BGP Additional Paths

The BGP Additional Paths feature allows the advertisement of multiple paths through the same peering session for the same prefix without the new paths implicitly replacing any previous paths. This behavior promotes path diversity and reduces multi-exit discriminator (MED) oscillations.

Finding Feature Information

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

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

Information About BGP Additional Paths

Problem That Additional Paths Can Solve

BGP routers and route reflectors (RRs) propagate only their best path over their sessions. The advertisement of a prefix replaces the previous announcement of that prefix (this behavior is known as an implicit withdraw). The implicit withdraw can achieve better scaling, but at the cost of path diversity.

Path hiding can prevent efficient use of BGP multipath, prevent hitless planned maintenance, and can lead to MED oscillations and suboptimal hot-potato routing. Upon nexthop failures, path hiding also inhibits fast and

local recovery because the network has to wait for BGP control plane convergence to restore traffic. The BGP Additional Paths feature provides a generic way of offering path diversity; the Best External or Best Internal features offer path diversity only in limited scenarios.

The BGP Additional Paths feature provides a way for multiple paths for the same prefix to be advertised without the new paths implicitly replacing the previous paths. Thus, path diversity is achieved instead of path hiding.

Path-Hiding Scenario

This section describes in more detail how path hiding can occur. In the following figure, we have prefix p with paths p1 and p2 advertised from BR1 and BR4 to RR1. RR1 selects the best path of the two and then advertises to PE only p1.

*Figure 1: RR Hiding an Additional Path*
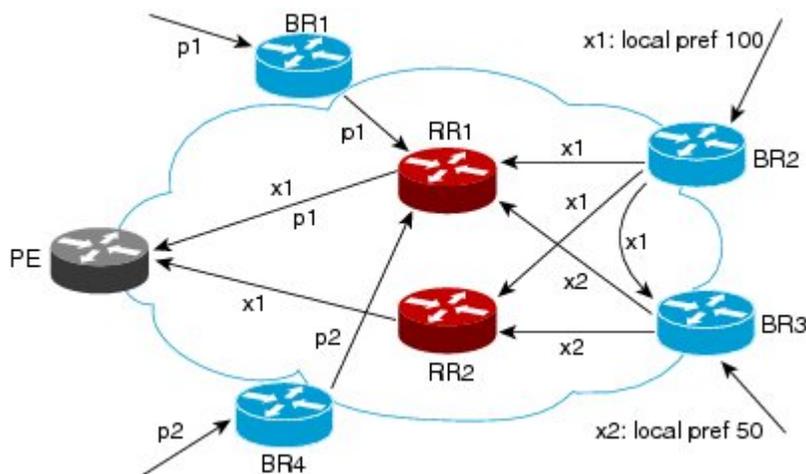
In the figure above, we also see prefix x with path x1 being advertised from BR2 to BR3 (which has path x2) with local preference 100. BR3 also has path x2, but due to routing policy, BR3 will advertise to the RRs x1 (not shown) instead of x2, and x2 will be suppressed. A user could enable the advertisement of best external on BR3 and thereby advertise x2 to the RRs, but, again, the RRs advertise only the best path.

Suboptimal Hot-Potato Routing Scenario

In order to minimize internal transport costs, transit ISPs try to forward packets to the closest exit point (according to Interior Gateway Protocol [IGP] cost). This behavior is known as hot-potato routing. In the distributed RR cluster model of the figure below, assume traffic coming from LA must go to Mexico. All links have the same IGP cost. If there are two exit points toward Mexico—one toward Austin and one toward Atlanta—the border router will try to send traffic to Austin based on the lower IGP cost from LA toward Austin than toward Atlanta. In a centralized RR model where the central RR resides where RR3 is (and RR1,

RR2, RR4, and RR5 do not exist), the closest exit point toward Mexico, as seen from RR3, might be Atlanta. Sending the traffic from LA toward Atlanta results in suboptimal hot-potato routing, which is not desirable.

