# Configuring EIGRP

Enhanced Interior Gateway Routing Protocol (EIGRP) is an enhanced version of the IGRP developed by Cisco. The convergence properties and the operating efficiency of EIGRP have improved substantially over IGRP, IGRP is now obsolete.

The convergence technology is based on research conducted at SRI International and employs an algorithm referred to as the Diffusing Update Algorithm (DUAL). This algorithm guarantees loop-free operation at every instant throughout a route computation and allows all devices involved in a topology change to synchronize at the same time. Routers that are not affected by topology changes are not involved in recomputations.

# Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for EIGRP" section on page 63.

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

# Contents

# Information About EIGRP

## EIGRP Features

- Increased network width—With IP Routing Information Protocol (RIP), the largest possible width of your network is 15 hops. When EIGRP is enabled, the largest possible width is increased to 255 hops, and the EIGRP metric is large enough to support thousands of hops. The default maximum number of EIGRP hops is 100.

- Fast convergence—The DUAL algorithm allows routing information to converge quickly.

- Partial updates—EIGRP sends incremental updates when the state of a destination changes, instead of sending the entire contents of the routing table. This feature minimizes the bandwidth required for EIGRP packets.

- Neighbor discovery mechanism—This is a simple hello mechanism used to learn about neighboring routers. It is protocol-independent.

- Variable-length subnet masks (VLSMs).

- Arbitrary route summarization.

- Scaling—EIGRP scales to large networks.

# EIGRP Autonomous System Configuration

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

In EIGRP autonomous system configuration, EIGRP Virtual Private Networks (VPNs) can be configured only under IPv4 address family configuration mode. A virtual routing and forwarding (VRF) instance and route distinguisher must be defined before the address family session can be created.

It is recommended that you configure an autonomous system number when the address family is configured, either by entering the *autonomous-system-number* argument with the **address-family** command or separately using the **autonomous-system** command.

# EIGRP Named Configuration

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

In EIGRP named configuration, EIGRP VPNs can be configured in IPv4 and IPv6 named configurations. A virtual routing and forwarding (VRF) instance and a route distinguisher may or may not be used to create the address family.

If a VRF is not used in creating the address family, the EIGRP VPN instance assumes the role of default route distinguisher and will communicate with the default route distinguisher of other routers in the same network.

EIGRP VPNs can be configured under EIGRP named configurations. A VRF and route distinguisher must be defined before the address family session can be created.

A single EIGRP routing process can support multiple VRFs. The number of VRFs that can be configured is limited only by available system resources on the router, which is determined by the number of VRFs, running processes, and available memory. However, only a single VRF can be supported by each VPN, and redistribution between different VRFs is not supported.

# EIGRP Neighbor Relationship Maintenance

Neighbor relationship maintenance is the process that routers use to dynamically learn of other routers on their directly attached networks. Routers must also discover when their neighbors become unreachable or inoperative. Neighbor relationship maintenance is achieved with low overhead by routers periodically sending small hello packets. As long as hello packets are received, the Cisco IOS software can determine that a neighbor is alive and functioning. When this status is determined, the neighboring routers can exchange routing information.

The reliable transport protocol is responsible for guaranteed, ordered delivery of EIGRP packets to all neighbors. It supports intermixed transmission of multicast and unicast packets. Some EIGRP packets must be sent reliably and others need not be. For efficiency, reliability is provided only when necessary. For example, on a multiaccess network that has multicast capabilities (such as Ethernet) it is not necessary to send hello packets reliably to all neighbors individually. Therefore, EIGRP sends a single multicast hello with an indication in the packet informing the receivers that the packet need not be acknowledged. Other types of packets (such as updates) require acknowledgment, which is indicated in

the packet. The reliable transport has a provision to send multicast packets quickly when unacknowledged packets are pending. This provision helps to ensure that convergence time remains low in the presence of varying speed links.

# DUAL Finite State Machine

The DUAL finite state machine embodies the decision process for all route computations. It tracks all routes advertised by all neighbors. DUAL uses the distance information (known as a metric) to select efficient, loop-free paths. DUAL selects routes to be inserted into a routing table based on feasible successors. A successor is a neighboring router used for packet forwarding that has a least-cost path to a destination that is guaranteed not to be part of a routing loop. When there are no feasible successors but there are neighbors advertising the destination, a recomputation must occur. This is the process whereby a new successor is determined. The amount of time required to recompute the route affects the convergence time. Recomputation is processor-intensive; it is advantageous to avoid unneeded recomputation. When a topology change occurs, DUAL will test for feasible successors. If there are feasible successors, DUAL will use any feasible successors it finds in order to avoid unnecessary recomputation.

# Protocol-Dependent Modules

The protocol-dependent modules are responsible for network-layer protocol-specific tasks. An example is the EIGRP module, which is responsible for sending and receiving EIGRP packets that are encapsulated in IP. It is also responsible for parsing EIGRP packets and informing DUAL of the new information received. EIGRP asks DUAL to make routing decisions, but the results are stored in the IP routing table. Also, EIGRP is responsible for redistributing routes learned by other IP routing protocols.

# EIGRP Metric Weights

EIGRP uses the minimum bandwidth on the path to a destination network and the total delay to compute routing metrics. You can use the **metric weights** (EIGRP) command to adjust the default behavior of EIGRP routing and metric computations. For example, this adjustment allows you to tune system behavior to allow for satellite transmission. EIGRP metric defaults have been carefully selected to provide optimal performance in most networks.

> **Note** Adjusting EIGRP metric weights can dramatically affect network performance. Because of the complexity of this task, we recommend that you do not change the default values without guidance from an experienced network designer.

By default, the EIGRP composite metric is a 32-bit quantity that is a sum of the segment delays and the lowest segment bandwidth (scaled and inverted) for a given route. The formula used to scale and invert the bandwidth value is $10^7$/minimum Bw in kilobits per second.

For a network of homogeneous media, this metric reduces to a hop count. For a network of mixed media (FDDI, Ethernet, and serial lines running from 9600 bits per second to T1 rates), the route with the lowest metric reflects the most desirable path to a destination.

## Mismatched K Values

EIGRP K values are the metrics that EIGRP uses to calculate routes. Mismatched K values (EIGRP metrics) can prevent neighbor relationships from being established and can negatively impact network convergence. The following example explains this behavior between two EIGRP peers (ROUTER-A and ROUTER-B).

The following configuration is applied to ROUTER-A. The K values are changed with the **metric weights** command. A value of 2 is entered for the *k1* argument to adjust the bandwidth calculation. The value of 1 is entered for the *k3* argument to adjust the delay calculation.

```
Router(config)# hostname Router-A
Router-A(config)# interface serial 0
Router-A(config-if)# ip address 10.1.1.1 255.255.255.0
Router-A(config-if)# exit
Router-A(config)# router eigrp virtual-name1
Router-A(config-router)# address-family ipv4 autonomous-system 4533
Router-A(config-router-af)# network 10.1.1.0 0.0.0.255
Router-A(config-router-af)# metric weights 0 2 0 1 0 0
```

The following configuration is applied to ROUTER-B. However, the **metric weights** command is not applied and the default K values are used. The default K values are 1, 0, 1, 0, and 0.

```
Router(config)# hostname Router-B
Router-B(config)# interface serial 0
Router-B(config-if)# ip address 10.1.1.2 255.255.255.0
Router-B(config-if)# exit
Router-B(config)# router eigrp virtual-name1
Router-B(config-router)# address-family ipv4 autonomous-system 4533
Router-B(config-router-af)# network 10.1.1.0 0.0.0.255
```

The bandwidth calculation is set to 2 on ROUTER-A and set to 1 (by default) on ROUTER-B. This configuration prevents these peers from forming a neighbor relationship.

The following error message is displayed in the console of ROUTER-B because the K values are mismatched:

```
*Apr 26 13:48:41.811: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 10.1.1.1 (Ethernet0/0) is
down: K-value mismatch
```

There are two scenarios where this error message can be displayed:

- The two routers are connected on the same link and configured to establish a neighbor relationship. However, each router is configured with different K values.

- The K-value mismatch error message can also be displayed if one of the two peers has transmitted a "goodbye" message, and the receiving router does not support this message. In this case, the receiving router will interpret this message as a K-value mismatch.

# Goodbye Message

The goodbye message is a feature designed to improve EIGRP network convergence. The goodbye message is broadcast when an EIGRP routing process is shut down to inform adjacent peers about the impending topology change. This feature allows supporting EIGRP peers to synchronize and recalculate neighbor relationships more efficiently than would occur if the peers discovered the topology change after the hold timer expired.

The following message is displayed by routers that run a supported release when a goodbye message is received:

```
 *Apr 26 13:48:42.523: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 10.1.1.1
(Ethernet0/0) is down: Interface Goodbye received
```

A Cisco router that runs a software release that does not support the goodbye message can misinterpret the message as a K-value mismatch and display the following message:

```
 *Apr 26 13:48:41.811: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor     10.1.1.1
(Ethernet0/0) is down: K-value mismatch
```

✎

**Note**    The receipt of a goodbye message by a nonsupporting peer does not disrupt normal network operation. The nonsupporting peer will terminate the session when the hold timer expires. The sending and receiving routers will reconverge normally after the sender reloads.

# Routing Metric Offset Lists

An offset list is the mechanism for increasing incoming and outgoing metrics to routes learned via EIGRP. An offset list provides a local mechanism for increasing the value of routing metrics. Optionally, you can limit the offset list with either an access list or an interface.

✎

**Note**    Offset lists are available only in IPv4 configurations. IPv6 configurations do not support offset lists.

# EIGRP Cost Metrics

EIGRP receives dynamic raw radio link characteristics and computes a composite EIGRP cost metric based on a proprietary formula. To avoid churn in the network as a result of the change in the link characteristics, a tunable dampening mechanism is used.

EIGRP uses the metric weights along with a set of vector metrics to compute the composite metric for local RIB installation and route selections. The EIGRP composite metric is calculated using the formula:

EIGRP Metric = 256*((K1*Bw) + (K2*Bw)/(256-Load) + (K3*Delay)*(K5/(Reliability + K4)))

Table 1 lists the EIGRP vector metrics and their descriptions.

*Table 1*        ***EIGRP Vector Metrics***

| Vector Metric | Description |
|---|---|
| bandwidth | Minimum bandwidth of the route in kilobits per second. It can be 0 or any positive integer. The bandwidth for the formula is scaled and inverted by the following formula:<br><br>(10^7/minimum Bw in kilobits per second) |
| delay | Route delay in tens of microseconds. |
| delay reliability | Likelihood of successful packet transmission expressed as a number between 0 and 255. The value 255 means 100 percent reliability; 0 means no reliability. |

*Table 1          EIGRP Vector Metrics (continued)*

| Vector Metric | Description |
|---|---|
| load | Effective load of the route expressed as a number from 0 to 255 (255 is 100 percent loading). |
| mtu | Minimum maximum transmission unit (MTU) size of the route in bytes. It can be 0 or any positive integer. |

EIGRP monitors metric weights on an interface to allow for the tuning of EIGRP metric calculations and indicate type of service (ToS). Table 2 lists the K values and their default.

*Table 2          EIGRP K-Value Defaults*

| Setting | Default Value |
|---|---|
| K1 | 1 |
| K2 | 0 |
| K3 | 1 |
| K4 | 0 |
| K5 | 0 |

Most configurations use the delay and bandwidth metrics, with bandwidth taking precedence. The default formula of 256*(Bw + Delay) is the EIGRP metric. The bandwidth for the formula is scaled and inverted by the following formula:

($10^7$/minimum Bw in kilobits per second)

**Note**     You can change the weights, but these weights must be the same on all the routers.

For example, look at a link whose bandwidth to a particular destination is 128k and the delay is 84,000 microseconds.

Using the cut-down formula, the EIGRP metric calculation would simplify to 256*(Bw + Delay), resulting in the following value:

Metric = 256*($10^7$/128 + 84000/10)= 256*86525 = 22150400

To calculate route delay, divide the delay value by 10 to get the true value in tenths of microseconds.

When EIGRP calculates the delay for Mobile Ad Hoc Networks (MANET) and the delay is obtained from a router interface, the delay is always calculated in tens of microseconds. In most cases, when using MANET, you will not use the interface delay, but rather the delay that is advertised by the radio. The delay you will receive from the radio is in microseconds, so you must adjust the cut-down formula as follows:

Metric = (256*($10^7$/128) + (84000*256)/10) = 20000000 + 2150400 = 22150400

# Route Summarization

You can configure EIGRP to perform automatic summarization of subnet routes into network-level routes. For example, you can configure subnet 172.16.1.0 to be advertised as 172.16.0.0 over interfaces that have subnets of 192.168.7.0 configured. Automatic summarization is performed when two or more **network** (EIGRP) router configuration or address family configuration commands are configured for the EIGRP process. By default, this feature is enabled.

Route summarization works in conjunction with the **ip summary-address eigrp** command available in interface configuration mode for autonomous system configurations and with the **summary-address** (EIGRP) command for named configurations in which additional summarization can be performed. If automatic summarization is in effect, there usually is no need to configure network level summaries using the **ip summary-address eigrp** command.

# Summary Aggregate Addresses

You can configure a summary aggregate address for a specified interface. If any more specific routes are in the routing table, EIGRP will advertise the summary address out the interface with a metric equal to the minimum of all more specific routes.
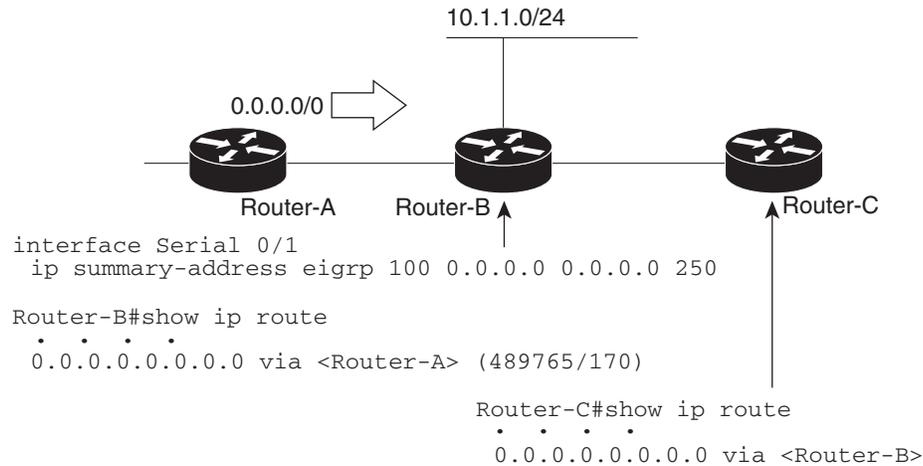
# Floating Summary Routes

You can use a floating summary route when configuring the **ip summary-address eigrp** command for autonomous system configurations or the **summary-address** (EIGRP) command for named configurations. The floating summary route is created by applying a default route and administrative distance at the interface level, or address family interface level. The following scenarios illustrate the behavior of floating summary routes.

