UNIT IV ROUTING

Routing and protocols: Unicast routing – Distance Vector Routing – RIP – Link State Routing – OSPF-– Path-vector routing – BGP – Multicast Routing: DVMRP – PIM.

1.ROUTING AND PROTOCOLS

Routing is the process of moving information from a source to a destination across the internetwork. Typically, at least one intermediary node is encountered along the path.

Routing protocols are mechanisms for exchanging routing information between routers to make routing decisions. Routing protocols can facilitate effective and efficient communication between computer networks.

A router consults a routing table when a packet is ready to be forwarded

• The routing table specifies the optimum path for the packet: static or dynamic

• Internet needs dynamic routing tables to be updated as soon as there is a change

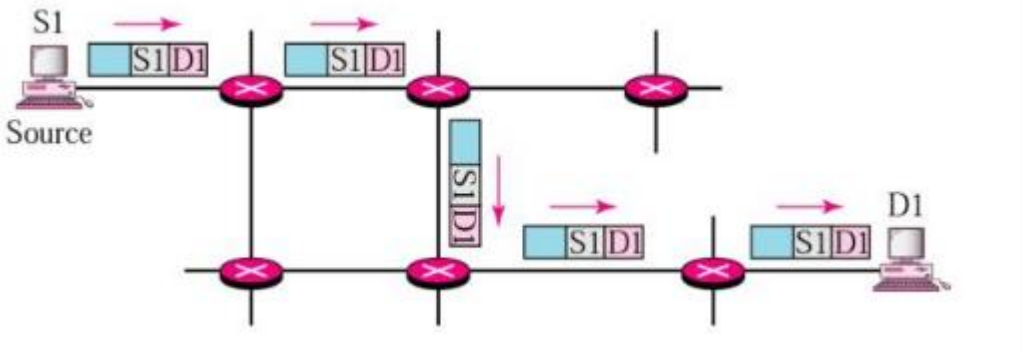
• Routing protocols is a combination of rules and procedures for dynamic routing tables

• The routing protocols also include procedures for combining information received from other routers

• Unicast routing and multicasting routing

2.UNICAST ROUTING

In unicast routing, the router forwards the received packet through only one of its ports.



Metric

Metric: a cost assigned for passing through a network

• Total metric is equal to the sum of the metrics of networks that comprise the route

• Router chooses the route with shortest (smallest) metric

• RIP (Routing Information Protocol): hop count

• OSPF (Open Shortest Path First): allows administrator to assign a cost based on the type of service required

• BGP (Border Gateway Protocol): criterion is the policy

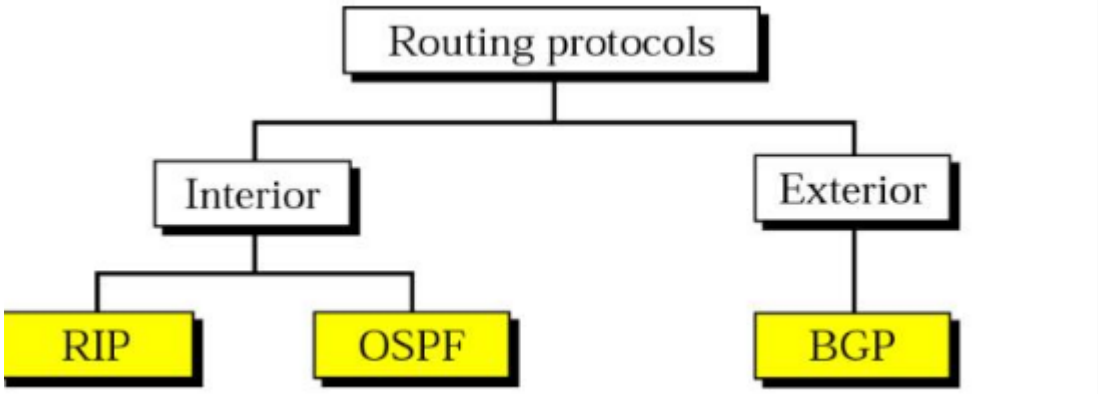
Interior and exterior routing

• AS (autonomous system): A group of networks and routers under the authority of a single administration

• Interior: inside an AS

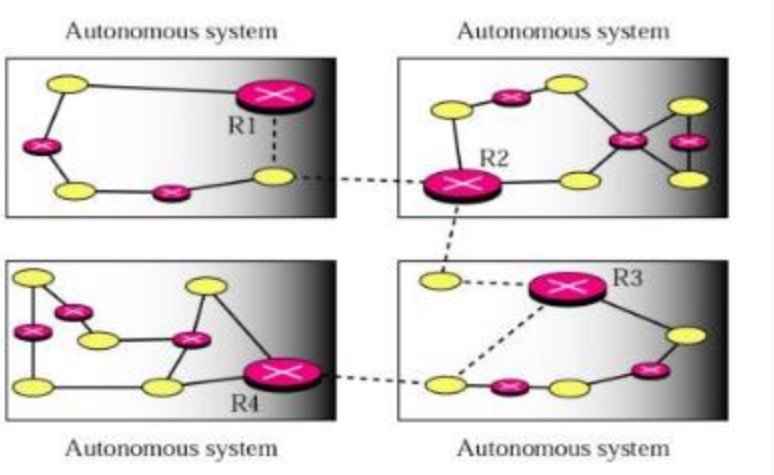
• Exterior: between ASs Computer Networks

Unicast routing protocols

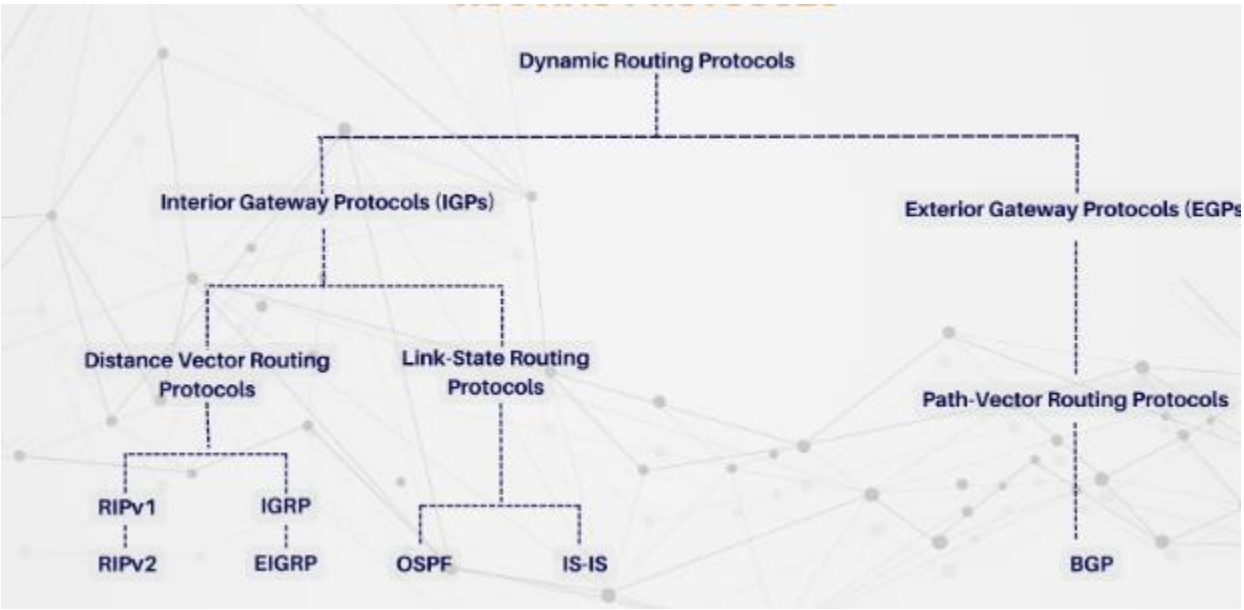


Interior and exterior routing

R1, R2, R3, and R4 use a interior and an exterior routing protocol.



