

to contain information detailing remote networks. This information provides a more complete view of the overall environment.

A robust routing protocol provides the ability to dynamically build and manage the information in the IP routing table. As network topology changes occur, the routing tables are updated with minimal or no manual intervention. This chapter details several IP routing protocols and how each protocol manages this information.

Note

In other sections of this book, the position of each protocol within the layered model of the OSI protocol stack is shown. The routing function is included as part of the internetwork layer. However, the primary function of a routing protocol is to exchange routing information with other routers. In this respect, routing protocols behave more like an application protocol. Therefore, this chapter makes no attempt to represent the position of these protocols within the overall protocol stack.

Note

Early IP routing documentation often referred to an IP router as an *IP gateway*.

4.1 Autonomous systems

The definition of an autonomous system (AS) is integral to understanding the function and scope of a routing protocol. An AS is defined as a logical portion of a larger IP network. An AS is normally comprised of an internetwork within an organization. It is administered by a single management authority. As shown in Figure 59, an AS may connect to other autonomous systems managed by the same organization. Alternatively, it may connect to other public or private networks.

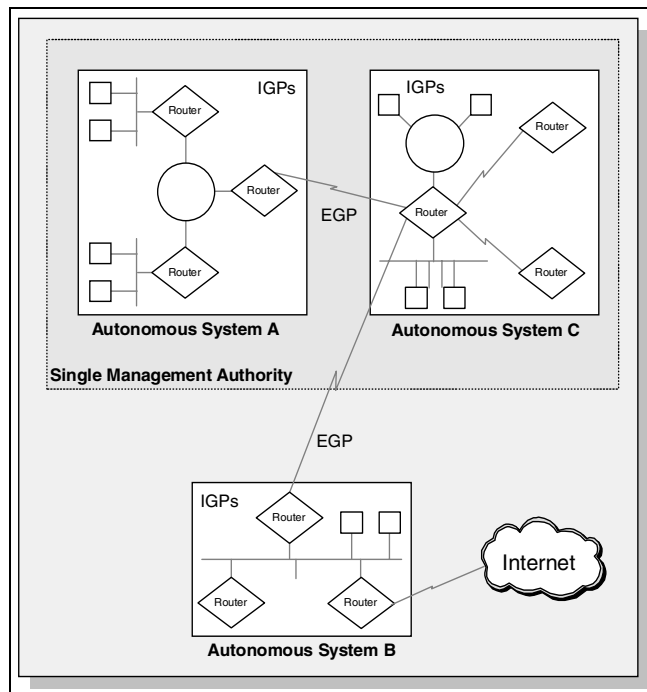


Figure 59. Autonomous systems

Some routing protocols are used to determine routing paths within an AS. Others are used to interconnect a set of autonomous systems:

- Interior Gateway Protocols (IGPs): Interior gateway protocols allow routers to exchange information within an AS. Examples of these protocols are Open Short Path First (OSPF) and Routing Information Protocol (RIP).
- Exterior Gateway Protocols (EGPs): Exterior gateway protocols allow the exchange of summary information between autonomous systems. An example of this type of routing protocol is Border Gateway Protocol (BGP).

Figure 59 depicts the interaction between interior and exterior protocols. It shows the interior protocols used to maintain routing information within each AS. The figure also shows the exterior protocols maintaining the routing information between autonomous systems.

Within an AS, multiple interior routing processes may be used. When this occurs, the AS must appear to other autonomous systems as having a single,

coherent interior routing plan. The AS must present a consistent view of the internal destinations

4.2 Types of IP routing and IP routing algorithms

Routing algorithms are used to build and maintain the IP routing table on a device. There are two primary methods used to build the routing table:

- Static routing: Static routing use preprogrammed definitions representing paths through the network.
- Dynamic routing: Dynamic routing algorithms allow routers to automatically discover and maintain awareness of the paths through the network. This automatic discovery can use a number of currently available dynamic routing protocols. The difference between these protocols is the way they discover and calculate new routes to destination networks. They can be classified into three broad categories:
 - Distance vector protocols
 - Link state protocols
 - Hybrid protocols

The remainder of this section details the operation of each algorithm.

There are several reasons for the multiplicity of protocols:

- Routing within a network and routing between networks typically have different requirements for security, stability, and scalability. Different routing protocols have been developed to address these requirements.
- New protocols have been developed to address the observed deficiencies in established protocols.
- Different-sized networks can use different routing algorithms. Small to medium-sized networks often use routing protocols that reflect the simplicity of the environment. However, these protocols do not scale to support large, interconnected networks. More complex routing algorithms are required to support these environments.

4.2.1 Static routing

Static routing is manually performed by the network administrator. The administrator is responsible for discovering and propagating routes through the network. These definitions are manually programmed in every routing device in the environment.

Once a device has been configured, it simply forwards packets out the predetermined ports. There is no communication between routers regarding the current topology of the network.

In small networks with minimal redundancy, this process is relatively simple to administer. However, there are several disadvantages to this approach for maintaining IP routing tables:

- Static routes require a considerable amount of coordination and maintenance in non-trivial network environments.
- Static routes cannot dynamically adapt to the current operational state of the network. If a destination subnetwork becomes unreachable, the static routes pointing to that network remain in the routing table. Traffic continues to be forwarded toward that destination. Unless the network administrator updates the static routes to reflect the new topology, traffic is unable to use any alternate paths that may exist.

Normally, static routes are used only in simple network topologies. However, there are additional circumstances when static routing can be attractive. For example, static routes can be used:

- To manually define a default route. This route is used to forward traffic when the routing table does not contain a more specific route to the destination.
- To define a route that is not automatically advertised within a network.
- When utilization or line tariffs make it undesirable to send routing advertisement traffic through lower-capacity WAN connections.
- When complex routing policies are required. For example, static routes can be used to guarantee that traffic destined for a specific host traverses a designated network path.
- To provide a more secure network environment. The administrator is aware of all subnetworks defined in the environment. The administrator specifically authorizes all communication permitted between these subnetworks.
- To provide more efficient resource utilization. This method of routing table management requires no network bandwidth to advertise routes between neighboring devices. It also uses less processor memory and CPU cycles to calculate network paths.

4.2.2 Distance vector routing

Distance vector algorithms are examples of dynamic routing protocols. These algorithms allow each device in the network to automatically build and maintain a local IP routing table.

The principle behind distance vector routing is simple. Each router in the internetwork maintains the *distance* or *cost* from itself to every known destination. This value represents the overall desirability of the path. Paths associated with a smaller cost value are more attractive to use than paths associated with a larger value. The path represented by the smallest cost becomes the preferred path to reach the destination.

This information is maintained in a *distance vector table*. The table is periodically advertised to each neighboring router. Each router processes these advertisements to determine the best paths through the network.

The main advantage of distance vector algorithms is that they are typically easy to implement and debug. They are very useful in small networks with limited redundancy. However, there are several disadvantages with this type of protocol:

- During an adverse condition, the length of time for every device in the network to produce an accurate routing table is called the *convergence time*. In large, complex internetworks using distance vector algorithms, this time can be excessive. While the routing tables are converging, networks are susceptible to inconsistent routing behavior. This can cause routing loops or other types of unstable packet forwarding.
- To reduce convergence time, a limit is often placed on the maximum number of hops contained in a single route. Valid paths exceeding this limit are not usable in distance vector networks.
- Distance vector routing tables are periodically transmitted to neighboring devices. They are sent even if no changes have been made to the contents of the table. This may cause noticeable periods of increased utilization in reduced capacity environments.

Enhancements to the basic distance vector algorithm have been developed to reduce the convergence and instability exposures. These enhancements are described in 4.3.5, “Convergence and counting to infinity” on page 148.

RIP and BGP are two popular examples of distance vector routing protocols.

4.2.3 Link state routing

The growth in the size and complexity of networks in recent years has necessitated the development of more robust routing algorithms. These algorithms address the shortcoming observed in distance vector protocols.

These algorithms use the principle of a *link state* to determine network topology. A link state is the description of an interface on a router (for example, IP address, subnet mask, type of network) and its relationship to neighboring routers. The collection of these link states forms a link state database.

The process used by link state algorithms to determine network topology is straightforward:

- Each router identifies all other routing devices on the directly connected networks.
- Each router advertises a list of all directly connected network links and the associated cost of each link. This is performed through the exchange of link state advertisements (LSAs) with other routers in the network.
- Using these advertisements, each router creates a database detailing the current network topology. The topology database in each router is identical.
- Each router uses the information in the topology database to compute the most desirable routes to each destination network. This information is used to update the IP routing table.

4.2.3.1 Shortest-Path First (SPF) algorithm

The SPF algorithm is used to process the information in the topology database. It provides a tree-representation of the network. The device running the SPF algorithm is the root of the tree. The output of the algorithm is the list of shortest-paths to each destination network. Figure 60 provides an example of the shortest-path algorithm executed on router A.

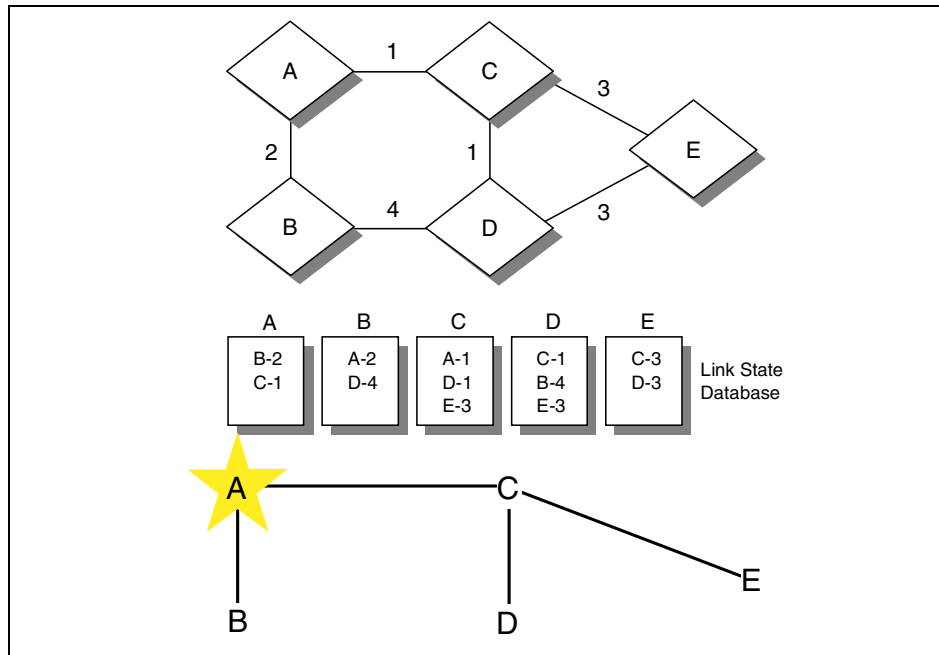


Figure 60. Shortest-Path First (SPF) example

Because each router is processing the same set of LSAs, each router creates an identical link state database. However, because each device occupies a different place in the network topology, application of the SPF algorithm produces a different tree for each router.

The OSPF protocol is a popular example of a link state routing protocol.

4.2.4 Hybrid routing

The last category of routing protocols is hybrid protocols. These protocols attempt to combine the positive attributes of both distance vector and link state protocols. Like distance vector, hybrid protocols use metrics to assign a preference to a route. However, the metrics are more accurate than conventional distance vector protocols. Like link state algorithms, routing updates in hybrid protocols are event driven rather than periodic. Networks using hybrid protocols tend to converge more quickly than networks using distance vector protocols. Finally, these protocols potentially reduce the overhead of link state updates and distance vector advertisements.

Although open hybrid protocols exist, this category is almost exclusively associated with the proprietary EIGRP algorithm. EIGRP was developed by Cisco Systems, Inc.

4.3 Routing Information Protocol (RIP)

RIP is an example of an interior gateway protocol designed for use within small autonomous systems. RIP is based on the Xerox XNS routing protocol. Early implementations of RIP were readily accepted because the code was incorporated in the Berkeley Software Distribution (BSD) UNIX-based operating system. RIP is a distance vector protocol.

In mid-1988, the IETF issued RFC 1058, which describes the standard operations of a RIP system. However, the RFC was issued after many RIP implementations had been completed. For this reason, some RIP systems do not support the entire set of enhancements to the basic distance vector algorithm (for example, poison reverse and triggered updates).

4.3.1 RIP packet types

The RIP protocol specifies two packet types. These packets may be sent by any device running the RIP protocol:

- Request packets: A request packet queries neighboring RIP devices to obtain their distance vector table. The request indicates if the neighbor should return either a specific subset or the entire contents of the table.
- Response packets: A response packet is sent by a device to advertise the information maintained in its local distance vector table. The table is sent during the following situations:
 - The table is automatically sent every 30 seconds.
 - The table is sent as a response to a request packet generated by another RIP node.
 - If triggered updates are supported, the table is sent when there is a change to the local distance vector table. Triggered updates are presented in 4.3.5.3, “Triggered updates” on page 152.

When a response packet is received by a device, the information contained in the update is compared against the local distance vector table. If the update contains a lower cost route to a destination, the table is updated to reflect the new path.

4.3.2 RIP packet format

RIP uses a specific packet format to share information about the distances to known network destinations. RIP packets are transmitted using UDP datagrams. RIP sends and receives datagrams using UDP port 520.

RIP datagrams have a maximum size of 512 octets. Updates larger than this size must be advertised in multiple datagrams. In LAN environments, RIP datagrams are sent using the MAC all-stations broadcast address and an IP network broadcast address. In point-to-point or non-broadcast environments, datagrams are specifically addressed to the destination device.

The RIP packet format is shown in Figure 61.

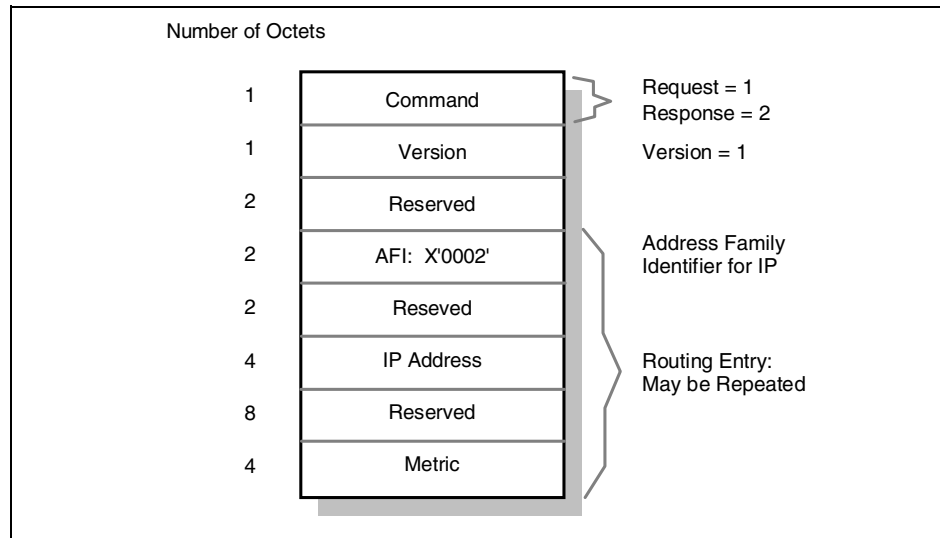


Figure 61. RIP packet format

A 512 byte packet size allows a maximum of 25 routing entries to be included in a single RIP advertisement.

4.3.3 RIP modes of operation

RIP hosts have two modes of operation:

- Active mode: Devices operating in active mode advertise their distance vector table and also receive routing updates from neighboring RIP hosts. Routing devices are typically configured to operate in active mode.
- Passive (or silent) mode: Devices operating in this mode simply receive routing updates from neighboring RIP devices. They do not advertise their

distance vector table. End stations are typically configured to operate in passive mode.

