Advanced Networking
Unit 2

-Madhavi Dave

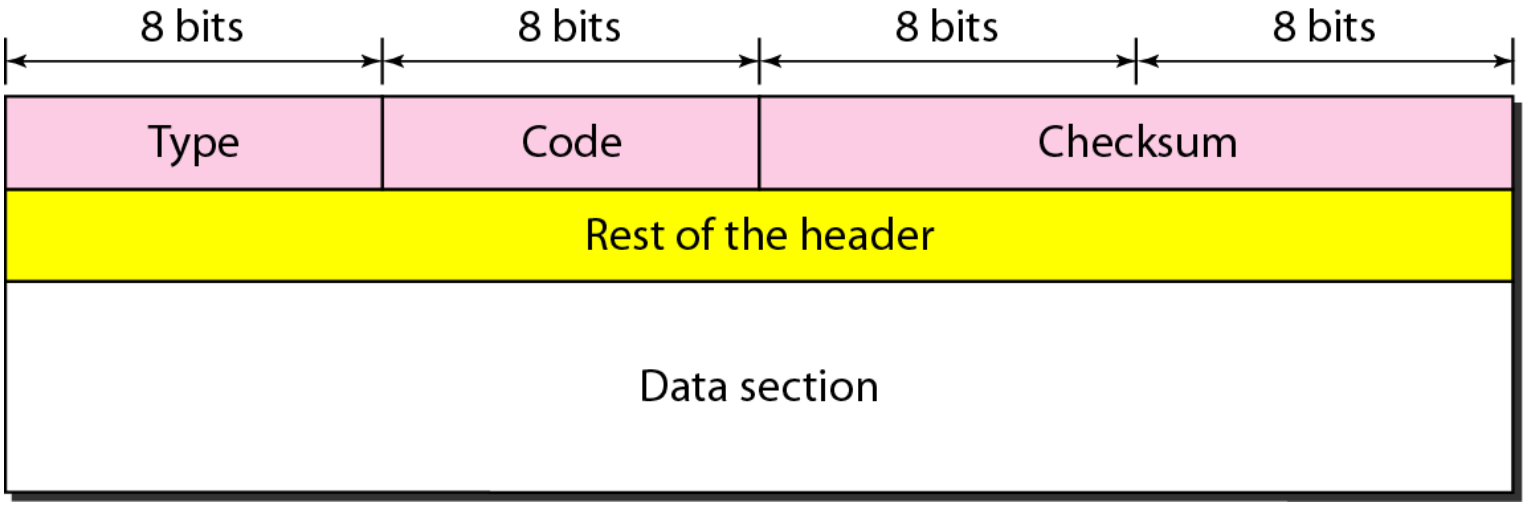
ICMP

- IP protocol has two deficiencies: lack of error control and lack of assistance mechanisms.
- The IP protocol has no error-reporting or error-correcting mechanism.
- What happens if something goes wrong? What happens if a router must discard a datagram because it cannot find a router to the final destination, or because the time-to-live field has a zero value? What happens if the final destination host must discard all fragments of a datagram because it has not received all fragments within a predetermined time limit?
- These are examples of situations where an error has occurred and the IP protocol has no built-in mechanism to notify the original host.

- The IP protocol also lacks a mechanism for host and management queries. A host sometimes needs to determine if a router or another host is alive. And sometimes a network administrator needs information from another host or router.
- The Internet Control Message Protocol (ICMP) has been designed to compensate for the above two deficiencies. It is a companion to the IP protocol.

- **Types of Messages**: ICMP messages are divided into two broad categories: error-reporting messages and query messages.
- The error-reporting messages report problems that a router or a host (destination) may encounter when it processes an IP packet.
- The query messages, which occur in pairs, help a host or a network manager get specific information from a router or another host

- For example, nodes can discover their neighbors. Also, hosts can discover and learn about routers on their network, and routers can help a node redirect its messages.
- **Message Format**: An ICMP message has an 8-byte header and a variable-size data section. Although the general format of the header is different for each message type, the first 4 bytes are common to all.

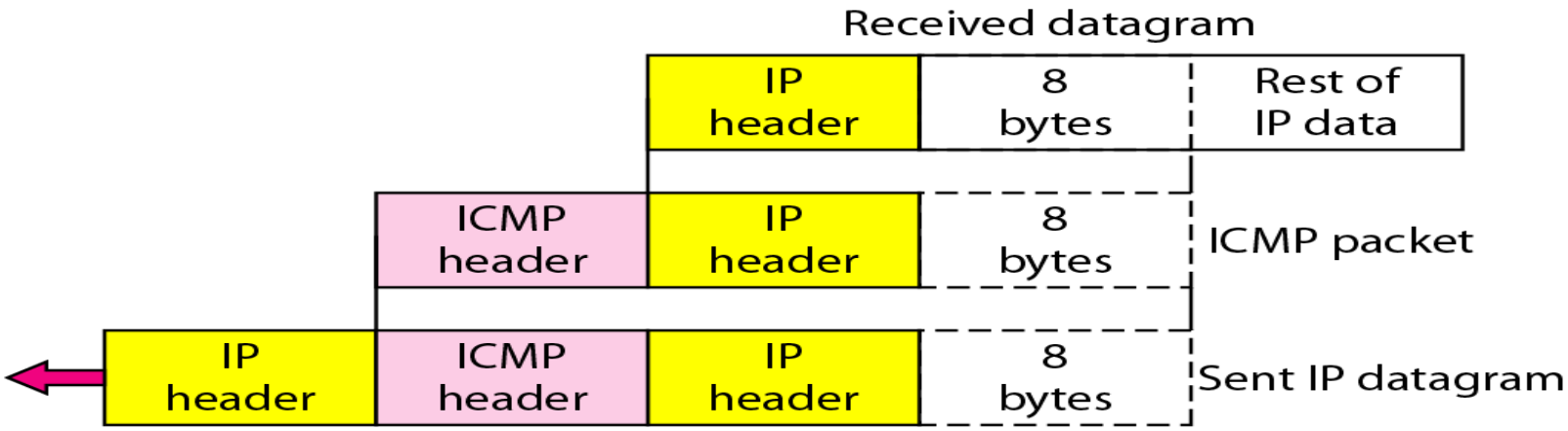


- ICMP type, defines the type of the message.
- The code field specifies the reason for the particular message type.
- The last common field is the checksum field.
- The rest of the header is specific for each message type.
- The data section in error messages carries information for finding the original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

