

CNND Unit IV

Network Layer

Introduction: Network-

Layer Services, Packet Switching, Network-Layer Performance, Network-Layer Congestion, Structure of A Router, Network Layer Protocols: IPv4 Datagram Format, IPv4 Addresses, Forwarding of IP Packets,

ICMPv4,

Unicast Routing: General Idea, Routing Algorithms, Unicast Routing Protocols, Multicast Routing : Introduction, Multicasting Basics, Intradomain Routing Protocols,

Next generation IP: Packet Format , IPv6 Addressing , Transition from IPv4 to IPv6, ICMPv6,

Mobile IP: Addressing , Agents , Three Phases , Inefficiency in Mobile IP.



Introduction:

- The Internet is made of many networks (or links) connected through the connecting devices.
- In other words, the Internet is an internetwork, a combination of LANs and WANs.
- To better understand the role of the network layer (or the internetwork layer), we need to think about the connecting devices (routers or switches) that connect the LANs and WANs.



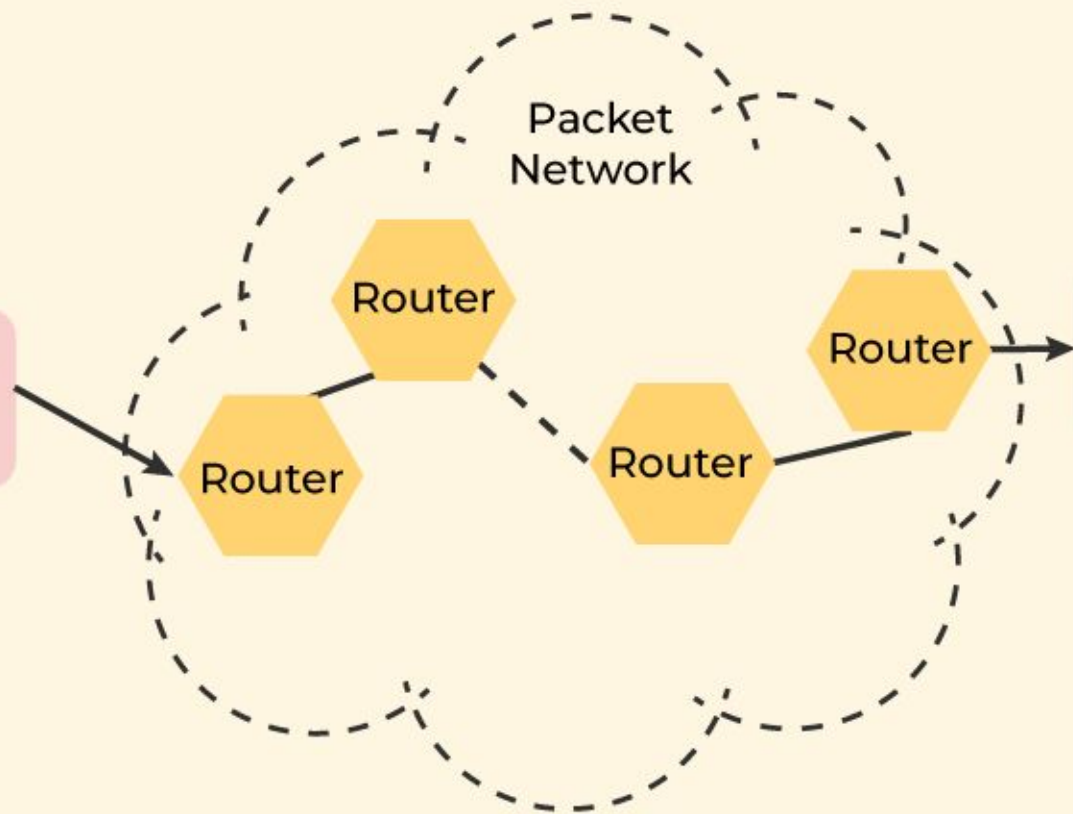
Network-Layer Services

- **Packetizing**

- The process of encapsulating the data received from the upper layers of the network (also called payload) in a network layer packet at the source and decapsulating the payload from the network layer packet at the destination is known as packetizing.
- The source host adds a header that contains the source and destination address and some other relevant information required by the network layer protocol to the payload received from the upper layer protocol and delivers the packet to the data link layer.
- The destination host receives the network layer packet from its data link layer, decapsulates the packet, and delivers the payload to the corresponding upper layer protocol.
- The routers in the path are not allowed to change either the source or the destination address.
- The routers in the path are not allowed to decapsulate the packets they receive unless they need to be fragmented.



Traffic Source
with Shaper



Packet
Network

Router

Router

Router

Router

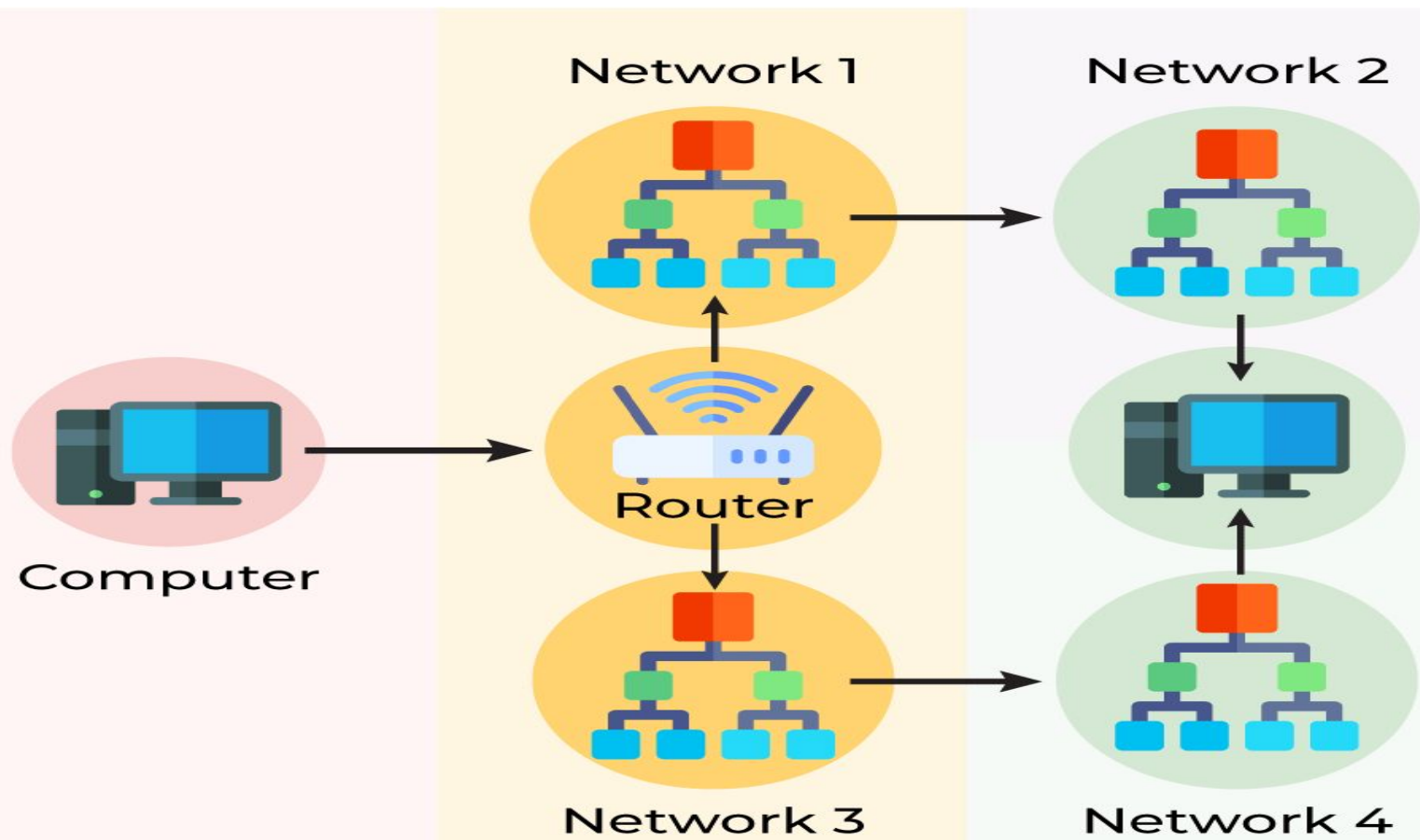
Receiver

Network-Layer Services

- **Routing**

- Routing is the process of moving data from one device to another device.
- These are two other services offered by the network layer.
- In a network, there are a number of routes available from the source to the destination.
- The network layer specifies some strategies which find out the best possible route.
- This process is referred to as routing.
- There are a number of routing protocols that are used in this process and they should be run to help the routers coordinate with each other and help in establishing communication throughout the network.





Network-Layer Services

- **Forwarding**

- Forwarding is simply defined as the action applied by each router when a packet arrives at one of its interfaces.
- When a router receives a packet from one of its attached networks, it needs to forward the packet to another attached network (unicast routing) or to some attached networks (in the case of multicast routing).
- Routers are used on the network for forwarding a packet from the local network to the remote network.
- So, the process of routing involves packet forwarding from an entry interface out to an exit interface.





Internet



Router



Client



Client

Network-Layer Services

- **Error Control**

- Although it can be implemented in the network layer, it is usually not preferred because the data packet in a network layer may be fragmented at each router, which makes error-checking inefficient in the network layer.

- **Flow Control**

- It regulates the amount of data a source can send without overloading the receiver. If the source produces data at a very faster rate than the receiver can consume it, the receiver will be overloaded with data. To control the flow of data, the receiver should send feedback to the sender to inform the latter that it is overloaded with data. There is a lack of flow control in the design of the network layer. It does not directly provide any flow control. The datagrams are sent by the sender when they are ready, without any attention to the readiness of the receiver.

- **Congestion Control**

- Congestion occurs when the number of datagrams sent by the source is beyond the capacity of the network or routers. This is another issue in the network layer protocol. If congestion continues, sometimes a situation may arrive where the system collapses and no datagrams are delivered. Although congestion control is indirectly implemented in the network layer, still there is a lack of congestion control in the network layer.

Network-Layer Services

- **Quality of Service**

- As the Internet has allowed new applications such as multimedia communication (in particular real-time communication of audio and video), the quality of service (QoS) of the communication has become more and more important.
- The Internet has thrived by providing better quality of service to support these applications. However, to keep the network layer untouched, these provisions are mostly implemented in the upper layer.

- **Security**

- Another issue related to communication at the network layer is security. Security was not a concern when the Internet was originally designed because it was used by a small number of users at universities for research activities; other people had no access to the Internet. The network layer was designed with no security provision.
- Today, however, security is a big concern. To provide security for a connectionless network layer, we need to have another virtual level that changes the connectionless service to a connection-oriented service. This virtual layer, called IPSec.




Packet Switching

- **Packet Switching** in computer networks is a method of transferring data to a network in the form of packets.
- In order to transfer the file fast and efficiently manner over the network and minimize the transmission latency, the data is broken into small pieces of variable length, called **Packet**.
- At the destination, all these small parts (packets) have to be reassembled, belonging to the same file.
- A packet is composed of a payload and various control information.
- No pre-setup or reservation of resources is needed.



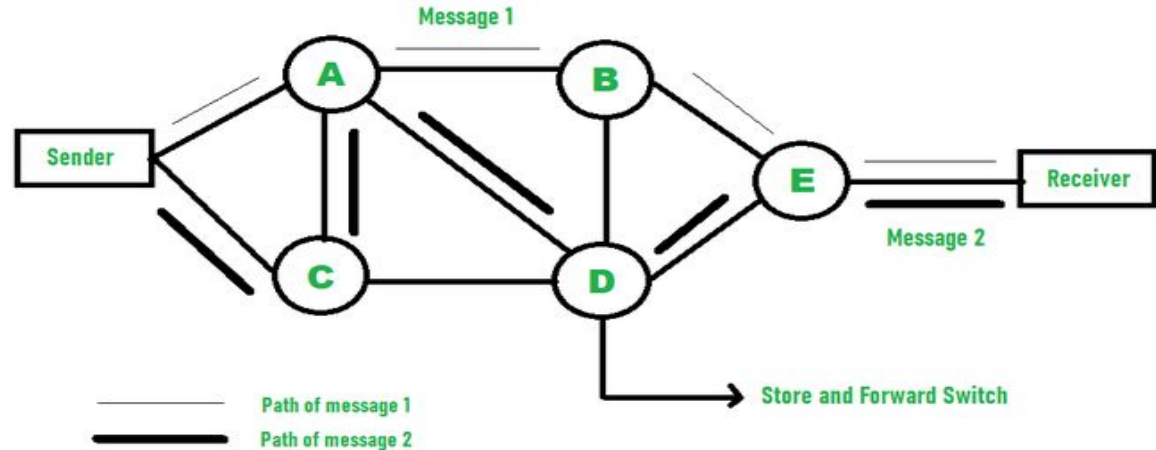
Packet Switching

- Packet Switching uses the **Store and Forward** technique while switching the packets; while forwarding the packet each hop first stores that packet then forwards.
 - This technique is very beneficial because packets may get discarded at any hop for some reason.
 - More than one path is possible between a pair of sources and destinations.
 - Each packet contains the Source and destination address using which they independently travel through the network.
 - In other words, packets belonging to the same file may or may not travel through the same path.
 - If there is congestion at some path, packets are allowed to choose different paths possible over an existing network.
- 

Packet Switching

Types of Delays in Packet Switching

- **Transmission Delay:** Time required by the **spent** station to transmit data to the link.
- **Propagation Delay:** Time of data propagation through the link.
- **Queueing Delay:** Time spent by the packet at the destination's queue.
- **Processing Delay:** Processing time for data at the destination.

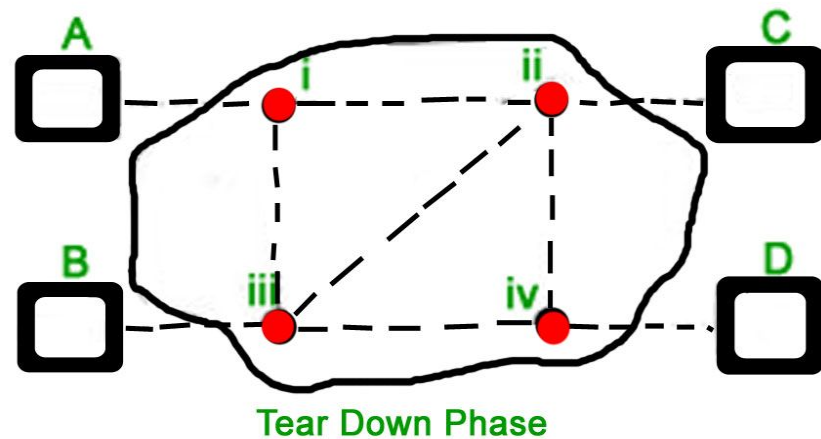
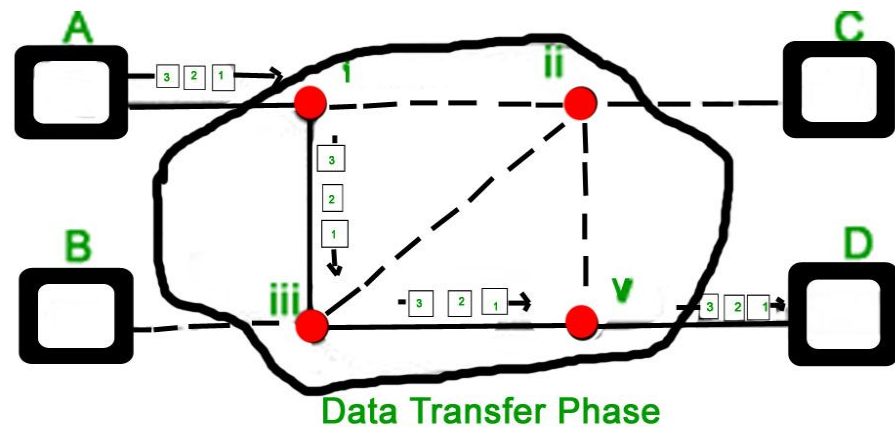
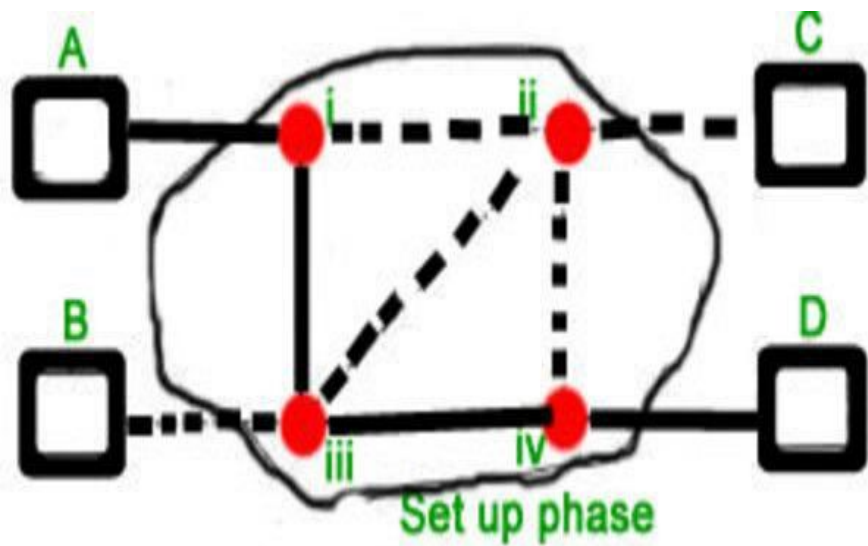


Types of Packet Switching

1. Connection-oriented Packet Switching (Virtual Circuit)

- Before starting the transmission, it establishes a logical path or virtual connection using a signaling protocol, between sender and receiver and all packets belongs to this flow will follow this predefined route.
- Virtual Circuit ID is provided by switches/routers to uniquely identify this virtual connection. Data is divided into small units and all these small units are appended with help of sequence numbers. Packets arrive in order at the destination. Overall, three phases take place here- The setup, data transfer and tear-down phases.
- All address information is only transferred during the setup phase. Once the route to a destination is discovered, entry is added to the switching table of each intermediate node. During data transfer, packet header (local header) may contain information such as length, timestamp, sequence number, etc.
- Connection-oriented switching is very useful in switched WAN. Some popular protocols which use the Virtual Circuit Switching approach are X.25, Frame-Relay, ATM, and MPLS(Multi-Protocol Label Switching).



