UNIT II:
Network Layer (Layer 3):

Network layer works for the transmission of data from one host to the other located in different networks. It also takes care of packet routing i.e. selection of the shortest path to transmit the packet, from the number of routes available. The sender & receiver's IP address are placed in the header by the network layer.

The functions of the Network layer are:

1. **Routing:** The network layer protocols determine which route is suitable from source to destination. This function of network layer is known as routing.
2. **Logical Addressing:** In order to identify each device on internetwork uniquely, network layer defines an addressing scheme. The sender & receiver's IP address are placed in the header by network layer. Such an address distinguishes each device uniquely and universally.

   * Segment in Network layer is referred as **Packet**.

** Network layer is implemented by networking devices such as routers.

Introduction and IPv4 Datagram Header

The network layer is the third layer (from bottom) in the OSI Model. The network layer is concerned with the delivery of a packet across multiple networks. The network layer is considered the backbone of the OSI Model. It selects and manages the best logical path for data transfer between nodes. This layer contains hardware devices such as routers, bridges, firewalls, and switches, but it actually creates a logical image of the most efficient communication route and implements it with a physical medium. Network layer protocols exist in every host or router. The router examines the header fields of all the IP packets that pass through it. Internet Protocol and Netware IPX/SPX are the most common protocols associated with the network layer.

In the OSI model, the network layer responds to requests from the layer above it (transport layer) and issues requests to the layer below it (data link layer).

Responsibilities of Network Layer:

Packet forwarding/Routing of packets: Relaying of data packets from one network segment to another by nodes in a computer network

Connectionless communication(IP): A data transmission method used in packet-switched networks in which each data unit is separately addressed and routed based on information carried by it

Fragmentation of data packets: Splitting of data packets that are too large to be transmitted on the network

There are two types of network transmission techniques, circuit switched network and packet switched network.

Circuit Switch vs Packet Switch:

In circuit switched network, a single path is designated for transmission of all the data packets. Whereas in case of a packet-switched network, each packet may be sent through a different path to reach the destination.

In a circuit switched network, the data packets are received in order whereas in a packet switched network, the data packets may be received out of order.

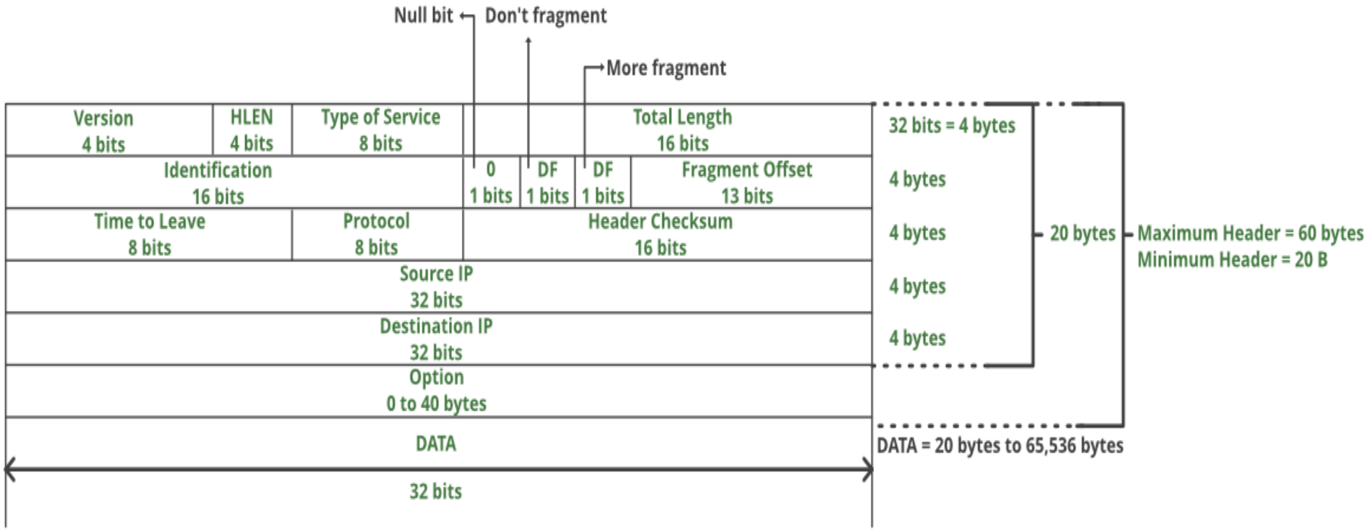
The packet switching is further subdivided into Virtual circuits and Datagram.

IPv4:

- IPv4 is a connectionless protocol used for packet-switched networks.
- The IPv4 addresses are unique and universal.
- IPv4 uses 32-bit (4 byte) addressing, which gives 2^{32} addresses. IPv4 addresses are written in the dot-decimal notation, which comprises of four octets of the address expressed individually in decimal and separated by periods, for instance, 192.168.1.5.

IPv4 Datagram Header

Size of the header is 20 to 60 bytes.



VERSION: Version of the IP protocol (4 bits), which is 4 for IPv4

HLEN: IP header length (4 bits), which is the number of 32 bit words in the header. The minimum value for this field is 5 and the maximum is 15.

Type of service: Low Delay, High Throughput, Reliability (8 bits)

Total Length: Length of header + Data (16 bits), which has a minimum value 20 bytes and the maximum is 65,535 bytes.

Identification: Unique Packet Id for identifying the group of fragments of a single IP datagram (16 bits)

Flags: 3 flags of 1 bit each : reserved bit (must be zero), do not fragment flag, more fragments flag (same order)

Fragment Offset: Represents the number of Data Bytes ahead of the particular fragment in the particular Datagram. Specified in terms of number of 8 bytes, which has the maximum value of 65,528 bytes.

Time to live: Datagram’s lifetime (8 bits), It prevents the datagram to loop through the network by restricting the number of Hops taken by a Packet before delivering to the Destination.

