IP Routing Protocol-Independent Commands_ A through R

accept-lifetime

To set the time period during which the authentication key on a key chain is received as valid, use the **accept-lifetime**command inkey chain key configuration mode. To revert to the default value, use the **no** form of this command.

accept-lifetime commandaccept-lifetime *start-time* {**infinite**| *end-time*| **duration** *seconds*}

no accept-lifetime [*start-time* {**infinite**| *end-time*| **duration** *seconds*}]

Syntax Description

start-time	Beginning time that the key specified by the **key** command is valid to be received. The syntax can be either of the following: *hh* **:** *mm* **:** *ss Month date year* *hh* **:** *mm* **:** *ss date Month year* • *hh* --hours • *mm* --minutes • *ss*-- s econds • *Month*-- first three letters of the month • *date*-- date (1-31) • *year*-- y ear (four digits) The default start time and the earliest acceptable date is January 1, 1993.
infinite	Key is valid to be received from the *start-time* value on.
end-time	Key is valid to be received from the *start-time* value until the *end-time*value. The syntax is the same as that for the *start-time*value. The *end-time* value must be after the *start-time*value. The default end time is an infinite time period.
duration *seconds*	Length of time (in seconds) that the key is valid to be received. The range is from 1 to 2147483646.

Command Default

The authentication key on a key chain is received as valid forever (the starting time is January 1, 1993, and the ending time is infinite).

Command Modes

Key chain key configuration (config-keychain-key)

Command History

Release	Modification
11.1	This command was introduced.
12.4(6)T	Support for IPv6 was added.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

Only DRP Agent, Enhanced Interior Gateway Routing Protocol (EIGRP), and Routing Information Protocol (RIP) Version 2 use key chains.

Specify a *start-time* value and one of the following values: **infinite**, *end-time*, or **duration** *seconds*.

We recommend running Network Time Protocol (NTP) or some other time synchronization method if you assign a lifetime to a key.

If the last key expires, authentication will continue and an error message will be generated. To disable authentication, you must manually delete the last valid key.

Examples

The following example configures a key chain named chain1. The key named key1 will be accepted from 1:30 p.m. to 3:30 p.m. and will be sent from 2:00 p.m. to 3:00 p.m. The key named key2 will be accepted from 2:30 p.m. to 4:30 p.m. and will be sent from 3:00 p.m. to 4:00 p.m. The overlap allows for migration of keys or a discrepancy in the set time of the router. There is a 30-minute leeway on each side to handle time differences.

```
Router(config)# interface ethernet 0
Router(config-if)# ip rip authentication key-chain chain1
Router(config-if)# ip rip authentication mode md5
!
Router(config)# router rip
Router(config-router)# network 172.19.0.0
Router(config-router)# version 2
!
Router(config)# key chain chain1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string key1
Router(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain)# key-string key2
Router(config-keychain)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Router(config-keychain)# send-lifetime 15:00:00 Jan 25 1996 duration 3600
```

The following example configures a key chain named chain1 for EIGRP address-family. The key named key1 will be accepted from 1:30 p.m. to 3:30 p.m. and be sent from 2:00 p.m. to 3:00 p.m. The key named key2 will be accepted from 2:30 p.m. to 4:30 p.m. and be sent from 3:00 p.m. to 4:00 p.m. The overlap allows for

migration of keys or a discrepancy in the set time of the router. There is a 30-minute leeway on each side to handle time differences.

```
Router(config)# router
 eigrp virtual-name
Router(config-router)# address-family ipv4 autonomous-system 4453
Router(config-router-af)# network 10.0.0.0
Router(config-router-af)# af-interface ethernet0/0
Router(config-router-af-interface)# authentication key-chain trees
Router(config-router-af-interface)# authentication mode md5
Router(config-router-af-interface)# exit
Router(config-router-af)# exit
Router(config-router)# exit
Router(config)# key chain chain1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string key1
Router(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string key2
Router(config-keychain-key)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 15:00:00 Jan 25 1996 duration 3600
```

Related Commands

Command	Description
key	Identifies an authentication key on a key chain.
key chain	Defines an authentication key-chain needed to enable authentication for routing protocols.
key-string (authentication)	Specifies the authentication string for a key.
send-lifetime	Sets the time period during which an authentication key on a key chain is valid to be sent.
show key chain	Displays authentication key information.

bfd all-interfaces

To enable Bidirectional Forwarding Detection (BFD) for all interfaces participating in the routing process, use the **bfd all-interfaces** command in router configuration or address family interface configuration mode. To disable BFD for all neighbors on a single interface, use the **no** form of this command.

bfd all-interfaces

no bfd all-interfaces

Syntax Description

This command has no arguments or keywords.

Command Default

BFD is disabled on the interfaces participating in the routing process.

Command Modes

Router configuration (config-router)

Address family interface configuration (config-router-af)

Command History

Release	Modification
12.2(18)SXE	This command was introduced.
12.0(31)S	This command was integrated into Cisco IOS Release 12.0(31)S.
12.4(4)T	This command was integrated into Cisco IOS Release 12.4(4)T.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS Release XE 2.1 and implemented on the Cisco ASR 1000 Series Aggregation Services Routers.
12.2(33)SRE	This command was modified. Support for IPv6 was added.
15.0(1)M	This command was modified. The **bfd all-interfaces** command in named router configuration mode was replaced by the **bfd** command in address family interface mode.
15.1(2)T	This command was modified. Support for IPv6 was added.
Cisco IOS XE Release 3.3	This command was modified. Support for the Routing Information Protocol (RIP) was added.
15.2(4)S	This command was modified. Support for IPv6 was added.
Cisco IOS XE Release 3.7S	This command was modified. Support for IPv6 was added.

Usage Guidelines

There are two methods to configure routing protocols to use BFD for failure detection. To enable BFD for all interfaces, enter the **bfd all-interfaces** command in router configuration mode. In Cisco IOS Release 12.4(24)T, Cisco IOS 12.2(33)SRA, and earlier releases, the **bfd all-interfaces** command works in router configuration mode and address family interface mode.

In Cisco IOS Release 15.0(1)M and later releases, the **bfd all-interfaces** command in named router configuration mode is replaced by the **bfd** command in address family interface configuration mode. Use the **bfd** command in address family interface configuration mode to achieve the same functionality as that of the **bfd all-interfaces** command in router configuration mode.

Examples

The following example shows how to enable BFD for all Enhanced Interior Gateway Routing Protocol (EIGRP) neighbors:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp 123
Router(config-router)# bfd all-interfaces
Router(config-router)# end
```

The following example shows how to enable BFD for all Intermediate System-to-Intermediate System (IS-IS) neighbors:

```
Router> enable
Router# configure terminal
Router(config)# router isis tag1
Router(config-router)# bfd all-interfaces
Router(config-router)# end
```

The following example shows how to enable BFD for all Open Shortest Path First (OSPF) neighbors:

```
Router> enable
Router# configure terminal
Router(config)# router ospf 123
Router(config-router)# bfd all-interfaces
Router(config-router)# end
```

The following example shows how to enable BFD for all EIGRP neighbors, using the **bfd** command in address family interface configuration mode:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp my_eigrp
Router(config-router)# address family ipv4 autonomous-system 100
Router(config-router-af)# af-interface FastEthernet 0/0
Router(config-router-af)# bfd
```

The following example shows how to enable BFD for all Routing Information Protocol (RIP) neighbors:

```
Router> enable
Router# configure terminal
Router(config)# router rip
Router(config-router)# bfd all-interfaces
Router(config-router)# end
```

The following example shows how to enable IPv6 BFD for all IS-IS neighbors, in address family interface configuration mode:

```
Router> enable
Router# configure terminal
Router(config)# router isis
Router(config-router)# address family ipv6
```

```
Router(config-router-af)# bfd all-interfaces
Router(config-router-af)# end
```

Related Commands

Command	Description
bfd	Sets the baseline BFD session parameters on an interface.

dampening

To configure a device to automatically dampen a flapping session, use the **dampening** command in interface configuration mode. To disable automatic dampening, use the **no** form of this command.

dampening [*half-life-period reuse-threshold suppress-threshold max-suppress-time*] [*restart-penalty*]

no dampening

Syntax Description

half-life-period	(optional) Time (in seconds) after which a penalty is decreased. Once the route has been assigned a penalty, the penalty is decreased by half after the half-life period expires. The range of the half-life period is from 1 to 30 seconds. The default time is 5 seconds.
reuse-threshold	(optional) Reuse value based on the number of penalties. When the accumulated penalty decreases enough to fall below this value, the route is unsuppressed. The range of the reuse value is from 1 to 20000; the default is 1000.
suppress-threshold	(optional) Value of the accumulated penalty that triggers the router to dampen a flapping interface. A route is suppressed when its penalty exceeds this limit. The range is from 1 to 20000; the default is 2000.
max-suppress-time	(optional) Maximum time (in seconds) a route can be suppressed. The range is from 1 to 20000; the default is four times the *half-life-periodvalue*. If the *half-life*-period value is allowed to default, the maximum suppress time defaults to 20 seconds.
restart-penalty	(optional) Penalty to applied to the interface when it comes up for the first time after the router reloads. The configurable range is from 1 to 18000 penalties. The default is 2000 penalties. This argument is not required for any other configurations.

Command Default

This command is disabled by default. To manually configure the timer for the restart-penalty argument, the value for all arguments must be manually entered.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.0(22)S	This command was introduced.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.
12.2(18)SXD	This command was integrated into Cisco IOS Release 12.2(18)SXD.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(31)SB2	This command was integrated into Cisco IOS Release 12.2(31)SB2.

Usage Guidelines

The IP Event Dampening feature will function on a subinterface but cannot be configured on only the subinterface. Only the primary interface can be configured with this feature. Primary interface configuration is applied to all subinterfaces by default.

When an interface is dampened, the interface is dampened to both IP and Connectionless Network Services (CLNS) routing equally. The interface is dampened to both IP and CLNS because integrated routing protocols such as Intermediate System-to-Intermediate System (IS-IS), IP, and CLNS routing protocols are closely interconnected, so it is impossible to apply dampening separately.

Copying a dampening configuration from virtual templates to virtual access interfaces is not supported because dampening has limited usefulness to existing applications using virtual templates. Virtual access interfaces are released when an interface flaps, and new connections and virtual access interfaces are acquired when the interface comes up and is made available to the network. Because dampening states are attached to the interface, the dampening states would not survive an interface flap.

If the **dampening**command is applied to an interface that already has dampening configured, all dampening states are reset and the accumulated penalty will be set to 0. If the interface has been dampened, the accumulated penalty will fall into the reuse threshold range, and the dampened interface will be made available to the network. The flap counts, however, are retained.

Examples

The following example sets the half life to 30 seconds, the reuse threshold to 1500, the suppress threshold to 10000, and the maximum suppress time to 120 seconds:

```
interface Ethernet 0/0
 dampening 30 1500 10000 120
```

The following example configures the router to apply a penalty of 500 on Ethernet interface 0/0 when the interface comes up for the first time after the router is reloaded:

```
interface Ethernet 0/0
 dampening 5 500 1000 20 500
```

Related Commands

Command	Description
clear counters	Clears the interface counters.

Command	Description
show dampening interface	Displays a summary of interface dampening.
show interface dampening	Displays a summary of the dampening parameters and status.

ip policy route-map

To identify a route map to use for policy routing on an interface, use the **ip policy route-map** command in interface configuration mode. Todisable policy routing on the interface, use the **no**form of this command.

ip policy route-map *map-tag*

no ip policy route-map

Syntax Description

map-tag	Name of the route map to use for policy routing. The name must match a *map-tag* value specified by a **route-map** command.

Command Default

No policy routing occurs on the interface.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
11.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
Cisco IOS XE Release 2.2	This command was integrated into Cisco IOS Release 12.2(33)SRA.

Usage Guidelines

You might enable policy routing if you want your packets to take a route other than the obvious shortest path.

