UNIT 4

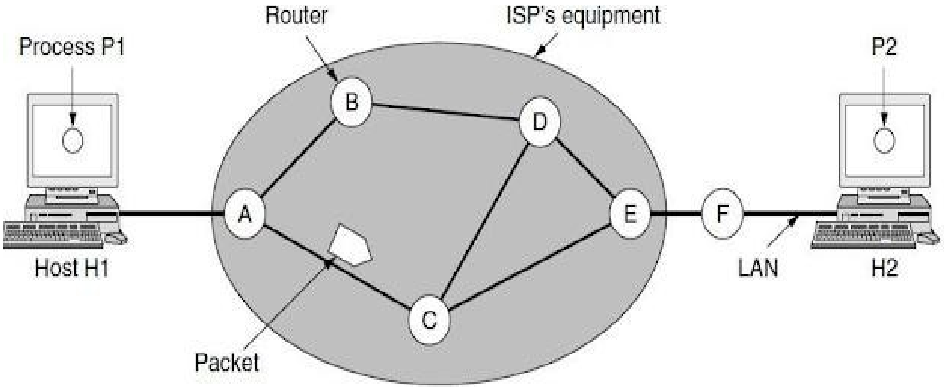
Network Layer - Design Issues - Routing Algorithms - Congestion Control Algorithms - IP Protocol - IP Addresses - Internet Control Protocols.

Functions of Network Layer

- ✓ Routing – find a path from one host to another host.
- ✓ Congestion control – mechanisms to prevent hosts from flooding the network.
- ✓ Quality of Service (QoS) - transmission rates, error rates, and other characteristics can be measured, improved, and, to some extent, guaranteed in advance.
- ✓ Internetworking provides translation between subnet using different protocols.

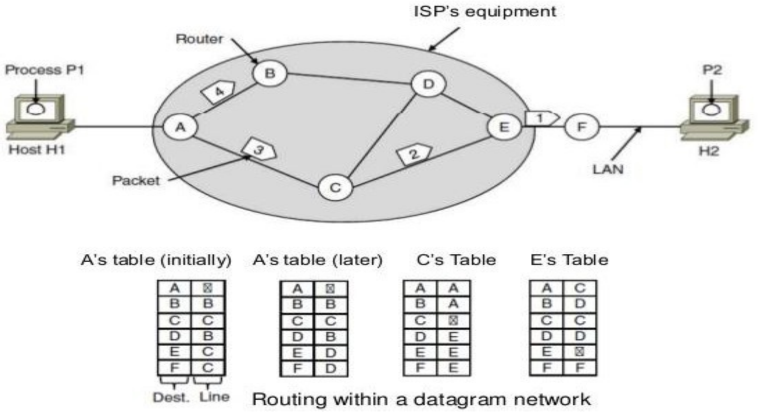
Network layer design issues

- ✓ Store-and-Forward Packet Switching
- ✓ Services Provided to the Transport Layer
- ✓ Implementation of Connectionless Service
- ✓ Implementation of Connection-Oriented Service
  - **Store-and-Forward Packet Switching**



- The major components of the network are the carrier's equipment (routers connected by transmission lines), shown inside the shaded oval, and the
- Customers' equipment, shown outside the oval.
- Host H1 is directly connected to one of the carrier's routers, A, by a leased line. In contrast, H2 is on a LAN with a router, F, owned and operated by the customer. This router also has a leased line to the carrier's equipment.
- We have shown F as being outside the oval because it does not belong to the carrier, but in terms of construction, software, and protocols, it is probably no different from the carrier's routers.
- This equipment is used as follows. A host with a packet to send transmits it to the nearest router, either on its own LAN or over a point- to-point link to the carrier. The packet is stored there until it has fully arrived so the checksum can be verified.
- Then it is forwarded to the next router along the path until it reaches the destination host, where it is delivered. This mechanism is store- and-forward packet switching.

- **Services Provided to the Transport Layer**
  - The network layer provides services to the transport layer at the network layer/transport layer interface. An important question is what kind of services the network layer provides to the transport layer.
  - The network layer services have been designed with the following goals in mind.
    - 1. The services should be independent of the router technology.
    - 2. The transport layer should be shielded from the number, type, and topology of the routers present.
    - 3. Network addresses available to transport layer should use be uniform, even across LANs and WANs.
- **Implementation of Connectionless Service**
  - No advance setup is needed.
  - The packets are frequently called datagrams.
  - The subnet is called a datagram network.
  - The routing algorithm is the algorithm that manages the tables and makes the routing decision.



A's table (initially)

Dest.	Line
A	☒
B	B
C	C
D	B
E	C
F	C

A's table (later)

A	☒
B	B
C	C
D	B
E	D
F	D

C's Table

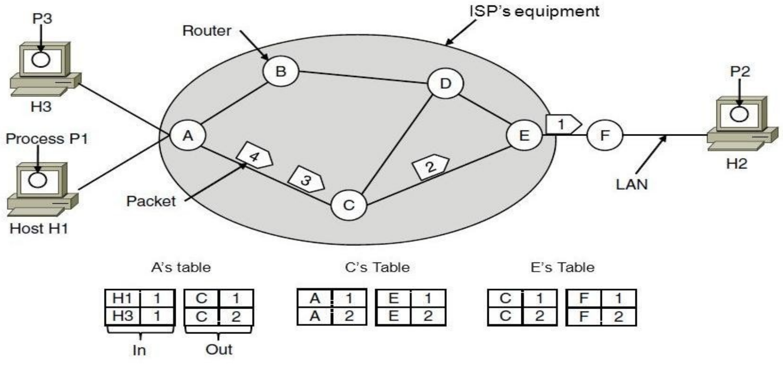
A	A
B	A
C	☒
D	E
E	E
F	E

E's Table

A	C
B	D
C	C
D	D
E	☒
F	F

Routing within a datagram network

- **Implementation of Connection-Oriented Service**
  - A path from the source router to the destination router must be established before any data packets can be sent.
  - The connection is called a VC (virtual circuit).
  - The network is called a virtual-circuit network.
  - To distinguish packets from different hosts, replacing connection identifiers in outgoing packets is called label switching.



A's table

In		Out	
H1	1	C	1
H3	1	C	2

C's Table

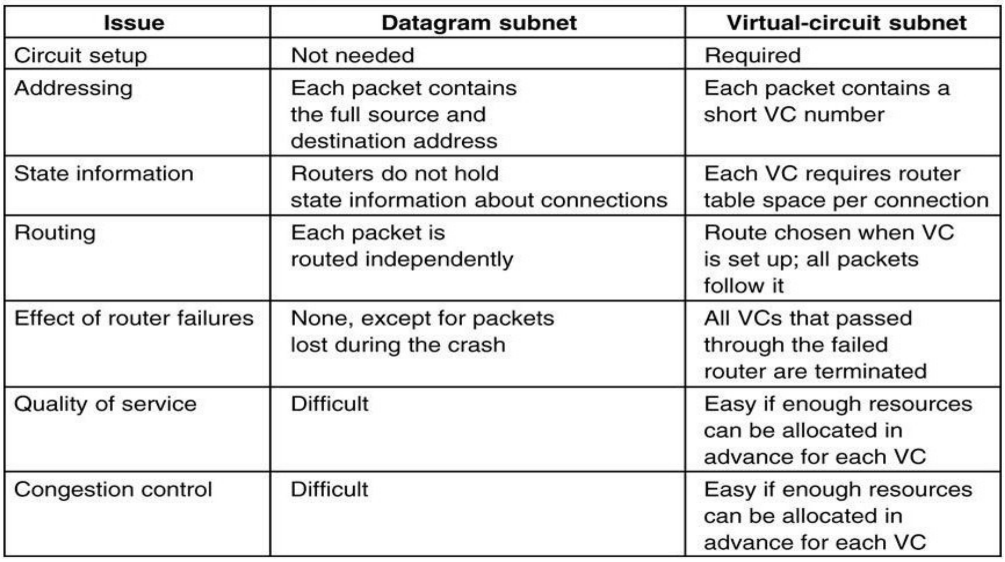
A	1	E	1
A	2	E	2

E's Table

C	1	F	1
C	2	F	2

Routing within a virtual-circuit network

- ✓ **Comparison of Virtual-Circuit and Datagram Network**



Issue	Datagram subnet	Virtual-circuit subnet
Circuit setup	Not needed	Required
Addressing	Each packet contains the full source and destination address	Each packet contains a short VC number
State information	Routers do not hold state information about connections	Each VC requires router table space per connection
Routing	Each packet is routed independently	Route chosen when VC is set up; all packets follow it
Effect of router failures	None, except for packets lost during the crash	All VCs that passed through the failed router are terminated
Quality of service	Difficult	Easy if enough resources can be allocated in advance for each VC
Congestion control	Difficult	Easy if enough resources can be allocated in advance for each VC

Routing Algorithms

- ✓ Routing is the process of moving packets from the source to a destination in internetworking.
- ✓ Routing protocols use a routing algorithm which is a mathematical formula to forward the packet to its destination.
- ✓ The main function of the network layer is routing packets from the source machine to the destination machine.
- ✓ In most networks, packets will require multiple hops to make the journey.
- ✓ More than one route is possible in every network; however the shortest route should be selected.
- ✓ The shortest route means, a route which passes through the least number of nodes to reach the destination.
- ✓ The routing algorithm is designed to find the shortest route and it is part of network software.
- ✓ Routing Table: To route IP packets, a host or a router has a routing table with entries for each destination or a combination of destinations.
    - o A static routing table contains information, which is entered manually. The administrator enters the route for each destination into the table.
    - o Dynamic routing table is updated periodically by using dynamic protocols like RIP, OSPF or BGP.
- ✓ **The routing algorithm can be classified into two types:**
    - o **Static (non-adaptive) routing algorithms**
        - In this type, the network topology determines the final path. All the possible paths which are already calculated are loaded into the routing table.
        - Static routing is suitable for small networks.
        - The disadvantage of static routing is, inability to respond quickly in case of network failure.
    - o **Dynamic (Adaptive) routing algorithms**
        - The dynamic routing algorithms can change their routing decision on the basis of some changes made in the topology.

- Each router can check the network status by communicating with the neighbours. So, the changes in the topology are reflected to all routers.
- Finally, the router can calculate the suitable path to the final destination.
- The disadvantage of this type is complexity in the router.