Figure 1 shows a network with three routers, Router-A, Router-B, and Router-C. Router-A learns a default route from elsewhere in the network and then advertises this route to Router-B. Router-B is configured so that only a default summary route is advertised to Router-C. The default summary route is applied to serial interface 0/1 on Router-B with the following configuration for an autonomous system configuration:

```
Router(config)# interface Serial 0/1
Router(config-if)# ip summary-address eigrp 100 0.0.0.0 0.0.0.0
```

The default summary route is applied to serial interface 0/1 on Router-B with the following configuration for a named configuration:

```
Router(config-router-af)# af-interface serial0/1
Router(config-router-af-interface)# summary-address 192.168.0.0 255.255.0.0 95
```

*Figure 1*        *Floating Summary Route Applied to Router-B*

```
                              10.1.1.0/24


          0.0.0.0/0 ⬜⟹

        Router-A     Router-B ↑                ↑ Router-C
interface Serial 0/1
  ip summary-address eigrp 100 0.0.0.0 0.0.0.0 250

Router-B#show ip route
  .   .   .   .
  0.0.0.0.0.0.0.0 via <Router-A> (489765/170)

                              Router-C#show ip route
                                .   .   .   .
                                0.0.0.0.0.0.0.0 via <Router-B>
```

The configuration of the default summary route on Router-B sends a 0.0.0.0/0 summary route to Router-C and blocks all other routes, including the 10.1.1.0/24 route, from being advertised to Router-C. However, this also generates a local discard route on Router-B, a route for 0.0.0.0/0 to the null 0 interface with an administrative distance of 5. When this route is created, it overrides the EIGRP learned default route. Router-B will no longer be able to reach destinations that it would normally reach through the 0.0.0.0.0/0 route.

This problem is resolved by applying a floating summary route to the interface on Router-B that connects to Router-C. The floating summary route is applied by configuring an administrative distance for the default summary route on the interface of Router-B with the following statement for an autonomous system configuration:

```
Router(config-if)# ip summary-address eigrp 100 0.0.0.0 0.0.0.0 250
```

The floating summary route is applied by configuring an administrative distance for the default summary route on the interface of Router-B with the following statement for a named configuration:
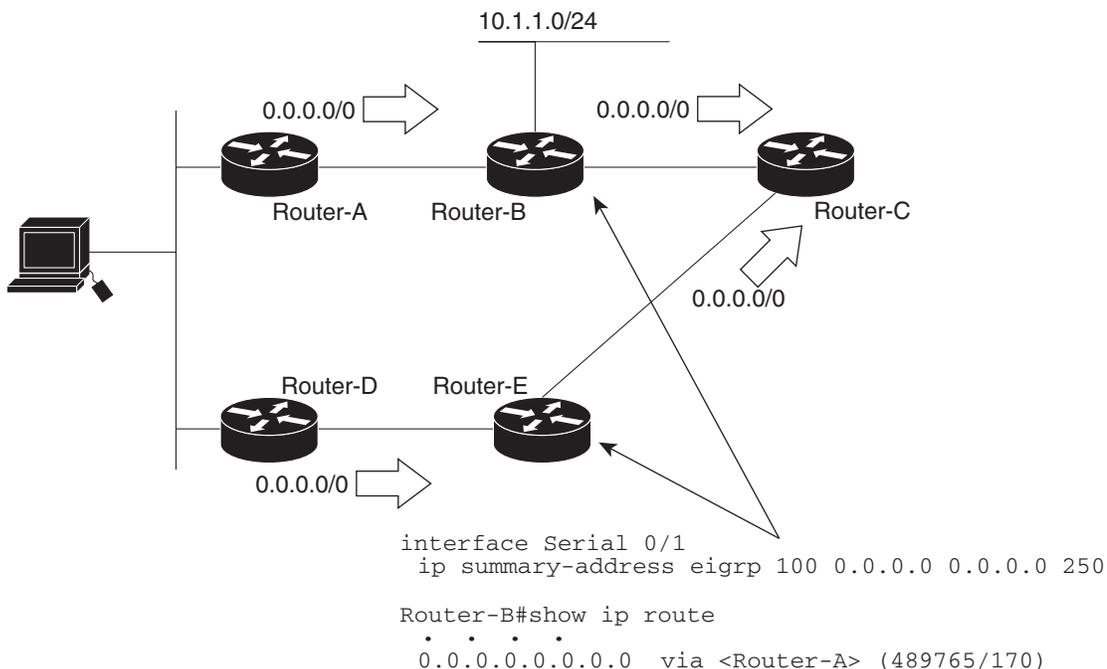
```
Router(config-router-af-interface)# summary-address eigrp 100 0.0.0.0 0.0.0.0 250
```

The administrative distance of 250, applied in the above statement, is now assigned to the discard route generated on Router-B. The 0.0.0.0/0, from Router-A, is learned through EIGRP and installed in the local routing table. Routing to Router-C is restored.

If Router-A loses the connection to Router-B, Router-B will continue to advertise a default route to Router-C, which allows traffic to continue to reach destinations attached to Router-B. However, traffic destined to networks to Router-A or behind Router-A will be dropped when it reaches Router-B.

Figure 2 shows a network with two connections from the core, Router-A and Router-D. Both Router-B and Router-E have floating summary routes configured on the interfaces connected to Router-C. If the connection between Router-E and Router-C fails, the network will continue to operate normally. All traffic will flow from Router-C through Router-B to the hosts attached to Router-A and Router-D.

*Figure 2*          *Floating Summary Route Applied for Dual-Homed Remotes*



```
interface Serial 0/1
  ip summary-address eigrp 100 0.0.0.0 0.0.0.0 250

Router-B#show ip route
  .  .  .  .
  0.0.0.0.0.0.0.0  via <Router-A> (489765/170)
```

However, if the link between Router-A and Router-B fails, the network may incorrectly direct traffic because Router-B will continue to advertise the default route (0.0.0.0/0) to Router-C. In this scenario, Router-C still forwards traffic to Router-B, but Router-B drops the traffic. To avoid this problem, you should configure the summary address with an administrative distance on only single-homed remote routers or areas where there is only one exit point between two segments of the network. If two or more exit points exist (from one segment of the network to another), configuring the floating default route can cause a black hole to be formed.

# EIGRP Route Authentication

EIGRP route authentication provides message digest algorithm 5 (MD5) authentication of routing updates from the EIGRP routing protocol. The MD5 keyed digest in each EIGRP packet prevents the introduction of unauthorized or false routing messages from unapproved sources.

Each key has its own key identifier (specified with the **key** *number* key chain configuration command), which is stored locally. The combination of the key identifier and the interface associated with the message uniquely identifies the authentication algorithm and MD5 authentication key in use.

You can configure multiple keys with lifetimes. Only one authentication packet is sent, regardless of how many valid keys exist. The software examines the key numbers in order from lowest to highest, and uses the first valid key it encounters. Note that the router needs to know the time to configure keys with lifetimes. Refer to the Network Time Protocol (NTP) and calendar commands in the "Performing Basic System Management" module of the *Cisco IOS Network Management Configuration Guide*.

For autonomous system and named configuration examples of route authentication, see the and the .

## Hello Packets and the Hold-Time Intervals

You can adjust the interval between hello packets and the hold time. This is a protocol-independent parameter that works for IP and IPX.

Routing devices periodically send hello packets to each other to dynamically learn of other routers on their directly attached networks. This information is used to discover neighbors and to learn when neighbors become unreachable or inoperative.

By default, hello packets are sent every 5 seconds. The exception is on low-speed, nonbroadcast multiaccess (NBMA) media, where the default hello interval is 60 seconds. Low speed is considered to be a rate of T1 or slower, as specified with the **bandwidth** interface configuration command. The default hello interval remains 5 seconds for high-speed NBMA networks. Note that for the purposes of EIGRP, Frame Relay and Switched Multimegabit Data Service (SMDS) networks may or may not be considered to be NBMA. These networks are considered NBMA only if the interface has not been configured to use physical multicasting.

You can configure the hold time on a specified interface for a particular EIGRP routing process designated by the autonomous system number. The hold time is advertised in hello packets and indicates to neighbors the length of time they should consider the sender valid. The default hold time is three times the hello interval, or 15 seconds. For slow-speed NBMA networks, the default hold time is 180 seconds.

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

**Note**  Do not adjust the hold time without advising your technical support personnel.

## Split Horizon

Split horizon controls the sending of EIGRP update and query packets. This is a protocol-independent parameter that works for IP and IPX. When split horizon is enabled on an interface, update and query packets are not sent for destinations for which this interface is the next hop. Controlling update and query packets in this manner reduces the possibility of routing loops.

By default, split horizon is enabled on all interfaces.

Split horizon blocks route information from being advertised by a router out of any interface from which that information originated. This behavior usually optimizes communications among multiple routing devices, particularly when links are broken. However, with nonbroadcast networks (such as Frame Relay and SMDS), situations can arise for which this behavior is less than ideal. For these situations, including networks in which you have EIGRP configured, you may want to disable split horizon.

## Link Bandwidth Percentage

By default, EIGRP packets consume a maximum of 50 percent of the link bandwidth, as configured with the **bandwidth** interface configuration command for autonomous system configurations, and with the **bandwidth-percent** command for named configurations. You might want to change that value if a

different level of link utilization is required or if the configured bandwidth does not match the actual link bandwidth (it may have been configured to influence route metric calculations). This is a protocol-independent parameter that works for IP and IPX.

# EIGRP Stub Routing

The EIGRP Stub Routing feature improves network stability, reduces resource utilization, and simplifies stub router configuration.

Stub routing is commonly used in a hub-and-spoke network topology. In a hub-and-spoke network, one or more end (stub) networks are connected to a remote router (the spoke) that is connected to one or more distribution routers (the hub). The remote router is adjacent only to one or more distribution routers. The only route for IP traffic to follow into the remote router is through a distribution router. This type of configuration is commonly used in WAN topologies where the distribution router is directly connected to a WAN. The distribution router can be connected to many more remote routers. Often, the distribution router will be connected to many remote routers. In a hub-and-spoke topology, the remote router must forward all nonlocal traffic to a distribution router, so it becomes unnecessary for the remote router to hold a complete routing table. Generally, the distribution router need not send anything more than a default route to the remote router.

When using the EIGRP Stub Routing feature, you need to configure the distribution and remote routers to use EIGRP, and to configure only the remote router as a stub. Only specified routes are propagated from the remote (stub) router. The stub router responds to all queries for summaries, connected routes, redistributed static routes, external routes, and internal routes with the message "inaccessible." A router that is configured as a stub will send a special peer information packet to all neighboring routers to report its status as a stub router.

Any neighbor that receives a packet informing it of the stub status will not query the stub router for any routes, and a router that has a stub peer will not query that peer. The stub router will depend on the distribution router to send the proper updates to all peers.

Figure 3 shows a simple hub-and-spoke configuration.

***Figure 3        Simple Hub-and-Spoke Network***
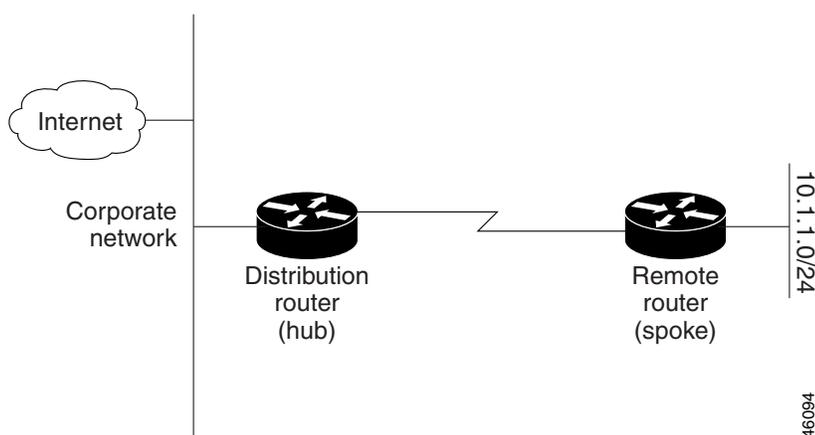


The stub routing feature by itself does not prevent routes from being advertised to the remote router. In the example in Figure 3, the remote router can access the corporate network and the Internet through the distribution router only. Having a full route table on the remote router, in this example, would serve no functional purpose because the path to the corporate network and the Internet would always be through the distribution router. The larger route table would only reduce the amount of memory required by the

remote router. Bandwidth and memory can be conserved by summarizing and filtering routes in the distribution router. The remote router need not receive routes that have been learned from other networks because the remote router must send all nonlocal traffic, regardless of destination, to the distribution router. If a true stub network is desired, the distribution router should be configured to send only a default route to the remote router. The EIGRP Stub Routing feature does not automatically enable summarization on the distribution router. In most cases, the network administrator will need to configure summarization on the distribution routers.

**Note**    When configuring the distribution router to send only a default route to the remote router, you must use the **ip classless** command on the remote router. By default, the **ip classless** command is enabled in all Cisco IOS images that support the EIGRP Stub Routing feature.