4.3.4 Calculating distance vectors

The distance vector table describes each destination network. The entries in this table contain the following information:

- The destination network (vector) described by this entry in the table.
- The associated cost (distance) of the most attractive path to reach this destination. This provides the ability to differentiate between multiple paths to a destination. In this context, the terms distance and cost can be misleading. They have no direct relationship to physical distance or monetary cost.
- The IP address of the next-hop device used to reach the destination network.

Each time a routing table advertisement is received by a device, it is processed to determine if any destination can be reached via a lower cost path. This is done using the RIP distance vector algorithm. The algorithm can be summarized as:

- At router initialization, each device contains a distance vector table listing each directly attached networks and configured cost. Typically, each network is assigned a cost of 1. This represents a single hop through the network. The total number of hops in a route is equal to the total cost of the route. However, cost can be changed to reflect other measurements such as utilization, speed, or reliability.
- Each router periodically (typically every 30 seconds) transmits its distance vector table to each of its neighbors. The router may also transmit the table when a topology change occurs.
- Each router uses this information to update its local distance vector table:
 - The total cost to each destination is calculated by adding the cost reported in a neighbor's distance vector table to the cost of the link to that neighbor. The path with the least cost is stored in the distance vector table.
 - All updates automatically supersede the previous information in the distance vector table. This allows RIP to maintain the integrity of the routes in the routing table.
- The IP routing table is updated to reflect the least-cost path to each destination.

Figure 62 illustrates the distance vector tables for three routers within a simple internetwork.

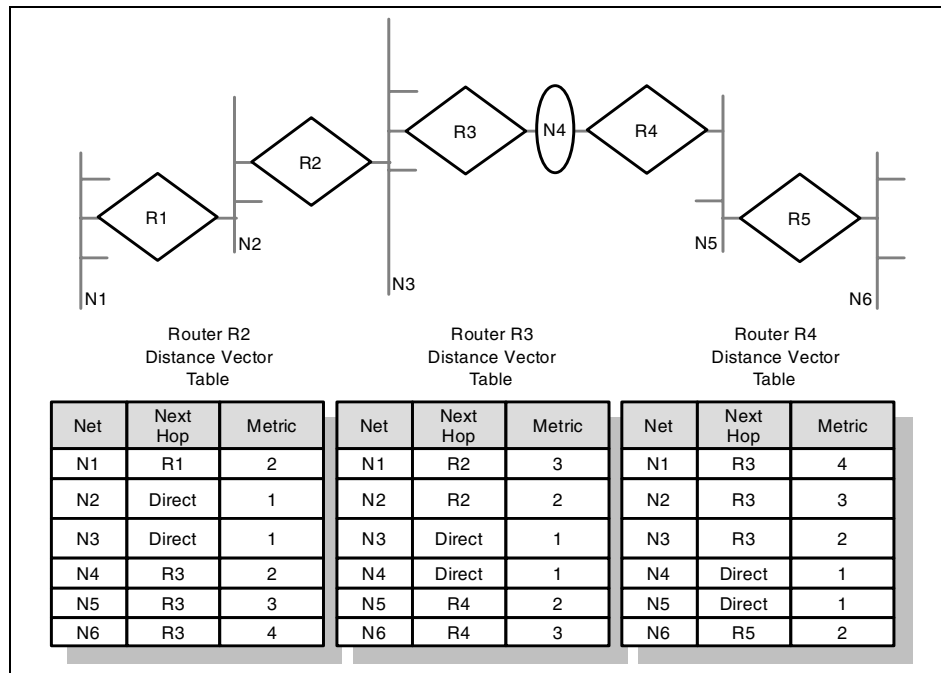


Figure 62. A sample distance vector routing table

4.3.5 Convergence and counting to infinity

Given sufficient time, this algorithm will correctly calculate the distance vector table on each device. However, during this convergence time, erroneous routes may propagate through the network. This problem is shown in Figure 63.

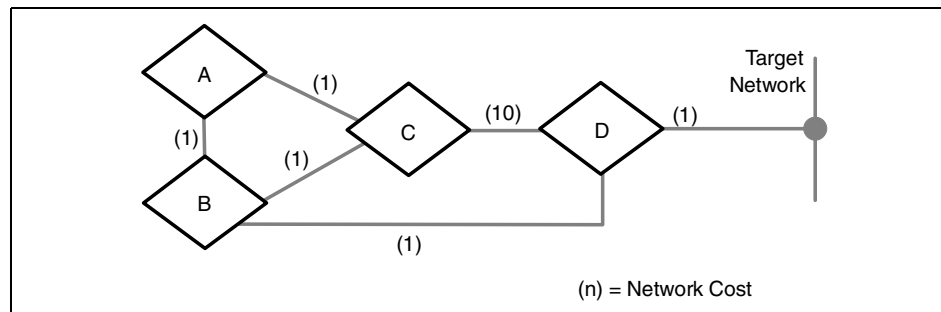


Figure 63. Counting to infinity sample network

This network contains four interconnected routers. Each link has a cost of 1, except for the link connecting router C and router D; this link has a cost of 10. The costs have been defined so that forwarding packets on the link connecting router C and router D is undesirable.

Once the network has converged, each device has routing information describing all networks. For example, to reach the target network, the routers have the following information:

- Router D to the target network: Directly connected network. Metric 1.
- Router B to the target network: Next hop is router D. Metric is 2.
- Router C to the target network: Next hop is router B. Metric is 3.
- Router A to the target network: Next hop is router B. Metric is 3.

Consider an adverse condition where the link connecting router B and router D fails. Once the network has reconverged, all routes use the link connecting router C and router D to reach the target network. However, this reconvergence time can be considerable. Figure 64 illustrates how the routes to the target network are updated throughout the reconvergence period. For simplicity, this figure assumes all routers send updates at the same time.

Time	→	→	→	→	→	→	→	→	→	→	→	→	
D:	Direct	1	Direct	1	Direct	1	Direct	1	Direct	1	Direct	1
B:	Unreachable		C	4	C	5	C	6		C	11	C	12
C:	B	3	A	4	A	5	A	6		A	11	D	11
A:	B	3	C	4	C	5	C	6	C	11	C	12

Figure 64. Network convergence sequence

Reconvergence begins when router B notices that the route to router D is unavailable. Router B is able to immediately remove the failed route because the link has timed-out. However, a considerable amount of time passes before the other routers remove their references to the failed route. This is described in the sequence of updates shown in Figure 64:

1. Prior to the adverse condition occurring, router A and router C have a route to the target network via router B.
2. The adverse condition occurs when the link connecting router D and router B fails. Router B recognizes that its preferred path to the target network is now invalid.

3. Router A and router C continue to send updates reflecting the route via router B. This route is actually invalid since the link connecting router D and router B has failed.
4. Router B receives the updates from router A and router C. Router B believes it should now route traffic to the target network through either router A or router C. In reality, this is not a valid route, since the routes in router A and router C are vestiges of the previous route through router B.
5. Using the routing advertisement sent by router B, router A and router C are able to determine that the route via router B has failed. However, router A and router C now believe the preferred route exists via the partner.

Network convergence continues as router A and router C engage in an extended period of mutual deception. Each device claims to be able to reach the target network via the partner device. The path to reach the target network now contains a routing loop.

The manner in which the costs in the distance vector table increment gives rise to the term *counting to infinity*. The costs continues to increment, theoretically to infinity. To minimize this exposure, whenever a network is unavailable, the incrementing of metrics through routing updates must be halted as soon as it is practical to do so. In a RIP environment, costs continue to increment until they reach a maximum value of 16. This limit is defined in the RFC.

A side effect of the metric limit is that it also limits the number of hops a packet can traverse from source network to destination network. In a RIP environment, any path exceeding 15 hops is considered invalid. The routing algorithm will discard these paths.

There are two enhancements to the basic distance vector algorithm that can minimize the counting to infinity problem:

- Split horizon with poison reverse
- Triggered updates

These enhancements do not impact the maximum metric limit.

4.3.5.1 Split horizon

The excessive convergence time caused by counting to infinity may be reduced with the use of split horizon. This rule dictates that routing information is prevented from exiting the router on an interface through which the information was received.

The basic split horizon rule is not supported in RFC 1058. Instead, the standard specifies the enhanced split horizon with poison reverse algorithm. The basic rule is presented here for background and completeness. The enhanced algorithm is reviewed in the next section.

The incorporation of split horizon modifies the sequence of routing updates shown in Figure 64. The new sequence is shown in Figure 65. The tables show that convergence occurs considerably faster using the split horizon rule.

Time	→	→	→	→				
D:	Direct	1	Direct	1	Direct	1	Direct	1
B:	Unreachable		Unreachable		Unreachable	C	12	
C:	B	3	A	4	D	11	D	11
A:	B	3	C	4	Unreachable	C	12	
Note: Faster Routing Table Convergence								

Figure 65. Network convergence with split horizon

The limitation to this rule is that each node must wait for the route to the unreachable destination to time out before the route is removed from the distance vector table. In RIP environments, this timeout is at least three minutes after the initial outage. During that time, the device continues to provide erroneous information to other nodes about the unreachable destination. This propagates routing loops and other routing anomalies.

4.3.5.2 Split horizon with poison reverse

Poison reverse is an enhancement to the standard split horizon implementation. It is supported in RFC 1058. With poison reverse, all known networks are advertised in each routing update. However, those networks learned through a specific interface are advertised as unreachable in the routing announcements sent out to that interface.

This drastically improves convergence time in complex, highly-redundant environments. With poison reverse, when a routing update indicates that a network is unreachable, routes are immediately removed from the routing table. This breaks erroneous, looping routes before they can propagate through the network. This approach differs from the basic split horizon rule where routes are eliminated through timeouts.

Poison reverse has no benefit in networks with no redundancy (single path networks).

One disadvantage to poison reverse is that it may significantly increase the size of routing announcements exchanged between neighbors. This is because all routes in the distance vector table are included in each announcement. While this is generally not an issue on local area networks, it can cause periods of increased utilization on lower-capacity WAN connections.

4.3.5.3 Triggered updates

Like split horizon with poison reverse, algorithms implementing triggered updates are designed to reduce network convergence time. With triggered updates, whenever a router changes the cost of a route, it immediately sends the modified distance vector table to neighboring devices. This mechanism ensures that topology change notifications are propagated quickly, rather than at the normal periodic interval.

Triggered updates are supported in RFC 1058.

4.3.6 RIP limitations

There are a number of limitations observed in RIP environments:

- **Path cost limits:** The resolution to the counting to infinity problem enforces a maximum cost for a network path. This places an upper limit on the maximum network diameter. Networks requiring paths greater than 15 hops must use an alternate routing protocol.
- **Network-intensive table updates:** Periodic broadcasting of the distance vector table can result in increased utilization of network resources. This can be a concern in reduced-capacity segments.
- **Relatively slow convergence:** RIP, like other distance vector protocols, is relatively slow to converge. The algorithms rely on timers to initiate routing table advertisements.
- **No support for variable length subnet masking:** Route advertisements in a RIP environment do not include subnet masking information. This makes it impossible for RIP networks to deploy variable length subnet masks.

4.4 Routing Information Protocol Version 2 (RIP-2)

The IETF recognizes two versions of RIP:

- **RIP Version 1 (RIP-1):** This protocol is described in RFC 1058.
- **RIP Version 2 (RIP-2):** RIP-2 is also a distance vector protocol designed for use within an AS. It was developed to address the limitations observed

in RIP-1. RIP-2 is described in RFC 1723. The standard was published in late 1994.

In practice, the term RIP refers to RIP-1. Whenever the reader encounters the term RIP in TCP/IP literature, it is safe to assume the reference is to RIP Version 1 unless otherwise stated. This same convention is used in this document. However, when the two versions are being compared, the term RIP-1 is used to avoid confusion.

RIP-2 is similar to RIP-1. It was developed to extend RIP-1 functionality in small networks. RIP-2 provides these additional benefits not available in RIP-1:

- Support for CIDR and VLSM: RIP-2 supports supernetting (that is, CIDR) and variable-length subnet masking. This support was the major reason the new standard was developed. This enhancement positions the standard to accommodate a degree of addressing complexity not supported in RIP-1.
- Support for multicasting: RIP-2 supports the use of multicasting rather than simple broadcasting of routing announcements. This reduces the processing load on hosts not listening for RIP-2 messages. To ensure interoperability with RIP-1 environments, this option is configured on each network interface.
- Support for authentication: RIP-2 supports authentication of any node transmitting route advertisements. This prevents fraudulent sources from corrupting the routing table.
- Support for RIP-1: RIP-2 is fully interoperable with RIP-1. This provides backward-compatibility between the two standards.

As noted in the RIP-1 section, one notable shortcoming in the RIP-1 standard is the implementation of the metric field. RIP-1 specifies the metric as a value between 0 and 16. To ensure compatibility with RIP-1 networks, RIP-2 preserves this definition. In both standards, networks paths with a hop-count greater than 15 are interpreted as unreachable.

4.4.1 RIP-2 packet format

The original RIP-1 specification was designed to support future enhancements. The RIP-2 standard was able to capitalize on this feature. RIP-2 developers noted that a RIP-1 packet already contains a version field and that 50 percent of the octets are unused.

Figure 66 illustrates the contents of a RIP-2 packet. The packet is shown with authentication information. The first entry in the update contains either a

routing entry or an authentication entry. If the first entry is an authentication entry, 24 additional routing entries can be included in the message. If there is no authentication information, 25 routing entries can be provided.

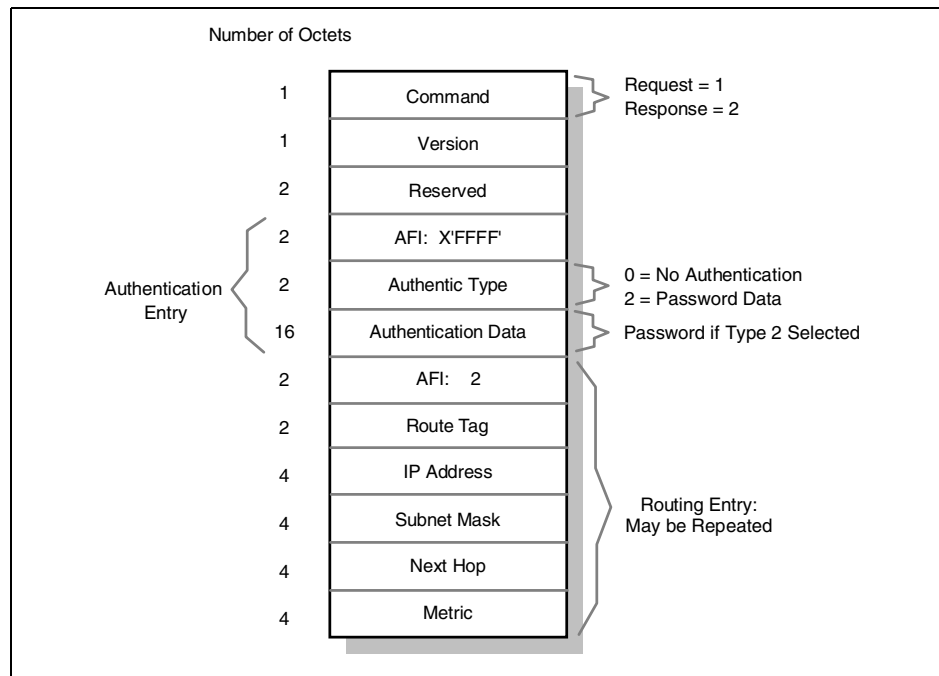


Figure 66. RIP-2 packet format

The use of the command field, IP address field, and metric field in a RIP-2 message is identical to the use in a RIP-1 message. Otherwise, the changes implemented in a RIP-2 packets include:

- **Version:** The value contained in this field must be two. This instructs RIP-1 routers to ignore any information contained in the previously unused fields.
- **AFI (Address Family):** A value of x'0002' indicates the address contained in the network address field is an IP address. An value of x'FFFF' indicates an authentication entry.
- **Authentication Type:** This field defines the remaining 16 bytes of the authentication entry. A value of 0 indicates *no* authentication. A value of two indicates the authentication data field contains password data.
- **Authentication Data:** This field contains a 16-byte password.