Protocol: Name of the protocol to which the data is to be passed (8 bits)

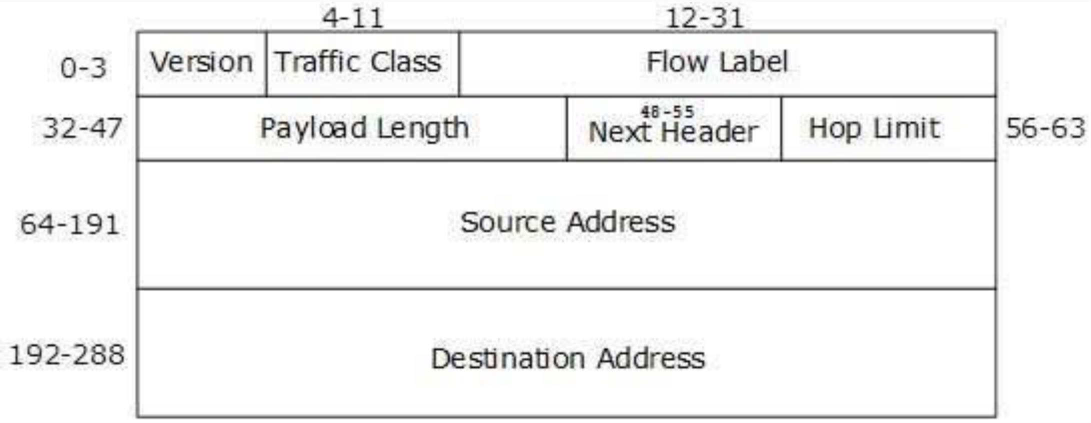
Header Checksum: 16 bits header checksum for checking errors in the datagram header

Source IP address: 32 bits IP address of the sender

Destination IP address: 32 bits IP address of the receiver

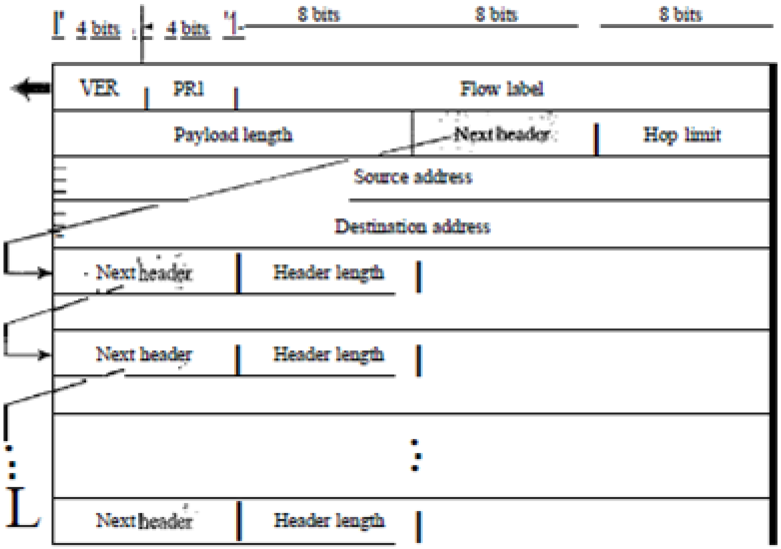
Option: Optional information such as source route, record route. Used by the Network administrator to check whether a path is working or not.

IPv6: Header



[Image: IPv6 Fixed Header]

Format ofan IPv6 datagram



1	**Version** (4-bits): It represents the version of Internet Protocol, i.e. 0110.
2	**Traffic Class** (8-bits): These 8 bits are divided into two parts. The most significant 6 bits are used for Type of Service to let the Router Known what services should be provided to this packet. The least significant 2 bits are used for Explicit Congestion Notification (ECN).
3	**Flow Label** (20-bits): This label is used to maintain the sequential flow of the packets belonging to a communication. The source labels the sequence to help the router identify that a particular packet belongs to a specific flow of information. This field helps avoid re-ordering of data packets. It is designed for streaming/real-time media.
4	**Payload Length** (16-bits): This field is used to tell the routers how much information a particular packet contains in its payload. Payload is composed of Extension Headers and Upper Layer data. With 16 bits, up to 65535 bytes can be indicated; but if the Extension Headers contain Hop-by-Hop Extension Header, then the payload may exceed 65535 bytes and this field is set to 0.
5	**Next Header** (8-bits): This field is used to indicate either the type of Extension Header, or if the Extension Header is not present then it indicates the Upper Layer PDU. The values for the type of Upper Layer PDU are same as IPv4's.
6	**Hop Limit** (8-bits): This field is used to stop packet to loop in the network infinitely. This is same as TTL in IPv4. The value of Hop Limit field is decremented by 1 as it passes a link (router/hop). When the field reaches 0 the packet is discarded.
7	**Source Address** (128-bits): This field indicates the address of originator of the packet.
8	**Destination Address** (128-bits): This field provides the address of intended recipient of the packet.

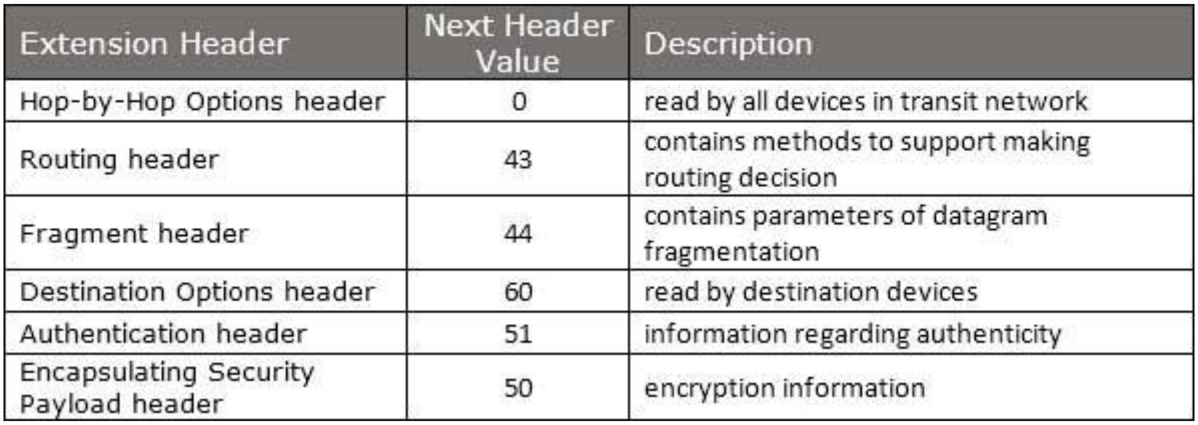
Extension Headers

In IPv6, the Fixed Header contains only that much information which is necessary, avoiding those information which is either not required or is rarely used. All such information is put between the Fixed Header and the Upper layer header in the form of Extension Headers. Each Extension Header is identified by a distinct value.