Error Reporting

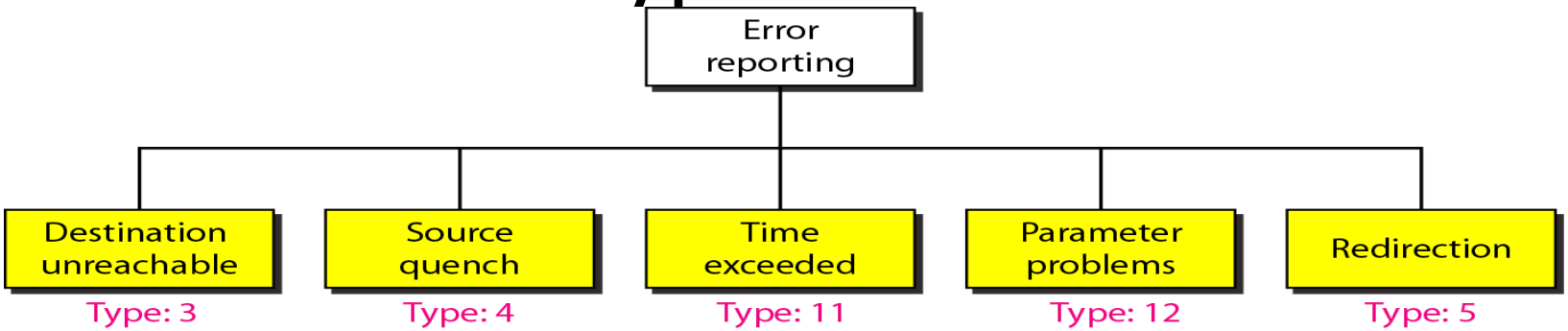
- One of the main responsibilities of ICMP is to report errors.
- ICMP does not correct errors-it simply reports them. Error correction is left to the higher-level protocols. Error messages are always sent to the original.
- **Five types of errors** are handled: destination unreachable, source quench, time exceeded, parameter problems, and redirection.
- The following are important points about ICMP error messages:
  - No ICMP error message will be generated in response to a datagram carrying an ICMP error message.
  - No ICMP error message will be generated for a fragmented datagram that is not the first fragment.
  - No ICMP error message will be generated for a datagram having a multicast address.
  - No ICMP error message will be generated for a datagram having a special address such as 127.0.0.0 or 0.0.0.0

- Contents of data field for the error messages:



- IP header is needed for source addresses and other information and first 8 bytes of data will be used to get port number sequence number to inform UDP and TCP protocol about error.

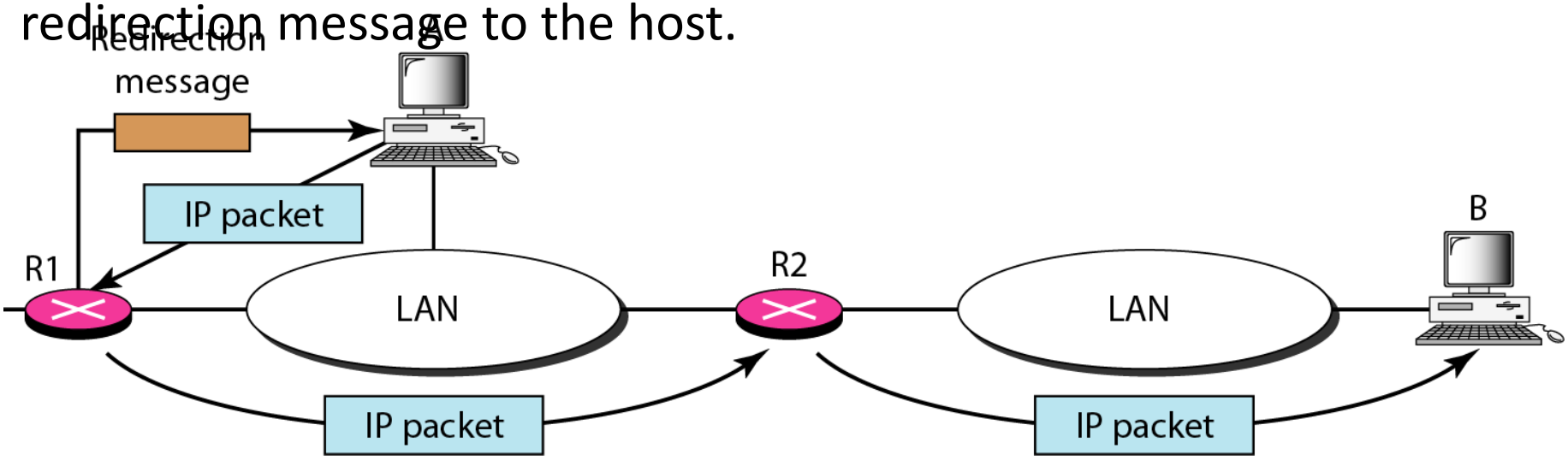
Types of Error



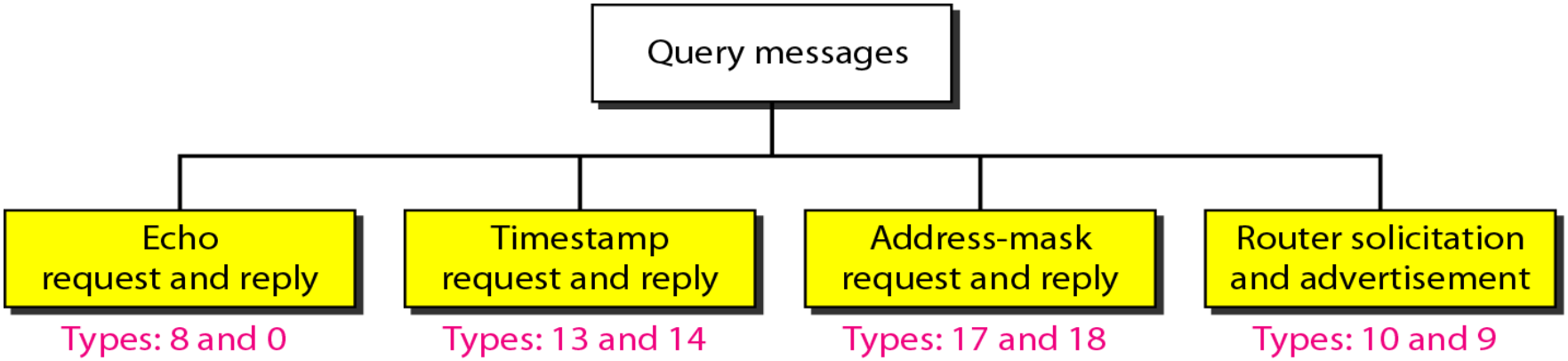
- ***Destination Unreachable:*** When a router cannot route a datagram or a host cannot deliver a datagram, the datagram is discarded and the router or the host sends a destination-unreachable message back to the source host that initiated the datagram. Messages can be created by either a router or the destination host.
- ***Source Quench:*** IP protocol is a connectionless protocol. The source host never knows if it is producing datagrams faster than can be forwarded by routers or processed by the destination host. When a router or host discards a datagram due to congestion, it sends a source-quench message to the sender of the datagram. This message has two purposes. First, it informs the source that the datagram has been discarded. Second, it warns the source that there is congestion somewhere in the path and that the source should slow down (quench) the sending process

- ***Time Exceeded:*** *This message is generated in two cases: 1)* If there are errors in one or more routing tables, a packet can travel in a loop or a cycle, going from one router to the next or visiting a series of routers endlessly. When the time-to-live value reaches 0, after decrementing, the router discards the datagram and send this message to source.2) a time-exceeded message is also generated when not all fragments that make up a message arrive at the destination host within a certain time limit.
- ***Parameter Problem:*** If a router or the destination host discovers an ambiguous or missing value in any field of the datagram, it discards the datagram and sends a parameter-problem message back to the source.