- **Route Tag:** This field is intended to differentiate between internal and external routes. Internal routes are learned via RIP-2 within the same network or AS.
- **Subnet Mask:** This field contains the subnet mask of the referenced network.
- **Next Hop:** This field contains a recommendation about the next hop the router should use when sending datagrams to the referenced network.

4.4.2 RIP-2 limitations

RIP-2 was developed to address many of the limitations observed in RIP-1. However, the path cost limits and slow convergence inherent in RIP-1 networks are also concerns in RIP-2 environments.

In addition to these concerns, there are limitations to the RIP-2 authentication process. The RIP-2 standard does not encrypt the authentication password. It is transmitted in clear text. This makes the network vulnerable to attack by anyone with direct physical access to the environment.

4.5 RIPng for IPv6

RIPng was developed to allow routers within an IPv6-based network to exchange information used to compute routes. It is documented in RFC 2080. Additional information regarding IPv6 is presented in Chapter 17, “IP Version 6” on page 559.

Like the other protocols in the RIP family, RIPng is a distance vector protocol designed for use within a small autonomous system. RIPng uses the same algorithms, timers, and logic used in RIP-2.

RIPng has many of the same limitations inherent in other distance vector protocols. Path cost restrictions and convergence time remain a concern in RIPng networks.

4.5.1 Differences between RIPng and RIP-2

There are two important distinctions between RIP-2 and RIPng:

- **Support for authentication:** The RIP-2 standard includes support for authenticating a node transmitting routing information. RIPng does not include any native authentication support. Rather, RIPng uses the security features inherent in IPv6. In addition to authentication, these security features provide the ability to encrypt each RIPng packet. This can control the set of devices that receive the routing information.

One consequence of using IPv6 security features is that the AFI field within the RIPng packet is eliminated. There is no longer a need to distinguish between authentication entries and routing entries within an advertisement.

- Support for IPv6 addressing formats: The fields contained in RIPng packets were updated to support the longer IPv6 address format.

4.5.2 RIPng packet format

RIPng packets are transmitted using UDP datagrams. RIPng sends and receives datagrams using UDP port number 521.

The format of a RIPng packet is similar to the RIP-2 format. Specifically both packets contain a 4 octet command header followed by a set of 20 octet route entries. The RIPng packet format is shown in Figure 67.

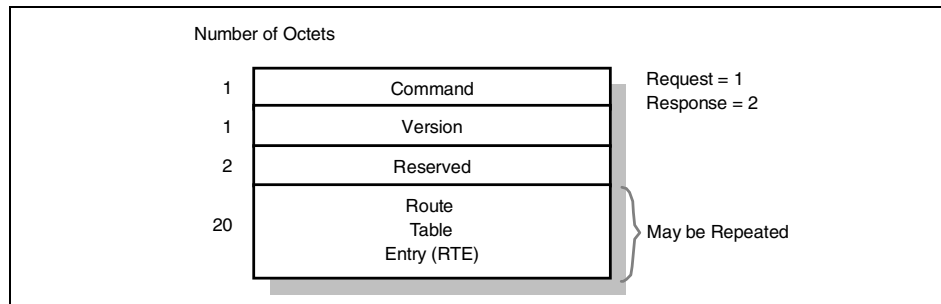


Figure 67. RIPng packet format

The use of the command field and the version field is identical to the use in a RIP-2 packet. However, the fields containing routing information have been updated to accommodate the 16 octet IPv6 address. These fields are used differently than the corresponding fields in a RIP-1 or RIP-2 packet. The format of the RTE is shown in Figure 68.

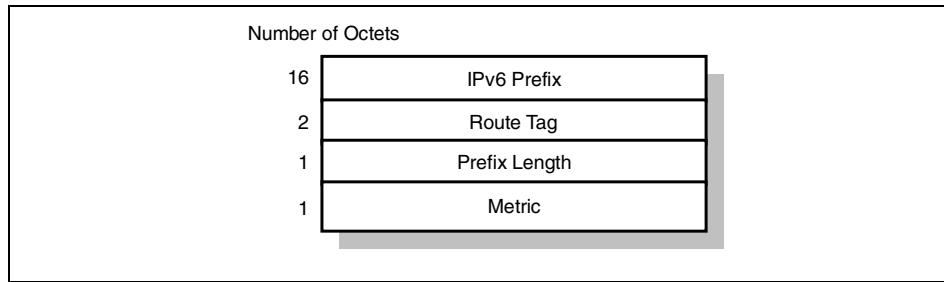


Figure 68. Route Table Entry (RTE)

In RIPng, the combination of the IP prefix and the prefix length identifies the route to be advertised. The metric remains encoded in a 1 octet field. This length is sufficient since RIPng uses a maximum hop-count of 16.

Another difference between RIPng and RIP-2 is the process used to determine the next hop. In RIP-2, each route table entry contains a next hop field. In RIPng, including this information in each RTE would have doubled the size of the advertisement. Therefore, in RIPng, the next hop is included in a special type of RTE. The specified next hop applies to each subsequent routing table entry in the advertisement. The format of an RTE used to specify the next hop is shown in Figure 69.

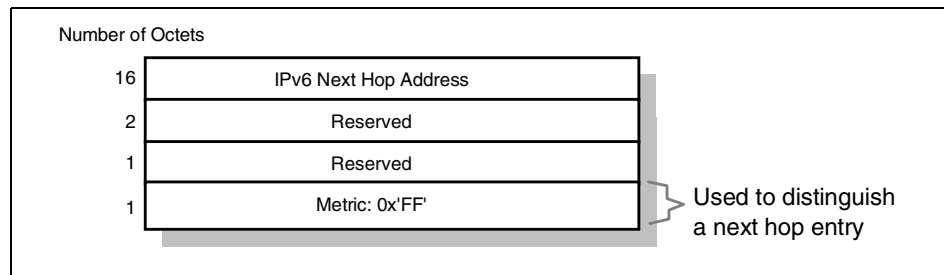


Figure 69. Next Hop Route Table Entry (RTE)

The next hop RTE is identified by a value of 0xFF in the metric field. This reserved value is outside the valid range of metrics.

The use of RTEs and next hop RTEs is shown in Figure 70.

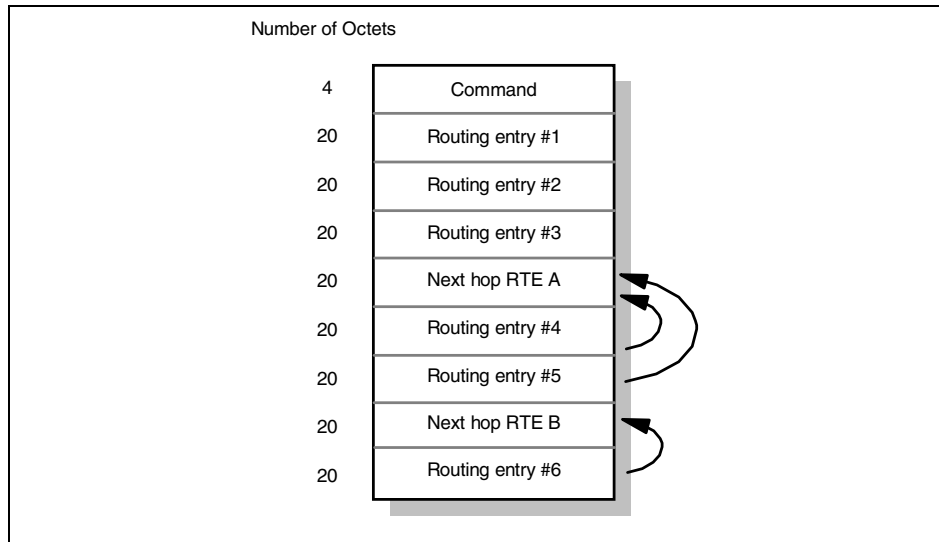


Figure 70. Using the RIPng RTE

In this example, the first three routing entries do not have a corresponding next hop RTE. The address prefixes specified by these entries will be routed through the advertising router. The prefixes included in routing entries 4 and 5 will route through the next hop address specified in the next hop RTE A. The prefix included in routing entry 6 will route through the next hop address specified in the next hop RTE B.

4.6 Open Shortest Path First (OSPF)

The Open Shortest Path First (OSPF) protocol is another example of an interior gateway protocol. It was developed as a non-proprietary routing alternative to address the limitations of RIP. Initial development started in 1988 and was finalized in 1991. Subsequent updates to the protocol continue to be published. The current version of the standard is documented in RFC 2328.

OSPF provides a number of features not found in distance vector protocols. Support for these features has made OSPF a widely-deployed routing protocol in large networking environments. In fact, RFC 1812 – Requirements for IPv4 Routers, lists OSPF as the only required dynamic routing protocol. The following features contribute to the continued acceptance of the OSPF standard:

- Equal cost load balancing: The simultaneous use of multiple paths may provide more efficient utilization of network resources.
- Logical partitioning of the network: This reduces the propagation of outage information during adverse conditions. It also provides the ability to aggregate routing announcements that limit the advertisement of unnecessary subnet information.
- Support for authentication: OSPF supports the authentication of any node transmitting route advertisements. This prevents fraudulent sources from corrupting the routing tables.
- Faster convergence time: OSPF provides instantaneous propagation of routing changes. This expedites the convergence time required to update network topologies.
- Support for CIDR and VLSM: This allows the network administrator to efficiently allocate IP address resources.

OSPF is a link state protocol. As with other link state protocols, each OSPF router executes the SPF algorithm (refer to 4.2.3.1, “Shortest-Path First (SPF) algorithm” on page 143) to process the information stored in the link state database. The algorithm produces a shortest-path tree detailing the preferred routes to each destination network.

4.6.1 OSPF terminology

OSPF uses specific terminology to describe the operation of the protocol.

4.6.1.1 OSPF areas

OSPF networks are divided into a collection of *areas*. An area consists of a logical grouping of networks and routers. The area may coincide with geographic or administrative boundaries. Each area is assigned a 32-bit *area ID*.

Subdividing the network provides the following benefits:

- Within an area, every router maintains an identical topology database describing the routing devices and links within the area. These routers have no knowledge of topologies outside the area. They are only aware of routes to these external destinations. This reduces the size of the topology database maintained by each router.
- Areas limit the potentially explosive growth in the number of link state updates. Most LSAs are distributed only within an area.

- Areas reduce the CPU processing required to maintain the topology database. The SPF algorithm is limited to managing changes within the area.

Backbone area and area 0

All OSPF networks contain at least one area. This area is known as area 0 or the backbone area. Additional areas may be created based on network topology or other design requirements.

In networks containing multiple areas, the backbone physically connects to all other areas. OSPF expects all areas to announce routing information directly into the backbone. The backbone then announces this information into other areas.

Figure 71 depicts a network with a backbone area and 4 additional areas.

4.6.1.2 Intra-area, area border and AS boundary routers

There are three classifications of routers in an OSPF network. Figure 71 illustrates the interaction of these devices.

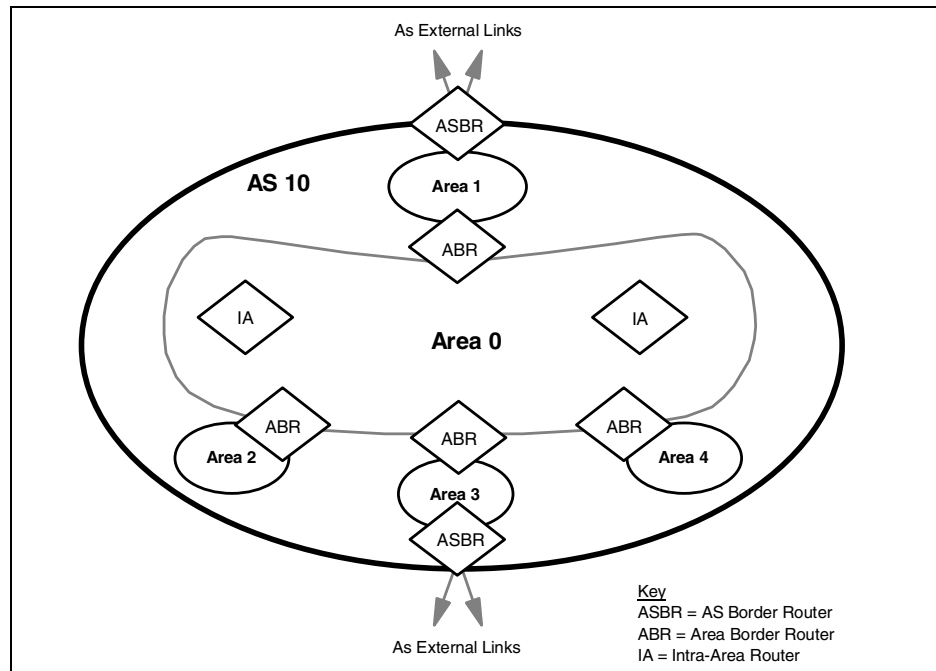


Figure 71. OSPF router types

Where:

- Intra-Area Routers: This class of router is logically located entirely within an OSPF area. Intra-area routers maintain a topology database for their local area.
- Area Border Routers (ABR): This class of router is logically connected to two or more areas. One area must be the backbone area. An ABR is used to interconnect areas. They maintain a separate topology database for each attached area. ABRs also execute separate instances of the SPF algorithm for each area.
- AS Boundary Routers (ASBR): This class of router is located at the periphery of an OSPF internetwork. It functions as a gateway exchanging reachability between the OSPF network and other routing environments. ASBRs are responsible for announcing AS external link advertisements through the AS. External link advertisements are further detailed in 4.6.6, “OSPF route redistribution” on page 170.

Each router is assigned a 32-bit *router ID (RID)*. The RID uniquely identifies the device. One popular implementation assigns the RID from the lowest-numbered IP address configured on the router.

4.6.1.3 Physical network types

OSPF categorizes network segments into three types. The frequency and types of communication occurring between OSPF devices connected to these networks is impacted by the network type:

- Point-to-point: Point-to-point networks directly link two routers.
- Multi-access: Multi-access networks support the attachment of more than two routers. They are further subdivided into two types:
 - Broadcast networks have the capability of simultaneously directing a packet to all attached routers. This capability uses an address that is recognized by all devices. Ethernet and token-ring LANs are examples of OSPF broadcast multi-access networks.
 - Non-broadcast networks do not have broadcasting capabilities. Each packet must be specifically addressed to every router in the network. X.25 and frame relay networks are examples of OSPF non-broadcast multi-access networks.
- Point-to-Multipoint: Point-to-multipoint networks are a special case of multi-access, non-broadcast networks. In a point-to-multipoint network, a device is not required to have a direct connection to every other device. This is known as a partially meshed environment.

4.6.1.4 Neighbor routers and adjacencies

Routers that share a common network segment establish a neighbor relationship on the segment. Routers must agree on the following information to become neighbors:

- Area-id: The routers must belong to the same OSPF area.
- Authentication: If authentication is defined, the routers must specify the same password.
- Hello and dead intervals: The routers must specify the same timer intervals used in the Hello protocol. This protocol is further described in 4.6.2, “OSPF packet types” on page 165.
- Stub area flag: The routers must agree that the area is configured as a stub area. Stub areas are further described in 4.6.7, “OSPF stub areas” on page 172.

Once two routers have become neighbors, an adjacency relationship can be formed between the devices. Neighboring routers are considered adjacent when they have synchronized their topology databases. This occurs through the exchange of link state information.

