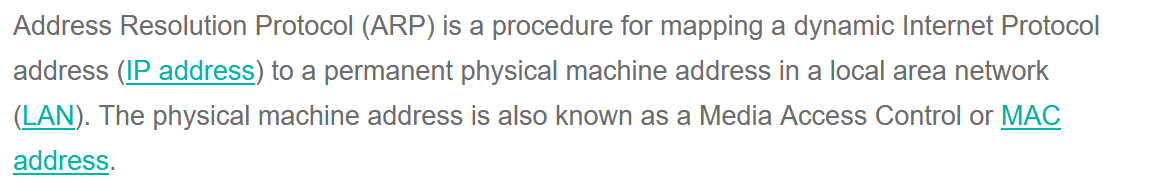
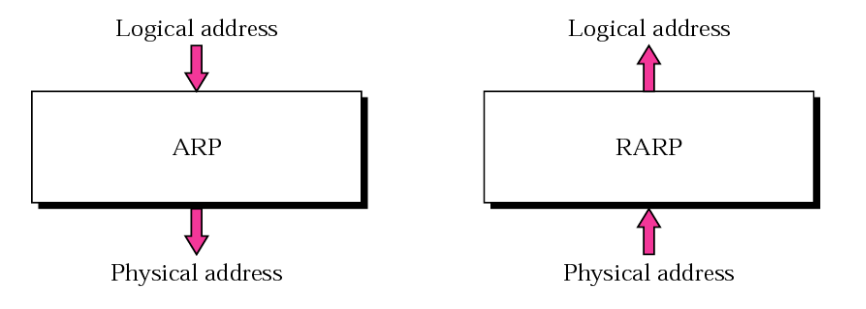
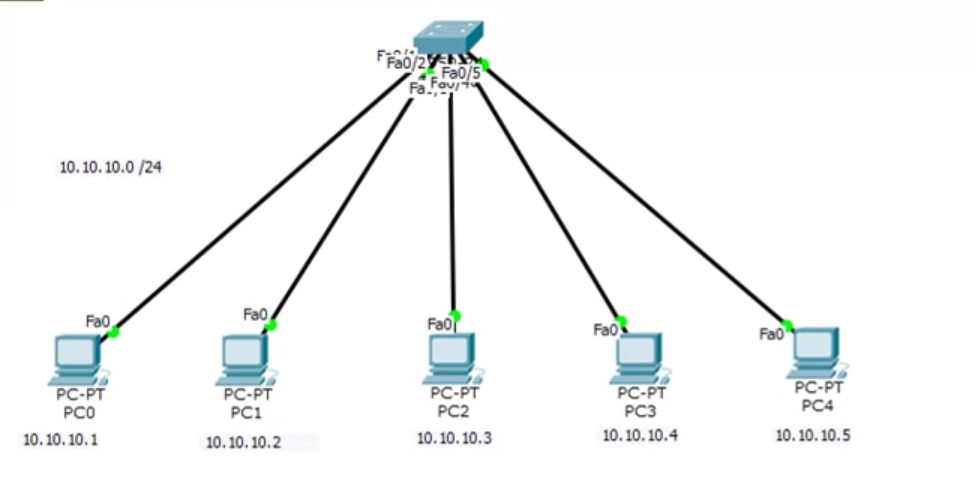
Logical Address 🡪 IP Address  
Physical Address 🡪 MAC Address

# **ARP (Address Resolution Protocol)**







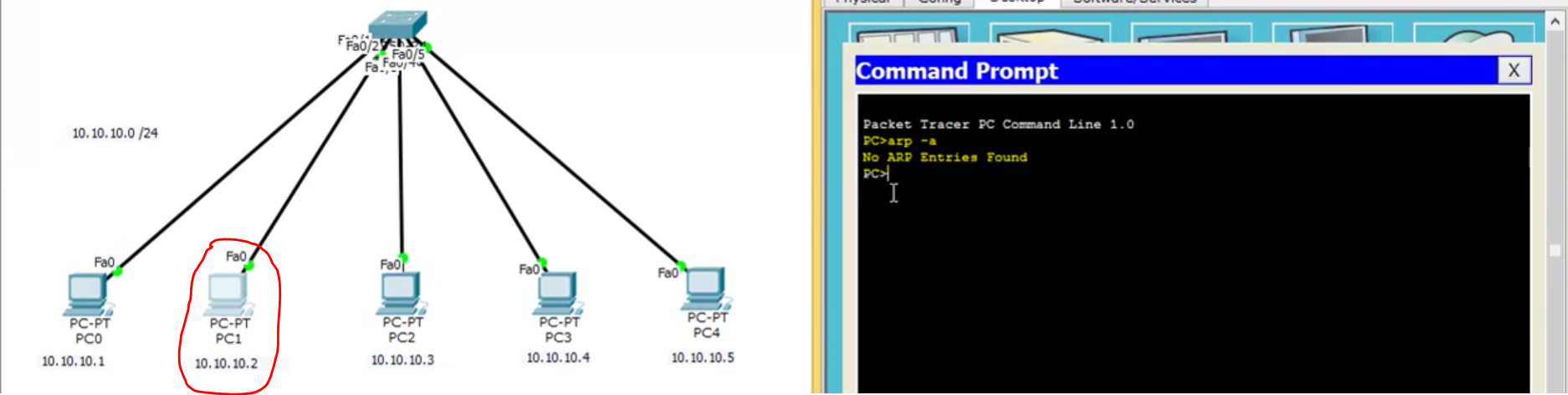
10.10.10.0/24 🡪 IP address of the entire network  
10.10.10.1 🡪 IP address of PC-0  
10.10.10.2 🡪 IP address of PC-1  
10.10.10.3 🡪 IP address of PC-2  
10.10.10.4 🡪 IP address of PC-3  
10.10.10.5 🡪 IP address of PC-4

For a PC to communicate to another PC within a local network, only IP address of that particular PC is not sufficient. Particular PC’s MAC address is also required.