Figure 2: Distributed RR Cluster
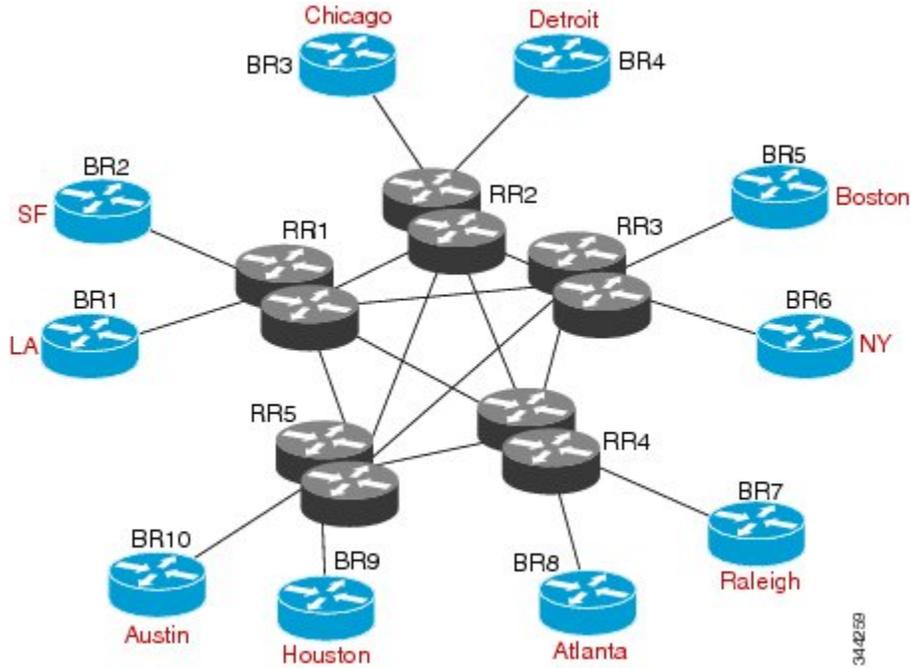
DMVPN Scenario

In Dynamic Multipoint Virtual Private Network (DMVPN) deployments, BGP is being used for scaling. In the figure below, Z is connected to both spokes S6 (NY) and S7 (Boston). The S7 links to the hubs have lower IGP costs than the S6 links to the hubs. There are physical links not shown that connect S5 to S6 and S6 to S7, with IGP costs lower than those to the hubs. Spokes S6 and S7 will send an update to both hubs H1 (Chicago) and H2 (Detroit). The RR hubs will then select the best path based on their lower IGP cost, which

might be S7. The spoke S5 (Raleigh) will receive two updates from the RRs for Z with S7 being the next hop, even though, in this scenario, it might be preferable to pick S6 (NY) as the next hop.

Figure 3: DMVPN Deployment
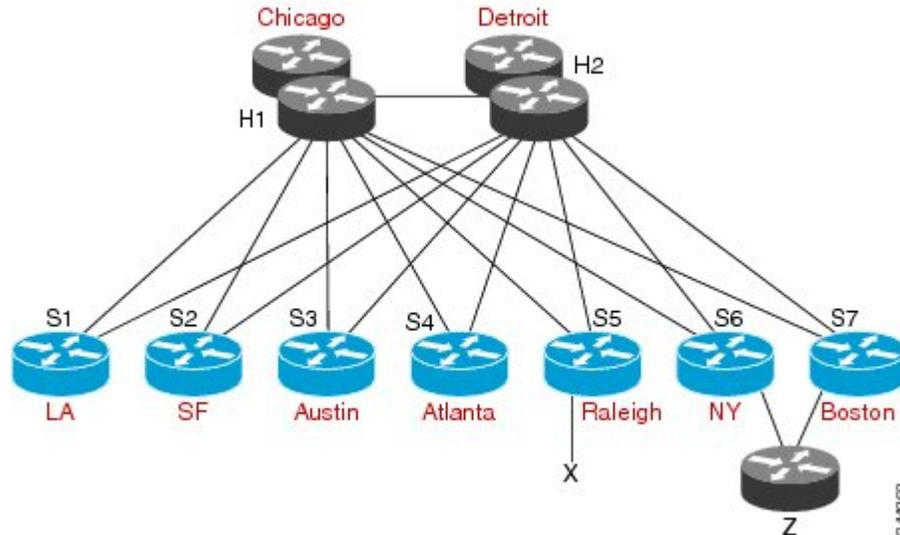
Benefits of BGP Additional Paths

BGP routers and route reflectors (RR) propagate only their best path over their sessions. The advertisement of a prefix replaces the previous announcement of that prefix (this in known as an implicit withdraw).

