11

IP Routing—Configuring Static Routes

Contents

Overview

Unlike a simple switch, a router can route a packet from one network to another. When the ProCurve Secure Router receives a packet, it matches the packet's destination address to a route in its routing table. This route specifies the interface through which the router must forward the packet in order for the packet to reach its destination.

This chapter describes the ProCurve Secure Router's routing table and explains how to add static routes to this table. In this chapter you will also learn how to configure a default route. In a small network with a single WAN connection, static and default routes provide the simplest and most reliable configuration for IP routing.

The ProCurve Secure Router also supports several routing protocols that allow the router to discover routing information from other routers. You should implement at least one of these protocols when your network has a large or complicated topology. *Chapter 15: IP Routing—Configuring RIP, OSPF, BGP, and PBR* in the *Advanced Management and Configuration Guide* describes how to configure these protocols.

Before configuring routing, you should understand the basics of IP addressing and networks. You should also understand how a router uses its routing table to forward traffic.

IP Addressing

Devices route packets by looking at their Layer 3 headers, typically their IP headers. (Currently, the ProCurve Secure Router only routes IP traffic.)

A packet's IP header contains a field for its source address and a field for its destination address. The router reads the destination IP address to determine where it should forward the packet.

An IP address is a field that uniquely identifies a host or device in the Internet or other network. In IP version 4 (IPv4) this field is 32 bits. A 32-bit IP address divides into four 8-bit octets. Typically, you will see IP addresses written in digital form. Therefore, IP address 11000000.10101000.000101101.01100011 is usually written as 192.168.45.99.

Unlike MAC addresses, IP addresses are not permanent or hardware specific. A host can change its address, and it can receive a temporary address from a server. However, public IP addresses must be unique and globally significant. (Otherwise, hosts could never be certain that data would arrive at the destination they intended.) Certain IP addresses are reserved for private networks; these addresses are locally significant and can be used by any number of different private organizations.

Networks

A network is a group of hosts that share a network address. Traffic between these hosts can be forwarded by bridges or switches. However, when a packet must be sent into a new network—that is, when its source and destination have different network addresses—the packet must be routed.

Network Addresses and Subnet Masks

A network address is the first part of a host's IP address. The second part of the IP address uniquely identifies the host within that network.

A subnet mask defines which bits identify the network and which identify the individual host. The subnet mask consists of 32 bits—first, a string of continuous ones; then, a string of continuous zeros.

All bits in the IP address that correspond with a one in the subnet mask are network bits; all bits that correspond with a zero are the host bits. (See Figure 11-1.)

Networks can be of varying sizes, depending on how many bits are allocated for the network address and how many for the host address. The greater the number of network bits, the fewer the addresses the network contains. (Because most of the bits define the network, there are fewer bits in which to store different addresses on that network.)

The first address (all zero host bits) in every network is reserved for identifying the network, and the last address (all one host bits) for broadcasting.

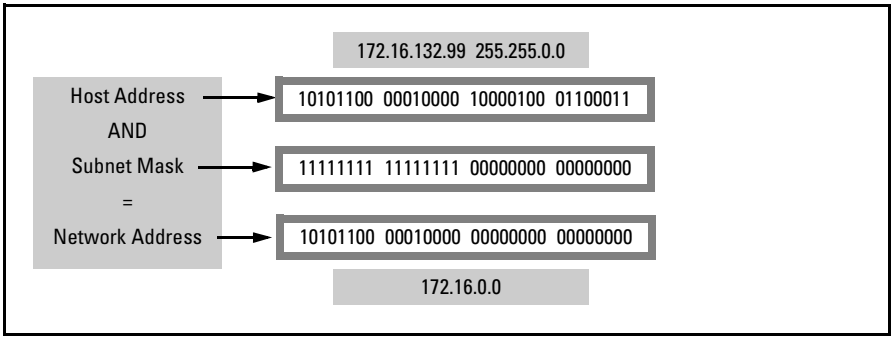

Figure 11-1. Subnet Masks

Classful Networks

In the early days of IP addressing, routing protocols did not always use subnet masks. The address itself needed to identify which bits were network bits and which host bits. Classful networks met this need. The first four bits of a classful IP address identified how many octets belonged to the network address.

Classful network addresses always end evenly at the end of an octet:

- Class A networks have 8-bit network addresses. They are identified by a 0 in the first bit. Therefore, the 126 class A networks range from 1.0.0.0 to 126.0.0.0. (127.0.0.0 is reserved for loopback and 0.0.0.0 for default routes.) Each class A network can accommodate up to 16,777,214 hosts.
- Class B network addresses always start with 10 in the first two bits, which indicates that the network has a 16-bit network address. The 16,384 class B networks range from 128.0.0.0 to 191.255.0.0. Each network can accommodate up to 65,534 hosts.
- Class C networks have 24-bit network addresses and always start with 110 in the first three bits. The 2,097,252 class C networks range from 192.0.0.0 to 223.255.255.0. Each class C network can accommodate up to 254 hosts.
- Class D networks have 32-bit network addresses and always include 1110 in the first four bits. These networks are used for multicasting and range from 224.0.0.0 to 239.255.255.255.

You might notice that this schema leaves networks beginning with 1111 undefined. Such networks are called Class E networks and have not been assigned a specific function.

CIDR

Classful networks condense more information into fewer bits: a router can resolve an address into its network and host bits without a 32-bit subnet mask. However, classful networks do not use IP addresses efficiently. Class C networks only provide addresses for 254 hosts, while Class B networks provide addresses for 65,534.

Many organizations need more addresses than a Class C network provides, but fewer than a Class B network does. Using Class C networks, an organization must request another network every time it needs more addresses. However, if the organization requests a Class B network so that it will have sufficient addresses, it usually wastes the vast majority of these addresses.

Most IP routers today support Classless Inter-Domain Routing (CIDR), which allows network administrators to define networks of any size. CIDR typically uses a prefix length instead of a subnet mask; the number in the prefix is the number of network bits in the address. For example, a network address with the subnet mask 255.255.0.0 has a /16 prefix length.