Eg: PC1 wants to communicate with PC3.

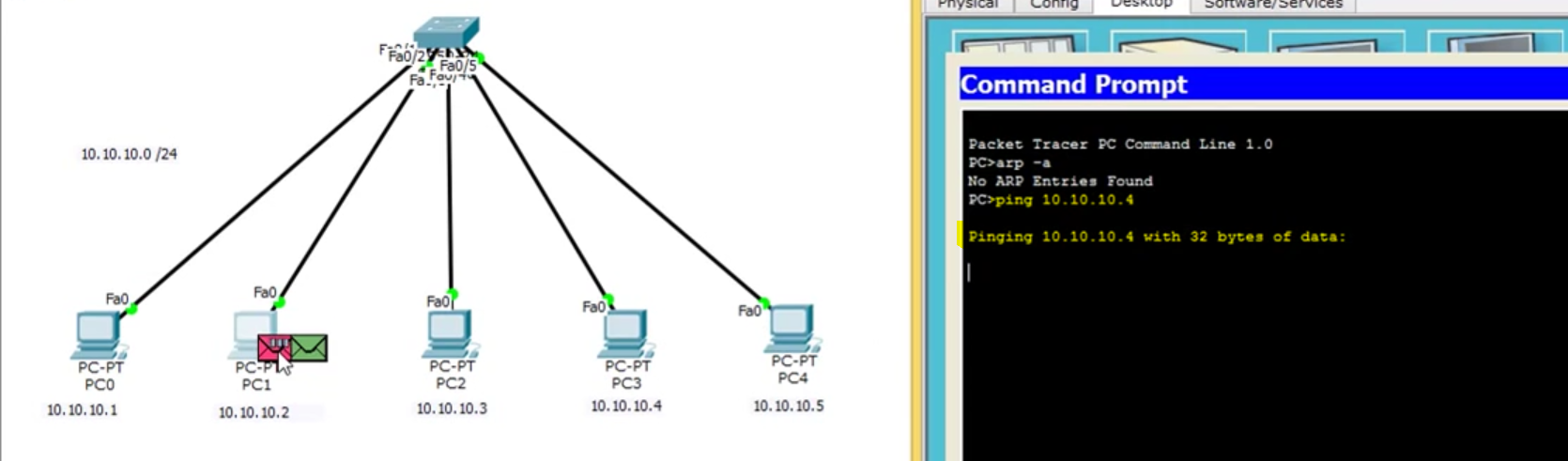
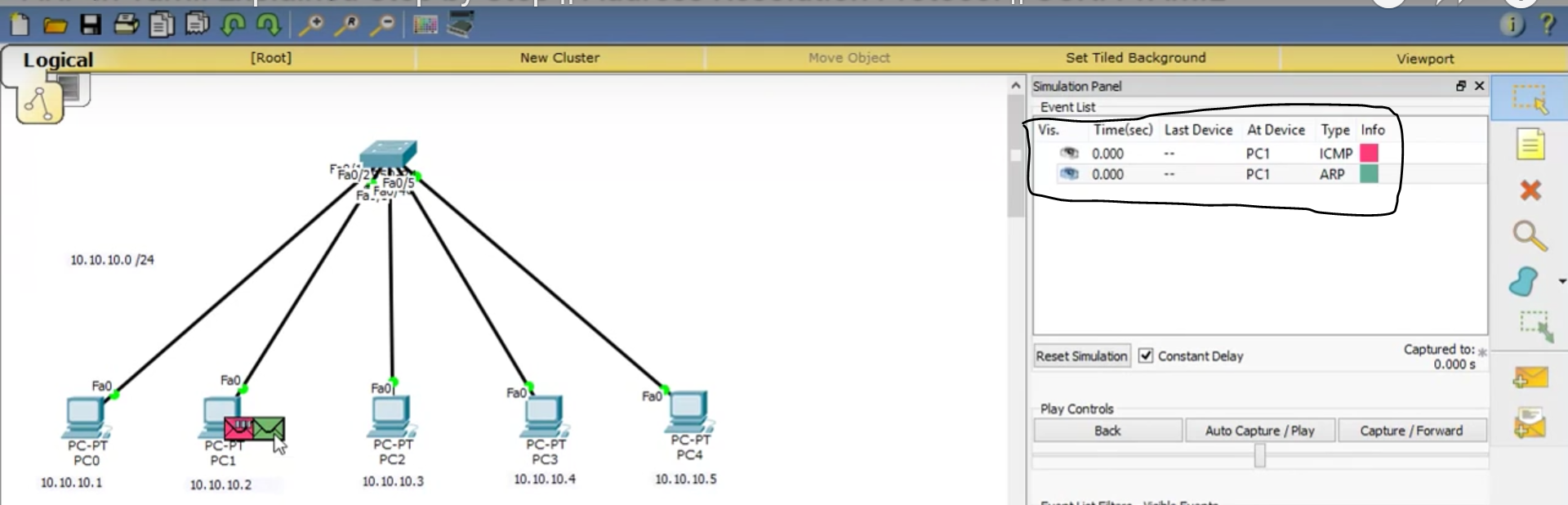
## **Checking Local Cache table**

We will check in the cache table of PC1 whether PC3’s MAC address is already available/not.

  
Cache table 🡪 PC 3’s MAC address is available. Without generating any ARP request ,PC1 can send the ICP packets directly with PC3.

