



IP Routing Commands

- [accept-lifetime](#), on page 4
- [address-family ipv6 \(OSPF\)](#), on page 7
- [address-family l2vpn](#), on page 8
- [aggregate-address](#), on page 11
- [area nssa](#), on page 14
- [area virtual-link](#), on page 16
- [auto-summary \(BGP\)](#), on page 19
- [authentication \(BFD\)](#), on page 22
- [bfd](#), on page 23
- [bfd all-interfaces](#), on page 25
- [bfd check-ctrl-plane-failure](#), on page 26
- [bfd echo](#), on page 27
- [bfd slow-timers](#), on page 29
- [bfd template](#), on page 31
- [bfd-template single-hop](#), on page 32
- [bgp graceful-restart](#), on page 33
- [clear proximity ip bgp](#), on page 35
- [default-information originate \(OSPF\)](#), on page 39
- [default-metric \(BGP\)](#), on page 41
- [distance \(OSPF\)](#), on page 43
- [eigrp log-neighbor-changes](#), on page 46
- [eigrp log-neighbor-warnings](#), on page 48
- [ip authentication key-chain eigrp](#), on page 50
- [ip authentication mode eigrp](#), on page 51
- [ip bandwidth-percent eigrp](#), on page 52
- [ip cef load-sharing algorithm](#), on page 53
- [ip community-list](#), on page 54
- [ip prefix-list](#), on page 59
- [ip hello-interval eigrp](#), on page 62
- [ip hold-time eigrp](#), on page 63
- [ip load-sharing](#), on page 64
- [ip network-broadcast](#), on page 65
- [ip ospf database-filter all out](#), on page 66

- [ip ospf name-lookup](#), on page 67
- [ip split-horizon eigrp](#), on page 68
- [ip summary-address eigrp](#), on page 69
- [ip route static bfd](#), on page 71
- [ipv6 route static bfd](#), on page 73
- [metric weights \(EIGRP\)](#), on page 74
- [neighbor advertisement-interval](#), on page 76
- [neighbor default-originate](#), on page 78
- [neighbor description](#), on page 80
- [neighbor ebgp-multihop](#), on page 81
- [neighbor maximum-prefix \(BGP\)](#), on page 82
- [neighbor peer-group \(assigning members\)](#), on page 84
- [neighbor peer-group \(creating\)](#), on page 86
- [neighbor route-map](#), on page 89
- [neighbor update-source](#), on page 91
- [network \(BGP and multiprotocol BGP\)](#), on page 93
- [network \(EIGRP\)](#), on page 95
- [nsf \(EIGRP\)](#), on page 97
- [offset-list \(EIGRP\)](#), on page 99
- [redistribute \(IP\)](#), on page 101
- [redistribute \(IPv6\)](#), on page 109
- [redistribute maximum-prefix \(OSPF\)](#), on page 112
- [rewrite-evpn-rt-asn](#), on page 114
- [route-map](#), on page 115
- [router-id](#), on page 118
- [router bgp](#), on page 119
- [router eigrp](#), on page 122
- [router ospf](#), on page 123
- [router ospfv3](#), on page 125
- [send-lifetime](#), on page 126
- [set community](#), on page 129
- [set ip next-hop \(BGP\)](#), on page 131
- [show ip bgp](#), on page 133
- [show ip bgp neighbors](#), on page 145
- [show ip bgp ipv6 unicast](#), on page 160
- [show ip eigrp interfaces](#), on page 162
- [show ip eigrp neighbors](#), on page 165
- [show ip eigrp topology](#), on page 168
- [show ip eigrp traffic](#), on page 173
- [show ip ospf](#), on page 175
- [show ip ospf border-routers](#), on page 183
- [show ip ospf database](#), on page 184
- [show ip ospf interface](#), on page 193
- [show ip ospf neighbor](#), on page 196
- [show ip ospf virtual-links](#), on page 202
- [summary-address \(OSPF\)](#), on page 203

- [timers throttle spf, on page 205](#)

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 in key chain key configuration mode. To revert to the default value, use the **no** form of this command.

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

Syntax Description

| | |
|--------------------------------|--|
| local | Specifies the time in local timezone. |
| <i>start-time</i> | Beginning time that the key specified by the key command is valid to be received. The syntax can be either of the following: <i>hh : mm : ss month date year</i> <i>hh : mm : ss date month year</i> <ul style="list-style-type: none"> • <i>hh</i>: Hours • <i>mm</i>: Minutes • <i>ss</i>: Seconds • <i>month</i>: First three letters of the month • <i>date</i>: Date (1-31) • <i>year</i>: Year (four digits) <p>The default start time and the earliest acceptable date is January 1, 1993.</p> |
| infinite | Key is valid to be received from the <i>start-time</i> value on. |
| <i>end-time</i> | Key is valid to be received from the <i>start-time</i> value until the <i>end-time</i> value. The syntax is the same as that for the <i>start-time</i> value. The <i>end-time</i> value must be after the <i>start-time</i> value. The default end time is an infinite time period. |
| duration <i>seconds</i> | 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 |
|-------------------------------|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Bengaluru 17.5.1 | The new range of the duration keyword is from 1 to 2147483646. |

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.

```
Device(config)# interface GigabitEthernet1/0/1
Device(config-if)# ip rip authentication key-chain chain1
Device(config-if)# ip rip authentication mode md5
Device(config-if)# exit
Device(config)# router rip
Device(config-router)# network 172.19.0.0
Device(config-router)# version 2
Device(config-router)# exit
Device(config)# key chain chain1
Device(config-keychain)# key 1
Device(config-keychain-key)# key-string key1
Device(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Device(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Device(config-keychain-key)# exit
Device(config-keychain)# key 2
Device(config-keychain)# key-string key2
Device(config-keychain)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Device(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.

```
Device(config)# router eigrp 10
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# network 10.0.0.0
Device(config-router-af)# af-interface ethernet0/0
Device(config-router-af-interface)# authentication key-chain trees
Device(config-router-af-interface)# authentication mode md5
Device(config-router-af-interface)# exit
Device(config-router-af)# exit
Device(config-router)# exit
Device(config)# key chain chain1
Device(config-keychain)# key 1
Device(config-keychain-key)# key-string key1
Device(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Device(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Device(config-keychain-key)# exit
Device(config-keychain)# key 2
Device(config-keychain-key)# key-string key2
```

```
Device(config-keychain-key) # accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Device(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. |

address-family ipv6 (OSPF)

To enter the address family configuration mode for configuring routing sessions, such as Open Shortest Path First (OSPF), that uses the standard IPv6 address prefixes, use the **address-family ipv6** command in the router configuration mode. To disable the address family configuration mode, use the **no** form of this command.

```
address-family ipv6 [unicast ][{vrf vrf-name }]  
no address-family ipv6 [unicast ][{vrf vrf-name }]
```

Syntax Description

| | |
|-----------------|--|
| unicast | (Optional) Specifies the IPv6 unicast address prefixes. |
| vrf | (Optional) Specifies all the VPN routing and forwarding (VRF) instance tables or a specific VRF table for an IPv6 address. |
| <i>vrf-name</i> | (Optional) A specific VRF table for an IPv6 address. |

Command Default

IPv6 address prefixes are not enabled. Unicast address prefixes are the default when the IPv6 address prefixes are configured.

Command Modes

Router configuration (config-router)

Command History

| Release | Modification |
|------------------------------|--------------|
| Cisco IOS XE Everest 16.5.1a | This command |

Usage Guidelines

The **address-family ipv6** command places the router in address family configuration mode (prompt: config-router-af), from which you can configure routing sessions that use the standard IPv6 address prefixes.

Examples

The following example shows how to place the router in address family configuration mode:

```
Device> enable  
Device# configure terminal  
Device(config)# router ospfv3 1  
Device(config-router)# address-family ipv6 unicast  
Device(config-router-af)#
```

Related Commands

| Command | Description |
|----------------------|--|
| router ospfv3 | Enters OSPFv3 router configuration mode. |

address-family l2vpn

To enter address family configuration mode to configure a routing session using Layer 2 Virtual Private Network (VPN) endpoint provisioning address information, use the **address-family l2vpn** command in router configuration mode. To remove the Layer 2 VPN address family configuration from the running configuration, use the **no** form of this command.

```
address-family l2vpn [{evpn | vpls}]
no address-family l2vpn [{evpn | vpls}]
```

| | | |
|--|--|---|
| Syntax Description | evpn | (Optional) Specifies L2VPN Ethernet Virtual Private Network (EVPN) endpoint provisioning address information. |
| | vpls | (Optional) Specifies L2VPN Virtual Private LAN Service (VPLS) endpoint provisioning address information. |
| Command Default | No Layer 2 VPN endpoint provisioning support is enabled. | |
| Command Modes | Router configuration (config-router) | |
| Command History | Release | Modification |
| | Cisco IOS XE Everest 16.6.1 | This command was introduced. |
| Usage Guidelines | The address-family l2vpn command places the device in address family configuration mode (prompt: config-router-af), from which you can configure routing sessions that support Layer 2 VPN endpoint provisioning. | |
| | BGP support for the Layer 2 VPN address family introduces a BGP-based autodiscovery mechanism to distribute Layer 2 VPN endpoint provisioning information. BGP uses a separate Layer 2 VPN routing information base (RIB) to store endpoint provisioning information, which is updated each time any Layer 2 virtual forwarding instance (VFI) is configured. Prefix and path information is stored in the Layer 2 VPN database, allowing BGP to make best-path decisions. When BGP distributes the endpoint provisioning information in an update message to all its BGP neighbors, the endpoint information is used to set up a pseudowire mesh to support Layer 2 VPN-based services. | |
| | The BGP autodiscovery mechanism facilitates the setting up of Layer 2 VPN services, which are an integral part of the Cisco IOS Virtual Private LAN Service (VPLS) feature. VPLS enables flexibility in deploying services by connecting geographically dispersed sites as a large LAN over high-speed Ethernet in a robust and scalable IP MPLS network. | |
| The multiprotocol capability for address family Layer 2 VPN EVPN is advertised when the Address Family Identifier (AFI) is enabled under the internal BGP (iBGP) and external BGP (eBGP) neighbors for both IPv4 and IPv6 neighbors. | | |



Note Routing information for address family IPv4 is advertised by default for each BGP routing session configured with the **neighbor remote-as** command unless you configure the **no bgp default ipv4-unicast** command before configuring the **neighbor remote-as** command.

Examples

In this example, two provider edge (PE) devices are configured with VPLS endpoint provisioning information that includes Layer 2 VFI, VPN, and VPLS IDs. BGP neighbors are configured and activated under Layer 2 VPN address family to ensure that the VPLS endpoint provisioning information is saved to a separate Layer 2 VPN RIB and then distributed to other BGP peers in BGP update messages. When the endpoint information is received by the BGP peers, a pseudowire mesh is set up to support Layer 2 VPN-based services.

Device A

```
Device> enable
Device# configure terminal
Device(config)# l2 vfi customerA autodiscovery
Device(config-vfi)# vpn id 100
Device(config-vfi)# vpls-id 45000:100
Device(config-vfi)# exit
Device(config)# l2 vfi customerB autodiscovery
Device(config-vfi)# vpn id 200
Device(config-vfi)# vpls-id 45000:200
Device(config-vfi)# exit
Device(config)# router bgp 45000
Device(config-router)# no bgp default ipv4-unicast
Device(config-router)# bgp log-neighbor-changes
Device(config-router)# neighbor 172.16.1.2 remote-as 45000
Device(config-router)# neighbor 172.21.1.2 remote-as 45000
Device(config-router)# address-family l2vpn vpls
Device(config-router-af)# neighbor 172.16.1.2 activate
Device(config-router-af)# neighbor 172.16.1.2 send-community extended
Device(config-router-af)# neighbor 172.21.1.2 activate
Device(config-router-af)# neighbor 172.21.1.2 send-community extended
Device(config-router-af)# end
```

Device B

```
Device> enable
Device# configure terminal
Device(config)# l2 vfi customerA autodiscovery
Device(config-vfi)# vpn id 100
Device(config-vfi)# vpls-id 45000:100
Device(config-vfi)# exit
Device(config)# l2 vfi customerB autodiscovery
Device(config-vfi)# vpn id 200
Device(config-vfi)# vpls-id 45000:200
Device(config-vfi)# exit
Device(config)# router bgp 45000
Device(config-router)# no bgp default ipv4-unicast
Device(config-router)# bgp log-neighbor-changes
Device(config-router)# neighbor 172.16.1.1 remote-as 45000
Device(config-router)# neighbor 172.22.1.1 remote-as 45000
```

```
Device(config-router)# address-family l2vpn vpls
Device(config-router-af)# neighbor 172.16.1.1 activate
Device(config-router-af)# neighbor 172.16.1.1 send-community extended
Device(config-router-af)# neighbor 172.22.1.1 activate
Device(config-router-af)# neighbor 172.22.1.1 send-community extended
Device(config-router-af)# end
```

Related Commands

| Command | Description |
|--------------------------|--|
| neighbor activate | Enables the exchange of information with a BGP neighboring router. |

aggregate-address

To create an aggregate entry in a Border Gateway Protocol (BGP) database, use the **aggregate-address** command in address family or router configuration mode. To disable this function, use the **no** form of this command.

```
aggregate-address address mask [as-set] [as-confed-set] [summary-only] [suppress-map map-name]
[advertise-map map-name] [attribute-map map-name]
no aggregate-address address mask [as-set] [as-confed-set] [summary-only] [suppress-map
map-name] [advertise-map map-name] [attribute-map map-name]
```

Syntax Description

| | |
|--------------------------------------|---|
| <i>address</i> | Aggregate address. |
| <i>mask</i> | Aggregate mask. |
| as-set | (Optional) Generates autonomous system set path information. |
| as-confed-set | (Optional) Generates autonomous confederation set path information. |
| summary-only | (Optional) Filters all more-specific routes from updates. |
| suppress-map <i>map-name</i> | (Optional) Specifies the name of the route map used to select the routes to be suppressed. |
| advertise-map <i>map-name</i> | (Optional) Specifies the name of the route map used to select the routes to create AS_SET origin communities. |
| attribute-map <i>map-name</i> | (Optional) Specifies the name of the route map used to set the attribute of the aggregate route. |

Command Default

The atomic aggregate attribute is set automatically when an aggregate route is created with this command unless the **as-set** keyword is specified.

Command Modes

Address family configuration (config-router-af)

Router configuration (config-router)

Command History

Table 1:

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

You can implement aggregate routing in BGP and Multiprotocol BGP (mBGP) either by redistributing an aggregate route into BGP or mBGP, or by using the conditional aggregate routing feature.

Using the **aggregate-address** command with no keywords will create an aggregate entry in the BGP or mBGP routing table if any more-specific BGP or mBGP routes are available that fall within the specified range. (A longer prefix that matches the aggregate must exist in the Routing Information Base (RIB).) The aggregate route will be advertised as coming from your autonomous system and will have the atomic aggregate attribute

set to show that information might be missing. (By default, the atomic aggregate attribute is set unless you specify the **as-set** keyword.)

Using the **as-set** keyword creates an aggregate entry using the same rules that the command follows without this keyword, but the path advertised for this route will be an AS_SET consisting of all elements contained in all paths that are being summarized. Do not use this form of the **aggregate-address** command when aggregating many paths, because this route must be continually withdrawn and updated as autonomous system path reachability information for the summarized routes changes.

Using the **as-confed-set** keyword creates an aggregate entry using the same rules that the command follows without this keyword. This keyword performs the same function as the **as-set** keyword, except that it generates autonomous confed set path information.

Using the **summary-only** keyword not only creates the aggregate route (for example, 192.*.*.) but also suppresses advertisements of more-specific routes to all neighbors. If you want to suppress only advertisements to certain neighbors, you may use the **neighbor distribute-list** command, with caution. If a more-specific route leaks out, all BGP or mBGP routers will prefer that route over the less-specific aggregate you are generating (using longest-match routing).

Using the **suppress-map** keyword creates the aggregate route but suppresses advertisement of specified routes. You can use the **match** clauses of route maps to selectively suppress some more-specific routes of the aggregate and leave others unsuppressed. IP access lists and autonomous system path access lists match clauses are supported.

Using the **advertise-map** keyword selects specific routes that will be used to build different components of the aggregate route, such as AS_SET or community. This form of the **aggregate-address** command is useful when the components of an aggregate are in separate autonomous systems and you want to create an aggregate with AS_SET, and advertise it back to some of the same autonomous systems. You must remember to omit the specific autonomous system numbers from the AS_SET to prevent the aggregate from being dropped by the BGP loop detection mechanism at the receiving router. IP access lists and autonomous system path access lists **match** clauses are supported.

Using the **attribute-map** keyword allows attributes of the aggregate route to be changed. This form of the **aggregate-address** command is useful when one of the routes forming the AS_SET is configured with an attribute such as the community no-export attribute, which would prevent the aggregate route from being exported. An attribute map route map can be created to change the aggregate attributes.

AS-Set Example

In the following example, an aggregate BGP address is created in router configuration mode. The path advertised for this route will be an AS_SET consisting of all elements contained in all paths that are being summarized.

```
Device(config)#router bgp 50000
Device(config-router)#aggregate-address 10.0.0.0 255.0.0.0 as-set
```

Summary-Only Example

In the following example, an aggregate BGP address is created in address family configuration mode and applied to the multicast database under the IP Version 4 address family. Because the **summary-only** keyword is configured, more-specific routes are filtered from updates.

```
Device(config)#router bgp 50000
```

```
Device(config-router)#address-family ipv4 multicast
Device(config-router-af)#aggregate-address 10.0.0.0 255.0.0.0 summary-only
```

Conditional Aggregation Example

In the following example, a route map called MAP-ONE is created to match on an AS-path access list. The path advertised for this route will be an AS_SET consisting of elements contained in paths that are matched in the route map.

```
Device(config)#ip as-path access-list 1 deny ^1234_
Device(config)#ip as-path access-list 1 permit .*
Device(config)#!
Device(config)#route-map MAP-ONE
Device(config-route-map)#match ip as-path 1
Device(config-route-map)#exit
Device(config)#router bgp 50000
Device(config-router)#address-family ipv4
Device(config-router-af)#aggregate-address 10.0.0.0 255.0.0.0 as-set advertise-map
MAP-ONE
Router(config-router-af)#end
```

Related Commands

| Command | Description |
|----------------------------------|---|
| 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. |
| ip as-path access-list | Defines a BGP autonomous system path access list. |
| 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. |
| neighbor distribute-list | Distributes BGP neighbor information in an access list. |
| route-map (IP) | Defines the conditions for redistributing routes from one routing protocol into another, or enables policy routing. |

area nssa

To configure a not-so-stubby area (NSSA), use the **area nssa** command in router address family topology or router configuration mode. To remove the NSSA distinction from the area, use the **no** form of this command.

```
area nssa command area area-id nssa [no-redistribution] [default-information-originate [metric]
[metric-type]] [no-summary] [nssa-only]
no area area-id nssa [no-redistribution] [default-information-originate [metric] [metric-type]]
[no-summary] [nssa-only]
```

Syntax Description

| | |
|--------------------------------------|---|
| <i>area-id</i> | Identifier for the stub area or NSSA. The identifier can be specified as either a decimal value or an IP address. |
| no-redistribution | (Optional) Used when the router is an NSSA Area Border Router (ABR) and you want the redistribute command to import routes only into the normal areas, but not into the NSSA area. |
| default-information-originate | (Optional) Used to generate a Type 7 default into the NSSA area. This keyword takes effect only on the NSSA ABR or the NSSA Autonomous System Boundary Router (ASBR). |
| metric | (Optional) Specifies the OSPF default metric. |
| metric-type | (Optional) Specifies the OSPF metric type for default routes. |
| no-summary | (Optional) Allows an area to be an NSSA but not have summary routes injected into it. |
| nssa-only | (Optional) Limits the default advertisement to this NSSA area by setting the propagate (P) bit in the type-7 LSA to zero. |

Command Default

No NSSA area is defined.

Command Modes

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

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

To remove the specified area from the software configuration, use the **no area area-id** command (with no other keywords). That is, the **no area area-id** command removes all area options, including **area authentication**, **area default-cost**, **area nssa**, **area range**, **area stub**, and **area virtual-link**.

Release 12.2(33)SRB

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

Examples

The following example makes area 1 an NSSA area:

```
router ospf 1
 redistribute rip subnets
 network 172.19.92.0 0.0.0.255 area 1
 area 1 nssa
```

Related Commands

| Command | Description |
|---------------------|---|
| redistribute | Redistributes routes from one routing domain into another routing domain. |

area virtual-link

To define an Open Shortest Path First (OSPF) virtual link, use the **area virtual-link** command in router address family topology, router configuration, or address family configuration mode. To remove a virtual link, use the **no** form of this command.

```
area area-id virtual-link router-id authentication key-chain chain-name [hello-interval seconds]
[retransmit-interval seconds] [transmit-delay seconds] [dead-interval seconds] [ttl-security hops
hop-count]
```

```
no area area-id virtual-link router-id authentication key-chain chain-name
```

Syntax Description

Table 2:

| | |
|---|--|
| <i>area-id</i> | Area ID assigned to the virtual link. This can be either a decimal value or a valid IPv6 prefix. There is no default. |
| <i>router-id</i> | Router ID associated with the virtual link neighbor. The router ID appears in the show ip ospf or show ipv6 display command. There is no default. |
| authentication | Enables virtual link authentication. |
| key-chain | Configures a key-chain for cryptographic authentication keys. |
| <i>chain-name</i> | Name of the authentication key that is valid. |
| hello-interval <i>seconds</i> | (Optional) Specifies the time (in seconds) between the hello packets that the Cisco IOS software sends on an interface. The hello interval is an unsigned integer value to be advertised in the hello packets. The value must be the same for all routers and access servers attached to a common network. The range is from 1 to 8192. The default is 10. |
| retransmit-interval <i>seconds</i> | (Optional) Specifies the time (in seconds) between link-state advertisement (LSA) retransmissions for adjacencies belonging to the interface. The retransmit interval is the expected round-trip delay between any two routers on the attached network. The value must be greater than the expected round-trip delay. The range is from 1 to 8192. The default is 5. |
| transmit-delay <i>seconds</i> | (Optional) Specifies the estimated time (in seconds) required to send a link-state update packet on the interface. The integer value that must be greater than zero. LSAs in the update packet have their age incremented by this amount before transmission. The range is from 1 to 8192. The default value is 1. |

| | |
|---|--|
| dead-interval <i>seconds</i> | (Optional) Specifies the time (in seconds) that hello packets are not seen before a neighbor declares the router down. The dead interval is an unsigned integer value. The default is four times the hello interval, or 40 seconds. As with the hello interval, this value must be the same for all routers and access servers attached to a common network. |
| ttl-security hops <i>hop-count</i> | (Optional) Configures Time-to-Live (TTL) security on a virtual link. The <i>hop-count</i> argument range is from 1 to 254. |

Command Default

No OSPF virtual link is defined.

Command Modes

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

Router configuration (config-router)

Address family configuration (config-router-af)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

In OSPF, all areas must be connected to a backbone area. A lost connection to the backbone can be repaired by establishing a virtual link.

The shorter the hello interval, the faster topological changes will be detected, but more routing traffic will ensue. The setting of the retransmit interval should be conservative, or needless retransmissions will result. The value should be larger for serial lines and virtual links.

You should choose a transmit delay value that considers the transmission and propagation delays for the interface.

To configure a virtual link in OSPF for IPv6, you must use a router ID instead of an address. In OSPF for IPv6, the virtual link takes the router ID rather than the IPv6 prefix of the remote router.

Use the **ttl-security hops** *hop-count* keywords and argument to enable checking of TTL values on OSPF packets from neighbors or to set TTL values sent to neighbors. This feature adds an extra layer of protection to OSPF.



Note In order for a virtual link to be properly configured, each virtual link neighbor must include the transit area ID and the corresponding virtual link neighbor router ID. To display the router ID, use the **show ip ospf** or the **show ipv6 ospf** command in privileged EXEC mode.



Note To remove the specified area from the software configuration, use the **no area** *area-id* command (with no other keywords). That is, the **no area** *area-id* command removes all area options, such as **area default-cost**, **area nssa**, **area range**, **area stub**, and **area virtual-link**.

Release 12.2(33)SRB

If you plan to configure the Multitopology Routing (MTR) feature, you need to enter the **area virtual-link** command in router address family topology configuration mode in order for this OSPF router configuration command to become topology-aware.

Examples

The following example establishes a virtual link with default values for all optional parameters:

```
Device(config)# ipv6 router ospf 1
Device(config)# log-adjacency-changes
Device(config)# area 1 virtual-link 192.168.255.1
```

The following example establishes a virtual link in OSPF for IPv6:

```
Device(config)# ipv6 router ospf 1
Device(config)# log-adjacency-changes
Device(config)# area 1 virtual-link 192.168.255.1 hello-interval 5
```

The following example shows how to configure TTL security for a virtual link in OSPFv3 for IPv6:

```
Device(config)# router ospfv3 1
Device(config-router)# address-family ipv6 unicast vrf vrf1
Device(config-router-af)# area 1 virtual-link 10.1.1.1 ttl-security hops 10
```

The following example shows how to configure the authentication using a key chain for virtual-links:

```
Device(config)# area 1 virtual-link 192.168.255.1 authentication key-chain ospf-chain-1
```

Related Commands

| Command | Description |
|--------------------------|--|
| area | Configures OSPFv3 area parameters. |
| show ip ospf | Enables the display of general information about OSPF routing processes. |
| show ipv6 ospf | Enables the display of general information about OSPF routing processes. |
| ttl-security hops | Enables checking of TTL values on OSPF packets from neighbors or setting TTL values sent to neighbors. |

auto-summary (BGP)

To configure automatic summarization of subnet routes into network-level routes, use the **auto-summary** command in address family or router configuration mode. To disable automatic summarization and send subprefix routing information across classful network boundaries, use the **no** form of this command.

auto-summary
no auto-summary

Syntax Description

This command has no arguments or keywords.

Command Default

Automatic summarization is disabled by default (the software sends subprefix routing information across classful network boundaries).

Command Modes

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

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

BGP automatically summarizes routes to classful network boundaries when this command is enabled. Route summarization is used to reduce the amount of routing information in routing tables. Automatic summarization applies to connected, static, and redistributed routes.



Note The MPLS VPN Per VRF Label feature does not support auto-summary.

By default, automatic summarization is disabled and BGP accepts subnets redistributed from an Interior Gateway Protocol (IGP). To block subnets and create summary subprefixes to the classful network boundary when crossing classful network boundaries, use the **auto-summary** command.

To advertise and carry subnet routes in BGP when automatic summarization is enabled, use an explicit **network** command to advertise the subnet. The **auto-summary** command does not apply to routes injected into BGP via the **network** command or through iBGP or eBGP.

Why auto-summary for BGP Is Disabled By Default

When **auto-summary** is enabled, routes injected into BGP via redistribution are summarized on a classful boundary. Remember that a 32-bit IP address consists of a network address and a host address. The subnet mask determines the number of bits used for the network address and the number of bits used for the host address. The IP address classes have a natural or standard subnet mask, as shown in the table below.

Table 3: IP Address Classes

| Class | Address Range | Standard Mask |
|-------|--------------------------|--------------------|
| A | 1.0.0.0 to 126.0.0.0 | 255.0.0.0 or /8 |
| B | 128.1.0.0 to 191.254.0.0 | 255.255.0.0 or /16 |

| Class | Address Range | Standard Mask |
|-------|----------------------------|----------------------|
| C | 192.0.1.0 to 223.255.254.0 | 255.255.255.0 or /24 |

Reserved addresses include 128.0.0.0, 191.255.0.0, 192.0.0.0, and 223.255.255.0.

When using the standard subnet mask, Class A addresses have one octet for the network, Class B addresses have two octets for the network, and Class C addresses have three octets for the network.

Consider the Class B address 156.26.32.1 with a 24-bit subnet mask, for example. The 24-bit subnet mask selects three octets, 156.26.32, for the network. The last octet is the host address. If the network 156.26.32.1/24 is learned via an IGP and is then redistributed into BGP, if **auto-summary** were enabled, the network would be automatically summarized to the natural mask for a Class B network. The network that BGP would advertise is 156.26.0.0/16. BGP would be advertising that it can reach the entire Class B address space from 156.26.0.0 to 156.26.255.255. If the only network that can be reached via the BGP router is 156.26.32.0/24, BGP would be advertising 254 networks that cannot be reached via this router. This is why the **auto-summary (BGP)** command is disabled by default.

Examples

In the following example, automatic summarization is enabled for IPv4 address family prefixes:

```
Device(config)#router bgp 50000
Device(config-router)#address-family ipv4 unicast
Device(config-router-af)#auto-summary
Device(config-router-af)#network 7.7.7.7 255.255.255.255
```

In the example, there are different subnets, such as 7.7.7.6 and 7.7.7.7 on Loopback interface 6 and Loopback interface 7, respectively. Both **auto-summary** and a **network** command are configured.

```
Device#show ip interface brief
Interface          IP-Address      OK? Method Status          Protocol
Ethernet0/0        100.0.1.7       YES NVRAM    up              up
Ethernet0/1        unassigned      YES NVRAM    administratively down down
Ethernet0/2        unassigned      YES NVRAM    administratively down down
Ethernet0/3        unassigned      YES NVRAM    administratively down down
Ethernet1/0        108.7.9.7       YES NVRAM    up              up
Ethernet1/1        unassigned      YES NVRAM    administratively down down
Ethernet1/2        unassigned      YES NVRAM    administratively down down
Ethernet1/3        unassigned      YES NVRAM    administratively down down
Loopback6          7.7.7.6         YES NVRAM    up              up
Loopback7          7.7.7.7         YES NVRAM    up              up
```

Note that in the output below, because of the **auto-summary** command, the BGP routing table displays the summarized route 7.0.0.0 instead of 7.7.7.6. The 7.7.7.7/32 network is displayed because it was configured with the **network** command, which is not affected by the **auto-summary** command.

```
Device#show ip bgp
BGP table version is 10, local router ID is 7.7.7.7
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, x best-external
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop          Metric LocPrf Weight Path
*> 6.6.6.6/32       100.0.1.6         0           0 6 i
*> 7.0.0.0          0.0.0.0           0           32768 ? <-- summarization
*> 7.7.7.7/32       0.0.0.0           0           32768 i <-- network command
```

```

r>i9.9.9.9/32      108.7.9.9      0    100      0  i
*> 100.0.0.0      0.0.0.0         0          32768 ?
r> 100.0.1.0/24   100.0.1.6       0          0  6  ?
*> 108.0.0.0      0.0.0.0         0          32768 ?
r>i108.7.9.0/24   108.7.9.9       0    100      0  ?
*>i200.0.1.0      108.7.9.9

```

Related Commands

| Command | Description |
|--|--|
| 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. |
| network (BGP and multiprotocol BGP) | Specifies the networks to be advertised by BGP and multiprotocol BGP. |

authentication (BFD)

To configure authentication in a Bidirectional Forwarding Detection (BFD) template for single hop sessions, use the **authentication** command in BFD configuration mode. To disable authentication in BFD template for single-hop sessions, use the **no** form of this command

authentication *authentication-type* **keychain** *keychain-name*
no authentication *authentication-type* **keychain** *keychain-name*

| | | |
|---------------------------|--------------------------------------|---|
| Syntax Description | <i>authentication-type</i> | Authentication type. Valid values are md5, meticulous-md5, meticulous-sha1, and sha-1. |
| | keychain <i>keychain-name</i> | Configures an authentication key chain with the specified name. The maximum number of characters allowed in the name is 32. |

Command Default Authentication in BFD template for single hop sessions is not enabled.

Command Modes BFD configuration (config-bfd)

| Command History | Release | Modification |
|------------------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines You can configure authentication in single hop templates. We recommend that you configure authentication to enhance security. Authentication must be configured on each BFD source-destination pair, and authentication parameters must match on both devices.

Examples The following example shows how to configure authentication for the template1 BFD single-hop template:

```
Device>enable
Device#configuration terminal
Device(config)#bfd-template single-hop template1
Device(config-bfd)#authentication sha-1 keychain bfd-singlehop
```

bfd

To set the baseline Bidirectional Forwarding Detection (BFD) session parameters on an interface, use the **bfd** interface configuration mode. To remove the baseline BFD session parameters, use the **no** form of this command

bfd interval *milliseconds* **min_rx** *milliseconds* **multiplier** *multiplier-value*
no bfd interval *milliseconds* **min_rx** *milliseconds* **multiplier** *multiplier-value*

| Syntax Description | Parameter | Description |
|--------------------|---|--|
| | interval <i>milliseconds</i> | Specifies the rate, in milliseconds, at which BFD control packets will be sent to BFD peers. The valid range for the milliseconds argument is from 50 to 9999. |
| | min_rx <i>milliseconds</i> | Specifies the rate, in milliseconds, at which BFD control packets will be expected to be received from BFD peers. The valid range for the milliseconds argument is from 50 to 9999. |
| | multiplier <i>multiplier-value</i> | Specifies the number of consecutive BFD control packets that must be missed from a BFD peer before BFD declares that the peer is unavailable and the Layer 3 BFD peer is informed of the failure. The valid range for the multiplier-value argument is from 3 to 50. |

Command Default No baseline BFD session parameters are set.

Command Modes Interface configuration (config-if)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines The **bfd** command can be configured on SVI, Ethernet and port-channel interfaces. If BFD runs on a port channel interface, BFD has a timer value restriction of $750 * 3$ milliseconds.

The **bfd interval** configuration is not removed when:

- an IPv4 address is removed from an interface
- an IPv6 address is removed from an interface
- IPv6 is disabled from an interface
- an interface is shutdown
- IPv4 CEF is disabled globally or locally on an interface
- IPv6 CEF is disabled globally or locally on an interface

The **bfd interval** configuration is removed when the subinterface on which it is configured is removed.



Note If we configure bfd interval command in interface config mode, then bfd echo mode is enabled by default. We need to enable either no ip redirect (if BFD echo is needed) or no bfd echo in interface config mode.

Before using BFD echo mode, you must disable sending Internet Control Message Protocol (ICMP) redirect messages by entering the no ip redirect command, in order to avoid high CPU utilization.

Examples

The following example shows the BFD session parameters set for Gigabit Ethernet 1/0/3:

```
Device>enable
Device#configuration terminal
Device(config)#interface gigabitethernet 1/0/3
Device(config-if)#bfd interval 100 min_rx 100 multiplier 3
```


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) | |
| Command History | Release | Modification |
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Usage Guidelines | To enable BFD for all interfaces, enter 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:

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

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

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

bfd check-ctrl-plane-failure

To enable Bidirectional Forwarding Detection (BFD) control plane failure checking for the Intermediate System-to-Intermediate System (IS-IS) routing protocol, use the **bfd check-control-plane-failure** command in router configuration mode. To disable control plane failure detection, use the **no** form of this command

bfd check-ctrl-plane-failure
no bfd check-ctrl-plane-failure

Syntax Description This command has no arguments or keywords.

Command Default BFD control plane failure checking is disabled.

Command Modes Router configuration (config-router)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines The `bfd check-ctrl-plane-failure` command can be configured for an IS-IS routing process only. The command is not supported on other protocols.

When a switch restarts, a false BFD session failure can occur, where neighboring routers behave as if a true forwarding failure has occurred. However, if the `bfd check-ctrl-plane-failure` command is enabled on a switch, the router can ignore control plane related BFD session failures. We recommend that you add this command to the configuration of all neighboring routers just prior to a planned router restart, and that you remove the command from all neighboring routers when the restart is complete.

Examples

The following example enables BFD control plane failure checking for the IS-IS routing protocol:

```
Device>enable
Device#configuration terminal
Device(config)#router isis
Device(config-router)#bfd check-ctrl-plane-failure
Device(config-router)#end
```

bfd echo

To enable Bidirectional Forwarding Detection (BFD) echo mode, use the **bfd echo** command in interface configuration mode. To disable BFD echo mode, use the **no** form of this command

bfd echo
no bfd echo

| Syntax Description | This command has no arguments or keywords. | | | | |
|------------------------------|--|---------|--------------|------------------------------|------------------------------|
| Command Default | BFD echo mode is enabled by default if BFD is configured using bfd interval command in interface configuration mode. | | | | |
| Command Modes | Interface configuration (config-if) | | | | |
| Command History | <table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>Cisco IOS XE Everest 16.5.1a</td> <td>This command was introduced.</td> </tr> </tbody> </table> | Release | Modification | Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Release | Modification | | | | |
| Cisco IOS XE Everest 16.5.1a | This command was introduced. | | | | |
| Usage Guidelines | <p>Echo mode is enabled by default. Entering the no bfd echo command without any keywords turns off the sending of echo packets and signifies that the switch is unwilling to forward echo packets received from BFD neighbor switches.</p> <p>When echo mode is enabled, the desired minimum echo transmit interval and required minimum transmit interval values are taken from the bfd interval <i>milliseconds</i> min_rx <i>milliseconds</i> parameters, respectively.</p> | | | | |



Note Before using BFD echo mode, you must disable sending Internet Control Message Protocol (ICMP) redirect messages by entering the **no ip redirects** command, in order to avoid high CPU utilization.