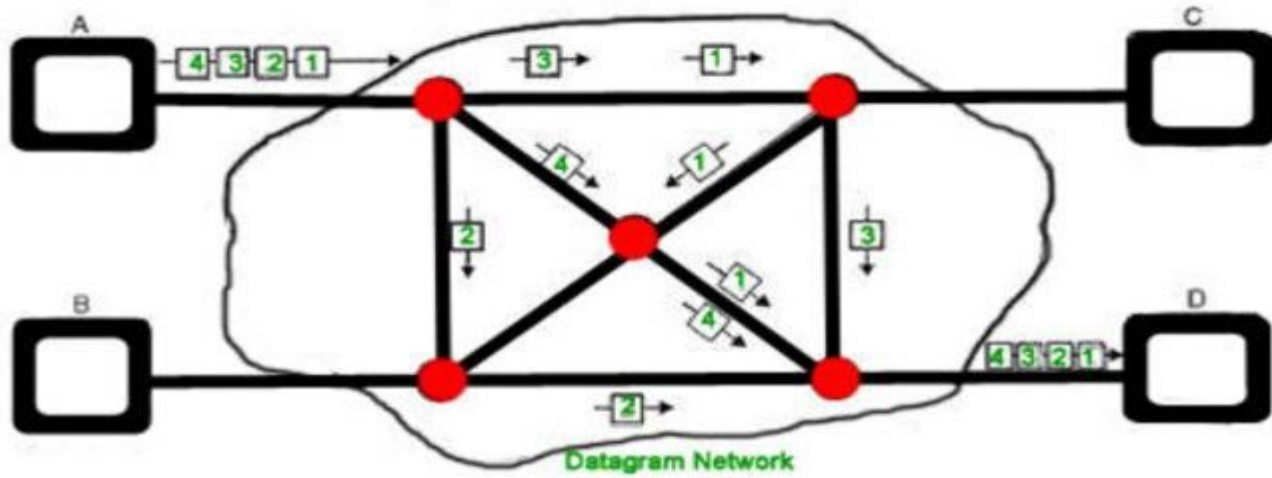


Phases in virtual circuit packet switching

2. Connectionless Packet Switching (Datagram)

- Unlike Connection-oriented packet switching, In Connectionless Packet Switching each packet contains all necessary addressing information such as source address, destination address, port numbers, etc.
- Packets belonging to one flow may take different routes because routing decisions are made dynamically, so the packets that arrived at the destination might be out of order.
- It has no connection setup and teardown phase, like Virtual Circuits.
- Packet delivery is not guaranteed in connectionless packet switching, so reliable delivery must be provided by end systems using additional protocols.



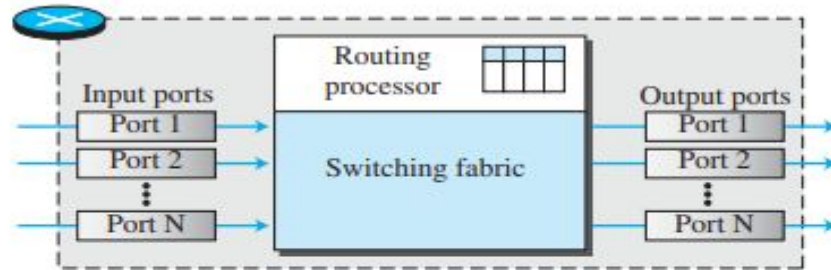


Datagram Packet Switching

A—R1—R2—BA is the sender (start) R1, R2 are two routers that store and forward data B is receiver(destination)

Structure of A Router

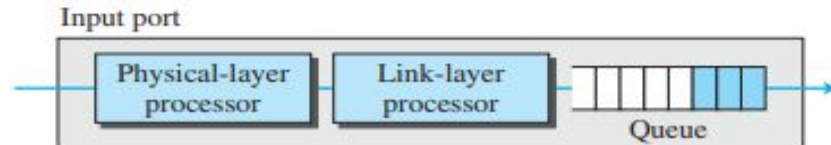
Figure 4.16 Router components



Input Ports

Figure 4.17 shows a schematic diagram of an input port.

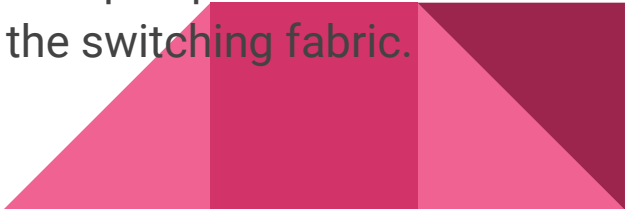
Figure 4.17 Input port



Structure of A Router

A router has four components: **input ports, output ports, the routing processor, and the switching fabric**, as shown in Figure 4.16.

Input Ports

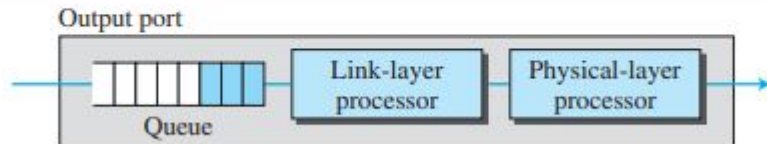
- An input port performs the physical- and link-layer functions of the router.
 - The bits are constructed from the received signal.
 - The packet is decapsulated from the frame, checked for errors, and discarded if corrupted.
 - The packet is then ready to be processed by the network layer. In addition to a physical-layer processor and a link-layer processor, the input port has buffers (queues) to hold the packets before they are directed to the switching fabric.
- 

Structure of A Router

Output Ports

- An output port performs the same functions as an input port, but in the reverse order.
- First the outgoing packets are queued, then each packet is encapsulated in a frame, and finally the physical-layer functions are applied to the frame to create the signal to be sent on the line. Figure 4.18 shows a schematic diagram of an output port.

Figure 4.18 *Output port*



Structure of A Router

Routing Processor

- The routing processor performs the functions of the network layer.
- The destination address is used to find the address of the next hop and, at the same time, the output port number from which the packet is sent out.
- This activity is sometimes referred to as table lookup because the routing processor searches the forwarding table.
- In the newer routers, this function of the routing processor is being moved to the input ports to facilitate and expedite the process



Structure of A Router

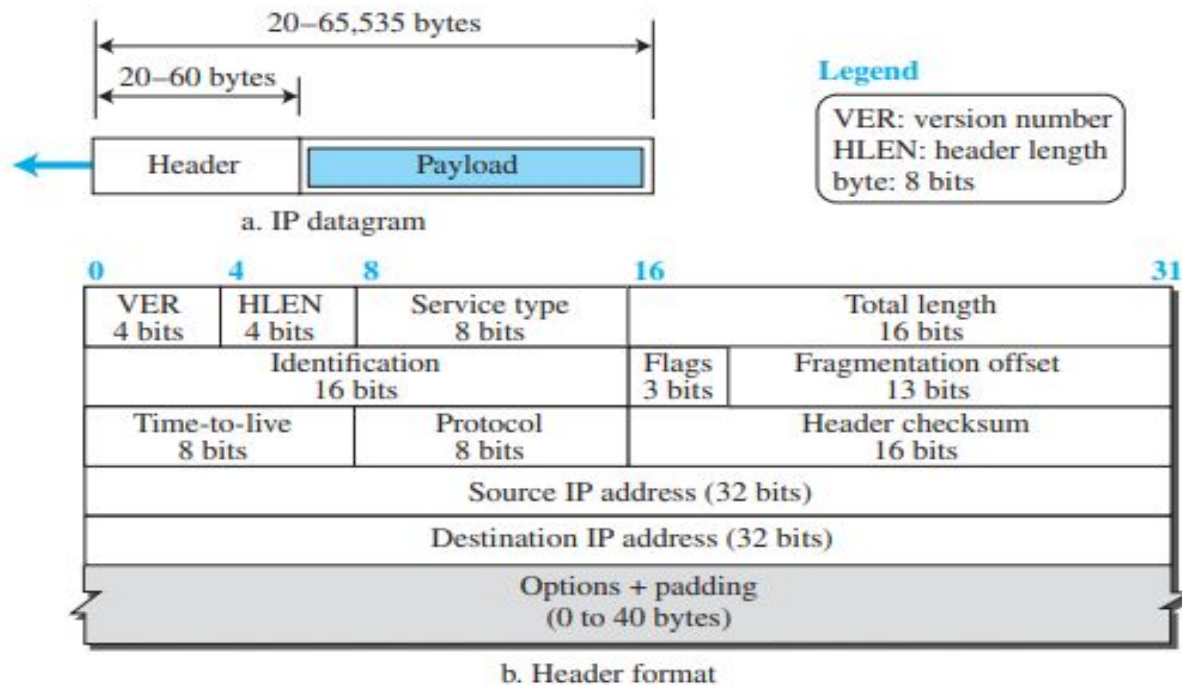
- **Switching Fabric:** This is the heart of the Router, It connects the input ports with the output ports. It is kind of a network inside a networking device. The switching fabric can be implemented in several ways some of the prominent ones are:
 - **Switching via memory:** In this, we have a processor which copies the packet from input ports and sends it to the appropriate output port. It works as a traditional CPU with input and output ports acting as input and output devices.
 - **Switching via bus:** In this implementation, we have a bus that connects all the input ports to all the output ports. On receiving a packet and determining which output port it must be delivered to, the input port puts a particular token on the packet and transfers it to the bus. All output ports can see the packets but they will be delivered to the output port whose token has been put in, the token is then scraped off by that output port and the packet is forwarded
 - **Switching via interconnection network:** This is a more sophisticated network, here instead of a single bus we use a $2N$ bus to connect n input ports to n output ports.

IPv4 Datagram Format

- Packets used by the IP are called datagrams.
- A datagram is a variable-length packet consisting of two parts: header and payload (data).
- The header is 20 to 60 bytes in length and contains information essential to routing and delivery. It is customary in TCP/IP to show the header in 4-byte sections.



Figure 4.24 *IP datagram*



IPv4 Datagram Format

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: defines how the datagram should be handled. 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.



IPv4 Datagram Format

Identification: It is a 16-bit field used to identify the fragments of an original IP datagram. If an IP packet is fragmented during the transmission, Each fragment contains the same identification number. It helps to identify the original IP packet they belong to.

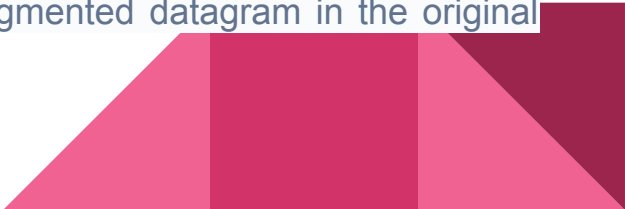
Flags: 3 flags of 1 bit each :

reserved bit (must be zero),

do not fragment flag, –DF bit stands for Do Not Fragment bit. It is a 1-bit field, and its value may be 0 or 1. When it is set to 0, it allows intermediate devices to fragment the datagram if necessary. When it is set to 1, it instructs the intermediary devices to avoid IP datagram fragmentation at all costs.

more fragments flag (same order) –MF bit stands for More Fragments bit. It is a 1-bit field, and its value may be 0 or 1. If it is set to 0, it indicates to the receiver that the current datagram is either the final fragment of the set or the only fragment. If it is set to 1, it indicates to the receiver that the current datagram is a fragment of some larger datagram.

Fragment Offset: It is a 13-bit field. It tells the exact position of the fragmented datagram in the original unfragmented IP datagram.



IPv4 Datagram Format

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.

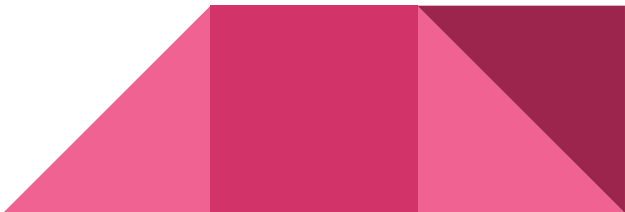
Protocol: Name of the protocol to which the data is to be passed (8 bits)

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.

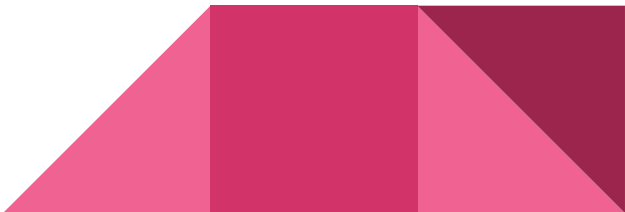


IPv4 Addresses

IP Address

- The identifier used in the IP layer of the TCP/IP protocol suite to identify the connection of each device to the Internet is called the Internet address or IP address.
- An IPv4 address is a 32-bit address that uniquely and universally defines the connection of a host or a router to the Internet.
- IPv4 addresses are unique in the sense that each address defines one, and only one, connection to the Internet.
- If a device has two connections to the Internet, via two networks, it has two IPv4 addresses.