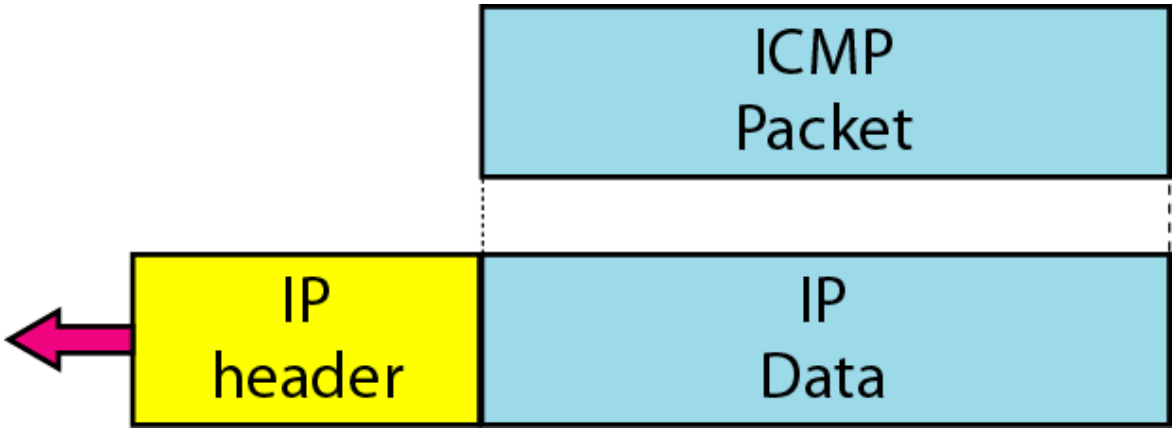
Redirection: For efficiency, hosts do not take part in the routing update process. The host usually knows the IP address of only one router, the default router. The host may send a datagram, which is destined for another network, to the wrong router. In this case, the router that receives the datagram will forward the datagram to the correct router and to update the routing table of the host, it sends a redirection message to the host.



Query Messages



Encapsulation of ICMP query messages



- ICMP can diagnose some network problems . For that a node sends a message that is answered in a specific format by the destination node.
- ***Echo Request and Reply:*** Network managers and users utilize this pair of messages to identify network problems. The combination of echo-request and echo-reply messages determines whether two systems (hosts or routers) can communicate with each other.

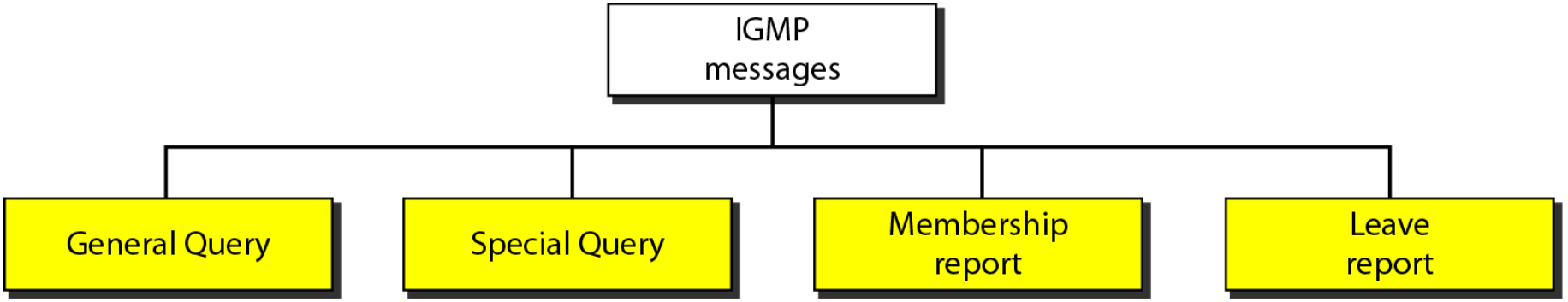
- ***Timestamp Request and Reply:*** Two machines (hosts or routers) can use the timestamp request and timestamp reply messages to determine the round-trip time needed for an IP datagram to travel between them. It can also be used to synchronize the clocks in two machines.

- ***Address-Mask Request and Reply:*** A host may know its IP address, but it may not know the corresponding mask. To obtain its mask, a host sends an address-mask-request message to a router on the LAN. If the host knows the address of the router, it sends the request directly to the router. If it does not know, it broadcasts the message. The router receiving the address-mask-request message responds with an address-mask-reply message, providing the necessary mask for the host.
- ***Router Solicitation and Advertisement:*** A host can broadcast (or multicast) a router-solicitation message to know the information about near by routers. The router or routers that receive the solicitation message broadcast their routing information using the router-advertisement message. A router can also periodically send router-advertisement messages even if no host has solicited. When a router sends out an advertisement, it announces not only its own presence but also the presence of all routers on the network of which it is aware.