Examples

The following example configures echo mode between BFD neighbors:

```
Device>enable
Device#configuration terminal
Device(config)#interface GigabitEthernet 1/0/3
Device(config-if)#bfd echo
```

The following output from the **show bfd neighbors details** command shows that the BFD session neighbor is up and using BFD echo mode. The relevant command output is shown in bold in the output.

```
Device#show bfd neighbors details
OurAddr      NeighAddr  LD/RD  RH/RS  Holdown(mult)  State Int
172.16.1.2   172.16.1.1  1/6    Up     0 (3 )         Up    Fa0/1
Session state is UP and using echo function with 100 ms interval.
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 3
Received MinRxInt: 1000000, Received Multiplier: 3
Holdown (hits): 3000(0), Hello (hits): 1000(337)
Rx Count: 341, Rx Interval (ms) min/max/avg: 1/1008/882 last: 364 ms ago
Tx Count: 339, Tx Interval (ms) min/max/avg: 1/1016/886 last: 632 ms ago
Registered protocols: EIGRP
```

```
Uptime: 00:05:00
Last packet: Version: 1          - Diagnostic: 0
                State bit: Up    - Demand bit: 0
                Poll bit: 0       - Final bit: 0
                Multiplier: 3     - Length: 24
                My Discr.: 6      - Your Discr.: 1
                Min tx interval: 1000000 - Min rx interval: 1000000
                Min Echo interval: 50000
```

bfd slow-timers

To configure the Bidirectional Forwarding Detection (BFD) slow timers value, use the **bfd slow-timers** command in interface configuration mode. To change the slow timers used by BFD, use the **no** form of this command

```
bfd slow-timers [milliseconds]  
no bfd slow-timers
```

| | |
|------------------------|---|
| Command Default | The BFD slow timer value is 1000 milliseconds |
|------------------------|---|

| | |
|----------------------|-------------------------------|
| Command Modes | Global configuration (config) |
|----------------------|-------------------------------|

| | | |
|------------------------|------------------------------|------------------------------|
| Command History | Release | Modification |
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Examples

The following example shows how to configure the BFD slow timers value to 14,000 milliseconds:

```
Device(config)#bfd slow-timers 14000
```

The following output from the show bfd neighbors details command shows that the BFD slow timers value of 14,000 milliseconds has been implemented. The values for the MinTxInt and MinRxInt will correspond to the configured value for the BFD slow timers. The relevant command output is shown in bold.

```
Device#show bfd neighbors details  
OurAddr      NeighAddr  LD/RD  RH/RS  Holdown(mult)  State  Int  
172.16.1.2   172.16.1.1  1/6    Up      0 (3 )         Up     Fa0/1  
Session state is UP and using echo function with 100 ms interval.  
Local Diag: 0, Demand mode: 0, Poll bit: 0  
MinTxInt: 14000, MinRxInt: 14000, Multiplier: 3  
Received MinRxInt: 1000000, Received Multiplier: 3  
Holdown (hits): 3600(0), Hello (hits): 1200(337)  
Rx Count: 341, Rx Interval (ms) min/max/avg: 1/1008/882 last: 364 ms ago  
Tx Count: 339, Tx Interval (ms) min/max/avg: 1/1016/886 last: 632 ms ago  
Registered protocols: EIGRP  
Uptime: 00:05:00  
Last packet: Version: 1                - Diagnostic: 0  
                State bit: Up          - Demand bit: 0  
                Poll bit: 0            - Final bit: 0  
                Multiplier: 3          - Length: 24  
                My Discr.: 6           - Your Discr.: 1  
                Min tx interval: 1000000 - Min rx interval: 1000000  
                Min Echo interval: 50000
```



Note

- If the BFD session is down, then the BFD control packets will be sent with the slow timer interval.
 - If the BFD session is up, then if echo is enabled, then BFD control packets will be sent in negotiated slow timer interval and echo packets will be sent in negotiated configured BFD interval. If echo is not enabled, then BFD control packets will be sent in negotiated configured interval.
-

bfd template

To create a Bidirectional Forwarding Detection (BFD) template and to enter BFD configuration mode, use the **bfd-template** command in global configuration mode. To remove a BFD template, use the **no** form of this command

```
bfd template template-name  
no bfd template template-name
```

Command Default A BFD template is not bound to an interface.

Command Modes Interface configuration (config-if)

| Command History | Release | Modification |
|------------------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines Even if you have not created the template by using the **bfd-template** command, you can configure the name of the template under an interface, but the template is considered invalid until you define the template. You do not have to reconfigure the template name again. It becomes valid automatically.

Examples

```
Device> enable  
Device#configuration terminal  
Device(config)#interface GigabitEthernet 1/3/0  
Device(config-if)#bfd template template1
```

bfd-template single-hop

To bind a single hop Bidirectional Forwarding Detection (BFD) template to an interface, use the **bfd template** command in interface configuration mode. To unbind single-hop BFD template from an interface, use the **no** form of this command

bfd-template single-hop *template-name*
no bfd-template single-hop *template-name*

| | |
|---------------------------|--|
| Syntax Description | single-hop Creates the single-hop BFD template. |
| | <i>template-name</i> Template name. |

Command Default A BFD template does not exist.

Command Modes Global configuration (config)

| | | |
|------------------------|------------------------------|------------------------------|
| Command History | Release | Modification |
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines The bfd-template command allows you to create a BFD template and places the device in BFD configuration mode. The template can be used to specify a set of BFD interval values. BFD interval values specified as part of the BFD template are not specific to a single interface.

Examples

The following example shows how to create a BFD template and specify BFD interval values:

```
Device>enable
Device#configuration terminal
Device(config)#bfd-template single-hop node1
Device(bfd-config)#interval min-tx 100 min-rx 100 multiplier 3
Device(bfd-config)#echo
```

The following example shows how to create a BFD single-hop template and configure BFD interval values and an authentication key chain:

```
Device> enable
Device#configuration terminal
Device(config)#bfd-template single-hop template1
Device(bfd-config)#interval min-tx 200 min-rx 200 multiplier 3
Device(bfd-config)#authentication keyed-sha-1 keychain bfd_singlehop
```



Note BFD echo is not enabled by default in the bfd-template configuration. This needs to be configured explicitly.

bgp graceful-restart

To enable the Border Gateway Protocol (BGP) graceful restart capability globally for all BGP neighbors, use the **bgp graceful-restart** command in address family or in router configuration mode. To disable the BGP graceful restart capability globally for all BGP neighbors, use the **no** form of this command.

bgp graceful-restart [{**extended** | **restart-time** *seconds* | **stalepath-time** *seconds*}] [**all**]
no bgp graceful-restart

| Syntax Description | | |
|--------------------|--------------------------------------|---|
| | extended | (Optional) Enables BGP graceful restart extension. |
| | restart-time <i>seconds</i> | (Optional) Sets the maximum time period that the local router will wait for a graceful-restart-capable neighbor to return to normal operation after a restart event occurs. The default value for this argument is 120 seconds. The configurable range of values is from 1 to 3600 seconds. |
| | stalepath-time <i>seconds</i> | (Optional) Sets the maximum time period that the local router will hold stale paths for a restarting peer. All stale paths are deleted after this timer expires. The default value for this argument is 360 seconds. The configurable range of values is from 1 to 3600 seconds. |
| | all | (Optional) Enables BGP graceful restart capability for all address family modes. |

Command Default The following default values are used when this command is entered without any keywords or arguments:
restart-time : 120 seconds **stalepath-time**: 360 seconds



Note Changing the restart and stalepath timer values is not required to enable the BGP graceful restart capability. The default values are optimal for most network deployments, and these values should be adjusted only by an experienced network operator.

Command Modes Address-family configuration (config-router-af)
 Router configuration (config-router)

Command History *Table 4:*

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines The **bgp graceful-restart** command is used to enable or disable the graceful restart capability globally for all BGP neighbors in a BGP network. The graceful restart capability is negotiated between nonstop forwarding (NSF)-capable and NSF-aware peers in OPEN messages during session establishment. If the graceful restart

capability is enabled after a BGP session has been established, the session will need to be restarted with a hard reset.

The graceful restart capability is supported by NSF-capable and NSF-aware routers. A router that is NSF-capable can perform a stateful switchover (SSO) operation (graceful restart) and can assist restarting peers by holding routing table information during the SSO operation. A router that is NSF-aware functions like a router that is NSF-capable but cannot perform an SSO operation.

The BGP graceful restart capability is enabled by default when a supporting version of Cisco IOS software is installed. The default timer values for this feature are optimal for most network deployments. We recommend that they are adjusted only by experienced network operators. When adjusting the timer values, the restart timer should not be set to a value greater than the hold time that is carried in the OPEN message. If consecutive restart operations occur, routes (from a restarting router) that were previously marked as stale will be deleted.



Note Changing the restart and stalepath timer values is not required to enable the BGP graceful restart capability. The default values are optimal for most network deployments, and these values should be adjusted only by an experienced network operator.

Examples

In the following example, the BGP graceful restart capability is enabled:

```
Device#configure terminal
Device(config)#router bgp 65000
Device(config-router)#bgp graceful-restart
```

In the following example, the restart timer is set to 130 seconds:

```
Device#configure terminal
Device(config)#router bgp 65000
Device(config-router)#bgp graceful-restart restart-time 130
```

In the following example, the stalepath timer is set to 350 seconds:

```
Device#configure terminal
Device(config)#router bgp 65000
Device(config-router)#bgp graceful-restart stalepath-time 350
```

In the following example, the **extended** keyword is used:

```
Device#configure terminal
Device(config)#router bgp 65000
Device(config-router)#bgp graceful-restart extended
```

Related Commands

Table 5:

| Command | Description |
|------------------------------|--|
| show ip bgp | Displays entries in the BGP routing table. |
| show ip bgp neighbors | Displays information about the TCP and BGP connections to neighbors. |

clear proximity ip bgp

To reset Border Gateway Protocol (BGP) connections using hard or soft reconfiguration, use the **clear proximity ip bgp** command in privileged EXEC mode.

```
clear proximity ip bgp {* | all autonomous-system-number neighbor-address | peer-group group-name}
[ {in [prefix-filter] | out | slow | soft [ {in [prefix-filter] | out | slow} ] } ]
```

| Syntax | Description |
|-------------------------------------|---|
| * | Specifies that all current BGP sessions will be reset. |
| all | (Optional) Specifies the reset of all address family sessions. |
| <i>autonomous-system-number</i> | Number of the autonomous system in which all BGP peer sessions will be reset. Number in the range from 1 to 65535. <ul style="list-style-type: none"> In Cisco IOS Release 12.0(32)SY8, 12.0(33)S3, 12.2(33)SRE, 12.2(33)XNE, 12.2(33)SX11, 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. |
| <i>neighbor-address</i> | Specifies that only the identified BGP neighbor will be reset. The value for this argument can be an IPv4 or IPv6 address. |
| peer-group <i>group-name</i> | Specifies that only the identified BGP peer group will be reset. |
| in | (Optional) Initiates inbound reconfiguration. If neither the in nor out keywords are specified, both inbound and outbound sessions are reset. |
| prefix-filter | (Optional) Clears the existing outbound route filter (ORF) prefix list to trigger a new route refresh or soft reconfiguration, which updates the ORF prefix list. |
| out | (Optional) Initiates inbound or outbound reconfiguration. If neither the in nor out keywords are specified, both inbound and outbound sessions are reset. |
| slow | (Optional) Clears slow-peer status forcefully and moves it to original update group. |
| soft | (Optional) Initiates a soft reset. Does not tear down the session. |

Command Modes

Privileged EXEC (#)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

The **clear proximity ip bgp** command can be used to initiate a hard reset or soft reconfiguration. A hard reset tears down and rebuilds the specified peering sessions and rebuilds the BGP routing tables. A soft reconfiguration uses stored prefix information to reconfigure and activate BGP routing tables without tearing down existing peering sessions. Soft reconfiguration uses stored update information, at the cost of additional memory for storing the updates, to allow you to apply new BGP policy without disrupting the network. Soft reconfiguration can be configured for inbound or outbound sessions.



Note Due to the complexity of some of the keywords available for the **clear proximity ip bgp** command, some of the keywords are documented as separate commands. All of the complex keywords that are documented separately start with **clear ip bgp**. For example, for information on resetting BGP connections using hard or soft reconfiguration for all BGP neighbors in IPv4 address family sessions, refer to the **clear ip bgp ipv4** command.

Generating Updates from Stored Information

To generate new inbound updates from stored update information (rather than dynamically) without resetting the BGP session, you must preconfigure the local BGP router using the **neighbor soft-reconfiguration inbound** command. This preconfiguration causes the software to store all received updates without modification regardless of whether an update is accepted by the inbound policy. Storing updates is memory intensive and should be avoided if possible.

Outbound BGP soft configuration has no memory overhead and does not require any preconfiguration. You can trigger an outbound reconfiguration on the other side of the BGP session to make the new inbound policy take effect.

Use this command whenever any of the following changes occur:

- Additions or changes to the BGP-related access lists
- Changes to BGP-related weights
- Changes to BGP-related distribution lists
- Changes to BGP-related route maps

Dynamic Inbound Soft Reset

The route refresh capability, as defined in RFC 2918, allows the local router to reset inbound routing tables dynamically by exchanging route refresh requests to supporting peers. The route refresh capability does not store update information locally for non-disruptive policy changes. It instead relies on dynamic exchange with supporting peers. Route refresh is advertised through BGP capability negotiation. All BGP routers must support the route refresh capability.

To determine if a BGP router supports this capability, use the **show ip bgp neighbors** command. The following message is displayed in the output when the router supports the route refresh capability:

```
Received route refresh capability from peer.
```

If all BGP routers support the route refresh capability, use the **clear proximity ip bgp** command with the **in** keyword. You need not use the **soft** keyword, because soft reset is automatically assumed when the route refresh capability is supported.



Note After configuring a soft reset (inbound or outbound), it is normal for the BGP routing process to hold memory. The amount of memory that is held depends on the size of routing tables and the percentage of the memory chunks that are utilized. Partially used memory chunks will be used or released before more memory is allocated from the global router pool.

Examples

In the following example, a soft reconfiguration is initiated for the inbound session with the neighbor 10.100.0.1, and the outbound session is unaffected:

```
Device#clear proximity ip bgp 10.100.0.1 soft in
```

In the following example, the route refresh capability is enabled on the BGP neighbor routers and a soft reconfiguration is initiated for the inbound session with the neighbor 172.16.10.2, and the outbound session is unaffected:

```
Device#clear proximity ip bgp 172.16.10.2 in
```

In the following example, a hard reset is initiated for sessions with all routers in the autonomous system numbered 35700:

```
Device#clear proximity ip bgp 35700
```

In the following example, a hard reset is initiated for sessions with all routers in the 4-byte autonomous system numbered 65538 in asplain notation. This example requires Cisco IOS Release 12.0(32)SY8, 12.0(33)S3, 12.2(33)SRE, 12.2(33)XNE, 12.2(33)SX11, Cisco IOS XE Release 2.4, or a later release.

```
Device#clear proximity ip bgp 65538
```

In the following example, a hard reset is initiated for sessions with all routers in the 4-byte autonomous system numbered 1.2 in asdot notation. This example requires Cisco IOS Release 12.0(32)SY8, 12.0(32)S12, 12.2(33)SRE, 12.2(33)XNE, 12.2(33)SX11, 12.4(24)T, and Cisco IOS XE Release 2.3, or a later release.

```
Device#clear proximity ip bgp 1.2
```

Related Commands

| Command | Description |
|---|---|
| bgp slow-peer split-update-group dynamic permanent | Moves a dynamically detected slow peer to a slow update group. |
| clear ip bgp ipv4 | Resets BGP connections using hard or soft reconfiguration for IPv4 address family sessions. |
| clear ip bgp ipv6 | Resets BGP connections using hard or soft reconfiguration for IPv6 address family sessions. |

| Command | Description |
|--|--|
| clear ip bgp vpnv4 | Resets BGP connections using hard or soft reconfiguration for VPNv4 address family sessions. |
| clear ip bgp vpnv6 | Resets BGP connections using hard or soft reconfiguration for VPNv6 address family sessions. |
| neighbor slow-peer split-update-group dynamic permanent | Moves a dynamically detected slow peer to a slow update group. |
| neighbor soft-reconfiguration | Configures the Cisco IOS software to start storing updates. |
| router bgp | Configures the BGP routing process. |
| show ip bgp | Displays entries in the BGP routing table. |
| show ip bgp neighbors | Displays information about BGP and TCP connections to neighbors. |
| slow-peer split-update-group dynamic permanent | Moves a dynamically detected slow peer to a slow update group. |

default-information originate (OSPF)

To generate a default external route into an Open Shortest Path First (OSPF) routing domain, use the **default-information originate** command in router configuration or router address family topology configuration mode. To disable this feature, use the **no** form of this command.

default-information originate [**always**] [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-name*]

no default-information originate [**always**] [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-name*]

Syntax Description

| | |
|--------------------------------------|---|
| always | (Optional) Always advertises the default route regardless of whether the software has a default route. Note The always keyword includes the following exception when the route map is used. When a route map is used, the origination of the default route by OSPF is not bound to the existence of a default route in the routing table and the always keyword is ignored. |
| metric <i>metric-value</i> | (Optional) Metric used for generating the default route. If you omit a value and do not specify a value using the default-metric router configuration command, the default metric value is 10. The value used is specific to the protocol. |
| metric-type <i>type-value</i> | (Optional) External link type associated with the default route that is advertised into the OSPF routing domain. It can be one of the following values: <ul style="list-style-type: none"> • Type 1 external route. • Type 2 external route. The default is type 2 external route. |
| route-map <i>map-name</i> | (Optional) The routing process will generate the default route if the route map is satisfied. |

Command Default

This command is disabled by default. No default external route is generated into the OSPF routing domain.

Command Modes

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

Command History

| | |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
|------------------------------|------------------------------|

Usage Guidelines

Whenever you use the **redistribute** or the **default-information** router configuration command to redistribute routes into an OSPF routing domain, the Cisco IOS software automatically becomes an Autonomous System Boundary Router (ASBR). However, an ASBR does not, by default, generate a default route into the OSPF routing domain. The software must still have a default route for itself before it generates one, except when you have specified the **always** keyword.

When a route map is used, the origination of the default route by OSPF is not bound to the existence of a default route in the routing table.

Release 12.2(33)SRB

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

Examples

The following example specifies a metric of 100 for the default route that is redistributed into the OSPF routing domain and specifies an external metric type of 1:

```
router ospf 109
 redistribute eigrp 108 metric 100 subnets
 default-information originate metric 100 metric-type 1
```

Related Commands

| Command | Description |
|----------------------------|--|
| default-information | Accepts exterior or default information into Enhanced Interior Gateway Routing Protocol (EIGRP) processes. |
| default-metric | Sets default metric values for routes. |
| redistribute (IP) | Redistributes routes from one routing domain into another routing domain. |

default-metric (BGP)

To set a default metric for routes redistributed into Border Gateway Protocol (BGP), use the **default-metric** command in address family or router configuration mode. To remove the configured value and return BGP to default operation, use the **no** form of this command.

default-metric *number*
no default-metric *number*

| | |
|---------------------------|---|
| Syntax Description | <i>number</i> Default metric value applied to the redistributed route. The range of values for this argument is from 1 to 4294967295. |
|---------------------------|---|

| | |
|------------------------|---|
| Command Default | <p>The following is default behavior if this command is not configured or if the no form of this command is entered:</p> <ul style="list-style-type: none"> • The metric of redistributed interior gateway protocol (IGP) routes is set to a value that is equal to the interior BGP (iBGP) metric. • The metric of redistributed connected and static routes is set to 0. |
|------------------------|---|

When this command is enabled, the metric for redistributed connected routes is set to 0.

| | |
|----------------------|--|
| Command Modes | <p>Address family configuration (config-router-af)</p> <p>Router configuration (config-router)</p> |
|----------------------|--|

Command History

Table 6:

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

| | |
|-------------------------|---|
| Usage Guidelines | <p>The default-metric command is used to set the metric value for routes redistributed into BGP and can be applied to any external BGP (eBGP) routes received and subsequently advertised internally to iBGP peers.</p> <p>This value is the Multi Exit Discriminator (MED) that is evaluated by BGP during the best path selection process. The MED is a non-transitive value that is processed only within the local autonomous system and adjacent autonomous systems. The default metric is not set if the received route has a MED value.</p> |
|-------------------------|---|



| | |
|-------------|---|
| Note | When enabled, the default-metric command applies a metric value of 0 to redistributed connected routes. The default-metric command does not override metric values that are applied with the redistribute command. |
|-------------|---|

Examples

In the following example, a metric of 1024 is set for routes redistributed into BGP from OSPF:

```
Device(config)#router bgp 50000
Device(config-router)#address-family ipv4 unicast

Device(config-router-af)#default-metric 1024
```

```
Device(config-router-af)#redistribute ospf 10
Device(config-router-af)#end
```

In the following configuration and output examples, a metric of 300 is set for eBGP routes received and advertised internally to an iBGP peer.

```
Device(config)#router bgp 65501
Device(config-router)#no synchronization
Device(config-router)#bgp log-neighbor-changes
Device(config-router)#network 172.16.1.0 mask 255.255.255.0
Device(config-router)#neighbor 172.16.1.1 remote-as 65501
Device(config-router)#neighbor 172.16.1.1 soft-reconfiguration inbound
Device(config-router)#neighbor 192.168.2.2 remote-as 65502
Device(config-router)#neighbor 192.168.2.2 soft-reconfiguration inbound
Device(config-router)#default-metric 300
Device(config-router)#no auto-summary
```

After the above configuration, some routes are received from the eBGP peer at 192.168.2.2 as shown in the output from the **show ip bgp neighbors received-routes** command.

```
Device#show ip bgp neighbors 192.168.2.2 received-routes

BGP table version is 7, local router ID is 192.168.2.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
               Network      Next Hop          Metric LocPrf Weight Path
*> 172.17.1.0/24    192.168.2.2          0      100     0 i
```

After the received routes from the eBGP peer at 192.168.2.2 are advertised internally to iBGP peers, the output from the **show ip bgp neighbors received-routes** command shows that the metric (MED) has been set to 300 for these routes.

```
Device#show ip bgp neighbors 172.16.1.2 received-routes

BGP table version is 2, local router ID is 172.16.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
               Network      Next Hop          Metric LocPrf Weight Path
* i172.16.1.0/24    172.16.1.2          0      100     0 i
* i172.17.1.0/24    192.168.2.2        300    100     0 65502 i
Total number of prefixes 2
```

Related Commands

| Command | Description |
|--------------------------|---|
| redistribute (IP) | Redistributes routes from one routing domain into another routing domain. |

distance (OSPF)

To define an administrative distance, use the **distance** command in router configuration mode or VRF configuration mode. To remove the **distance** command and restore the system to its default condition, use the **no** form of this command.

```
distance weight
[ip-address wildcard-mask [access-list name]]
no distance weight ip-address wildcard-mask [access-list-name]
```

Syntax Description

| | |
|-------------------------|---|
| <i>weight</i> | Administrative distance. Range is 10 to 255. Used alone, the <i>weight</i> argument specifies a default administrative distance that the software uses when no other specification exists for a routing information source. Routes with a distance of 255 are not installed in the routing table. The table in the “Usage Guidelines” section lists the default administrative distances. |
| <i>ip-address</i> | (Optional) IP address in four-part dotted-decimal notation. |
| <i>wildcard-mask</i> | (Optional) Wildcard mask in four-part, dotted-decimal format. A bit set to 1 in the <i>wildcard-mask</i> argument instructs the software to ignore the corresponding bit in the address value. |
| <i>access-list-name</i> | (Optional) Name of an IP access list to be applied to incoming routing updates. |

Command Default

If this command is not specified, the administrative distance is the default. The table in the “Usage Guidelines” section lists the default administrative distances.

Command Modes

Router configuration (config-router)
VRF configuration (config-vrf)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

To use this command, you must be in a user group associated with a task group that includes the appropriate task IDs. If the user group assignment is preventing you from using a command contact your AAA administrator for assistance.

An administrative distance is an integer from 10 to 255. In general, the higher the value, the lower the trust rating. An administrative distance of 255 means that the routing information source cannot be trusted at all and should be ignored. Weight values are subjective; no quantitative method exists for choosing weight values.

If an access list is used with this command, it is applied when a network is being inserted into the routing table. This behavior allows you to filter networks based on the IP prefix supplying the routing information. For example, you could filter possibly incorrect routing information from networking devices not under your administrative control.

The order in which you enter **distance** commands can affect the assigned administrative distances, as shown in the “Examples” section. The following table lists default administrative distances.

Table 7: Default Administrative Distances

| Rate Source | Default Distance |
|-------------------------------|------------------|
| Connected interface | 0 |
| Static route out on interface | 0 |
| Static route to next hop | 1 |
| EIGRP summary route | 5 |
| External BGP | 20 |
| Internal EIGRP | 90 |
| OSPF | 110 |
| IS-IS | 115 |
| RIP version 1 and 2 | 120 |
| External EIGRP | 170 |
| Internal BGP | 200 |
| Unknown | 255 |

Task ID

| Task ID | Operations |
|---------|-------------|
| ospf | read, write |

Examples

In the following example, the **router ospf** command sets up Open Shortest Path First (OSPF) routing instance 1. The first **distance** command sets the default administrative distance to 255, which instructs the software to ignore all routing updates from networking devices for which an explicit distance has not been set. The second **distance** command sets the administrative distance for all devices on the network 192.168.40.0 to 90.

```
Device#configure terminal
Device(config)#router ospf 1
Device(config-ospf)#distance 255
Device(config-ospf)#distance 90 192.168.40.0 0.0.0.255
```

Related Commands

| Command | Description |
|----------------------|--|
| distance bgp | Allows the use of external, internal, and local administrative distances that could be a better route to a BGP node. |
| distance ospf | Allows the use of external, internal, and local administrative distances that could be a better route to an OSPF node. |

| Command | Description |
|--------------------|--------------------------------------|
| router ospf | Configures the OSPF routing process. |

eigrp log-neighbor-changes

To enable the logging of changes in Enhanced Interior Gateway Routing Protocol (EIGRP) neighbor adjacencies, use the **eigrp log-neighbor-changes** command in router configuration mode, address-family configuration mode, or service-family configuration mode. To disable the logging of changes in EIGRP neighbor adjacencies, use the **no** form of this command.

eigrp log-neighbor-changes
no eigrp log-neighbor-changes

Syntax Description

This command has no arguments or keywords.

Command Default

Adjacency changes are logged.

Command Modes

Router configuration (config-router) Address-family configuration (config-router-af) Service-family configuration (config-router-sf)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

This command enables the logging of neighbor adjacency changes to monitor the stability of the routing system and to help detect problems. Logging is enabled by default. To disable the logging of neighbor adjacency changes, use the **no** form of this command.

To enable the logging of changes for EIGRP address-family neighbor adjacencies, use the **eigrp log-neighbor-changes** command in address-family configuration mode.

To enable the logging of changes for EIGRP service-family neighbor adjacencies, use the **eigrp log-neighbor-changes** command in service-family configuration mode.

Examples

The following configuration disables logging of neighbor changes for EIGRP process 209:

```
Device(config)# router eigrp 209
Device(config-router)# no eigrp log-neighbor-changes
```

The following configuration enables logging of neighbor changes for EIGRP process 209:

```
Device(config)# router eigrp 209
Device(config-router)# eigrp log-neighbor-changes
```

The following example shows how to disable logging of neighbor changes for EIGRP address-family with autonomous-system 4453:

```
Device(config)# router eigrp virtual-name
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# no eigrp log-neighbor-changes
Device(config-router-af)# exit-address-family
```

The following configuration enables logging of neighbor changes for EIGRP service-family process 209:

```
Device(config)# router eigrp 209
Device(config-router)# service-family ipv4 autonomous-system 4453
Device(config-router-sf)# eigrp log-neighbor-changes
Device(config-router-sf)# exit-service-family
```

Related Commands

| Command | Description |
|-------------------------------|--|
| address-family (EIGRP) | Enters address-family configuration mode to configure an EIGRP routing instance. |
| exit-address-family | Exits address-family configuration mode. |
| exit-service-family | Exits service-family configuration mode. |
| router eigrp | Configures the EIGRP routing process. |
| service-family | Specifies service-family configuration mode. |

eigrp log-neighbor-warnings

To enable the logging of Enhanced Interior Gateway Routing Protocol (EIGRP) neighbor warning messages, use the **eigrp log-neighbor-warnings** command in router configuration mode, address-family configuration mode, or service-family configuration mode. To disable the logging of EIGRP neighbor warning messages, use the **no** form of this command.

eigrp log-neighbor-warnings [*seconds*]
no eigrp log-neighbor-warnings

Syntax Description

| | |
|----------------|--|
| <i>seconds</i> | (Optional) The time interval (in seconds) between repeated neighbor warning messages. The range is from 1 to 65535. The default is 10. |
|----------------|--|

Command Default

Neighbor warning messages are logged at 10-second intervals.

Command Modes

Router configuration (config-router) Address-family configuration (config-router-af) Service-family configuration (config-router-sf)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

When neighbor warning messages occur, they are logged by default. With this command, you can disable and enable neighbor warning messages, and you can configure the interval between repeated neighbor warning messages.

To enable the logging of warning messages for an EIGRP address family, use the **eigrp log-neighbor-warnings** command in address-family configuration mode.

To enable the logging of warning messages for an EIGRP service family, use the **eigrp log-neighbor-warnings** command in service-family configuration mode.

Examples

The following command will log neighbor warning messages for EIGRP process 209 and repeat the warning messages in 5-minute (300 seconds) intervals:

```
Device(config)# router eigrp 209
Device(config-router)# eigrp log-neighbor-warnings 300
```

The following example logs neighbor warning messages for the service family with autonomous system number 4453 and repeats the warning messages in five-minute (300 second) intervals:

```
Device(config)# router eigrp virtual-name
Device(config-router)# service-family ipv4 autonomous-system 4453
Device(config-router-sf)# eigrp log-neighbor-warnings 300
```

The following example logs neighbor warning messages for the address family with autonomous system number 4453 and repeats the warning messages in five-minute (300 second) intervals:

```
Device(config)# router eigrp virtual-name
```



```
Device(config-router)# address-family ipv4 autonomous-system 4453  
Device(config-router-af)# eigrp log-neighbor-warnings 300
```

Related Commands

| Command | Description |
|-------------------------------|--|
| address-family (EIGRP) | Enters address-family configuration mode to configure an EIGRP routing instance. |
| exit-address-family | Exits address-family configuration mode. |
| exit-service-family | Exits service-family configuration mode. |
| router eigrp | Configures the EIGRP routing process. |
| service-family | Specifies service-family configuration mode. |

ip authentication key-chain eigrp

To enable authentication of Enhanced Interior Gateway Routing Protocol (EIGRP) packets, use the **ip authentication key-chain eigrp** command in interface configuration mode. To disable such authentication, use the **no** form of this command.

ip authentication key-chain eigrp *as-number* *key-chain*

no ip authentication key-chain eigrp *as-number* *key-chain*

Syntax Description

| | |
|------------------|---|
| <i>as-number</i> | Autonomous system number to which the authentication applies. |
| <i>key-chain</i> | Name of the authentication key chain. |

Command Default

No authentication is provided for EIGRP packets.

Command Modes

Interface configuration (config-if) Virtual network interface (config-if-vnet)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Examples

The following example applies authentication to autonomous system 2 and identifies a key chain named SPORTS:

```
Device (config-if) #ip authentication key-chain eigrp 2 SPORTS
```

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 mode eigrp | Specifies the type of authentication used in EIGRP packets. |
| key | Identifies an authentication key on a key chain. |
| key chain | Enables authentication of 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. |

ip authentication mode eigrp

To specify the type of authentication used in Enhanced Interior Gateway Routing Protocol (EIGRP) packets, use the **ip authentication mode eigrp** command in interface configuration mode. To disable that type of authentication, use the **no** form of this command.

```
ip authentication mode eigrp as-number md5
no ip authentication mode eigrp as-number md5
```

| Syntax Description | |
|--------------------|---|
| <i>as-number</i> | Autonomous system number. |
| md5 | Keyed Message Digest 5 (MD5) authentication. |

Command Default No authentication is provided for EIGRP packets.

Command Modes Interface configuration (config-if) Virtual network interface (config-if-vnet)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines Configure authentication to prevent unapproved sources from introducing unauthorized or false routing messages. When authentication is configured, an MD5 keyed digest is added to each EIGRP packet in the specified autonomous system.

Examples

The following example configures the interface to use MD5 authentication in EIGRP packets in autonomous system 10:

```
Device(config-if) #ip authentication mode eigrp 10 md5
```

| 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 | Enables authentication of 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. |

ip bandwidth-percent eigrp

To configure the percentage of bandwidth that may be used by Enhanced Interior Gateway Routing Protocol (EIGRP) on an interface, use the **ip bandwidth-percent eigrp** command in interface configuration mode. To restore the default value, use the **no** form of this command.

```
ip bandwidth-percent eigrp as-number percent
no ip bandwidth-percent eigrp as-number percent
```

| Syntax Description | |
|--------------------|--|
| <i>as-number</i> | Autonomous system number. |
| <i>percent</i> | Percent of bandwidth that EIGRP may use. |

Command Default EIGRP may use 50 percent of available bandwidth.

Command Modes Interface configuration (config-if) Virtual network interface (config-if-vnet)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines EIGRP will use up to 50 percent of the bandwidth of a link, as defined by the **bandwidth** interface configuration command. This command may be used if some other fraction of the bandwidth is desired. Note that values greater than 100 percent may be configured. The configuration option may be useful if the bandwidth is set artificially low for other reasons.

Examples

The following example allows EIGRP to use up to 75 percent (42 kbps) of a 56-kbps serial link in autonomous system 209:

```
Device(config)#interface serial 0
Device(config-if)#bandwidth 56
Device(config-if)#ip bandwidth-percent eigrp 209 75
```

| Related Commands | Command | Description |
|------------------|------------------------------|--|
| | bandwidth (interface) | Sets a bandwidth value for an interface. |

ip cef load-sharing algorithm

To select a Cisco Express Forwarding load-balancing algorithm, use the **ip cef load-sharing algorithm** command in global configuration mode. To return to the default universal load-balancing algorithm, use the **no** form of this command.

ip cef load-sharing algorithm {**original** | [**universal** [*id*]]}
no ip cef load-sharing algorithm

| Syntax Description | original | Sets the load-balancing algorithm to the original algorithm based on a source and destination hash. |
|--------------------|-----------|---|
| | universal | Sets the load-balancing algorithm to the universal algorithm that uses a source and destination and an ID hash. |
| | <i>id</i> | (Optional) Fixed identifier. |

Command Default The universal load-balancing algorithm is selected by default. If you do not configure the fixed identifier for a load-balancing algorithm, the router automatically generates a unique ID.

Command Modes Global configuration (config)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines The original Cisco Express Forwarding load-balancing algorithm produced distortions in load sharing across multiple devices because of the use of the same algorithm on every device. When the load-balancing algorithm is set to universal mode, each device on the network can make a different load sharing decision for each source-destination address pair, and that resolves load-balancing distortions.

Examples The following example shows how to enable the Cisco Express Forwarding original load-balancing algorithm:

```
Device> enable
Device# configure terminal
Device(config)# ip cef load-sharing algorithm original
Device(config)# exit
```

| Related Commands | Command | Description |
|------------------|------------------------|--|
| | ip load-sharing | Enables load balancing for Cisco Express Forwarding. |

ip community-list

To configure a BGP community list and to control which routes are permitted or denied based on their community values, use the **ip community-list** command in global configuration mode. To delete the community list, use the **no** form of this command.

Standard Community Lists

```
ip community-list {standard | standard list-name} {deny | permit} [community-number] [AA:NN]
[internet] [local-as] [no-advertise] [no-export] [gshut]
no ip community-list {standard | standard list-name}
```

Expanded Community Lists

```
ip community-list {expanded | expanded list-name} {deny | permit} regex
no ip community-list {expanded | expanded list-name}
```

Syntax Description

| | |
|----------------------------------|---|
| <i>standard</i> | Standard community list number from 1 to 99 to identify one or more permit or deny groups of communities. |
| standard <i>list-name</i> | Configures a named standard community list. |
| deny | Denies routes that match the specified community or communities. |
| permit | Permits routes that match the specified community or communities. |
| <i>community-number</i> | (Optional) 32-bit number from 1 to 4294967200. A single community can be entered or multiple communities can be entered, each separated by a space. |

| | |
|---------------------|--|
| <i>AA :NN</i> | (Optional) Autonomous system number and network number entered in the 4-byte new community format. This value is configured with two 2-byte numbers separated by a colon. A number from 1 to 65535 can be entered for each 2-byte number. A single community can be entered or multiple communities can be entered, each separated by a space. |
| internet | (Optional) Specifies the Internet community. Routes with this community are advertised to all peers (internal and external). |
| local-as | (Optional) Specifies the local-as community. Routes with community are advertised to only peers that are part of the local autonomous system or to only peers within a subautonomous system of a confederation. These routes are not advertised to external peers or to other subautonomous systems within a confederation. |
| no-advertise | (Optional) Specifies the no-advertise community. Routes with this community are not advertised to any peer (internal or external). |
| no-export | (Optional) Specifies the no-export community. Routes with this community are advertised to only peers in the same autonomous system or to only other subautonomous systems within a confederation. These routes are not advertised to external peers. |

| | |
|----------------------------------|---|
| gshut | (Optional) Specifies the Graceful Shutdown (GSHUT) community. |
| <i>expanded</i> | Expanded community list number from 100 to 500 to identify one or more permit or deny groups of communities. |
| expanded <i>list-name</i> | Configures a named expanded community list. |
| <i>regexp</i> | Regular expression that is used to specify a pattern to match against an input string. Note Regular expressions can be used only with expanded community lists. |

Command Default BGP community exchange is not enabled by default.

Command Modes Global configuration (config)

Command History *Table 8:*

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines The **ip community-list** command is used to filter BGP routes based on one or more community values. BGP community values are configured as a 32-bit number (old format) or as a 4-byte number (new format). The new community format is enabled when the **ip bgp-community new-format** command is entered in global configuration mode. The new community format consists of a 4-byte value. The first two bytes represent the autonomous system number, and the trailing two bytes represent a user-defined network number. Named and numbered community lists are supported.

BGP community exchange is not enabled by default. The exchange of BGP community attributes between BGP peers is enabled on a per-neighbor basis with the **neighbor send-community** command. The BGP community attribute is defined in RFC 1997 and RFC 1998.

The Internet community is applied to all routes or prefixes by default, until any other community value is configured with this command or the **set community** command.

Use a route map to reference a community list and thereby apply policy routing or set values.

Community List Processing

Once a **permit** value has been configured to match a given set of communities, the community list defaults to an implicit deny for all other community values. Unlike an access list, it is feasible for a community list to contain only **deny** statements.

- When multiple communities are configured in the same **ip community-list** statement, a logical AND condition is created. All community values for a route must match the communities in the community list statement to satisfy an AND condition.
- When multiple communities are configured in separate **ip community-list** statements, a logical OR condition is created. The first list that matches a condition is processed.

Standard Community Lists

Standard community lists are used to configure well-known communities and specific community numbers. A maximum of 16 communities can be configured in a standard community list. If you attempt to configure more than 16 communities, the trailing communities that exceed the limit are not processed or saved to the running configuration file.

Expanded Community Lists

Expanded community lists are used to filter communities using a regular expression. Regular expressions are used to configure patterns to match community attributes. The order for matching using the * or + character is longest construct first. Nested constructs are matched from the outside in. Concatenated constructs are matched beginning at the left side. If a regular expression can match two different parts of an input string, it will match the earliest part first. For more information about configuring regular expressions, see the “Regular Expressions” appendix of the *Terminal Services Configuration Guide*.