Network administrators can subdivide classful networks into smaller, variable-length networks by changing the prefix length.

For example, your organization is using the Class B network 172.16.0.0. Your organization needs at least six subnets, each with at least 500 hosts. With future expansion, your organization will need ten subnets. You round this number up to the nearest power of two and decide to divide the network into sixteen subnets. You calculate that each of the sixteen subnets can hold 4,094 hosts, which more than meets your organization's requirements.

To subdivide the network, you add one bit to the prefix length for every time you divide the network in half. For example, half of a /16 network is a /17 network, a fourth of a /16 network is a /18 network, and so forth. Sixteen is 2^4, so in the scenario outlined above, you would divide the 16-bit network four times, into sixteen 20-bit subnets:

- 172.16.0.0 /20 (255.255.240.0)
- 172.16.16.0 /20
- 172.16.32.0 /20
- 172.16.48.0 /20
- ...
- 172.16.240.0 /20

When you use prefix lengths in this way, the bit length becomes, in a sense, part of the address. 172.16.0.0 /20 is a different network than 172.16.0.0 /16. The second is the network address for the entire class B network, while the first is a network that includes only hosts from 172.16.0.1 to 172.16.15.254.

Therefore, when you define routes to variable-length subnets, you must always be careful to specify the correct bit length. If a router thinks that it knows a route to network 172.16.0.0 /16 when the route should actually be to 172.16.0.0 /20, it may misroute traffic to the other fifteen 20-bit networks in the 172.16.0.0/16 range.

Routing Table

A routing table stores the following information for each network that the router knows how to reach:

- destination network address
- subnet mask
- next-hop address
- forwarding interface
- metric
- administrative distance

Destination Network Address and Subnet Mask

The destination network address and subnet mask identify the route. When a router receives a packet, it matches the packet's destination IP address to a network address in the routing table. The subnet mask defines how many bits the router examines when matching the two addresses. For example, a routing table entry for 172.16.0.0 with a subnet mask 255.255.0.0 refers to all packets destined to IP addresses of which the first 16 bits are 172.16.

If a packet matches more than one entry, the router uses the more-specific route (the route with a longer subnet mask), which it assumes is more accurate for that packet.

The subnet mask condenses the routing table: an individual router's table need not include a separate entry for each host or subnet in the 172.16.0.0/16 network when the next hop to all these destinations is the same. Routers nearer a particular destination may include more specific entries that allow them to forward traffic to individual networks that have been subdivided from a larger network.

Next-Hop Address and Forwarding Interface

A route's next-hop address and forwarding interface instruct the router how to forward packets that match the destination address for the route.

The next-hop address is the address of the next directly-connected device en route to the destination address. The router determines the forwarding interface for the route by looking up, in its routing table, the interface that connects to the next-hop address. (Because the next-hop address should be a directly connected device, the routing table will automatically include this information.)

Only a forwarding interface is absolutely necessary for a route. When you add a static route to the routing table, you can specify a forwarding interface instead of a next-hop address. The next-hop address is then listed as 0.0.0.0.

Administrative Distance and Metric

A router may learn more than one route to the same destination. The router compares the administrative distances and metrics of identical routes to select the single best route that it will add to its routing table. (You can also enable the router to select more than one best route. See "Load Sharing" on page 11-11.)

The ProCurve Secure Router uses administrative distance to compare routes learned by different routing protocols or methods. The ProCurve Secure Router uses metrics to compare routes learned by the same routing protocol. That is, each routing protocol used on a router has its own database of routes. When a routing protocol knows more than one route to a destination, it selects the route with the lowest metric as its best route. The router then compares the best routes of each method and selects the route with the lowest administrative distance.

A route's administrative distance indicates how reliable the router considers the method through which it discovered the route. The lower the administrative distance the more trustworthy the route.

If you are only using static routes, you generally do not need to worry about administrative distance. However, if you are using static routing in conjunction with a routing protocol, you should understand how the ProCurve Secure Router uses administrative distance to choose between identical routes learned using different methods. The ProCurve Secure Router always selects the route with the lower administrative distance. For example, statically configured routes have a default administrative distance of 1, while Routing Information Protocol (RIP) routes have a default administrative distance of 120. When the router knows an identical RIP and static route, it only adds the static route to the routing table.

A route's metric is the cost of sending traffic on that route and can be based on various criteria:

- number of hops to the destination
- link conditions:
  - bandwidth
  - delay
  - reliability
- organization policies
  - monetary cost
  - autonomous systems through which the packet must travel

Number of hops and bandwidth are among the most common criteria for computing a route's metric.

Each routing protocol has its own method for computing a route's metric. The protocol compares the metric of identical routes to determine the best route. The protocol chooses the route with the smallest metric.

Other Information Stored in a Route

Routing tables can also include information such as:

- route type—whether the destination subnet is directly attached or remote
- source of the route—directly connected, statically configured, or discovered with a routing protocol
- route age
- maximum transmission unit (MTU) over the link used in the route

The ProCurve Secure Router tracks all of these parameters. When you view your router's routing table, you can see the route type and source of the route.

A routing table should, most importantly, provide reliable routes that get traffic to its destination. Ideally, routes should also minimize congestion and delay. One of your must important tasks when configuring your ProCurve Secure Router is to construct a routing table with reliable best routes.

Static Routing

The most straightforward method for constructing a routing table is static routing. Static routes are routes that you manually add to the routing table. When you enter a static route, you specify the destination network address and subnet mask and either the next-hop address or forwarding interface for that destination.

Dynamic Routing Protocols

Routers can also construct their routing tables using dynamic routing protocols. The ProCurve Secure Router supports three routing protocols, each of which it can use alone or in conjunction with the others:

- RIP versions 1 and 2
- Open Shortest Path First (OSPF) version 2
- Border Gateway Protocol (BGP) version 4

See *Chapter 15: IP Routing—Configuring RIP, OSPF, BGP, and PBR* in the *ProCurve Secure Router Advanced Management and Configuration Guide* to learn how to configure these protocols.

Static Routing Versus Dynamic Routing

Static routing is secure because it provides you the tightest control over traffic flow: you determine exactly which connection the router uses to forward traffic to each destination. Static routing is also relatively reliable (although it does open room for human error).