When Extension Headers are used, IPv6 Fixed Header's Next Header field points to the first Extension Header. If there is one more Extension Header, then the first Extension Header's 'Next-Header' field points to the second one, and so on. The last Extension Header's 'Next-Header' field points to the Upper Layer Header. Thus, all the headers points to the next one in a linked list manner.

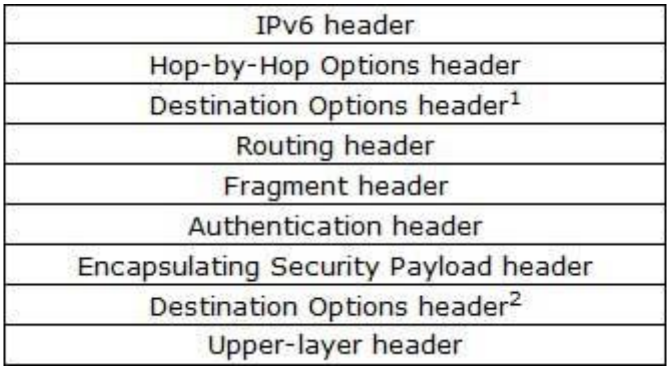
If the Next Header field contains the value 59, it indicates that there are no headers after this header, not even Upper Layer Header.

The following Extension Headers must be supported as per RFC 2460:



Extension Header	Next Header Value	Description
Hop-by-Hop Options header	0	read by all devices in transit network
Routing header	43	contains methods to support making routing decision
Fragment header	44	contains parameters of datagram fragmentation
Destination Options header	60	read by destination devices
Authentication header	51	information regarding authenticity
Encapsulating Security Payload header	50	encryption information

The sequence of Extension Headers should be:

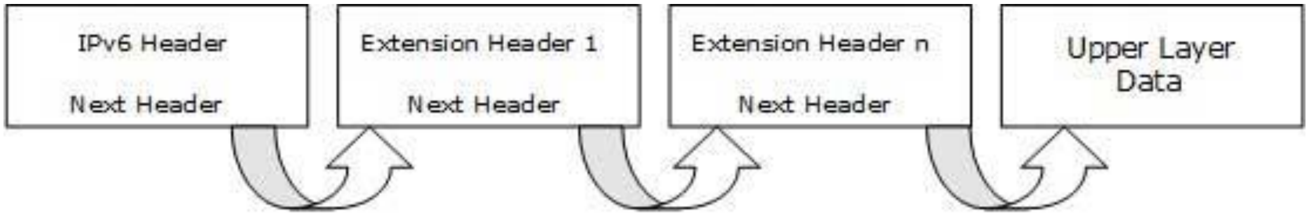


IPv6 header
Hop-by-Hop Options header
Destination Options header[1]
Routing header
Fragment header
Authentication header
Encapsulating Security Payload header
Destination Options header[2]
Upper-layer header

These headers:

- 1. should be processed by First and subsequent destinations.
- 2. should be processed by Final Destination.

Extension Headers are arranged one after another in a linked list manner, as depicted in the following diagram:

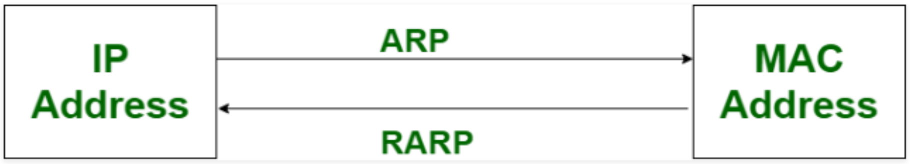


Address Mapping

- The delivery of a packet to a host or router requires two levels of addressing
  1. Logical and
  2. Physical
- We need to map logical address to its corresponding physical address and vice versa.
- This can be done by static mapping and dynamic mapping
- Static mapping involves in the creation a table that associates a logical address with a physical address. This address mapping table is stored in each device on the network.
- Static mapping has some limitations because physical addresses may change in the following ways
  1. A machine could change its Network Interface Card (NIC), which results in a new Media Access Control(MAC) address
  2. In some LANs such as Local Talk, the MAC address changes every time the computer is turned on.
  3. A mobile device can move from one physical network to another, which results in a change in its MAC address
- To implement these changes, a static mapping table must be updated periodically. This overhead could affect network performance.
- In dynamic mapping each time a device knows one of the two addresses (IP address or MAC address)

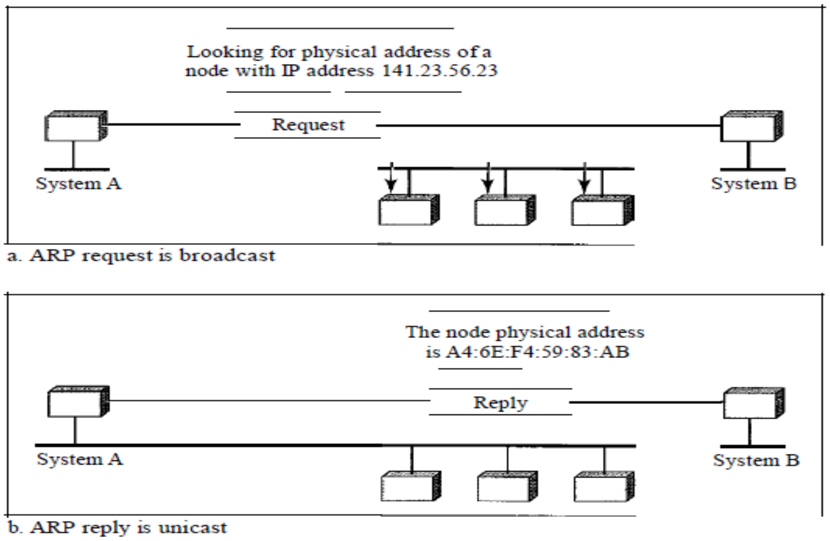
Mapping Logical to Physical Address:ARP(Address Resolution Protocol)

- It is an Address Resolution Protocol.
- It is a network layer protocol used to address resolution or address mapping in TCP/IP protocol suite
- The purpose of ARP is to find out MAC address of a device in LAN for corresponding IP address, which network application is trying to communicate.
- ARP is never generated for the device of other network.



