# IP Routing: Protocol-Independent Configuration Guide, Cisco IOS XE Release 3E

**Americas Headquarters**
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

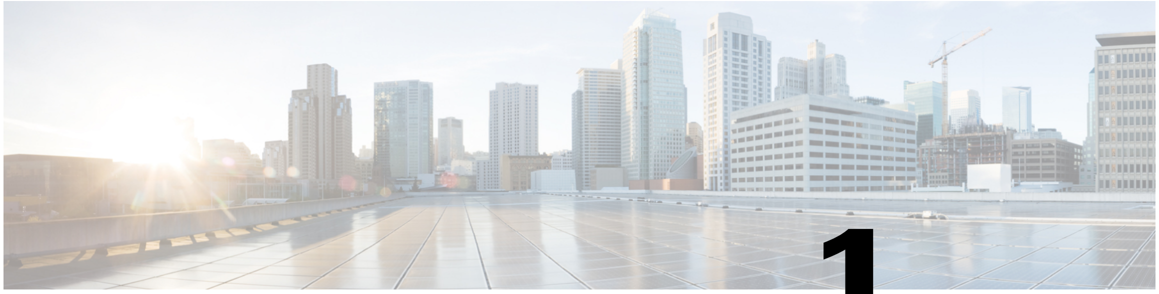
NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

# CONTENTS

CHAPTER 1



CHAPTER 1

# Basic IP Routing

This module describes how to configure basic IP routing. The Internet Protocol (IP) is a network layer (Layer 3) protocol that contains addressing information and some control information that enables packets to be routed. IP is documented in RFC 791 and is the primary network layer protocol in the Internet protocol suite.

## Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

## Information About Basic IP Routing

### Variable-Length Subnet Masks

Enhanced Interior Gateway Routing Protocol (EIGRP), Intermediate System-to-Intermediate System (IS-IS), Open Shortest Path First (OSPF), Routing Information Protocol (RIP) Version 2, and static routes support variable-length subnet masks (VLSMs). With VLSMs, you can use different masks for the same network number on different interfaces, which allows you to conserve IP addresses and more efficiently use available address space. However, using VLSMs also presents address assignment challenges for the network administrator and ongoing administrative challenges.

Refer to RFC 1219 for detailed information about VLSMs and how to correctly assign addresses.

**Note** Consider your decision to use VLSMs carefully. You can easily make mistakes in address assignments and you will generally find that the network is more difficult to monitor using VLSMs.

The best way to implement VLSMs is to keep your existing addressing plan in place and gradually migrate some networks to VLSMs to recover address space.

# Static Routes

Static routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. Static routes can be important if the device cannot build a route to a particular destination. They are also useful for specifying a gateway of last resort to which all unroutable packets will be sent.

To configure a static route, use the **ip route** *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [*distance*] [*name*] [**permanent** | **track** *number*] [**tag** *tag*] global configuration command.

Static routes remains in the device configuration until you remove them (using the **no ip route** global configuration command). However, you can override static routes with dynamic routing information through prudent assignment of administrative distance values. An administrative distance is a rating of the trustworthiness of a routing information source, such as an individual router or a group of routers. Numerically, an administrative distance is an integer from 0 to 255. In general, the higher the value, the lower the trust rating. An administrative distance of 255 means the routing information source cannot be trusted at all and should be ignored.

Each dynamic routing protocol has a default administrative distance, as listed in the table below. If you want a static route to be overridden by information from a dynamic routing protocol, simply ensure that the administrative distance of the static route is higher than that of the dynamic protocol.

***Table 1: Default Administrative Distances for Dynamic Routing Protocols***

| Route Source | Default Distance |
|---|---|
| Connected interface | 0 |
| Static route | 1 |
| Enhanced Interior Gateway Routing Protocol (EIGRP) summary route | 5 |
| External Border Gateway Protocol (BGP) | 20 |
| Internal EIGRP | 90 |
| Interior Gateway Routing Protocol (IGRP) | 100 |
| Open Shortest Path First (OSPF) | 110 |

| Route Source | Default Distance |
|---|---|
| intermediate System to Intermediate System (IS-IS) | 115 |
| Routing Information Protocol (RIP) | 120 |
| Exterior Gateway Routing Protocol (EGP) | 140 |
| On Demand Routing (ODR) | 160 |
| External EIGRP | 170 |
| Internal BGP | 200 |
| Unknown | 255 |

Static routes that point to an interface are advertised via RIP, EIGRP, and other dynamic routing protocols, regardless of whether **redistribute static** router configuration commands are specified for those routing protocols. These static routes are advertised because static routes that point to an interface are considered in the routing table to be connected and hence lose their static nature. However, if you define a static route to an interface that is not one of the networks defined in a **network** command, no dynamic routing protocols will advertise the route unless a **redistribute static** command is specified for these protocols.

When an interface goes down, all static routes through that interface are removed from the IP routing table. Also, when the software can no longer find a valid next hop for the address specified as the address of the forwarding device in a static route, the static route is removed from the IP routing table.

**Note** A packet with an E-class source address (240.0.0.0/4) gets dropped on Cisco ASR 1000 Series Aggregation Services Routers, although RFC 1812 (Requirements for IP Version 4 Routers) defines this behavior only for destination addresses and not specifically for source addresses.

# Default Routes

Default routes, also known as gateways of last resort, are used to route packets that are addressed to networks not explicitly listed in the routing table. A device might not be able to determine routes to all networks. To provide complete routing capability, network administrators use some devices as smart devices and give the remaining devices default routes to the smart device. (Smart devices have routing table information for the entire internetwork.) Default routes can be either passed along dynamically or configured manually into individual devices.

Most dynamic interior routing protocols include a mechanism for causing a smart device to generate dynamic default information, which is then passed along to other devices.

You can configure a default route by using the following commands:

- **ip default-gateway**
- **ip default-network**

- **ip route 0.0.0.0 0.0.0.0**

You can use the **ip default-gateway** global configuration command to define a default gateway when IP routing is disabled on a device. For instance, if a device is a host, you can use this command to define a default gateway for the device. You can also use this command to transfer a Cisco software image to a device when the device is in boot mode. In boot mode, IP routing is not enabled on the device.

Unlike the **ip default-gateway** command, the **ip default-network** command can be used when IP routing is enabled on a device. When you specify a network by using the **ip default-network** command, the device considers routes to that network for installation as the gateway of last resort on the device.

Gateways of last resort configured by using the **ip default-network** command are propagated differently depending on which routing protocol is propagating the default route. For Interior Gateway Routing Protocol (IGRP) and Enhanced Interior Gateway Routing Protocol (EIGRP) to propagate the default route, the network specified by the **ip default-network** command must be known to IGRP or EIGRP. The network must be an IGRP- or EIGRP-derived network in the routing table, or the static route used to generate the route to the network must be redistributed into IGRP or EIGRP or advertised into these protocols by using the **network** command. The Routing Information Protocol (RIP) advertises a route to network 0.0.0.0 if a gateway of last resort is configured by using the **ip default-network** command. The network specified in the **ip default-network** command need not be explicitly advertised under RIP.

Creating a static route to network 0.0.0.0 0.0.0.0 by using the **ip route 0.0.0.0 0.0.0.0** command is another way to set the gateway of last resort on a device. As with the **ip default-network** command, using the static route to 0.0.0.0 is not dependent on any routing protocols. However, IP routing must be enabled on the device. IGRP does not recognize a route to network 0.0.0.0. Therefore, it cannot propagate default routes created by using the **ip route 0.0.0.0 0.0.0.0** command. Use the **ip default-network** command to have IGRP propagate a default route.

EIGRP propagates a route to network 0.0.0.0, but the static route must be redistributed into the routing protocol.

Depending on your release of the Cisco software, the default route created by using the **ip route 0.0.0.0 0.0.0.0** command is automatically advertised by RIP devices. In some releases, RIP does not advertise the default route if the route is not learned via RIP. You might have to redistribute the route into RIP by using the **redistribute** command.

Default routes created using the **ip route 0.0.0.0 0.0.0.0** command are not propagated by Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (IS-IS). Additionally, these default routes cannot be redistributed into OSPF or IS-IS by using the **redistribute** command. Use the **default-information originate** command to generate a default route into an OSPF or IS-IS routing domain.

## Default Network

Default networks are used to route packets to destinations not established in the routing table. You can use the **ip default-network** *network-number* global configuration command to configure a default network when IP routing is enabled on the device. When you configure a default network, the device considers routes to that network for installation as the gateway of last resort on the device.

## Gateway of Last Resort

When default information is being passed along through a dynamic routing protocol, no further configuration is required. The system periodically scans its routing table to choose the optimal default network as its default route. In the case of the Routing Information Protocol (RIP), there is only one choice, network 0.0.0.0. In the case of Enhanced Interior Gateway Routing Protocol (EIGRP), there might be several networks that can be candidates for the system default. Cisco software uses both administrative distance and metric information to determine the default route (gateway of last resort). The selected default route appears in the gateway of last resort display of the **show ip route** privileged EXEC command.

If dynamic default information is not being passed to the software, candidates for the default route are specified with the **ip default-network** global configuration command. In this usage, the **ip default-network** command takes an unconnected network as an argument. If this network appears in the routing table from any source (dynamic or static), it is flagged as a candidate default route and is a possible choice as the default route.

If the device has no interface on the default network, but does have a route to it, it considers this network as a candidate default path. The route candidates are examined and the best one is chosen, based on administrative distance and metric. The gateway to the best default path becomes the gateway of last resort.

## Maximum Number of Paths

By default, most IP routing protocols install a maximum of four parallel routes in a routing table. Static routes always install six routes. The exception is Border Gateway Protocol (BGP), which by default allows only one path (the best path) to a destination. However, BGP can be configured to use equal and unequal cost multipath load sharing.

The number of parallel routes that you can configure to be installed in the routing table is dependent on the installed version of Cisco software. To change the maximum number of parallel paths allowed, use the **maximum-paths** *number-paths* command in router configuration mode.

## Multi-Interface Load Splitting

Multi-interface load splitting allows you to efficiently control traffic that travels across multiple interfaces to the same destination. The **traffic-share min** router configuration command specifies that if multiple paths are available to the same destination, only paths with the minimum metric will be installed in the routing table. The number of paths allowed is never more than six. For dynamic routing protocols, the number of paths is controlled by the **maximum-paths** router configuration command. The static route source can install six paths. If more paths are available, the extra paths are discarded. If some installed paths are removed from the routing table, pending routes are added automatically.

## Routing Information Redistribution

In addition to running multiple routing protocols simultaneously, Cisco software can be configured to redistribute information from one routing protocol to another. For example, you can configure a device to readvertise Enhanced Interior Gateway Routing Protocol (EIGRP)-derived routes using the Routing Information Protocol (RIP), or to readvertise static routes using the EIGRP protocol. Redistribution from one routing protocol to another can be configured in all of the IP-based routing protocols.

You also can conditionally control the redistribution of routes between routing domains by configuring route maps between the two domains. A route map is a route/packet filter that is configured with permit and deny statements, match and set clauses, and sequence numbers.

Although redistribution is a protocol-independent feature, some of the **match** and **set** commands are specific to a particular protocol.

One or more **match** commands and one or more **set** commands are configured in route map configuration mode. If there are no **match** commands, then everything matches. If there are no **set** commands, then no set action is performed.

To define a route map for redistribution, use the **route-map** *map-tag* [**permit** | **deny**] [*sequence-number*] global configuration command.