While this behavior may achieve better scaling, it can prevent path diversity, which tends to be poor or completely lost. The behavior in turn prevents efficient use of BGP multipath, prevents hitless planned maintenance, and can lead to multi-exit discriminator (MED) oscillations and suboptimal hot-potato routing. It also inhibits fast and local recovery upon nexthop failures, because the network has to wait for BGP control plane convergence to restore traffic.

The BGP Additional Paths feature is a BGP extension that allows the advertisement of multiple paths for the same prefix without the new paths implicitly replacing any previous paths. This behavior promotes path diversity and reduces MED oscillations.

BGP Additional Paths Functionality

The BGP Additional Paths feature is implemented by adding a path identifier to each path in the NLRI. The path identifier (ID) can be considered as something similar to a route distinguisher (RD) in VPNs, except that a path ID can apply to any address family. Path IDs are unique to a peering session and are generated for each network. The path identifier is used to prevent a route announcement from implicitly withdrawing the previous one. The Additional Paths feature allows the advertisement of more paths, in addition to the bestpath. The Additional Paths feature allows the advertisement of multiple paths for the same prefix, without the new paths implicitly replacing any previous paths.

The BGP Additional Paths feature requires the user to take three general steps:

1. Specify whether the device can send, receive, or send and receive additional paths. This is done at the address family level or the neighbor level, and is controlled by either the **bgp additional-paths** {**send** [**receive**] | **receive**} command or the **neighbor additional-paths** {**send** [**receive**] | **receive**} command, respectively. During session establishment, two BGP neighbors negotiate the Additional Path capabilities (whether they can send and/or receive) between them.

2. Select a set or sets of candidate paths for advertisement by specifying selection criteria (using the **bgp additional-paths select** command).

3. Advertise for a neighbor a set or sets of additional paths from the candidate paths marked (using the **neighbor advertise additional-paths** command).

To send or receive additional paths, the Additional Path capability must be negotiated. If it isn't negotiated, even if the selection criteria are such that more than the bestpath is marked and the neighbor is configured to advertise the marked paths, the selections would be useless because without the capability negotiated, only the bestpath can be sent.

Configuring BGP to send or receive additional paths triggers negotiation of additional path capability with the device's peers. Neighbors that have negotiated the capability will be grouped together in an update group (if other update group policies allow), and in a separate update group from those peers that have not negotiated the capability. Therefore, additional path capability causes the neighbor's update group membership to be recalculated.

Additional Path Selection

There are three path selection (path marking) policies, and they are not mutually exclusive. They are specified per address family, using the **bgp additional-paths select** command. They are:

- **best 2** or **best 3** (**best 2** means the bestpath and 2nd best path; the 2nd best path is the one computed by eliminating best-path from the best-computation algorithm. Similarly, **best 3** means the bestpath, 2nd best path, and 3rd best path; the 3rd best path is the one computed by eliminating bestpath and 2nd best path from the best-computation algorithm.)

- **group-best** (calculates the group-best for prefixes during bestpath calculation; described further below)

- **all** (all paths with unique next hops are eligible for selection)

Definition of the group-best Selection

The **group-best** keyword is part of the following commands:

- **advertise additional-paths**

- **bgp additional-paths select**

- **match additional-paths advertise-set**

- **neighbor advertise additional-paths**

The **group-best** is the set of paths that are the best paths from the paths of the same AS. For example, suppose there are three autonomous systems: AS 100, 200, and 300. Paths p101, p102, and p103 are from AS 100; p201, p202, and p203 are from AS200; and p301, p302, and p303 are from AS300. If we run the BGP bestpath algorithm on the paths from each AS, the algorithm will select one bestpath from each set of paths from that AS. Assuming p101 is the best from AS100, p201 is the best from AS200, and p301 is the best from AS300, then the **group-best** is the set of p101, p201, and p301.