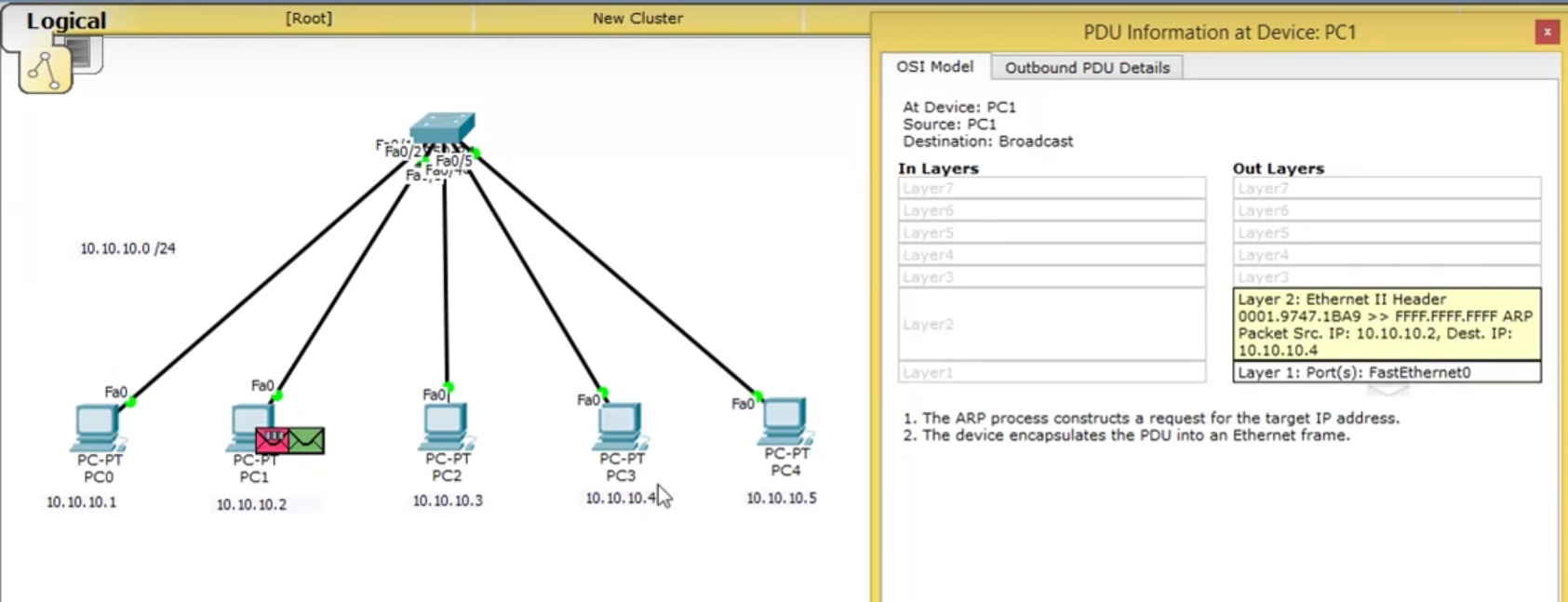
## **MAC Address not available**

Cache table 🡪 If PC 3’s MAC address is not available in PC 1’s cache table,  
PC1 will send the ARP request to the PC whose IP is 10.10.10.4 to the switch.   
**ARP packet 🡪 green envelope**  
**ICMP packet 🡪 red envelope**

## **Inside ARP packet**

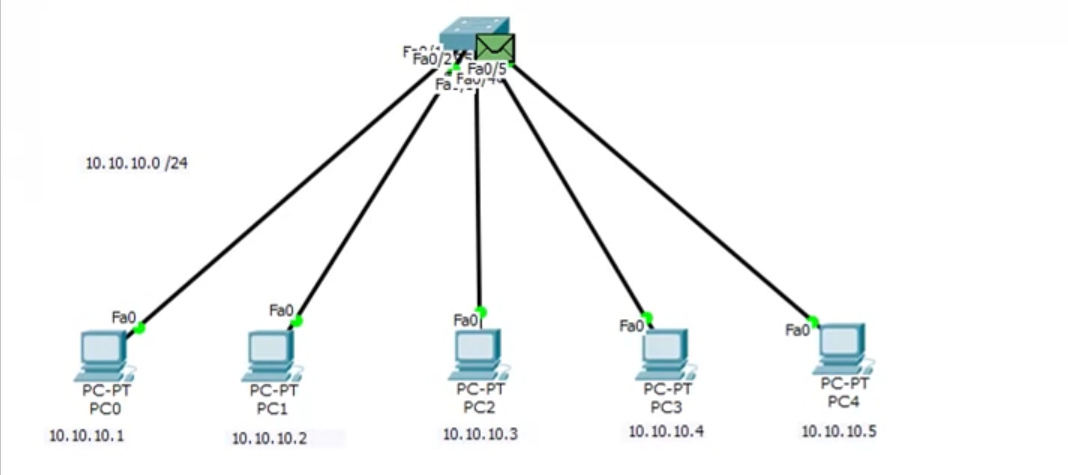
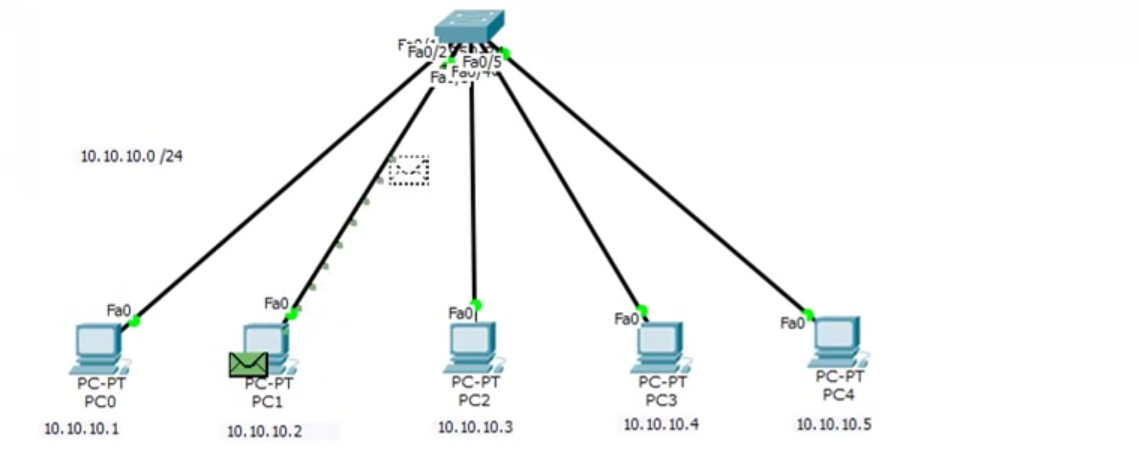
What is inside ARP packet ??



