CS-204: COMPUTER NETWORKS

Lecture 7

Chapter 21- Network Layer: Address Mapping & Error Reporting Protocols

Instructor: Dr. Vandana Kushwaha

1. INTRODUCTION

IP (Internet Protocol) was designed as a **best-effort delivery protocol**, but it **lacks some features** such as **flow control** and **error control**. To make IP more responsive it takes help of other protocols as depicted in Figure 21.1:

- Protocols to create a mapping between physical and logical addresses: **ARP** (Address Resolution Protocol).
- Protocols to create a reverse mapping i.e. mapping a physical address to a logical address: **RARP, BOOTP**, and **DHCP**.
- Lack of flow and error control in the Internet Protocol has resulted in another protocol, **ICMP**. It reports congestion and some types of errors in the network or destination host.

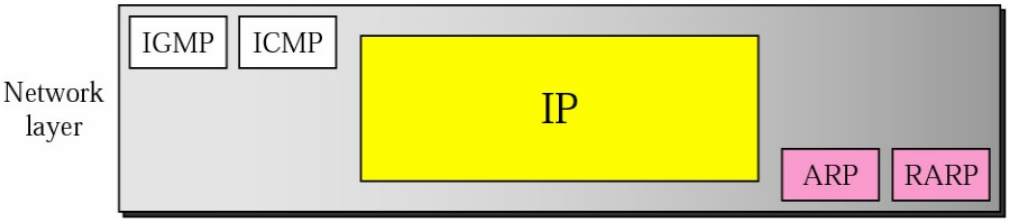


Figure 21.1 Position of ARP and RARP Protocols

2. ADDRESS MAPPING

An internet is made of a combination of physical networks connected by internetworking devices such as routers. A packet starting from a source host may pass through several different physical networks before finally reaching the destination host. The hosts and routers are recognized at the **network level** by their **logical (IP) addresses**, while at the **physical level**, they are recognized by their **physical (MAC) addresses.** Thus delivery of a packet to a host or a router requires **two levels** of **addressing**: **logical (IP)** and **physical (MAC)**.

We need to be able to map a logical address to its corresponding physical address and vice versa. These can be done by using either static or dynamic mapping.

2.1. Static mapping

Static mapping involves in the creation of a table that associates a logical address with a physical address. This table is stored in each machine on the network. Each machine that knows, for example, the IP address of another machine but not its physical address can look it up in the table. Static mapping has some **limitations** because physical addresses may change in the following ways:

- A machine could change its NIC (Network Interface Card), resulting in a new physical address.
- In some LANs, such as LocalTalk, the physical address changes every time the computer is turned on.
- A mobile computer can move from one physical network to another, resulting in a change in its physical address.

To implement these changes, a static mapping table must be updated periodically. This overhead could affect network performance.

2.2. Dynamic mapping

In such mapping each time a machine knows one of the two addresses (logical or physical), it can use a protocol to find the other one.

3. Mapping Logical to Physical Address: ARP

ARP stands for **Address Resolution Protocol** which is one of the most important protocols of the Network layer in the OSI model. ARP finds the physical address, also known as Media Access Control (MAC) address, of a host from its known IP address Figure 21.2.

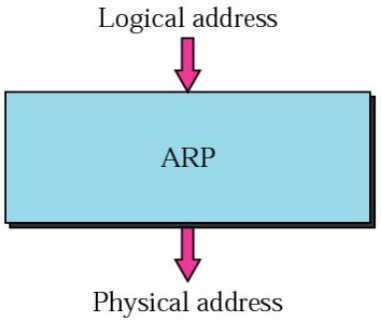


Figure 21.2: ARP Mapping

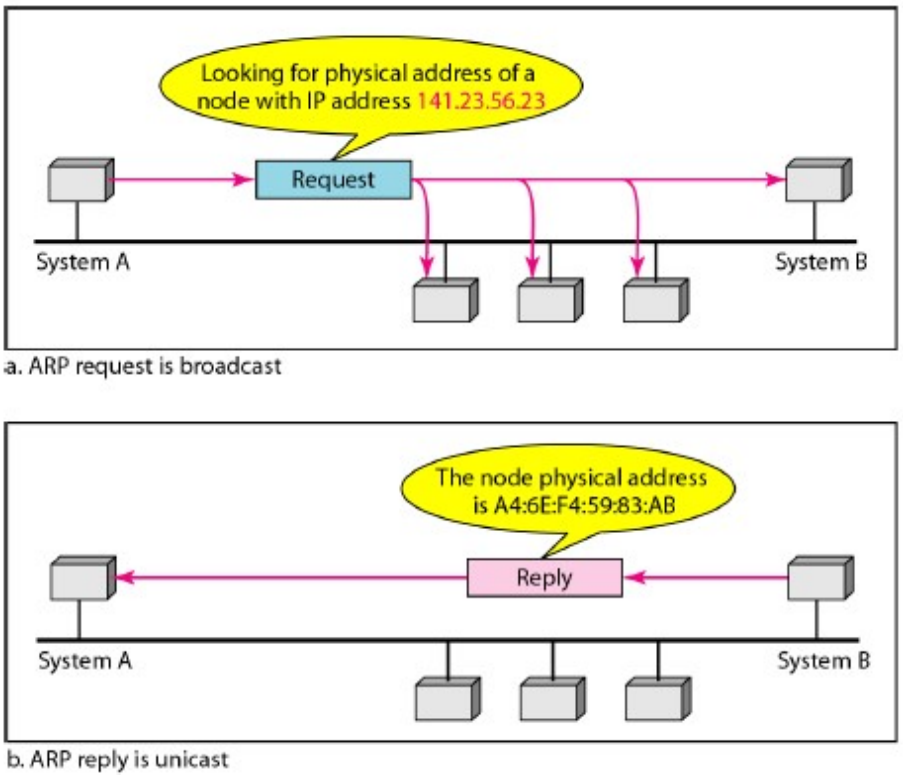


Figure 21.3 ARP operation

Following **steps** are involved in logical to physical address mapping:

a. The host or the router sends an **ARP query packet**. The ARP query packet includes the physical and IP addresses of the sender and the IP address of the receiver. As the sender does not know the physical address of the receiver, the **ARP query is broadcast over the network** (see Figure 21.3).

b. Every host or router on the network receives and processes the ARP query packet, but only the intended recipient recognizes its IP address and sends back an **ARP response packet**.

c. The ARP response packet contains the recipient's IP and physical addresses. The ARP response packet is **unicast directly to the inquirer** (host/router) by using the physical address received in the query packet.

