Configuring EIGRP

This chapter describes how to configure the Enhanced Interior Gateway Routing Protocol (EIGRP) on the Cisco NX-OS device.

- About EIGRP, on page 1
- Prerequisites for EIGRP, on page 8
- Guidelines and Limitations for EIGRP, on page 8
- Default Settings, on page 9
- Configuring Basic EIGRP, on page 10
- Configuring Advanced EIGRP, on page 14
- Configuring Virtualization for EIGRP, on page 29
- Verifying the EIGRP Configuration, on page 30
- Monitoring EIGRP, on page 31
- Configuration Examples for EIGRP, on page 31
- Related Topics, on page 32
- Additional References, on page 32

About EIGRP

EIGRP combines the benefits of distance vector protocols with the features of link-state protocols. EIGRP sends out periodic Hello messages for neighbor discovery. Once EIGRP learns a new neighbor, it sends a one-time update of all the local EIGRP routes and route metrics. The receiving EIGRP router calculates the route distance based on the received metrics and the locally assigned cost of the link to that neighbor. After this initial full route table update, EIGRP sends incremental updates to only those neighbors affected by the route change. This process speeds convergence and minimizes the bandwidth used by EIGRP.

EIGRP Components

- Reliable Transport Protocol, on page 2
- Neighbor Discovery and Recovery, on page 2
- Diffusing Update Algorithm, on page 2

Reliable Transport Protocol

The Reliable Transport Protocol guarantees ordered delivery of EIGRP packets to all neighbors. (See Neighbor Discovery and Recovery, on page 2.) The Reliable Transport Protocol supports an intermixed transmission of multicast and unicast packets. The reliable transport can send multicast packets quickly when unacknowledged packets are pending. This provision helps to ensure that the convergence time remains low for various speed links. See the Configuring Advanced EIGRP, on page 14 section for details about modifying the default timers that control the multicast and unicast packet transmissions.

The Reliable Transport Protocol includes the following message types:

- Hello—Used for neighbor discovery and recovery. By default, EIGRP sends a periodic multicast Hello message on the local network at the configured hello interval. By default, the hello interval is 5 seconds.
- Acknowledgment—Verify reliable reception of Updates, Queries, and Replies.
- Updates—Send to affected neighbors when routing information changes. Updates include the route destination, address mask, and route metrics such as delay and bandwidth. The update information is stored in the EIGRP topology table.
- Queries and Replies—Sent as part of the Diffusing Update Algorithm used by EIGRP.

Neighbor Discovery and Recovery

EIGRP uses the Hello messages from the Reliable Transport Protocol to discover neighboring EIGRP routers on directly attached networks. EIGRP adds neighbors to the neighbor table. The information in the neighbor table includes the neighbor address, the interface it was learned on, and the hold time, which indicates how long EIGRP should wait before declaring a neighbor unreachable. By default, the hold time is three times the hello interval or 15 seconds.

EIGRP sends a series of Update messages to new neighbors to share the local EIGRP routing information. This route information is stored in the EIGRP topology table. After this initial transmission of the full EIGRP route information, EIGRP sends Update messages only when a routing change occurs. These Update messages contain only the new or changed information and are sent only to the neighbors affected by the change.

EIGRP also uses the Hello messages as a keepalive to its neighbors. As long as Hello messages are received, Cisco NX-OS can determine that a neighbor is alive and functioning.

Diffusing Update Algorithm

The Diffusing Update Algorithm (DUAL) calculates the routing information based on the destination networks in the topology table. The topology table includes the following information:

- IPv4 or IPv6 address/mask—The network address and network mask for this destination.
- Successors—The IP address and local interface connection for all feasible successors or neighbors that advertise a shorter distance to the destination than the current feasible distance.
- Feasibility distance (FD)—The lowest calculated distance to the destination. The feasibility distance is the sum of the advertised distance from a neighbor plus the cost of the link to that neighbor.

DUAL uses the distance metric to select efficient, loop-free paths. DUAL selects routes to insert into the Unicast Routing Information Base (RIB) based on feasible successors. When a topology change occurs, DUAL looks for feasible successors in the topology table. If there are feasible successors, DUAL selects the feasible successor with the lowest feasible distance and inserts that into the unicast RIB, avoiding unnecessary recomputation.

When there are no feasible successors but there are neighbors advertising the destination, DUAL transitions from the passive state to the active state and triggers a recomputation to determine a new successor or next-hop router to the destination. The amount of time required to recompute the route affects the convergence time. EIGRP sends Query messages to all neighbors, searching for feasible successors. Neighbors that have a feasible successor send a Reply message with that information. Neighbors that do not have feasible successors trigger a DUAL recomputation.

EIGRP Route Updates

When a topology change occurs, EIGRP sends an Update message with only the changed routing information to affected neighbors. This Update message includes the distance information to the new or updated network destination.

The distance information in EIGRP is represented as a composite of available route metrics, including bandwidth, delay, load utilization, and link reliability. Each metric has an associated weight that determines if the metric is included in the distance calculation. You can configure these metric weights. You can fine-tune link characteristics to achieve optimal paths, but we recommend that you use the default settings for most configurable metrics.

Internal Route Metrics

Internal routes are routes that occur between neighbors within the same EIGRP autonomous system. These routes have the following metrics:

- Next hop—The IP address of the next-hop router.
- Delay—The sum of the delays configured on the interfaces that make up the route to the destination network. The delay is configured in tens of microseconds.
- Bandwidth—The calculation from the lowest configured bandwidth on an interface that is part of the route to the destination.

Note Cisco recommends that you use the default bandwidth value. This bandwidth parameter is also used by EIGRP.

- MTU—The smallest maximum transmission unit value along the route to the destination.
- Hop count—The number of hops or routers that the route passes through to the destination. This metric is not directly used in the DUAL computation.
- Reliability—An indication of the reliability of the links to the destination.
- Load—An indication of how much traffic is on the links to the destination.

By default, EIGRP uses the bandwidth and delay metrics to calculate the distance to the destination. You can modify the metric weights to include the other metrics in the calculation.

Wide Metrics

EIGRP supports wide (64-bit) metrics to improve route selection on higher-speed interfaces or bundled interfaces. Routers supporting wide metrics can interoperate with routers that do not support wide metrics as follows:

- A router that supports wide metrics—Adds local wide metrics values to the received values and sends the information on.
- A router that does not support wide metrics—Sends any received metrics on without changing the values.

EIGRP uses the following equation to calculate path cost with wide metrics:

metric = [k1 x bandwidth + (k2 x bandwidth)/(256 – load) + k3 x delay + k6 x extended attributes] x [k5/(reliability + k4)]

Since the unicast RIB cannot support 64-bit metric values, EIGRP wide metrics uses the following equation with a RIB scaling factor to convert the 64-bit metric value to a 32-bit value:

RIB Metric = (Wide Metric / RIB scale value)

where the RIB scale value is a configurable parameter.

