



Implementing BFD

Bidirectional forwarding detection (BFD) provides low-overhead, short-duration detection of failures in the path between adjacent forwarding engines. BFD allows a single mechanism to be used for failure detection over any media and at any protocol layer, with a wide range of detection times and overhead. The fast detection of failures provides immediate reaction to failure in the event of a failed link or neighbor.



Note Cisco NCS 5000 Routers do not support BFD dampening. If it is enabled by default, disable it first before configuring any session. If dampening is disabled after configuring the BFD session, reload the device for it to take effect.



Tip You can programmatically configure BFD and retrieve operational data using `openconfig-bfd.yang` OpenConfig data model. To get started with using data models, see the *Programmability Configuration Guide*.

- [BFD over Bundle, on page 1](#)
- [BFD over Bundle and BFD over Logical Bundle, on page 4](#)
- [Enabling BFD Sessions on Bundle Members, on page 4](#)
- [Specifying the BFD Destination Address on a Bundle, on page 5](#)
- [Configuring the Minimum Thresholds for Maintaining an Active Bundle, on page 6](#)
- [Configuring BFD Packet Transmission Intervals and Failure Detection Times on a Bundle, on page 8](#)
- [Configure BFD over Bundles IETF Mode Support on a Per Bundle Basis, on page 9](#)
- [Multihop BFD over BVI, on page 10](#)

BFD over Bundle

BFD over Bundle feature enables BFD sessions to monitor the status of individual bundle member links. BFD notifies the bundle manager immediately when one of the member links goes down, and reduces the bandwidth used by the bundle.

Restrictions for BFD over Bundle

The following are the restrictions in using BFD over Bundle feature:

- It is only supported in IETF mode.

- It is only supported on the main bundle interface; it is not supported on bundle subinterfaces.
- It is not supported on routing protocols, such as OSPF, ISIS, and BGP.
- When the BFD timer is configured to 3.3 ms, which is the most aggressive timer, 256 sessions can be brought up.
- If BFD timer is configured to greater than 100 ms, 300 BFD sessions can be brought up simultaneously.
- BFD echo mode and encryption is not supported.
- BFD dampening is not supported.

Configure BFD over Bundle

Configuring BFD over bundle involves the following steps:

- Enable and Disable IPv6 checksum calculations for BFD on a router .
- Specify the mode, BFD packet transmission intervals, and failure detection times on a bundle.



Note Repeat the same configuration steps in the destination router.

```

/* Enable and Disable IPv6 checksum calculations for BFD on a router. */

Router(config-if)# bfd
Router(config-bfd-if)# ipv6 checksum disable
Router(config-bfd-if)# dampening disable
Router(config-bfd-if)# commit

/* Specify the mode, BFD packet transmission intervals, and failure detection times on a
bundle */

Router(config)# interface Bundle-Ether 3739
Router(config-if)# bfd mode ietf
Router(config-if)# bfd address-family ipv4 multiplier 3
Router(config-if)# bfd address-family ipv4 destination 10.23.1.2
Router(config-if)# bfd address-family ipv4 fast-detect
Router(config-if)# bfd address-family ipv4 minimum-interval 100
Router(config-if)# bfd address-family ipv6 multiplier 3
Router(config-if)# bfd address-family ipv6 destination 2001:DB8:1::2
Router(config-if)# bfd address-family ipv6 fast-detect
Router(config-if)# bfd address-family ipv6 minimum-interval 100
Router(config-if)# ipv4 address 10.23.1.1 255.255.255.252
Router(config-if)# ipv6 address 2001:DB8:1::2/120
Router(config-if)# load-interval 30
Router(config-if)# commit
Router(config)# interface TenGigE 0/0/0/0
Router(config-if)# bundle id 3739 mode active