Advertise a Subset of the Paths Selected

Take care when you select a set of paths but want to advertise a different set of paths. If the set of paths you want to advertise is not a subset of the selected paths, then you will not advertise the paths you want advertised.

The following example configures the additional paths selected to be the group-best and all selections. However, the paths configured to be advertised to the neighbor are the best 3 paths. Because the selection and advertise policy are not the same, the subsequent message is displayed. In these cases, only the bestpath is advertised.

```
Device(config)# router bgp 100
Device(config-router)# address-family ipv4
Device(config-router-af)# bgp additional-paths send receive
Device(config-router-af)# bgp additional-paths select group-best all
Device(config-router-af)# neighbor 192.168.2.2 advertise additional-paths best 3
 % BGP: AF level 'bgp additional-paths select' more restrictive than advertising policy.
This is a reminder that AF level additional-path select commands are needed.
```

How to Configure BGP Additional Paths

Configuring Additional Paths per Address Family

To select which paths are candidates to be additional paths, you can perform any combination of Steps 6, 7, and 8, as long as you perform at least one of those steps.

If you want to disable additional paths per neighbor, see the "Disabling Additional Paths per Neighbor" section.

SUMMARY STEPS

1. **enable**
2. **configure   terminal**
3. **router bgp**   *as-number*
4. **address-family ipv4** [**unicast** | **multicast**]
5. **bgp   additional-paths** {**send** [**receive**] | **receive**}
6. **bgp additional-paths select group-best**
7. **bgp additional-paths select best** *number*
8. **bgp additional-paths select all**
9. **neighbor** {*ip-address* | *ipv6-address* | *peer-group-name* } **advertise additional-paths** [**best** *number*] [**group-best**] [**all**]
10. **end**

DETAILED STEPS

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

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Device# configure terminal	Enters global configuration mode.
Step 3	**router bgp** *as-number* **Example:** Device(config)# router bgp 65000	Configures a BGP routing process and enters router configuration mode. • The *as-number* argument indicates the number of an autonomous system that identifies the router to other BGP routers and tags the routing information passed along. Valid numbers are from 0 to 65535. Private autonomous system numbers that can be used in internal networks range from 64512 to 65535.
Step 4	**address-family ipv4** [**unicast** | **multicast**] **Example:** Device(config-router)# address-family ipv4	Enters address family configuration mode. • The following address families are supported: IPv4 unicast, IPv4 multicast, IPv4 unicast + label, IPv6 unicast, IPv6 multicast, and IPv6 multicast + label.
Step 5	**bgp additional-paths** {**send** [**receive**] | **receive**} **Example:** Device(config-router-af)# bgp additional-paths send receive	Enables BGP additional paths to be sent only, received only, or sent and received, after negotiation with the neighbor is completed. • This example enables additional paths to be sent and received.
Step 6	**bgp additional-paths select group-best** **Example:** Device(config-router-af)# bgp additional-paths select group-best	(Optional) Calculates the group-best for prefixes during bestpath calculation.
Step 7	**bgp additional-paths select best** *number* **Example:** Device(config-router-af)# bgp additional-paths select best 3	(Optional) Calculates the specified number of best paths, including the advertisement of the bestpath. • The value of *number* can be 2 or 3.
Step 8	**bgp additional-paths select all** **Example:** Device(config-router-af)# bgp additional-paths select all	(Optional) Specifies that all paths with unique next hops are eligible for selection.
Step 9	**neighbor** {*ip-address* | *ipv6-address* | *peer-group-name* } **advertise additional-paths** [**best** *number*] [**group-best**] [**all**]	Specifies which selection methods control the additional paths that are advertised to the neighbor.

Command or Action	Purpose
Example: Device(config-router-af)# neighbor 192.168.0.1 advertise additional-paths best 3 group-best all	
Step 10 **end** **Example:** Device(config-router-af)# end	(Optional) Exits to privileged EXEC mode.