Let us consider an **example** In Figure 21.3a, the system on the left (A) has a packet that needs to be delivered to another system (B) with IP address 141.23.56.23. System A needs to pass the packet to its data link layer for the actual delivery, but it does not know the physical address of the recipient. It uses the services of ARP by asking the ARP protocol to send a broadcast ARP request packet to ask for the physical address of a system with an IP address of 141.23.56.23.This packet is received by every system on the physical network, but only system B will answer it, as shown in Figure 21.3b. System B sends an ARP reply packet that includes its physical address. Now system A can send all the packets it has for this destination by using the physical address it received.

3.1. Cache Memory

Using ARP is inefficient if system A needs to broadcast an ARP request for each IP packet it needs to send to system B. ARP can be useful if the ARP reply is cached (kept in cache memory for a while) because a system normally sends several packets to the same destination. A system that receives an ARP reply stores the mapping in the cache memory and keeps it for 20 to 30 minutes unless the space in the cache is exhausted. Before sending an ARP request, the system first checks its cache to see if it can find the mapping.

3.2. ARP Packet Format

Figure 21.4 shows the format of an ARP packet.

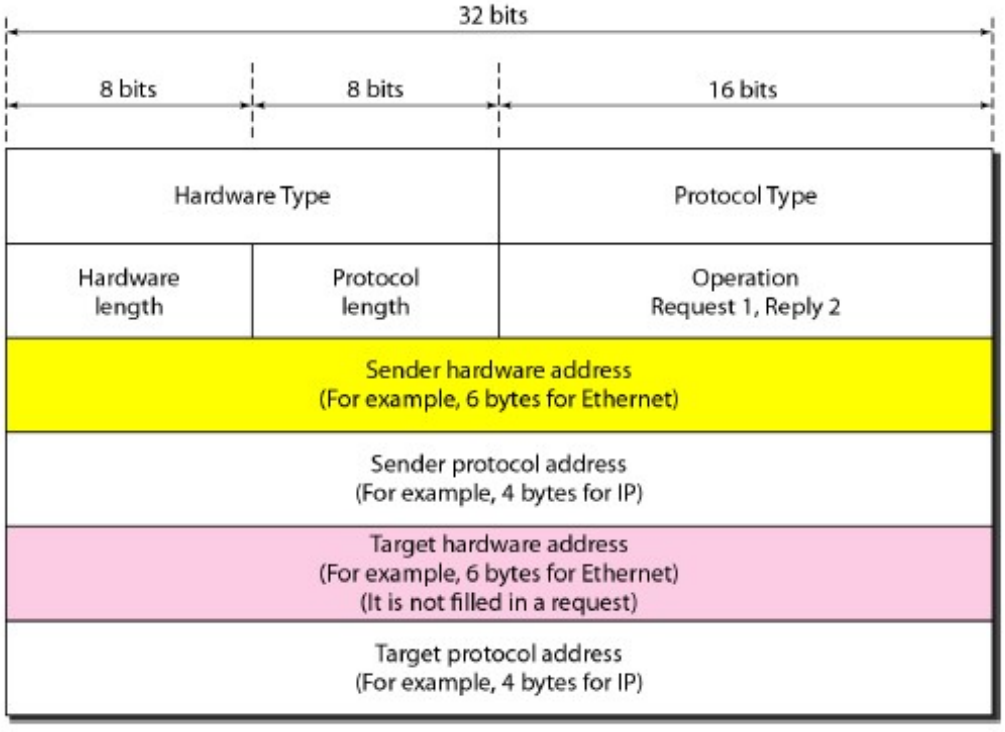


Figure 21.4 ARP packet

The fields are as follows:

a. **Hardware type.** This is a 16-bit field defining the type of the network on which ARP is running. Each LAN has been assigned an integer based on its type. For **example**, Ethernet is given type 1. ARP can be used on any physical network.

b. **Protocol type.** This is a 16-bit field defining the protocol. For example, the value of this field for the IPv4 protocol is 080016, ARP can be used with any higher-level protocol.

c. **Hardware length**. This is an 8-bit field defining the length of the physical address in bytes. For example, for Ethernet the value is 6.

d. **Protocol length**. This is an 8-bit field defining the length of the logical address in bytes. For example, for the IPv4 protocol the value is 4.

e. **Operation**. This is a 16-bit field defining the type of packet. Two packet types are defined: ARP request (1) and ARP reply (2).

f. **Sender hardware address**. This is a variable-length field defining the physical address of the sender. For example, for Ethernet this field is 6 bytes long.

g. **Sender protocol address**. This is a variable-length field defining the logical (for example, IP) address of the sender. For the IP protocol, this field is 4 bytes long.

h. **Target hardware address**. This is a variable-length field defining the physical address of the target. For example, for Ethernet this field is 6 bytes long. For an ARP request message, this field is all 0s because the sender does not know the physical address of the target.

i. **Target protocol address**. This is a variable-length field defining the logical (for example, IP) address of the target. For the IPv4 protocol, this field is 4 bytes long.

3.3. Encapsulation

An ARP packet is encapsulated directly into a data link frame. For example, in Figure 21.5 an ARP packet is encapsulated in an Ethernet frame. Note that the type field indicates that the data carried by the frame are an ARP packet.

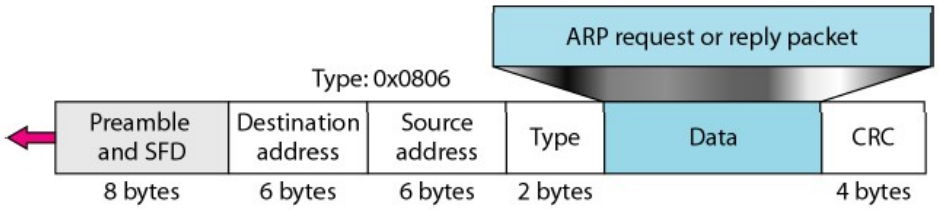


Figure 21.5 Encapsulation of ARP packet

3.4. ARP Operation

Let us see how ARP functions on a typical internet. First we describe the steps involved. Then we discuss the four cases in which a host or router needs to use ARP. These are the steps involved in an ARP process:

1. The sender knows the IP address of the target. We will see how the sender obtains this shortly.

2. IP asks ARP to create an ARP request message, filling in the sender physical address, the sender IP address, and the target IP address. The target physical address field is filled with 0s.

3. The message is passed to the data link layer where it is encapsulated in a frame by using the physical address of the sender as the source address and the physical broadcast address as the destination address.

4. Every host or router receives the frame. Because the frame contains a broadcast destination address, all stations remove the message and pass it to ARP. All machines except the one targeted drop the packet. The target machine recognizes its IP address.