✓ **Intra-domain Routing and Inter-domain Routing**

- An autonomous system is a group of the networks and the routers, which are operated by the network administrator. Internet can be divided into autonomous systems.
- Routing inside an autonomous system is referred to as intra-domain routing.
  - Protocols for Intra-domain routing are called as interior gateway protocols.
  - Distance vector and link state routing are the examples of Intra-domain routing.
    - Example: RIP and OSPF
- Routing between two or more autonomous systems can be referred to as inter-domain routing.
  - Protocol for Inter-domain routing are also called as exterior gateway protocols.
  - Path vector is an example of an inter-domain routing.
    - Example: BGP

Distance Vector Routing

✓ Distance vector routing is the dynamic routing algorithm and also known as **Bellman-Ford** routing algorithm and **Ford- Fulkerson** algorithm.

✓ It was designed for small network topologies.

✓ In this algorithm, node router constructs a table containing the distance (total cost of path) to all other nodes and distributes that vector to its immediate neighbors.

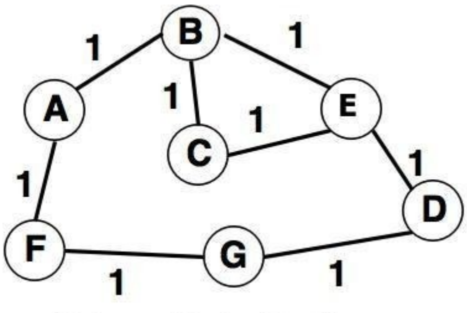
✓ For distance vector routing, it is assumed that each node knows the cost of the link to each of its directly connected neighbors.

✓ A link, which is 'down' is assigned as an infinite cost.

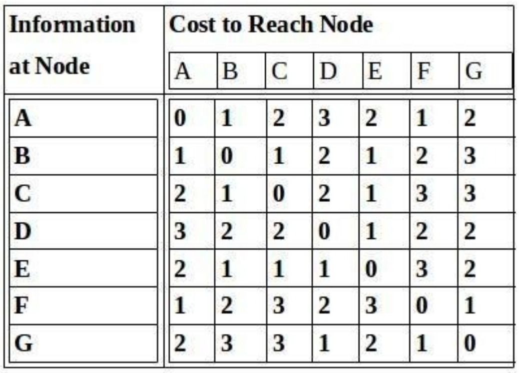
✓ Every node sends a message to its directly connected neighbors

- **For example:** A sends its information to B and F.

✓ After communicating to each directly connected node the shortest path can be easy to compute (as shown in above table).



Distance Vector Routing



Information at Node	Cost to Reach Node						
	A	B	C	D	E	F	G
A	0	1	2	3	2	1	2
B	1	0	1	2	1	2	3
C	2	1	0	2	1	3	3
D	3	2	2	0	1	2	2
E	2	1	1	1	0	3	2
F	1	2	3	2	3	0	1
G	2	3	3	1	2	1	0

Routing Information Protocol (RIP)

- RIP is a dynamic, distance vector routing protocol based around the Berkely BSD application *routed* and was developed for smaller IP based networks. RIP uses UDP port 520 for route updates. RIP calculates the best route based on hop count. Like all distance vector routing protocols, RIP takes some time to converge. While RIP requires less CPU power and RAM than some other routing protocols, RIP does have some limitations:
    - Metric: Hop Count
        - Since RIP calculates the best route to a destination based solely on how many hops it is to the destination network, RIP tends to be inefficient in network using more than one LAN protocol, such as Fast Ethernet and serial or Token Ring. This is because RIP prefers paths with the shortest hop count. The path with the shortest hop count might be over the slowest link in the network.
    - Hop Count Limit
        - RIP cannot handle more than 15 hops. Anything more than 15 hops away is considered unreachable by RIP. This fact is used by RIP to prevent routing loops.
    - Classful Routing Only
        - RIP is a classful routing protocol. RIP cannot handle classless routing. RIP v1 advertises all networks it knows as classful networks, so it is impossible to subnet a network properly via VLSM if you are running RIP v1, which
- However, it must be pointed out that RIP is the only routing protocol that all routing devices and software support, so in a mixed equipment environment, RIP may be your only option for dynamic routing.

- RIP MESSAGES
    - RIP updates are placed as UDP payload inside an IP datagram. Below is the base format of a RIP message.

command	version	zeroes
Address Family ID		zeroes
IP Address		
zeroes		
zeroes		
Metric		
Payload		

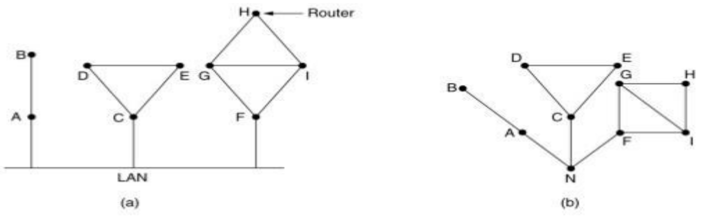
-
    - COMMAND types (field value)
        - REQUEST (1)- Request either a partial or full table update from another RIP router.
        - RESPONSE (2) - A response to a request. All route updates use this command in the command field.
        - TRACEON (3) / TRACEOFF (4) - Obsolete and ignored.
        - RESERVED (5) - Sun Microsystems uses this field for it's own purposes.
    - VERSION field - Describes which version of the RIP protocol it is (1 or 2).
    - ADDRESS FAMILY ID - Identifies which addressing protocol is being used (CLNS, IPX, IP etc.)
    - METRIC - Metric measures how 'good' a route is. RIP uses the number of hops as the metric. The route with the fewest number of hops is preferred.

- RIP ROUTING UPDATES
    - Routers running IP RIP broadcast the full list of all the routes they know every 30 seconds. When a router running RIP hears a broadcast it runs the distance vector algorithm to create a list of best routes.

- RIP TIMERS
    - The routing-update timer controls the time between routing updates. Default is usually 30 seconds, plus a small random delay to prevent all RIP routers from sending updates simultaneously.
    - The route-timeout timer controls when a route is no longer available. The default is usually 180 seconds. If a router has not seen the route in an update during this specified interval, it is dropped from the router's announcements. The route is maintained long enough for the router to advertise the route as down (hop count of 16).
    - The route-flush timer controls how long before a route is completely flushed from the routing table. The default setting is usually 120 seconds.

Link State Protocol

- Link state routing is the second family of routing protocols.
- 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 a shortest path computation.
- The idea behind link state routing is fairly simple and can be stated as five parts. Each router must do the following things to make it work:
    - 1. Discover its neighbors and learn their network addresses.
    - 2. Set the distance or cost metric to each of its neighbors.
    - 3. Construct a packet telling all it has just learned.
    - 4. Send this packet to and receive packets from all other routers.
    - 5. Compute the shortest path to every other router.
- Dijkstra’s algorithm can be run at each router to find the shortest path to every other router.
- **Learning about the Neighbors**
    - Each Link State enabled router periodically sends a HELLO message on each of its links.
    - Neighbor routers respond to these HELLO messages identifying themselves. Within the replies, network addresses of the routers are attached and are used by the HELLO initiator to build up its neighbor table.



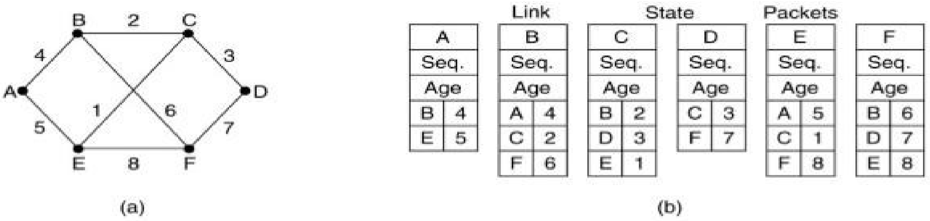
(a) Nine routers and a LAN. (b) A graph model of (a).

- **Setting Link Costs**

- The link state routing algorithm requires each link to have a distance or cost metric for finding shortest paths.
- The cost to reach neighbors can be set automatically, or configured by the network operator.
- A common choice is to make the cost inversely proportional to the bandwidth of the link.
  - For example, 1-Gbps Ethernet may have a cost of 1 and 100-Mbps Ethernet a cost of 10. This makes higher-capacity paths better choices.
  - The most direct way to determine this delay is to send over the line a special ECHO packet that the other side is required to send back immediately.
  - By measuring the round-trip time and dividing it by two, the sending router can get a reasonable estimate of the delay.

✓ **Building Link State Packets**

- Once the information needed for the exchange has been collected, the next step is for each router to build a packet containing all the data.
- The packet starts with the identity of the sender, followed by a sequence number and age and a list of neighbors.
- The cost to each neighbor is also given. An example network is presented in Fig. (a) with costs shown as labels on the lines.
- The corresponding link state packets for all six routers are shown in Fig. (b).



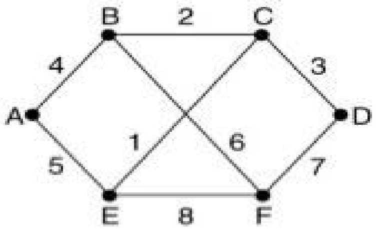
Link State Packets

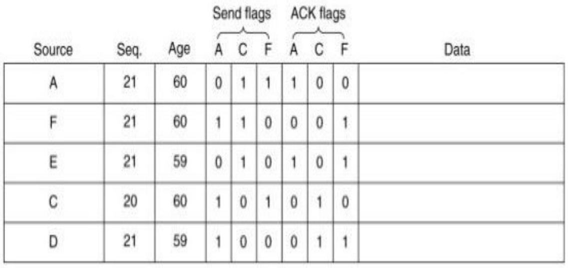
A	B	C	D	E	F
Seq.	Seq.	Seq.	Seq.	Seq.	Seq.
Age	Age	Age	Age	Age	Age
B 4	A 4	B 2	C 3	A 5	B 6
E 5	C 2	D 3	F 7	C 1	D 7
	F 6	E 1		F 8	E 8

(a) A subnet. (b) The link state packets for this subnet.

✓ **Distributing the Link State Packets**

- The trickiest part of the algorithm is distributing the link state packets. All of the routers must get all of the link state packets quickly and reliably.
- For flooding packets, here used basic distribution algorithm.
  - The fundamental idea is to use flooding to distribute the link state packets to all routers.
  - To keep the flood in check, each packet contains a sequence number that is incremented for each new packet sent. Routers keep track of all the (source router, sequence) pairs they see.
  - When a new link state packet comes in, it is checked against the list of packets already seen.
    - If it is new, it is forwarded on all lines except the one it arrived on.
    - If it is a duplicate, it is discarded.
    - If a packet with a sequence number lower than the highest one seen so far ever arrives, it is rejected as being obsolete as the router has more recent data.