4.6.1.5 Designated and backup designated router

The exchange of link state information between neighbors can create significant quantities of network traffic. To reduce the total bandwidth required to synchronize databases and advertise link state information, a router does not necessarily develop adjacencies with every neighboring device:

- Multi-access networks: Adjacencies are formed between an individual router and the (backup) designated router.
- Point-to-point networks: An adjacency is formed between both devices.

Each multi-access network elects a designated router (DR) and backup designated router (BDR). The DR performs two key functions on the network segment:

- It forms adjacencies with all routers on the multi-access network. This causes the DR to become the focal point for forwarding LSAs.
- It generates network link advertisements listing each router connected to the multi-access network. Additional information regarding network link advertisements is contained in 4.6.1.7, “Link state advertisements and flooding” on page 163.

The BDR forms the same adjacencies as the designated router. It assumes DR functionality when the DR fails.

Each router is assigned an 8-bit priority, indicating its ability to be selected as the DR or BDR. A router priority of zero indicates that the router is not eligible to be selected. The priority is configured on each interface in the router.

Figure 72 illustrates the relationship between neighbors. No adjacencies are formed between routers that are not selected to be the DR or BDR.

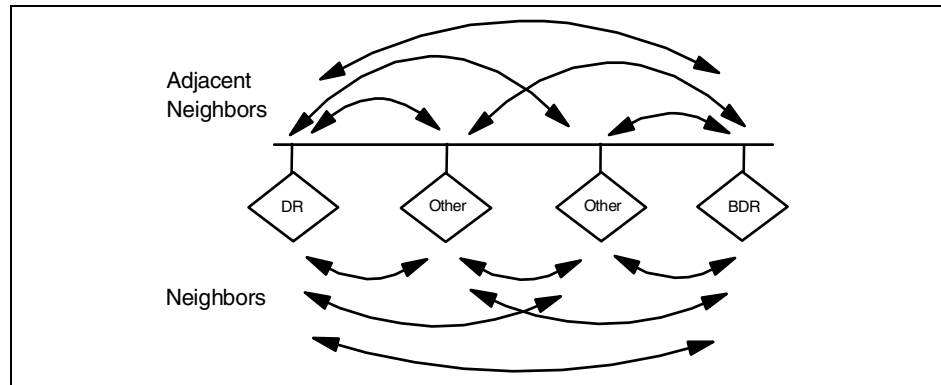


Figure 72. Relationship between adjacencies and neighbors

4.6.1.6 Link state database

The link state database is also called the *topology database*. It contains the set of link state advertisements describing the OSPF network and any external connections. Each router within the area maintains an identical copy of the link state database.

Note:

RFC 2328 uses the term link state database in preference to topology database. The former term has the advantage in that it describes the contents of the database. The latter term is more descriptive of the purpose of the database. This book has previously used the term topology database for this reason. However for the remainder of the OSPF section, we refer to it as the link state database.

4.6.1.7 Link state advertisements and flooding

The contents of an LSA describes an individual network component (that is, router, segment, or external destination). LSAs are exchanged between adjacent OSPF routers. This is done to synchronize the link state database on each device.

When a router generates or modifies an LSA, it must communicate this change throughout the network. The router starts this process by forwarding the LSA to each adjacent device. Upon receipt of the LSA, these neighbors store the information in their link state database and communicate the LSA to their neighbors. This store and forward activity continues until all devices receive the update. This process is called *reliable flooding*. Two steps are taken to ensure this flooding effectively transmits changes without overloading the network with excessive quantities of LSA traffic:

- Each router stores the LSA for a period of time before propagating the information to its neighbors. If, during that time, a new copy of the LSA arrives, the router replaces the stored version. However, if the new copy is outdated, it is discarded.
- To ensure reliability, each link state advertisement must be acknowledged. Multiple acknowledgements can be grouped together into a single acknowledgement packet. If an acknowledgement is not received, the original link state update packet is retransmitted.

Link state advertisements contain five types of information. Together these advertisements provide the necessary information needed to describe the entire OSPF network and any external environments:

- Router LSAs: This type of advertisement describes the state of the router's interfaces (links) within the area. They are generated by every OSPF router. The advertisements are flooded throughout the area.
- Network LSAs: This type of advertisement lists the routers connected to a multi-access network. They are generated by the DR on a multi-access segment. The advertisements are flooded throughout the area.
- Summary LSAs (Type-3 and Type-4): This type of advertisement is generated by an ABR. There are two types of summary link advertisements:
 - Type-3 summary LSAs describe routes to destinations in other areas within the OSPF network (inter-area destinations).
 - Type-4 summary LSAs describe routes to ASBRs.

Summary LSAs are used to exchange reachability information between areas. Normally, information is announced into the backbone area. The backbone then injects this information into other areas.

- AS external LSAs: This type of advertisement describes routes to destinations external to the OSPF network. They are generated by an ASBR. The advertisements are flooded throughout all areas in the OSPF network.

Figure 73 illustrates the different types of link state advertisements.

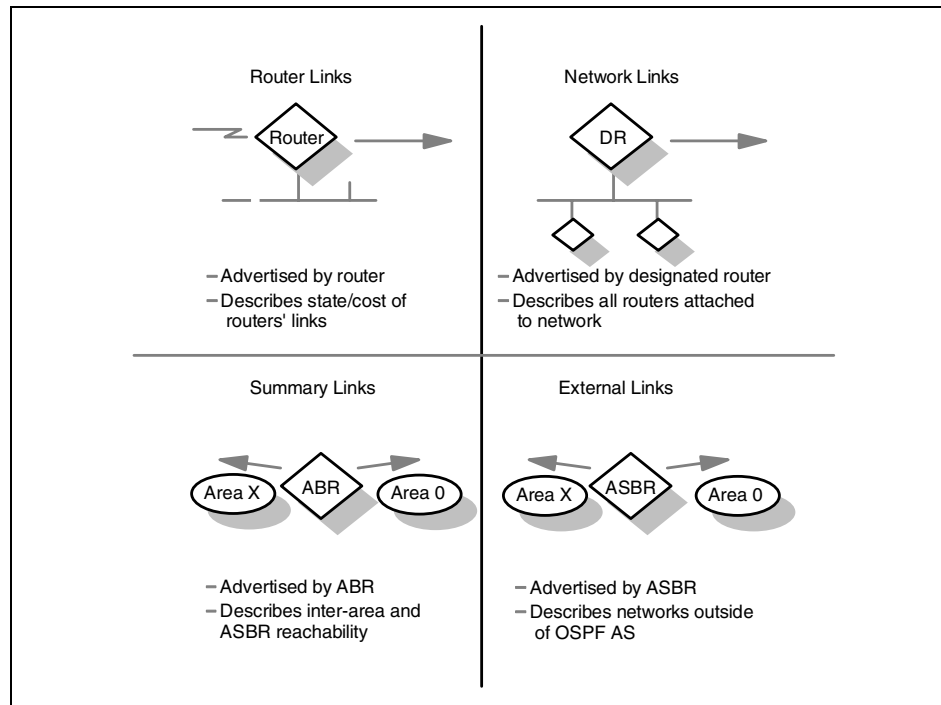


Figure 73. OSPF link state advertisements

4.6.2 OSPF packet types

OSPF packets are transmitted in IP datagrams. They are not encapsulated within TCP or UDP packets. The IP header uses protocol identifier 89. OSPF packets are sent with an IP ToS of 0 and an IP precedence of internetwork control. This is used to obtain preferential processing for the packets. Further discussion of ToS and IP precedence is located in 22.2, "Integrated Services" on page 782

Wherever possible, OSPF uses multicast facilities to communicate with neighboring devices. In broadcast and point-to-point environments, packets are sent to the reserved multicast address 224.0.0.5. RFC 2328 refers to this as the AllSPFRouters address. In non-broadcast environments, packets are addressed to the neighbor's specific IP address.

All OSPF packets share the common header shown in Figure 74. The header provides general information including area identifier, RID, checksum, and authentication information.

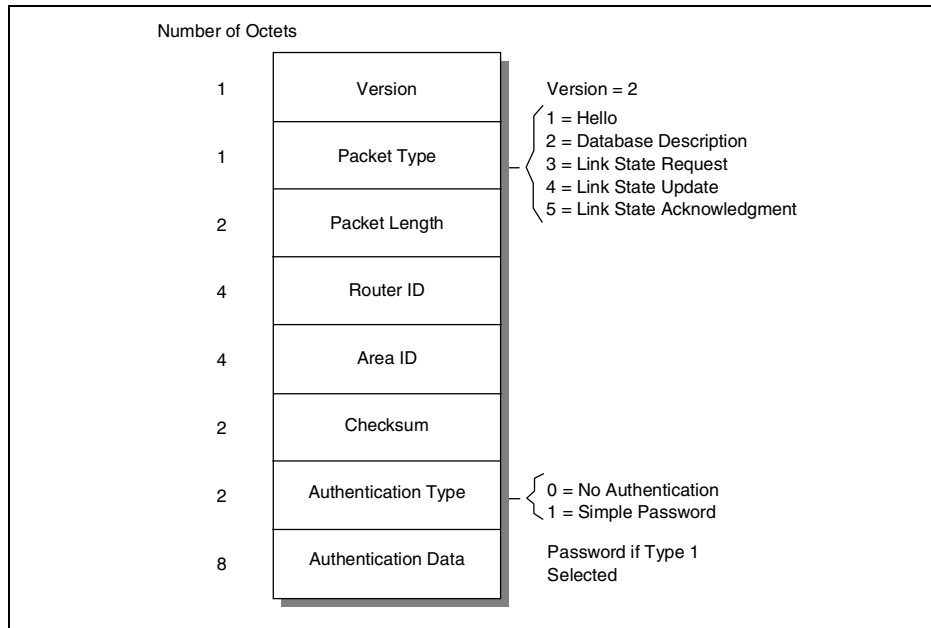


Figure 74. OSPF common header

The type field identifies the OSPF packet as one of five possible types:

- Hello: This packet type is used to discover and maintain neighbor relationships.
- Database description: This packet type describes the set of LSAs contained in the router's link state database
- Link state request: This packet type is used to request a more current instance of an LSA from a neighbor.
- Link state update: This packet type is used to provide a more current instance of an LSA to a neighbor.
- Link state acknowledgement: This packet type is used to acknowledge receipt of a newly received LSA.

The use of these packets is presented in the next section.

4.6.3 Neighbor communication

OSPF is responsible for determining the optimum set of paths through a network. To accomplish this, each router exchanges LSAs with other routers in the network. The OSPF protocol defines a number of activities to accomplish this information exchange:

- Discovering neighbors

- Electing a designated router
- Establishing adjacencies and synchronizing databases

The five OSPF packet types are used to support these information exchanges.

4.6.3.1 Discovering neighbors - the OSPF Hello protocol

The Hello protocol discovers and maintains relationships with neighboring routers. Hello packets are periodically sent out to each router interface. The packet contains the RID of other routers whose hello packets have already been received over the interface.

When a device sees its own RID in the hello packet generated by another router, these devices establish a neighbor relationship.

The hello packet also contains the router priority, DR identifier, and BDR identifier. These parameters are used to elect the DR on multi-access networks.

4.6.3.2 Electing a designated router

All multi-access networks must have a DR. A BDR may also be selected. The backup ensures there is no extended loss of routing capability if the DR fails.

The DR and BDR are selected using information contained in hello packets. The device with the highest OSPF router priority on a segment becomes the DR for that segment. The same process is repeated to select the BDR. In case of a tie, the router with the highest RID is selected. A router declared the DR is ineligible to become the BDR.

Once elected, the DR and BDR proceed to establish adjacencies with all routers on the multi-access segment.

4.6.3.3 Establishing adjacencies and synchronizing databases

Neighboring routers are considered adjacent when they have synchronized their link state databases. A router does not develop an adjacency with every neighboring device. On multi-access networks, adjacencies are formed only with the DR and BDR. This is a two step process:

Step 1: Database exchange process

The first phase of database synchronization is the database exchange process. This occurs immediately after two neighbors attempt to establish an adjacency. The process consists of an exchange of database description packets. The packets contain a list of the LSAs stored in the local database.

During the database exchange process, the routers form a master/slave relationship. The master is the first to transmit. Each packet is identified by a sequence number. Using this sequence number, the slave acknowledges each database description packet from the master. The slave also includes its own set of link state headers in the acknowledgements.

Step 2: Database loading

During the database exchange process, each router notes the link state headers for which the neighbor has a more current instance (all advertisements are time stamped). Once the process is complete, each router requests the more current information from the neighbor. This request is made with a link state request packet.

When a router receives a link state request, it must reply with a set of link state update packets providing the requested LSA. Each transmitted LSA is acknowledged by the receiver. This process is similar to the reliable flooding procedure used to transmit topology changes throughout the network.

Every LSA contains an age field indicating the time in seconds since the origin of the advertisement. The age continues to increase after the LSA is installed in the topology database. It also increases during each hop of the flooding process. When the maximum age is reached, the LSA is no longer used to determining routing information and is discarded from the link state database. This age is also used to distinguish between two otherwise identical copies of an advertisement.

4.6.4 OSPF neighbor state machine

The OSPF specification defines a set of neighbor states and the events that can cause a neighbor to transition from one state to another. A state machine is used to describe these transitions:

- **Down:** This is the initial state. It indicates that no recent information has been received from any device on the segment.
- **Attempt:** This state is used on non-broadcast networks. It indicates that a neighbor appears to be inactive. Attempts continue to reestablish contact.
- **Init:** Communication with the neighbor has started, but bidirectional communication has not been established. Specifically, a hello packet was received from the neighbor, but the local router was not listed in the neighbor's hello packet.
- **2-way:** Bidirectional communication between the two routers has been established. Adjacencies can be formed. Neighbors are eligible to be elected as designated routers.

- ExStart: The neighbors are starting to form an adjacency.
- Exchange: The two neighbors are exchanging their topology databases.
- Loading: The two neighbors are synchronizing their topology databases.
- Full: The two neighbors are fully adjacent and their databases are synchronized.

Network events cause a neighbor's OSPF state to change. For example, when a router receives a hello packet from a neighboring device, the OSPF neighbor state changes from Down to Init. When bidirectional communication has been established, the neighbor state changes from Init to 2-Way. RFC 2328 contains a complete description of the events causing a state change.

4.6.5 OSPF virtual links and transit areas

Virtual links are used when a network does not support the standard OSPF network topology. This topology defines a backbone area that directly connects to each additional OSPF area. The virtual link addresses two conditions:

- It may logically connect the backbone area when it is not contiguous.
- It may connect an area to the backbone when a direct connection does not exist.

A virtual link is established between two ABRs sharing a common non-backbone area. The link is treated as a point-to-point link. The common area is known as a *transit area*. Figure 75 illustrates the interaction between virtual links and transit areas when used to connect an area to the backbone.

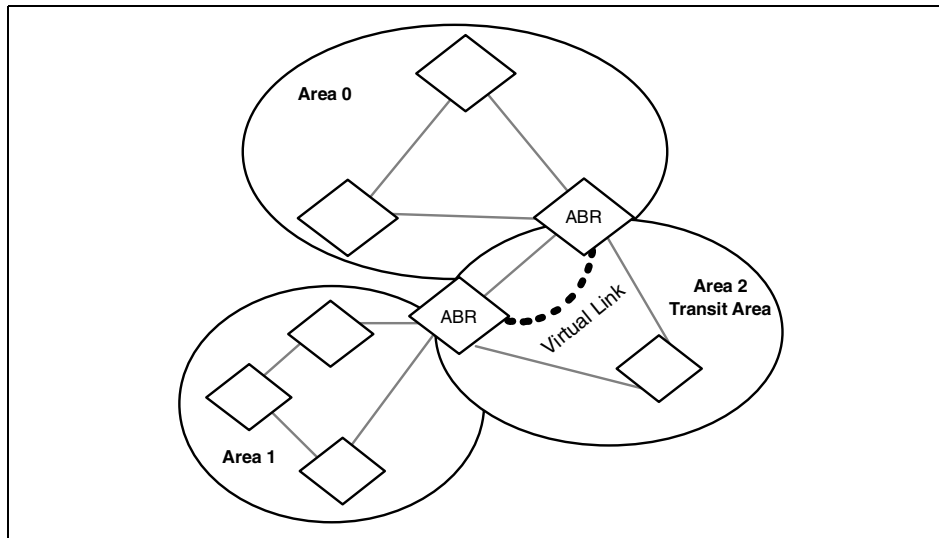


Figure 75. OSPF virtual link and transit areas

This diagram shows that area 1 does not have a direct connection to the backbone. Area 2 can be used as a transit area to provide this connection. A virtual link is established between the two ABRs located in area 2. Establishing this virtual link logically extends the backbone area to connect to area 1.

