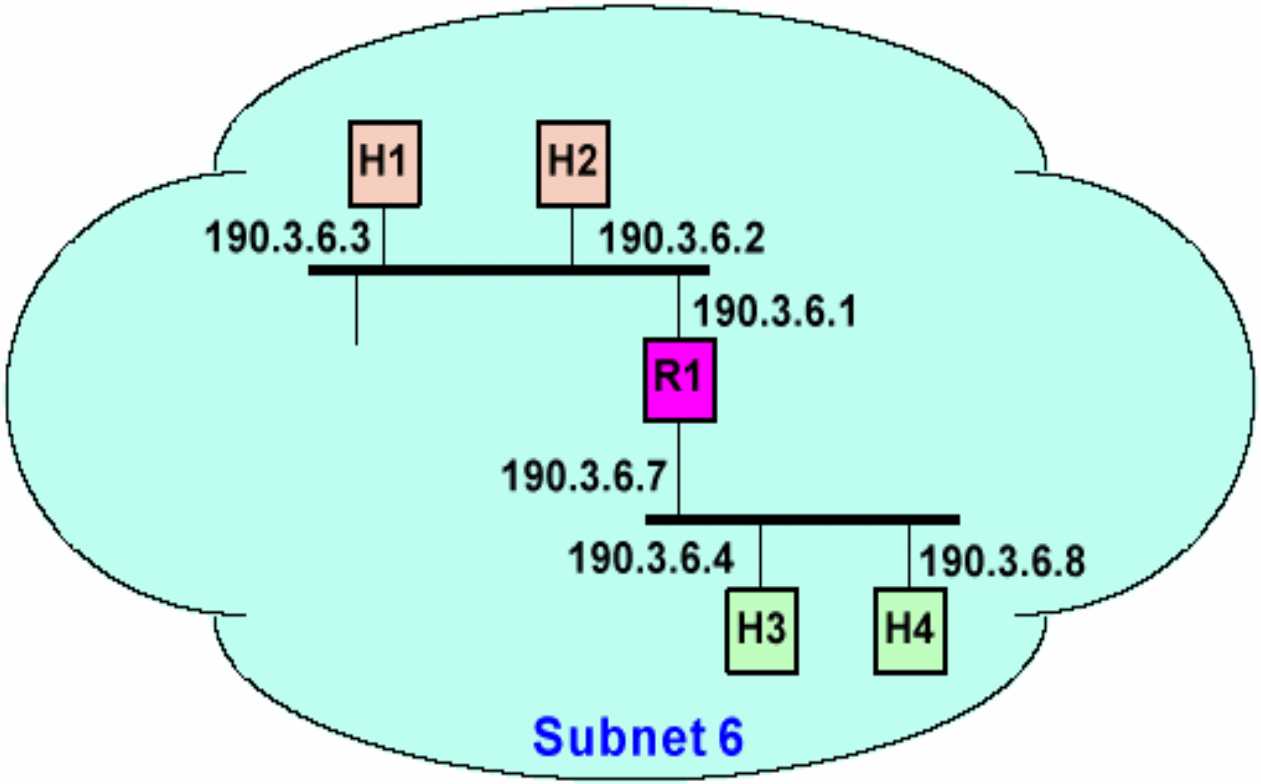
- Per ridurre il traffico sulla rete, ogni host mantiene una cache con le corrispondenze tra indirizzi logici e fisici
  - Prima di spedire una richiesta ARP controlla nella cache
- Il pacchetto ARP contiene indirizzo fisico e logico del mittente
  - Gli host che leggono il pacchetto possono aggiornare le loro ARP cache

Monitoraggio di ARP

- Con il comando *arp* è possibile leggere e modificare il contenuto della arp cache
  - **arp –a** (legge il contenuto di tutta la cache)
- Con il comando *tcpdump* è possibile monitorare tutto il traffico che viaggia sulla rete
  - È possibile filtrare solo i pacchetti spediti da un dato protocollo su una data interfaccia
  - **tcpdump arp** (legge solo i pacchetti arp)