On a router in a small network with a single exit to a remote site or the Internet, static routing is effective and simple to configure.

However, as a network expands, configuring all the necessary static routes can become more and more complicated and time-consuming. Ensuring that all routes remain accurate can also unduly burden an IT staff. Every time you want to add a connection or change a route, you must configure the change on every router in the network. Routers do not automatically respond to a failed connection, so traffic can be misrouted.

Note

Network monitoring can provide a mechanism for detecting failed static routes. See the *Advanced Management and Configuration Guide*, *Chapter 9: Network Monitoring*.

Dynamic routing can provide reliable routes. OSPF selects routes according to fairly sophisticated criteria, such as link state and bandwidth, and BGP, though complicated to configure, can take an organization's policies into account when selecting routes. What is the best route at one moment may not always be the best route, and dynamic routing protocols can track these changes. Dynamic routing also adapts well to changes in network topology, such as node failures and network expansion.

On the other hand, routing protocols consume bandwidth and CPU processes; routers must exchange updates and calculate the best routes. A router that has been carelessly configured may send updates to unauthorized devices, opening a security vulnerability. However, a well-designed network eliminates many of these problems.

You should not implement a dynamic routing protocol on a demand interface that is used with a dial-up connection because the routing updates may keep the line up longer than is necessary, costing your organization money. Instead, configure a static route that uses the demand interface as the forwarding interface. If you are using the dial-up connection for backup, you can configure a floating static route. (See "Configuring a Floating Static Route" on page 11-16.)

You can use static routing in conjunction with one or more dynamic routing protocols. A static route will always supersede a discovered route because static routes have low administrative distance. Table 11-1 shows the default administrative distance for the various types of routes that the ProCurve Secure Router can learn. As you can see, besides routes to directly connected networks, static routes are considered to be the most reliable.

Table 11-1. Hierarchy of Routes (Most Trusted to Least Trusted)

Route Type	Default Administrative Distance
directly connected	0
static	1
BGP	• 20 for external routes • 200 for internal and local routes
OSPF	110
RIP v1 and v2	120

Load Sharing

Typically, a routing table can only include one best route for each destination. If you enter more than one route to the same destination, the router will only add this route to its routing table if the first route that you entered is removed or if the forwarding interface for this route goes down. However, the ProCurve Secure Router can also implement load sharing, which enables it to activate up to routes to the same destination. This option enables the router to use redundant connections to the same remote site.

When you enable load sharing, the router can place up to six routes to the same destination in its active routing table. The routes must all have the same metric and administrative distance; otherwise, only the route with the lowest values will be selected.

The router can share traffic over the routes based on destination, assigning traffic destined to some hosts to one route and traffic destined to other hosts to another route. In this case, the traffic may not be exactly balanced over the multiple connections, but the more sessions the router supports, the more evenly balanced the traffic will be.

The router can also share the traffic in a round-robin manner, alternating between the routes every time it routes a new packet to the destination network. Configuring the router to load share in this way, however, can cause packets to arrive at the destination out of order and is not generally recommended.

Fast Caching

One of a router's tasks is to forward the packets it receives with a minimum of delay. However, the router must also accurately route packets, and looking up routes takes time and processing power. When a router uses process switching, it considers route lookup to be no more important than any other process and forces packets to wait in a queue until it finishes other tasks. When CPU usage spikes, packets can be delayed longer than acceptable.

Fast caching, or fast-switching cache, is designed to speed processing of packets that follow often-used routes. In addition to the routing table, the router keeps a fast-cache table, which contains entries for recently received packets. A fast-cache entry includes the destination address and the forwarding interface. When the router receives a packet, the CPU postpones other tasks to immediately check the fast-cache table for a matching entry. If the router finds a matching entry, it rewrites the packet's header and forwards it to the appropriate interface. (See Figure 11-2.) If the router does not find a match in the fast-cache table, it sends the packet to the appropriate queue to await processing. When the router processes these packets, it checks the routing table to determine where the packets should be forwarded.

On the ProCurve Secure Router, you can enable fast caching for individual interfaces. However, if you enable the firewall, the ProCurve Secure Router uses process switching because firewall features can require extensive computations. For example, the firewall must check packets for known cyber attacks, ensure packet integrity, track connections, and determine if packets match access control lists (ACLs).

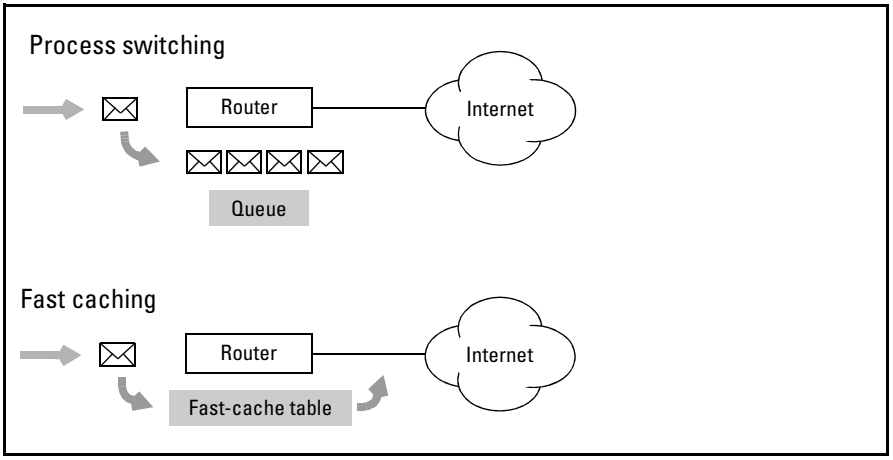

Figure 11-2. Fast Caching Versus Process Switching

Configuring Static Routes

Overview

A static route is a route that you add manually to a routing table. You can construct a router's entire table manually. (The table will also automatically include directly connected networks with a metric and an administrative distance of zero.)

