

# Reti di Calcolatori I

Prof. Roberto Canonico

Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione

Corso di Laurea in Ingegneria Informatica

A.A. 2020-2021

## Protocolli ARP e DHCP

**I lucidi presentati al corso sono uno strumento didattico  
che NON sostituisce i testi indicati nel programma del corso**

# Nota di copyright per le slide COMICS

## Nota di Copyright

Questo insieme di trasparenze è stato ideato e realizzato dai ricercatori del Gruppo di Ricerca COMICS del Dipartimento di Informatica e Sistemistica dell'Università di Napoli Federico II. Esse possono essere impiegate liberamente per fini didattici esclusivamente senza fini di lucro, a meno di un esplicito consenso scritto degli Autori. Nell'uso dovranno essere esplicitamente riportati la fonte e gli Autori. Gli Autori non sono responsabili per eventuali imprecisioni contenute in tali trasparenze né per eventuali problemi, danni o malfunzionamenti derivanti dal loro uso o applicazione.

**Autori:**

Simon Pietro Romano, Antonio Pescapè, Stefano Avallone,  
Marcello Esposito, Roberto Canonico, Giorgio Ventre

# Trasmissione di un pacchetto IP

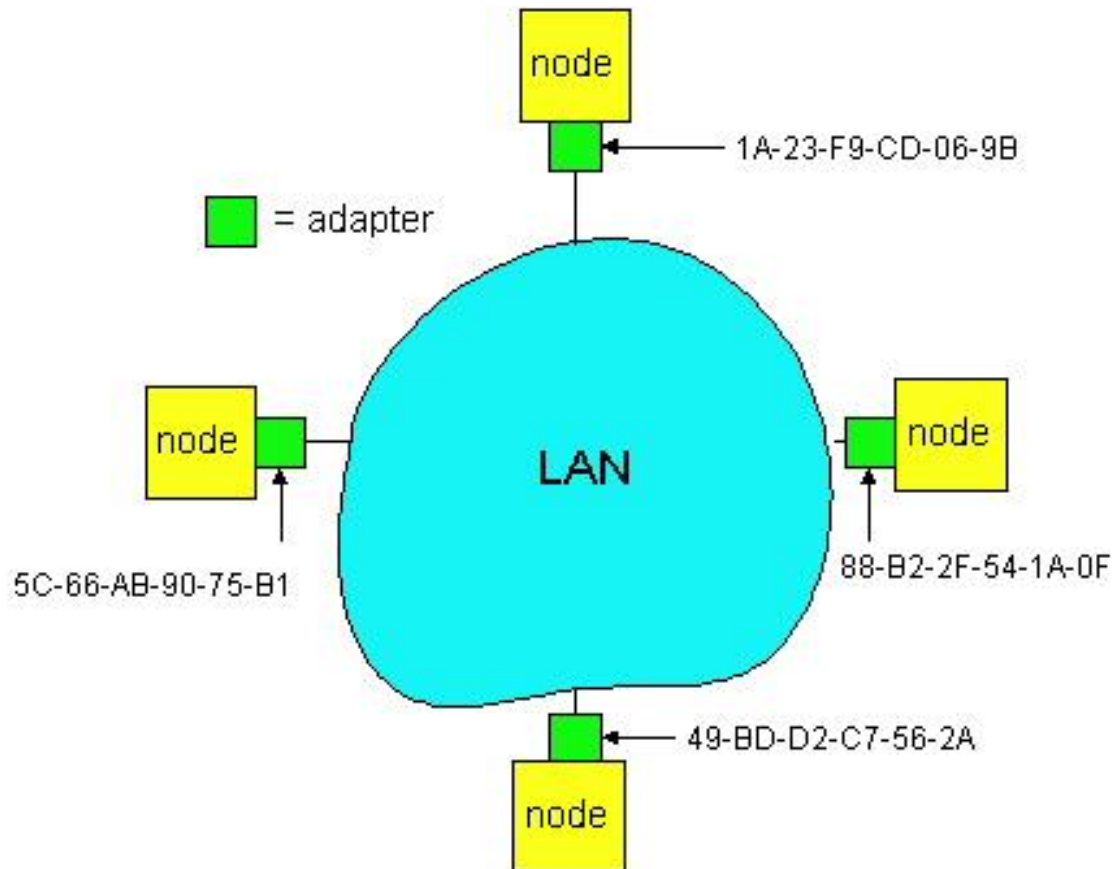
- Quando un host deve trasmettere un pacchetto IP, lo strato di livello inferiore incapsula il pacchetto in una frame di livello 2
- Le tecnologie di livello 2 sono molteplici
- Nel seguito consideriamo due casi
  - Interfacce di rete LAN basate su tecnologia Ethernet
  - Interfacce di rete WAN basate su tecnologie che realizzano collegamenti seriali punto-punto