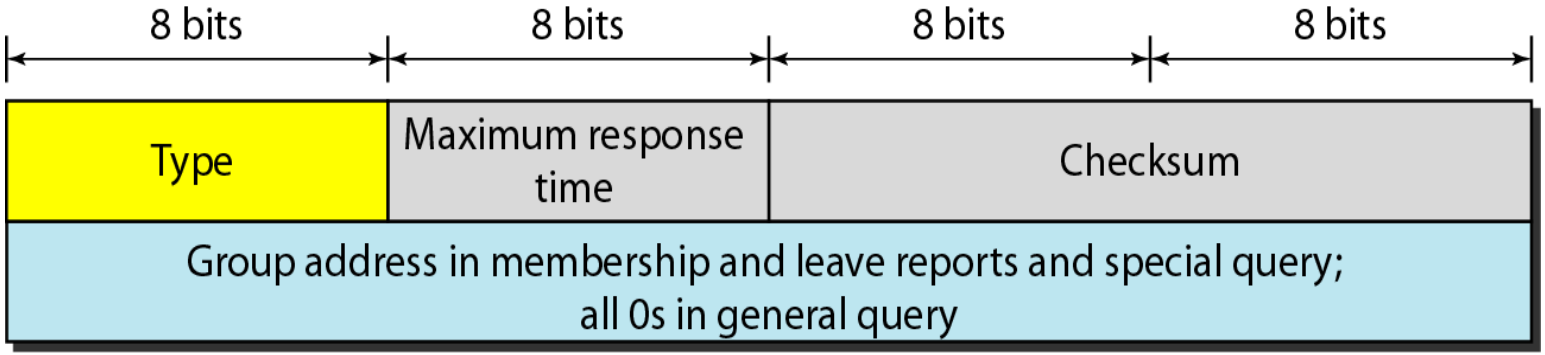
IGMP

- Some processes sometimes need to send the same message to a large number of receivers simultaneously.
- This is called multicasting, which is a one-to-many communication. Multicasting has many applications.
- For example, multiple stockbrokers can simultaneously be informed of changes in a stock price, or travel agents can be informed of a plane cancellation. Some other applications include distance learning and video-on-demand.
- The Internet Group Management Protocol (IGMP) is one of the necessary, protocols that is involved in multicasting.
- **Group Management:** IGMP is a group management protocol. It helps a multicast router create and update a list of loyal members related to each router interface

IGMP Messages



IGMP message format



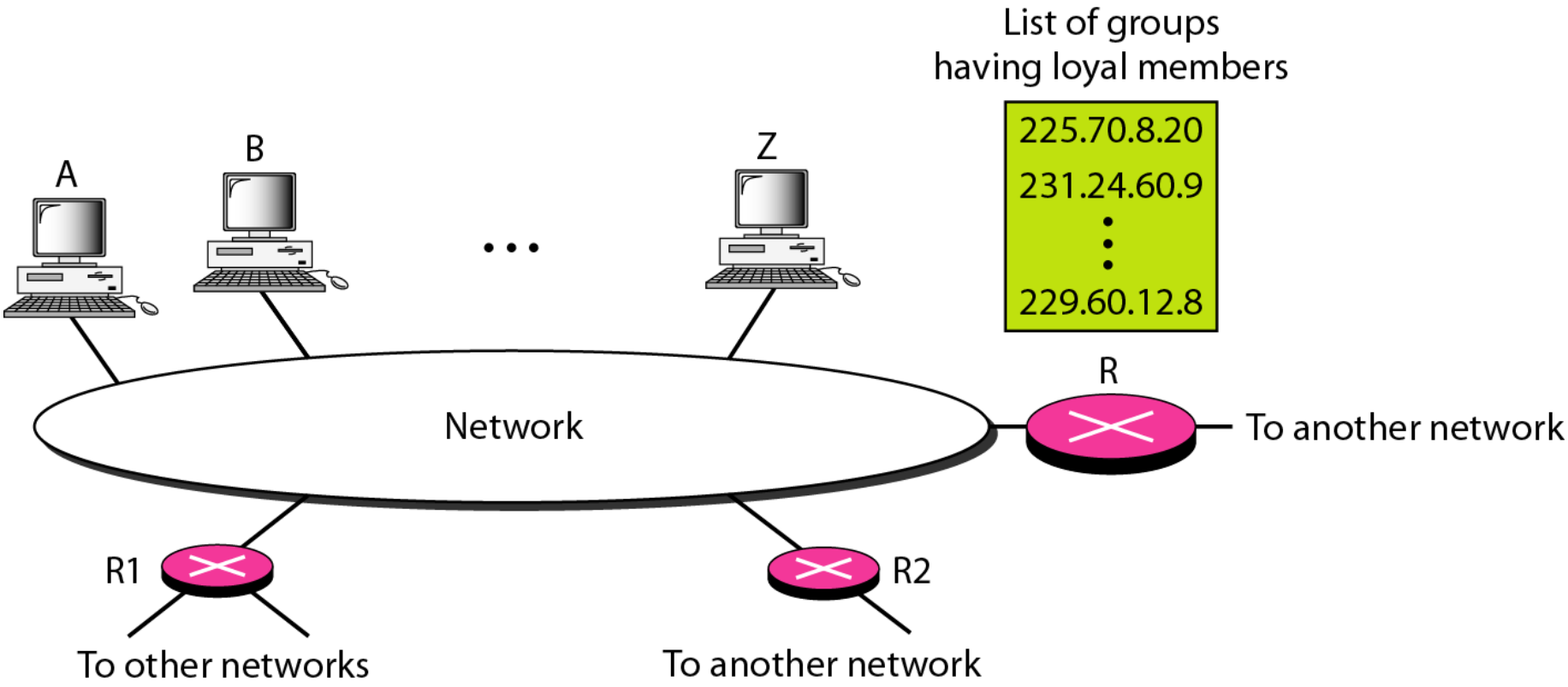
Type	*Value*
General or special query	0x11 or 00010001
Membership report	0x16 or 00010110
Leave report	0x17 or 00010111

- **IGMP type field:**
- **Maximum Response Time**. This 8-bit field defines the amount of time in which a query must be answered. The value is in tenths of a second.
- Checksum. This is a 16-bit field carrying the checksum. The checksum is calculated over the 8-byte message.
- Group address. The value of this field is 0 for a general query message. The value defines the groupid (multicast address of the group) in the special query, the membership report, and the leave report messages.

IGMP Operation

- A multicast router connected to a network has a list of multicast addresses of the groups with at least one loyal member in that network.
- For each group, there is one router that has the duty of distributing the multicast packets destined for that group.
- A host or multicast router can have membership in a group. When a host has membership, it means that one of its processes (an application program) receives multicast packets from some group. When a router has membership, it means that a network connected to one of its other interfaces receives these multicast packets.

- ***Joining a Group****:* A host or a router can join a group. A host maintains a list of processes that have membership in a group. When a process wants to join a new group, it sends its request to the host. The host adds the name of the process and the name of the requested group to its list. If this is the first entry for this particular group, the host sends a membership report message.
- The protocol requires that the membership report be sent twice, one after the other within a few moments. In this way, if the first one is lost or damaged, the second one replaces it.