Without the stub feature, even after the routes that are sent from the distribution router to the remote router have been filtered or summarized, a problem might occur. If a route is lost somewhere in the corporate network, EIGRP could send a query to the distribution router, which in turn would send a query to the remote router even if routes are being summarized. If there is a problem communicating over the WAN link between the distribution router and the remote router, an EIGRP stuck in active (SIA) condition could occur and cause instability elsewhere in the network. The EIGRP Stub Routing feature allows a network administrator to prevent queries from being sent to the remote router.

## Dual-Homed Remote Topology

In addition to a simple hub-and-spoke network where a remote router is connected to a single distribution router, the remote router can be dual-homed to two or more distribution routers. This configuration adds redundancy and introduces unique issues, and the stub feature helps to address some of these issues.

A dual-homed remote router will have two or more distribution (hub) routers. However, the principles of stub routing are the same as they are with a hub-and-spoke topology. Figure 4 shows a common dual-homed remote topology with one remote router, but 100 or more routers could be connected on the same interfaces on distribution router 1 and distribution router 2. The remote router will use the best route to reach its destination. If distribution router 1 experiences a failure, the remote router can still use distribution router 2 to reach the corporate network.

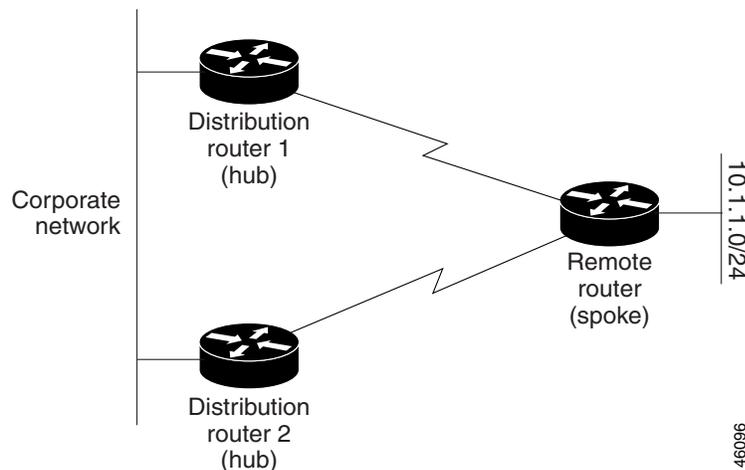*Figure 4*        *Simple Dual-Homed Remote Topology*

Figure 4 shows a simple dual-homed remote with one remote router and two distribution routers. Both distribution routers maintain routes to the corporate network and stub network 10.1.1.0/24.

Dual-homed routing can introduce instability into an EIGRP network. In Figure 5, distribution router 1 is directly connected to network 10.3.1.0/24. If summarization or filtering is applied on distribution router 1, the router will advertise network 10.3.1.0/24 to all of its directly connected EIGRP neighbors (distribution router 2 and the remote router).

**Figure 5** *Dual-Homed Remote Topology with Distribution Router 1 Connected to Two Networks*
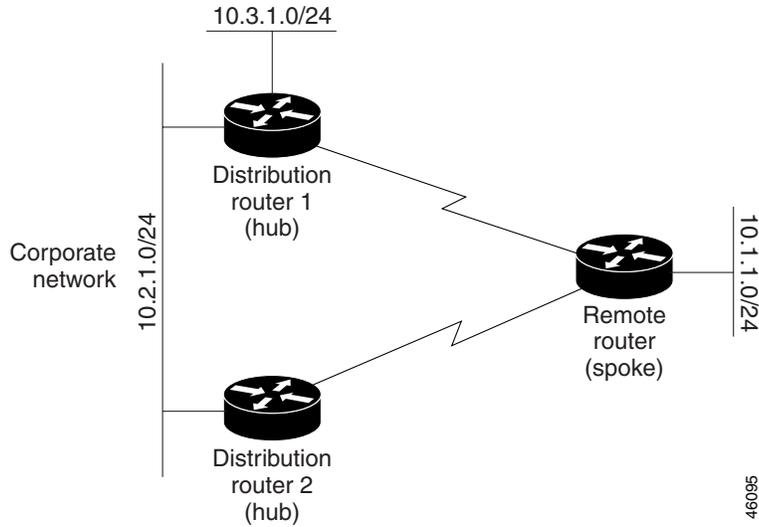


Figure 5 shows a simple dual-homed remote router where distribution router 1 is connected to both network 10.3.1.0/24 and network 10.2.1.0/24.

If the 10.2.1.0/24 link between distribution router 1 and distribution router 2 has failed, the lowest cost path to network 10.3.1.0/24 from distribution router 2 is through the remote router (see Figure 6). This route is not desirable because the traffic that was previously traveling across the corporate network 10.2.1.0/24 would now be sent across a much lower bandwidth connection. The over utilization of the lower bandwidth WAN connection can cause a number of problems that might affect the entire corporate network. The use of the lower bandwidth route that passes through the remote router might cause WAN EIGRP distribution routers to be dropped. Serial lines on distribution and remote routers could also be dropped, and EIGRP SIA errors on the distribution and core routers could occur.

*Figure 6*        *Dual-Homed Remote Topology with a Failed Route to a Distribution Router*
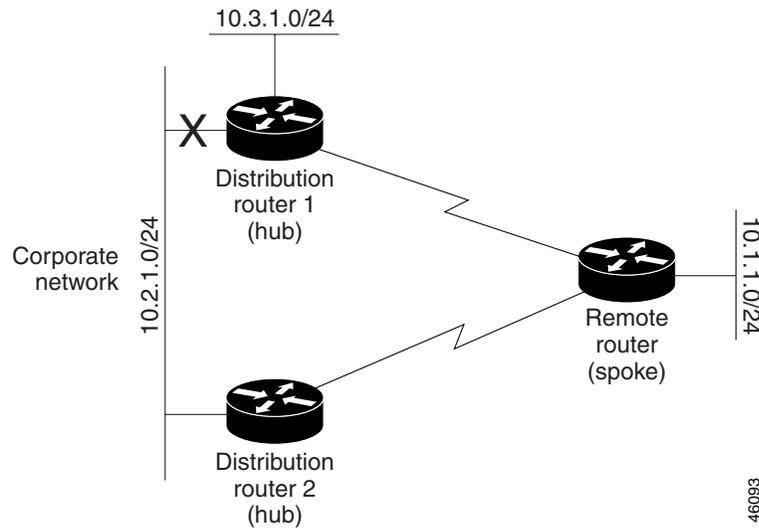


It is not desirable for traffic from distribution router 2 to travel through any remote router in order to reach network 10.3.1.0/24. If the links are sized to manage the load, it would be acceptable to use one of the backup routes. However, most networks of this type have remote routers located at remote offices with relatively slow links. This problem can be prevented if proper summarization is configured on the distribution router and remote router.

It is typically undesirable for traffic from a distribution router to use a remote router as a transit path. A typical connection from a distribution router to a remote router would have much less bandwidth than a connection at the network core. Attempting to use a remote router with a limited bandwidth connection as a transit path would generally produce excessive congestion to the remote router. The EIGRP Stub Routing feature can prevent this problem by preventing the remote router from advertising core routes back to distribution routers. Routes learned by the remote router from distribution router 1 will not be advertised to distribution router 2. Because the remote router will not advertise core routes to distribution router 2, the distribution router will not use the remote router as a transit for traffic destined for the network core.

The EIGRP Stub Routing feature can help to provide greater network stability. In the event of network instability, this feature prevents EIGRP queries from being sent over limited bandwidth links to nontransit routers. Instead, distribution routers to which the stub router is connected answer the query on behalf of the stub router. This feature greatly reduces the chance of further network instability due to congested or problematic WAN links. The EIGRP Stub Routing feature also simplifies the configuration and maintenance of hub-and-spoke networks. When stub routing is enabled in dual-homed remote configurations, it is no longer necessary to configure filtering on remote routers to prevent those remote routers from appearing as transit paths to the hub routers.

⚠
**Caution**    The EIGRP Stub Routing feature should be used only on stub routers. A stub router is defined as a router connected to the network core or distribution layer through which core transit traffic should not flow. A stub router should not have any EIGRP neighbors other than distribution routers. Ignoring this restriction will cause undesirable behavior.

✎

**Note** Multiaccess interfaces such as ATM, Ethernet, Frame Relay, ISDN PRI, and X.25 are supported by the EIGRP Stub Routing feature only when all routers on that interface, except the hub, are configured as stub routers.

# EIGRP Stub Routing Leak Map Support

In EIGRP stub routing configurations where there is a remote site with more than one router, only one of the remote routers can be configured as the stub router. If you have two distribution layer routers, and two routers at a remote site, there is no way to declare both remote routers as stub routers. If one remote router is configured as a stub router, the other remote router cannot learn routes toward the network core if the link between the stub router and the distribution layer router fails and cannot route around the failed link.

The stub router cannot readvertise routes it has learned from any neighboring EIGRP router. To resolve this, a leak map configuration can be added to the EIGRP stub routing feature that allows a selected set of learned routes to be readvertised to other peers. The set of routes allowed through the stub router are specified using a standard route map, so that routes can be matched based on tags, prefixes, or interfaces. These routes are marked using the site of origin code mechanism, which prevents the routes permitted through the stub from being readvertised into the core of the network.

Configure the **eigrp stub** command with the **leak-map** keyword to configure the EIGRP stub routing feature to reference a leak map that identifies routes that are allowed to be advertised on an EIGRP stub router that would normally have been suppressed.

# How to Configure EIGRP

# Enabling EIGRP: Autonomous System Configuration

Perform this task to enable EIGRP and create an EIGRP routing process. EIGRP sends updates to the interfaces in the specified networks. If you do not specify the network of an interface, the interface will not be advertised in any EIGRP update.

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

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router eigrp** *autonomous-system-number*
4. **network** *network-number*

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| Step 1 | `enable`<br><br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br><br>- Enter your password if prompted. |
| Step 2 | `configure terminal`<br><br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| Step 3 | `router eigrp autonomous-system-number`<br><br>**Example:**<br>`Router(config)# router eigrp 1` | Configures an EIGRP routing process and enters router configuration mode.<br><br>- A maximum of 30 EIGRP routing processes can be configured. |
| Step 4 | `network network-number`<br><br>**Example:**<br>`Router(config-router)# network 172.16.0.0` | Associates networks with an EIGRP routing process. |

# Enabling EIGRP: Named Configuration

Perform this task to enable EIGRP and to create an EIGRP routing process. EIGRP sends updates to the interfaces in the specified networks. If you do not specify the network of an interface, the interface will not be advertised in any EIGRP update.

Configuring the **router eigrp** command with the *virtual-instance-name* argument creates an EIGRP configuration referred to as an EIGRP named configuration. 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.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router eigrp** *virtual-instance-name*
4. **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*

   or

   **address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*
5. **network** *ip-address* [*wildcard-mask*]

## DETAILED STEPS

|  | Command or Action | Purpose |
|---|---|---|
| Step 1 | `enable`<br><br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| Step 2 | `configure terminal`<br><br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| Step 3 | `router eigrp` *virtual-instance-name*<br><br>**Example:**<br>`Router(config)# router eigrp virtual-name1` | Configures the EIGRP routing process. |

| | Command or Action | Purpose |
|---|---|---|
| Step 4 | **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>or<br><br>**address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>**Example:**<br>Router(config-router)# address-family ipv4 autonomous-system 45000 | Enters address family configuration mode to configure an EIGRP IPv4 or IPv6 routing instance. |
| Step 5 | **network** *ip-address* [*wildcard-mask*]<br><br>**Example:**<br>Router(config-router-af)# network 172.16.0.0 | Specifies a network for the EIGRP routing process. |

# Configuring Optional EIGRP Parameters: Autonomous System Configuration

Perform this task to configure optional EIGRP parameters including applying offsets to routing metrics, adjusting EIGRP metrics, and disabling automatic summarization in an EIGRP autonomous system configuration.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router eigrp** *autonomous-system*
4. **network** *ip-address* [*wildcard-mask*]
5. **passive-interface** [**default**] [*interface-type interface-number*]
6. **offset-list** [*access-list-number* | *access-list-name*] {**in** | **out**} *offset* [*interface-type interface-number*]
7. **metric weights** *tos k1 k2 k3 k4 k5*
8. **no auto-summary**
9. **exit**

**DETAILED STEPS**

| | Command or Action | Purpose |
|---|---|---|
| Step 1 | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| Step 2 | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |
| Step 3 | **router eigrp** *autonomous-system*<br><br>**Example:**<br>Router(config)# router eigrp 1 | Enables an EIGRP routing process and enters router configuration mode.<br><br>• A maximum of 30 EIGRP routing processes can be configured. |
| Step 4 | **network** *ip-address* [*wildcard-mask*]<br><br>**Example:**<br>Router(config-router)# network 172.16.0.0 | Associates networks with an EIGRP routing process. |
| Step 5 | **passive-interface** [**default**] [*interface-type interface-number*]<br><br>**Example:**<br>Router(config-router)# passive-interface | (Optional) Suppresses EIGRP hello packets and routing updates on interfaces while still including the interface addresses in the topology database. |
| Step 6 | **offset-list** [*access-list-number* \| *access-list-name*] {**in** \| **out**} *offset* [*interface-type interface-number*]<br><br>**Example:**<br>Router(config-router)# offset-list 21 in 10 ethernet 0 | (Optional) Applies an offset to routing metrics. |
| Step 7 | **metric weights** *tos k1 k2 k3 k4 k5*<br><br>**Example:**<br>Router(config-router)# metric weights 0 2 0 2 0 0 | (Optional) Adjusts the EIGRP metric or K value.<br><br>• EIGRP uses the following formula to determine the total metric to the network:<br><br>EIGRP Metric = 256*((K1*Bw) + (K2*Bw)/(256-Load) + (K3*Delay)*(K5/(Reliability + K4)))<br><br>**Note** If K5 is 0 then (K5/ (Reliability + K4)) is defined as 1. |

| | Command or Action | Purpose |
|---|---|---|
| Step 8 | `no auto-summary`<br><br>**Example:**<br>`Router(config-router)# no auto-summary` | (Optional) Disables automatic summarization.<br><br>**Note**     Automatic summarization is enabled by default. |
| Step 9 | `exit`<br><br>**Example:**<br>`Router(config-router)# exit` | Exits router configuration mode. |

# Configuring Optional EIGRP Parameters: Named Configuration

Perform this task to configure optional EIGRP named configuration parameters including applying offsets to routing metrics, adjusting EIGRP metrics, and disabling automatic summarization.

