

Corso di Laurea in Ingegneria Informatica



**Corso di Reti di Calcolatori
a.a. 2009/10**

**Roberto Canonico (roberto.canonico@unina.it)
Antonio Pescapè (pescape@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



	<u>Type</u>	<u>Code</u>	<u>description</u>
• I messaggi sono individuati da un tipo e da un codice	0	0	echo reply (ping)
	3	0	dest. network unreachable
	3	1	dest host unreachable
	3	2	dest protocol unreachable
	3	3	dest port unreachable
	3	6	dest network unknown
	3	7	dest host unknown
	4	0	source quench (congestion control - not used)
	8	0	echo request (ping)
	9	0	route advertisement
	10	0	router discovery
	11	0	TTL expired
	12	0	bad IP header

3

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 azzererà invierà indietro un messaggio "time exceeded"
 - » in questo modo si riesce a determinare il percorso fino alla destinazione

4

Esempio di traceroute (1/9)



```
C:\Documents and Settings\sprmano>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.
```

5

Esempio di traceroute (2/9)



The screenshot shows a packet capture window titled "<capture> - Ethereal". The packet list pane shows a single packet (No. 1) at time 0.000000, source java.comics.unina.it, destination grid.grid.unina.it, protocol ICMP, and info Echo (ping) request. The packet details pane is expanded to show the following structure:

- Frame 1 (106 on wire, 106 captured)
- Ethernet II
 - Destination: 00:30:bd:96:28:fa (BELKIN_96:28:fa)
 - Source: 00:02:2d:09:17:be (java.comics.unina.it)
 - Type: IP (0x0800)
- Internet Protocol, Src Addr: java.comics.unina.it (192.168.2.6), Dst Addr: grid.grid.unina.it (143.225.229.3)
 - Version: 4
 - Header length: 20 bytes
 - Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
 - Total Length: 92
 - Identification: 0x2042
 - Flags: 0x00
 - Fragment offset: 0
- Time to live: 64
- Protocol: ICMP (0x01)
 - Header checksum: 0x61cc (correct)
 - Source: java.comics.unina.it (192.168.2.6)
 - Destination: grid.grid.unina.it (143.225.229.3)
- Internet Control Message Protocol
 - Type: 8 (Echo (ping) request)
 - Code: 0
 - Checksum: 0xaeff (correct)
 - Identifier: 0x0300
 - Sequence number: 46:00
 - Data (64 bytes)

6

Esempio di traceroute (3/9)



The screenshot shows the Wireshark interface with a packet capture of an ICMP Echo request and its response. The packet list pane shows two packets:

No.	Time	Source	Destination	Protocol	Info
1	0.000000	java.comics.unina.it	grid.grid.unina.it	ICMP	Echo (ping) request
2	0.002111	192.168.2.1	java.comics.unina.it	ICMP	Time-to-live exceeded

The packet details pane for packet 2 is expanded, showing the following structure:

- Frame 2 (134 on wire, 134 captured)
- Ethernet II
- Internet Protocol, Src Addr: 192.168.2.1 (192.168.2.1), Dst Addr: java.comics.unina.it (192.168.2.6)
- Internet Control Message Protocol
 - Type: 8 (Echo (ping) request)
 - Code: 0 (TTL equals 0 during transit)
 - Checksum: 0xF4ff (correct)
- Internet Protocol, Src Addr: java.comics.unina.it (192.168.2.6), Dst Addr: grid.grid.unina.it (143.225.229.3)
 - Version: 4
 - Header length: 20 bytes
 - Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
 - Total Length: 92
 - Identification: 0x2042
 - Flags: 0x00
 - Fragment offset: 0
 - Time to live: 1
 - Protocol: ICMP (0x01)
 - Header checksum: 0x61cc (correct)
 - Source: java.comics.unina.it (192.168.2.6)
 - Destination: grid.grid.unina.it (143.225.229.3)
- Internet Control Message Protocol
 - Type: 8 (Echo (ping) request)
 - Code: 0
 - Checksum: 0xaeff (correct)
 - Identifier: 0x0300
 - Sequence number: 46:00
 - Data (64 bytes)

7

Esempio di traceroute (4/9)



The screenshot shows the Wireshark interface with a packet capture of an ICMP Echo request and its response. The packet list pane shows three packets:

No.	Time	Source	Destination	Protocol	Info
5	0.004468	java.comics.unina.it	grid.grid.unina.it	ICMP	Echo (ping) request
6	0.006490	192.168.2.1	java.comics.unina.it	ICMP	Time-to-live exceeded
7	1.008388	java.comics.unina.it	grid.grid.unina.it	ICMP	Echo (ping) request

The packet details pane for packet 7 is expanded, showing the following structure:

- Frame 7 (106 on wire, 106 captured)
- Ethernet II
- Internet Protocol, Src Addr: java.comics.unina.it (192.168.2.6), Dst Addr: grid.grid.unina.it (143.225.229.3)
 - Version: 4
 - Header length: 20 bytes
 - Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
 - Total Length: 92
 - Identification: 0x2045
 - Flags: 0x00
 - Fragment offset: 0
 - Time to live: 2
 - Protocol: ICMP (0x01)
 - Header checksum: 0x60c9 (correct)
 - Source: java.comics.unina.it (192.168.2.6)
 - Destination: grid.grid.unina.it (143.225.229.3)
- Internet Control Message Protocol
 - Type: 8 (Echo (ping) request)
 - Code: 0
 - Checksum: 0xabff (correct)
 - Identifier: 0x0300
 - Sequence number: 49:00
 - Data (64 bytes)

8

Esempio di traceroute (5/9)



No.	Time	Source	Destination	Protocol	Info
5	0.004468	java.comics.unina.it	grid.grid.unina.it	ICMP	Echo (ping) request
6	0.006490	192.168.2.1	java.comics.unina.it	ICMP	Time-to-live exceeded
7	1.006318	java.comics.unina.it	grid.grid.unina.it	ICMP	Echo (ping) request
8	1.011137	217.9.64.193	java.comics.unina.it	ICMP	Time-to-live exceeded

Frame 8 (70 on wire, 70 captured)
Ethernet II
Internet Protocol, Src Addr: 217.9.64.193 (217.9.64.193), Dst Addr: java.comics.unina.it (192.168.2.6)
Internet Control Message Protocol
Type: 8 (Time-to-live exceeded)
Code: 0 (TTL equals 0 during transit)
Checksum: 0xf4ff (correct)
Internet Protocol, Src Addr: java.comics.unina.it (192.168.2.6), Dst Addr: grid.grid.unina.it (143.225.229.3)
Version: 4
Header length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
Total Length: 92
Identification: 0x2045
Flags: 0x00
Fragment offset: 0
Time to live: 1
Protocol: ICMP (0x01)
Header checksum: 0x61c9 (correct)
Source: java.comics.unina.it (192.168.2.6)
Destination: grid.grid.unina.it (143.225.229.3)
Internet Control Message Protocol
Type: 8 (Echo (ping) request)
Code: 0
Checksum: 0xabff (correct)
Identifier: 0x0300
Sequence number: 49:00

9

Esempio di traceroute (6/9)



No.	Time	Source	Destination	Protocol	Info
13	2.023553	java.comics.unina.it	grid.grid.unina.it	ICMP	Echo (ping) request

Frame 13 (106 on wire, 106 captured)
Ethernet II
Internet Protocol, Src Addr: java.comics.unina.it (192.168.2.6), Dst Addr: grid.grid.unina.it (143.225.229.3)
Version: 4
Header length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
Total Length: 92
Identification: 0x2049
Flags: 0x00
Fragment offset: 0
Time to live: 3
Protocol: ICMP (0x01)
Header checksum: 0x5fc5 (correct)
Source: java.comics.unina.it (192.168.2.6)
Destination: grid.grid.unina.it (143.225.229.3)
Internet Control Message Protocol
Type: 8 (Echo (ping) request)
Code: 0
Checksum: 0xa8ff (correct)
Identifier: 0x0300
Sequence number: 4c:00
Data (64 bytes)

10

Esempio di traceroute (7/9)



<capture> - Ethereal

File Edit Capture Display Tools Help

No.	Time	Source	Destination	Protocol	Info
13	2.023553	java.comics.unina.it	grid.grid.unina.it	ICMP	Echo (ping) request
14	2.034492	192.55.101.129	java.comics.unina.it	ICMP	Time-to-live exceeded

Frame 14 (70 on wire, 70 captured)