- The other routes use only interior routing protocols



3.DISTANCE-VECTOR ROUTING

The **distance-vector (DV) routing** uses the goal we discussed in the introduction, to find the best route. In distance- vector routing, the first thing each node creates is its own least-cost tree with the rudimentary information it has about its immediate neighbors. The incomplete trees are exchanged between immediate neighbors to make the trees more and more complete and to represent the whole internet. We can say that in distance-vector routing, a router continuously tells all of its neighbors what it knows about the whole internet (although the knowledge can be incomplete).

Bellman-Ford Equation

The heart of distance-vector routing is the famous **Bellman-Ford** equation. This equationis used to find the least cost (shortest distance) between a source node, ***x***, and a destination node, ***y***, through some intermediary nodes (**a, b,c, ...**) when the costs between the source and the intermediary nodes

and the least costs between the intermediary nodes and the destination are given. The following shows the general case in which D***ij*** is the shortest distance and***cij*** is the cost between nodes ***i*** and ***j***.

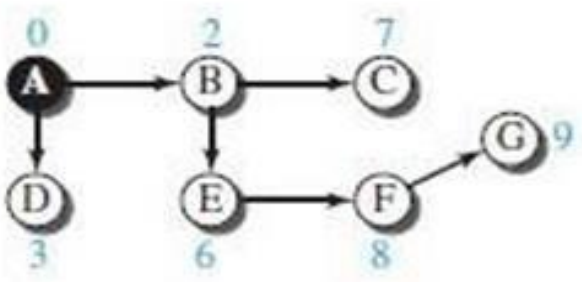
$$D_{xy} = \min\{(c_{xa} + D_{ay}), (c_{xb} + D_{by}), (c_{xc} + D_{cy}), \ldots\}$$

In distance-vector routing, normally we want to update an existing least cost with a leastcost through an intermediary node, such as ***z***, if the latter is shorter. In this case, the equation becomes simpler, as shown below:

$$D_{xy} = \min\{D_{xy}, (c_{xz} + D_{zy})\}$$

Distance Vectors

The concept of a **distance vector** is the rationale for the name *distance-vector routing*. A least-cost tree is a combination of least-cost paths from the root of the tree to all destinations. These paths are graphically glued together to form the tree. Distance- vector routing unglues these paths and creates a *distance vector*, a one- dimensional array to represent the tree.



a. Tree for node A

	A
A	0
B	2
C	7
D	3
E	6
F	8
G	9

b. Distance vector for node A

ALGORITHM

At each node x,

Initialization

for all destinations y in N:

Dx(y) = c(x,y) // If y is not a neighbor then c(x,y) = ∞for eachneighbor w

Dw(y) = ? for all destination y in N.foreach neighbor w

send distance vector Dx = [Dx(y) : y in N] to w

loop

wait(until I receive any distance vector fromsome neighbor w)for each y in N:

Dx(y) = minv{c(x,v)+Dv(y)}

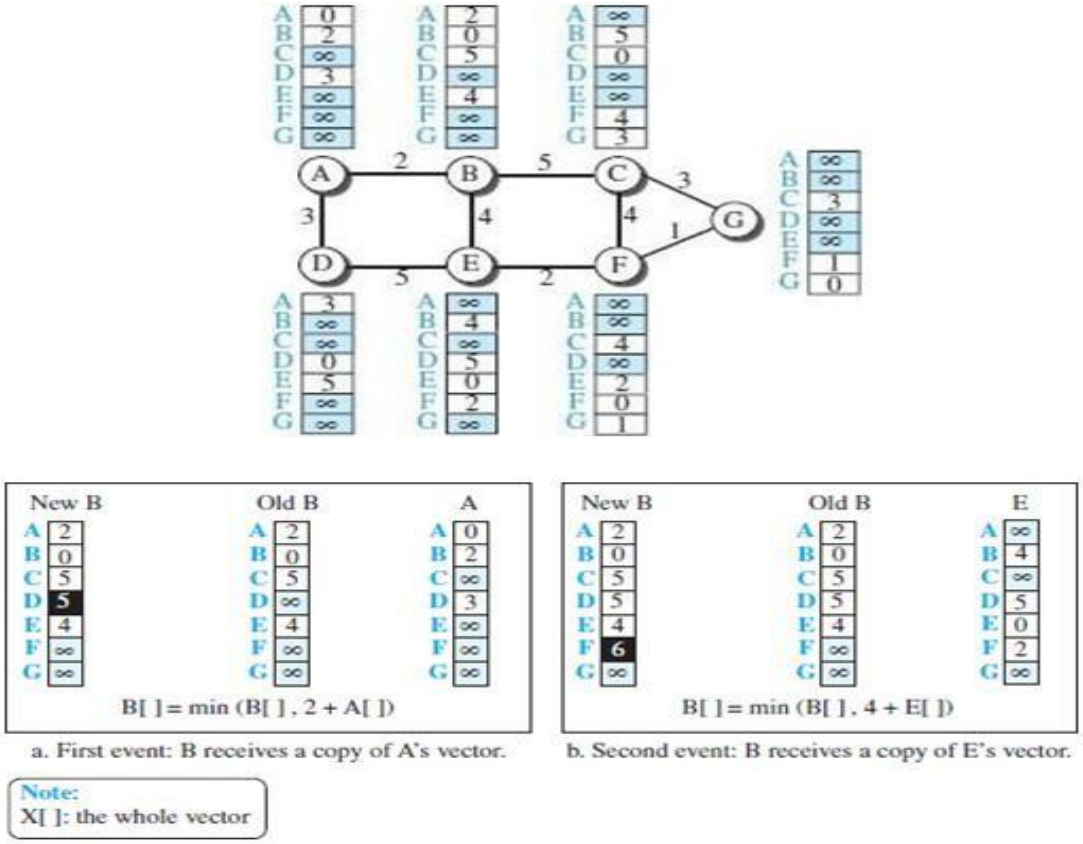
If Dx(y) is changed for any destination y

Send distance vector Dx = [Dx(y) : y in N] to all neighbors

forever

When we know only one distance between two nodes, it is the least cost. Figure below shows all

distance vectors for our internet. However, we need to mention that these vectors are made asynchronously, when the corresponding node has been booted; the existence of all of them in a figure does not mean synchronous creation of them. For example, node A thinks that it is not connected to node G because the corresponding cell shows the least cost of infinity. To improve these vectors, the nodes in the internet need to help each other by exchanging information. After each node has created its vector, it sends a copy of the vector to all its immediate neighbors.



a. First event: B receives a copy of A's vector.

b. Second event: B receives a copy of E's vector.