**SUMMARY STEPS**

1. **enable**

2. **configure terminal**

3. **router eigrp** *virtual-instance-name*

4. **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*

   or

   **address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*

5. **network** *ip-address* [*wildcard-mask*]

6. **metric weights** *tos k1 k2 k3 k4 k5*

7. **af-interface** {**default** | *interface-type interface-number*}

8. **passive-interface** [**default**] [*interface-type interface-number*]

9. **bandwidth-percent** *maximum-bandwidth-percentage*

10. **exit-af-interface**

11. **topology** {**base** | *topology-name* **tid** *number*}

12. **offset-list** [*access-list-number* | *access-list-name*] {**in** | **out**} *offset* [*interface-type interface-number*]

13. **no auto-summary**

14. **exit-af-topology**

**DETAILED STEPS**

| | Command or Action | Purpose |
|---|---|---|
| Step 1 | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| Step 2 | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |
| Step 3 | **router eigrp** *virtual-instance-name*<br><br>**Example:**<br>Router(config)# router eigrp virtual-name1 | Enables an EIGRP routing process and enters router configuration mode. |
| Step 4 | **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>or<br><br>**address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>**Example:**<br>Router(config-router)# address-family ipv4 autonomous-system 45000 | Enters address family configuration mode to configure an EIGRP IPv4 or IPv6 routing instance. |
| Step 5 | **network** *ip-address* [*wildcard-mask*]<br><br>**Example:**<br>Router(config-router-af)# network 172.16.0.0 | Specifies a network for the EIGRP routing process. |
| Step 6 | **metric weights** *tos k1 k2 k3 k4 k5*<br><br>**Example:**<br>Router(config-router-af)# metric weights 0 2 0 2 0 0 | (Optional) Adjusts the EIGRP metric or K value.<br><br>• EIGRP uses the following formula to determine the total metric to the network:<br><br>EIGRP Metric = 256*((K1*Bw) + (K2*Bw)/(256-Load) + (K3*Delay)*(K5/(Reliability + K4)))<br><br>**Note** If K5 is 0 then (K5/ (Reliability + K4)) is defined as 1. |
| Step 7 | **af-interface** {**default** \| *interface-type interface-number*}<br><br>**Example:**<br>Router(config-router-af)# af-interface ethernet0/0 | Enters address family interface configuration mode and configures interface-specific EIGRP commands. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 8** | **passive-interface** [**default**] [*interface-type interface-number*] | Suppresses EIGRP hello packets and routing updates on interfaces while still including the interface addresses in the topology database. |
| | **Example:**<br>Router(config-router-af-interface)#<br>passive-interface | |
| **Step 9** | **bandwidth-percent** *maximum-bandwidth-percentage* | Configures the percentage of bandwidth that may be used by an EIGRP address family on an interface. |
| | **Example:**<br>Router(config-router-af-interface)#<br>bandwidth-percent 75 | |
| **Step 10** | **exit-af-interface** | Exits address family interface configuration mode. |
| | **Example:**<br>Router(config-router-af-interface)#<br>exit-af-interface | |
| **Step 11** | **topology** {**base** \| *topology-name* **tid** *number*} | Configures an EIGRP process to route IP traffic under the specified topology instance and enters address family topology configuration mode. |
| | **Example:**<br>Router(config-router-af)# topology base | |
| **Step 12** | **offset-list** [*access-list-number* \| *access-list-name*] {**in** \| **out**} *offset* [*interface-type interface-number*] | (Optional) Applies an offset to routing metrics. |
| | **Example:**<br>Router(config-router-af-topology)# offset-list<br>21 in 10 ethernet 0 | |
| **Step 13** | **no auto-summary** | (Optional) Disables automatic summarization. |
| | | **Note** Automatic summarization is enabled by default. |
| | **Example:**<br>Router(config-router-af-topology)# no<br>auto-summary | |
| **Step 14** | **exit-af-topology** | Exits address family topology configuration mode. |
| | **Example:**<br>Router(config-router-af-topology)#<br>exit-af-topology | |

# Configuring EIGRP Redistribution: Autonomous System Configuration

Perform this task to configure redistribution of non-EIGRP protocol metrics into EIGRP metrics and to configure the EIGRP administrative distance in an EIGRP autonomous system configuration.

You must use a default metric to redistribute a protocol into EIGRP, unless you use the **redistribute** command.

Metric defaults have been carefully set to work for a wide variety of networks. Take great care when changing these values.

Default metrics are supported only when you are redistributing from EIGRP or static routes.

An administrative distance is a rating of the trustworthiness of a routing information source, such as an individual router or a group of routers. Numerically, an administrative distance is an integer from 0 to 255. In general, the higher the value, the lower the trust rating. An administrative distance of 255 means the routing information source cannot be trusted at all and should be ignored.

## SUMMARY STEPS

1. **enable**

2. **configure terminal**

3. **router eigrp** *autonomous-system*

4. **network** *ip-address* [*wildcard-mask*]

5. **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**]

6. **distance eigrp** *internal-distance external-distance*

7. **default-metric** *bandwidth delay reliability loading mtu*

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| Step 1 | `enable`<br><br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| Step 2 | `configure terminal`<br><br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| Step 3 | `router eigrp` *autonomous-system*<br><br>**Example:**<br>`Router(config)# router eigrp 1` | Enables an EIGRP routing process and enters router configuration mode.<br><br>• A maximum of 30 EIGRP routing processes can be configured. |
| Step 4 | `network` *ip-address* [*wildcard-mask*]<br><br>**Example:**<br>`Router(config-router)# network 172.16.0.0` | Associates networks with an EIGRP routing process. |
| Step 5 | `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`]<br><br>**Example:**<br>`Router(config-router)# redistribute rip` | Redistributes routes from one routing domain into another routing domain. |

| | Command or Action | Purpose |
|---|---|---|
| Step 6 | **distance eigrp** *internal-distance* *external-distance*<br><br>**Example:**<br>Router(config-router)# distance eigrp 80 130 | Allows the use of two administrative distances—internal and external—that could be a better route to a node. |
| Step 7 | **default-metric** *bandwidth delay reliability* *loading mtu*<br><br>**Example:**<br>Router(config-router)# default-metric 1000 100 250 100 1500 | Sets metrics for EIGRP. |

# Configuring EIGRP Route Summarization: Autonomous System Configuration

Perform this task to configure EIGRP to perform automatic summarization of subnet routes into network-level routes in an EIGRP AS configuration.

**SUMMARY STEPS**

1. **enable**
2. **configure terminal**
3. **router eigrp** *autonomous-system*
4. **exit**
5. **interface** *type*/*number*
6. **ip summary-address eigrp** *as-number ip-address mask* [*admin-distance*] [**leak-map** *name*]
7. **ip bandwidth-percent eigrp** *as-number percent*

**DETAILED STEPS**

| | | |
|---|---|---|
| Step 1 | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| Step 2 | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |
| Step 3 | **router eigrp** *autonomous-system*<br><br>**Example:**<br>Router(config)# router eigrp 101 | Enables an EIGRP routing process and enters router configuration mode.<br><br>• A maximum of 30 EIGRP routing processes can be configured. |

| Step 4 | **exit** | Exits router configuration mode. |
|--------|----------|----------------------------------|
| | **Example:**<br>Router(config-router)# exit | |
| Step 5 | **interface** *type***/***number* | Enters interface configuration mode. |
| | **Example:**<br>Router(config)# interface ethernet0/0 | |
| Step 6 | **ip summary-address eigrp** *as-number ip-address mask* [*admin-distance*] [**leak-map** *name*] | (Optional) Configures a summary aggregate address. |
| | **Example:**<br>Router(config-if)# ip summary-address eigrp 100<br>0.0.0.0 0.0.0.0 | |
| Step 7 | **ip bandwidth-percent eigrp** *as-number percent* | (Optional) Configures the percentage of bandwidth that may be used by EIGRP on an interface. |
| | **Example:**<br>Router(config-if)# ip bandwidth-percent eigrp<br>209 75 | |

# Configuring EIGRP Route Summarization: Named Configuration

Perform this task to configure EIGRP to perform automatic summarization of subnet routes into network-level routes in an EIGRP named configuration.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router eigrp** *virtual-instance-name*
4. **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*

   or

   **address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*
5. **af-interface** *interface-type interface-number*
6. **summary-address** *ip-address mask* [*administrative-distance* [**leak-map** *leak-map-name*]]
7. **exit-af-interface**

**DETAILED STEPS**

| | Command or Action | Purpose |
|---|---|---|
| Step 1 | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| Step 2 | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |
| Step 3 | **router eigrp** *virtual-instance-name*<br><br>**Example:**<br>Router(config)# router eigrp virtual-name1 | Enables an EIGRP routing process and enters router configuration mode. |
| Step 4 | **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>or<br><br>**address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>**Example:**<br>Router(config-router)# address-family ipv4 autonomous-system 45000 | Enters address family configuration mode to configure an EIGRP IPv4 or IPv6 routing instance. |
| Step 5 | **af-interface** *interface-type interface-number*}<br><br>**Example:**<br>Router(config-router-af)# af-interface ethernet0/0 | Enters address family interface configuration mode and configures interface-specific EIGRP commands. |
| Step 6 | **summary-address** *ip-address mask* [*administrative-distance* [**leak-map** *leak-map-name*]]<br><br>**Example:**<br>Router(config-router-af-interface)# summary-address 192.168.0.0 255.255.0.0 | Configures a summary address for EIGRP. |
| Step 7 | **exit-af-interface**<br><br>**Example:**<br>Router(config-router-af-interface)# exit-af-interface | Exits address family interface configuration mode. |

# Configuring EIGRP Event Logging: Autonomous System Configuration

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router eigrp** *autonomous-system*
4. **eigrp event-log-size** *size*
5. **eigrp log-neighbor-changes**
6. **eigrp log-neighbor-warnings** [*seconds*]

## DETAILED STEPS

| | | |
|---|---|---|
| **Step 1** | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |
| **Step 3** | **router eigrp** *autonomous-system*<br><br>**Example:**<br>Router(config)# router eigrp 101 | Enables an EIGRP routing process and enters router configuration mode.<br><br>• A maximum of 30 EIGRP routing processes can be configured. |
| **Step 4** | **eigrp event-log-size** *size*<br><br>**Example:**<br>Router(config-router)# eigrp event-log-size 5000010 | (Optional) Sets the size of the EIGRP event log. |
| **Step 5** | **eigrp log-neighbor-changes**<br><br>**Example:**<br>Router(config-router)# eigrp log-neighbor-changes | (Optional) Enables logging of EIGRP neighbor adjacency changes.<br><br>• By default, the system logs EIGRP neighbor adjacency changes to help you monitor the stability of the routing system and detect problems. |
| **Step 6** | **eigrp log-neighbor-warnings** [*seconds*]<br><br>**Example:**<br>Router(config-router)# eigrp log-neighbor-warnings 300 | (Optional) Enables the logging of EIGRP neighbor warning messages. |

# Configuring EIGRP Event Logging: Named Configuration

## SUMMARY STEPS

1. **enable**

2. **configure terminal**

3. **router eigrp** *virtual-instance-name*

4. **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*

   or

   **address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*

5. **eigrp log-neighbor-warnings** [*seconds*]

6. **eigrp log-neighbor-changes**

7. **topology** {**base** | *topology-name* **tid** *number*}

8. **eigrp event-log-size** *size*

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br><br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br><br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **router eigrp** *virtual-instance-name*<br><br>**Example:**<br>`Router(config)# router eigrp virtual-name1` | Enables an EIGRP routing process and enters router configuration mode. |
| **Step 4** | **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>or<br><br>**address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>**Example:**<br>`Router(config-router)# address-family ipv4 autonomous-system 45000` | Enters address family configuration mode to configure an EIGRP IPv4 or IPv6 routing instance. |

| | Command or Action | Purpose |
|---|---|---|
| Step 5 | **eigrp log-neighbor-warnings** [*seconds*]<br><br>**Example:**<br>Router(config-router-af)# eigrp<br>log-neighbor-warnings 300 | (Optional) Enables the logging of EIGRP neighbor warning messages. |
| Step 6 | **eigrp log-neighbor-changes**<br><br>**Example:**<br>Router(config-router-af)# eigrp<br>log-neighbor-changes | (Optional) Enables logging of EIGRP neighbor adjacency changes.<br><br>• By default, the system logs EIGRP neighbor adjacency changes to help you monitor the stability of the routing system and detect problems. |
| Step 7 | **topology** {**base** \| *topology-name* **tid** *number*}<br><br>**Example:**<br>Router(config-router-af)# topology base | Configures an EIGRP process to route IP traffic under the specified topology instance and enters address family topology configuration mode. |
| Step 8 | **eigrp event-log-size** *size*<br><br>**Example:**<br>Router(config-router-af-topology)# eigrp<br>event-log-size 10000 | (Optional) Sets the size of the EIGRP event log. |

# Configuring Equal and Unequal Cost Load Balancing: Autonomous System Configuration

## SUMMARY STEPS

1. **enable**

2. **configure terminal**

3. **router eigrp** *autonomous-system*

4. **traffic-share balanced**

5. **maximum-paths** *number-of-paths*

6. **variance** *multiplier*

## DETAILED STEPS

| | | |
|---|---|---|
| Step 1 | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| Step 2 | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |

| Step 3 | `router eigrp` *autonomous-system* | Enables an EIGRP routing process and enters router configuration mode. |
| | **Example:** | • A maximum of 30 EIGRP routing processes can be configured. |
| | Router(config)# router eigrp 101 | |
| Step 4 | `traffic-share balanced` | Controls how traffic is distributed among routes when multiple routes for the same destination network have different costs. |
| | **Example:** | |
| | Router(config-router)# traffic-share balanced | |
| Step 5 | `maximum-paths` *number-of-paths* | Controls the maximum number of parallel routes that an IP routing protocol can support. |
| | **Example:** | |
| | Router(config-router)# maximum-paths 5 | |
| Step 6 | `variance` *multiplier* | Controls load balancing in an internetwork based on EIGRP. |
| | **Example:** | |
| | Router(config-router)# variance 1 | |