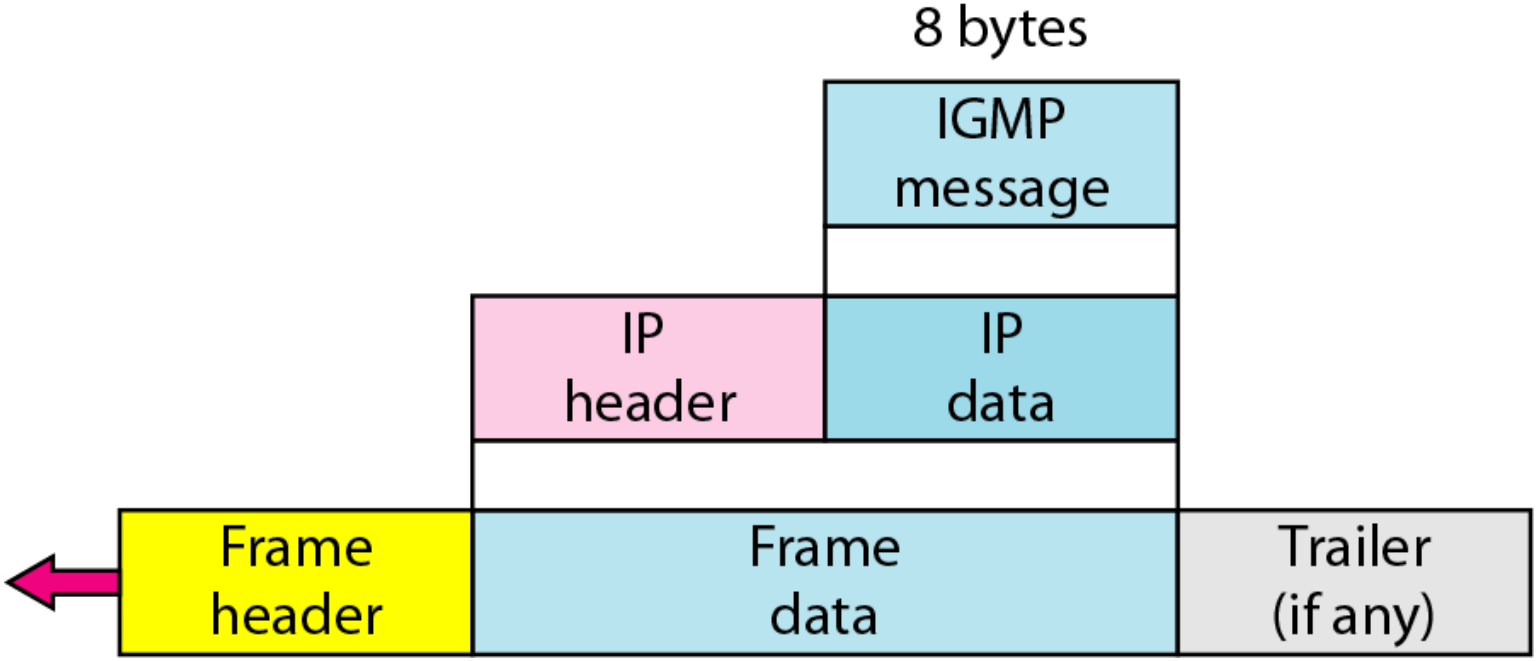

List of groups
having loyal members
225.70.8.20
231.24.60.9
229.60.12.8
A
B
Z
R
Network
To another network
R1
R2
To other networks
To another network

- ***Leaving a Group:*** When a host sees that no process is interested in a specific group, it sends a leave report. when a multicast router receives a leave report, it cannot immediately purge that group from its list because the report comes from just one host or router; there may be other hosts or routers that are still interested in that group.
- To make sure, the router sends a special query message and inserts the groupid, or multicast address, related to the group. The router allows a specified time for any host or router to respond. If, during this time, no interest (membership report) is received, the router assumes that there are no loyal members in the network and purges the group from its list.

- ***Monitoring Membership:*** The router periodically (by default, every 125 s) sends a general query message. In this message, the group address field is set to 0.0.0.0. This means the query for membership continuation is for all groups in which a host is involved, not just one.
- The router expects an answer for each group in its group list; even new groups may respond. The query message has a maximum response time of 10 s. When a host or router receives the general query message, it responds with a membership report if it is interested in a group. However, if there is a common interest (two hosts, for example, are interested in the same group), only one response is sent for that group to prevent unnecessary traffic. This is called a delayed response.

- ***Delayed Response:*** To prevent unnecessary traffic, IGMP uses a delayed response strategy. When a host or router receives a query message, it does not respond immediately; it delays the response. Each host or router uses a random number to create a timer, which expires between 1 and 10s.
- **Query Router:** Query messages may create a lot of responses. To prevent unnecessary traffic, IGMP designates one router as the query router for each network. Only this designated router sends the query message, and the other routers are passive.

- Encapsulation



Netstat Utility: The netstat utility can be used to find the multicast addresses supported by an interface. We use *netstat with three options: -n, -r, and -a. The -n option gives the numeric versions of IP* addresses, the -r option gives the routing table, and the -a option gives all addresses

IP Routing

Introduction

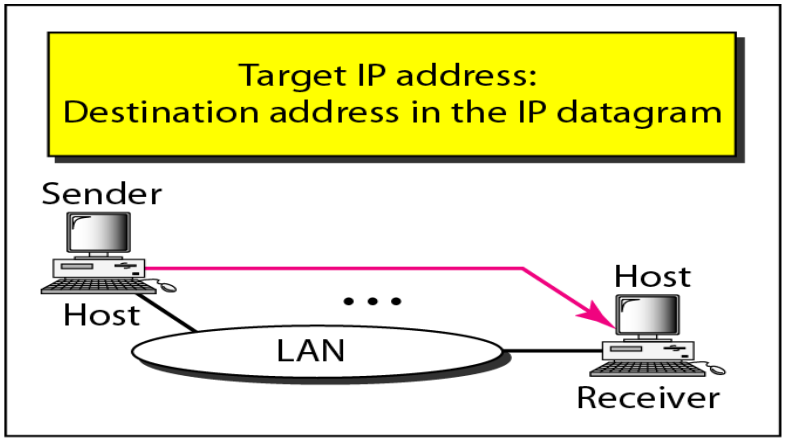
- In a packet switching system, ***routing*** refers to the process of choosing a path over which to send packets, and ***router*** refers to a computer making the choice.
- Routing occurs at several levels.
- ***IP forwarding,*** which is also called ***internet routing***
- IP routing chooses a path over which a datagram should be sent.
- Unlike routing within a single network, the **IP** routing algorithm must choose how to send a datagram across multiple physical networks.