EIGRP wide metrics introduce the following two new metric values represented as k6 in the EIGRP metrics configuration:

- Jitter—Measured in microseconds and accumulated across all links in the route path.
- Energy—Measured in watts per kilobit and accumulated across all links in the route path.

EIGRP prefers a path with low or no jitter or energy metric values over a path with higher values.

Note EIGRP wide metrics are sent with a TLV version of 2. For more information, see the Enabling Wide Metrics, on page 25 section.

External Route Metrics

External routes are routes that occur between neighbors in different EIGRP autonomous systems. These routes have the following metrics:

- Next hop—The IP address of the next-hop router.
- Router ID—The router ID of the router that redistributed this route into EIGRP.
- AS number—The autonomous system number of the destination.
- Protocol ID—A code that represents the routing protocol that learned the destination route.
- Tag—An arbitrary tag that can be used for route maps.
- Metric—The route metric for this route from the external routing protocol.

EIGRP and the Unicast RIB

EIGRP adds all learned routes to the EIGRP topology table and the unicast RIB. When a topology change occurs, EIGRP uses these routes to search for a feasible successor. EIGRP also listens for notifications from the unicast RIB for changes in any routes redistributed to EIGRP from another routing protocol.

Advanced EIGRP

You can use the advanced features of EIGRP to optimize your EIGRP configuration.

Address Families

EIGRP supports both IPv4 and IPv6 address families. For backward compatibility, you can configure EIGRPv4 in route configuration mode or in IPv4 address family mode. You must configure EIGRP for IPv6 in address family mode.

Address family configuration mode includes the following EIGRP features:

- Authentication
- AS number
- Default route
- Metrics
- Distance
- Graceful restart
- Logging
- Load balancing
- Redistribution
- Router ID
- Stub router
- Timers

You cannot configure the same feature in more than one configuration mode. For example, if you configure the default metric in router configuration mode, you cannot configure the default metric in address family mode.

Authentication

You can configure authentication on EIGRP messages to prevent unauthorized or invalid routing updates in your network. EIGRP authentication supports MD5 authentication digest.

You can configure the EIGRP authentication per virtual routing and forwarding (VRF) instance or interface using keychain management for the authentication keys. Keychain management allows you to control changes to the authentication keys used by MD5 authentication digest. See the *Cisco Nexus 3400-S NX-OS Security Configuration Guide* for more details about creating keychains.

For MD5 authentication, you configure a password that is shared at the local router and all remote EIGRP neighbors. When an EIGRP message is created, Cisco NX-OS creates an MD5 one-way message digest based on the message itself and the encrypted password and sends this digest along with the EIGRP message. The receiving EIGRP neighbor validates the digest using the same encrypted password. If the message has not changed, the calculation is identical, and the EIGRP message is considered valid.

MD5 authentication also includes a sequence number with each EIGRP message that is used to ensure that no message is replayed in the network.

Stub Routers

You can use the EIGRP stub routing feature to improve network stability, reduce resource usage, and simplify stub router configuration. Stub routers connect to the EIGRP network through a remote router. See the Stub Routing section.

When using EIGRP stub routing, you need to configure the distribution and remote routers to use EIGRP and configure only the remote router as a stub. EIGRP stub routing does not automatically enable summarization on the distribution router. Usually, you need to configure summarization on the distribution routers.

Without EIGRP stub routing, even after the routes that are sent from the distribution router to the remote router have been filtered or summarized, a problem might occur. For example, if a route is lost somewhere in the corporate network, EIGRP could send a query to the distribution router. The distribution router could then send a query to the remote router even if routes are summarized. If a problem communicating over the WAN link between the distribution router and the remote router occurs, EIGRP could get stuck in an active condition and cause instability elsewhere in the network. EIGRP stub routing allows you to prevent queries to the remote router.

Route Summarization

You can configure a summary aggregate address for a specified interface. Route summarization simplifies route tables by replacing several more-specific addresses with an address that represents all the specific addresses. For example, you can replace 10.1.1.0/24, 10.1.2.0/24, and 10.1.3.0/24 with one summary address, 10.1.0.0/16.

If more specific routes are in the routing table, EIGRP advertises the summary address from the interface with a metric equal to the minimum metric of the more specific routes.

Note EIGRP does not support automatic route summarization.

Route Redistribution

You can use EIGRP to redistribute static routes, routes that are learned by other EIGRP autonomous systems, or routes from other protocols. You must configure a route map with the redistribution to control which routes are passed into EIGRP. A route map allows you to filter routes based on attributes such as the destination, origination protocol, route type, route tag, and so on. See Configuring Route Policy Manager.

You also configure the default metric that is used for all imported routes into EIGRP.

You use distribute lists to filter routes from routing updates. These filtered routes are applied to each interface with the **ip distribute-list eigrp** command.

Load Balancing

You can use load balancing to allow a router to distribute traffic over all the router network ports that are the same distance from the destination address. Load balancing increases the usage of network segments, which increases effective network bandwidth.

Cisco NX-OS supports the Equal Cost Multiple Paths (ECMP) feature with up to 16 equal-cost paths in the EIGRP route table and the unicast RIB. You can configure EIGRP to load balance traffic across some or all of those paths.

Note EIGRP in Cisco NX-OS does not support unequal cost load balancing.

Split Horizon

You can use split horizon to ensure that EIGRP never advertises a route out of the interface where it was learned.

Split horizon is a method that controls the sending of EIGRP update and query packets. When you enable split horizon on an interface, Cisco NX-OS does not send update and query packets for destinations that were learned from this interface. Controlling update and query packets in this manner reduces the possibility of routing loops.

Split horizon with poison reverse configures EIGRP to advertise a learned route as unreachable back through the interface from which EIGRP learned the route.

EIGRP uses split horizon or split horizon with poison reverse in the following scenarios:

- Exchanging topology tables for the first time between two routers in startup mode.
- Advertising a topology table change.
- Sending a Query message.

By default, the split horizon feature is enabled on all interfaces.

BFD

This feature supports bidirectional forwarding detection (BFD) for IPv4 and IPv6. BFD is a detection protocol designed to provide fast forwarding-path failure detection times. BFD provides subsecond failure detection between two adjacent devices and can be less CPU-intensive than protocol hello messages because some of the BFD load can be distributed onto the data plane on supported modules. See the *Cisco Nexus 3400-S NX-OS Interfaces Configuration Guide* for more information.

Virtualization Support

EIGRP supports virtual routing and forwarding instances (VRFs).