- Ethernet II
- Internet Protocol, Src Addr: 192.55.101.129 (192.55.101.129), Dst Addr: java.comics.unina.it (192.168.2.6)
- Internet Control Message Protocol
 - Type: 11 (Time-to-live exceeded)
 - Code: 0 (TTL equals 0 during transit)
 - Checksum: 0xf4ff (correct)
- Internet Protocol, Src Addr: java.comics.unina.it (192.168.2.6), Dst Addr: grid.grid.unina.it (143.225.229.3)
 - Version: 4
 - Header length: 20 bytes
 - Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
 - Total Length: 92
 - Identification: 0x2049
 - Flags: 0x00
 - Fragment offset: 0
 - Time to live: 1
 - Protocol: ICMP (0x01)
 - Header checksum: 0x61c5 (correct)
 - Source: java.comics.unina.it (192.168.2.6)
 - Destination: grid.grid.unina.it (143.225.229.3)
- Internet Control Message Protocol
 - Type: 8 (Echo (ping) request)
 - Code: 0
 - Checksum: 0xa8ff (correct)
 - Identifier: 0x0300
 - Sequence number: 4c:00

11

Esempio di traceroute (8/9)



<capture> - Ethereal

File Edit Capture Display Tools Help

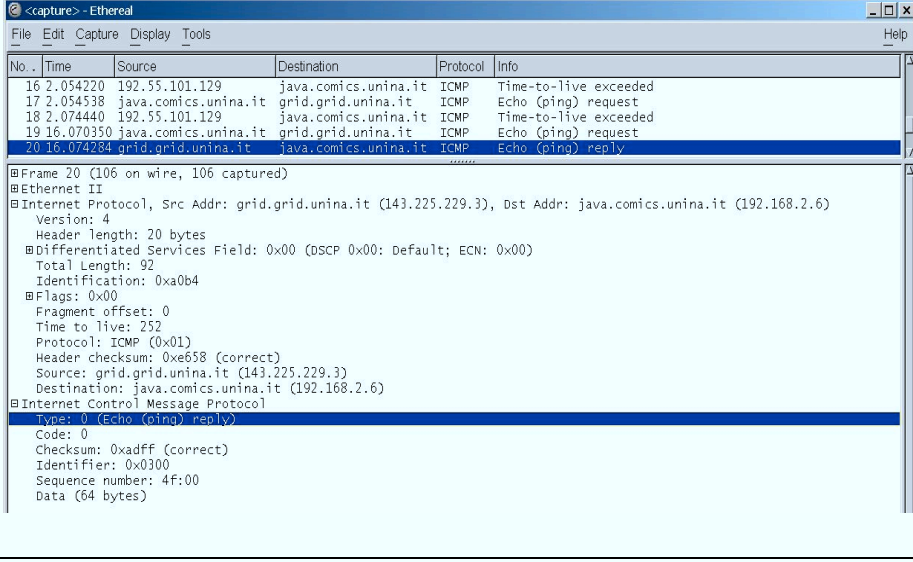
No.	Time	Source	Destination	Protocol	Info
15	2.034848	java.comics.unina.it	grid.grid.unina.it	ICMP	Echo (ping) request
16	2.054220	192.55.101.129	java.comics.unina.it	ICMP	Time-to-live exceeded
17	2.054538	java.comics.unina.it	grid.grid.unina.it	ICMP	Echo (ping) request
18	2.074440	192.55.101.129	java.comics.unina.it	ICMP	Time-to-live exceeded
19	16.070350	192.55.101.129	java.comics.unina.it	ICMP	Echo (ping) request

Frame 19 (106 on wire, 106 captured)

- Ethernet II
- Internet Protocol, Src Addr: java.comics.unina.it (192.168.2.6), Dst Addr: grid.grid.unina.it (143.225.229.3)
 - Version: 4
 - Header length: 20 bytes
 - Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
 - Total Length: 92
 - Identification: 0x2053
 - Flags: 0x00
 - Fragment offset: 0
 - Time to live: 4
 - Protocol: ICMP (0x01)
 - Header checksum: 0x5ebb (correct)
 - Source: java.comics.unina.it (192.168.2.6)
 - Destination: grid.grid.unina.it (143.225.229.3)
- Internet Control Message Protocol
 - Type: 8 (Echo (ping) request)
 - Code: 0
 - Checksum: 0xa5ff (correct)
 - Identifier: 0x0300
 - Sequence number: 4f:00
 - Data (64 bytes)