* Source MAC Address (PC-1) 🡪 0001.9747.1BA9
* Source IP Address (PC-1) 🡪 10.10.10.2
* Destination IP Address (PC-3) 🡪 10.10.10.4
* Destination MAC Address (PC-3) 🡪 FFFF.FFFF.FFFF   
  [ Initially PC-3’s MAC address is not known]

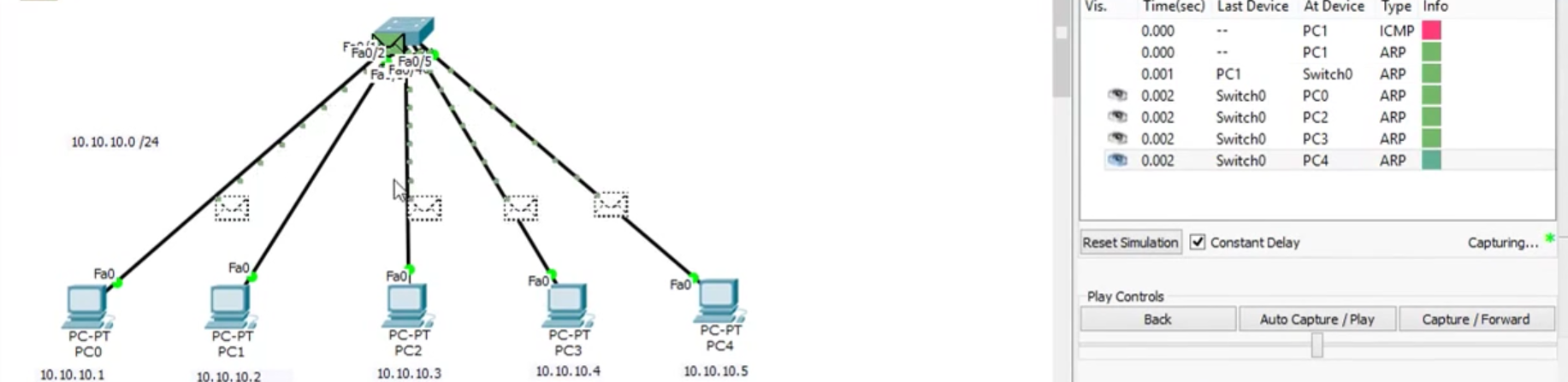
## **ARP packet to switch**

Then that **ARP packet 🡪 green envelope** is sent to the switch

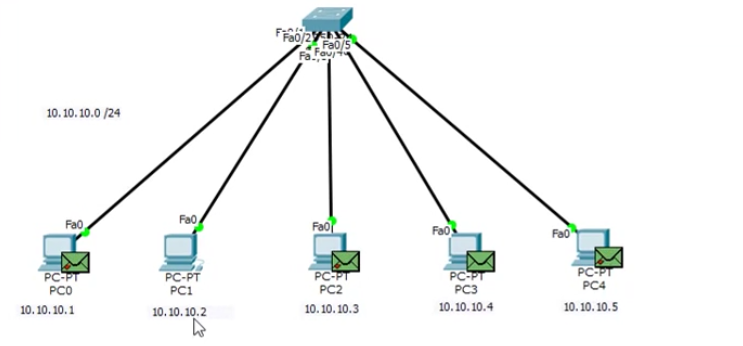
 **ARP packet 🡪 green envelope** reached the switch.

## **(Request) Broadcasting the ARP packets to the entire network**

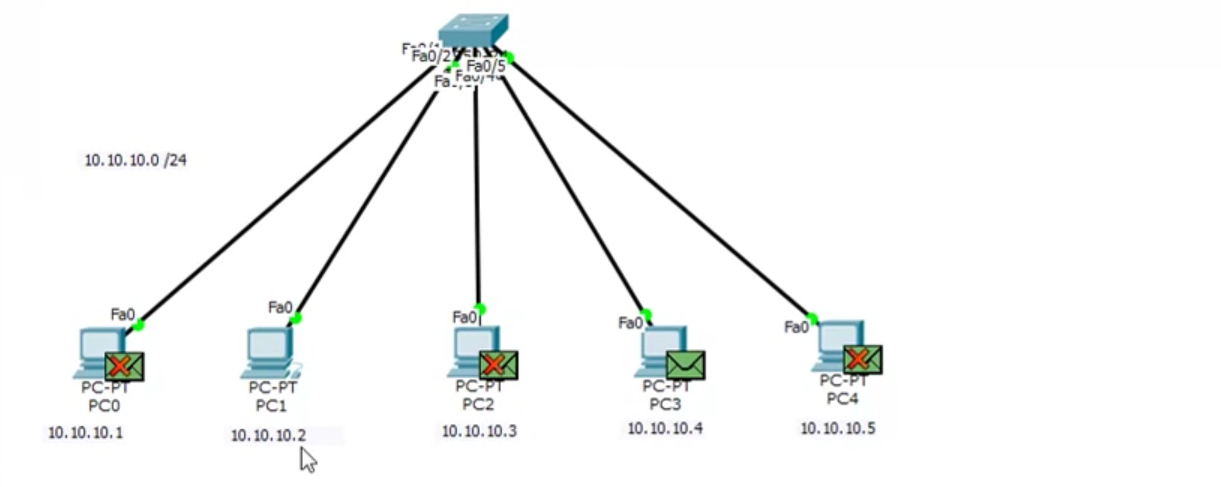
Request 🡪 Broad-cast



Then that switch will broadcast the **ARP packet 🡪 green envelope** to the entire PC’s in the network except PC-1 (since ARP packet was sent from PC-1)