The **ip policy route-map** command identifies a route map to use for policy routing. Each**route-map**command has a list of **match** and **set** commands associated with it. The **match** commands specify the *matchc riteria*--the conditions under which policy routing is allowed for the interface, based on the destination IP address of the packet. The **set** commands specify the *setactions*--the particular policy routing actions to perform if the criteria enforced by the **match** commands are met. The **no ip policy route-map** command deletes the pointer to the route map.

Policy routing can be performed on any match criteria that can be defined in an extended IP access list when using the**match ip address**command and referencing an extended IP access list.

The policy route map needs to reconfigured in an interface in the following scenarios:

- When a policy route map is applied to an interface with VRF configuration, the route map is removed and this information is sent to the CEF.
- When an interface is configured with a policy route map and VRF, the route map is removed whenever the VRF value changes.

Examples

The following example sends packets with the destination IP address of 172.21.16.18 to a router at IP address 172.30.3.20:

```
interface serial 0
 ip policy route-map wethersfield
!
route-map wethersfield
 match ip address 172.21.16.18
 set ip next-hop 172.30.3.20
```

Related Commands

Command	Description
match ip address	Distributes any routes that have a destination network number address that is permitted by a standard or extended access list, and performs policy routing on packets.
match length	Bases policy routing on the Level 3 length of a packet.
route-map (IP)	Defines the conditions for redistributing routes from one routing protocol into another, or enables policy routing.
set default interface	Indicates where to output packets that pass a match clause of a route map for policy routing and have no explicit route to the destination.
set interface	Indicates where to output packets that pass a match clause of route map for policy routing.
set ip default next-hop	Indicates where to output packets that pass a match clause of a route map for policy routing and for which the Cisco IOS software has no explicit route to a destination.
set ip next-hop	Indicates where to output packets that pass a match clause of a route map for policy routing.

key

To identify an authentication key on a key chain, use the **key** command in key-chain configuration mode. To remove the key from the key chain, use the **no** form of this command.

key commandkey *key-id*

no key *key-id*

Syntax Description

key-id	Identification 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.

Command Default

No key exists on the key chain.

Command Modes

Key-chain configuration (config-keychain)

Command History

Release	Modification
11.1	This command was introduced.
12.4(6)T	Support for IPv6 was added.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

Only DRP Agent, Enhanced Interior Gateway Routing Protocol (EIGRP), and Routing Information Protocol (RIP) Version 2 use key chains.

It is useful to have multiple keys on a key chain so that the software can sequence through the keys as they become invalid after time, based on the **accept-lifetime** and **send-lifetime** key chain key command settings.

Each key has its own key identifier, which is stored locally. The combination of the key identifier and the interface associated with the message uniquely identifies the authentication algorithm and Message Digest 5 (MD5) authentication key in use. Only one authentication packet is sent, regardless of the number of valid keys. The software starts looking at the lowest key identifier number and uses the first valid key.

If the last key expires, authentication will continue and an error message will be generated. To disable authentication, you must manually delete the last valid key.

To remove all keys, remove the key chain by using the **no key chain** command.

Examples

The following example configures a key chain named chain1. The key named key1 will be accepted from 1:30 p.m. to 3:30 p.m. and be sent from 2:00 p.m. to 3:00 p.m. The key named key2 will be accepted from 2:30 p.m. to 4:30 p.m. and be sent from 3:00 p.m. to 4:00 p.m. The overlap allows for migration of keys or a discrepancy in the set time of the router. There is a 30-minute leeway on each side to handle time differences.

```
Router(config)# interface ethernet 0
Router(config-if)# ip rip authentication key-chain chain1
Router(config-if)# ip rip authentication mode md5
!
Router(config)# router rip
Router(config-router)# network 172.19.0.0
Router(config-router)# version 2
!
Router(config)# key chain chain1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string key1
Router(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string key2
Router(config-keychain-key)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 15:00:00 Jan 25 1996 duration 3600
```

The following named configuration example configures a key chain named chain1 for EIGRP address-family. The key named key1 will be accepted from 1:30 p.m. to 3:30 p.m. and be sent from 2:00 p.m. to 3:00 p.m. The key named key2 will be accepted from 2:30 p.m. to 4:30 p.m. and be sent from 3:00 p.m. to 4:00 p.m. The overlap allows for migration of keys or a discrepancy in the set time of the router. There is a 30-minute leeway on each side to handle time differences.

```
Router(config)# router
eigrp virtual-name
Router(config-router)# address-family ipv4 autonomous-system 4453
Router(config-router-af)# network 10.0.0.0
Router(config-router-af)# af-interface ethernet0/0
Router(config-router-af-interface)# authentication key-chain trees
Router(config-router-af-interface)# authentication mode md5
Router(config-router-af-interface)# exit
Router(config-router-af)# exit
Router(config-router)# exit
Router(config)# key chain chain1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string key1
Router(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string key2
Router(config-keychain-key)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 15:00:00 Jan 25 1996 duration 3600
```

The following named configuration example configures a key chain named chain1 for EIGRP service-family. The key named key1 will be accepted from 1:30 p.m. to 3:30 p.m. and be sent from 2:00 p.m. to 3:00 p.m. The key named key2 will be accepted from 2:30 p.m. to 4:30 p.m. and be sent from 3:00 p.m. to 4:00 p.m. The overlap allows for migration of keys or a discrepancy in the set time of the router. There is a 30-minute leeway on each side to handle time differences.

```
Router(config)# eigrp virtual-name
Router(config-router)# service-family ipv4 autonomous-system 4453
Router(config-router-sf)# network 10.0.0.0
Router(config-router-sf)# sf-interface ethernet0/0
Router(config-router-sf-interface)# authentication key-chain trees
```

```
Router(config-router-sf-interface)# authentication mode md5
Router(config-router-sf-interface)# exit
Router(config-router-sf)# exit
Router(config-router)# exit
Router(config)# key chain chain1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string key1
Router(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string key2
Router(config-keychain-key)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 15:00:00 Jan 25 1996 duration 3600
```

Related Commands

Command	Description
accept-lifetime	Sets the time period during which the authentication key on a key chain is received as valid.
ip authentication key-chain eigrp	Enables authentication of EIGRP packets.
key chain	Defines an authentication key chain needed to enable authentication for routing protocols.
key-string (authentication)	Specifies the authentication string for a key.
send-lifetime	Sets the time period during which an authentication key on a key chain is valid to be sent.
show key chain	Displays authentication key information.

key chain

To define an authentication key chain needed to enable authentication for routing protocols and enter key-chain configuration mode, use the **key chain** command in global configuration mode. To remove the key chain, use the **no** form of this command.

key chain commandkey chain *name-of-chain*

no key chain *name-of-chain*

Syntax Description

name-of-chain	Name of a key chain. A key chain must have at least one key and can have up to 2147483647 keys.

Command Default

No key chain exists.

Command Modes

Global configuration (config)

Command History

Release	Modification
11.1	This command was introduced.
12.4(6)T	Support for IPv6 was added.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

Only DRP Agent, Enhanced Interior Gateway Routing Protocol (EIGRP), and Routing Information Protocol (RIP) Version 2 use key chains.

You must configure a key chain with keys to enable authentication.

Although you can identify multiple key chains, we recommend using one key chain per interface per routing protocol. Upon specifying the **key chain** command, you enter key chain configuration mode.

Examples

The following example configures a key chain named chain1. The key named key1 will be accepted from 1:30 p.m. to 3:30 p.m. and be sent from 2:00 p.m. to 3:00 p.m. The key named key2 will be accepted from

2:30 p.m. to 4:30 p.m. and be sent from 3:00 p.m. to 4:00 p.m. The overlap allows for migration of keys or a discrepancy in the set time of the router. There is a 30-minute leeway on each side to handle time differences.

```
Router(config)# interface ethernet 0
Router(config-if)# ip rip authentication key-chain chain1
Router(config-if)# ip rip authentication mode md5
!
Router(config)# router rip
Router(config-router)# network 172.19.0.0
Router(config-router)# version 2
!
Router(config)# key chain chain1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string key1
Router(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string key2
Router(config-keychain-key)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 15:00:00 Jan 25 1996 duration 3600
```

The following named configuration example configures a key chain named chain1 for EIGRP address-family. The key named key1 will be accepted from 1:30 p.m. to 3:30 p.m. and be sent from 2:00 p.m. to 3:00 p.m. The key named key2 will be accepted from 2:30 p.m. to 4:30 p.m. and be sent from 3:00 p.m. to 4:00 p.m. The overlap allows for migration of keys or a discrepancy in the set time of the router. There is a 30-minute leeway on each side to handle time differences.

```
Router(config)# router eigrp virtual-name
Router(config-router)# address-family ipv4 autonomous-system 4453
Router(config-router-af)# network 10.0.0.0
Router(config-router-af)# af-interface ethernet0/0
Router(config-router-af-interface)# authentication key-chain trees
Router(config-router-af-interface)# authentication mode md5
Router(config-router-af-interface)# exit
Router(config-router-af)# exit
Router(config-router)# exit
Router(config)# key chain chain1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string key1
Router(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string key2
Router(config-keychain-key)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 15:00:00 Jan 25 1996 duration 3600
```

The following named configuration example configures a key chain named trees for service-family. The key named chestnut will be accepted from 1:30 pm to 3:30 pm and be sent from 2:00 pm to 3:00 pm. The key birch will be accepted from 2:30 pm to 4:30 pm and be sent from 3:00 pm to 4:00 pm. The overlap allows for migration of keys or a discrepancy in the set time of the router. There is a 30-minute leeway on each side to handle time differences.

```
Router(config)# router eigrp virtual-name
Router(config-router)# service-family ipv4 autonomous-system 4453
Router(config-router-sf)# sf-interface ethernet
Router(config-router-sf-interface)# authentication key chain trees
Router(config-router-sf-interface)# authentication mode md5
Router(config-router-sf-interface)# exit
Router(config-router-sf)# exit
Router(config-router)# exit
Router(config)# key chain chain1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string chestnut
Router(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
```

```
Router(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string birch
Router(config-keychain-key)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 15:00:00 Jan 25 1996 duration 3600
```

Related Commands

Command	Description
accept-lifetime	Sets the time period during which the authentication key on a key chain is received as valid.
ip rip authentication key-chain	Enables authentication for RIP Version 2 packets and specifies the set of keys that can be used on an interface.
ip authentication key-chain eigrp	Enables authentication of EIGRP packets.
key	Identifies an authentication key on a key chain.
key-string (authentication)	Specifies the authentication string for a key.
send-lifetime	Sets the time period during which an authentication key on a key chain is valid to be sent.
show key chain	Displays authentication key information.

key-string (authentication)

To specify the authentication string for a key, use the **key-string**(authentication) command in key chain key configuration mode. To remove the authentication string, use the **no** form of this command.

key-string commandkey-string *text*

no key-string *text*

Syntax Description

text	Authentication string that must be sent and received in the packets using the routing protocol being authenticated. The string can contain from 1 to 80 uppercase and lowercase alphanumeric characters.

Command Default

No authentication string for a key exists.

Command Modes

Key chain key configuration (config-keychain-key)

Command History

Release	Modification
11.1	This command was introduced.
12.4(6)T	Support for IPv6 was added.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

Only DRP Agent, Enhanced Interior Gateway Routing Protocol (EIGRP), and Routing Information Protocol (RIP) Version 2 use key chains. Each key can have only one key string.

If password encryption is configured (with the **service password-encryption** command), the software saves the key string as encrypted text. When you write to the terminal with the **more system:running-config** command, the software displays key-string 7 encrypted text.