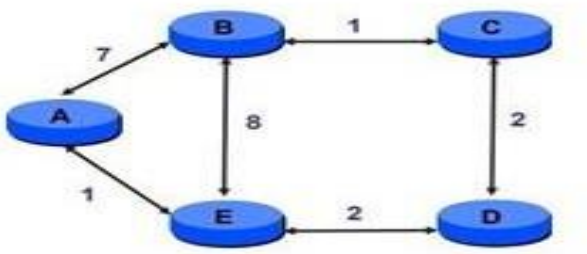
Note:
X[ ]: the whole vector

In the first event, node A has sent its vector to node B. Node B updates its vector using the cost cBA =2. In the second event, node E has sent its vector to node B. Node B updates its vector using the cost cEA =4. After the first event, node B has one improvement in its vector: its least cost to node D has changed from infinity to 5 (via node A).After the second event, node B has one more improvement in its vector; its least cost to node F has changed from infinity to 6 (via node E).

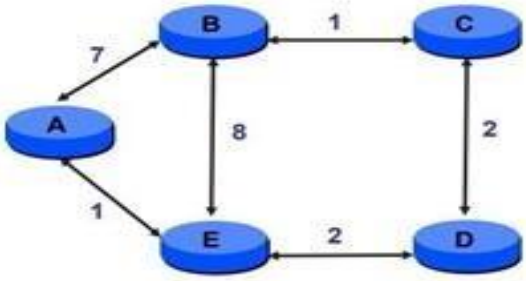
Count to Infinity

A problem with distance-vector routing is that any decrease in cost (good news) propagates quickly, but any increasein cost (bad news) will propagate slowly. For a routingprotocol to work properly, if a link is broken (cost becomes infinity), every other router should be aware of it immediately, but in distance-vector routing, this takes some time. Theproblem is referred to as *count to infinity*.

Example:



Info at node	Distance to Node				
	A	B	C	D	E
A	0	7	∞	∞	1
B	7	0	1	∞	8
C	∞	1	0	2	∞
D	∞	∞	2	0	2
E	1	8	∞	2	0

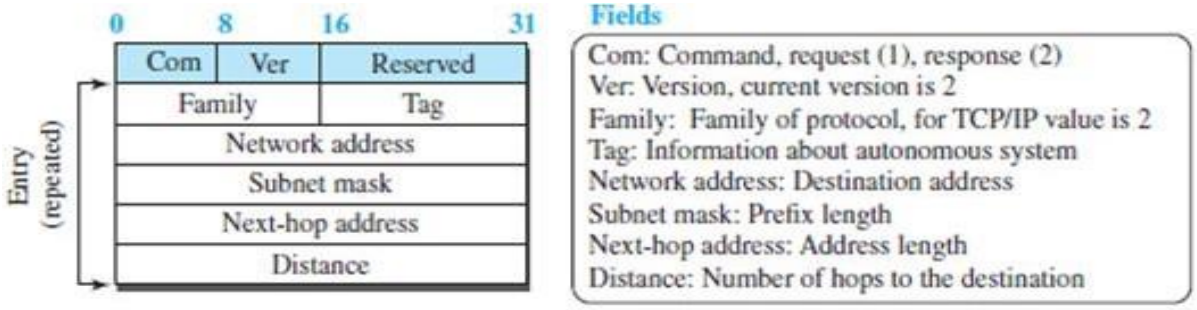


Info at node	Distance to Node				
	A	B	C	D	E
A	0	6	5	3	1
B	6	0	1	3	5
C	5	1	0	2	4
D	3	3	2	0	2
E	1	5	4	2	0

4.ROUTING INFORMATION PROTOCOL (RIP)

The **Routing Information Protocol** (**RIP**) is one of the most widely used intra domain routing protocols based on the distance-vector routing algorithm RIP routers advertise the cost of reaching different networks instead of reaching other nodes in a theoretical graph. In other words, the cost is defined between a router and the network in which thedestination host is located. Second, to make the implementation of the cost simpler. A forwarding table in RIP is a three-column table in which the first column is the addressof the destination network, the second column is the address of the next router to which the packet should be forwarded, and the third column is the cost (the number of hops)

to reach the destination network. RIP has gone through two versions: RIP-1 and RIP-2. The second version is backward compatible with the first section; it allows the use of more information in the RIP messages that wereset to 0 in the first version.



RIP has two types of messages: request and response. A request message is sent by a router that has just come up or by a router that has some time-out entries. A request message can ask about specific entries or all entries. A response(or update) message can beeither solicited or unsolicited. A solicited response message is sent only in answer to a request message. It contains information about the destination specified in the corresponding request message.

5.LINK-STATE ROUTING

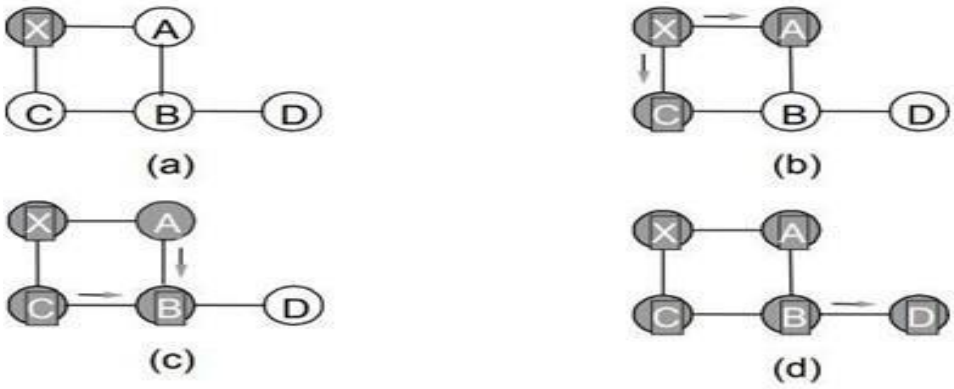
A routing algorithm that directly follows our discussion for creating least-cost trees and forwarding tables is **link- state (LS) routing.** This method uses the term link-state to define the characteristic of a link (an edge) that represents a network in the internet. Inthis algorithm the cost associated with an edge defines the state of the link. Links with lower costs are preferred to links with higher costs; if the cost of a link is infinity, it means that the link does not exist or has been broken.

Link-State Database (LSDB)

To create a least-cost tree with this method, each node needs to have a complete map of thenetwork, which means it needs to know the state of each link. The collection of states for alllinks is called the **link-state database** (**LSDB**)**.** Each node can send some greeting messages to all its immediate neighbors (those nodes to which it is connected directly) to collect two pieces of information for each neighboring node: the identity of the node andthe cost of the link. The combination of these two pieces ofinformation is called the LS packet (LSP).

Flooding

Each router sends link-state information out its links The next node sends it out through allof its links Except theone where the information arrived



Formation of Least-Cost Trees

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

This iterative algorithm uses the following steps:

1. The node chooses itself as the root of the tree, creating a tree with a single node, and sets the total cost of each node based on the information in the LSDB.