Configuring Additional Paths per Neighbor

To select which paths are candidates to be additional paths, you can perform any combination of Steps 6, 7, and 8, as long as you perform at least one of those steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *as-number*
4. **address-family ipv4** [**unicast** | **multicast**]
5. **neighbor** {*ip-address* | *ipv6-address* | *peer-group-name*} **additional-paths** {**send** [**receive**] | **receive**}
6. **bgp additional-paths select group-best**
7. **bgp additional-paths select best** *number*
8. **bgp additional-paths select all**
9. **neighbor** {*ip-address* | *ipv6-address* | *peer-group-name*} **advertise additional-paths** [**best** *number*] [**group-best**] [**all**]
10. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**router bgp** *as-number* **Example:** `Device(config)# router bgp 65000`	Configures a BGP routing process and enters router configuration mode. • The *as-number* argument indicates the number of an autonomous system that identifies the router to other BGP routers and tags the routing information passed along. Valid numbers are from 0 to 65535. Private autonomous system numbers that can be used in internal networks range from 64512 to 65535.
Step 4	**address-family ipv4** [**unicast** | **multicast**] **Example:** `Device(config-router)# address-family ipv4 unicast`	Enters address family configuration mode. • The following address families are supported: IPv4 unicast, IPv4 multicast, IPv4 unicast + label, IPv6 unicast, IPv6 multicast, and IPv6 multicast + label.
Step 5	**neighbor** {*ip-address* | *ipv6-address* | *peer-group-name*} **additional-paths** {**send** [**receive**] | **receive**} **Example:** `Device(config-router-af)# neighbor 192.168.1.2 additional-paths send receive`	Enables the neighbor to send or receive additional paths after negotiation is completed. • This example enables the neighbor to send and receive additional paths. • Note that this command overrides any send or receive capability that might have been configured at the address-family level.
Step 6	**bgp additional-paths select group-best** **Example:** `Device(config-router-af)# bgp additional-paths select group-best`	(Optional) Calculates the group-best for prefixes during bestpath calculation.
Step 7	**bgp additional-paths select best** *number* **Example:** `Device(config-router-af)# bgp additional-paths select best 3`	(Optional) Calculates the specified number of best paths, including the selection of the bestpath. • The value of *number* can be 2 or 3.
Step 8	**bgp additional-paths select all** **Example:** `Device(config-router-af)# bgp additional-paths select all`	(Optional) Specifies that all paths with unique next hops are eligible for selection.

	Command or Action	Purpose
Step 9	neighbor {*ip-address*	*ipv6-address*	*peer-group-name*} advertise additional-paths [best *number*] [group-best] [all] Example: `Device(config-router-af)# neighbor 192.168.1.2 advertise additional-paths best 3 group-best all`	Specifies the selection methods that control which additional paths are advertised for the neighbor.
Step 10	end Example: `Device(config-router-af)# end`	(Optional) Exits to privileged EXEC mode.

Configuring Additional Paths Using a Peer Policy Template

In this configuration task example, the capability to send and receive additional paths and the selection criteria are configured for the address family, and then the template is configured.

SUMMARY STEPS

1. enable
2. configure terminal
3. router bgp *autonomous-system-number*
4. address-family ipv4 unicast
5. bgp additional-paths {send [receive] | receive}
6. bgp additional-paths select [best *number*] [group-best] [all]
7. template peer-policy *policy-template-name*
8. additional-paths {send [receive] | receive}
9. advertise additional-paths [best *number*] [group-best] [all]
10. exit
11. address-family ipv4 unicast
12. neighbor {*ip-address* | *ipv6-address* | *peer-group-name*} remote-as *autonomous-system-number*
13. neighbor *ip-address* inherit peer-policy *policy-template-name*
14. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Device> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**router bgp** *autonomous-system-number* **Example:** `Device(config)# router bgp 45000`	Enters router configuration mode and creates a BGP routing process.
Step 4	**address-family ipv4 unicast** **Example:** `Device(config-router)# address-family ipv4 unicast`	Configures the IPv4 address family.
Step 5	**bgp additional-paths** {**send** [**receive**] | **receive**} **Example:** `Device(config-router)# bgp additional-paths send receive`	Enables BGP additional paths to be sent only, received only, or sent and received for the peers in the address family.
Step 6	**bgp additional-paths select** [**best** *number*] [**group-best**] [**all**] **Example:** `Device(config-router)# bgp additional-paths select best 3 group-best all`	Causes the system to calculate BGP additional paths that can be candidates for advertisement in addition to a bestpath.
Step 7	**template peer-policy** *policy-template-name* **Example:** `Device(config-router)# template peer-policy rr-client-pt1`	Enters policy-template configuration mode and creates a peer policy template.
Step 8	**additional-paths** {**send** [**receive**] | **receive**} **Example:** `Device(config-router-ptmp)# additional-paths send receive`	Enables BGP additional paths to be sent only, received only, or sent and received for the peers covered by the peer policy template.