# Trasmissione di un pacchetto IP su LAN

- L'incapsulamento di un pacchetto IP in una frame Ethernet richiede la conoscenza degli indirizzi di livello 2 (*MAC address*) dell'interfaccia mittente e destinataria su ogni hop che il pacchetto attraversa
- Due scenari:
  1. Mittente e destinatario IP del pacchetto sono nella stessa subnet IP
    - Un solo hop
  2. Mittente e destinatario IP del pacchetto sono in subnet IP diverse, collegate mediante uno o più router
    - Molteplici hop: mittente-router, [router-router, ...], router-destinatario

# Indirizzi MAC

Ogni scheda di rete in una LAN ha un indirizzo MAC (di 48 bit) univoco cablato nell'hardware della scheda dal costruttore



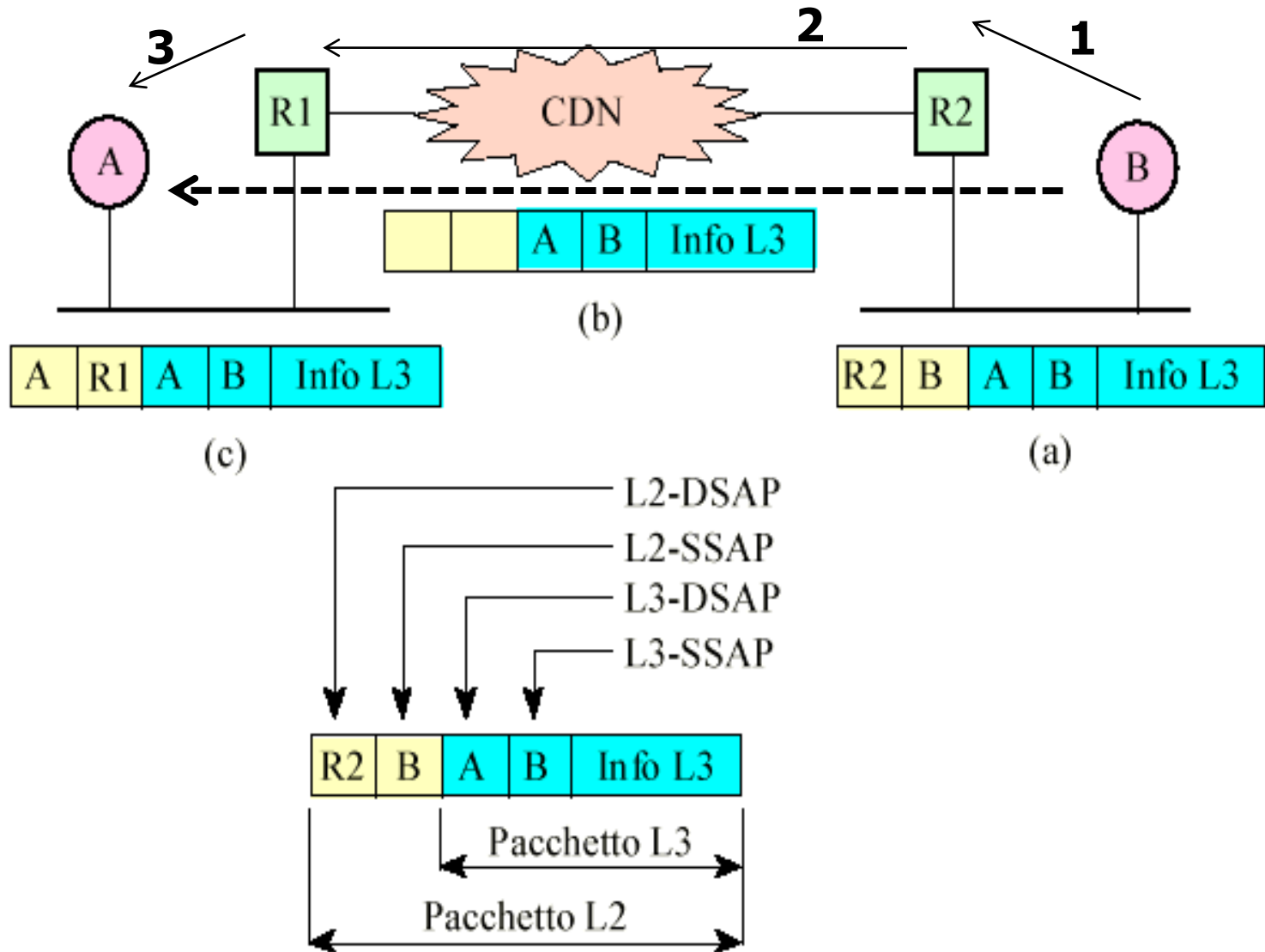
# Struttura della trama Ethernet II



- Il **preambolo** è una sequenza fissa che serve a sincronizzare i clock di mittente e destinatario
  - 7 byte con una sequenza 10101010 seguiti da un byte (SFD) con la sequenza 10101011
- Gli **indirizzi MAC** della scheda NIC del **destinatario** e del **mittente** occupano 6+6 byte
- Il campo **Type** (2 byte) indica il protocollo di livello rete del pacchetto trasportato dalla frame nella parte **Data** (di lunghezza variabile)
  - Type = 0x0800                    indica il protocollo IPv4
  - Type = 0x86DD                    indica il protocollo IPv6
  - Type = 0x0806                    indica il protocollo ARP (*Address Resolution Protocol*)
  - Cfr.: <https://www.iana.org/assignments/ieee-802-numbers/ieee-802-numbers.xhtml>