Address space

- It is the total number of addresses used by the protocol.
 - If a protocol uses N bits to define an address, the address space is 2^N because each bit can have two different values (0 and 1) and N bits can have 2^N values.
 - The address space of IPv4 is 2^{32} or 4,294,967,296.
- 

Classful Addressing

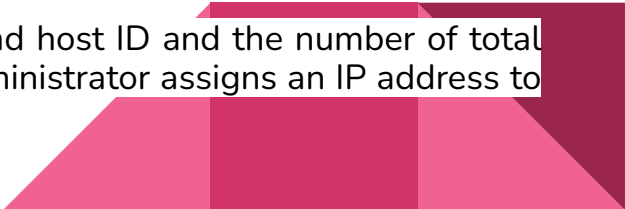
The 32-bit IP address is divided into five sub-classes. These are:

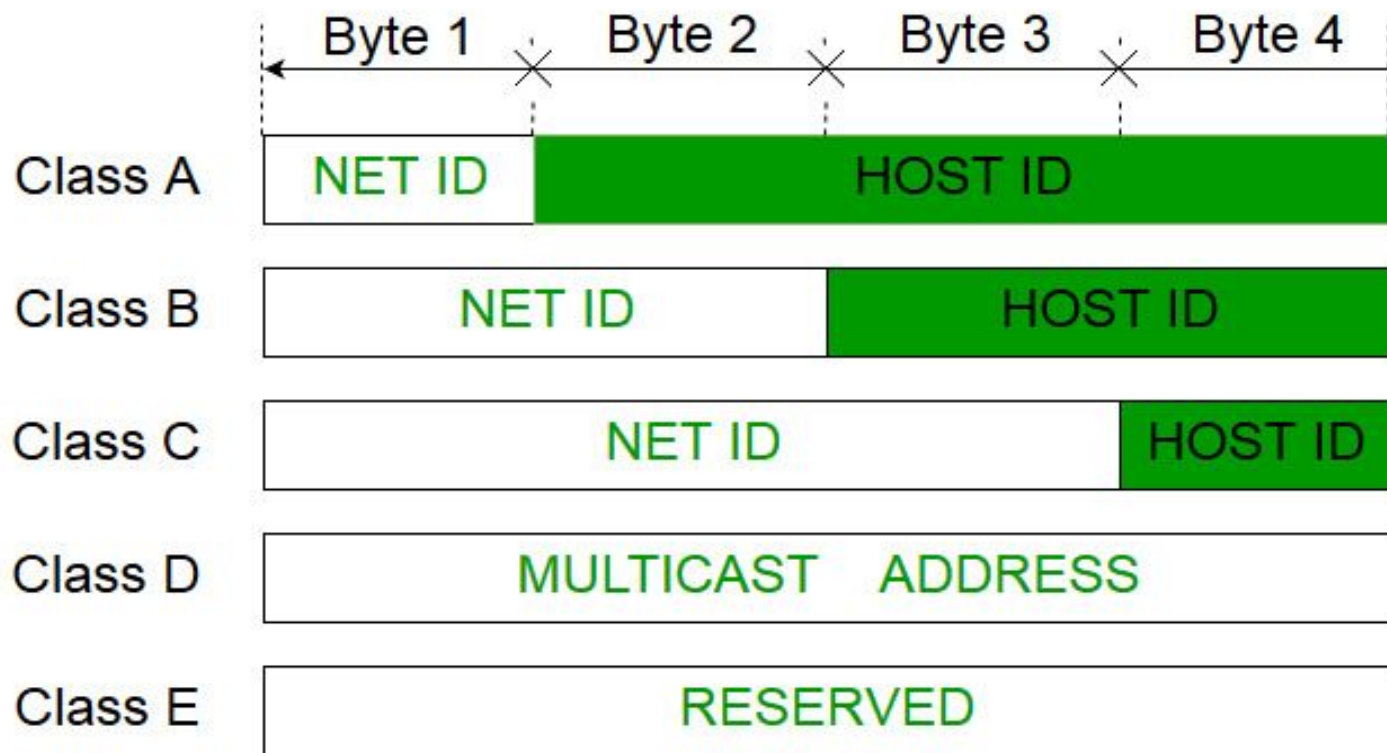
- Class A
- Class B
- Class C
- Class D
- Class E

Each of these classes has a valid range of IP addresses. Classes D and E are reserved for multicast and experimental purposes respectively. The order of bits in the first octet determines the classes of the IP address. The IPv4 address is divided into two parts:

- Network ID
- Host ID

The class of IP address is used to determine the bits used for network ID and host ID and the number of total networks and hosts possible in that particular class. Each ISP or network administrator assigns an IP address to each device that is connected to its network.





Class A

IP addresses belonging to class A are assigned to the networks that contain a large number of hosts.

- The network ID is 8 bits long.
- The host ID is 24 bits long.

The higher-order bit of the first octet in class A is always set to 0. The remaining 7 bits in the first octet are used to determine network ID. The 24 bits of host ID are used to determine the host in any network. The default subnet mask for Class A is 255.0.0.0. Therefore, class A has a total of:

- $2^{24} - 2 = 16,777,214$ host ID

IP addresses belonging to class A ranges from 0 – 127.



Class A

Class B

IP address belonging to class B is assigned to networks that range from medium-sized to large-sized networks.

- The network ID is 16 bits long.
- The host ID is 16 bits long.

The higher-order bits of the first octet of IP addresses of class B are always set to 10. The remaining 14 bits are used to determine the network ID. The 16 bits of host ID are used to determine the host in any network. The default subnet mask for class B is 255.255.0.0. Class B has a total of:

- $2^{14} = 16384$ network address
- $2^{16} - 2 = 65534$ host address

IP addresses belonging to class B ranges from 128 – 191.



Class B

Class C

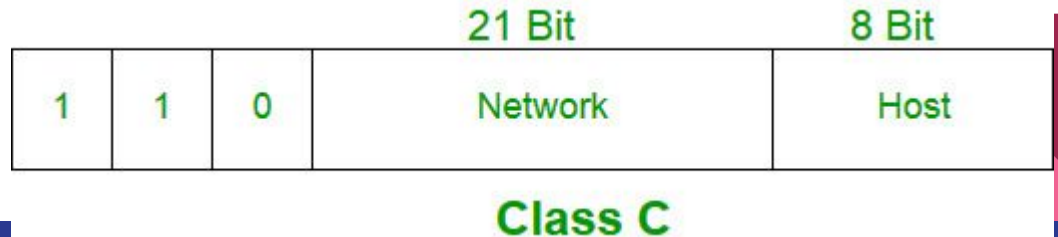
IP addresses belonging to class C are assigned to small-sized networks.

- The network ID is 24 bits long.
- The host ID is 8 bits long.

The higher-order bits of the first octet of IP addresses of class C is always set to 110. The remaining 21 bits are used to determine the network ID. The 8 bits of host ID are used to determine the host in any network. The default subnet mask for class C is 255.255.255.0. Class C has a total of:

- $2^{21} = 2097152$ network address
- $2^8 - 2 = 254$ host address

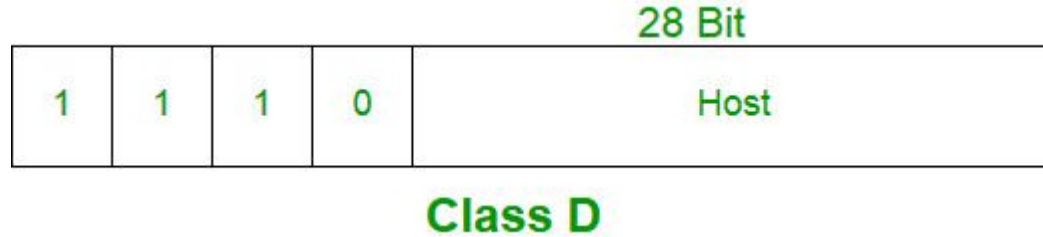
IP addresses belonging to class C range from 192 – 223.



Class D

IP address belonging to class D is reserved for multi-casting. The higher-order bits of the first octet of IP addresses belonging to class D is always set to 1110. The remaining bits are for the address that interested hosts recognize.

Class D does not possess any subnet mask. IP addresses belonging to class D range from 224 – 239.



Class E

IP addresses belonging to class E are reserved for experimental and research purposes. IP addresses of class E range from 240 – 255. This class doesn't have any subnet mask. The higher-order bits of the first octet of class E are always set to 1111.



Class E

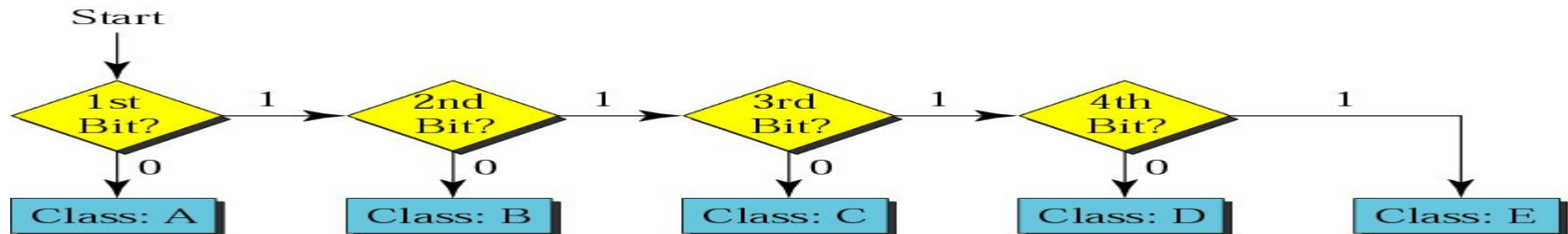
Recognizing classes

	First byte	Second byte	Third byte	Fourth byte
Class A	0			
Class B	10			
Class C	110			
Class D	1110			
Class E	1111			

a. Binary notation

	First byte	Second byte	Third byte	Fourth byte
Class A	0-127			
Class B	128-191			
Class C	192-223			
Class D	224-239			
Class E	240-255			

b. Dotted-decimal notation



Example:

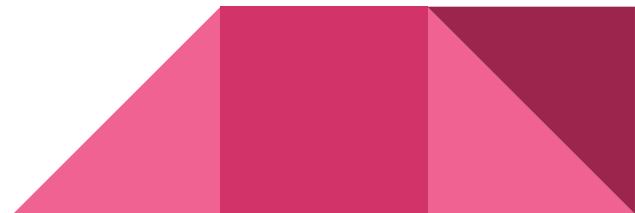
Find the class of each address:

a. 00000001 00001011 00001011 11101111

b. 11000001 10000011 00011011 11111111

c. 10100111 11011011 10001011 01101111

d. 11110011 10011011 11111011 00001111



Solution

- a. The first bit is 0. This is a class A address.
- b. The first 2 bits are 1; the third bit is 0. This is a class C address.
- c. The first bit is 0; the second bit is 1. This is a class B address.
- d. The first 4 bits are 1s. This is a class E address..



Example:

Find the class of each address:

a. 227.12.14.87

b. 193.14.56.22

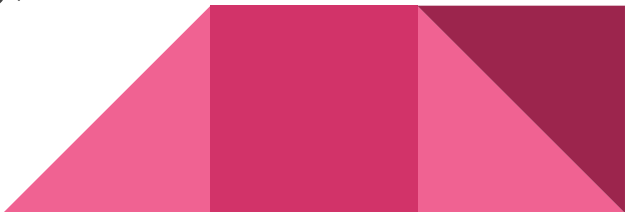
c. 14.23.120.8

d. 252.5.15.111

e. 134.11.78.56



Solution

- a. The first byte is 227 (between 224 and 239); the class is D.
 - b. The first byte is 193 (between 192 and 223); the class is C.
 - c. The first byte is 14 (between 0 and 127); the class is A.
 - d. The first byte is 252 (between 240 and 255); the class is E.
 - e. The first byte is 134 (between 128 and 191); the class is B.
- 

Netid and Hostid

Class A:	<table><tr><td>0</td><td>7 bits Network ID</td><td>24 bits Host ID</td></tr></table>	0	7 bits Network ID	24 bits Host ID	Very large Network (2^24 hosts !)		
0	7 bits Network ID	24 bits Host ID					
Class B:	<table><tr><td>1</td><td>0</td><td>14 bits Network ID</td><td>16 bits Host ID</td></tr></table>	1	0	14 bits Network ID	16 bits Host ID	Medium size Network (Most popular !!!)	
1	0	14 bits Network ID	16 bits Host ID				
Class C:	<table><tr><td>1</td><td>1</td><td>0</td><td>21 bits Network ID</td><td>8 bits Host ID</td></tr></table>	1	1	0	21 bits Network ID	8 bits Host ID	Small Network
1	1	0	21 bits Network ID	8 bits Host ID			
Class D:	<table><tr><td>1</td><td>1</td><td>1</td><td>0</td><td>Multicast Group ID</td></tr></table>	1	1	1	0	Multicast Group ID	Multicast Address
1	1	1	0	Multicast Group ID			
Class E:	<table><tr><td>1</td><td>1</td><td>1</td><td>1</td><td></td></tr></table>	1	1	1	1		Reserved (unused)
1	1	1	1				

Netid and Hostid

Here, each class has a fixed number of hostid and netid.

Network part :

The network part is also known as net id which is used to classify the network to which the host is connected.

Host part :

The host part is also known as the host id which is the part of the IP address which is used to uniquely identify the host on a network.



Subnet Mask

The 32-bit IP address contains information about the host and its network. It is very necessary to distinguish both.

For this, routers use Subnet Mask, which is as long as the size of the network address in the IP address.

Subnet Mask is also 32 bits long. If the IP address in binary is ANDed with its Subnet Mask, the result yields the Network address.

For example, say the IP Address is 192.168.1.152 and the Subnet Mask is 255.255.255.0 then –

IP	192.168.1.152	11000000	10101000	00000001	10011000	} ANDed
Mask	255.255.255.0	11111111	11111111	11111111	00000000	
Network	192.168.1.0	11000000	10101000	00000001	00000000	Result

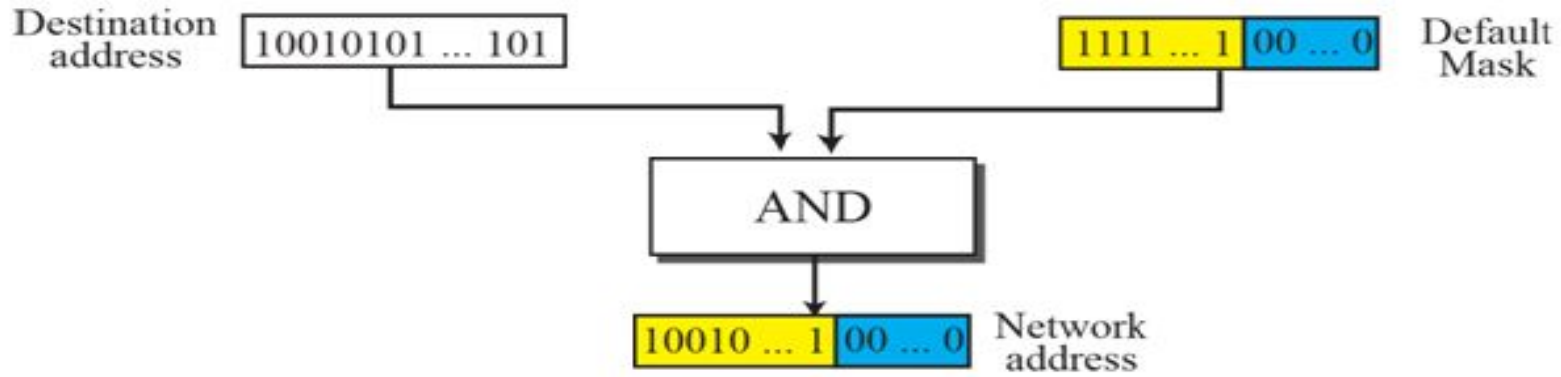
This way the Subnet Mask helps extract the Network ID and the Host from an IP Address. It can be identified now that 192.168.1.0 is the Network number and 192.168.1.152 is the host on that network.

Subnet Mask

Table 1.8. Default Subnet Masks

IP Address Class	Default Subnet Mask
Class A	255.0.0.0
Class B	255.255.0.0
Class C	255.255.255.0

Finding a network address using the default mask



To extract the network address from destination address of a packet, a router uses the AND operation.

Example

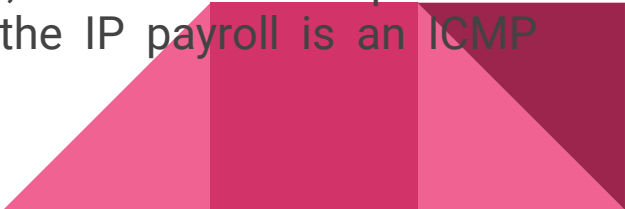
A router receives a packet with the destination address 201.24.67.32. Show how the router finds the network address of the packet.

Solution

Since the class of the address is C, we assume that the router applies the default mask for class B, 255.255.0.0 to find the network address.

Destination address	→	201	.	24	.	67	.	32
Default mask	→	255	.	255	.	0	.	0
Network address	→	201	.	24	.	0	.	0

ICMPv4

- The IPv4 has no error-reporting or error-correcting mechanism. In 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 manager needs information from another host or router.
 - **The Internet Control Message Protocol version 4 (ICMPv4)** has been designed to compensate for the above two deficiencies. It is a companion to the IP protocol.
 - ICMP itself is a network-layer protocol. However, its messages are not passed directly to the data-link layer as would be expected.
 - Instead, the messages are first encapsulated inside IP datagrams before going to the lower layer.
 - When an IP datagram encapsulates an ICMP message, the value of the protocol field in the IP datagram is set to 1 to indicate that the IP payload is an ICMP message.
- 

ICMPv4

Messages

- ICMPv4 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.
- For example, hosts can discover their neighbor hosts or routers on their network and routers can help a node redirect its messages.