Proxy ARP

- Permette di usare la stessa subnet su due o più reti fisiche diverse



Reverse ARP

- Il protocollo RARP svolge il ruolo opposto ad ARP
  - fisico → logico
- Usato per sistemi diskless:
  - X terminal, diskless workstation
  - Al boot non conoscono il loro indirizzo IP

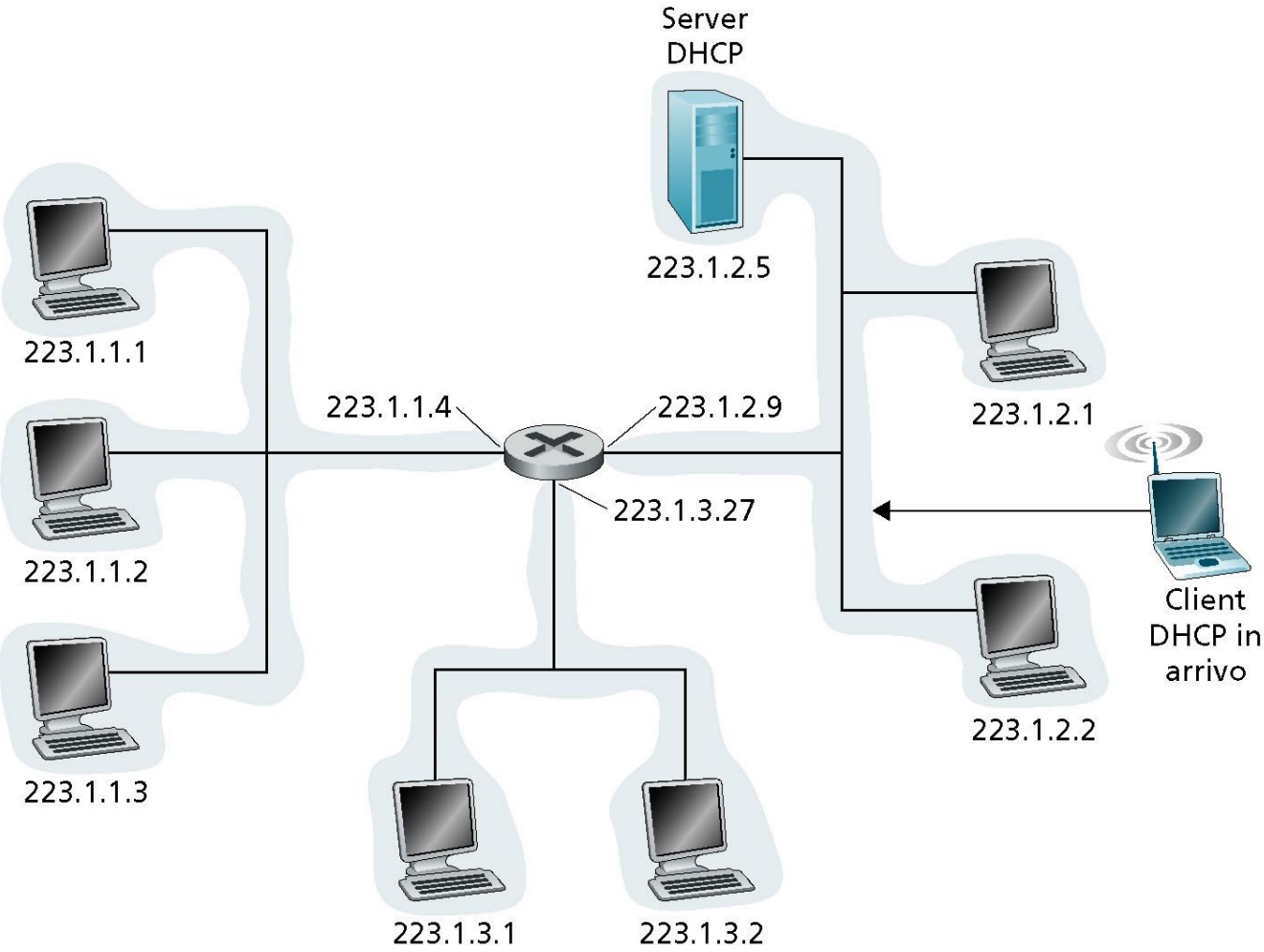
Scenario RARP

- *A* conosce il proprio indirizzo MAC, ma non conosce il proprio indirizzo IP
- L'host B (server RARP) conosce l'indirizzo IP di A