# Configuring Equal and Unequal Cost Load Balancing: Named Configuration

**SUMMARY STEPS**

1. **enable**

2. **configure terminal**

3. **router eigrp** *virtual-instance-name*

4. **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*

   or

   **address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*

5. **topology** {**base** | *topology-name* **tid** *number*}

6. **traffic-share balanced**

7. **maximum-paths** *number-of-paths*

8. **variance** *multiplier*

**DETAILED STEPS**

| | Command or Action | Purpose |
|---|---|---|
| Step 1 | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>- Enter your password if prompted. |
| Step 2 | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |
| Step 3 | **router eigrp** *virtual-instance-name*<br><br>**Example:**<br>Router(config)# router eigrp virtual-name1 | Enables an EIGRP routing process and enters router configuration mode. |
| Step 4 | **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>or<br><br>**address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>**Example:**<br>Router(config-router)# address-family ipv4 autonomous-system 45000 | Enters address family configuration mode to configure an EIGRP IPv4 or IPv6 routing instance. |
| Step 5 | **topology** {**base** \| *topology-name* **tid** *number*}<br><br>**Example:**<br>Router(config-router-af)# topology base | Configures an EIGRP process to route IP traffic under the specified topology instance and enters address family topology configuration mode. |
| Step 6 | **traffic-share balanced**<br><br>**Example:**<br>Router(config-router-af-topology)# traffic-share balanced | Controls how traffic is distributed among routes when multiple routes for the same destination network have different costs. |
| Step 7 | **maximum-paths** *number-of-paths*<br><br>**Example:**<br>Router(config-router-af-topology)# maximum-paths 5 | Controls the maximum number of parallel routes that an IP routing protocol can support. |
| Step 8 | **variance** *multiplier*<br><br>**Example:**<br>Router(config-router-af-topology)# variance 1 | Controls load balancing in an internetwork based on EIGRP. |

# Configuring EIGRP Route Authentication: Autonomous System Configuration

Before you can configure EIGRP route authentication, you must enable EIGRP.

Perform the following task to configure authentication of EIGRP packets in an EIGRP autonomous system configuration.

## SUMMARY STEPS

1. **enable**

2. **configure terminal**

3. **interface** *type slot*

4. **ip authentication mode eigrp** *autonomous-system* **md5**

5. **ip authentication key-chain eigrp** *autonomous-system key-chain*

6. **exit**

7. **key chain** *name-of-chain*

8. **key** *key-id*

9. **key-string** *text*

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

11. **send-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*}

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | `enable`<br><br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| **Step 2** | `configure terminal`<br><br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | `interface` *type slot*<br><br>**Example:**<br>`Router(config)# interface Ethernet0` | Configures an interface type and enters interface configuration mode. |
| **Step 4** | `ip authentication mode eigrp` *autonomous-system* `md5`<br><br>**Example:**<br>`Router(config-if) ip authentication mode eigrp 1 md5` | Enables MD5 authentication in EIGRP packets. |

| | Command or Action | Purpose |
|---|---|---|
| Step 5 | **ip authentication key-chain eigrp** *autonomous-system key-chain*<br><br>**Example:**<br>Router(config-if)# ip authentication key-chain eigrp 1 keychain1 | Enables authentication of EIGRP packets. |
| Step 6 | **exit**<br><br>**Example:**<br>Router(config-if)# exit | Exits to global configuration mode. |
| Step 7 | **key chain** *name-of-chain*<br><br>**Example:**<br>Router(config)# key chain keychain1 | Identifies a key chain and enters key chain configuration mode. |
| Step 8 | **key** *key-id*<br><br>**Example:**<br>Router(config-keychain)# key 1 | Identifies the key number and enters key chain key configuration mode. |
| Step 9 | **key-string** *text*<br><br>**Example:**<br>Router(config-keychain-key)# key-string 0987654321 | Identifies the key string. |
| Step 10 | **accept-lifetime** *start-time* {**infinite** \| *end-time* \| **duration** *seconds*}<br><br>**Example:**<br>Router(config-keychain-key)# accept-lifetime 04:00:00 Jan 4 2007 infinite | (Optional) Specifies the time period during which the key can be received. |
| Step 11 | **send-lifetime** *start-time* {**infinite** \| *end-time* \| **duration** *seconds*}<br><br>**Example:**<br>Router(config-keychain-key)# send-lifetime 04:00:00 Dec 4 2006 infinite | (Optional) Specifies the time period during which the key can be sent. |

# Configuring EIGRP Route Authentication: Named Configuration

## Prerequisites

Before you can configure EIGRP route authentication, you must enable EIGRP.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **router eigrp** *virtual-instance-name*

4. **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system**
   *autonomous-system-number*

   or

   **address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*

5. **network** *ip-address* [*wildcard-mask*]

6. **af-interface** {**default** | *interface-type interface-number*}

7. **authentication key-chain** *name-of-chain*

8. **authentication mode md5**

9. **exit-af-interface**

10. **exit-address-family**

11. **exit**

12. **key chain** *name-of-chain*

13. **key** *key-id*

14. **key-string** *text*

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

16. **send-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*}

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| Step 1 | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| Step 2 | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |
| Step 3 | **router eigrp** *virtual-instance-name*<br><br>**Example:**<br>Router(config)# router eigrp virtual-name1 | Enables an EIGRP routing process and enters router configuration. |

| | Command or Action | Purpose |
|---|---|---|
| Step 4 | **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number* <br><br> or <br><br> **address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number* <br><br> **Example:** <br> Router(config-router)# address-family ipv4 autonomous-system 45000 | Enters address family configuration mode to configure an EIGRP IPv4 or IPv6 routing instance. |
| Step 5 | **network** *ip-address* [*wildcard-mask*] <br><br> **Example:** <br> Router(config-router-af)# network 172.16.0.0 | Associates networks with an EIGRP routing process. |
| Step 6 | **af-interface** {**default** \| *interface-type interface-number*} <br><br> **Example:** <br> Router(config-router-af)# af-interface ethernet0/0 | Enters address family interface configuration mode and configures interface-specific EIGRP commands. |
| Step 7 | **authentication key-chain** *name-of-chain* <br><br> **Example:** <br> Router(config-router-af-interface)# authentication key-chain SITE1 | Specifies an authentication key chain for EIGRP. |
| Step 8 | **authentication mode md5** <br><br> **Example:** <br> Router(config-router-af-interface)# authentication mode md5 | Specifies the type of authentication used in an EIGRP address family for the EIGRP instance. |
| Step 9 | **exit-af-interface** <br><br> **Example:** <br> Router(config-router-af-interface)# exit-af-interface | Exits address family interface configuration mode. |
| Step 10 | **exit-address-family** <br><br> **Example:** <br> Router(config-router-af)# exit-address-family | Exits address family configuration mode. |
| Step 11 | **exit** <br><br> **Example:** <br> Router(config-router)# exit | Exits to global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| Step 12 | **key chain** *name-of-chain*<br><br>**Example:**<br>Router(config)# key chain keychain1 | Identifies a key chain and enters key chain configuration mode. |
| Step 13 | **key** *key-id*<br><br>**Example:**<br>Router(config-keychain)# key 1 | Identifies the key number and enters key chain key configuration mode. |
| Step 14 | **key-string** *text*<br><br>**Example:**<br>Router(config-keychain-key)# key-string 0987654321 | Identifies the key string. |
| Step 15 | **accept-lifetime** *start-time* {**infinite** \| *end-time* \| **duration** *seconds*}<br><br>**Example:**<br>Router(config-keychain-key)# accept-lifetime 04:00:00 Jan 4 2007 infinite | (Optional) Specifies the time period during which the key can be received. |
| Step 16 | **send-lifetime** *start-time* {**infinite** \| *end-time* \| **duration** *seconds*}<br><br>**Example:**<br>Router(config-keychain-key)# send-lifetime 04:00:00 Dec 4 2006 infinite | (Optional) Specifies the time period during which the key can be sent. |

# Adjusting the Interval Between Hello Packets and the Hold Time: Autonomous System Configuration

## Prerequisites

Do not adjust the hold time without advising your technical support personnel.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router eigrp** *autonomous-system-number*
4. **exit**
5. **interface** *slot*/*port*
6. **ip hello-interval eigrp** *autonomous-system-number seconds*
7. **ip hold-time eigrp** *autonomous-system-number seconds*

**DETAILED STEPS**

|        | Command or Action | Purpose |
|--------|-------------------|---------|
| Step 1 | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| Step 2 | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |
| Step 3 | **router eigrp** *autonomous-system-number*<br><br>**Example:**<br>Router(config)# router eigrp 101 | Enables an EIGRP routing process and enters router configuration mode.<br><br>• A maximum of 30 EIGRP routing processes can be configured. |
| Step 4 | **exit**<br><br>**Example:**<br>Router(config-router)# exit | Exits to global configuration mode. |
| Step 5 | **interface** *slot***/***port*<br><br>**Example:**<br>Router(config)# interface Ethernet0/1 | Enters interface configuration mode. |
| Step 6 | **ip hello-interval eigrp** *autonomous-system-number seconds*<br><br>**Example:**<br>Router(config-if)# ip hello-interval eigrp 109 10 | Configures the hello interval for an EIGRP routing process. |
| Step 7 | **ip hold-time eigrp** *autonomous-system-number seconds*<br><br>**Example:**<br>Router(config-if)# ip hold-time eigrp 109 40 | Configures the hold time for an EIGRP routing process.<br><br>**Note** Do not adjust the hold time without advising your technical support personnel. |

# Adjusting the Interval Between Hello Packets and the Hold Time: Named Configuration

## Prerequisites

Do not adjust the hold time without advising your technical support personnel.

**SUMMARY STEPS**

1. **enable**

2. **configure terminal**

3. **router eigrp** *virtual-instance-name*

4. **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system**
   *autonomous-system-number*

   or

   **address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*

5. **af-interface** {**default** | *interface-type interface-number*}

6. **hello-interval** *seconds*

7. **hold-time** *seconds*

**DETAILED STEPS**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br><br>**Example:**<br>`Router> enable` | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br><br>**Example:**<br>`Router# configure terminal` | Enters global configuration mode. |
| **Step 3** | **router eigrp** *virtual-instance-name*<br><br>**Example:**<br>`Router(config)# router eigrp virtual-name1` | Enables an EIGRP routing process and enters router configuration mode. |
| **Step 4** | **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>or<br><br>**address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>**Example:**<br>`Router(config-router)# address-family ipv4 autonomous-system 45000` | Enters address family configuration mode to configure an EIGRP IPv4 or IPv6 routing instance. |
| **Step 5** | **af-interface** {**default** | *interface-type interface-number*}<br><br>**Example:**<br>`Router(config-router-af)# af-interface ethernet0/0` | Enters address family interface configuration mode and configures interface-specific EIGRP commands. |

| | Command or Action | Purpose |
|---|---|---|
| Step 6 | **hello-interval** *seconds*<br><br>**Example:**<br>Router(config-router-af-interface)#<br>hello-interval 10 | Configures the hello interval for an EIGRP address family named configuration. |
| Step 7 | **hold-time** *seconds*<br><br>**Example:**<br>Router(config-router-af-interface)# hold-time 50 | Configures the hold time for an EIGRP address family named configuration. |

# Disabling Split Horizon: Autonomous System Configuration

Split horizon controls the sending of EIGRP update and query packets. When split horizon is enabled on an interface, update and query packets are not sent for destinations for which this interface is the next hop. Controlling update and query packets in this manner reduces the possibility of routing loops.

By default, split horizon is enabled on all interfaces.

Perform the following task to disable split horizon for an EIGRP autonomous system configuration.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *slot*/*port*
4. **no ip split-horizon eigrp** *autonomous-system-number*

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| Step 1 | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| Step 2 | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| Step 3 | `interface` *slot*/*port* <br><br> **Example:** <br> `Router(config)# interface Ethernet0/1` | Enters interface configuration mode. |
| Step 4 | `no ip split-horizon eigrp` *autonomous-system-number* <br><br> **Example:** <br> `Router(config-if)# no ip split-horizon eigrp 101` | Disables split horizon. |

# Disabling Split Horizon and Next-Hop-Self: Named Configuration

Perform this task to disable EIGRP split horizon and next-hop-self for an EIGRP named configuration.

By default, split horizon is enabled on all interfaces.