- ARP associates an IP address with its physical address on a typical physical network, such as LAN, each device on a link is identified by physical address that is usually imprinted on the NIC.

Figure *ARP operation*


Looking for physical address of a node with IP address 141.23.56.23
Request
System A
System B

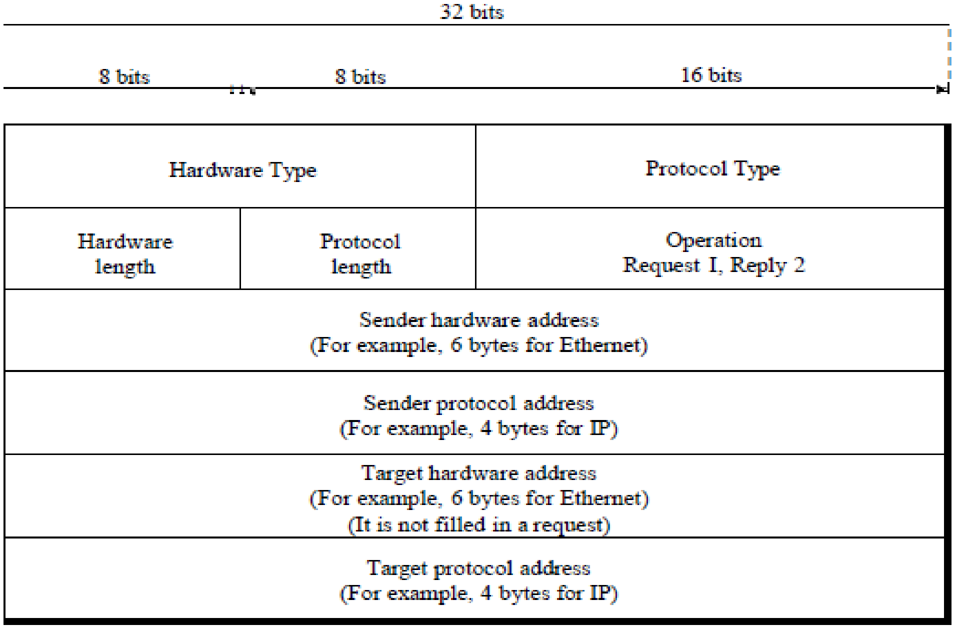
a. ARP request is broadcast

The node physical address is A4:6E:F4:59:83:AB
Reply
System A
System B

b. ARP reply is unicast

ARP Packet format

Figure *ARP packet*



32 bits

8 bits	8 bits	16 bits
Hardware Type		Protocol Type
Hardware length	Protocol length	Operation Request I, Reply 2
Sender hardware address (For example, 6 bytes for Ethernet)		
Sender protocol address (For example, 4 bytes for IP)		
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled in a request)		
Target protocol address (For example, 4 bytes for IP)		

- **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.
- **Protocol type**. This is a 16-bit field defining the protocol. For example, the value of this field for the IPv4 protocol is 0800_{16}, ARP can be used with any higher-level protocol.
- **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.

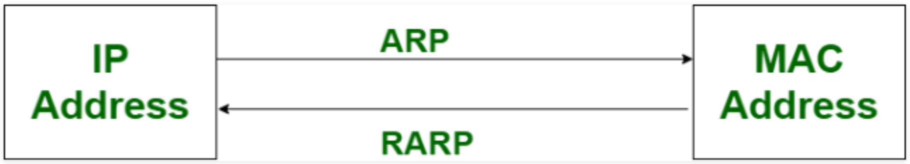
- **Protocol length.** This is an 8-bit field defining the length of the logical address in bytes. For example, for the IPv4 protocol the value is 4.
- **Operation.** This is a 16-bit field defining the type of packet. Two packet types are defined: ARP request (1) andARP reply (2).
- **Sender hardware address.** This is a variable-length field defining the physical address of the sender. For example, for Ethernet this field is 6 bytes long.
- **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.
- **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 allOs because the sender does not know the physical address of the target.
- **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.

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

RARP: (Reverse Address Resolution Protocol)