- Il campo **FCS** (*Frame Control Sequence*) contiene una sequenza di bit utile al destinatario della frame per rilevare eventuali errori (**CRC** – *Cyclic Redundancy Check*)
- Il campo Data ha una lunghezza minima di 46 byte ed una lunghezza massima di 1500 byte

L'indirizzo MAC destinazione FF:FF:FF:FF:FF:FF indica un frame trasmesso in **broadcast**: tutti gli host della rete locale (inclusi i router) ricevono il pacchetto

# Incapsulamento di pacchetti IP a livello 2



# ARP: due scenari tipici

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



# ARP - Address Resolution Protocol

- Primo scenario:
  - A deve spedire un datagram ad un host B appartenente alla medesima sottorete IP di A (*rete logica* in terminologia ARP)
  - A conosce l'indirizzo IP di B, ma non il suo indirizzo fisico
- Il protocollo ARP consente ad A di conoscere l'indirizzo fisico (MAC address) corrispondente all'indirizzo IP di B:
  - 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

<b>Hardware Type</b>		<b>Protocol Type</b>
<b>HLEN</b>	<b>PLEN</b>	<b>Operation</b>
<b>Sender Hardware Address</b>		
<b>Sender HW Address</b>		<b>Sender IP Address</b>
<b>Sender IP Address</b>		<b>Target HW Address</b>
<b>Target Hardware Address</b>		
<b>Target IP Address</b>		

# Incapsulamento dei pacchetti ARP

- Il protocollo ARP interagisce direttamente con il livello data link
- Il pacchetto ARP viene incapsulato in un frame come un protocollo di livello 3
  - 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)