Graceful Restart and High Availability

Cisco NX-OS supports nonstop forwarding and graceful restart for EIGRP.

You can use nonstop forwarding for EIGRP to forward data packets along known routes in the FIB while the EIGRP routing protocol information is being restored following a failover. With nonstop forwarding (NSF), peer networking devices do not experience routing flaps. During failover, data traffic is forwarded through intelligent modules while the standby supervisor becomes active.

If a Cisco NX-OS system experiences a cold reboot, the device does not forward traffic to the system and removes the system from the network topology. In this scenario, EIGRP experiences a stateless restart, and all neighbors are removed. Cisco NX-OS applies the startup configuration, and EIGRP rediscovers the neighbors and shares the full EIGRP routing information again.

A dual-supervisor platform that runs Cisco NX-OS can experience a stateful supervisor switchover. Before the switchover occurs, EIGRP uses a graceful restart to announce that EIGRP will be unavailable for some

time. During a switchover, EIGRP uses nonstop forwarding to continue forwarding traffic based on the information in the FIB, and the system is not taken out of the network topology.

The graceful restart-capable router uses Hello messages to notify its neighbors that a graceful restart operation has started. When a graceful restart-aware router receives a notification from a graceful restart-capable neighbor that a graceful restart operation is in progress, both routers immediately exchange their topology tables. The graceful restart-aware router performs the following actions to assist the restarting router as follows:

- The router expires the EIGRP Hello hold timer to reduce the time interval set for Hello messages. This process allows the graceful restart-aware router to reply to the restarting router more quickly and reduces the amount of time required for the restarting router to rediscover neighbors and rebuild the topology table.
- The router starts the route-hold timer. This timer sets the period of time that the graceful restart-aware router will hold known routes for the restarting neighbor. The default time period is 240 seconds.
- The router notes in the peer list that the neighbor is restarting, maintains adjacency, and holds known routes for the restarting neighbor until the neighbor signals that it is ready for the graceful restart-aware router to send its topology table or the route-hold timer expires. If the route-hold timer expires on the graceful restart-aware router, the graceful restart-aware router discards held routes and treats the restarting router as a new router that joins the network and reestablishes adjacency.

After the switchover, Cisco NX-OS applies the running configuration, and EIGRP informs the neighbors that it is operational again.

Multiple EIGRP Instances

Cisco NX-OS supports multiple instances of the EIGRP protocol that run on the same system. Every instance uses the same system router ID. You can optionally configure a unique router ID for each instance. For the number of supported EIGRP instances, see the *Cisco Nexus 3400-S NX-OS Verified Scalability Guide*.

Prerequisites for EIGRP

EIGRP has the following prerequisites:

- You must enable EIGRP (see the Enabling the EIGRP Feature, on page 10 section).

Guidelines and Limitations for EIGRP

EIGRP has the following configuration guidelines and limitations:

- A metric configuration (either through the default-metric configuration option or through a route map) is required for redistribution from any other protocol, connected routes, or static routes (see Configuring Route Policy Manager).
- For graceful restart, an NSF-aware router must be up and completely converged with the network before it can assist an NSF-capable router in a graceful restart operation.
- For graceful restart, an NSF-aware router must be up and completely converged with the network before it can assist an NSF-capable router in a graceful restart operation.

- For graceful restart, neighboring devices participating in the graceful restart must be NSF-aware or NSF-capable.
- Cisco NX-OS EIGRP is compatible with EIGRP in the Cisco IOS software.
- Do not change the metric weights without a good reason. If you change the metric weights, you must apply the change to all EIGRP routers in the same autonomous system.
- A mix of standard metrics and wide metrics in an EIGRP network with interface speeds of 1 Gigabit or greater might result in suboptimal routing.
- Consider using stubs for larger networks.
- Avoid redistribution between different EIGRP autonomous systems because the EIGRP vector metric will not be preserved.
- The **no** {**ip** | **ipv6**} **next-hop-self** command does not guarantee reachability of the next hop.
- The {**ip** | **ipv6**} **passive-interface eigrp** command suppresses neighbors from forming.
- Cisco NX-OS does not support IGRP or connecting IGRP and EIGRP clouds.
- Autosummarization is disabled by default and cannot be enabled.
- Cisco NX-OS supports only IPv4 and IPv6.
- High availability is not supported with EIGRP aggressive timers.

Note If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Default Settings

The table lists the default settings for EIGRP parameters.

Table 1: Default Settings for EIGRP Parameters

Parameters	Default
Administrative distance	• Internal routes—90 • External routes—170
Bandwidth percent	50 percent

Parameters	Default
Default metric for redistributed routes	• Bandwidth—100000 Kb/s • Delay—100 (10-microsecond units) • Reliability—255 • Loading—1 • MTU—1500
EIGRP feature	Disabled
Hello interval	5 seconds
Hold time	15 seconds
Equal-cost paths	8
Metric weights	1 0 1 0 0 0
Next-hop address advertised	IP address of local interface
NSF convergence time	120
NSF route-hold time	240
NSF signal time	20
Redistribution	Disabled
Split horizon	Enabled

Configuring Basic EIGRP

Enabling the EIGRP Feature

You must enable EIGRP before you can configure EIGRP.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	[**no**] **feature eigrp**	Enables the EIGRP feature.

	Command or Action	Purpose
	Example: `switch(config)# feature eigrp`	The **no** option disables the EIGRP feature and removes all associated configurations.
Step 3	(Optional) **show feature** **Example:** `switch(config)# show feature`	Displays information about enabled features.
Step 4	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	Saves this configuration change.

Creating an EIGRP Instance

You can create an EIGRP instance and associate an interface with that instance. You assign a unique autonomous system number for this EIGRP process (see the Autonomous Systems section). Routes are not advertised or accepted from other autonomous systems unless you enable route redistribution.

Before you begin

You must enable EIGRP (see the Enabling the EIGRP Feature, on page 10 section).

EIGRP must be able to obtain a router ID (for example, a configured loopback address), or you must configure the router ID option.

If you configure an instance tag that does not qualify as an AS number, you must configure the AS number explicitly or this EIGRP instance remains in the shutdown state. For IPv6, this number must be configured under the address family.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	[**no**] **router eigrp** *instance-tag*
	Example:
	`switch(config)# router eigrp Test1`
	`switch(config-router)#`
		Creates a new EIGRP process with the configured instance tag. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.
		If you configure an *instance-tag* that does not qualify as an AS number, you must use the **autonomous-system** command to configure the AS number explicitly, or this EIGRP instance will remain in the shutdown state.