- Soluzione
  - RARP request sulla rete (in brodacast)
  - *B* risponde con un messaggio RARP reply contenente l'indirizzo IP di *A*

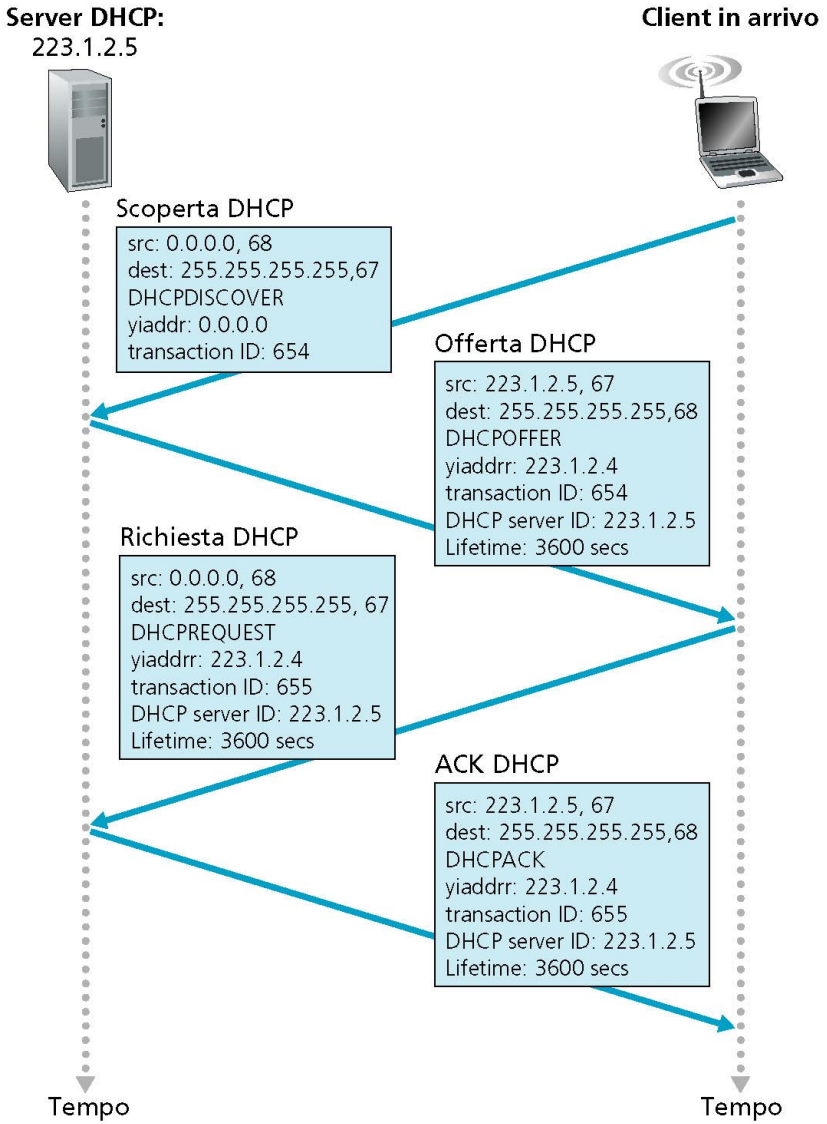
Altre soluzioni per boot remoto

- Il protocollo RARP è stato sostituito da altri protocolli più flessibili e potenti:
  - BOOTP: BOOTstrap Protocol
  - DHCP: Dynamic Host Configuration Protocol
  - Utilizzati per assegnare dinamicamente gli indirizzi agli host di una rete IP