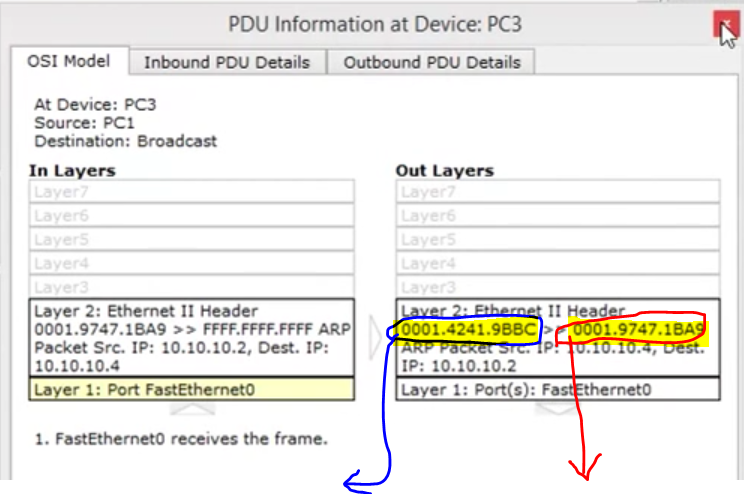
  
Now the ARP packet have successfully reached to all other PC’s in the entire network.

## **(Response) From the entire network**

Now all the PC’s opens the ARP packet and checks the destination IP address.  
PC-0 🡪 destination IP address not matching X  
PC-2 🡪 destination IP address not matching X  
PC-3 🡪 destination IP address matching   
PC-4 🡪 destination IP address not matching X

X marks indicates that 10.10.10.4 does not belong to PC-0, PC-2 and PC-4.  
10.10.10.4 belongs to PC-3. So there is no X mark.

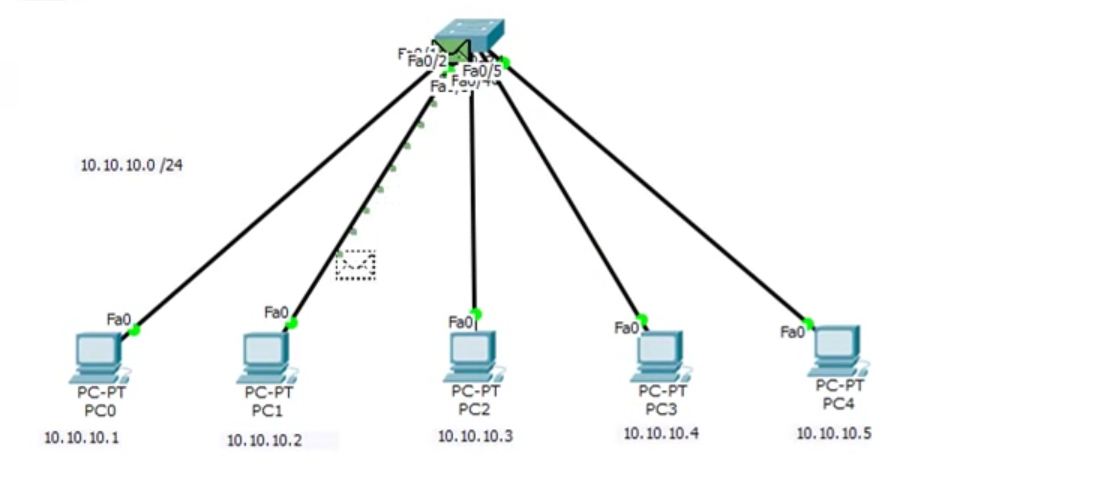
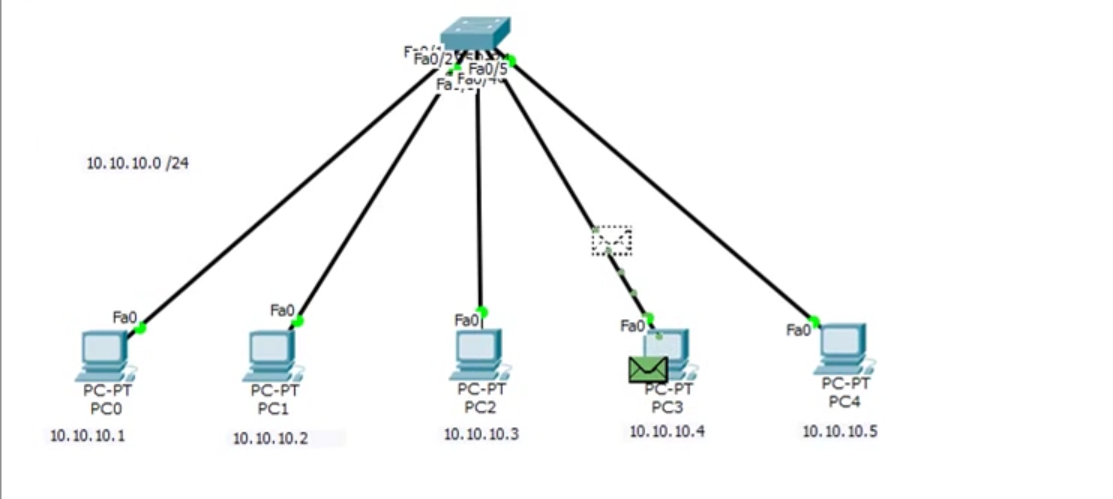
## **Opening the message in PC-3**