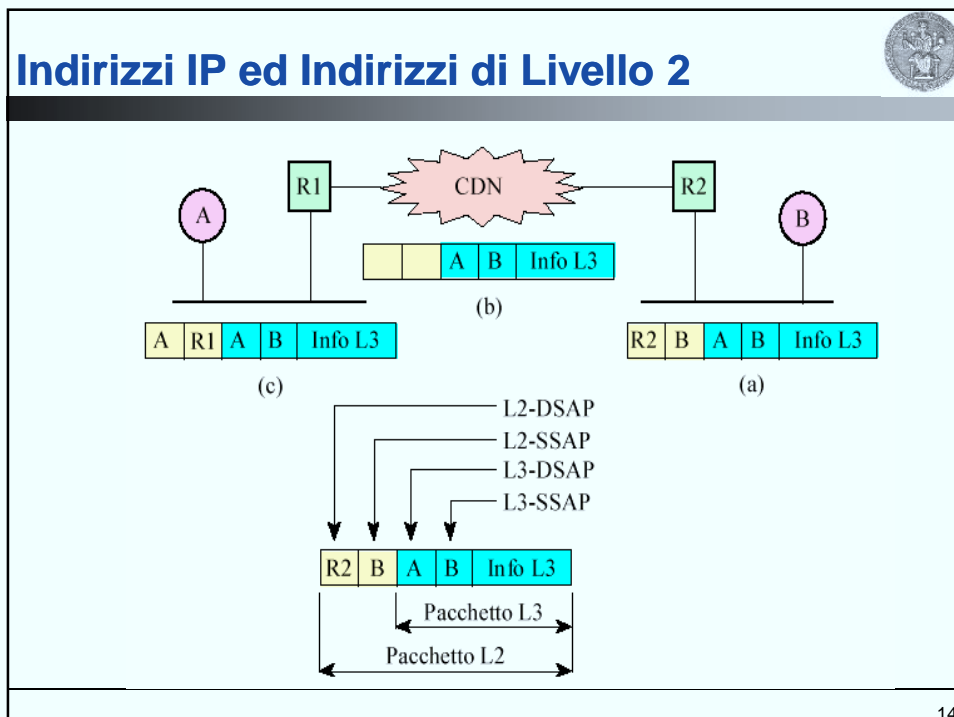
12

Esempio di traceroute (9/9)



13

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

15

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*

16

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		

17

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

18

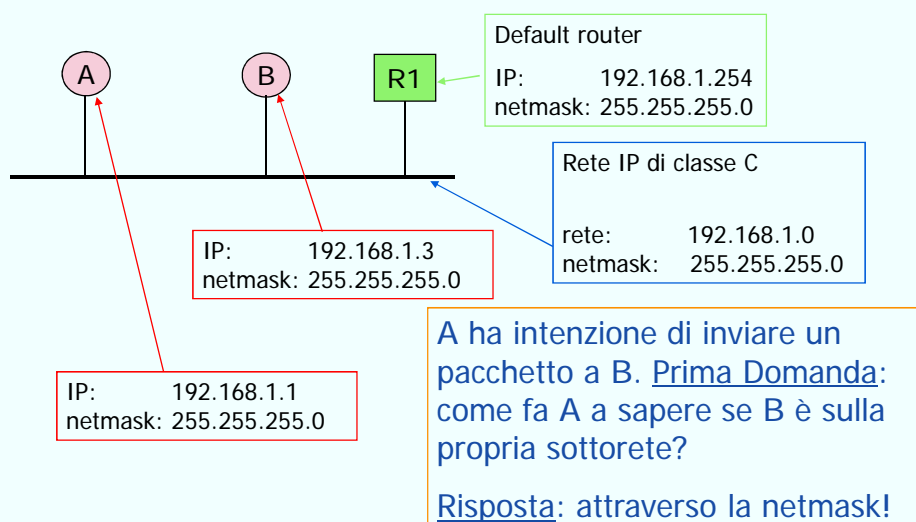
ARP: scenari tipici



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

21

ARP: primo caso (1/3)



22

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.

