

IP Routing: BFD Configuration Guide, Cisco IOS XE 17 (Cisco ASR 900 Series)

First Published: 2020-01-13

Last Modified: 2026-06-08

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/c/en/us/about/legal/trademarks.html. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2026 Cisco Systems, Inc. All rights reserved.



CONTENTS

CHAPTER 1 **Feature History** 1

CHAPTER 2 **Bidirectional Forwarding Detection** 3

- Prerequisites for Bidirectional Forwarding Detection 3
- Restrictions for Bidirectional Forwarding Detection 3
- Information About Bidirectional Forwarding Detection 4
  - BFD Operation 4
    - Neighbor Relationships 5
    - BFD Detection of Failures 5
    - BFD Version Interoperability 6
    - BFD Process on the RP 6
    - BFD Support for Nonstop Forwarding with Stateful Switchover 6
    - BFD Support for Stateful Switchover 7
    - BFD Support for Static Routing 9
    - BFD on Multiple Hops 10
  - BFD over Routed Pseudowire 10
    - Restrictions for Routed Pseudowire on the RSP3 Module 10
  - Benefits of Using BFD for Failure Detection 10
- How to Configure Bidirectional Forwarding Detection 11
  - Configuring BFD Session Parameters on the Interface 11
  - Configuring BFD Support for Static Routing 13
  - Configuring BFD Echo Mode 15
    - Prerequisites 16
    - Restrictions 16
    - Configuring the BFD Slow Timer 17
    - Disabling BFD Echo Mode Without Asymmetry 18

Configuring BFD on Routed Pseudowire on the RSP3 Module 18
Verifying BFD on Routed Pseudowire 20
Creating and Configuring BFD Templates 20
Configuring a Single-Hop Template 20
Configuring a Multihop Template 21
Configuring a BFD Map 23
Configuration Examples for Bidirectional Forwarding Detection 24
Example: Configuring BFD in an EIGRP Network with Echo Mode Enabled by Default 24
Example: Configuring BFD in an OSPF Network 27
Example: Configuring BFD in a BGP Network 31
Example: Configuring BFD in an IS-IS Network 34
Example: Configuring BFD in an HSRP Network 36
Example: Configuring BFD Support for Static Routing 37
BFD Dampening 38
Restriction 39
Configuring BFD Dampening 39
Examples for BFD Dampening 40
Micro BFD 42
Features of Micro BFD 43
Functions of Micro BFD 43
Restrictions for Micro BFD 43
Steps to Configure Micro BFD on Layer 3 Port Channel 44
Configure Port-Channel 44
Verify Port Channel 45
Configure Micro BFD on Port Channel 45
Verify Micro BFD Session 47
Configure Micro BFD on Port Channel with EFPs 48
Verify Micro BFD on Port Channel with EFPs 50

CHAPTER 3 **Static Route Support for BFD over IPv6 53**
Restrictions for Static Route for BFD over IPv6 53
Information About Static Route Support for BFD over IPv6 53
BFDv6 Associated Mode 53
BFDv6 Unassociated Mode 54

How to Configure Bidirectional Forwarding Detection for IPv6 54
Specifying a Static BFDv6 Neighbor 54
Associating an IPv6 Static Route with a BFDv6 Neighbor 55
Configuration Examples for Static Route Support for BFD over IPv6 56
Example: Specifying an IPv6 Static BFDv6 Neighbor 56
Example: Associating an IPv6 Static Route with a BFDv6 Neighbor 56

CHAPTER 4 **BFD on BDI Interfaces** 57
Information About BFD on Bridge Domain Interfaces 57
BFD on Bridge Domain Interfaces 57
How to Configure BFD on BDI Interfaces 57
Enabling BFD on a Bridge Domain Interface 57
Associating an Ethernet Flow Point with a Bridge Domain 59
Configuration Examples for BFD on BDI Interfaces 61
Examples for BFD on BDI Interfaces 61

CHAPTER 5 **BFD on IP Unnumbered Interfaces** 63
Information about BFD on IP Unnumbered Interfaces 63
Prerequisites for BFD on IP Unnumbered Interfaces 63
Restrictions for BFD on IP Unnumbered Interfaces 64
Configuring BFD on IP Unnumbered Interfaces 64
Verifying BFD on IP Unnumbered Interface Configuration 67
Example: Configuration of BFD on IP Unnumbered Interfaces 68
Additional References 73
Feature Information for BFD on IP Unnumbered Interfaces 73

CHAPTER 6 **IS-IS Client for BFD** 75
Prerequisites for IS-IS Client for BFD 75
Restrictions for IS-IS Client for BFD 75
Information About IS-IS Client for BFD 75
IS-IS BFD Topology 75
IS-IS BFD Session Creation 76
IS-IS BFD IPv6 Session Deletion 76
How to Configure ISIS IPv6 Client for BFD 76

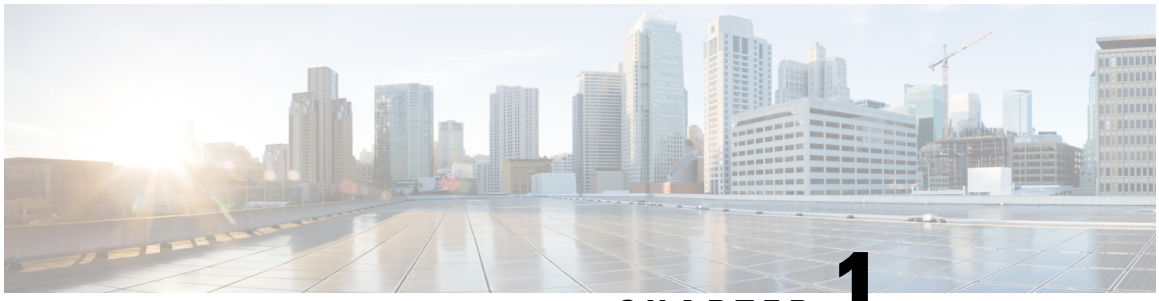
Configuring IS-IS IPv6 Client Support for BFD on an Interface 76
Configuring IS-IS IPv6 Client Support for BFD on All Interfaces 77
Configuration Examples for ISIS IPv6 Client for BFD 79
Example: IS-IS IPv6 Client Support for BFD on a Single Interface 79
Example: IS-IS IPv6 Client Support for BFD on All Interfaces 79