It finds the logical address for a machine that knows only its physical address

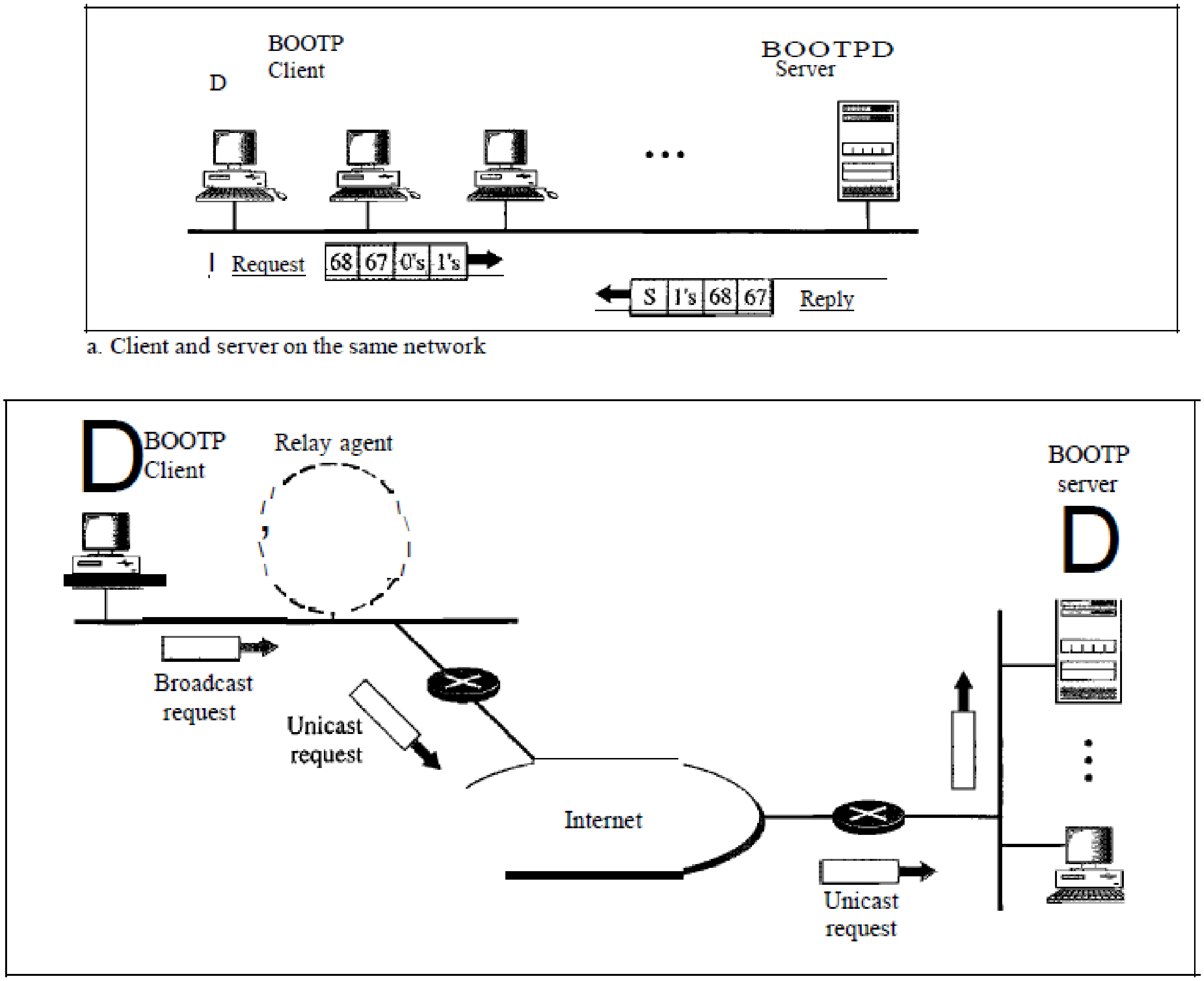
In ARP receivers MAC address is fetched through ARP (32-bit). IP address mapped into (48-bit) MAC address whereas, In RARP,IP address is fetched through server. Through RARP (48-bit) MAC address of 48 bits mapped into (32-bit) IP address



BOOTP: The Bootstrap Protocol

The **BOOTP** protocol is a client/server protocol designed to provide physical address to logical address mapping BOOTP is an application layer protocol. The administrator may put the client & the server on the same network or on different networks. BOOTP messages are encapsulated in a UDP packet and the UDP packet itself is encapsulated in an IP packet.

Figure *BOOTP client and server on the same and different network*



a. Client and server on the same network

b. Client and server on different networks

-

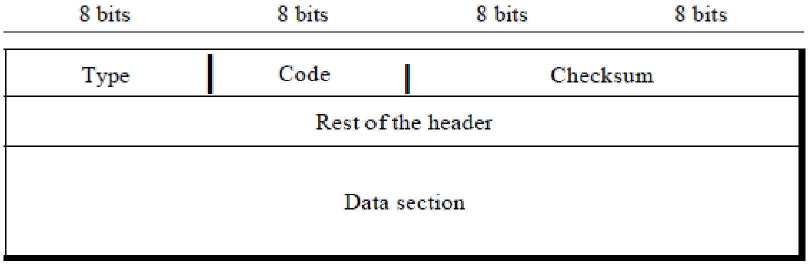
- Client determines its own hardware address.
- Client sends its hardware address in UDP datagram to server.
- After receiving datagram server looks up hardware address of client
  - ➢ if client knows its own IP address
  - ➢ if client does not know its own IP address
  - ➢ use of ARP on server
- On replay BOOTP client record its own IP address & begin bootstrap process.

ICMP: Internet Control Message Protocol

- **ICMP** is designed to overcome the following two problems in the IP protocol
    - IP protocol has no error-reporting or error-correcting mechanism
    - IP protocol also lacks a mechanism for host and management queries.

  To overcome the above problem ICMP has been designed. ICMP is a companion to the IP protocol
- This is a Network Layer protocol
- Types of Messages: There are two types if messages
    - **Error-reporting messages:** Error Reporting messages report or a host may encounter while processing of IP Packet.
    - **Query messages:** It helps to get specific information from a router or another host.

Message Format



Type(8 bits): Defines the type of the message.

Code(8 bits): The code field specifies the reason for the particular message type.

Checksum(16 bits): The last common field is the checksum field which checks the error.

Rest of the header: The rest of the header is specific for each message type.

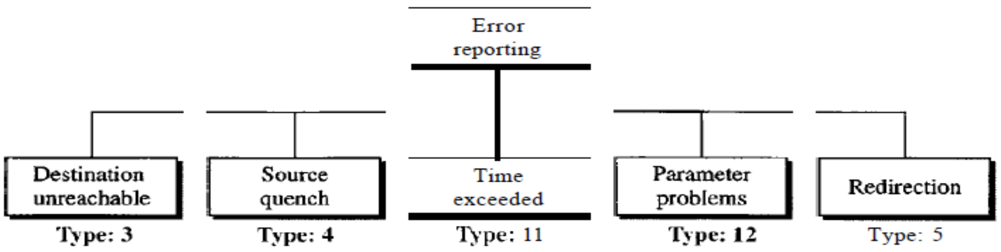
Data Section: The data section in error messages carries information for finding the original packet that had the error. In query messages, the data section carries extra information based on the type of the query.

Error Reporting:

ICMP always reports error messages to the original source. ICMP does not correct the errors.