Examples

The following example configures a key chain named chain1. The key named key1 will be accepted from 1:30 p.m. to 3:30 p.m. and be sent from 2:00 p.m. to 3:00 p.m. The key named key2 will be accepted from

2:30 p.m. to 4:30 p.m. and be sent from 3:00 p.m. to 4:00 p.m. The overlap allows for migration of keys or a discrepancy in the set time of the router. There is a 30-minute leeway on each side to handle time differences.

```
Router(config)# interface ethernet 0
Router(config-if)# ip rip authentication key-chain chain1
Router(config-if)# ip rip authentication mode md5
!
Router(config)# router rip
Router(config-router)# network 172.19.0.0
Router(config-router)# version 2
!
Router(config)# key chain chain1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string key1
Router(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string key2
Router(config-keychain-key)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 15:00:00 Jan 25 1996 duration 3600
```

The following example configures a key chain named chain1 for EIGRP address-family. The key named key1 will be accepted from 1:30 p.m. to 3:30 p.m. and be sent from 2:00 p.m. to 3:00 p.m. The key named key2 will be accepted from 2:30 p.m. to 4:30 p.m. and be sent from 3:00 p.m. to 4:00 p.m. The overlap allows for migration of keys or a discrepancy in the set time of the router. There is a 30-minute leeway on each side to handle time differences.

```
Router(config)# eigrp virtual-name
Router(config-router)# address-family ipv4 autonomous-system 4453
Router(config-router-af)# network 10.0.0.0
Router(config-router-af)# af-interface ethernet0/0
Router(config-router-af-interface)# authentication key-chain trees
Router(config-router-af-interface)# authentication mode md5
Router(config-router-af-interface)# exit
Router(config-router-af)# exit
Router(config-router)# exit
Router(config)# key chain chain1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string key1
Router(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string key2
Router(config-keychain-key)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 15:00:00 Jan 25 1996 duration 3600
```

Related Commands

Command	Description
accept-lifetime	Sets the time period during which the authentication key on a key chain is received as valid.
ip authentication key-chain eigrp	Enables authentication of EIGRP packets.
key	Identifies an authentication key on a key chain.
key chain	Defines an authentication key-chain needed to enable authentication for routing protocols.

Command	Description
send-lifetime	Sets the time period during which an authentication key on a key chain is valid to be sent.
service password-encryption	Encrypts passwords.
show key chain	Displays authentication key information.

match interface (IP)

To distribute any routes that have their next hop out one of the interfaces specified, use the **matchinterface** command in route-map configuration mode. To remove the **matchinterface** entry, use the **no** form of this command.

match interface *interface-type interface-number* [*... interface-type interface-number*]

no match interface *interface-type interface-number* [*... interface-type interface-number*]

Syntax Description

interface- type	Interface type.
interface- number	Interface number.

Command Default

No match interfaces are defined.

Command Modes

Route-map configuration (config-route-map)

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
15.1(2)S	This command was integrated into Cisco IOS Release 15.1(2)S.

Usage Guidelines

An ellipsis (...) in the command syntax indicates that your command input can include multiple values for the *interface-typeinterface-number* arguments .

Use the **route-map** global configuration command, and the **match** and **set** route-map configuration commands, to define the conditions for redistributing routes from one routing protocol into another. Each **route-map** command has a list of **match** and **set** commands associated with it. The **match** commands specify the *matchcriteria*--the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the *setactions*--the particular redistribution actions to perform if the criteria enforced by the **match** commands are met. The **noroute-map** command deletes the route map.

The **match** route-map configuration command has multiple formats. The **match** commands may be given in any order, and all **match** commands must “pass” to cause the route to be redistributed according to the

*setactions*given with the **set** commands. The **no** forms of the **match** commands remove the specified match criteria.

A route map can have several parts. Any route that does not match at least one **match** clause relating to a **route-map** command will be ignored; that is, the route will not be advertised for outbound route maps and will not be accepted for inbound route maps. If you want to modify only some data, you must configure a second route map section with an explicit match specified.

Examples

In the following example, routes that have their next hop out Ethernet interface 0 will be distributed:

```
route-map name
 match interface ethernet 0
```

Related Commands

Command	Description
match as-path	Matches a BGP autonomous system path access list.
match community	Matches a BGP community.
match ip address	Distributes any routes that have a destination network number address that is permitted by a standard or extended access list, and performs policy routing on packets.
match ip next-hop	Redistributes any routes that have a next hop router address passed by one of the access lists specified.
match ip route-source	Redistributes routes that have been advertised by routers and access servers at the address specified by the access lists.
match metric (IP)	Redistributes routes with the metric specified.
match route-type (IP)	Redistributes routes of the specified type.
match tag	Redistributes routes in the routing table that match the specified tags.
route-map (IP)	Defines the conditions for redistributing routes from one routing protocol into another, or enables policy routing.
set as-path	Modifies an autonomous system path for BGP routes.
set automatic-tag	Automatically computes the tag value.
set community	Sets the BGP communities attribute.
set level (IP)	Indicates where to import routes.

Command	Description
set local-preference	Specifies a preference value for the autonomous system path.
set metric (BGP, OSPF, RIP)	Sets the metric value for a routing protocol.
set metric-type	Sets the metric type for the destination routing protocol.
set next-hop	Specifies the address of the next hop.
set tag (IP)	Sets a tag value of the destination routing protocol.
set weight	Specifies the BGP weight for the routing table.

match ip address

To distribute any routes that have a destination network number address that is permitted by a standard access list, an extended access list, or a prefix list, or to perform policy routing on packets, use the **matchipaddress** command in route-map configuration mode. To remove the **matchipaddress** entry, use the **no** form of this command.

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 ...*]}

no 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 ...*]}

Syntax Description

access-list-number...	Number of a standard or extended access list. It can be an integer from 1 to 199. The ellipsis indicates that multiple values can be entered.
access-list-name...	Name of a standard or extended access list. It can be an integer from 1 to 199. The ellipsis indicates that multiple values can be entered.
prefix-list	Distributes routes based on a prefix list.
prefix-list-name...	Name of a specific prefix list. The ellipsis indicates that multiple values can be entered.

Command Default

No access list numbers or prefix lists are specified.

Command Modes

Route-map configuration (config-route-map)

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
15.1(2)S	This command was integrated into Cisco IOS Release 15.1(2)S.

Usage Guidelines

An ellipsis (...) in the command syntax indicates that your command input can include multiple values for the *access-list-number,access-list-name,orprefix-list-name*arguments .

Like matches in the same route map subblock are filtered with “or” semantics. If any one match clause is matched in the entire route map subblock, this match is treated as a successful match. Dissimilar match clauses are filtered with “and” semantics. So dissimilar matches are filtered logically. If the first set of conditions is not met, the second match clause is filtered. This process continues until a match occurs or there are no more match clauses.

Use route maps to redistribute routes or to subject packets to policy routing. Both purposes are described in this section.

Redistribution

Use the **route-map** global configuration command, and the **match** and **set** route-map configuration commands, to define the conditions for redistributing routes from one routing protocol into another. Each **route-map** command has a list of **match** and **set** commands associated with it. The **match** commands specify the *matchcriteria*--the conditions under which redistribution is allowed for the current **route-map**command. The**set** commands specify the *setactions*--the particular redistribution actions to perform if the criteria enforced by the **match** commands are met. The **noroute-map** command deletes the route map.

The **match** route-map configuration command has multiple formats. The **match** commands can be given in any order, and all **match** commands must “pass” to cause the route to be redistributed according to the *setactions*given with the **set** commands. The **no** forms of the **match** commands remove the specified match criteria.

When you are passing routes through a route map, a route map can have several sections that contain specific **match** clauses. Any route that does not match at least one **match** clause relating to a **route-map** command will be ignored; that is, the route will not be advertised for outbound route maps and will not be accepted for inbound route maps. If you want to modify only some data, you must configure a second route map section with an explicit match specified.

Policy Routing

Another purpose of route maps is to enable policy routing. The match ip address command allows you to policy route packets based on criteria that can be matched with an extended access list; for example, a protocol, protocol service, and source or destination IP address. To define the conditions for policy routing packets, use the **ippolicyroute-map** interface configuration command, in addition to the **route-map** global configuration command, and the **match** and **set** route-map configuration commands. Each **route-map** command has a list of **match** and **set** commands associated with it. The **match** commands specify the *matchcriteria*--the conditions under which policy routing occurs. The **set** commands specify the *setactions*--the particular routing actions to perform if the criteria enforced by the **match**commands are met. You might want to policy route packets based on their source, for example, using an access list.

Examples

In the following example, routes that have addresses specified by access list numbers 5 or 80 will be matched:

```
Router(config)# route-map name
Router(config-route-map)#  match ip address 5 80
```

Route maps that use prefix lists can be used for route filtering, default origination, and redistribution in other routing protocols. In the following example, a default route 0.0.0.0/0 is conditionally originated when there exists a prefix 10.1.1.0/24 in the routing table:

```
Router(config)# ip prefix-list cond permit 10.1.1.0/24
!
```

```
Router(config)# route-map default-condition permit 10
Router(config-route-map)# match ip address prefix-list cond
!
Router(config)# router rip
Router(config-router)# default-information originate route-map default-condition
```

In the following policy routing example, packets that have addresses specified by access list numbers 6 or 25 will be routed to Ethernet interface 0:

```
Router(config)# interface serial 0
Router(config-if)# ip policy route-map chicago
!
Router(config)# route-map chicago
Router(config-route-map)# match ip address 6 25
Router(config-route-map)# set interface ethernet 0
```

Related Commands

Command	Description
ip local policy route-map	Identifies a route map to use for policy routing on an interface.
ip policy route-map	Identifies a route map to use for policy routing on an interface.
match as-path	Matches a BGP autonomous system path access list.
match community	Matches a BGP community.
match interface (IP)	Distributes any routes that have their next hop out one of the interfaces specified.
match ip next-hop	Redistributes any routes that have a next hop router address passed by one of the access lists specified.
match ip route-source	Redistributes routes that have been advertised by routers and access servers at the address specified by the access lists.
match length	Bases policy routing on the Level 3 length of a packet.
match metric (IP)	Redistributes routes with the metric specified.
match route-type (IP)	Redistributes routes of the specified type.
match tag	Redistributes routes in the routing table that match the specified tags.
route-map (IP)	Defines the conditions for redistributing routes from one routing protocol into another, or enables policy routing.
set as-path	Modifies an autonomous system path for BGP routes.
set automatic-tag	Automatically computes the tag value.

Command	Description
set community	Sets the BGP communities attribute.
set default interface	Indicates where to output packets that pass a match clause of a route map for policy routing and have no explicit route to the destination.
set interface	Indicates where to output packets that pass a match clause of a route map for policy routing.
set ip default next-hop	Indicates where to output packets that pass a match clause of a route map for policy routing and for which the Cisco IOS software has no explicit route to a destination.
set ip next-hop	Indicates where to output packets that pass a match clause of a route map for policy routing.
set level (IP)	Indicates where to import routes.
set local-preference	Specifies a preference value for the autonomous system path.
set metric (BGP, OSPF, RIP)	Sets the metric value for a routing protocol.
set metric-type	Sets the metric type for the destination routing protocol.
set next-hop	Specifies the address of the next hop.
set tag (IP)	Sets a tag value of the destination routing protocol.
set weight	Specifies the BGP weight for the routing table.

match ip next-hop

To redistribute any routes that have a next hop router address passed by one of the access lists specified, use the **matchipnext-hop** command in route-map configuration mode. To remove the next hop entry, use the **no** form of this command.

match ip next-hop {*access-list-number*| *access-list-name*} [*... access-list-number*| *... access-list-name*]

no match ip next-hop {*access-list-number*| *access-list-name*} [*... access-list-number*| *... access-list-name*]

Syntax Description

access-list-number access-list-name	Number or name of a standard or extended access list. It can be an integer from 1 to 199.

Command Default

Routes are distributed freely, without being required to match a next hop address.

Command Modes

Route-map configuration (config-route-map)

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
15.1(2)S	This command was integrated into Cisco IOS Release 15.1(2)S.

Usage Guidelines

An ellipsis (...) in the command syntax indicates that your command input can include multiple values for the *access-list-number*or*access-list-name* argument .