The metrics of one routing protocol do not necessarily translate into the metrics of another. For example, the RIP metric is a hop count and the EIGRP metric is a combination of five metric values. In such situations, a dynamic metric is assigned to the redistributed route. Redistribution in these cases should be applied consistently and carefully with inbound filtering to avoid routing loops.

Removing options that you have configured for the **redistribute** command requires careful use of the **no redistribute** command to ensure that you obtain the result that you are expecting.

## Supported Metric Translations

This section describes supported automatic metric translations between the routing protocols. The following descriptions assume that you have not defined a default redistribution metric that replaces metric conversions:

- The Routing Information Protocol (RIP) can automatically redistribute static routes. It assigns static routes a metric of 1 (directly connected).
- The Border Gateway Protocol (BGP) does not normally send metrics in its routing updates.
- The Enhanced Interior Gateway Routing Protocol (EIGRP) can automatically redistribute static routes from other EIGRP-routed autonomous systems as long as the static route and any associated interfaces are covered by an EIGRP network statement. EIGRP assigns static routes a metric that identifies them as directly connected. EIGRP does not change the metrics of routes derived from EIGRP updates from other autonomous systems.

**Note** Note that any protocol can redistribute routes from other routing protocols as long as a default metric is configured.

## Protocol Differences in Implementing the no redistribute Command

**Caution** Removing options that you have configured for the **redistribute** command requires careful use of the **no redistribute** command to ensure that you obtain the result that you are expecting. In most cases, changing or disabling any keyword will not affect the state of other keywords.

Different protocols implement the **no redistribute** command differently as follows:

- In Border Gateway Protocol (BGP), Open Shortest Path First (OSPF), and Routing Information Protocol (RIP) configurations, the **no redistribute** command removes only the specified keywords from the **redistribute** commands in the running configuration. They use the *subtractive keyword* method when redistributing from other protocols. For example, in the case of BGP, if you configure **no redistribute static route-map interior**, only the route map is removed from the redistribution, leaving **redistribute static** in place with no filter.
- The **no redistribute isis** command removes the Intermediate System to Intermediate System (IS-IS) redistribution from the running configuration. IS-IS removes the entire command, regardless of whether IS-IS is the redistributed or redistributing protocol.

- The Enhanced Interior Gateway Routing Protocol (EIGRP) used the subtractive keyword method prior to EIGRP component version rel5. Starting with EIGRP component version rel5, the **no redistribute** command removes the entire **redistribute** command when redistributing from any other protocol.

## Sources of Routing Information Filtering

Filtering sources of routing information prioritizes routing information from different sources because some pieces of routing information might be more accurate than others. An administrative distance is a rating of the trustworthiness of a routing information source, such as an individual device or a group of devices. In a large network, some routing protocols and some devices can be more reliable than others as sources of routing information. Also, when multiple routing processes are running in the same device for IP, the same route could be advertised by more than one routing process. By specifying administrative distance values, you enable the device to intelligently discriminate between sources of routing information. The device always picks the route whose routing protocol has the lowest administrative distance.

There are no general guidelines for assigning administrative distances because each network has its own requirements. You must determine a reasonable matrix of administrative distances for the network as a whole.

For example, consider a device using the Enhanced Interior Gateway Routing Protocol (EIGRP) and the Routing Information Protocol (RIP). Suppose you trust the EIGRP-derived routing information more than the RIP-derived routing information. In this example, because the default EIGRP administrative distance is lower than the default RIP administrative distance, the device uses the EIGRP-derived information and ignores the RIP-derived information. However, if you lose the source of the EIGRP-derived information (because of a power shutdown at the source network, for example), the device uses the RIP-derived information until the EIGRP-derived information reappears.

**Note** You can also use administrative distance to rate the routing information from devices that are running the same routing protocol. This application is generally discouraged if you are unfamiliar with this particular use of administrative distance because it can result in inconsistent routing information, including forwarding loops.

**Note** The weight of a route can no longer be set with the **distance** command. To set the weight for a route, use a route map.

## Authentication Key Management and Supported Protocols

Key management is a method of controlling the authentication keys used by routing protocols. Not all protocols support key management. Authentication keys are available for Director Response Protocol (DRP) Agent, Enhanced Interior Gateway Routing Protocol (EIGRP), and Routing Information Protocol (RIP) Version 2.

You can manage authentication keys by defining key chains, identifying the keys that belong to the key chain, and specifying how long each key is valid. Each key has its own key identifier (specified using the **key chain** configuration command), which is stored locally. The combination of the key identifier and the interface associated with the message uniquely identifies the authentication algorithm and the message digest algorithm 5 (MD5) authentication key in use.

You can configure multiple keys with lifetimes. Only one authentication packet is sent, regardless of how many valid keys exist. The software examines the key numbers in ascending order and uses the first valid key it encounters. The lifetimes allow for overlap during key changes.

# How to Configure Basic IP Routing

## Redistributing Routing Information

You can redistribute routes from one routing domain into another, with or without controlling the redistribution with a route map. To control which routes are redistributed, configure a route map and reference the route map from the **redistribute** command.

The tasks in this section describe how to define the conditions for redistributing routes (a route map), how to redistribute routes, and how to remove options for redistributing routes, depending on the protocol being used.

### Defining Conditions for Redistributing Routes

Route maps can be used to control route redistribution (or to implement policy-based routing). To define conditions for redistributing routes from one routing protocol into another, configure the **route-map** command. Then use at least one **match** command in route map configuration mode, as needed. At least one **match** command is used in this task because the purpose of the task is to illustrate how to define one or more conditions on which to base redistribution.

**Note** A route map is not required to have **match** commands; it can have only **set** commands. If there are no **match** commands, everything matches the route map.

**Note** There are many more **match** commands not shown in this table. For additional **match** commands, see the *Cisco IOS Master Command List*.

| Command or Action | Purpose |
|---|---|
| **match as-path** *path-list-number* | Matches a BGP autonomous system path access list. |
| **match community** {*standard-list-number* \| *expanded-list-number* \| *community-list-name* **match community** [**exact**] } | Matches a BGP community. |
| **match ip address** {*access-list-number* [*access-list-number...* \| *access-list-name...*] \| *access-list-name* [*access-list-number...* \| *access-list-name*] \| **prefix-list** *prefix-list-name* [*prefix-list-name...*] } | Matches routes that have a destination network address that is permitted to policy route packets or is permitted by a standard access list, an extended access list, or a prefix list. |
| **match metric** *metric-value* | Matches routes with the specified metric. |
| **match ip next-hop** {*access-list-number* \| *access-list-name*} [*access-list-number* \| *access-list-name*] | Matches a next-hop device address passed by one of the specified access lists. |
| **match tag** *tag-value* [*tag-value*] | Matches the specified tag value. |
| **match interface** *type number* [*type number*] | Matches routes that use the specified interface as the next hop. |
| **match ip route-source** {*access-list-number* \| *access-list-name*} [*access-list-number* \| *access-list-name*] | Matches the address specified by the advertised access lists. |
| **match route-type** {**local** \| **internal** \| **external** [**type-1** \| **type-2**] \| **level-1** \| **level-2**} | Matches the specified route type. |

To optionally specify the routing actions for the system to perform if the match criteria are met (for routes that are being redistributed by the route map), use one or more **set** commands in route map configuration mode, as needed.

**Note** A route map is not required to have **set** commands; it can have only **match** commands.

**Note** There are more **set** commands not shown in this table. For additional **set** commands, see the *Cisco IOS Master Command List*.

| Command or Action | Purpose |
|---|---|
| **set community** {*community-number* [**additive**] [**well-known**] \| **none**} | Sets the community attribute (for BGP). |
| **set dampening** *halflife reuse suppress max-suppress-time* | Sets route dampening parameters (for BGP). |
| **set local-preference** *number-value* | Assigns a local preference value to a path (for BGP). |
| **set origin** {**igp** \| **egp** *as-number* \| **incomplete**} | Sets the route origin code. |
| **set as-path**{**tag** \| **prepend** *as-path-string* } | Modifies the autonomous system path (for BGP). |
| **set next-hop** *next-hop* | Specifies the address of the next hop. |
| **set automatic-tag** | Enables automatic computation of the tag table. |
| **set level** {**level-1** \| **level-2** \| **level-1-2** \| **stub-area** \| **backbone**} | Specifies the areas to import routes. |
| **set metric** *metric-value* | Sets the metric value for redistributed routes (for any protocol, except EIGRP). |
| **set metric** *bandwidth delay reliability load mtu* | Sets the metric value for redistributed routes (for EIGRP only). |
| **set metric-type** {**internal** \| **external** \| **type-1** \| **type-2**} | Sets the metric type for redistributed routes. |
| **set metric-type internal** | Sets the Multi Exit Discriminator (MED) value on prefixes advertised to the external BGP neighbor to match the Interior Gateway Protocol (IGP) metric of the next hop. |
| **set tag** *tag-value* | Sets a tag value to be applied to redistributed routes. |

## Redistributing Routes from One Routing Domain to Another

Perform this task to redistribute routes from one routing domain into another and to control route redistribution. This task shows how to redistribute OSPF routes into a BGP domain.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *autonomous-system*
4. **redistribute** *protocol process-id*
5. **default-metric** *number*
6. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **router bgp** *autonomous-system*<br>**Example:**<br>`Device(config)# router bgp 109` | Enables a BGP routing process and enters router configuration mode. |
| **Step 4** | **redistribute** *protocol process-id*<br>**Example:**<br>`Device(config-router)# redistribute ospf 2 1` | Redistributes routes from the specified routing domain into another routing domain. |
| **Step 5** | **default-metric** *number*<br>**Example:**<br>`Device(config-router)# default-metric 10` | Sets the default metric value for redistributed routes.<br>**Note** The metric value specified in the **redistribute** command supersedes the metric value specified using the **default-metric** command. |
| **Step 6** | **end**<br>**Example:**<br>`Device(config-router)# end` | Exits router configuration mode and returns to privileged EXEC mode. |

## Removing Options for Redistribution Routes

**Caution** Removing options that you have configured for the **redistribute** command requires careful use of the **no redistribute** command to ensure that you obtain the result that you are expecting.

Different protocols implement the **no redistribute** command differently as follows:

- In BGP, OSPF, and RIP configurations, the **no redistribute** command removes only the specified keywords from the **redistribute** commands in the running configuration. They use the *subtractive keyword* method when redistributing from other protocols. For example, in the case of BGP, if you configure **no redistribute static route-map interior**, only the route map is removed from the redistribution, leaving **redistribute static** in place with no filter.
- The **no redistribute isis** command removes the IS-IS redistribution from the running configuration. IS-IS removes the entire command, regardless of whether IS-IS is the redistributed or redistributing protocol.
- EIGRP used the subtractive keyword method prior to EIGRP component version rel5. Starting with EIGRP component version rel5, the **no redistribute** command removes the entire **redistribute** command when redistributing from any other protocol.
- For the **no redistribute connected** command, the behavior is subtractive if the **redistribute** command is configured under the **router bgp** or the **router ospf** command. The behavior is complete removal of the command if it is configured under the **router isis** or the **router eigrp** command.