2. The node selects one node, among all nodes not in the tree, which is closest to the root, and adds this to the tree. After this node is added to the tree, the cost of all other nodes not in the tree needs to be updated because the paths may have been changed.

3. The node repeats step 2 until all nodes are added to the tree.The forward search algorithm:

1. Initialize the Confirmed list with an entry for myself; this entry has a cost of 0.

2. For the node just added to the Confirmed list in the previous step, call it node Next and select its LSP.

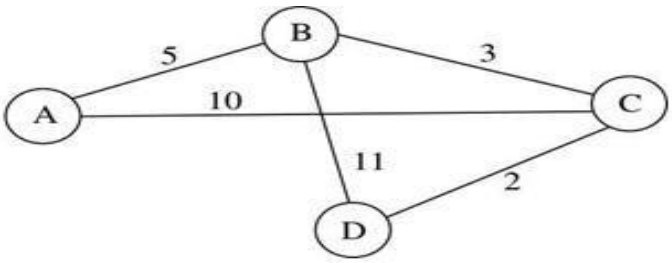
3. For each neighbor (Neighbor) of Next, calculate the cost (Cost) to reach this Neighbor as the sum of the cost from myself to Next and from Next to Neighbor.

a) If Neighbor is currently on neither the Confirmed nor the Tentative list, then add (Neighbor, Cost, Nexthop) to the Tentative list, where Nexthop is the direction I go to reach Next

b) If Neighbor is currently on the Tentative list, and the Cost is less than the currently listed cost for the Neighbor, then replace the current entry with (Neighbor, Cost, Nexthop) where Nexthop is the direction I go to reach Next

4. If the Tentative list is empty, stop.

Otherwise, pick the entry from the Tentative list with the lowest cost, move it to the Confirmed list, and return to Step 2.



Step	Confirmed	Tentative	Comments
1	(D,0,–)		Since D is the only new member of the confirmed list, look at its LSP.
2	(D,0,–)	(B,11,B) (C,2,C)	D's LSP says we can reach B through B at cost 11, which is better than anything else on either list, so put it on Tentative list; same for C.
3	(D,0,–) (C,2,C)	(B,11,B)	Put lowest-cost member of Tentative (C) onto Confirmed list. Next, examine LSP of newly confirmed member (C).
4	(D,0,–) (C,2,C)	(B,5,C) (A,12,C)	Cost to reach B through C is 5, so replace (B,11,B). C's LSP tells us that we can reach A at cost 12.
5	(D,0,–) (C,2,C) (B,5,C)	(A,12,C)	Move lowest-cost member of Tentative (B) to Confirmed, then look at its LSP.
6	(D,0,–) (C,2,C) (B,5,C)	(A,10,C)	Since we can reach A at cost 5 through B, replace the Tentative entry.
7	(D,0,–) (C,2,C) (B,5,C) (A,10,C)		Move lowest-cost member of Tentative (A) to Confirmed, and we are all done.

6.OPEN SHORTEST PATH FIRST (OSPF)

Open Shortest Path First (**OSPF**) is also an intradomain routing protocol like RIP, butit is based on the link-staterouting protocol

Metric

In OSPF, like RIP, the cost of reaching a destination from the host is calculated from the source router to the destination network. However, each link (network) can be assigned aweight based on the throughput, round-trip time, reliability, and so on. An administrationcan also decide to use the hop count as the cost. An interesting point about the cost in OSPF is that different service types (TOSs) can have different weights as the cost.

Link-State Advertisement

OSPF is based on the link-state routing algorithm, which requires that a router advertise the state of each link to all neighbors for the formation of the LSDB. When we discussed the link-state algorithm, we used the graph theory and assumed that each router is a node and each network between two routers is an edge. The situation is different in the real world, in which we need to advertise the existence of different entities as nodes, the different types of links that connect each node to its neighbors, and the different types of cost associated with each link.

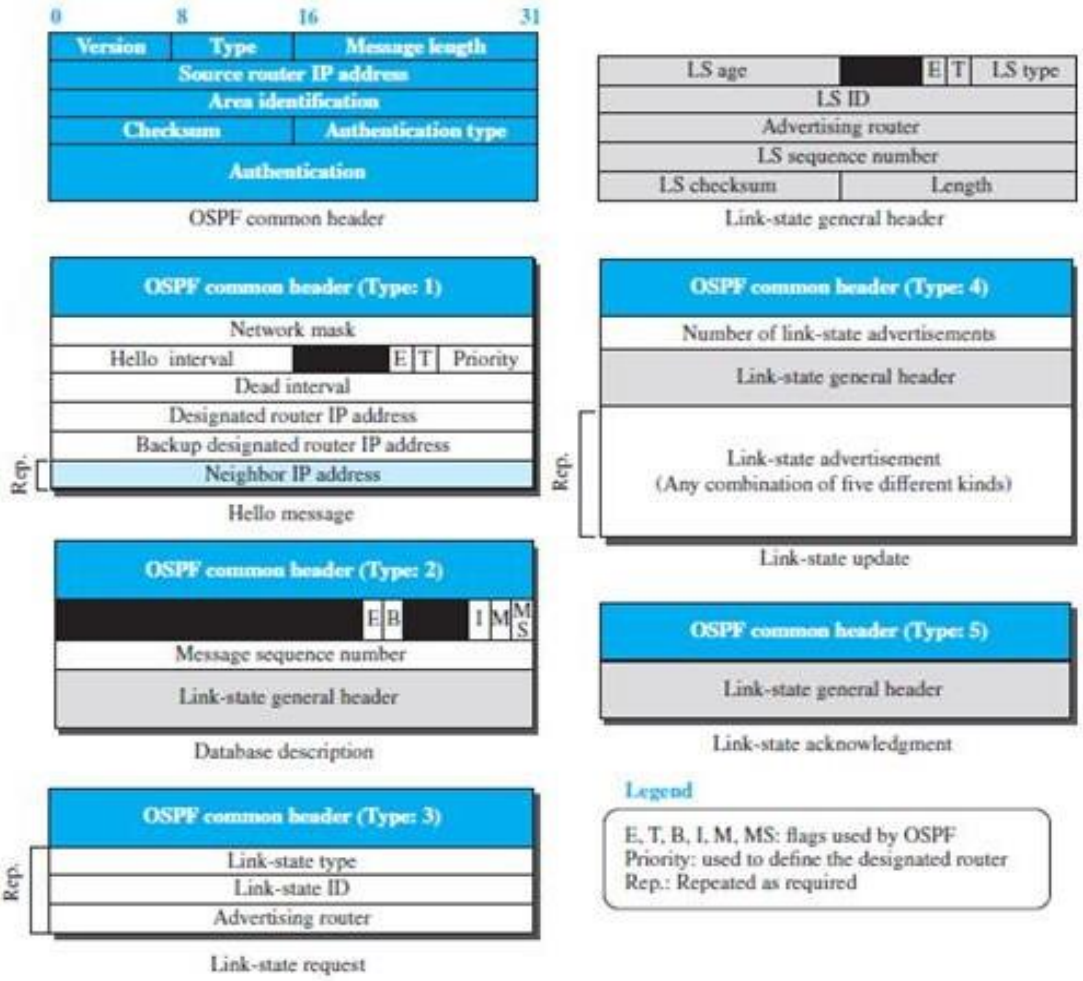
OSPF Implementation

OSPF is implemented as a program in the network layer, using the service of the IP for propagation. An IP datagram that carries a message from OSPF sets the value of the protocol field to 89.