Direct Delivery

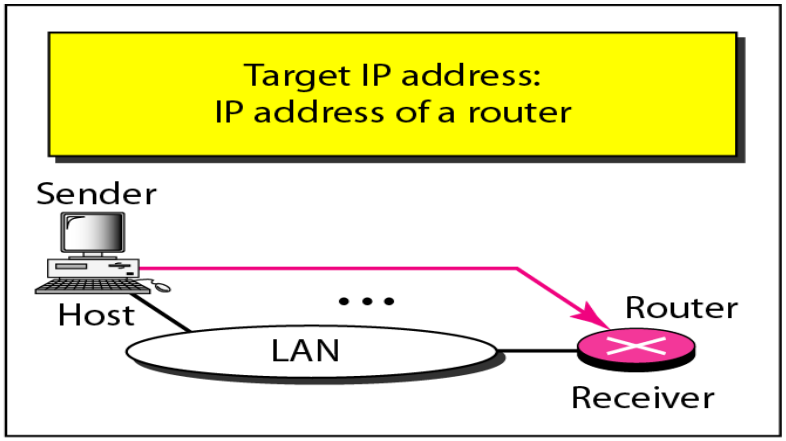
- Direct delivery, the transmission of a datagram from one machine across a single physical network directly to another, is the basis on which all internet communication rests.
- Two machines can engage in direct delivery only if they both attach directly to the same underlying physical transmission system

Indirect Delivery

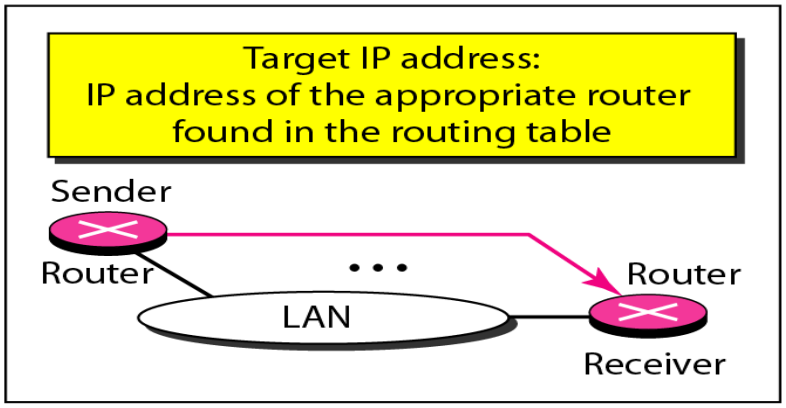
- ***Indirect delivery*** occurs when the destination is not on a directly attached network, forcing the sender to pass the datagram to a router for delivery.
- Sender must identify a router to which the datagram can be sent.
- The router must then forward the datagram toward its destination network.



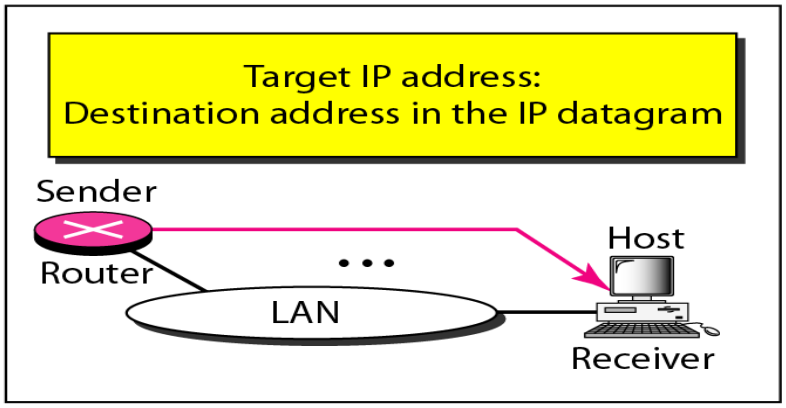
Case 1. A host has a packet to send to another host on the same network.



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



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



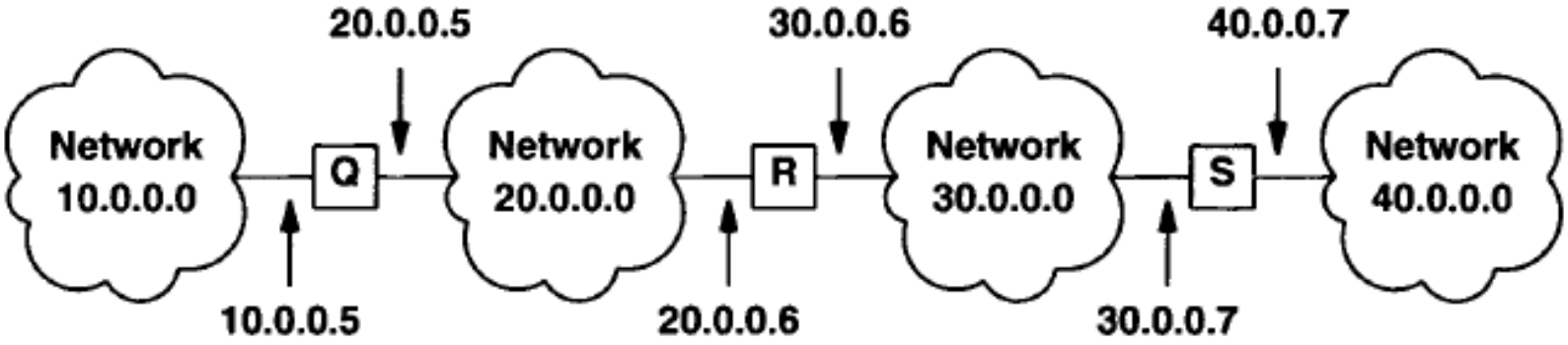
Case 4. A router receives a packet to be sent to a host on the same network.

Table driven IP Routing

- IP routing algorithm employs an ***Internet routing table*** (***IP routing*** table) on each machine that stores information about possible destinations and how to reach them.
- Because both hosts and routers route datagrams, both have IP routing tables.
- Whenever the IP routing software in a host or router needs to transmit a datagram, it consults the routing table to decide where to send the datagram.

Next-Hop Routing

- Using the network portion of a destination address instead of the complete host address makes routing efficient and keeps routing tables small.
- More important, it helps hide information, keeping the details of specific hosts confined to the local environment in which those hosts operate.
- Typically, a routing table contains pairs (N, ***R),*** where N is the **IP** address of a destination ***network,*** and R is the IP address of the "next" router along the path to network N.
- The routing table in a router ***R*** only specifies one step along the path from R to a destination network - the router does not know the complete path to a destination.



(a)

TO REACH HOSTS ON NETWORK	ROUTE TO THIS ADDRESS
20.0.0.0	DELIVER DIRECTLY
30.0.0.0	DELIVER DIRECTLY
10.0.0.0	20.0.0.5
40.0.0.0	30.0.0.7

(b)

Figure 82 (a) **An example internet with 4 networks and 3 routers, and (b) the routing table in R.**

Default Routing

- Another technique used to hide information and keep routing table sizes small consolidates multiple entries into a default case.
- The idea is to have the IP routing software first look in the routing table for the destination network.
- If no route appears in the table, the routing routines send the datagram to a ***default router.***
- Default routing is especially useful when a site has a small set of local addresses and only one connection to the rest of the internet.