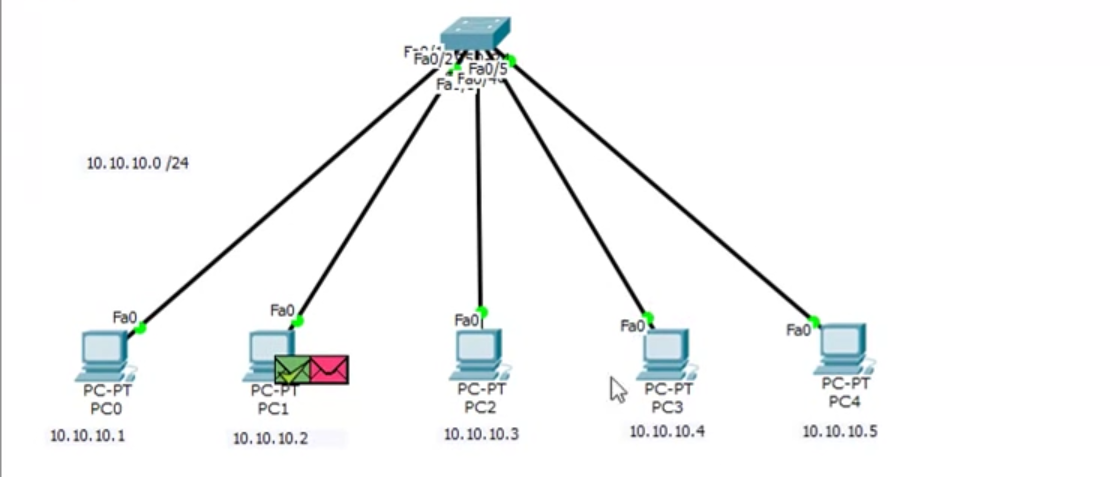
PC-3’s MAC address PC-1’s MAC address

Source IP (PC-3) 🡪 10.10.10.4  
Destination IP (PC-1) 🡪 10.10.10.2

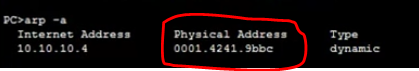
## **Sending the reply message from PC-3 to PC-1**



## **PC-1 knows the MAC address of PC-3**

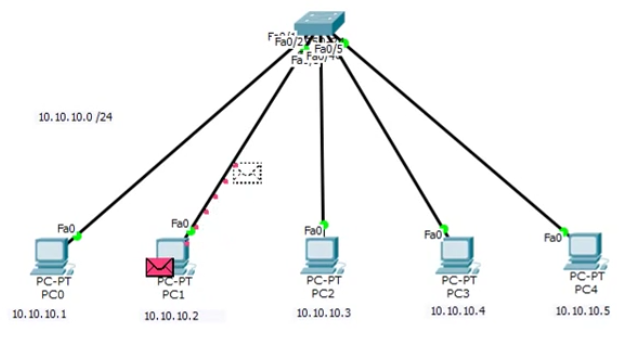


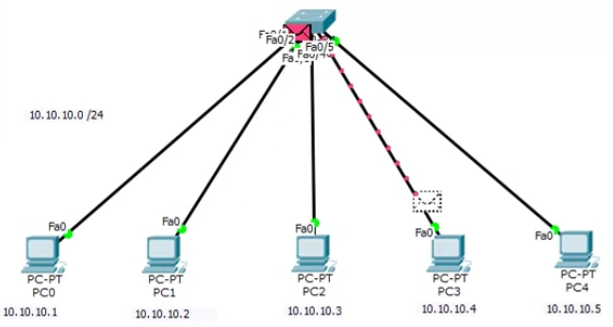
* Indicates that PC-1 learns the MAC address of PC-3 and PC-1 is ready to send the ICMP packets.



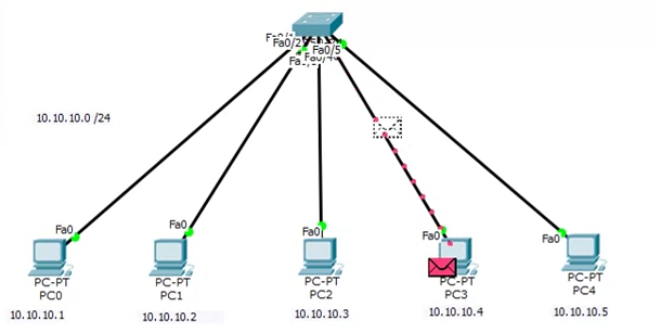
## **PC-1 to PC-3 (message)**

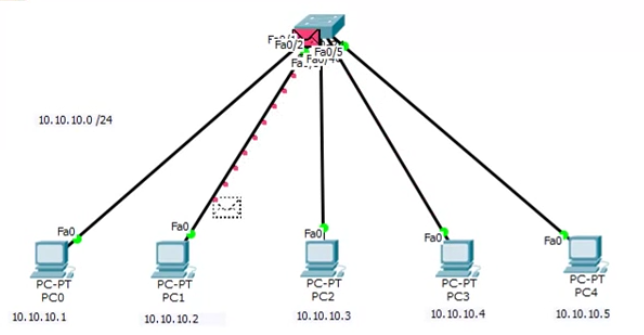
PC-1 sends the **ICMP packet 🡪 red envelope** directly to PC-3 via the switch.

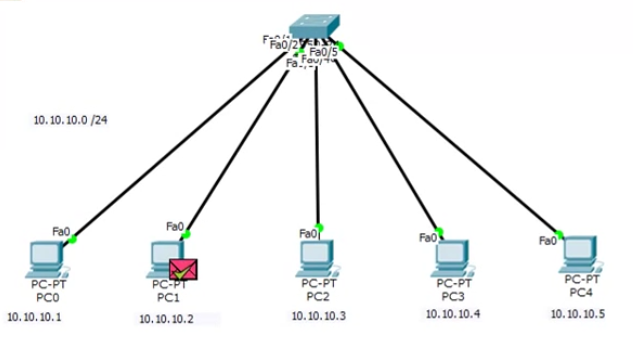
PC-1 sends the ICMP packet (message) to PC-3 via a switch.