Use the route-map global configuration command, and the **match** and **set** route-map configuration commands, to define the conditions for redistributing routes from one routing protocol into another. Each **route-map** command has a list of **match** and **set** commands associated with it. The **match** commands specify the *matchcriteria*--the conditions under which redistribution is allowed for the current route-mapcommand. The **set** commands specify the *setactions*--the particular redistribution actions to perform if the criteria enforced by the **match** commands are met. The **noroute-map** command deletes the route map.

The **match** route-map configuration command has multiple formats. The **match** commands can be given in any order, and all **match** commands must “pass” to cause the route to be redistributed according to the *setactions*given with the **set** commands. The **no** forms of the **match** commands remove the specified match criteria.

When you are passing routes through a route map, a route map can have several parts. Any route that does not match at least one **match** clause relating to a **route-map** command will be ignored; that is, the route will not be advertised for outbound route maps and will not be accepted for inbound route maps. If you want to modify only some data, you must configure a second route map section with an explicit match specified.

Examples

The following example distributes routes that have a next hop router address passed by access list 5 or 80 will be distributed:

```
Router(config)# route-map name
Router(config-route-map)# match ip next-hop 5 80
```

Related Commands

Command	Description
match as-path	Matches a BGP autonomous system path access list.
match community	Matches a BGP community.
match interface (IP)	Distributes any routes that have their next hop out one of the interfaces specified.
match ip address	Distributes any routes that have a destination network number address that is permitted by a standard or extended access list, and performs policy routing on packets.
match ip route-source	Redistributes routes that have been advertised by routers and access servers at the address specified by the access lists.
match metric (IP)	Redistributes routes with the metric specified.
match route-type (IP)	Redistributes routes of the specified type.
match tag	Redistributes routes in the routing table that match the specified tags.
route-map (IP)	Defines the conditions for redistributing routes from one routing protocol into another, or enables policy routing.
set as-path	Modifies an autonomous system path for BGP routes.
set automatic-tag	Automatically computes the tag value.
set community	Sets the BGP communities attribute.
set level (IP)	Indicates where to import routes.

Command	Description
set local-preference	Specifies a preference value for the autonomous system path.
set metric (BGP, OSPF, RIP)	Sets the metric value for a routing protocol.
set metric-type	Sets the metric type for the destination routing protocol.
set next-hop	Specifies the address of the next hop.
set tag (IP)	Sets a tag value of the destination routing protocol.
set weight	Specifies the BGP weight for the routing table.

match ip route-source

To match routes that have been advertised by routers and access servers at the address specified by the access lists, use the **matchiproute-source** command in route-map configuration mode. To remove the route-source entry, use the **no** form of this command.

match ip route-source [*access-list-number* [*... access-list-number*]] [*expanded-access-list* [*... expanded-access-list*]] [*access-list-name* [*... access-list-name*]] [**prefix-list** *name* [*...***prefix-list** *name*]] **[redistribution-source]**

no match ip route-source [*access-list-number* [*... access-list-number*]] [*expanded-access-list* [*... expanded-access-list*]] [*access-list-name* [*... access-list-name*]] [**prefix-list** *name* [*...***prefix-list** *name*]] **[redistribution-source]**

Syntax Description

access-list-number	(Optional) Number of a standard access list. The range is from 1 to 199.
expanded-access-list	(Optional) Number of an expanded access list. The range is from 1300 to 1999.
access-list-name	(Optional) Name of a standard access list.
prefix-list *name*	(Optional) Configures the match entries of a specified prefix list.
redistribution-source	(Optional) Specifies the route redistribution source for Enhanced Interior Gateway Routing Protocol (EIGRP).

Command Default

No filtering of the routes is applied on the route source.

Command Modes

Route-map configuration (config-route-map)

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Release	Modification
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 2.1 and implemented on Cisco ASR 1000 Series Aggregation Services Routers.

Usage Guidelines

An ellipsis (...) in the command syntax indicates that your command input can include multiple values for the *access-list-number* argument, the *expanded-access-list* argument, the *access-list-name* argument, and the **prefix-list***name* keyword and argument pair.

Use the **route-map** global configuration command, and the **match** and **set** route-map configuration commands, to define the conditions for redistributing routes from one routing protocol into another. Each **route-map** command has a list of **match** and **set** commands associated with it. The **match** commands specify the match criteria--the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the set actions--the particular redistribution actions to perform if the criteria enforced by the **match** commands are met. The **no route-map** command deletes the route map.

The **match** route-map configuration command has multiple formats. The **match** commands can be given in any order, and all **match** commands must "pass" to cause the route to be redistributed according to the set actions given with the **set** commands. The **no** forms of the **match** commands remove the specified match criteria.

A route map can have several parts. Any route that does not match at least one **match** clause relating to a **route-map** command will be ignored; that is, the route will not be advertised for outbound route maps and will not be accepted for inbound route maps. If you want to modify only some data, you must configure the second route map section with an explicit match specified.

Examples

The following example shows how to match routes that are advertised by routers and access servers at the address specified by access list 5 and expanded access list 1335:

```
Router(config)# route-map R1
Router(config-route-map)# match ip route-source 5 1335
```

Related Commands

Command	Description
match as-path	Matches a BGP autonomous system path access list.
match community	Matches a BGP community.
match interface (IP)	Distributes any routes that have their next hop from one of the interfaces specified.
match ip address	Distributes any routes that have a destination network number address that is permitted by a standard or extended access list, and performs policy routing on packets.
match ip redistribution-source	Filters the external EIGRP routes that have been advertised by routers and access servers at the address specified by the access lists.

Command	Description
match ip next-hop	Redistributes any routes that have a next hop router address passed by one of the access lists specified.
match metric (IP)	Redistributes routes with the metric specified.
match route-type (IP)	Redistributes routes of the specified type.
match tag	Redistributes routes in the routing table that match the specified tags.
route-map (IP)	Defines the conditions for redistributing routes from one routing protocol into another, or enables policy routing.
set as-path	Modifies an autonomous system path for BGP routes.
set automatic-tag	Automatically computes the tag value.
set community	Sets the BGP communities attribute.
set level (IP)	Indicates where to import routes.
set local-preference	Specifies a preference value for the autonomous system path.
set metric (BGP, OSPF, RIP)	Sets the metric value for a routing protocol.
set metric-type	Sets the metric type for the destination routing protocol.
set next-hop	Specifies the address of the next hop.
set tag (IP)	Sets a tag value for the destination routing protocol.
set weight	Specifies the BGP weight for the routing table.

match ipv6 address

To distribute IPv6 routes that have a prefix permitted by a prefix list or to specify an IPv6 access list to be used to match packets for policy-based routing (PBR) for IPv6, use the **match ipv6 address** command in route-map configuration mode. To remove the **match ipv6 address** entry, use the **no** form of this command.

match ipv6 address {**prefix-list** *prefix-list-name*| *access-list-name*}

no match ipv6 address

Syntax Description

prefix-list *prefix-list-name*	Specifies the name of an IPv6 prefix list.
access-list-name	Name of the IPv6 access list. Names cannot contain a space or quotation mark or begin with a numeric.

Command Default

No routes are distributed based on the destination network number or an access list.

Command Modes

Route-map configuration (config-route-map)

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.3(7)T	This command was modified. The *access-list-name* argument was added.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SXI4	This command was modified. The **prefix-list** *prefix-list-name* keyword-argument pair argument is not supported in Cisco IOS Release 12.2(33)SXI4.
Cisco IOS XE Release 3.2S	This command was integrated into Cisco IOS XE Release 3.2S.
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)SY.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

Use the **route-map** command and the **match** and **set** commands to define the conditions for redistributing routes from one routing protocol to another. Each **route-map** command has a list of **match** and **set** commands associated with it. The **match** commands specify the match criteria--the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the set actions, which are the particular redistribution actions to be performed if the criteria enforced by the **match** commands are met.

The **match ipv6 address** command can be used to specify either an access list or a prefix list. When using PBR, you must use the *access-list-name* argument; the **prefix-list** *prefix-list-name* keyword-argument pair argument will not work.

Examples

In the following example, IPv6 routes that have addresses specified by the prefix list named marketing are matched:

```
Device(config)# route-map name
Device(config-route-map)# match ipv6 address prefix-list marketing
```

In the following example, IPv6 routes that have addresses specified by an access list named marketing are matched:

```
Device(config)# route-map
Device(config-route-map)# match ipv6 address marketing
```

Related Commands

Command	Description
match as-path	Matches a BGP autonomous system path access list.
match community	Matches a BGP community.
match ipv6 address	Specifies an IPv6 access list to be used to match packets for PBR for IPv6.
match ipv6 next-hop	Distributes IPv6 routes that have a next-hop prefix permitted by a prefix list.
match ipv6 route-source	Distributes IPv6 routes that have been advertised by routers at an address specified by a prefix list.
match length	Bases policy routing on the Level 3 length of a packet.
match metric	Redistributes routes with the specified metric.
match route-type	Redistributes routes of the specified type.
route-map	Defines conditions for redistributing routes from one routing protocol into another.
set as-path	Modifies an autonomous system path for BGP routes.
set community	Sets the BGP community attribute.

Command	Description
set default interface	Specifies the default interface to output packets that pass a match clause of a route map for policy routing and have no explicit route to the destination.
set interface	Specifies the default interface to output packets that pass a match clause of a route map for policy routing.
set ipv6 default next-hop	Specifies an IPv6 default next hop to which matching packets will be forwarded.
set ipv6 next-hop (PBR)	Indicates where to output IPv6 packets that pass a match clause of a route map for policy routing.
set ipv6 precedence	Sets the precedence value in the IPv6 packet header.
set level	Indicates where to import routes.
set local preference	Specifies a preference value for the autonomous system path.
set metric	Sets the metric value for a routing protocol.
set metric-type	Sets the metric type for the destination routing protocol.
set tag	Sets a tag value of the destination routing protocol.
set weight	Specifies the BGP weight for the routing table.

match length

To base policy routing on the Level 3 length of a packet, use the **match length** command in route-map configuration mode. To remove the entry, use the **no** form of this command.

match length *minimum-length maximum-length*

no match length *minimum-length maximum-length*

Syntax Description

minimum-length	Minimum Level 3 length of the packet allowed for a match. The range is from 0 to 0x7FFFFFFF.
maximum-length	Maximum Level 3 length of the packet allowed for a match. The range is from 0 to 0x7FFFFFFF.

Command Default

No policy routing occurs on the length of a packet.

Command Modes

Route-map configuration (config-route-map)

Command History

Release	Modification
10.0	This command was introduced.
12.3(7)T	This command was modified. This command was updated for use in configuring IPv6 policy-based routing (PBR).
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
Cisco IOS XE Release 3.2S	This command was integrated into Cisco IOS XE Release 3.2S.
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)SY.

Usage Guidelines

In IPv4, use the **ip policy route-map** interface configuration command, the **route-map** global configuration command, and the **match** and **set** route-map configuration commands to define the conditions for policy routing packets. The **ip policy route-map** command identifies a route map by name. Each **route-map** command has a list of **match** and **set** commands associated with it. The **match** commands specify the **match criteria—**the conditions under which policy routing occurs. The **set** commands specify the **set actions—**the particular routing actions to perform if the criteria enforced by the **match** commands are met.

In PBR for IPv6, use the **ipv6 policy route-map** or **ipv6 local policy route-map** command to define conditions for policy routing packets.

In IPv4, the **match** route-map configuration command has multiple formats. The **match** commands can be issued in any order, and all **match** commands must "pass" to cause the packet to be routed according to the **set actions** given with the **set** commands. The **no** forms of the **match** commands remove the specified match criteria.

In IPv4, you might want to base your policy routing on the length of packets so that your interactive traffic and bulk traffic are directed to different routers.