```

Running Configuration

```

bfd
  ipv6 checksum disable

```

```

dampening disable!
!

interface Bundle-Ether3739
bfd mode ietf
bfd address-family ipv4 multiplier 3
bfd address-family ipv4 destination 10.23.1.2
bfd address-family ipv4 fast-detect
bfd address-family ipv4 minimum-interval 100
bfd address-family ipv6 multiplier 3
bfd address-family ipv6 destination 2001:DB8:1::2
bfd address-family ipv6 fast-detect
bfd address-family ipv6 minimum-interval 100
ipv4 address 10.23.1.1 255.255.255.252
ipv6 address 2001:DB8:1::2/120
load-interval 30
!

interface TenGigE 0/0/0/0
bundle id 3739 mode active

```

Verification

The following show command outputs displays the status of BFD sessions on bundle members:

```
/* Verify the details of the IPv4 BFD session in the source router. */
```

```
Router# show bfd session
```

Interface	Dest Addr	Local det	time(int*mult)	State	Echo	Async	H/W	NPU
Te0/0/0/0	10.23.1.2	0s(0s*0)	300ms(100ms*3)	UP	Yes			0/RP0/CPU0
BE3739	10.23.1.2	n/a	n/a	UP	No	n/a		

```
/* Verify the details of the IPv4 BFD session in the destination router. */
```

```
Router# show bfd session
```

Interface	Dest Addr	Local det	time(int*mult)	State	Echo	Async	H/W	NPU
Te0/6/0/0	10.23.1.1	0s(0s*0)	300ms(100ms*3)	UP	No	n/a		
BE3739	10.23.1.1	n/a	n/a	UP	No	n/a		

```
/* Verify the details of the IPv6 BFD session in the source router. */
```

```
Router# show bfd ipv6 session
```

Interface	Dest Addr	Local det	time(int*mult)	State	H/W	NPU	Echo	Async
Te0/0/0/0	10:23:1::2	Yes		0/RP0/0s	(0s*0)	00ms(100ms*3)	UP	
BE3739	10:23:1::2	No		n/a	n/a	n/a	UP	

```
/* Verify the details of the IPv6 BFD session in the destination router. */
```

```
Router# show bfd ipv6 session
```

Interface	Dest Addr	Local det	time(int*mult)	State	H/W	NPU	Echo	Async
Te0/6/0/0	10:23:1::1	No	n/a	0s(0s*0)		300ms(100ms*3)	UP	
BE3739	10:23:1::1	No	n/a	n/a		n/a	UP	

BFD over Bundle and BFD over Logical Bundle

Link Aggregation Control Protocol (LACP) allows a network device to negotiate an automatic bundling of links by sending LACP packets to their directly connected peer. LACP provides a keep-alive mechanism for the link members. While the default keep-alive is 30s, it is configurable to up to 1s. LACP can detect failures on a per-physical-member link. However, the LACP timers do not fulfill the criteria of current fast convergence requirements.

Differences between BFD over Bundle and BFD over Logical Bundle

BFD over Bundle (BoB) (RFC 7130) has a BFD session on each bundle member. The client is the bundle manager. If a BFD session goes down on a specific member link, the whole bundle interface goes down. That is, when the member link goes down, the number of available links falls below the required minimum. Hence the routing session is brought down.

BFD over Logical Bundle (BLB) (RFC 5880) treats a bundle interface with all its members as a single interface. BLB is a multipath (MP) single-hop session. If BLB is configured on a bundle there is only one single BFD session that is active. This implies that only one bundle member is being monitored by BFD at any given time. The client is one of the routing protocols. When BFD detects a failure, the client brings down the routing session.

The mode (BoB or BLB) is determined by how you configure BFD:

- You can enable BoB by configuring BFD under a Bundle-Ether interface.
- You can enable BLB by configuring BFD under a Bundle-Ether interface on a routing client.

Link Aggregation Control Protocol (LACP) allows a network device to negotiate an automatic bundling of links by sending LACP packets to their directly connected peer. LACP provides a keep-alive mechanism for the link members. While the default keep-alive is 30s, it is configurable to up to 1s. LACP can detect failures on a per-physical-member link. However, the LACP timers do not fulfill the criteria of current fast convergence requirements.

Enabling BFD Sessions on Bundle Members

To enable BFD sessions on bundle member links, complete these steps:

SUMMARY STEPS

1. **configure**
2. **interface Bundle-Ether** *bundle-id*
3. **bfd address-family ipv4 fast-detect**
4. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example:	Enters mode.