The following OSPF commands illustrate how various options are removed from the redistribution in router configuration mode.

| Command or Action | Purpose |
|---|---|
| **no redistribute connected metric 1000 subnets** | Removes the configured metric value of 1000 and the configured subnets and retains the **redistribute connected** command in the configuration. |
| **no redistribute connected metric 1000** | Removes the configured metric value of 1000 and retains the **redistribute connected subnets** command in the configuration. |
| **no redistribute connected subnets** | Removes the configured subnets and retains the **redistribute connected metric** *metric-value* command in the configuration. |
| **no redistribute connected** | Removes the **redistribute connected** command and any of the options that were configured for the command. |

# Configuring Routing Information Filtering

**Note** When routes are redistributed between Open Shortest Path First (OSPF) processes, no OSPF metrics are preserved.

## Controlling the Advertising of Routes in Routing Updates

To prevent other devices from learning one or more routes, you can suppress routes from being advertised in routing updates. To suppress routes from being advertised in routing updates, use the **distribute-list** {*access-list-number* | *access-list-name*} **out** [*interface-name* | *routing-process* | *as-number*] command in router configuration mode.

You cannot specify an interface name in Open Shortest Path First (OSPF). When used for OSPF, this feature applies only to external routes.

## Controlling the Processing of Routing Updates

You might want to avoid processing certain routes that are listed in incoming updates (this does not apply to Open Shortest Path First [OSPF] or Intermediate System to Intermediate System [IS-IS]). To suppress routes in incoming updates, use the **distribute-list** {*access-list-number* | *access-list-name*} **in** [*interface-type interface-number*] command in router configuration mode.

## Filtering Sources of Routing Information

To filter sources of routing information, use the **distance** *ip-address wildcard- mask* [*ip-standard-acl* | *ip-extended-acl* | *access-list-name*] command in router configuration mode.

# Managing Authentication Keys

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **key chain** *name-of-chain*
4. **key** *number*
5. **key-string** *text*
6. **accept-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*}
7. **send-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*}
8. **end**
9. **show key chain**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>You can configure multiple keys with lifetimes. Only one authentication packet is sent, regardless of how many valid keys exist. The software examines the key numbers in ascending order and uses the first valid key it encounters. The lifetimes allow for overlap during key changes.<br>`Device> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | **key chain** *name-of-chain*<br>**Example:**<br>`Device(config)# key chain chain1` | Defines a key chain and enters key-chain configuration mode. |
| **Step 4** | **key** *number*<br>**Example:**<br>`Device(config-keychain)# key 1` | Identifies number of an authentication key on a key chain. The range of keys is from 0 to 2147483647. The key identification numbers need not be consecutive. |
| **Step 5** | **key-string** *text*<br>**Example:**<br>`Device(config-keychain-key)# key-string string1` | Identifies the key string. |
| **Step 6** | **accept-lifetime** *start-time* {**infinite** \| *end-time* \| **duration** *seconds*}<br>**Example:**<br>`Device(config-keychain-key)# accept-lifetime 13:30:00 Dec 22 2011 duration 7200` | Specifies the time period during which the key can be received. |
| **Step 7** | **send-lifetime** *start-time* {**infinite** \| *end-time* \| **duration** *seconds*}<br>**Example:**<br>`Device(config-keychain-key)# send-lifetime 14:30:00 Dec 22 2011 duration 3600` | Specifies the time period during which the key can be sent. |
| **Step 8** | **end**<br>**Example:**<br>`Device(config-keychain-key)# end` | Exits key-chain key configuration mode and returns to privileged EXEC mode. |
| **Step 9** | **show key chain**<br>**Example:**<br>`Device# show key chain` | (Optional) Displays authentication key information. |

# Monitoring and Maintaining the IP Network

## Clearing Routes from the IP Routing Table

You can remove all contents of a particular table. Clearing a table may become necessary when the contents of the particular structure have become, or are suspected to be, invalid.

To clear one or more routes from the IP routing table, use the **clear ip route** {*network* [*mask*] | *****} command in privileged EXEC mode.

### Displaying System and Network Statistics

You can use the following **show** commands to display system and network statistics. You can display specific statistics such as contents of IP routing tables, caches, and databases. You can also display information about node reachability and discover the routing path that packets leaving your device are taking through the network. This information can an be used to determine resource utilization and solve network problems.

| Command or Action | Purpose |
|---|---|
| **show ip cache policy** | Displays cache entries in the policy route cache. |
| **show ip local policy** | Displays the local policy route map if one exists. |
| **show ip policy** | Displays policy route maps. |
| **show ip protocols** | Displays the parameters and current state of the active routing protocols. |
| **show ip route** [*ip-address* [*mask*] [**longer-prefixes**] \| *protocol* [*process-id*] \| **list** {**access-list-number** \| *access-list-name*} \| **static download**] | Displays the current state of the routing table. |
| **show ip route summary** | Displays the current state of the routing table in summary form. |
| **show ip route supernets-only** | Displays supernets. |
| **show key chain** [*name-of-chain*] | Displays authentication key information. |
| **show route-map** [*map-name*] | Displays all route maps configured or only the one specified. |

# Configuration Examples for Basic IP Routing

## Example: Variable-Length Subnet Mask

The following example uses two different subnet masks for the class B network address of 172.16.0.0. A subnet mask of /24 is used for LAN interfaces. The /24 mask allows 265 subnets with 254 host IP addresses on each subnet. The final subnet of the range of possible subnets using a /24 mask (172.16.255.0) is reserved for use on point-to-point interfaces and assigned a longer mask of /30. The use of a /30 mask on 172.16.255.0 creates 64 subnets (172.16.255.0 to 172.16.255.252) with 2 host addresses on each subnet.

**Caution:** To ensure unambiguous routing, you must not assign 172.16.255.0/24 to a LAN interface in your network.

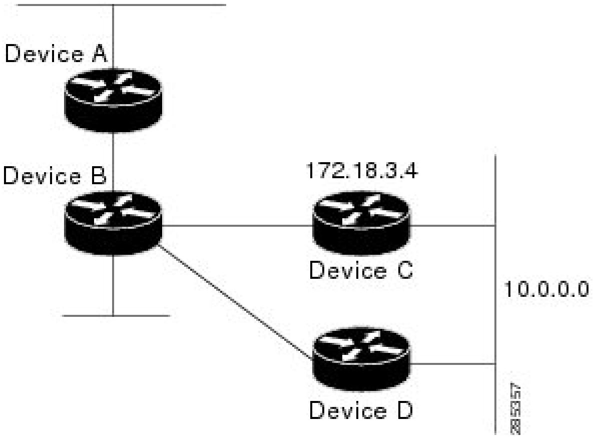
```
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# ip address 172.16.1.1 255.255.255.0
Device(config-if)# ! 8 bits of host address space reserved for GigabitEthernet interfaces
Device(config-if)# exit
Device(config)# interface Serial 0/0/0
Device(config-if)# ip address 172.16.255.5 255.255.255.252
Device(config-if)# ! 2 bits of address space reserved for point-to-point serial interfaces
Device(config-if)# exit
Device(config)# router rip
Device(config-router)# network 172.16.0.0
Device(config-router)# ! Specifies the network directly connected to the device
```

# Example: Overriding Static Routes with Dynamic Protocols

In the following example, packets for network 10.0.0.0 from Device B (where the static route is installed) will be routed through 172.18.3.4 if a route with an administrative distance less than 110 is not available. The figure below illustrates this example. The route learned by a protocol with an administrative distance of less than 110 might cause Device B to send traffic destined for network 10.0.0.0 via the alternate path through Device D.

```
Device(config)# ip route 10.0.0.0 255.0.0.0 172.18.3.4 110
```

*Figure 1: Overriding Static Routes*



# Example: IP Default Gateway as a Static IP Next Hop When IP Routing Is Disabled

The following example shows how to configure IP address 172.16.5.4 as the default route when IP routing is disabled:

```
Device> enable
Device# configure terminal
```

```
Device(conf)# no ip routing
Device(conf)# ip default-gateway 172.16.15.4
```

# Examples: Administrative Distances

In the following example, the **router eigrp** global configuration command configures Enhanced Interior Gateway Routing Protocol (EIGRP) routing in autonomous system 1. The **network** command configuration specifies EIGRP routing on networks 192.168.7.0 and 172.16.0.0. The first **distance** router configuration command sets the default administrative distance to 255, which instructs the device to ignore all routing updates from devices for which an explicit distance has not been set. The second **distance** command sets the administrative distance to 80 for internal EIGRP routes and to 100 for external EIGRP routes. The third **distance** command sets the administrative distance to 120 for the device with the address 172.16.1.3.

```
Device(config)# router eigrp 1
Device(config-router)# network 192.168.7.0
Device(config-router)# network 172.16.0.0
Device(config-router)# distance 255
Device(config-router)# distance eigrp 80 100
Device(config-router)# distance 120 172.16.1.3 0.0.0.0
```

**Note** The **distance eigrp** command must be used to set the administrative distance for EIGRP-derived routes.

The following example assigns the device with the address 192.168.7.18 an administrative distance of 100 and all other devices on subnet 192.168.7.0 an administrative distance of 200:

```
Device(config-router)# distance 100 192.168.7.18 0.0.0.0
Device(config-router)# distance 200 192.168.7.0 0.0.0.255
```

However, if you reverse the order of these two commands, all devices on subnet 192.168.7.0 are assigned an administrative distance of 200, including the device at address 192.168.7.18:

```
Device(config-router)# distance 200 192.168.7.0 0.0.0.255
Device(config-router)# distance 100 192.168.7.18 0.0.0.0
```

**Note** Assigning administrative distances can be used to solve unique problems. However, administrative distances should be applied carefully and consistently to avoid the creation of routing loops or other network failures.

In the following example, the distance value for IP routes learned is 90. Preference is given to these IP routes rather than routes with the default administrative distance value of 110.

```
Device(config)# router isis
Device(config-router)# distance 90 ip
```

# Example: Static Routing Redistribution

In the example that follows, three static routes are specified, two of which are to be advertised. The static routes are created by specifying the **redistribute static** router configuration command and then specifying an access list that allows only those two networks to be passed to the Enhanced Interior Gateway Routing Protocol (EIGRP) process. Any redistributed static routes should be sourced by a single device to minimize the likelihood of creating a routing loop.

```
Device(config)# ip route 192.168.2.0 255.255.255.0 192.168.7.65
Device(config)# ip route 192.168.5.0 255.255.255.0 192.168.7.65
Device(config)# ip route 10.10.10.0 255.255.255.0 10.20.1.2
Device(config)# !
Device(config)# access-list 3 permit 192.168.2.0 0.0.255.255
Device(config)# access-list 3 permit 192.168.5.0 0.0.255.255
Device(config)# access-list 3 permit 10.10.10.0 0.0.0.255
Device(config)# !
Device(config)# router eigrp 1
Device(config-router)# network 192.168.0.0
Device(config-router)# network 10.10.10.0
Device(config-router)# redistribute static metric 10000 100 255 1 1500
Device(config-router)# distribute-list 3 out static
```

# Examples: EIGRP Redistribution

Each Enhanced Interior Gateway Routing Protocol (EIGRP) routing process provides routing information to only one autonomous system. The Cisco software must run a separate EIGRP process and maintain a separate routing database for each autonomous system that it services. However, you can transfer routing information between these routing databases.