When you use static routing in exclusion of other routing protocols, the router will not share its routing table with other routers. This means that the hosts serviced by this router will only be able to reach a destination if you add an entry for that destination. In large and complicated networks, configuring static routing can be prohibitively time-consuming and cumbersome. However, in a relatively uncomplicated environment with few subnets, you can quickly configure the necessary routes while maintaining tight control over your network.

Static routing is best suited for networks that have:

- a simple topology and a single router at each site
- a single destination for traffic—for example, to an Internet service provider (ISP)
- only one path for IP traffic

You can use static routing with dynamic routing. In this case, you supplement routes discovered through various protocols with manually added routes. You can configure the router to advertise these routes using a routing protocol, or you can keep the routes private. (See *Chapter 15: IP Routing—Configuring RIP, OSPF, BGP, and PBR* in the *Advanced Management and Configuration Guide* to learn how to configure a routing protocol.)

For example, you can run a routing protocol, but configure a static default route. (See "Configuring a Default Route" on page 11-17.)

Configuring a Static Route

When you configure a static route, you must enter the following information:

- destination address and subnet mask
- next-hop address or forwarding interface

By default, the administrative distance for a static route is 1 and the metric 0. You can view the kind of information the ProCurve Secure Router stores in its routing table in Figure 11-3.

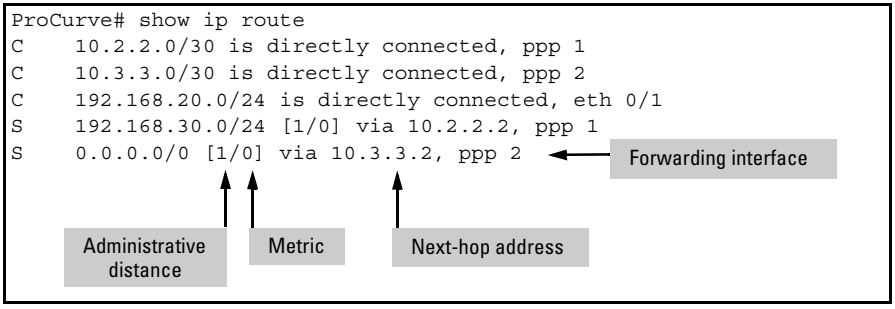

Figure 11-3. Routing Table with Static Routes

The destination address is the network address for the destination subnet. The subnet mask indicates how long this network address is. (The ProCurve Secure Router also allows you to enter a prefix length instead of a subnet mask.) When the router looks for a route that matches a packet's destination, it only compares the bits specified by the subnet mask.

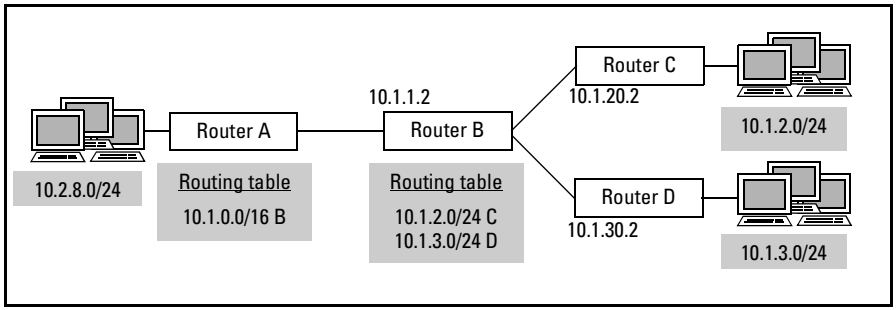

Figure 11-4. Prefix Lengths with Static Routing

You add routes to the routing table from the global configuration mode context. Enter this command:

Syntax: ip route <*destination network A.B.C.D*> <*subnet mask* | /*prefix length*> <*next hop A.B.C.D* | *forwarding interface ID*> [track <*name*>] [<*administrative distance*>]

You should make the network address and subnet as short as possible for the next-hop address to still be valid for all matching packets. For example, to configure a route to network 10.1.3.0 /24 on Router A shown in Figure 11-4, you could enter a route to the entire 10.1.0.0 /16 network:

```
ProCurve(config)# ip route 10.1.0.0 255.255.0.0 10.1.1.2
```

You would have to configure a more specific route to network 10.1.3.0 /24 on Router B:

```
ProCurve(config)# ip route 10.1.3.0 255.255.255.0 10.1.30.2
```

For point-to-point connections, instead of the next-hop IP address, you can specify the forwarding interface (for example, PPP 1 or Frame Relay 1.103). It is often a good idea to specify the forwarding interface rather than the next-hop address, particularly when connecting to an external network, because IP addresses can change without notice. The route in the routing table includes the specified forwarding interface, but forwards traffic to any next-hop neighbor that connects to the interface. See Figure 11-5.

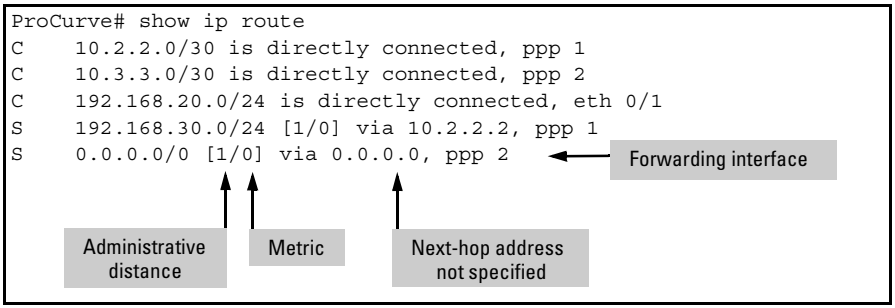

Figure 11-5. Static Route with a Forwarding Interface

Specifying administrative distance is optional. By default, static routes have an administrative distance of 1 and are considered to be more reliable than any other routes (except those to directly connected networks).

Associating the route with a track is also optional. A track monitors the route using network monitor probes, removing the route should it fail at any point. For more information on tracks and network monitoring, see the *Advanced Configuration and Management Guide*, *Chapter 9: Network Monitoring*.