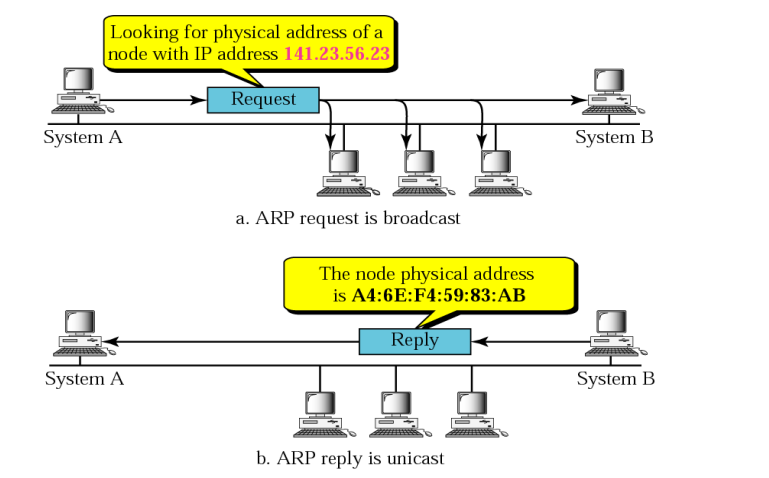
PC-3 receives the ICMP packet (message) from PC-1 via a switch.

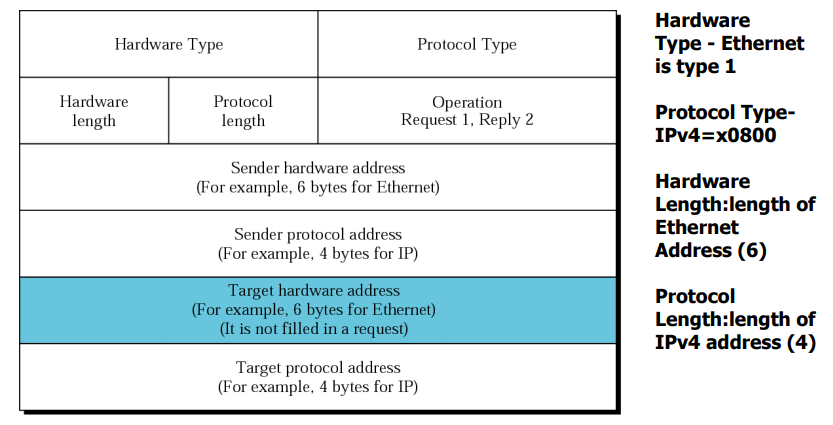
## **PC-3 to PC-1 (reply)**

  
PC-3 sends the ICMP packet (reply) to PC-1 via a switch.

  
PC-1 receives the ICMP packet (reply) via a switch from PC-3.

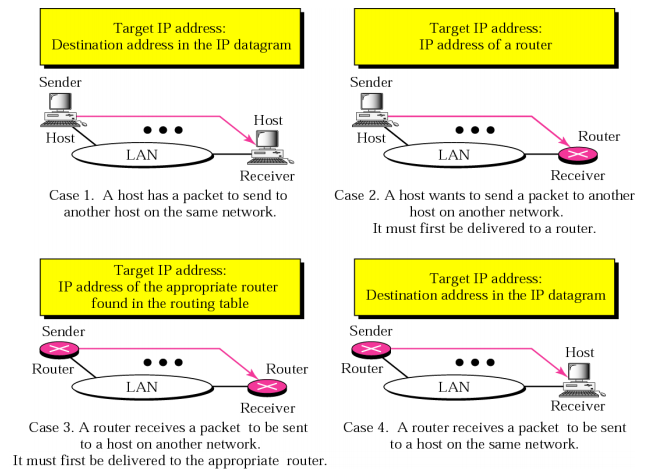
  
ICMP packet is successfully received by the PC-1

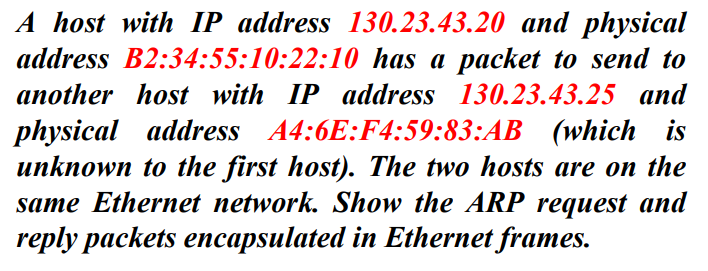


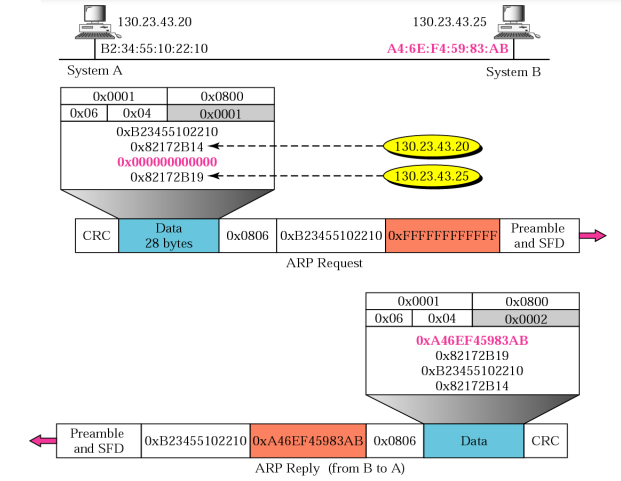


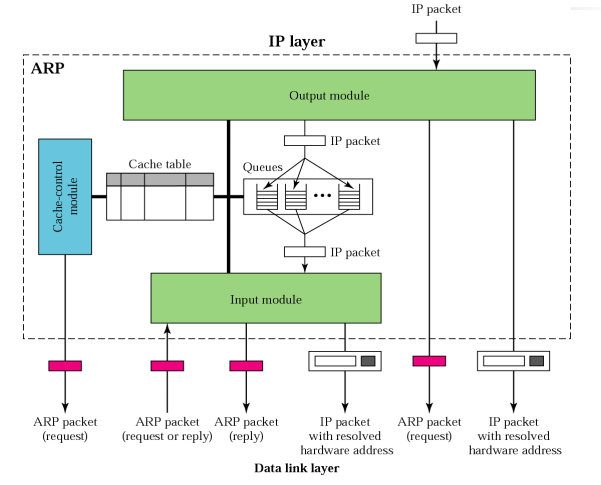
Hardware-type 🡪 what type of network ??, ethernet, wifi, carrier usage etc.  
Protocol-type 🡪 IPv4 [32 bits = 4 bytes] (or) IPv6 [128 bits = 16 bytes]  
Hardware length 🡪 Length of the MAC address.  
Protocol length 🡪 Length of IPv4, IPv6 and MAC address [48 bits = 6 bytes]  
Operation 🡪 Request=1 (Broadcasting from the switch)  
Response=1 (Unicasting from a particular PC)