	Command or Action	Purpose
		Use the **no** option with this command to delete the EIGRP process and all associated configuration. **Note** You should also remove any EIGRP commands configured in interface mode if you remove the EIGRP process.
Step 3	(Optional) **autonomous-system** *as-number* **Example:** `switch(config-router)# autonomous-system 33`	Configures a unique AS number for this EIGRP instance. The range is from 1 to 65535.
Step 4	(Optional) **log-adjacency-changes** **Example:** `switch(config-router)# log-adjacency-changes`	Generates a system message whenever an adjacency changes state. This command is enabled by default.
Step 5	(Optional) **log-neighbor-warnings** [*seconds*] **Example:** `switch(config-router)# log-neighbor-warnings`	Generates a system message whenever a neighbor warning occurs. You can configure the time between warning messages, from 1 to 65535, in seconds. The default is 10 seconds. This command is enabled by default.

Step 6	Required: **interface** *interface-type slot*/*port*
	Example:
	`switch(config-router)# interface ethernet 1/2`
	`switch(config-if)#`
		Enters interface configuration mode. Use **?** to determine the slot and port ranges.

Step 7	Required: {**ip** | **ipv6**} **router eigrp** *instance-tag* **Example:** `switch(config-if)# ip router eigrp Test1`	Associates this interface with the configured EIGRP process. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.
Step 8	(Optional) **show** {**ip** | **ipv6**} **eigrp interfaces** **Example:** `switch(config-if)# show ip eigrp interfaces`	Displays information about EIGRP interfaces.
Step 9	(Optional) **copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	Saves this configuration change.

Example

Note You should also remove any EIGRP commands configured in interface mode if you remove the EIGRP process.

This example shows how to create an EIGRP process and configure an interface for EIGRP:

```
switch# configure terminal
switch(config)# router eigrp Test1
switch(config-router)# interface ethernet 1/2
switch(config-if)# ip router eigrp Test1
switch(config-if)# no shutdown
switch(config-if)# copy running-config startup-config
```

For more information about other EIGRP parameters, see the Configuring Advanced EIGRP, on page 14 section.

Restarting an EIGRP Instance

You can restart an EIGRP instance. This action clears all neighbors for the instance.

To restart and EIGRP instance and remove all associated neighbors, use the following commands in global configuration mode:

Procedure

	Command or Action	Purpose
Step 1	(Optional) **flush-routes** **Example:** `switch(config)# flush-routes`	Flushes all EIGRP routes in the unicast RIB when this EIGRP instance restarts.
Step 2	**restart eigrp** *instance-tag* **Example:** `switch(config)# restart eigrp Test1`	Restarts the EIGRP instance and removes all neighbors. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.

Shutting Down an EIGRP Instance

You can gracefully shut down an EIGRP instance. This action removes all routes and adjacencies but preserves the EIGRP configuration.

To disable an EIGRP instance, use the following command in router configuration mode:

Procedure

	Command or Action	Purpose
Step 1	**shutdown** **Example:** `switch(config-router)# shutdown`	Disables this instance of EIGRP. The EIGRP router configuration remains.

Configuring a Passive Interface for EIGRP

You can configure a passive interface for EIGRP. A passive interface does not participate in EIGRP adjacency, but the network address for the interface remains in the EIGRP topology table.

To configure a passive interface for EIGRP, use the following command in interface configuration mode:

Procedure

	Command or Action	Purpose
Step 1	{**ip** | **ipv6**} **passive-interface eigrp** *instance-tag* **Example:** `switch(config-if)# ip passive-interface eigrp tag10`	Suppresses EIGRP hellos, which prevents neighbors from forming and sending routing updates on an EIGRP interface. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.

Shutting Down an EIGRP Instance

You can gracefully shut down an EIGRP instance. This action removes all routes and adjacencies but preserves the EIGRP configuration.

To disable an EIGRP instance, use the following command in router configuration mode:

Procedure

	Command or Action	Purpose
Step 1	**shutdown** **Example:** `switch(config-router)# shutdown`	Disables this instance of EIGRP. The EIGRP router configuration remains.

Configuring Advanced EIGRP

Configuring Authentication in EIGRP

You can configure authentication between neighbors for EIGRP. See the Authentication, on page 5 section.

You can configure EIGRP authentication for the EIGRP process or for individual interfaces. The interface EIGRP authentication configuration overrides the EIGRP process-level authentication configuration.

Before you begin

Enable EIGRP. See the Enabling the EIGRP Feature, on page 10 section.

Ensure that all neighbors for an EIGRP process share the same authentication configuration, including the shared authentication key.

Create the keychain for this authentication configuration. For more information, see the *Cisco Nexus 3400-S NX-OS Security Configuration Guide*.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**router eigrp** *instance-tag*
	Example:
	`switch(config)# router eigrp Test1`
	`switch(config-router)#`
		Creates a new EIGRP process with the configured instance tag. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.
		If you configure an *instance-tag* that does not qualify as an autonomous system number, you must use the **autonomous-system** command to configure the AS number explicitly. If you do not, this EIGRP instance remains in the shutdown state.

Step 3	**address-family** {**ipv4** | **ipv6**} **unicast**
	Example:
	`switch(config-router)# address-family ipv4 unicast`
	`switch(config-router-af)#`
		Enters the address-family configuration mode. This command is optional for IPv4.

Step 4	**authentication key-chain** *key-chain* **Example:** `switch(config-router-af)# authentication key-chain routeKeys`	Associates a keychain with this EIGRP process for this VRF. The keychain can be any case-sensitive, alphanumeric string up to 20 characters.
Step 5	**authentication mode md5** **Example:** `switch(config-router-af)# authentication mode md5`	Configures MD5 message digest authentication mode for this VRF.
Step 6	**interface** *interface-type slot/port* **Example:**	Enters interface configuration mode. Use **?** to find the supported interfaces.

	Command or Action	Purpose
	`switch(config-router-af) interface ethernet 1/2` `switch(config-if)#`	
Step 7	{**ip** | **ipv6**} **router eigrp** *instance-tag* **Example:** `switch(config-if)# ip router eigrp Test1`	Associates this interface with the configured EIGRP process. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.
Step 8	{**ip** | **ipv6**} **authentication key-chain eigrp** *instance-tag keychain* **Example:** `switch(config-if)# ip authentication key-chain eigrp Test1 routeKeys`	Associates a keychain with this EIGRP process for this interface. This configuration overrides the authentication configuration set in the router VRF mode. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.
Step 9	{**ip** | **ipv6**} **authentication mode eigrp** *instance-tag* **md5** **Example:** `switch(config-if)# ip authentication mode eigrp Test1 md5`	Configures the MD5 message digest authentication mode for this interface. This configuration overrides the authentication configuration set in the router VRF mode. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.
Step 10	(Optional) **copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	Saves this configuration change.