Examples

In the following example, a standard community list is configured that permits routes from network 10 in autonomous system 50000:

```
Device(config)#ip community-list 1 permit 50000:10
```

In the following example, a standard community list is configured that permits only routes from peers in the same autonomous system or from subautonomous system peers in the same confederation:

```
Device(config)#ip community-list 1 permit no-export
```

In the following example, a standard community list is configured to deny routes that carry communities from network 40 in autonomous system 65534 and from network 60 in autonomous system 65412. This example shows a logical AND condition; all community values must match in order for the list to be processed.

```
Device(config)#ip community-list 2 deny 65534:40 65412:60
```

In the following example, a named, standard community list is configured that permits all routes within the local autonomous system or permits routes from network 20 in autonomous system 40000. This example shows a logical OR condition; the first match is processed.

```
Device(config)#ip community-list standard RED permit local-as  
Device(config)#ip community-list standard RED permit 40000:20
```

In the following example, a standard community list is configured that denies routes with the GSHUT community and permits routes with the local-AS community. This example shows a logical OR condition; the first match is processed.

```
Device(config)#ip community-list 18 deny gshut
Device(config)#ip community-list 18 permit local-as
```

In the following example, an expanded community list is configured that denies routes that carry communities from any private autonomous system:

```
Device(config)#ip community-list 500 deny _64[6-9][0-9][0-9]_|_65[0-9][0-9][0-9]_
```

In the following example, a named expanded community list is configured that denies routes from network 1 to 99 in autonomous system 50000:

```
Device(config)#ip community-list expanded BLUE deny 50000:[0-9][0-9]_
```

Related Commands

| Command | Description |
|-----------------------------------|---|
| match community | Defines a BGP community that must match the community of a route. |
| neighbor send-community | Allows BGP community exchange with a neighbor. |
| neighbor shutdown graceful | Configures the BGP Graceful Shutdown feature. |
| route-map (IP) | Defines the conditions for redistributing routes from one routing protocol into another, or enables policy routing. |
| set community | Sets the BGP communities attribute. |
| set comm-list delete | Removes communities from the community attribute of an inbound or outbound update. |
| show ip bgp community | Displays routes that belong to specified BGP communities. |
| show ip bgp regexp | Displays routes that match a locally configured regular expression. |

ip prefix-list

To create a prefix list or to add a prefix-list entry, use the **ip prefix-list** command in global configuration mode. To delete a prefix-list entry, use the **no** form of this command.

```
ip prefix-list {list-name [seq number] {deny | permit} network/length [ge ge-length] [le le-length]
| description description | sequence-number}
no ip prefix-list {list-name [seq number] [{deny | permit} network/length [ge ge-length] [le
le-length]} | description description | sequence-number}
```

Syntax Description

| | |
|-------------------------|--|
| <i>list-name</i> | Configures a name to identify the prefix list. Do not use the word “detail” or “summary” as a list name because they are keywords in the show ip prefix-list command. |
| seq | (Optional) Applies a sequence number to a prefix-list entry. |
| <i>number</i> | (Optional) Integer from 1 to 4294967294. If a sequence number is not entered when configuring this command, default sequence numbering is applied to the prefix list. The number 5 is applied to the first prefix entry, and subsequent unnumbered entries are incremented by 5. |
| deny | Denies access for a matching condition. |
| permit | Permits access for a matching condition. |
| <i>network / length</i> | Configures the network address and the length of the network mask in bits. The network number can be any valid IP address or prefix. The bit mask can be a number from 1 to 32. |
| ge | (Optional) Specifies the lesser value of a range (the “from” portion of the range description) by applying the <i>ge-length</i> argument to the range specified. Note The ge keyword represents the greater than or equal to operator. |
| <i>ge-length</i> | (Optional) Represents the minimum prefix length to be matched. |
| le | (Optional) Specifies the greater value of a range (the “to” portion of the range description) by applying the <i>le-length</i> argument to the range specified. Note The le keyword represents the less than or equal to operator. |
| <i>le-length</i> | (Optional) Represents the maximum prefix length to be matched. |
| description | (Optional) Configures a descriptive name for the prefix list. |
| <i>description</i> | (Optional) Descriptive name of the prefix list, from 1 to 80 characters in length. |
| sequence-number | (Optional) Enables or disables the use of sequence numbers for prefix lists. |

Command Default

No prefix lists or prefix-list entries are created.

Command Modes

Global configuration (config)

Command History

Table 9:

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

Use the **ip prefix-list** command to configure IP prefix filtering. Prefix lists are configured with **permit** or **deny** keywords to either permit or deny a prefix based on a matching condition. An implicit deny is applied to traffic that does not match any prefix-list entry.

A prefix-list entry consists of an IP address and a bit mask. The IP address can be for a classful network, a subnet, or a single host route. The bit mask is a number from 1 to 32.

Prefix lists are configured to filter traffic based on a match of an exact prefix length or a match within a range when the **ge** and **le** keywords are used. The **ge** and **le** keywords are used to specify a range of prefix lengths and provide more flexible configuration than using only the *network/length* argument. A prefix list is processed using an exact match when neither the **ge** nor **le** keyword is specified. If only the **ge** value is specified, the range is the value entered for the **ge ge-length** argument to a full 32-bit length. If only the **le** value is specified, the range is from the value entered for the *network/length* argument to the **le le-length** argument. If both the **ge ge-length** and **le le-length** keywords and arguments are entered, the range is between the values used for the *ge-length* and *le-length* arguments.

The following formula shows this behavior:

$$\text{length} < \mathbf{ge} \text{ ge-length} < \mathbf{le} \text{ le-length} \leq 32$$

If the **seq** keyword is configured without a sequence number, the default sequence number is 5. In this scenario, the first prefix-list entry is assigned the number 5 and subsequent prefix list entries increment by 5. For example, the next two entries would have sequence numbers 10 and 15. If a sequence number is entered for the first prefix list entry but not for subsequent entries, the subsequent entry numbers increment by 5. For example, if the first configured sequence number is 3, subsequent entries will be 8, 13, and 18. Default sequence numbers can be suppressed by entering the **no ip prefix-list** command with the **seq** keyword.

Evaluation of a prefix list starts with the lowest sequence number and continues down the list until a match is found. When an IP address match is found, the permit or deny statement is applied to that network and the remainder of the list is not evaluated.



Tip For best performance, the most frequently processed prefix list statements should be configured with the lowest sequence numbers. The **seq number** keyword and argument can be used for resequencing.

A prefix list is applied to inbound or outbound updates for a specific peer by entering the **neighbor prefix-list** command. Prefix list information and counters are displayed in the output of the **show ip prefix-list** command. Prefix-list counters can be reset by entering the **clear ip prefix-list** command.

Examples

In the following example, a prefix list is configured to deny the default route 0.0.0.0/0:

```
Device(config)#ip prefix-list RED deny 0.0.0.0/0
```

In the following example, a prefix list is configured to permit traffic from the 172.16.1.0/24 subnet:

```
Device(config)#ip prefix-list BLUE permit 172.16.1.0/24
```

In the following example, a prefix list is configured to permit routes from the 10.0.0.0/8 network that have a mask length that is less than or equal to 24 bits:

```
Device(config)#ip prefix-list YELLOW permit 10.0.0.0/8 le 24
```

In the following example, a prefix list is configured to deny routes from the 10.0.0.0/8 network that have a mask length that is greater than or equal to 25 bits:

```
Device(config)#ip prefix-list PINK deny 10.0.0.0/8 ge 25
```

In the following example, a prefix list is configured to permit routes from any network that have a mask length from 8 to 24 bits:

```
Device(config)#ip prefix-list GREEN permit 0.0.0.0/0 ge 8 le 24
```

In the following example, a prefix list is configured to deny any route with any mask length from the 10.0.0.0/8 network:

```
Device(config)#ip prefix-list ORANGE deny 10.0.0.0/8 le 32
```

Related Commands

| Command | Description |
|-----------------------------------|--|
| clear ip prefix-list | Resets the prefix list entry counters. |
| ip prefix-list description | Adds a text description of a prefix list. |
| ip prefix-list sequence | Enables or disables default prefix-list sequencing. |
| 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. |
| neighbor prefix-list | Filters routes from the specified neighbor using a prefix list. |
| show ip prefix-list | Displays information about a prefix list or prefix list entries. |

ip hello-interval eigrp

To configure the hello interval for an Enhanced Interior Gateway Routing Protocol (EIGRP) process, use the **ip hello-interval eigrp** command in interface configuration mode. To restore the default value, use the **no** form of this command.

```
ip hello-interval eigrp as-number seconds
no ip hello-interval eigrp as-number [seconds]
```

Syntax Description

| | |
|------------------|--|
| <i>as-number</i> | Autonomous system number. |
| <i>seconds</i> | Hello interval (in seconds). The range is from 1 to 65535. |

Command Default

The hello interval for low-speed, nonbroadcast multiaccess (NBMA) networks is 60 seconds and 5 seconds for all other networks.

Command Modes

Interface configuration (config-if) Virtual network interface (config-if-vnet)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

The default of 60 seconds applies only to low-speed, NBMA media. Low speed is considered to be a rate of T1 or slower, as specified with the **bandwidth** interface configuration command. Note that for the purposes of EIGRP, Frame Relay and Switched Multimegabit Data Service (SMDS) networks may be considered to be NBMA. These networks are considered NBMA if the interface has not been configured to use physical multicasting; otherwise, they are considered not to be NBMA.

Examples

The following example sets the hello interval for Ethernet interface 0 to 10 seconds:

```
Device(config)#interface ethernet 0
Device(config-if)#ip hello-interval eigrp 109 10
```

Related Commands

| Command | Description |
|------------------------------|---|
| bandwidth (interface) | Sets a bandwidth value for an interface. |
| ip hold-time eigrp | Configures the hold time for a particular EIGRP routing process designated by the autonomous system number. |

ip hold-time eigrp

To configure the hold time for an Enhanced Interior Gateway Routing Protocol (EIGRP) process, use the **ip hold-time eigrp** command in interface configuration mode. To restore the default value, use the **no** form of this command.

ip hold-time eigrp *as-number seconds*
no ip hold-time eigrp *as-number seconds*

| Syntax Description | |
|--------------------|---|
| <i>as-number</i> | Autonomous system number. |
| <i>seconds</i> | Hold time (in seconds). The range is from 1 to 65535. |

Command Default The EIGRP hold time is 180 seconds for low-speed, nonbroadcast multiaccess (NBMA) networks and 15 seconds for all other networks.

Command Modes Interface configuration (config-if) Virtual network interface (config-if-vnet)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines On very congested and large networks, the default hold time might not be sufficient time for all routers and access servers to receive hello packets from their neighbors. In this case, you may want to increase the hold time.

We recommend that the hold time be at least three times the hello interval. If a router does not receive a hello packet within the specified hold time, routes through this router are considered unavailable.

Increasing the hold time delays route convergence across the network.

The default of 180 seconds hold time and 60 seconds hello interval apply only to low-speed, NBMA media. Low speed is considered to be a rate of T1 or slower, as specified with the **bandwidth** interface configuration command.

Examples

The following example sets the hold time for Ethernet interface 0 to 40 seconds:

```
Device(config)#interface ethernet 0
Device(config-if)#ip hold-time eigrp 109 40
```

| Related Commands | Command | Description |
|------------------|--------------------------------|--|
| | bandwidth (interface) | Sets a bandwidth value for an interface. |
| | ip hello-interval eigrp | Configures the hello interval for the EIGRP routing process designated by an autonomous system number. |

ip load-sharing

To enable load balancing for Cisco Express Forwarding on an interface, use the **ip load-sharing** command in interface configuration mode. To disable load balancing for Cisco Express Forwarding on the interface, use the **no** form of this command.

```
ip load-sharing { per-destination }
no ip load-sharing
```

Syntax Description

| | |
|------------------------|---|
| per-destination | Enables per-destination load balancing for Cisco Express Forwarding on the interface. |
|------------------------|---|

Command Default

Per-destination load balancing is enabled by default when you enable Cisco Express Forwarding.

Command Modes

Interface configuration (config-if)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

Per-destination load balancing allows the device to use multiple, equal-cost paths to achieve load sharing. Packets for a given source-destination host pair are guaranteed to take the same path, even if multiple, equal-cost paths are available. Traffic for different source-destination host pairs tends to take different paths.

Examples

The following example shows how to enable per-destination load balancing:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# ip load-sharing per-destination
```


ip network-broadcast

To receive and accept the network-prefix-directed broadcast packets, configure the **ip network-broadcast** command at the interface of the device.

```
ip network-broadcast
```

Syntax Description This command has no arguments or keywords.

Command Default This command is disabled by default.

Command Modes Interface configuration (config-if)

| Command History | Release | Modification |
|-----------------|-------------------------------|------------------------------|
| | Cisco IOS XE Amsterdam 17.3.1 | This command was introduced. |

Usage Guidelines Configure the **ip network-broadcast** command at the ingress interface before configuring the **ip directed-broadcast** command at the egress interface. This ensures that the network-prefix-directed broadcast packets are received and accepted.

The **ip network-broadcast** command is disabled by default. If you do not configure this command, the network-prefix-directed broadcast packets are silently discarded.

Example

The following example shows how to enable the network to accept the network-prefix-directed broadcast packets at ingress and then configure the directed broadcast-to-physical broadcast translation on the egress interface.

```
Device# configure terminal
Device(config)#interface gigabitethernet 1/0/2
Device(config-if)#ip network-broadcast
Device(config-if)#exit
Device(config)#interface gigabitethernet 1/0/3
Device(config-if)#ip directed-broadcast
Device(config-if)#exit
```

ip ospf database-filter all out

To filter outgoing link-state advertisements (LSAs) to an Open Shortest Path First (OSPF) interface, use the **ip ospf database-filter all out** command in interface or virtual network interface configuration modes. To restore the forwarding of LSAs to the interface, use the **no** form of this command.

```
ip ospf database-filter all out [disable]
no ip ospf database-filter all out
```

Syntax Description

| | |
|----------------|--|
| disable | (Optional) Disables the filtering of outgoing LSAs to an OSPF interface; all outgoing LSAs are flooded to the interface. |
| Note | This keyword is available only in virtual network interface mode. |

Command Default

This command is disabled by default. All outgoing LSAs are flooded to the interface.

Command Modes

Interface configuration (config-if)

Virtual network interface (config-if-vnet)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

This command performs the same function that the **neighbor database-filter** command performs on a neighbor basis.

If the **ip ospf database-filter all out** command is enabled for a virtual network and you want to disable it, use the **disable** keyword in virtual network interface configuration mode.

Examples

The following example prevents filtering of OSPF LSAs to broadcast, nonbroadcast, or point-to-point networks reachable through Ethernet interface 0:

```
Device(config)#interface ethernet 0
Device(config-if)#ip ospf database-filter all out
```

Related Commands

| Command | Description |
|---------------------------------|--|
| neighbor database-filter | Filters outgoing LSAs to an OSPF neighbor. |

ip ospf name-lookup

To configure Open Shortest Path First (OSPF) to look up Domain Name System (DNS) names for use in all OSPF **show EXEC** command displays, use the **ip ospf name-lookup** command in global configuration mode. To disable this function, use the **no** form of this command.

ip ospf name-lookup
no ip ospf name-lookup

Syntax Description This command has no arguments or keywords.

Command Default This command is disabled by default.

Command Modes Global configuration

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines This command makes it easier to identify a router because the router is displayed by name rather than by its router ID or neighbor ID.

Examples

The following example configures OSPF to look up DNS names for use in all OSPF **show EXEC** command displays:

```
Device(config)#ip ospf name-lookup
```

ip split-horizon eigrp

To enable Enhanced Interior Gateway Routing Protocol (EIGRP) split horizon, use the **ip split-horizon eigrp** command in interface configuration mode. To disable split horizon, use the **no** form of this command.

ip split-horizon eigrp *as-number*
no ip split-horizon eigrp *as-number*

Syntax Description

| | |
|------------------|---------------------------|
| <i>as-number</i> | Autonomous system number. |
|------------------|---------------------------|

Command Default

The behavior of this command is enabled by default.

Command Modes

Interface configuration (config-if)
 Virtual network interface (config-if-vnet)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

Use the **no ip split-horizon eigrp** command to disable EIGRP split horizon in your configuration.

Examples

The following is an example of how to enable EIGRP split horizon:

```
Device(config-if)#ip split-horizon eigrp 101
```

Related Commands

| Command | Description |
|-------------------------------|--|
| ip split-horizon (RIP) | Enables the split horizon mechanism. |
| neighbor (EIGRP) | Defines a neighboring router with which to exchange routing information. |

ip summary-address eigrp

To configure address summarization for the Enhanced Interior Gateway Routing Protocol (EIGRP) on a specified interface, use the **ip summary-address eigrp** command in interface configuration or virtual network interface configuration mode. To disable the configuration, use the **no** form of this command.

```
ip summary-address eigrp as-number ip-address mask [admin-distance] [leak-map name]  
no ip summary-address eigrp as-number ip-address mask
```

Syntax Description

| | |
|-----------------------------|--|
| <i>as-number</i> | Autonomous system number. |
| <i>ip-address</i> | Summary IP address to apply to an interface. |
| <i>mask</i> | Subnet mask. |
| <i>admin-distance</i> | (Optional) Administrative distance. Range: 0 to 255. Note Starting with Cisco IOS XE Release 3.2S, the <i>admin-distance</i> argument was removed. Use the summary-metric command to configure the administrative distance. |
| leak-map <i>name</i> | (Optional) Specifies the route-map reference that is used to configure the route leaking through the summary. |

Command Default

- An administrative distance of 5 is applied to EIGRP summary routes.
- EIGRP automatically summarizes to the network level, even for a single host route.
- No summary addresses are predefined.
- The default administrative distance metric for EIGRP is 90.

Command Modes

Interface configuration (config-if)

Virtual network interface configuration (config-if-vnet)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

The **ip summary-address eigrp** command is used to configure interface-level address summarization. EIGRP summary routes are given an administrative-distance value of 5. The administrative-distance metric is used to advertise a summary without installing it in the routing table.

By default, EIGRP summarizes subnet routes to the network level. The **no auto-summary** command can be entered to configure the subnet-level summarization.

The summary address is not advertised to the peer if the administrative distance is configured as 255.

EIGRP Support for Leaking Routes

Configuring the **leak-map** keyword allows a component route that would otherwise be suppressed by the manual summary to be advertised. Any component subset of the summary can be leaked. A route map and access list must be defined to source the leaked route.

The following is the default behavior if an incomplete configuration is entered:

- If the **leak-map** keyword is configured to reference a nonexistent route map, the configuration of this keyword has no effect. The summary address is advertised but all component routes are suppressed.
- If the **leak-map** keyword is configured but the access list does not exist or the route map does not reference the access list, the summary address and all component routes are advertised.

If you are configuring a virtual-network trunk interface and you configure the **ip summary-address eigrp** command, the *admin-distance* value of the command is not inherited by the virtual networks running on the trunk interface because the administrative distance option is not supported in the **ip summary-address eigrp** command on virtual network subinterfaces.

Examples

The following example shows how to configure an administrative distance of 95 on Ethernet interface 0/0 for the 192.168.0.0/16 summary address:

```
Device(config)#router eigrp 1
Device(config-router)#no auto-summary
Device(config-router)#exit
Device(config)#interface Ethernet 0/0
Device(config-if)#ip summary-address eigrp 1 192.168.0.0 255.255.0.0 95
```

The following example shows how to configure the 10.1.1.0/24 subnet to be leaked through the 10.2.2.0 summary address:

```
Device(config)#router eigrp 1
Device(config-router)#exit
Device(config)#access-list 1 permit 10.1.1.0 0.0.0.255
Device(config)#route-map LEAK-10-1-1 permit 10
Device(config-route-map)#match ip address 1
Device(config-route-map)#exit
Device(config)#interface Serial 0/0
Device(config-if)#ip summary-address eigrp 1 10.2.2.0 255.0.0.0 leak-map LEAK-10-1-1
Device(config-if)#end
```

The following example configures GigabitEthernet interface 0/0/0 as a virtual network trunk interface:

```
Device(config)#interface gigabitethernet 0/0/0
Device(config-if)#vnet global
Device(config-if-vnet)#ip summary-address eigrp 1 10.3.3.0 255.0.0.0 33
```

Related Commands

| Command | Description |
|-----------------------------|---|
| auto-summary (EIGRP) | Configures automatic summarization of subnet routes to network-level routes (default behavior). |
| summary-metric | Configures fixed metrics for an EIGRP summary aggregate address. |

ip route static bfd

To specify static route bidirectional forwarding detection (BFD) neighbors, use the **ip route static bfd** command in global configuration mode. To remove a static route BFD neighbor, use the **no** form of this command

```
ip route static bfd { interface-type interface-number ip-address | vrf vrf-name } [group group-name]
[passive] [unassociate]
no ip route static bfd { interface-type interface-number ip-address | vrf vrf-name } [group group-name]
[passive] [unassociate]
```

| Syntax Description | | |
|--------------------|--|---|
| | <i>interface-type interface-number</i> | Interface type and number. |
| | <i>ip-address</i> | IP address of the gateway, in A.B.C.D format. |
| | vrf <i>vrf-name</i> | Specifies Virtual Routing and Forwarding (VRF) instance and the destination vrf name. |
| | group <i>group-name</i> | (Optional) Assigns a BFD group. The <i>group-name</i> is a character string of up to 32 characters specifying the BFD group name. |
| | unassociate | (Optional) Unassociates the static route configured for a BFD. |

Command Default No static route BFD neighbors are specified.

Command Modes Global configuration (config)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines Use the **ip route static bfd** command to specify static route BFD neighbors. All static routes that have the same interface and gateway specified in the configuration share the same BFD session for reachability notification.

All static routes that specify the same values for the *interface-type*, *interface-number*, and *ip-address* arguments will automatically use BFD to determine gateway reachability and take advantage of fast failure detection.

The **group** keyword assigns a BFD group. The static BFD configuration is added to the VPN routing and forwarding (VRF) instance with which the interface is associated. The **passive** keyword specifies the passive member of the group. Adding static BFD in a group without the **passive** keyword makes the BFD an active member of the group. A static route should be tracked by the active BFD configuration in order to trigger a BFD session for the group. To remove all the static BFD configurations (active and passive) of a specific group, use the **no ip route static bfd** command and specify the BFD group name.

The **unassociate** keyword specifies that a BFD neighbor is not associated with static route, and the BFD sessions are requested if an interface has been configured with BFD. This is useful in bringing up a BFDv4 session in the absence of an IPv4 static route. If the unassociate keyword is not provided, then the IPv4 static routes are associated with BFD sessions.

BFD requires that BFD sessions are initiated on both endpoint devices. Therefore, this command must be configured on each endpoint device.

The BFD static session on a switch virtual interface (SVI) is established only after the **bfd interval milliseconds min_rx milliseconds multiplier multiplier-value** command is disabled and enabled on that SVI.

To enable the static BFD sessions, perform the following steps:

1. Enable BFD timers on the SVI.

```
bfd interval milliseconds min_rx milliseconds multiplier multiplier-value
```

2. Enable BFD for the static IP route

```
ip route static bfd interface-type interface-number ip-address
```

3. Disable and enable the BFD timers on the SVI again.

```
no bfd interval milliseconds min_rx milliseconds multiplier multiplier-value
```

```
bfd interval milliseconds min_rx milliseconds multiplier multiplier-value
```

Examples

The following example shows how to configure BFD for all static routes through a specified neighbor, group, and active member of the group:

```
Device#configuration terminal
Device(config)#ip route static bfd GigabitEthernet 1/0/1 10.1.1.1 group group1
```

The following example shows how to configure BFD for all static routes through a specified neighbor, group, and passive member of the group:

```
Device#configuration terminal
Device(config)#ip route static bfd GigabitEthernet 1/0/1 10.2.2.2 group group1 passive
```

The following example shows how to configure BFD for all static routes in an unassociated mode without the group and passive keywords:

```
Device#configuration terminal
Device(config)#ip route static bfd GigabitEthernet 1/0/1 10.2.2.2 unassociate
```


ipv6 route static bfd

To specify static route Bidirectional Forwarding Detection for IPv6 (BFDv6) neighbors, use the **ipv6 route static bfd** command in global configuration mode. To remove a static route BFDv6 neighbor, use the **no** form of this command

ipv6 route static bfd [*vrf vrf-name*] *interface-type interface-number ipv6-address* [**unassociated**]
no ipv6 route static bfd

| Syntax Description | | |
|--------------------|--|--|
| | <i>vrf vrf-name</i> | (Optional) Name of the virtual routing and forwarding (VRF) instance by which static routes should be specified. |
| | <i>interface-type interface-number</i> | Interface type and number. |
| | <i>ipv6-address</i> | IPv6 address of the neighbor. |
| | unassociated | (Optional) Moves a static BFD neighbor from associated mode to unassociated mode. |

Command Default No static route BFDv6 neighbors are specified.

Command Modes Global configuration (config)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines Use the `ipv6 route static bfd` command to specify static route neighbors. All of the static routes that have the same interface and gateway specified in the configuration share the same BFDv6 session for reachability notification. BFDv6 requires that BFDv6 sessions are initiated on both endpoint routers. Therefore, this command must be configured on each endpoint router. An IPv6 static BFDv6 neighbor must be fully specified (with the interface and the neighbor address) and must be directly attached.

All static routes that specify the same values for `vrf vrf-name`, `interface-type interface-number`, and `ipv6-address` will automatically use BFDv6 to determine gateway reachability and take advantage of fast failure detection.

Examples

The following example creates a neighbor on Ethernet interface 0/0 with an address of 2001::1:

```
Device#configuration terminal
Device(config)#ipv6 route static bfd ethernet 0/0 2001::1
```

The following example converts the neighbor to unassociated mode:

```
Device#configuration terminal
Device(config)#ipv6 route static bfd ethernet 0/0 2001::1 unassociated
```

metric weights (EIGRP)

To tune the Enhanced Interior Gateway Routing Protocol (EIGRP) metric calculations, use the **metric weights** command in router configuration mode or address family configuration mode. To reset the values to their defaults, use the **no** form of this command.

Router Configuration

```
metric weights tos k1 k2 k3 k4 k5
no metric weights
```

Address Family Configuration

```
metric weights tos [k1 [k2 [k3 [k4 [k5 [k6]]]]]]
no metric weights
```

Syntax Description

| | |
|--------------------------|---|
| <i>tos</i> | Type of service. This value must always be zero. |
| <i>k1 k2 k3 k4 k5 k6</i> | <p>(Optional) Constants that convert an EIGRP metric vector into a scalar quantity. Valid values are 0 to 255. Given below are the default values:</p> <ul style="list-style-type: none"> • <i>k1</i>: 1 • <i>k2</i>: 0 • <i>k3</i>: 1 • <i>k4</i>: 0 • <i>k5</i>: 0 • <i>k6</i>: 0 <p>Note In address family configuration mode, if the values are not specified, default values are configured. The <i>k6</i> argument is supported only in address family configuration mode.</p> |

Command Default

EIGRP metric K values are set to their default values.

Command Modes

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

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

Use this command to alter the default behavior of EIGRP routing and metric computation and to allow the tuning of the EIGRP metric calculation for a particular type of service (ToS).

If *k5* equals 0, the composite EIGRP metric is computed according to the following formula:

$$\text{metric} = [k1 * \text{bandwidth} + (k2 * \text{bandwidth}) / (256 - \text{load}) + k3 * \text{delay} + K6 * \text{extended metrics}]$$

If k5 does not equal zero, an additional operation is performed:

$$\text{metric} = \text{metric} * [\text{k5}/(\text{reliability} + \text{k4})]$$

$$\text{Scaled Bandwidth} = 10^7 / \text{minimum interface bandwidth (in kilobits per second)} * 256$$

Delay is in tens of microseconds for classic mode and pico seconds for named mode. In classic mode, a delay of hexadecimal FFFFFFFF (decimal 4294967295) indicates that the network is unreachable. In named mode, a delay of hexadecimal FFFFFFFFFF (decimal 281474976710655) indicates that the network is unreachable.

Reliability is given as a fraction of 255. That is, 255 is 100 percent reliability or a perfectly stable link.

Load is given as a fraction of 255. A load of 255 indicates a completely saturated link.

Examples

The following example shows how to set the metric weights to slightly different values than the defaults:

```
Device(config)#router eigrp 109
Device(config-router)#network 192.168.0.0
Device(config-router)#metric weights 0 2 0 2 0 0
```

The following example shows how to configure an address-family metric weight to ToS: 0; K1: 2; K2: 0; K3: 2; K4: 0; K5: 0; K6:1:

```
Device(config)#router eigrp virtual-name
Device(config-router)#address-family ipv4 autonomous-system 4533
Device(config-router-af)#metric weights 0 2 0 2 0 0 1
```

Related Commands

| Command | Description |
|-------------------------------|--|
| address-family (EIGRP) | Enters address family configuration mode to configure an EIGRP routing instance. |
| bandwidth (interface) | Sets a bandwidth value for an interface. |
| delay (interface) | Sets a delay value for an interface. |
| ipv6 router eigrp | Configures an IPv6 EIGRP routing process. |
| metric holddown | Keeps new EIGRP routing information from being used for a certain period of time. |
| metric maximum-hops | Causes IP routing software to advertise routes with a hop count higher than what is specified by the command (EIGRP only) as unreachable routes. |
| router eigrp | Configures an EIGRP routing process. |

neighbor advertisement-interval

To set the minimum route advertisement interval (MRAI) between the sending of BGP routing updates, use the **neighbor advertisement-interval** command in address family or router configuration mode. To restore the default value, use the **no** form of this command.

neighbor {*ip-address**peer-group-name*} **advertisement-interval** *seconds*

no neighbor {*ip-address**peer-group-name*} **advertisement-interval** *seconds*

Syntax Description

| | |
|------------------------|---|
| <i>ip-address</i> | IP address of the neighbor. |
| <i>peer-group-name</i> | Name of a BGP peer group. |
| <i>seconds</i> | Time (in seconds) is specified by an integer ranging from 0 to 600. |

Command Default

eBGP sessions not in a VRF: 30 seconds

eBGP sessions in a VRF: 0 seconds

iBGP sessions: 0 seconds

Command Modes

Router configuration (config-router)

Command History

Table 10:

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

When the MRAI is equal to 0 seconds, BGP routing updates are sent as soon as the BGP routing table changes.

If you specify a BGP peer group by using the *peer-group-name* argument, all the members of the peer group will inherit the characteristic configured with this command.

Examples

The following router configuration mode example sets the minimum time between sending BGP routing updates to 10 seconds:

```
router bgp 5
 neighbor 10.4.4.4 advertisement-interval 10
```

The following address family configuration mode example sets the minimum time between sending BGP routing updates to 10 seconds:

```
router bgp 5
 address-family ipv4 unicast
 neighbor 10.4.4.4 advertisement-interval 10
```

Related Commands

| Command | Description |
|---------------------------------------|--|
| 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. |
| neighbor peer-group (creating) | Creates a BGP peer group. |

neighbor default-originate

To allow a BGP speaker (the local router) to send the default route 0.0.0.0 to a neighbor for use as a default route, use the **neighbor default-originate** command in address family or router configuration mode. To send no route as a default, use the **no** form of this command.

```
neighbor {ip-addresspeer-group-name} default-originate [route-map map-name]  
no neighbor {ip-addresspeer-group-name} default-originate [route-map map-name]
```

| Syntax Description | | |
|--------------------|----------------------------------|--|
| | <i>ip-address</i> | IP address of the neighbor. |
| | <i>peer-group-name</i> | Name of a BGP peer group. |
| | route-map <i>map-name</i> | (Optional) Name of the route map. The route map allows route 0.0.0.0 to be injected conditionally. |

Command Default No default route is sent to the neighbor.

Command Modes Address family configuration (config-router-af)
Router configuration (config-router)

Command History *Table 11:*

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines This command does not require the presence of 0.0.0.0 in the local router. When used with a route map, the default route 0.0.0.0 is injected if the route map contains a **match ip address** clause and there is a route that matches the IP access list exactly. The route map can contain other match clauses also.

You can use standard or extended access lists with the **neighbor default-originate** command.

Examples

In the following router configuration example, the local router injects route 0.0.0.0 to the neighbor 172.16.2.3 unconditionally:

```
router bgp 109
network 172.16.0.0
neighbor 172.16.2.3 remote-as 200
neighbor 172.16.2.3 default-originate
```

In the following example, the local router injects route 0.0.0.0 to the neighbor 172.16.2.3 only if there is a route to 192.168.68.0 (that is, if a route with any mask exists, such as 255.255.255.0 or 255.255.0.0):

```
router bgp 109
network 172.16.0.0
neighbor 172.16.2.3 remote-as 200
neighbor 172.16.2.3 default-originate route-map default-map
!
```

```
route-map default-map 10 permit
  match ip address 1
!
access-list 1 permit 192.168.68.0
```

In the following example, the last line of the configuration has been changed to show the use of an extended access list. The local router injects route 0.0.0.0 to the neighbor 172.16.2.3 only if there is a route to 192.168.68.0 with a mask of 255.255.0.0:

```
router bgp 109
  network 172.16.0.0
  neighbor 172.16.2.3 remote-as 200
  neighbor 172.16.2.3 default-originate route-map default-map
!
route-map default-map 10 permit
  match ip address 100
!
access-list 100 permit ip host 192.168.68.0 host 255.255.0.0
```

Related Commands

| Command | Description |
|----------------------------------|--|
| 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. |
| neighbor ebgp-multihop | Accepts and attempts BGP connections to external peers residing on networks that are not directly connected. |

neighbor description

To associate a description with a neighbor, use the **neighbor description** command in router configuration mode or address family configuration mode. To remove the description, use the **no** form of this command.

neighbor {*ip-address**peer-group-name*} **description** *text*
no neighbor {*ip-address**peer-group-name*} **description** [*text*]

Syntax Description

| | |
|------------------------|---|
| <i>ip-address</i> | IP address of the neighbor. |
| <i>peer-group-name</i> | Name of an EIGRP peer group. This argument is not available in address-family configuration mode. |
| <i>text</i> | Text (up to 80 characters in length) that describes the neighbor. |

Command Default

There is no description of the neighbor.

Command Modes

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

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Examples

In the following examples, the description of the neighbor is “peer with example.com”:

```
Device(config)#router bgp 109
Device(config-router)#network 172.16.0.0
Device(config-router)#neighbor 172.16.2.3 description peer with example.com
```

In the following example, the description of the address family neighbor is “address-family-peer”:

```
Device(config)#router eigrp virtual-name
Device(config-router)#address-family ipv4 autonomous-system 4453
Device(config-router-af)#network 172.16.0.0
Device(config-router-af)#neighbor 172.16.2.3 description address-family-peer
```

Related Commands

| Command | Description |
|-------------------------------|--|
| address-family (EIGRP) | Enters address family configuration mode to configure an EIGRP routing instance. |
| network (EIGRP) | Specifies the network for an EIGRP routing process. |
| router eigrp | Configures the EIGRP address family process. |

neighbor ebgp-multihop

To accept and attempt BGP connections to external peers residing on networks that are not directly connected, use the **neighbor ebgp-multihop** command in router configuration mode. To return to the default, use the **no** form of this command.

```
neighbor {ip-addressipv6-addresspeer-group-name} ebgp-multihop [ttl]  
no neighbor {ip-addressipv6-addresspeer-group-name} ebgp-multihop
```

Syntax Description

| | |
|------------------------|--|
| <i>ip-address</i> | IP address of the BGP-speaking neighbor. |
| <i>ipv6-address</i> | IPv6 address of the BGP-speaking neighbor. |
| <i>peer-group-name</i> | Name of a BGP peer group. |
| <i>ttl</i> | (Optional) Time-to-live in the range from 1 to 255 hops. |

Command Default

Only directly connected neighbors are allowed.

Command Modes

Router configuration (config-router)

Command History

Table 12:

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

This feature should be used only under the guidance of Cisco technical support staff.

If you specify a BGP peer group by using the *peer-group-name* argument, all the members of the peer group will inherit the characteristic configured with this command.

To prevent the creation of loops through oscillating routes, the multihop will not be established if the only route to the multihop peer is the default route (0.0.0.0).

Examples

The following example allows connections to or from neighbor 10.108.1.1, which resides on a network that is not directly connected:

```
Device(config)#router bgp 109  
Device(config-router)#neighbor 10.108.1.1 ebgp-multihop
```

Related Commands

| Command | Description |
|---|---|
| neighbor advertise-map non-exist-map | Allows a BGP speaker (the local router) to send the default route 0.0.0.0 to a neighbor for use as a default route. |
| neighbor peer-group (creating) | Creates a BGP peer group. |
| network (BGP and multiprotocol BGP) | Specifies the list of networks for the BGP routing process. |

neighbor maximum-prefix (BGP)

To control how many prefixes can be received from a neighbor, use the **neighbor maximum-prefix** command in router configuration mode. To disable this function, use the **no** form of this command.

```
neighbor {ip-addresspeer-group-name} maximum-prefix maximum [threshold] [restart restart-interval]
[warning-only]
no neighbor {ip-addresspeer-group-name} maximum-prefix maximum
```

Syntax Description

| | |
|-------------------------|---|
| <i>ip-address</i> | IP address of the neighbor. |
| <i>peer-group-name</i> | Name of a Border Gateway Protocol (BGP) peer group. |
| <i>maximum</i> | Maximum number of prefixes allowed from the specified neighbor. The number of prefixes that can be configured is limited only by the available system resources on a router. |
| <i>threshold</i> | (Optional) Integer specifying at what percentage of the <i>maximum-prefix</i> limit the router starts to generate a warning message. The range is from 1 to 100; the default is 75. |
| restart | (Optional) Configures the router that is running BGP to automatically reestablish a peering session that has been disabled because the maximum-prefix limit has been exceeded. The restart timer is configured with the <i>restart-interval</i> argument. |
| <i>restart-interval</i> | (Optional) Time interval (in minutes) that a peering session is reestablished. The range is from 1 to 65535 minutes. |
| warning-only | (optional) Allows the router to generate a sys-log message when the <i>maximum-prefix</i> limit is exceeded, instead of terminating the peering session. |

Command Default

This command is disabled by default. Peering sessions are disabled when the maximum number of prefixes is exceeded. If the *restart-interval* argument is not configured, a disabled session will stay down after the maximum-prefix limit is exceeded.

threshold : 75 percent

Command Modes

Router configuration (config-router)

Command History

Table 13:

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

The **neighbor maximum-prefix** command allows you to configure a maximum number of prefixes that a Border Gateway Protocol (BGP) routing process will accept from the specified peer. This feature provides a mechanism (in addition to distribute lists, filter lists, and route maps) to control prefixes received from a peer.

When the number of received prefixes exceeds the maximum number configured, BGP disables the peering session (by default). If the **restart** keyword is configured, BGP will automatically reestablish the peering

session at the configured time interval. If the **restart** keyword is not configured and a peering session is terminated because the maximum prefix limit has been exceeded, the peering session will not be reestablished until the **clear ip bgp** command is entered. If the **warning-only** keyword is configured, BGP sends only a log message and continues to peer with the sender.

There is no default limit on the number of prefixes that can be configured with this command. Limitations on the number of prefixes that can be configured are determined by the amount of available system resources.

Examples

In the following example, the maximum prefixes that will be accepted from the 192.168.1.1 neighbor is set to 1000:

```
Device(config)#router bgp 40000
Device(config-router)#network 192.168.0.0
Device(config-router)#neighbor 192.168.1.1 maximum-prefix 1000
```

In the following example, the maximum number of prefixes that will be accepted from the 192.168.2.2 neighbor is set to 5000. The router is also configured to display warning messages when 50 percent of the maximum-prefix limit (2500 prefixes) has been reached.