In the following configuration, network 10.0.0.0 is configured under EIGRP autonomous system 1 and network 192.168.7.0 is configured under EIGRP autonomous system 101:

```
Device(config)# router eigrp 1
Device(config-router)# network 10.0.0.0
Device(config-router)# exit
Device(config)# router eigrp 101
Device(config-router)# network 192.168.7.0
```

In the following example, routes from the 192.168.7.0 network are redistributed into autonomous system 1 (without passing any other routing information from autonomous system 101):

```
Device(config)# access-list 3 permit 192.168.7.0
Device(config)# !
Device(config)# route-map 101-to-1 permit 10
Device(config-route-map)# match ip address 3
Device(config-route-map)# set metric 10000 100 1 255 1500
Device(config-route-map)# exit
Device(config)# router eigrp 1
Device(config-router)# redistribute eigrp 101 route-map 101-to-1
Device(config-router)#!
```

The following example is an alternative way to redistribute routes from the 192.168.7.0 network into autonomous system 1. Unlike the previous configuration, this method does not allow you to set the metric for redistributed routes.

```
Device(config)# access-list 3 permit 192.168.7.0
Device(config)# !
Device(config)# router eigrp 1
Device(config-router)# redistribute eigrp 101
Device(config-router)# distribute-list 3 out eigrp 101
Device(config-router)# !
```

# Example: Mutual Redistribution Between EIGRP and RIP

Consider a WAN at a university that uses the Routing Information Protocol (RIP) as an interior routing protocol. Assume that the university wants to connect its WAN to regional network 172.16.0.0, which uses the Enhanced Interior Gateway Routing Protocol (EIGRP) as the routing protocol. The goal in this case is to advertise the networks in the university network to devices in the regional network.

Mutual redistribution is configured between EIGRP and RIP in the following example:

```
Device(config)# access-list 10 permit 172.16.0.0
Device(config)# !
Device(config)# router eigrp 1
Device(config-router)# network 172.16.0.0
Device(config-router)# redistribute rip metric 10000 100 255 1 1500
Device(config-router)# default-metric 10
Device(config-router)# distribute-list 10 out rip
Device(config-router)# exit
Device(config)# router rip
Device(config-router)# redistribute eigrp 1
Device(config-router)# !
```

In this example, an EIGRP routing process is started. The **network** router configuration command specifies that network 172.16.0.0 (the regional network) is to send and receive EIGRP routing information. The **redistribute** router configuration command specifies that RIP-derived routing information be advertised in routing updates. The **default-metric** router configuration command assigns an EIGRP metric to all RIP-derived routes. The **distribute-list** router configuration command instructs the Cisco software to use access list 10 (not defined in this example) to limit the entries in each outgoing update. The access list prevents unauthorized advertising of university routes to the regional network.

## Example: Mutual Redistribution Between EIGRP and BGP

In the following example, mutual redistribution is configured between the Enhanced Interior Gateway Routing Protocol (EIGRP) and the Border Gateway Protocol (BGP).

Routes from EIGRP routing process 101 are injected into BGP autonomous system 50000. A filter is configured to ensure that the correct routes are advertised, in this case, three networks. Routes from BGP autonomous system 50000 are injected into EIGRP routing process 101. The same filter is used.

```
Device(config)# ! All networks that should be advertised from R1 are controlled with ACLs:

Device(config)# access-list 1 permit 172.18.0.0 0.0.255.255
Device(config)# access-list 1 permit 172.16.0.0 0.0.255.255
Device(config)# access-list 1 permit 172.25.0.0 0.0.255.255
Device(config)# ! Configuration for router R1:
Device(config)# router bgp 50000
Device(config-router)# network 172.18.0.0
Device(config-router)# network 172.16.0.0
Device(config-router)# neighbor 192.168.10.1 remote-as 2
Device(config-router)# neighbor 192.168.10.15 remote-as 1
Device(config-router)# neighbor 192.168.10.24 remote-as 3
Device(config-router)# redistribute eigrp 101
Device(config-router)# distribute-list 1 out eigrp 101
Device(config-router)# exit
Device(config)# router eigrp 101
Device(config-router)# network 172.25.0.0
Device(config-router)# redistribute bgp 50000
Device(config-router)# distribute-list 1 out bgp 50000
Device(config-router)# !
```

**Caution** BGP should be redistributed into an Interior Gateway Protocol (IGP) when there are no other suitable options. Redistribution from BGP into any IGP should be applied with proper filtering by using distribute lists, IP prefix lists, and route map statements to limit the number of prefixes.

# Examples: OSPF Routing and Route Redistribution

OSPF typically requires coordination among many internal devices, area border routers (ABRs), and Autonomous System Boundary Routers (ASBRs). At a minimum, OSPF-based devices can be configured with all default parameter values, with no authentication, and with interfaces assigned to areas.

This section provides the following configuration examples:

- The first example shows simple configurations illustrating basic OSPF commands.
- The second example shows configurations for an internal device, ABR, and ASBR within a single, arbitrarily assigned OSPF autonomous system.
- The third example illustrates a more complex configuration and the application of various tools available for controlling OSPF-based routing environments.

## Examples: Basic OSPF Configuration

The following example illustrates a simple OSPF configuration that enables OSPF routing process 1, attaches Gigabit Ethernet interface 0/0/0 to area 0.0.0.0, and redistributes RIP into OSPF and OSPF into RIP:

```
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# ip address 172.16.1.1 255.255.255.0
Device(config-if)# ip ospf cost 1
Device(config-if)# exit
Device(config)# interface GigabitEthernet 1/0/0
Device(config-if)# ip address 172.17.1.1 255.255.255.0
Device(config-if)# exit
Device(config)# router ospf 1
Device(config-router)# network 172.18.0.0 0.0.255.255 area 0.0.0.0
Device(config-router)# redistribute rip metric 1 subnets
Device(config-router)# exit
Device(config)# router rip
Device(config-router)# network 172.17.0.0
Device(config-router)# redistribute ospf 1
Device(config-router)# default-metric 1
Device(config-router)# !
```

The following example illustrates the assignment of four area IDs to four IP address ranges. In the example, OSPF routing process 1 is initialized, and four OSPF areas are defined: 10.9.50.0, 2, 3, and 0. Areas 10.9.50.0, 2, and 3 mask specific address ranges, whereas area 0 enables OSPF for all other networks.

```
Device(config)# router ospf 1
Device(config-router)# network 172.18.20.0 0.0.0.255 area 10.9.50.0
Device(config-router)# network 172.18.0.0 0.0.255.255 area 2
Device(config-router)# network 172.19.10.0 0.0.0.255 area 3
Device(config-router)# network 0.0.0.0 255.255.255.255 area 0
Device(config-router)# exit
Device(config)# ! GigabitEthernet interface 0/0/0 is in area 10.9.50.0:
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# ip address 172.18.20.5 255.255.255.0
Device(config-if)# exit
Device(config)# ! GigabitEthernet interface 1/0/0 is in area 2:
Device(config)# interface GigabitEthernet 1/0/0
Device(config-if)# ip address 172.18.1.5 255.255.255.0
Device(config-if)# exit
Device(config)# ! GigabitEthernet interface 2/0/0 is in area 2:
Device(config)# interface GigabitEthernet 2/0/0
Device(config-if)# ip address 172.18.2.5 255.255.255.0
Device(config-if)# exit
Device(config)# ! GigabitEthernet interface 3/0/0 is in area 3:
```

```
Device(config)# interface GigabitEthernet 3/0/0
Device(config-if)# ip address 172.19.10.5 255.255.255.0
Device(config-if)# exit
Device(config)# ! GigabitEthernet interface 4/0/0 is in area 0:
Device(config)# interface GigabitEthernet 4/0/0
Device(config-if)# ip address 172.19.1.1 255.255.255.0
Device(config-if)# exit
Device(config)# ! GigabitEthernet interface 5/0/0 is in area 0:
Device(config)# interface GigabitEthernet 5/0/0
Device(config-if)# ip address 10.1.0.1 255.255.0.0
Device(config-if)# !
```

Each **network** router configuration command is evaluated sequentially, so the specific order of these commands in the configuration is important. The Cisco software sequentially evaluates the *address/wildcard-mask* pair for each interface. See the *IP Routing Protocols Command Reference* for more information.

Consider the first **network** command. Area ID 10.9.50.0 is configured for the interface on which subnet 172.18.20.0 is located. Assume that a match is determined for Gigabit Ethernet interface 0/0/0. Gigabit Ethernet interface 0/0/0 is attached to Area 10.9.50.0 only.

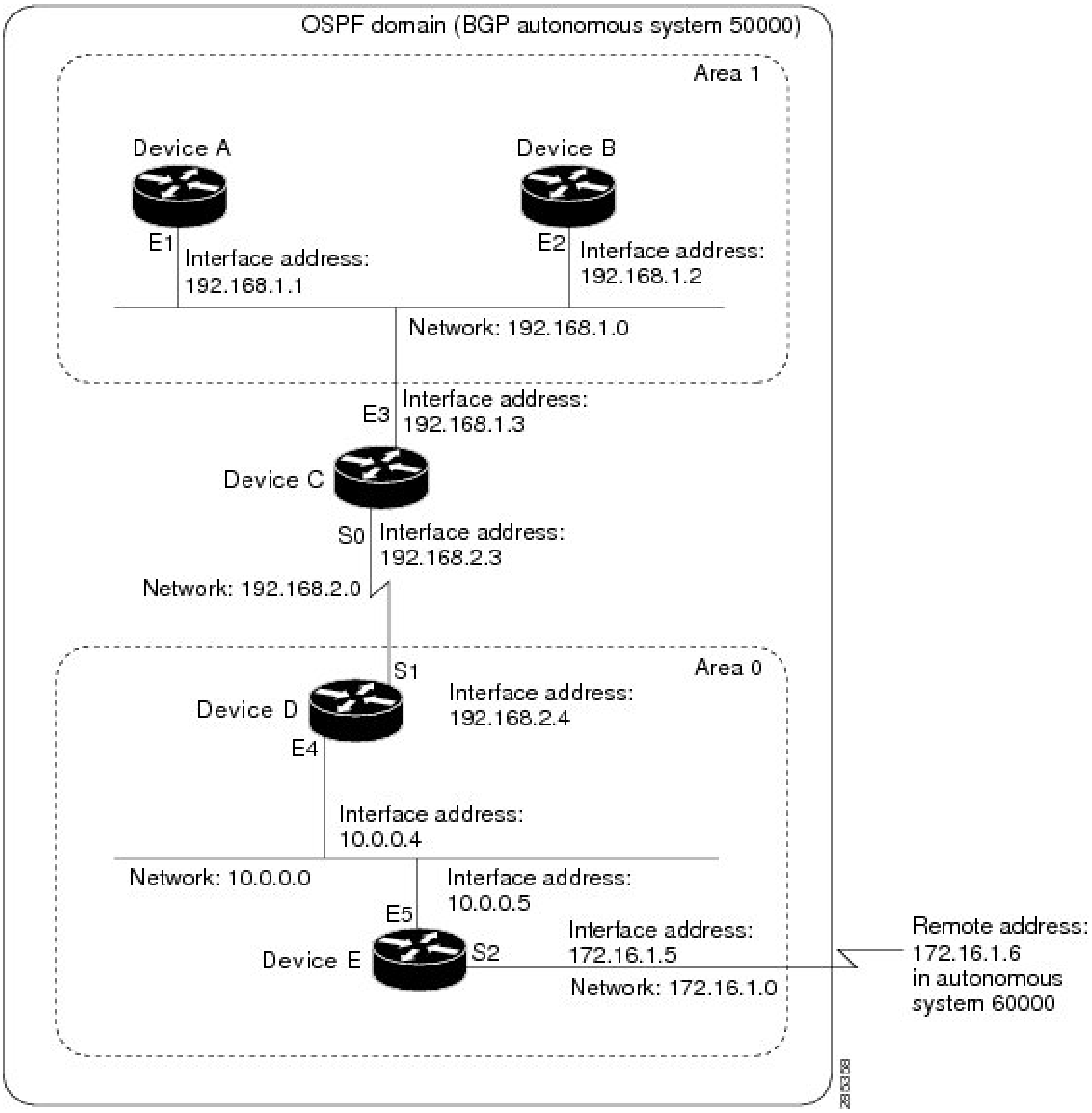
The second **network** command is evaluated next. For Area 2, the same process is then applied to all interfaces (except Gigabit Ethernet interface 0/0/0). Assume that a match is determined for Gigabit Ethernet interface 1/0/0. OSPF is then enabled for that interface, and Gigabit Ethernet 1/0/0 is attached to Area 2.