OSPF Messages – OSPF is a very complex protocol. It uses five different types of messages.These are as follows:

1. **Hello message (Type 1) –** It is used by the routers to introduce itself to the other routers.
2. **Database description message (Type 2) –** It is normally send in response to the Hello message.
3. **Link-state request message (Type 3) –** It is used by the routers that needinformation about specific Link-State packet.
4. **Link-state update message (Type 4) –** It is the main OSPF message for building Link-State Database.
5. **Link-state acknowledgement message (Type 5) –** It is used to create reliability in theOSPF protocol.Basic link state algorithm plus many features:

- Authentication of routing messages
- Extra hierarchy: Partition into routing areas » “Border” router pretends to be directlyconnected to all routers inan area (answers for them)
- Load balancing: Multiple equal cost routes



7.PATH-VECTOR ROUTING

Path Vector Routing is a routing algorithm in unicast routing protocol of network layer, and it is useful for interdomain routing. The principle of path vector routing is similar to that of distance vector routing. It assumes that there is one node in each autonomous system that acts on behalf of the entire autonomous system is called Speaker node .The speaker node in an AS creates a routing cable and advertises to the speaker node in the neighbouring asAS.A speaker node advertises the path, not the metrics of the nodes, in its autonomous system or other autonomous systems

It is the initial table for each speaker node in a system made four ASs. Here Node A1 is thespeaker node for AS1, B1 for AS2, C1 for AS3 and D1 for AS4, Node A1 creates an initial table that shows A1 to A5 and these are located in AS1, it can be reached through it.

A speaker in an autonomous system shares its table with immediate neighbours ,here Node A1 share its table with nodes B1 and C1 , Node C1 share its table with nodes A1,B1 and D1 , Node B1 share its table with nodes A1 andC1 , Node D1 share its table with node C1 If router A1 receives a packet for nodes A3 , it knows that the path is in AS1,but if it receives a packet for D1,it knows that the packet should go from AS1,to AS2 and then to AS3 ,then the routing table shows that path completely on the other hand if the node D1 in AS4 receives a packet for node A2,it knows it should go through AS4,AS3,and AS.

FUNCTIONS

- PREVENTION OF LOOP The creation of loop can be avoided in path vector routing .A router

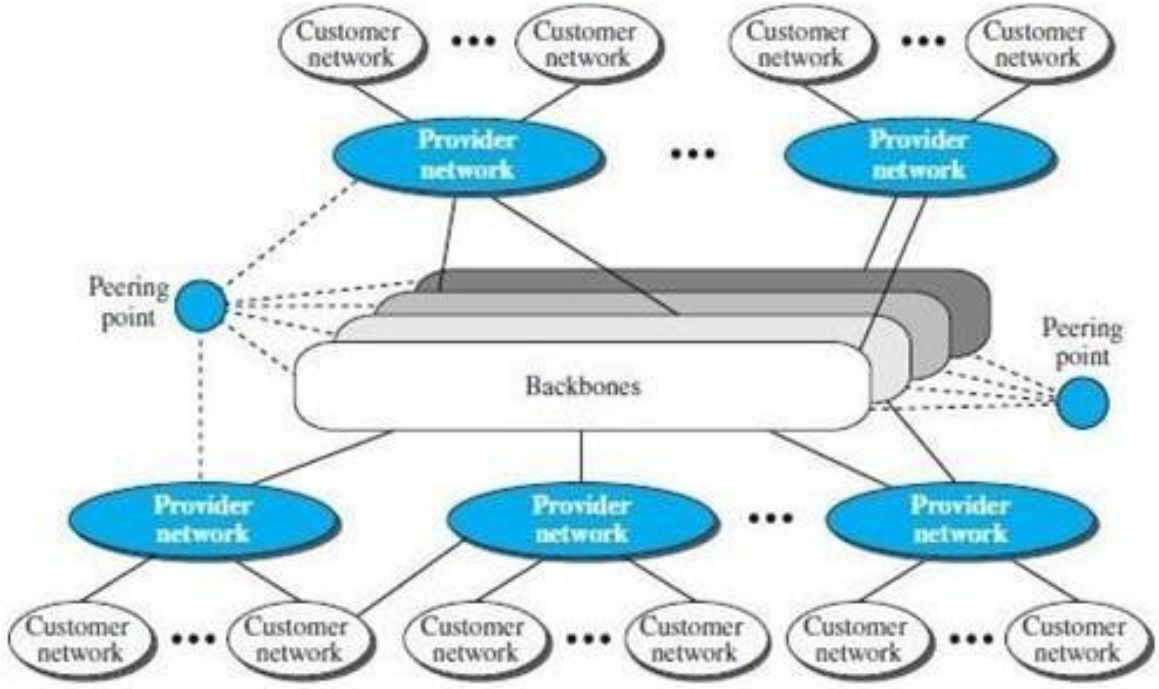
receives a messageit checks to see if its autonomous system is in the path list to the destination if it is looping is involved and the message is ignored

- POLICY ROUTING When a router receives a messages it can check the path, if one of the autonomous system listed in the path against its policy, it can ignore its path and destination it does not update its routing table with this path or it does not send the messages to its neighbours.

- OPTIMUM PATH A path to a destination that is the best for the organization that runs the autonomous system

Internet Structure

The Internet has changed from a tree-like structure, with a single backbone, to a multi- backbone structure run by different private corporations today. There are several backbones run by private communication companies that provide global connectivity. These backbones are connected by some peering points that allow connectivity between backbones. At a lower level, there are some provider networks that use the backbones for global connectivity but provide services to Internet customers. Finally, there are some customer networks that use the services provided by the provider networks. Any of these three entities (backbone, provider network, or customer network) can be called an Internet Service Provider or ISP. They provide services, but at different levels. Hierarchical routing means considering each ISP as an **autonomous system (AS).**Each AS can run a routing protocol that meets its needs, but the global Internet runs a global protocol to glue all ASs together. The routing protocol run in each AS is referred to as intra-AS routing protocol, intradomain routing protocol, or interior gateway protocol (IGP); the global routing protocol is referred to as inter-AS routing protocol,interdomain routing protocol, or exterior gateway protocol (EGP).



The autonomous systems,however, are not categorized according to their size; they are categorized according to the way they are connected to other ASs. We have stub ASs, multihomed ASs, and transient ASs.

Stub AS. A stub AS has only one connection to another AS. The data traffic can be either initiated or terminated ina stub AS; the data cannot pass through it. A good example of a stub AS is the customer network, which is either the source or the sink of data.

Multihomed AS. A multihomed AS can have more than one connection to other ASs, but it does not allow data traffic to pass through it. A good example of such an AS is some of the customer ASs that may use the services of more than one provider
network, but their policy does not allow data to be passed through them.

Transient AS. A transient AS is connected to more than one other AS and also allows the traffic to pass through. The provider networks and the backbone are good examples of transient ASs.