	Command or Action	Purpose
	<code>RP/0/RP0/CPU0:router# configure</code>	
Step 2	interface Bundle-Ether <i>bundle-id</i> Example: <code>RP/0/RP0/CPU0:router(config)# interface Bundle-Ether 1</code>	Enters interface configuration mode for the specified bundle ID.
Step 3	bfd address-family ipv4 fast-detect Example: <code>RP/0/RP0/CPU0:router(config-if)# bfd address-family ipv4 fast-detect</code>	Enables IPv4 BFD sessions on bundle member links.
Step 4	Use the commit or end command.	commit —Saves the configuration changes and remains within the configuration session. end —Prompts user to take one of these actions: <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration session, without committing the configuration changes.

Specifying the BFD Destination Address on a Bundle

To specify the BFD destination address on a bundle, complete these steps:

SUMMARY STEPS

1. **configure**
2. **interface Bundle-Ether *bundle-id***
3. **bfd address-family ipv4 destination *ip-address***
4. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: <code>RP/0/RP0/CPU0:router# configure</code>	Enters mode.

	Command or Action	Purpose
Step 2	interface Bundle-Ether <i>bundle-id</i> Example: <pre>RP/0/RP0/CPU0:router(config)# interface Bundle-Ether 1</pre>	Enters interface configuration mode for the specified bundle ID.
Step 3	bfd address-family ipv4 destination <i>ip-address</i> Example: <pre>RP/0/RP0/CPU0:router(config-if)# bfd address-family ipv4 destination 10.20.20.1</pre>	Specifies the primary IPv4 address assigned to the bundle interface on a connected remote system, where <i>ip-address</i> is the 32-bit IP address in dotted-decimal format (A.B.C.D).
Step 4	Use the commit or end command.	commit —Saves the configuration changes and remains within the configuration session. end —Prompts user to take one of these actions: <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration session, without committing the configuration changes.

Configuring the Minimum Thresholds for Maintaining an Active Bundle

The bundle manager uses two configurable minimum thresholds to determine whether a bundle can be brought up or remain up, or is down, based on the state of its member links.

- Minimum active number of links
- Minimum active bandwidth available

Whenever the state of a member changes, the bundle manager determines whether the number of active members or available bandwidth is less than the minimum. If so, then the bundle is placed, or remains, in DOWN state. Once the number of active links or available bandwidth reaches one of the minimum thresholds, then the bundle returns to the UP state.

To configure minimum bundle thresholds, complete these steps:

SUMMARY STEPS

1. **configure**
2. **interface Bundle-Ether** *bundle-id*
3. **bundle minimum-active bandwidth** *kbps*
4. **bundle minimum-active links** *links*

5. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters mode.
Step 2	interface Bundle-Ether <i>bundle-id</i> Example: RP/0/RP0/CPU0:router(config)# interface Bundle-Ether 1	Enters interface configuration mode for the specified bundle ID.
Step 3	bundle minimum-active bandwidth <i>kbps</i> Example: RP/0/RP0/CPU0:router(config-if)# bundle minimum-active bandwidth 580000	Sets the minimum amount of bandwidth required before a bundle can be brought up or remain up. The range is from 1 through a number that varies depending on the platform and the bundle type.
Step 4	bundle minimum-active links <i>links</i> Example: RP/0/RP0/CPU0:router(config-if)# bundle minimum-active links 2	Sets the number of active links required before a bundle can be brought up or remain up. The range is from 1 to 32. Note When BFD is started on a bundle that is already active, the BFD state of the bundle is declared when the BFD state of all the existing active members is known.
Step 5	Use the commit or end command.	commit —Saves the configuration changes and remains within the configuration session. end —Prompts user to take one of these actions: <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration session, without committing the configuration changes.

Configuring BFD Packet Transmission Intervals and Failure Detection Times on a Bundle

BFD asynchronous packet intervals and failure detection times for BFD sessions on bundle member links are configured using a combination of the **bfd address-family ipv4 minimum-interval** and **bfd address-family ipv4 multiplier** interface configuration commands on a bundle.