EIGRP will, by default, set the next-hop value to the local outbound interface address for routes that it is advertising, even when advertising those routes back out the same interface where it learned them. Perform this task to change this default to instruct EIGRP to use the received next hop value when advertising these routes. Disabling next-hop-self is primarily useful in Dynamic Multipoint VPN (DMVPN) spoke-to-spoke topologies.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router eigrp** *virtual-instance-name*
4. **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*

   or

   **address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*
5. **af-interface** {**default** | *interface-type interface-number*}
6. **no split-horizon**
7. **no next-hop-self**

**DETAILED STEPS**

| | Command or Action | Purpose |
|---|---|---|
| Step 1 | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| Step 2 | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |
| Step 3 | **router eigrp** *virtual-instance-name*<br><br>**Example:**<br>Router(config)# router eigrp virtual-name1 | Enables an EIGRP routing process and enters router configuration mode. |
| Step 4 | **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>or<br><br>**address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>**Example:**<br>Router(config-router)# address-family ipv4 autonomous-system 45000 | Enters address family configuration mode to configure an EIGRP IPv4 or IPv6 routing instance. |
| Step 5 | **af-interface** {**default** | *interface-type interface-number*}<br><br>**Example:**<br>Router(config-router-af)# af-interface ethernet0/0 | Enters address family interface configuration mode and configures interface-specific EIGRP commands. |
| Step 6 | **no split-horizon**<br><br>**Example:**<br>Router(config-router-af-interface)# no split-horizon | Disables EIGRP split horizon. |
| Step 7 | **no next-hop-self**<br><br>**Example:**<br>Router(config-router-af-interface)# no next-hop-self | (Optional) Instructs an EIGRP router to use the received next hop rather than the local outbound interface address as the next hop. |

# Configuring EIGRP Stub Routing: Autonomous System Configuration

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router eigrp** *autonomous-system-number*
4. **network** *ip-address* [**wildcard-mask**]
5. **eigrp stub** [**receive-only**] [**leak-map** *name*] [**connected**] [**static**] [**summary**] [**redistributed**]
6. **exit**
7. **exit**
8. **show ip eigrp neighbors** [*interface-type* | *as-number* | **static** | **detail**]

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| **Step 2** | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |
| **Step 3** | **router eigrp** *autonomous-system-number*<br><br>**Example:**<br>Router(config)# router eigrp 1 | Configures a remote or distribution router to run an EIGRP process and enters router configuration mode. |
| **Step 4** | **network** *ip-address* [**wildcard-mask**]<br><br>**Example:**<br>Router(config-router)# network 172.16.0.0 | Specifies the network address of the EIGRP distribution router. |
| **Step 5** | **eigrp stub** [**receive-only**] [**leak-map** *name*] [**connected**] [**static**] [**summary**] [**redistributed**]<br><br>**Example:**<br>Router(config-router)# eigrp stub connected static | Configures a remote router as an EIGRP stub router. |
| **Step 6** | **exit**<br><br>**Example:**<br>Router(config-router)# exit | Exits router configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| Step 7 | `exit`<br><br>**Example:**<br>`Router(config)# exit` | Exits global configuration mode. |
| Step 8 | `show ip eigrp neighbors` [*interface-type* \| *as-number* \| `static` \| `detail`]<br><br>**Example:**<br>`Router# show ip eigrp neighbors detail`<br><br>`IP-EIGRP neighbors for process 1`<br>`H   Address              Interface   Hold`<br>`Uptime   SRTT   RTO  Q  Seq Type`<br>`                                (sec)`<br>`(ms)      Cnt Num`<br>`0   10.1.1.2             Se3/1       11`<br>`00:00:59   1  4500  0  7`<br>`   Version 12.1/1.2, Retrans: 2, Retries: 0`<br>`   Stub Peer Advertising ( CONNECTED SUMMARY )`<br>`Routes` | (Optional) Verifies that a remote router has been configured as a stub router with EIGRP.<br><br>Enter this command from the distribution router. The last line of the output displays the stub status of the remote or spoke router. |

# Configuring EIGRP Stub Routing: Named Configuration

Perform the following task to configure EIGRP stub routing for an EIGRP named configuration.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router eigrp** *virtual-instance-name*
4. **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*

   or

   **address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*
5. **network** *ip-address* [**wildcard-mask**]
6. **eigrp stub** [**receive-only**] [**leak-map** *name*] [**connected**] [**static** ] [**summary**] [**redistributed**]
7. **exit-address-family**
8. **exit**
9. **exit**
10. **show eigrp address-family** {**ipv4** \| **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **neighbors** [**static**] [**detail**] [*interface-type interface-number*]

**DETAILED STEPS**

| | Command or Action | Purpose |
|---|---|---|
| Step 1 | **enable**<br><br>**Example:**<br>Router> enable | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| Step 2 | **configure terminal**<br><br>**Example:**<br>Router# configure terminal | Enters global configuration mode. |
| Step 3 | **router eigrp** *virtual-instance-name*<br><br>**Example:**<br>Router(config)# router eigrp virtual-name1 | Enables an EIGRP routing process in global configuration mode and enters router configuration mode. |
| Step 4 | **address-family ipv4** [**multicast**] [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>or<br><br>**address-family ipv6** [**unicast**] [**vrf** *vrf-name*] **autonomous-system** *autonomous-system-number*<br><br>**Example:**<br>Router(config-router)# address-family ipv4 autonomous-system 45000 | Enters address family configuration mode to configure an EIGRP IPv4 or IPv6 routing instance. |
| Step 5 | **network** *ip-address* [**wildcard-mask**]<br><br>**Example:**<br>Router(config-router-af)# network 172.16.0.0 | Specifies the network address of the EIGRP distribution router. |
| Step 6 | **eigrp stub** [**receive-only**] [**leak-map** *name*] [**connected**] [**static** ] [**summary**] [**redistributed**]<br><br>**Example:**<br>Router(config-router-af) eigrp stub leak-map map1 | Configures a router as a stub using EIGRP. |
| Step 7 | **exit-address-family**<br><br>**Example:**<br>Router(config-router-af)# exit-address-family | Exits address family configuration mode. |
| Step 8 | **exit**<br><br>**Example:**<br>Router(config-router)# exit | Exits to global configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 9** | `exit`<br><br>**Example:**<br>`Router(config-router)# exit` | Exits to privileged EXEC mode. |
| **Step 10** | `show eigrp address-family {ipv4 \| ipv6} [vrf vrf-name] [autonomous-system-number] [multicast] neighbors [static] [detail] [interface-type interface-number]`<br><br>**Example:**<br>`Router# show eigrp address-family ipv4 neighbors detail` | (Optional) Displays the neighbors that are discovered by EIGRP. |

# Monitoring and Maintaining EIGRP: Autonomous System Configuration

**SUMMARY STEPS**

1. **enable**

2. **show ip eigrp** [**vrf** {*vrf-name* | **\***}] [*autonomous-system-number*] **accounting**

3. **show ip eigrp events** [*starting-event-number ending-event-number*] [**type**]

4. **show ip eigrp interfaces** [**vrf** {*vrf-name* | **\***}] [*autonomous-system-number*] [*type number*] [**detail**]

5. **show ip eigrp** [**vrf** {*vrf-name* | **\***}] [*autonomous-system-number*] **neighbors** [*interface-type* | **static** | **detail**]

6. **show ip eigrp** [**vrf** {*vrf-name* | **\***}] [*autonomous-system-number*] **topology** [*ip-address* [*mask*]] | [**name**] [**active** | **all-links** | **detail-links** | **pending** | **summary** | **zero-successors**]

7. **show ip eigrp** [**vrf** {*vrf-name* | **\***}] [*autonomous-system-number*] **traffic**

**DETAILED STEPS**

**Step 1** **enable**

Enables privileged EXEC mode. Enter your password if prompted.

```
Router# enable
```

**Step 2** **show ip eigrp** [**vrf** {*vrf-name* | **\***}] [*autonomous-system-number*] **accounting**

This command displays prefix accounting information for EIGRP processes. The following is sample output from the command:

```
Router# show ip eigrp vrf VRF1 accounting

EIGRP-IPv4 Accounting for AS(100)/ID(10.0.2.1) VRF(VRF1)
Total Prefix Count: 4  States: A-Adjacency, P-Pending, D-Down
State Address/Source    Interface       Prefix     Restart   Restart/
                                        Count      Count     Reset(s)
   P    Redistributed    ----               0         3          211
   A    10.0.1.2         Et0/0              2         0           84
   P    10.0.2.4         Se2/0              0         2          114
   D    10.0.1.3         Et0/0              0         3            0
```

**Step 3**     **show ip eigrp events** [*starting-event-number ending-event-number*] [**type**]

This command displays the EIGRP event log. The following is sample output from the command:

```
Router# show ip eigrp events

1    02:37:58.171 NSF stale rt scan, peer: 10.0.0.0
2    02:37:58.167 Metric set: 10.0.0.1/24 284700416
3    02:37:58.167 FC sat rdbmet/succmet: 284700416 0
4    02:37:58.167 FC sat nh/ndbmet: 10.0.0.2 284700416
5    02:37:58.167 Find FS: 10.0.0.0/24 284700416
6    02:37:58.167 Rcv update met/succmet: 284956416 284700416
7    02:37:58.167 Rcv update dest/nh: 10.0.0.0/24 10.0.0.1
8    02:37:58.167 Peer nsf restarted: 10.0.0.1 Tunnel0
9    02:36:38.383 Metric set: 10.0.0.0/24 284700416
10   02:36:38.383 RDB delete: 10.0.0.0/24 10.0.0.1
11   02:36:38.383 FC sat rdbmet/succmet: 284700416 0
12   02:36:38.383 FC sat nh/ndbmet: 0.0.0.0 284700416
```

**Step 4**     **show ip eigrp interfaces** [**vrf** {*vrf-name* | **\***}] [*autonomous-system-number*] [*type number*] [**detail**]

This command displays information about interfaces that are configured for EIGRP. The following is sample output from the command:

```
Router# 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
```

**Step 5**     **show ip eigrp** [**vrf** {*vrf-name* | **\***}] [*autonomous-system-number*] **neighbors** [*interface-type* | **static** | **detail**]

This command displays neighbors discovered by EIGRP. The following is sample output from this command:

```
Router# show ip eigrp neighbors

H    Address              Interface      Hold Uptime   SRTT   RTO   Q   Seq
                                         (sec)         (ms)         Cnt Num
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    10.1.2.3             Et0/1            11 00:20:39  2202   5000   0   5
```

**Step 6**     **show ip eigrp** [**vrf** {*vrf-name* | **\***}] [*autonomous-system-number*] **topology** [*ip-address* [*mask*]] | [**name**] [**active** | **all-links** | **detail-links** | **pending** | **summary** | **zero-successors**]

This command displays entries in the EIGRP topology table. The following is sample output from this command:

```
Router# 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 1.1.1.2 (409600/128256), Ethernet0/0
P 172.16.1.0/24, 1 successors, FD is 409600
        via 1.1.1.2 (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
```

**Step 7**    **show ip eigrp** [**vrf** {*vrf-name* | **\***}] [*autonomous-system-number*] **traffic**

This command displays the number of EIGRP packets sent and received. The following is sample output from the command:

```
Router# 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)
```

# Monitoring and Maintaining EIGRP: Named Configuration

## SUMMARY STEPS

1. **enable**

2. **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **accounting**

3. **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **events** [*starting-event-number ending-event-number*] [**errmsg** [*starting-event-number ending-event-number*]] [**sia** [*starting-event-number ending-event-number*]] [**type**]

4. **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **interfaces** [**detail**] [*interface-type interface-number*]

5. **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **neighbors** [**static**] [**detail**] [*interface-type interface-number*]

6. **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **timers**

7. **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **topology** [*topology-name*] [*ip-address*] [**active**] [**all-links**] [**detail-links**] [**pending**] [**summary**] [**zero-successors**] [**route-type** {**connected** | **external** | **internal** | **local** | **redistributed** | **summary** | **vpn**}]

8. **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **traffic**

9. **show eigrp plugins** [*plugin-name*] [**detailed**]

10. **show eigrp protocols** [**vrf** *vrf-name*]

## DETAILED STEPS

**Step 1** **enable**

Enables privileged EXEC mode. Enter your password if prompted.

```
Router# enable
```

**Step 2** **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **accounting**

This command displays prefix accounting information for EIGRP processes. The following is sample output from the command:

```
Router# show eigrp address-family ipv4 22 accounting

EIGRP-IPv4 VR(saf) Accounting for AS(22)/ID(10.0.0.1)
Total Prefix Count: 3   States: A-Adjacency, P-Pending, D-Down
State Address/Source     Interface      Prefix    Restart   Restart/
                                        Count     Count     Reset(s)
  A    10.0.0.2          Et0/0             2         0          0
  P    10.0.2.4          Se2/0             0         2         114
  D    10.0.1.3          Et0/0             0         3          0
```

**Step 3** **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **events** [*starting-event-number ending-event-number*] [**errmsg** [*starting-event-number ending-event-number*]] [**sia** [*starting-event-number ending-event-number*]] [**type**]

This command displays information about EIGRP address-family events. The following is sample output from the command:

```
Router# show eigrp address-family ipv4 3 events

Event information for AS 3:
1 15:37:47.015 Change queue emptied, entries: 1
2 15:37:47.015 Metric set: 10.0.0.0/24 307200
3 15:37:47.015 Update reason, delay: new if 4294967295
4 15:37:47.015 Update sent, RD: 10.0.0.0/24 4294967295
5 15:37:47.015 Update reason, delay: metric chg 4294967295
6 15:37:47.015 Update sent, RD: 10.0.0.0/24 4294967295
7 15:37:47.015 Route installed: 10.0.0.0/24 1.1.1.2
8 15:37:47.015 Route installing: 10.0.0.0/24 10.0.1.2
```

**Step 4** **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **interfaces** [**detail**] [*interface-type interface-number*]

This command displays information about interfaces that are configured for EIGRP. The following is sample output from the command:

```
Router# show eigrp address-family ipv4 4453 interfaces

EIGRP-IPv4 VR(Virtual-name) Address-family Neighbors for AS(4453)
                Xmit Queue   Mean   Pacing Time   Multicast    Pending
Interface   Peers Un/Reliable SRTT  Un/Reliable   Flow Timer   Services
Se0           1      0/0       28      0/15          127           0
Se1           1      0/0       44      0/15          211           0
```

**Step 5** **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **neighbors** [**static**] [**detail**] [*interface-type interface-number*]

This command displays the neighbors that are discovered by EIGRP. The following is sample output from the command:

```
Router# show eigrp address-family ipv4 4453 neighbors

EIGRP-IPv4 VR(Virtual-name) Address-family Neighbors for AS(4453)
Address               Interface    Hold Uptime   SRTT   RTO    Q     Seq
                                   (sec)         (ms)   (ms)   Cnt   Num
172.16.81.28          Ethernet1    13   0:00:41  0      11     4     20
172.16.80.28          Ethernet0    14   0:02:01  0      10     12    24
172.16.80.31          Ethernet0    12   0:02:02  0      4      5     20
```

**Step 6**  **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **timers**

This command displays information about EIGRP timers and expiration times. The following is sample output from the command:

```
Router# show eigrp address-family ipv4 4453 timers

EIGRP-IPv4 VR(Virtual-name) Address-family Timers for AS(4453)
Hello Process
Expiration Type
| 1.022 (parent)
| 1.022 Hello (Et0/0)

Update Process
Expiration Type
| 14.984 (parent)
| 14.984 (parent)
| 14.984 Peer holding

SIA Process
Expiration Type for Topo(base)
| 0.000 (parent)
```