ICMPv4

Message Format


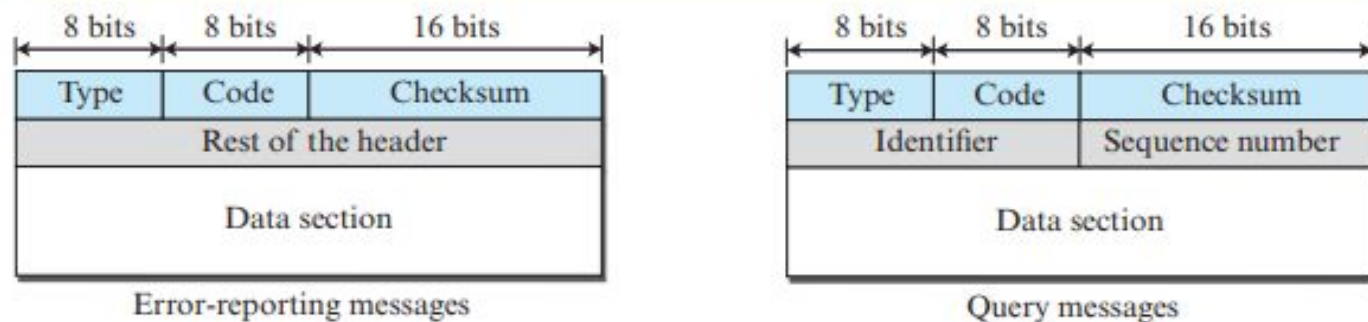
- An ICMPv4 message has an 8-byte header and a variable-size data section.
 - Although the general format of the header is different for each message type, the first 4 bytes are common to all.
 - As Figure shows, the first field, **ICMP type**, defines the type of the message.
 - The **code field** specifies the reason for the particular message type.
 - The last common field is the **checksum** field.
 - The rest of the header is specific to each message type.
 - 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 query.
- 

Figure 4.54 General format of ICMP messages



Type and code values

Error-reporting messages

- 03:** Destination unreachable (codes 0 to 15)
- 04:** Source quench (only code 0)
- 05:** Redirection (codes 0 to 3)
- 11:** Time exceeded (codes 0 and 1)
- 12:** Parameter problem (codes 0 and 1)

Query messages

- 08 and 00:** Echo request and reply (only code 0)
- 13 and 14:** Timestamp request and reply (only code 0)

Note: See the book website for more explanation about the code values.

ICMPv4

- **Error-reporting messages**

The error-reporting message means that the router encounters a problem when it processes an IP packet then it reports a message.

- **Destination unreachable (type 3).** This message uses different codes (0 to 15) to define the type of error message and the reason why a datagram has not reached its final destination. For example, code 0 tells the source that a host is unreachable. This may happen, for example, when we use the HTTP protocol to access a web page, but the server is down. The message “destination host is not reachable” is created and sent back to the source.
- **Source quench (type 4)** message, which informs the sender that the network has encountered congestion and the datagram has been dropped; the source needs to slow down sending more datagrams. In other words, ICMP adds a kind of congestion control mechanism to the IP protocol by using this type of message.
- **Redirection message (type 5)** is used when the source uses a wrong router to send out its message. The router redirects the message to the appropriate router, but informs the source that it needs to change its default router in the future. The IP address of the default router is sent in the message.

- **The time-to-live (TTL)** field in the IP datagram and explained that it prevents a datagram from being aimlessly circulated in the Internet. When the TTL value becomes 0, the datagram is dropped by the visiting router and a time exceeded message (type 11) with code 0 is sent to the source to inform it about the situation. The time-exceeded message (with code 1) can also be sent when not all fragments of a datagram arrive within a predefined period of time.
- **A parameter problem message (type 12)** can be sent when either there is a problem in the header of a datagram (code 0) or some options are missing or cannot be interpreted (code 1).



ICMPv4

- **Query messages**

The query messages are those messages that help the host to get the specific information of another host. For example, suppose there are a client and a server, and the client wants to know whether the server is live or not, then it sends the ICMP message to the server.



ICMPv4

- **The echo request (type 8) and the echo reply (type 0)** pair of messages are used by a host or a router to test the liveness of another host or router. A host or router sends an echo request message to another host or router; if the latter is alive, it responds with an echo reply message. We shortly see the applications of this pair in two debugging tools: ping and traceroute.
- **The timestamp request (type 13) and the timestamp reply (type 14)** pair of messages are used to find the round-trip time between two devices or to check whether the clocks in two devices are synchronized. The timestamp request message sends a 32-bit number, which defines the time the message is sent. The timestamp reply resends that number, but also includes two new 32-bit numbers representing the time the request was received and the time the response was sent. If all timestamps represent Universal time, the sender can calculate the one-way and round-trip time.

UNICAST ROUTING

- In an internet, the goal of the network layer is to deliver a datagram from its source to its destination or destinations.
- If a datagram is destined for only one destination (one-to-one delivery), we have **unicast routing**.
- If the datagram is destined for several destinations (one-to-many delivery), we have **multicast routing**.
- Finally, if the datagram is supposed to be delivered to all hosts in the internet (one-to-all), we have **broadcast routing**.



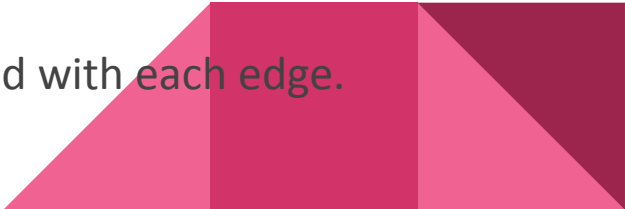
UNICAST ROUTING

- In unicast routing, a packet is routed, hop by hop, from its source to its destination by the help of forwarding tables.
- The source host needs no forwarding table because it delivers its packet to the default router in its local network.
- The destination host needs no forwarding table either because it receives the packet from its default router in its local network.
- This means that only the routers that glue together the networks in the internet need forwarding tables.
- With the above explanation, routing a packet from its source to its destination means routing the packet from a source router (the default router of the source host) to a destination router (the router connected to the destination network).
- There are several routes that a packet can travel from the source to the destination; what must be determined is which route the packet should take.



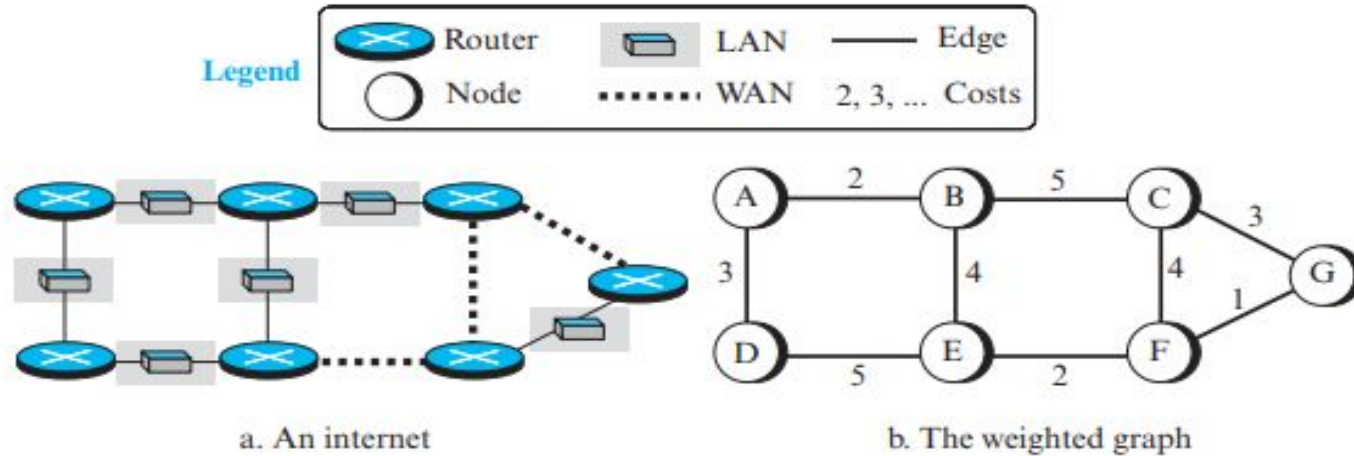
UNICAST ROUTING

An Internet as a Graph

- To find the best route, an internet can be modeled as a graph.
 - A graph in computer science is a set of nodes and edges (lines) that connect the nodes.
 - To model an internet as a graph, we can think of each router as a node and each network between a pair of routers as an edge.
 - An internet is, in fact, modeled as a weighted graph, in which each edge is associated with a cost.
 - If a weighted graph is used to represent a geographical area, the nodes can be cities and the edges can be roads connecting the cities; the weights, in this case, are distances between cities.
 - In routing, however, the cost of an edge has a different interpretation in different routing protocols.
 - For the moment, we assume that there is a cost associated with each edge.
- 

UNICAST ROUTING

Figure 4.56 *An internet and its graphical representation*



Routing Algorithms

Distance-Vector Routing

Link-State Routing

Path-Vector Routing

Unicast Routing Protocols

Routing Information Protocol (RIP)

Open Shortest Path First (OSPF)

Border Gateway Protocol Version 4 (BGP4)



Distance-Vector Routing

- A **distance-vector routing (DVR)** protocol requires that a router inform its neighbors of topology changes periodically.
- Historically known as the old ARPANET routing algorithm (or known as Bellman-Ford algorithm).
- **Bellman Ford Basics** – Each router maintains a Distance Vector table containing the distance between itself and ALL possible destination nodes.
- Distances, based on a chosen metric, are computed using information from the neighbors' distance vectors.



Distance-Vector Routing

Information kept by DV router -

- Each router has an ID
- Associated with each link connected to a router, there is a link cost (static or dynamic).
- Intermediate hops

Distance Vector Table Initialization -

- Distance to itself = 0
- Distance to ALL other routers = infinity number.



Distance-Vector Routing

Distance Vector Algorithm –

1. A router transmits its distance vector to each of its neighbors in a routing packet.
2. Each router receives and saves the most recently received distance vector from each of its neighbors.
3. A router recalculates its distance vector when:
 - It receives a distance vector from a neighbor containing different information than before.
 - It discovers that a link to a neighbor has gone down.



Distance-Vector Routing

The DV calculation is based on minimizing the cost to each destination

$D_x(y)$ = Estimate of least cost from x to y

$C(x,v)$ = Node x knows cost to each neighbor v

$D_x = [D_x(y): y \in N]$ = Node x maintains distance vector

Node x also maintains its neighbors' distance vectors

– For each neighbor v , x maintains $D_v = [D_v(y): y \in N]$



Distance-Vector Routing

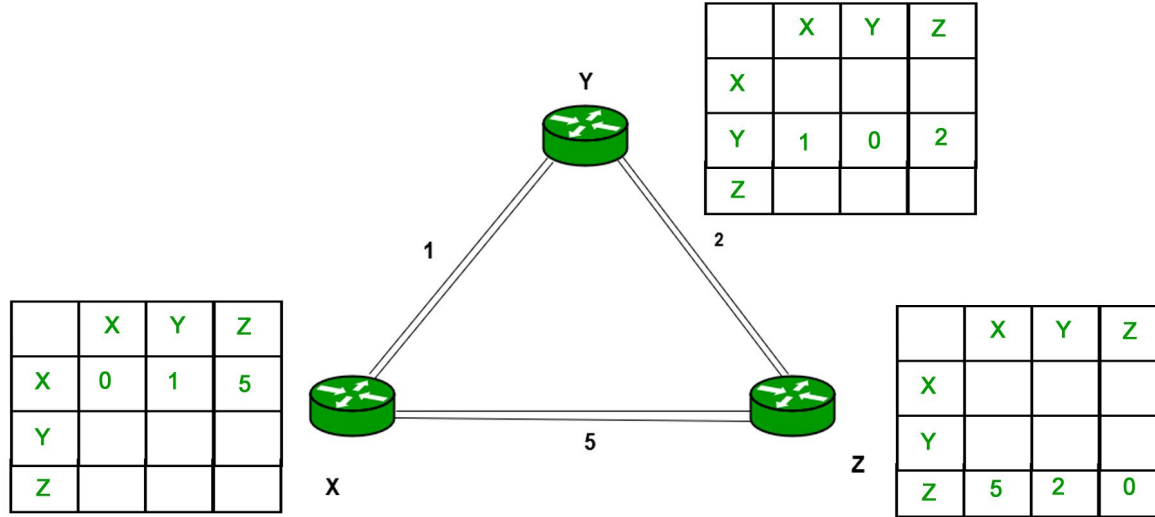
- From time-to-time, each node sends its own distance vector estimate to neighbors.
- When a node x receives new DV estimate from any neighbor v , it saves v 's distance vector and it updates its own DV using B-F equation:

$$D_x(y) = \min \{ C(x,v) + D_v(y), D_x(y) \} \text{ for each node } y \in N$$



Distance-Vector Routing

Example – Consider 3-routers X, Y and Z as shown in figure. Each router have their routing table. Every routing table will contain distance to the destination nodes.

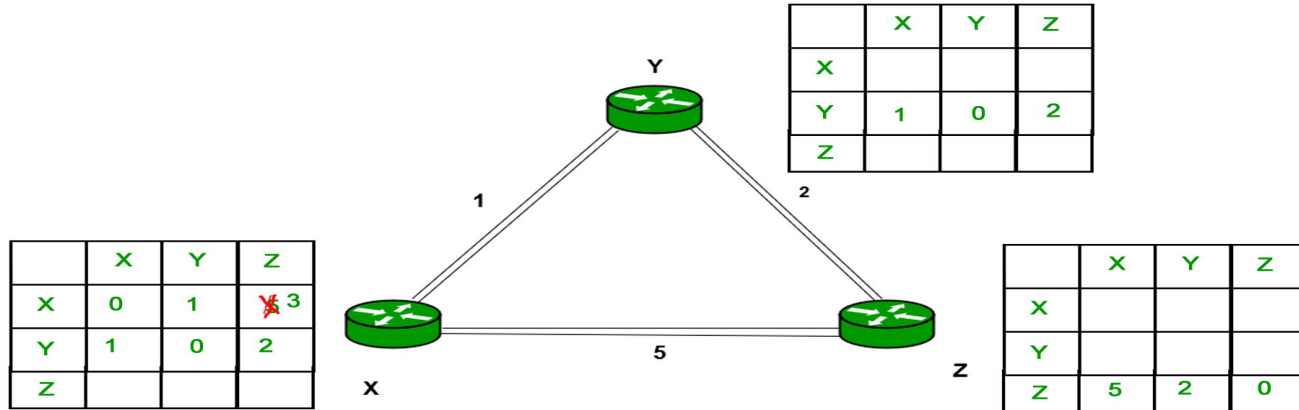


Distance-Vector Routing

Consider router X, X will share its routing table to neighbors and neighbors will share their routing table to it. The distance from node X to destination will be calculated using the Bellman-Ford equation.