The BFD control packet interval is configured directly using the **bfd address-family ipv4 minimum-interval** command. The failure detection times are determined by a combination of the interval and multiplier values in these commands.

To configure the minimum transmission interval and failure detection times for BFD asynchronous mode control packets on bundle member links, complete these steps:

SUMMARY STEPS

1. **configure**
2. **interface Bundle-Ether** *bundle-id*
3. **bfd address-family ipv4 minimum-interval** *milliseconds*
4. **bfd address-family ipv4 multiplier** *multiplier*
5. Use the **commit** or **end** command.

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: RP/0/RP0/CPU0:router# configure	Enters mode.
Step 2	interface Bundle-Ether <i>bundle-id</i> Example: RP/0/RP0/CPU0:router(config)# interface Bundle-Ether 1	Enters interface configuration mode for the specified bundle ID.
Step 3	bfd address-family ipv4 minimum-interval <i>milliseconds</i> Example: RP/0/RP0/CPU0:router(config-if)#bfd address-family ipv4 minimum-interval 2000 Note Specifies the minimum interval, in milliseconds, for asynchronous mode control packets on IPv4 BFD sessions on bundle member links. The range is from 4 to 30000.	

	Command or Action	Purpose
Step 4	bfd address-family ipv4 multiplier <i>multiplier</i> Example: <pre>RP/0/RP0/CPU0:router(config-if)#bfd address-family ipv4 multiplier 30</pre>	<p>Specifies a number that is used as a multiplier with the minimum interval to determine BFD control packet failure detection times and transmission intervals for IPv4 BFD sessions on bundle member links. The range is from 2 to 50. The default is 3.</p> <p>Note Although the command allows you to configure a minimum of 2, the supported minimum is 3.</p>
Step 5	Use the commit or end command.	<p>commit —Saves the configuration changes and remains within the configuration session.</p> <p>end —Prompts user to take one of these actions:</p> <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration session, without committing the configuration changes.

Configure BFD over Bundles IETF Mode Support on a Per Bundle Basis

To configure BFD over Bundles IETF mode support on a per bundle basis use these steps:

SUMMARY STEPS

1. **configure**
2. **interface Bundle-Ether** *bundle-id*
3. **bfd mode ietf**
4. **bfd address-family ipv4 fast-detect**
5. Use the **commit** or **end** command.
6. **show bundle bundle-ether** *bundle-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: <pre>RP/0/RP0/CPU0:router# configure</pre>	Enters mode.

	Command or Action	Purpose
Step 2	interface Bundle-Ether <i>bundle-id</i> Example: <pre>RP/0/RP0/CPU0:router(config)# interface Bundle-Ether 1</pre>	Enters interface configuration mode for the specified bundle ID.
Step 3	bfd mode ietf Example: <pre>RP/0/RP0/CPU0:router(config-if)# bfd mode ietf</pre>	Enables IETF mode for BFD over bundle for the specified bundle.
Step 4	bfd address-family ipv4 fast-detect Example: <pre>RP/0/RP0/CPU0:router(config-if)# bfd address-family ipv4 fast-detect</pre>	Enables IPv4 BFD sessions on the specified bundle.
Step 5	Use the commit or end command.	commit —Saves the configuration changes and remains within the configuration session. end —Prompts user to take one of these actions: <ul style="list-style-type: none"> • Yes — Saves configuration changes and exits the configuration session. • No —Exits the configuration session without committing the configuration changes. • Cancel —Remains in the configuration session, without committing the configuration changes.
Step 6	show bundle bundle-ether <i>bundle-id</i>	Displays the selected bundle mode.

Multihop BFD over BVI

Table 1: Feature History Table

Feature Name	Release Information	Feature Description

Multihop BFD over Bridge Group Virtual Interface (BVI)	Release 7.4.1	<p>The multihop BFD over Bridge Group Virtual Interface (BVI) feature introduces support for multihop BFD over (BVI). You can set up a multihop BFD session between two endpoints that have IP connectivity. This session is between a unique source-destination address pair that the client provides.</p> <p>This feature allows you to extend BFD on arbitrary paths. These arbitrary paths can span multiple network hops, hence detecting link failures.</p>
--	---------------	---

Multihop BFD over BVI feature allows you to configure both routing and bridging on the same interface using Integrated Routing Bridging (IRB). IRB enables you to route between a bridged domain and a routed domain with the Bridge Group Virtual Interface (BVI).