**Step 7**  **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **topology** [*topology-name*] [*ip-address*] [**active**] [**all-links**] [**detail-links**] [**pending**] [**summary**] [**zero-successors**] [**route-type** {**connected** | **external** | **internal** | **local** | **redistributed** | **summary** | **vpn**}]

This command displays entries in the EIGRP topology table. The following is sample output from the command:

```
Router# show eigrp address-family ipv4 4453 topology

EIGRP-IPv4 VR(Virtual-name) Topology Table for AS(4453)/ID(10.0.0.1)
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - Reply status, s - sia Status
P  10.17.17.0/24, 1 successors, FD is 409600
        via 10.10.10.2 (409600/128256), Ethernet3/0
P  172.16.19.0/24, 1 successors, FD is 409600
        via 10.10.10.2 (409600/128256), Ethernet3/0
P  192.168.10.0/24, 1 successors, FD is 281600
        via Connected, Ethernet3/0
P  10.10.10.0/24, 1 successors, FD is 281600
        via Redistributed (281600/0)
```

**Step 8**     **show eigrp address-family** {**ipv4** | **ipv6**} [**vrf** *vrf-name*] [*autonomous-system-number*] [**multicast**] **traffic**

This command displays the number of EIGRP packets that are sent and received. The following is sample output from the command:

```
Router# show eigrp address-family ipv4 4453 traffic

EIGRP-IPv4 VR(virtual-name) Address-family Traffic Statistics for AS(4453)
  Hellos sent/received: 122/122
  Updates sent/received: 3/1
  Queries sent/received: 0/0
  Replies sent/received: 0/0
  Acks sent/received: 0/3
  SIA-Queries sent/received: 0/0
  SIA-Replies sent/received: 0/0
  Hello Process ID: 128
  PDM Process ID: 191
  Socket Queue: 0/2000/1/0 (current/max/highest/drops)
  Input Queue: 0/2000/1/0 (current/max/highest/drops
```

**Step 9**     **show eigrp plugins** [*plugin-name*] [**detailed**]

This command displays general information including the versions of the EIGRP protocol features that are currently running. The following is sample output from the command:

```
Router# show eigrp plugins

EIGRP feature plugins:::
    eigrp-release     :   5.00.00 : Portable EIGRP Release
                      :  19.00.00 : Source Component Release(rel5)
    igrp2             :   3.00.00 : Reliable Transport/Dual Database
    bfd               :   1.01.00 : BFD Platform Support
    mtr               :   1.00.01 : Multi-Topology Routing(MTR)
    eigrp-pfr         :   1.00.01 : Performance Routing Support
    ipv4-af           :   2.01.01 : Routing Protocol Support
    ipv4-sf           :   1.01.00 : Service Distribution Support
    external-client   :   1.02.00 : Service Distribution Client Support
    ipv6-af           :   2.01.01 : Routing Protocol Support
    ipv6-sf           :   1.01.00 : Service Distribution Support
    snmp-agent        :   1.01.01 : SNMP/SNMPv2 Agent Support
```

**Step 10**     **show eigrp protocols** [**vrf** *vrf-name*]

This command displays general information about EIGRP protocols that are currently running. The following is sample output from the command:

```
Router# show eigrp protocols

EIGRP-IPv4 Protocol for AS(10)
Metric weight K1=1, K2=0, K3=1, K4=0, K5=0
NSF-aware route hold timer is 240
Router-ID: 1.1.1.1
Topology : 0 (base)
Active Timer: 3 min
Distance: internal 90 external 170
Maximum path: 4
Maximum hopcount 100
Maximum metric variance 1
EIGRP-IPv4 Protocol for AS(5) VRF(VRF1)
Metric weight K1=1, K2=0, K3=1, K4=0, K5=0
NSF-aware route hold timer is 240
Router-ID: 1.1.1.1
Topology : 0 (base)
Active Timer: 3 min
```

```
Distance: internal 90 external 170
Maximum path: 4
Maximum hopcount 100
Maximum metric variance 1
Total Prefix Count: 0
Total Redist Count: 0
```

# Configuration Examples for EIGRP

## Example: Enabling EIGRP: Autonomous System Configuration

The following example shows how to enable EIGRP in autonomous system configurations:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp 1
Router(config-router)# network 172.16.0.0
```

# Example: Enabling EIGRP: Named Configuration

The following example shows how to enable EIGRP in named configurations:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 45000
Router(config-router-af)# network 172.16.0.0
```

# Example: EIGRP Parameters: Autonomous System Configuration

The following example shows how to configure optional EIGRP autonomous system configuration parameters including applying offsets to routing metrics, adjusting EIGRP metrics, and disabling automatic summarization:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp 1
Router(config-router)# network 172.16.0.0
Router(config-router)# passive-interface
Router(config-router)# offset-list 21 in 10 ethernet 0
Router(config-router)# metric weights 0 2 0 2 0 0
Router(config-router)# no auto-summary
Router(config-router)# exit
```

# Example: EIGRP Parameters: Named Configuration

The following example shows how to configure optional EIGRP named configuration parameters including applying offsets to routing metrics, adjusting EIGRP metrics, and disabling automatic summarization.

```
Router> enable
Router# configure terminal
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 45000
Router(config-router-af)# network 172.16.0.0
Router(config-router-af)# metric weights 0 2 0 2 0 0
Router(config-router-af)# af-interface ethernet0/0
Router(config-router-af-interface)# passive-interface
Router(config-router-af-interface)# bandwidth-percent 75
Router(config-router-af-interface)# exit-af-interface
Router(config-router-af-interface)# topology base
Router(config-router-af-topology)# offset-list 21 in 10 ethernet 0
Router(config-router-af-topology)# no auto-summary
Router(config-router-af-topology)# exit-af-topology
```

# Example: EIGRP Redistribution: Autonomous System Configuration

The following example shows how to configure redistribution of non-EIGRP protocol metrics into EIGRP metrics and to configure the EIGRP administrative distance in an EIGRP autonomous system configuration:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp 1
```

```
Router(config-router)# network 172.16.0.0
Router(config-router)# redistribute rip
Router(config-router)# distance eigrp 80 130
Router(config-router)# default-metric 1000 100 250 100 1500
```

# Example: EIGRP Route Summarization: Autonomous System Configuration

The following example configures route summarization on an interface and also configures the automatic summary feature for an EIGRP autonomous system configuration. This configuration causes EIGRP to summarize network 10.0.0.0 out Ethernet interface 0 only.

```
Router> enable
Router# configure terminal
Router(config)# router eigrp 101
Router(config-router)# exit
Router(config)# interface ethernet0
Router(config-if)# ip summary-address eigrp 100 0.0.0.0 0.0.0.0
Router(config-if)# ip bandwidth-percent eigrp 209 75
```

**Note** You should not use the **ip summary-address eigrp** summarization command to generate the default route (0.0.0.0) from an interface. This causes the creation of an EIGRP summary default route to the null 0 interface with an administrative distance of 5. The low administrative distance of this default route can cause this route to displace default routes learned from other neighbors from the routing table. If the default route learned from the neighbors is displaced by the summary default route, or if the summary route is the only default route present, all traffic destined for the default route will not leave the router, instead, this traffic will be sent to the null 0 interface, where it is dropped.

The recommended way to send only the default route out a given interface is to use a **distribute-list** command. You can configure this command to filter all outbound route advertisements sent out the interface with the exception of the default (0.0.0.0).

# Example: EIGRP Route Summarization: Named Configuration

The following example configures route summarization on an interface and also configures the automatic summary feature for an EIGRP named configuration. This configuration causes EIGRP to summarize network 192.168.0.0 out Ethernet interface 0/0 only.

```
Router> enable
Router# configure terminal
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 45000
Router(config-router-af)# af-interface ethernet0/0
Router(config-router-af-interface)# summary-address 192.168.0.0 255.255.0.0
Router(config-router-af-interface)# exit-af-interface
```

# Example: EIGRP Event Logging: Autonomous System Configuration

The following example shows how to configure EIGRP event logging parameters for an EIGRP autonomous system configuration:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp 1
```

```
Router(config-router)# eigrp event-log-size 5000
Router(config-router)# eigrp log-neighbor-changes
Router(config-router)# eigrp log-neighbor-warnings 300
```

# Example: EIGRP Event Logging: Named Configuration

The following example shows how to configure EIGRP event logging parameters, including setting the size of the EIGRP event log, for an EIGRP named configuration:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 45000
Router(config-router-af)# eigrp log-neighbor-warnings 300
Router(config-router-af)# eigrp log-neighbor-changes
Router(config-router-af)# topology base
Router(config-router-af-topology)# eigrp event-log-size 10000
```

# Example: Equal and Unequal Cost Load Balancing: Autonomous System Configuration

The following example shows how to configure traffic distribution among routes, the maximum number of parallel routes, and load balancing in an EIGRP named configuration network:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp 1
Router(config-router)# traffic-share balanced
Router(config-router)# maximum-paths 5
Router(config-router)# variance 1
```

# Example: Equal and Unequal Cost Load Balancing: Named Configuration

The following example shows how to configure traffic distribution among routes, the maximum number of parallel routes, and load-balancing in an EIGRP named configuration network:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 45000
Router(config-router-af)# topology base
Router(config-router-af-topology)# traffic-share balanced
Router(config-router-af-topology)# maximum-paths 5
Router(config-router-af-topology)# variance 1
```

# Example: EIGRP Route Authentication: Autonomous System Configuration

The following example enables MD5 authentication on EIGRP packets in autonomous system 1. Figure 7 shows the scenario.

*Figure 7*        ***EIGRP Route Authentication Scenario***
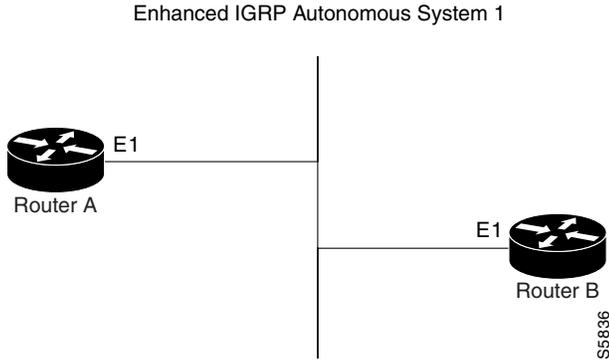
Enhanced IGRP Autonomous System 1



### Router A Configuration

```
Router> enable
Router(config)# configure terminal
Router(config)# router eigrp 1
Router(config-router)# exit
Router(config)# interface ethernet 1
Router(config-if)# ip authentication mode eigrp 1 md5
Router(config-if)# ip authentication key-chain eigrp 1 key1
Router(config-if)# exit
Router(config)# key chain key1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string 0987654321
Router(config-keychain-key)# accept-lifetime 04:00:00 Dec 4 2006 infinite
Router(config-keychain-key)# send-lifetime 04:00:00 Dec 4 2006 04:48:00 Dec 4 1996
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string 1234567890
Router(config-keychain-key)# accept-lifetime 04:00:00 Jan 4 2007 infinite
Router(config-keychain-key)# send-lifetime 04:45:00 Jan 4 2007 infinite
```

### Router B Configuration

```
Router> enable
Router(config)# configure terminal
Router(config)# router eigrp 1
Router(config-router)# exit
Router(config)# interface ethernet 1
Router(config-if)# ip authentication mode eigrp 1 md5
Router(config-if)# ip authentication key-chain eigrp 1 key2
Router(config-if)# exit
Router(config)# key chain key2
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string 0987654321
Router(config-keychain-key)# accept-lifetime 04:00:00 Dec 4 2006 infinite
Router(config-keychain-key)# send-lifetime 04:00:00 Dec 4 2006 infinite
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string 1234567890
Router(config-keychain-key)# accept-lifetime 04:00:00 Jan 4 2007 infinite
Router(config-keychain-key)# send-lifetime 04:45:00 Jan 4 2007 infinite
```

Router A will accept and attempt to verify the MD5 digest of any EIGRP packet with a key equal to 1. It will also accept a packet with a key equal to 2. All other MD5 packets will be dropped. Router A will send all EIGRP packets with key 2.

Router B will accept key 1 or key 2, and will use key 1 to send MD5 authentication, because key 1 is the first valid key off the key chain. Key 1 will no longer be valid to be used for sending after December 4, 2006. After this date, key 2 would be used to send MD5 authentication, because it is valid until January 4, 2007.

# Example: EIGRP Route Authentication: Named Configuration

The following example enables MD5 authentication on EIGRP packets in a named configuration. Figure 7 shows the scenario.

### Router A Configuration

```
Router> enable
Router# configure terminal
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 45000
Router(config-router-af)# network 172.16.0.0
Router(config-router-af)# af-interface ethernet0/0
Router(config-router-af-interface)# authentication key-chain SITE1
Router(config-router-af-interface)# authentication mode md5
Router(config-router-af-interface)# exit-af-interface
Router(config-router-af)# exit-address-family
Router(config-router)# exit
Router(config)# key chain SITE1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string 0987654321
Router(config-keychain-key)# accept-lifetime 04:00:00 Dec 4 2006 infinite
Router(config-keychain-key)# send-lifetime 04:00:00 Dec 4 2006 infinite
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string 1234567890
Router(config-keychain-key)# accept-lifetime 04:00:00 Jan 4 2007 infinite
Router(config-keychain-key)# send-lifetime 04:45:00 Jan 4 2007 infinite
```

### Router B Configuration

```
Router> enable
Router# configure terminal
Router(config)# router eigrp virtual-name2
Router(config-router)# address-family ipv4 autonomous-system 45000
Router(config-router-af)# network 172.16.0.0
Router(config-router-af)# af-interface ethernet0/0
Router(config-router-af-interface)# authentication key-chain SITE2
Router(config-router-af-interface)# authentication mode md5
Router(config-router-af-interface)# exit-af-interface
Router(config-router-af)# exit-address-family
Router(config-router)# exit
Router(config)# key chain SITE2
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string 0987654321
Router(config-keychain-key)# accept-lifetime 04:00:00 Jan 4 2007 infinite
Router(config-keychain-key)# send-lifetime 04:00:00 Dec 4 2006 infinite
```

Router A will accept and attempt to verify the MD5 digest of any EIGRP packet with a key equal to 1. It will also accept a packet with a key equal to 2. All other MD5 packets will be dropped. Router A will send all EIGRP packets with key 2.