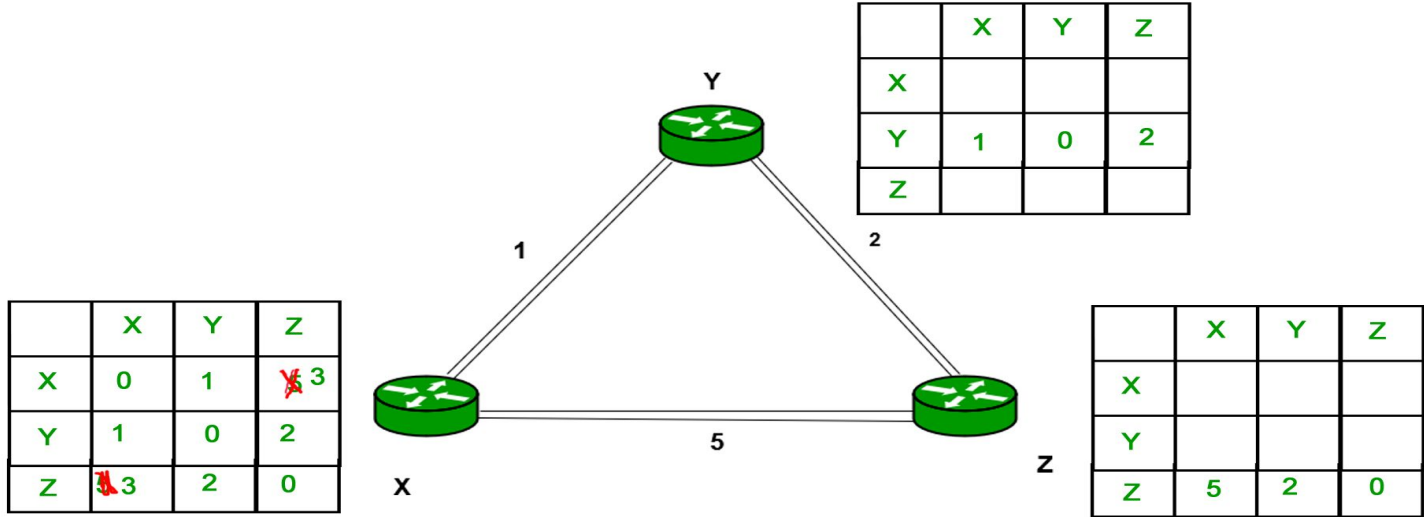
$$D_x(y) = \min \{ C(x,v) + D_v(y) \} \text{ for each node } y \in N$$

As we can see that the distance will be less going from X to Z when Y is an intermediate node (hop), so it will be updated in routing table X.



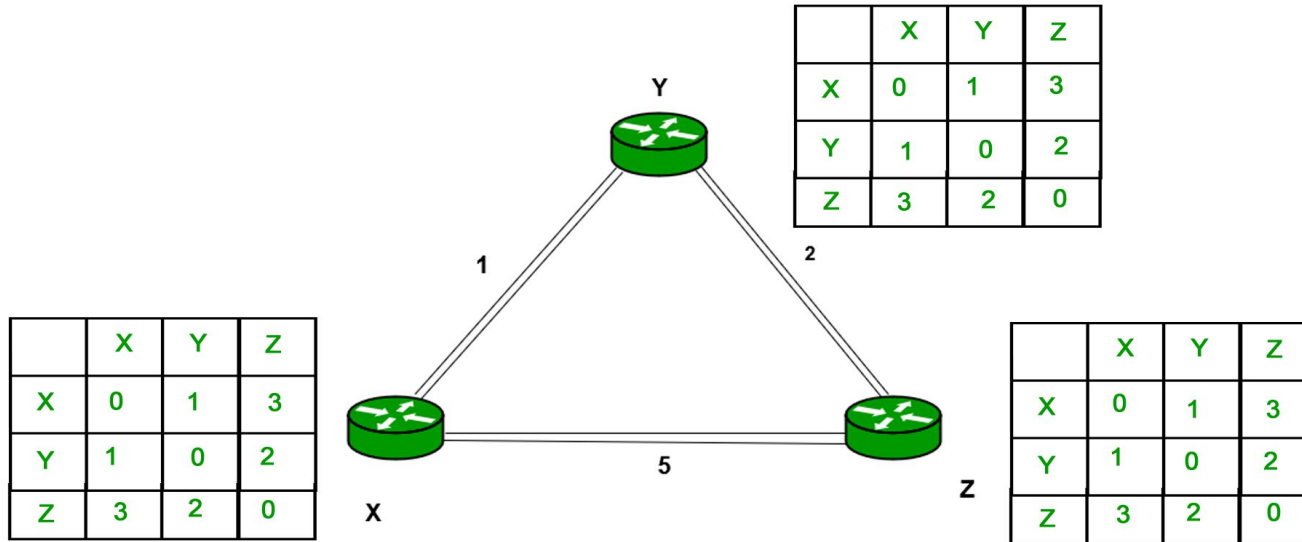
Distance-Vector Routing

Similarly for Z also –



Distance-Vector Routing

Finally the routing table for all –



Link-State Routing

- Link state routing is the second family of routing protocols.
- While distance-vector routers use a distributed algorithm to compute their routing tables, link-state routing uses link-state routers to exchange messages that allow each router to learn the entire network topology.
- Based on this learned topology, each router is then able to compute its routing table by using the shortest path computation.
- Link state routing is a technique in which each router shares the knowledge of its neighborhood with every other router i.e. the internet work. The three keys to understand the link state routing algorithm.



Link-State Routing

The three keys to understand the link state routing algorithm.

1. **Knowledge about the neighborhood:** Instead of sending its routing table, a router sends the information about its neighborhood only. A router broadcast its identities and cost of the directly attached links to other routers.
2. **Flooding:** Each router sends the information to every other router on the internetwork. This process is known as flooding. Every router that receives the packet sends the copies to all the neighbors. Finally each and every router receives a copy of the same information.
3. **Information Sharing:** A router send the information to every other router only when the change occurs in the information.



Link-State Routing

Link state routing has two phase:

1. **Reliable Flooding: Initial state**– Each node knows the cost of its neighbors. Final state- Each node knows the entire graph.
2. **Route Calculation:** Each node uses Dijkstra's algorithm on the graph to calculate the optimal routes to all nodes. The link state routing algorithm is also known as Dijkstra's algorithm which is used to find the shortest path from one node to every other node in the network.



Link-State Routing

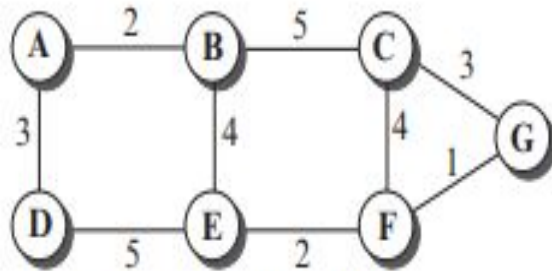
Features of Link State Routing Protocols

- **Link State Packet:** A small packet that contains routing information.
- **Link-State Database:** A collection of information gathered from the link-state packet.
- **Shortest Path First Algorithm (Dijkstra algorithm):** A calculation performed on the database results in the shortest path
- **Routing Table:** A list of known paths and interfaces.



Link-State Routing

Figure 4.63 *Example of a link-state database*



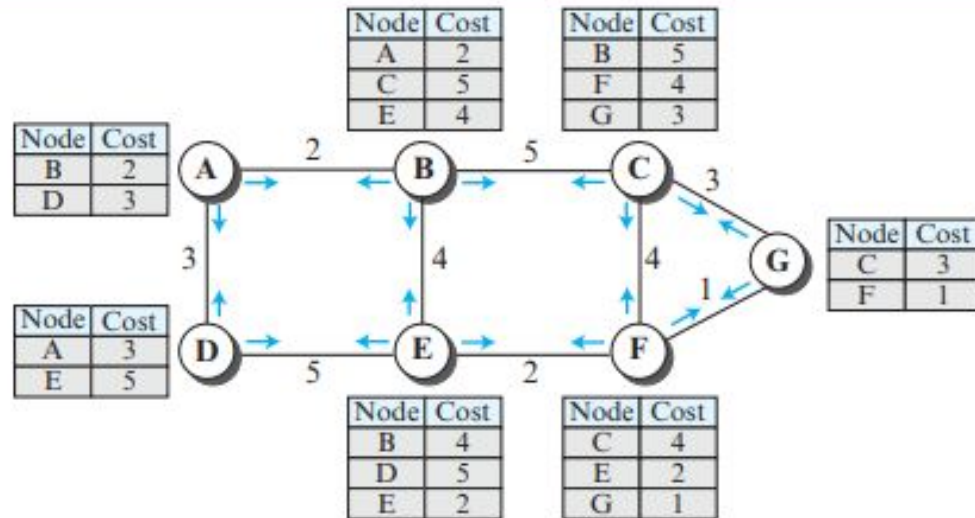
a. The weighted graph

	A	B	C	D	E	F	G
A	0	2	∞	3	∞	∞	∞
B	2	0	5	∞	4	∞	∞
C	∞	5	0	∞	∞	4	3
D	3	∞	∞	0	5	∞	∞
E	∞	4	∞	5	0	2	∞
F	∞	∞	4	∞	2	0	1
G	∞	∞	3	∞	∞	1	0

b. Link state database

Link-State Routing

Figure 4.64 *LSPs created and sent out by each node to build LSDB*

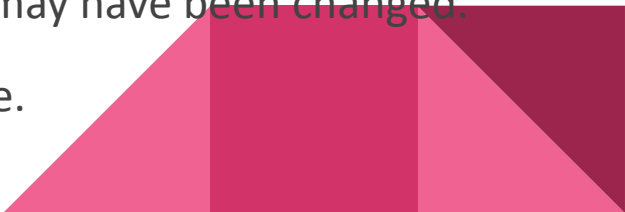


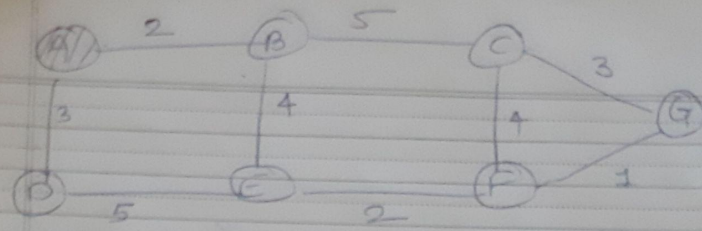
Link-State Routing

Formation of Least-Cost Trees

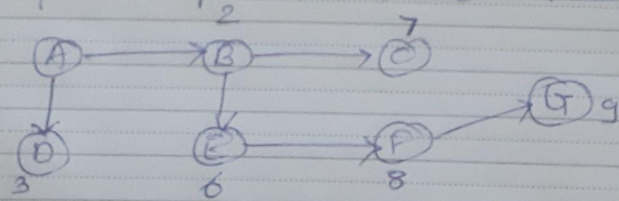
To create a least-cost tree for itself, using the shared LSDB, each node needs to run the famous Dijkstra Algorithm.

This iterative algorithm uses the following steps:

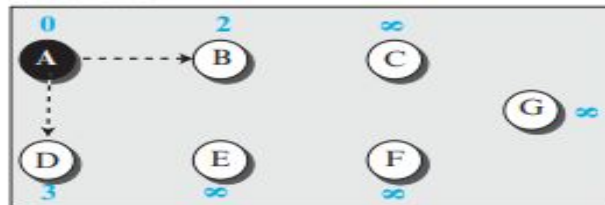
1. The node chooses itself as the root of the tree, creating a tree with a single node, and sets the total cost of each node based on the information in the LSDB.
 2. The node selects one node, among all nodes not in the tree, which is closest to the root, and adds this to the tree. After this node is added to the tree, the cost of all other nodes not in the tree needs to be updated because the paths may have been changed.
 3. The node repeats step 2 until all nodes are added to the tree.
- 



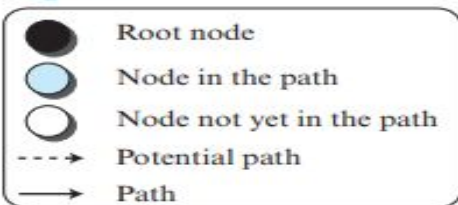
Iteration	Nodes	B	C	D	E	F	G
Initial	A	2	∞	3	∞	∞	∞
1	A, B	2	7	3	6	∞	∞
2	A, B, D	2	7	3	6	∞	∞
3	A, B, D, E	2	7	3	6	8	∞
4	A, B, D, E, C	2	7	3	6	8	10
5	A, B, D, E, C, F	2	7	3	6	8	9
6	A, B, D, E, C, F, G	2	7	3	6	8	9



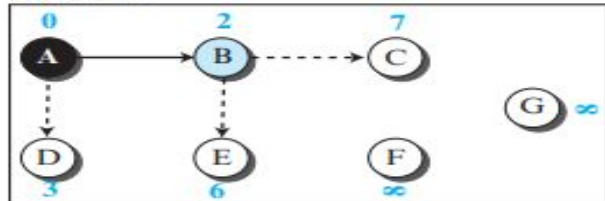
Initialization



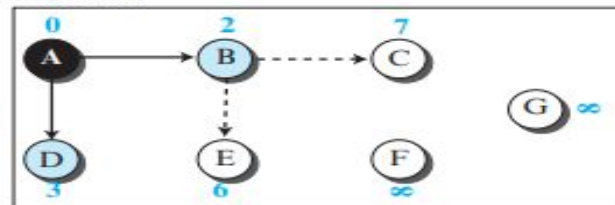
Legend



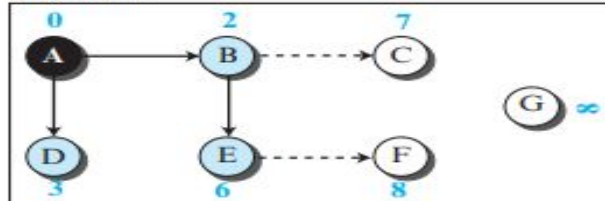
Iteration 1



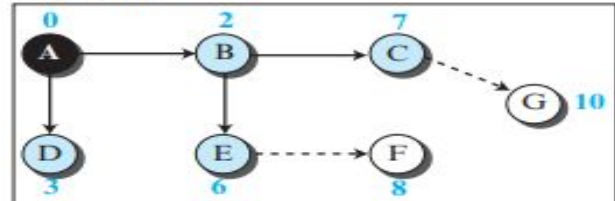
Iteration 2



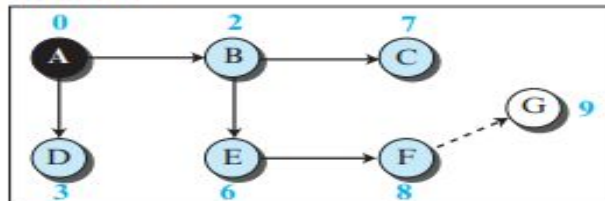
Iteration 3



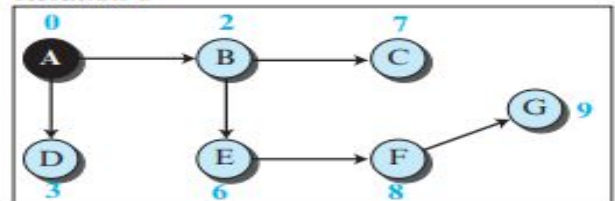
Iteration 4




Iteration 5



Iteration 6



Path-Vector Routing

- Both link-state and distance-vector routing are based on the least-cost goal.
 - However, there are instances where this goal is not the priority. For example, assume that there are some routers in the internet that a sender wants to prevent its packets from going through.
 - For example, a router may belong to an organization that does not provide enough security or that belongs to a commercial rival of the sender which might inspect the packet for obtaining information.
 - Least-cost routing does not prevent a packet from passing through an area when that area is in the least-cost path. In other words, the least-cost goal, applied by LS or DV routing, does not allow a sender to apply specific policies to the route a packet may take.
 - Aside from safety and security, there are occasions, in which the mere goal of routing is reachability: to allow the packet to reach its destination more efficiently without assigning costs to the route.
- 

Path-Vector Routing

- To respond to these demands, a third routing algorithm, called **path-vector (PV) routing** has been devised.
- Path-vector routing does not have the drawbacks of LS or DV routing as described above because it is not based on least-cost routing.
- The best route is determined by the source using the policy it imposes on the route.
- In other words, the source can control the path.
- Although path-vector routing is not actually used in an internet, and is mostly designed to route a packet between ISPs.