Examples

In the following example, packets 3 to 200 bytes long, inclusive, will be routed to FDDI interface 0:

```
interface serial 0
 ip policy route-map interactive
!
route-map interactive
 match length 3 200
 set interface fddi 0
```

In the following example for IPv6, packets 3 to 200 bytes long, inclusive, will be routed to FDDI interface 0:

```
interface Ethernet0/0
  ipv6 policy-route-map interactive
!
route-map interactive
 match length 3 200
 set interface fddi 0
```

Related Commands

Command	Description
ip local policy route-map	Identifies a route map to be used for policy routing on an interface.
ipv6 local policy route-map	Configures IPv6 PBR for IPv6 originated packets.
ipv6 policy route-map	Configures IPv6 PBR on an interface.
match ip address	Distributes any routes that have a destination network number address that is permitted by a standard or extended access list, and performs policy routing on packets.
match ipv6 address	Specifies an IPv6 access list to be used to match packets for IPv6 PBR.
match length	Bases policy routing on the Level 3 length of a packet.
route-map (IP)	Defines the conditions for redistributing routes from one routing protocol to another, or enables policy routing.

Command	Description
set default interface	Indicates where to output packets that pass a match clause of a route map for policy routing and have no explicit route to the destination.
set interface	Indicates where to output packets that pass a match clause of route map for policy routing.
set ip default next-hop	Indicates where to output packets that pass a match clause of a route map for policy routing and for which the Cisco software has no explicit route to a destination.
set ipv6 default next-hop	Specifies an IPv6 default next hop to which matching packets will be forwarded.
set ip next-hop	Indicates where to output packets that pass a match clause of a route map for policy routing.
set ipv6 next-hop (PBR)	Indicates where to output IPv6 packets that pass a match clause of a route map for policy routing.
set ipv6 precedence	Sets the precedence value in the IPv6 packet header.

match metric (IP)

To redistribute r outes with the specified metric, use the **matchmetric** command in route-map configuration mode. To remove the entry for the redistributed route from the routing table, use the **no** form of this command.

match metric {*metric-value*| **external** *metric-value*} [*+-deviation-number*]

no match metric {*metric-value*| **external** *metric-value*} [*+-deviation-number*]

Syntax Description

metric-value	Internal route metric, which can be an Enhanced Interior Gateway Routing Protocol (EIGRP) five-part metric. The range is from 1 to 4294967295.
external	External protocol associated with a route and interpreted by a source protocol.
+- *deviation-number*	(Optional) A standard deviation number that will offset the number configured for the metric-value argument. The deviation-number argument can be any number. There is no default. **Note** When you specify a deviation of the metric with the + and - keywords, the router will match any metric that falls inclusively in that range.

Command Default

No filtering is performed on a metric value.

Command Modes

Route-map configuration (config-route-map)

Command History

Release	Modification
11.2	This command was introduced.
12.3(8)T	The **external** and **+-**keywords and *deviation-number* argument were added.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.

Usage Guidelines

Use the **route-map** global configuration command and the **match** and **set** route-map configuration commands to define the conditions for redistributing routes from one routing protocol into another. Each **route-map**

command has a list of **match** and **set** commands associated with it. The **match** commands specify the *matchcriteria*--the conditions under which redistribution is allowed for the current **route-map**command. The **set** commands specify the *setactions*--the particular redistribution actions to perform if the criteria enforced by the **match** commands are met. The **noroute-map** command deletes the route map.

The **match** route-map configuration command has multiple formats. The **match** commands can be given in any order, and all **match** commands must "pass" to cause the route to be redistributed according to the set actionsgiven with the **set** commands. The **no** forms of the **match** commands remove the specified match criteria.

A route map can have several parts. Any route that does not match at least one **match** clause relating to a **route-map**command will be ignored; that is, the route will not be advertised for outbound route maps and will not be accepted for inbound route maps. If you want to modify only some data, you must configure a second route map section with an explicit match specified.

Note An external protocol route metric is not the same as the EIGRP assigned route metric which is a figure computed using EIGRP vectorized metric components (delay, bandwidth, reliability, load, and MTU).

Examples

In the following example, routes with the metric 5 will be redistributed:

```
Router(config)# route-map name
Router(config-route-map)# match metric 5
```

In the following example, any metric that falls inclusively in the range from 400 to 600 is matched:

```
Router(config)# route-map name
Router(config-route-map)# match metric 500 +- 100
```

The following example shows how to configure a route map to match an EIGRP external protocol metric route with an allowable deviation of 100, a source protocol of BGP, and an autonomous system 45000. When the two match clauses are true, the tag value of the destination routing protocol is set to 5. The route map is used to distribute incoming packets for an EIGRP process.

```
Router(config)# route-map metric_range
Router(config-route-map)# match metric external 500 +- 100
Router(config-route-map)# match source-protocol bgp 45000
Router(config-route-map)# set tag 5
!
Router(config)# router eigrp 45000
Router(config-router)# network 172.16.0.0
Router(config-router)# distribute-list route-map metric_range in
```

Related Commands

Command	Description
match as-path	Matches a BGP autonomous system path access list.
match community	Matches a BGP community.
match interface (IP)	Distributes any routes that have their next hop out one of the interfaces specified.

Command	Description
match ip address	Distributes any routes that have a destination network number address that is permitted by a standard or extended access list, and performs policy routing on packets.
match ip next-hop	Redistributes any routes that have a next hop router address passed by one of the access lists specified.
match ip route-source	Redistributes routes that have been advertised by routers and access servers at the address specified by the access lists.
match route-type (IP)	Redistributes routes of the specified type.
match tag	Redistributes routes in the routing table that match the specified tags.
route-map (IP)	Defines the conditions for redistributing routes from one routing protocol into another, or enables policy routing.
set as-path	Modifies an autonomous system path for BGP routes.
set automatic-tag	Automatically computes the tag value.
set community	Sets the BGP communities attribute.
set level (IP)	Indicates where to import routes.
set local-preference	Specifies a preference value for the autonomous system path.
set metric (BGP, OSPF, RIP)	Sets the metric value for a routing protocol.
set metric-type	Sets the metric type for the destination routing protocol.
set next-hop	Specifies the address of the next hop.
set tag (IP)	Sets a tag value of the destination routing protocol.

match route-type (IP)

To redistribute routes of the specified type, use the **matchroute-type** command in route-map configuration mode. To remove the route type entry, use the **no** form of this command.

match route-type {**local**| **internal**| **external** [**type-1**| **type-2**]| **level-1**| **level-2**}

no match route-type {**local**| **internal**| **external** [**type-1**| **type-2**]| **level-1**| **level-2**}

Syntax Description

local	Locally generated Border Gateway Protocol (BGP) routes.
internal	Open Shortest Path First (OSPF) intra-area and interarea routes or Enhanced Interior Gateway Routing Protocol (EIGRP) internal routes.
external [**type-1**|**type-2**	OSPF external routes, or EIGRP external routes. For OSPF, the **externaltype-1** keyword matches only Type 1 external routes and the **externaltype-2** keyword matches only Type 2 external routes.
level-1	Intermediate System-to-Intermediate System (IS-IS) Level 1 routes.
level-2	IS-IS Level 2 routes.

Command Default

This command is disabled by default.

Command Modes

Route-map configuration

Command History

Release	Modification
10.0	This command was introduced.
11.2	The **local** and **external** [**type-1** | **type-2**] keywords were added.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.

Usage Guidelines

Use the **route-map** global configuration command, and the **match** and **set** route-map configuration commands, to define the conditions for redistributing routes from one routing protocol into another. Each **route-map** command has a list of **match** and **set** commands associated with it. The **match** commands specify the *matchcriteria*--the conditions under which redistribution is allowed for the current **route-map**command. The **set** commands specify the *setactions*--the particular redistribution actions to perform if the criteria enforced by the **match** commands are met. The **noroute-map** command deletes the route map.

The **match** route-map configuration command has multiple formats. The **match** commands can be given in any order, and all **match** commands must "pass" to cause the route to be redistributed according to the *setactions*given with the **set** commands. The **no** forms of the **match** commands remove the specified match criteria.

A route map can have several parts. Any route that does not match at least one **match** clause relating to a **route-map** command will be ignored; that is, the route will not be advertised for outbound route maps and will not be accepted for inbound route maps. If you want to modify only some data, you must configure second route map section with an explicit match specified.

Examples

The following example redistributes internal routes:

```
route-map name
 match route-type internal
```

Related Commands

Command	Description
match as-path	Matches a BGP autonomous system path access list.
match community	Matches a BGP community.
match interface (IP)	Distributes any routes that have their next hop out one of the interfaces specified.
match ip address	Distributes any routes that have a destination network number address that is permitted by a standard or extended access list, and performs policy routing on packets.
match ip next-hop	Redistributes any routes that have a next hop router address passed by one of the access lists specified.
match ip route-source	Redistributes routes that have been advertised by routers and access servers at the address specified by the access lists.
match metric (IP)	Redistributes routes with the metric specified.
match tag	Redistributes routes in the routing table that match the specified tags.

Command	Description
route-map (IP)	Defines the conditions for redistributing routes from one routing protocol into another, or enables policy routing.
set as-path	Modifies an autonomous system path for BGP routes.
set automatic-tag	Automatically computes the tag value.
set community	Sets the BGP communities attribute.
set level (IP)	Indicates where to import routes.
set local-preference	Specifies a preference value for the autonomous system path.
set metric (BGP, OSPF, RIP)	Sets the metric value for a routing protocol.
set metric-type	Sets the metric type for the destination routing protocol.
set next-hop	Specifies the address of the next hop.
set tag (IP)	Sets a tag value of the destination routing protocol.
set weight	Specifies the BGP weight for the routing table.

match tag

To filter routes that match specific route tags, use the **match tag** command in route-map configuration mode. To remove the tag entry, use the **no** form of this command.

match tag {*tag-value*| *tag-value-dotted-decimal*} [*... tag-value* | *... tag-value-dotted-decimal*]

no match tag {*tag-value*| *tag-value-dotted-decimal*} [*... tag-value* | *... tag-value-dotted-decimal*]

Syntax Description

tag-value	Route tag value in plain decimals. The valid range is from 0 to 4294967295.
tag-value-dotted-decimal	Route tag value in dotted decimals. The valid range is from 0.0.0.0 to 255.255.255.255.

Command Default

No match tag values are defined.

Command Modes

Route-map configuration (config-route-map)

Command History

Release	Modification
10.0	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
Cisco IOS XE Release 2.1	This command was implemented on Cisco ASR 1000 Series Aggregation Services Routers.
15.1(2)S	This command was integrated into Cisco IOS Release 15.1(2)S.
15.2(2)S	This command was modified. The *tag-value-dotted-decimal* argument was added to support tag values in dotted-decimal format.
Cisco IOS XE Release 3.6S	This command was modified. The *tag-value-dotted-decimal* argument was added to support tag values in dotted-decimal format.

Usage Guidelines

Ellipses (...) in the command syntax indicate that your command input can include multiple values for the *tag-value* and the *tag-value-dotted-decimal* arguments.

Examples

The following example shows how to match a route with a tag value of 5:

```
Device(config)# route-map name
Device(config-route-map)# match tag 5
```

The following example shows how to match a route with a tag value of 10.10.10.10:

```
Device(config)# route-map name
Device(config-route-map)# match tag 10.10.10.10
```

Related Commands

Command	Description
match as-path	Matches a BGP autonomous system path specified by an access list.
match community	Matches a BGP community.
match ip address	Distributes any route that has a destination address that performs policy routing on packets and is permitted by a standard or extended access list.
route-map (IP)	Defines conditions for redistributing routes from one routing protocol into another, or enables policy routing.
set automatic-tag	Automatically computes the tag value.
set level (IP)	Indicates where to import routes.
set local-preference	Specifies a preference value for autonomous system paths that pass a route map.
set metric (BGP-OSPF-RIP)	Sets the metric value for a routing protocol.
set metric-type	Sets the metric type for the destination routing protocol.
set next-hop	Specifies the address of the next hop.
set tag (IP)	Sets a tag value for a route.

maximum-paths

To control the maximum number of parallel routes that an IP routing protocol can support, use the **maximum-paths** command in router address family topology or router configuration mode. To restore the default number of parallel routes, use the **no** form of this command.

maximum-paths *number-of-paths*

no maximum-paths *number-of-paths*

Syntax Description

number-of-paths	Maximum number of parallel routes that an IP routing protocol installs in a routing table. Valid values vary by Cisco IOS release and platform. For more information on valid values, use the question mark (**?**) online help function.