CHAPTER 7 **BFD IPv6 Encapsulation Support** 81
Prerequisites for BFD IPv6 Encapsulation Support 81
Restrictions for BFD IPv6 Encapsulation Support 81
Information About BFD IPv6 Encapsulation Support 82
Overview of the BFDv6 Protocol 82
BFDv6 Registration 82
BFDv6 Global and Link-Local Addresses 82
BFD for IPv4 and IPv6 on the Same Interface 83
How to Configure BFD IPv6 Encapsulation Support 83
Configuring BFD Session Parameters on the Interface 83
Configuration Examples for BFD IPv6 Encapsulation Support 84
Example: Configuring BFD Session Parameters on the Interface 84

CHAPTER 8 **Configuring BFD Debug Enhancement** 87
Prerequisites for BFD Debug Enhancement 87
Restrictions for BFD Debug Enhancement 87
Information About BFD Debug Enhancement 87
Overview of BFD Debug Enhancement 87
BFD Debug Enhancement - Event Tracer subsystem 88
BFD Debug Enhancement - Event Log Architecture 88
How to Configure BFD Debug Enhancement 88
Configuring BFD Debug Enhancement 88
Configuration Examples for BFD Debug Enhancement 90
Example: Configuring BFD Debug Enhancement 90

CHAPTER 9 **Bidirectional Forwarding Detection Configuration of Deterministic Hardware Offload** 93
Information About BFD Deterministic Hardware Offload 95
Configuring BFD Deterministic Hardware Offload 95

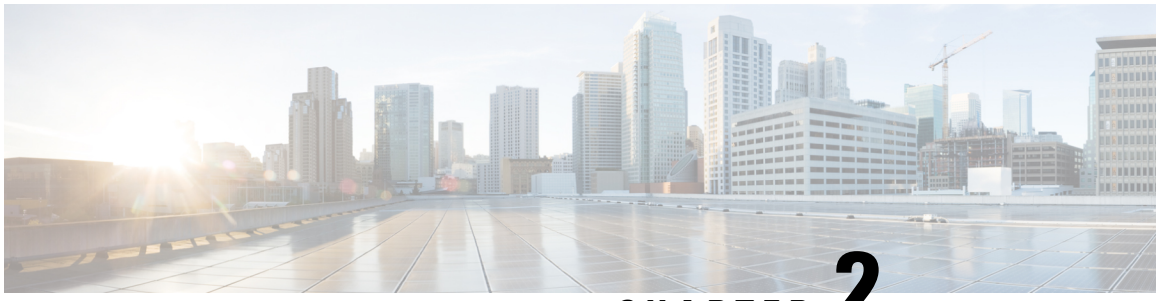
Example for Configuring BFD Deterministic Hardware Offload 96
Related Documents 96
Standards 96



CHAPTER 1

Feature History

Feature	Description
Cisco IOS XE Dublin 17.11.1	
Micro BFD Support on Port Channel with EFPs	A Micro Bidirectional Forwarding Detection (Micro-BFD) session detects failures in member links of a port channel. You can now enable Micro-BFD sessions for a port channel on which Ethernet flow Point (EFP) or service instance is configured. This feature ensures that traffic is forwarded to a member link only when the micro-BFD session for that member link is in the UP state. As part of this feature, the **source-service-instance** *number* keyword has been added to the **port-channel bfd** command. The specified service instance provides the source IP address for the micro-BFD session. This feature is supported on the Cisco RSP3 module.
Cisco IOS XE Bengaluru 17.6.1	
Micro BFD over LAG Convergence Optimization	Starting with 17.6.x release, the convergence for port-channel failures with Fast Reroute (FRR) is less than 50 milliseconds, when min-links is configured and equal to the total-links available under the port-channel. This feature is supported on the Cisco RSP3 module.
Cisco IOS XE Amsterdam 17.3.1	
BFD Dampening	Bidirectional Forwarding Detection (BFD) is a detection protocol that is designed to provide fast forwarding path failure detection for encapsulations, topologies, and routing protocols. BFD provides a consistent failure detection method. BFD detects forwarding path failures at a uniform rate, rather than the variable rates for different routing protocol. This feature is supported on the RSP2 module.



CHAPTER 2

Bidirectional Forwarding Detection

This document describes how to enable the Bidirectional Forwarding Detection (BFD) protocol. BFD is a detection protocol that is designed to provide fast forwarding path failure detection times for all media types, encapsulations, topologies, and routing protocols. It includes a description of how to configure multihop BFD sessions.

BFD provides a consistent failure detection method for network administrators, in addition to fast forwarding path failure detection. Because the network administrator can use BFD to detect forwarding path failures at a uniform rate, rather than the variable rates for different routing protocol hello mechanisms, network profiling and planning will be easier, and reconvergence time will be consistent and predictable.

Note BFD packets carries precedence of 7, and get prioritized by default. If you are applying any policy-map to change the priority for precedence 7, then BFD packets will also get subjected for the change. Avoid BFD packets drops due to queue congestion.

- Prerequisites for Bidirectional Forwarding Detection, on page 3
- Restrictions for Bidirectional Forwarding Detection, on page 3
- Information About Bidirectional Forwarding Detection, on page 4
- How to Configure Bidirectional Forwarding Detection, on page 11
- Configuration Examples for Bidirectional Forwarding Detection, on page 24
- BFD Dampening, on page 38
- Micro BFD, on page 42

Prerequisites for Bidirectional Forwarding Detection

- Cisco Express Forwarding and IP routing must be enabled on all participating routers

Restrictions for Bidirectional Forwarding Detection

- Minimum value supported on **bfd interval** command is 50ms.
- On RSP3, hardware supports BFD intervals of 3ms, 6ms, 9ms, 20ms, 50ms, 100ms, 200ms and 1000ms. If a user tries to configure unsupported values, it takes the lower supported value(in ms) by default. This is applicable to BFD hardware offloaded sessions.