This process of attaching interfaces to OSPF areas continues for all **network** commands. Note that the last **network** command in this example is a special case. With this command, all available interfaces (not explicitly attached to another area) are attached to Area 0.

## Example: Internal Device ABR and ASBRs Configuration

The figure below provides a general network map that illustrates a sample configuration for several devices within a single OSPF autonomous system.

*Figure 2: Example OSPF Autonomous System Network Map*



In this configuration, five devices are configured in OSPF autonomous system 1:

- Device A and Device B are both internal devices within area 1.
- Device C is an OSPF ABR. Note that for Device C, area 1 is assigned to E3 and Area 0 is assigned to S0.

- Device D is an internal device in area 0 (backbone area). In this case, both **network** router configuration commands specify the same area (area 0, or the backbone area).
- Device E is an OSPF ASBR. Note that the Border Gateway Protocol (BGP) routes are redistributed into OSPF and that these routes are advertised by OSPF.

> **Note** Definitions of all areas in an OSPF autonomous system need not be included in the configuration of all devices in the autonomous system. You must define only the *directly* connected areas. In the example that follows, routes in Area 0 are learned by the devices in area 1 (Device A and Device B) when the ABR (Device C) injects summary link state advertisements (LSAs) into area 1.

Autonomous system 60000 is connected to the outside world via the BGP link to the external peer at IP address 172.16.1.6.

Following is the sample configuration for the general network map shown in the figure above.

### Device A Configuration--Internal Device

```
Device(config)# interface GigabitEthernet 1/0/0
Device(config-if)# ip address 192.168.1.1 255.255.255.0
Device(config-if)# exit
Device(config)# router ospf 1
Device(config-router)# network 192.168.1.0 0.0.0.255 area 1
Device(config-router)# exit
```

### Device B Configuration--Internal Device

```
Device(config)# interface GigabitEthernet 2/0/0
Device(config-if)# ip address 192.168.1.2 255.255.255.0
Device(config-if)# exit
Device(config)# router ospf 1
Device(config-router)# network 192.168.1.0 0.0.0.255 area 1
Device(config-router)# exit
```

### Device C Configuration--ABR

```
Device(config)# interface GigabitEthernet 3/0/0
Device(config-if)# ip address 192.168.1.3 255.255.255.0
Device(config-if)# exit
Device(config)# interface Serial 0/0/0
Device(config-if)# ip address 192.168.2.3 255.255.255.0
Device(config-if)# exit
Device(config)# router ospf 1
Device(config-router)# network 192.168.1.0 0.0.0.255 area 1
Device(config-router)# network 192.168.2.0 0.0.0.255 area 0
Device(config-router)# exit
```

### Device D Configuration--Internal Device

```
Device(config)# interface GigabitEthernet 4/0/0
Device(config-if)# ip address 10.0.0.4 255.0.0.0
Device(config-if)# exit
Device(config)# interface Serial 1/0/0
Device(config-if)# ip address 192.168.2.4 255.255.255.0
Device(config-if)# exit
Device(config)# router ospf 1
Device(config-router)# network 192.168.2.0 0.0.0.255 area 0
Device(config-router)# network 10.0.0.0 0.255.255.255 area 0
Device(config-router)# exit
```

### Device E Configuration--ASBR

```
Device(config)# interface GigabitEthernet 5/0/0
Device(config-if)# ip address 10.0.0.5 255.0.0.0
Device(config-if)# exit
Device(config)# interface Serial 2/0/0
Device(config-if)# ip address 172.16.1.5 255.255.255.0
Device(config-if)# exit
Device(config)# router ospf 1
Device(config-router)# network 10.0.0.0 0.255.255.255 area 0
Device(config-router)# redistribute bgp 50000 metric 1 metric-type 1
Device(config-router)# exit
Device(config)# router bgp 50000
Device(config-router)# network 192.168.0.0
Device(config-router)# network 10.0.0.0
Device(config-router)# neighbor 172.16.1.6 remote-as 60000
```

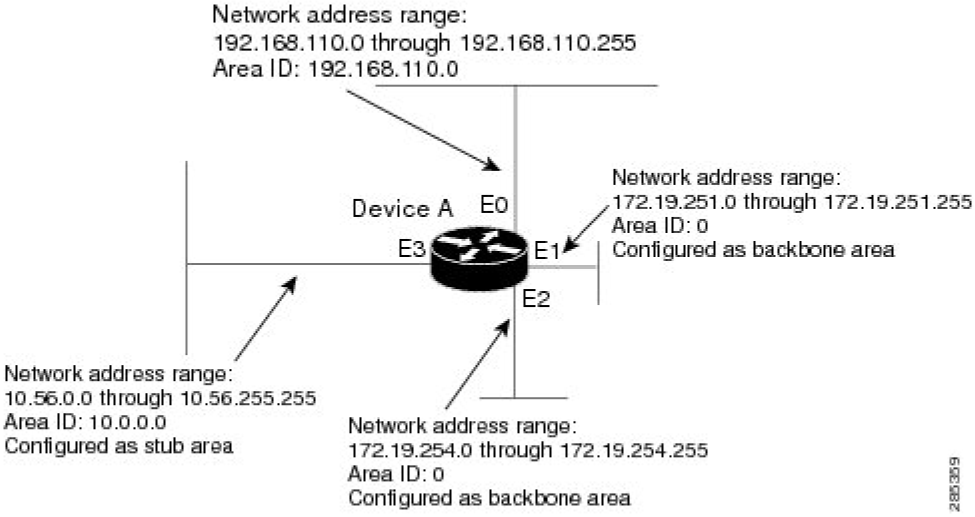
## Example: Complex OSPF Configuration

The following sample configuration accomplishes several tasks in setting up an ABR. These tasks can be split into two general categories:

- Basic OSPF configuration
- Route redistribution

The specific tasks outlined in this configuration are detailed briefly in the following descriptions. The figure below illustrates the network address ranges and area assignments for the interfaces.

*Figure 3: Interface and Area Specifications for OSPF Configuration Example*


Network address range:
192.168.110.0 through 192.168.110.255
Area ID: 192.168.110.0
Device A
E0
E1
E2
E3
Network address range:
172.19.251.0 through 172.19.251.255
Area ID: 0
Configured as backbone area
Network address range:
10.56.0.0 through 10.56.255.255
Area ID: 10.0.0.0
Configured as stub area
Network address range:
172.19.254.0 through 172.19.254.255
Area ID: 0
Configured as backbone area
285369

The basic configuration tasks in this example are as follows:

- Configure address ranges for Gigabit Ethernet interface 0/0/0 through Gigabit Ethernet interface 3/0/0.
- Enable OSPF on each interface.

- Set up an OSPF authentication password for each area and network.
- Assign link-state metrics and other OSPF interface configuration options.
- Create a stub area with area ID 10.0.0.0. (Note that the **authentication** and **stub** options of the **area** router configuration command are specified with separate **area** command entries, but they can be merged into a single **area** command.)
- Specify the backbone area (area 0).

Configuration tasks associated with redistribution are as follows:

- Redistribute the Enhanced Interior Gateway Routing Protocol (EIGRP) and the Routing Information Protocol (RIP) into OSPF with various options set (including metric-type, metric, tag, and subnet).
- Redistribute EIGRP and OSPF into RIP.

The following is an example OSPF configuration:

```
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# ip address 192.168.110.201 255.255.255.0
Device(config-if)# ip ospf authentication-key abcdefgh
Device(config-if)# ip ospf cost 10
Device(config-if)# exit
Device(config)# interface GigabitEthernet 1/0/0
Device(config-if)# ip address 172.19.251.201 255.255.255.0
Device(config-if)# ip ospf authentication-key ijklmnop
Device(config-if)# ip ospf cost 20
Device(config-if)# ip ospf retransmit-interval 10
Device(config-if)# ip ospf transmit-delay 2
Device(config-if)# ip ospf priority 4
Device(config-if)# exit
Device(config)# interface GigabitEthernet 2/0/0
Device(config-if)# ip address 172.19.254.201 255.255.255.0
Device(config-if)# ip ospf authentication-key abcdefgh
Device(config-if)# ip ospf cost 10
Device(config-if)# exit
Device(config)# interface GigabitEthernet 3/0/0
Device(config-if)# ip address 10.56.0.201 255.255.0.0
Device(config-if)# ip ospf authentication-key ijklmnop
Device(config-if)# ip ospf cost 20
Device(config-if)# ip ospf dead-interval 80
Device(config-if)# exit
```

In the following configuration, OSPF is on network 172.19.0.0:

```
Device(config)# router ospf 1
Device(config-router)# network 10.0.0.0 0.255.255.255 area 10.0.0.0
Device(config-router)# network 192.168.110.0 0.0.0.255 area 192.168.110.0
Device(config-router)# network 172.19.0.0 0.0.255.255 area 0
Device(config-router)# area 0 authentication
Device(config-router)# area 10.0.0.0 stub
Device(config-router)# area 10.0.0.0 authentication
Device(config-router)# area 10.0.0.0 default-cost 20
Device(config-router)# area 192.168.110.0 authentication
Device(config-router)# area 10.0.0.0 range 10.0.0.0 255.0.0.0
Device(config-router)# area 192.168.110.0 range 192.168.110.0 255.255.255.0
Device(config-router)# area 0 range 172.19.251.0 255.255.255.0
Device(config-router)# area 0 range 172.19.254.0 255.255.255.0
Device(config-router)# redistribute eigrp 200 metric-type 2 metric 1 tag 200 subnets
Device(config-router)# redistribute rip metric-type 2 metric 1 tag 200
Device(config-router)# exit
```

In the following configuration, EIGRP autonomous system 1 is on 172.19.0.0:

```
Device(config)# router eigrp 1
Device(config-router)# network 172.19.0.0
Device(config-router)# exit
Device(config)# ! RIP for 192.168.110.0:
Device(config)# router rip
Device(config-router)# network 192.168.110.0
Device(config-router)# redistribute eigrp 1 metric 1
Device(config-router)# redistribute ospf 201 metric 1
Device(config-router)# exit
```

## Example: Default Metric Values Redistribution

The following example shows a device in autonomous system 1 that is configured to run both the Routing Information Protocol (RIP) and the Enhanced Interior Gateway Routing Protocol (EIGRP). The example advertises EIGRP-derived routes using RIP and assigns the EIGRP-derived routes a RIP metric of 10.

```
Device(config)# router rip
Device(config-router)# redistribute eigrp 1
Device(config-router)# default-metric 10
Device(config-router)# exit
```