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

IP di B	192.168.1.2
	AND
netmask A	255.255.255.0
	=
	192.168.1.0

23

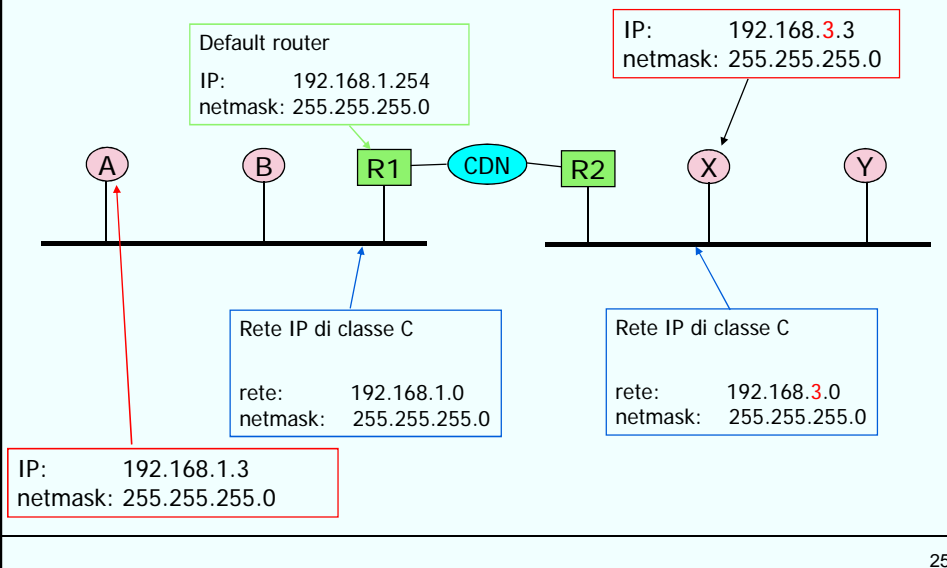
ARP: primo caso (3/3)



- 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

24

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

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

IP di X	192.168.3.3
	AND
netmask A	255.255.255.0
	=
	192.168.3.0

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

27

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

28

Monitoraggio di ARP



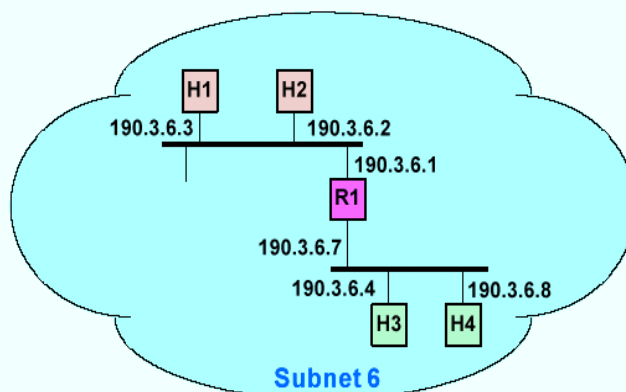
- Con il comando *arp* è possibile leggere e modificare il contenuto della arp cache
 - **arp -a** (legge il contenuto di tutta la cache)
- Con il comando *tcpdump* (o con un software tipo Ethereal...) è possibile monitorare tutto il traffico che viaggia sulla rete
 - È possibile filtrare solo i pacchetti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

29

Proxy ARP



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



30

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

31

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 broadcast)
 - *B* risponde con un messaggio **RARP reply** contenente l'indirizzo IP di *A*

32

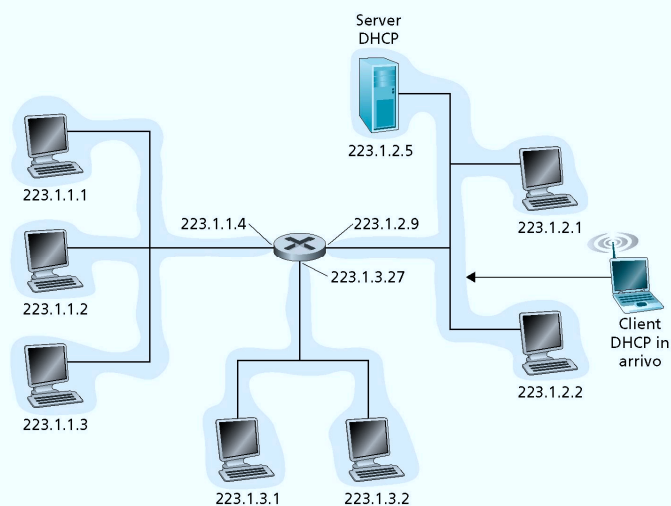
Altre soluzioni per boot remoto



- Il protocollo RARP è stato sostituito da altri protocolli più flessibili e potenti:
 - BOOTP: **BOOT**strap **P**rotocol
 - DHCP: **D**ynamic **H**ost **C**onfiguration **P**rotocol
 - Utilizzati per assegnare dinamicamente gli indirizzi agli host di una rete IP