Example: If a user tries to configure 150ms interval, the hardware BFD is programmed with 100ms timer

- BFD templates must be used to configure 3.3ms BFD. For more information, see Creating and Configuring BFD Templates, on page 20.
- For BFD over IPv6 hardware offload and software based sessions, only async mode is supported on the RSP3 module. Echo mode is not supported.
- Starting with Cisco IOS XE Release 3.18SP, BFD over port-channel (PoCH) is supported on the RSP3 module in software offload mode only. The minimum timer value supported is 200 milliseconds.
- BFD for MPLS-TP is not supported on the RSP3 module in Cisco IOS XE Release 3.16.
- BFD Echo Mode is not supported on the RSP3 module in Cisco IOS XE Release 3.16.

  Starting with Cisco IOS XE Fuji Release 16.9.1, BFD Echo mode is supported in the RSP3 module.
- Moving from echo to non-echo and non-echo to echo multiple times without any delay between each iteration may cause a few BFD sessions to flap. So multiple times echo to non-echo/non-echo to echo transition is not recommended and if this scenario is required then it should be done with a delay of at least 20 seconds at each step.
- The loopback packets are not prioritized for BFD echo sessions. Apply QoS policy to increase the priority of the packets.
- BFD packets for both, echo and non-echo sessions are not subjected to the Egress QoS. These packets are sent in the high-priority queue. This is applicable only on the Cisco ASR900 RSP1 and RSP2 modules.
- BFD with authentication is supported only in software. Hardware offload is not supported for BFD with authentication.
- The minimum timer value supported for the BFD software session is 200 ms.
- BFD over IPv4 unicast GRE tunnel is not supported.

Information About Bidirectional Forwarding Detection

BFD Operation

BFD provides a low-overhead, short-duration method of detecting failures in the forwarding path between two adjacent routers, including the interfaces, data links, and forwarding planes.

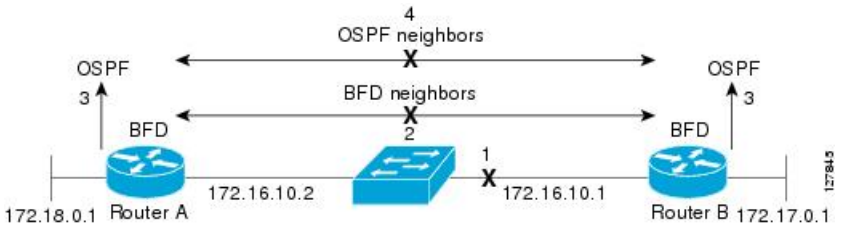
BFD is a detection protocol that is enabled at the interface and protocol levels. Cisco supports BFD asynchronous mode, which depends on the sending of BFD control packets between two systems to activate and maintain BFD neighbor sessions between routers. Therefore, in order for a BFD session to be created, BFD must be configured on both systems (or BFD peers). Once BFD has been enabled on the interfaces and at the router level for the appropriate protocols (NHRP and the routing protocol on overlay), a BFD session is created, BFD timers are negotiated, and the BFD peers will begin to send BFD control packets to each other at the negotiated interval.

Neighbor Relationships

BFD provides fast BFD peer failure detection times independently of all media types, encapsulations, topologies, and routing protocols BGP, EIGRP, IS-IS, and OSPF. By sending rapid failure detection notices to the routing protocols in the local router to initiate the routing table recalculation process, BFD contributes to greatly reduced overall network convergence time. The figure below shows a simple network with two routers running OSPF and BFD. When OSPF discovers a neighbor (1) it sends a request to the local BFD process to initiate a BFD neighbor session with the OSPF neighbor router (2). The BFD neighbor session with the OSPF neighbor router is established (3).



The figure below shows what happens when a failure occurs in the network (1). The BFD neighbor session with the OSPF neighbor router is torn down (2). BFD notifies the local OSPF process that the BFD neighbor is no longer reachable (3). The local OSPF process tears down the OSPF neighbor relationship (4). If an alternative path is available, the routers will immediately start converging on it.



A routing protocol needs to register with BFD for every neighbor it acquires. Once a neighbor is registered, BFD initiates a session with the neighbor if a session does not already exist.

OSPF registers with BFD when:

- A neighbor finite state machine (FSM) transitions to full state.
- Both OSPF BFD and BFD are enabled.

On broadcast interfaces, OSPF establishes a BFD session only with the designated router (DR) and backup designated router (BDR), but not between any two routers in DROTHER state.

Note A single BFD session notifies all protocols. For example, if OSFP and PIM neighbors exist, then a single BFD session notifies both the protocols.

BFD Detection of Failures

Once a BFD session has been established and timer negations are complete, BFD peers send BFD control packets that act in the same manner as an IGP hello protocol to detect liveliness, except at a more accelerated rate. The following information should be noted:

- BFD is a forwarding path failure detection protocol. BFD detects a failure, but the routing protocol must take action to bypass a failed peer.
- Cisco devices will use one BFD session for multiple client protocols in the Cisco implementation of BFD for Cisco IOS Releases 12.2(18)SXE, 12.0(31)S, and 12.4(4)T. For example, if a network is running OSPF and EIGRP across the same link to the same peer, only one BFD session will be established, and BFD will share session information with both routing protocols.

BFD Version Interoperability

All BFD sessions come up as Version 1 by default and will be interoperable with Version 0. The system automatically performs BFD version detection, and BFD sessions between neighbors will run in the highest common BFD version between neighbors. For example, if one BFD neighbor is running BFD Version 0 and the other BFD neighbor is running Version 1, the session will run BFD Version 0. The output from the **show bfd neighbors** [**details**] command will verify which BFD version a BFD neighbor is running.



Note RSP3 supports only Version 1 and do not support BFD version interoperability.

See the Example Configuring BFD in an EIGRP Network with Echo Mode Enabled by Default for an example of BFD version detection.

BFD Process on the RP

Client Interaction

The BFD process on the RP will handle the interaction with clients, which create and delete BFD sessions.