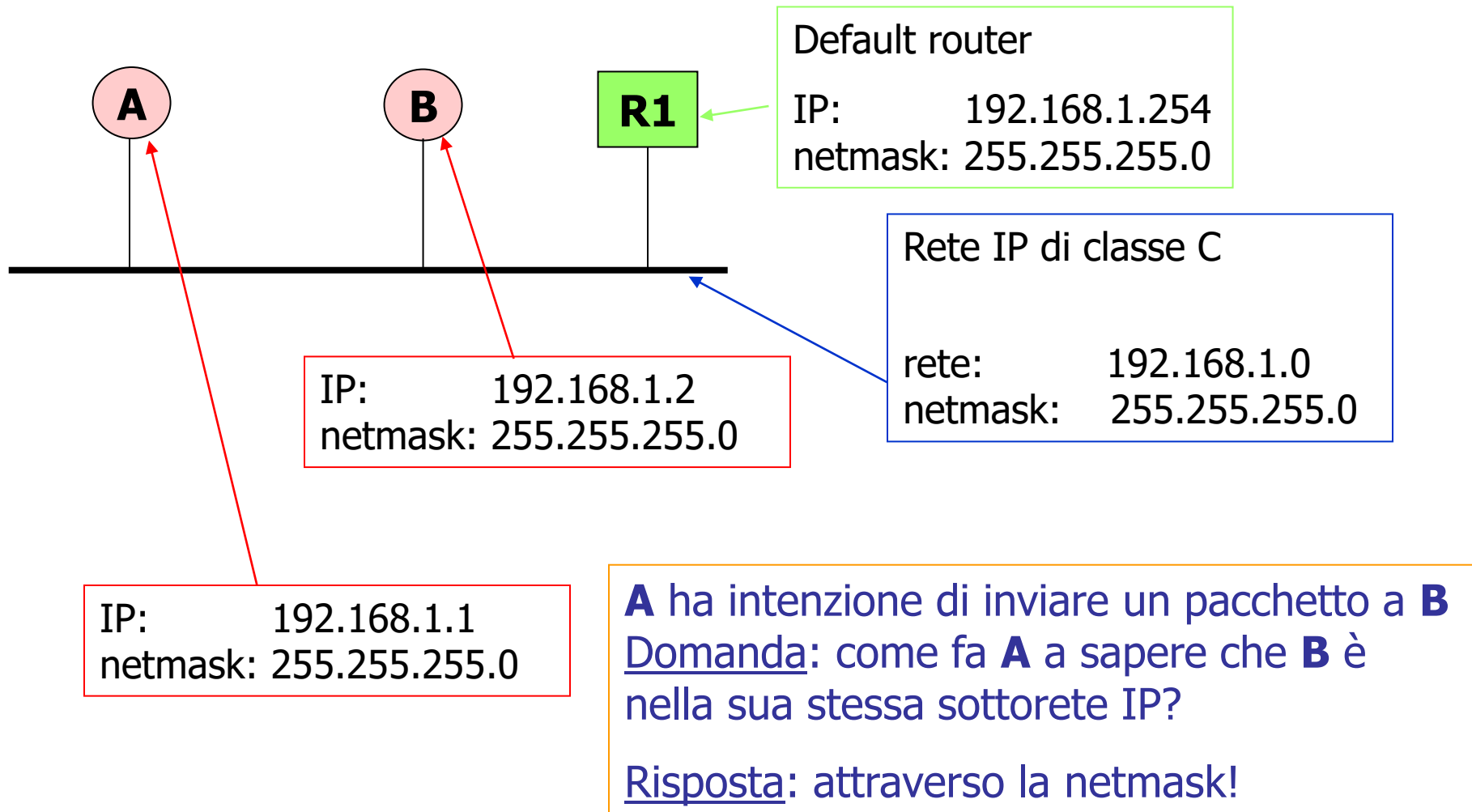
▣ Ethernet II

- 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: 00000000000000000000000000000000...