	Command or Action	**Purpose**
Step 9	**advertise additional-paths** [**best** *number*] [**group-best**] [**all**] **Example:** `Device(config-router-ptmp)# advertise additional-paths best 3 group-best all`	Specifies the selection methods that control which additional paths are advertised for the peers covered by the peer policy template.
Step 10	**exit** **Example:** `Device(config-router-ptmp)# exit`	Exits policy-template configuration mode and returns to router configuration mode.
Step 11	**address-family ipv4 unicast** **Example:** `Device(config-router)# address-family ipv4 unicast`	Configures the IPv4 address family.
Step 12	**neighbor** {*ip-address* | *ipv6-address* | *peer-group-name*} **remote-as** *autonomous-system-number* **Example:** `Device(config-router-af)# neighbor 192.168.1.1 remote-as 45000`	Adds an entry to the BGP neighbor table.
Step 13	**neighbor** *ip-address* **inherit peer-policy** *policy-template-name* **Example:** `Device(config-router-af)# neighbor 192.168.1.1 inherit peer-policy rr-client-pt1`	Sends a peer policy template to a neighbor so that the neighbor can inherit the configuration.
Step 14	**end** **Example:** `Device(config-router-af)# end`	Exits address family configuration mode and returns to privileged EXEC mode.

Filtering and Setting Actions for Additional Paths

You can optionally use a route map to filter the paths to be advertised by matching on the tags of additional paths that are candidates to be advertised. (These tags are the advertise-sets that are configured with the **bgp additional-paths select** command.) Paths that have the same path marking (tag) as the marking that is

configured in the **match additional-paths advertise-set** command match the route map entry (and are permitted or denied).

You can also optionally set one or more actions to take for those paths that pass the route map. This task happens to use the **set metric** command to illustrate using a route map with the **match additional-paths advertise-set** command. Of course, other **set** commands are available that are not shown in this task.

Why set a metric for paths marked with **all** (all paths with a unique next hop)? Suppose the neighbor 2001:DB8::1037 is receiving the same route from different neighbors. Routes received from the local device have a metric of 565 and routes from another device perhaps have a metric of 700. Routes with metric 565 will have precedence over the routes with metric 700.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **route-map** *map-tag* [**permit** | **deny**] [*sequence-number*]
4. **match additional-paths advertise-set** [**best** *number*] [**best-range** *start-range end-range*] [**group-best**] [**all**]
5. **set metric** *metric-value*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**route-map** *map-tag* [**permit**	**deny**] [*sequence-number*] **Example:** `Device(config)# route-map additional_path1 permit 10`	Creates a route map.
Step 4	**match additional-paths advertise-set** [**best** *number*] [**best-range** *start-range end-range*] [**group-best**] [**all**] **Example:** `Device(config-route-map)# match additional-paths advertise-set best 3`	Matches on any path that is tagged with the specified path selection policy. • You must specify at least one selection method; you can specify more than one selection method in the command. • Specifying **best** *number* is incompatible with specifying **best-range**.

	Command or Action	Purpose
		• Specifying **best 1** will match only the bestpath.
		• Specifying **best-range 1 1** will match only the bestpath.
		• Only one **match additional-paths advertise-set** command is allowed per route map. A subsequent **match additional-paths advertise-set** command will overwrite the previous command.
Step 5	**set metric** *metric-value* **Example:** `Device(config-route-map)# set metric 500`	Sets the metric of the additional paths that pass the match criteria. • Note that other **set** commands can be used to take action on the paths that pass the route map. This example happens to use the **set metric** command.

What to Do Next

After creating the route map, you would reference the route map in the **neighbor route-map out** command. Thus, the route map is applied to paths being advertised (outgoing) to neighbors. Then you would use the **neighbor advertise additional-paths** command to advertise the additional paths. See the "Example: BGP Additional Paths" section to see the route map in context.