5. The target machine replies with an ARP reply message that contains its physical address. The message is unicast.

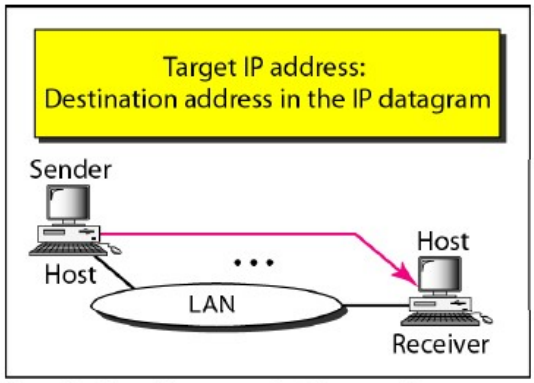
6. The sender receives the reply message. It now knows the physical address of the target machine.

7. The IP datagram, which carries data for the target machine, is now encapsulated in a frame and is unicast to the destination.

3.5. Four Different Cases of ARP Operation

The following are **four different cases** in which the services of ARP can be used (see Figure 21.6(a) to 21.6(d)).

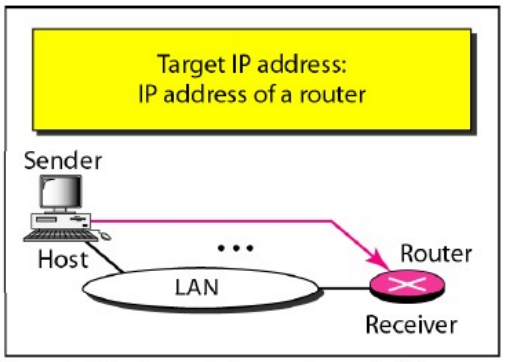
Case 1: *The sender is a host and wants to send a packet to another host on the same network*. In this case, the logical address that must be mapped to a physical address is the destination IP address in the datagram header.



Case 1. A host has a packet to send to another host on the same network.

Figure 21.6(a): Case 1

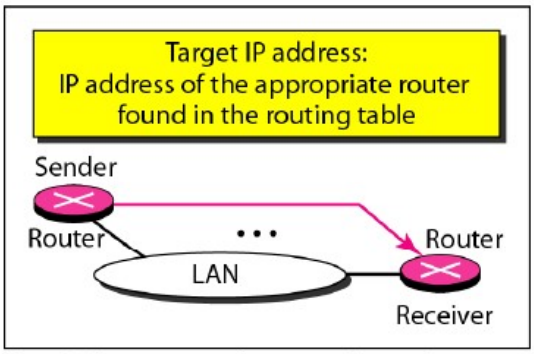
Case 2: *The sender is a host and wants to send a packet to another host on another network*. In this case, the host looks at its routing table and finds the IP address of the next hop (router) for this destination. If it does not have a routing table, it looks for the IP address of the default router. The IP address of the router becomes the logical address that must be mapped to a physical address.



Case 2. A host wants to send a packet to another host on another network. It must first be delivered to a router.

Figure 21.6(b): Case 2

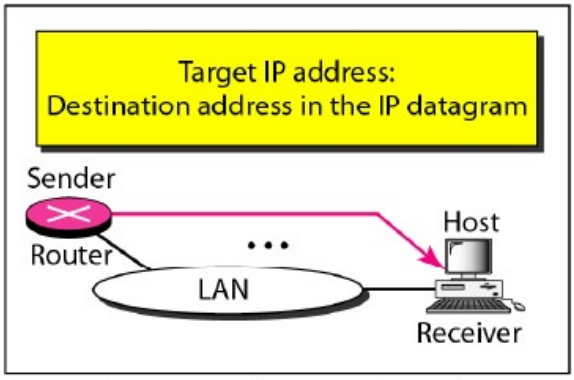
Case 3: *The sender is a router that has received a datagram destined for a host on another network.* It checks its routing table and finds the IP address of the next router. The IP address of the next router becomes the logical address that must be mapped to a physical address.



Case 3. A router receives a packet to be sent to a host on another network. It must first be delivered to the appropriate router.

Figure 21.6(c): Case 3

Case 4: *The sender is a router that has received a datagram destined for a host on the same network.* The destination IP address of the datagram becomes the logical address that must be mapped to a physical address.



Case 4. A router receives a packet to be sent to a host on the same network.

Figure 21.6(d): Case 4

Example 21.1

A host with IP address 130.23.43.20 and physical address B2:34:55: 10:22: 10 has a packet to send to another host with IP address 130.23.43.25 and physical address A4:6E:F4:59:83:AB (which is unknown to the first host). The two hosts are on the same Ethernet network. Show the ARP request and reply packets encapsulated in Ethernet frames.

Solution

Figure 21.7 shows the **ARP request** and **reply packets**. Note that the **ARP data field** in this case is **28 bytes**, and that the individual addresses do not fit in the 4-byte boundary. That is why we do not show the regular 4-byte boundaries for these addresses.

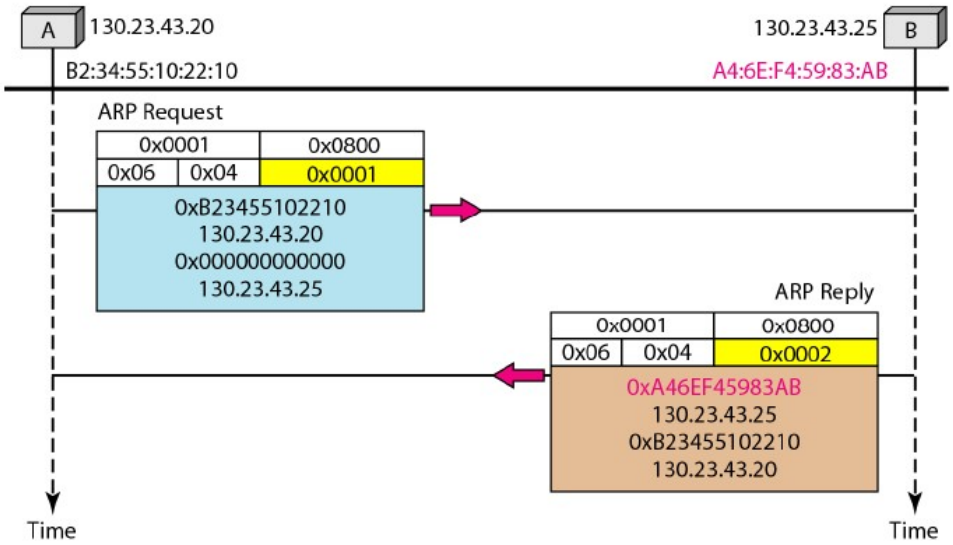


Figure 21.7 Example 21.1, an ARP request and reply

3.6. Proxy ARP

A proxy ARP is an ARP that acts on behalf of a set of hosts. Whenever a router running a proxy ARP receives an ARP request looking for the IP address of one of these hosts, the router sends an ARP reply announcing its own hardware (physical) address. After the router receives the actual IP packet, it sends the packet to the appropriate host or router. Let us give an example. In Figure 21.8 the ARP installed on the right-hand host will answer only to an ARP request with a target IP address of 141.23.56.23.