Command Default

The default number of parallel routes vary by Cisco IOS release and platform.

Command Modes

Router address family topology configuration (config-router-af-topology)

Router configuration (config-router)

Command History

Release	Modification
12.2(8)T	This command was introduced.
12.2(14)SX	This command was implemented on the Supervisor Engine 720.
12.2(17d)SXB	Support for this command on the Supervisor Engine 2 was extended to Cisco IOS Release 12.2(17d)SXB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SRB	This command was modified. This command was made available in router address family topology configuration mode.
12.2(33)SXH	This command was modified. The maximum number of paths was changed from 8 to 16 for Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 3.5S	In Cisco IOS XE Release 3.5S, support was added for the Cisco ASR 903 Router.

Usage Guidelines

Release 12.2(33)SRB

If you plan to configure the Multi-Topology Routing (MTR) feature, you need to enter the **maximum-paths** command in router address family topology configuration mode for this Open Shortest Path First (OSPF) router configuration command to become aware of the topology.

Examples

The following example shows how to allow a maximum of 16 paths to a destination in an OSPF routing process:

```
Router(config)# router ospf 3
Router(config-router)# maximum-paths 16
```

nsf

To enable and configure Cisco NSF, use the **nsf** command in router configuration mode. To disable NSF, uses the **no** form of this command.

nsf [**enforce global**]

nsf [{**cisco**| **ietf**}| **interface wait** *seconds*| **interval** *minutes*| **t3** [**adjacency**| **manual** *seconds*]]

no nsf

Syntax Description

enforce global	(Optional) Cancels OSPF NSF restart when non-NSF-aware neighbors are detected.
cisco	Specifies the Cisco proprietary IS-IS NSF method of checkpointing if the active RP fails over.
ietf	Specifies the IETF IS-IS NSF method of protocol modification if the active RP fails over.
interface wait *seconds*	(Optional) Specifies how long to wait for an interface to come up after failover before it proceeds with the Cisco NSF process; valid values are from 1 to 60 seconds.
interval *minutes*	(Optional) Specifies how long to wait after a route processor stabilizes before restarting; valid values are from 0 to 1440 minutes.
t3 adjacency	(Optional) Specifies that the time that IETF NSF waits for the LSP database to synchronize is determined by the adjacency holdtime advertised to the neighbors of the specified RP before switchover.
t3 manual *seconds*	(Optional) Specifies the time to wait after the NSF database synchronizes before informing other nodes to remove the restarting node from consideration as a transit; valid values are from 5 to 3600 seconds.

Command Default

The default settings are as follows:

- NSF is disabled.
- **enforce global** --Enabled.
- interval *minutes*--5 minutes.
- interface **wait***seconds*--10 seconds.

- t3 manual *seconds*--30 seconds.

Command Modes

Router configuration

Command History

Release	Modification
12.2(18)SXD	Support for this command was introduced on the Supervisor Engine 720.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
15.2(1)E	This command was integrated into Cisco IOS Release 15.2(1)E.

Usage Guidelines

The **nsf** command is not supported on Cisco 7600 series routers that are configured with a Supervisor Engine 2.

The **nsfinterfacewait**command can be used if Cisco proprietary IS-IS NSF is configured or if the Internet Engineering Task Force (IETF) IS-IS NSF is enabled using the **nsft3** manual command. You can use this command if an interface is slow to come up.

Note Cisco NSF is required only if the Cisco 7600 series router is expected to perform Cisco NSF during a restart. If the Cisco 7600 series router is expected to cooperate with a neighbor that is doing a Cisco NSF restart only, the switch must be NSF capable by default (running a version of code that supports Cisco NSF), but Cisco NSF does not have to be configured on the switch.

The **nsf** commands are a subset of the **router** command and affects all the interfaces that are covered by the designated process. Cisco NSF supports the BGP, OSPF, IS-IS, and EIGRP protocols. The configuration commands that enable NSF processing are as follows:

- **nsf** under the **routerospf** command
- **nsf ietf** under the **routerisis** command
- **bgp graceful-restart** under the **routerbgp** command

These commands must be issued as part of the router's running configuration. During the restart, these commands are restored to activate the NSF processing.

The [{cisco | ietf } | interface **wait***seconds* | interval *minutes* | t3 [adjacency | manual *seconds*] keywords and arguments apply to IS-IS only.

The {**enforceglobal**} keywords apply to OSPF only.

BGP NSF Guidelines

BGP support in NSF requires that neighbor networking devices be NSF-aware devices; that is, they must have the graceful restart capability and advertise that capability in the OPEN message during session establishment. If an NSF-capable router discovers that a particular BGP neighbor does not have the graceful restart capability enabled, it will not establish an NSF-capable session with that neighbor. All other neighbors that have a

graceful restart capability will continue to have NSF-capable sessions with this NSF-capable networking device. Enter the **bgpgraceful-restart** router configuration command to enable the graceful restart capability.

EIRGP NSF Guidelines

A router may be an NSF-aware router but may not be participating in helping out the NSF restarting neighbor because it is coming up from a cold start.

IS-IS NSF Guidelines

If you configure IETF on the networking device, but neighbor routers are not IETF-compatible, NSF will abort after the switchover.

Use these two keywords when configuring IS-IS NSF:

- **ietf** --Internet Engineering Task Force IS-IS--After a supervisor engine switchover, the NSF-capable router sends the IS-IS NSF restart requests to the neighboring NSF-aware devices.
- **cisco** --Cisco IS-IS. Full adjacency and LSP information is saved (checkpointed) to the standby supervisor engine. After a switchover, the newly active supervisor engine maintains its adjacencies using the checkpointed data to quickly rebuild its routing tables.

OSPF NSF Guidelines

OSPF NSF requires that all neighbor networking devices be NSF-aware devices. If an NSF-capable router discovers that it has non-NSF aware neighbors on a particular network segment, it will disable the NSF capabilities for that segment. The other network segments that are composed entirely of NSF-capable or NSF-aware routers will continue to provide NSF capabilities.

OSPF NSF supports NSF/SSO for IPv4 traffic only. OSPFv3 is not supported with NSF/SSO. Only OSPFv2 is supported with NSF/SSO.

Examples

This example shows how to enable NSF for all OSPF-process interfaces:

```
Router(config)# router ospf 109
Router(config-router)# nsf
```

This example shows how to disable NSF for all OSPF-process interfaces:

```
Router(config)# router ospf 109
Router(config-router)# no nsf
```

Related Commands

Command	Description
router	Enables a routing process.

passive-interface

To disable sending routing updates on an interface, use the **passive-interface** command in router configuration mode. To re-enable the sending of routing updates, use the **no** form of this command.

passive-interface commandpassive-interface [default] i *nterface-type interface-number*

no passive-interface *interface-type interface-number*

Syntax Description

default	(Optional) Causes all interfaces to become passive.
interface-type	Interface type.
interface-number	Interface number.

Command Default

Routing updates are sent on the interface.

Command Modes

Router configuration (config-router)

Command History

Release	Modification
10.0	This command was introduced.
12.0	This command was modified. The **default** keyword was added.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 2.1.
15.1(2)S	This command was integrated into Cisco IOS Release 15.1(2)S.

Usage Guidelines

If you disable the sending of routing updates on an interface, the particular subnet will continue to be advertised to other interfaces, and updates from other routers on that interface continue to be received and processed.

The **default** keyword sets all interfaces as passive by default. You can then configure individual interfaces where adjacencies are desired using the **nopassive-interface** command. The **default** keyword is useful in

Internet service provider (ISP) and large enterprise networks where many of the distribution routers have more than 200 interfaces.

For the Open Shortest Path First (OSPF) protocol, OSPF routing information is neither sent nor received through the specified router interface. The specified interface address appears as a stub network in the OSPF domain.

For the Intermediate System-to-Intermediate System (IS-IS) protocol, this command instructs IS-IS to advertise the IP addresses for the specified interface without actually running IS-IS on that interface. The **no** form of this command for IS-IS disables advertising IP addresses for the specified address.

Note For IS-IS you must keep at least one active interface and configure the interface with the **iprouterisis** command.

The use of the **passive-interface** command in Enhanced Interior Gateway Routing Protocol (EIGRP) suppresses the exchange of hello packets on the interface and thus stops routing updates from being advertised, and it also suppresses incoming routing updates. For more information on passive interfaces, see http://www.cisco.com/en/US/tech/tk365/technologies_tech_note09186a0080093f0a.shtml .

Examples

The following example sends EIGRP updates to all interfaces on network 10.108.0.0 except Ethernet interface 1:

```
router eigrp 109
 network 10.108.0.0
 passive-interface ethernet 1
```

The following configuration enables IS-IS on Ethernet interface 1 and serial interface 0 and advertises the IP addresses of Ethernet interface 0 in its link-state protocol data units (PDUs):

```
router isis Finance
 passive-interface Ethernet 0
interface Ethernet 1
 ip router isis Finance
interface serial 0
 ip router isis Finance
```

The following example sets all interfaces as passive and then activates Ethernet interface 0:

```
router ospf 100
 passive-interface default
 no passive-interface ethernet0
 network 10.108.0.1 0.0.0.255 area 0
```

redistribute (IP)

To redistribute routes from one routing domain into another routing domain, use the **redistribute** command in the appropriate configuration mode. To disable all or some part of the redistribution (depending on the protocol), use the **no** form of this command. See the "Usage Guidelines" section for detailed, protocol-specific behaviors.

redistribute *protocol* [*process-id*] {**level-1**| **level-1-2** | **level-2**} [*autonomous-system-number*] [**metric** {*metric-value* | **transparent**}] [**metric-type** *type-value*] [**match** {**internal** | **external 1** | **external 2**}] [**tag** *tag-value*] [**route-map** *map-tag*] **[subnets] [nssa-only]**

no redistribute *protocol* [*process-id*] {**level-1 | level-1-2 | level-2**} [*autonomous-system-number*] [**metric** {*metric-value* | **transparent**}] [**metric-type** *type-value*] [**match** {**internal** | **external 1** | **external 2**}] [**tag** *tag-value*] [**route-map** *map-tag*] **[subnets] [nssa-only]**

Syntax Description

protocol	Source protocol from which routes are being redistributed. It can be one of the following keywords: **application**, **bgp**, **connected**, **eigrp**, **isis**, **mobile**, **ospf**, **rip**, or **static** [**ip**]. The **static** [**ip**] keyword is used to redistribute IP static routes. The optional **ip** keyword is used when redistributing into the Intermediate System-to-Intermediate System (IS-IS) protocol. The **application** keyword is used to redistribute an application from one routing domain to another. Starting from Cisco IOS XE Release 3.12S, you can redistribute more than one application to different routing protocols such as IS-IS, OSPF, Border Gateway Protocol (BGP), Enhanced Interior Gateway Routing Protocol (EIGRP) and Routing Information Protocol (RIP). The **connected** keyword refers to routes that are established automatically by virtue of having enabled IP on an interface. For routing protocols such as Open Shortest Path First (OSPF) and IS-IS, these routes will be redistributed as external to the autonomous system.