A virtual link is used only to transmit routing information. It does not carry regular traffic between the remote area and the backbone. This traffic, in addition to the virtual link traffic, is routed using the standard intra-area routing within the transit area.

4.6.6 OSPF route redistribution

Route redistribution is the process of introducing external routes into an OSPF network. These routes may be either static routes or routes learned via another routing protocol. They are advertised into the OSPF network by an ASBR. These routes become OSPF external routes. The ASBR advertises these routes by flooding OSPF AS external LSAs throughout the entire OSPF network.

The routes describe an end to end path consisting of two portions:

- External portion: This is the portion of the path external to the OSPF network. When these routes are distributed into OSPF, the ASBR assigns

an initial cost. This cost represents the *external cost* associated with traversing the external portion of the path.

- Internal portion: This is the portion of the path internal to the OSPF network. Costs for this portion of the network are calculated using standard OSPF algorithms.

OSPF differentiates between two types of external routes. They differ in the way the cost of the route is calculated. The ASBR is configured to redistribute the route as:

- External type 1: The total cost of the route is the sum of the external cost and any internal OSPF costs.
- External type 2: The total cost of the route is always the external cost. This ignores any internal OSPF costs required to reach the ASBR.

Figure 76 illustrates an example of the types of OSPF external routes.

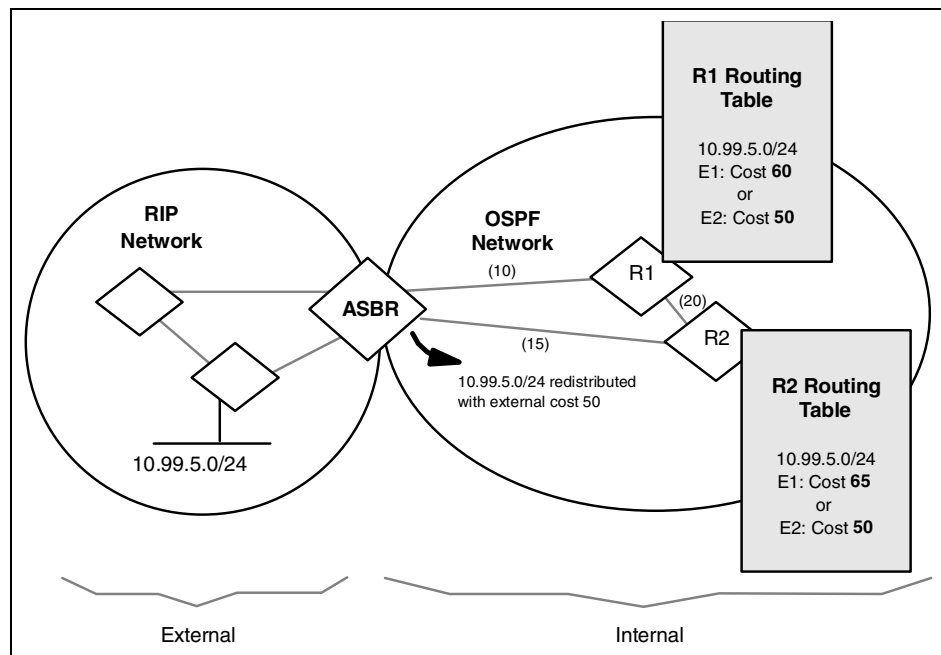


Figure 76. OSPF route redistribution

In this example, the ASBR is redistributing the 10.99.5.0/24 route into the OSPF network. This subnet is located within the RIP network. The route is announced into OSPF with an external cost of 50. This represents the cost for the portion of the path traversing the RIP network.

- If the ASBR redistributed the route as an E1 route, R1 will contain an external route to this subnet with a cost of 60 (50 + 10). R2 will have an external route with a cost of 65 (50 + 15).
- If the ASBR redistributed the route as an E2 route, both R1 and R2 will contain an external route to this subnet with a cost of 50. Any costs associated with traversing segments within the OSPF network are not included in the total cost to reach the destination.

4.6.7 OSPF stub areas

OSPF allows certain areas to be defined as a stub area. A stub area is created when the ABR connecting to a stub area excludes AS external LSAs from being flooded into the area. This is done to reduce the size of the link state database maintained within the stub area routers. Since there are no specific routes to external networks, routing to these destinations is based on a default route generated by the ABR. The link state databases maintained within the stub area contain only the default route and the routes from within the OSPF environment (for example, intra-area and inter-area routes).

Since a stub area does not allow external LSAs, a stub area cannot contain an ASBR. No external routes can be generated from within the stub area.

Stub areas can be deployed when there is a single exit point connecting the area to the backbone. An area with multiple exit points can also be a stub area. However, there is no guarantee that packets exiting the area will follow an optimal path. This is due to the fact that each ABR generates a default route. There is no ability to associate traffic with a specific default routes.

All routers within the area must be configured as stub routers. This configuration is verified through the exchange of hello packets.

4.6.7.1 Not-so-stubby areas

An extension to the stub area concept is the *not-so-stubby area (NSSA)*. This alternative is documented in RFC 1587. An NSSA is similar to a stub area in that the ABR servicing the NSSA does not flood any external routes into the NSSA. The only routes flooded into the NSSA are the default route and any other routes from within the OSPF environment (for example, intra-area and inter-area).

However, unlike a stub area, an ASBR can be located within an NSSA. This ASBR can generate external routes. Therefore, the link state databases maintained within the NSSA contain the default route, routes from within the OSPF environment (for example, intra-area and inter-area routes), and the external routes generated by the ASBR within the area.

The ABR servicing the NSSA floods the external routes from within the NSSA throughout the rest of the OSPF network.

4.6.8 OSPF route summarization

Route summarization is the process of consolidating multiple contiguous routing entries into a single advertisement. This reduces the size of the link state database and the IP routing table. In an OSPF network, summarization is performed at a border router. There are two types of summarization:

- **Inter-area route summarization:** Inter-area summarization is performed by the ABR for an area. It is used to summarize route advertisements originating within the area. The summarized route is announcement into the backbone. The backbone receives the aggregated route and announces the summary into other areas.
- **External route summarization:** This type of summarization applies specifically to external routes injected into OSPF. This is performed by the ASBR distributing the routes into the OSPF network.

Figure 77 illustrates an example of OSPF route summarization.

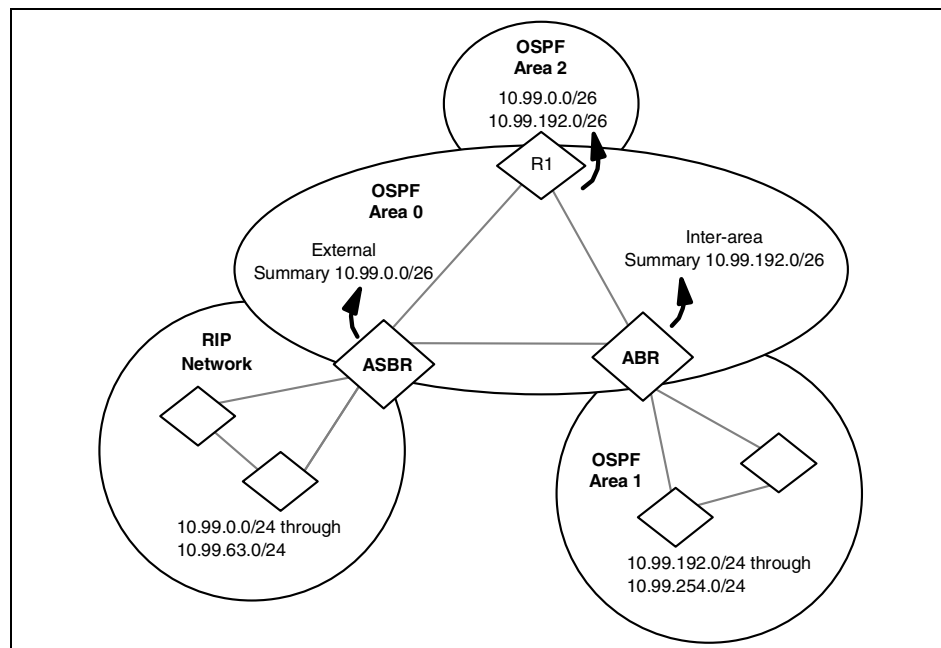


Figure 77. OSPF route summarization

In this figure, the ASBR is advertising a single summary route for the 64 subnetworks located in the RIP environment. This single summary route is flooded throughout the entire OSPF network. In addition, the ABR is generating a single summary route for the 64 subnetworks located in area 1. This summary route is flooded through area 0 and area 2. Depending on the configuration of the ASBR, the inter-area summary route may also be redistributed into the RIP network.

4.7 Enhanced Interior Gateway Routing Protocol (EIGRP)

The Enhanced Interior Gateway Routing Protocol (EIGRP) is categorized as a hybrid routing protocol. Similar to a distance vector algorithm, EIGRP uses metrics to determine network paths. However, like a link state protocol, topology updates in an EIGRP environment are event driven.

EIGRP, as the name implies, is an interior gateway protocol designed for use within an AS. In properly designed networks, EIGRP has the potential for improved scalability and faster convergence over standard distance vector algorithms. EIGRP is also better positioned to support complex, highly redundant networks.

EIGRP is a proprietary protocol developed by Cisco Systems, Inc. At the time of this writing, it is not an IETF standard protocol.

4.7.1 Features of EIGRP

EIGRP provides several benefits. Some of these benefits are also available in distance vector or link state algorithms.

- **Faster convergence:** EIGRP maintains a list of alternate routes that can be used if a preferred path fails. When the path fails, the new route is immediately installed in the IP routing table. No route recomputation is performed.
- **Partial routing updates:** When EIGRP discovers a neighboring router, each device exchanges their entire routing table. After the initial information exchange, only routing table changes are propagated. There is no periodic rebroadcasting of the entire routing table.
- **Low bandwidth utilization:** During normal network operations, only hello packets are transmitted through a stable network.
- **CIDR and VLSM:** EIGRP supports supernetting and variable length subnet masks. This allows the network administrator to efficiently allocate IP address resources.

- Route summarization: EIGRP supports the ability to summarize routing announcements. This limits the advertisement of unnecessary subnet information.
- Multiple protocols: EIGRP can provide network layer routing for AppleTalk, IPX and IP networks.
- Unequal cost load balancing: EIGRP supports the simultaneous use of multiple unequal cost paths to a destination. Each route is installed in the IP routing table. EIGRP also intelligently load balances traffic over the multiple paths.

4.7.2 Terminology

EIGRP uses specific terminology to describe the operation of the protocol:

- Successor: For a specific destination, the successor is the neighbor router currently used for packet forwarding. This device has the least-cost path to the destination and is guaranteed not to be participating in a routing loop. To reach the target network shown in Figure 78, router B is the current successor for router A.
- Feasible Successor: A feasible successor assumes forwarding responsibility when the current successor router fails. The set of feasible successors represent the devices that can become a successor without requiring a route recomputation or introducing routing loops.

The set of feasible successors to a destination is determined by reviewing the complete list of minimum cost paths advertised by neighboring routers. From this list, neighbors that have an advertised metric less than the current routing table metric are considered feasible successors.

Figure 78 provides an example of a feasible successor relationship.

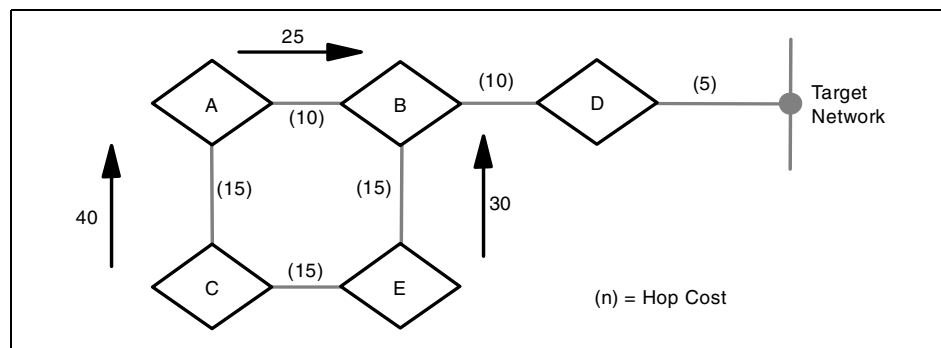


Figure 78. EIGRP feasible successors

In this diagram, the costs to reach the target network are shown. For example, the cost from router C to the target network is 40 (15 + 10 + 10 + 5). The cost from router E to the target network is 30 (15 + 10 + 5).

Router E is advertising a cost (30) that is less than the current routing table metric on router C (40). Therefore, router C recognizes router E as a feasible successor to reach the target network. Note that the reverse is not true. The cost advertised by router C (40) is more than the current route on router E (30). Therefore, router E does not recognize router C as a feasible successor to the destination network.

- Neighbor table: EIGRP maintains a table to track the state of each adjacent neighbor. The table contains the address and interface used to reach the neighbor. It also contains the last sequence number contained in a packet from the neighbor. This allows the reliable transport mechanism of EIGRP to detect out-of-order packets.
- Topology table: EIGRP uses a topology table to install routes into the IP routing table. The topology table lists all destination networks currently advertised by neighboring routers. The table contains all the information needed to build a set of distances and vectors to each destination. This information includes:
 - Smallest bandwidth available on a segment used to reach this destination.
 - Total delay, reliability, and loading of the path.
 - Minimum MTU used on the path.
 - The feasible distance of the path. This represents the best metric along the path to the destination network. It including the metric used to reach the neighbor advertising the path.
 - The reported distance of the path. This represents the total metric along the path to a destination network as advertised by an upstream neighbor.
 - The source of the route. EIGRP marks external routes. This provides the ability to implement policy controls that customize routing patterns.

An entry in the topology table can have one of two states:

- Passive state: The router is not performing a route recomputation for the entry.
 - Active state: The router is performing a route recomputation for the entry. If a feasible successor exists for a route, the entry never enters this state. This avoids processor-intensive route recomputation.
- Reliable transport protocol: EIGRP can guarantee the ordered delivery of packets to a neighbor. However, not all types of packets must be reliably transmitted. For example, in a network that supports multicasting, there is

no need to send individual, acknowledged hello packets to each neighbor. To provide efficient operation, reliability is provided only when needed. This improves convergence time in networks containing varying speed connections.

4.7.3 Neighbor discovery and recovery

EIGRP can dynamically learn about other routers on directly attached networks. This is similar to the Hello protocol used for neighbor discovery in an OSPF environment.

Devices in an EIGRP network exchange hello packets to verify each neighbor is operational. Like OSPF, the frequency used to exchange packets is based on the network type. Packets are exchanged at a five second interval on high bandwidth links (for example, LAN segments). Otherwise, hello packets on lower bandwidth connections are exchanged every 60 seconds.

Like OSPF, EIGRP uses a hold timer to remove inactive neighbors. This timer indicates the amount of time that a device will continue to consider a neighbor active without receiving a hello packet from the neighbor.