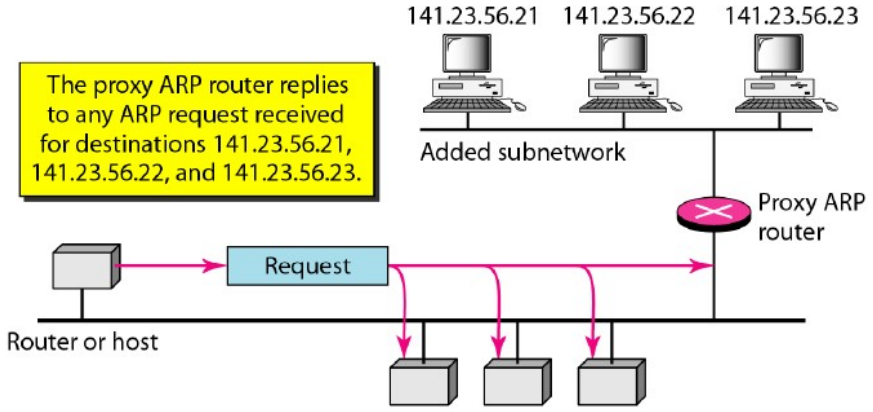


Figure 21.8 Proxy ARP

However, the administrator may need to create a subnet without changing the whole system to recognize subnetted addresses. One solution is to add a router running a proxy ARP. In this case, the router acts on behalf of all the hosts installed on the subnet. When it receives an ARP request with a target IP address that matches the address of one of its host (141.23.56.21, 141.23.56.22, or 141.23.56.23), it sends an ARP reply and announces its hardware address as the target hardware address. When the router receives the IP packet, it sends the packet to the appropriate host.

4. Mapping Physical to Logical Address: RARP, BOOTP, and DHCP

There are occasions in which a **host knows its physical address, but needs to know its logical address** Figure 21.9. This may happen in **two cases**:

Case 1: *A diskless station is just booted*. The station can find its physical address by checking its interface, but it does not know its IP address.

Case 2: *An organization does not have enough IP addresses to assign to each station*; it needs to assign IP addresses on demand. The station can send its physical address and ask for a short time lease.

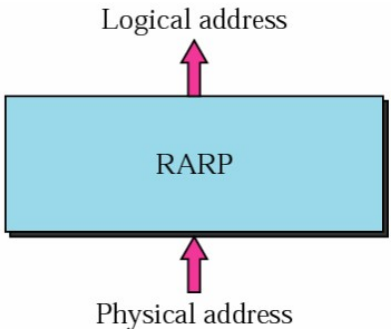


Figure 21.9: RARP Mapping

4.1. RARP

Reverse Address Resolution Protocol (RARP) finds the logical address for a machine that knows only its physical address. To create an IP datagram, a host or a router needs to know its own IP address or addresses. The IP address of a machine is usually read from its configuration file stored on a disk file. However, a **diskless machine** is usually booted from ROM, which has minimum booting information. The ROM is installed by the manufacturer. It cannot include the IP address because the IP addresses on a network are assigned by the network administrator. The machine can get its physical address (by reading its NIC, for example), which is unique locally. It can then use the physical address to get the logical address by using the RARP protocol.

4.1.1. RARP Operation

RARP operation is displayed in Figure 21.10

- **a.** A **RARP request** is created and **broadcast** on the local network.
- **b.** Another machine on the local network that knows all the IP addresses will respond with a **RARP reply**.
- **c.** The requesting machine must be running a **RARP client** program; the responding machine must be running a **RARP server** program.

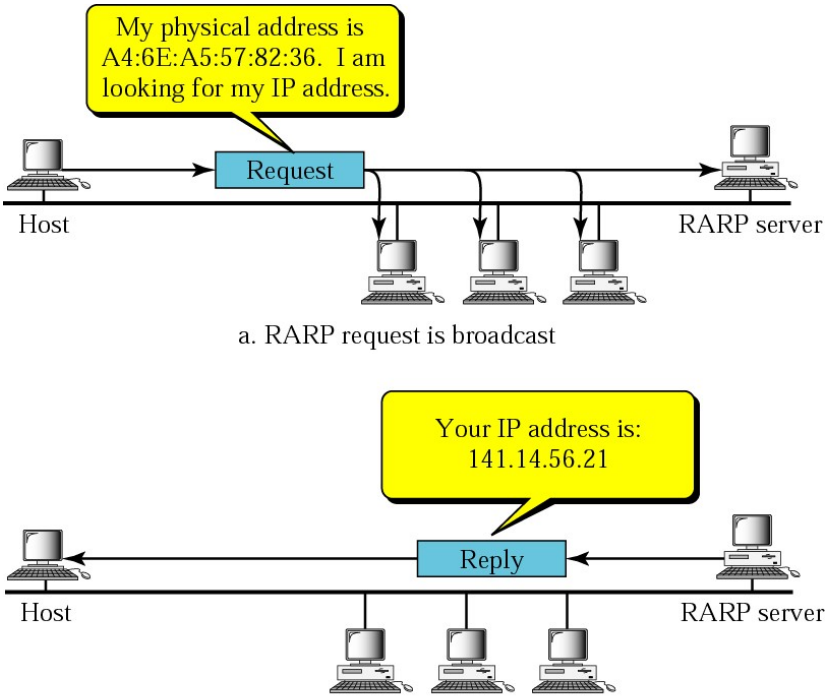


Figure 21.10 RARP Operation

4.1.2. RARP Packet Format & Encapsulation

- The format of the RARP packet is the same as the ARP packet format as displayed in Figure 21.4, except that the Operation field. It's value is 3 for RARP request message and 4 for RARP reply message.
- An RARP packet is also encapsulated directly into a data link frame just like ARP packet as displayed in Figure 21.5.

4.1.3. Limitations of RARP:

- As broadcasting is done at the data link layer. The physical broadcast address, all 1's in the case of Ethernet, does not pass the boundaries of a network.
- This means that if an administrator has several networks or several subnets, it needs to assign a RARP server for each network or subnet.
- This is the reason that **RARP is almost obsolete**.
- Two protocols, BOOTP and DHCP, are replacing RARP.