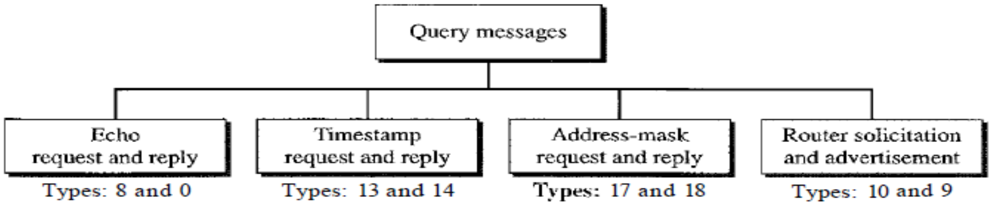
Types of Error-reporting messages

Error-reporting messages



Query Messages: Types of Query Messages

Figure *Query messages*



DELIVERY

The delivery of a packet to its final destination is accomplished by using two different methods of delivery,

- direct and
- indirect

Direct Delivery

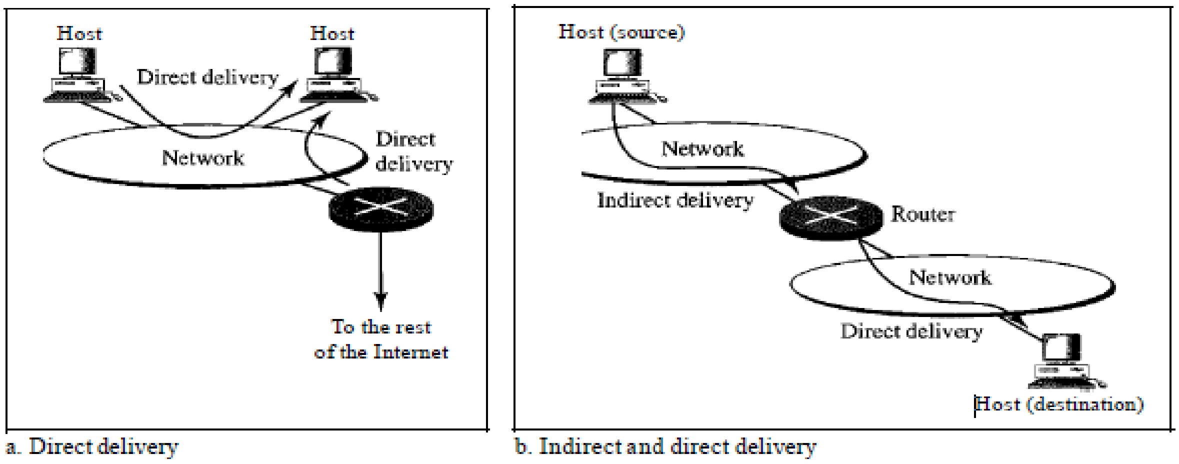
In a direct delivery, the final destination of the packet is a host connected to the same physical network as the deliverer. Direct delivery occurs when the source and destination of the packet are located on the same physical network or when the delivery is between the last router and the destination host.

The sender can easily determine if the delivery is direct. It can extract the network address of the destination (using the mask) and compare this address with the addresses of the networks to which it is connected. If a match is found, the delivery is direct.

Indirect Delivery

If the destination host is not on the same network as the deliverer, the packet is delivered indirectly. In an indirect delivery, the packet goes from router to router until it reaches the one connected to the same physical network as its final destination. Note that a delivery always involves one direct delivery but zero or more indirect deliveries. Note also that the last delivery is always a direct delivery.

Figure *Direct and indirect delivery*



a. Direct delivery b. Indirect and direct delivery

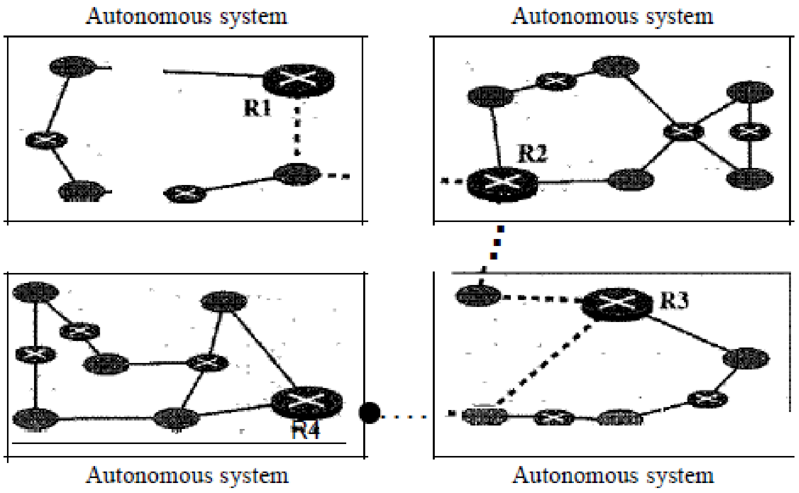
UNICAST ROUTING PROTOCOLS

A routing table can be either static or dynamic. A *static table* is one with manual entries.
A *dynamic table* is one that is updated automatically when there is a change somewhere in the internet

Intra- and Interdomain Routing

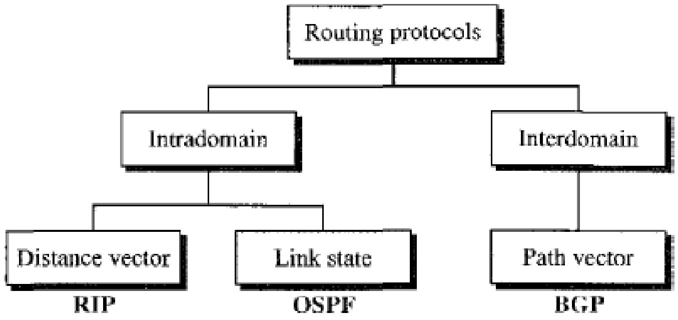
Today, an internet can be so large that one routing protocol cannot handle the task of updating the routing tables of all routers. For this reason, an internet is divided into autonomous systems. An autonomous system (AS) is a group of networks and routers under the authority of a single administration. Routing inside an autonomous system is referred to as intradomain routing. Routing between autonomous systems is referred to as interdomain routing. Each autonomous system can choose one or more intradomain routing protocols to handle routing inside the autonomous system. However, only one interdomain routing protocol handles routing between autonomous systems as shown in below figure.

Figure *Autonomous systems*



Routing Information Protocol (RIP) is an implementation of the distance vector protocol. Open Shortest Path First (OSPF) is an implementation of the link state protocol. Border Gateway Protocol (BGP) is an implementation of the path vector protocol.