The BVI is a virtual interface within the router that acts like a normal, routed interface that does not support bridging, but represents the comparable bridge group to routed interfaces within the router.

Restrictions

- The minimum Multihop BFD timer for the BVI interface is 50 msec.
- The **multihop ttl-drop-threshold** command is not supported.
- The Multihop BFD over BVI or IRB functionality is supported only in asynchronous mode and does not support echo mode.
- The Multihop BFD over BVI feature is not supported over MPLS and SR core.

Supported Functionality

- This feature is supported in both IPv4 and IPv6.
- BFD Multihop over BVI feature supports on client BGP.
- BFD Multihop supports only over IP core.
- BFD Multihop supports on all currently supported media-type for BFD single-hop.

Configuration

```

/* Configure a BVI interface and assign an IP address */
Router(config)# interface BVI1
Router(config-if)# host-routing
Router(config-if)# mtu 8986
Router(config-if)# ipv4 address 10.1.1.1 255.255.255.0
Router(config-if)# ipv6 address 10:1:1::1/120

/* Configure the Layer 2 AC interface */
Router(config-if)# interface TenGigE0/5/0/6/0.1 l2transport
Router(config-subif)# encapsulation dot1q 1
Router(config-subif)# rewrite ingress tag pop 1 symmetric

/* Configure L2VPN Bridge Domain */
Router(config-subif)# l2vpn
Router(config-subif)# bridge group 1
Router(config-subif)# bridge-domain 1

```

```
Router(config-l2vpn-bg-bd) # interface TenGigE0/5/0/6/0.1
Router(config-l2vpn-bg-bd) # routed interface BVI1
```

Running Configuration

```
interface BVI1
  host-routing
  mtu 8986
  ipv4 address 10.1.1.1 255.255.255.0
  ipv6 address 10:1:1::1/120
  !
interface TenGigE0/5/0/6/0.1 l2transport
  encapsulation dot1q 1
  rewrite ingress tag pop 1 symmetric
  !
l2vpn
  bridge group 1
  bridge-domain 1
  interface TenGigE0/5/0/6/0.1
  !
  routed interface BVI1
  !
```

Repeat the configuration on the peer router.

```
/* Configure BGP as the routing protocol */
Router(config) # router bgp 1
Router(config-bgp) # neighbor 2.2.1.1
Router(config-bgp-nbr) # remote-as 1
Router(config-bgp-nbr) # bfd fast-detect
Router(config-bgp-nbr) # bfd minimum-interval 300
Router(config-bgp-nbr) # update-source Loopback1
Router(config-bgp-nbr) # address-family ipv4 unicast

/* Configure reachability to the BGP neighbour IP either via static or IGP*/
Router(config-bgp-nbr-af) # router static
Router(config-static) # address-family ipv4 unicast
Router(config-static-afi) # 2.2.1.1/32 10.1.1.2

/* Configure the line cards to allow hosting of Multipath BFD sessions. */
Router(config-static-afi) # bfd
Router(config-bfd) #

router bgp 1
neighbor 2.2.1.1
  remote-as 1
  bfd fast-detect
  bfd minimum-interval 300
  update-source Loopback1
  address-family ipv4 unicast
  !
router static
  address-family ipv4 unicast
  2.2.1.1/32 10.1.1.2
  !
bfd
  multipath include location !
```



Note To avoid the unsupported three-level recursion on BVI interfaces on the first and second generation of line cards, you must not configure the BVI interface as the next-hop in the static route configuration.

Verification

```
Router# show bfd session destination 2.2.1.1
Fri May 28 14:35:52.566 IST
```

Src Addr	Dest Addr	VRF Name	Local det time(int*mult)	H/W NPU
		Echo	Async	State
1.1.1.1	2.2.1.1	default		Yes
		n/a	900ms (300ms*3)	UP