Configuring a Floating Static Route

When the router has a redundant connection to a network, it needs two routes to that network, one of which uses the primary interface as the forwarding interface and one of which uses the redundant interface. However, the routing table can only include a single active route to a particular network. (See "Configuring Load Sharing" on page 11-20 for an exception to this rule.)

You can configure a floating static route that uses the redundant, or backup interface, and that will only appear if the forwarding interface for the primary route goes down. You configure the floating static route by assigning it a higher administrative distance than that for the primary route.

For example, your router can reach remote site 192.168.115.0 /24 through the PPP 1 interface. If this connection goes down, it can reach the remote site through the backup PPP 2 interface. Configure the routes as follows:

```
ProCurve(config)# ip route 192.168.115.0 /24 ppp 1
ProCurve(config)# ip route 192.168.115.0 /24 ppp 2 2
```

You can also configure a floating static route that only appears when a route discovered using a routing protocol becomes invalid and is removed from the routing table. Simply, specify an administrative distance in the floating static route that is higher than that for the protocol.

For example, your router has learned a route to network 192.168.115.0 /24 by running OSPF on the PPP 1 interface. The router uses an ISDN module for backup. Configure a floating static route through the demand interface that will only appear if the PPP 1 interface fails:

```
ProCurve(config)# ip route 192.168.115.0 /24 demand 1 120
```

Because OSPF routes have an administrative distance of 110, specify **120** for the floating static route's administrative distance. (Refer to Table 11-1 on page 11-11 for the administrative distance of various routing protocols.)

Configuring a Default Route

A default route is a special static route that applies to all traffic. Typically, when the router receives a packet that it does not know how to forward, it drops it. A default route allows the router to forward all such packets toward the destination most likely to be able to route them.

To configure a default route, enter a route to a destination address of all zeros with an all-zero subnet mask. The all-zero subnet mask indicates to the router that a packet's IP address does not have to match any of the destination address bits in order for the route to be valid. Because the router always matches traffic to the most specific route, it will only use the default route for traffic that would otherwise be dropped.

To configure the default route, move to the global configuration mode context and enter this command:

Syntax: ip route 0.0.0.0 [0.0.0.0 | /0] <*next hop A.B.C.D | forwarding interface ID*> [track <*name*>] [<*administrative distance*>]

The ProCurve Secure Router allows you to enter the default route in CIDR notation.

Instead of configuring a route to a default next-hop address, you can configure a default forwarding interface. A default route is often used to forward external traffic. In this case, specifying the WAN interface as the default forwarding interface can be a good idea so that the default remains valid no matter what IP address the remote router has.

For example, your router connects to the Internet with a PPP connection. You could configure the following default route for all external traffic:

```
ProCurve(config)# ip route 0.0.0.0 0.0.0.0 ppp 1
```

Default routes can be especially useful for routers with a single point-to-point WAN connection. If necessary, add static routes for any local subnets that are not directly connected to the Ethernet ports. (Directly connected networks are automatically added.) Then add a default route for all other traffic through the WAN interface.

For example, to configure Router A shown in Figure 11-6, you would enter:

```
ProCurve(config)# ip route 192.168.10.0 /24 192.168.12.2
ProCurve(config)# ip route 0.0.0.0 /0 ppp 1
```

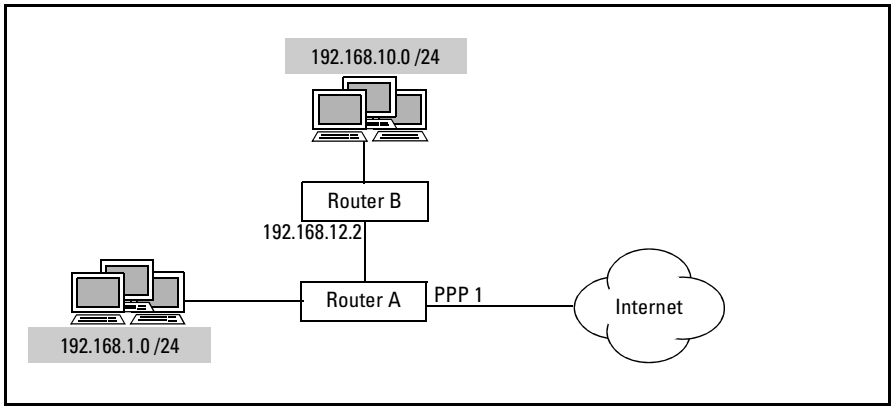

Figure 11-6. Default Routing

Default routes are used with dynamic routing as well as static routing. For example, OSPF stub routers in an OSPF network do not receive many of the OSPF link state advertisements (LSAs). This keeps the protocol's overhead down and stub router memory uncluttered with routes that are not needed. Instead, stub routers can receive a default route for all external traffic.

Configuring a Route through the Null Interface

When the router matches a packet to a route through the null interface, it drops the packet. You can use the null interface to force the router to drop certain traffic.

To configure a null route, enter this command from the global configuration mode context:

Syntax: ip route <*A.B.C.D*> <*subnet mask* | /*prefix length*> null 0 [<*administrative distance*>]

You might configure a route through the null interface in order to drop traffic to network addresses that do not yet exist in your network.

For example, an organization has allocated the address space 192.168.20.0 /24 to a remote site. However, currently the site is only using half of the addresses. Network management have divided the network into two /25 subnets and left the second subnet (192.168.20.128 /25) unused. You can prevent the local router from forwarding traffic across the WAN link that will only dropped by the remote router. Enter this command:

```
ProCurve(config)# ip route 192.168.20.128 /25 null 0
```

You could also use a null route in order to force the router to:

- drop traffic to destinations that you have determined to be unauthorized

  However, a better way to control traffic is to use an ACL or an ACP. (See *Chapter 5: Applying Access Control to Router Interfaces* of the *Advanced Management and Configuration Guide*.)

- advertise a route not included in its routing table

  When a router uses a routing protocol, its routing table must include a route in order to advertise that route. You could configure a null route if you wanted the router to advertise a route, but not to forward traffic using that route. (For more information on this topic, see “Advertising Local Networks” on page 15-73 in *Chapter 15: IP Routing—Configuring RIP, OSPF, BGP, and PBR* of the *Advanced Management and Configuration Guide*.)