Example

This example shows how to configure MD5 message digest authentication for EIGRP over Ethernet interface 1/2:

```
switch# configure terminal
switch(config)# router eigrp Test1
switch(config-router)# exit
switch(config)# interface ethernet 1/2
switch(config-if)# ip router eigrp Test1
switch(config-if)# ip authentication key-chain eigrp Test1 routeKeys
switch(config-if)# ip authentication mode eigrp Test1 md5
switch(config-if)# copy running-config startup-config
```

Configuring EIGRP Stub Routing

You can configure a router for EIGRP stub routing.

To configure a router for EIGRP stub routing, use the following command in address-family configuration mode:

Procedure

	Command or Action	Purpose
Step 1	**stub** [**direct** | **receive-only** | **redistributed** [**direct**] **leak-map** *map-name*] **Example:** `switch(config-router-af)# eigrp stub redistributed`	Configures a remote router as an EIGRP stub router. The map name can be any case-sensitive, alphanumeric string up to 20 characters.
Step 2	(Optional) **show ip eigrp neighbor detail** **Example:** `switch(config-router-af)# show ip eigrp neighbor detail`	Verifies that the router has been configured as a stub router.

Example

This example shows how to configure a stub router to advertise directly connected and redistributed routes:

```
switch# configure terminal
switch(config)# router eigrp Test1
switch(config-router)# address-family ipv6 unicast
switch(config-router-af)# stub direct redistributed
switch(config-router-af)# copy running-config startup-config
```

Use the **show ip eigrp neighbor detail** command to verify that a router has been configured as a stub router. The last line of the output shows the stub status of the remote or spoke router.

This example shows the output from the **show ip eigrp neighbor detail** command:

```
Router# show ip eigrp neighbor detail
IP-EIGRP neighbors for process 201
H  Address            Interface          Hold Uptime   SRTT   RTO  Q  Seq Type
                                          (sec)         (ms)       Cnt Num
0  10.1.1.2           Se3/1                11 00:00:59    1  4500  0  7
  Version 12.1/1.2,   Retrans: 2,  Retries: 0
 Stub Peer Advertising ( CONNECTED SUMMARY ) Routes
```

Configuring a Summary Address for EIGRP

You can configure a summary aggregate address for a specified interface. If any more specific routes are in the routing table, EIGRP advertises the summary address out the interface with a metric equal to the minimum of all more specific routes. See the Route Summarization, on page 6 section.

To configure a summary aggregate address, use the following command in interface configuration mode:

Procedure

	Command or Action	Purpose
Step 1	{**ip** | **ipv6**} **summary-address eigrp** *instance-tag ip-prefix/length* [*distance* | **leak-map** *map-name*] **Example:** `switch(config-if)# ip summary-address eigrp Test1 192.0.2.0/8`	Configures a summary aggregate address as either an IP address and network mask or an IP prefix/length. The instance tag and map name can be any case-sensitive, alphanumeric string up to 20 characters. You can optionally configure the administrative distance for this aggregate address. The default administrative distance is 5 for aggregate addresses.

Example

This example shows how to cause EIGRP to summarize network 192.0.2.0 out Ethernet 1/2 only:

```
switch# configure terminal
switch(config)# interface ethernet 1/2
switch(config-if) ip summary eigrp Test1 192.0.2.0 255.255.255.0
```

Redistributing Routes into EIGRP

You can redistribute routes in EIGRP from other routing protocols.

Before you begin

Enable EIGRP. See Enabling the EIGRP Feature, on page 10.

Configure the metric (either through the default-metric configuration option or through a route map) for routes that are redistributed from any other protocol.

Create a route map to control the types of routes that are redistributed into EIGRP. See Configuring Route Policy Manager.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**router eigrp** *instance-tag*
	Example:
	`switch(config)# router eigrp Test1`
	`switch(config-router)#`
		Creates a new EIGRP process with the configured instance tag. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.

	Command or Action	Purpose
		If you configure an *instance-tag* that does not qualify as an autonomous system number, you must use the **autonomous-system** command to configure the autonomous system number explicitly. Otherwise, this EIGRP instance remains in the shutdown state.
Step 3	**address-family** {**ipv4** | **ipv6**} **unicast** **Example:** `switch(config-router)# address-family ipv4 unicast switch(config-router-af)#`	Enters the address-family configuration mode. This command is optional for IPv4.
Step 4	**redistribute** {**bgp** *as* | {**eigrp** | **isis** | **ospf** | **ospfv3** | **rip**} *instance-tag* | **direct** | **static**} **route-map** *map-name* **Example:** `switch(config-router-af)# redistribute bgp 100 route-map BGPFilter`	Injects routes from one routing domain into EIGRP. The instance tag and map name can be any case-sensitive, alphanumeric string up to 20 characters.
Step 5	**default-metric** *bandwidth delay reliability loading mtu* **Example:** `switch(config-router-af)# default-metric 500000 30 200 1 1500`	Sets the metrics that are assigned to routes learned through route redistribution. The default values are as follows: • bandwidth—100000 Kbps • delay—100 (10 microsecond units) • reliability—255 • loading—1 • MTU—1492
Step 6	(Optional) **show** {**ip** | **ipv6**} **eigrp route-map statistics redistribute** **Example:** `switch(config-router-af)# show ip eigrp route-map statistics redistribute bgp`	Displays information about EIGRP route map statistics.
Step 7	(Optional) **copy running-config startup-config** **Example:** `switch(config-router-af)# copy running-config startup-config`	Saves this configuration change.

Example

The following example shows how to redistribute BGP into EIGRP for IPv4:

```
switch# configure terminal
switch(config)# router eigrp Test1
switch(config-router)# redistribute bgp 100 route-map BGPFilter
switch(config-router)# default-metric 500000 30 200 1 1500
switch(config-router)# copy running-config startup-config
```

Limiting the Number of Redistributed Routes

Route redistribution can add many routes to the EIGRP route table. You can configure a maximum limit to the number of routes accepted from external protocols. EIGRP provides the following options to configure redistributed route limits:

- Fixed limit—Logs a message when EIGRP reaches the configured maximum. EIGRP does not accept any more redistributed routes. You can optionally configure a threshold percentage of the maximum where EIGRP logs a warning when that threshold is passed.
- Warning only—Logs a warning only when EIGRP reaches the maximum. EIGRP continues to accept redistributed routes.
- Withdraw—Starts the timeout period when EIGRP reaches the maximum. After the timeout period, EIGRP requests all redistributed routes if the current number of redistributed routes is less than the maximum limit. If the current number of redistributed routes is at the maximum limit, EIGRP withdraws all redistributed routes. You must clear this condition before EIGRP accepts more redistributed routes. You can optionally configure the timeout period.

Before you begin

You must enable EIGRP (see the Enabling the EIGRP Feature, on page 10 section).