Session Management for the BFD Process on the RP

The BFD RP process will primarily own all BFD sessions on the router. It will pass the session creation and deletion requests to the BFD processes on all LCs. BFD LC sessions will have no knowledge of sessions being added or deleted by the clients. Only the BFD RP process will send session addition and deletion commands to the BFD LC process.

Session Database Management

The BFD RP process will maintain a database of all the BFD sessions on the router. This database will contain only the minimum required information.

Process EXEC Commands

The BFD RP process services the BFD **show** commands.

BFD Support for Nonstop Forwarding with Stateful Switchover

Typically, when a networking device restarts, all routing peers of that device detect that the device went down and then came back up. This transition results in a routing flap, which could spread across multiple routing domains. Routing flaps caused by routing restarts create routing instabilities, which are detrimental to the overall network performance. Nonstop forwarding (NSF) helps to suppress routing flaps in devices that are enabled with stateful switchover (SSO), thereby reducing network instability.

NSF allows for the forwarding of data packets to continue along known routes while the routing protocol information is being restored after a switchover. With NSF, peer networking devices do not experience routing flaps. Data traffic is forwarded through intelligent line cards or dual forwarding processors while the standby RP assumes control from the failed active RP during a switchover. The ability of line cards and forwarding processors to remain up through a switchover and to be kept current with the Forwarding Information Base (FIB) on the active RP is key to NSF operation.

In devices that support dual RPs, SSO establishes one of the RPs as the active processor; the other RP is designated as the standby processor, and then synchronizes information between them. A switchover from the active to the standby processor occurs when the active RP fails, when it is removed from the networking device, or when it is manually taken down for maintenance.

BFD Support for Stateful Switchover

The BFD protocol provides short-duration detection of failures in the path between adjacent forwarding engines. In network deployments that use dual RP routers or switches (to provide redundancy), the routers have a graceful restart mechanism that protects the forwarding state during a switchover between the active RP and the standby RP.

The dual RPs have variable switchover times that depend on the ability of the hardware to detect a communication failure. When BFD is running on the RP, some platforms are not able to detect a switchover before the BFD protocol times out; these platforms are referred to as slow switchover platforms.

Stateful BFD on the Standby RP

To ensure a successful switchover to the standby RP, the BFD protocol uses checkpoint messages to send session information from the active RP Cisco IOS instance to the standby RP Cisco IOS instance. The session information includes local and remote discriminators, adjacent router timer information, BFD setup information, and session-specific information such as the type of session and the session version. In addition, the BFD protocol sends session creation and deletion checkpoint messages to create or delete a session on the standby RP.

The BFD sessions on the standby RP do not receive or send packets and do not process expired timers. These sessions wait for a switchover to occur and then send packets for any active sessions so that sessions do not time out on adjacent routers.

When the BFD protocol on the standby RP is notified of a switchover it changes its state to active, registers itself with Cisco Express Forwarding so that it can receive packets, and then sends packets for any elements that have expired.

BFD also uses checkpoint messages to ensure that sessions created by clients on the active RP are maintained during a switchover. When a switchover occurs, BFD starts an SSO reclaim timer. Clients must reclaim their sessions within the duration specified by the reclaim timer or else the session is deleted.

BFD Timer Values

Table 1: BFD Timer Values on Cisco A900-RSP1A-55, Cisco A900-RSP1B-55

Maximum Number of BFD Sessions	Chassis Type	BFD Session Type	Minimum Timer Value (ms) [1]	Clients	Comments
511	RSP1	Async	3.3	BGP, OSFP, ISIS, EIGRP, MPLS-TP	NA
255	RSP1	Echo	3.3	BGP, OSFP, ISIS, EIGRP,	NA

[1] For Port-Channel, the minimum timer value is 750 multiplier 3.

Table 2: BFD Timer Values on Cisco A900-RSP2A-128, A900-RSP2A-64

Maximum Number of BFD Sessions	Chassis Type	BFD Session Type	Minimum Timer Value (ms) [2]	Clients	Comments
1023	RSP2	Async	3.3	BGP, OSFP, ISIS, EIGRP, MPLS-TP	NA
1023	RSP2	Echo	3.3	BGP, OSFP, ISIS, EIGRP,	NA

[2] For Port-Channel, the minimum timer value is 750 multiplier 3.

Table 3: BFD Timer Values on Cisco RSP3C-400-S

Maximum Number of BFD Sessions	Chassis Type	BFD Session Type	Minimum Timer Value (ms) [3]	Clients	Comments
1023 (IPv4) 64 (IPv6)	RSP3	Async	3.3	BGP, OSFP, ISIS, EIGRP	NA
64 (Software)	RSP3	Async	200	BGP, OSFP, ISIS, EIGRP	NA
511 (Hardware) (no client) 64 (Software)	RSP3	BFD echo	3.3	NA	NA
255 (Hardware) Scale 64 (Software)	RSP3	BFD echo	3.3	BGP, OSFP, ISIS, EIGRP	NA

[3] Port Channel is supported on RSP3 module in software. Minimum timer value for BFD over Port Channel is 200 ms. Hardware offload is not supported .

Note On Cisco RSP3C-400-S, software BFD sessions may be impacted by CPU spikes.

Note When BFD over IPv6 is configured, the maximum scale of BFD over IPv4 reduces to 64.

Note Whenever BFD discriminators are exhausted an error message “BFD discrminators exhausted. No more sessions can be created” with traceback will be printed.



Note Effective from the Cisco IOS XE Fuji 16.9.x onwards, Port-channel BFD (Micro-BFD as per RFC7130) on RSP3 is supported in both hardware offloaded and software offloaded.

Hardware offloaded session timers are supported from **3.3 ms to 200 ms** and software offloaded session timers are from **200 ms to 999 ms**.

BFD Support for Static Routing

Unlike dynamic routing protocols, such as OSPF and BGP, static routing has no method of peer discovery. Therefore, when BFD is configured, the reachability of the gateway is completely dependent on the state of the BFD session to the specified neighbor. Unless the BFD session is up, the gateway for the static route is considered unreachable, and therefore the affected routes will not be installed in the appropriate Routing Information Base (RIB).