## Examples: Redistribution With and Without Route Maps

The examples in this section illustrate the use of redistribution, with and without route maps. Examples from both the IP and Connectionless Network Service (CLNS) routing protocols are given. The following example redistributes all Open Shortest Path First (OSPF) routes into the Enhanced Interior Gateway Routing Protocol (EIGRP):

```
Device(config)# router eigrp 1
Device(config-router)# redistribute ospf 101
Device(config-router)# exit
```

The following example redistributes Routing Information Protocol (RIP) routes with a hop count equal to 1 into OSPF. These routes will be redistributed into OSPF as external link state advertisements (LSAs) with a metric of 5, metric a type of type 1, and a tag equal to 1.

```
Device(config)# router ospf 1
Device(config-router)# redistribute rip route-map rip-to-ospf
Device(config-router)# exit
Device(config)# route-map rip-to-ospf permit
Device(config-route-map)# match metric 1
Device(config-route-map)# set metric 5
Device(config-route-map)# set metric-type type 1
Device(config-route-map)# set tag 1
Device(config-route-map)# exit
```

The following example redistributes OSPF learned routes with tag 7 as a RIP metric of 15:

```
Device(config)# router rip
Device(config-router)# redistribute ospf 1 route-map 5
Device(config-router)# exit
Device(config)# route-map 5 permit
Device(config-route-map)# match tag 7
Device(config-route-map)# set metric 15
```

The following example redistributes OSPF intra-area and interarea routes with next hop devices on serial interface 0/0/0 into the Border Gateway Protocol (BGP) with an INTER_AS metric of 5:

```
Device(config)# router bgp 50000
Device(config-router)# redistribute ospf 1 route-map 10
Device(config-router)# exit
Device(config)# route-map 10 permit
Device(config-route-map)# match route-type internal
Device(config-route-map)# match interface serial 0/0/0
Device(config-route-map)# set metric 5
```

The following example redistributes two types of routes into the integrated IS-IS routing table (supporting both IP and CLNS). The first type is OSPF external IP routes with tag 5; these routes are inserted into Level 2 IS-IS link-state packets (LSPs) with a metric of 5. The second type is ISO-IGRP derived CLNS prefix routes that match CLNS access list 2000; these routes will be redistributed into IS-IS as Level 2 LSPs with a metric of 30.

```
Device(config)# router isis
Device(config-router)# redistribute ospf 1 route-map 2
Device(config-router)# redistribute iso-igrp nsfnet route-map 3

Device(config-router)# exit
Device(config)# route-map 2 permit
Device(config-route-map)# match route-type external
Device(config-route-map)# match tag 5
Device(config-route-map)# set metric 5
Device(config-route-map)# set level level-2
Device(config-route-map)# exit
Device(config)# route-map 3 permit
Device(config-route-map)# match address 2000
Device(config-route-map)# set metric 30
Device(config-route-map)# exit
```

With the following configuration, OSPF external routes with tags 1, 2, 3, and 5 are redistributed into RIP with metrics of 1, 1, 5, and 5, respectively. The OSPF routes with a tag of 4 are not redistributed.

```
Device(config)# router rip
Device(config-router)# redistribute ospf 101 route-map 1
Device(config-router)# exit
Device(config)# route-map 1 permit
Device(config-route-map)# match tag 1 2
Device(config-route-map)# set metric 1
Device(config-route-map)# exit
Device(config)# route-map 1 permit
Device(config-route-map)# match tag 3
Device(config-route-map)# set metric 5
Device(config-route-map)# exit
Device(config)# route-map 1 deny
Device(config-route-map)# match tag 4
Device(config-route-map)# exit
Device(config)# route map 1 permit
Device(config-route-map)# match tag 5
Device(config-route-map)# set metric 5
Device(config-route-map)# exit
```

Given the following configuration, a RIP learned route for network 172.18.0.0 and an ISO-IGRP learned route with prefix 49.0001.0002 will be redistributed into an IS-IS Level 2 LSP with a metric of 5:

```
Device(config)# router isis
Device(config-router)# redistribute rip route-map 1
Device(config-router)# redistribute iso-igrp remote route-map 1
Device(config-router)# exit
Device(config)# route-map 1 permit
Device(config-route-map)# match ip address 1
Device(config-route-map)# match clns address 2
Device(config-route-map)# set metric 5
Device(config-route-map)# set level level-2
Device(config-route-map)# exit
Device(config)# access-list 1 permit 172.18.0.0 0.0.255.255
Device(config)# clns filter-set 2 permit 49.0001.0002...
```

The following configuration example illustrates how a route map is referenced by the **default-information** router configuration command. This type of reference is called conditional default origination. OSPF will originate the default route (network 0.0.0.0) with a type 2 metric of 5 if 172.20.0.0 is in the routing table.

```
Device(config)# route-map ospf-default permit
Device(config-route-map)# match ip address 1
Device(config-route-map)# set metric 5
Device(config-route-map)# set metric-type type-2
Device(config-route-map)# exit
Device(config)# access-list 1 172.20.0.0 0.0.255.255
Device(config)# router ospf 101
Device(config-router)# default-information originate route-map ospf-default
```

# Examples: Key Management

The following example configures a key chain named chain1. In this example, the software always accepts and sends key1 as a valid key. The key key2 is accepted from 1:30 p.m. to 3:30 p.m. and is sent from 2:00 p.m. to 3:00 p.m. The overlap allows for migration of keys or discrepancy in the set time of the device. Likewise, the key key3 immediately follows key2, and there is 30-minutes on each side to handle time-of-day differences.

```
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# ip rip authentication key-chain chain1
Device(config-if)# ip rip authentication mode md5
Device(config-if)# exit
Device(config)# router rip
Device(config-router)# network 172.19.0.0
Device(config-router)# version 2
Device(config-router)# exit
Device(config)# key chain chain1
Device(config-keychain)# key 1
Device(config-keychain-key)# key-string key1
Device(config-keychain-key)# exit
Device(config-keychain)# key 2
Device(config-keychain-key)# key-string key2
Device(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 2005 duration 7200
Device(config-keychain-key)# send-lifetime 14:00:00 Jan 25 2005 duration 3600
Device(config-keychain-key)# exit
Device(config-keychain)# key 3
Device(config-keychain-key)# key-string key3
Device(config-keychain-key)# accept-lifetime 14:30:00 Jan 25 2005 duration 7200
Device(config-keychain-key)# send-lifetime 15:00:00 Jan 25 2005 duration 3600
Device(config-keychain-key)# end
```

The following example configures a key chain named chain1:

```
Device(config)# key chain chain1
Device(config-keychain)# key 1
Device(config-keychain-key)# key-string key1
Device(config-keychain-key)# exit
Device(config-keychain)# key 2
Device(config-keychain-key)# key-string key2
Device(config-keychain-key)# accept-lifetime 00:00:00 Dec 5 2004 23:59:59 Dec 5 2005
Device(config-keychain-key)# send-lifetime 06:00:00 Dec 5 2004 18:00:00 Dec 5 2005
Device(config-keychain-key)# exit
Device(config-keychain)# exit
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# ip address 172.19.104.75 255.255.255.0 secondary 172.19.232.147
255.255.255.240
Device(config-if)# ip rip authentication key-chain chain1
Device(config-if)# media-type 10BaseT
Device(config-if)# exit
Device(config)# interface GigabitEthernet 1/0/0
Device(config-if)# no ip address
Device(config-if)# shutdown
Device(config-if)# media-type 10BaseT
Device(config-if)# exit
Device(config)# interface Fddi 0
Device(config-if)# ip address 10.1.1.1 255.255.255.0
Device(config-if)# no keepalive
Device(config-if)# exit
Device(config)# interface Fddi 1/0/0
Device(config-if)# ip address 172.16.1.1 255.255.255.0
Device(config-if)# ip rip send version 1
Device(config-if)# ip rip receive version 1
Device(config-if)# no keepalive
Device(config-if)# exit
Device(config)# router rip
Device(config-router)# version 2
Device(config-router)# network 172.19.0.0
Device(config-router)# network 10.0.0.0
Device(config-router)# network 172.16.0.0
```

# Additional References

### Related Documents

| Related Topic | Document Title |
|---|---|
| Cisco IOS commands | Cisco IOS Master Command List, All Releases |
| IP routing protocol-independent commands | Cisco IOS IP Routing: Protocol-Independent Command Reference |

### Technical Assistance

| Description | Link |
|---|---|
| The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password. | http://www.cisco.com/cisco/web/support/index.html |

# Feature Information for Basic IP Routing

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

*Table 2: Feature Information for Basic IP Routing*

| Feature Name | Releases | Feature Information |
|---|---|---|
| IP Routing | Cisco IOS XE Release 3.2SE<br>Cisco IOS XE Release 3.2SE<br>Cisco IOS XE Release 3.3SE | The IP Routing feature introduced basic IP routing features that are documented throughout this module and also in other IP Routing Protocol modules.<br>In Cisco IOS XE Release 3.2SE, support was added for the Cisco Catalyst 3850 Series Switches.<br>In Cisco IOS XE Release 3.2SE, support was added for the Cisco 5700 Series Wireless LAN Controllers.<br>In Cisco IOS XE Release 3.3SE, support was added for the Cisco Catalyst 3650 Series Switches and Cisco Catalyst 3850 Series Switches. |

CHAPTER 2

# Fast-Switched Policy Routing

Currently policy-based routing is process-switched within the software, which means that on most platforms the switching rate was approximately 1000 to 10,000 packets per second. Such rates are not fast enough for many applications. The Fast-Switched Policy Routing feature allows customers to direct certain traffic profiles to take specified paths through the network, enabling source-based routing and leveraging security.

## Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

## Prerequisites for Fast-Switched Policy Routing

Policy routing must be configured before you configure fast-switched policy routing.

# Information About Fast-Switched Policy Routing

## Fast-Switched Policy Routing

IP policy routing can be fast-switched. Prior to fast-switched policy routing, policy routing could only be process -switched, which meant that on most platforms, the switching rate was approximately 1000 to 10,000 packets per second. Such rates were not fast enough for many applications. With fast-switched policy routing, users who need policy routing to occur at faster speeds can implement policy routing without slowing down the device.

Fast-switched policy routing supports all **match** commands and most **set** commands, except for the following:

- **set ip default**
- **set interface**

The **set interface** command is supported only over point-to-point links, unless there is a route cache entry that uses the same interface that is specified in the command in the route map.

To configure fast-switched policy routing, use the **ip route-cache policy** interface configuration command.

# Additional References

### Related Documents

| Related Topic | Document Title |
|---|---|
| Cisco IOS commands | Cisco IOS Master Command List, All Releases |
| IP routing protocol-independent commands | Cisco IOS IP Routing: Protocol-Independent Command Reference |

### Technical Assistance

| Description | Link |
|---|---|
| The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password. | http://www.cisco.com/cisco/web/support/index.html |

# Feature Information for Fast-Switched Policy Routing

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

***Table 3: Feature Information for Fast-Switched Policy Routing***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Fast-Switched Policy Routing | Cisco IOS XE Release 3.2SE<br>Cisco IOS XE Release 3.2SE<br>Cisco IOS XE Release 3.3SE | Currently policy-based routing is process-switched within the software, which means that on most platforms the switching rate was approximately 1000 to 10,000 packets per second. Such rates are not fast enough for many applications. The Fast-Switched Policy Routing feature allows customers to direct certain traffic profiles to take specified paths through the network, enabling source-based routing and leveraging security.<br>In Cisco IOS XE Release 3.2SE, support was added for the Cisco Catalyst 3850 Series Switches.<br>In Cisco IOS XE Release 3.2SE, support was added for the Cisco 5700 Series Wireless LAN Controllers.<br>In Cisco IOS XE Release 3.3SE, support was added for the Cisco Catalyst 3650 Series Switches and Cisco Catalyst 3850 Series Switches. |

CHAPTER 3

# Policy-Based Routing

The Policy-Based Routing feature is a process whereby a device puts packets through a route map before routing the packets. The route map determines which packets are routed next to which device. Policy-based routing is a more flexible mechanism for routing packets than destination routing.

## Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

## Prerequisites for Policy-Based Routing

For Policy-Based Routing, IPBase is a minimum licensing requirement.

# Information About Policy-Based Routing

## Policy-Based Routing

Policy-based routing is a process whereby the device puts packets through a route map before routing them. The route map determines which packets are routed to which device next. You might enable policy-based routing if you want certain packets to be routed some way other than the obvious shortest path. Possible applications for policy-based routing are to provide equal access, protocol-sensitive routing, source-sensitive routing, routing based on interactive versus batch traffic, and routing based on dedicated links. Policy-based routing is a more flexible mechanism for routing packets than destination routing.

To enable policy-based routing, you must identify which route map to use for policy-based routing and create the route map. The route map itself specifies the match criteria and the resulting action if all of the match clauses are met.

To enable policy-based routing on an interface, indicate which route map the device should use by using the **ip policy route-map** *map-tag* command in interface configuration mode. A packet arriving on the specified interface is subject to policy-based routing except when its destination IP address is the same as the IP address of the device's interface. This **ip policy route-map** command disables fast switching of all packets arriving on this interface.

To define the route map to be used for policy-based routing, use the **route-map** *map-tag* [**permit** | **deny**] [*sequence-number*] global configuration command.

Only **set ip next-hop** command can be used under route-map configuration mode when you configure policy-based routing.

To define the criteria by which packets are examined to learn if they will be policy-based routed, use either the **match length** *minimum-length maximum-length* command or the **match ip address** {*access-list-number* | *access-list-name*} [*access-list-number* | *access-list-name*] command or both in route map configuration mode. No match clause in the route map indicates all packets.

To display the cache entries in the policy route cache, use the **show ip cache policy** command.

**Note** Mediatrace will show statistics of incorrect interfaces with policy-based routing (PBR) if the PBR does not interact with CEF or Resource Reservation Protocol (RSVP). Hence configure PBR to interact with CEF or RSVP directly so that mediatrace collects statistics only on tunnel interfaces and not physical interfaces.

# How to Configure Policy-Based Routing

## Configuring Policy-Based Routing

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **no switchport**
5. **ip policy route-map** *map-tag*
6. **exit**
7. **route-map** *map-tag* [**permit** | **deny**] [*sequence-number*]
8. Enter one or both of the following commands:
    - **match length**
    - **match ip address**
9. **set ip next-hop** *ip-address* [...*ip-address*]
10. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **interface** *type number*<br>**Example:**<br>`Device(config)# interface gigabitethernet 1/0/0` | Configures an interface type and enters interface configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 4** | **no switchport**<br>**Example:**<br>`Device(config-if)# no switchport` | Puts an interface into Layer 3 mode. |
| **Step 5** | **ip policy route-map** *map-tag*<br>**Example:**<br>`Device(config-if)# ip policy route-map equal-access` | Identifies a route map to use for policy routing on an interface. |
| **Step 6** | **exit**<br>**Example:**<br>`Device(config-if)# exit` | Returns to global configuration mode. |
| **Step 7** | **route-map** *map-tag* [**permit** \| **deny**] [*sequence-number*]<br>**Example:**<br>`Device(config)# route-map equal-access permit 10` | Defines the conditions for redistributing routes from one routing protocol into another routing protocol or enables policy-based routing and enters route-map configuration mode.<br>**Note** Only **set ip next-hop** command can be used under route-map configuration mode when you configure policy-based routing.<br>• *map-tag*—A meaningful name for the route map.<br>• **permit**—(Optional) If the match criteria are met for this route map, and the **permit** keyword is specified, the route is redistributed as controlled by the set actions. In the case of policy routing, the packet is policy routed. If the match criteria are not met, and the **permit** keyword is specified, the next route map with the same map tag is tested. If a route passes none of the match criteria for the set of route maps sharing the same name, it is not redistributed by that set.<br>• **deny**—(Optional) If the match criteria are met for the route map and the **deny** keyword is specified, the route is not redistributed. In the case of policy routing, the packet is not policy routed, and no further route maps sharing the same map tag name will be examined. If the packet is not policy routed, the normal forwarding algorithm is used.<br>• *sequence-number*—(Optional) Number that indicates the position a new route map will have in the list of route maps already configured with the same name. If used with the **no** form of this command, the position of the route map **configure terminal** should be deleted. |
| **Step 8** | Enter one or both of the following commands:<br>• **match length**<br>• **match ip address** | Define the criteria by which packets are examined to learn if they will be policy-based routed. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Device(config-route-map)# match ip address 1` | |
| **Step 9** | **set ip next-hop** *ip-address* [...*ip-address*]<br>**Example:**<br>`Device(config-route-map)# set ip next-hop 172.16.6.6` | Specifies where to output packets that pass a match clause of a route map for policy routing. |
| **Step 10** | **end**<br>**Example:**<br>`Device(config-route-map)# end` | Exits route-map configuration mode and returns to privileged EXEC mode. |

# Configuration Examples for Policy-Based Routing

# Additional References

### Related Documents

| Related Topic | Document Title |
|---|---|
| Cisco IOS commands | Cisco IOS Master Command List, All Releases |
| IP routing protocol-independent commands | Cisco IOS IP Routing: Protocol-Independent Command Reference |

### Technical Assistance

| Description | Link |
|---|---|
| The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password. | http://www.cisco.com/cisco/web/support/index.html |

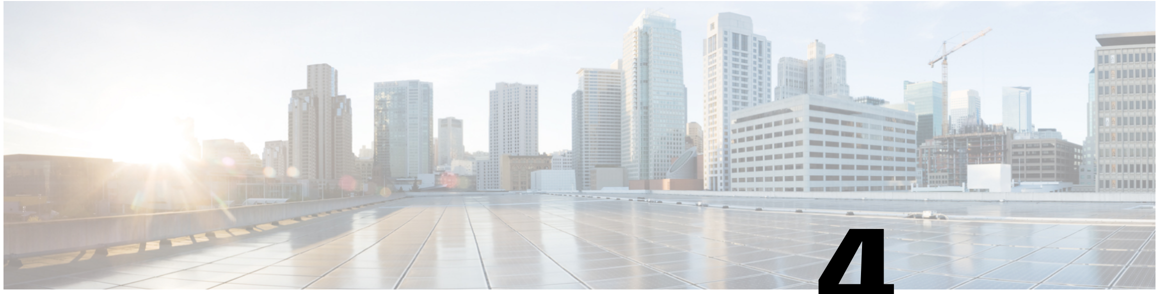
# Feature Information for Policy-Based Routing

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

***Table 4: Feature Information for Policy-Based Routing***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Policy-Based Routing | Cisco IOS XE Release 3.2SE<br>Cisco IOS XE Release 3.3SE | The Policy-Based Routing feature is a process whereby a device puts packets through a route map before routing the packets. The route map determines which packets are routed next to which device. Policy-Based Routing introduces a more flexible mechanism for routing packets than destination routing.<br>In Cisco IOS XE Release 3.2SE, support was added for the Cisco Catalyst 3850 Series Switches.<br>In Cisco IOS XE Release 3.3SE , support was added for the Cisco Catalyst 3650 Series Switches and Cisco Catalyst 3850 Series Switches.<br>The following command was introduced or modified: **ip policy route-map**. |



CHAPTER 4

# IPv6 Policy-Based Routing

Policy-based routing (PBR) in both IPv6 and IPv4 allows a user to manually configure how received packets should be routed. PBR allows the user to identify packets by using several attributes and to specify the next hop or the output interface to which the packet should be sent. PBR also provides a basic packet-marking capability.

# Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

# Information About IPv6 Policy-Based Routing

## Policy-Based Routing Overview

Policy-based routing (PBR) gives you a flexible means of routing packets by allowing you to configure a defined policy for traffic flows, which lessens reliance on routes derived from routing protocols. Therefore, PBR gives you more control over routing by extending and complementing the existing mechanisms provided by routing protocols. PBR allows you to set the IPv6 precedence. For a simple policy, you can use any one of these tasks; for a complex policy, you can use all of them. It also allows you to specify a path for certain traffic, such as priority traffic over a high-cost link. IPv6 PBR is supported on Cisco ASR 1000 Series platform.

PBR for IPv6 may be applied to both forwarded and originated IPv6 packets. For forwarded packets, PBR for IPv6 will be implemented as an IPv6 input interface feature, supported in the following forwarding paths:

- Process
- Cisco Express Forwarding (formerly known as CEF)
- Distributed Cisco Express Forwarding

Policies can be based on the IPv6 address, port numbers, protocols, or packet size.

PBR allows you to perform the following tasks:

- Classify traffic based on extended access list criteria. Access lists, then, establish the match criteria.
- Set IPv6 precedence bits, giving the network the ability to enable differentiated classes of service.
- Route packets to specific traffic-engineered paths; you might need to route them to allow a specific quality of service (QoS) through the network.

PBR allows you to classify and mark packets at the edge of the network. PBR marks a packet by setting precedence value. The precedence value can be used directly by devices in the network core to apply the appropriate QoS to a packet, which keeps packet classification at your network edge.

# How Policy-Based Routing Works

All packets received on an interface with policy-based routing (PBR) enabled are passed through enhanced packet filters called route maps. The route maps used by PBR dictate the policy, determining where to forward packets.

Route maps are composed of statements. The route map statements can be marked as permit or deny, and they are interpreted in the following ways:

- If a packet matches all match statements for a route map that is marked as permit, the device attempts to policy route the packet using the set statements. Otherwise, the packet is forwarded normally.
- If the packet matches any match statements for a route map that is marked as deny, the packet is not subject to PBR and is forwarded normally.
- If the statement is marked as permit and the packets do not match any route map statements, the packets are sent back through normal forwarding channels and destination-based routing is performed.

You must configure policy-based routing (PBR) on the interface that receives the packet, and not on the interface from which the packet is sent.

## Packet Matching

Policy-based routing (PBR) for IPv6 will match packets using the **match ipv6 address** command in the associated PBR route map. Packet match criteria are those criteria supported by IPv6 access lists, as follows:

- Input interface

- Source IPv6 address (standard or extended access control list [ACL])
- Destination IPv6 address (standard or extended ACL)
- Protocol (extended ACL)
- Source port and destination port (extended ACL)
- DSCP (extended ACL)
- Flow-label (extended ACL)
- Fragment (extended ACL)

Packets may also be matched by length using the **match length** command in the PBR route map.

Match statements are evaluated first by the criteria specified in the **match ipv6 address** command and then by the criteria specified in the **match length** command. Therefore, if both an ACL and a length statement are used, a packet will first be subject to an ACL match. Only packets that pass the ACL match will be subject to the length match. Finally, only packets that pass both the ACL and the length statement will be policy routed.

## Packet Forwarding Using Set Statements

Policy-based routing (PBR) for IPv6 packet forwarding is controlled by using a number of set statements in the PBR route map. These set statements are evaluated individually in the order shown, and PBR will attempt to forward the packet using each of the set statements in turn. PBR evaluates each set statement individually, without reference to any prior or subsequent set statement.