Router B will accept key 1 or key 2, and will use key 1 to send MD5 authentication, because key 1 is the first valid key off the key chain. Key 1 will no longer be valid to be used for sending after December 4, 2006. After this date key 2 would be used to send MD5 authentication, because it is valid until January 4, 2007.

# Example: Adjusting the Interval Between Hello Packets and the Hold Time: Autonomous System Configuration

The following example shows how to adjust the interval between hello packets and the hold time in an EIGRP autonomous system configuration:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp 1
Router(config-router)# exit
Router(config)# interface Ethernet0/1
Router(config-if)# ip hello-interval eigrp 109 10
Router(config-if)# ip hold-time eigrp 109 40
```

# Example: Adjusting the Interval Between Hello Packets and the Hold Time: Named Configuration

The following example shows how to adjust the interval between hello packets and the hold time in an EIGRP named configuration:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 45000
Router(config-router-af)# af-interface ethernet0/0
Router(config-router-af-interface)# hello-interval 10
Router(config-router-af-interface)# hold-time 50
```

# Example: Disabling Split Horizon: Autonomous System Configuration

Split horizon is enabled on all interfaces by default. The following example shows how to disable split horizon for an EIGRP autonomous system configuration:

```
Router> enable
Router# configure terminal
Router(config)# router eigrp 1
Router(config-router)# exit
Router(config)# interface Ethernet0/1
Router(config-if)# no ip split-horizon eigrp 101
```

# Example: Disabling Split Horizon and Next-Hop-Self: Named Configuration

Split horizon is enabled on all interfaces by default. The following example shows how to disable split horizon in an EIGRP named configuration.

EIGRP will, by default, set the next-hop value to the local outbound interface address for routes that it is advertising, even when advertising those routes back out the same interface where it learned them. The following example shows how to change this default to instruct EIGRP to use the received next hop value when advertising these routes in an EIGRP named configuration. Disabling next-hop-self is primarily useful in Dynamic Multipoint VPN (DMVPN) spoke-to-spoke topologies.

```
Router> enable
Router# configure terminal
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 45000
Router(config-router-af)# af-interface ethernet0/0
Router(config-router-af-interface)# no split-horizon
Router(config-router-af-interface)# no next-hop-self
```

# Example: EIGRP Stub Routing: Autonomous System Configuration

A router that is configured as a stub with the **eigrp stub** command shares connected and summary routing information with all neighbor routers by default. Six keywords can be used with the **eigrp stub** command to modify this behavior:

- **receive-only**
- **leak-map**
- **connected**
- **static**
- **summary**
- **redistributed**

This section provides configuration examples for all forms of the **eigrp stub** command for an EIGRP autonomous system configuration.

### eigrp stub Command: Example

In the following example, the **eigrp stub** command is used to configure the router as a stub that advertises connected and summary routes:

```
Router(config)# router eigrp 1
Router(config-router)# network 10.0.0.0
Router(config-router)# eigrp stub
```

### eigrp stub connected static Command: Example

In the following example, the **eigrp stub** command is issued with the **connected** and **static** keywords to configure the router as a stub that advertises connected and static routes (sending summary routes will not be permitted):

```
Router(config)# router eigrp 1
Router(config-router)# network 10.0.0.0
Router(config-router)# eigrp stub connected static
```

### eigrp stub receive-only Command: Example

In the following example, the **eigrp stub** command is issued with the **receive-only** keyword to configure the router as a receive-only neighbor (connected, summary, and static routes will not be sent):

```
Router(config)# router eigrp 1
Router(config-router)# network 10.0.0.0
Router(config-router)# eigrp stub receive-only
```

**eigrp stub redistributed Command: Example**

In the following example, the **eigrp stub** command is issued with the **redistributed** keyword to configure the router to advertise other protocols and autonomous systems:

```
Router(config)# router eigrp 1
Router(config-router)# network 10.0.0.0
Router(config-router)# eigrp stub redistributed
```

**eigrp stub leak-map Command: Example**

In the following example, the **eigrp stub** command is issued with the **leak-map** *name* keyword and argument pair to configure the router to reference a leak map that identifies routes that would have been suppressed:

```
Router(config)# router eigrp
Router(config-router)# network 10.0.0.0
Router(config-router) eigrp stub leak-map map1
```

# Example: EIGRP Stub Routing: Named Configuration

A router that is configured as a stub with the **eigrp stub** command shares connected and summary routing information with all neighbor routers by default. Six keywords can be used with the **eigrp stub** command to modify this behavior:

- **receive-only**
- **leak-map**
- **connected**
- **static**
- **summary**
- **redistributed**

This section provides configuration examples for all forms of the **eigrp stub** command for an EIGRP named configuration.

**eigrp stub Command: Example**

In the following example, the **eigrp stub** command is used to configure the router as a stub that advertises connected and summary routes:

```
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 4453
Router(config-router-af)# network 10.0.0.0
Router(config-router-af) eigrp stub
```

**eigrp stub connected static Command: Example**

In the following named configuration example, the **eigrp stub** command is issued with the **connected** and **static** keywords to configure the router as a stub that advertises connected and static routes (sending summary routes will not be permitted):

```
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 4453
Router(config-router-af)# network 10.0.0.0
Router(config-router-af)# eigrp stub connected static
```

### eigrp stub receive-only Command: Example

In the following named configuration example, the **eigrp stub** command is issued with the **receive-only** keyword to configure the router as a receive-only neighbor (connected, summary, and static routes will not be sent):

```
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 4453
Router(config-router-af)# network 10.0.0.0
Router(config-router-af)# eigrp stub receive-only
```

### eigrp stub redistributed Command: Example

In the following named configuration example, the **eigrp stub** command is issued with the **redistributed** keyword to configure the router to advertise other protocols and autonomous systems:

```
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 4453
Router(config-router-af)# network 10.0.0.0
Router(config-router-af) eigrp stub redistributed
```

### eigrp stub leak-map Command: Example

In the following named configuration example, the **eigrp stub** command is issued with the **leak-map** *name* keyword and argument pair to configure the router to reference a leak map that identifies routes that would normally have been suppressed:

```
Router(config)# router eigrp virtual-name1
Router(config-router)# address-family ipv4 autonomous-system 4453
Router(config-router-af)# network 10.0.0.0
Router(config-router-af) eigrp stub leak-map map1
```

# Additional References

## Related Documents

| Related Topic | Document Title |
|---|---|
| EIGRP message authentication | *EIGRP Message Authentication Configuration Example* |
| EIGRP commands | *Cisco IOS IP Routing: EIGRP Command Reference* |
| Protocol-independent features that work with EIGRP | *Configuring IP Routing Protocol-Independent Features* |
| Cisco IOS commands | *Cisco IOS Master Command List, All Releases* |
| EIGRP L2/L3 API and Tunable Metric for Mobile Adhoc Networks feature | *EIGRP L2/L3 API and Tunable Metric for Mobile Adhoc Networks* |
| EIGRP Technology White Papers | *Enhanced Interior Gateway Routing Protocol* |
| EIGRP FAQ | *EIGRP Frequently Asked Questions* |
| EIGRP Technology Support | *Enhanced Interior Gatweway Routing Protocol* |

# Standards

| Standard | Title |
|---|---|
| No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature. | — |

# MIBs

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

# RFCs

| RFC | Title |
|---|---|
| None | — |

# Technical Assistance

| Description | Link |
|---|---|
| The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password. | http://www.cisco.com/cisco/web/support/index.html |

# Feature Information for EIGRP

Table 3 lists the features in this module and provides links to specific configuration information.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

**Note**      Table 3 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

***Table 3        Feature Information for EIGRP Features***

| Feature Name | Releases | Feature Information |
|---|---|---|
| Enhanced Interior Gateway Routing Protocol (EIGRP) | 11.2(1) 12.2(33)SRA 15.0(1)M 12.2(33)SRE 12.2(33)XNE 15.0(1)S | EIGRP is an enhanced version of the IGRP developed by Cisco. EIGRP uses the same distance vector algorithm and distance information as IGRP. However, the convergence properties and the operating efficiency of EIGRP have improved substantially over IGRP, and IGRP is obsolete.<br><br>The following commands were introduced or modified by this feature:<br><br>**auto-summary** (EIGRP), **clear ip eigrp neighbors**, **default-information**, **default-metric** (EIGRP), **distance** (EIGRP), **eigrp log-neighbor-changes**, **eigrp log-neighbor-warnings**, **eigrp router-id**, **ip bandwidth-percent eigrp**, **ip hello-interval eigrp**, **ip hold-time eigrp**, **ip next-hop-self eigrp**, **ip split-horizon eigrp**, **ip summary-address eigrp**, **metric maximum-hops**, **metric weights** (EIGRP), **neighbor** (EIGRP), **network** (EIGRP), **offset-list** (EIGRP), **router eigrp**, **set metric** (EIGRP), **show ip eigrp accounting**, **show ip eigrp interfaces**, **show ip eigrp neighbors**, **show ip eigrp topology**, **show ip eigrp traffic**, **show ip eigrp vrf accounting**, **show ip eigrp vrf interfaces**, **show ip eigrp vrf neighbors**, **show ip eigrp vrf topology**, **show ip eigrp vrf traffic**, **timers active-time**, **traffic-share balanced**, **variance** (EIGRP). |

*Table 3* **Feature Information for EIGRP Features (continued)**

| Feature Name | Releases | Feature Information |
| --- | --- | --- |
| | | In Cisco IOS Release 15.0(1)M, 12.2(33)SRE, and 12.2(33)XNE, the following commands were introduced or modified: **address-family** (EIGRP), **af-interface**, **autonomous-system** (EIGRP), **auto-summary** (EIGRP), **bandwidth percent**, **clear eigrp address-family neighbors, clear ip eigrp neighbors, debug eigrp address-family neighbor, debug eigrp address-family notifications, default-information, default-metric** (EIGRP), **distance** (EIGRP), **eigrp event-log-size, eigrp log-neighbor- changes, eigrp log-neighbor-warnings, eigrp router-id, exit-address-family, exit-af-interface, exit-af-topology, hello-interval, hold-time, match extcommunity, metric maximum-hops, metric weights, next-hop-self, offset-list** (EIGRP), **passive-interface** (EIGRP), **router eigrp, show eigrp address-family accounting, show eigrp address-family events, show eigrp address-family interfaces, show eigrp address-family neighbors, show eigrp address-family timers, show eigrp address-family topology, show eigrp address-family traffic, show eigrp plugins, show eigrp protocols, show eigrp tech-support, show ip eigrp accounting, show ip eigrp events, show ip eigrp interfaces, show ip eigrp neighbors, show ip eigrp topology, show ip eigrp traffic, shutdown** (address-family), **split-horizon** (EIGRP), **summary-address** (EIGRP), **timers active-time**, **traffic-share balanced**, **variance** (EIGRP). In Cisco IOS Release 15.0(1)M, 12.2(33)SRE, and 12.2(33)XNE, the following commands were replaced: **clear ip eigrp vrf neighbors, eigrp interface, log-neighbor-warnings, show ip eigrp vrf accounting, show ip eigrp vrf interfaces, show ip eigrp vrf neighbors, show ip eigrp vrf topology, show ip eigrp vrf traffic.** |

**Table 3** *Feature Information for EIGRP Features (continued)*

| Feature Name | Releases | Feature Information |
|---|---|---|
| EIGRP Stub Routing | Cisco IOS XE 3.1.0SG 12.0(7)T 12.0(15)S 15.0(1)M 12.2(33)SRE 12.2(33)XNE 15.0(1)S | The EIGRP Stub Routing feature improves network stability, reduces resource utilization, and simplifies stub router configuration. Stub routing is commonly used in a hub-and-spoke network topology. In a hub-and-spoke network, one or more end (stub) networks are connected to a remote router (the spoke) that is connected to one or more distribution routers (the hub). The remote router is adjacent only to one or more distribution routers.<br><br>The following sections provide information about this feature:<br><br>• EIGRP Stub Routing, page 12<br>• EIGRP Stub Routing Leak Map Support, page 16<br>• Configuring EIGRP Stub Routing: Autonomous System Configuration, page 43<br>• Configuring EIGRP Stub Routing: Named Configuration, page 44<br>• Example: EIGRP Stub Routing: Autonomous System Configuration, page 59<br>• Example: EIGRP Stub Routing: Named Configuration, page 60<br><br>The following command was introduced by this feature: **eigrp stub**. |

*Table 3* **Feature Information for EIGRP Features (continued)**

| Feature Name | Releases | Feature Information |
|---|---|---|
| IP Enhanced IGRP Route Authentication | 11.3(1)<br>12.2(33)SRA<br>15.0(1)M<br>12.2(33)SRE<br>12.2(33)XNE<br>15.0(1)S | The IP Enhanced IGRP route authentication feature provides MD5 authentication of routing updates from the EIGRP routing protocol.<br><br>The following sections provide information about this feature:<br><br>• EIGRP Route Authentication, page 10<br>• Configuring EIGRP Route Authentication: Autonomous System Configuration, page 33<br>• Configuring EIGRP Route Authentication: Named Configuration, page 34<br>• Example: EIGRP Route Authentication: Autonomous System Configuration, page 55<br>• Example: EIGRP Route Authentication: Named Configuration, page 57<br><br>The following commands were introduced or modified by this feature: **accept-lifetime**, **ip authentication key-chain eigrp**, **ip authentication mode eigrp**, **key chain**, **key**, **key-string, send-lifetime**<br><br>In Cisco IOS Release 15.0(1)M, 12.2(33)SRE, and 12.2(33)XNE, the following commands were introduced or modified: **authentication mode** (EIGRP), **authentication key-chain** (EIGRP) |