4.2. BOOTP

The Bootstrap Protocol (BOOTP) is a client/server based protocol at application layer, designed to **provide physical address** to **logical address mapping**. The administrator may put the client and the server on the same network or on different networks, as shown in Figure 21.11a and Figure 21.11b respectively. **BOOTP** messages are **encapsulated** in a **UDP packet**, and the UDP packet itself is encapsulated in an **IP packet**, as shown in Figure 21.12.

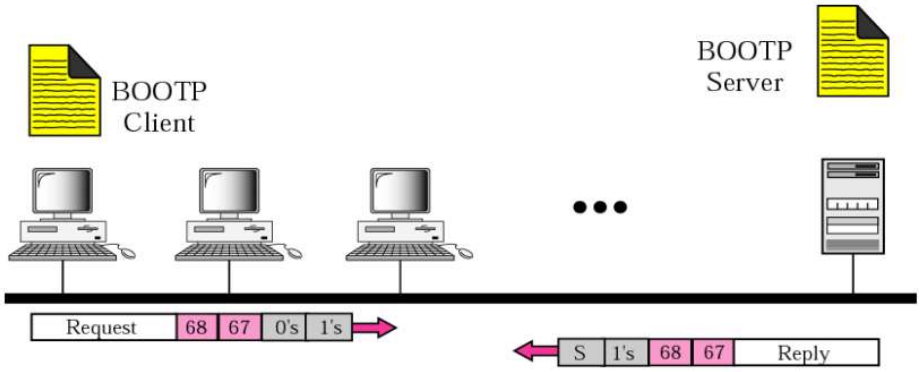


Figure 21.11a BOOTP client and server on the same network

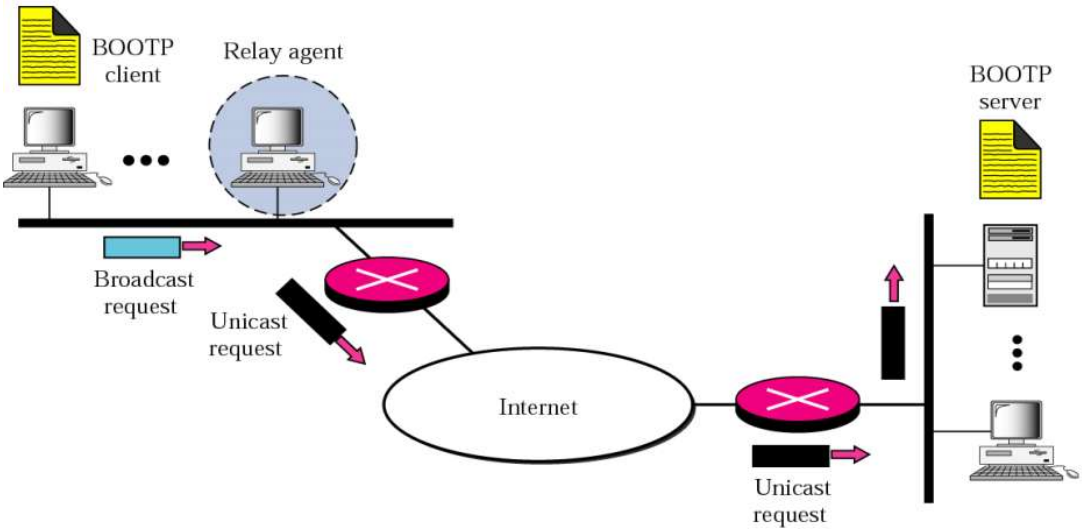


Figure 21.11b BOOTP client and server on the same and different network

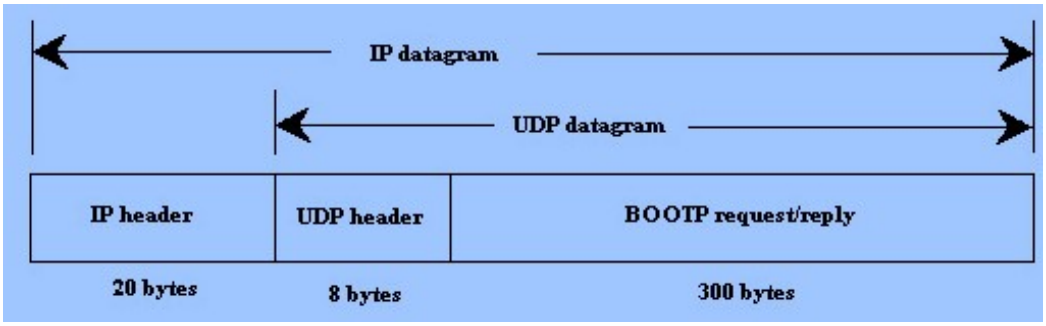


Figure 21.12 BOOTP data Encapsulation

4.2.1. BOOTP Operation

There are two cases of BOOTP operation described below:

Case 1: Client and server on same network (*Figure 21.11a)*

1. When a BOOTP client is started, it has no IP address, so it broadcasts a message containing its MAC address onto the network. This message is called a "BOOTP request," and it is picked up by the BOOTP server, which replies to the client with the following information that the client needs:
    a. The client's IP address, subnet mask, and default gateway address.
    b. The IP address and host name of the BOOTP server.
    c. The IP address of the server that has the boot image, which the client needs to load its operating system.
2. When the client receives this information from the BOOTP server, it configures and initializes its TCP/IP protocol stack, and then connects to the server on which the boot image is shared.

Case 2 : Client and server on different networks(Figure 21.11b**)**

1. If the server exists on some distant network the BOOTP request is broadcast because the client does not know the IP address of the server.
2. The client simply uses all as 0's the source address and all 1's as the destination address.
3. But a broadcast IP datagram cannot pass through any router. To solve the problem, there is a need for an intermediary.
4. One of the hosts in local network (or a router that can be configured to operate at the application layer) can be used as a relay . The host in this case is called a **relay agent**.
5. The relay agent knows the unicast address of a BOOTP server. When it receives this type of packet, it encapsulates the message in a unicast datagram and sends the request to the BOOTP server.
6. The packet, carrying a unicast destination address, is routed by any router and reaches the BOOTP server.
7. The BOOTP server knows the message comes from a relay agent because one of the fields in the request message defines the IP address of the relay agent.
8. BOOTP server sends a BOOTP reply message to the relay agent.
9. The relay agent, after receiving the reply, sends it to the BOOTP client.