4.7.4 The DUAL algorithm

A typical distance vector protocol uses periodic updates to compute the best path to a destination. It uses distance, next hop, and local interface costs to determine the path. Once this information is processed, it is discarded. EIGRP does not rely on periodic updates to converge on the topology. Instead, it builds a topology table containing each of its neighbor's advertisements. Unlike a distance vector protocol, this data is not discarded.

EIGRP processes the information in the topology table to determine the best paths to each destination network. EIGRP implements an algorithm known as DUAL (Diffusing Update ALgorithm). This algorithm provides several benefits:

- The DUAL algorithm guarantees loop-free operations throughout the route computation and convergence period.
- The DUAL algorithm allows all routers to synchronize at the same time. This is unlike a RIP environment, in which the propagation of routing updates causes devices to converge at different rates.
- The DUAL algorithm allows routers not involved with a topology change to avoid route recomputation.

The DUAL algorithm is used to find the set of feasible successors for a destination. When an adverse condition occurs in the network, the alternate route is immediately added to the IP routing table. This avoids unnecessary

computation to determine an alternate path. If no feasible successor is known, a route recomputation occurs. This behavior is shown in Figure 79 and Figure 80.

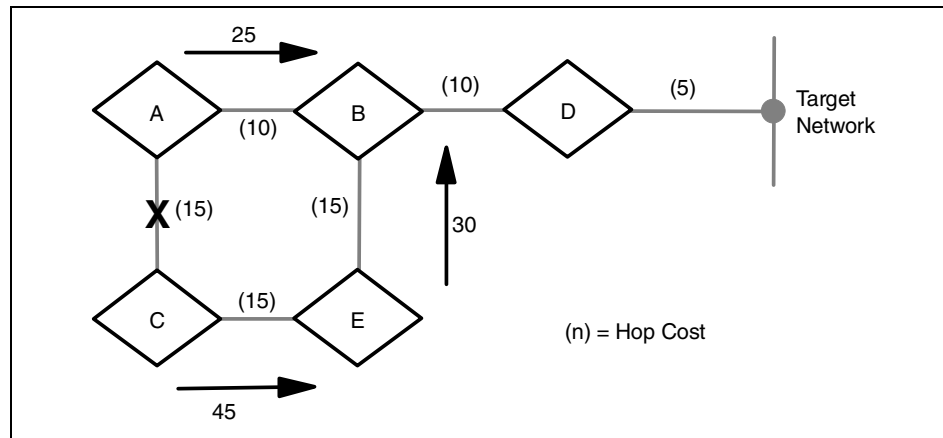


Figure 79. Using a feasible successor

In this example, router C uses router E as a feasible successor to reach the target network. If the connection between router A and router C fails, router C will immediately reroute traffic through router E. The new route is updated in the IP routing table.

4.7.4.1 Route recomputation

A route recomputation occurs when there is no known feasible successor to the destination. The process starts with a router sending a multicast query packet to determine if any neighbor is aware of a feasible successor to the destination. A neighbor replies if it has an feasible successor. If the neighbor does not have feasible successor, the neighbor may return a query indicating it also is performing a route recomputation.

Figure 80 shows an example of querying to determine a feasible successor. In this example, router E does not have a feasible successor to the target network. When the link connecting router E and router B fails, router E must determine a new path. Router E sends a multicast query to each of its neighbors. Router C has a feasible successor and responds to router E. Router E updates its IP routing table with the new path at a cost of 55.

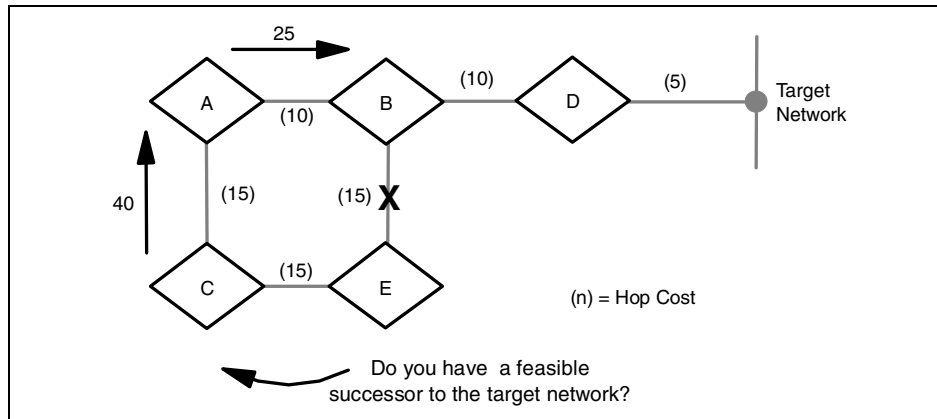


Figure 80. Query for a feasible successor

When the link to a neighbor fails, all routes that used that neighbor as the only feasible successor require a route recomputation.

4.7.4.2 EIGRP metrics

EIGRP uses a mathematical formula to determine the metric associated with a path. By default, the formula references the minimum bandwidth of a segment used to reach the destination. It also sums the delays on the path. The default formula to determine the metric is:

$$\left[\left(\frac{10^7}{\text{minbandwidth}} \right) + \text{sumofdelays} \right] \times 256$$

EIGRP supports the inclusion of other measurements in the metric calculation.

4.7.5 EIGRP packet types

EIGRP uses five types of packets to establish neighbor relationships and advertise routing information:

- Hello/Acknowledgement: These packets are used for neighbor discovery. They are multicast advertised on each network segment. Unicast responses to the hello packet are returned.

A hello packet without any data is considered an acknowledgement.

- Updates: These packets are used to convey reachability information for each destination. When a new neighbor is discovered, unicast update packets are exchanged to allow each neighbor to build their topology

table. Other types of advertisements (e.g., metric changes) use multicast packets. Update packets are always transmitted reliably.

- Queries and replies: These packets are exchanged when a destination enters an active state. A multicast query packet is sent to determine if any neighbor contains a feasible successor to the destination. Unicast reply packets are sent to indicate that the neighbor does not need to go into an active state because a feasible successor has been identified. Query and reply packets are transmitted reliably.
- Request: These packets are used to obtain specific information from a neighbor. These packets are used in route server applications.

4.8 Exterior Gateway Protocol (EGP)

EGP is an exterior gateway protocol of historical merit. It was one of the first protocols developed for communication between autonomous systems. It is described in RFC 904.

EGP assumes the network contains a single backbone and a single path exists between any two autonomous systems. Due to this limitation, the current use of EGP is minimal. In practice, EGP has been replaced by BGP.

EGP is based on periodic polling using a hello/I-hear-you message exchange. These are used to monitor neighbor reachability and solicit update responses.

The gateway connecting to an AS is permitted to advertise only those destination networks reachable within the local AS. It does not advertise reachability information about its EGP neighbors outside the AS.

4.9 Border Gateway Protocol (BGP)

The Border Gateway Protocol (BGP) is an exterior gateway protocol. It was originally developed to provide a loop-free method of exchanging routing information between autonomous systems. BGP has since evolved to support aggregation and summarization of routing information.

BGP is an IETF draft standard protocol described in RFC 1771. The version described in this RFC is BGP Version 4. Following standard convention, this document uses the term BGP when referencing BGP Version 4.

4.9.1 BGP concepts and terminology

BGP uses specific terminology to describe the operation of the protocol. Figure 81 is used to illustrate this terminology.

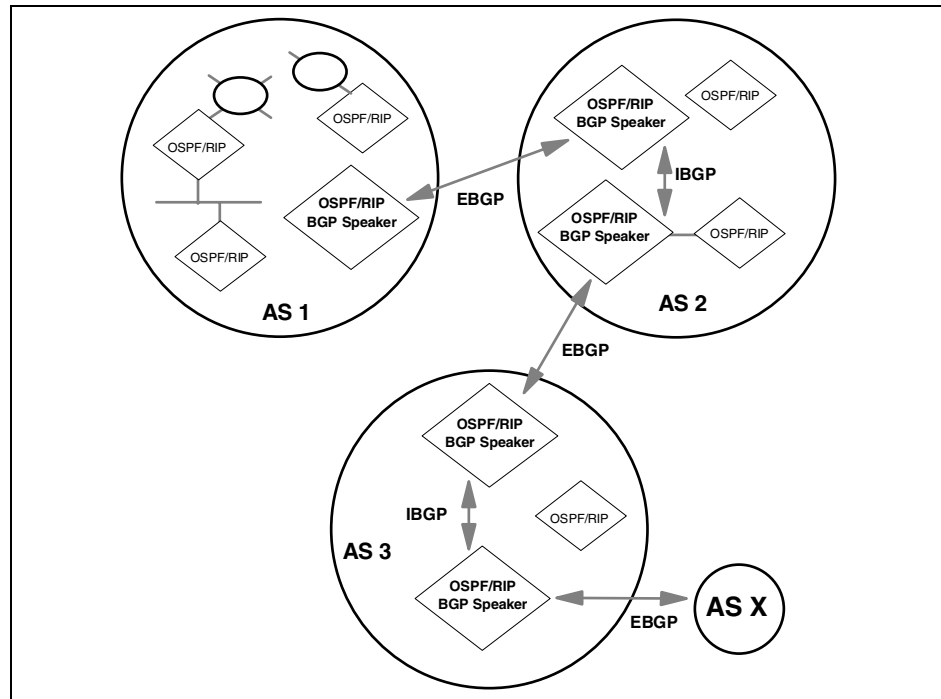


Figure 81. Components of a BGP network

- BGP speaker: A router configured to support BGP.
- BGP neighbors (peers): A pair of BGP speakers that exchange routing information. There are two types of BGP neighbors:
 - Internal (IBGP) neighbor: A pair of BGP speakers within the same AS.
 - External (EBGP) neighbor: A pair of BGP neighbors, each in a different AS. These neighbors typically share a directly connected network.
- BGP session: A TCP session connecting two BGP neighbors. The session is used to exchange routing information. The neighbors monitor the state of the session by sending keepalive messages.¹

¹ This keepalive message is implemented in the application layer. It is independent of the keepalive message available in many TCP implementations.

- Traffic type: BGP defines two types of traffic:
 - Local: Traffic local to an AS either originates or terminates within the AS. Either the source or the destination IP address resides in the AS.
 - *Transit*: Any traffic that is not local traffic is transit traffic. One of the goals of BGP is to minimize the amount of transit traffic.
- AS type: BGP defines three types of autonomous systems:
 - Stub: A stub AS has a single connection to one other AS. A stub AS carries only local traffic.
 - Multihomed: A multihomed AS has connections to two or more autonomous systems. However, a multihomed AS has been configured so that it does not forward transit traffic.
 - Transit: A transit AS has connections to two or more autonomous systems and carries both local and transit traffic. The AS may impose policy restrictions on the types of transit traffic that will be forwarded.

Depending on the configuration of the BGP devices within AS 2 in Figure 81, this autonomous system may be either a multihomed AS or a transit AS.

- AS number: A 16-bit number uniquely identifying an AS.
- AS path: A list of AS numbers describing a route through the network. A BGP neighbor communicates paths to its peers.
- Routing policy: A set of rules constraining the flow of data packets through the network. Routing policies are not defined in the BGP protocol. Rather, they are used to configure a BGP device. For example, a BGP device may be configured so that:
 - A multihomed AS can refuse to act as a transit AS. This is accomplished by advertising only those networks contained within the AS.
 - A multihomed AS can perform transit AS routing for a restricted set of adjacent autonomous systems. It does this by tailoring the routing advertisements sent to EBGp peers.
 - An AS can optimize traffic to use a specific AS path for certain categories of traffic.
- Network layer reachability information (NLRI): NLRI is used by BGP to advertise routes. It consists of a set of networks represented by the tuple <length,prefix>. For example, the tuple <14,220.24.106.0> represents the CIDR route 220.24.106.0/14.

- Routes and paths: A route associates a destination with a collection of attributes describing the path to the destination. The destination is specified in NRI format. The path is reported as a collection of path attributes. This information is advertised in UPDATE messages. Additional information describing the UPDATE message is located in 4.9.3, “Protocol description” on page 185.

4.9.2 IBGP and EBGP communication

BGP does not replace the IGP operating within an AS. Instead, it cooperates with the IGP to establish communication between autonomous systems. BGP within an AS is used to advertise the local IGP routes. These routes are advertised to BGP peers in other autonomous systems. Figure 82 illustrates the communication that occurs between BGP peers. This example shows four autonomous systems. AS 2, AS 3 and AS 4 each have an EBGP connection to AS 1. A full mesh of IBGP sessions exists between BGP devices within AS 1.

Network 10.0.0.0/8 is located within AS 3. Using BGP, the existence of this network is advertised to the rest of the environment:

- R4 in AS 3 uses its EBGP connection to announce the network to AS 1.
- R1 in AS 1 uses its IBGP connections to announce the network to R2 and R3.
- R2 in AS 1 uses its EBGP session to announce the network into AS 2. R3 in AS 1 uses its EBGP session 5 to announce the network into AS 4.

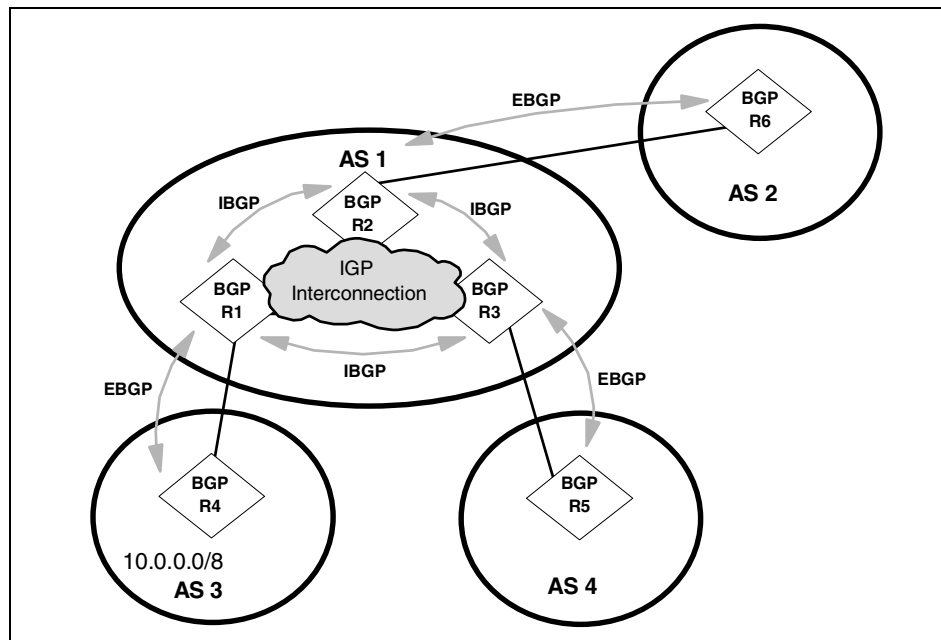


Figure 82. EBGP and IBGP communication

Several additional operational issues are shown in Figure 82:

- Role of BGP and the IGP: The diagram shows that while BGP alone carries information between autonomous systems, both BGP and the IGP are used to carry information through an AS.
- Establishing the TCP session between peers: Before establishing a BGP session, a device verifies that routing information is available to reach the peer:
 - EBGP peers: EBGP peers typically share a directly connected network. The routing information needed to exchange BGP packets between these peers is trivial.
 - IBGP peers: IBGP peers can be located anywhere within the AS. They do not need to be directly connected. BGP relies on the IGP to locate a peer. Packet forwarding between IBGP peers uses IGP-learned routes.
- Full mesh of BGP sessions within an AS: IBGP speakers assume a full mesh of BGP sessions have been established between peers in the same AS. In Figure 82, all three BGP peers in AS 1 are interconnected with BGP sessions.

When a BGP speaker receives a route update from an IBGP peer, the receiving speaker uses EBGP to propagate the update to external peers.