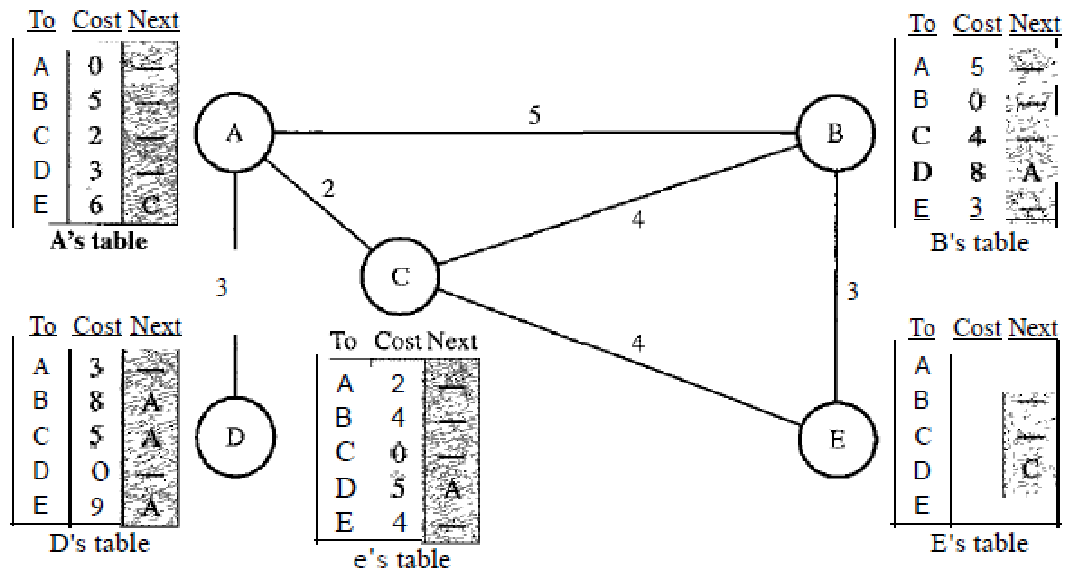
Figure *Popular routing protocols*



Distance Vector Routing

In distance vector routing, the least-cost route between any two nodes is the route with minimum distance. In this protocol, as the name implies, each node maintains a vector (table) of minimum distances to every node. The table at each node also guides the packets to the desired node by showing the next stop in the route (next-hop routing).

Figure *Distance vector routing tables*



To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

A's table

To	Cost	Next
A	5	—
B	0	—
C	4	—
D	8	A
E	3	—

B's table

To	Cost	Next
A	3	—
B	8	A
C	5	A
D	0	—
E	9	A

D's table

To	Cost	Next
A	2	—
B	4	—
C	0	—
D	5	A
E	4	—

e's table

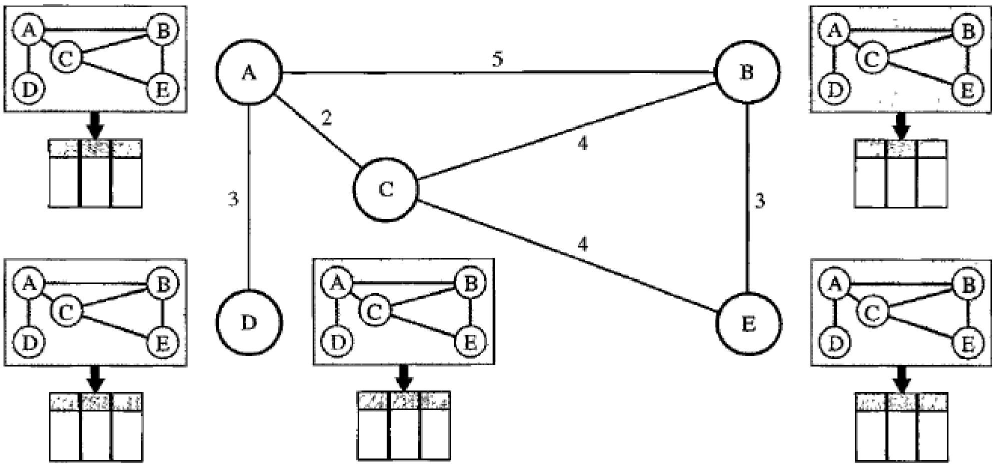
To	Cost	Next
A		
B		—
C		—
D		C
E		

E's table

Link State Routing

Link state routing has a different philosophy from that of distance vector routing. In link state routing, if each node in the domain has the entire topology of the domainthe list of nodes and links, how they are connected including the type, cost (metric), and condition of the links (up or down)-the node can use Dijkstra's algorithm to build a routing table

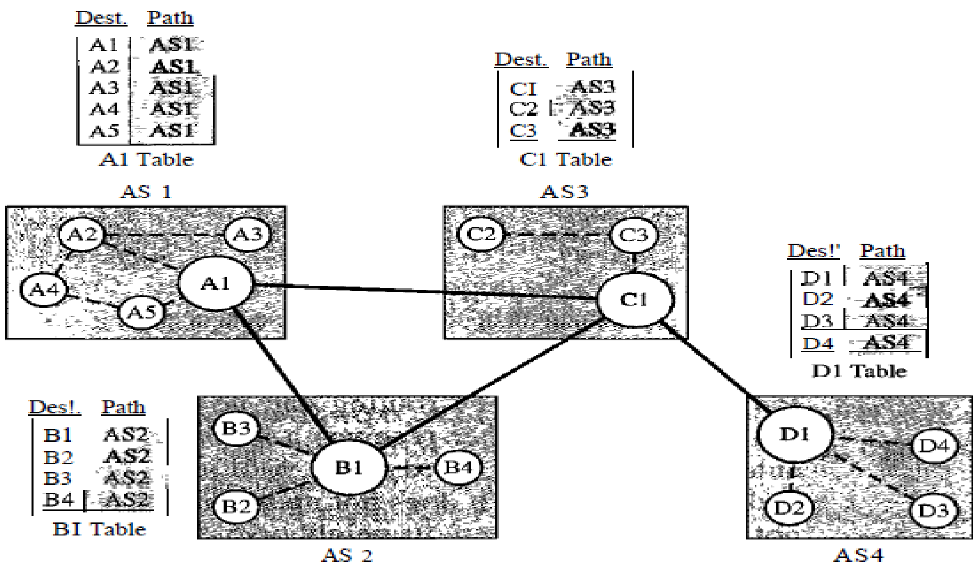
Figure *Concept oflink state routing*



Path Vector Routing

Distance vector and link state routing are both intradomain routing protocols. They can be used inside an autonomous system, but not between autonomous systems. These two protocols are not suitable for interdomain routing mostly because of scalability. Both of these routing protocols become intractable when the domain of operation becomes large. Distance vector routing is subject to instability if there are more than a few hops in the domain of operation. Link state routing needs a huge amount of resources to calculate routing tables. It also creates heavy traffic because of flooding. There is a need for a third routing protocol which we call path vector routing.