4.2.2. Limitations of BOOTP

- BOOTP is **not a dynamic configuration** protocol.
- When a client requests its IP address, the BOOTP server consults a table that matches the physical address of the client with its IP address.
- This implies that the binding between the physical address and the IP address of the client already exists. The binding is predetermined i.e. static.
- However, what if a host moves from one physical network to another? What if a host wants a temporary IP address?
- BOOTP cannot handle these situations because the binding between the physical and IP addresses is static and fixed in a table until changed by the administrator.
- BOOTP is a **static configuration** protocol.

4.3. DHCP

The **Dynamic Host Configuration Protocol** (DHCP) has been devised to provide static and dynamic address allocation that can be manual or automatic as required.

- **Static Address Allocation** In this capacity DHCP acts as BOOTP does. It is backward compatible with BOOTP, which means a host running the BOOTP client can request a static address from a DHCP server. A DHCP server has a database that statically binds physical addresses to IP addresses.
- **Dynamic Address Allocation** DHCP has a second database with a pool of available IP addresses. This second database makes DHCP dynamic. When a DHCP client requests a temporary IP address, the DHCP server goes to the pool of available (unused) IP addresses and assigns an IP address for a negotiable period of time.

- When a DHCP client sends a DHCP request to a DHCP server, the server first checks its static database. If an entry with the requested physical address exists in the static database, the permanent IP address of the client is returned.
- On the other hand, if the entry does not exist in the static database, the server selects an IP address from the available pool, assigns the address to the client, and adds the entry to the dynamic database.
- The dynamic aspect of DHCP is needed when a host moves from network to network or is connected and disconnected from a network (as is a subscriber to a service provider).
- DHCP provides temporary IP addresses for a limited time. The addresses assigned from the pool are temporary addresses.
- The DHCP server issues a lease for a specific time. When the lease expires, the client must either stop using the IP address or renew the lease.
- The server has the option to agree or disagree with the renewal. If the server disagrees, the client stops using the address.

4.3.1. DHCP Operation

DHCP provides an automated way to distribute and update IP addresses and other configuration information on a network. A DHCP server provides this information to a DHCP client through the exchange of a series of messages, known as the DHCP conversation or the DHCP transaction displayed in Figure 21.13.

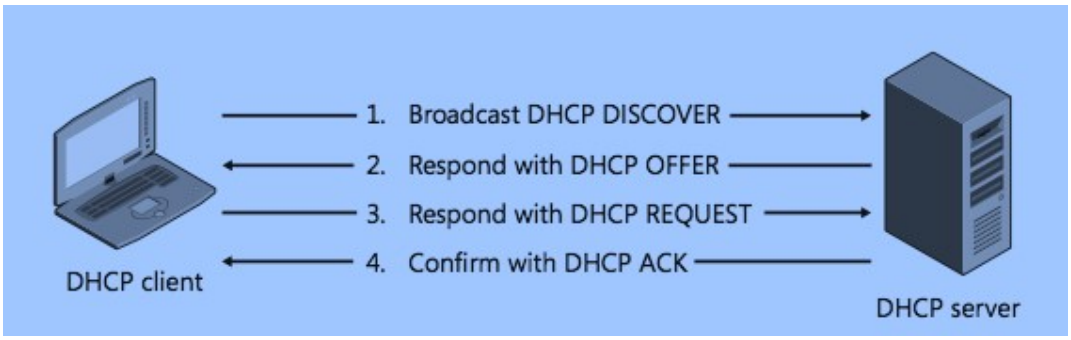


Figure 21.13: DHCP Operation

DHCP client goes through the **four step** process:

1. A DHCP client sends a broadcast packet (**DHCP Discover**) to discover DHCP servers on the LAN segment.

2. The DHCP servers receive the **DHCP Discover** packet and respond with **DHCP Offer** packets, offering IP addressing information.

3. If the client receives the **DHCP Offer** packets from multiple DHCP servers, the first **DHCP Offer** packet is accepted. The client responds by broadcasting a **DHCP Request** packet, requesting network parameters from a single server.

4. The DHCP server approves the lease with a **DHCP Acknowledgement (DHCP Ack)** packet. The packet includes the lease duration and other configuration information.

5. ICMP

IP provides unreliable and connectionless datagram delivery. It was designed this way to make efficient use of network resources. The IP protocol is a best-effort delivery service that delivers a datagram from its original source to its final destination. However, **IP protocol has two deficiencies:** lack of error control and lack of assistance mechanisms.

- The IP protocol has no error-reporting or error-correcting mechanism.
- What happens if something goes wrong?
- What happens if a router must discard a datagram because it cannot find a router to the final destination, or because the time-to-live field has a zero value?
- What happens if the final destination host must discard all fragments of a datagram because it has not received all fragments within a predetermined time limit?

These are examples of situations where an error has occurred and the IP protocol has no built-in mechanism to notify the original host.

- The IP protocol also lacks a mechanism for host and management queries.
- A host sometimes needs to determine if a router or another host is alive.
- And sometimes a network administrator needs information from another host or router.

The Internet Control Message Protocol (ICMP) has been designed to compensate for the above two deficiencies. It is a companion to the IP protocol.

5.1. Types of Messages

ICMP messages are divided into two broad categories: **Error-reporting messages and Query messages**

- The **error-reporting messages** report problems that a router or a host (destination) may encounter when it processes an IP packet.
- The **query messages**, which occur in pairs, help a host or a network manager get specific information from a router or another host.

5.2. Message Format

An ICMP message has an **8-byte header** and a **variable-size data section**. Although the general format of the header is different for each message type, the first 4 bytes are common to all. As Figure 21.14 shows:

- The first field, **ICMP type**, defines the type of the message.
- The **code field** specifies the reason for the particular message type.
- The last common field is the **checksum field** used for securing ICMP header.
- The rest of the header is specific for each message type.
- The **data section** in error messages carries information for finding the original packet that had the error.
- In ICMP query messages, the data section carries extra information based on the type of the query.

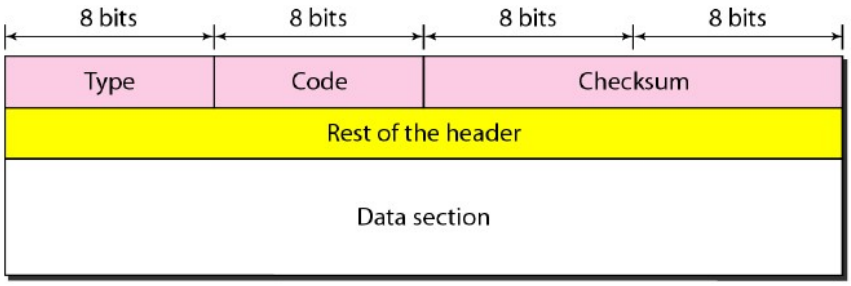


Figure 21.14 General format of ICMP messages

5.3. ICMP Encapsulation:

ICMP itself is a network layer protocol. However it's messages are not passed directly to data link layer. Instead the messages are first encapsulated inside IP datagrams before going to the lower layer (see Figure 21.15).