8.BGP – BORDER GATEWAY PROTOCOL VERSION 4 (BGP4)

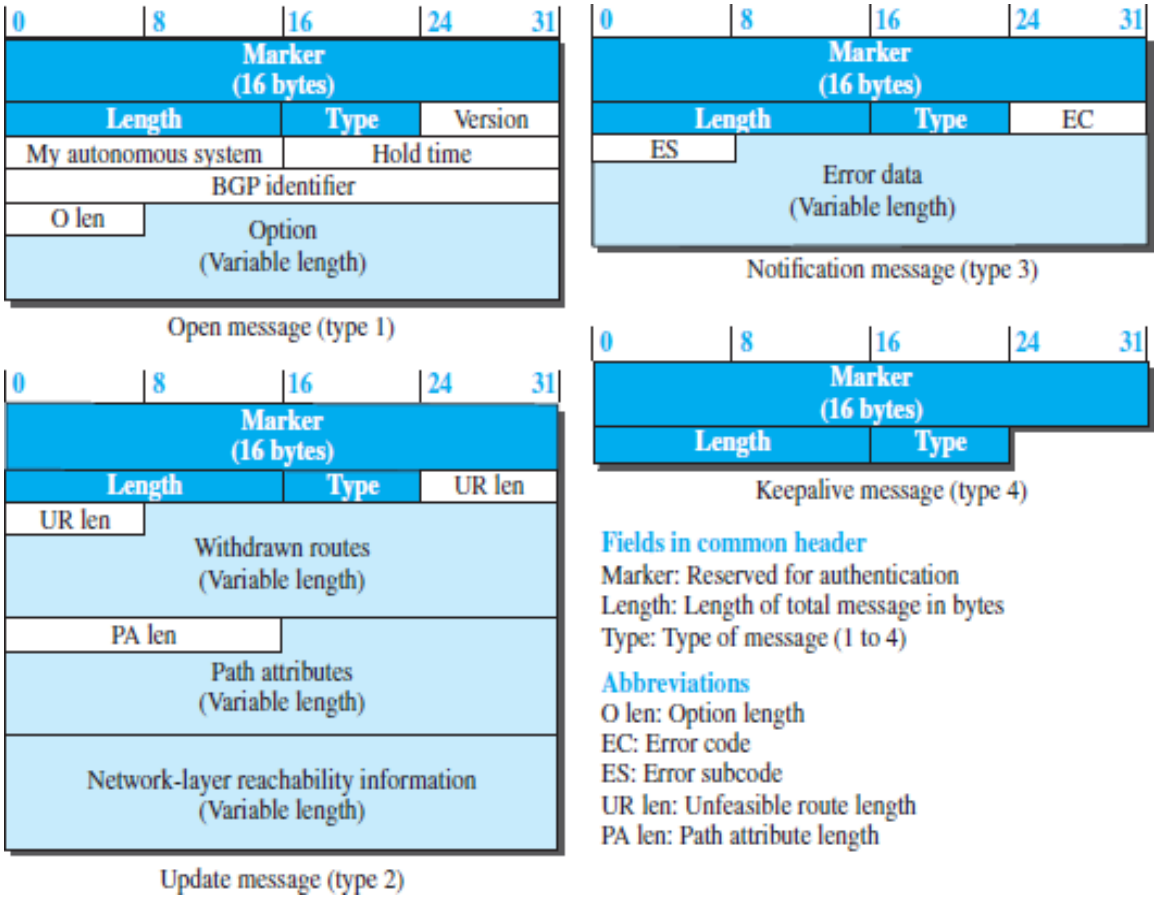
The **Border Gateway Protocol version 4 (BGP4)** is the only interdomain routing protocol used in the Internet today. BGP4 is based on the path-vector algorithm wedescribed before, but it is tailored to provide information about the reachability of networks in the Internet. BGP routing information is usually exchanged between competingbusiness entities in the form of internet service providers (ISPs) in an open, hostile environment -- the public internet. BGP is very security-focused. For example, all adjacent routers have to be configured manually based on routing policies. To permit the transfer of routing information between neighboring ISPs, BGP requires peering agreements, which comprise the necessary terms and conditions for exchanging traffic. The protocol adheres to these agreements, while also evaluating routing tables and information along multiple routes between ISPs. All other routing protocols are concerned solely with finding the optimal path toward all known destinations. BGP can't take this simplistic approach because the peering agreements between ISPs almost always result in complex routing policies.

Messages

BGP uses four types of messages for communication between the BGP speakers across theASs and inside an AS:
open, update, keepalive, and notification. All BGP packets share the same common header.

- **Open Message.** To create a neighborhood relationship, a router running BGP opens a TCP connection with a neighbor and sends an open message.
- **Update Message.** The update message is the heart of the BGP protocol. It is used by a router to withdraw destinations that have been advertised previously, to announce a route to a new destination, or both. Note that BGP can withdraw several destinations that were advertised before, but it can only advertise one new destination (or multiple destinations with the same path attributes) in a single update message.
- **Keepalive Message.** The BGP peers that are running exchange keepalive messagesregularly (before their hold time expires) to tell each other that they are alive.

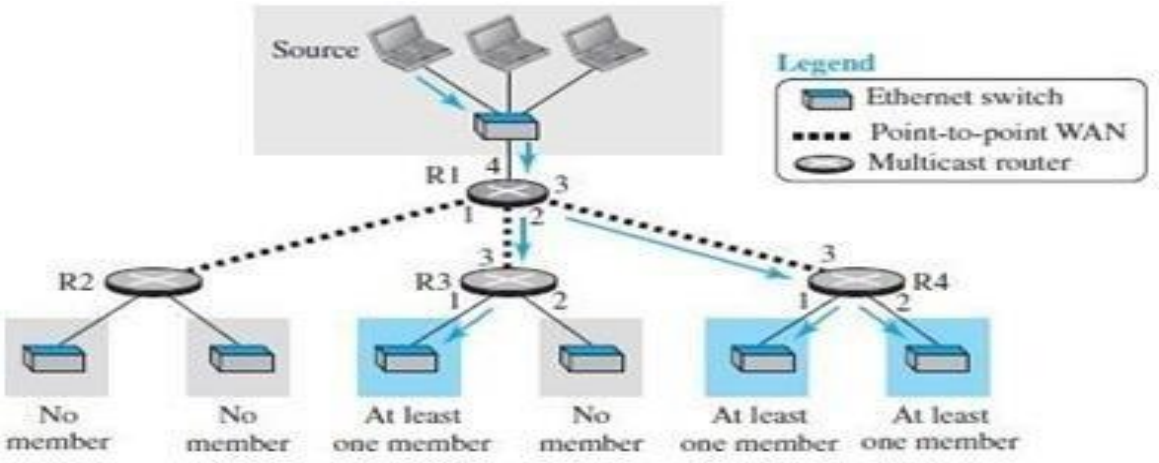
❑ **Notification.** A notification message is sent by a router whenever an error conditionis detected or a router wants to close the session.



9.MULTICAST ROUTING

In multicasting, there is one source and a group of destinations. The relationship is one to many. In this type of communication, the source address is a unicast address, but the destination address is a group address, a group of oneor more destination networks in which there is at least one member of the group that is interested in receiving the

multicast datagram. The group address defines the members of the group.