▣ 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: primo caso A→B (1/3)



# ARP: primo caso (2/3)

- Ogni computer ha un indirizzo IP ed una netmask
- La netmask serve ad individuare la propria sottorete IP
  - In Windows digitare il comando: `ipconfig /all`
- Il computer **A** esegue una AND bit-a-bit tra l'indirizzo IP destinazione e la propria netmask.
  - Nel caso precedente:

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

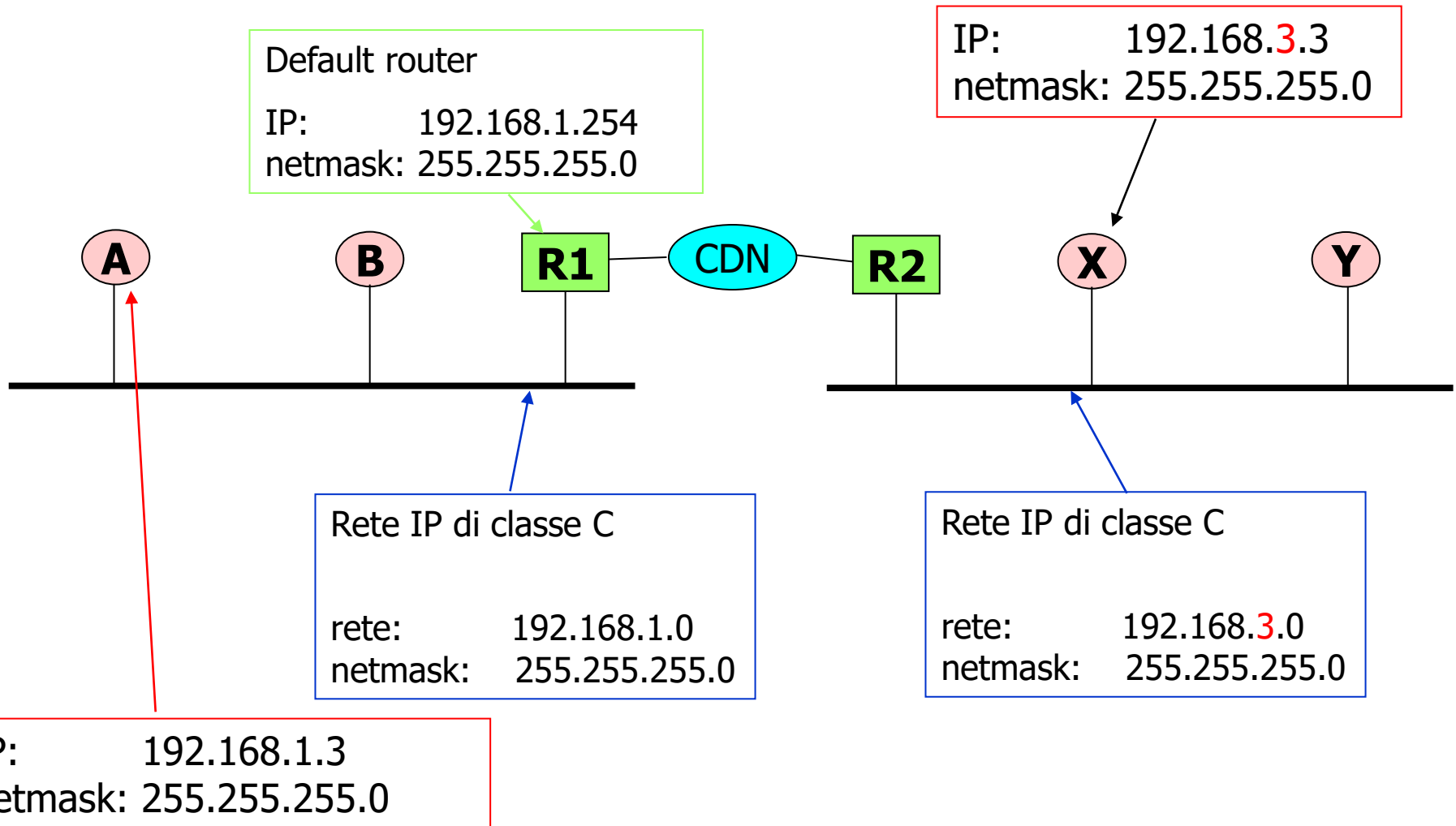
IP di B	192.168.1.2
	AND
netmask A	255.255.255.0
	=
	192.168.1.0