Configuring Load Sharing

Your ProCurve Secure Router may have more than one connection to the same remote site or to the Internet. However, a router can typically select a single best route for a destination; without further configuration, traffic destined to the site will travel over only one of the connections.

For example, your router provides a connection to one ISP through its PPP 1 interface. For redundancy, you connect the router to a second ISP through the PPP 2 interface. You configure a default route through PPP 1. All Internet traffic is carried over this WAN connection, and the redundant connection is unused unless the first connection fails—not a cost-effective solution.

Load sharing allows the router to place up to six routes to the same destination in its routing table. (See Figure 11-7.) The routes must have the same metric and administrative distance. When load-sharing is implemented, the router will sends some traffic over one route and some traffic over the other route.

To enable load sharing, enter this command from the global configuration mode context:

Syntax: ip load-sharing [per-destination | per-packet]

You can configure the router to balance traffic:

- per destination
- per packet

When the router balances traffic per destination, it assigns packets to routes based on the packets' source and destination addresses. That is, when the router must forward a packet to a destination for which multiple routes exit, it hashes the packet's source and destination and, according to this value, assigns the packet to a route. (The router performs the hash function such that a source and destination can only resolve to as many different values as routes are available in the routing table.) Therefore, per-destination load sharing does not balance traffic exactly equally; two successive packets may be sent over the same route, even if they have different source and destination addresses. Packets in the same session always take the same route because they have the same source and destination address. The more traffic that the router supports, the more evenly it will balance the traffic.

When the router balances traffic per packet, it sends each new packet over each route in turn. Although this option balances traffic more exactly, it is not generally recommended. Because each successive packet takes a different route, packets may arrive at the destination out of order.

```
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP
       IA - OSPF inter area, N1 - OSPF NSSA external type 1
       N2 - OSPF NSSA external type 2, E1 - OSPF external type 1
       E2 - OSPF external type 2

Gateway of last resort is 0.0.0.0 to network 0.0.0.0

S    0.0.0.0/0 [1/0] via 0.0.0.0, ppp 1
               [1/0] via 0.0.0.0, ppp 2
               [1/0] via 0.0.0.0, ppp 3
C    10.1.1.0/30 is directly connected, ppp 1
C    10.1.1.1/32 is directly connected, ppp 1
C    10.1.1.4/30 is directly connected, ppp 2
C    10.1.1.5/32 is directly connected, ppp 2
C    10.1.1.8/30 is directly connected, ppp 3
C    10.1.1.9/32 is directly connected, ppp 3
C    192.168.50.0/24 is directly connected, eth 0/1
C    192.168.51.0/24 is directly connected, eth 0/2
```

Multiple static routes

Figure 11-7. Routing Table with Load Sharing

After enabling load sharing, add the multiple static routes. For example, enter:

```
ProCurve(config)# ip route 0.0.0.0 /0 ppp 1
ProCurve(config)# ip route 0.0.0.0 /0 ppp 2
ProCurve(config)# ip route 0.0.0.0 /0 ppp 3
```

The routing table can hold up to six routes for the same destination. If you enter more than six routes, then the router will learn the extra routes, but not add them to the routing table. If you delete one of the routes in the routing table, or if the forwarding interface for one of these routes fails, then one of the extra routes will take its place.

Enabling Fast Caching

The ProCurve Secure Router can route incoming packets using either:

- process switching
- fast caching

A router using process switching:

- places packets in a queue to await processing
- looks up routes in the routing table, which contains all routes

A router using fast caching:

- interrupts other processes to serve packets immediately
- looks up routes in the fast-cache table, which contains only recently-used routes

Fast caching is a valuable tool for speeding packets through the router and maintaining quality of service (QoS).

By default, fast caching is enabled on:

- Ethernet interfaces
- Point-to-Point Protocol (PPP) interfaces
- Frame Relay subinterfaces

Although fast caching is not enabled on Asynchronous Transfer Mode (ATM) subinterfaces by default, ATM subinterfaces also support it.

You can disable fast caching on specific interfaces. If you disable fast caching, the ProCurve Secure Router will use process switching. With process switching, the router places all packets in the appropriate queue, where they wait until the router can process them.

You enable and disable fast caching for individual interfaces. One interface can use fast caching and another interface can use process switching.

To enable or disable fast caching on an interface, you must first move to the configuration mode context for that interface. Then enter this command:

Syntax: [no] ip route-cache

For example:

```
ProCurve(config)# int eth 0/1
ProCurve(config-eth 0/1)# no ip route-cache
```

Note

Fast caching is forcibly disabled when you use the following processes:

- the ProCurve Secure Router OS firewall
- any firewall processes, such as ACLs and ACPs
- policy based routing (PBR)

If you enable the firewall, the ProCurve Secure Router must use process switching because firewall features require the router to make more-extensive computations than simple route determination, including checks for attacks and packet filtering according to an access policy. Similarly, PBR requires the router to screen packets to determine whether to route them according to a route map or according to the routing table.

To optimize packet switching for firewall processes, the ProCurve Secure Router uses a separate table so that it does not have to check long ACLs each time it receives a packet. This table speeds up firewall computations.

Troubleshooting Static Routing

When you receive reports that traffic is not reaching its destination, first attempt to ping the destination from the router to verify that a host or other network node is not the root of the problem. If the ping confirms that the router cannot reach the destination, next view the routing table.

Note

The **show** and **debug** commands described in the following sections are enable mode commands. You can also enter the commands from configuration mode contexts by adding the **do** option.