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**router eigrp** *instance-tag*
	Example:
	`switch(config)# router eigrp Test1`
	`switch(config-router)#`
		Creates a new EIGRP instance with the configured instance tag.

Step 3	**redistribute** {**bgp** *id* | **direct** | **eigrp** *id* | **isis** *id* | **ospf** *id* | **rip** *id* | **static**} **route-map** *map-name* **Example:** `switch(config-router)# redistribute bgp route-map FilterExternalBGP`	Redistributes the selected protocol into EIGRP through the configured route map.

	Command or Action	Purpose
Step 4	**redistribute maximum-prefix** *max* [*threshold*] [**warning-only** | **withdraw** [*num-retries timeout*]] **Example:** `switch(config-router)# redistribute maximum-prefix 1000 75 warning-only`	Specifies a maximum number of prefixes that EIGRP distributes. The range is from 1 to 65535. Optionally specifies the following: • *threshold*—Percentage of maximum prefixes that triggers a warning message. • **warning-only**—Logs a warning message when the maximum number of prefixes is exceeded. • **withdraw**—Withdraws all redistributed routes. Optionally tries to retrieve the redistributed routes. The *num-retries* range is from 1 to 12. The *timeout* is from 60 to 600 seconds. The default is 300 seconds. Use the **clear ip eigrp redistribution** command if all routes are withdrawn.
Step 5	(Optional) **show running-config eigrp** **Example:** `switch(config-router)# show running-config eigrp`	Displays the EIGRP configuration.
Step 6	(Optional) **copy running-config startup-config** **Example:** `switch(config-router)# copy running-config startup-config`	Saves this configuration change.

Example

This example shows how to limit the number of redistributed routes into EIGRP:

```
switch# configure terminal
switch(config)# router eigrp Test1
switch(config-router)# redistribute bgp route-map FilterExternalBGP
switch(config-router)# redistribute maximum-prefix 1000 75
```

Configuring Load Balancing in EIGRP

You can configure load balancing in EIGRP. You can configure the number of Equal Cost Multiple Path (ECMP) routes using the **maximum-paths** option.

Before you begin

You must enable EIGRP (see the Enabling the EIGRP Feature, on page 10 section).

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**router eigrp** *instance-tag*
	Example:
	`switch(config)# router eigrp Test1`
	`switch(config-router)#`
		Creates a new EIGRP process with the configured instance tag. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.
		If you configure an *instance-tag* that does not qualify as an AS number, you must use the **autonomous-system** command to configure the AS number explicitly or this EIGRP instance will remain in the shutdown state.

Step 3	**address-family** {**ipv4** | **ipv6**} **unicast**
	Example:
	`switch(config-router)# address-family ipv4`
	`unicast`
	`switch(config-router-af)#`
		Enters the address-family configuration mode. This command is optional for IPv4.

Step 4	**maximum-paths** *num-paths* **Example:** `switch(config-router-af)# maximum-paths 5`	Sets the number of equal cost paths that EIGRP accepts in the route table. The range is from 1 to 32. The default is 8.

Step 5	(Optional) **copy running-config startup-config**
	Example:
	`switch(config-router-af)# copy running-config`
	`startup-config`
		Saves this configuration change.

Example

This example shows how to configure equal cost load balancing for EIGRP over IPv4 with a maximum of six equal cost paths:

```
switch# configure terminal
switch(config)# router eigrp Test1
switch(config-router)# maximum-paths 6
switch(config-router)# copy running-config startup-config
```

Configuring Graceful Restart for EIGRP

You can configure graceful restart or nonstop forwarding for EIGRP.

Note Graceful restart is enabled by default.

Before you begin

You must enable EIGRP (see the Enabling the EIGRP Feature, on page 10 section).

An NSF-aware router must be up and completely converged with the network before it can assist an NSF-capable router in a graceful restart operation.

Neighboring devices participating in the graceful restart must be NSF aware or NSF capable.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `switch# configure terminal`	Enters global configuration mode.

Step 2	**router eigrp** *instance-tag*
	Example:
	`switch(config)# router eigrp Test1`
	`switch(config-router)#`
		Creates a new EIGRP process with the configured instance tag. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.
		If you configure an *instance-tag* that does not qualify as an AS number, you must use the **autonomous-system** command to configure the AS number explicitly or this EIGRP instance remains in the shutdown state.

Step 3	**address-family** {**ipv4** | **ipv6**} **unicast**
	Example:
	`switch(config-router)# address-family ipv4`
	`unicast`
	`switch(config-router-af)#`
		Enters the address-family configuration mode. This command is optional for IPv4.

Step 4	**graceful-restart**
	Example:
	`switch(config-router-af)#`
	`graceful-restart`
		Enables graceful restart. This feature is enabled by default.

	Command or Action	Purpose
Step 5	**timers nsf converge** *seconds* **Example:** `switch(config-router-af)# timers nsf converge 100`	Sets the time limit for the convergence after a switchover. The range is from 60 to 180 seconds. The default is 120.
Step 6	**timers nsf route-hold** *seconds* **Example:** `switch(config-router-af)# timers nsf route-hold 200`	Sets the hold time for routes learned from the graceful restart-aware peer. The range is from 20 to 300 seconds. The default is 240.
Step 7	**timers nsf signal** *seconds* **Example:** `switch(config-router-af)# timers nsf signal 15`	Sets the time limit for signaling a graceful restart. The range is from 10 to 30 seconds. The default is 20.
Step 8	(Optional) **copy running-config startup-config** **Example:** `switch(config-router-af)# copy running-config startup-config`	Saves this configuration change.

Example

This example shows how to configure graceful restart for EIGRP over IPv6 using the default timer values:

```
switch# configure terminal
switch(config)# router eigrp Test1
switch(config-router)# address-family ipv6 unicast
switch(config-router-af)# graceful-restart
switch(config-router-af)# copy running-config startup-config
```

Adjusting the Interval Between Hello Packets and the Hold Time

You can adjust the interval between Hello messages and the hold time.

By default, Hello messages are sent every 5 seconds. The hold time is advertised in Hello messages and indicates to neighbors the length of time that they should consider the sender valid. The default hold time is three times the hello interval, or 15 seconds.

On very congested and large networks, the default hold time might not be sufficient time for all routers to receive hello packets from their neighbors. In this case, you might want to increase the hold time. To change the hold time, use the step 2 command in interface configuration mode:

Procedure

	Command or Action	Purpose
Step 1	{**ip** | **ipv6**} **hello-interval eigrp** *instance-tag seconds* **Example:** `switch(config-if)# ip hello-interval eigrp Test1 30`	Configures the hello interval for an EIGRP routing process. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters. The range is from 1 to 65535 seconds. The default is 5.
Step 2	{**ip** | **ipv6**} **hold-time eigrp** *instance-tag seconds* **Example:** `switch(config-if)# ipv6 hold-time eigrp Test1 30`	Configures the hold time for an EIGRP routing process. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters. The range is from 1 to 65535 seconds.