For a BFD session to be successfully established, BFD must be configured on the interface on the peer and there must be a BFD client registered on the peer for the address of the BFD neighbor. When an interface is used by dynamic routing protocols, the latter requirement is usually met by configuring the routing protocol instances on each neighbor for BFD. When an interface is used exclusively for static routing, this requirement must be met by configuring static routes on the peers.

If a BFD configuration is removed from the remote peer while the BFD session is in the up state, the updated state of the BFD session is not signaled to IPv4 static. This will cause the static route to remain in the RIB. The only workaround is to remove the IPv4 static BFD neighbor configuration so that the static route no longer tracks BFD session state. Also, if you change the encapsulation type on a serial interface to one that is unsupported by BFD, BFD will be in a down state on that interface. The workaround is to shut down the interface, change to a supported encapsulation type, and then reconfigure BFD.

A single BFD session can be used by an IPv4 static client to track the reachability of next hops through a specific interface. You can assign a BFD group for a set of BFD-tracked static routes. Each group must have one active static BFD configuration, one or more passive BFD configurations, and the corresponding static routes to be BFD-tracked. Nongroup entries are BFD-tracked static routes for which a BFD group is not assigned. A BFD group must accommodate static BFD configurations that can be part of different VRFs. Effectively, the passive static BFD configurations need not be in the same VRF as that of the active configuration.

For each BFD group, there can be only one active static BFD session. You can configure the active BFD session by adding a static BFD configuration and a corresponding static route that uses the BFD configuration. The BFD session in a group is created only when there is an active static BFD configuration and the static route that uses the static BFD configuration. When the active static BFD configuration or the active static route is removed from a BFD group, all the passive static routes are withdrawn from the RIB. Effectively, all the passive static routes are inactive until an active static BFD configuration and a static route to be tracked by the active BFD session are configured in the group.

Similarly, for each BFD group, there can be one or more passive static BFD configurations and their corresponding static routes to be BFD-tracked. Passive static session routes take effect only when the active BFD session state is reachable. Though the active BFD session state of the group is reachable, the passive static route is added to the RIB only if the corresponding interface state is up. When a passive BFD session is removed from a group, it will not affect the active BFD session if one existed, or the BFD group reachability status.

BFD on Multiple Hops

Cisco IOS Release 15.1(3)S and later releases support BFD

on arbitrary paths, which might span multiple network hops. The BFD Multihop feature provides subsecond forwarding failure detection for a destination more than one hop, and up to 255 hops, away.

A BFD multihop session is set up between a unique source-destination address pair provided by the client. A session can be set up between two endpoints that have IP connectivity.

You must configure the **bfd-template** and **bfd map** commands to create a multihop template and associate it with one or more maps of destinations and associated BFD timers. You can enable authentication and configure a key chain for BFD multihop sessions.

Multi-hop BFD over IPv6 is supported in software mode only.

BFD over Routed Pseudowire

Starting Cisco IOS XE Gibraltar Release 16.11.1, BFD over Routed Pseudowire is supported on the router.

Routed VPLS is the ability to route or bridge frames to and from the pseudowire. Routed VPLS is configured by assigning the IP address under a bridge domain interface (BDI), in addition to the configuring the **vfi** command. Multi-Point (VPLS) is supported.

Both the virtual forwarding interface (VFI) and the IP address is configured under the BDI. This configuration makes the BDI multi-functional and unique to other previously possible interfaces.

Following configurations are supported on the BDI:

- MPLS configuration
- LDP configuration

Routed pseudowire supports Layer3 routing in addition to Layer2 bridging of frames to, and from the pseudowire.

Restrictions for Routed Pseudowire on the RSP3 Module

- Routed pseudowire does not support IPv6 BFD sessions.
- Routed pseudowire does not support point to multipoint sessions.
- Ensure that the minimum timer configuration is greater than the convergence time of the core network to avoid unnecessary flaps. For example, if the core network convergence time is 50 ms, BFD timer that is configured must be 50 ms *3.

⚠

Caution A session flap may be seen while moving from explicit null to implicit null configuration or vice versa.

Benefits of Using BFD for Failure Detection

When you deploy any feature, it is important to consider all the alternatives and be aware of any trade-offs being made.

The closest alternative to BFD in conventional EIGRP, IS-IS, and OSPF deployments is the use of modified failure detection mechanisms for EIGRP, IS-IS, and OSPF routing protocols.

If you set EIGRP hello and hold timers to their absolute minimums, the failure detection rate for EIGRP falls to within a one- to two-second range.

If you use fast hellos for either IS-IS or OSPF, these Interior Gateway Protocol (IGP) protocols reduce their failure detection mechanisms to a minimum of one second.

There are several advantages to implementing BFD over reduced timer mechanisms for routing protocols:

- BFD on the CPU operates under interrupt like CEF switched traffic. EIGRP, IS-IS and OSPF protocol hellos are handled in the process switching path. This provides BFD greater scalability and reliability over protocol hellos.
- Although reducing the EIGRP, IS-IS, and OSPF timers can result in minimum detection timer of one to two seconds, BFD can provide failure detection in less than one second.
- Because BFD is not tied to any particular routing protocol, it can be used as a generic and consistent failure detection mechanism for EIGRP, IS-IS, and OSPF.
- Because some parts of BFD can be distributed to the data plane, it can be less CPU-intensive than the reduced EIGRP, IS-IS, and OSPF timers, which exist wholly at the control plane.

How to Configure Bidirectional Forwarding Detection

Configuring BFD Session Parameters on the Interface

The steps in this procedure show how to configure BFD on the interface by setting the baseline BFD session parameters on an interface. Repeat the steps in this procedure for each interface over which you want to run BFD sessions to BFD neighbors.

Note RSP3 Module eysupports only the following BFD interval timers:

3.3ms, 6.6ms, 10ms, 20ms, 50ms, 100ms, 200ms ,999ms. It is recommended that peer should also configure the same timer values.

Procedure

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 **bfd-template single-hop** *template-name*

Example:

```
Router(config)# bfd-template single-hop bfdtemplate1
```

Creates a single-hop BFD template and enters BFD configuration mode.