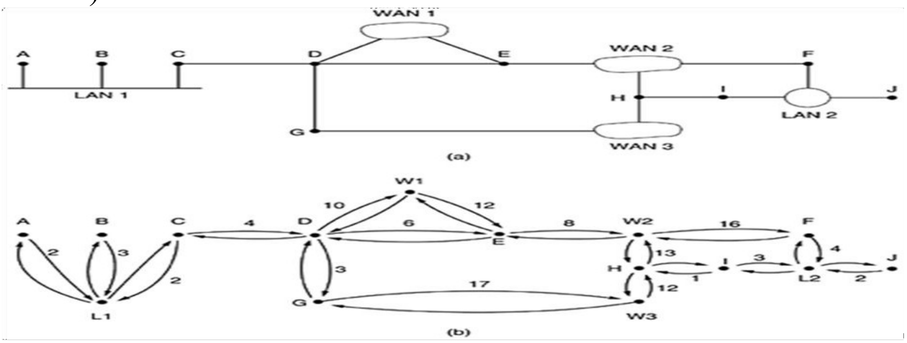


Source	Seq.	Age	Send flags			ACK flags			Data
			A	C	F	A	C	F	
A	21	60	0	1	1	1	0	0	
F	21	60	1	1	0	0	0	1	
E	21	59	0	1	0	1	0	1	
C	20	60	1	0	1	0	1	0	
D	21	59	1	0	0	0	1	1	

- ✓ **Computing the New Routes**
  - o Once a router has accumulated a full set of link state packets, it can construct the entire network graph because every link is represented.
  - o Every link is, in fact, represented twice, once for each direction. The different directions may even have different costs.
  - o The shortest-path computations may then find different paths from router A to B than from router B to A.
  - o Now Dijkstra's algorithm can be run locally to construct the shortest paths to all possible destinations.
  - o The results of this algorithm tell the router which link to use to reach each destination.

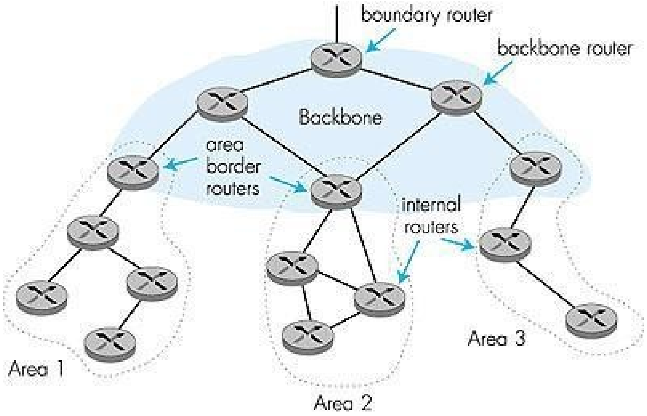
Open Shortest Path First (OSPF)

- ✓ The Internet is made up of a large number of autonomous systems. Each AS is operated by a different organization and can use its own routing algorithm inside.
  - o For example, the internal networks of companies X, Y, and Z are usually seen as three ASes if all three are on the Internet.
- ✓ All three may use different routing algorithms internally. Nevertheless, having standards, even for internal routing, simplifies the implementation at the boundaries between ASes and allows reuse of code.
- ✓ A routing algorithm within an AS is called an interior gateway protocol; an algorithm for routing between ASes is called an exterior gateway protocol.
- ✓ The original Internet interior gateway protocol was a distance vector protocol (RIP) based on the Bellman-Ford algorithm inherited from the ARPANET.
- ✓ In 1988, the Internet Engineering Task Force began work on a successor. That successor, called OSPF (Open Shortest Path First), became a standard in 1990.
- ✓ OSPF supports three kinds of connections and networks:
  - o 1. Point-to-point lines between exactly two routers.
  - o 2. Multiaccess networks with broadcasting (e.g., most LANs).
  - o 3. Multiaccess networks without broadcasting (e.g., most packet-switched WANs).



(a) An autonomous system. (b) A graph representation of (a).

- Many of the ASes in the Internet are themselves large and nontrivial to manage.
- OSPF allows them to be divided into numbered areas, where an area is a network or a set of contiguous networks.
- Every AS has a backbone area, called area 0. All areas are connected to the backbone, possibly by tunnels, so it is possible to go from any area in the AS to any other area in the AS via the backbone. A tunnel is represented in the graph as an arc and has a cost. Each router that is connected to two or more areas is part of the backbone.



- The five types of OSPF messages:

Message type	**Description**
Hello	Used to discover who the neighbors are Link state
update	Provides the sender’s costs to its neighbors Link
state ack	Acknowledges link state update
Database description	Announces which updates the sender has Link state
request	Requests information from the partner

  - **Hello**
    - Hello messages are used as a form of greeting, to allow a router to discover other adjacent routers on its local links and networks.
    - The messages establish relationships between neighboring devices and communicate key parameters about how OSPF is to be used in the autonomous system or area.
    - During normal operation, routers send hello messages to their neighbors at regular intervals (the hello interval); if a router stops receiving hello messages from a neighbor, after a set period (the dead interval) the router will assume the neighbor has gone down.
  - **Database Description (DBD)**
    - Database description messages contain descriptions of the topology of the autonomous system or area. They convey the contents of the link- state database (LSDB) for the area from one router to another. Communicating a large LSDB may require several messages to be sent by having the sending device designated as a master device and sending messages in sequence, with the slave (recipient of the LSDB information) responding with acknowledgements.
  - **Link State Request (LSR)**
    - Link state request messages are used by one router to request updated information about a portion of the LSDB from another router. The message specifies the link(s) for which the requesting device wants more current information.

- **Link State Update (LSU)**
  - *Link state update* messages contain updated information about the state of certain links on the LSDB. They are sent in response to a Link State Request message, and also broadcast or multicast by routers on a regular basis. Their contents are used to update the information in the LSDBs of routers that receive them.
- **Link State Acknowledgment (LSAck)**
  - *Link state acknowledgement* messages provide reliability to the link- state exchange process, by explicitly acknowledging receipt of a Link State Update message.

Destination Sequenced Distance Vector routing (DSDV)

- ✓ Destination Sequenced Distance Vector (DSDV) routing is a table driven or proactive and hop-by-hop distance vector routing protocol based on the concept of the classical distributed Bellman-Ford Algorithm, in which each mobile node maintains a routing table that contains the number of hops to reach the destination node in the shortest path, all available destinations in the network and the sequence number fixed by the destination node to prevent looping problem.
- ✓ To maintain the regularity of routing tables in a dynamically varying topology, each and every neighbor node exchanges its routing tables periodically at regular time intervals.
- ✓ A node also broadcasts its routing table to neighbors if any change occurs in the routing table due to changes in the local topology.
- ✓ Updating of the routing table is both time-driven and event-driven and is done by two methods:
  - **Incremental updates:** An incremental update takes a single Network Data Packet Unit (NDPU) which is used when a node does not detect considerable changes in topology.
  - **Full dump updates:** A full dump update may take multiple NDPUs. It is done either when an incremental update wants more than a single network data packet unit or when the topology changes significantly.
- ✓ If there is a space in the incremental update packet, then those entries may be included whose sequence number has changed when routing table information is modified. When two routes to a destination are received from two different neighbors, the one with greatest number is selected. If equal the smaller hop count is selected.
- ✓ DSDV protocol guarantees loop free paths and count to infinity problem is reduced.
- ✓ DSDV protocol has four different phases, which are described as follows :
  - **Route advertisements:**
    - Each node has to maintain a routing table in which all the accessible destinations within the network and the number of hops to every destination are saved.
    - Each entry in the table has a sequence number fixed by the destination node. This number enables the mobile node to distinguish stale routes from new ones, and also keep away from the formation of routing loops.
    - These routing tables are transmitting to its immediate neighbors periodically at regular intervals time.

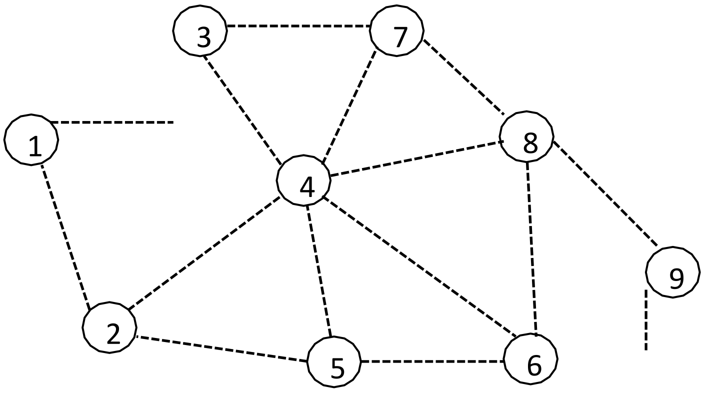


Figure 1. Route establishments in DSDV.

- **Routing table entry construction:**
  - The packet broadcast by each node has the new sequence number and the information in the packet for each new route are the destination address, the number of hops to reach the destination and the sequence number of the information received about that destination and stamped through the destination.

Table 1 Routing table at node 1

Destination	Next hop	Number of hops	Sequence number
1	1	0	20
2	2	1	16
3	3	1	78
4	3	2	48
5	2	2	52
6	2	3	18
7	3	2	78
8	2	3	58
9	2	4	24

- **Response to changes in topology:**
  - Each and every immediate neighbor node in the network can exchange its routing tables periodically at regular time intervals to maintain the consistency of routing tables in a dynamically changing topology. A node updates its routing table and broadcasts immediately when significant new information is available.

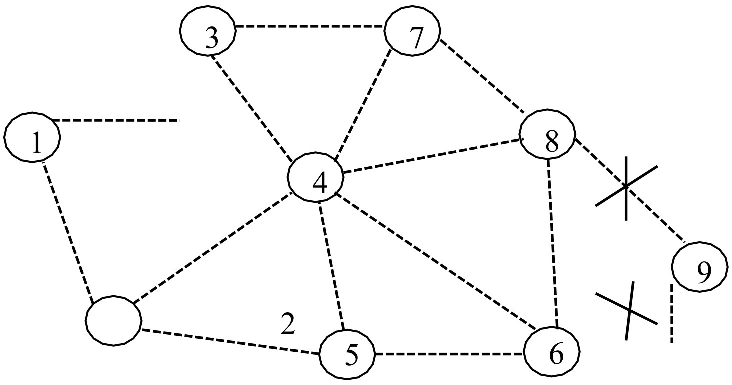


Figure 2 Route failures in DSDV

Table 2 Routing table at node 1 (Link failure)

Destination	Next hop	Number of hops	Sequence number
1	1	0	20
2	2	1	16
3	3	1	78
4	3	2	48
5	2	2	52
6	2	3	18
7	3	2	78
8	2	3	58
9	2	∞	24

- **Route selection:**
  - If the source node gets new routing information through an incremental packet, it compares with available routing information from previous routing packets.