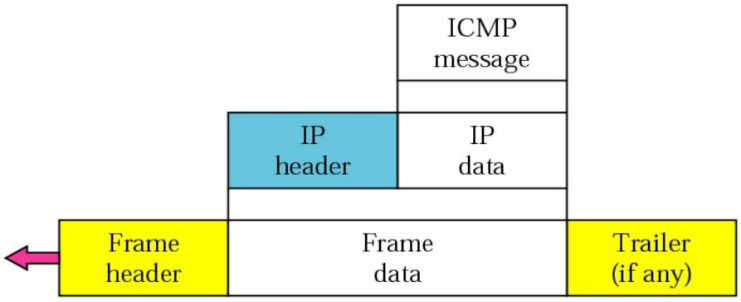


Figure 21.15 Contents of data field for the error messages

5.4. Error Reporting Messages

One of the main responsibilities of ICMP is to report errors. Although technology has produced increasingly reliable transmission media, errors still exist and must be handled. IP is an unreliable protocol. This means that error checking and error control are not a concern of IP. ICMP was designed, in part, to compensate for this shortcoming. However, ICMP does not correct errors-it simply reports them. Error correction is left to the higher-level protocols.

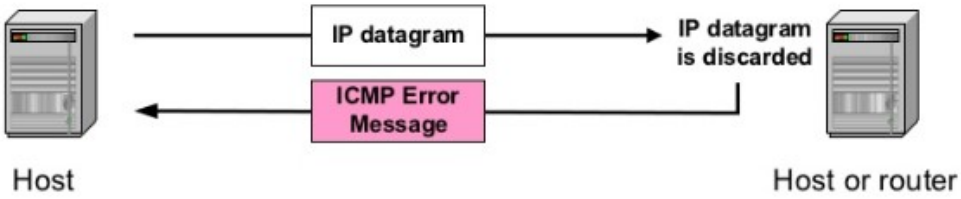


Figure 21.16 ICMP Error Reporting Message

- Error messages are typically sent when a datagram is discarded due to some error as displayed in Figure 12.16.
- Error messages are always sent to the original source because the only information available in the datagram about the route is the source and destination IP addresses.
- **Five types of errors** are handled: *destination unreachable, source quench, time exceeded, parameter problems, and redirection* (see Figure 21.17).

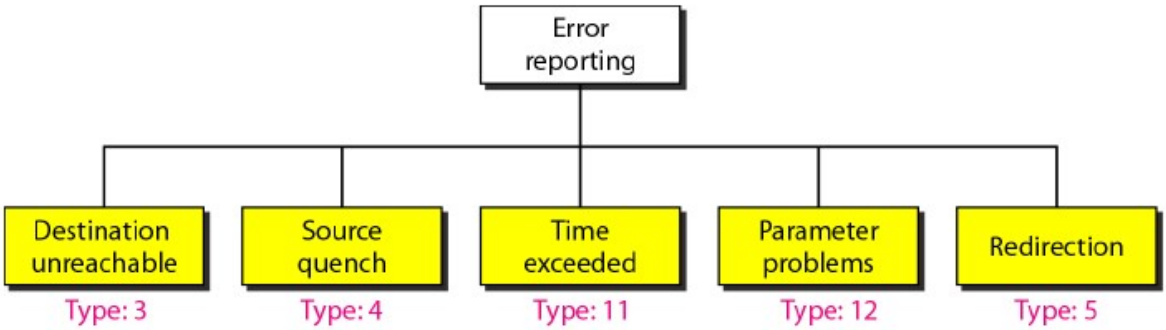


Figure 21.17 Error-reporting messages

a. Destination Unreachable

When a router cannot route a datagram or a host cannot deliver a datagram, the datagram is discarded and the router or the host sends a destination-unreachable message back to the source host that initiated the datagram. Note that destination-unreachable messages can be created by either a router or the destination host.

b. Source Quench

The IP protocol is a connectionless protocol. IP does not have a flow control mechanism embedded in the protocol. The lack of flow control can create a major problem in the operation of IP: *congestion*. The source host never knows if the routers or the destination host has been overwhelmed with datagrams. The source host never knows if it is producing datagrams faster than can be forwarded by routers or processed by the destination host.

- The lack of flow control can create congestion in routers or the destination host. In this case, the router or the host has no choice but to discard some of the datagrams.
- **The source-quench message in ICMP was designed to add a kind of flow control to the IP.**

- When a router or host discards a datagram due to congestion**, it sends a source-quench message to the sender of the datagram**. This message has **two purposes**.
- **First**, it informs the source that the datagram has been discarded.
- **Second**, it warns the source that there is congestion somewhere in the path and that the source should slow down (quench) the sending process.

c. Time Exceeded

The time-exceeded message is generated in **two cases**:

Case1: As routers use routing tables to find the next hop (next router) that must receive the packet. If there are errors in one or more routing tables, a packet can travel in a loop or a cycle, going from one router to the next or visiting a series of routers endlessly. Each datagram contains a field called *time to live* that controls this situation. When a datagram visits a router, the value of this field is decremented by 1. When the time-to-live value reaches 0, after decrementing, the router discards the datagram. However, when the datagram is discarded, a time-exceeded message must be sent by the router to the original source.

Case2: A time-exceeded message is also generated when not all fragments that make up a message arrive at the destination host within a certain time limit.

d. Parameter Problem

Any **ambiguity in the header** part of a datagram can create serious problems as the datagram travels through the Internet. If a router or the destination host discovers an ambiguous or missing value in any field of the datagram, it discards the datagram and sends a parameter-problem message back to the source.

e. Redirection

When a router needs to send a packet destined for another network, it must know the IP address of the next appropriate router. The same is true if the sender is a host. Both routers and hosts, then, must have a routing table to find the address of the router or the next router. Routers take part in the routing update process, and are supposed to be updated constantly. Routing is dynamic. However, for efficiency, hosts do not take part in the routing update process because there are many more hosts in an internet than routers.

- Updating the routing tables of hosts dynamically produces unacceptable traffic.
- The hosts usually use static routing. When a host comes up, its routing table has a limited number of entries. It usually knows the IP address of only one router, the default router.
- For this reason, the host may send a datagram, which is destined for another network, to the wrong router.
- In this case, the router that receives the datagram will forward the datagram to the correct router.
- However, to update the routing table of the host, it sends a redirection message to the host.
- This concept of redirection is shown in Figure 21.18. Host A wants to send a datagram to host B.