Monitoring the Routing Table

To view the routing table, enter this enable mode command:

Syntax: show ip route

The screen displays the destinations to which the router can route traffic. (See Figure 11-8.) For each destination, the routing table also records:

- the method the router used to discover the route
  - B—BGP
  - C—directly connected
  - O—OSPF
  - R—RIP
  - S—entered manually (static)
- the administrative distance—the trustworthiness of the route, used to choose between two identical routes discovered through different methods
- the metric—the cost for the route
- the next-hop address
- the forwarding interface

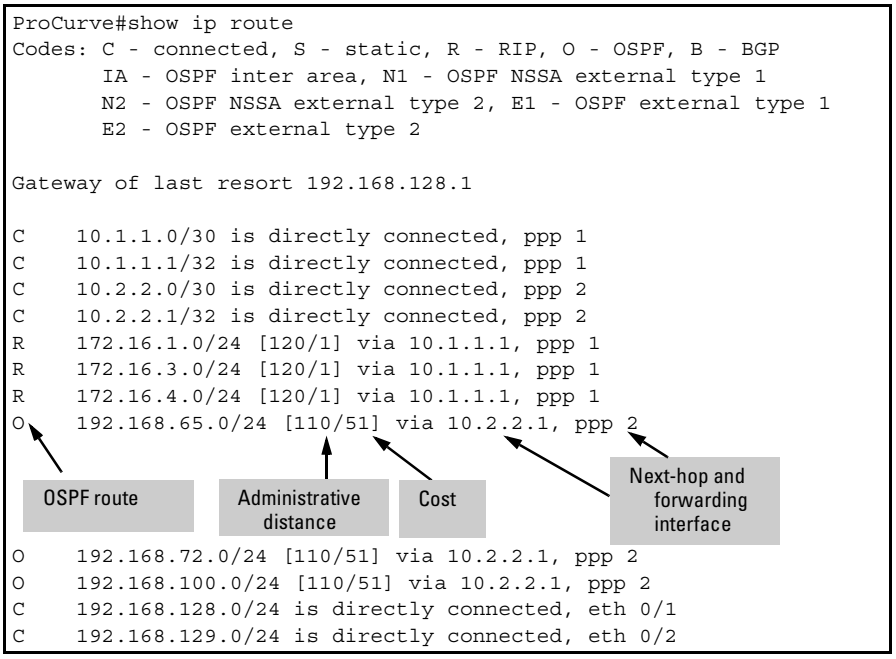

Figure 11-8. Routing Table

You can also view specific portions of the routing table. Use the commands in Table 11-2.

Table 11-2. Viewing the Routing Table

Table Section	Command Syntax
directly connected routes	**show ip route connected**
statically entered routes	**show ip route static**
BGP	**show ip route bgp**
RIP	**show ip route rip**
OSPF	**show ip route ospf**
routes displayed in table format	**show ip route table**
the number of routes stored in the routing table	**show ip route summary**

Using the Routing Table to Troubleshoot Static Routing

Several problems can prevent the router from using static routes to forward traffic to its destination correctly:

- You have not added a route to the destination.
- The router cannot use the route.
- The route to the destination is faulty.

Enter the **show ip route** command to determine what route, if any, the router is using to forward traffic to the destination in question.

When the routing table does not include a route for the destination, you should try adding the route. If adding new static routes on each new device become too cumbersome, you can configure a dynamic routing protocol. See *Chapter 15: IP Routing—Configuring RIP, OSPF, BGP, and PBR* in the *Advanced Management and Configuration Guide*.

Even if you have configured a static route for a destination, you may not see that route when you enter the **show ip route** command. The routing table only displays the routes that the router can use to forward packets. The router may know routes that it is not using because:

- the forwarding interface is down
- the router knows an identical route with a smaller metric or administrative distance
- the router knows an identical route with the same metric and administrative distance and load sharing is *not* enabled

If a static route will not appear in the routing table, verify that the associated forwarding interface is up. If necessary, troubleshoot that interface. If you have configured a next hop address for the static route, you should check the routing table to ensure that it includes a route to that next hop.

If you want the router to use more than one route to the same destination, you must enable load sharing with the **ip load-sharing** command.

If you see a route to the destination that hosts cannot reach, several problems could be causing traffic to be misrouted:

- Another router en route to the destination cannot route the traffic—In this case, you should use the **traceroute** command to pinpoint the router that is not forwarding the traffic. (See "Monitoring Routes" on page 11-27.) Remember that in order for a ping to be successful, routers must also know a route back to the source of the ping. You should always make sure that routes are two-way: the local router knows routes to remote destinations, and remote routers know routes to the local networks.
- The route in the local routing table is invalid—Check for miskeyed information such as the wrong interface number for the forwarding interface. You must remove the route before re-entering the route with the correct information. (When you configure more than one static route to the same destination, the router automatically assigns the second route a higher administrative distance. Therefore, if you fail to remove the faulty route, your correction will not take affect.)
- Your router's routing table includes the correct route, but it also includes a more-specific, incorrect route. For example, the router may have discovered a more-specific route using a routing protocol. See "Clearing Routes" on page 11-28 to learn how to remove dynamic routes from the table. See *Chapter 15: IP Routing—Configuring RIP, OSPF, BGP, and PBR* in the *Advanced Management and Configuration Guide* to learn how to troubleshoot routing protocols.

Monitoring Routes

You can monitor the route that packets actually take through the network by using the **traceroute** command. Enter the command followed by the destination address for the route you want to trace:

Syntax: traceroute <*A.B.C.D*>

The router sends out a series of pings with steadily incrementing TTLs, so that each successive ping reaches one hop closer to the destination. The router records the addresses of the routers that return the pings, thus building up a list of every hop between itself and the destination. (See Figure 11-9.)

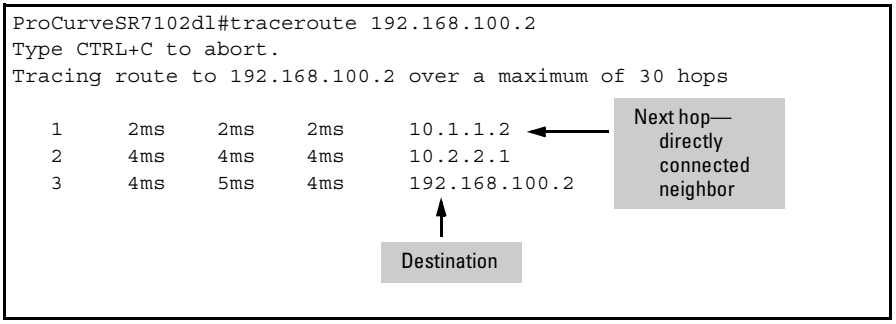

Figure 11-9. Traceroute Command

Tracing routes allows you to monitor actual traffic flow (although in a necessarily limited fashion). When traffic does not reach its destination, you can determine which network node cannot forward it. You can then troubleshoot the device with the problem.