```
Device(config)#router bgp 40000
Device(config-router)#network 192.168.0.0
Device(config-router)#neighbor 192.168.2.2 maximum-prefix 5000 50
```

In the following example, the maximum number of prefixes that will be accepted from the 192.168.3.3 neighbor is set to 2000. The router is also configured to reestablish a disabled peering session after 30 minutes.

```
Device(config)#router bgp 40000
Device(config-router) network 192.168.0.0
Device(config-router)#neighbor 192.168.3.3 maximum-prefix 2000 restart 30
```

In the following example, warning messages will be displayed when the threshold of the maximum-prefix limit ($500 \times 0.75 = 375$) for the 192.168.4.4 neighbor is exceeded:

```
Device(config)#router bgp 40000
Device(config-router)#network 192.168.0.0
Device(config-router)#neighbor 192.168.4.4 maximum-prefix 500 warning-only
```

Related Commands

| Command | Description |
|---------------------|---|
| clear ip bgp | Resets a BGP connection using BGP soft reconfiguration. |

neighbor peer-group (assigning members)

To configure a BGP neighbor to be a member of a peer group, use the **neighbor peer-group** command in address family or router configuration mode. To remove the neighbor from the peer group, use the **no** form of this command.

neighbor {*ip-address*|*ipv6-address*} **peer-group** *peer-group-name*

no neighbor {*ip-address*|*ipv6-address*} **peer-group** *peer-group-name*

Syntax Description

| | |
|------------------------|---|
| <i>ip-address</i> | IP address of the BGP neighbor that belongs to the peer group specified by the <i>peer-group-name</i> argument. |
| <i>ipv6-address</i> | IPv6 address of the BGP neighbor that belongs to the peer group specified by the <i>peer-group-name</i> argument. |
| <i>peer-group-name</i> | Name of the BGP peer group to which this neighbor belongs. |

Command Default

There are no BGP neighbors in a peer group.

Command Modes

Address family configuration (config-router-af)

Router configuration (config-router)

Command History

Table 14:

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

The neighbor at the IP address indicated inherits all the configured options of the peer group.



Note Using the **no** form of the **neighbor peer-group** command removes all of the BGP configuration for that neighbor, not just the peer group association.

Examples

The following router configuration mode example assigns three neighbors to the peer group named **internal**:

```
Device(config)#router bgp 100
Device(config-router)#neighbor internal peer-group
Device(config-router)#neighbor internal remote-as 100
Device(config-router)#neighbor internal update-source loopback 0
Device(config-router)#neighbor internal route-map set-med out
Device(config-router)#neighbor internal filter-list 1 out
Device(config-router)#neighbor internal filter-list 2 in
Device(config-router)#neighbor 172.16.232.53 peer-group internal
Device(config-router)#neighbor 172.16.232.54 peer-group internal
```

```
Device(config-router)#neighbor 172.16.232.55 peer-group internal
Device(config-router)#neighbor 172.16.232.55 filter-list 3 in
```

The following address family configuration mode example assigns three neighbors to the peer group named internal:

```
Device(config)#router bgp 100
Device(config-router)#address-family ipv4 unicast
Device(config-router)#neighbor internal peer-group
Device(config-router)#neighbor internal remote-as 100
Device(config-router)#neighbor internal update-source loopback 0
Device(config-router)#neighbor internal route-map set-med out
Device(config-router)#neighbor internal filter-list 1 out
Device(config-router)#neighbor internal filter-list 2 in
Device(config-router)#neighbor 172.16.232.53 peer-group internal
Device(config-router)#neighbor 172.16.232.54 peer-group internal
Device(config-router)#neighbor 172.16.232.55 peer-group internal
Device(config-router)#neighbor 172.16.232.55 filter-list 3 in
```

Related Commands

| Command | Description |
|---------------------------------------|--|
| 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 vpv4 | 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. |
| neighbor peer-group (creating) | Creates a BGP peer group. |
| neighbor shutdown | Disables a neighbor or peer group. |

neighbor peer-group (creating)

To create a BGP or multiprotocol BGP peer group, use the **neighbor peer-group** command in address family or router configuration mode. To remove the peer group and all of its members, use the **no** form of this command.

neighbor *peer-group-name* **peer-group**
no neighbor *peer-group-name* **peer-group**

| Syntax Description | |
|--------------------|--|
| | <i>peer-group-name</i> Name of the BGP peer group. |

Command Default There is no BGP peer group.

Command Modes Address family configuration (config-router-af)
 Router configuration (config-router)

Command History *Table 15:*

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines Often in a BGP or multiprotocol BGP speaker, many neighbors are configured with the same update policies (that is, same outbound route maps, distribute lists, filter lists, update source, and so on). Neighbors with the same update policies can be grouped into peer groups to simplify configuration and make update calculation more efficient.



Note Peer group members can span multiple logical IP subnets, and can transmit, or pass along, routes from one peer group member to another.

Once a peer group is created with the **neighbor peer-group** command, it can be configured with the **neighbor** commands. By default, members of the peer group inherit all the configuration options of the peer group. Members also can be configured to override the options that do not affect outbound updates.

All the peer group members will inherit the current configuration as well as changes made to the peer group. Peer group members will always inherit the following configuration options by default:

- remote-as (if configured)
- version
- update-source
- outbound route-maps
- outbound filter-lists
- outbound distribute-lists

- minimum-advertisement-interval
- next-hop-self

If a peer group is not configured with a remote-as option, the members can be configured with the **neighbor** *{ip-address | peer-group-name}* **remote-as** command. This command allows you to create peer groups containing external BGP (eBGP) neighbors.

Examples

The following example configurations show how to create these types of neighbor peer group:

- internal Border Gateway Protocol (iBGP) peer group
- eBGP peer group
- Multiprotocol BGP peer group

In the following example, the peer group named internal configures the members of the peer group to be iBGP neighbors. By definition, this is an iBGP peer group because the **router bgp** command and the **neighbor remote-as** command indicate the same autonomous system (in this case, autonomous system 100). All the peer group members use loopback 0 as the update source and use set-med as the outbound route map. The **neighbor internal filter-list 2 in** command shows that, except for 172.16.232.55, all the neighbors have filter list 2 as the inbound filter list.

```
router bgp 100
neighbor internal peer-group
neighbor internal remote-as 100
neighbor internal update-source loopback 0
neighbor internal route-map set-med out
neighbor internal filter-list 1 out
neighbor internal filter-list 2 in
neighbor 172.16.232.53 peer-group internal
neighbor 172.16.232.54 peer-group internal
neighbor 172.16.232.55 peer-group internal
neighbor 172.16.232.55 filter-list 3 in
```

The following example defines the peer group named external-peers without the **neighbor remote-as** command. By definition, this is an eBGP peer group because each individual member of the peer group is configured with its respective autonomous system number separately. Thus the peer group consists of members from autonomous systems 200, 300, and 400. All the peer group members have the set-metric route map as an outbound route map and filter list 99 as an outbound filter list. Except for neighbor 172.16.232.110, all of them have 101 as the inbound filter list.

```
router bgp 100
neighbor external-peers peer-group
neighbor external-peers route-map set-metric out
neighbor external-peers filter-list 99 out
neighbor external-peers filter-list 101 in
neighbor 172.16.232.90 remote-as 200
neighbor 172.16.232.90 peer-group external-peers
neighbor 172.16.232.100 remote-as 300
neighbor 172.16.232.100 peer-group external-peers
neighbor 172.16.232.110 remote-as 400
neighbor 172.16.232.110 peer-group external-peers
neighbor 172.16.232.110 filter-list 400 in
```

In the following example, all members of the peer group are multicast-capable:

```

router bgp 100
neighbor 10.1.1.1 remote-as 1
neighbor 172.16.2.2 remote-as 2
address-family ipv4 multicast
neighbor mygroup peer-group
neighbor 10.1.1.1 peer-group mygroup
neighbor 172.16.2.2 peer-group mygroup
neighbor 10.1.1.1 activate
neighbor 172.16.2.2 activate

```

Related Commands

| Command | Description |
|----------------------------------|--|
| 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. |
| clear ip bgp peer-group | Removes all the members of a BGP peer group. |
| show ip bgp peer-group | Displays information about BGP peer groups. |

neighbor route-map

To apply a route map to incoming or outgoing routes, use the **neighbor route-map** command in address family or router configuration mode. To remove a route map, use the **no** form of this command.

neighbor {*ip-address**peer-group-name* | *ipv6-address*[%]} **route-map** *map-name* {**in** | **out**}
no neighbor {*ip-address**peer-group-name* | *ipv6-address*[%]} **route-map** *map-name* {**in** | **out**}

Syntax Description

| | |
|------------------------|--|
| <i>ip-address</i> | IP address of the neighbor. |
| <i>peer-group-name</i> | Name of a BGP or multiprotocol BGP peer group. |
| <i>ipv6-address</i> | IPv6 address of the neighbor. |
| % | (Optional) IPv6 link-local address identifier. This keyword needs to be added whenever a link-local IPv6 address is used outside the context of its interface. |
| <i>map-name</i> | Name of a route map. |
| in | Applies route map to incoming routes. |
| out | Applies route map to outgoing routes. |

Command Default

No route maps are applied to a peer.

Command Modes

Router configuration (config-router)

Command History

Table 16:

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

When specified in address family configuration mode, this command applies a route map to that particular address family only. When specified in router configuration mode, this command applies a route map to IPv4 or IPv6 unicast routes only.

If an outbound route map is specified, it is proper behavior to only advertise routes that match at least one section of the route map.

If you specify a BGP or multiprotocol BGP peer group by using the *peer-group-name* argument, all the members of the peer group will inherit the characteristic configured with this command. Specifying the command for a neighbor overrides the inbound policy that is inherited from the peer group.

The % keyword is used whenever link-local IPv6 addresses are used outside the context of their interfaces. This keyword does not need to be used for non-link-local IPv6 addresses.

Examples

The following router configuration mode example applies a route map named internal-map to a BGP incoming route from 172.16.70.24:

```
router bgp 5
```

```
neighbor 172.16.70.24 route-map internal-map in
route-map internal-map
match as-path 1
set local-preference 100
```

The following address family configuration mode example applies a route map named internal-map to a multiprotocol BGP incoming route from 172.16.70.24:

```
router bgp 5
address-family ipv4 multicast
neighbor 172.16.70.24 route-map internal-map in
route-map internal-map
match as-path 1
set local-preference 100
```

Related Commands

| Command | Description |
|----------------------------------|--|
| 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 IP Version 4 address prefixes. |
| address-family ipv6 | Enters address family configuration mode for configuring routing sessions such as BGP that use standard IPv6 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 VPN Version 4 address prefixes. |
| address-family vpnv6 | Places the router in address family configuration mode for configuring routing sessions that use standard VPNv6 address prefixes. |
| neighbor remote-as | Creates a BGP peer group. |

neighbor update-source

To have the Cisco software allow Border Gateway Protocol (BGP) sessions to use any operational interface for TCP connections, use the **neighbor update-source** command in router configuration mode. To restore the interface assignment to the closest interface, which is called the best local address, use the **no** form of this command.

neighbor {*ip-address* | *ipv6-address*[%]} [*peer-group-name*] **update-source** *interface-type* *interface-number*
neighbor {*ip-address* | *ipv6-address*[%]} [*peer-group-name*] **update-source** *interface-type* *interface-number*

Syntax Description

| | |
|-------------------------|--|
| <i>ip-address</i> | IPv4 address of the BGP-speaking neighbor. |
| <i>ipv6-address</i> | IPv6 address of the BGP-speaking neighbor. |
| % | (Optional) IPv6 link-local address identifier. This keyword needs to be added whenever a link-local IPv6 address is used outside the context of its interface. |
| <i>peer-group-name</i> | Name of a BGP peer group. |
| <i>interface-type</i> | Interface type. |
| <i>interface-number</i> | Interface number. |

Command Default

Best local address

Command Modes

Router configuration (config-router)

Command History

Table 17:

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

This command can work in conjunction with the loopback interface feature described in the “Interface Configuration Overview” chapter of the Cisco IOS Interface and Hardware Component Configuration Guide.

If you specify a BGP peer group by using the *peer-group-name* argument, all the members of the peer group will inherit the characteristic configured with this command.

The **neighbor update-source** command must be used to enable IPv6 link-local peering for internal or external BGP sessions.

The % keyword is used whenever link-local IPv6 addresses are used outside the context of their interfaces and for these link-local IPv6 addresses you must specify the interface they are on. The syntax becomes <IPv6 local-link address>%<interface name>, for example, FE80::1%Ethernet1/0. Note that the interface type and number must not contain any spaces, and be used in full-length form because name shortening is not supported in this situation. The % keyword and subsequent interface syntax is not used for non-link-local IPv6 addresses.

Examples

The following example sources BGP TCP connections for the specified neighbor with the IP address of the loopback interface rather than the best local address:

```
Device(config)#router bgp 65000
Device(config-router)#network 172.16.0.0
Device(config-router)#neighbor 172.16.2.3 remote-as 110
Device(config-router)#neighbor 172.16.2.3 update-source Loopback0
```

The following example sources IPv6 BGP TCP connections for the specified neighbor in autonomous system 65000 with the global IPv6 address of loopback interface 0 and the specified neighbor in autonomous system 65400 with the link-local IPv6 address of Fast Ethernet interface 0/0. Note that the link-local IPv6 address of FE80::2 is on Ethernet interface 1/0.

```
Device(config)#router bgp 65000
Device(config-router)#neighbor 3ffe::3 remote-as 65000
Device(config-router)#neighbor 3ffe::3 update-source Loopback0
Device(config-router)#neighbor fe80::2%Ethernet1/0 remote-as 65400
Device(config-router)#neighbor fe80::2%Ethernet1/0 update-source FastEthernet 0/0
Device(config-router)#address-family ipv6
Device(config-router)#neighbor 3ffe::3 activate
Device(config-router)#neighbor fe80::2%Ethernet1/0 activate
Device(config-router)#exit-address-family
```

Related Commands

| Command | Description |
|---------------------------|--|
| neighbor activate | Enables the exchange of information with a BGP neighboring router. |
| neighbor remote-as | Adds an entry to the BGP or multiprotocol BGP neighbor table. |

network (BGP and multiprotocol BGP)

To specify the networks to be advertised by the Border Gateway Protocol (BGP) and multiprotocol BGP routing processes, use the **network** command in address family or router configuration mode. To remove an entry from the routing table, use the **no** form of this command.

network {*network-number* [**mask** *network-mask*]*nsap-prefix*} [**route-map** *map-tag*]
no network {*network-number* [**mask** *network-mask*]*nsap-prefix*} [**route-map** *map-tag*]

| Syntax Description | | |
|---------------------------------|--|--|
| <i>network-number</i> | | Network that BGP or multiprotocol BGP will advertise. |
| mask <i>network-mask</i> | | (Optional) Network or subnetwork mask with mask address. |
| <i>nsap-prefix</i> | | Network service access point (NSAP) prefix of the Connectionless Network Service (CLNS) network that BGP or multiprotocol BGP will advertise. This argument is used only under NSAP address family configuration mode. |
| route-map <i>map-tag</i> | | (Optional) Identifier of a configured route map. The route map should be examined to filter the networks to be advertised. If not specified, all networks are advertised. If the keyword is specified, but no route map tags are listed, no networks will be advertised. |

Command Default No networks are specified.

Command Modes Address family configuration (config-router-af)
 Router configuration (config-router)

Command History *Table 18:*

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines BGP and multiprotocol BGP networks can be learned from connected routes, from dynamic routing, and from static route sources.

The maximum number of **network** commands you can use is determined by the resources of the router, such as the configured NVRAM or RAM.

Examples

The following example sets up network 10.108.0.0 to be included in the BGP updates:

```
Device(config)#router bgp 65100
Device(config-router)#network 10.108.0.0
```

The following example sets up network 10.108.0.0 to be included in the multiprotocol BGP updates:

```
Device(config)#router bgp 64800
```

```
Device(config-router)#address family ipv4 multicast
Device(config-router)#network 10.108.0.0
```

The following example advertises NSAP prefix 49.6001 in the multiprotocol BGP updates:

```
Device(config)#router bgp 64500
Device(config-router)#address-family nsap
Device(config-router)#network 49.6001
```

Related Commands

| Command | Description |
|--|---|
| address-family ipv4 (BGP) | Enters the router in address family configuration mode for configuring routing sessions such as BGP, RIP, or static routing sessions that use standard IP Version 4 address prefixes. |
| address-family vpnv4 | Enters 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. |
| default-information originate (BGP) | Allows the redistribution of network 0.0.0.0 into BGP. |
| route-map (IP) | Defines the conditions for redistributing routes from one routing protocol into another. |
| router bgp | Configures the BGP routing process. |

network (EIGRP)

To specify the network for an Enhanced Interior Gateway Routing Protocol (EIGRP) routing process, use the **network** command in router configuration mode or address-family configuration mode. To remove an entry, use the **no** form of this command.

```
network ip-address [wildcard-mask]
no network ip-address [wildcard-mask]
```

| Syntax Description | |
|----------------------|--|
| <i>ip-address</i> | IP address of the directly connected network. |
| <i>wildcard-mask</i> | (Optional) EIGRP wildcard bits. Wildcard mask indicates a subnetwork, bitwise complement of the subnet mask. |

Command Default No networks are specified.

Command Modes Router configuration (config-router) Address-family configuration (config-router-af)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines When the **network** command is configured for an EIGRP routing process, the router matches one or more local interfaces. The **network** command matches only local interfaces that are configured with addresses that are within the same subnet as the address that has been configured with the **network** command. The router then establishes neighbors through the matched interfaces. There is no limit to the number of network statements (**network** commands) that can be configured on a router.

Use a wildcard mask as a shortcut to group networks together. A wildcard mask matches everything in the network part of an IP address with a zero. Wildcard masks target a specific host/IP address, entire network, subnet, or even a range of IP addresses.

When entered in address-family configuration mode, this command applies only to named EIGRP IPv4 configurations. Named IPv6 and Service Advertisement Framework (SAF) configurations do not support this command in address-family configuration mode.

Examples

The following example configures EIGRP autonomous system 1 and establishes neighbors through network 172.16.0.0 and 192.168.0.0:

```
Device(config)#router eigrp 1
Device(config-router)#network 172.16.0.0
Device(config-router)#network 192.168.0.0
Device(config-router)#network 192.168.0.0 0.0.255.255
```

The following example configures EIGRP address-family autonomous system 4453 and establishes neighbors through network 172.16.0.0 and 192.168.0.0:

```
Device(config)#router eigrp virtual-name
Device(config-router)#address-family ipv4 autonomous-system 4453
```

```
Device(config-router-af)#network 172.16.0.0  
Device(config-router-af)#network 192.168.0.0
```

Related Commands

| Command | Description |
|-------------------------------|--|
| address-family (EIGRP) | Enters address-family configuration mode to configure an EIGRP routing instance. |
| router eigrp | Configures the EIGRP address-family process. |

nsf (EIGRP)

To enable Cisco nonstop forwarding (NSF) operations for the Enhanced Interior Gateway Routing Protocol (EIGRP), use the **nsf** command in router configuration or address family configuration mode. To disable EIGRP NSF and to remove the EIGRP NSF configuration from the running-configuration file, use the **no** form of this command.

nsf
no nsf

Syntax Description This command has no arguments or keywords.

Command Default EIGRP NSF is disabled.

Command Modes Router configuration (config-router)
Address family configuration (config-router-af)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines The **nsf** command is used to enable or disable EIGRP NSF support on an NSF-capable router. NSF is supported only on platforms that support High Availability.

Examples The following example shows how to disable NSF:

```
Device#configure terminal
Device(config)#router eigrp 101
Device(config-router)#no nsf
Device(config-router)#end
```

The following example shows how to enable EIGRP IPv6 NSF:

```
Device#configure terminal
Device(config)#router eigrp virtual-name-1
Device(config-router)#address-family ipv6 autonomous-system 10
Device(config-router-af)#nsf
Device(config-router-af)#end
```

| Related Commands | Command | Description |
|------------------|--|---|
| | debug eigrp address-family ipv6 notifications | Displays information about EIGRP address family IPv6 event notifications. |
| | debug eigrp nsf | Displays notifications and information about NSF events for an EIGRP routing process. |
| | debug ip eigrp notifications | Displays information and notifications for an EIGRP routing process. |

| Command | Description |
|---|---|
| show ip protocols | Displays the parameters and the current state of the active routing protocol process. |
| show ipv6 protocols | Displays the parameters and the current state of the active IPv6 routing protocol process. |
| timers graceful-restart purge-time | Sets the graceful-restart purge-time timer to determine how long an NSF-aware router that is running EIGRP must hold routes for an inactive peer. |
| timers nsf converge | Sets the maximum time that the restarting router must wait for the end-of-table notification from an NSF-capable or NSF-aware peer. |
| timers nsf signal | Sets the maximum time for the initial restart period. |

offset-list (EIGRP)

To add an offset to incoming and outgoing metrics to routes learned via Enhanced Interior Gateway Routing Protocol (EIGRP), use the **offset-list** command in router configuration mode or address family topology configuration mode. To remove an offset list, use the **no** form of this command.

offset-list {*access-list-number**access-list-name*} {**in** | **out**} *offset* [*interface-type interface-number*]
no offset-list {*access-list-number**access-list-name*} {**in** | **out**} *offset* [*interface-type interface-number*]

| Syntax Description | | |
|---|--|--|
| <i>access-list-number</i> <i>access-list-name</i> | | Standard access list number or name to be applied. Access list number 0 indicates all networks (networks, prefixes, or routes). If the <i>offset</i> value is 0, no action is taken. |
| in | | Applies the access list to incoming metrics. |
| out | | Applies the access list to outgoing metrics. |
| <i>offset</i> | | Positive offset to be applied to metrics for networks matching the access list. If the offset is 0, no action is taken. |
| <i>interface-type</i> | | (Optional) Interface type to which the offset list is applied. |
| <i>interface-number</i> | | (Optional) Interface number to which the offset list is applied. |

Command Default No offset values are added to incoming or outgoing metrics to routes learned via EIGRP.

Command Modes Router configuration (config-router) Address family topology configuration (config-router-af-topology)

Command History *Table 19:*

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines The offset value is added to the routing metric. An offset list with an interface type and interface number is considered extended and takes precedence over an offset list that is not extended. Therefore, if an entry passes the extended offset list and the normal offset list, the offset of the extended offset list is added to the metric.

Examples

In the following example, the router applies an offset of 10 to the delay component of the router only to access list 21:

```
Device(config-router)#offset-list 21 out 10
```

In the following example, the router applies an offset of 10 to routes learned from Ethernet interface 0:

```
Device(config-router)#offset-list 21 in 10 ethernet 0
```

In the following example, the router applies an offset of 10 to routes learned from Ethernet interface 0 in an EIGRP named 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)#offset-list 21 in 10 ethernet0
```

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

| | |
|-----------------|---|
| <i>protocol</i> | <p>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].</p> <p>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.</p> <p>The application keyword is used to redistribute an application from one routing domain to another. 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).</p> <p>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.</p> |
|-----------------|---|

| | |
|-----------------------------------|--|
| <i>process-id</i> | <p>(Optional) For the application keyword, this is the name of an application.</p> <p>For the bgp or eigrp keyword, this is an autonomous system number, which is a 16-bit decimal number.</p> <p>For the isis keyword, this is an optional <i>tag</i> 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.</p> <p>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.</p> <p>For the rip keyword, no <i>process-id</i> value is needed.</p> <p>For the application keyword, this is the name of an application.</p> <p>By default, no process ID is defined.</p> |
| 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. |
| <i>autonomous-system-number</i> | <p>(Optional) Autonomous system number for the redistributed route. The range is from 1 to 65535.</p> <ul style="list-style-type: none"> 4-byte autonomous system numbers are supported in the range from 1.0 to 65535.65535 in asdot notation only. <p>For more details about autonomous system number formats, see the router bgp command.</p> |
| metric <i>metric-value</i> | (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 <i>type value</i> | <p>(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:</p> <ul style="list-style-type: none"> • 1—Type 1 external route • 2—Type 2 external route <p>If a metric-type is not specified, the Cisco IOS software adopts a Type 2 external route.</p> <p>For IS-IS, it can be one of two values:</p> <ul style="list-style-type: none"> • internal—IS-IS metric that is < 63. • external—IS-IS metric that is > 64 < 128. <p>The default is internal.</p> |
| match { internal external1 external2 } | <p>(Optional) Specifies the criteria by which OSPF routes are redistributed into other routing domains. It can be one of the following:</p> <ul style="list-style-type: none"> • 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. <p>The default is internal.</p> |
| tag <i>tag-value</i> | <p>(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.</p> |
| route-map | <p>(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.</p> |
| <i>map-tag</i> | <p>(Optional) Identifier of a configured route map.</p> |

| | |
|------------------|--|
| subnets | (Optional) For redistributing routes into OSPF. Note Irrespective of whether the subnets keyword is configured or not, the subnets functionality is enabled by default. This automatic addition results in the redistribution of classless OSPF routes. |
| 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 |
|------------------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

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.
- An EIGRP routing process is configured when you issue the **router eigrp** command and then specify a network for the process using the **network** sub-command. Suppose that you have not configured an EIGRP routing process, and that you have configured redistribution of routes from such an EIGRP process into BGP, OSPF, or RIP. If you use the **no redistribute eigrp** command to change or disable a parameter

in the **redistribute eigrp** command, the **no redistribute eigrp** command removes the entire **redistribute eigrp** command instead of changing or disabling a specific parameter.

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.)



Note The **show ip ospf [topology-info]** command will display **subnets** keyword irrespective of whether the **subnets** keyword is configured or not. This is because the subnets functionality is enabled by default for OSPF.

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.

4-Byte Autonomous System Number Support

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.

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.

```
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 |
|---|--|
| default-information originate (OSPF) | Generates a default route into an OSPF routing domain. |
| router bgp | Configures the BGP routing process. |
| router eigrp | Configures the EIGRP address-family process. |

redistribute (IPv6)

To redistribute IPv6 routes from one routing domain into another routing domain, use the **redistribute** command in IPv6 address family configuration mode. To disable redistribution, use the **no** form of this command.

```
redistribute protocol [ {process-id} ] [ include-connected {level-1 | level-1-2 | level-2} ] [ {as-number} ] [ metric metric-value ] [ metric-type type-value ] [ nssa-only ] [ {tag tag-value} ] [ route-map map-tag ]
```

```
no redistribute protocol [ {process-id} ] [ include-connected {level-1 | level-1-2 | level-2} ] [ {as-number} ] [ metric metric-value ] [ metric-type type-value ] [ nssa-only ] [ {tag tag-value} ] [ route-map map-tag ]
```

| Syntax Description | |
|--------------------------------------|---|
| <i>protocol</i> | Source protocol from which routes are redistributed. It can be one of the following keywords: bgp , connected , eigrp , isis , lisp , nd , omp , ospf (ospfv3), rip , or static . |
| <i>process-id</i> | (Optional) For the bgp or eigrp keyword, the process ID is an autonomous system number, which is a 16-bit decimal number. For the isis keyword, the process ID is an optional value that defines a meaningful name for a routing process. You can specify only one Intermediate System-to-Intermediate System (IS-IS) process per router. Creating a name for a routing process means that you use names when configuring routing. For the ospf keyword, the process ID is the number that is assigned administratively when the Open Shortest Path First (OSPF) for the IPv6 routing process is enabled. For the rip keyword, the process ID is an optional value that defines a meaningful name for an IPv6 Routing Information Protocol (RIP) routing process. |
| include-connected | (Optional) Allows the target protocol to redistribute routes that are learned by the source protocol and connected prefixes on those interfaces over which the source protocol is running. |
| level-1 | Specifies that for IS-IS, Level 1 routes are redistributed into other IPv6 routing protocols independently. |
| level-1-2 | Specifies that for IS-IS, both Level 1 and Level 2 routes are redistributed into other IPv6 routing protocols. |
| level-2 | Specifies that for IS-IS, Level 2 routes are redistributed into other IPv6 routing protocols independently. |
| <i>as-number</i> | (Optional) Autonomous system number for the redistributed route. |
| metric <i>metric-value</i> | (Optional) When redistributing from one OSPF process to another OSPF process on the same router, the metric is 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. |

| | |
|---|--|
| metric-type <i>type-value</i> | (Optional) Specifies the external link type that is associated with the default route that is advertised into the routing domain. It can be one of two values: <ul style="list-style-type: none"> • 1: Type 1 external route • 2: Type 2 external route <p>If no value is specified for the metric-type keyword, the Cisco IOS software adopts a Type 2 external route.</p> |
| nssa-only | (Optional) Limits redistributed routes to not-so-stubby area (NSSA) |
| tag <i>tag-value</i> | (Optional) Specifies the 32-bit decimal value that is attached to each external route. This is not used by OSPF itself. It might be used to communicate information between Autonomous System Boundary Routers (ASBRs). If none is specified, then the remote autonomous system number is used for routes from the BGP and the Exterior Gateway Protocol (EGP); for other protocols, zero (0) is used. |
| route-map | (Optional) Specifies the route map that is checked to filter the import of routes from this source routing protocol to the current routing protocol. If the route-map keyword is not specified, all the routes are redistributed. If this keyword is specified, but no route map tags are listed, no routes are imported. |
| <i>map-tag</i> | (Optional) Identifier of a configured route map. |

Command Modes

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

Command History

| Release | Modification |
|------------------------------|----------------|
| Cisco IOS XE Everest 16.5.1a | This command w |

Usage Guidelines

Changing or disabling a keyword does not affect the state of other keywords.

IS-IS ignores configured redistribution of routes, if any that are configured with the **include-connected** keyword. IS-IS advertises a prefix on an interface if either IS-IS is running over the interface or the interface is configured as passive.

Routes that are learned from IPv6 routing protocols are redistributed into IPv6 IS-IS 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.

For IPv6 RIP, use the **redistribute** command to advertise static routes as if they were directly connected routes.



Note Advertising static routes as directly connected routes might cause routing loops if improperly configured.

Redistributed IPv6 RIP routing information is always filtered by the **distribute-list prefix-list** command in router configuration mode. Using the **distribute-list prefix-list** command ensures that only those routes that are intended by the administrator are passed along to the receiving routing protocol.



Note The **metric** value that is specified in the **redistribute** command for IPv6 RIP supersedes the **metric** value that is specified using the **default-metric** command.

In IPv4, if you redistribute a protocol, by default, you also redistribute the subnet on the interfaces over which the protocol is running. In IPv6, this is not the default behavior. To redistribute the subnet on the interfaces over which the protocol is running in IPv6, use the **include-connected** keyword. In IPv6, this functionality is not supported when the source protocol is BGP.

When the **no redistribute** command is configured, the parameter settings are ignored when the client protocol is IS-IS or EIGRP.

IS-IS redistribution is removed completely when IS-IS Level 1 and Level 2 are removed by you. IS-IS level settings can be configured using the **redistribute** command only.

The default redistribute type is restored to OSPFv3 when all route type values are removed by you.

Specify the **nssa-only** keyword to clear the propagate bit (P-bit) when external routes are redistributed into an NSSA. Doing so prevents corresponding NSSA external link state advertisements (LSAs) from being translated into other areas.

Examples

The following example shows how to configure IPv6 IS-IS to redistribute IPv6 BGP routes. The metric is specified as 5, and the metric type is set to 1.

```
Device> enable
Device# configure terminal
Device(config)# router isis
Device(config-router)# address-family ipv6
Device(config-router-af)# redistribute bgp 64500 metric 5 metric-type 1
```

The following example shows how to redistribute IPv6 BGP routes into the IPv6 RIP routing process named cisco:

```
Device> enable
Device# configure terminal
Device(config)# router rip cisco
Device(config-router)# redistribute bgp 42
```

The following example shows how to redistribute IS-IS for IPv6 routes into the OSPFv3 for IPv6 routing process 1:

```
Device> enable
Device# configure terminal
Device(config)# router ospfv3 1
Device(config-router)# address-family ipv6
Device(config-router-af)# redistribute isis 1 metric 32 metric-type 1 tag 85
```

redistribute maximum-prefix (OSPF)

To limit the number of prefixes that are redistributed into Open Shortest Path First (OSPF) or to generate a warning when the number of prefixes that are redistributed into OSPF reaches a maximum, use the **redistribute maximum-prefix** command in router configuration mode. To remove the values, use the **no** form of this command.

redistribute maximum-prefix *maximum* [{*percentage*}] [{**warning-only**}]
no redistribute

Syntax Description

| | |
|---------------------|--|
| <i>maximum</i> | Integer from 1 to 4294967295 that specifies the maximum number of IP or IPv6 prefixes that can be redistributed into OSPF. When the warning-only keyword is configured, the maximum value specifies the number of prefixes that can be redistributed into OSPF before the system logs a warning message. Redistribution is not limited. The maximum number of IP or IPv6 prefixes that are allowed to be redistributed into OSPF, or the number of prefixes that are allowed to be redistributed into OSPF before the system logs a warning message, depends on whether the warning-only keyword is present. There is no default value for the maximum argument. If the warning-only keyword is also configured, this value does not limit redistribution; it is simply the number of redistributed prefixes that, when reached, causes a warning message to be logged. |
| <i>percentage</i> | (Optional) Integer from 1 to 100 that specifies the threshold value, as a percentage, at which a warning message is generated. The default percentage is 75. |
| warning-only | (Optional) Causes a warning message to be logged when the number of prefixes that are defined by the <i>maximum</i> argument has been exceeded. Additional redistribution is not prevented. |

Command Default

The default percentage is 75.

Command Modes

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

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

A network can be severely flooded if many IP or IPv6 prefixes are injected into the OSPF, perhaps by redistributing Border Gateway Protocol (BGP) into OSPF. Limiting the number of redistributed prefixes prevents this potential problem.

When the **redistribute maximum-prefix** command is configured and the number of redistributed prefixes reaches the maximum value that is configured, no more prefixes are redistributed (unless the **warning-only** keyword is configured).

Examples

The following example shows how two warning messages are logged; the first if the number of prefixes redistributed reaches 85 percent of 600 (510 prefixes), and the second if the number of redistributed routes reaches 600. However, the number of redistributed routes is not limited.

```
Device> enable
Device# configure terminal
Device(config)# router ospfv3 11
Device(config-router)# address-family ipv6
Device(config-router-af)# redistribute eigrp 10 subnets
Device(config-router-af)# redistribute maximum-prefix 600 85 warning-only
```

The following example shows how to set a maximum of 10 prefixes that can be redistributed into an OSPFv3 process:

```
Device> enable
Device# configure terminal
Device(config)# router ospfv3 10
Device(config-router)# address-family ipv6 unicast
Device(config-router-af)# redistribute maximum-prefix 10
Device(config-router-af)# redistribute connected
```

rewrite-evpn-rt-asn

To enable the rewrite of the autonomous system number (ASN) portion of the EVPN route target extended community with the ASN of the target eBGP EVPN peer, use the **rewrite-evpn-rt-asn** command in address family configuration mode. Use the **no** form of the command to disable the rewrite of ASN.

rewrite-evpn-rt-asn
no rewrite-evpn-rt-asn

Syntax Description This command has no arguments or keywords.

Command Modes Address-family configuration (config-router-af)

Command History

| Release | Modification |
|--------------------------------|-----------------------------|
| Cisco IOS XE Gibraltar 16.12.1 | The command was introduced. |

Usage Guidelines

The **rewrite-evpn-rt-asn** command is required for the route target auto feature to be used to configure EVPN route targets. Route target auto feature is implemented on all border leaf switches that support BGP EVPN.

The **rewrite-evpn-rt-asn** command only affects the following:

- EVPN address family.
- Inbound route-reception.
- Routes from eBGP peers.
- Route-type 2 and route-type 5 of EVPN prefixes.
- route target extended community inside the BGP update.

The **rewrite-evpn-rt-asn** command only works on type 0 and on type 2 of route-target extended communities.



Note Run this command only when route target auto feature is being used and matching route targets are not manually configured on all switches.

The following example shows how to enable rewrite of ASN using the **rewrite-evpn-rt-asn** command:

```
Device# configure terminal
Device(config)# router bgp 10000
Device(config-router)# address-family l2vpn evpn
Device(config-router-af)# rewrite-evpn-rt-asn
```

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] ordering-seq sequence-name
no route-map map-tag [{permit | deny}] [sequence-number] ordering-seq sequence-name
```

| Syntax Description | | |
|--|--|--|
| <i>map-tag</i> | Name for the route map. | |
| permit | (Optional) Permits only the routes matching the route map to be forwarded or redistributed. | |
| deny | (Optional) Blocks routes matching the route map from being forwarded or redistributed. | |
| <i>sequence-number</i> | (Optional) Number that indicates the position a new route map will have in the list of route maps already configured with the same name. | |
| ordering-seq <i>sequence-name</i> | (Optional) Orders the route maps based on the string provided. | |

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 |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

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 these purposes are described here.

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*, that is, the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the *set actions*, that is, the redistribution actions to be performed if the criteria enforced by the **match** commands are met. If the **route-map** command is enabled and the user does not specify any action, then the **permit** action is applied by default. The **no route-map** command deletes the route map.

The **match** route-map configuration command has multiple formats. The **match** commands can be run in any order, and all the **match** commands must match to cause the route to be redistributed according to the *set actions* specified 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 passing routes through a route map, the route map can have several parts. Any route that does not match at least one **match** clause relating to a **route-map** command is ignored, that is, the route is not advertised for outbound route maps, and is not accepted for inbound route maps. If you want to modify only some data, 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 can 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 other route maps sharing the same map tag name are 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. We recommend that you policy route packets some way other than the obvious shortest path.

The *sequence-number* argument works as follows:

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

If the **no route-map map-tag** command is specified (without the *sequence-number* argument), the entire 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 the Open Shortest Path First (OSPF). These routes will be redistributed to the OSPF as external link-state advertisements (LSAs) with a metric of 5, metric type of type1, and a tag equal to 1.

```
Device> enable
Device# configure terminal
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 the OSPF. These routes will be redistributed to the OSPF as external LSAs, with a tag equal to 42, and a metric type equal to type1.