Step 4 **interval min-tx** *milliseconds* **min-rx** *milliseconds* **multiplier** *multiplier-value*

Example:

```
Router(bfd-config)# interval min-tx 120 min-rx 100 multiplier 3
```

Configures the transmit and receive intervals between BFD packets, and specifies the number of consecutive BFD control packets that must be missed before BFD declares that a peer is unavailable.

Step 5 **interface gigabitethernet** *number*

Example:

```
Device(config)# interface gigabitethernet 0/0/0
```

Specifies the Gigabit Ethernet interface and enters interface configuration mode.

Step 6 Perform one of the following steps:

- **ip address** *ipv4-address mask*
- **ipv6 address** *ipv6-address*/*mask*

Example:

Configuring an IPv4 address for the interface:

```
Device(config-if)# ip address 10.201.201.1 255.255.255.0
```

Configuring an IPv6 address for the interface:

```
Device(config-if)# ipv6 address 2001:DB8::/32
```

Configures an IP address for the interface.

Step 7 **bfd template** *template name*

Enables the BFD template.

Step 8 **end**

Example:

```
Device(config-if)# end
```

Exits interface configuration mode and returns to privileged EXEC mode.

Configuring BFD Support for Static Routing

Perform this task to configure BFD support for static routing. Repeat the steps in this procedure on each BFD neighbor. For more information, see the "Example: Configuring BFD Support for Static Routing" section.

Procedure

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 **bfd-template single-hop** *template-name*

Example:

```
Router(config)# bfd-template single-hop bfdtemplate1
```

Creates a single-hop BFD template and enters BFD configuration mode.

Step 4 **interval min-tx** *milliseconds* **min-rx** *milliseconds* **multiplier** *multiplier-value*

Example:

```
Router(bfd-config)# interval min-tx 120 min-rx 100 multiplier 3
```

Configures the transmit and receive intervals between BFD packets, and specifies the number of consecutive BFD control packets that must be missed before BFD declares that a peer is unavailable.

Step 5 **interface** *type number*

Example:

```
Device(config)# interface serial 2/0
```

Configures an interface and enters interface configuration mode.

Step 6 Perform one of the following steps:

- **ip address** *ipv4-address mask*
- **ipv6 address** *ipv6-address*/*mask*

Example:

Configuring an IPv4 address for the interface:

```
Device(config-if)# ip address 10.201.201.1 255.255.255.0
```

Configuring an IPv6 address for the interface:

```
Device(config-if)# ipv6 address 2001:db8:1:1::1/32
```

Configures an IP address for the interface.

Step 7 **bfd template** *template name*

Enables the BFD template.

Step 8 **exit**

Example:

```
Device(config-if)# exit
```

Exits interface configuration mode and returns to global configuration mode.

Step 9 Perform one of the following steps:

- **ip route static bfd** *interface-type interface-number ip-address* [**group** *group-name* [**passive**]]
- **ipv6 route static bfd** *interface-type interface-number ip-address* [**unaasosiated**]

Example:

```
Device(config)# ip route static bfd serial 2/0 10.1.1.1 group group1 passive
Device(config)# ipv6 route static bfd TenGigabitEthernet 0/0/7 10:1:1::2
```

Specifies a static route BFD neighbor.

- The *interface-type*, *interface-number*, and *ip-address* arguments are required because BFD support exists only for directly connected neighbors.

Step 10 Perform one of the following steps:

- **ip route** [**vrf** *vrf-name*] *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [**dhcp**] [*distance*] [**name** *next-hop-name*] [**permanent** | **track** *number*] [**tag** *tag*]
- **ipv6 route** [**vrf** *vrf-name*] *ipv6 prefix*/*mask* {*ipv6-address* | *interface-type interface-number* [*ipv6-address*]} [**name** *next-hop-name*] [**track** *number*] [**tag** *tag*]

Example:

```
Device(config)# ip route 10.0.0.1 255.0.0.0

Device(config)# ipv6 route 10:1:1::/64 TenGigabitEthernet0/0/7 10:1:1::2
```

Specifies a static route BFD neighbor.

Step 11 **exit**

Example:

```
Device(config)# exit
```

Exits global configuration mode and returns to privileged EXEC mode.

Step 12 Perform one of the following steps:

- **show ip static route**
- **show ipv6 static**

Example:

```
Device# show ip static route
Device# show ipv6 static route
```

(Optional) Displays static route database information.

Step 13 Perform one of the following steps:

- **show ip static route bfd**
- **show ipv6 static bfd**

Example:

```
Device# show ip static route bfd
Device# show ipv6 static route bfd
```

(Optional) Displays information about the static BFD configuration from the configured BFD groups and nongroup entries.

Step 14 **exit**

Example:

```
Device# exit
```

Exits privileged EXEC mode and returns to user EXEC mode.

Configuring BFD Echo Mode

BFD echo mode is enabled by default, but you can disable it such that it can run independently in each direction.

BFD echo mode works with asynchronous BFD. Echo packets are sent by the forwarding engine and forwarded back along the same path in order to perform detection--the BFD session at the other end does not participate in the actual forwarding of the echo packets. The echo function and the forwarding engine are responsible for the detection process; therefore, the number of BFD control packets that are sent out between two BFD neighbors is reduced. In addition, because the forwarding engine is testing the forwarding path on the remote (neighbor) system without involving the remote system, there is an opportunity to improve the interpacket delay variance, thereby achieving quicker failure detection times than when using BFD Version 0 with BFD control packets for the BFD session.

Echo mode is described as without asymmetry when it is running on both sides (both BFD neighbors are running echo mode).

BFD Slow TImer

BFD can use the slow timer to slow down the asycnhronous session when the echo mode is enabled, and reduce the number of BFD control packets that are sent between two BFD neighbors. Also, the forwarding engine tests the forwarding path on the remote (neighbor) system without involving the remote system, so there is less interpacket delay variability and faster failure detection times.

Prerequisites

BFD must be running on all participating routers.

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

The baseline parameters for BFD sessions on the interfaces over which you want to run BFD sessions to BFD neighbors must be configured. See the Configuring BFD Session Parameters on the Interface section for more information.