Figure *Initial routing tables in path vector routing*



Dest.	Path
A1	AS1
A2	AS1
A3	AS1
A4	AS1
A5	AS1

A1 Table

Dest.	Path
CI	AS3
C2	AS3
C3	AS3

C1 Table

Des!'	Path
D1	AS4
D2	AS4
D3	AS4
D4	AS4

D1 Table

Des!.	Path
B1	AS2
B2	AS2
B3	AS2
B4	AS2

BI Table

MULTICAST ROUTING PROTOCOLS

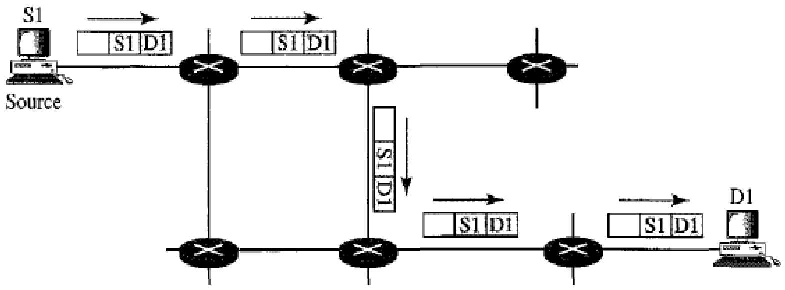
Unicast, Multicast, **and** Broadcast:

A message can be unicast, multicast, or broadcast.

Unicasting

In unicast communication, there is one source and one destination. The relationship between the source and the destination is one-to-one. In this type of communication, both the source and destination addresses, in the IP datagram, are the unicast addresses assigned to the hosts.

Figure 22.33 *Unicasting*



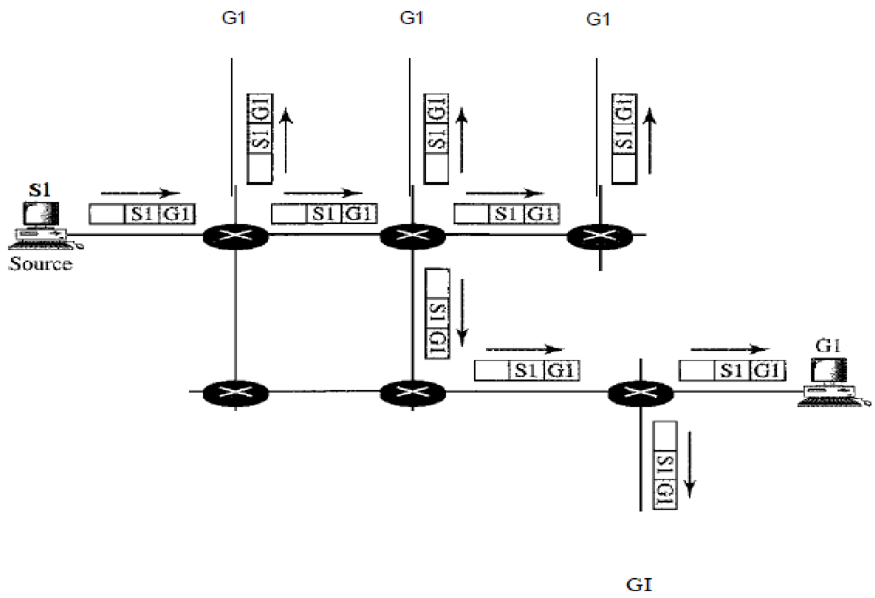
A unicast packet starts from the source S1 and passes through routers to reach the destination D1. In unicasting, when a router receives a packet, it forwards the packet through only one of its interfaces (the one belonging to the optimum path) as defined in the routing table. The router may discard the packet if it cannot find the destination address in its routing table.

In unicasting, the router forwards the received packet through only one of its interfaces.

Multicasting

In multicast communication, there is one source and a group of destinations. The relationship is one-to-many. In this type of communication, the source address is a unicast address, but the destination address is a group address, which defines one or more destinations. The group address identifies the members of the group.

Figure *Multicasting*



A multicast packet starts from the source S1 and goes to all destinations that belong to group G1. In multicasting, when a router receives a packet, it may forward it through several of its interfaces **.**

In multicasting, the router may forward the received packet through several of its interfaces.

Broadcasting

In broadcast communication, the relationship between the source and the destination is one-to-all. There is only one source, but all the other hosts are the destinations. The Internet does not explicitly support broadcasting because of the huge amount of traffic it would create and because of the bandwidth it would need. Imagine the traffic generated in the Internet if one person wanted to send a message to everyone else connected to the Internet.

Unicast and Multicast Routing

In this section, we first discuss the idea of optimal routing, common in all multicast protocols. We then give an overview of multicast routing protocols.

Optimal Routing: Shortest Path Trees

The process of optimal interdomain routing eventually results in the finding of the *shortest path tree.* The root of the tree is the source, and the leaves are the potential destinations. The path from the root to each destination is the shortest path. However, the number of trees and the formation of the trees in unicast and multicast routing are different. Let us discuss each separately.

Unicast Routing In unicast routing, when a router receives a packet to forward, it needs to find the shortest path to the destination of the packet. The router consults its routing table for that particular destination. The next-hop entry corresponding to the destination is the start of the shortest path. The router knows the shortest path for each destination, which means that the router has a shortest path tree to optimally reach all destinations. In other words, each line of the routing table is a shortest path; the whole routing table is a shortest path tree. In unicast routing, each router needs only one shortest path tree to forward a packet; however, each router has its own shortest path tree.

Multicast Routing When a router receives a multicast packet, the situation is different from when it receives a unicast packet. A multicast packet may have destinations in more than one network. Forwarding of a single packet to members of a group requires a shortest path tree. If we have *n* groups, we may need *n* shortest path trees. We can imagine the complexity of multicast routing. Two approaches have been used to solve the problem: source-based trees and group-shared trees.