You may set multiple forwarding statements in the PBR for IPv6 route map. The following set statements may be specified:

- IPv6 next hop. The next hop to which the packet should be sent. The next hop must be present in the Routing Information Base (RIB), it must be directly connected, and it must be a global IPv6 address. If the next hop is invalid, the set statement is ignored.
- Output interface. A packet is forwarded out of a specified interface. An entry for the packet destination address must exist in the IPv6 RIB, and the specified output interface must be in the set path. If the interface is invalid, the statement is ignored.
- Default IPv6 next hop. The next hop to which the packet should be sent. It must be a global IPv6 address. This set statement is used only when there is no explicit entry for the packet destination in the IPv6 RIB.
- Default output interface. The packet is forwarded out of a specified interface. This set statement is used only when there is no explicit entry for the packet destination in the IPv6 RIB.

**Note** The order in which PBR evaluates the set statements is the order in which they are listed above. This order may differ from the order in which route-map set statements are listed by **show** commands.

## When to Use Policy-Based Routing

Policy-based routing (PBR) can be used if you want certain packets to be routed some way other than the obvious shortest path. For example, PBR can be used to provide the following functionality:

- Equal access
- Protocol-sensitive routing
- Source-sensitive routing
- Routing based on interactive traffic versus batch traffic
- Routing based on dedicated links

Some applications or traffic can benefit from Quality of Service (QoS)-specific routing; for example, you could transfer stock records to a corporate office on a higher-bandwidth, higher-cost link for a short time while sending routine application data such as e-mail over a lower-bandwidth, lower-cost link.

# How to Enable IPv6 Policy-Based Routing

## Enabling IPv6 PBR on an Interface

To enable Policy-Based Routing (PBR) for IPv6, you must create a route map that specifies the packet match criteria and desired policy-route action. Then you associate the route map on the required interface. All packets arriving on the specified interface that match the match clauses will be subject to PBR.

In PBR, the **set vrf** command decouples the virtual routing and forwarding (VRF) instance and interface association and allows the selection of a VRF based on access control list (ACL)-based classification using existing PBR or route-map configurations. It provides a single router with multiple routing tables and the ability to select routes based on ACL classification. The router classifies packets based on ACL, selects a routing table, looks up the destination address, and then routes the packet.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **route-map** *map-tag* [**permit** | **deny**] [*sequence-number*]
4. Do one of the following:
   - **match length** *minimum-length maximum-length*
   - **match ipv6 address** {**prefix-list** *prefix-list-name* | *access-list-name*}
5. Do one of the following:
   - **set ipv6 precedence** *precedence-value*
   - **set ipv6 next-hop** *global-ipv6-address* [*global-ipv6-address...*]
   - **set interface** *type number* [*...type number*]
   - **set ipv6 default next-hop** *global-ipv6-address* [*global-ipv6-address...*]
   - **set default interface** *type number* [*...type number*]
   - **set vrf** *vrf-name*
6. **exit**
7. **interface** *type number*
8. **ipv6 policy route-map** *route-map-name*
9. **end**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **route-map** *map-tag* [**permit** \| **deny**] [*sequence-number*]<br>**Example:**<br>`Device(config)# route-map rip-to-ospf permit` | Defines the conditions for redistributing routes from one routing protocol into another, or enables policy routing, and enters route-map configuration mode. |
| **Step 4** | Do one of the following: | Specifies the match criteria. |

| | Command or Action | Purpose |
|---|---|---|
| | • **match length** *minimum-length maximum-length*<br>• **match ipv6 address** {**prefix-list** *prefix-list-name* \| *access-list-name*}<br>**Example:**<br>`Device(config-route-map)# match length 3 200`<br>**Example:**<br>`Device(config-route-map)# match ipv6 address marketing` | • You can specify any or all of the following:<br>• Matches the Level 3 length of the packet.<br>• Matches a specified IPv6 access list.<br>• If you do not specify a **match** command, the route map applies to all packets. |
| **Step 5** | Do one of the following:<br>• **set ipv6 precedence** *precedence-value*<br>• **set ipv6 next-hop** *global-ipv6-address* [*global-ipv6-address...*]<br>• **set interface** *type number* [*...type number*]<br>• **set ipv6 default next-hop** *global-ipv6-address* [*global-ipv6-address...*]<br>• **set default interface** *type number* [*...type number*]<br>• **set vrf** *vrf-name*<br>**Example:**<br>`Device(config-route-map)# set ipv6 precedence 1`<br>**Example:**<br>`Device(config-route-map)# set ipv6 next-hop 2001:DB8:2003:1::95`<br>**Example:**<br>`Device(config-route-map)# set interface GigabitEthernet 0/0/1`<br>**Example:**<br>`Device(config-route-map)# set ipv6 default next-hop 2001:DB8:2003:1::95` | Specifies the action or actions to take on the packets that match the criteria.<br>• You can specify any or all of the following:<br>• Sets precedence value in the IPv6 header.<br>• Sets next hop to which to route the packet (the next hop must be adjacent).<br>• Sets output interface for the packet.<br>• Sets next hop to which to route the packet, if there is no explicit route for this destination.<br>• Sets output interface for the packet, if there is no explicit route for this destination.<br>• Sets VRF instance selection within a route map for a policy-based routing VRF selection. |

| | Command or Action | Purpose |
|---|---|---|
| | **Example:**<br>`Device(config-route-map)# set default interface GigabitEthernet 0/0/0`<br>**Example:**<br>`Device(config-route-map)# set vrf vrfname` | |
| **Step 6** | **exit**<br>**Example:**<br>`Device(config-route-map)# exit` | Exits route-map configuration mode and returns to global configuration mode. |
| **Step 7** | **interface** *type number*<br>**Example:**<br>`Device(config)# interface FastEthernet 1/0` | Specifies an interface type and number, and places the router in interface configuration mode. |
| **Step 8** | **ipv6 policy route-map** *route-map-name*<br>**Example:**<br>`Device(config-if)# ipv6 policy-route-map interactive` | Identifies a route map to use for IPv6 PBR on an interface. |
| **Step 9** | **end**<br>**Example:**<br>`Device(config-if)# end` | Exits interface configuration mode and returns to privileged EXEC mode. |

# Enabling Local PBR for IPv6

Packets that are generated by the device are not normally policy routed. Perform this task to enable local IPv6 policy-based routing (PBR) for such packets, indicating which route map the device should use.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 local policy route-map** *route-map-name*
4. **end**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br>**Example:**<br>`Device# configure terminal` | Enters global configuration mode. |
| **Step 3** | **ipv6 local policy route-map** *route-map-name*<br>**Example:**<br>`Device(config)# ipv6 local policy route-map pbr-src-90` | Configures IPv6 PBR for packets generated by the device. |
| **Step 4** | **end**<br>**Example:**<br>`Device(config)# end` | Returns to privileged EXEC mode. |

# Verifying the Configuration and Operation of PBR for IPv6

### SUMMARY STEPS

1. **enable**
2. **show ipv6 policy**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 2** | **show ipv6 policy**<br>**Example:**<br>`Device# show ipv6 policy` | Displays IPv6 policy routing packet activity. |

# Troubleshooting PBR for IPv6

Policy routing analyzes various parts of the packet and then routes the packet based on certain user-defined attributes in the packet.

### SUMMARY STEPS

1. **enable**
2. **show route-map** [*map-name* | **dynamic** [*dynamic-map-name* | **application** [*application-name*]] | **all**] [**detailed**]
3. **debug ipv6 policy** [*access-list-name*]

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br>**Example:**<br>`Device> enable` | Enables privileged EXEC mode.<br>• Enter your password if prompted. |
| **Step 2** | **show route-map** [*map-name* \| **dynamic** [*dynamic-map-name* \| **application** [*application-name*]] \| **all**] [**detailed**]<br>**Example:**<br>`Device# show route-map` | Displays all route maps configured or only the one specified. |
| **Step 3** | **debug ipv6 policy** [*access-list-name*]<br>**Example:**<br>`Device# debug ipv6 policy` | Enables debugging of the IPv6 policy routing packet activity. |

# Configuration Examples for IPv6 Policy-Based Routing

## Example: Enabling PBR on an Interface

In the following example, a route map named pbr-dest-1 is created and configured, specifying packet match criteria and desired policy-route action. PBR is then enabled on GigabitEthernet interface 0/0/1.

```
ipv6 access-list match-dest-1
  permit ipv6 any 2001:DB8:2001:1760::/32
route-map pbr-dest-1 permit 10
  match ipv6 address match-dest-1
  set interface GigabitEthernet 0/0/0
interface GigabitEthernet0/0/1
  ipv6 policy-route-map interactive
```

## Example: Enabling Local PBR for IPv6

In the following example, packets with a destination IPv6 address that match the IPv6 address range allowed by access list pbr-src-90 are sent to the device at IPv6 address 2001:DB8:2003:1::95:

```
ipv6 access-list src-90
  permit ipv6 host 2001:DB8:2003::90 2001:DB8:2001:1000::/64
route-map pbr-src-90 permit 10
  match ipv6 address src-90
  set ipv6 next-hop 2001:DB8:2003:1::95
ipv6 local policy route-map pbr-src-90
```

## Example: show ipv6 policy Command Output

The **show ipv6 policy** command displays PBR configuration, as shown in the following example:

```
Device# show ipv6 policy

Interface                 Routemap
GigabitEthernet0/0/0      src-1
```

## Example: Verifying Route-Map Information

The following sample output from the **show route-map** command displays specific route-map information, such as a count of policy matches:

```
Device# show route-map

route-map bill, permit, sequence 10
  Match clauses:
  Set clauses:
  Policy routing matches:0 packets, 0 bytes
```

# Additional References for IPv6 Policy-Based Routing

### Related Documents

| Related Topic | Document Title |
|---|---|
| Cisco IOS commands | Cisco IOS Master Command List, All Releases |
| IP Routing Protocol-Independent commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples | Cisco IOS IP Routing: Protocol-Independent Command Reference |

### MIBs

| MIBs | MIBs Link |
|---|---|
| No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature. | To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL:<br>http://www.cisco.com/go/mibs |

### Technical Assistance

| Description | Link |
|---|---|
| The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password. | http://www.cisco.com/cisco/web/support/index.html |

# Feature Information for IPv6 Policy-Based Routing

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

*Table 5: Feature Information for IPv6 Policy-Based Routing*

| Feature Name | Releases | Feature Information |
|---|---|---|
| IPv6 Policy-Based Routing | Cisco IOS XE 3.6E | Policy-based routing for IPv6 allows a user to manually configure how received packets should be routed.<br>In Cisco IOS XE Release 3.6E, this feature was supported on the following platforms:<br>• Catalyst 3650 Series Switches<br>• Catalyst 3850 Series Switches<br>• Catalyst 4500E Supervisor Engine 7L-E<br>• Catalyst 4500-X Series Switches<br>The following commands were introduced or modified: **debug fm ipv6 pbr**, **debug ipv6 policy**, **ipv6 local policy route-map**, **ipv6 policy route-map**, **match ipv6 address**, **match length**, **route-map**, **set default interface**, **set interface**, **set ipv6 default next-hop**, **set ipv6 next-hop (PBR)**, **set ipv6 precedence**, **set vrf**, **show fm ipv6 pbr all**, **show fm ipv6 pbr interface**, **show ipv6 policy**, and **show route-map**. |