Restrictions

- BFD echo mode is supported only on single hop. Multi-hop is not supported.
- BFD Echo mode is not supported on the RSP3 module in releases before Cisco IOS Fuji Release 16.9.1. The below restrictions apply if you bring up an echo BFD session. The following error message is displayed:

  BFD template model:

  ```
  Please remove bfd template <template name> from the
  interface
  add no echo to template
  reapply bfd template <template name> on interface
  ```

  BFD legacy configuration:

  ```
  removal bfd interval <value> min_rx <value> multiplier
  <value> or no bfd interval will also suffice
  Now remove the bfd client configurations also i.e
  Ex the client is ospf remove bfd all-interfaces under
  ospf
  Apply the  bfd interval <value> min_rx <value>
  multiplier <value>
  add no bfd echo
  add back the bfd client configuration
  ```

  If you configure BFD in non-echo mode and then move to echo RSP3, following error message is displayed:

  ```
  "Echo Mode for ld = 0x%x not supported on RSP3\n"
          "add no echo to template name if configured via template\n"
          "or add no bfd echo to interface if configured\n"
          "via legacy(i.e bfd interval on the interface)
  ```

- All BFD sessions use a single timer value for the echo interval.
- BFD echo is supported onl on IPv4.
- Minimum Timer value supported in 3.3ms.
- BFD interval timer (Tx) values supported are (in ms) are 3.3, 6.6, 9.9, 20, 100, 200.

- Maximum mumber of BFD echo sessions supported is 511, however the recommended scale per client is 255 (for example, 255 OSPF and 255 ISIS).
- The BFD Clients supported are static route, OSPF/OSPFv3, BGP, EIGRP, ISIS, and HSRP.
- When the session is in the UP state, use the **shutdown** and the **no shutdown** command on both sides to move BFD session from non echo mode to echo mode or vice versa.
- If the BFD interval is configured with unsupported timer values, the BFD tx timer adjusts to its lower supported timer value.

Configuring the BFD Slow Timer

The steps in this procedure show how to change the value of the BFD slow timer. Repeat the steps in this procedure for each BFD router.

Procedure

Step 1 **enable**

Example:

```
Switch> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Switch# configure terminal
```

Enters global configuration mode.

Step 3 **bfd slow-timer** *milliseconds*

Example:

```
Switch(config)# bfd slow-timer 12000
```

Configures the BFD slow timer.

Step 4 **end**

Example:

```
Switch(config)# end
```

Exits global configuration mode and returns the router to privileged EXEC mode.

Disabling BFD Echo Mode Without Asymmetry

The steps in this procedure show how to disable BFD echo mode without asymmetry—no echo packets will be sent by the router, and the router will not forward BFD echo packets that are received from any neighbor routers.

Repeat the steps in this procedure for each BFD router.

Procedure

Step 1 **enable**

Example:

```
Router> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 3 **no bfd echo**

Example:

```
Router(config)# no bfd echo
```

Disables BFD echo mode.

- Use the **no** form to disable BFD echo mode.

  This command is not applicable for Cisco RSP3 module.

Step 4 **end**

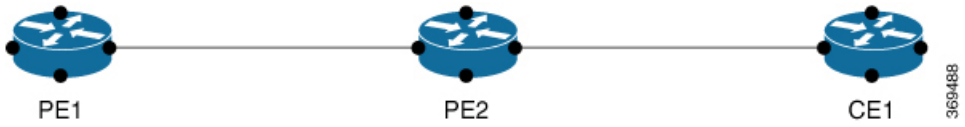
Example:

```
Router(config)# end
```

Exits global configuration mode and returns to privileged EXEC mode.

Configuring BFD on Routed Pseudowire on the RSP3 Module

Topology and Configuration



The Core consists of the PE1 and PE2 nodes where PE1 is the Routed pseudowire, and the CE node is the customer Layer3 interface. The BFD session is configured between the PE1 and CE node. VPLS is configured between PE1 and PE2.

Note The BFD session can also be configured between the two PE nodes.

Configuration on the PE1 node

```
VPLS Configuration:
         ethernet evc EVC10
interface TenGigabitEthernet0/3/0
 service instance 10 ethernet EVC10
 encapsulation dot1q 10
 rewrite ingress tag pop 1 symmetric
 bridge-domain 10
         l2 vfi VPLS10 manual EVC10
          vpn id 10

          neighbor 192.168.10.1 encapsulation mpls

ROUTED PSEUDOWIRE configuration:
        interface BDI10
         ip address 192.0.2.1 255.255.255.0
         ip ospf 4 area 0
         bfd template BFD_HW-50ms

BFD TEMPLATE Confguration:
        router ospf 4
         bfd all-interfaces
        bfd-template single-hop BFD_HW-50ms
         interval min-tx 50 min-rx 50 multiplier 3
```

Configuration on the PE2 node

```
VPLS Configuration:
         l2 vfi VPLS10 manual EVC10
          vpn id 10
          bridge-domain 10
          neighbor 192.168.10.11 encapsulation mpls

ACCESS INTERFACE Configuration:
         interface te0/0/0
                          service instance 10 ethernet
          encapsulation dot1q 10
          rewrite ingress tag pop 1 symmetric
          bridge-domain 10
```

Configuration on the CE node

```
INTERFACE Configuration connected to PE2:
         interface te0/02/0
         service instance 10 ethernet
          encapsulation dot1q 10
          rewrite ingress tag pop 1 symmetric
          bridge-domain 10
BDI Configuration:
```

```
         interface BDI10
         ip address 192.0.2.2 255.255.255.0
         ip ospf 12 area 0
         bfd template BFD_HW-50ms
BFD Template Configuration:
        bfd-template single-hop BFD_HW-50ms
        interval  min-tx 50 min-rx 50 multiplier 3
        router ospf 12
        bfd all-interfaces
```

Verifying BFD on Routed Pseudowire

Use the **show bfd neighbors** command to verify the BFD over routed pseudowire configuration.