Since the receiving speaker assumes a full mesh of IBGP sessions have been established, it does not propagate the update to other IBGP peers.

For example, assume there was no IBGP session between R1 and R3 in Figure 82. R1 receives the update about 10.0.0.0/8 from AS 3. R1 forwards the update to its BGP peers, namely R2. R2 receives the IBGP update and forwards it to its EBGP peers, namely R6. No update is sent to R3. If R3 needs to receive this information, R1 and R3 must be configured to be BGP peers.

4.9.3 Protocol description

BGP establishes a reliable TCP connection between peers. Sessions are established using TCP port 179. BGP assumes the transport connection will manage fragmentation, retransmission, acknowledgement, and sequencing.

When two speakers initially form a BGP session, they exchange their entire routing table. This routing information contains the complete AS path used to reach each destination. The information avoids the routing loops and counting-to-infinity behavior observed in RIP networks. Once the entire table has been exchanged, changes to the table are communicated as incremental updates.

4.9.3.1 BGP packet types

All BGP packets contain a standard header. The header specifies the BGP packet type. The valid BGP packet types include:

- OPEN²: This message type is used to establish a BGP session between two peer nodes.
- UPDATE: This message type is used to transfer routing information between BGP peers.
- NOTIFICATION: This message is sent when an error condition is detected.
- KEEPALIVE: This message is used to determine if peers are reachable.

Figure 83 shows the flow of these message types between two autonomous systems.

² RFC 1771 uses uppercase to name BGP messages. The same convention is used in this section.

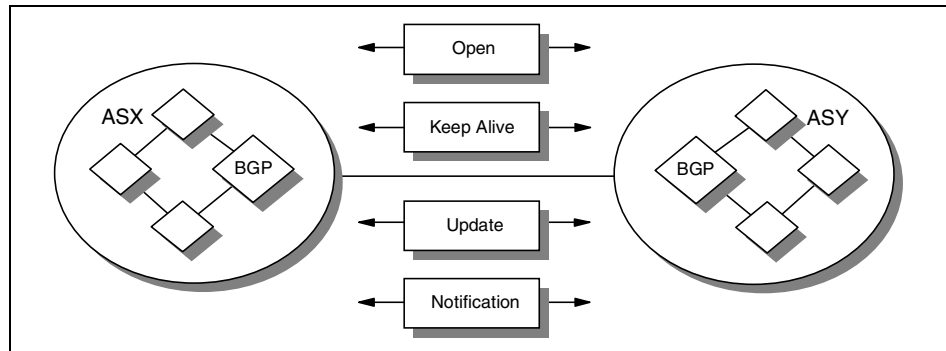


Figure 83. BGP message flows between BGP speakers

4.9.3.2 Opening and confirming a BGP connection

Once a TCP session has been established between two peer nodes, each router sends an OPEN message to the neighbor. The open message includes:

- The originating router's AS number and BGP router identifier.
- A suggested value for the hold timer. The function of this timer is discussed in the next section.
- Optional parameters. This information is used to authenticate a peer.

An OPEN message contains support for authenticating the identity of a BGP peer. However, the BGP standard does not specify a specific authorization mechanism. This allows BGP peers to select any supported authorization scheme.

An OPEN message is acknowledged by a KEEPALIVE message. Once peer routers have established a BGP connection, they can exchange additional information.

4.9.3.3 Maintaining the BGP connection

BGP does not use any transport-based keep-alive to determine if peers are reachable. Instead, BGP messages are periodically exchanged between peers. If no messages are received from the peer for the duration specified by the hold timer, the originating router assumes an error has occurred. When this happens, an error notification is sent to the peer and the connection is closed.

RFC 1771 recommends a 90 second hold timer and a 30 second keepalive timer.

4.9.3.4 Sending reachability information

Reachability information is exchanged between peers in UPDATE messages. BGP does not require a periodic refresh of the entire BGP routing table. Therefore, each BGP speaker must retain a copy of the current BGP routing table used by each peer. This information is maintained for the duration of the connection. Once neighbors have performed the initial exchange of complete routing information, only incremental updates to that information are exchanged.

An UPDATE message is used to advertise feasible routes or withdraw infeasible routes. The message may simultaneously advertise a feasible route and withdraw multiple infeasible routes from service. Figure 84 depicts the format of an UPDATE message:

- Network Layer Reachability Information (NLRI)
- Path attributes (Path attributes are discussed in 4.9.4.1, “Path attributes” on page 188)
- Withdrawn routes

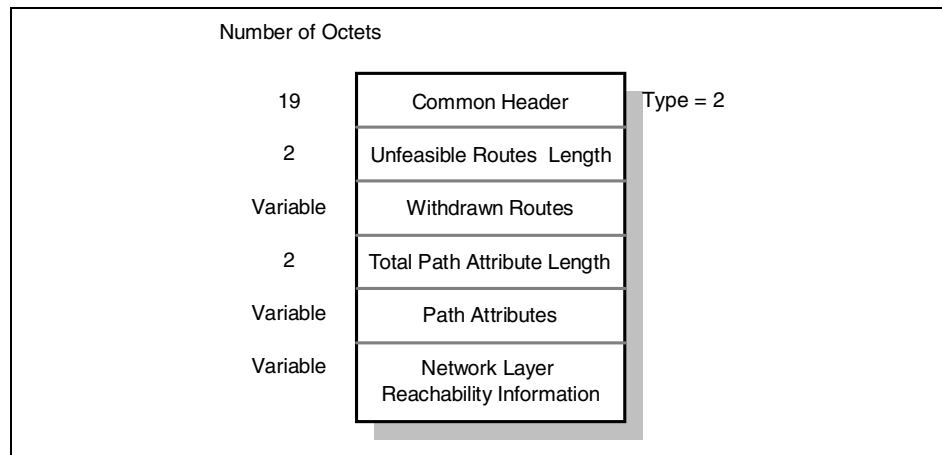


Figure 84. BGP UPDATE message

Several path attributes may be used to describe a route.

Withdrawn routes

The unfeasible routes length field indicates the total length of the withdrawn routes field.

The withdrawn routes field provides a list of IP addresses prefixes that are not feasible or are no longer in service. These addresses need to be

withdrawn from the BGP routing table. The withdrawn routes are represented in the same tuple-format as the NLRI.

4.9.3.5 Notification of error conditions

A BGP device may observe error conditions impacting the connection to a peer. NOTIFICATION messages are sent to the neighbor when these conditions are detected. Once the message is sent, the BGP transport connection is closed. This means all resources for the BGP connection are deallocated. The routing table entries associated with the remote peer are marked as invalid. Finally, other peers are notified that these routes are invalid.

Notification messages include an error code and an error subcode.

The error codes provided by BGP include:

- Message header error
- OPEN message error
- UPDATE message error
- Hold timer expired
- Finite state machine error
- Cease

The error subcode further qualifies the specific error. Each error code may have multiple subcodes associated with it.

4.9.4 Path selection

BGP is a distance vector protocol. In traditional distance vector protocols, a single metric (for example, hop-count) is associated with a path. The best path is obtained by comparing the metrics of each feasible route. However, inter-AS routing complicates this process. There are no universally agreed-upon metrics that can be used to evaluate external paths. Each AS has its own set of criteria for path evaluation.

4.9.4.1 Path attributes

Path attributes are used to describe and evaluate a route. Peers exchange path attributes along with other routing information. When a device advertises a route, it may add or modify the path attributes before advertising the route to a peer. The combination of attributes are used to select the best path.

Each path attribute is placed into one of four separate categories:

- Well-known mandatory: The attribute must be recognized by all BGP implementations. It must be sent in every UPDATE message.

- Well-known discretionary: The attribute must be recognized by all BGP implementations. However, it is not required to be sent in every UPDATE message.
- Optional transitive: It is not required that every BGP implementation recognize this type of attribute. A path with an unrecognized optional transitive attribute is accepted and simply forwarded to other BGP peers.
- Optional non-transitive: It is not required that every BGP implementation recognize this type of attribute. These attributes can be ignored and not passed along to other BGP peers.

BGP defines seven attribute types to define an advertised route:

- ORIGIN: This attribute defines the origin of the path information. Valid selections are IGP (interior to the AS), EGP, or INCOMPLETE. This is a well-known mandatory attribute.
- AS_PATH: This attribute defines the set of autonomous systems which must be traversed to reach the advertised network. Each BGP device prepends its AS number onto the AS path sequence before sending the routing information to an EBGP peer. Using the sample network depicted in Figure 82 on page 184, R4 advertises network 10.0.0.0 with an AS_PATH of 3. When the update traverses AS 1, R2 prepends its own AS number to it. When the routing update reaches R6, the AS_PATH attribute for network 10.0.0.0 is <1 3>. This is a well-known mandatory attribute.
- NEXT_HOP: This attribute defines the IP address of the next hop used to reach the destination. This is a well-known mandatory attribute.

For routing updates received over EBGP connections, the next hop is typically the IP address of the EBGP neighbor in the remote AS. BGP specifies that this next hop is passed without modification to each IBGP neighbor. As a result, each IBGP neighbor must have a route to reach the neighbor in the remote AS. Figure 85 illustrates this interaction.

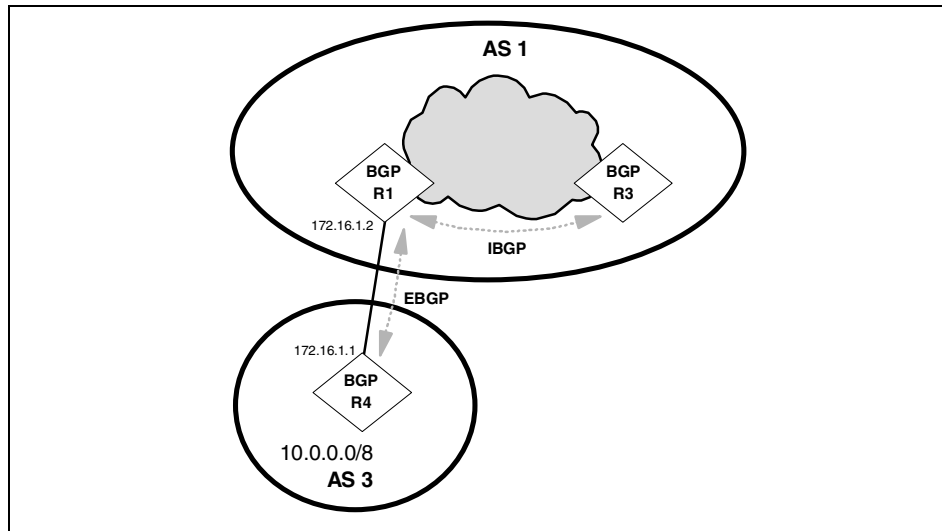


Figure 85. NEXT_HOP attribute

In this example, when a routing update for network 10.0.0.0/8 is sent from AS 3, R1 receives the update with the NEXT_HOP attribute set to 172.16.1.1. When this update is forwarded to R3, the next hop address remains 172.16.1.1. R3 must have appropriate routing information to reach this address. Otherwise, R3 will drop packets destined for AS 3 if the next hop is inaccessible.

- **MULTI_EXIT_DISC** (multi-exit discriminator, MED): This attribute is used to discriminate among multiple exit points to a neighboring AS. If this information is received from an EBGP peer, it is propagated to each IBGP peer. This attribute is not propagated to peers in other autonomous systems. If all other attributes are equal, the exit point with the lowest MED value is preferred. This is an optional non-transitive attribute.
- **LOCAL_PREF** (local preference): This attribute is used by a BGP speaker to inform other speakers within the AS of the originating speaker's degree of preference for the advertised route. Unlike MED, this attribute is used only within an AS. The value of the local preference is not distributed outside an AS. If all other attributes are equal, the route with the higher degree of preference is preferred. This is a well-known discretionary attribute.
- **ATOMIC_AGGREGATE**: This attribute is used when a BGP peer receives advertisements for the same destination identified in multiple, non-matching routes (that is, overlapping routes). One route describes a smaller set of destinations (a more specific prefix), other routes describe a

larger set of destinations (a less specific prefix). This attribute is used by the BGP speaker to inform peers that it has selected the less specific route without selecting the more specific route. This is a well-known discretionary attribute.

A route with this attribute included may actually traverse autonomous systems not listed in the AS_PATH.

- **AGGREGATOR:** This attribute indicates the last AS number that formed the aggregate route, followed by the IP address of the BGP speaker that formed the aggregate route. Further information about route aggregation is located in 4.9.6, “BGP aggregation” on page 193. This is an optional transitive attribute.

4.9.4.2 Decision process

The process to select the best path uses the path attributes describing each route. The attributes are analyzed and a *degree of preference* is assigned. Since there may be multiple paths to a given destination, the route selection process determines the degree of preference for each feasible route. The path with the highest degree of preference is selected as the best path. This is the path advertised to each BGP neighbor.

Route aggregation can also be performed during this process.

Where there are multiple paths to a destination, BGP tracks each individual path. This allows faster convergence to the alternate path when the primary path fail.

4.9.5 BGP synchronization

Figure 86 shows an example of an AS providing transit service. In this example, AS 1 is used to transport traffic between AS 3 and AS 4. Within AS 1, R2 is not configured for BGP. However, R2 is used for communication between R1 and R3. Traffic between these two BGP nodes physically traverses through R2.

Using the routing update flow described earlier, the 10.0.0.0/8 network is advertised using the EBGP connection between R4 and R1. R1 passes the network advertisement to R3 using its existing IBGP connection. Since R2 is not configured for BGP, it is unaware of any networks in AS 3. A problem occurs if R3 needs to communicate with a device in AS 3. R3 passes the traffic to R2. However, since R2 does not have any routes to AS 3 networks, the traffic is dropped.

If R3 advertises the 10.0.0.0/8 network to AS 4, the problem continues. If AS 4 needs to communicate with a device in AS 3, the packets are forwarded from R5 to R3. R3 forwards the packets to R2 where they are discarded.

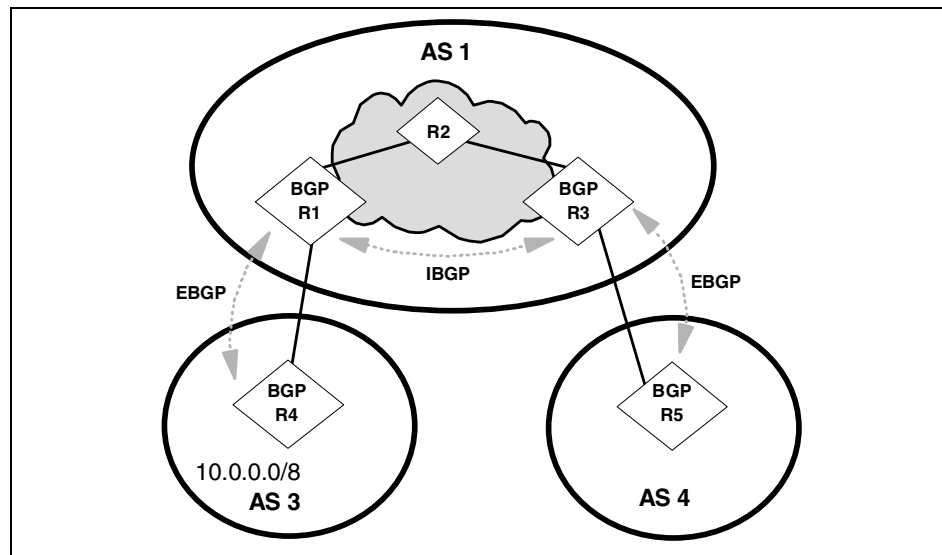


Figure 86. BGP synchronization

This situation is addressed by the synchronization rule of BGP. The rule states that a transit AS will not advertise a route before all routers within the AS have learned about the route. In this example, R3 will not advertise the existence of the networks in AS 3 until R2 has built a proper routing table.