```
Device> enable
Device# configure terminal
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 the EIGRP as external, with a metric of 5, and a tag equal to 1:

```
Device> enable
Device# configure terminal
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 | Matches values from the routing table. |
| router eigrp | Configures the EIGRP address-family process. |
| set | Sets values in the destination routing protocol |
| show route-map | Displays all route maps configured or only the one specified. |

router-id

To use a fixed router ID, use the **router-id** command in router configuration mode. To force Open Shortest Path First (OSPF) to use the previous OSPF router ID behavior, use the **no** form of this command.

router-id *ip-address*

no router-id *ip-address*

Syntax Description

| | |
|-------------------|---------------------------------|
| <i>ip-address</i> | Router ID in IP address format. |
|-------------------|---------------------------------|

Command Default

No OSPF routing process is defined.

Command Modes

Router configuration

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

You can configure an arbitrary value in the IP address format for each router. However, each router ID must be unique.

If this command is used on an OSPF router process which is already active (has neighbors), the new router-ID is used at the next reload or at a manual OSPF process restart. To manually restart the OSPF process, use the clear ip ospf command.

Examples

The following example specifies a fixed router-id:

```
router-id 10.1.1.1
```

Related Commands

| Command | Description |
|----------------------|---|
| clear ip ospf | Clears redistribution based on the OSPF routing process ID. |
| router ospf | Configures the OSPF routing process. |

router bgp

To configure the Border Gateway Protocol (BGP) routing process, use the **router bgp** command in global configuration mode. To remove a BGP routing process, use the **no** form of this command.

router bgp *autonomous-system-number*
no router bgp *autonomous-system-number*

| | | |
|---------------------------|---------------------------------|--|
| Syntax Description | <i>autonomous-system-number</i> | Number of an autonomous system that identifies the router to other BGP routers and tags the routing information that is passed along. Number in the range from 1 to 65535. |
|---------------------------|---------------------------------|--|

Command Default No BGP routing process is enabled by default.

Command Modes Global configuration (config)

| | | |
|------------------------|--------------------------------|------------------------------|
| Command History | Release | Modification |
| | Cisco IOS XE Gibraltar 16.12.1 | This command was introduced. |

Usage Guidelines This command allows you to set up a distributed routing core that automatically guarantees the loop-free exchange of routing information between autonomous systems.

Cisco has implemented the following two methods of representing autonomous system numbers:

- **Asplain**—Decimal value notation where both 2-byte and 4-byte autonomous system numbers are represented by their decimal value. For example, 65526 is a 2-byte autonomous system number and 234567 is a 4-byte autonomous system number.
- **Asdot**—Autonomous system dot notation where 2-byte autonomous system numbers are represented by their decimal value and 4-byte autonomous system numbers are represented by a dot notation. For example, 65526 is a 2-byte autonomous system number and 1.169031 is a 4-byte autonomous system number (this is dot notation for the 234567 decimal number).

For details about the third method of representing autonomous system numbers, see [RFC 5396](#).



Note In Cisco IOS releases that include 4-byte ASN support, command accounting and command authorization that include a 4-byte ASN number are sent in the asplain notation irrespective of the format that is used on the command-line interface.

Asplain as Default Autonomous System Number Formatting

The Cisco implementation of 4-byte autonomous system numbers uses asplain as the default display format for autonomous system numbers, but you can configure 4-byte autonomous system numbers in both the asplain and asdot format. In addition, the default format for matching 4-byte autonomous system numbers in regular expressions is asplain, so you must ensure that any regular expressions to match 4-byte autonomous system numbers are written in the asplain format. If you want to change the default **show** command output to display 4-byte autonomous system numbers in the asdot format, use the **bgp asnotation dot** command under router

configuration mode. When the asdot format is enabled as the default, any regular expressions to match 4-byte autonomous system numbers must be written using the asdot format, or the regular expression match will fail. The tables below show that although you can configure 4-byte autonomous system numbers in either asplain or asdot format, only one format is used to display **show** command output and control 4-byte autonomous system number matching for regular expressions, and the default is asplain format. To display 4-byte autonomous system numbers in **show** command output and to control matching for regular expressions in the asdot format, you must configure the **bgp asnotation dot** command. After enabling the **bgp asnotation dot** command, a hard reset must be initiated for all BGP sessions by entering the **clear ip bgp *** command.



Note If you are upgrading to an image that supports 4-byte autonomous system numbers, you can still use 2-byte autonomous system numbers. The **show** command output and regular expression match are not changed and remain in asplain (decimal value) format for 2-byte autonomous system numbers regardless of the format configured for 4-byte autonomous system numbers.

Table 21: Default Asplain 4-Byte Autonomous System Number Format

| Format | Configuration Format | Show Command Output and Regular Expression Match Format |
|---------|--|---|
| asplain | 2-byte: 1 to 65535 4-byte: 65536 to 4294967295 | 2-byte: 1 to 65535 4-byte: 65536 to 4294967295 |
| asdot | 2-byte: 1 to 65535 4-byte: 1.0 to 65535.65535 | 2-byte: 1 to 65535 4-byte: 65536 to 4294967295 |

Table 22: Asdot 4-Byte Autonomous System Number Format

| Format | Configuration Format | Show Command Output and Regular Expression Match Format |
|---------|--|---|
| asplain | 2-byte: 1 to 65535 4-byte: 65536 to 4294967295 | 2-byte: 1 to 65535 4-byte: 1.0 to 65535.65535 |
| asdot | 2-byte: 1 to 65535 4-byte: 1.0 to 65535.65535 | 2-byte: 1 to 65535 4-byte: 1.0 to 65535.65535 |

Reserved and Private Autonomous System Numbers

The Cisco implementation of BGP supports [RFC 4893](#). RFC 4893 was developed to allow BGP to support a gradual transition from 2-byte autonomous system numbers to 4-byte autonomous system numbers. A new reserved (private) autonomous system number, 23456, was created by RFC 4893 and this number cannot be configured as an autonomous system number in the Cisco IOS CLI.

[RFC 5398](#), *Autonomous System (AS) Number Reservation for Documentation Use*, describes new reserved autonomous system numbers for documentation purposes. Use of the reserved numbers allow configuration examples to be accurately documented and avoids conflict with production networks if these configurations are literally copied. The reserved numbers are documented in the IANA autonomous system number registry. Reserved 2-byte autonomous system numbers are in the contiguous block, 64496 to 64511 and reserved 4-byte autonomous system numbers are from 65536 to 65551 inclusive.

Private 2-byte autonomous system numbers are still valid in the range from 64512 to 65534 with 65535 being reserved for special use. Private autonomous system numbers can be used for internal routing domains but must be translated for traffic that is routed out to the Internet. BGP should not be configured to advertise

private autonomous system numbers to external networks. Cisco IOS software does not remove private autonomous system numbers from routing updates by default. Cisco recommends that ISPs filter private autonomous system numbers.



Note Autonomous system number assignment for public and private networks is governed by the IANA. For information about autonomous system numbers, including reserved number assignment, or to apply to register an autonomous system number, see the following URL: <http://www.iana.org/>.

Examples

The following example shows how to configure a BGP process for autonomous system 45000 and configures two external BGP neighbors in different autonomous systems using 2-byte autonomous system numbers:

```
Device> enable
Device# configure terminal
Device(config)# router bgp 45000
Device(config-router)# neighbor 192.168.1.2 remote-as 40000
Device(config-router)# neighbor 192.168.3.2 remote-as 50000
Device(config-router)# neighbor 192.168.3.2 description finance
Device(config-router)# address-family ipv4
Device(config-router-af)# neighbor 192.168.1.2 activate
Device(config-router-af)# neighbor 192.168.3.2 activate
Device(config-router-af)# no auto-summary
Device(config-router-af)# no synchronization
Device(config-router-af)# network 172.17.1.0 mask 255.255.255.0
Device(config-router-af)# exit-address-family
```

The following example shows how to configure a BGP process for autonomous system 65538 and configures two external BGP neighbors in different autonomous systems using 4-byte autonomous system numbers in asplain notation. This example is supported in Cisco IOS Release 12.0(32)SY8, 12.0(33)S3, 12.2(33)SRE, 12.2(33)XNE, 12.2(33)SXH, Cisco IOS XE Release 2.4, and later releases.

```
Device> enable
Device# configure terminal
Device(config)# router bgp 65538
Device(config-router)# neighbor 192.168.1.2 remote-as 65536
Device(config-router)# neighbor 192.168.3.2 remote-as 65550
Device(config-router)# neighbor 192.168.3.2 description finance
Device(config-router)# address-family ipv4
Device(config-router-af)# neighbor 192.168.1.2 activate
Device(config-router-af)# neighbor 192.168.3.2 activate
Device(config-router-af)# no auto-summary
Device(config-router-af)# no synchronization
Device(config-router-af)# network 172.17.1.0 mask 255.255.255.0
Device(config-router-af)# exit-address-family
```

Related Commands

| Command | Description |
|--|---|
| neighbor remote-as | Adds an entry to the BGP or multiprotocol BGP neighbor table. |
| network (BGP and multiprotocol BGP) | Specifies the list of networks for the BGP routing process. |

router eigrp

To configure the EIGRP routing process, use the **router eigrp** command in global configuration mode. To remove an EIGRP routing process, use the **no** form of this command.

router eigrp {*autonomous-system-number**virtual-instance-name*}
no router eigrp {*autonomous-system-number**virtual-instance-name*}

Syntax Description

| | |
|---------------------------------|--|
| <i>autonomous-system-number</i> | Autonomous system number that identifies the EIGRP services to the other EIGRP address-family routers. It is also used to tag routing information. Valid range is from 1 to 65535. |
| <i>virtual-instance-name</i> | EIGRP virtual instance name. This name must be unique among all the address-family router processes on a single router, but need not be unique among routers. |

Command Default

No EIGRP processes are configured.

Command Modes

Global configuration (config)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

Configuring the **router eigrp** command with the *autonomous-system-number* argument creates an EIGRP configuration referred to as autonomous system (AS) configuration. An EIGRP AS configuration creates an EIGRP routing instance that can be used for tagging routing information.

Configuring the **router eigrp** command with the *virtual-instance-name* argument creates an EIGRP configuration referred to as EIGRP named configuration. An EIGRP named configuration does not create an EIGRP routing instance by itself. An EIGRP named configuration is a base configuration that is required to define address-family configurations under it that are used for routing.

Examples

The following example shows how to configure EIGRP process 109:

```
Device> enable
Device# configure terminal
Device(config)# router eigrp 109
```

The following example configures an EIGRP address-family routing process and assigns it the name *virtual-name*:

```
Device> enable
Device# configure terminal
Device(config)# router eigrp virtual-name
```

router ospf

To configure an OSPF routing process, use the **router ospf** command in global configuration mode. To terminate an OSPF routing process, use the **no** form of this command.

```
router ospf process-id [vrf vrf-name]
no router ospf process-id [vrf vrf-name]
```

| Syntax Description | |
|----------------------------|--|
| <i>process-id</i> | Internally used identification parameter for an OSPF routing process. It is locally assigned, and can be a positive integer. A unique value is assigned for each OSPF routing process. |
| vrf <i>vrf-name</i> | (Optional) Specifies the name of the VPN routing and forwarding (VRF) instance to associate with the OSPF VRF processes. |

Command Default No OSPF routing process is defined.

Command Modes Global configuration (config)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines You can specify multiple OSPF routing processes in each router.

After you enter the **router ospf** command, you can enter the maximum number of paths. There can be between 1 and 32 paths.

Examples

The following example shows how to configure an OSPF routing process and assign a process number of 109:

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

The following example shows a basic OSPF configuration using the **router ospf** command to configure the OSPF VRF instance processes for the first, second, and third VRFs:

```
Device> enable
Device# configure terminal
Device(config)# router ospf 12 vrf first
Device(config)# router ospf 13 vrf second
Device(config)# router ospf 14 vrf third
Device(config)# exit
```

The following example shows how to use the **maximum-paths** option:

```
Device> enable
Device# configure terminal
Device(config)# router ospf
Device(config-router)# maximum-paths 2
Device(config-router)# exit
```

Related Commands

| Command | Description |
|--------------|--|
| network area | Defines the interfaces on which OSPF runs, and defines the area ID for those interfaces. |

router ospfv3

To enter Open Shortest Path First Version 3 (OSPFv3) through router configuration mode, use the **router ospfv3** command in global configuration mode.

```
router ospfv3 [{process-id}]
```

Syntax Description

process-id (Optional) Internal identification. The number that is used here is the number assigned administratively when enabling the OSPFv3 routing process. The range is 1-65535.

Command Default

OSPFv3 routing process is disabled by default.

Command Modes

Global configuration (config)

Command History

| Release | Modification |
|------------------------------|-----------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced |

Usage Guidelines

Use the **router ospfv3** command to enter OSPFv3 router configuration mode. From this mode, you can enter address-family configuration mode for IPv6 or IPv4, and then configure the IPv6 or IPv4 address family.

Examples

The following example shows how to enter OSPFv3 router configuration mode:

```
Device> enable
Device# configure terminal
Device(config)# router ospfv3 1
Device(config-router)#
```

Related Commands

| Command | Description |
|----------------------------|--|
| address-family ipv6 | Enters IPv6 address family configuration mode. |

send-lifetime

To set the time period during which an authentication key on a key chain is valid to be sent, use the **send-lifetime** command in key chain key configuration mode. To revert to the default value, use the **no** form of this command.

```
send-lifetime [ local ] start-time { infinite end-time | duration seconds }
no send-lifetime
```

Syntax Description

| | |
|--------------------------------|--|
| local | Specifies the time in local timezone. |
| <i>start-time</i> | Beginning time that the key specified by the key command is valid to be sent. The syntax can be either of the following: <i>hh : mm : ss month date year</i> <i>hh : mm : ss date month year</i> <ul style="list-style-type: none"> • <i>hh</i>: Hours • <i>mm</i>: Minutes • <i>ss</i>: Seconds • <i>month</i>: First three letters of the month • <i>date</i>: Date (1-31) • <i>year</i>: Year (four digits) <p>The default start time and the earliest acceptable date is January 1, 1993.</p> |
| infinite | Key is valid to be sent from the <i>start-time</i> value on. |
| <i>end-time</i> | Key is valid to be sent from the <i>start-time</i> value until the <i>end-time</i> value. The syntax is the same as that for the <i>start-time</i> value. The <i>end-time</i> value must be after the <i>start-time</i> value. The default end time is an infinite time period. |
| duration <i>seconds</i> | Length of time (in seconds) that the key is valid to be sent. The range is from 1 to 2147483646. |

Command Default

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 |
|-------------------------------|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Bengaluru 17.5.1 | The new range of the duration keyword is from 1 to 2147483646. |

Usage Guidelines

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 intend to set lifetimes on keys.

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 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.

```
Device(config)# interface GigabitEthernet1/0/1
Device(config-if)# ip rip authentication key-chain chain1
Device(config-if)# ip rip authentication mode md5
Device(config-if)# exit
Device(config)# router rip
Device(config-router)# network 172.19.0.0
Device(config-router)# version 2
Device(config-router)# exit
Device(config)# key chain chain1
Device(config-keychain)# key 1
Device(config-keychain-key)# key-string key1
Device(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Device(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Device(config-keychain-key)# exit
Device(config-keychain)# key 2
Device(config-keychain)# key-string key2
Device(config-keychain)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Device(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.

```
Device(config)# router eigrp 10
Device(config-router)# address-family ipv4 autonomous-system 4453
Device(config-router-af)# network 10.0.0.0
Device(config-router-af)# af-interface ethernet0/0
Device(config-router-af-interface)# authentication key-chain trees
Device(config-router-af-interface)# authentication mode md5
Device(config-router-af-interface)# exit
Device(config-router-af)# exit
Device(config-router)# exit
Device(config)# key chain chain1
Device(config-keychain)# key 1
Device(config-keychain-key)# key-string key1
Device(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Device(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Device(config-keychain-key)# exit
Device(config-keychain)# key 2
Device(config-keychain-key)# key-string key2
Device(config-keychain-key)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Device(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. |
| 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. |
| show key chain | Displays authentication key information. |

set community

To set the BGP communities attribute, use the **set community** route map configuration command. To delete the entry, use the **no** form of this command.

```
set community {community-number [additive] [well-known-community] | none}
no set community
```

| Syntax Description | |
|-----------------------------|--|
| <i>community-number</i> | Specifies that community number. Valid values are from 1 to 4294967200, no-export , or no-advertise . |
| additive | (Optional) Adds the community to the already existing communities. |
| <i>well-known-community</i> | (Optional) Well know communities can be specified by using the following keywords: <ul style="list-style-type: none"> • internet • local-as • no-advertise • no-export |
| none | (Optional) Removes the community attribute from the prefixes that pass the route map. |

Command Default No BGP communities attributes exist.

Command Modes Route-map configuration (config-route-map)

Command History *Table 23:*

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines You must have a match clause (even if it points to a “permit everything” list) if you want to set tags.

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 **set** route map configuration commands specify the redistribution *set actions* to be performed when all of the match criteria of a route map are met. When all match criteria are met, all set actions are performed.

Examples

In the following example, routes that pass the autonomous system path access list 1 have the community set to 109. Routes that pass the autonomous system path access list 2 have the community set to no-export (these routes will not be advertised to any external BGP [eBGP] peers).

```
route-map set_community 10 permit
match as-path 1
set community 109
route-map set_community 20 permit
match as-path 2
set community no-export
```

In the following similar example, routes that pass the autonomous system path access list 1 have the community set to 109. Routes that pass the autonomous system path access list 2 have the community set to local-as (the router will not advertise this route to peers outside the local autonomous system).

```
route-map set_community 10 permit
match as-path 1
set community 109
route-map set_community 20 permit
match as-path 2
set community local-as
```

Related Commands

| Command | Description |
|------------------------------|---|
| ip community-list | Creates a community list for BGP and control access to it. |
| match community | Matches a BGP community. |
| route-map (IP) | Defines the conditions for redistributing routes from one routing protocol into another, or enables policy routing. |
| set comm-list delete | Removes communities from the community attribute of an inbound or outbound update. |
| show ip bgp community | Displays routes that belong to specified BGP communities. |

set ip next-hop (BGP)

To indicate where to output packets that pass a match clause of a route map for policy routing, use the **set ip next-hop** command in route-map configuration mode. To delete an entry, use the **no** form of this command.

```
set ip next-hop ip-address[ {...ip-address} ][ {peer-address} ]
no set ip next-hop ip-address[ {...ip-address} ][ {peer-address} ]
```

Syntax Description

| | |
|---------------------|--|
| <i>ip-address</i> | IP address of the next hop to which packets are output. It need not be an adjacent router. |
| peer-address | (Optional) Sets the next hop to be the BGP peering address. |

Command Default

This command is disabled by default.

Command Modes

Route-map configuration (config-route-map)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

An ellipsis (...) in the command syntax indicates that your command input can include multiple values for the *ip-address* argument.

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.

If the first next hop specified with the **set ip next-hop** command is down, the optionally specified IP addresses are tried in turn.

When the **set ip next-hop** command is used with the **peer-address** keyword in an inbound route map of a BGP peer, the next hop of the received matching routes will be set to be the neighbor peering address, overriding any third-party next hops. So the same route map can be applied to multiple BGP peers to override third-party next hops.

When the **set ip next-hop** command is used with the **peer-address** keyword in an outbound route map of a BGP peer, the next hop of the advertised matching routes will be set to be the peering address of the local router, thus disabling the next hop calculation. The **set ip next-hop** command has finer granularity than the (per-neighbor) **neighbor next-hop-self** command, because you can set the next hop for some routes, but not others. The **neighbor next-hop-self** command sets the next hop for all routes sent to that neighbor.

The set clauses can be used in conjunction with one another. They are evaluated in the following order:

1. **set ip next-hop**
2. **set interface**
3. **set ip default next-hop**

4. set default interface



Note To avoid a common configuration error for reflected routes, do not use the **set ip next-hop** command in a route map to be applied to BGP route reflector clients.

Configuring the **set ip next-hop ...ip-address** command on a VRF interface allows the next hop to be looked up in a specified VRF address family. In this context, the *...ip-address* argument matches that of the specified VRF instance.

Examples

In the following example, three routers are on the same FDDI LAN (with IP addresses 10.1.1.1, 10.1.1.2, and 10.1.1.3). Each is in a different autonomous system. The **set ip next-hop peer-address** command specifies that traffic from the router (10.1.1.3) in remote autonomous system 300 for the router (10.1.1.1) in remote autonomous system 100 that matches the route map is passed through the router bgp 200, rather than sent directly to the router (10.1.1.1) in autonomous system 100 over their mutual connection to the LAN.

```
Device(config)#router bgp 200
Device(config)#neighbor 10.1.1.3 remote-as 300
Device(config)#neighbor 10.1.1.3 route-map set-peer-address out
Device(config)#neighbor 10.1.1.1 remote-as 100
Device(config)#route-map set-peer-address permit 10
Device(config)#set ip next-hop peer-address
```

Related Commands

| Command | Description |
|--------------------------------|---|
| ip policy route-map | Identifies a route map to use for policy routing 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 length | Bases policy routing on the Level 3 length of a packet. |
| neighbor next-hop-self | Disables next hop processing of BGP updates on the router. |
| route-map (IP) | Defines the conditions for redistributing routes from one routing protocol to 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 that 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. |

show ip bgp

To display entries in the Border Gateway Protocol (BGP) routing table, use the **show ip bgp** command in user EXEC or privileged EXEC mode.

```
show ip bgp [{ip-address [{mask [{longer-prefixes [{injected}] | shorter-prefixes [{length}] |
best-path-reason | bestpath | multipaths | subnets}] | best-path-reason | bestpath | internal |
multipaths}] | all | oer-paths | prefix-list name | pending-prefixes | route-map name | version
{version-number | recent offset-value}]]
```

Syntax Description

| | |
|-------------------------------|--|
| <i>ip-address</i> | (Optional) IP address entered to filter the output to display only a particular host or network in the BGP routing table. |
| <i>mask</i> | (Optional) Mask to filter or match hosts that are part of the specified network. |
| longer-prefixes | (Optional) Displays the specified route and all more-specific routes. |
| injected | (Optional) Displays more-specific prefixes injected into the BGP routing table. |
| shorter-prefixes | (Optional) Displays the specified route and all less-specific routes. |
| <i>length</i> | (Optional) The prefix length. The range is a number from 0 to 32. |
| bestpath | (Optional) Displays the best path for this prefix. |
| best-path-reason | (Optional) Displays the reason why a path loses to the bestpath. Note If the best-path is yet to be selected, then the output will be 'Best Path Evaluation: No best path' |
| internal | (Optional) Displays the internal details for this prefix. |
| multipaths | (Optional) Displays multipaths for this prefix. |
| subnets | (Optional) Displays the subnet routes for the specified prefix. |
| all | (Optional) Displays all address family information in the BGP routing table. |
| oer-paths | (Optional) Displays Optimized Edge Routing (OER) controlled prefixes in the BGP routing table. |
| prefix-list name | (Optional) Filters the output based on the specified prefix list. |
| pending-prefixes | (Optional) Displays prefixes that are pending deletion from the BGP routing table. |
| route-map name | (Optional) Filters the output based on the specified route map. |
| version version-number | (Optional) Displays all prefixes with network versions greater than or equal to the specified version number. The range is from 1 to 4294967295. |
| recent offset-value | (Optional) Displays the offset from the current routing table version. The range is from 1 to 4294967295. |

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History**Command History**

| Release | Modification |
|--------------------------------|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Gibraltar 16.10.1 | The best-path-reason keyword was added to this command. BGP Path Installation Time-Stamp was added to the output of the command. BGP Peak Prefix Watermark was added to the output of the command. |

Usage Guidelines

The **show ip bgp** command is used to display the contents of the BGP routing table. The output can be filtered to display entries for a specific prefix, prefix length, and prefixes injected through a prefix list, route map, or conditional advertisement.

When changes are made to the network address, the network version number is incremented. Use the **version** keyword to view a specific network version.

show ip bgp: Example

The following sample output displays the BGP routing table:

```
Device#show ip bgp

BGP table version is 6, local router ID is 10.0.96.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath, b backup-path, x best-external, f
RT-Filter, a additional-path
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

      Network          Next Hop           Metric LocPrf Weight Path
-----
N*  10.0.0.1           10.0.0.3             0           0 3 ?
N*>
Nr  10.0.0.0/8          10.0.0.3             0           0 3 ?
Nr>
Nr  10.0.0.0/24         10.0.0.3             0           0 3 ?
V*> 10.0.2.0/24         0.0.0.0              0          32768 i
Vr> 10.0.3.0/24         10.0.3.5             0           0 4 ?
```

The table below describes the significant fields shown in the display.

Table 24: show ip bgp Field Descriptions

| Field | Description |
|-----------------------|--|
| BGP table version | Internal version number of the table. This number is incremented whenever the table changes. |
| local router ID | IP address of the router. |
| Status codes | Status of the table entry. The status is displayed at the beginning of each line in the table. It can be one of the following values: <ul style="list-style-type: none"> • s—The table entry is suppressed. • d—The table entry is dampened. • h—The table entry history. • *—The table entry is valid. • >—The table entry is the best entry to use for that network. • i—The table entry was learned via an internal BGP (iBGP) session. • r—The table entry is a RIB-failure. • S—The table entry is stale. • m—The table entry has multipath to use for that network. • b—The table entry has a backup path to use for that network. • x—The table entry has a best external route to use for the network. |
| Origin codes | Origin of the entry. The origin code is placed at the end of each line in the table. It can be one of the following values: <ul style="list-style-type: none"> • a—Path is selected as an additional path. • i—Entry originated from an Interior Gateway Protocol (IGP) and was advertised with a network router configuration command. • e—Entry originated from an Exterior Gateway Protocol (EGP). • ?—Origin of the path is not clear. Usually, this is a router that is redistributed into BGP from an IGP. |
| RPKI validation codes | If shown, the RPKI validation state for the network prefix, which is downloaded from the RPKI server. The codes are shown only if the bgp rpki server or neighbor announce rpki state command is configured. |
| Network | IP address of a network entity. |
| Next Hop | IP address of the next system that is used when forwarding a packet to the destination network. An entry of 0.0.0.0 indicates that the router has some non-BGP routes to this network. |
| Metric | If shown, the value of the interautonomous system metric. |

| Field | Description |
|------------|--|
| LocPrf | Local preference value as set with the set local-preference route-map configuration command. The default value is 100. |
| Weight | Weight of the route as set via autonomous system filters. |
| Path | Autonomous system paths to the destination network. There can be one entry in this field for each autonomous system in the path. |
| (stale) | Indicates that the following path for the specified autonomous system is marked as “stale” during a graceful restart process. |
| Updated on | The time at which the path is received or updated. |

show ip bgp (4-Byte Autonomous System Numbers): Example

The following sample output shows the BGP routing table with 4-byte autonomous system numbers, 65536 and 65550, shown under the Path field. This example requires Cisco IOS Release 12.0(32)SY8, 12.0(33)S3, 12.2(33)SRE, 12.2(33)XNE, 12.2(33)SX11, Cisco IOS XE Release 2.4, or a later release.

```
Device#show ip bgp

BGP table version is 4, local router ID is 172.16.1.99
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network        Next Hop           Metric LocPrf Weight Path
*> 10.1.1.0/24    192.168.1.2         0           0 65536  i
*> 10.2.2.0/24    192.168.3.2         0           0 65550  i
*> 172.16.1.0/24  0.0.0.0             0           0 32768  i
```

show ip bgp network: Example

The following sample output displays information about the 192.168.1.0 entry in the BGP routing table:

```
Device#show ip bgp 192.168.1.0

BGP routing table entry for 192.168.1.0/24, version 22
Paths: (2 available, best #2, table default)
  Additional-path
  Advertised to update-groups:
    3
  10 10
    192.168.3.2 from 172.16.1.2 (10.2.2.2)
      Origin IGP, metric 0, localpref 100, valid, internal, backup/repair
  10 10
    192.168.1.2 from 192.168.1.2 (10.3.3.3)
      Origin IGP, localpref 100, valid, external, best , recursive-via-connected
```

The following sample output displays information about the 10.3.3.3 255.255.255.255 entry in the BGP routing table:


```

Device#show ip bgp 10.3.3.3 255.255.255.255

BGP routing table entry for 10.3.3.3/32, version 35
Paths: (3 available, best #2, table default)
Multipath: eBGP
Flag: 0x860
  Advertised to update-groups:
    1
    200
      10.71.8.165 from 10.71.8.165 (192.168.0.102)
        Origin incomplete, localpref 100, valid, external, backup/repair
        Only allowed to recurse through connected route
    200
      10.71.11.165 from 10.71.11.165 (192.168.0.102)
        Origin incomplete, localpref 100, weight 100, valid, external, best
        Only allowed to recurse through connected route
    200
      10.71.10.165 from 10.71.10.165 (192.168.0.104)
        Origin incomplete, localpref 100, valid, external,
        Only allowed to recurse through connected route

```

The table below describes the significant fields shown in the display.

Table 25: show ip bgp ip-address Field Descriptions

| Field | Description |
|-----------------------------|---|
| BGP routing table entry for | IP address or network number of the routing table entry. |
| version | Internal version number of the table. This number is incremented whenever the table changes. |
| Paths | The number of available paths, and the number of installed best paths. This line displays “Default-IP-Routing-Table” when the best path is installed in the IP routing table. |
| Multipath | This field is displayed when multipath load sharing is enabled. This field will indicate if the multipaths are iBGP or eBGP. |
| Advertised to update-groups | The number of each update group for which advertisements are processed. |
| Origin | Origin of the entry. The origin can be IGP, EGP, or incomplete. This line displays the configured metric (0 if no metric is configured), the local preference value (100 is default), and the status and type of route (internal, external, multipath, best). |
| Extended Community | This field is displayed if the route carries an extended community attribute. The attribute code is displayed on this line. Information about the extended community is displayed on a subsequent line. |

show ip bgp all: Example

The following is sample output from the **show ip bgp** command entered with the **all** keyword. Information about all configured address families is displayed.

show ip bgp

Device#show ip bgp all

For address family: IPv4 Unicast *****

BGP table version is 27, local router ID is 10.1.1.1

Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure

Origin codes: i - IGP, e - EGP, ? - incomplete

| Network | Next Hop | Metric | LocPrf | Weight | Path |
|-------------------|---------------|--------|--------|--------|---------------------------|
| *> 10.1.1.0/24 | 0.0.0.0 | 0 | | 32768 | ? |
| *> 10.13.13.0/24 | 0.0.0.0 | 0 | | 32768 | ? |
| *> 10.15.15.0/24 | 0.0.0.0 | 0 | | 32768 | ? |
| *>i10.18.18.0/24 | 172.16.14.105 | 1388 | 91351 | 0 | 100 e |
| *>i10.100.0.0/16 | 172.16.14.107 | 262 | 272 | 0 | 1 2 3 i |
| *>i10.100.0.0/16 | 172.16.14.105 | 1388 | 91351 | 0 | 100 e |
| *>i10.101.0.0/16 | 172.16.14.105 | 1388 | 91351 | 0 | 100 e |
| *>i10.103.0.0/16 | 172.16.14.101 | 1388 | 173 | 173 | 100 e |
| *>i10.104.0.0/16 | 172.16.14.101 | 1388 | 173 | 173 | 100 e |
| *>i10.100.0.0/16 | 172.16.14.106 | 2219 | 20889 | 0 | 53285 33299 51178 47751 e |
| *>i10.101.0.0/16 | 172.16.14.106 | 2219 | 20889 | 0 | 53285 33299 51178 47751 e |
| * 10.100.0.0/16 | 172.16.14.109 | 2309 | | 0 | 200 300 e |
| *> | 172.16.14.108 | 1388 | | 0 | 100 e |
| * 10.101.0.0/16 | 172.16.14.109 | 2309 | | 0 | 200 300 e |
| *> | 172.16.14.108 | 1388 | | 0 | 100 e |
| *> 10.102.0.0/16 | 172.16.14.108 | 1388 | | 0 | 100 e |
| *> 172.16.14.0/24 | 0.0.0.0 | 0 | | 32768 | ? |
| *> 192.168.5.0 | 0.0.0.0 | 0 | | 32768 | ? |
| *> 10.80.0.0/16 | 172.16.14.108 | 1388 | | 0 | 50 e |
| *> 10.80.0.0/16 | 172.16.14.108 | 1388 | | 0 | 50 e |

For address family: VPNv4 Unicast *****

BGP table version is 21, local router ID is 10.1.1.1

Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure

Origin codes: i - IGP, e - EGP, ? - incomplete

| Network | Next Hop | Metric | LocPrf | Weight | Path |
|---|-------------|--------|--------|--------|---------------------------|
| Route Distinguisher: 1:1 (default for vrf vpn1) | | | | | |
| *> 10.1.1.0/24 | 192.168.4.3 | 1622 | | 0 | 100 53285 33299 51178 |
| {27016,57039,16690} e | | | | | |
| *> 10.1.2.0/24 | 192.168.4.3 | 1622 | | 0 | 100 53285 33299 51178 |
| {27016,57039,16690} e | | | | | |
| *> 10.1.3.0/24 | 192.168.4.3 | 1622 | | 0 | 100 53285 33299 51178 |
| {27016,57039,16690} e | | | | | |
| *> 10.1.4.0/24 | 192.168.4.3 | 1622 | | 0 | 100 53285 33299 51178 |
| {27016,57039,16690} e | | | | | |
| *> 10.1.5.0/24 | 192.168.4.3 | 1622 | | 0 | 100 53285 33299 51178 |
| {27016,57039,16690} e | | | | | |
| *>i172.17.1.0/24 | 10.3.3.3 | 10 | 30 | 0 | 53285 33299 51178 47751 ? |
| *>i172.17.2.0/24 | 10.3.3.3 | 10 | 30 | 0 | 53285 33299 51178 47751 ? |
| *>i172.17.3.0/24 | 10.3.3.3 | 10 | 30 | 0 | 53285 33299 51178 47751 ? |
| *>i172.17.4.0/24 | 10.3.3.3 | 10 | 30 | 0 | 53285 33299 51178 47751 ? |
| *>i172.17.5.0/24 | 10.3.3.3 | 10 | 30 | 0 | 53285 33299 51178 47751 ? |

For address family: IPv4 Multicast *****

BGP table version is 11, local router ID is 10.1.1.1

Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure

Origin codes: i - IGP, e - EGP, ? - incomplete

| Network | Next Hop | Metric | LocPrf | Weight | Path |
|--------------------|---------------|--------|--------|--------|-----------------------------|
| *> 10.40.40.0/26 | 172.16.14.110 | 2219 | | 0 | 21 22 {51178,47751,27016} e |
| * | 10.1.1.1 | 1622 | | 0 | 15 20 1 {2} e |
| *> 10.40.40.64/26 | 172.16.14.110 | 2219 | | 0 | 21 22 {51178,47751,27016} e |
| * | 10.1.1.1 | 1622 | | 0 | 15 20 1 {2} e |
| *> 10.40.40.128/26 | 172.16.14.110 | 2219 | | 0 | 21 22 {51178,47751,27016} e |
| * | 10.1.1.1 | 2563 | | 0 | 15 20 1 {2} e |
| *> 10.40.40.192/26 | 10.1.1.1 | 2563 | | 0 | 15 20 1 {2} e |

```
*> 10.40.41.0/26    10.1.1.1          1209             0 15 20 1 {2} e
*>i10.102.0.0/16   10.1.1.1          300    500        0 5 4 {101,102} e
*>i10.103.0.0/16   10.1.1.1          300    500        0 5 4 {101,102} e
For address family: NSAP Unicast *****
BGP table version is 1, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network                Next Hop                Metric LocPrf Weight Path
*  i45.0000.0002.0001.000c.00    49.0001.0000.0000.0a00                100    0 ?
*  i46.0001.0000.0000.0000.0a00    49.0001.0000.0000.0a00                100    0 ?
*  i47.0001.0000.0000.000b.00    49.0001.0000.0000.0a00                100    0 ?
*  i47.0001.0000.0000.000e.00    49.0001.0000.0000.0a00
```

show ip bgp longer-prefixes: Example

The following is sample output from the **show ip bgp longer-prefixes** command:

```
Device#show ip bgp 10.92.0.0 255.255.0.0 longer-prefixes
BGP table version is 1738, local router ID is 192.168.72.24
Status codes: s suppressed, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network                Next Hop                Metric LocPrf Weight Path
*> 10.92.0.0              10.92.72.30             8896             32768 ?
*                          10.92.72.30             0 109 108 ?
*> 10.92.1.0              10.92.72.30             8796             32768 ?
*                          10.92.72.30             0 109 108 ?
*> 10.92.11.0             10.92.72.30            42482            32768 ?
*                          10.92.72.30             0 109 108 ?
*> 10.92.14.0             10.92.72.30             8796             32768 ?
*                          10.92.72.30             0 109 108 ?
*> 10.92.15.0             10.92.72.30             8696             32768 ?
*                          10.92.72.30             0 109 108 ?
*> 10.92.16.0             10.92.72.30             1400             32768 ?
*                          10.92.72.30             0 109 108 ?
*> 10.92.17.0             10.92.72.30             1400             32768 ?
*                          10.92.72.30             0 109 108 ?
*> 10.92.18.0             10.92.72.30             8876             32768 ?
*                          10.92.72.30             0 109 108 ?
*> 10.92.19.0             10.92.72.30             8876             32768 ?
*                          10.92.72.30             0 109 108 ?
```

show ip bgp shorter-prefixes: Example

The following is sample output from the **show ip bgp shorter-prefixes** command. An 8-bit prefix length is specified.

```
Device#show ip bgp 172.16.0.0/16 shorter-prefixes 8
*> 172.16.0.0            10.0.0.2                0 ?
*                          10.0.0.2                0 200 ?
```

show ip bgp prefix-list: Example

The following is sample output from the **show ip bgp prefix-list** command:

```
Device#show ip bgp prefix-list ROUTE

BGP table version is 39, local router ID is 10.0.0.1
Status codes:s suppressed, d damped, h history, * valid, > best, i -
internal
Origin codes:i - IGP, e - EGP, ? - incomplete
   Network      Next Hop          Metric LocPrf Weight Path
*> 192.168.1.0  10.0.0.2          0             0 ?
*                10.0.0.2          0             0 200 ?
```

show ip bgp route-map: Example

The following is sample output from the **show ip bgp route-map** command:

```
Device#show ip bgp route-map LEARNED_PATH

BGP table version is 40, local router ID is 10.0.0.1
Status codes:s suppressed, d damped, h history, * valid, > best, i -
internal
Origin codes:i - IGP, e - EGP, ? - incomplete
   Network      Next Hop          Metric LocPrf Weight Path
*> 192.168.1.0  10.0.0.2          0             0 ?
*                10.0.0.2          0             0 200 ?
```

show ip bgp (Additional Paths): Example

The following output indicates (for each neighbor) whether any of the additional path tags (group-best, all, best 2 or best 3) are applied to the path. A line of output indicates rx pathid (received from neighbor) and tx pathid (announcing to neighbors). Note that the “Path advertised to update-groups:” is now per-path when the BGP Additional Paths feature is enabled.