Table 3 Routing table at node 1 (update discard based on sequence number)

Destination	Next hop	Number of hops	Sequence number
1	1	0	20
2	2	1	16
3	3	1	78
4	3	2	48
5	2	2	52
6	2	3	18
7	3	2	78
8	2	3	58
9	2	4	24

Table 4 Routing table at node 1 (update discard based on the number of hops)

Destination	Next hop	Number of hops	Sequence number
1	1	0	20
2	2	1	16
3	3	1	78
4	3	2	48
5	2	2	52
6	2	3	18
7	3	2	78

8	2	3	58
9	2	4	24

- The route uses the recent sequence number information. Routes with existing sequence numbers are removed. A route with a sequence number equal to an existing route is elected if it is more cost effective. Then an existing route may be removed or stored as a less preferable route.
- Table 3 indicates the routing table at node 1 for update discard based on sequence number. Table 4 indicates the routing table at node 1 for update discard based on the number of hops. Table 5 indicates the updated routing table at node 1.

Table 5 Updated routing table at node 1

Destination	Next hop	Number of hops	Sequence number
1	1	0	20
2	2	1	16
3	3	1	78
4	3	2	48
5	2	2	~~52~~ 54
6	2	3	18
7	3	2	78
8	2	3	58
9	2	4	24

Dynamic Source routing Protocol

- ✓ Dynamic Source Routing (DSR) protocol is a simple and efficient routing protocol. It is also known as on-demand protocol because the packets are forwarded only when there is demand by the nodes. It does not need any network infrastructure as it is based on demand.
- ✓ The protocol consists of route discovery and route maintenance. They are used together to discover and maintain routers.
- ✓ DSR avoids flooding and it does not need the up-to-date routing information. Hence the operation of DSR has been evaluated on a variety of patterns and communication patterns in an ad hoc network.
- ✓ DSR divides the task of routing into two distinct problems. They are route discovery and route maintenance.In route discovery, A node only tries to discover a route to a destination if it has to send something to this destination and there is currently no known route.
- ✓ In route maintenance, if a node is continuously sending packets via a route, it has to make sure that the route is held upright. As soon as a node detects problems with the current route, it has to find an alternative.
- ✓ The above two mechanisms are based on demand. When compared to other protocols, DSR is independent of periodic update of routing table. It does not use any periodic routing advertisement, link status sensing, neighbor detection packets.

- ✓ DSR allows unidirectional links and asymmetric routes. A node can send the packets to other nodes while the opponent is free i.e. idle. It increases the overall performance and network connectivity.
- ✓ DSR also acts as an interface between different types of wireless networks.
  - **DSR route discovery:**

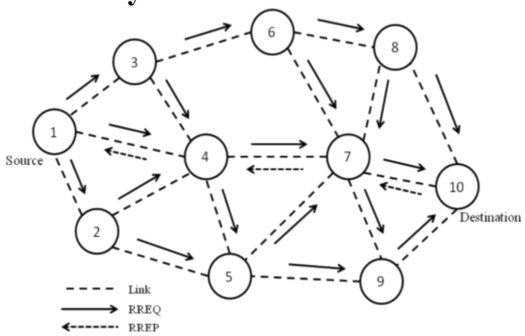


Figure 1 An example for DSR route discovery

  - Route discovery is the mechanism by which a node wishing to send a packet to a destination node obtains a source route.
  - A route is discovered only when a node sends a packet to the other node. Route discovery takes place while a node that wants to send a packet first establishes a route by sending ROUTE REQUEST message to all the other nodes available in the network.
  - If the request has reached the target node, the ROUTE REPLY message is being transferred to the initiator node. Otherwise, if the target node has already received this ROUTE REQUEST message, then it drops the request, or else the nodes append their own address to a list of traversed hops in the packet and broadcast this updated route request.
  - The address of the intermediate node is maintained in the ROUTE REQUEST message for future use.
  - The copy of the original packet is saved in the local buffer called send buffer by the sending nodes.
  - To reduce the overhead, once a node discovers a route, it can send various packets to the same target node.
  - Additional route discovery features:
    - Caching overhead,
    - Replying to route requests using cached routes,
    - Preventing route reply storms,
    - Route request hop limits.
  - **DSR route maintainance:**

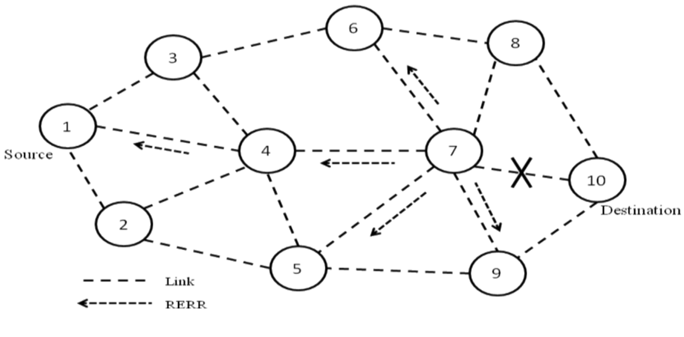


Figure 2 An example for DSR route maintenance

- After the discovery of a route, there should be maintenance of a route.
- Route maintenance involves the maintenance of a route as long as the node sends packets along this route.
- When a node has discovered a route, a passive acknowledgement is being sent.
- A ROUTE ERROR message is sent to the sender when the packet is retransmitted maximum number of times and no acknowledgement is being received.
- Figure 2 shows the link failure between node 7 and node 10. Hence the need for route maintenance.

Congestion Control

- What is congestion?

A state occurring in network layer when the message traffic is so heavy that it slows down network response time.

Effects of Congestion

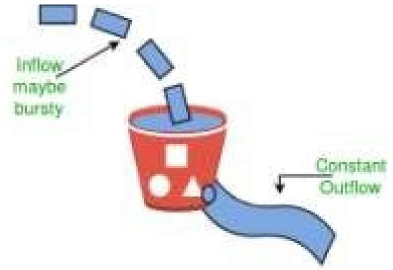
- As delay increases, performance decreases.
- If delay increases, retransmission occurs, making situation worse.

Congestion control algorithms

- **Leaky Bucket Algorithm**

Let us consider an example to understand

Imagine a bucket with a small hole in the bottom. No matter at what rate water enters the bucket, the outflow is at constant rate. When the bucket is full with water additional water entering spills over the sides and is lost.



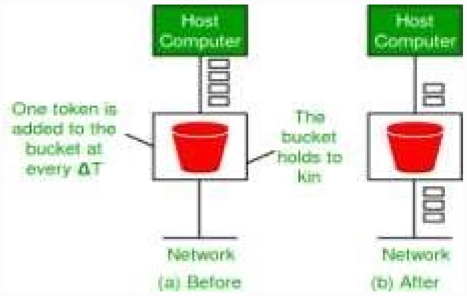
Similarly, each network interface contains a leaky bucket and the following steps are involved in leaky bucket algorithm:

1. When host wants to send packet, packet is thrown into the bucket.
2. The bucket leaks at a constant rate, meaning the network interface transmits packets at a constant rate.
3. Bursty traffic is converted to a uniform traffic by the leaky bucket.
4. In practice the bucket is a finite queue that outputs at a finite rate.

- **Token bucket Algorithm**
  - Need of token bucket Algorithm:-
  - The leaky bucket algorithm enforces output pattern at the average rate, no matter how bursty the traffic is.
  - So in order to deal with the bursty traffic we need a flexible algorithm so that the data is not lost.
  - One such algorithm is token bucket algorithm.

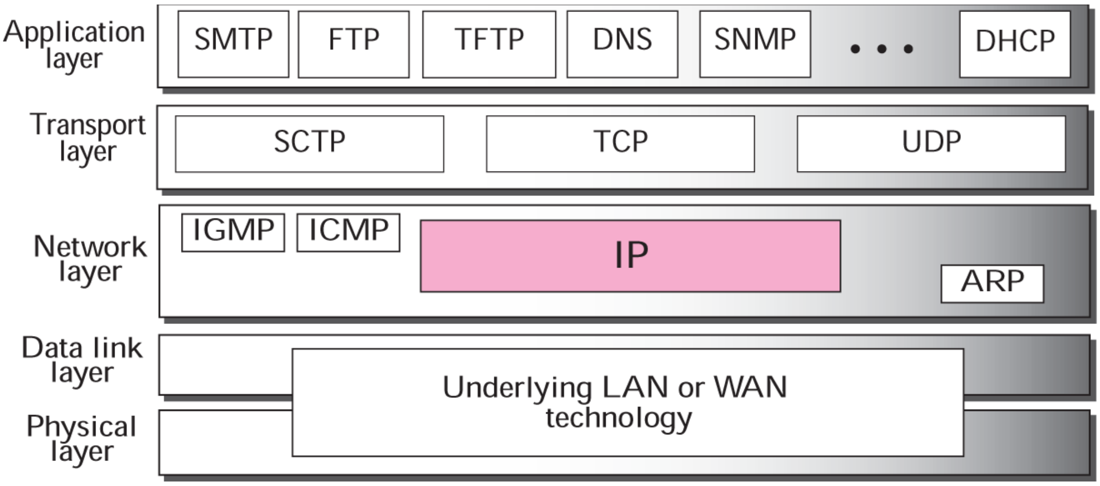
Steps of this algorithm can be described as follows:

1. In regular intervals tokens are thrown into the bucket. ƒ
2. The bucket has a maximum capacity. ƒ
3. If there is a ready packet, a token is removed from the bucket, and the packet is sent.
4. If there is no token in the bucket, the packet cannot be sent.



Internet protocol version 4 (IPv4)

- ✓ The Internet Protocol (IP) is the transmission mechanism used by the TCP/IP protocols at the network layer.
- ✓ Figure shows the position of IP in the suite.

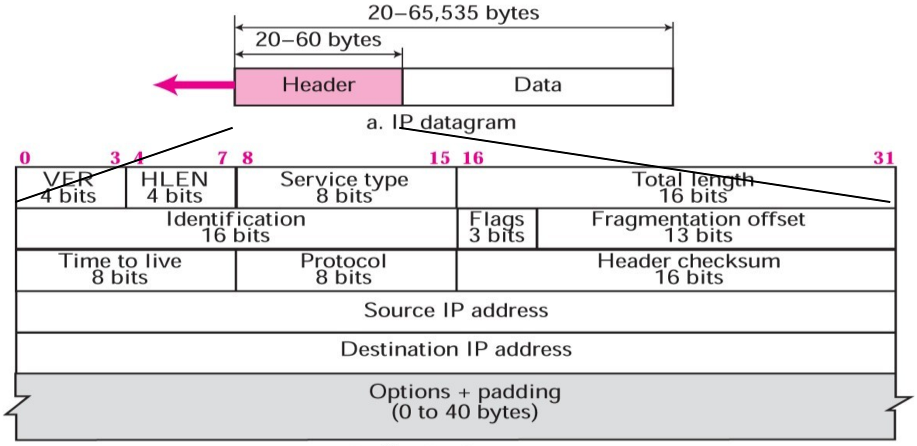


- ✓ IP is an unreliable and connectionless datagram protocol.
- ✓ It is a best-effort delivery service. The term best-effort means that IP packets can be corrupted, lost, arrive out of order, or delayed and may create congestion for the network.
- ✓ If reliability is important, IP must be paired with a reliable protocol such as TCP.
- ✓ IP is also a connectionless protocol for a packet switching network that uses the datagram approach.