There are three methods to implement the synchronization rule:

- Enable BGP on all devices within the transit AS. In this solution, R2 would have an IBGP session with both R1 and R3. R2 learns of the 10.0.0.0/8 network at the same time it is advertised to R3. At that time, R3 announces the routes to its peer in AS 4.
- Redistribute the routes into the IGP used within the transit area. In this solution, R1 redistributes the 10.0.0.0/8 network into the IGP within AS 1. R3 learns of the network via two routing protocols: BGP and the IGP. Once R3 learns of the network via the IGP, it is certain that other routers within the AS have also learned of the routes. At that time, R3 announces the routes to its peer in AS 4.
- Encapsulate the transit traffic across the AS. In this solution, transit traffic is encapsulated within IP datagrams addressed to the exit gateway. Since this does not require the IGP to carry exterior routing information, no

synchronization is required between BGP and the IGP. R3 can immediately announce the routes to its peer in AS 4.

4.9.6 BGP aggregation

The major improvement introduced in BGP Version 4 was support for CIDR and route aggregation. These features allow BGP peers to consolidate multiple contiguous routing entries into a single advertisement. It significantly enhances the scalability of BGP into large internetworking environments. These functions are illustrated in Figure 87.

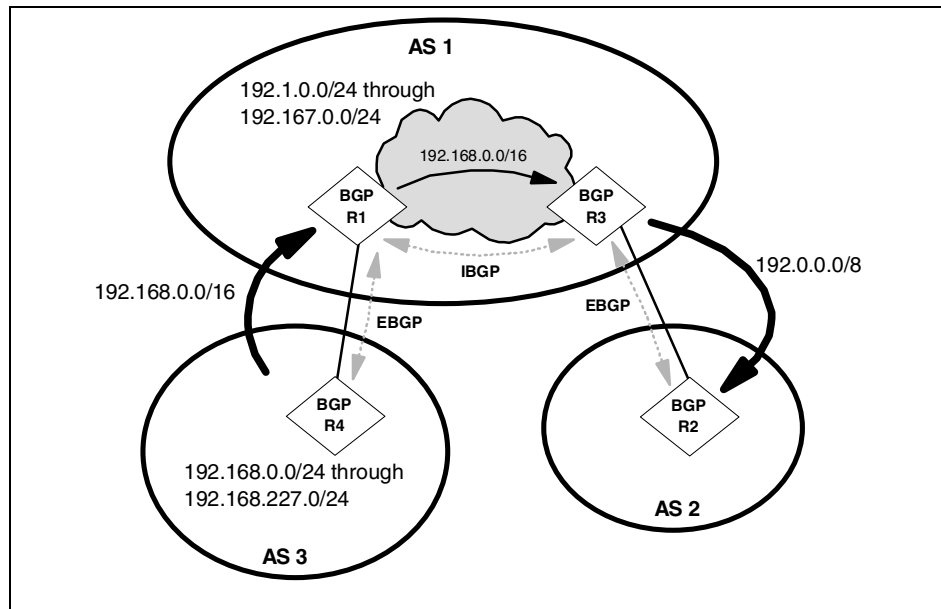


Figure 87. BGP route aggregation

This diagram depicts three autonomous systems interconnected via BGP. In this example, networks 192.168.0.0 through 192.168.227.0 are located within AS 3. To reduce the size of routing announcements, R4 aggregates these individual networks into a single route entry prior to advertising into AS 1. The single entry 192.168.0.0/16 represents a valid CIDR supernet even though it is an illegal class C network.

BGP aggregate routes contain additional information within the AS_PATH path attribute. When aggregate entries are generated from a set of more specific routes, the AS_PATH attributes of the more specific routes are combined. For example in Figure 87, the aggregate route 192.0.0.0/8 is announced from AS 1 into AS 2. This aggregate represents the set of more

specific routes deployed within AS 1 and AS 3. When this aggregate route is sent to AS 2, the AS_PATH attribute consists of <1 3>. This is done to prevent routing information loops. A loop could occur if AS 1 generated an aggregate with an AS_PATH attribute of <1>. If AS 2 had a direct connection to AS 3, the route with the less-specific AS_PATH advertised from AS 1 could generate a loop. This is because AS 2 does not know this aggregate contains networks located within AS 3.

4.9.7 BGP confederations

BGP requires that all speakers within a single AS have a fully meshed set of IBGP connections. This can be a scaling problem in networks containing a large number of IBGP peers. The use of BGP confederations addresses this problem.

A BGP confederation creates a set of autonomous systems that represent a single AS to peers external to the confederation. This removes the full mesh requirement and reduces management complexity.

Figure 88 illustrates the operation of a BGP confederation. In this sample network, AS 1 contains 8 BGP speakers. A standard BGP network would require 28 IBGP sessions to fully mesh the speakers.

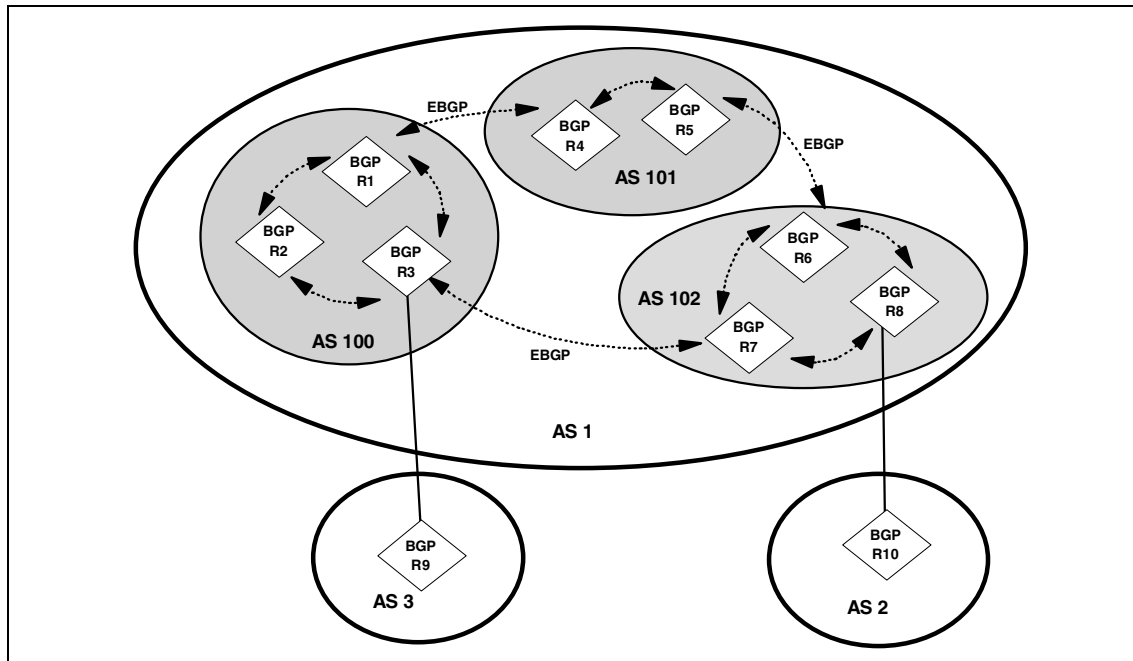


Figure 88. BGP confederations

A confederation divides the AS into a set of domains. In this example, AS 1 contains three domains. Devices within a domain have a fully meshed set of IBGP connections. Each domain also has an EBGP connection to other domains within the confederation. In the example network, R1, R2, and R3 have fully meshed IBGP sessions. R1 has an EBGP session within the confederation to R4. R3 has an EBGP session outside the confederation to R9.

Each router in the confederation is assigned a confederation ID. A member of the confederation uses this ID in all communications with devices outside the confederation. In this example, each router is assigned a confederation ID of AS 1. All communications from AS 1 to AS 2 or AS 3 appear to have originated from the confederation ID of AS 1.

Even though communication between domains within a confederation occurs with EBGP, the domains exchange routing updates as if they were connected via IBGP. Specifically, the information contained in the NEXT_HOP, MULTI_EXIT_DESC, and LOCAL_PREF attributes is preserved between domains. The confederation appears to be a single AS to other autonomous systems.

BGP confederations are described in RFC 3065. At the time of this writing, this is a proposed standard. Regardless, BGP confederations have been widely deployed throughout the Internet. Numerous vendors support this feature.

4.9.8 BGP route reflectors

Route reflectors are another solution to address the requirement for a full mesh of IBGP sessions between peers in an AS. As noted previously, when a BGP speaker receives an update from an IBGP peer, the receiving speaker propagates the update only to EBGP peers. The receiving speaker does not forward the update to other IBGP peers. Route reflectors relax this restriction. BGP speakers are permitted to advertise IBGP learned routes to certain IBGP peers.

Figure 89 depicts an environment utilizing route reflectors. R1 is configured as a route reflector for R2 and R3. R2 and R3 are route reflector clients of R1. No IBGP session is defined between R2 and R3. When R3 receives an EBGP update from AS 3, it is passed to R1 using IBGP. Since R1 is configured as a reflector, R1 forwards the IBGP update to R2.

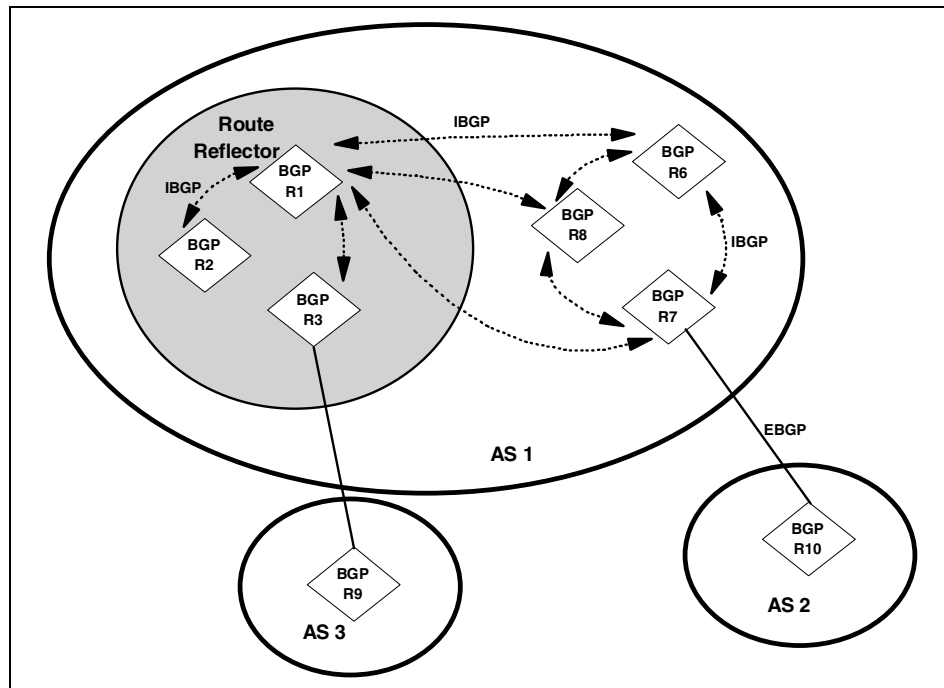


Figure 89. BGP route reflector

Figure 89 also illustrates the interaction between route reflectors and conventional BGP speakers within an AS. In this figure, R1, R2, and R3 are in the route reflector domain. R6, R7, and R8 are conventional BGP speakers containing a full mesh of IBGP peer connections. In addition, each of these speakers is peered with the route reflector. This configuration permits full IBGP communication within AS 1.

Although not shown in Figure 89, an AS can contain more than one route reflector. When this occurs, each reflector treats other reflectors as a conventional IBGP peer.

Route reflectors are described in RFC 2796. At the time of this writing, this is a proposed standard.

4.10 Routing protocol selection

The choice of a routing protocol is a major decision for the network administrator. It has a major impact to overall network performance. The selection depends on network complexity, size, and administrative policies. The protocol chosen for one type of network may not be appropriate for other types of networks. Each unique environment must be evaluated against a number of fundamental design requirements:

- Scalability to large environments: The potential growth of the network dictates the importance of this requirement. If support is needed for large, highly-redundant networks, link state or hybrid algorithms should be considered. Distance vector algorithms do not scale into these environments.
- Stability during outages: Distance vector algorithms may introduce network instability during outage periods. The counting to infinity problems (4.3.5, “Convergence and counting to infinity” on page 148) may cause routing loops or other non-optimal routing paths. Link state or hybrid algorithms reduce the potential for these problems.
- Speed of convergence: Triggered updates provide the ability to immediately initiate convergence when a failure is detected. All three types of protocols support this feature. One contributing factor to convergence is the time required to detect a failure. In OSPF and EIGRP networks, a series of hello packets must be missed before convergence begins. In RIP environments, subsequent route advertisements must be missed before convergence is initiated. These detection times increase the time required to restore communication.

- **Metrics:** Metrics provide the ability to groom appropriate routing paths through the network. Link state algorithms consider bandwidth when calculating routes. EIGRP improves this to include network delay in the route calculation.
- **Support for VLSM:** The availability of IP address ranges dictates the importance of this requirement. In environments with a constrained supply of addresses, the network administrator must develop an addressing scheme that intelligently overlays the network. VLSM is a major component of this plan. The use of private addresses ranges may also address this concern.
- **Vendor interoperability:** The types of devices deployed in a network indicate the importance of this requirement. If the network contains equipment from a number of vendors, standard routing protocols should be used. The IETF has dictated the operating policies for the distance vector and link state algorithms described in this document. Implementing these algorithms avoids any interoperability problems encountered with non-standard protocols.
- **Ease of implementation:** Distance vector protocols are the simplest routing protocol to configure and maintain. Because of this, these protocols have the largest implementation base. Limited training is required to perform problem resolution in these environments.

In small, non-changing environments, static routes are also simple to implement. These definitions change only when sites are added or removed from the network.

The administrator must assess the importance of each of these requirements when determining the appropriate routing protocol for an environment.

4.11 Additional functions performed by the router

The main functions performed by a router relate to managing the IP routing table and forwarding data. However, the router should be able to provide information alerting other devices to potential network problems. This information is provided via the ICMP protocol described in 3.2, “Internet Control Message Protocol (ICMP)” on page 102. The information includes:

- **ICMP Destination Unreachable:** The destination address specified in the IP packet references an unknown IP network.
- **ICMP Redirect:** Redirect forwarding of traffic to a more suitable router along the path to the destination.

- ICMP Source Quench: Congestion problems (for example, too many incoming datagrams for the available buffer space) have been encountered in a device along the path to the destination.
- ICMP Time Exceeded: The Time-to-Live field of an IP datagram has reached zero. The packet is not able to be delivered to the final destination.

In addition, each IP router should support the following base ICMP operations and messages:

- Parameter problem: This message is returned to the packet's source if a problem with the IP header is found. The message indicates the type and location of the problem. The router discards the errored packet.
- Address mask request/reply: A router must implement support for receiving ICMP Address Mask Request messages and responding with ICMP Address Mask Reply messages.
- Timestamp: The router must return a Timestamp Reply to every Timestamp message that is received. It should be designed for minimum variability in delay. To synchronize the clock on the router, the UDP Time Server Protocol or the Network Time Protocol (NTP) may be used.
- Echo request/reply: A router must implement an ICMP Echo server function that receives requests sent to the router, and sends corresponding replies. The router may choose to ignore ICMP echo requests addressed to IP broadcast or IP multicast addresses.

4.12 Routing processes in UNIX-based systems

This chapter has focused on protocols available in standard IP routers. However, several of these protocols are also available in UNIX-based systems. These protocols are often implemented using one of two processes:

- routed (pronounced route-D): This is a basic routing process for interior routing. It is supplied with the majority of TCP/IP implementations. It implements the RIP protocol.
- gated (pronounced gate-D): This is a more sophisticated process allowing for both interior and exterior routing. It can implement a number of protocols including OSPF, RIP-2, and BGP-4.