Example

Use the **show ip eigrp interface detail** command to verify the timer configuration.

Disabling Split Horizon

You can use split horizon to block route information from being advertised by a router out of any interface from which that information originated. Split horizon usually optimizes communications among multiple routing devices, particularly when links are broken.

By default, split horizon is enabled on all interfaces.

To disable split horizon, use the following command in interface configuration mode:

Procedure

	Command or Action	Purpose
Step 1	**no** {**ip** | **ipv6**} **split-horizon eigrp** *instance-tag* **Example:** `switch(config-if)# no ip split horizon eigrp Test1`	Disables split horizon.

Enabling Wide Metrics

To enable wide metrics and optionally configure a scaling factor for the RIB, use the following commands in router or address family configuration mode:

Procedure

	Command or Action	Purpose
Step 1	**metrics version 64bit** **Example:** `switch(config-router)# metrics version 64bit`	Enables 64-bit metric values.
Step 2	(Optional) **metrics rib-scale** *value* **Example:** `switch(config-router)#`	Configures the scaling factor used to convert the 64-bit metric values to 32 bit in the RIB. The range is from 1 to 255. The default value is 128.

Tuning EIGRP

You can configure optional parameters to tune EIGRP for your network.

You can configure the following optional parameters in address-family configuration mode:

Procedure

	Command or Action	Purpose
Step 1	**default-information originate** [**always** | **route-map** *map-name*] **Example:** `switch(config-router-af)# default-information originate always`	Originates or accepts the default route with prefix 0.0.0.0/0. When a route-map is supplied, the default route is originated only when the route map yields a true condition. The route-map name can be any case-sensitive, alphanumeric string up to 20 characters.
Step 2	**distance** *internal external* **Example:** `switch(config-router-af)# distance 25 100`	Configures the administrative distance for this EIGRP process. The range is from 1 to 255. The *internal* value sets the distance for routes learned from within the same autonomous system (the default value is 90). The *external* value sets the distance for routes learned from an external autonomous system (the default value is 170).
Step 3	**metric max-hops** *hop-count* **Example:** `switch(config-router-af)# metric max-hops 70`	Sets the maximum allowed hops for an advertised route. Routes over this maximum are advertised as unreachable. The range is from 1 to 255. The default is 100.
Step 4	**metric weights** *tos k1 k2 k3 k4 k5 k6* **Example:** `switch(config-router-af)# metric weights 0 1 3 2 1 0`	Adjusts the EIGRP metric or K value. EIGRP uses the following formula to determine the total metric to the network:

	Command or Action	Purpose
		metric = [k1 x bandwidth + (k2 x bandwidth)/(256 – load) + k3 x delay + k6 x extended attributes] * [k5/(reliability + k4)] Default values and ranges are as follows: • TOS—0. The range is from 0 to 8. • k1—1. The range is from 0 to 255. • k2—0. The range is from 0 to 255. • k3—1. The range is from 0 to 255. • k4—0. The range is from 0 to 255. • k5—0. The range is from 0 to 255. • k6—0. The range is from 0 to 255.
Step 5	**nsf await-redist-proto-convergence** **Example:** `switch(config-router-af)# nsf await-redist-proto-convergence`	Causes EIGRP to wait for the convergence of redistributed protocols before installing its own routes in the Routing Information Base (RIB) during nonstop forwarding (NSF). This command is useful in switchover scenarios when NSF is in progress and you want EIGRP to wait for BGP to converge and install its routes. It prevents EIGRP from installing transient routes and modifying the Forwarding Information Base (FIB) entries before BGP converges and EIGRP finds an alternate path to a destination. **Note** If you use this command when mutual redistribution is configured between EIGRP and BGP (for example, in a PE-CE environment), some traffic loss might occur because the provider-edge (PE) router will not install EIGRP routes into the RIB until BGP routes are available. This behavior delays the routes that the customer-edge (CE) router learns from EIGRP and advertises to the peer PE router.
Step 6	**timers active-time** {*time-limit* | **disabled**} **Example:** `switch(config-router-af)# timers active-time 200`	Sets the time the router waits in minutes (after sending a query) before declaring the route to be stuck in the active (SIA) state. The range is from 1 to 65535. The default is 3.
Step 7	(Optional) {**ip** | **ipv6**} **bandwidth eigrp** *instance-tag bandwidth*	Configures the bandwidth metric for EIGRP on an interface. The instance tag can be any

	Command or Action	Purpose
	Example: `switch(config-if)# ip bandwidth eigrp Test1 30000`	case-sensitive, alphanumeric string up to 20 characters. The bandwidth range is from 1 to 2,560,000,000 Kbps.
Step 8	{**ip** | **ipv6**} **bandwidth-percent eigrp** *instance-tag percent* **Example:** `switch(config-if)# ip bandwidth-percent eigrp Test1 30`	Configures the percentage of bandwidth that EIGRP might use on an interface. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters. The percent range is from 0 to 100. The default is 50.
Step 9	[**no**] {**ip** | **ipv6**} **delay eigrp** *instance-tag delay* **Example:** `switch(config-if)# ip delay eigrp Test1 100`	Configures the delay metric for EIGRP on an interface. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters. The delay range is from 1 to 16777215 (in tens of microseconds).
Step 10	{**ip** | **ipv6**} **distribute-list eigrp** *instance-tag* {**prefix-list** *name* | **route-map** *map-name*} {**in** | **out**} **Example:** `switch(config-if)# ip distribute-list eigrp Test1 route-map EigrpTest in`	Configures the route filtering policy for EIGRP on this interface. The instance tag, prefix list name, and route-map name can be any case-sensitive, alphanumeric string up to 20 characters.
Step 11	[**no**] {**ip** | **ipv6**} **next-hop-self eigrp** *instance-tag* **Example:** `switch(config-if)# ipv6 next-hop-self eigrp Test1`	Configures EIGRP to use the received next-hop address rather than the address for this interface. The default is to use the IP address of this interface for the next-hop address. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.
Step 12	{**ip** | **ipv6**} **offset-list eigrp** *instance-tag* {**prefix-list** *name* | **route-map** *map-name*} {**in** | **out**} *offset* **Example:** `switch(config-if)# ip offset-list eigrp Test1 prefix-list EigrpList in`	Adds an offset to incoming and outgoing metrics to routes learned by EIGRP. The instance tag, prefix list name, and route-map name can be any case-sensitive, alphanumeric string up to 20 characters.
Step 13	{**ip** | **ipv6**} **passive-interface eigrp** *instance-tag* **Example:** `switch(config-if)# ip passive-interface eigrp Test1`	Suppresses EIGRP hellos, which prevents neighbors from forming and sending routing updates on an EIGRP interface. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.