Datagrams

- ✓ Packets in the network (internet) layer are called datagrams.
- ✓ A datagram is a variable-length packet consisting of two parts:
  - o Header - It is 20 to 60 bytes in length and contains information essential to routing and delivery.
  - o Data – Payload data.

- ✓ Figure shows the IP datagram format.



a. IP datagram

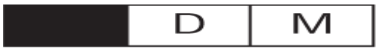
b. Header format

- ✓ **Version (VER) :**
  - o This 4-bit field defines the version of the IP protocol. Currently the version is 4. This field tells the IP software running in the processing machine that the datagram has the format of version 4.

- ✓ **Header length (HLEN):**
  - o This 4-bit field defines the total length of the datagram header in 4-byte words. This field is needed because the length of the header is variable (between 20 and 60 bytes).
    - ▪ When there are no options, the header length is 20 bytes.
    - ▪ When the option field is at its maximum size 60 bytes.
- ✓ **Service type:**
  - o In the original design of IP header, this field was referred to as type of service (TOS)
  - o TOS component is used to determine the type of service that must be provided by the Internet layer depending on the type of application for which the data transfer needs to be done.
  - o It has 8 bits field.
    - ▪ The first three bits filed are known as precedence bits (ignored as today).
    - ▪ The next 4 bits represent type of service
      - • TOS are :
        - o 0000 - Normal
        - o 0001 - Minimizing Cost
        - o 0010 - Maximize reliability.
        - o 0100 - Maximize throughput
        - o 1000 - Minimize delay.
    - ▪ Last bit is unused.
- ✓ **Total length:**
  - o This is a 16-bit field that defines the total length (header plus data) of the IP datagram in bytes.
  - o To find the length of the data coming from the upper layer, subtract the header length from the total length.

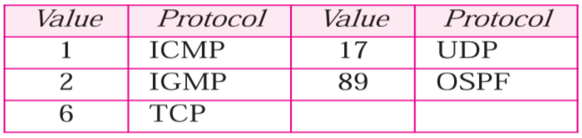
    Length of data = total length - header length
  - o The header length can be found by multiplying the value in the HLEN field by four.

- **Identification:**
  - This 16-bit field identifies a datagram originating from the source host. The combination of the identification and source IP address must uniquely define a datagram as it leaves the source host.
  - All fragments of a datagram contain the same identification value.
  - This allows the destination host to determine which fragment belongs to which datagram.
- **Flags:**
  - This is a three-bit field.



    - The first bit is reserved (not used).
    - The second bit (D) is called the **do not fragment bit.**
      - If its value is 1, the machine must not fragment the datagram
      - If its value is 0, the datagram can be fragmented if necessary.
    - The third bit (M) is called the **more fragment bit.**
      - If its value is 1, it means the datagram is not the last fragment; there are more fragments after this one.
      - If its value is 0, it means this is the last or only fragment.

- **Fragmentation offset:**
  - This 13-bit field shows the relative position of this fragment with respect to the whole datagram.
  - It is the offset of the data in the original datagram measured in units of 8 bytes.
- **Time to live:**
  - A datagram has a limited lifetime in its travel through an internet.
  - This field was originally designed to hold a timestamp, which was decremented by each visited router.
  - The datagram was discarded when the value became zero.
- **Protocol:**
  - This 8-bit field defines the higher-level protocol that uses the services of the IP layer.
  - An IP datagram can encapsulate data from several higher level protocols such as TCP, UDP, ICMP, and IGMP.
  - This field specifies the final destination protocol to which the IP datagram should be delivered.
  - Some of the value of this field for different higher-level protocols



Value	*Protocol*	*Value*	*Protocol*
1	ICMP	17	UDP
2	IGMP	89	OSPF
6	TCP		

- **Checksum:**
  - To provide basic protection against corruption in transmission.
  - Example – CRC

- ✓ **Source address:**
  - o This 32-bit field defines the IP address of the source. This field must remain unchanged during the time the IP datagram travels from the source host to the destination host.
- ✓ **Destination address:**
  - o This 32-bit field defines the IP address of the destination. This field must remain unchanged during the time the IP datagram travels from the source host to the destination host.
- ✓ **Options:**
  - o One or more several types of options may be included after the standard header in certain IP datagrams.
- ✓ **Padding:**
  - o The variable part comprises the options, which can be a maximum of 40 bytes. Options, as the name implies, are not required for a datagram. They can be used for network testing and debugging.
- ✓ **Data:**
  - o The data to be transmitted in the datagram.

Internet protocol version 6 (IPv6)

IPv6 - Introduction

- ✓ Several reasons for the need of a new protocol, Internet Protocol version 6 (IPv6).
  - o The main reason was the address depletion.
  - o Other reasons are related to the slowness of the process due to some unnecessary processing, the need for new options, support for multimedia, and the desperate need for security.

Comparison between IPv4 and IPv6 Headers

IPv4	IPv6
Source and destination address are 32bits or 4Bytes Example: 192.168.0.1	Source and destination address are 128bits or 16Bytes Example: 2001:0db8:85a3:0000:0000:8a2e:0370:7334
Header contains checksum	Header does not contains checksum
Header contains options	All optional data in moved to IPv6 extension headers
Broadcast addresses are used to send packets to all nodes on a subnet.	No Broadcast addresses, Instead link local scope all nodes multicast address is used.
Manual or DHCP based IP configuration.	Nodes are capable of auto configuration.
Fragmentation is done by sending host and also router which slows down the process.	Fragmentation is done only by the sender of the packet.
IPSec header support is optional	IPSec header support is required.

No identification of packet flow in IP header.	Flow label field is used to identify the packet flow and prioritized delivery

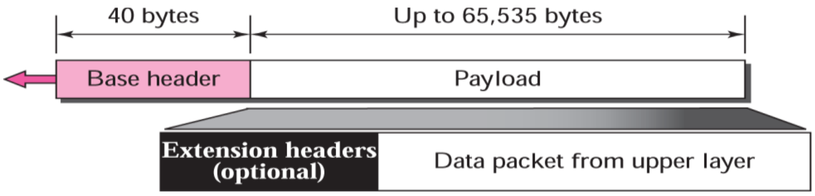
Advantages of IPv6

- ✓ **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 upper-layer 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.

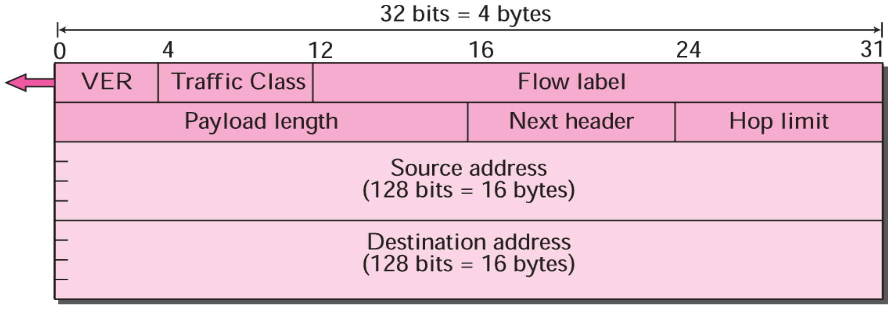
IPv6 Header

- ✓ The IPv6 packet is shown in below. Each packet is composed of a mandatory base header followed by the payload.
- ✓ The payload consists of two parts:
  - o optional extension headers and
  - o data from an upper layer.
- ✓ The base header occupies 40 bytes, whereas the extension headers and data from the upper layer contain up to 65,535 bytes of information.

- ✓ IPv6 datagram

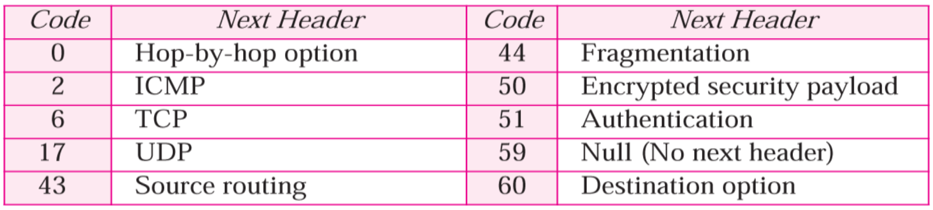


- ✓ Format of the base header



  - o These fields are as follows:
    - o **Version:** This 4-bit field defines the version number of the IP. For IPv6, the value is 6.

- **Traffic Class:** This 8-bit field is used to distinguish different payloads with different delivery requirements. It replaces the service class 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.
- **Payload length:** The 2-byte payload length field defines the length of the IP datagram excluding the base header.
- **Next header:** The **next header** is an 8-bit field defining the header that follows the base header in the datagram. The next header is either one of the optional extension headers used by IP or the header of an encapsulated packet such as UDP or TCP.



Code	*Next Header*	*Code*	*Next Header*
0	Hop-by-hop option	44	Fragmentation
2	ICMP	50	Encrypted security payload
6	TCP	51	Authentication
17	UDP	59	Null (No next header)
43	Source routing	60	Destination option

- **Hop limit:** This 8-bit **hop limit** field serves the same purpose as the TTL field in IPv4.
- **Source address:** The source address field is a 16-byte (128-bit) Internet address that identifies the original source of the datagram.
- **Destination address:** The destination address field is a 16-byte (128-bit) Internet address that usually identifies the final destination of the datagram. However, if source routing is used, this field contains the address of the next router.

✓ **Flow Label:**

- In version 6, the flow label has been directly added to the format of the IPv6 datagram to allow us to use IPv6 as as connection-oriented protocol.
- To a router, a flow is a sequence of packets that share the same characteristics, such as traveling the same path, using the same resources, having the same kind of security, and so on. A router that supports the handling of flow labels has a flow label table.
- The table has an entry for each active flow label; each entry defines the services required by the corresponding flow label. When the router receives a packet, it consults its flow. It then provides the packet with the services mentioned in the entry.