```
Device#show ip bgp 10.0.0.1 255.255.255.224

BGP routing table entry for 10.0.0.1/28, version 82
Paths: (10 available, best #5, table default)
  Path advertised to update-groups:
    21      25
  Refresh Epoch 1
  20 50, (Received from a RR-client)
    192.0.2.1 from 192.0.2.1 (192.0.2.1)
      Origin IGP, metric 200, localpref 100, valid, internal, all
      Originator: 192.0.2.1, Cluster list: 2.2.2.2
      mpls labels in/out 16/nolabel
      rx pathid: 0, tx pathid: 0x9
      Updated on Aug 14 2018 18:30:39 PST
  Path advertised to update-groups:
    18      21
  Refresh Epoch 1
  30
    192.0.2.2 from 192.0.2.2 (192.0.2.2)
      Origin IGP, metric 200, localpref 100, valid, internal, group-best, all
```

```

    Originator: 192.0.2.2, Cluster list: 4.4.4.4
    mpls labels in/out 16/nolabel
    rx pathid: 0x1, tx pathid: 0x8
    Updated on Aug 14 2018 18:30:39 PST
  Path advertised to update-groups:
    16      18      19      20      21      22      24
    25      27
  Refresh Epoch 1
  10
    192.0.2.3 from 192.0.2.3 (192.0.2.3)
    Origin IGP, metric 200, localpref 100, valid, external, best2, all
    mpls labels in/out 16/nolabel
    rx pathid: 0, tx pathid: 0x7
    Updated on Aug 14 2018 18:30:39 PST
  Path advertised to update-groups:
    20      21      22      24      25
  Refresh Epoch 1
  10
    192.0.2.4 from 192.0.2.4 (192.0.2.4)
    Origin IGP, metric 300, localpref 100, valid, external, best3, all
    mpls labels in/out 16/nolabel
    rx pathid: 0, tx pathid: 0x6
    Updated on Jun 17 2018 11:12:30 PST
  Path advertised to update-groups:
    10      13      17      18      19      20      21
    22      23      24      25      26      27      28
  Refresh Epoch 1
  10
    192.0.2.5 from 192.0.2.5 (192.0.2.5)
    Origin IGP, metric 100, localpref 100, valid, external, best
    mpls labels in/out 16/nolabel
    rx pathid: 0, tx pathid: 0x0
    Updated on Jun 17 2018 11:12:30 PST
  Path advertised to update-groups:
    21
  Refresh Epoch 1
  30
    192.0.2.6 from 192.0.2.6 (192.0.2.6)
    Origin IGP, metric 200, localpref 100, valid, internal, all
    Originator: 192.0.2.6, Cluster list: 5.5.5.5
    mpls labels in/out 16/nolabel
    rx pathid: 0x1, tx pathid: 0x5
    Updated on Jun 17 2018 11:12:30 PST
  Path advertised to update-groups:
    18      23      24      26      28
  Refresh Epoch 1
  60 40, (Received from a RR-client)
    192.0.2.7 from 192.0.2.7 (192.0.2.7)
    Origin IGP, metric 250, localpref 100, valid, internal, group-best
    Originator: 192.0.2.7, Cluster list: 3.3.3.3
    mpls labels in/out 16/nolabel
    rx pathid: 0x2, tx pathid: 0x2
    Updated on Jun 17 2018 11:12:30 PST
  Path advertised to update-groups:
    25
  Refresh Epoch 1
  30 40, (Received from a RR-client)
    192.0.2.8 from 192.0.2.8 (192.0.2.8)
    Origin IGP, metric 200, localpref 100, valid, internal, all
    Originator: 192.0.2.8, Cluster list: 2.2.2.2
    mpls labels in/out 16/nolabel
    rx pathid: 0x1, tx pathid: 0x3
    Updated on Jun 17 2018 11:12:30 PST
  Path advertised to update-groups:

```

```

18          21          23          24          25          26          28
Refresh Epoch 1
20 40, (Received from a RR-client)
  192.0.2.9 from 192.0.2.9 (192.0.2.9)
    Origin IGP, metric 200, localpref 100, valid, internal, group-best, all
    Originator: 192.0.2.9, Cluster list: 2.2.2.2
    mpls labels in/out 16/nolabel
    rx pathid: 0x1, tx pathid: 0x4
    Updated on Jun 17 2018 18:34:12 PST
Path advertised to update-groups:
  21
Refresh Epoch 1
30 40
  192.0.2.9 from 192.0.2.9 (192.0.2.9)
    Origin IGP, metric 100, localpref 100, valid, internal, all
    Originator: 192.0.2.9, Cluster list: 4.4.4.4
    mpls labels in/out 16/nolabel
    rx pathid: 0x1, tx pathid: 0x1
    Updated on Jun 17 2018 18:34:12 PST

```

show ip bgp network (BGP Attribute Filter): Example

The following is sample output from the **show ip bgp** command that displays unknown and discarded path attributes:

```

Device#show ip bgp 192.0.2.0/32

BGP routing table entry for 192.0.2.0/32, version 0
Paths: (1 available, no best path)
  Refresh Epoch 1
  Local
    192.168.101.2 from 192.168.101.2 (192.168.101.2)
      Origin IGP, localpref 100, valid, internal
      unknown transitive attribute: flag 0xE0 type 0x81 length 0x20
        value 0000 0000 0000 0000 0000 0000 0000 0000
              0000 0000 0000 0000 0000 0000 0000 0000

      unknown transitive attribute: flag 0xE0 type 0x83 length 0x20
        value 0000 0000 0000 0000 0000 0000 0000 0000
              0000 0000 0000 0000 0000 0000 0000 0000

      discarded unknown attribute: flag 0x40 type 0x63 length 0x64
        value 0000 0000 0000 0000 0000 0000 0000 0000
              0000 0000 0000 0000 0000 0000 0000 0000

```

show ip bgp version: Example

The following is sample output from the **show ip bgp version** command:

```

Device#show ip bgp version

BGP table version is 5, local router ID is 10.2.4.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure, S Stale, m multipath, b backup-path, x best-external
Origin codes: i - IGP, e - EGP, ? - incomplete
Network Next Hop Metric LocPrf Weight Path
*> 192.168.34.2/24 10.0.0.1 0 0 1 ?

```

```
*> 192.168.35.2/24 10.0.0.1 0 0 1 ?
```

The following example shows how to display the network version:

```
Device#show ip bgp 192.168.34.2 | include version
```

```
BGP routing table entry for 192.168.34.2/24, version 5
```

The following sample output from the **show ip bgp version recent** command displays the prefix changes in the specified version:

```
Device#show ip bgp version recent 2
```

```
BGP table version is 5, local router ID is 10.2.4.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure, S Stale, m multipath, b backup-path, x best-external
Origin codes: i - IGP, e - EGP, ? - incomplete
```

| Network | Next Hop | Metric | LocPrf | Weight | Path |
|----------------------|----------|--------|--------|--------|------|
| *> 192.168.134.1/28 | 10.0.0.1 | 0 | | 0 | 1 ? |
| *> 192.168.134.19/28 | 10.0.0.1 | 0 | | 0 | 1 ? |
| *> 192.168.134.34/28 | 10.0.0.1 | 0 | | 0 | 1 ? |

```
Device#show ip bgp 80.230.70.96 best-path-reason
```

```
BGP routing table entry for 192.168.3.0/24, version 72
```

```
Paths: (2 available, best #2, table default)
```

```
Advertised to update-groups:
```

```
2
```

```
Refresh Epoch 1
```

```
2
```

```
10.0.101.1 from 10.0.101.1 (10.0.101.1)
```

```
Origin IGP, localpref 100, valid, external
```

```
Extended Community: RT:100:100
```

```
rx pathid: 0, tx pathid: 0
```

```
Updated on Aug 14 2018 18:34:12 PST
```

```
Best Path Evaluation: Path is younger
```

```
Refresh Epoch 1
```

```
1
```

```
10.0.96.254 from 10.0.96.254 (10.0.96.254)
```

```
Origin IGP, localpref 100, valid, external, best
```

```
rx pathid: 0, tx pathid: 0x0
```

```
Updated on Aug 14 2018 18:30:39 PST
```

```
Best Path Evaluation: Overall best path
```

The following sample output for the **show ip bgp summary** command shows the peak watermarks and their time-stamps for the peak number of route entries per neighbor bases:

```
Device#show ip bgp all summary
```

```
For address family: IPv4 Unicast
```

```
BGP router identifier 10.10.10.10, local AS number 1
```

```
BGP table version is 27, main routing table version 27
```

```
2 network entries using 496 bytes of memory
```

```
2 path entries using 272 bytes of memory
```

```
1/1 BGP path/bestpath attribute entries using 280 bytes of memory
```

```
1 BGP extended community entries using 24 bytes of memory
```

```
0 BGP route-map cache entries using 0 bytes of memory
```

```
0 BGP filter-list cache entries using 0 bytes of memory
```

```
BGP using 1072 total bytes of memory
```

```
BGP activity 58/54 prefixes, 110/106 paths, scan interval 60 secs
```

```
20 networks peaked at 00:03:50 Jul 28 2018 PST (00:00:32.833 ago)
```

show ip bgp

```
Neighbor      V          AS MsgRcvd MsgSent   TblVer  InQ  OutQ Up/Down  State/PfxRcd
11.11.11.11   4          1      0      0         1    0    0 00:20:09 Idle
```

```
For address family: L2VPN E-VPN
BGP router identifier 10.10.10.10, local AS number 1
BGP table version is 183, main routing table version 183
2 network entries using 688 bytes of memory
2 path entries using 416 bytes of memory
2/2 BGP path/bestpath attribute entries using 560 bytes of memory
1 BGP extended community entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 1688 total bytes of memory
BGP activity 58/54 prefixes, 110/106 paths, scan interval 60 secs
30 networks peaked at 00:35:36 Jul 28 2018 PST (00:00:47.321 ago)
```

```
Neighbor      V          AS MsgRcvd MsgSent   TblVer  InQ  OutQ Up/Down  State/PfxRcd
11.11.11.11   4          1      0      0         1    0    0 00:20:09 Idle
```

Related Commands

| Command | Description |
|------------------------------------|--|
| 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. |
| clear ip bgp | Resets BGP connections using hard or soft reconfiguration. |
| ip bgp community new-format | Configures BGP to display communities in the format AA:NN. |
| ip prefix-list | Creates a prefix list or adds a prefix-list entry. |
| route-map | Defines the conditions for redistributing routes from one routing protocol into another routing protocol. |
| router bgp | Configures the BGP routing process. |

show ip bgp neighbors

To display information about Border Gateway Protocol (BGP) and TCP connections to neighbors, use the **show ip bgp neighbors** command in user or privileged EXEC mode.

```
show ip bgp [{ipv4 {multicast | unicast} | vpnv4 all | vpnv6 unicast all}] neighbors [{slow ip-address
| ipv6-address [{advertised-routes | dampened-routes | flap-statistics | paths [reg-exp] | policy [detail]
| received prefix-filter | received-routes | routes}}}]
```

Syntax Description

| | |
|-------------------------------|---|
| ipv4 | (Optional) Displays peers in the IPv4 address family. |
| multicast | (Optional) Specifies IPv4 multicast address prefixes. |
| unicast | (Optional) Specifies IPv4 unicast address prefixes. |
| vpnv4 all | (Optional) Displays peers in the VPNv4 address family. |
| vpnv6 unicast all | (Optional) Displays peers in the VPNv6 address family. |
| slow | (Optional) Displays information about dynamically configured slow peers. |
| <i>ip-address</i> | (Optional) IP address of the IPv4 neighbor. If this argument is omitted, information about all neighbors is displayed. |
| <i>ipv6-address</i> | (Optional) IP address of the IPv6 neighbor. |
| advertised-routes | (Optional) Displays all routes that have been advertised to neighbors. |
| dampened-routes | (Optional) Displays the dampened routes received from the specified neighbor. |
| flap-statistics | (Optional) Displays the flap statistics of the routes learned from the specified neighbor (for external BGP peers only). |
| paths <i>reg-exp</i> | (Optional) Displays autonomous system paths learned from the specified neighbor. An optional regular expression can be used to filter the output. |
| policy | (Optional) Displays the policies applied to this neighbor per address family. |
| detail | (Optional) Displays detailed policy information such as route maps, prefix lists, community lists, access control lists (ACLs), and autonomous system path filter lists. |
| received prefix-filter | (Optional) Displays the prefix list (outbound route filter [ORF]) sent from the specified neighbor. |
| received-routes | (Optional) Displays all received routes (both accepted and rejected) from the specified neighbor. |
| routes | (Optional) Displays all routes that are received and accepted. The output displayed when this keyword is entered is a subset of the output displayed by the received-routes keyword. |

Command Default The output of this command displays information for all neighbors.

Command Modes User EXEC (>)
Privileged EXEC (#)

Command History

| Command History | Release | Modification |
|-----------------|--------------------------------|--|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| | Cisco IOS XE Gibraltar 16.10.1 | BGP Peak Prefix Watermark was added to the command output. |

Usage Guidelines Use the **show ip bgp neighbors** command to display BGP and TCP connection information for neighbor sessions. For BGP, this includes detailed neighbor attribute, capability, path, and prefix information. For TCP, this includes statistics related to BGP neighbor session establishment and maintenance.

Prefix activity is displayed based on the number of prefixes that are advertised and withdrawn. Policy denials display the number of routes that were advertised but then ignored based on the function or attribute that is displayed in the output.

Examples

Example output is different for the various keywords available for the **show ip bgp neighbors** command. Examples using the various keywords appear in the following sections.

show ip bgp neighbors: Example

The following example shows output for the BGP neighbor at 10.108.50.2. This neighbor is an internal BGP (iBGP) peer. This neighbor supports the route refresh and graceful restart capabilities.

```
Device#show ip bgp neighbors 10.108.50.2

BGP neighbor is 10.108.50.2, remote AS 1, internal link
  BGP version 4, remote router ID 192.168.252.252
  BGP state = Established, up for 00:24:25
  Last read 00:00:24, last write 00:00:24, hold time is 180, keepalive interval is
    60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received(old & new)
    MPLS Label capability: advertised and received
    Graceful Restart Capability: advertised
    Address family IPv4 Unicast: advertised and received
  Message statistics:
    InQ depth is 0
    OutQ depth is 0

      Sent      Rcvd
  Opens:           3         3
  Notifications:   0         0
  Updates:         0         0
  Keepalives:     113       112
  Route Refresh:   0         0
  Total:          116       115
  Default minimum time between advertisement runs is 5 seconds
  For address family: IPv4 Unicast
  BGP additional-paths computation is enabled
```

```

BGP advertise-best-external is enabled
BGP table version 1, neighbor version 1/0
Output queue size : 0
  Index 1, Offset 0, Mask 0x2
  1 update-group member

Prefix activity:
  Prefixes Current:      0      0
  Prefixes Total:       0      0
  Implicit Withdraw:    0      0
  Explicit Withdraw:    0      0
  Used as bestpath:    n/a     0
  Used as multipath:    n/a     0
                        Outbound Inbound
Local Policy Denied Prefixes:
  Total:                0      0
Number of NLRI in the update sent: max 0, min 0
Connections established 3; dropped 2
Last reset 00:24:26, due to Peer closed the session
External BGP neighbor may be up to 2 hops away.
Connection state is ESTAB, I/O status: 1, unread input bytes: 0
Connection is ECN Disabled
Local host: 10.108.50.1, Local port: 179
Foreign host: 10.108.50.2, Foreign port: 42698
Enqueued packets for retransmit: 0, input: 0 mis-ordered: 0 (0 bytes)
Event Timers (current time is 0x68B944):
Timer      Starts   Wakeups      Next
Retrans    27        0           0x0
TimeWait   0          0           0x0
AckHold    27        18          0x0
SendWnd    0          0           0x0
KeepAlive  0          0           0x0
GiveUp     0          0           0x0
PmtuAger   0          0           0x0
DeadWait   0          0           0x0
iss: 3915509457  snduna: 3915510016  sndnxt: 3915510016  sndwnd: 15826
irs: 233567076  rcvnxt: 233567616  rcvwnd: 15845  delrcvwnd: 539
SRTT: 292 ms, RTTO: 359 ms, RTV: 67 ms, KRTT: 0 ms
minRTT: 12 ms, maxRTT: 300 ms, ACK hold: 200 ms
Flags: passive open, nagle, gen tcbs
IP Precedence value : 6
Datagrams (max data segment is 1460 bytes):
Rcvd: 38 (out of order: 0), with data: 27, total data bytes: 539
Sent: 45 (retransmit: 0, fastretransmit: 0, partialack: 0, Second Congestion: 08

```

The table below describes the significant fields shown in the display. Fields that are preceded by the asterisk character (*) are displayed only when the counter has a nonzero value.

Table 26: show ip bgp neighbors Field Descriptions

| Field | Description |
|--|--|
| BGP neighbor | IP address of the BGP neighbor and its autonomous system number. |
| remote AS | Autonomous system number of the neighbor. |
| local AS 300 no-prepend (not shown in display) | Verifies that the local autonomous system number is not prepended to received external routes. This output supports the hiding of the local autonomous systems when a network administrator is migrating autonomous systems. |

| Field | Description |
|-----------------------------|--|
| internal link | “internal link” is displayed for iBGP neighbors; “external link” is displayed for external BGP (eBGP) neighbors. |
| BGP version | BGP version being used to communicate with the remote router. |
| remote router ID | IP address of the neighbor. |
| BGP state | Finite state machine (FSM) stage of session negotiation. |
| up for | Time, in hh:mm:ss, that the underlying TCP connection has been in existence. |
| Last read | Time, in hh:mm:ss, since BGP last received a message from this neighbor. |
| last write | Time, in hh:mm:ss, since BGP last sent a message to this neighbor. |
| hold time | Time, in seconds, that BGP will maintain the session with this neighbor without receiving messages. |
| keepalive interval | Time interval, in seconds, at which keepalive messages are transmitted to this neighbor. |
| Neighbor capabilities | BGP capabilities advertised and received from this neighbor. “advertised and received” is displayed when a capability is successfully exchanged between two routers. |
| Route refresh | Status of the route refresh capability. |
| MPLS Label capability | Indicates that MPLS labels are both sent and received by the eBGP peer. |
| Graceful Restart Capability | Status of the graceful restart capability. |
| Address family IPv4 Unicast | IP Version 4 unicast-specific properties of this neighbor. |
| Message statistics | Statistics organized by message type. |
| InQ depth is | Number of messages in the input queue. |
| OutQ depth is | Number of messages in the output queue. |
| Sent | Total number of transmitted messages. |
| Revd | Total number of received messages. |
| Opens | Number of open messages sent and received. |
| Notifications | Number of notification (error) messages sent and received. |
| Updates | Number of update messages sent and received. |
| Keepalives | Number of keepalive messages sent and received. |

| Field | Description |
|---------------------------------|---|
| Route Refresh | Number of route refresh request messages sent and received. |
| Total | Total number of messages sent and received. |
| Default minimum time between... | Time, in seconds, between advertisement transmissions. |
| For address family: | Address family to which the following fields refer. |
| BGP table version | Internal version number of the table. This is the primary routing table with which the neighbor has been updated. The number increments when the table changes. |
| neighbor version | Number used by the software to track prefixes that have been sent and those that need to be sent. |
| 1 update-group member | Number of the update-group member for this address family. |
| Prefix activity | Prefix statistics for this address family. |
| Prefixes Current | Number of prefixes accepted for this address family. |
| Prefixes Total | Total number of received prefixes. |
| Implicit Withdraw | Number of times that a prefix has been withdrawn and readvertised. |
| Explicit Withdraw | Number of times that a prefix has been withdrawn because it is no longer feasible. |
| Used as bestpath | Number of received prefixes installed as best paths. |
| Used as multipath | Number of received prefixes installed as multipaths. |
| * Saved (soft-reconfig) | Number of soft resets performed with a neighbor that supports soft reconfiguration. This field is displayed only if the counter has a nonzero value. |
| * History paths | This field is displayed only if the counter has a nonzero value. |
| * Invalid paths | Number of invalid paths. This field is displayed only if the counter has a nonzero value. |
| Local Policy Denied Prefixes | Prefixes denied due to local policy configuration. Counters are updated for inbound and outbound policy denials. The fields under this heading are displayed only if the counter has a nonzero value. |
| * route-map | Displays inbound and outbound route-map policy denials. |
| * filter-list | Displays inbound and outbound filter-list policy denials. |
| * prefix-list | Displays inbound and outbound prefix-list policy denials. |
| * Ext Community | Displays only outbound extended community policy denials. |
| * AS_PATH too long | Displays outbound AS_PATH length policy denials. |

| Field | Description |
|---|---|
| * AS_PATH loop | Displays outbound AS_PATH loop policy denials. |
| * AS_PATH confed info | Displays outbound confederation policy denials. |
| * AS_PATH contains AS 0 | Displays outbound denials of autonomous system 0. |
| * NEXT_HOP Martian | Displays outbound martian denials. |
| * NEXT_HOP non-local | Displays outbound nonlocal next-hop denials. |
| * NEXT_HOP is us | Displays outbound next-hop-self denials. |
| * CLUSTER_LIST loop | Displays outbound cluster-list loop denials. |
| * ORIGINATOR loop | Displays outbound denials of local originated routes. |
| * unsuppress-map | Displays inbound denials due to an unsuppress map. |
| * advertise-map | Displays inbound denials due to an advertise map. |
| * VPN Imported prefix | Displays inbound denials of VPN prefixes. |
| * Well-known Community | Displays inbound denials of well-known communities. |
| * SOO loop | Displays inbound denials due to site-of-origin. |
| * Bestpath from this peer | Displays inbound denials because the best path came from the local router. |
| * Suppressed due to dampening | Displays inbound denials because the neighbor or link is in a dampening state. |
| * Bestpath from iBGP peer | Displays inbound denials because the best path came from an iBGP neighbor. |
| * Incorrect RIB for CE | Displays inbound denials due to RIB errors for a customer edge (CE) router. |
| * BGP distribute-list | Displays inbound denials due to a distribute list. |
| Number of NLRIs... | Number of network layer reachability attributes in updates. |
| Current session network count peaked... | Displays the peak number of networks observed in the current session. |
| Highest network count observed at... | Displays the peak number of networks observed since startup. |
| Connections established | Number of times a TCP and BGP connection has been successfully established. |
| dropped | Number of times that a valid session has failed or been taken down. |
| Last reset | Time, in hh:mm:ss, since this peering session was last reset. The reason for the reset is displayed on this line. |

| Field | Description |
|--|--|
| External BGP neighbor may be... | Indicates that the BGP time to live (TTL) security check is enabled. The maximum number of hops that can separate the local and remote peer is displayed on this line. |
| Connection state | Connection status of the BGP peer. |
| unread input bytes | Number of bytes of packets still to be processed. |
| Connection is ECN Disabled | Explicit congestion notification status (enabled or disabled). |
| Local host: 10.108.50.1, Local port: 179 | IP address of the local BGP speaker. BGP port number 179. |
| Foreign host: 10.108.50.2, Foreign port: 42698 | Neighbor address and BGP destination port number. |
| Enqueued packets for retransmit: | Packets queued for retransmission by TCP. |
| Event Timers | TCP event timers. Counters are provided for starts and wakeups (expired timers). |
| Retrans | Number of times a packet has been retransmitted. |
| TimeWait | Time waiting for the retransmission timers to expire. |
| AckHold | Acknowledgment hold timer. |
| SendWnd | Transmission (send) window. |
| KeepAlive | Number of keepalive packets. |
| GiveUp | Number of times a packet is dropped due to no acknowledgment. |
| PmtuAger | Path MTU discovery timer. |
| DeadWait | Expiration timer for dead segments. |
| iss: | Initial packet transmission sequence number. |
| snduna: | Last transmission sequence number that has not been acknowledged. |
| sndnxt: | Next packet sequence number to be transmitted. |
| sndwnd: | TCP window size of the remote neighbor. |
| irs: | Initial packet receive sequence number. |
| rcvnxt: | Last receive sequence number that has been locally acknowledged. |
| rcvwnd: | TCP window size of the local host. |

| Field | Description |
|----------------------|--|
| delrcvwnd: | Delayed receive window—data the local host has read from the connection, but has not yet subtracted from the receive window the host has advertised to the remote host. The value in this field gradually increases until it is higher than a full-sized packet, at which point it is applied to the rcvwnd field. |
| SRTT: | A calculated smoothed round-trip timeout. |
| RTTO: | Round-trip timeout. |
| RTV: | Variance of the round-trip time. |
| KRTT: | New round-trip timeout (using the Karn algorithm). This field separately tracks the round-trip time of packets that have been re-sent. |
| minRTT: | Shortest recorded round-trip timeout (hard-wire value used for calculation). |
| maxRTT: | Longest recorded round-trip timeout. |
| ACK hold: | Length of time the local host will delay an acknowledgment to carry (piggyback) additional data. |
| IP Precedence value: | IP precedence of the BGP packets. |
| Datagrams | Number of update packets received from a neighbor. |
| Rcvd: | Number of received packets. |
| out of order: | Number of packets received out of sequence. |
| with data | Number of update packets sent with data. |
| total data bytes | Total amount of data received, in bytes. |
| Sent | Number of update packets sent. |
| Second Congestion | Number of update packets with data sent. |
| Datagrams: Rcvd | Number of update packets received from a neighbor. |
| retransmit | Number of packets retransmitted. |
| fastretransmit | Number of duplicate acknowledgments retransmitted for an out of order segment before the retransmission timer expires. |
| partialack | Number of retransmissions for partial acknowledgments (transmissions before or without subsequent acknowledgments). |
| Second Congestion | Number of second retransmissions sent due to congestion. |

show ip bgp neighbors (4-Byte Autonomous System Numbers)

The following partial example shows output for several external BGP neighbors in autonomous systems with 4-byte autonomous system numbers, 65536 and 65550. This example requires Cisco IOS Release 12.0(32)SY8, 12.0(33)S3, 12.2(33)SRE, 12.2(33)XNE, 12.2(33)SX11, Cisco IOS XE Release 2.4, or a later release.

```
Device#show ip bgp neighbors
```

```
BGP neighbor is 192.168.1.2, remote AS 65536, external link
  BGP version 4, remote router ID 0.0.0.0
  BGP state = Idle
  Last read 02:03:38, last write 02:03:38, hold time is 120, keepalive interval is 70
seconds
  Configured hold time is 120, keepalive interval is 70 seconds
  Minimum holdtime from neighbor is 0 seconds
.
.
.
BGP neighbor is 192.168.3.2, remote AS 65550, external link
  Description: finance
  BGP version 4, remote router ID 0.0.0.0
  BGP state = Idle
  Last read 02:03:48, last write 02:03:48, hold time is 120, keepalive interval is 70
seconds
  Configured hold time is 120, keepalive interval is 70 seconds
  Minimum holdtime from neighbor is 0 seconds
```

show ip bgp neighbors advertised-routes

The following example displays routes advertised for only the 172.16.232.178 neighbor:

```
Device#show ip bgp neighbors 172.16.232.178 advertised-routes
```

```
BGP table version is 27, local router ID is 172.16.232.181
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete
Network          Next Hop          Metric LocPrf Weight Path
*>i10.0.0.0      172.16.232.179    0      100      0  ?
*> 10.20.2.0     10.0.0.0          0              32768 i
```

The table below describes the significant fields shown in the display.

Table 27: show ip bgp neighbors advertised-routes Field Descriptions

| Field | Description |
|-------------------|---|
| BGP table version | Internal version number of the table. This is the primary routing table with which the neighbor has been updated. The number increments when the table changes. |
| local router ID | IP address of the local BGP speaker. |

| Field | Description |
|--------------|--|
| Status codes | Status of the table entry. The status is displayed at the beginning of each line in the table. It can be one of the following values: <ul style="list-style-type: none"> • s—The table entry is suppressed. • d—The table entry is dampened and will not be advertised to BGP neighbors. • h—The table entry does not contain the best path based on historical information. • *—The table entry is valid. • >—The table entry is the best entry to use for that network. • i—The table entry was learned via an internal BGP (iBGP) session. |
| Origin codes | Origin of the entry. The origin code is placed at the end of each line in the table. It can be one of the following values: <ul style="list-style-type: none"> • i—Entry originated from Interior Gateway Protocol (IGP) and was advertised with a network router configuration command. • e—Entry originated from Exterior Gateway Protocol (EGP). • ?—Origin of the path is not clear. Usually, this is a route that is redistributed into BGP from an IGP. |
| Network | IP address of a network entity. |
| Next Hop | IP address of the next system used to forward a packet to the destination network. An entry of 0.0.0.0 indicates that there are non-BGP routes in the path to the destination network. |
| Metric | If shown, this is the value of the interautonomous system metric. This field is not used frequently. |
| LocPrf | Local preference value as set with the set local-preference route-map configuration command. The default value is 100. |
| Weight | Weight of the route as set via autonomous system filters. |
| Path | Autonomous system paths to the destination network. There can be one entry in this field for each autonomous system in the path. |

show ip bgp neighbors check-control-plane-failure

The following is sample output from the **show ip bgp neighbors** command entered with the **check-control-plane-failure** option configured:

```
Device#show ip bgp neighbors 10.10.10.1
```

```
BGP neighbor is 10.10.10.1, remote AS 10, internal link
  Fall over configured for session
  BFD is configured. BFD peer is Up. Using BFD to detect fast fallover (single-hop) with
  c-bit check-control-plane-failure.
```

```
Inherits from template cbit-tps for session parameters
BGP version 4, remote router ID 10.7.7.7
BGP state = Established, up for 00:03:55
Last read 00:00:02, last write 00:00:21, hold time is 180, keepalive interval is 60 seconds

Neighbor sessions:
  1 active, is not multisession capable (disabled)
Neighbor capabilities:
  Route refresh: advertised and received(new)
  Four-octets ASN Capability: advertised and received
  Address family IPv4 Unicast: advertised and received
  Enhanced Refresh Capability: advertised and received
  Multisession Capability:
  Stateful switchover support enabled: NO for session 1
```

show ip bgp neighbors paths

The following is sample output from the **show ip bgp neighbors** command entered with the **paths** keyword:

```
Device#show ip bgp neighbors 172.29.232.178 paths 10

Address      Refcount Metric Path
0x60E577B0      2      40 10 ?
```

The table below describes the significant fields shown in the display.

Table 28: show ip bgp neighbors paths Field Descriptions

| Field | Description |
|----------|---|
| Address | Internal address where the path is stored. |
| Refcount | Number of routes using that path. |
| Metric | Multi Exit Discriminator (MED) metric for the path. (The name of this metric for BGP versions 2 and 3 is INTER_AS.) |
| Path | Autonomous system path for that route, followed by the origin code for that route. |

show ip bgp neighbors received prefix-filter

The following example shows that a prefix list that filters all routes in the 10.0.0.0 network has been received from the 192.168.20.72 neighbor:

```
Device#show ip bgp neighbors 192.168.20.72 received prefix-filter

Address family:IPv4 Unicast
ip prefix-list 192.168.20.72:1 entries
  seq 5 deny 10.0.0.0/8 le 32
```

The table below describes the significant fields shown in the display.

Table 29: show ip bgp neighbors received prefix-filter Field Descriptions

| Field | Description |
|----------------|---|
| Address family | Address family mode in which the prefix filter is received. |
| ip prefix-list | Prefix list sent from the specified neighbor. |

show ip bgp neighbors policy

The following sample output shows the policies applied to the neighbor at 192.168.1.2. The output displays both inherited policies and policies configured on the neighbor device. Inherited policies are policies that the neighbor inherits from a peer group or a peer-policy template.

```
Device#show ip bgp neighbors 192.168.1.2 policy

Neighbor: 192.168.1.2, Address-Family: IPv4 Unicast
Locally configured policies:
  route-map ROUTE in
Inherited policies:
  prefix-list NO-MARKETING in
  route-map ROUTE in
  weight 300
  maximum-prefix 10000
```

BGP Attribute Filter and Enhanced Attribute Error Handling

The following is sample output from the **show ip bgp neighbors** command that indicates the discard attribute values and treat-as-withdraw attribute values configured. It also provides a count of received Updates matching a treat-as-withdraw attribute, a count of received Updates matching a discard attribute, and a count of received malformed Updates that are treat-as-withdraw.

```
Device#show ip bgp vpnv4 all neighbors 10.0.103.1

BGP neighbor is 10.0.103.1, remote AS 100, internal link
Path-attribute treat-as-withdraw inbound
Path-attribute treat-as-withdraw value 128
Path-attribute treat-as-withdraw 128 in: count 2
Path-attribute discard 128 inbound
Path-attribute discard 128 in: count 2

      Outbound   Inbound
Local Policy Denied Prefixes:  -----  -----
MALFORM treat as withdraw:      0          1
Total:                          0          1
```

BGP Additional Paths

The following output indicates that the neighbor is capable of advertising additional paths and sending additional paths it receives. It is also capable of receiving additional paths and advertised paths.

```
Device#show ip bgp neighbors 10.108.50.2
```

```
BGP neighbor is 10.108.50.2, remote AS 1, internal link
  BGP version 4, remote router ID 192.168.252.252
  BGP state = Established, up for 00:24:25
  Last read 00:00:24, last write 00:00:24, hold time is 180, keepalive interval is 60 seconds
```

```
Neighbor capabilities:
  Additional paths Send: advertised and received
  Additional paths Receive: advertised and received
  Route refresh: advertised and received(old & new)
  Graceful Restart Capabilty: advertised and received
  Address family IPv4 Unicast: advertised and received
```

BGP—Multiple Cluster IDs

In the following output, the cluster ID of the neighbor is displayed. (The vertical bar and letter “i” for “include” cause the device to display only lines that include the user’s input after the “i”, in this case, “cluster-id.”) The cluster ID displayed is the one directly configured through a neighbor or a template.

```
Device#show ip bgp neighbors 192.168.2.2 | i cluster-id

Configured with the cluster-id 192.168.15.6
```

BGP Peak Prefix Watermark

The following sample output shows the peak watermarks and their timestamps displayed for the peak number of route entries per neighbor bases:

```
Device#show ip bgp ipv4 unicast neighbors 11.11.11.11

BGP neighbor is 11.11.11.11, remote AS 1, internal link
  BGP version 4, remote router ID 0.0.0.0
  BGP state = Idle, down for 00:01:43
  Neighbor sessions:
    0 active, is not multiseession capable (disabled)
  Stateful switchover support enabled: NO
  Do log neighbor state changes (via global configuration)
  Default minimum time between advertisement runs is 0 seconds
```

```
For address family: IPv4 Unicast
  BGP table version 27, neighbor version 1/27
  Output queue size : 0
  Index 0, Advertise bit 0
  Slow-peer detection is disabled
  Slow-peer split-update-group dynamic is disabled
```

| Prefix activity: | Sent | Rcvd |
|--------------------|------|------|
| Prefixes Current: | 0 | 0 |
| Prefixes Total: | 0 | 0 |
| Implicit Withdraw: | 0 | 0 |
| Explicit Withdraw: | 0 | 0 |
| Used as bestpath: | n/a | 0 |
| Used as multipath: | n/a | 0 |
| Used as secondary: | n/a | 0 |

| | Outbound | Inbound |
|-------------------------------|----------|---------|
| Local Policy Denied Prefixes: | ----- | ----- |
| Total: | 0 | 0 |

Number of NLRIs in the update sent: max 2, min 0

```

Current session network count peaked at 20 entries at 00:00:23 Aug 8 2018 PST (00:01:29.156
ago).
Highest network count observed at 20 entries at 23:55:32 Aug 7 2018 PST (00:06:20.156
ago).
Last detected as dynamic slow peer: never
Dynamic slow peer recovered: never
Refresh Epoch: 1
Last Sent Refresh Start-of-rib: never
Last Sent Refresh End-of-rib: never
Last Received Refresh Start-of-rib: never
Last Received Refresh End-of-rib: never

Refresh activity:
Refresh Start-of-RIB
Refresh End-of-RIB
Sent
----
0
0
Rcvd
----
0
0

```

BGP Soft Inbound and Outbound Refresh Time

In the following example, the times of occurrence of the soft inbound and outbound refresh, to or from the given neighbour, are displayed:

```
Device#show ip bgp l2vpn evpn neighbors 11.11.11.11
```

```

BGP neighbor is 11.11.11.11, remote AS 1, internal link
BGP version 4, remote router ID 11.11.11.11
BGP state = Established, up for 00:14:06
Last read 00:00:21, last write 00:00:28, hold time is 180, keepalive
.....
Do log neighbor state changes (via global configuration)

Default minimum time between advertisement runs is 0 seconds

```

```

For address family: L2VPN E-VPN
Session: 11.11.11.11
BGP table version 30, neighbor version 30/0
Output queue size : 0
Index 1, Advertise bit 0
1 update-group member
Community attribute sent to this neighbor
Extended-community attribute sent to this neighbor
.....
.....
Last detected as dynamic slow peer: never
Dynamic slow peer recovered: never
Refresh Epoch: 2
Last Sent Refresh Start-of-rib: never
Last Sent Refresh End-of-rib: never
Last Received Refresh Start-of-rib: 00:14:06
Last Received Refresh End-of-rib: 00:14:06
Refresh-In took 0 seconds

Refresh activity:
Refresh Start-of-RIB
Refresh End-of-RIB
Sent
----
0
0
Rcvd
----
1
1

```

```

Address tracking is enabled, the RIB does have a route to 11.11.11.11
Route to peer address reachability Up: 1; Down: 0
Last notification 00:14:07
Connections established 1; dropped 0
.....

```

```

.....
Packets received in fast path: 0, fast processed: 0, slow path: 0
fast lock acquisition failures: 0, slow path: 0
TCP Semaphore      0x7FA8A0AE7BA0  FREE

```

Related Commands

| Command | Description |
|--|--|
| 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. |
| bgp enhanced-error | Restores the default behavior of treating Update messages that have a malformed attribute as withdrawn, or includes iBGP peers in the Enhanced Attribute Error Handling feature. |
| neighbor path-attribute discard | Configures the device to discard unwanted Update messages from the specified neighbor that contain a specified path attribute. |
| neighbor path-attribute treat-as-withdraw | Configures the device to withdraw from the specified neighbor unwanted Update messages that contain a specified attribute. |
| neighbor send-label | Enables a BGP router to send MPLS labels with BGP routes to a neighboring BGP router. |
| neighbor send-label explicit-null | Enables a BGP router to send MPLS labels with explicit-null information for a CSC-CE router and BGP routes to a neighboring CSC-PE router. |
| router bgp | Configures the BGP routing process. |

show ip bgp ipv6 unicast

To display entries in the Internet Protocol version 6 (IPv6) Border Gateway Protocol (BGP) routing table, use the **show ip bgp ipv6 unicast** command in user EXEC or privileged EXEC mode.