Displaying Additional Path Information

Perform either Step 2 or Step 3 in this task to see information about BGP additional paths.

SUMMARY STEPS

1. **enable**
2. **show ip bgp neighbors** [*ip-address*]
3. **show ip bgp** [*network*]

DETAILED STEPS

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

	Command or Action	**Purpose**
Step 2	**show ip bgp neighbors** [*ip-address*] **Example:** `Device# show ip bgp neighbors 192.168.1.1`	Displays the capabilities of the neighbor to send and receive additional paths.
Step 3	**show ip bgp** [*network*] **Example:** `Device# show ip bgp 192.168.0.0`	Displays the additional path selections and path ID for the network.

Disabling Additional Paths per Neighbor

If you had configured the sending or receiving of additional paths on a per neighbor basis (with the **neighbor additional-paths** command), and you wanted to disable that functionality, you would use the **no neighbor additional-paths** command.

However, if you had configured the sending or receiving of additional paths for an address family (with the **bgp additional-paths** command), and you wanted to disable that functionality for a neighbor, you would use the **neighbor additional-paths disable** command. Disabling additional paths also works if the functionality was inherited from a template.

Perform this task to disable additional path capability for a neighbor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router bgp** *as-number*
4. **address-family ipv6** [**unicast** | **multicast**]
5. **bgp additional-paths** {**send** [**receive**] | **receive**}
6. **neighbor** {*ip-address* | *ipv6-address* | *peer-group-name*} **additional-paths disable**
7. **end**

DETAILED STEPS

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

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Device# configure terminal	Enters global configuration mode.
Step 3	**router bgp** *as-number* **Example:** Device(config)# router bgp 65000	Configures a BGP routing process and enters router configuration mode. • The *as-number* argument indicates the number of an autonomous system that identifies the router to other BGP routers and tags the routing information passed along. Valid numbers are from 0 to 65535. Private autonomous system numbers that can be used in internal networks range from 64512 to 65535.
Step 4	**address-family ipv6** [**unicast** | **multicast**] **Example:** Device(config-router)# address-family ipv6 unicast	Enters address family configuration mode.
Step 5	**bgp additional-paths** {**send** [**receive**] | **receive**} **Example:** Device(config-router-af)# bgp additional-paths send receive	Enables BGP additional paths to be sent or received for the neighbors in the address family.
Step 6	**neighbor** {*ip-address* | *ipv6-address* | *peer-group-name*} **additional-paths disable** **Example:** Device(config-router-af)# neighbor 2001:DB8::1 additional-paths disable	Disables BGP additional paths from being sent to or received from the specified neighbor. • The additional path functionality is still enabled for the rest of the neighbors in the address family.
Step 7	**end** **Example:** Device(config-router-af)# end	(Optional) Exits to privileged EXEC mode.

Configuration Examples for BGP Additional Paths

Example: BGP Additional Path Send and Receive Capabilities

In this example, R1's address is 192.168.1.1; its neighbor is R2, which has address 192.168.1.2. Updates are sent from R2 to R1 with additional-paths (all paths advertised). Updates are sent from R1 to R2 with only the classic BGP bestpath advertised because R2 is only able to send additional paths, not receive additional paths.

R1

```
router bgp 1
 address-family ipv4 unicast
 bgp additional-paths select all
 neighbor 192.168.1.2 additional-paths send receive
 neighbor 192.168.1.2 advertise additional-paths all
```

R2

```
router bgp 2
 address-family ipv4 unicast
 bgp additional-paths select all
 neighbor 192.168.1.1 additional-paths send
 neighbor 192.168.1.1 advertise additional-paths all
```

Example: BGP Additional Paths

In the following example, for every address family, there are one or more eBGP neighbors not shown in the configuration that are sending routes to the local device. The eBGP routes learned from those neighbors are advertised toward the neighbors shown in the configuration below and the path attributes are changed. The example configures that:

- The route map called add_path1 specifies that all the paths are advertised toward neighbor 192.168.101.15, but any path that is marked with **best 2** will have its metric set to 780 before being sent toward that neighbor.