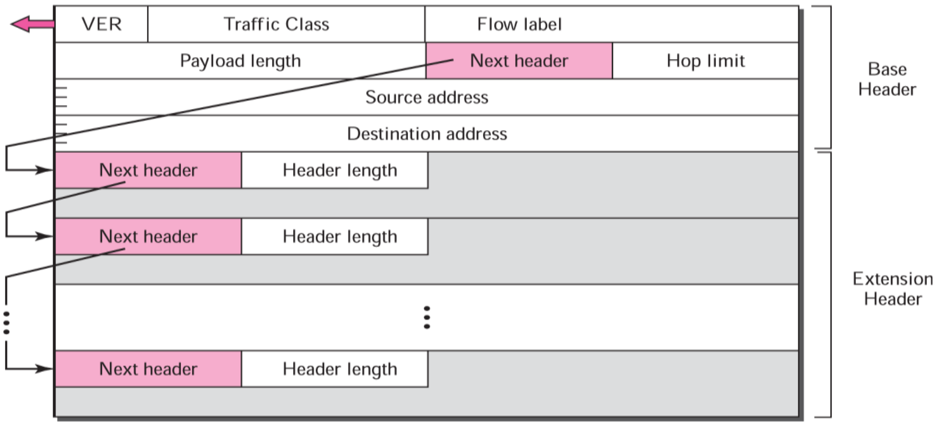
- In its simplest form, a flow label can be used to speed up the processing of a packet by a router. When a router receives a packet, instead of consulting the routing table and going through a routing algorithm to define the address of the next hop, it can easily look in a flow label table for the next hop label table to find the corresponding entry for the flow label value defined in the packet. It then provides the packet with the services mentioned in the entry.
- In its more sophisticated form, a flow label can be used to support the transmission of real-time audio and video. Real-time audio or video,

particularly in digital form, requires resources such as high bandwidth, large buffers, long processing time, and so on.

- The use of real-time data and the reservation of these resources require other protocols such as Real-Time Protocol (RTP) and Resource Reservation Protocol (RSVP) in addition to IPv6.
- To allow the effective use of flow labels, three rules have been defined:
  1. The flow label is assigned to a packet by the source host. The label is a random number between 1 and 2^{24}– 1. A source must not reuse a flow label for a new flow while the existing flow is still alive.
  2. If a host does not support the flow label, it sets this field to zero. If a router does not support the flow label, it simply ignores it.
  3. All packets belonging to the same flow have the same source, same destination, same priority, and same options.

IPv6 extension headers

- ✓ The length of the base header is fixed at 40 bytes. However, to give more functionality to the IP datagram, the base header can be followed by up to six extension headers. Many of these headers are options in IPv4. Figure shows the extension header format.



- ✓ Six types of extension headers have been defined. These are **hop-by-hop option, source routing, fragmentation, authentication, encrypted security payload, and destination option.**

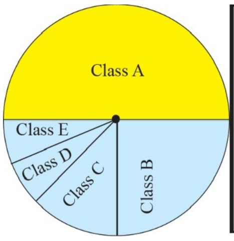
IP ADDRESS

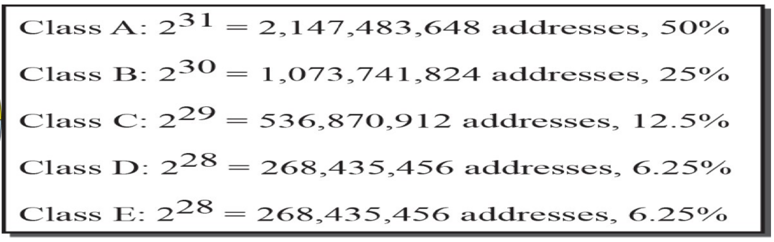
- ✓ An IPv4 address is 32 bits long.
- ✓ The IPv4 addresses are unique and universal.
- ✓ Address Space : The address space of IPv4 is 232 or 4,294,967,296.
- ✓ Notation :
  - Binary notation (base 2) – In binary notation, an IPv4 address is displayed as 32 bits. To make the address more readable, one or more spaces is usually inserted between each octet (8 bits). Each octet is often referred to as a byte.
    - 01110101 10010101 00011101 11101010
  - Dotted-decimal notation (base 256)- IPv4 address is usually written in decimal form with a decimal point (dot) separating the bytes.
    - 192.168.10.1

- Hexadecimal notation (base 16) - Each hexadecimal digit is equivalent to four bits. This means that a 32-bit address has 8 hexadecimal digits.
  - 0X810B0BEF

CLASSFUL ADDRESSES

- ✓ IP addresses, when started a few decades ago, used the concept of classes. This architecture is called Classful addressing.
- ✓ In the mid-1990s, a new architecture, called classless addressing, was introduced that supersedes the original architecture.
- ✓ Classes:
  - In Classful addressing, the IP address space is divided into five classes: A, B, C, D, and E. Each class occupies some part of the whole address space.





Class A: 2^{31} = 2,147,483,648 addresses, 50%

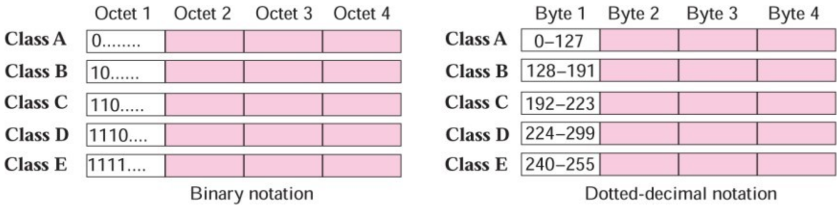
Class B: 2^{30} = 1,073,741,824 addresses, 25%

Class C: 2^{29} = 536,870,912 addresses, 12.5%

Class D: 2^{28} = 268,435,456 addresses, 6.25%

Class E: 2^{28} = 268,435,456 addresses, 6.25%

- ✓ Finding the class of address
  - Find the class of an address when the address is given either in binary or dotted-decimal notation.
  - In the binary notation, the first few bits can immediately tell us the class of the address.
  - In the dotted-decimal notation, the value of the first byte can give the class of an address.



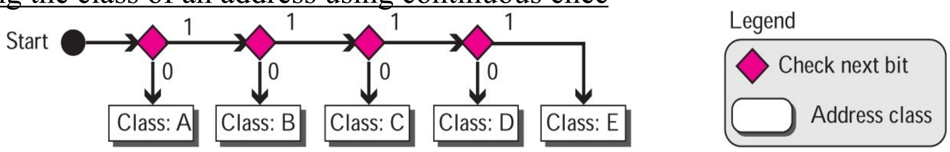
	Octet 1	Octet 2	Octet 3	Octet 4
Class A	0........			
Class B	10......			
Class C	110.....			
Class D	1110....			
Class E	1111....			

Binary notation

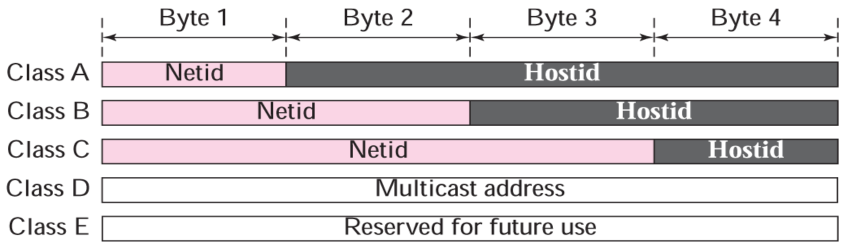
	Byte 1	Byte 2	Byte 3	Byte 4
Class A	0–127			
Class B	128–191			
Class C	192–223			
Class D	224–299			
Class E	240–255			

Dotted-decimal notation

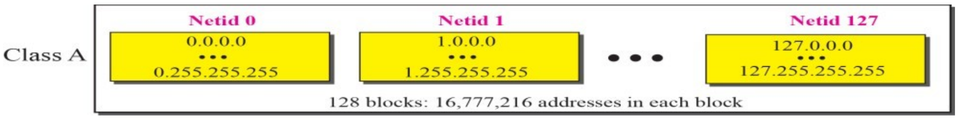
- ✓ Finding the class of an address using continuous chec



- ✓ Netid and Hostid
  - In classful addressing, an IP address in classes A, B, and C is divided into netid and hostid. These parts are of varying lengths, depending on the class of the address. Figure shows the netid and hostid bytes.
  - Classes D and E are not divided into netid and hostid.

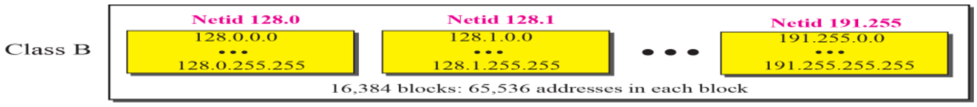


    - In class A, 1 byte defines the netid and 3 bytes define the hostid.
    - In class B, 2 bytes define the netid and 2 bytes define the hostid.
    - In class C, 3 bytes define the netid and 1 byte defines the hostid.
- ✓ Classes and Blocks
  - Each class is divided into a fixed number of blocks with each block having a fixed size.
  - **Class A:**
    - Only 1 byte in class A defines the netid and the leftmost bit should be 0, the next 7 bits can be changed to find the number of blocks in this class.
    - Therefore, class A is divided into 2^7 = 128 blocks that can be assigned to 128 organizations.
    - Each block in this class contains 16,777,216 addresses.
    - Many addresses are wasted in this class.

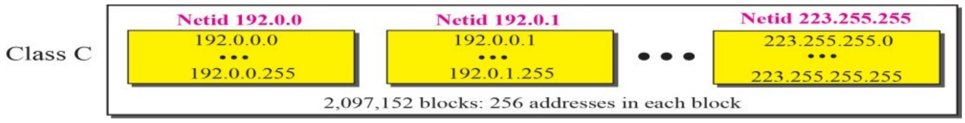


  - **Class B:**
    - Two bytes in class B defines the netid and the leftmost bit should be 10, the next 14 bits can be changed to find the number of blocks in this class.

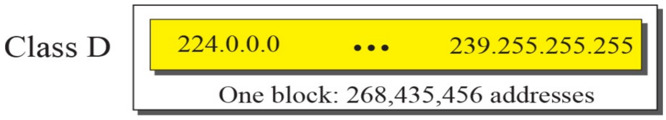
    - Therefore, class B is divided into 2^14 = 16,384 blocks that can be assigned to 16,384 organizations.
    - Each block in this class contains 65,536 addresses.
    - Many addresses are wasted in this class



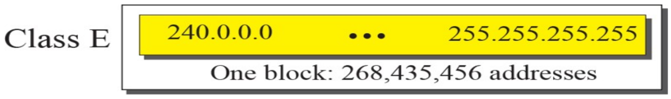
- **Class C**
    - Three bytes in class C defines the netid and the leftmost bit should be 110, the next 21 bits can be changed to find the number of blocks in this class.
    - Therefore, class B is divided into 2^21 = 2,097,152 blocks that can be assigned to 2,097,152 organizations.
    - Each block in this class contains 256 addresses.