When traffic can take more than one route through a network, you can use the **traceroute** command to discover which path routers have selected. If you determine that routers are using high-cost paths unnecessarily, you can make adjustments accordingly. For example, you can configure a routing protocol, such as OSPF, that takes link cost into account. Or you can configure PBR to allow the router to forward traffic over different paths depending on certain characteristics of the traffic. (See *Chapter 15: IP Routing—Configuring RIP, OSPF, BGP, and PBR* in the *Advanced Management and Configuration Guide*.)

Clearing Routes

In addition to the routes that you add to your router's routing table, your router may learn routes using a dynamic routing protocol. If your router has learned unreliable routes, you can clear them using this command:

Syntax: clear ip route [* | <*A.B.C.D*> <*subnet mask* | /*prefix length*>

You can enter *, which clears all routes, or the destination for the specific route you want to remove.

Note

Clearing a route is not necessarily enough to solve a problem. Unless you address the reason that the router learned the inaccurate route, the router may only learn the inaccurate route again.

If your router should not be receiving dynamic routes at all, then you should enter these commands:

```
ProCurve(config)# no router rip
ProCurve(config)# no router ospf
ProCurve(config)# no router bgp <AS>
```

If your *do* want your router to use a routing protocol in addition to static routes, you should troubleshoot the routing protocol as described in *Chapter 15: IP Routing—Configuring RIP, OSPF, BGP, and PBR* in the *Advanced Management and Configuration Guide*.

The **clear** command only removes learned routes. To clear a static route, you must enter the **no** form of the command you used to enter it:

Syntax: no ip route <*destination A.B.C.D*> <*subnet mask* | /*prefix length*> <*next hop A.B.C.D* | *forwarding interface ID*>

Remember that, unlike the **clear ip route** command, the **no ip route** command is entered from the global configuration mode context.

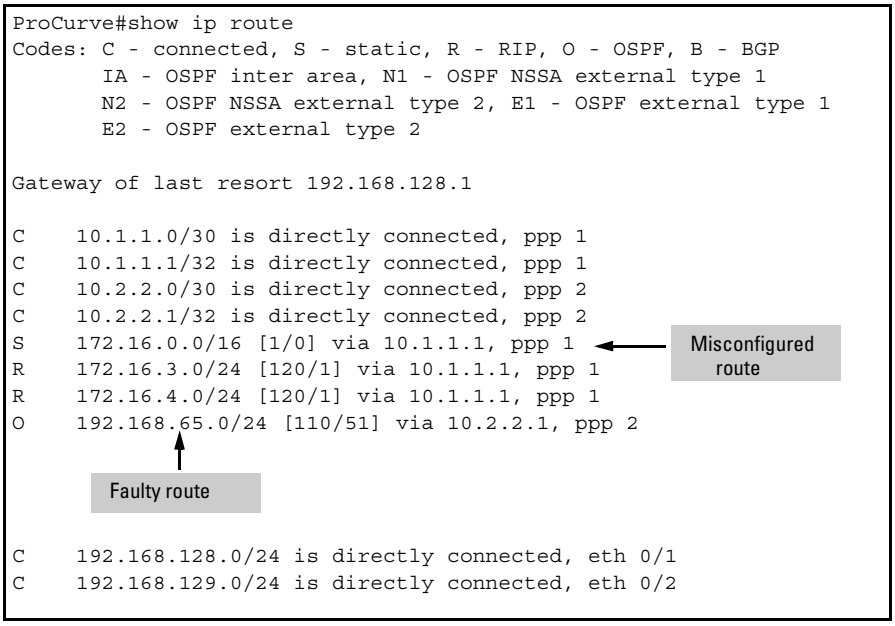

Figure 11-10. Clearing Routes

For example, your router has the routes in the routing table shown in Figure 11-10. The routes to 192.168.65.0/24 and 172.168.0.0/16 are faulty and you want to clear them. The first is a learned route, so you enter:

ProCurve# clear ip route 192.168.65.0 /24

The second is a static route, so you move to the global configuration mode context and enter:

ProCurve(config)# no ip route 172.168.0.0 /16 ppp 1

Quick Start

This section provides the commands you must enter to quickly configure static routes.

Only a minimal explanation is provided. If you need additional information about any of these options, check "Contents" on page 11-1 to locate the section that contains the explanation you need.

Static Routing

Static routing may be good solution for your WAN if:

- you are connecting remote sites that each only have one router
- the router only needs to route traffic to an ISP
- only one path is available to forward IP traffic

Connecting Simple Remote Sites

1. Configure a route to the remote network using the remote router's WAN IP address as the next-hop address:

   Syntax: ip route <*destination network A.B.C.D*> <*subnet mask* | */prefix length*> <*next hop A.B.C.D* | *forwarding interface ID*> [track <*name*>] [<*administrative distance*>]

   For example:

   ```
   ProCurve(config)# ip route 192.168.3.0 /24 10.2.2.1
   ```

   You can alternatively specify the connecting WAN interface on the local router as the forwarding interface:

   ```
   ProCurve(config)# ip route 192.168.3.0 /24 ppp 1
   ```

   For Frame Relay connections, use the Frame Relay subinterface for the PVC you want to use as the forwarding interface.

   It can be a good idea to use the logical interface as the reference for the route because IP addresses could change.

2. If necessary, add a route to another remote network.

Routing Traffic to an ISP

Configure a default route to the ISP router:

ProCurve(config)# ip route 0.0.0.0 /0 ppp 1

Syntax: ip route 0.0.0.0 /0 <*subnet mask* | /*prefix length*> <*next hop A.B.C.D* | *forwarding interface ID*> [track <*name*>] [<*administrative distance*>]

Again, you should specify the WAN interface as the forwarding interface so that the route is still valid even if the IP address changes.