Host-Specific Routing

- Mostly routing is based on networks and not on individual host, but most IP routing software allows per-host routes to be specified as a special case.
- Having per-host routes gives the local network administrator more control over network use, permits testing, and can also be used to control access for security purposes.
- When debugging network connections or routing tables, the ability to specify a special route to one individual machine turns out to be especially useful.

IP Routing Algorithm

RouteDatagram (Datagram, **RoutingTable**)

Extract destination **IP** address, D, from the datagram
    and compute the network prefix, N;
if N matches any directly connected network address
    deliver datagram to destination D over that network
    (This involves resolving D to a physical address,
    encapsulating the datagram, and sending the frame.)
else if the table contains a host-specific route for D
    send datagram to next-hop specified in table
else if the table contains a route for network N
    send datagram to next-hop specified in table
else if the table contains a default route
    send datagram to the default router specified in table
else declare a routing error;

Handling Incoming Datagram

- When an IP datagram arrives at a host, the network interface software delivers it to the IP module for processing.
- If the datagram's destination address matches the host's IP address, **IP** software on the host accepts the datagram and passes it to the appropriate higher-level protocol software for further processing.
- If the destination IP address does not match, a host is required to discard the datagram

Handling Incoming Datagram

- Unlike hosts, routers perform forwarding. When an IP datagram arrives at a router, it is delivered to the IP software.
- Again, two cases arise: the datagram could have reached its final destination, or it may need to travel further.
- A machine must also accept datagrams that were broadcast on the physical network if their destination IP address is the limited IP broadcast address or the directed broadcast address for that network.
- Routers also propagate routing information to ensure that their routing tables are consistent.

UDP

Introduction

- The operating systems in most computers support multiprogramming, which means they permit multiple application programs to execute simultaneously.
- we refer to each executing program as a ***process, task, application program***
- processes are created and destroyed dynamically

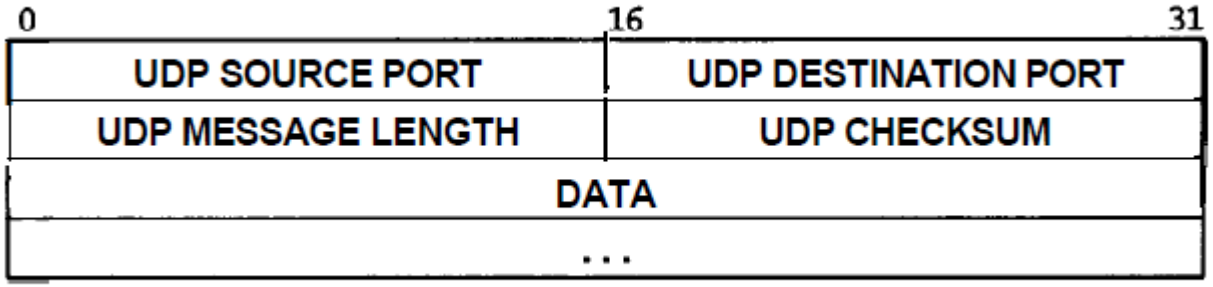
UDP

- ***User Datagram Protocol*** or ***UDP*** provides the primary mechanism that application programs use to send datagrams to other application programs.
- UDP provides protocol ports used to distinguish among multiple programs executing on a single machine.
- each UDP message contains both a destination port number and a source port number, making it possible for the UDP software at the destination to deliver the message to the correct recipient and for the recipient to send a reply.

UDP

- UDP uses the underlying Internet Protocol to transport a message from one machine to another, and provides the same unreliable, connectionless datagram delivery semantics as IP.
- It does not use acknowledgements to make sure messages arrive, it does not order incoming messages, and it does not provide feedback to control the rate at which information flows between the machines. Thus, UDP messages can be lost, duplicated, or arrive out of order.
- Furthermore, packets can arrive faster than the recipient can process them.

UDP Format



- user datagram consists of two parts: a UDP header and a UDP data area.

UDP Format

- The ***SOURCE PORT*** and ***DESTINATION PORT*** fields contain the 16-bit UDP protocol port numbers used to de-multiplex datagrams among the processes waiting to receive them.
- The ***SOURCE PORT*** is optional. When used, it specifies the port to which replies should be sent; if not used, it should be zero.
- The ***LENGTH*** field contains a count of octets in the UDP datagram, including the UDP header and the user data.
- Thus, the minimum value for ***LENGTH*** is eight, the length of the header alone.
- The UDP checksum is optional and need not be used at all; a value of zero in the ***CHECKSUM*** field means that the checksum has not been computed.

UDP Pseudo Header

- The UDP checksum covers more information than is present in the UDP datagram alone.
- To compute the checksum, UDP prepends a ***pseudo-header*** to the UDP datagram, appends an octet of zeros to pad the datagram to an exact multiple of 16 bits, and computes the checksum over the entire object.

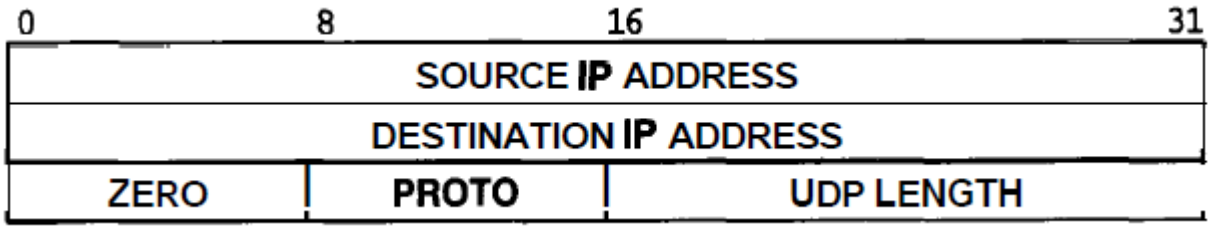
UDP Pseudo Header

- The purpose of using a pseudo-header is to venfy that the UDP datagram has
- reached its correct destination.
- The key to understanding the pseudo-header lies in realizing that the correct destination consists of a specific machine and a specific protocol port within that machine.
- The UDP header itself specifies only the protocol port number.

UDP Pseudo Header

- Thus, to verify the destination, UDP on the sending machine computes a checksum that covers the destination IP address as well as the UDP datagram.
- At the ultimate destination, UDP software verifies the checksum using the destination **IP** address obtained from the header of the IP datagram that carried the UDP message.
- If the checksums agree, then it must be true that the datagram has reached the intended destination host as well as the correct protocol port within that host.