- **Class D**
    - One block of class D addresses.
    - It is designed for multicasting.
    - Each address in this class is used to define one group of hosts on the Internet.
    - When a group is assigned an address in this class, every host that is a member of this group will have a multicast address in addition to its normal (unicast) address.



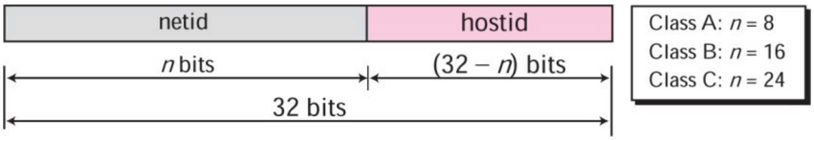
- **Class E**
    - One block of class E addresses.
    - It was designed for use as reserved addresses,



✓ Two-Level Addressing

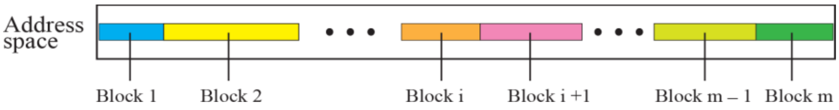
- The whole purpose of IPv4 addressing is to define a destination for an Internet packet.
- When classful addressing was designed, it was assumed that the whole Internet is divided into many networks and each network connects many hosts.
- A network was normally created by an organization that wanted to be connected to the Internet.
- The Internet authorities allocated a block of addresses to the organization (in class A, B, or C).
- Each address in classful addressing contains two parts: netid and hostid.

- The netid defines the network;
- The hostid defines a particular host connected to that network.

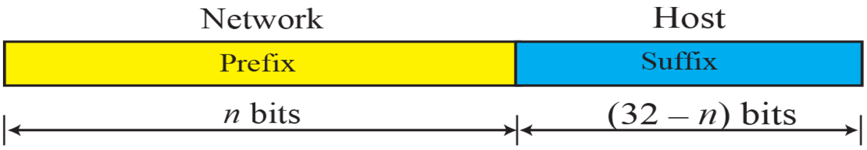


CLASSLESS ADDRESSING

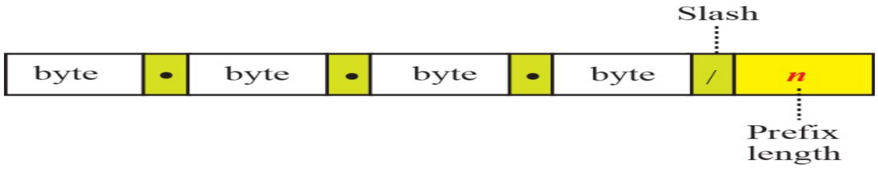
- ✓ Subnetting and super netting in classful addressing did not really solve the address depletion problem. With the growth of the Internet, it was clear that a larger address space was needed as a long-term solution. Although the long-range solution has already been devised and is called IPv6, a short-term solution was also devised to use the same address space but to change the distribution of addresses to provide a fair share to each organization. The short-term solution still uses IPv4 addresses, but it is called classless addressing.
- ✓ Variable-Length Blocks
  - In classless addressing, the whole address space is divided into variable length blocks.
- ✓ Number of Addresses in a Block
  - There is only one condition on the number of addresses in a block; it must be a power of 2 (2, 4, 8, . . .).
    - A household may be given a block of 2 addresses.
    - A small business may be given 16 addresses.
    - A large organization may be given 1024 addresses.



- ✓ Two-Level Addressing
  - In classful addressing, two-level addressing was provided by dividing an address into *netid and hostid.*
    - The netid defined the network
    - The hostid defined the host in the network.
  - In classless addressing, the block is actually divided into two parts, the prefix and the suffix.
    - The prefix plays the same role as the netid
    - The suffix plays the same role as the hostid.
  - All addresses in the block have the same prefix; each address has a different suffix.
  - The prefix length in classless addressing can be 1 to 32.



- ✓ **Slash notation**
  - In classless addressing, we need to include the prefix length to each address if we need to find the block of the address.
  - In this case, the prefix length, n, is added to the address separated by a slash. The notation is informally referred to as slash notation.
  - The slash notation is formally referred to as classless interdomain routing or CIDR notation.



- ✓ **Example**

1. A small organization is given a block with the beginning address and the prefix length 205.16.37.24/29 (in slash notation). What is the range of the block?

Solution

The beginning address is 205.16.37.24. To find the last address we keep the first 29 bits and change the last 3 bits to 1s.
Beginning:11001111 00010000 00100101 00011000
Ending : 11001111 00010000 00100101 00011111
There are only 8 addresses in this block.

IPv6 addressing format

- ✓ An IPv6 address is made of 128 bits divided into eight 16-bits blocks. Each block is then converted into 4-digit Hexadecimal numbers separated by colon symbols.
- ✓ For example, given below is a 128 bit IPv6 address represented in binary format and divided into eight 16-bits blocks:

0010000000000001 0000000000000000 0011001000111000 1101111111100001
0000000001100011 0000000000000000 0000000000000000 1111111011111011

- ✓ Each block is then converted into Hexadecimal and separated by ':' symbol:

2001:0000:3238:DFE1:0063:0000:0000:FEFB

- ✓ Even after converting into Hexadecimal format, IPv6 address remains long. IPv6 provides some rules to shorten the address. The rules are as follows:
  - Rule.1: Discard leading Zero(es):
    - In Block 6, 0036, the leading two 0s can be omitted, such as (6th block):
    - 2001:0000:3238:DFE1:1263:36:0000:FEFB
  - Rule.2: If two of more blocks contain consecutive zeroes, omit them all and replace with double colon sign ::, such as (6th and 7th block):
    - 2001:0000:3238:DFE1:1263::FEFB

    - Consecutive blocks of zeroes can be replaced only once by :: so if there are still blocks of zeroes in the address, they can be shrunk down to a single zero, such as (2nd block):
        - 2001:0:3238:DFE1:1263::FEFB

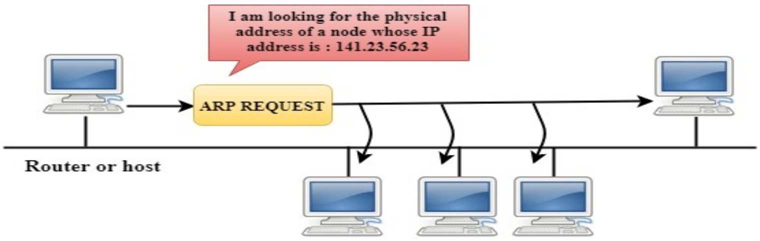
Internet Control Protocol:

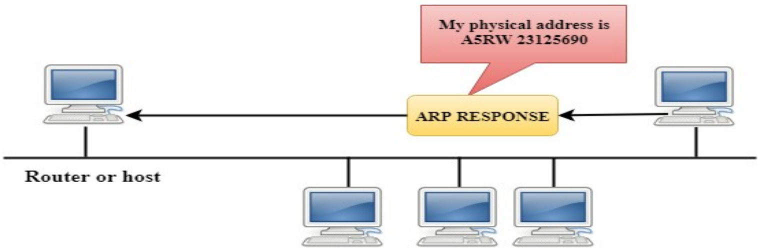
ARP

- ARP stands for Address Resolution Protocol.
- It is used to associate an IP address with the MAC address.
- Each device on the network is recognized by the MAC address imprinted on the NIC. Therefore, we can say that devices need the MAC address for communication on a local area network. MAC address can be changed easily. For example, if the NIC on a particular machine fails, the MAC address changes but IP address does not change. ARP is used to find the MAC address of the node when an internet address is known.

How ARP works

If the host wants to know the physical address of another host on its network, then it sends an ARP query packet that includes the IP address and broadcast it over the network. Every host on the network receives and processes the ARP packet, but only the intended recipient recognizes the IP address and sends back the physical address. The host holding the datagram adds the physical address to the cache memory and to the datagram header, then sends back to the sender.


I am looking for the physical address of a node whose IP address is : 141.23.56.23
ARP REQUEST
Router or host

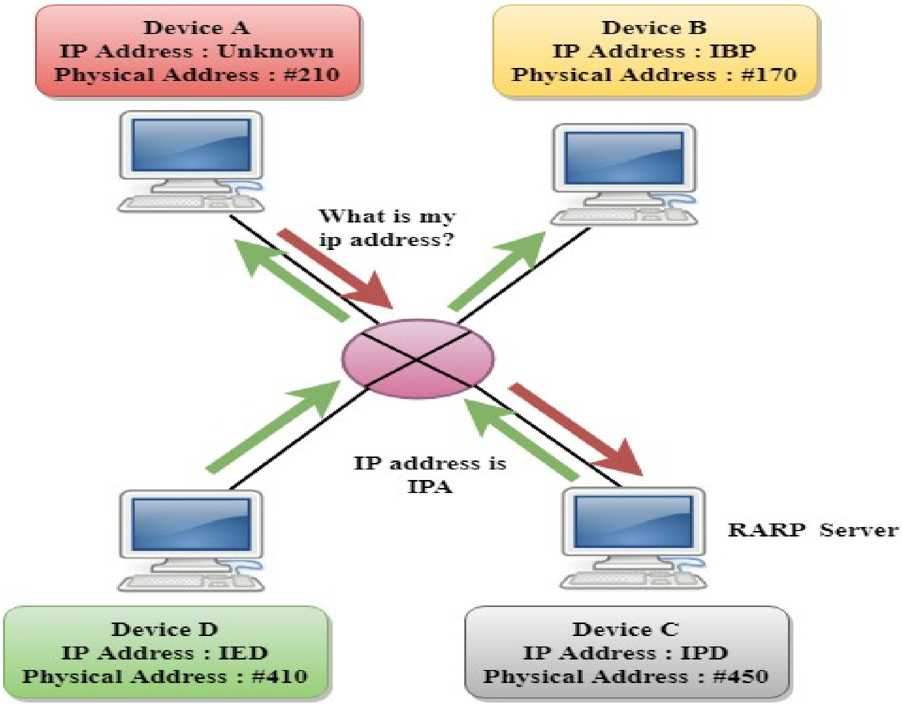


There are two types of ARP entries:

- **Dynamic entry:** It is an entry which is created automatically when the sender broadcast its message to the entire network. Dynamic entries are not permanent, and they are removed periodically.
- **Static entry:** It is an entry where someone manually enters the IP to MAC address association by using the ARP command utility.