process-id	(Optional) For the **application** keyword, this is the name of an application. For the **bgp** or **eigrp** keyword, this is an autonomous system number, which is a 16-bit decimal number. For the **isis** keyword, this is an optional *tag* value that defines a meaningful name for a routing process. Creating a name for a routing process means that you use names when configuring routing. You can configure a router in two routing domains and redistribute routing information between these two domains. For the **ospf** keyword, this is an appropriate OSPF process ID from which routes are to be redistributed. This identifies the routing process. This value takes the form of a nonzero decimal number. For the **rip** keyword, no *process-id* value is needed. For the **application** keyword, this is the name of an application. By default, no process ID is defined.
level-1	Specifies that, for IS-IS, Level 1 routes are redistributed into other IP routing protocols independently.
level-1-2	Specifies that, for IS-IS, both Level 1 and Level 2 routes are redistributed into other IP routing protocols.
level-2	Specifies that, for IS-IS, Level 2 routes are redistributed into other IP routing protocols independently.
autonomous-system-number	(Optional) Autonomous system number for the redistributed route. The range is from 1 to 65535. • In Cisco IOS Release 12.0(32)SY8, 12.0(33)S3, 12.2(33)SRE, 12.2(33)XNE, 12.2(33)SXI1, Cisco IOS XE Release 2.4, and later releases, 4-byte autonomous system numbers are supported in the range from 65536 to 4294967295 in asplain notation and in the range from 1.0 to 65535.65535 in asdot notation. • In Cisco IOS Release 12.0(32)S12, 12.4(24)T, and Cisco IOS XE Release 2.3, 4-byte autonomous system numbers are supported in the range from 1.0 to 65535.65535 in asdot notation only. For more details about autonomous system number formats, see the **router bgp** command.

metric *metric-value*	(Optional) When redistributing from one OSPF process to another OSPF process on the same router, the metric will be carried through from one process to the other if no metric value is specified. When redistributing other processes to an OSPF process, the default metric is 20 when no metric value is specified. The default value is 0.
metric transparent	(Optional) Causes RIP to use the routing table metric for redistributed routes as the RIP metric.
metric-type *type value*	(Optional) For OSPF, specifies the external link type associated with the default route advertised into the OSPF routing domain. It can be one of two values: • **1—**Type 1 external route • **2—**Type 2 external route If a **metric-type** is not specified, the Cisco IOS software adopts a Type 2 external route. For IS-IS, it can be one of two values: • **internal—**IS-IS metric that is < 63. • **external—**IS-IS metric that is > 64 < 128. The default is **internal**.
match {**internal** | **external1** | **external2**}	(Optional) Specifies the criteria by which OSPF routes are redistributed into other routing domains. It can be one of the following: • **internal—**Routes that are internal to a specific autonomous system. • **external 1—**Routes that are external to the autonomous system, but are imported into OSPF as Type 1 external routes. • **external 2—**Routes that are external to the autonomous system, but are imported into OSPF as Type 2 external routes. The default is **internal**.
tag *tag-value*	(Optional) Specifies the 32-bit decimal value attached to each external route. This is not used by OSPF itself. It may be used to communicate information between Autonomous System Boundary Routers (ASBRs). If none is specified, the remote autonomous system number is used for routes from BGP and Exterior Gateway Protocol (EGP); for other protocols, zero (0) is used.

route-map	(Optional) Specifies the route map that should be interrogated to filter the importation of routes from this source routing protocol to the current routing protocol. If not specified, all routes are redistributed. If this keyword is specified, but no route map tags are listed, no routes will be imported.
map-tag	(Optional) Identifier of a configured route map.
subnets	(Optional) For redistributing routes into OSPF, the scope of redistribution for the specified protocol. By default, no subnets are defined.
nssa-only	(Optional) Sets the nssa-only attribute for all routes redistributed into OSPF.

Command Default

Route redistribution is disabled.

Command Modes

Router configuration (config-router)

Address family configuration (config-af)

Address family topology configuration (config-router-af-topology)

Command History

Release	Modification
10.0	This command was introduced.
12.0(5)T	This command was modified. Address family configuration mode was added.
12.0(22)S	This command was modified. Address family support under EIGRP was added.
12.2(15)T	This command was modified. Address family support under EIGRP was added.
12.2(18)S	This command was modified. Address family support under EIGRP was added.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SRB	This command was modified. Address family topology support under EIGRP was added.
12.2(14)SX	This command was integrated into Cisco IOS Release 12.2(14)SX.
12.0(32)S12	This command was modified. Support for 4-byte autonomous system numbers in asdot notation only was added.

Release	Modification
12.0(32)SY8	This command was modified. Support for 4-byte autonomous system numbers in asplain and asdot notation was added.
12.4(24)T	This command was modified. Support for 4-byte autonomous system numbers in asdot notation only was added.
Cisco IOS XE Release 2.3	This command was modified. Support for 4-byte autonomous system numbers in asdot notation only was added.
12.2(33)SXI1	This command was modified. Support for 4-byte autonomous system numbers in asplain and asdot notation was added.
12.0(33)S3	This command was modified. Support for asplain notation was added, and the default format for 4-byte autonomous system numbers is asplain.
Cisco IOS XE Release 2.4	This command was modified. Support for asplain notation was added, and the default format for 4-byte autonomous system numbers is asplain.
15.0(1)M	This command was modified. The **nssa-only** keyword was added.
12.2(33)SRE	This command was modified. Support for 4-byte autonomous system numbers in asplain and asdot notation was added.
15.1(1)SG	This command was modified. Support for 4-byte autonomous system numbers in asplain and asdot notation was added.
Cisco IOS XE Release 3.3SG	This command was modified. Support for 4-byte autonomous system numbers in asplain and asdot notation was added.
15.1(2)SNG	This command was implemented on the Cisco ASR 901 Series Aggregation Services Routers
Cisco IOS XE Release 3.9S	This command was modified. The **subnets** keyword was deprecated for OSPF classful redistribution.
15.2(1)E	This command was integrated into Cisco IOS Release 15.2(1)E.
Cisco IOS XE Release 3.12S	This command was modified. Support for redistribution of more than one application from one routing domain to another routing domain was added.

Usage Guidelines

Using the no Form of the redistribute Command

Caution Removing options that you have configured for the **redistribute** command requires careful use of the **no** form of the **redistribute** command to ensure that you obtain the result that you are expecting. Changing or disabling any keyword may or may not affect the state of other keywords, depending on the protocol.

It is important to understand that different protocols implement the **no** form of the **redistribute** command differently:

- 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.

Additional Usage Guidelines for the redistribute Command

A router receiving a link-state protocol with an internal metric will consider the cost of the route from itself to the redistributing router plus the advertised cost to reach the destination. An external metric only considers the advertised metric to reach the destination.

Routes learned from IP routing protocols can be redistributed at Level 1 into an attached area or at Level 2. The **level-1-2** keyword allows both Level 1 and Level 2 routes in a single command.

Redistributed routing information must be filtered by the **distribute-list out** router configuration command. This guideline ensures that only those routes intended by the administrator are passed along to the receiving routing protocol.

Whenever you use the **redistribute** or the **default-information** router configuration commands to redistribute routes into an OSPF routing domain, the router automatically becomes an ASBR. However, an ASBR does not, by default, generate a default route into the OSPF routing domain.

When routes are redistributed into OSPF from protocols other than OSPF or BGP, and no metric has been specified with the **metric-type** keyword and *type-value* argument, OSPF will use 20 as the default metric. When routes are redistributed into OSPF from BGP, OSPF will use 1 as the default metric. When routes are redistributed from one OSPF process to another OSPF process, autonomous system external and not-so-stubby-area (NSSA) routes will use 20 as the default metric. When intra-area and inter-area routes are redistributed between OSPF processes, the internal OSPF metric from the redistribution source process is advertised as the external metric in the redistribution destination process. (This is the only case in which the routing table metric will be preserved when routes are redistributed into OSPF.)

When routes are redistributed into OSPF, only routes that are not subnetted are redistributed if the **subnets** keyword is not specified.

Note Depending on your release the **subnets**keyword is automatically appended when you use the **redistribute** *ospf* command. This automatic addition results in the redistribution of classless OSPF routes.

On a router internal to an NSSA area, the **nssa-only** keyword causes the originated type-7 NSSA LSAs to have their propagate (P) bit set to zero, which prevents area border routers from translating these LSAs into type-5 external LSAs. On an area border router that is connected to an NSSA and normal areas, the **nssa-only** keyword causes the routes to be redistributed only into the NSSA areas.

Routes configured with the **connected** keyword affected by this **redistribute** command are the routes not specified by the **network** router configuration command.

You cannot use the **default-metric** command to affect the metric used to advertise connected routes.

Note The **metric** value specified in the **redistribute** command supersedes the **metric** value specified in the **default-metric** command.

The default redistribution of Interior Gateway Protocol (IGP) or Exterior Gateway Protocol (EGP) into BGP is not allowed unless the **default-information originate** router configuration command is specified.

Release 12.2(33)SRB

If you plan to configure the Multi-Topology Routing (MTR) feature, you need to enter the **redistribute** command in address family topology configuration mode in order for this OSPF configuration command to become topology-aware.

4-Byte Autonomous System Number Support

In Cisco IOS Release 12.0(32)SY8, 12.0(33)S3, 12.2(33)SRE, 12.2(33)XNE, 12.2(33)SXI1, Cisco IOS XE Release 2.4, and later releases, the Cisco implementation of 4-byte autonomous system numbers uses asplain—65538 for example—as the default regular expression match and output display format for autonomous system numbers, but you can configure 4-byte autonomous system numbers in both the asplain format and the asdot format as described in RFC 5396. To change the default regular expression match and output display of 4-byte autonomous system numbers to asdot format, use the **bgp asnotation dot** command.

In Cisco IOS Release 12.0(32)S12, 12.4(24)T, and Cisco IOS XE Release 2.3, the Cisco implementation of 4-byte autonomous system numbers uses asdot—1.2, for example—as the only configuration format, regular expression match, and output display, with no asplain support.

Examples

The following example shows how OSPF routes are redistributed into a BGP domain:

```
Device(config)# router bgp 109
Device(config-router)# redistribute ospf
```

The following example shows how to redistribute EIGRP routes into an OSPF domain:

```
Device(config)# router ospf 110
Device(config-router)# redistribute eigrp
```

The following example shows how to redistribute the specified EIGRP process routes into an OSPF domain. The EIGRP-derived metric will be remapped to 100 and RIP routes to 200.

```
Device(config)# router ospf 109
```

```
Device(config-router)# redistribute eigrp 108 metric 100 subnets
Device(config-router)# redistribute rip metric 200 subnets
```

The following example shows how to configure BGP routes to be redistributed into IS-IS. The link-state cost is specified as 5, and the metric type is set to external, indicating that it has lower priority than internal metrics.

```
Device(config)# router isis
Device(config-router)# redistribute bgp 120 metric 5 metric-type external
```

The following example shows how to redistribute an application into an OSPF domain and specify a metric value of 5:

```
Device(config)# router ospf 4
Device(config-router)# redistribute application am metric 5
```

In the following example, network 172.16.0.0 will appear as an external LSA in OSPF 1 with a cost of 100 (the cost is preserved):

```
Device(config)# interface ethernet 0
Device(config-if)# ip address 172.16.0.1 255.0.0.0
Device(config-if)# exit
Device(config)# ip ospf cost 100
Device(config)# interface ethernet 1
Device(config-if)# ip address 10.0.0.1 255.0.0.0
!
Device(config)# router ospf 1
Device(config-router)# network 10.0.0.0 0.255.255.255 area 0
Device(config-if)# exit
Device(config-router)# redistribute ospf 2 subnet
Device(config)# router ospf 2
Device(config-router)# network 172.16.0.0 0.255.255.255 area 0
```

The following example shows how BGP routes are redistributed into OSPF and assigned the local 4-byte autonomous system number in asplain format. This example requires Cisco IOS Release 12.0(32)SY8, 12.0(33)S3, 12.2(33)SRE, 12.2(33)SXI1, Cisco IOS XE Release 2.4, or a later release.

```
Device(config)# router ospf 2
Device(config-router)# redistribute bgp 65538
```

The following example shows how to remove the **connected metric 1000 subnets** options from the **redistribute connected metric 1000 subnets** command and leave the **redistribute connected** command in the configuration:

```
Device(config-router)# no redistribute connected metric 1000 subnets
```

The following example shows how to remove the **metric 1000** options from the **redistribute connected metric 1000 subnets** command and leave the **redistribute connected subnets** command in the configuration:

```
Device(config-router)# no redistribute connected metric 1000
```