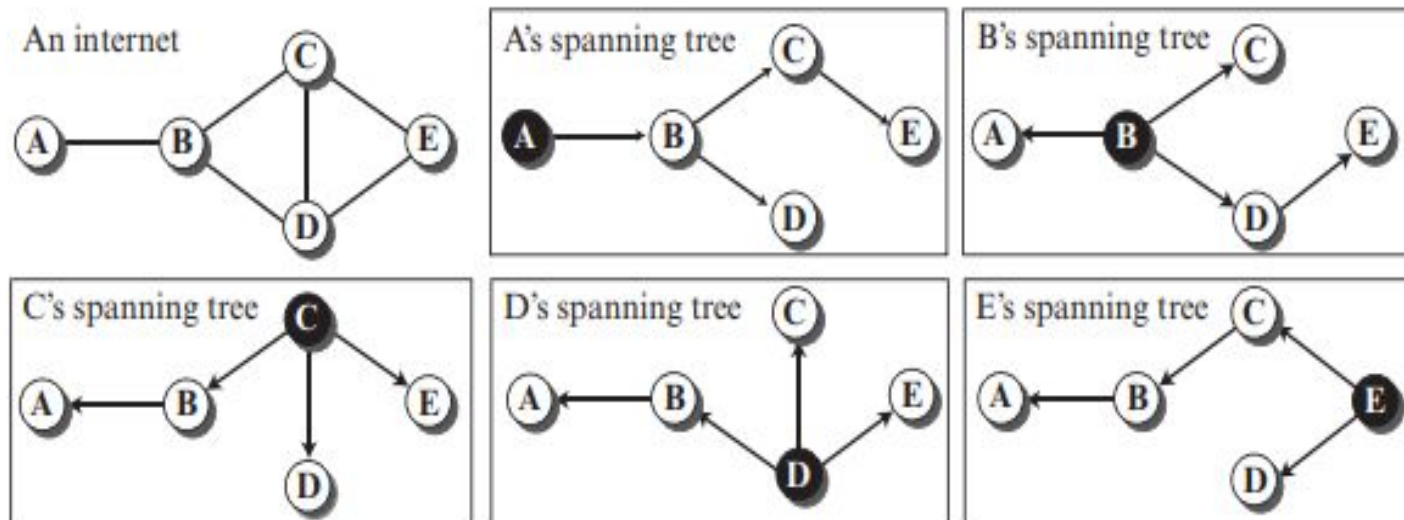


Path-Vector Routing


Spanning Trees

- In **path-vector routing**, the path from a source to all destinations is also determined by the best spanning tree.
- The best spanning tree, however, is not the least-cost tree; it is the tree determined by the source when it imposes its own policy.
- If there is more than one route to a destination, the source can choose the route that meets its policy best.
- A source may apply several policies at the same time.
- One of the common policies uses the minimum number of nodes to be visited (something similar to least-cost).
- Another common policy is to avoid some nodes as the middle node in a route.
- Figure 4.66 shows a small internet with only five nodes. Each source has created its own spanning tree that meets its policy.
- The policy imposed by all sources is to use the minimum number of nodes to reach a destination.
- The spanning tree selected by A and E is such that the communication does not pass through D as a middle node. Similarly, the spanning tree selected by B is such that the communication does not pass through C as a middle node.

Figure 4.66 *Spanning trees in path-vector routing*



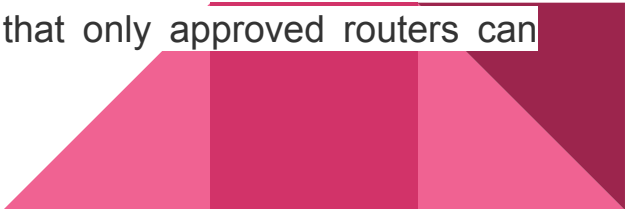
Routing Information Protocol (RIP)

1. Routing Information Protocol or RIP is one of the first routings protocols to be created.
 2. RIP is used in both Local Area Networks (LANs) and Wide Area Networks (WANs), and also runs on the Application layer of the OSI model.
 3. There are multiple versions of RIP including RIPv1 and RIPv2. The original version or RIPv1 determines network paths based on the IP destination and the hop count of the journey.
 4. RIPv1 interacts with the network by broadcasting its IP table to all routers connected to the network.
 5. RIPv2 is a little more sophisticated than this and sends its routing table onto a multicast address. RIPv2 also uses authentication to keep data more secure and chooses a subnet mask and gateway for future traffic.
 6. The main limitation of RIP is that it has a maximum hop count of 15 which makes it unsuitable for larger networks.
- 

Open Shortest Path First (OSPF)

1. Open Shortest Path First or OSPF protocol is a link-state IGP that was tailor-made for IP networks using the Shortest Path First (SPF) algorithm.
2. The SPF routing algorithm is used to calculate the shortest path spanning tree to ensure efficient data transmission of packets.
3. OSPF routers maintain databases detailing information about the surrounding topology of the network. This database is filled with data taken from Link State Advertisements (LSAs) sent by other routers.
4. LSAs are packets that detail information about how many resources a given path would take.
5. OSPF also uses the Dijkstra algorithm to recalculate network paths when the topology changes. This protocol is also relatively secure as it can authenticate protocol changes to keep data secure.
6. It is used by many organizations because it's scalable to large environments. Topology changes are tracked and OSPF can recalculate compromised packet routes if a previously-used route has been blocked.

Border Gateway Protocol (BGP)

1. Border Gateway Protocol or BGP is the routing protocol of the internet that is classified as a distance path vector protocol. BGP was designed to replace EGP with a decentralized approach to routing.
 2. The BGP Best Path Selection Algorithm is used to select the best routes for data packet transfers. If you don't have any custom settings then BGP will select routes with the shortest path to the destination.
 3. However many administrators choose to change routing decisions to criteria in line with their needs.
 4. The best routing path selection algorithm can be customized by changing the BGP cost community attribute. BGP can make routing decisions based on Factors such as weight, local preference, locally generated, AS_Path length, origin type, multi-exit discriminator, eBGP over iBGP, IGP metric, router ID, cluster list, and neighbor IP address.
 5. BGP only sends updated router table data when something changes. As a result, there is no auto-discovery of topology changes which means that the user has to configure BGP manually.
 6. In terms of security, the BGP protocol can be authenticated so that only approved routers can exchange data with each other.
- 

Next generation IP:

Packet Format ,

IPv6 Addressing ,

Transition from IPv4 to IPv6,

ICMPv6,



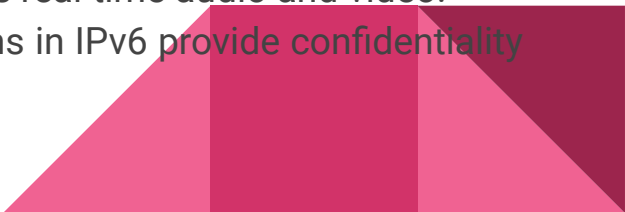
NEXT GENERATION IP

- The address depletion of IPv4 and other shortcomings of this protocol prompted a new version of IP protocol in the early 1990s.
- The new version, which is called Internet Protocol version 6 (IPv6) or IP new generation (IPng) was a proposal to augment the address space of IPv4 and at the same time redesign the format of the IP packet and revise some auxiliary protocols such as ICMP.



NEXT GENERATION IP

The following shows the main changes in the IPv6 protocol:

- **Larger address space.** An IPv6 address is 128 bits long. Compared with the 32-bit address of IPv4, this is a huge (296 times) increase in the address space.
 - **Better header format.** IPv6 uses a new header format in which options are separated from the base header and inserted, when needed, between the base header and the data. This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.
 - **New options.** IPv6 has new options to allow for additional functionalities.
 - **Allowance for extension.** IPv6 is designed to allow the extension of the protocol if required by new technologies or applications.
 - **Support for resource allocation.** In IPv6, the type-of-service field has been removed, but two new fields, traffic class and flow label, have been added to enable the source to request special handling of the packet. This mechanism can be used to support traffic such as real-time audio and video.
 - **Support for more security.** The encryption and authentication options in IPv6 provide confidentiality and integrity of the packet.
- 

Packet Format

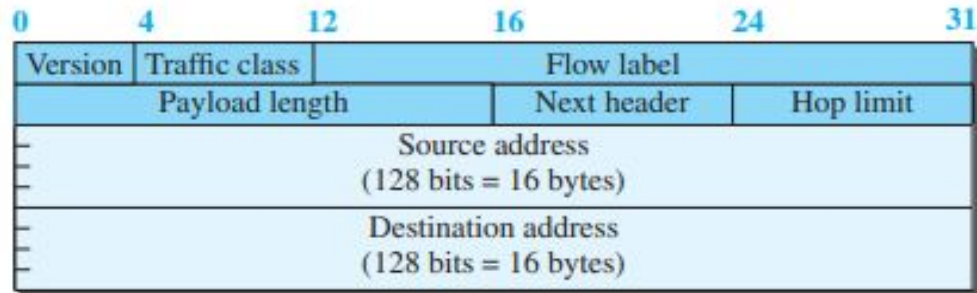
- Each packet is composed of a base header followed by the payload.
- The base header occupies 40 bytes, whereas payload can be up to 65,535 bytes of information.



Figure 4.101 *IPv6 datagram*



a. IPv6 packet



b. Base header

Packet Format

- **Version.** The 4-bit version field defines the version number of the IP. For IPv6, the value is 6.
- **Traffic class.** The 8-bit traffic class field is used to distinguish different payloads with different delivery requirements. It replaces the type-of-service field in IPv4.
- **Flow label.** The flow label is a 20-bit field that is designed to provide special handling for a particular flow of data. We will discuss this field later.
- **Payload length.** The 2-byte payload length field defines the length of the IP datagram excluding the header. Note that IPv4 defines two fields related to the length: header length and total length. In IPv6, the length of the base header is fixed (40 bytes); only the length of the payload needs to be defined.

Packet Format

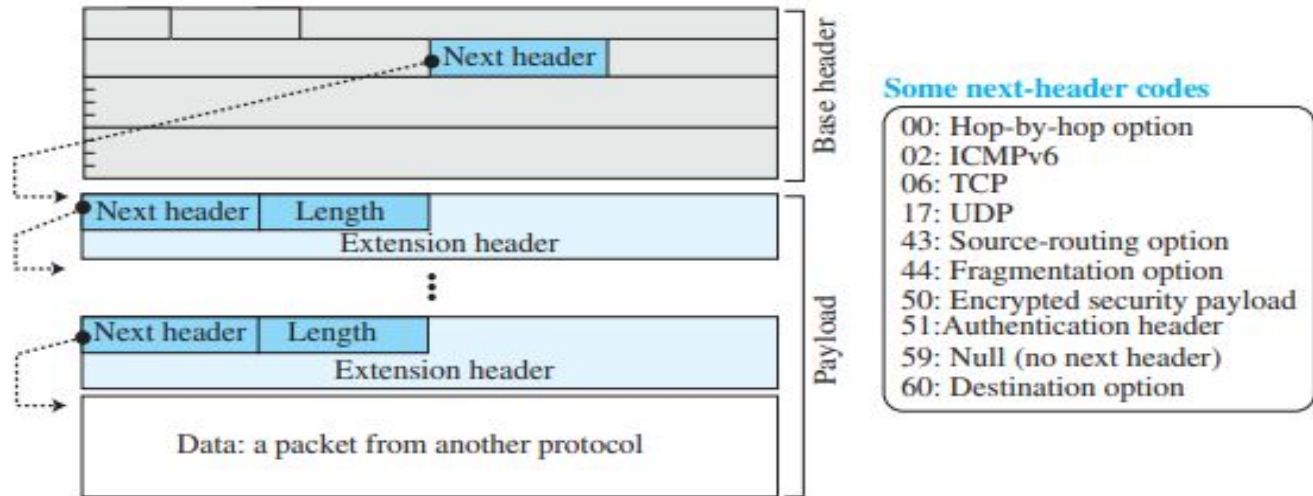
- **Next header.** The next header is an 8-bit field defining the type of first extension header (if present) or the type of the data that follows the base header in the datagram. This field is similar to the protocol field in IPv4, but we talk more about it when we discuss the payload.
- **Hop limit.** The 8-bit hop limit field serves the same purpose as the TTL field in IPv4.
- **Source and destination address.** The source address field is a 16-byte (128-bit) Internet address that identifies the original source of the datagram. The destination address field is a 16-byte (128-bit) Internet address that identifies the destination of the datagram.
-

Packet Format

- **Payload.** Compared to IPv4, the payload field in IPv6 has a different format and meaning, as shown in Figure 4.102.
- The payload in IPv6 means a combination of zero or more extension headers (options) followed by the data from other protocols (UDP, TCP, and so on).
- In IPv6, options, which are part of the header in IPv4, are designed as extension headers.
- The payload can have as many extension headers as required by the situation. Each extension header has two mandatory fields, next header and the length, followed by information related to the particular option.
- Note that each next header field value (code) defines the type of the next header (hop-by-hop option, sourcerouting option, . . .); the last next header field defines the protocol (UDP, TCP, . . .) that is carried by the datagram.

Packet Format

Figure 4.102 *Payload in an IPv6 datagram*



IPv6 Addressing

IPv6 was developed by Internet Engineering Task Force (IETF) to deal with the problem of IPv4 exhaustion. IPv6 is a 128-bits address having an address space of 2^{128} , which is way bigger than IPv4. IPv6 use Hexa-Decimal format separated by colon (:).

Components in Address format :

1. There are 8 groups and each group represents 2 Bytes (16-bits).
2. Each Hex-Digit is of 4 bits (1 nibble)
3. Delimiter used – colon (:)



IPv6 Addressing

In IPv6 representation, we have three addressing methods :

- Unicast
- Multicast
- Anycast



IPv6 Addressing

Addressing methods

1. *Unicast Address*

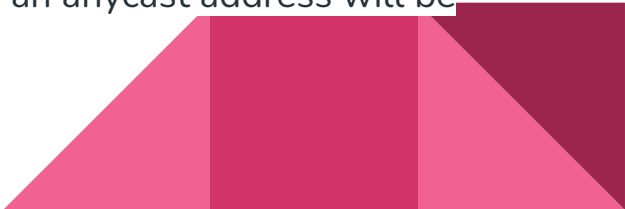
Unicast Address identifies a single network interface. A packet sent to a unicast address is delivered to the interface identified by that address.

2. *Multicast Address*


Multicast Address is used by multiple hosts, called as **groups**, to acquire a multicast destination address. These hosts need not be geographically together. If any packet is sent to this multicast address, it will be distributed to all interfaces corresponding to that multicast address. And every node is configured in the same way. In simple words, one data packet is sent to multiple destinations simultaneously.

3. *Anycast Address*

Anycast Address is assigned to a group of interfaces. Any packet sent to an anycast address will be delivered to only one member interface (mostly nearest host possible).



Transition from IPv4 to IPv6

- The first solution that comes to mind is to define a transition day on which every host or router should stop using the old version and start using the new version.
 - However, this is not practical; because of the huge number of systems on the Internet, the transition from IPv4 to IPv6 cannot happen suddenly.
 - It will take a considerable amount of time before every system in the Internet can move from IPv4 to IPv6.
 - The transition must be smooth to prevent any problems between IPv4 and IPv6 systems.
 - Three strategies have been devised by the IETF to help the transition: **dual stack, tunneling, and header translation.**
- 

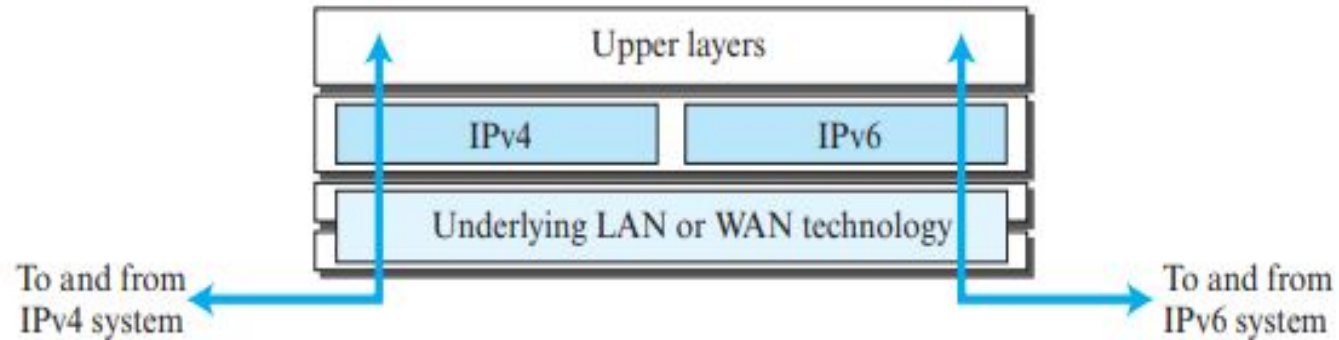
Dual Stack

- It is recommended that all hosts, before migrating completely to version 6, have a dual stack of protocols during the transition.
- In other words, a station must run IPv4 and IPv6 simultaneously until all the Internet uses IPv6. See Figure 4.106 for the layout of a dual-stack configuration.
- To determine which version to use when sending a packet to a destination, the source host queries the DNS.
- If the DNS returns an IPv4 address, the source host sends an IPv4 packet. If the DNS returns an IPv6 address, the source host sends an IPv6 packet



Dual Stack

Figure 4.106 *Dual stack*

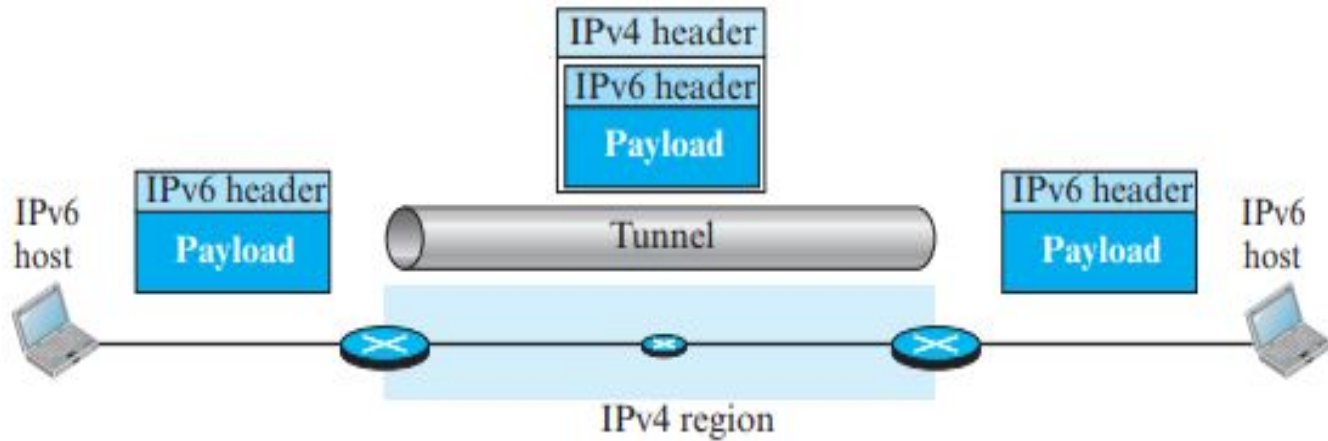


Tunneling

- Tunneling is a strategy used when two computers using IPv6 want to communicate with each other and the packet must pass through a region that uses IPv4.
- To pass through this region, the packet must have an IPv4 address. So the IPv6 packet is encapsulated in an IPv4 packet when it enters the region, and it leaves its capsule when it exits the region.
- It seems as if the IPv6 packet goes through a tunnel at one end and emerges at the other end.
- To make it clear that the IPv4 packet is carrying an IPv6 packet as data, the protocol value is set to 41.
- Tunneling is shown in Figure 4.107.