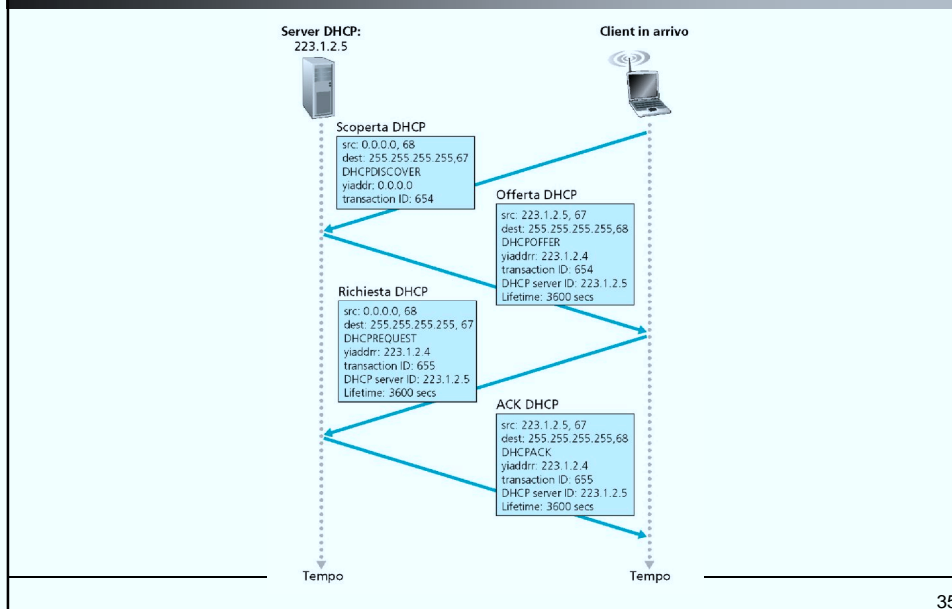
33

DHCP: scenario tipico



34

Interazione client-server via DHCP



35

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: 0xb2c065b
    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
    
```

36

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: 0xb2c065b
  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: 217.9.64.20
  IP Address: 217.9.64.3
  Option 15: Domain Name = "napoli.consortio-cini.it"
  Option 44: NetBIOS over TCP/IP Name Server = 217.9.64.200
  End Option
  Padding
  
```

37

DHCP request



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: 0xb2c065b
  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 31: 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
  
```

38

DHCP ACK



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	0.009355	192.168.2.1	192.168.2.13	DHCP	DHCP ACK - Transaction ID 0xb2c065b

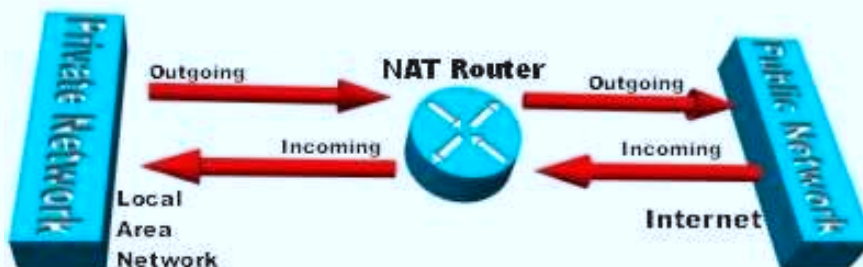
Frame 4 (590 on wire, 590 captured)
Ethernet II
Destination: 00:02:2d:09:17:be (Agere_09:17:be)
Source: 00:30:bd:96:28:fa (BELKIN_96:28:fa)
Type: IP (0x0800)
Internet Protocol, Src Addr: 192.168.2.1 (192.168.2.1), Dst Addr: 192.168.2.13 (192.168.2.13)
User Datagram Protocol, Src Port: bootps (67), Dst Port: bootpc (68)
Bootstrap Protocol
Message type: Boot Reply (2)
Hardware type: Ethernet
Hardware address length: 6
Hops: 0
Transaction ID: 0xb2c065b
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

39

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



40

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



41

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

42

NAT: un esempio