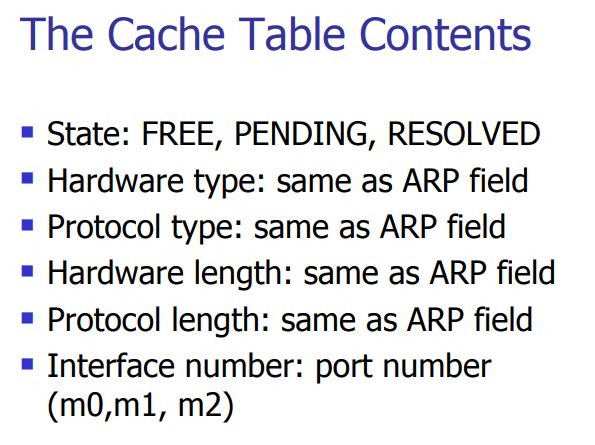
Sender Hardware address 🡪

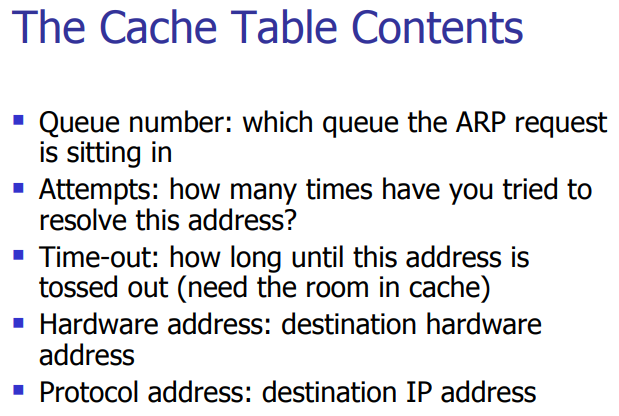


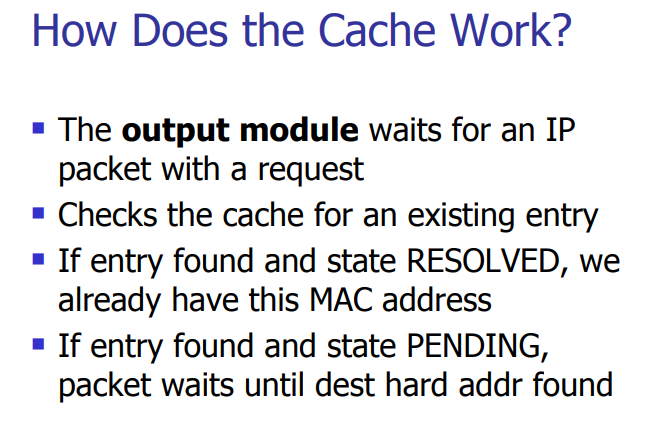


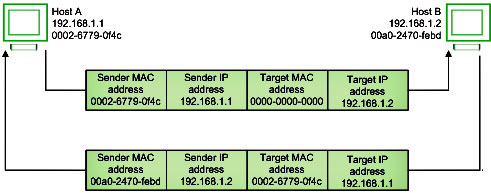


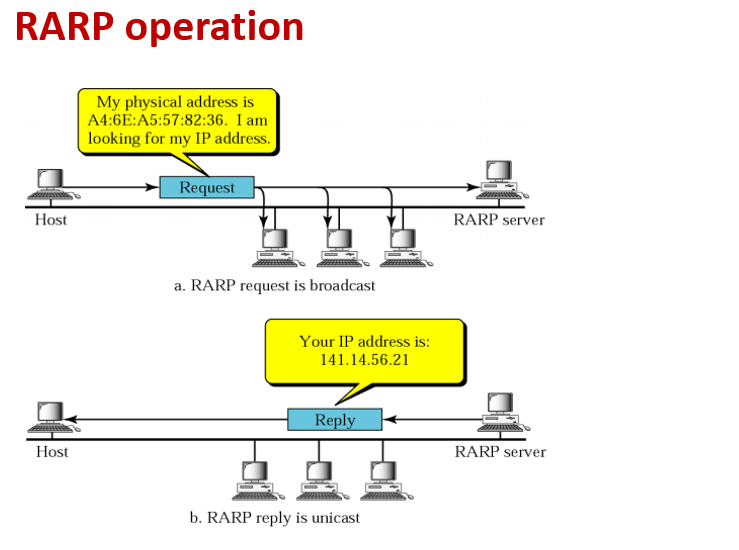




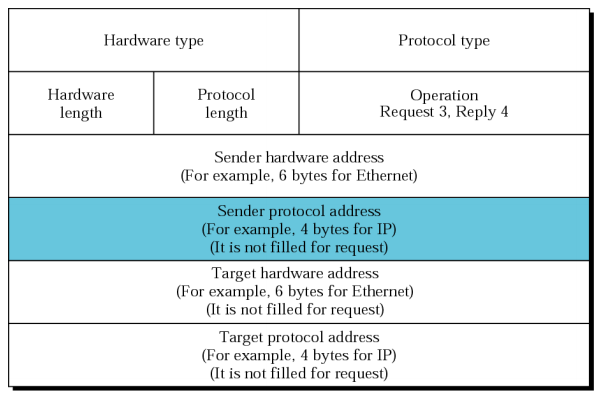








RARP header



# RARP (Reverse Address Resolution Protocol)