```
show ip bgp ipv6 unicast [ prefix / length ]
```

Syntax Description

| | |
|------------------------|--|
| <i>prefix / length</i> | <p>(Optional) IPv6 network number and length of the IPv6 prefix, entered to display a particular network in the IPv6 BGP routing table.</p> <ul style="list-style-type: none"> The <i>length</i> is a decimal value that indicates how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). A slash mark must precede the decimal value. |
|------------------------|--|

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

The **show ip bgp ipv6 unicast** command provides output similar to the **show ip bgp** command, except that it is IPv6 specific.

Examples

The following is sample output from the **show bgp ipv6 unicast prefix/length** command, showing the RPKI state of the path:

```
Device# show bgp ipv6 unicast 2010::1/128

BGP routing table entry for 2010::1/128, version 5
Paths: (1 available, best #1, table default)
  Advertised to update-groups:
    1          2
  Refresh Epoch 1
    3
  2002::1 (FE80::A8BB:CCFF:FE00:300) from 2002::1 (10.0.0.3)
    Origin IGP, metric 0, localpref 100, valid, external, best
    path 079ECBD0 RPKI State not found
```

The table below describes the significant fields shown in the display.

Table 30: show ip bgp ipv6 Field Descriptions

| Field | Description |
|-----------------------------|---|
| BGP routing table entry for | IPv6 prefix and prefix length, internal version number of the table. This number is incremented whenever the table changes. |

| Field | Description |
|--|--|
| Paths: | Number of routes available to destination. |
| Advertised to update-groups: | Update group numbers. |
| 3 | Autonomous system number. |
| 2002::1 (FE80::A8BB:CCFF:FE00:300) from 2002::1 (10.0.0.3) | Address of the neighbor from which the path was received, link local address of the neighbor, from address of the neighbor, BGP router ID of the neighbor. |
| Origin | Indicates the origin of the entry. |
| metric | If shown, the value of the interautonomous system metric. |
| localpref | Local preference value as set with the set local-preference route-map configuration command. The default value is 100. |
| valid | Path is legitimate. |
| external | Path is an External Border Gateway Protocol (EBGP) path. |
| best path | Path is flagged as the best path; number indicates which path in memory. |
| RPKI State | RPKI state of the network prefix shown at the beginning of the output. The state could be valid, invalid, or not found. |

Related Commands

| Command | Description |
|-----------------------|---|
| clear bgp ipv6 | Resets an IPv6 BGP connection or session. |

show ip eigrp interfaces

To display information about interfaces that are configured for the Enhanced Interior Gateway Routing Protocol (EIGRP), use the **show ip eigrp interfaces** command in user EXEC or privileged EXEC mode.

```
show ip eigrp [vrf vrf-name] [autonomous-system-number] interfaces [type number] [{detail}]
```

Syntax Description

| | |
|---------------------------------|--|
| vrf <i>vrf-name</i> | (Optional) Displays information about the specified virtual routing and forwarding (VRF) instance. |
| <i>autonomous-system-number</i> | (Optional) Autonomous system number whose output needs to be filtered. |
| <i>type</i> | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| <i>number</i> | (Optional) Interface or subinterface number. For more information about the numbering syntax for your networking device, use the question mark (?) online help function. |
| detail | (Optional) Displays detailed information about EIGRP interfaces for a specific EIGRP process. |

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

Use the **show ip eigrp interfaces** command to display active EIGRP interfaces and EIGRP-specific interface settings and statistics. The optional *type number* argument and the **detail** keyword can be entered in any order.

If an interface is specified, only information about that interface is displayed. Otherwise, information about all interfaces on which EIGRP is running is displayed.

If an autonomous system is specified, only the routing process for the specified autonomous system is displayed. Otherwise, all EIGRP processes are displayed.

This command can be used to display information about EIGRP named and EIGRP autonomous system configurations.

This command displays the same information as the **show eigrp address-family interfaces** command. Cisco recommends using the **show eigrp address-family interfaces** command.

Examples

The following is sample output from the **show ip eigrp interfaces** command:

```
Device#show ip eigrp interfaces
EIGRP-IPv4 Interfaces for AS(60)
      Xmit Queue   Mean   Pacing Time   Multicast   Pending
```

```

Interface    Peers    Un/Reliable    SRTT    Un/Reliable    Flow Timer    Routes
Di0          0        0/0            0       11/434        0             0
Et0          1        0/0            337     0/10          0             0
SE0:1.16    1        0/0            10      1/63          103           0
Tu0          1        0/0            330     0/16          0             0

```

The following sample output from the **show ip eigrp interfaces detail** command displays detailed information about all active EIGRP interfaces:

```
Device#show ip eigrp interfaces detail
```

```

EIGRP-IPv4 Interfaces for AS(1)
              Xmit Queue    PeerQ          Mean    Pacing Time    Multicast    Pending
Interface      Peers Un/Reliable    Un/Reliable  SRTT    Un/Reliable    Flow Timer    Routes
Et0/0          1      0/0            0/0         525     0/2            3264          0
Hello-interval is 5, Hold-time is 15
  Split-horizon is enabled
  Next xmit serial <none>
  Packetized sent/expedited: 3/0
  Hello's sent/expedited: 6/2
  Un/reliable mcasts: 0/6  Un/reliable ucasts: 7/4
  Mcast exceptions: 1  CR packets: 1  ACKs suppressed: 0
  Retransmissions sent: 1  Out-of-sequence rcvd: 0
  Topology-ids on interface - 0
  Authentication mode is not set

```

The following sample output from the **show ip eigrp interfaces detail** command displays detailed information about a specific interface on which the **no ip next-hop self** command is configured along with the **no-ecmp-mode** option:

```
Device#show ip eigrp interfaces detail tunnel 0
```

```

EIGRP-IPv4 Interfaces for AS(1)
              Xmit Queue    PeerQ          Mean    Pacing Time    Multicast    Pending
Interface      Peers Un/Reliable    Un/Reliable  SRTT    Un/Reliable    Flow Timer    Routes
Tu0/0          2      0/0            0/0          2       0/0            50             0
Hello-interval is 5, Hold-time is 15
  Split-horizon is disabled
  Next xmit serial <none>
  Packetized sent/expedited: 24/3
  Hello's sent/expedited: 28083/9
  Un/reliable mcasts: 0/19  Un/reliable ucasts: 18/64
  Mcast exceptions: 5  CR packets: 5  ACKs suppressed: 0
  Retransmissions sent: 52  Out-of-sequence rcvd: 2
  Next-hop-self disabled, next-hop info forwarded, ECMP mode Enabled
  Topology-ids on interface - 0
  Authentication mode is not set

```

The table below describes the significant fields shown in the displays.

Table 31: show ip eigrp interfaces Field Descriptions

| Field | Description |
|-----------|---|
| Interface | Interface on which EIGRP is configured. |
| Peers | Number of directly connected EIGRP neighbors. |

| Field | Description |
|---------------------------|--|
| PeerQ Un/Reliable | Number of unreliable and reliable packets queued for transmission to specific peers on the interface. |
| Xmit Queue Un/Reliable | Number of packets remaining in the Unreliable and Reliable transmit queues. |
| Mean SRTT | Mean smooth round-trip time (SRTT) interval (in seconds). |
| Pacing Time Un/Reliable | Pacing time (in seconds) used to determine when EIGRP packets (unreliable and reliable) should be sent out of the interface . |
| Multicast Flow Timer | Maximum number of seconds for which the device will send multicast EIGRP packets. |
| Pending Routes | Number of routes in the transmit queue waiting to be sent. |
| Packetized sent/expedited | Number of EIGRP routes that have been prepared for sending packets to neighbors on an interface, and the number of times multiple routes were stored in a single packet. |
| Hello's sent/expedited | Number of EIGRP hello packets that have been sent on an interface and packets that were expedited. |

Related Commands

| Command | Description |
|---|--|
| show eigrp address-family interfaces | Displays information about address family interfaces configured for EIGRP. |
| show ip eigrp neighbors | Displays neighbors discovered by EIGRP. |

show ip eigrp neighbors

To display neighbors discovered by the Enhanced Interior Gateway Routing Protocol (EIGRP), use the **show ip eigrp neighbors** command in privileged EXEC mode.

show ip eigrp [**vrf** *vrf-name*] [*autonomous-system-number*] **neighbors** [{**static** | **detail**}] [*interface-type interface-number*]

| Syntax Description | | |
|--|--|--|
| vrf <i>vrf-name</i> | (Optional) Displays information about the specified VPN Routing and Forwarding (VRF) instance. | |
| <i>autonomous-system-number</i> | (Optional) Autonomous-system-number-specific output is displayed. | |
| static | (Optional) Displays static neighbors. | |
| detail | (Optional) Displays detailed neighbor information. | |
| <i>interface-type interface-number</i> | (Optional) Interface-specific output is displayed. | |

Command Modes Privileged EXEC (#)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines The **show ip eigrp neighbors** command can be used to display information about EIGRP named and EIGRP autonomous-system configurations. Use the **show ip eigrp neighbors** command to display dynamic and static neighbor states. You can use this command for also debugging certain types of transport problems.

This command displays the same information as the **show eigrp address-family neighbors** command. Cisco recommends that you use the **show eigrp address-family neighbors** command.

Examples

The following is sample output from the **show ip eigrp neighbors** command:

```
Device#show ip eigrp neighbors
H   Address                Interface      Hold Uptime    SRTT  RTO  Q  Seq
   (sec)                  (ms)          (sec)
0   10.1.1.2                 Et0/0         13 00:00:03 1996 5000 0 5
2   10.1.1.9                 Et0/0         14 00:02:24 206  5000 0 5
1   10.1.1.2.3              Et0/1         11 00:20:39 2202 5000 0 5
```

The table below describes the significant fields shown in the display.

Table 32: show ip eigrp neighbors Field Descriptions

| Field | Description |
|-----------|---|
| Address | IP address of the EIGRP peer. |
| Interface | Interface on which the router is receiving hello packets from the peer. |

| Field | Description |
|---------|---|
| Hold | Time in seconds for which EIGRP waits to hear from the peer before declaring it down. |
| Uptime | Elapsed time (in hours:minutes: seconds) since the local router first heard from this neighbor. |
| SRTT | Smooth round-trip time. This is the number of milliseconds required for an EIGRP packet to be sent to this neighbor and for the local router to receive an acknowledgment of that packet. |
| RTO | Retransmission timeout (in milliseconds). This is the amount of time the software waits before resending a packet from the retransmission queue to a neighbor. |
| Q Cnt | Number of EIGRP packets (update, query, and reply) that the software is waiting to send. |
| Seq Num | Sequence number of the last update, query, or reply packet that was received from this neighbor. |

The following is sample output from the **show ip eigrp neighbors detail** command:

```
Device#show ip eigrp neighbors detail
EIGRP-IPv4 VR(foo) Address-Family Neighbors for AS(1)
H   Address                Interface          Hold Uptime    SRTT   RTO   Q   Seq
   (sec)                (ms)            Cnt Num
0   192.168.10.1           Gi2/0             12 00:00:21 1600  5000  0   3
   Static neighbor (Lisp Encap)
   Version 8.0/2.0, Retrans: 0, Retries: 0, Prefixes: 1
   Topology-ids from peer - 0
```

The table below describes the significant fields shown in the display.

Table 33: show ip eigrp neighbors detail Field Descriptions

| Field | Description |
|------------|---|
| H | This column lists the order in which a peering session was established with the specified neighbor. The order is specified with sequential numbering starting with 0. |
| Address | IP address of the EIGRP peer. |
| Interface | Interface on which the router is receiving hello packets from the peer. |
| Hold | Time in seconds for which EIGRP waits to hear from the peer before declaring it down. |
| Lisp Encap | Indicates that routes from this neighbor are LISP encapsulated. |
| Uptime | Elapsed time (in hours:minutes: seconds) since the local router first heard from this neighbor. |
| SRTT | Smooth round-trip time. This is the number of milliseconds required for an EIGRP packet to be sent to this neighbor and for the local router to receive an acknowledgment of that packet. |
| RTO | Retransmission timeout (in milliseconds). This is the amount of time the software waits before resending a packet from the retransmission queue to a neighbor. |
| Q Cnt | Number of EIGRP packets (update, query, and reply) that the software is waiting to send. |

| Field | Description |
|---------|--|
| Seq Num | Sequence number of the last update, query, or reply packet that was received from this neighbor. |
| Version | The software version that the specified peer is running. |
| Retrans | Number of times that a packet has been retransmitted. |
| Retries | Number of times an attempt was made to retransmit a packet. |

Related Commands

| Command | Description |
|--|---|
| show eigrp address-family neighbors | Displays neighbors discovered by EIGRP. |

show ip eigrp topology

To display Enhanced Interior Gateway Routing Protocol (EIGRP) topology table entries, use the **show ip eigrp topology** command in user EXEC or privileged EXEC mode.

show ip eigrp topology [{ *network* [{ *mask* }] *prefix* | **active** | **all-links** | **detail-links** | **pending** | **secondary-paths** | **summary** | **zero-successors** }

| Syntax Description | | |
|--------------------|------------------------|--|
| | <i>network</i> | (Optional) Network address. |
| | <i>mask</i> | (Optional) Network mask. |
| | <i>prefix</i> | (Optional) Network prefix in the format <i><network>/<length></i> , for example, 192.168.0.0/16. |
| | active | (Optional) Displays all topology entries that are in the active state. |
| | all-links | (Optional) Displays all the entries in the EIGRP topology table (including nonfeasible successor sources). |
| | detail-links | (Optional) Displays all the topology entries with additional details. |
| | pending | (Optional) Displays all the entries in the EIGRP topology table that are either waiting for an update from a neighbor or to reply to a neighbor. |
| | secondary-paths | (Optional) Displays the secondary paths in the topology. |
| | summary | (Optional) Displays a summary of the EIGRP topology table. |
| | zero-successors | (Optional) Displays the available routes that have zero successors. |

Command Default If this command is used without any of the optional keywords, only topology entries with feasible successors are displayed and only feasible paths are shown.

Command Modes User EXEC (>)
Privileged EXEC (#)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines Use the **show ip eigrp topology** command to display topology entries, feasible and nonfeasible paths, metrics, and states. This command can be used without any arguments or keywords to display only topology entries with feasible successors and feasible paths. The **all-links** keyword displays all the paths, whether feasible or not, and the **detail-links** keyword displays additional details about these paths.

Use this command to display information about EIGRP named and EIGRP autonomous system configurations. This command displays the same information as the **show eigrp address-family topology** command. We recommend that you use the **show eigrp address-family topology** command.

Examples

The following is a sample output from the **show ip eigrp topology** command:

```
Device# show ip eigrp topology

EIGRP-IPv4 Topology Table for AS(1)/ID(10.0.0.1)
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - Reply status, s - sia status
P 10.0.0.0/8, 1 successors, FD is 409600
   via 192.0.2.1 (409600/128256), Ethernet0/0
P 192.16.1.0/24, 1 successors, FD is 409600
   via 192.0.2.1 (409600/128256), Ethernet0/0
P 10.0.0.0/8, 1 successors, FD is 281600
   via Summary (281600/0), Null0
P 10.0.1.0/24, 1 successors, FD is 281600
   via Connected, Ethernet0/0
```

The following is a sample output from the **show ip eigrp topology prefix** command, and displays detailed information about a single prefix. The prefix shown is an EIGRP internal route.

```
Device# show ip eigrp topology 10.0.0.0/8

EIGRP-IPv4 VR(vr1) Topology Entry for AS(1)/ID(10.1.1.2) for 10.0.0.0/8
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 82329600, RIB is 643200
  Descriptor Blocks:
    10.1.1.1 (Ethernet2/0), from 10.1.1.1, Send flag is 0x0
      Composite metric is (82329600/163840), route is Internal
      Vector metric:
        Minimum bandwidth is 16000 Kbit
        Total delay is 631250000 picoseconds
        Reliability is 255/255
        Load is 1/55
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 10.1.1.1
```

The following is a sample output from the **show ip eigrp topology prefix** command, and displays detailed information about a single prefix. The prefix shown is an EIGRP external route.

```
Device# show ip eigrp topology 192.16.1.0/24

EIGRP-IPv4 Topology Entry for AS(1)/ID(10.0.0.1) for 192.16.1.0/24
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 409600, RIB is 643200
  Descriptor Blocks:
    172.16.1.0/24 (Ethernet0/0), from 10.0.1.2, Send flag is 0x0
      Composite metric is (409600/128256), route is External
      Vector metric:
        Minimum bandwidth is 10000 Kbit
        Total delay is 6000 picoseconds
        Reliability is 255/255
        Load is 1/55
        Minimum MTU is 1500
        Hop count is 1
        Originating router is 192.16.1.0/24
      External data:
        AS number of route is 0
        External protocol is Connected, external metric is 0
        Administrator tag is 0 (0x00000000)
```

The following is a sample output from the **show ip eigrp topology prefix** command displays Equal Cost Multipath (ECMP) mode information when the **no ip next-hop-self** command is configured without the **no-ecmp-mode** keyword in an EIGRP topology. The ECMP mode provides information

about the path that is being advertised. If there is more than one successor, the top-most path is advertised as the default path over all the interfaces, and ECMP Mode: Advertise by default is displayed in the output. If any path other than the default path is advertised, ECMP Mode: Advertise out <Interface name> is displayed.

The topology table displays entries of routes for a particular prefix. The routes are sorted based on metric, next-hop, and infosource. In a Dynamic Multipoint VPN (DMVPN) scenario, routes with the same metric and next hop are sorted based on infosource. The top route in the ECMP is always advertised.

```
Device# show ip eigrp topology 192.168.10.0/24

EIGRP-IPv4 Topology Entry for AS(1)/ID(10.10.100.100) for 192.168.10.0/24
State is Passive, Query origin flag is 1, 2 Successor(s), FD is 284160
Descriptor Blocks:
  10.100.1.0 (Tunnel0), from 10.100.0.1, Send flag is 0x0
    Composite metric is (284160/281600), route is Internal
    Vector metric:
      Minimum bandwidth is 10000 Kbit
      Total delay is 1100 microseconds
      Reliability is 255/255
      Load is 1/55
      Minimum MTU is 1400
      Hop count is 1
      Originating router is 10.10.1.1
    ECMP Mode: Advertise by default
  10.100.0.2 (Tunnel1), from 10.100.0.2, Send flag is 0x0
    Composite metric is (284160/281600), route is Internal
    Vector metric:
      Minimum bandwidth is 10000 Kbit
      Total delay is 1100 microseconds
      Reliability is 255/255
      Load is 1/55
      Minimum MTU is 1400
      Hop count is 1
      Originating router is 10.10.2.2
    ECMP Mode: Advertise out Tunnel1
```

The following is a sample output from the **show ip eigrp topology all-links** command, and displays all the paths, including those that are not feasible:

```
Device# show ip eigrp topology all-links

EIGRP-IPv4 Topology Table for AS(1)/ID(10.0.0.1)
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
P 172.16.1.0/24, 1 successors, FD is 409600, serno 14
   via 10.10.1.2 (409600/128256), Ethernet0/0
   via 10.1.4.3 (2586111744/2585599744), Serial3/0, serno 18
```

The following is a sample output from the **show ip eigrp topology detail-links** command, and displays additional details about routes:

```
Device# show ip eigrp topology detail-links

EIGRP-IPv4 Topology Table for AS(1)/ID(10.0.0.1)
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
P 10.0.0.0/8, 1 successors, FD is 409600, serno 6
   via 10.10.1.2 (409600/128256), Ethernet0/0
P 172.16.1.0/24, 1 successors, FD is 409600, serno 14
   via 10.10.1.2 (409600/128256), Ethernet0/0
P 10.0.0.0/8, 1 successors, FD is 281600, serno 3
```

```

    via Summary (281600/0), Null0
P 10.1.1.0/24, 1 successors, FD is 281600, serno 1
    via Connected, Ethernet0/0

```

The following table describes the significant fields shown in the above examples:

Table 34: show ip eigrp topology Field Descriptions

| Field | Description |
|------------|---|
| Codes | <p>State of this topology table entry. Passive and Active refer to the EIGRP state with respect to the destination. Update, Query, and Reply refer to the type of packet that is being sent.</p> <ul style="list-style-type: none"> • P - Passive: Indicates that no EIGRP computations are being performed for this route. • A - Active: Indicates that EIGRP computations are being performed for this route. • U - Update: Indicates that a pending update packet is waiting to be sent for this route. • Q - Query: Indicates that a pending query packet is waiting to be sent for this route. • R - Reply: Indicates that a pending reply packet is waiting to be sent for this route. • r - Reply status: Indicates that EIGRP has sent a query for the route and is waiting for a reply from the specified path. • s - sia status: Indicates that the EIGRP query packet is in stuck-in-active (SIA) status. |
| successors | Number of successors. This number corresponds to the number of next hops in the IP routing table. If successors is capitalized, then the route or the next hop is in a transition state. |
| serno | Serial number. |
| FD | Feasible distance. This is the best metric to reach the destination or the best metric that was known when the route became active. This value is used in the feasibility condition check. If the reported distance of the device is less than the feasible distance, the feasibility condition is met and that route becomes a feasible successor. After the software determines that it has a feasible successor, the software need not send a query for that destination. |
| via | Next-hop address that advertises the passive route. |

Related Commands

| Command | Description |
|------------------------------------|--|
| show eigrp address-family topology | Displays entries in the EIGRP address-family topology table. |

show ip eigrp traffic

To display the number of Enhanced Interior Gateway Routing Protocol (EIGRP) packets sent and received, use the **show ip eigrp traffic** command in privileged EXEC mode.

show ip eigrp [**vrf** {*vrf-name* | *}] [*autonomous-system-number*] **traffic**

| Syntax Description | | |
|---------------------------------|--|--|
| vrf <i>vrf-name</i> | (Optional) Displays information about the specified VRF. | |
| vrf * | (Optional) Displays information about all VRFs. | |
| <i>autonomous-system-number</i> | (Optional) Autonomous system number. | |

Command Modes Privileged EXEC (#)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines This command can be used to display information about EIGRP named configurations and EIGRP autonomous-system (AS) configurations.

This command displays the same information as the **show eigrp address-family traffic** command. Cisco recommends using the **show eigrp address-family traffic** command.

Examples

The following is sample output from the **show ip eigrp traffic** command:

```
Device#show ip eigrp traffic
EIGRP-IPv4 Traffic Statistics for AS(60)
Hellos sent/received: 21429/2809
Updates sent/received: 22/17
Queries sent/received: 0/0
Replies sent/received: 0/0
Acks sent/received: 16/13
SIA-Queries sent/received: 0/0
SIA-Replies sent/received: 0/0
Hello Process ID: 204
PDM Process ID: 203
Socket Queue: 0/2000/2/0 (current/max/highest/drops)
Input Queue: 0/2000/2/0 (current/max/highest/drops)
```

The table below describes the significant fields shown in the display.

Table 35: show ip eigrp traffic Field Descriptions

| Field | Description |
|-----------------------|---|
| Hellos sent/received | Number of hello packets sent and received. |
| Updates sent/received | Number of update packets sent and received. |
| Queries sent/received | Number of query packets sent and received. |

| Field | Description |
|---------------------------|---|
| Replies sent/received | Number of reply packets sent and received. |
| Acks sent/received | Number of acknowledgement packets sent and received. |
| SIA-Queries sent/received | Number of stuck in active query packets sent and received. |
| SIA-Replies sent/received | Number of stuck in active reply packets sent and received. |
| Hello Process ID | Hello process identifier. |
| PDM Process ID | Protocol-dependent module IOS process identifier. |
| Socket Queue | The IP to EIGRP Hello Process socket queue counters. |
| Input queue | The EIGRP Hello Process to EIGRP PDM socket queue counters. |

Related Commands

| Command | Description |
|--|---|
| show eigrp address-family traffic | Displays the number of EIGRP packets sent and received. |

show ip ospf

To display general information about Open Shortest Path First (OSPF) routing processes, use the **show ip ospf** command in user EXEC or privileged EXEC mode.

show ip ospf [*process-id*]

| | |
|---------------------------|--|
| Syntax Description | <i>process-id</i> (Optional) Process ID. If this argument is included, only information for the specified routing process is included. |
|---------------------------|--|

Command Modes User EXEC Privileged EXEC

| | | |
|------------------------|------------------------------|------------------------------|
| Command History | Mainline Release | Modification |
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Examples

The following is sample output from the **show ip ospf** command when entered without a specific OSPF process ID:

```
Device#show ip ospf

Routing Process "ospf 201" with ID 10.0.0.1 and Domain ID 10.20.0.1
Supports only single TOS(TOS0) routes
Supports opaque LSA
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
LSA group pacing timer 100 secs
Interface flood pacing timer 55 msec
Retransmission pacing timer 100 msec
Number of external LSA 0. Checksum Sum 0x0
Number of opaque AS LSA 0. Checksum Sum 0x0
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 2. 2 normal 0 stub 0 nssa
External flood list length 0
  Area BACKBONE (0)
    Number of interfaces in this area is 2
    Area has message digest authentication
    SPF algorithm executed 4 times
    Area ranges are
    Number of LSA 4. Checksum Sum 0x29BEB
    Number of opaque link LSA 0. Checksum Sum 0x0
    Number of DCbitless LSA 3
    Number of indication LSA 0
    Number of DoNotAge LSA 0
    Flood list length 0
  Area 172.16.26.0
    Number of interfaces in this area is 0
    Area has no authentication
    SPF algorithm executed 1 times
    Area ranges are
      192.168.0.0/16 Passive Advertise
    Number of LSA 1. Checksum Sum 0x44FD
    Number of opaque link LSA 0. Checksum Sum 0x0
    Number of DCbitless LSA 1
```

```

Number of indication LSA 1
Number of DoNotAge LSA 0
Flood list length 0

```

Cisco IOS Release 12.2(18)SXE, 12.0(31)S, and 12.4(4)T

The following is sample output from the **show ip ospf** command to verify that the BFD feature has been enabled for OSPF process 123. The relevant command output is shown in bold in the output.

```

Device#show ip ospf

Routing Process "ospf 123" with ID 172.16.10.1
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
Initial SPF schedule delay 5000 msecs
Minimum hold time between two consecutive SPF's 10000 msecs
Maximum wait time between two consecutive SPF's 10000 msecs
Incremental-SPF disabled
Minimum LSA interval 5 secs
Minimum LSA arrival 1000 msecs
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msecs
Retransmission pacing timer 66 msecs
Number of external LSA 0. Checksum Sum 0x000000
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
External flood list length 0
BFD is enabled
Area BACKBONE(0)
  Number of interfaces in this area is 2
  Area has no authentication
  SPF algorithm last executed 00:00:03.708 ago
  SPF algorithm executed 27 times
  Area ranges are
  Number of LSA 3. Checksum Sum 0x00AEF1
  Number of opaque link LSA 0. Checksum Sum 0x000000
  Number of DCbitless LSA 0
  Number of indication LSA 0
  Number of DoNotAge LSA 0
  Flood list length 0

```

The table below describes the significant fields shown in the display.

Table 36: show ip ospf Field Descriptions

| Field | Description |
|---|--|
| Routing process "ospf 201" with ID 10.0.0.1 | Process ID and OSPF router ID. |
| Supports... | Number of types of service supported (Type 0 only). |
| SPF schedule delay | Delay time (in seconds) of SPF calculations. |
| Minimum LSA interval | Minimum interval (in seconds) between link-state advertisements. |

| Field | Description |
|--|---|
| LSA group pacing timer | Configured LSA group pacing timer (in seconds). |
| Interface flood pacing timer | Configured LSA flood pacing timer (in milliseconds). |
| Retransmission pacing timer | Configured LSA retransmission pacing timer (in milliseconds). |
| Number of external LSA | Number of external link-state advertisements. |
| Number of opaque AS LSA | Number of opaque link-state advertisements. |
| Number of DCbitless external and opaque AS LSA | Number of demand circuit external and opaque link-state advertisements. |
| Number of DoNotAge external and opaque AS LSA | Number of do not age external and opaque link-state advertisements. |
| Number of areas in this router is | Number of areas configured for the router. |
| External flood list length | External flood list length. |
| BFD is enabled | BFD has been enabled on the OSPF process. |

The following is an excerpt of output from the **show ip ospf** command when the OSPF Forwarding Address Suppression in Type-5 LSAs feature is configured:

```

Device#show ip ospf
.
.
.
Area 2
  Number of interfaces in this area is 4
  It is a NSSA area
  Perform type-7/type-5 LSA translation, suppress forwarding address
.
.
.
Routing Process "ospf 1" with ID 192.168.0.1
  Supports only single TOS(TOS0) routes
  Supports opaque LSA
  Supports Link-local Signaling (LLS)
  Initial SPF schedule delay 5000 msec
  Minimum hold time between two consecutive SPF's 10000 msec
  Maximum wait time between two consecutive SPF's 10000 msec
  Incremental-SPF disabled
  Minimum LSA interval 5 secs
  Minimum LSA arrival 1000 msec
  LSA group pacing timer 240 secs
  Interface flood pacing timer 33 msec
  Retransmission pacing timer 66 msec
  Number of external LSA 0. Checksum Sum 0x0
  Number of opaque AS LSA 0. Checksum Sum 0x0
  Number of DCbitless external and opaque AS LSA 0
  Number of DoNotAge external and opaque AS LSA 0
  Number of areas in this router is 0. 0 normal 0 stub 0 nssa
  External flood list length 0

```

The table below describes the significant fields shown in the display.

Table 37: show ip ospf Field Descriptions

| Field | Description |
|--|--|
| Area | OSPF area and tag. |
| Number of interfaces... | Number of interfaces configured in the area. |
| It is... | Possible types are internal, area border, or autonomous system boundary. |
| Routing process "ospf 1" with ID 192.168.0.1 | Process ID and OSPF router ID. |
| Supports... | Number of types of service supported (Type 0 only). |
| Initial SPF schedule delay | Delay time of SPF calculations at startup. |
| Minimum hold time | Minimum hold time (in milliseconds) between consecutive SPF calculations. |
| Maximum wait time | Maximum wait time (in milliseconds) between consecutive SPF calculations. |
| Incremental-SPF | Status of incremental SPF calculations. |
| Minimum LSA... | Minimum time interval (in seconds) between link-state advertisements, and minimum arrival time (in milliseconds) of link-state advertisements, |
| LSA group pacing timer | Configured LSA group pacing timer (in seconds). |
| Interface flood pacing timer | Configured LSA flood pacing timer (in milliseconds). |
| Retransmission pacing timer | Configured LSA retransmission pacing timer (in milliseconds). |
| Number of... | Number and type of link-state advertisements that have been received. |
| Number of external LSA | Number of external link-state advertisements. |
| Number of opaque AS LSA | Number of opaque link-state advertisements. |
| Number of DCbitless external and opaque AS LSA | Number of demand circuit external and opaque link-state advertisements. |
| Number of DoNotAge external and opaque AS LSA | Number of do not age external and opaque link-state advertisements. |
| Number of areas in this router is | Number of areas configured for the router listed by type. |
| External flood list length | External flood list length. |

The following is sample output from the **show ip ospf** command. In this example, the user had configured the **redistribution maximum-prefix** command to set a limit of 2000 redistributed routes. SPF throttling was configured with the **timer throttlespf** command.

```
Device#show ip ospf 1
Routing Process "ospf 1" with ID 10.0.0.1
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
It is an autonomous system boundary router
Redistributing External Routes from,
    static, includes subnets in redistribution
Maximum limit of redistributed prefixes 2000
Threshold for warning message 75%
Initial SPF schedule delay 5000 msec
Minimum hold time between two consecutive SPF's 10000 msec
Maximum wait time between two consecutive SPF's 10000 msec
```

The table below describes the significant fields shown in the display.

Table 38: show ip ospf Field Descriptions

| Field | Description |
|---|--|
| Routing process "ospf 1" with ID 10.0.0.1 | Process ID and OSPF router ID. |
| Supports ... | Number of Types of Service supported. |
| It is ... | Possible types are internal, area border, or autonomous system boundary router. |
| Redistributing External Routes from | Lists of redistributed routes, by protocol. |
| Maximum limit of redistributed prefixes | Value set in the redistribution maximum-prefix command to set a limit on the number of redistributed routes. |
| Threshold for warning message | Percentage set in the redistribution maximum-prefix command for the threshold number of redistributed routes needed to cause a warning message. The default is 75 percent of the maximum limit. |
| Initial SPF schedule delay | Delay (in milliseconds) before initial SPF schedule for SPF throttling. Configured with the timer throttlespf command. |
| Minimum hold time between two consecutive SPF's | Minimum hold time (in milliseconds) between two consecutive SPF calculations for SPF throttling. Configured with the timer throttlespf command. |
| Maximum wait time between two consecutive SPF's | Maximum wait time (in milliseconds) between two consecutive SPF calculations for SPF throttling. Configured with the timer throttlespf command. |
| Number of areas | Number of areas in router, area addresses, and so on. |

The following is sample output from the **show ip ospf** command. In this example, the user had configured LSA throttling, and those lines of output are displayed in bold.

```

Device#show ip ospf 1
Routing Process "ospf 4" with ID 10.10.24.4
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
Initial SPF schedule delay 5000 msec
Minimum hold time between two consecutive SPF's 10000 msec
Maximum wait time between two consecutive SPF's 10000 msec
Incremental-SPF disabled
Initial LSA throttle delay 100 msec
Minimum hold time for LSA throttle 10000 msec

Maximum wait time for LSA throttle 45000 msec
Minimum LSA arrival 1000 msec
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msec
Retransmission pacing timer 66 msec
Number of external LSA 0. Checksum Sum 0x0
Number of opaque AS LSA 0. Checksum Sum 0x0
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
External flood list length 0
  Area 24
    Number of interfaces in this area is 2
    Area has no authentication
    SPF algorithm last executed 04:28:18.396 ago
    SPF algorithm executed 8 times
    Area ranges are
    Number of LSA 4. Checksum Sum 0x23EB9
    Number of opaque link LSA 0. Checksum Sum 0x0
    Number of DCbitless LSA 0
    Number of indication LSA 0
    Number of DoNotAge LSA 0
    Flood list length 0

```

The following is sample **show ip ospf** command. In this example, the user had configured the **redistribution maximum-prefix** command to set a limit of 2000 redistributed routes. SPF throttling was configured with the **timer throttle spf** command.

```

Device#show ip ospf 1
Routing Process "ospf 1" with ID 192.168.0.0
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
It is an autonomous system boundary router
Redistributing External Routes from,
  static, includes subnets in redistribution
  Maximum limit of redistributed prefixes 2000
  Threshold for warning message 75%
Initial SPF schedule delay 5000 msec
Minimum hold time between two consecutive SPF's 10000 msec
Maximum wait time between two consecutive SPF's 10000 msec

```

The table below describes the significant fields shown in the display.

Table 39: show ip ospf Field Descriptions

| Field | Description |
|---|---|
| Routing process "ospf 1" with ID 192.168.0.0. | Process ID and OSPF router ID. |
| Supports ... | Number of TOS supported. |
| It is ... | Possible types are internal, area border, or autonomous system boundary routers. |
| Redistributing External Routes from | Lists of redistributed routes, by protocol. |
| Maximum limit of redistributed prefixes | Value set in the redistributionmaximum-prefix command to set a limit on the number of redistributed routes. |
| Threshold for warning message | Percentage set in the redistributionmaximum-prefix command for the threshold number of redistributed routes needed to cause a warning message. The default is 75 percent of the maximum limit. |
| Initial SPF schedule delay | Delay (in milliseconds) before the initial SPF schedule for SPF throttling. Configured with the timersthrottlespf command. |
| Minimum hold time between two consecutive SPF's | Minimum hold time (in milliseconds) between two consecutive SPF calculations for SPF throttling. Configured with the timersthrottlespf command. |
| Maximum wait time between two consecutive SPF's | Maximum wait time (in milliseconds) between two consecutive SPF calculations for SPF throttling. Configured with the timersthrottlespf command. |
| Number of areas | Number of areas in router, area addresses, and so on. |

The following is sample output from the **show ip ospf** command. In this example, the user had configured LSA throttling, and those lines of output are displayed in bold.

```
Device#show ip ospf 1
Routing Process "ospf 4" with ID 10.10.24.4
  Supports only single TOS(TOS0) routes
  Supports opaque LSA
  Supports Link-local Signaling (LLS)
  Initial SPF schedule delay 5000 msec
  Minimum hold time between two consecutive SPF's 10000 msec
  Maximum wait time between two consecutive SPF's 10000 msec
  Incremental-SPF disabled
  Initial LSA throttle delay 100 msec
  Minimum hold time for LSA throttle 10000 msec
  Maximum wait time for LSA throttle 45000 msec
Minimum LSA arrival 1000 msec
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msec
Retransmission pacing timer 66 msec
Number of external LSA 0. Checksum Sum 0x0
Number of opaque AS LSA 0. Checksum Sum 0x0
Number of DChitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
```

```
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
External flood list length 0
  Area 24
    Number of interfaces in this area is 2
    Area has no authentication
    SPF algorithm last executed 04:28:18.396 ago
    SPF algorithm executed 8 times
    Area ranges are
    Number of LSA 4. Checksum Sum 0x23EB9
    Number of opaque link LSA 0. Checksum Sum 0x0
    Number of DCbitless LSA 0
    Number of indication LSA 0
    Number of DoNotAge LSA 0
    Flood list length 0
```

show ip ospf border-routers

To display the internal Open Shortest Path First (OSPF) routing table entries to an Area Border Router (ABR) and Autonomous System Boundary Router (ASBR), use the **show ip ospf border-routers** command in privileged EXEC mode.

show ip ospf border-routers

Syntax Description

This command has no arguments or keywords.

Command Modes

Privileged EXEC

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Examples

The following is sample output from the **show ip ospf border-routers** command:

```
Device#show ip ospf border-routers
OSPF Process 109 internal Routing Table
Codes: i - Intra-area route, I - Inter-area route
i 192.168.97.53 [10] via 172.16.1.53, Serial0, ABR, Area 0.0.0.3, SPF 3
i 192.168.103.51 [10] via 192.168.96.51, Serial0, ABR, Area 0.0.0.3, SPF 3
I 192.168.103.52 [22] via 192.168.96.51, Serial0, ASBR, Area 0.0.0.3, SPF 3
I 192.168.103.52 [22] via 172.16.1.53, Serial0, ASBR, Area 0.0.0.3, SPF 3
```

The table below describes the significant fields shown in the display.

Table 40: show ip ospf border-routers Field Descriptions

| Field | Description |
|-----------------|--|
| 192.168.97.53 | Router ID of the destination. |
| [10] | Cost of using this route. |
| via 172.16.1.53 | Next hop toward the destination. |
| Serial0 | Interface type for the outgoing interface. |
| ABR | The router type of the destination; it is either an ABR or ASBR or both. |
| Area | The area ID of the area from which this route is learned. |
| SPF 3 | The internal number of the shortest path first (SPF) calculation that installs this route. |

show ip ospf database

To display lists of information related to the Open Shortest Path First (OSPF) database for a specific router, use the **show ip ospf database** command in EXEC mode.