# ARP: primo caso (3/3)

- Se il computer **B** è sulla stessa sottorete IP la comunicazione avviene direttamente da **A** a **B**
  - A manda un pacchetto *ARP request* in broadcast per conoscere il MAC address di B
    - tale pacchetto contiene, nel campo **DEST IP**, l'indirizzo IP di B



# ARP: secondo caso A→X (1/2)



# ARP: secondo caso (2/2)

- Se **A** intende mandare un pacchetto a **X**, l'operazione di AND bit-a-bit tra la netmask e l'indirizzo IP di **X** fornisce un risultato differente
  - il destinatario non è nella stessa subnet IP del mittente

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, al primo hop il destinatario del livello 2 è l'interfaccia del router che appartiene alla subnet di A
- A 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

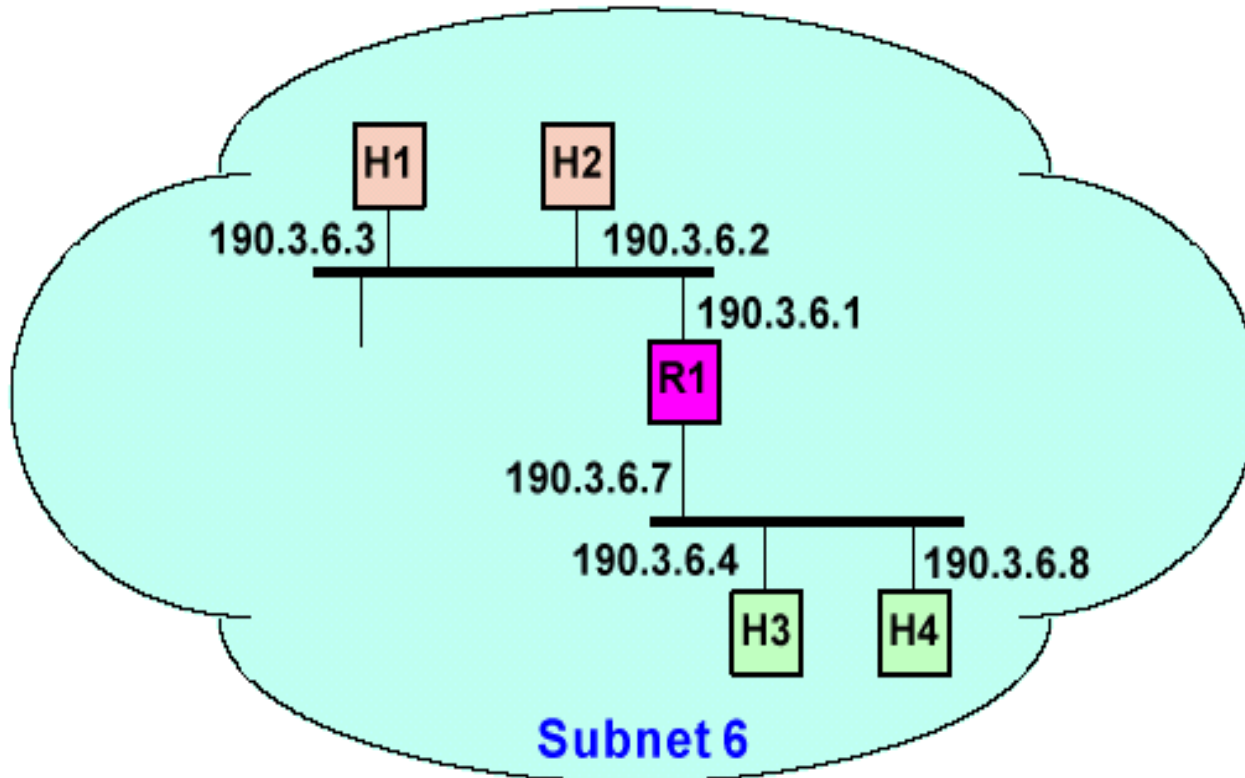
# Raffinamenti del protocollo

- 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* (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)