DHCP: scenario tipico



Interazione client-server via DHCP



DHCP discover

No. .	Time	Source	Destination	Protocol	Info
1	0.000000	0.0.0.0	255.255.255.255	DHCP	DHCP Discover - Transaction ID 0xb2c065b
2	0.004628	192.168.2.1	192.168.2.13	DHCP	DHCP Offer - Transaction ID 0xb2c065b
3	0.005392	0.0.0.0	255.255.255.255	DHCP	DHCP Request - Transaction ID 0xb2c065b
4	0.009656	192.168.2.1	192.168.2.13	DHCP	DHCP ACK - Transaction ID 0xb2c065b

```
⊞Frame 1 (342 on wire, 342 captured)
⊟Ethernet II
    Destination: ff:ff:ff:ff:ff:ff (ff:ff:ff:ff:ff:ff)
    Source: 00:02:2d:09:17:be (Agere_09:17:be)
    Type: IP (0x0800)
⊞Internet Protocol, Src Addr: 0.0.0.0 (0.0.0.0), Dst Addr: 255.255.255.255 (255.255.255.255)
⊟User Datagram Protocol, Src Port: bootpc (68), Dst Port: bootps (67)
    Source port: bootpc (68)
    Destination port: bootps (67)
    Length: 308
    Checksum: 0xae22 (correct)
⊟Bootstrap Protocol
    Message type: Boot Request (1)
    Hardware type: Ethernet
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0x0b2c065b
    Seconds elapsed: 0
    Broadcast flag: 0x0000
    Client IP address: 0.0.0.0 (0.0.0.0)
    Your (client) IP address: 0.0.0.0 (0.0.0.0)
    Next server IP address: 0.0.0.0 (0.0.0.0)
    Relay agent IP address: 0.0.0.0 (0.0.0.0)
    Client hardware address: 00:02:2d:09:17:be
    Server host name not given
    Boot file name not given
    Magic cookie: (OK)
    Option 53: DHCP Message Type = DHCP Discover
    Option 116: DHCP Auto-Configuration (1 bytes)
  ⊞Option 61: Client identifier
    Option 50: Requested IP Address = 192.168.2.13
    Option 12: Host Name = "java"
    Option 60: Vendor class identifier = "MSFT 5.0"
  ⊞Option 55: Parameter Request List
    End Option
    Padding
```