Two Approaches to Multicasting

In multicast routing, as in unicast routing, we need to create routing trees to optimally route the packets from their source to their destination.

Source-Based Tree Approach

In the **source-based tree** approach to multicasting, each router needs to create a separate tree for each source-group combination. In other words, if there are *m* groups and *n* sources in the internet, a router needs to create (*m* * *n*) routing trees. In each tree
,the corresponding source is the root, the members of the group are the leaves, and the router itself is somewhere on the tree.

Group-Shared Tree Approach

In the **group-shared tree** approach, we designate a router to act as the phony source foreach group. The designatedrouter, which is called the *core* router or the *rendezvous point* router, acts as the representative for the group. Any source that has a packet to send to a member of that group sends it to the core center (unicast communication) and the core center is responsible for multicasting. The core center creates one single routing tree with itself as the root and any routers with active members in the group as the leaves. In this approach, there are *m* core routers (one for each group) and each core router has a routing tree, for the total of *m* trees. This means that the number of routing trees is reduced from (*m* * *n*) in the source-based tree approach to *m* in this approach.

10.DVMRP

The **Distance Vector Multicast Routing Protocol** (**DVMRP**) is the extension of the Routing Information Protocol (RIP) which is used in unicast routing. It uses the sourcebased tree approach to multicasting. It is worth mentioning that each router in thisprotocol that receives a multicast packet to be forwarded implicitly creates a source-

based multicast tree in three steps:

1. The router uses an algorithm called *reverse path forwarding* (RPF) to simulate creating part of the optimal source-based tree between the source and itself.

2. The router uses an algorithm called *reverse path broadcasting* (RPB) to create a broadcast (spanning) tree whose root is the router itself and whose leaves are all networks in the internet.

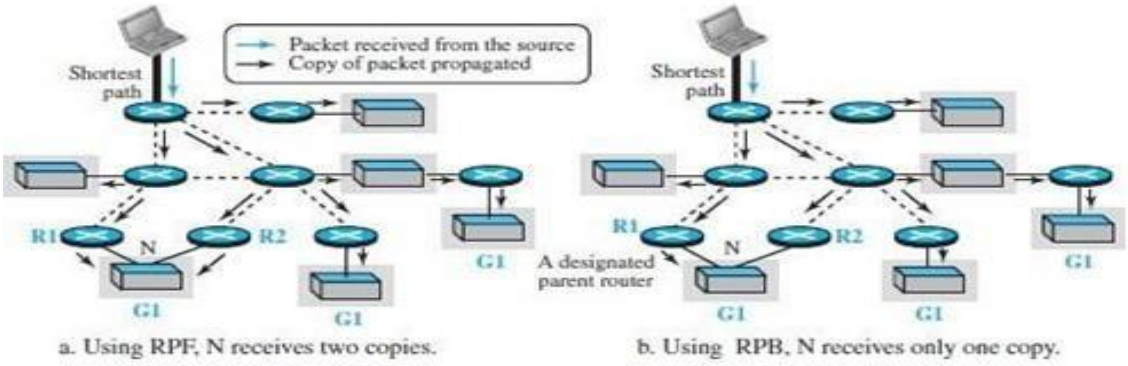
3. The router uses an algorithm called *reverse path multicasting* (RPM) to create a multicast tree by cutting some branches of the tree that end in networks with no member in the group.

Reverse Path Forwarding (RPF)

The first algorithm, **reverse path forwarding (RPF),** forces the router to forward a multicast packet from one specific interface: the one which has come through the shortest path from the source to the router. The router doesnot know the shortest path from the source to itself, but it can find which is the next router in the shortest path from itself to the source (reverse path). The router simply consults its unicast forwardingtable, pretending that it wants to send a packet to the source; the forwarding table gives the next router and the interface the message that the packet should be sent out in this reverse direction. The router uses this information to accept a multicast packet only if it arrives from this interface.

Reverse Path Broadcasting (RPB)

The RPF algorithm helps a router to forward only one copy received from a sourceand drop the rest. However, when we think about broadcasting in the second step, we need to remember that destinations are all the networks (LANs) in the internet. To be efficient, we need to prevent each network from receiving more than one copy of the packet. If a network is connected to more than one router, it may receive a copy of the packet from each router.Reverse path broadcasting (RPB) actually creates a broadcast tree from the graph that has been created by the RPF algorithm. RPB has cut those branches of the tree that cause cycles in the graph. If we use the shortest path criteria for choosingthe parent router, we have actually created a shortest-path broadcast tree.

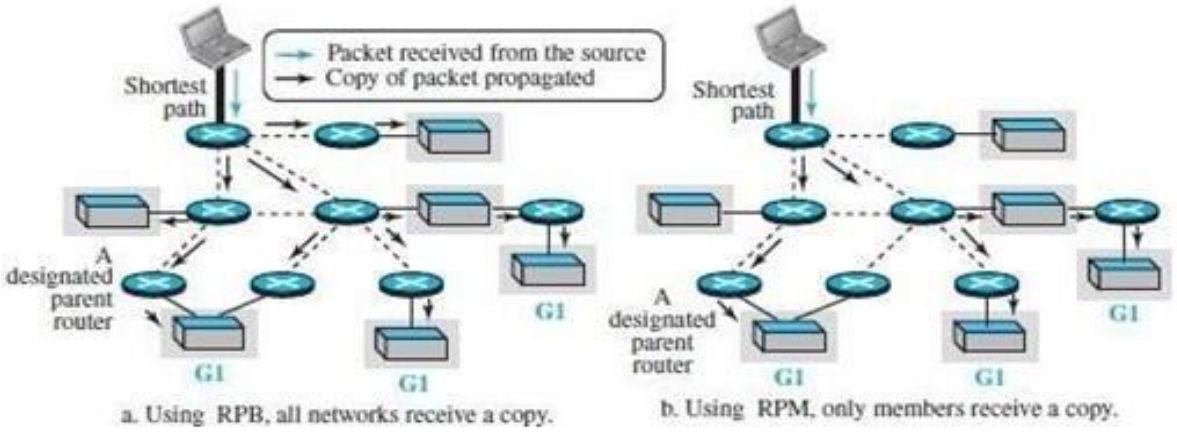


a. Using RPF, N receives two copies.

b. Using RPB, N receives only one copy.

Reverse Path Multicasting (RPM)

RPB does not multicast the packet, it broadcasts it. This is not efficient. To increase efficiency, the multicast packet must reach only those networks that have active members for that particular group. This is called **reverse path multicasting** (**RPM**)**.** To change the broadcast shortest-path tree to a multicast shortest-path tree, each router needs to prune (make inactive) the interfaces that do not reach a network with active members corresponding to a particular source-group combination. This step can be done bottom-up, from the leaves to the root.



a. Using RPB, all networks receive a copy.