- 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

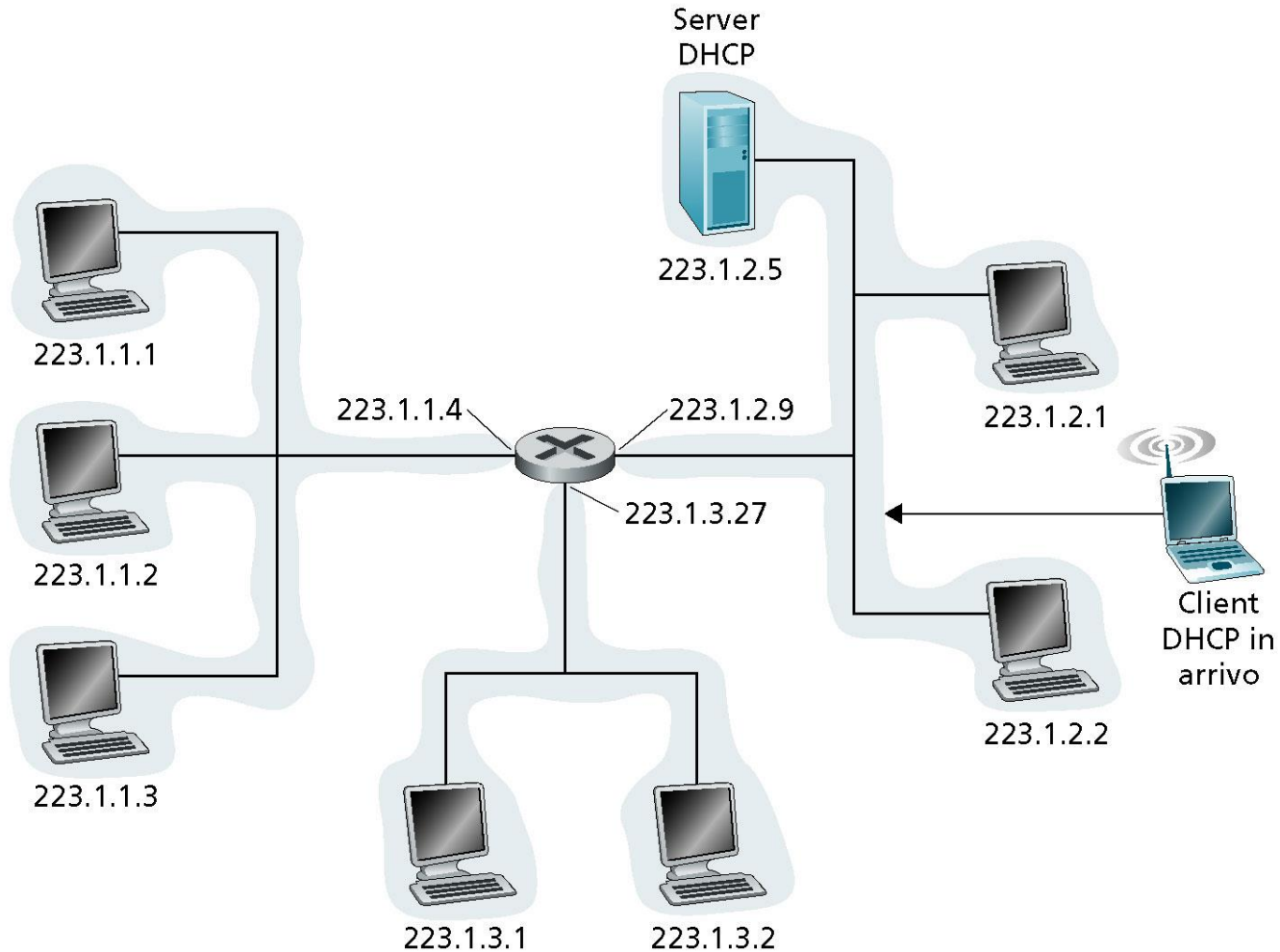
- *A* conosce il proprio indirizzo MAC, ma non conosce il proprio indirizzo IP
- L'host *B* (server RARP) conosce l'indirizzo IP di *A*
- Soluzione
  - **RARP request** sulla rete (in broadcast)
  - *B* risponde con un messaggio **RARP reply** contenente l'indirizzo IP di *A*