Figure 4.107 *Tunneling strategy*

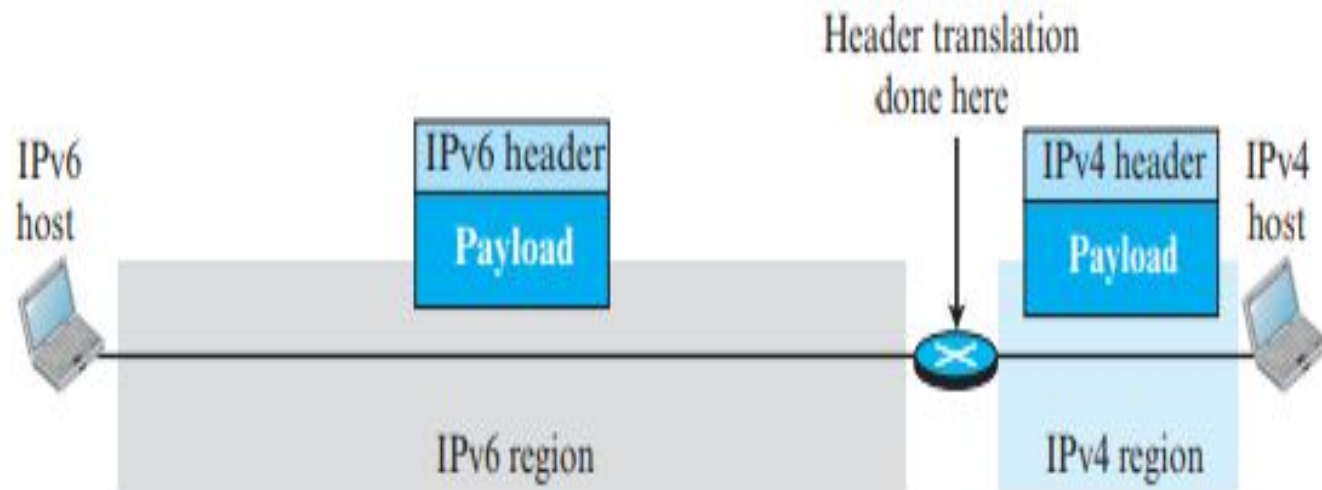


Header Translation

- Header translation is necessary when the majority of the Internet has moved to IPv6 but some systems still use IPv4.
- The sender wants to use IPv6, but the receiver does not understand IPv6.
- Tunneling does not work in this situation because the packet must be in the IPv4 format to be understood by the receiver.
- In this case, the header format must be totally changed through header translation.
- The header of the IPv6 packet is converted to an IPv4 header (see Figure 4.108



Figure 4.108 *Header translation strategy*



ICMPv6

Figure 4.109 *Comparison of network layer in version 4 and version 6*

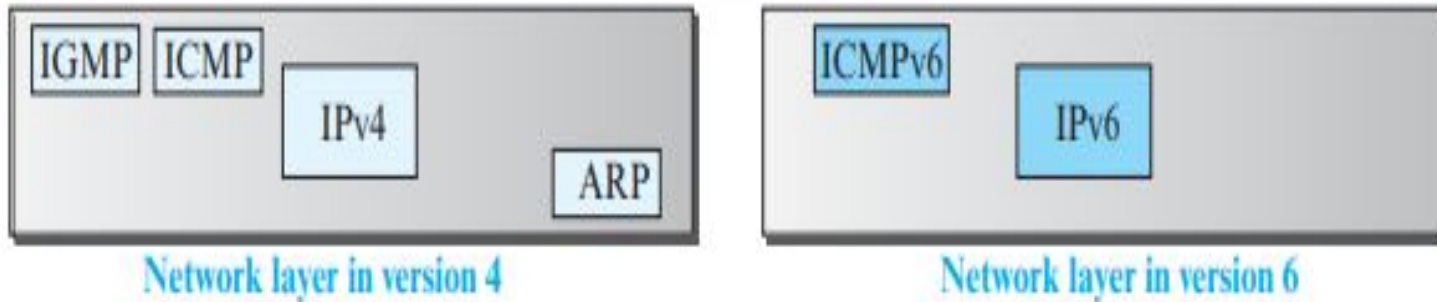
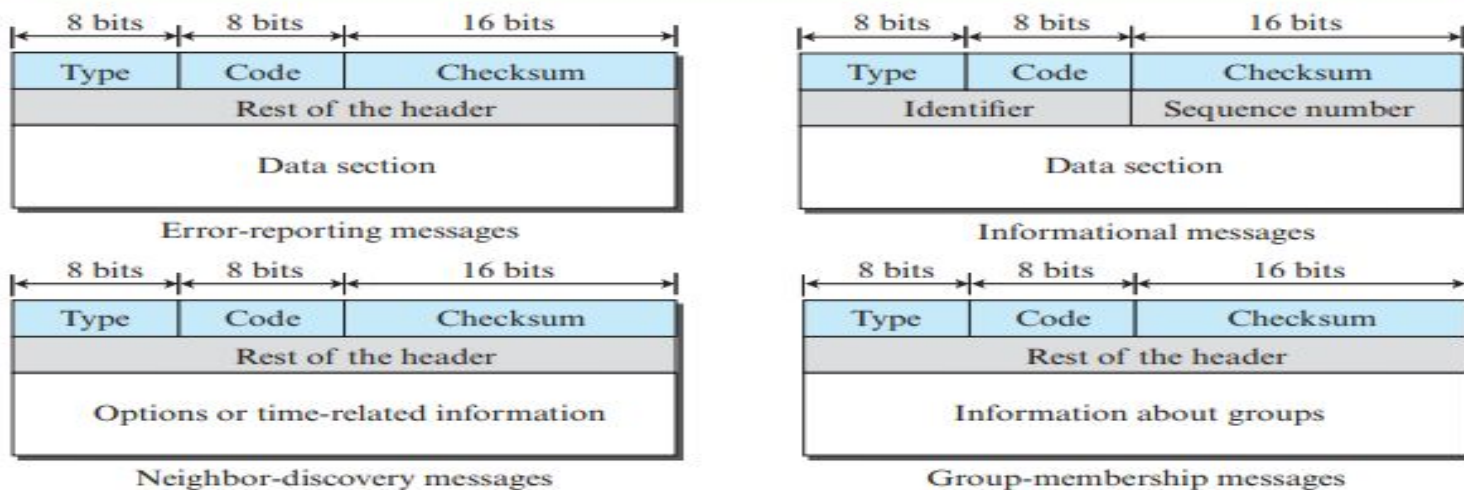


Figure 4.110 ICMPv6 messages



Type and code values

Error-reporting messages

- 01:** Destination unreachable (codes 0 to 6)
- 02:** Packet too big (code 0)
- 03:** Time exceeded (codes 0 and 1)
- 04:** Parameter problem (codes 0 to 2)

Neighbor-discovery messages

- 133 and 134:** Router solicitation and advertisement (code 0)
- 135 and 136:** Neighbor solicitation and advertisement (code 0)
- 137:** Redirection (codes 0 to 3)
- 141 and 142:** Inverse neighbor solicitation and advertisement (code 0)

Informational messages

- 128 and 129:** Echo request and reply (code 0)

Group-membership messages

- 130:** Membership query (code 0)
- 131:** Membership report

Note: See the book website for more explanation about messages.

Mobile IP:

Addressing ,

Agents ,

Three Phases ,

Inefficiency in Mobile IP.



MOBILE IP

- As mobile and personal computers such as notebooks become increasingly popular, we need to think about mobile IP, the extension of IP protocol that allows mobile computers to be connected to the Internet at any location where the connection is possible.



Addressing

- The main problem that must be solved in providing mobile communication using the IP protocol is addressing.


Stationary Hosts

- The original IP addressing was based on the assumption that a host is stationary, attached to one specific network.
- A router uses an IP address to route an IP datagram.
- An IP address has two parts: a prefix and a suffix.
- The prefix associates a host with a network. For example, the IP address 10.3.4.24/8 defines a host attached to the network 10.0.0.0/8.
- This implies that a host in the Internet does not have an address that it can carry with itself from one place to another.
- The address is valid only when the host is attached to the network. If the network changes, the address is no longer valid.
- Routers use this association to route a packet; they use the prefix to deliver the packet to the network to which the host is attached. This scheme works perfectly with stationary hosts

Mobile Hosts

When a host moves from one network to another, the IP addressing structure needs to be modified. Several solutions have been proposed.

Changing the Address

- One simple solution is to let the mobile host change its address as it goes to the new network.
 - The host can use DHCP to obtain a new address to associate it with the new network.
 - This approach has several drawbacks. First, the configuration files would need to be changed.
 - Second, each time the computer moves from one network to another, it must be rebooted.
 - Third, the DNS tables need to be revised so that every other host in the Internet is aware of the change.
 - Fourth, if the host roams from one network to another during a transmission, the data exchange will be interrupted.
 - This is because the ports and IP addresses of the client and the server must remain constant for the duration of the connection.
- 

Mobile Hosts

Two Addresses


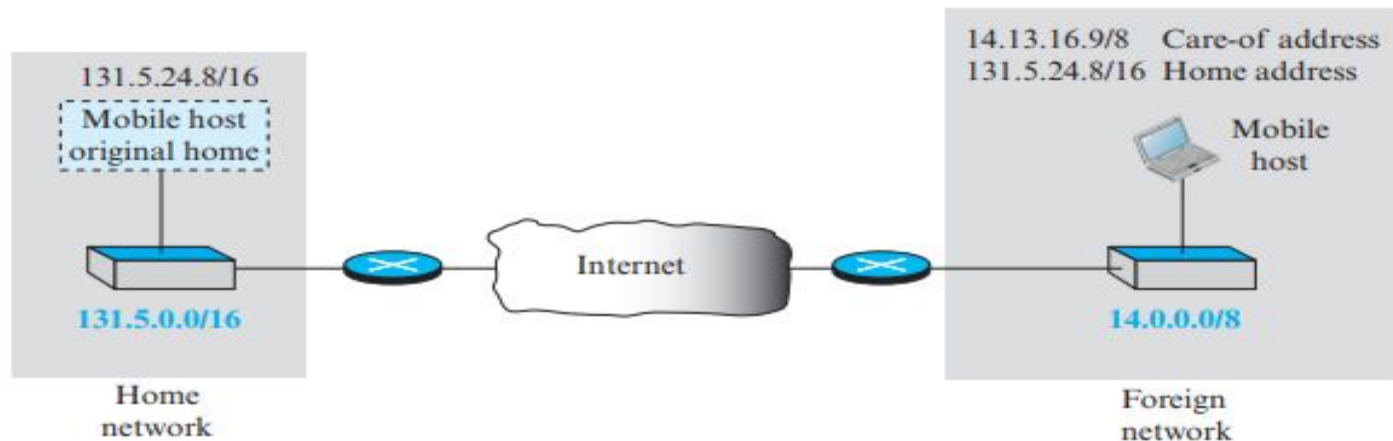
- The approach that is more feasible is the use of two addresses.
 - The host has its original address, called the home address, and a temporary address, called the care-of address.
 - The home address is permanent; it associates the host to its home network, the network that is the permanent home of the host.
 - The care-of address is temporary.
 - When a host moves from one network to another, the care-of address changes; it is associated with the foreign network, the network to which the host moves. Figure 6.52 shows the concept
- 

Figure 6.52 *Home address and care-of address*



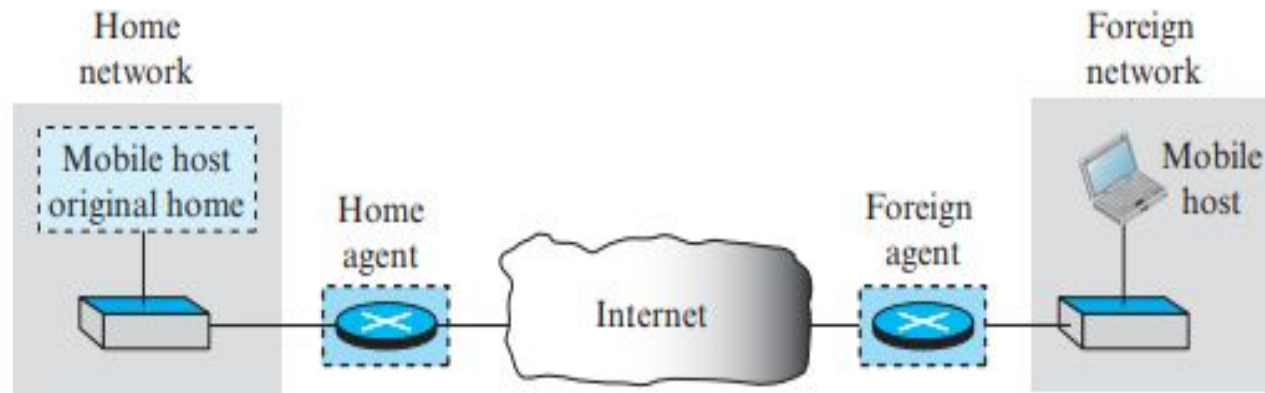
Mobile IP has two addresses for a mobile host: one home address and one care-of address. The home address is permanent; the care-of address changes as the mobile host moves from one network to another.

Agents

- To make the change of address transparent to the rest of the Internet requires a home agent and a foreign agent.
- Figure 6.53 shows the position of a home agent relative to the home network and a foreign agent relative to the foreign network.
- We have shown the home and the foreign agents as routers, but we need to emphasize that their specific function as an agent is performed in the application layer.
- In other words, they are both routers and hosts.



Figure 6.53 *Home agent and foreign agent*



Agents


Home Agent

- The home agent is usually a router attached to the home network of the mobile host.
- The home agent acts on behalf of the mobile host when a remote host sends a packet to the mobile host.
- The home agent receives the packet and sends it to the foreign agent.