```

show ip ospf [process-id area-id] database
show ip ospf [process-id area-id] database [adv-router [ip-address]]
show ip ospf [process-id area-id] database [asbr-summary] [link-state-id]
show ip ospf [process-id area-id] database [asbr-summary] [link-state-id] [adv-router [ip-address]]
show ip ospf [process-id area-id] database [asbr-summary] [link-state-id] [self-originate]
[link-state-id]
show ip ospf [process-id area-id] database [database-summary]
show ip ospf [process-id] database [external] [link-state-id]
show ip ospf [process-id] database [external] [link-state-id] [adv-router [ip-address]]
show ip ospf [process-id area-id] database [external] [link-state-id] [self-originate] [link-state-id]
show ip ospf [process-id area-id] database [network] [link-state-id]
show ip ospf [process-id area-id] database [network] [link-state-id] [adv-router [ip-address]]
show ip ospf [process-id area-id] database [network] [link-state-id] [self-originate] [link-state-id]
show ip ospf [process-id area-id] database [nssa-external] [link-state-id]
show ip ospf [process-id area-id] database [nssa-external] [link-state-id] [adv-router [ip-address]]
show ip ospf [process-id area-id] database [nssa-external] [link-state-id] [self-originate] [link-state-id]
show ip ospf [process-id area-id] database [router] [link-state-id]
show ip ospf [process-id area-id] database [router] [adv-router [ip-address]]
show ip ospf [process-id area-id] database [router] [self-originate] [link-state-id]
show ip ospf [process-id area-id] database [self-originate] [link-state-id]
show ip ospf [process-id area-id] database [summary] [link-state-id]
show ip ospf [process-id area-id] database [summary] [link-state-id] [adv-router [ip-address]]
show ip ospf [process-id area-id] database [summary] [link-state-id] [self-originate] [link-state-id]

```

Syntax Description

| | |
|--------------------------------|--|
| <i>process-id</i> | (Optional) Internal identification. It is locally assigned and can be any positive integer. The number used here is the number assigned administratively when enabling the OSPF routing process. |
| <i>area-id</i> | (Optional) Area number associated with the OSPF address range defined in the network router configuration command used to define the particular area. |
| adv-router [ip-address] | (Optional) Displays all the LSAs of the specified router. If no IP address is included, the information is about the local router itself (in this case, the same as self-originate). |

| | |
|-------------------------|--|
| <i>link-state-id</i> | <p>(Optional) Portion of the Internet environment that is being described by the advertisement. The value entered depends on the advertisement's LS type. It must be entered in the form of an IP address.</p> <p>When the link state advertisement is describing a network, the <i>link-state-id</i> can take one of two forms:</p> <p>The network's IP address (as in type 3 summary link advertisements and in autonomous system external link advertisements).</p> <p>A derived address obtained from the link state ID. (Note that masking a network links advertisement's link state ID with the network's subnet mask yields the network's IP address.)</p> <p>When the link state advertisement is describing a router, the link state ID is always the described router's OSPF router ID.</p> <p>When an autonomous system external advertisement (LS Type = 5) is describing a default route, its link state ID is set to Default Destination (0.0.0.0).</p> |
| asbr-summary | (Optional) Displays information only about the autonomous system boundary router summary LSAs. |
| database-summary | (Optional) Displays how many of each type of LSA for each area there are in the database, and the total. |
| external | (Optional) Displays information only about the external LSAs. |
| network | (Optional) Displays information only about the network LSAs. |
| nssa-external | (Optional) Displays information only about the NSSA external LSAs. |
| router | (Optional) Displays information only about the router LSAs. |
| self-originate | (Optional) Displays only self-originated LSAs (from the local router). |
| summary | (Optional) Displays information only about the summary LSAs. |

Command Modes EXEC

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines The various forms of this command deliver information about different OSPF link state advertisements.

Examples The following is sample output from the **show ip ospf database** command when no arguments or keywords are used:

```
Device#show ip ospf database
OSPF Router with id(192.168.239.66) (Process ID 300)
      Displaying Router Link States(Area 0.0.0.0)
  Link ID      ADV Router   Age         Seq#         Checksum     Link count
172.16.21.6   172.16.21.6   1731       0x80002CFB   0x69BC       8
```

show ip ospf database

```

172.16.21.5 172.16.21.5 1112 0x800009D2 0xA2B8 5
172.16.1.2 172.16.1.2 1662 0x80000A98 0x4CB6 9
172.16.1.1 172.16.1.1 1115 0x800009B6 0x5F2C 1
172.16.1.5 172.16.1.5 1691 0x80002BC 0x2A1A 5
172.16.65.6 172.16.65.6 1395 0x80001947 0xEEE1 4
172.16.241.5 172.16.241.5 1161 0x8000007C 0x7C70 1
172.16.27.6 172.16.27.6 1723 0x80000548 0x8641 4
172.16.70.6 172.16.70.6 1485 0x80000B97 0xEB84 6
    Displaying Net Link States (Area 0.0.0.0)
  Link ID      ADV Router      Age      Seq#      Checksum
172.16.1.3 192.168.239.66 1245    0x800000EC 0x82E
    Displaying Summary Net Link States (Area 0.0.0.0)
  Link ID      ADV Router      Age      Seq#      Checksum
172.16.240.0 172.16.241.5 1152    0x80000077 0x7A05
172.16.241.0 172.16.241.5 1152    0x80000070 0xAEB7
172.16.244.0 172.16.241.5 1152    0x80000071 0x95CB

```

The table below describes the significant fields shown in the display.

Table 41: show ip ospf Database Field Descriptions

| Field | Description |
|------------|--|
| Link ID | Router ID number. |
| ADV Router | Advertising router's ID. |
| Age | Link state age. |
| Seq# | Link state sequence number (detects old or duplicate link state advertisements). |
| Checksum | Fletcher checksum of the complete contents of the link state advertisement. |
| Link count | Number of interfaces detected for router. |

The following is sample output from the **show ip ospf database asbr-summary** command with the **asbr-summary** keyword:

```

Device#show ip ospf database asbr-summary
OSPF Router with id(192.168.239.66) (Process ID 300)
    Displaying Summary ASB Link States (Area 0.0.0.0)
LS age: 1463
Options: (No TOS-capability)
LS Type: Summary Links (AS Boundary Router)
Link State ID: 172.16.245.1 (AS Boundary Router address)
Advertising Router: 172.16.241.5
LS Seq Number: 80000072
Checksum: 0x3548
Length: 28
Network Mask: 0.0.0.0 TOS: 0 Metric: 1

```

The table below describes the significant fields shown in the display.

Table 42: show ip ospf database asbr-summary Field Descriptions

| Field | Description |
|---------------------|-------------------|
| OSPF Router with id | Router ID number. |
| Process ID | OSPF process ID. |

| Field | Description |
|--------------------|---|
| LS age | Link state age. |
| Options | Type of service options (Type 0 only). |
| LS Type | Link state type. |
| Link State ID | Link state ID (autonomous system boundary router). |
| Advertising Router | Advertising router's ID. |
| LS Seq Number | Link state sequence (detects old or duplicate link state advertisements). |
| Checksum | LS checksum (Fletcher checksum of the complete contents of the link state advertisement). |
| Length | Length in bytes of the link state advertisement. |
| Network Mask | Network mask implemented. |
| TOS | Type of service. |
| Metric | Link state metric. |

The following is sample output from the **show ip ospf database external** command with the **external** keyword:

```
Device#show ip ospf database external
OSPF Router with id(192.168.239.66) (Autonomous system 300)
    Displaying AS External Link States
LS age: 280
Options: (No TOS-capability)
LS Type: AS External Link
Link State ID: 10.105.0.0 (External Network Number)
Advertising Router: 172.16.70.6
LS Seq Number: 80000AFD
Checksum: 0xC3A
Length: 36
Network Mask: 255.255.0.0
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 1
    Forward Address: 0.0.0.0
    External Route Tag: 0
```

The table below describes the significant fields shown in the display.

Table 43: show ip ospf database external Field Descriptions

| Field | Description |
|---------------------|--|
| OSPF Router with id | Router ID number. |
| Autonomous system | OSPF autonomous system number (OSPF process ID). |
| LS age | Link state age. |
| Options | Type of service options (Type 0 only). |

| Field | Description |
|--------------------|---|
| LS Type | Link state type. |
| Link State ID | Link state ID (external network number). |
| Advertising Router | Advertising router's ID. |
| LS Seq Number | Link state sequence number (detects old or duplicate link state advertisements). |
| Checksum | LS checksum (Fletcher checksum of the complete contents of the LSA). |
| Length | Length in bytes of the link state advertisement. |
| Network Mask | Network mask implemented. |
| Metric Type | External Type. |
| TOS | Type of service. |
| Metric | Link state metric. |
| Forward Address | Forwarding address. Data traffic for the advertised destination will be forwarded to this address. If the forwarding address is set to 0.0.0.0, data traffic will be forwarded instead to the advertisement's originator. |
| External Route Tag | External route tag, a 32-bit field attached to each external route. This is not used by the OSPF protocol itself. |

The following is sample output from the **show ip ospf database network** command with the **network** keyword:

```
Device#show ip ospf database network
  OSPF Router with id(192.168.239.66) (Process ID 300)
    Displaying Net Link States(Area 0.0.0.0)

LS age: 1367
Options: (No TOS-capability)
LS Type: Network Links
Link State ID: 172.16.1.3 (address of Designated Router)
Advertising Router: 192.168.239.66
LS Seq Number: 800000E7
Checksum: 0x1229
Length: 52
Network Mask: 255.255.255.0
    Attached Router: 192.168.239.66
    Attached Router: 172.16.241.5
    Attached Router: 172.16.1.1
    Attached Router: 172.16.54.5
    Attached Router: 172.16.1.5
```

The table below describes the significant fields shown in the display.

Table 44: show ip ospf database network Field Descriptions

| Field | Description |
|---------------------|-------------------|
| OSPF Router with id | Router ID number. |
| Process ID 300 | OSPF process ID. |

| Field | Description |
|--------------------|---|
| LS age | Link state age. |
| Options | Type of service options (Type 0 only). |
| LS Type: | Link state type. |
| Link State ID | Link state ID of designated router. |
| Advertising Router | Advertising router's ID. |
| LS Seq Number | Link state sequence (detects old or duplicate link state advertisements). |
| Checksum | LS checksum (Fletcher checksum of the complete contents of the link state advertisement). |
| Length | Length in bytes of the link state advertisement. |
| Network Mask | Network mask implemented. |
| AS Boundary Router | Definition of router type. |
| Attached Router | List of routers attached to the network, by IP address. |

The following is sample output from the **show ip ospf database** command with the **router** keyword:

```
Device#show ip ospf database router
OSPF Router with id(192.168.239.66) (Process ID 300)
Displaying Router Link States(Area 0.0.0.0)
LS age: 1176
Options: (No TOS-capability)
LS Type: Router Links
Link State ID: 172.16.21.6
Advertising Router: 172.16.21.6
LS Seq Number: 80002CF6
Checksum: 0x73B7
Length: 120
AS Boundary Router
155 Number of Links: 8
Link connected to: another Router (point-to-point)
(link ID) Neighboring Router ID: 172.16.21.5
(Link Data) Router Interface address: 172.16.21.6
Number of TOS metrics: 0
TOS 0 Metrics: 2
```

The table below describes the significant fields shown in the display.

Table 45: show ip ospf database router Field Descriptions

| Field | Description |
|---------------------|-------------------|
| OSPF Router with id | Router ID number. |
| Process ID | OSPF process ID. |
| LS age | Link state age. |

| Field | Description |
|--------------------|---|
| Options | Type of service options (Type 0 only). |
| LS Type | Link state type. |
| Link State ID | Link state ID. |
| Advertising Router | Advertising router's ID. |
| LS Seq Number | Link state sequence (detects old or duplicate link state advertisements). |
| Checksum | LS checksum (Fletcher checksum of the complete contents of the link state advertisement). |
| Length | Length in bytes of the link state advertisement. |
| AS Boundary Router | Definition of router type. |
| Number of Links | Number of active links. |
| link ID | Link type. |
| Link Data | Router interface address. |
| TOS | Type of service metric (Type 0 only). |

The following is sample output from **show ip ospf database summary** command with the **summary** keyword:

```
Device#show ip ospf database summary
      OSPF Router with id(192.168.239.66) (Process ID 300)
      Displaying Summary Net Link States(Area 0.0.0.0)

LS age: 1401
Options: (No TOS-capability)
LS Type: Summary Links(Network)
Link State ID: 172.16.240.0 (summary Network Number)
Advertising Router: 172.16.241.5
LS Seq Number: 80000072
Checksum: 0x84FF
Length: 28
Network Mask: 255.255.255.0  TOS: 0  Metric: 1
```

The table below describes the significant fields shown in the display.

Table 46: show ip ospf database summary Field Descriptions

| Field | Description |
|---------------------|--|
| OSPF Router with id | Router ID number. |
| Process ID | OSPF process ID. |
| LS age | Link state age. |
| Options | Type of service options (Type 0 only). |
| LS Type | Link state type. |

| Field | Description |
|--------------------|---|
| Link State ID | Link state ID (summary network number). |
| Advertising Router | Advertising router's ID. |
| LS Seq Number | Link state sequence (detects old or duplicate link state advertisements). |
| Checksum | LS checksum (Fletcher checksum of the complete contents of the link state advertisement). |
| Length | Length in bytes of the link state advertisement. |
| Network Mask | Network mask implemented. |
| TOS | Type of service. |
| Metric | Link state metric. |

The following is sample output from **show ip ospf database database-summary** command with the **database-summary** keyword:

```

Device#show ip ospf database database-summary
OSPF Router with ID (10.0.0.1) (Process ID 1)
Area 0 database summary
  LSA Type      Count   Delete   Maxage
  Router        3       0       0
  Network       0       0       0
  Summary Net   0       0       0
  Summary ASBR  0       0       0
  Type-7 Ext    0       0       0
  Self-originated Type-7  0
Opaque Link    0       0       0
Opaque Area    0       0       0
Subtotal      3       0       0
Process 1 database summary
  LSA Type      Count   Delete   Maxage
  Router        3       0       0
  Network       0       0       0
  Summary Net   0       0       0
  Summary ASBR  0       0       0
  Type-7 Ext    0       0       0
  Opaque Link   0       0       0
  Opaque Area   0       0       0
  Type-5 Ext    0       0       0
  Self-originated Type-5  200
Opaque AS      0       0       0
Total         203    0       0

```

The table below describes the significant fields shown in the display.

Table 47: show ip ospf database database-summary Field Descriptions

| Field | Description |
|-------------------------|---|
| Area 0 database summary | Area number. |
| Count | Count of LSAs of the type identified in the first column. |

| Field | Description |
|----------------------------|---|
| Router | Number of router link state advertisements in that area. |
| Network | Number of network link state advertisements in that area. |
| Summary Net | Number of summary link state advertisements in that area. |
| Summary ASBR | Number of summary autonomous system boundary router (ASBR) link state advertisements in that area. |
| Type-7 Ext | Type-7 LSA count. |
| Self-originated Type-7 | Self-originated Type-7 LSA. |
| Opaque Link | Type-9 LSA count. |
| Opaque Area | Type-10 LSA count |
| Subtotal | Sum of LSAs for that area. |
| Delete | Number of link state advertisements that are marked "Deleted" in that area. |
| Maxage | Number of link state advertisements that are marked "Maxaged" in that area. |
| Process 1 database summary | Database summary for the process. |
| Count | Count of LSAs of the type identified in the first column. |
| Router | Number of router link state advertisements in that process. |
| Network | Number of network link state advertisements in that process. |
| Summary Net | Number of summary link state advertisements in that process. |
| Summary ASBR | Number of summary autonomous system boundary router (ASBR) link state advertisements in that process. |
| Type-7 Ext | Type-7 LSA count. |
| Opaque Link | Type-9 LSA count. |
| Opaque Area | Type-10 LSA count. |
| Type-5 Ext | Type-5 LSA count. |
| Self-Originated Type-5 | Self-originated Type-5 LSA count. |
| Opaque AS | Type-11 LSA count. |
| Total | Sum of LSAs for that process. |
| Delete | Number of link state advertisements that are marked "Deleted" in that process. |
| Maxage | Number of link state advertisements that are marked "Maxaged" in that process. |

show ip ospf interface

To display interface information related to Open Shortest Path First (OSPF), use the **show ip ospf interface** command in user EXEC or privileged EXEC mode.

show ip [ospf] [process-id] interface [type number] [brief] [multicast] [topology {topology-name | base}]

| Syntax Description | | |
|-------------------------------|--|--|
| <i>process-id</i> | (Optional) Process ID number. If this argument is included, only information for the specified routing process is included. The range is 1 to 65535. | |
| <i>type</i> | (Optional) Interface type. If the <i>type</i> argument is included, only information for the specified interface type is included. | |
| <i>number</i> | (Optional) Interface number. If the <i>number</i> argument is included, only information for the specified interface number is included. | |
| brief | (Optional) Displays brief overview information for OSPF interfaces, states, addresses and masks, and areas on the device. | |
| multicast | (Optional) Displays multicast information. | |
| topology topology-name | (Optional) Displays OSPF-related information about the named topology instance. | |
| topology base | (Optional) Displays OSPF-related information about the base topology. | |

Command Modes

User EXEC (>)

Privileged EXEC (#)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Examples

The following is sample output from the **show ip ospf interface** command when Ethernet interface 0/0 is specified:

```
Device#show ip ospf interface ethernet 0/0

Ethernet0/0 is up, line protocol is up
  Internet Address 192.168.254.202/24, Area 0
  Process ID 1, Router ID 192.168.99.1, Network Type BROADCAST, Cost: 10
  Topology-MTID      Cost      Disabled      Shutdown      Topology Name
    0                10         no           no           Base
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.99.1, Interface address 192.168.254.202
  Backup Designated router (ID) 192.168.254.10, Interface address 192.168.254.10
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:05
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
```

```

IETF NSF helper support enabled
Index 1/1, flood queue length 0
Next 0x0(0)/0x0(0)
Last flood scan length is 1, maximum is 1
Last flood scan time is 0 msec, maximum is 0 msec
Neighbor Count is 1, Adjacent neighbor count is 1
  Adjacent with neighbor 192.168.254.10 (Backup Designated Router)
Suppress hello for 0 neighbor(s)

```

In Cisco IOS Release 12.2(33)SRB, the following sample output from the **show ip ospf interface brief topology VOICE** command shows a summary of information, including a confirmation that the Multitopology Routing (MTR) VOICE topology is configured in the interface configuration:

```

Device#show ip ospf interface brief topology VOICE

VOICE Topology (MTID 10)
Interface  PID  Area          IP Address/Mask  Cost  State Nbrs F/C
Lo0        1    0            10.0.0.2/32      1    LOOP  0/0
Se2/0     1    0            10.1.0.2/30      10   P2P   1/1

```

The following sample output from the **show ip ospf interface brief topology VOICE** command displays details of the MTR VOICE topology for the interface. When the command is entered without the **brief** keyword, more information is displayed.

```

Device#show ip ospf interface topology VOICE

                VOICE Topology (MTID 10)
Loopback0 is up, line protocol is up
  Internet Address 10.0.0.2/32, Area 0
  Process ID 1, Router ID 10.0.0.2, Network Type LOOPBACK
  Topology-MTID   Cost   Disabled   Shutdown   Topology Name
    10            1     no        no        VOICE
Loopback interface is treated as a stub Host Serial2/0 is up, line protocol is up
  Internet Address 10.1.0.2/30, Area 0
  Process ID 1, Router ID 10.0.0.2, Network Type POINT_TO_POINT
  Topology-MTID   Cost   Disabled   Shutdown   Topology Name
    10            10     no        no        VOICE
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:03
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 10.0.0.1
  Suppress hello for 0 neighbor(s)

```

In Cisco IOS Release 12.2(33)SRC, the following sample output from the **show ip ospf interface** command displays details about the configured Time-to-Live (TTL) limits:

```

Device#show ip ospf interface ethernet 0
.
.
.
Strict TTL checking enabled
! or a message similar to the following is displayed
Strict TTL checking enabled, up to 4 hops allowed

```

.
.
.

The table below describes the significant fields shown in the displays.

Table 48: show ip ospf interface Field Descriptions

| Field | Description |
|---|--|
| Ethernet | Status of the physical link and operational status of the protocol. |
| Process ID | OSPF process ID. |
| Area | OSPF area. |
| Cost | Administrative cost assigned to the interface. |
| State | Operational state of the interface. |
| Nbrs F/C | OSPF neighbor count. |
| Internet Address | Interface IP address, subnet mask, and area address. |
| Topology-MTID | MTR topology Multitopology Identifier (MTID). A number assigned so that the protocol can identify the topology associated with information that it sends to its peers. |
| Transmit Delay | Transmit delay in seconds, interface state, and device priority. |
| Designated Router | Designated router ID and respective interface IP address. |
| Backup Designated router | Backup designated router ID and respective interface IP address. |
| Timer intervals configured | Configuration of timer intervals. |
| Hello | Number of seconds until the next hello packet is sent out this interface. |
| Strict TTL checking enabled | Only one hop is allowed. |
| Strict TTL checking enabled, up to 4 hops allowed | A set number of hops has been explicitly configured. |
| Neighbor Count | Count of network neighbors and list of adjacent neighbors. |

show ip ospf neighbor

To display Open Shortest Path First (OSPF) neighbor information on a per-interface basis, use the **show ip ospf neighbor** command in privileged EXEC mode.

show ip ospf neighbor [*interface-type interface-number*] [*neighbor-id*] [**detail**] [**summary**] [**per-instance**]

Syntax Description

| | |
|--|---|
| <i>interface-type interface-number</i> | (Optional) Type and number associated with a specific OSPF interface. |
| <i>neighbor-id</i> | (Optional) Neighbor hostname or IP address in A.B.C.D format. |
| detail | (Optional) Displays all neighbors given in detail (lists all neighbors). |
| summary | (Optional) Displays total number summary of all neighbors. |
| per-instance | (Optional) Displays total number of neighbors in each neighbor state. The output is printed for each configured OSPF instance separately. |

Command Modes

Privileged EXEC (#)

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Examples

The following sample output from the **show ip ospf neighbor** command shows a single line of summary information for each neighbor:

```
Device#show ip ospf neighbor
```

```
Neighbor ID    Pri   State           Dead Time   Address           Interface
10.199.199.137 1     FULL/DR         0:00:31    192.168.80.37    Ethernet0
172.16.48.1    1     FULL/DROTHER    0:00:33    172.16.48.1      Fddi0
172.16.48.200 1     FULL/DROTHER    0:00:33    172.16.48.200    Fddi0
10.199.199.137 5     FULL/DR         0:00:33    172.16.48.189    Fddi0
```

The following is sample output showing summary information about the neighbor that matches the neighbor ID:

```
Device#show ip ospf neighbor 10.199.199.137
```

```
Neighbor 10.199.199.137, interface address 192.168.80.37
  In the area 0.0.0.0 via interface Ethernet0
  Neighbor priority is 1, State is FULL
  Options 2
  Dead timer due in 0:00:32
  Link State retransmission due in 0:00:04
Neighbor 10.199.199.137, interface address 172.16.48.189
  In the area 0.0.0.0 via interface Fddi0
  Neighbor priority is 5, State is FULL
  Options 2
  Dead timer due in 0:00:32
```

```
Link State retransmission due in 0:00:03
```

If you specify the interface along with the neighbor ID, the system displays the neighbors that match the neighbor ID on the interface, as in the following sample display:

```
Device#show ip ospf neighbor ethernet 0 10.199.199.137

Neighbor 10.199.199.137, interface address 192.168.80.37
  In the area 0.0.0.0 via interface Ethernet0
  Neighbor priority is 1, State is FULL
  Options 2
  Dead timer due in 0:00:37
  Link State retransmission due in 0:00:04
```

You can also specify the interface without the neighbor ID to show all neighbors on the specified interface, as in the following sample display:

```
Device#show ip ospf neighbor fddi 0

   ID          Pri   State          Dead Time   Address      Interface
172.16.48.1    1   FULL/DROTHER  0:00:33    172.16.48.1  Fddi0
172.16.48.200  1   FULL/DROTHER  0:00:32    172.16.48.200 Fddi0
10.199.199.137 5   FULL/DR       0:00:32    172.16.48.189 Fddi0
```

The following is sample output from the **show ip ospf neighbor detail** command:

```
Device#show ip ospf neighbor detail

Neighbor 192.168.5.2, interface address 10.225.200.28
  In the area 0 via interface GigabitEthernet1/0/0
  Neighbor priority is 1, State is FULL, 6 state changes
  DR is 10.225.200.28 BDR is 10.225.200.30
  Options is 0x42
  LLS Options is 0x1 (LR), last OOB-Resync 00:03:08 ago
  Dead timer due in 00:00:36
  Neighbor is up for 00:09:46
  Index 1/1, retransmission queue length 0, number of retransmission 1
  First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
  Last retransmission scan length is 1, maximum is 1
  Last retransmission scan time is 0 msec, maximum is 0 msec
```

The table below describes the significant fields shown in the displays.

Table 49: show ip ospf neighbor detail Field Descriptions

| Field | Description |
|-------------------|---|
| Neighbor | Neighbor router ID. |
| interface address | IP address of the interface. |
| In the area | Area and interface through which the OSPF neighbor is known. |
| Neighbor priority | Router priority of the neighbor and neighbor state. |
| State | OSPF state. If one OSPF neighbor has enabled TTL security, the other side of the connection will show the neighbor in the INIT state. |

| Field | Description |
|---------------------------------|--|
| state changes | Number of state changes since the neighbor was created. This value can be reset using the clearipospfcountersneighbor command. |
| DR is | Router ID of the designated router for the interface. |
| BDR is | Router ID of the backup designated router for the interface. |
| Options | Hello packet options field contents. (E-bit only. Possible values are 0 and 2; 2 indicates area is not a stub; 0 indicates area is a stub.) |
| LLS Options..., last OOB-Resync | Link-Local Signaling and out-of-band (OOB) link-state database resynchronization performed hours:minutes:seconds ago. This is nonstop forwarding (NSF) information. The field indicates the last successful out-of-band resynchronization with the NSF-capable router. |
| Dead timer due in | Expected time in hours:minutes:seconds before Cisco IOS software will declare the neighbor dead. |
| Neighbor is up for | Number of hours:minutes:seconds since the neighbor went into the two-way state. |
| Index | Neighbor location in the area-wide and autonomous system-wide retransmission queue. |
| retransmission queue length | Number of elements in the retransmission queue. |
| number of retransmission | Number of times update packets have been re-sent during flooding. |
| First | Memory location of the flooding details. |
| Next | Memory location of the flooding details. |
| Last retransmission scan length | Number of link state advertisements (LSAs) in the last retransmission packet. |
| maximum | Maximum number of LSAs sent in any retransmission packet. |
| Last retransmission scan time | Time taken to build the last retransmission packet. |
| maximum | Maximum time, in milliseconds, taken to build any retransmission packet. |

The following is sample output from the **show ip ospf neighbor** command showing a single line of summary information for each neighbor. If one OSPF neighbor has enabled TTL security, the other side of the connection will show the neighbor in the INIT state.

```
Device#show ip ospf neighbor
```

```
Neighbor ID    Pri   State           Dead Time   Address           Interface
10.199.199.137 1     FULL/DR         0:00:31    192.168.80.37    Ethernet0
172.16.48.1    1     FULL/DROTHER    0:00:33    172.16.48.1      Fddi0
172.16.48.200 1     FULL/DROTHER    0:00:33    172.16.48.200    Fddi0
```

```
10.199.199.137 5 FULL/DR 0:00:33 172.16.48.189 Fddi0
172.16.1.201 1 INIT/DROTHER 00.00.35 10.1.1.201 Ethernet0/0
```

Cisco IOS Release 15.1(3)S

The following sample output from the **show ip ospf neighbor** command shows the network from the neighbor's point of view:

```
Device#show ip ospf neighbor 192.0.2.1
      OSPF Router with ID (192.1.1.1) (Process ID 1)

          Area with ID (0)

Neighbor with Router ID 192.0.2.1:
  Reachable over:
    Ethernet0/0, IP address 192.0.2.1, cost 10

  SPF was executed 1 times, distance to computing router 10

  Router distance table:
    192.1.1.1 i [10]
    192.0.2.1 i [0]
    192.3.3.3 i [10]
    192.4.4.4 i [20]
    192.5.5.5 i [20]

  Network LSA distance table:
    192.2.12.2 i [10]
    192.2.13.3 i [20]
    192.2.14.4 i [20]
    192.2.15.5 i [20]
```

The following is sample output from the **show ip ospf neighbor summary** command:

```
Device#show ip ospf neighbor summary

      Neighbor summary for all OSPF processes

DOWN          0
ATTEMPT      0
INIT         0
2WAY         0
EXSTART      0
EXCHANGE     0
LOADING      0
FULL         1
Total count  1      (Undergoing NSF 0)
```

The following is sample output from the **show ip ospf neighbor summary per-instance** command:

```
Device#show ip ospf neighbor summary

      OSPF Router with ID (1.0.0.10) (Process ID 1)

DOWN          0
ATTEMPT      0
INIT         0
2WAY         0
```

show ip ospf neighbor

```

EXSTART      0
EXCHANGE     0
LOADING      0
FULL         1
Total count  1      (Undergoing NSF 0)

```

Neighbor summary for all OSPF processes

```

DOWN         0
ATTEMPT     0
INIT        0
2WAY        0
EXSTART     0
EXCHANGE    0
LOADING     0
FULL        1
Total count  1      (Undergoing NSF 0)

```

Table 50: show ip ospf neighbor summary and show ip ospf neighbor summary per-instance Field Descriptions

| Field | Description |
|----------|--|
| DOWN | No information (hellos) has been received from this neighbor, but hello packets can still be sent to the neighbor in this state. |
| ATTEMPT | This state is only valid for manually configured neighbors in a Non-Broadcast Multi-Access (NBMA) environment. In Attempt state, the router sends unicast hello packets every poll interval to the neighbor, from which hellos have not been received within the dead interval. |
| INIT | This state specifies that the router has received a hello packet from its neighbor, but the receiving router's ID was not included in the hello packet. When a router receives a hello packet from a neighbor, it should list the sender's router ID in its hello packet as an acknowledgment that it received a valid hello packet. |
| 2WAY | This state designates that bi-directional communication has been established between two routers. |
| EXSTART | This state is the first step in creating an adjacency between the two neighboring routers. The goal of this step is to decide which router is active, and to decide upon the initial DD sequence number. Neighbor conversations in this state or greater are called adjacencies. |
| EXCHANGE | In this state, OSPF routers exchange database descriptor (DBD) packets. Database descriptors contain link-state advertisement (LSA) headers only and describe the contents of the entire link-state database. Each DBD packet has a sequence number which can be incremented only by the active router which is explicitly acknowledged by the secondary router. Routers also send link-state request packets and link-state update packets (which contain the entire LSA) in this state. The contents of the DBD received are compared to the information contained in the routers link-state database to check if new or more current link-state information is available with the neighbor. |

| Field | Description |
|---------|--|
| LOADING | In this state, the actual exchange of link state information occurs. Based on the information provided by the DBDs, routers send link-state request packets. The neighbor then provides the requested link-state information in link-state update packets. During the adjacency, if a device receives an outdated or missing LSA, it requests that LSA by sending a link-state request packet. All link-state update packets are acknowledged. |
| FULL | <p>In this state, devices are fully adjacent with each other. All the device and network LSAs are exchanged and the devices' databases are fully synchronized.</p> <p>Full is the normal state for an OSPF device. If a device is stuck in another state, it's an indication that there are problems in forming adjacencies. The only exception to this is the 2-way state, which is normal in a broadcast network. Devices achieve the full state with their DR and BDR only. Neighbors always see each other as 2-way.</p> |

show ip ospf virtual-links

To display parameters and the current state of Open Shortest Path First (OSPF) virtual links, use the **show ip ospf virtual-links** command in EXEC mode.

show ip ospf virtual-links

Syntax Description

This command has no arguments or keywords.

Command Modes

EXEC

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

The information displayed by the **show ip ospf virtual-links** command is useful in debugging OSPF routing operations.

Examples

The following is sample output from the **show ip ospf virtual-links** command:

```
Device#show ip ospf virtual-links
Virtual Link to router 192.168.101.2 is up
Transit area 0.0.0.1, via interface Ethernet0, Cost of using 10
Transmit Delay is 1 sec, State POINT_TO_POINT
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Hello due in 0:00:08
Adjacency State FULL
```

The table below describes the significant fields shown in the display.

Table 51: show ip ospf virtual-links Field Descriptions

| Field | Description |
|--|--|
| Virtual Link to router 192.168.101.2 is up | Specifies the OSPF neighbor, and if the link to that neighbor is up or down. |
| Transit area 0.0.0.1 | The transit area through which the virtual link is formed. |
| via interface Ethernet0 | The interface through which the virtual link is formed. |
| Cost of using 10 | The cost of reaching the OSPF neighbor through the virtual link. |
| Transmit Delay is 1 sec | The transmit delay (in seconds) on the virtual link. |
| State POINT_TO_POINT | The state of the OSPF neighbor. |
| Timer intervals... | The various timer intervals configured for the link. |
| Hello due in 0:00:08 | When the next hello is expected from the neighbor. |
| Adjacency State FULL | The adjacency state between the neighbors. |

summary-address (OSPF)

To create aggregate addresses for Open Shortest Path First (OSPF), use the **summary-address** command in router configuration mode. To restore the default, use the no form of this command.

summary-address **command** **summary-address** {*ip-address mask* | *prefix mask*} [**not-advertise**] [**tag tag**] [**nssa-only**]
no summary-address {*ip-address mask* | *prefix mask*} [**not-advertise**] [**tag tag**] [**nssa-only**]

Syntax Description

| | |
|----------------------|---|
| <i>ip-address</i> | Summary address designated for a range of addresses. |
| <i>mask</i> | IP subnet mask used for the summary route. |
| <i>prefix</i> | IP route prefix for the destination. |
| not-advertise | (Optional) Suppresses routes that match the specified prefix/mask pair. This keyword applies to OSPF only. |
| tag tag | (Optional) Specifies the tag value that can be used as a “match” value for controlling redistribution via route maps. This keyword applies to OSPF only. |
| nssa-only | (Optional) Sets the nssa-only attribute for the summary route (if any) generated for the specified prefix, which limits the summary to not-so-stubby-area (NSSA) areas. |

Command Default

This command behavior is disabled by default.

Command Modes

Router configuration

Command History

| Release | Modification |
|------------------------------|------------------------------|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines

R routes learned from other routing protocols can be summarized. The metric used to advertise the summary is the lowest metric of all the more specific routes. This command helps reduce the size of the routing table.

Using this command for OSPF causes an OSPF Autonomous System Boundary Router (ASBR) to advertise one external route as an aggregate for all redistributed routes that are covered by the address. For OSPF, this command summarizes only routes from other routing protocols that are being redistributed into OSPF. Use the **area range** command for route summarization between OSPF areas.

OSPF does not support the **summary-address 0.0.0.0 0.0.0.0** command.

Examples

In the following example, the summary address 10.1.0.0 includes address 10.1.1.0, 10.1.2.0, 10.1.3.0, and so on. Only the address 10.1.0.0 is advertised in an external link-state advertisement.

```
Device(config)#summary-address 10.1.0.0 255.255.0.0
```

Related Commands

| Command | Description |
|-----------------------------------|---|
| area range | Consolidates and summarizes routes at an area boundary. |
| ip ospf authentication-key | Assigns a password to be used by neighboring routers that are using the simple password authentication of OSPF. |
| ip ospf message-digest-key | Enables OSPF MD5 authentication. |

timers throttle spf

To turn on Open Shortest Path First (OSPF) shortest path first (SPF) throttling, use the **timers throttle spf** command in the appropriate configuration mode. To turn off OSPF SPF throttling, use the **no** form of this command.

timers throttle spf *spf-start spf-hold spf-max-wait*
no timers throttle spf *spf-start spf-hold spf-max-wait*

| Syntax Description | | |
|---------------------|--|--|
| <i>spf-start</i> | Initial delay to schedule an SPF calculation after a change, in milliseconds. Range is from 1 to 600000. In OSPF for IPv6, the default value is 5000. | |
| <i>spf-hold</i> | Minimum hold time between two consecutive SPF calculations, in milliseconds. Range is from 1 to 600000. In OSPF for IPv6, the default value is 10,000. | |
| <i>spf-max-wait</i> | Maximum wait time between two consecutive SPF calculations, in milliseconds. Range is from 1 to 600000. In OSPF for IPv6, the default value is 10,000. | |

Command Default SPF throttling is not set.

Command Modes Address family configuration (config-router-af) Router address family topology configuration (config-router-af-topology) Router configuration (config-router) OSPF for IPv6 router configuration (config-rtr)

| Command History | Release | Modification |
|-----------------|------------------------------|------------------------------|
| | Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Usage Guidelines The first wait interval between SPF calculations is the amount of time in milliseconds specified by the *spf-start* argument. Each consecutive wait interval is two times the current hold level in milliseconds until the wait time reaches the maximum time in milliseconds as specified by the *spf-max-wait* argument. Subsequent wait times remain at the maximum until the values are reset or a link-state advertisement (LSA) is received between SPF calculations.

Release 12.2(33)SRB

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

Release 15.2(1)T

When you configure the **ospfv3 network manet** command on any interface attached to the OSPFv3 process, the default values for the *spf-start*, *spf-hold*, and the *spf-max-wait* arguments are reduced to 1000 milliseconds, 1000 milliseconds, and 2000 milliseconds respectively.

Examples

The following example shows how to configure a router with the delay, hold, and maximum interval values for the **timers throttle spf** command set at 5, 1000, and 90,000 milliseconds, respectively.

```
router ospf 1
router-id 10.10.10.2
```

```
log-adjacency-changes
timers throttle spf 5 1000 90000
redistribute static subnets
network 10.21.21.0 0.0.0.255 area 0
network 10.22.22.0 0.0.0.255 area 00
```

The following example shows how to configure a router using IPv6 with the delay, hold, and maximum interval values for the **timers throttle spf** command set at 500, 1000, and 10,000 milliseconds, respectively.

```
ipv6 router ospf 1
event-log size 10000 one-shot
log-adjacency-changes
timers throttle spf 500 1000 10000
```

Related Commands

| Command | Description |
|-----------------------------|---|
| ospfv3 network manet | Sets the network type to Mobile Ad Hoc Network (MANET). |