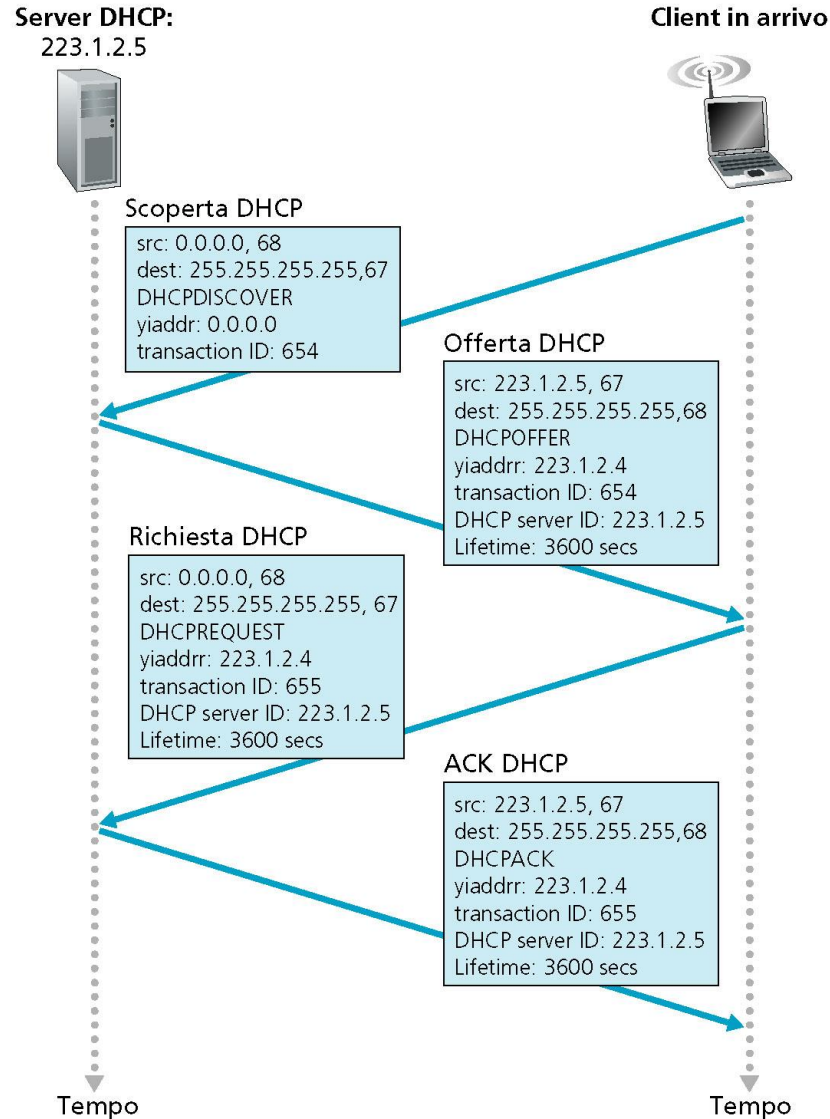
# Altre soluzioni per boot remoto

- Il protocollo RARP è stato sostituito da altri protocolli più flessibili e potenti:
  - BOOTP: **B**OOTstrap **P**rotocol
  - DHCP: **D**ynamic **H**ost **C**onfiguration **P**rotocol
  - 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.consortio-cini.it"  
 Option 44: NetBIOS over TCP/IP Name Server = 217.9.64.200  
 End Option  
 Padding

# 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: 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

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.conorzio-cini.it"  
   Option 44: NetBIOS over TCP/IP Name Server = 217.9.64.200  
   End Option  
   Padding