Agents

Foreign Agent

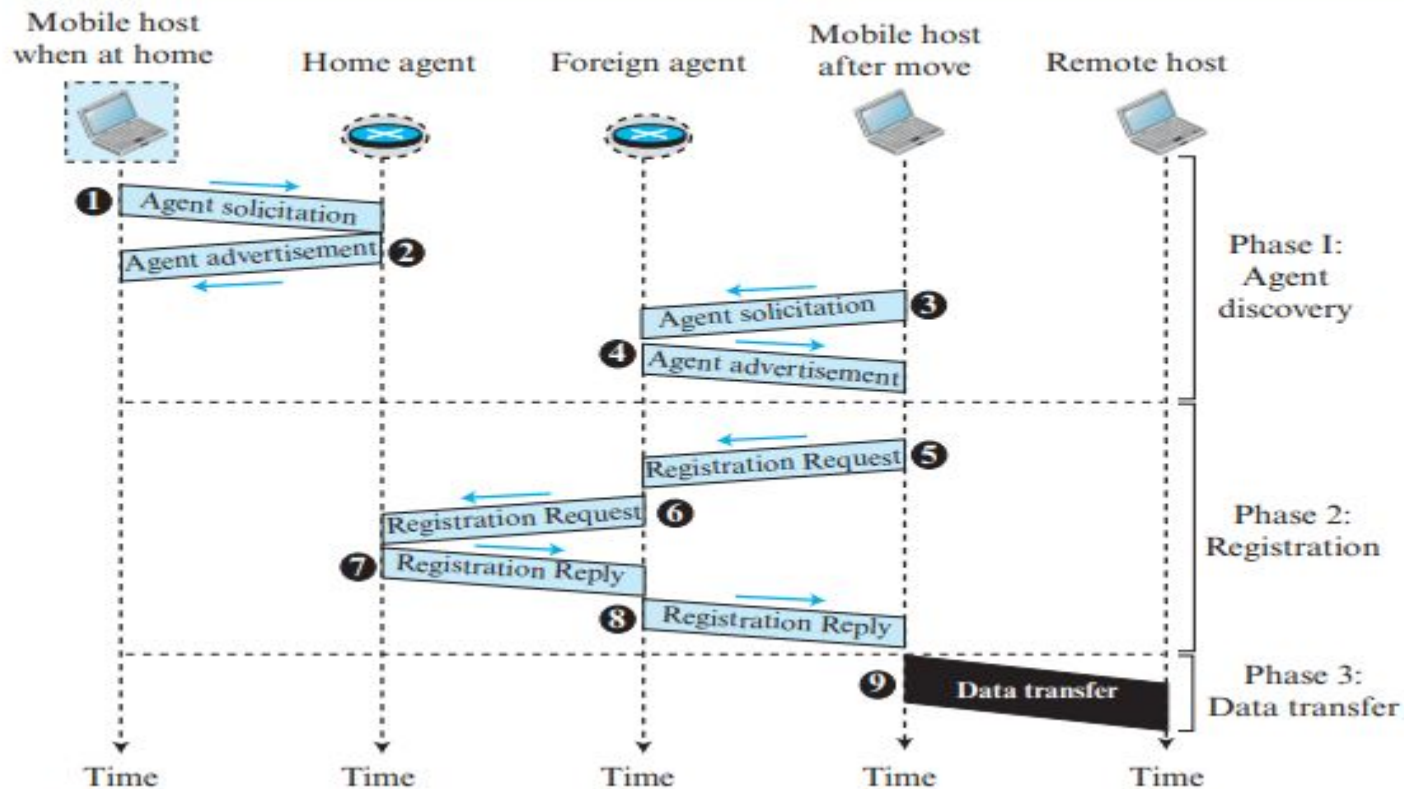
- The foreign agent is usually a router attached to the foreign network.
 - The foreign agent receives and delivers packets sent by the home agent to the mobile host.
 - The mobile host can also act as a foreign agent.
 - In other words, the mobile host and the foreign agent can be the same.
 - However, to do this, a mobile host must be able to receive a care-of address by itself, which can be done through the use of DHCP.
 - In addition, the mobile host needs the necessary software to allow it to communicate with the home agent and to have two addresses: its home address and its care-of address.
 - This dual addressing must be transparent to the application programs. When the mobile host acts as a foreign agent, the care-of address is called a collocated care-of address.
- 

Three Phases

- To communicate with a remote host, a mobile host goes through three phases: **agent discovery, registration, and data transfer**.
- The first phase, **agent discovery**, involves the mobile host, the foreign agent, and the home agent.
- The second phase, **registration**, also involves the mobile host and the two agents.
- Finally, in the third phase, **the remote host** is also involved. We discuss each phase separately.



Figure 6.54 *Remote host and mobile host communication*



Three Phases

Agent Discovery

- The first phase in mobile communication, agent discovery, consists of two subphases.
- A mobile host must discover (learn the address of) a home agent before it leaves its home network.
- A mobile host must also discover a foreign agent after it has moved to a foreign network.
- This discovery consists of learning the care-of address as well as the foreign agent's address. The discovery involves two types of messages: **advertisement and solicitation**.

Agent Advertisement

- When a router advertises its presence on a network using an ICMP router advertisement, it can append an agent advertisement to the packet if it acts as an agent.

Agent Solicitation

- When a mobile host has moved to a new network and has not received agent advertisements, it can initiate an agent solicitation. It can use the ICMP solicitation message to inform an agent that it needs assistance.

Three Phases

Registration

- The second phase in mobile communication is registration.
- After a mobile host has moved to a foreign network and discovered the foreign agent, it must register.
- There are four aspects of registration:
 - 1. The mobile host must register itself with the foreign agent.
 - 2. The mobile host must register itself with its home agent. This is normally done by the foreign agent on behalf of the mobile host.
 - 3. The mobile host must renew registration if it has expired.
 - 4. The mobile host must cancel its registration (deregistration) when it returns home.



Three Phases

Data Transfer

After agent discovery and registration, a mobile host can communicate with a remote host.

From Remote Host to Home Agent


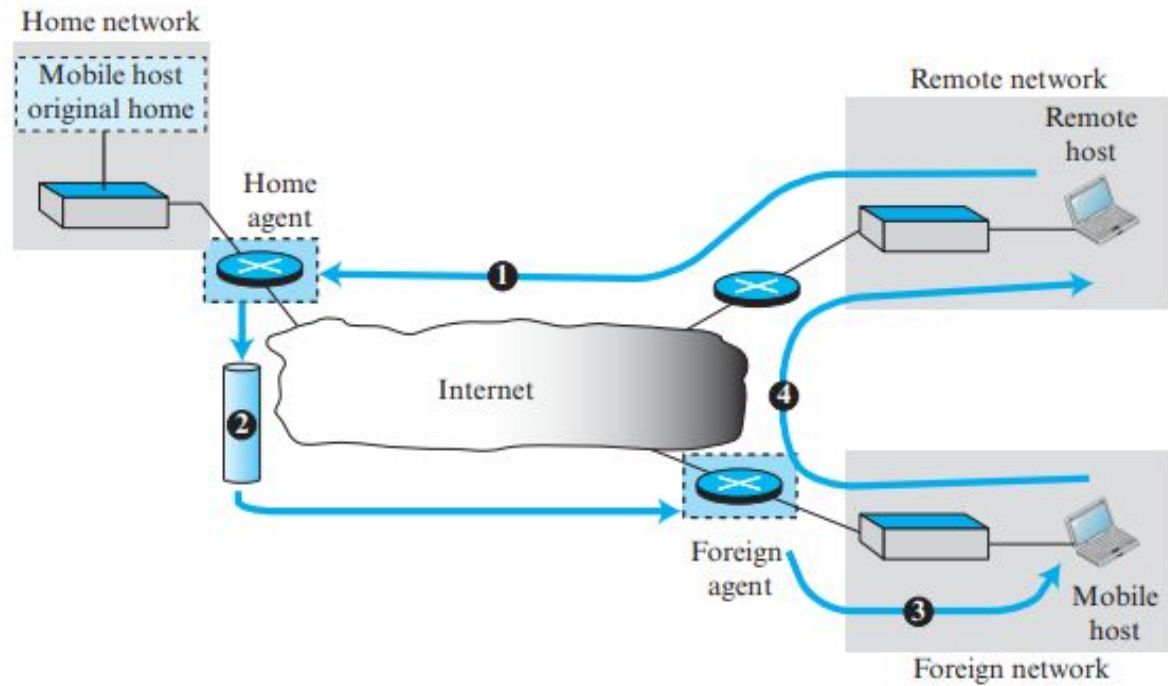
- When a remote host wants to send a packet to the mobile host, it uses its address as the source address and the home address of the mobile host as the destination address.
 - In other words, the remote host sends a packet as though the mobile host is at its home network. The packet, however, is intercepted by the home agent, which pretends it is the mobile host. This is done using the proxy ARP technique.
- 

Figure 6.58 *Data transfer*



Three Phases

From Home Agent to Foreign Agent

- After receiving the packet, the home agent sends the packet to the foreign agent, using the tunneling concept.
- The home agent encapsulates the whole IP packet inside another IP packet using its address as the source and the foreign agent's address as the destination. Path 2 of Figure 6.58 shows this step.

From Foreign Agent to Mobile Host

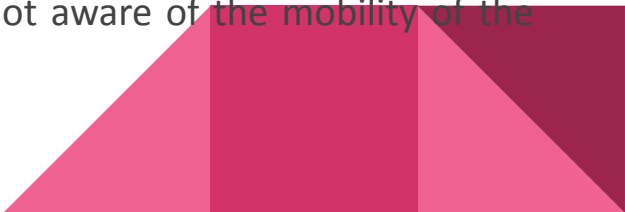
- When the foreign agent receives the packet, it removes the original packet.
- However, since the destination address is the home address of the mobile host, the foreign agent consults a registry table to find the care-of address of the mobile host.
- (Otherwise, the packet would just be sent back to the home network.) The packet is then sent to the care-of address. Path 3 of Figure 6.58 shows this step

Three Phases

From Mobile Host to Remote Host

- ❖ When a mobile host wants to send a packet to a remote host (for example, a response to the packet it has received), it sends as it does normally.
- ❖ The mobile host prepares a packet with its home address as the source, and the address of the remote host as the destination.
- ❖ Although the packet comes from the foreign network, it has the home address of the mobile host. Path 4 of Figure 6.58 shows this step.

Transparency

- ❖ In this data transfer process, the remote host is unaware of any movement by the mobile host.
 - ❖ The remote host sends packets using the home address of the mobile host as the destination address; it receives packets that have the home address of the mobile host as the source address.
 - ❖ The movement is totally transparent. The rest of the Internet is not aware of the mobility of the moving host.
- 

Inefficiency in Mobile IP

Communication involving mobile IP can be inefficient. The inefficiency can be severe or moderate. The severe case is called double crossing or 2X. The moderate case is called triangle routing or dog-leg routing.

Double Crossing


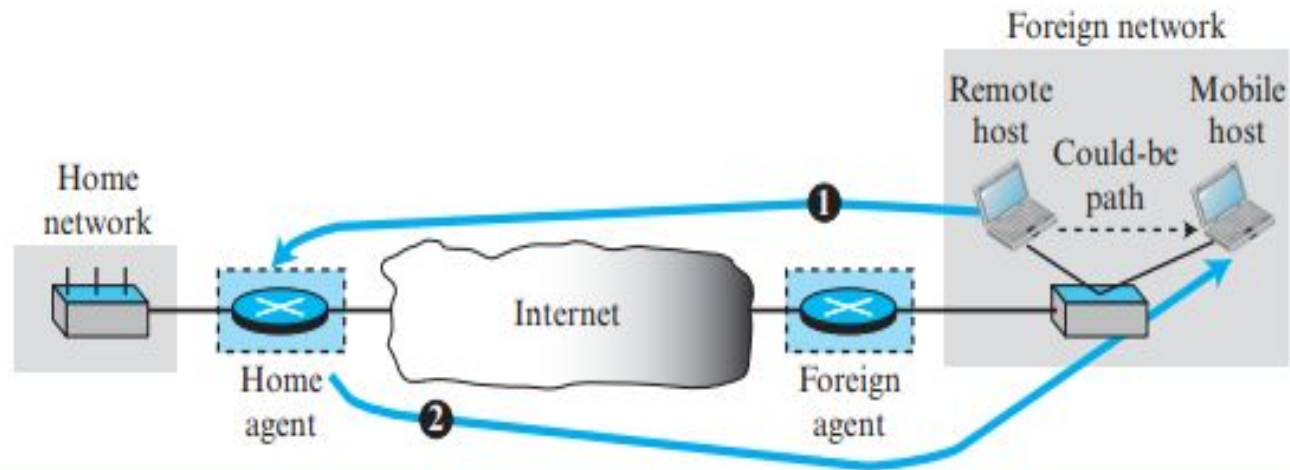
- Double crossing occurs when a remote host communicates with a mobile host that has moved to the same network (or site) as the remote host (see Figure 6.59).
 - When the mobile host sends a packet to the remote host, there is no inefficiency; the communication is local. However, when the remote host sends a packet to the mobile host, the packet crosses the Internet twice. Since a computer usually communicates with other local computers (principle of locality), the inefficiency from double crossing is significant.
- 

Figure 6.59 *Double crossing*



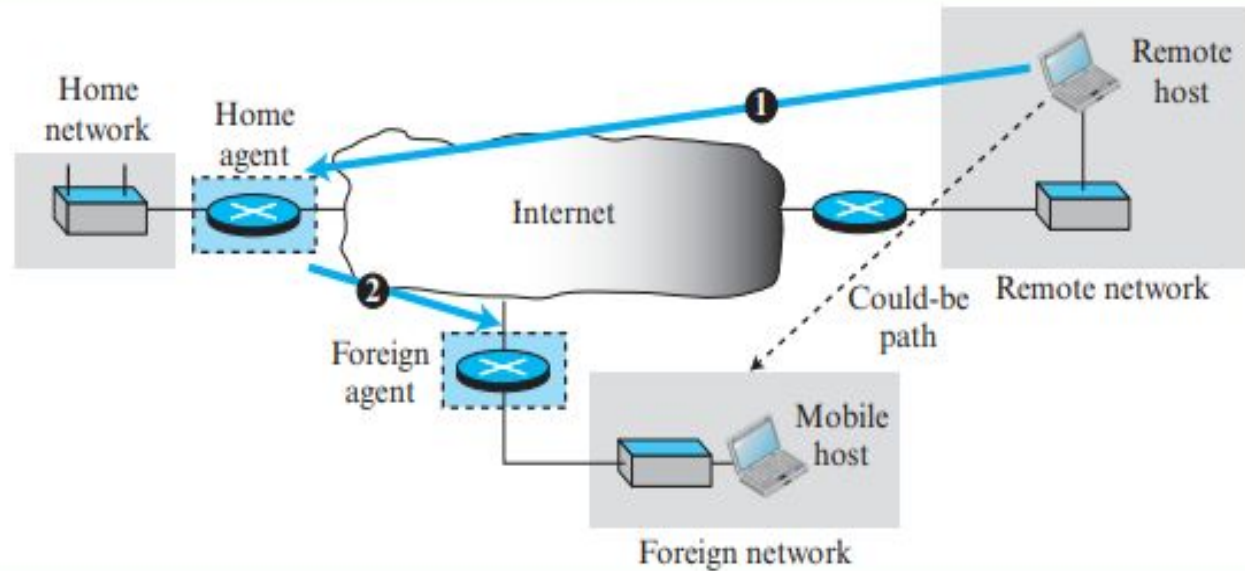
Inefficiency in Mobile IP

Triangle Routing

- Triangle routing, the less severe case, occurs when the remote host communicates with a mobile host that is not attached to the same network (or site) as the mobile host.
- When the mobile host sends a packet to the remote host, there is no inefficiency. However, when the remote host sends a packet to the mobile host, the packet goes from the remote host to the home agent and then to the mobile host. The packet travels the two sides of a triangle, instead of just one side (see Figure 6.60).



Figure 6.60 *Triangle routing*



Inefficiency in Mobile IP

Solution

- One solution to inefficiency is for the remote host to bind the care-of address to the home address of a mobile host.
 - For example, when a home agent receives the first packet for a mobile host, it forwards the packet to the foreign agent; it could also send an update binding packet to the remote host so that future packets to this host could be sent to the care-of address.
 - The remote host can keep this information in a cache.
 - The problem with this strategy is that the cache entry becomes outdated once the mobile host moves. In this case the home agent needs to send a warning packet to the remote host to inform it of the change
- 