Configuring Virtualization for EIGRP

You can create multiple VRFs and use the same or multiple EIGRP processes in each VRF. You assign an interface to a VRF.

Note Configure all other parameters for an interface after you configure the VRF for an interface. Configuring a VRF for an interface deletes all other configuration for that interface.

Before you begin

You must enable EIGRP (see the Enabling the EIGRP Feature, on page 10 section).

Create the VRFs.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`	

Step 2	**vrf context** *vrf-name*
	Example:
	`switch(config)# vrf context RemoteOfficeVRF`
	`switch(config-vrf)#`
		Creates a new VRF and enters VRF configuration mode. The VRF name can be any case-sensitive, alphanumeric string up to 20 characters.

Step 3	**router eigrp** *instance-tag*
	Example:
	`switch(config-vrf)# router eigrp Test1`
	`switch(config-router)#`
		Creates a new EIGRP process with the configured instance tag. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.
		If you configure an *instance-tag* that does not qualify as an AS number, you must use the **autonomous-system** command to configure the AS number explicitly or this EIGRP instance remains in the shutdown state.

Step 4	**interface ethernet** *slot//port*
	Example:
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		Enters interface configuration mode. Use **?** to find the slot and port ranges.

	Command or Action	Purpose
Step 5	**vrf member** *vrf-name* **Example:** `switch(config-if)# vrf member RemoteOfficeVRF`	Adds this interface to a VRF. The VRF name can be any case-sensitive, alphanumeric string up to 20 characters.
Step 6	{**ip** | **ipv6**} **router eigrp** *instance-tag* **Example:** `switch(config-if)# ip router eigrp Test1`	Adds this interface to the EIGRP process. The instance tag can be any case-sensitive, alphanumeric string up to 20 characters.
Step 7	**copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	Saves this configuration change.

Example

This example shows how to create a VRF and add an interface to the VRF:

```
switch# configure terminal
switch(config)# vrf context NewVRF
switch(config-vrf)# router eigrp Test1
switch(config-router)# interface ethernet 1/2
switch(config-if)# ip router eigrp Test1
switch(config-if)# vrf member NewVRF
switch(config-if)# copy running-config startup-config
```

Verifying the EIGRP Configuration

To display the EIGRP configuration information, perform one of the following tasks:

Command	Purpose
show {**ip** | **ipv6**} **eigrp** [*instance-tag*]	Displays a summary of the configured EIGRP processes.
show {**ip** | **ipv6**} **eigrp** [*instance-tag*] **interfaces** [*type number*] [**brief**] [**detail**]	Displays information about all configured EIGRP interfaces.
show {**ip** | **ipv6**} **eigrp** *instance-tag* **neighbors** [*type number*] [**detail**]	Displays information about all the EIGRP neighbors. Use this command to verify the EIGRP neighbor configuration.
show {**ip** | **ipv6**} **eigrp** [*instance-tag*] **route** [*ip-prefix/length*] [**active**] [**all-links**] [**detail-links**] [**pending**] [**summary**] [**zero-successors**] [**vrf** *vrf-name*]	Displays information about all the EIGRP routes.

Command	Purpose
show {**ip** | **ipv6**} **eigrp** [*instance-tag*] **topology** [*ip-prefix/length*] [**active**] [**all-links**] [**detail-links**] [**pending**] [**summary**] [**zero-successors**] [**vrf** *vrf-name*]	Displays information about the EIGRP topology table.
show running-configuration eigrp	Displays the current running EIGRP configuration.

Monitoring EIGRP

To display EIGRP statistics, use the following commands:

Command	Purpose
show { ip | ipv6 } eigrp [instance-tag] accounting [vrf vrf-name]	Displays accounting statistics for EIGRP.
show { ip | ipv6 } eigrp [instance-tag] route-map statistics redistribute	Displays redistribution statistics for EIGRP.
show { ip | ipv6 } eigrp [instance-tag] traffic [vrf vrf-name]	Displays traffic statistics for EIGRP.

Configuration Examples for EIGRP

This example shows how to configure EIGRP:

```
feature eigrp
interface ethernet 1/2
 ip address 192.0.2.55/24
 ip router eigrp Test1
  no shutdown
router eigrp Test1
  router-id 192.0.2.1
```

The following example shows how to use a route map with the **distribute-list** command to filter routes that are dynamically received from (or advertised to) EIGRP peers. The example configures a route map to match an EIGRP external protocol metric route with an allowable deviation of 100, a source protocol of BGP, and an autonomous system number of 45000. When the two match clauses are true, the tag value of the destination routing protocol is set to 5. The route map is used to distribute incoming packets for an EIGRP process.

```
switch(config)# route-map metric-range
switch(config-route-map)# match metric external 500 +- 100
switch(config-route-map)# match source-protocol bgp 45000
switch(config-route-map)# set tag 5
switch(config-route-map)# exit
switch(config)# router eigrp 1
switch(config-router)# exit
switch(config)# interface ethernet 1/2
switch(config-if)# ip address 172.16.0.0
switch(config-if)# ip router eigrp 1
switch(config-if)# ip distribute-list eigrp 1 route-map metric-range in
```

The following example shows how to use a route map with the redistribute command to allow routes that are redistributed from the routing table to be filtered with a route map before being admitted into an EIGRP

topology table. The example shows how to configure a route map to match EIGRP routes with a metric of 110, 200, or an inclusive range of 700 to 800. When the match clause is true, the tag value of the destination routing protocol is set to 10. The route map is used to redistribute EIGRP packets.

```
switch(config)# route-map metric-eigrp
switch(config-route-map)# match metric 110 200 750 +- 50
switch(config-route-map)# set tag 10
switch(config-route-map)# exit
switch(config)# router eigrp 1
switch(config-router)# redistribute eigrp route-map metric-eigrp
switch(config-router)# exit
switch(config)# interface ethernet 1/2
switch(config-if)# ip address 172.16.0.0
switch(config-if)# ip router eigrp 1
```

Related Topics

See Configuring Route Policy Manager, for more information about route maps.

Additional References

For additional information related to implementing EIGRP, see the following sections:

Related Documents

Related Topic	Document Title
EIGRP CLI commands	*Cisco Nexus 3000 Series NX-OS N3K Mode Command Reference*
Introduction to EIGRP Tech Note	Introduction to EIGRP Tech Note
EIGRP Frequently Asked Questions	EIGRP Frequently Asked Questions