The following example shows how to remove the **subnets** option from the **redistribute connected metric 1000 subnets** command and leave the **redistribute connected metric 1000** command in the configuration:

```
Device(config-router)# no redistribute connected subnets
```

The following example shows how to remove the **redistribute connected** command, and any of the options that were configured for the **redistribute connected** command, from the configuration:

```
Device(config-router)# no redistribute connected
```

The following example shows how EIGRP routes are redistributed into an EIGRP process in a named EIGRP configuration:

```
Device(config)# router eigrp virtual-name
Device(config-router)# address-family ipv4 autonomous-system 1
Device(config-router-af)# topology base
Device(config-router-af-topology)# redistribute eigrp 6473 metric 1 1 1 1 1
```

The following example shows how to set and disable the redistributions in EIGRP configuration. Note that, in the case of EIGRP, the **no** form of the commands removes the entire set of **redistribute** commands from the running configuration.

```
Device(config)# router eigrp 1
Device(config-router)# network 0.0.0.0
Device(config-router)# redistribute eigrp 2 route-map x
Device(config-router)# redistribute ospf 1 route-map x
Device(config-router)# redistribute bgp 1 route-map x
Device(config-router)# redistribute isis level-2 route-map x
Device(config-router)# redistribute rip route-map x

Device(config)# router eigrp 1
Device(config-router)# no redistribute eigrp 2 route-map x
Device(config-router)# no redistribute ospf 1 route-map x
Device(config-router)# no redistribute bgp 1 route-map x
Device(config-router)# no redistribute isis level-2 route-map x
Device(config-router)# no redistribute rip route-map x
Device(config-router)# end

Device# show running-config | section router eigrp 1

router eigrp 1
 network 0.0.0.0
```

The following example shows how to set and disable the redistributions in OSPF configuration. Note that the **no** form of the commands removes only the specified keywords from the **redistribute** command in the running configuration.

```
Device(config)# router ospf 1
Device(config-router)# network 0.0.0.0
Device(config-router)# redistribute eigrp 2 route-map x
Device(config-router)# redistribute ospf 1 route-map x
Device(config-router)# redistribute bgp 1 route-map x
Device(config-router)# redistribute isis level-2 route-map x
Device(config-router)# redistribute rip route-map x

Device(config)# router ospf 1
Device(config-router)# no redistribute eigrp 2 route-map x
Device(config-router)# no redistribute ospf 1 route-map x
Device(config-router)# no redistribute bgp 1 route-map x
Device(config-router)# no redistribute isis level-2 route-map x
Device(config-router)# no redistribute rip route-map x
Device(config-router)# end

Device# show running-config | section router ospf 1

router ospf 1
 redistribute eigrp 2
 redistribute ospf 1
 redistribute bgp 1
 redistribute rip
 network 0.0.0.0
```

The following example shows how to remove only the route map filter from the redistribution in BGP; redistribution itself remains in force without a filter:

```
Device(config)# router bgp 65000
Device(config-router)# no redistribute eigrp 2 route-map x
```

The following example shows how to remove the EIGRP redistribution to BGP:

```
Device(config)# router bgp 65000
Device(config-router)# no redistribute eigrp 2
```

Related Commands

Command	Description
address-family (EIGRP)	Enters address family configuration mode to configure an EIGRP routing instance.
address-family ipv4 (BGP)	Places the router in address family configuration mode for configuring routing sessions such as BGP, RIP, or static routing sessions that use standard IPv4 address prefixes.
address-family vpnv4	Places the router in address family configuration mode for configuring routing sessions such as BGP, RIP, or static routing sessions that use standard VPNv4 address prefixes.
bgp asnotation dot	Changes the default display and the regular expression match format of BGP 4-byte autonomous system numbers from asplain (decimal values) to dot notation.
default-information originate (BGP)	Allows the redistribution of network 0.0.0.0 into BGP.
default-information originate (IS-IS)	Generates a default route into an IS-IS routing domain.
default-information originate (OSPF)	Generates a default route into an OSPF routing domain.
distribute-list out (IP)	Suppresses networks from being advertised in updates.
route-map (IP)	Defines the conditions for redistributing routes from one routing protocol into another, or enables policy routing.
router bgp	Configures the BGP routing process.
router eigrp	Configures the EIGRP address-family process.
show route-map	Displays all route maps configured or only the one specified.

Command	Description
topology (EIGRP)	Configures an EIGRP process to route IP traffic under the specified topology instance and enters address family topology configuration mode.

route-map

To define conditions for redistributing routes from one routing protocol to another routing protocol, or to enable policy routing, use the **route-map** command in global configuration mode. To delete an entry, use the **no** form of this command.

route-map *map-tag* [**permit**| **deny**] [*sequence-number*]

no route-map *map-tag* [**permit**| **deny**] [*sequence-number*]

Syntax Description

map-tag	Name for the route map.
permit	(Optional) Permits only routes matching the route map to be forwarded or redistributed.
deny	(Optional) Blocks routes matching the route map from being forwarded or redistributed.
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.

Command Default

Policy routing is not enabled and conditions for redistributing routes from one routing protocol to another routing protocol are not configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
10.0	This command was introduced.
12.3(7)T	This command was updated for use in configuring IPv6 policy-based routing (PBR).
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
Cisco IOS XE Release 2.1	This command was implemented on Cisco ASR 1000 Series Aggregation Services Routers.
12.2(33)SXI4	This command was modified. It was integrated into Cisco IOS Release 12.2(33)SXI4.

Release	Modification
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)SY.

Usage Guidelines

Use the **route-map** command to enter route-map configuration mode.

Use route maps to redistribute routes or to subject packets to policy routing. Both purposes are described in this section.

Redistribution

Use the **route-map** global configuration command and the **match** and **set** route-map configuration commands to define the conditions for redistributing routes from one routing protocol to another. Each **route-map** command has a list of **match** and **set** commands associated with it. The **match** commands specify the *match criteria*--the conditions under which redistribution is allowed for the current **route-map**command. The **set** commands specify the *set actions*--the particular redistribution actions to be performed if the criteria enforced by the **match** commands are met. The **no route-map** command deletes the route map.

The **match** route-map configuration command has multiple formats. The **match** commands can be given in any order, and all **match** commands must "pass" to cause the route to be redistributed according to the *set actions* given with the **set** commands. The **no** forms of the **match** commands remove the specified match criteria.

Use route maps when you want detailed control over how routes are redistributed between routing processes. The destination routing protocol is the one you specify with the **router** global configuration command. The source routing protocol is the one you specify with the **redistribute** router configuration command. See the "Examples" section for an illustration of how route maps are configured.

When you are passing routes through a route map, a route map can have several parts. Any route that does not match at least one **match** clause relating to a **route-map** command will be ignored; that is, the route will not be advertised for outbound route maps and will not be accepted for inbound route maps. If you want to modify only some data, you must configure a second route map section with an explicit match specified.

The **redistribute** router configuration command uses the name specified by the *map-tag* argument to reference a route map. Multiple route maps may share the same map tag name.

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.

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.

Policy Routing

Another purpose of route maps is to enable policy routing. Use the **ip policy route-map** or **ipv6 policy route-map** command, in addition to the **route-map** command, and the **match** and **set** commands to define the conditions for policy routing packets. The **match** commands specify the conditions under which policy routing occurs. The **set** commands specify the routing actions to be performed if the criteria enforced by the **match** commands are met. You might want to policy route packets some way other than the obvious shortest path.

The *sequence-number* argument works as follows:

1. If no entry is defined with the supplied tag, an entry is created with the *sequence-number* argument set to 10.
2. If only one entry is defined with the supplied tag, that entry becomes the default entry for the following **route-map** command. The *sequence-number* argument of this entry is unchanged.
3. If more than one entry is defined with the supplied tag, an error message is printed to indicate that the *sequence-number* argument is required.

If the **no route-map** *map-tag* command is specified (with no *sequence-number* argument), the whole route map is deleted.

Examples

The following example shows how to redistribute Routing Information Protocol (RIP) routes with a hop count equal to 1 to Open Shortest Path First (OSPF). These routes will be redistributed to OSPF as external link-state advertisements (LSAs) with a metric of 5, metric type of Type 1, and a tag equal to 1.

```
Device(config)# router ospf 109
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 type1
Device(config-route-map)# set tag 1
```

The following example for IPv6 shows how to redistribute RIP routes with a hop count equal to 1 to OSPF. These routes will be redistributed into OSPF as external LSAs with a tag equal to 42 and a metric type equal to type1.

```
Device(config)# ipv6 router ospf 1
Device(config-router)# redistribute rip one route-map rip-to-ospfv3
Device(config-router)# exit
Device(config)# route-map rip-to-ospfv3
Device(config-route-map)# match tag 42
Device(config-route-map)# set metric-type type1
```

The following named configuration example shows how to redistribute Enhanced Interior Gateway Routing Protocol (EIGRP) addresses with a hop count equal to 1. These addresses are redistributed to EIGRP as external with a metric of 5 and a tag equal to 1:

```
Device(config)# router eigrp virtual-name1
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# topology base
Device(config-router-af-topology)# redistribute eigrp 6473 route-map
virtual-name1-to-virtual-name2
Device(config-router-af-topology)# exit-address-topology
Device(config-router-af)# exit-address-family
Device(config-router)# router eigrp virtual-name2
Device(config-router)# address-family ipv4 autonomous-system 6473
Device(config-router-af)# topology base
Device(config-router-af-topology)# exit-af-topology
Device(config-router-af)# exit-address-family
Device(config)# route-map virtual-name1-to-virtual-name2
Device(config-route-map)# match tag 42
Device(config-route-map)# set metric 5
Device(config-route-map)# set tag 1
```

Related Commands

Command	Description
ip policy route-map	Identifies a route map to use for policy routing on an interface.
ipv6 policy route-map	Configures IPv6 PBR on an interface.
match as-path	Matches a BGP autonomous system path access list.
match community	Matches a BGP community.
match interface (IP)	Distributes any routes that have their next hop on one of the specified interfaces.
match ip address	Distributes any routes that have a destination network number address that is permitted by a standard or extended access list, and performs policy routing on packets.
match ipv6 address	Distributes IPv6 routes that have a prefix permitted by a prefix list or specifies an IPv6 access list to use to match packets for PBR for IPv6.
match ip next-hop	Redistributes any routes that have a next-hop router address passed by one of the specified access lists.
match ip route-source	Redistributes routes that have been advertised by routers and access servers at the address specified by the access lists.
match length	Bases policy routing on the Level 3 length of a packet.
match metric (IP)	Redistributes routes with the metric specified.
match route-type (IP)	Redistributes routes of the specified type.
match tag	Redistributes routes in the routing table that match the specified tags.
router eigrp	Configures the EIGRP address-family process.
set as-path	Modifies an autonomous system path for BGP routes.
set automatic-tag	Automatically computes the tag value.
set community	Sets the BGP communities attribute.

Command	Description
set default interface	Indicates where to output packets that pass a match clause of a route map for policy routing and have no explicit route to the destination.
set interface	Indicates where to output packets that pass a match clause of a route map for policy routing.
set ipv6 default next-hop	Specifies an IPv6 default next hop to which matching packets will be forwarded.
set ip default next-hop verify-availability	Indicates where to output packets that pass a match clause of a route map for policy routing and for which the Cisco IOS software has no explicit route to a destination.
set ip next-hop	Indicates where to output packets that pass a match clause of a route map for policy routing.
set ipv6 next-hop (PBR)	Indicates where to output IPv6 packets that pass a match clause of a route map for PBR for IPv6.
set level (IP)	Indicates where to import routes.
set local preference	Specifies a preference value for the autonomous system path.
set metric (BGP, OSPF, RIP)	Sets the metric value for a routing protocol.
set metric type	Sets the metric type for the destination routing protocol.
set next-hop	Specifies the address of the next hop.
set tag (IP)	Sets a tag value of the destination routing protocol.
set weight	Specifies the BGP weight for the routing table.
show route-map	Displays all route maps configured or only the one specified.