DHCP offer

No.	Time	Source	Destination	Protocol	Info
1	0.000000	0.0.0.0	255.255.255.255	DHCP	DHCP Discover - Transaction ID 0xb2c065b
2	0.004628	192.168.2.1	192.168.2.13	DHCP	DHCP Offer - Transaction ID 0xb2c065b
3	0.005392	0.0.0.0	255.255.255.255	DHCP	DHCP Request - Transaction ID 0xb2c065b
4	0.009656	192.168.2.1	192.168.2.13	DHCP	DHCP ACK - Transaction ID 0xb2c065b

```
⊞Frame 2 (590 on wire, 590 captured)
⊟Ethernet II
    Destination: 00:02:2d:09:17:be (Agere_09:17:be)
    Source: 00:30:bd:96:28:fa (BELKIN_96:28:fa)
    Type: IP (0x0800)
⊞Internet Protocol, Src Addr: 192.168.2.1 (192.168.2.1), Dst Addr: 192.168.2.13 (192.168.2.13)
⊟User Datagram Protocol, Src Port: bootps (67), Dst Port: bootpc (68)
    Source port: bootps (67)
    Destination port: bootpc (68)
    Length: 556
    Checksum: 0xc29a (correct)
⊟Bootstrap Protocol
    Message type: Boot Reply (2)
    Hardware type: Ethernet
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0x0b2c065b
    Seconds elapsed: 0
    Broadcast flag: 0x0000
    Client IP address: 0.0.0.0 (0.0.0.0)
    Your (client) IP address: 192.168.2.13 (192.168.2.13)
    Next server IP address: 0.0.0.0 (0.0.0.0)
    Relay agent IP address: 0.0.0.0 (0.0.0.0)
    Client hardware address: 00:02:2d:09:17:be
    Server host name not given
    Boot file name not given
    Magic cookie: (OK)
    Option 53: DHCP Message Type = DHCP Offer
    Option 54: Server Identifier = 192.168.2.1
    Option 51: IP Address Lease Time = 12427 days, 7 hours, 45 minutes, 41 seconds
    Option 1: Subnet Mask = 255.255.255.0
    Option 3: Router = 192.168.2.1
  ⊟Option 6: Domain Name Server
      IP Address: 217.9.64.200
      IP Address: 217.9.64.220
      IP Address: 217.9.64.3
    Option 15: Domain Name = "napoli.consorzio-cini.it"
    Option 44: NetBIOS over TCP/IP Name Server = 217.9.64.200
    End Option
    Padding
```

DHCP request

File Edit Capture Display Tools

No.	Time	Source	Destination	Protocol	Info
1	0.000000	0.0.0.0	255.255.255.255	DHCP	DHCP Discover - Transaction ID 0xb2c065b
2	0.004628	192.168.2.1	192.168.2.13	DHCP	DHCP Offer - Transaction ID 0xb2c065b
3	0.005392	0.0.0.0	255.255.255.255	DHCP	DHCP Request - Transaction ID 0xb2c065b
4	0.009656	192.168.2.1	192.168.2.13	DHCP	DHCP ACK - Transaction ID 0xb2c065b

```
⊞Frame 3 (361 on wire, 361 captured)
⊟Ethernet II
   Destination: ff:ff:ff:ff:ff:ff (ff:ff:ff:ff:ff:ff)
   Source: 00:02:2d:09:17:be (Agere_09:17:be)
   Type: IP (0x0800)
⊞Internet Protocol, Src Addr: 0.0.0.0 (0.0.0.0), Dst Addr: 255.255.255.255 (255.255.255.255)
⊞User Datagram Protocol, Src Port: bootpc (68), Dst Port: bootps (67)
⊟Bootstrap Protocol
   Message type: Boot Request (1)
   Hardware type: Ethernet
   Hardware address length: 6
   Hops: 0
   Transaction ID: 0x0b2c065b
   Seconds elapsed: 0
   Broadcast flag: 0x0000
   Client IP address: 0.0.0.0 (0.0.0.0)
   Your (client) IP address: 0.0.0.0 (0.0.0.0)
   Next server IP address: 0.0.0.0 (0.0.0.0)
   Relay agent IP address: 0.0.0.0 (0.0.0.0)
   Client hardware address: 00:02:2d:09:17:be
   Server host name not given
   Boot file name not given
   Magic cookie: (OK)
   Option 53: DHCP Message Type = DHCP Request
 ⊟Option 61: Client identifier
     Hardware type: Ethernet
     Client hardware address: 00:02:2d:09:17:be
   Option 50: Requested IP Address = 192.168.2.13
   Option 54: Server Identifier = 192.168.2.1
   Option 12: Host Name = "java"
   Option 81: Client Fully Qualified Domain Name (23 bytes)
   Option 60: Vendor class identifier = "MSFT 5.0"
 ⊟Option 55: Parameter Request List
     1 = Subnet Mask
     15 = Domain Name
     3 = Router
     6 = Domain Name Server
     44 = NetBIOS over TCP/IP Name Server
     46 = NetBIOS over TCP/IP Node Type
     47 = NetBIOS over TCP/IP Scope
     31 = Perform Router Discover
     33 = Static Route
     Unknown Option Code: 249
     43 = Vendor-Specific Information
   End Option
```