- The route map called add_path2 specifies that any path that is marked with **best 3** will have its metric set to 640 and will be advertised toward neighbor 192.168.25.

- The route map called add_path3 specifies that any path that is marked with **group-best** will have its metric set to 825 and will be advertised toward neighbor 2001:DB8::1045.

- In the IPv6 multicast address family, all paths are candidates to be advertised and will be advertised toward neighbor 2001:DB8::1037.

```
router bgp 1
 neighbor 192.168.101.15 remote-as 1
 neighbor 192.168.101.25 remote-as 1
 neighbor 2001:DB8::1045 remote-as 1
 neighbor 2001:DB8::1037 remote-as 1
!
 address-family ipv4 unicast
  bgp additional-paths send receive
  bgp additional-paths select all best 3 group-best
  neighbor 192.168.101.15 activate
  neighbor 192.168.101.15 route-map add_path1 out
  neighbor 192.168.101.15 advertise additional-paths best 2
 exit-address-family
!
address-family ipv4 multicast
```

```
  bgp additional-paths send receive
  bgp additional-paths select all best 3 group-best
  neighbor 192.168.101.25 activate
  neighbor 192.168.101.25 route-map add_path2 out
  neighbor 192.168.101.25 advertise additional-paths best 3
 exit-address-family
!
 address-family ipv6 unicast
  bgp additional-paths send receive
  bgp additional-paths select group-best
  neighbor 2001:DB8::1045 activate
  neighbor 2001:DB8::1045 route-map add_path3 out
  neighbor 2001:DB8::1045 advertise additional-paths all group-best
 exit-address-family
!
 address-family ipv6 multicast
  bgp additional-paths send receive
  bgp additional-paths select all
  neighbor 2001:DB8::1037 activate
  neighbor 2001:DB8::1037 route-map add_path4 out
  neighbor 2001:DB8::1037 advertise additional-paths all
 exit-address-family
!
route-map add_path1 permit 10
 match additional-paths advertise-set best 2
 set metric 780
route-map add_path1 permit 20
!
route-map add_path2 permit 10
 match additional-paths advertise-set best 3
 set metric 640
!
route-map add_path3 permit 10
 match additional-paths advertise-set group-best
 set metric 825
!
```

Example: Neighbor Capabilities Override Address Family Capabilities

In the following example, the receive-only capability of the neighbor overrides the send and receive capability of the address family:

```
router bgp 65000
 address-family ipv6 multicast
 bgp additional-paths send receive
 bgp additional-paths select group-best
 neighbor 2001:DB8::1037 activate
 neighbor 2001:DB8::1037 additional-paths receive
 neighbor 2001:DB8::1037 advertise additional-paths group-best
!
```

Example: BGP Additional Paths Using a Peer Policy Template

```
router bgp 45000
address-family ipv4 unicast
 bgp additional-paths send receive
 bgp additional-paths select all group-best best 3
 template peer-policy rr-client-pt1
  additional-paths send receive
  advertise additional-paths group-best best 3
  exit
 address-family ipv4 unicast
  neighbor 192.168.1.1 remote-as 45000
  neighbor 192.168.1.1 inherit peer-policy rr-client-pt1
```

```
    end
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
BGP commands	Cisco IOS IP Routing: BGP Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 3107	*Carrying Label Information in BGP-4*
RFC 4271	*A Border Gateway Protocl (BGP-4)*
RFC 4760	*Multiprotocol Extensions for BGP-4*

Technical Assistance

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

Feature Information for BGP Additional Paths

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

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

Table 1: Feature Information for BGP Additional Paths

Feature Name	Releases	Feature Information
BGP Additional Paths	15.2(1)E	The BGP Additional Paths feature allows the advertisement of multiple paths for the same prefix without the new paths implicitly replacing any previous paths.
		The following commands were introduced:
		• **additional-paths**
		• **advertise additional-paths**
		• **bgp additional-paths**
		• **bgp additional-paths select**
		• **match additional-paths advertise-set**
		• **neighbor additional-paths**
		• **neighbor advertise additional-paths**
		The following commands were modified:
		• **show ip bgp**
		• **show ip bgp neighbors**