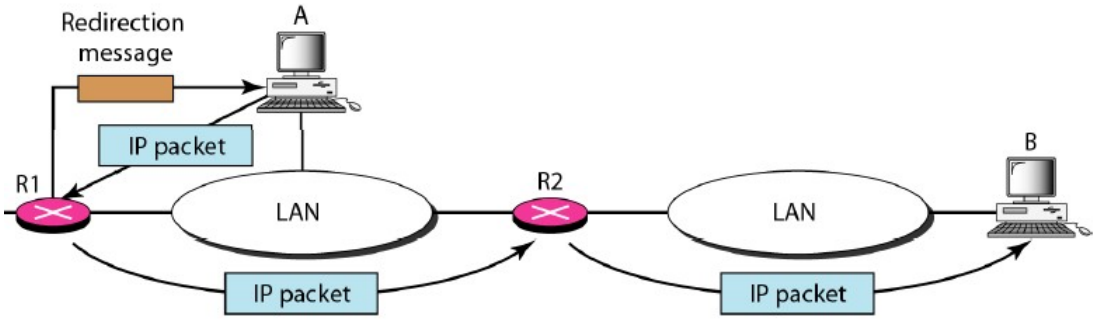


Figure 21.18 Redirection concept

- Router R2 is obviously the most efficient routing choice, but host A did not choose router R2. The datagram goes to R1 instead.
- Router R1, after consulting its table, finds that the packet should have gone to R2.
- It sends the packet to R2 and, at the same time, sends a redirection message to host A.
- Host A's routing table can now be updated.

5.5. ICMP Query Messages

In addition to error reporting, ICMP can diagnose some network problems. This is accomplished through the query messages, a group of **four different pairs of messages**, as shown in Figure 21.19.

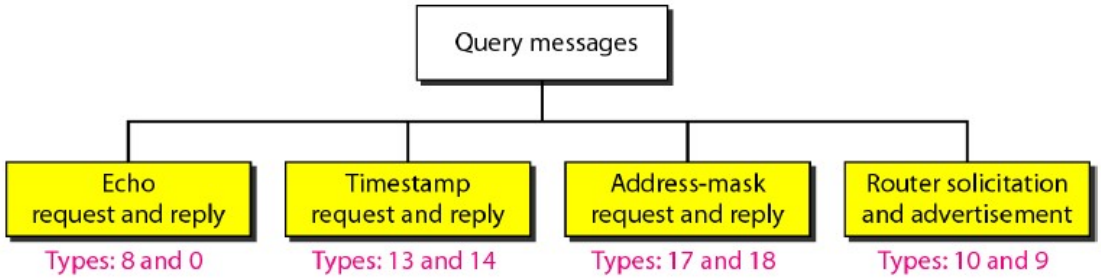


Figure 21.19 Query messages

- In this type of ICMP message, a node sends a ICMP request message that is answered in a specific format as ICMP reply by the destination node, depicted in Figure 21.20.

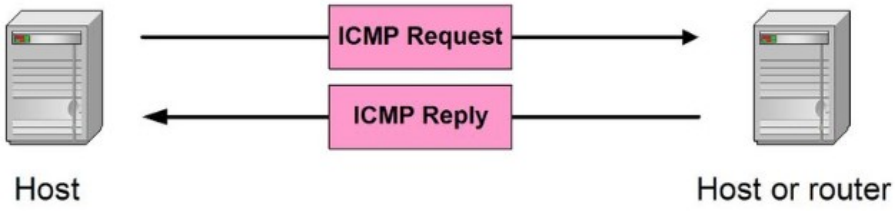


Figure 21.20 ICMP Query Message

- A query message is encapsulated in an IP packet, which in turn is encapsulated in a data link layer frame.
- However, in this case, no bytes of the original IP are included in the message, as shown in Figure 21.21.

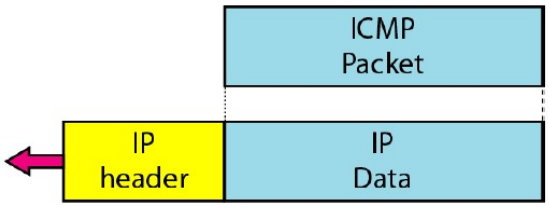


Figure 21.21 Encapsulation of ICMP query messages

a. Echo Request and Echo Reply

The echo-request and echo-reply messages are designed for diagnostic purposes. The combination of echo-request and echo-reply messages determines whether two systems (hosts or routers) can communicate with each other Figure 21.22. It also confirms that the intermediate routers are receiving, processing, and forwarding IP datagrams.

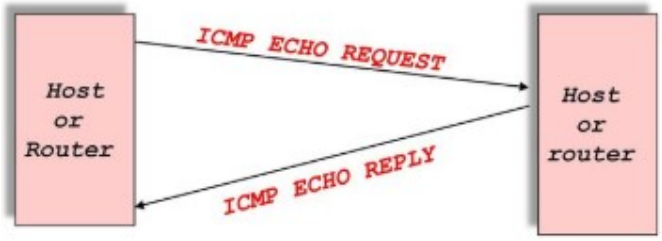


Figure 21.22 ICMP Echo Request and Echo Reply

Today, most systems provide a version of the ***ping* command** that can create a series (instead of just one) of echo-request and echo-reply messages, providing statistical information. We can use the *ping* program to find if a host is alive and responding.

b. Timestamp Request and Timestamp Reply

Two machines (hosts or routers) can use the timestamp request and timestamp reply messages to determine the round-trip time needed for an IP datagram to travel between them. It can also be used to synchronize the clocks in two machines.

c. Address-Mask Request and Address-Mask Reply

A host may know its IP address, but it may not know the corresponding mask. For example, a host may know its IP address as 159.31.17.24, but it may not know that the corresponding mask is /24. To obtain its mask, a host sends an address-mask-request message to a router on the LAN. If the host knows the address of the router, it sends the request directly to the router. If it does not know, it broadcasts the message. The router receiving the address-mask-request message responds with an address-mask-reply message, providing the necessary mask for the host. This can be applied to its full IP address to get its subnet address.

d. Router Solicitation and Router Advertisement

The router-solicitation and router-advertisement messages can help a host to check whether the neighboring routers are alive and functioning. A host can broadcast (or multicast) a router-solicitation message. The router or routers that receive the solicitation message broadcast their routing information using the router-advertisement message. A router can also periodically send router-advertisement messages even if no host has solicited. Note that when a router sends out an advertisement, it announces not only its own presence but also the presence of all routers on the network of which it is aware.

Reference:

1. B. A. Forouzan: Data Communications and Networking, Fourth edition, TMH .