RARP

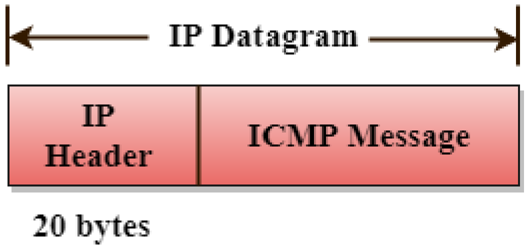
- RARP stands for **Reverse Address Resolution Protocol**.
- If the host wants to know its IP address, then it broadcast the RARP query packet that contains its physical address to the entire network. A RARP server on the network recognizes the RARP packet and responds back with the host IP address.
- The protocol which is used to obtain the IP address from a server is known as **Reverse Address Resolution Protocol**.
- The message format of the RARP protocol is similar to the ARP protocol.
- Like ARP frame, RARP frame is sent from one machine to another encapsulated in the data portion of a frame.



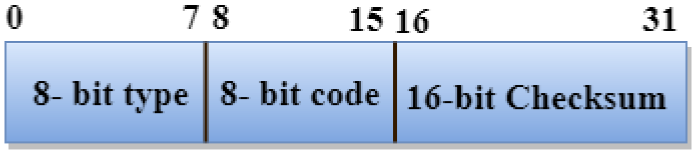
ICMP

- ICMP stands for Internet Control Message Protocol.

- The ICMP is a network layer protocol used by hosts and routers to send the notifications of IP datagram problems back to the sender.
- ICMP uses echo test/reply to check whether the destination is reachable and responding.
- ICMP handles both control and error messages, but its main function is to report the error but not to correct them.
- An IP datagram contains the addresses of both source and destination, but it does not know the address of the previous router through which it has been passed. Due to this reason, ICMP can only send the messages to the source, but not to the immediate routers.
- ICMP protocol communicates the error messages to the sender. ICMP messages cause the errors to be returned back to the user processes.
- ICMP messages are transmitted within IP datagram.



The Format of an ICMP message



- The first field specifies the type of the message.
- The second field specifies the reason for a particular message type.
- The checksum field covers the entire ICMP message.

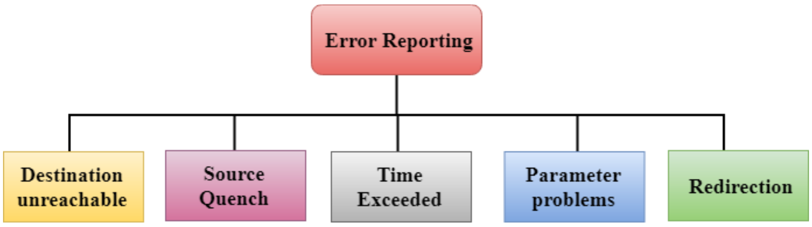
Error Reporting

ICMP protocol reports the error messages to the sender.

Five types of errors are handled by the ICMP protocol:

- Destination unreachable
- Source Quench
- Time Exceeded
- Parameter problems

- Redirection



- **Destination unreachable:** The message of "Destination Unreachable" is sent from receiver to the sender when destination cannot be reached, or packet is discarded when the destination is not reachable.
- **Source Quench:** The purpose of the source quench message is congestion control. The message sent from the congested router to the source host to reduce the transmission rate. ICMP will take the IP of the discarded packet and then add the source quench message to the IP datagram to inform the source host to reduce its transmission rate. The source host will reduce the transmission rate so that the router will be free from congestion.
- **Time Exceeded:** Time Exceeded is also known as "Time-To-Live". It is a parameter that defines how long a packet should live before it would be discarded.

There are two ways when Time Exceeded message can be generated:

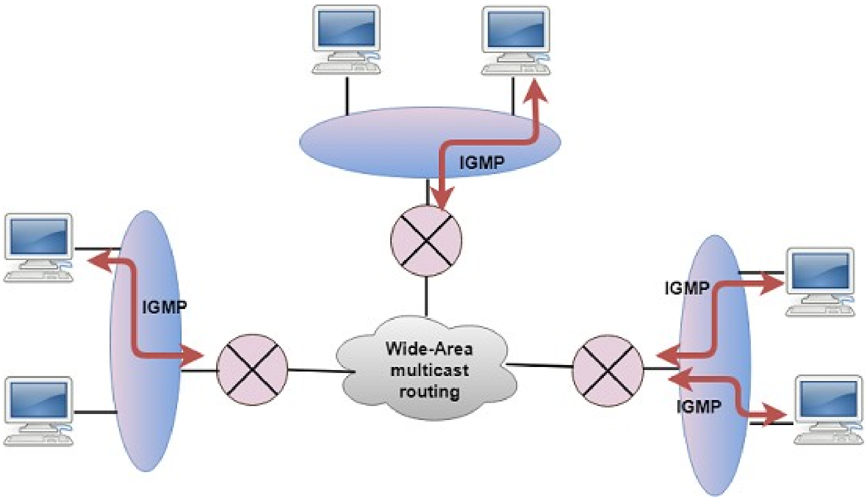
Sometimes packet discarded due to some bad routing implementation, and this causes the looping issue and network congestion. Due to the looping issue, the value of TTL keeps on decrementing, and when it reaches zero, the router discards the datagram. However, when the datagram is discarded by the router, the time exceeded message will be sent by the router to the source host.

When destination host does not receive all the fragments in a certain time limit, then the received fragments are also discarded, and the destination host sends time Exceeded message to the source host.

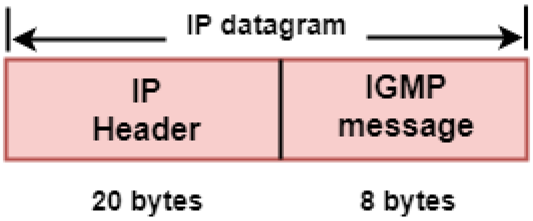
- **Parameter problems:** When a router or host discovers any missing value in the IP datagram, the router discards the datagram, and the "parameter problem" message is sent back to the source host.
- **Redirection:** Redirection message is generated when host consists of a small routing table. When the host consists of a limited number of entries due to which it sends the datagram to a wrong router. The router that receives a datagram will forward a datagram to a correct router and also sends the "Redirection message" to the host to update its routing table.

IGMP

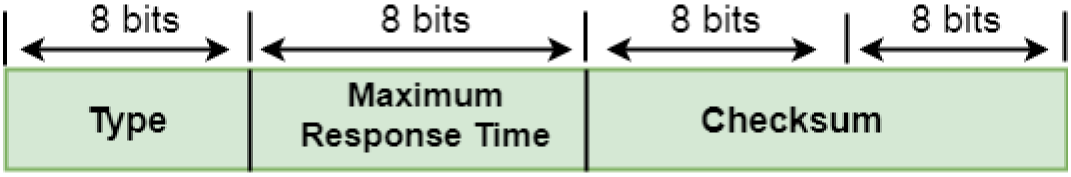
- IGMP stands for **Internet Group Message Protocol**.
- The IP protocol supports two types of communication:
  - **Unicasting:** It is a communication between one sender and one receiver. Therefore, we can say that it is one-to-one communication.
  - **Multicasting:** Sometimes the sender wants to send the same message to a large number of receivers simultaneously. This process is known as multicasting which has one-to-many communication.
- The IGMP protocol is used by the hosts and router to support multicasting.
- The IGMP protocol is used by the hosts and router to identify the hosts in a LAN that are the members of a group.



- IGMP is a part of the IP layer, and IGMP has a fixed-size message.
- The IGMP message is encapsulated within an IP datagram.



The Format of IGMP message



Where,

Type: It determines the type of IGMP message. There are three types of IGMP message: Membership Query, Membership Report and Leave Report.

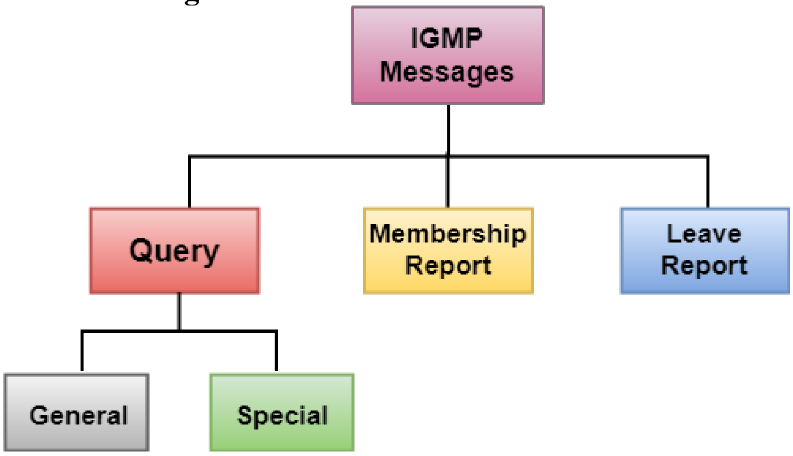
Maximum Response Time: This field is used only by the Membership Query message. It determines the maximum time the host can send the Membership Report message in response to the Membership Query message.

Checksum: It determines the entire payload of the IP datagram in which IGMP message is encapsulated.

Group Address: The behavior of this field depends on the type of the message sent.

- **For Membership Query**, the group address is set to zero for General Query and set to multicast group address for a specific query.
- **For Membership Report**, the group address is set to the multicast group address.
- **For Leave Group**, it is set to the multicast group address.

IGMP Messages



- **Membership Query message**
  - This message is sent by a router to all hosts on a local area network to determine the set of all the multicast groups that have been joined by the host.
  - It also determines whether a specific multicast group has been joined by the hosts on a attached interface.
  - The group address in the query is zero since the router expects one response from a host for every group that contains one or more members on that host.
- **Membership Report message**
  - The host responds to the membership query message with a membership report message.
  - Membership report messages can also be generated by the host when a host wants to join the multicast group without waiting for a membership query message from the router.

- Membership report messages are received by a router as well as all the hosts on an attached interface.
- Each membership report message includes the multicast address of a single group that the host wants to join.
- IGMP protocol does not care which host has joined the group or how many hosts are present in a single group. It only cares whether one or more attached hosts belong to a single multicast group.
- The membership Query message sent by a router also includes a "**Maximum Response time**". After receiving a membership query message and before sending the membership report message, the host waits for the random amount of time from 0 to the maximum response time. If a host observes that some other attached host has sent the "**Maximum Report message**", then it discards its "**Maximum Report message**" as it knows that the attached router already knows that one or more hosts have joined a single multicast group. This process is known as feedback suppression. It provides the performance optimization, thus avoiding the unnecessary transmission of a "**Membership Report message**".
- **Leave Report**

  When the host does not send the "Membership Report message", it means that the host has left the group. The host knows that there are no members in the group, so even when it receives the next query, it would not report the group.