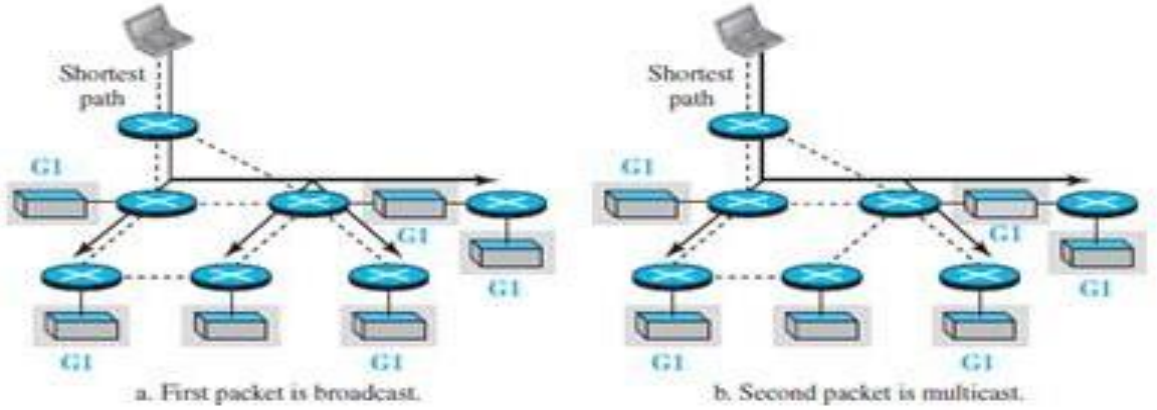
b. Using RPM, only members receive a copy.

11.PIM

Protocol Independent Multicast (PIM) is the name given to a common protocol that needs a unicast routing protocol for its operation, but the unicast protocol can be either a distance-vector protocol or a link-state protocol. In other words, PIM needs to use the forwarding table of a unicast routing protocol to find the next router in a path to the destination, but it does not matter how the forwarding table is created. PIM has another interesting feature: it can work in two different modes: dense and sparse.

Protocol Independent Multicast-Dense Mode (PIM-DM)

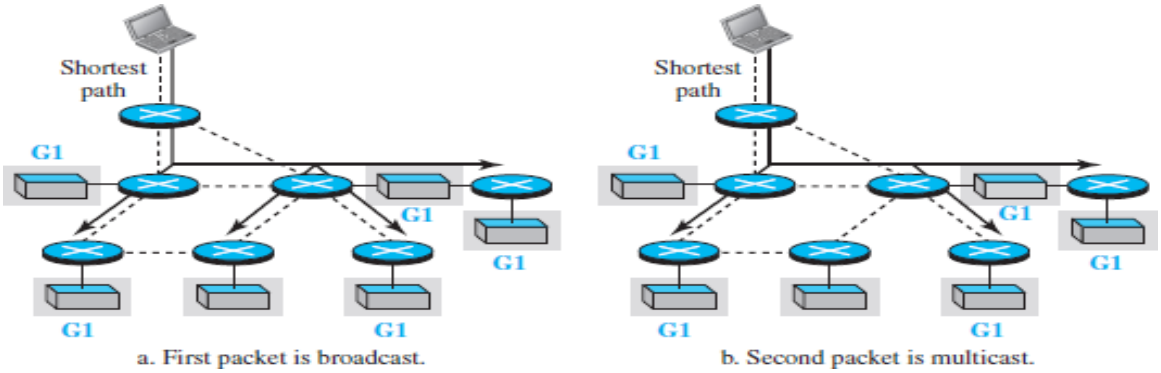
When the number of routers with attached members is large relative to the number of routers in the internet, PIM works in the dense mode and is called **PIM-DM**. In this mode, the protocol uses a source-based tree approach and is similar to DVMRP, but simpler. PIM-DM uses only two strategies described in DVMRP: RPF and RPM. The idea behind PIM-DM. The first packet is broadcast to all networks, which have or do not have members. After a prune message arrives from a router with no member, the second packet is only multicast.



a. First packet is broadcast. b. Second packet is multicast.

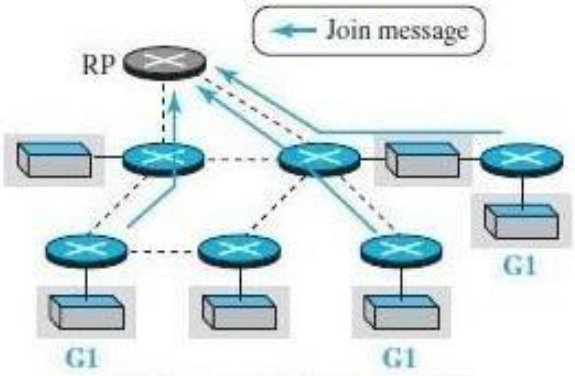
Protocol Independent Multicast-Sparse Mode (PIM-SM)

When the number of routers with attached members is small relative to the number of routers in the internet, PIM works in the sparse mode and is called **PIM-SM.** In this environment, the use of a protocol that broadcasts the packets until the tree is pruned is not justified; PIM-SM uses a group-shared tree approach to multicasting. The core router inPIM-SM is called the rendezvous point (RP). PIM-SM uses a complex algorithm to select one router amongall routers in the internet as the RP for a specific group. This means thatif we have m active groups, we need m RPs, although a router may serve more than one group.

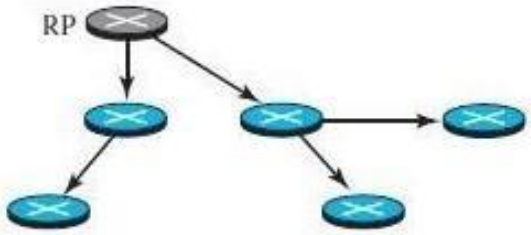


a. First packet is broadcast. b. Second packet is multicast.

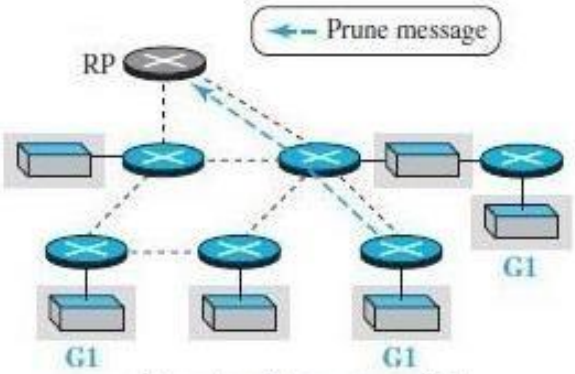
The idea is that each router helps to create the tree. To create a multicast tree rooted at the RP, PIM-SM uses *join* and *prune* messages. First, three networks join group G1 and form a multicast tree. Later, one of the networks leaves the group and the tree is pruned.The join message is used to add possible new branches to the tree; the **prune message** is used to cut branches that are not needed.



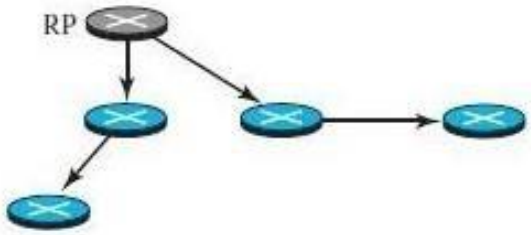
a. Three networks join group G1



b. Multicast tree after joins



c. One network leaves group G1



d. Multicast tree after pruning