UDP Pseudo Header Format



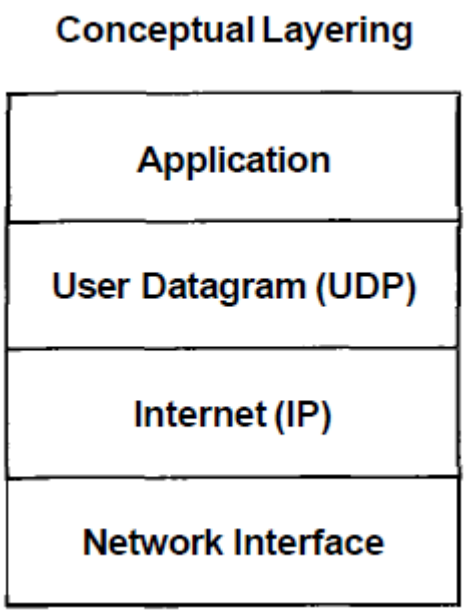
- The pseudo-header used in the UDP checksum computation consists of 12 octets of data arranged

UDP Pseudo Header Format

- The fields of the pseudo-header labeled ***SOURCE IP ADDRESS*** and ***DESTINATION IP ADDRESS*** contain the source and destination **IP** addresses that will be used when sending the UDP message.
- Field ***PROTO*** contains the ***IP*** protocol type code (17 for UDP), and the field labeled ***UDP LENGTH*** contains the length of the UDP datagram (not including the pseudo-header).
- To verify the checksum, the receiver must extract these fields from the IP header, assemble them into the pseudo-header format, and recompute the checksum.

UDP Encapsulation And Protocol Layering

- UDP lies in the layer above the Internet Protocol layer. Conceptually, application programs access UDP, which uses **IP** to send and receive datagrams



UDP Encapsulation And Protocol Layering

- Layering UDP above **IP** means that a complete UDP message, including the UDP header and data, is encapsulated in an IP datagram as it travels across an internet

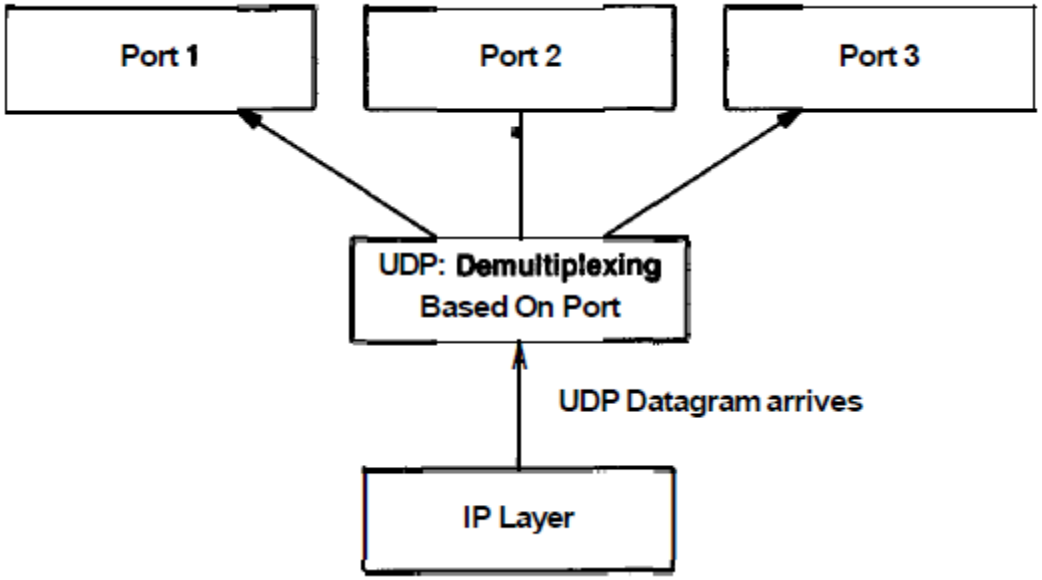
UDP Multiplexing and De-multiplexing

- UDP software provides example of multiplexing and demultiplexing.
- It accepts UDP datagrams from many application programs and passes them to **IP** for transmission.
- It accepts arriving UDP datagrams from **IP** and passes each to the appropriate application program.

UDP Multiplexing and De-multiplexing

- All multiplexing and demultiplexing between UDP software and application programs occur through the port mechanism.
- In practice, each application program must negotiate with the operating system to obtain a protocol port and an associated port number before it can send a UDP datagram
- Once the port has been assigned, any datagram the application program sends through the port will have that port number in its UDP SOURCE PORT field.

UDP Multiplexing and De-multiplexing



Reserved and Available UDP Ports

- Two computers need to agree on port numbers before they can intemperate.
- For example, when computer A wants to obtain a file from computer B, it needs to know what port the file transfer program on computer B uses.
- There are two fundamental approaches to port assignment.
- The first approach uses a central authority.
- Everyone agrees to allow a central authority to assign port numbers as needed and to publish the list of all assignments. Then all software is built according to the list.
- This approach is sometimes called ***universal assignment,*** and the port assignments specified by the authority are called ***well-known*** port ***assignments.***

Reserved and Available UDP Ports

- The second approach to port assignment uses dynamic binding. In the dynamic
- binding approach, ports are not globally known. Instead, whenever a program needs a
- port, the network software assigns one. To learn about the current port assignment on
- another computer, it is necessary to send a request that asks about the current port assignment
- (e.g., What port is the file transfer service using?). The target machine replies
- by giving the correct port number to use.

- The **TCP/IP** designers adopted a hybrid approach that assigns some port numbers a
- priori, but leaves many available for local sites or application programs. The assigned
- port numbers begin at low values and extend upward, leaving large integer values available
- for dynamic assignment. The table in Figure 12.6 lists some of the currently assigned
- UDP port numbers. The second column contains Internet standard assigned keywords,
- while the third contains keywords used on most UNIX systems.

Decimal	Keyword	UNIX Keyword	Description
0			Reserved
7	ECHO	echo	Echo
9	DISCARD	discard	Discard
11	USERS	systat	Active Users
13	DAYTIME	daytime	Daytime
15		netstat	Network status program
17	QUOTE	qotd	Quote of the Day
19	CHARGEN	chargen	Character Generator
37	TIME	time	Time
42	NAMESERVER	name	Host Name Server
43	NICNAME	whois	Who Is
53	DOMAIN	nameserver	Domain Name Server
67	BOOTPS	bootps	BOOTP or DHCP Server
68	BOOTPC	bootpc	BOOTP or DHCP Client
69	TFTP	tftp	Trivial File Transfer
88	KERBEROS	kerberos	Kerberos Security Service
111	SUNRPC	sunrpc	Sun Remote Procedure Call
123	NTP	ntp	Network Time Protocol
161		snmp	Simple Network Management Proto
162		snmp-trap	SNMP traps
512		biff	UNIX comsat
513		who	UNIX rwho daemon
514		syslog	System log
525		timed	Time daemon