DHCP ACK

File Edit Capture Display Tools

No.	Time	Source	Destination	Protocol	Info
1	0.000000	0.0.0.0	255.255.255.255	DHCP	DHCP Discover - Transaction ID 0xb2c065b
2	0.004628	192.168.2.1	192.168.2.13	DHCP	DHCP Offer - Transaction ID 0xb2c065b
3	0.005392	0.0.0.0	255.255.255.255	DHCP	DHCP Request - Transaction ID 0xb2c065b
4	0.009656	192.168.2.1	192.168.2.13	DHCP	DHCP ACK - Transaction ID 0xb2c065b

```
⊞Frame 4 (590 on wire, 590 captured)
⊟Ethernet II
    Destination: 00:02:2d:09:17:be (Agere_09:17:be)
    Source: 00:30:bd:96:28:fa (BELKIN_96:28:fa)
    Type: IP (0x0800)
⊞Internet Protocol, Src Addr: 192.168.2.1 (192.168.2.1), Dst Addr: 192.168.2.13 (192.168.2.13)
⊞User Datagram Protocol, Src Port: bootps (67), Dst Port: bootpc (68)
⊟Bootstrap Protocol
    Message type: Boot Reply (2)
    Hardware type: Ethernet
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0x0b2c065b
    Seconds elapsed: 0
    Broadcast flag: 0x0000
    Client IP address: 0.0.0.0 (0.0.0.0)
    Your (client) IP address: 192.168.2.13 (192.168.2.13)
    Next server IP address: 0.0.0.0 (0.0.0.0)
    Relay agent IP address: 0.0.0.0 (0.0.0.0)
    Client hardware address: 00:02:2d:09:17:be
    Server host name not given
    Boot file name not given
    Magic cookie: (OK)
    Option 53: DHCP Message Type = DHCP ACK
    Option 54: Server Identifier = 192.168.2.1
    Option 51: IP Address Lease Time = 12427 days, 13 hours, 37 minutes, 3 seconds
    Option 1: Subnet Mask = 255.255.255.0
    Option 3: Router = 192.168.2.1
  ⊟Option 6: Domain Name Server
      IP Address: 217.9.64.200
      IP Address: 217.9.64.220
      IP Address: 217.9.64.3
    Option 15: Domain Name = "napoli.consorzio-cini.it"
    Option 44: NetBIOS over TCP/IP Name Server = 217.9.64.200
    End Option
    Padding
```

NAT: Network Address Translation

- Network Address Translation (RFC 1631) consente ad un dispositivo di agire come intermediario tra Internet (rete pubblica) e una rete privata
- In questo modo, un unico indirizzo IP può rappresentare un intero gruppo di computer

NAT

- L'uso più comune del NAT è quello di mappare un insieme di indirizzi privati su di un unico indirizzo pubblico, utilizzando differenti porti per mantenere traccia delle diverse connessioni

NAT

- Quando il router riceve un pacchetto inviato da un computer della rete privata ad un computer esterno, salva in una tabella l'indirizzo e il porto del mittente, oltre ai nuovi valori che esso assegna
- Tale tabella viene consultata anche quando il router riceve un pacchetto dal computer destinazione

Source Computer	Source Computer's IP Address	Source Computer's Port	NAT Router's IP Address	NAT Router's Assigned Port Number
A	192.168.32.10	400	215.37.32.203	1
B	192.168.32.13	50	215.37.32.203	2
C	192.168.32.15	3750	215.37.32.203	3
D	192.168.32.18	206	215.37.32.203	4

NAT: un esempio