```
Router# show bfd neighbors details

IPv4 Sessions
NeighAddr LD/RD RH/RS State Int
192.0.2.1 1/1 Up Up BD10
Session state is UP and not using echo function.
Session Host: Hardware
OurAddr: 192.0.2.2
Handle: 1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holddown (hits): 0(0), Hello (hits): 50(0)
Rx Count: 12441
Tx Count: 12507
Elapsed time watermarks: 0 0 (last: 0)
Registered protocols: OSPF CEF
Template: BFD_HW-50ms
Uptime: 00:10:28
Last packet: Version: 1 - Diagnostic: 0
 State bit: Up - Demand bit: 0
 Poll bit: 0 - Final bit: 0
 C bit: 1
 Multiplier: 3 - Length: 24
 My Discr.: 1 - Your Discr.: 1
 Min tx interval: 50000 - Min rx interval: 50000
 Min Echo interval: 0
```

Creating and Configuring BFD Templates

You can configure a single-hop template to specify a set of BFD interval values. BFD interval values specified as part of the BFD template are not specific to a single interface. You can configure a multihop template to associate these values with one or more maps of destinations and associated BFD timers. You can enable authentication and configure a key chain for BFD multihop sessions.

Note ASR 900 RSP3 Module for Cisco IOS XE Release 3.16 supports only the following BFD interval timers: 3.3ms, 6.6ms, 10ms, 20ms, 50ms, 100ms, 200ms ,999ms.

Configuring a Single-Hop Template

Perform this task to create a BFD single-hop template and configure BFD interval timers.

Procedure

Step 1 **enable**

Example:

```
Router> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 3 **bfd-template single-hop** *template-name*

Example:

```
Router(config)# bfd-template single-hop bfdtemplate1
```

Creates a single-hop BFD template and enters BFD configuration mode.

Step 4 **interval min-tx** *milliseconds* **min-rx** *milliseconds* **multiplier** *multiplier-value*

Example:

```
Router(bfd-config)# interval min-tx 120 min-rx 100 multiplier 3
```

Configures the transmit and receive intervals between BFD packets, and specifies the number of consecutive BFD control packets that must be missed before BFD declares that a peer is unavailable.

Step 5 **end**

Example:

```
Router(bfd-config)# end
```

Exits BFD configuration mode and returns the router to privileged EXEC mode.

Configuring a Multihop Template

Perform this task to create a BFD multihop template and configure BFD interval timers, authentication, and key chain.

Procedure

Step 1 **enable**

Example:

```
Router> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 3 **bfd-template multi-hop** *template-name*

Example:

```
Router(config)# bfd-template multi-hop mh-template1
```

Creates a BFD multihop BFD template and enters BFD configuration mode.

Step 4 **interval min-tx** *milliseconds* **min-rx** *milliseconds* **multiplier** *multiplier-value*

Example:

```
Router(bfd-config)# interval min-tx 120 min-rx 100 multiplier 3
```

Configures the transmit and receive intervals between BFD packets, and specifies the number of consecutive BFD control packets that must be missed before BFD declares that a peer is unavailable.

Step 5 **authentication** *authentication-type* **keychain** *keychain-name*

Example:

```
Router(bfd-config)# authentication keyed-sha-1 keychain bfd-multihop
```

Configures authentication for the multihop template and specifies the authentication type.

Step 6 **end**

Example:

```
Router(bfd-config)# end
```

Exits BFD configuration mode and returns the router to privileged EXEC mode.

Configuring a BFD Map

Perform this task to configure a BFD map that associates the interval timers and authentication configured in a template with unique source-destination address pairs for multihop BFD sessions.

Before you begin

You must configure a BFD multihop template before you associate it with a map.

Procedure

Step 1 **enable**

Example:

```
Router> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 3 Perform one of the following steps:

- **bfd map ipv4** *destination length source-address length template-name*
- **bfd map ipv4 vrf** *vrf-name destination length* **vrf** *vrfname source-address length template-name*

Example:

```
Router(config)# bfd map ipv4 192.168.0.0/24 192.168.42.5/32 mh-template1
Router(config)# bfd map ipv4 vrf vpn1 192.168.0.0/24 vrf vpn2 192.168.42.5/32 BFD_RPW_MH
```

Configures a BFD map and associates it with the template.

Mention the VRF name for the source-address as well, if source address is part of a VRF.

Step 4 **end**

Example:

```
Router(config)# end
```

Exits BFD configuration mode and returns the router to privileged EXEC mode.

Configuration Examples for Bidirectional Forwarding Detection

Example: Configuring BFD in an EIGRP Network with Echo Mode Enabled by Default

In the following example, the EIGRP network contains RouterA, RouterB, and RouterC. The Gigabit Ethernet interface 0/0/1 on RouterA is connected to the same network as Gigabit Ethernet interface 0/0/1 on Router B. The Gigabit Ethernet interface 0/0/1 on RouterB is connected to the same network as Gigabit Ethernet interface 0/0/1 on RouterC.

RouterA and RouterB are running BFD Version 1, which supports echo mode, and RouterC is running BFD Version 0, which does not support echo mode. The BFD sessions between RouterC and its BFD neighbors are said to be running echo mode with asymmetry because echo mode will run on the forwarding path for RouteA and RouterB, and their echo packets will return along the same path for BFD sessions and failure detections, while their BFD neighbor RouterC runs BFD Version 0 and uses BFD controls packets for BFD sessions and failure detections.

The example, starting in global configuration mode, shows the configuration of BFD.

Note The following configurations are in legacy mode. Legacy mode is not supported on Cisco RSP3. To configure on Cisco RSP3, use the template mode configuration as shown in Configuring BFD Session Parameters on the Interface, on page 11 section.

Configuration for RouterA

```
interface Gigabtethernet0/0/0
 no shutdown
 ip address 10.4.9.14 255.255.255.0
 duplex auto
 speed auto
!
interface Gigabtethernet0/0/1
 ip address 172.16.1.1 255.255.255.0
 bfd interval 50 min_rx 50 multiplier 3
 no shutdown
 duplex auto
 speed auto
!
router eigrp 11
 network 172.16.0.0
 bfd all-interfaces
 auto-summary
!
```

Configuration for RouterB

```
!
interface Gigabtethernet0/0/0
 no shutdown
 ip address 10.4.9.34 255.255.255.0
```

```
 duplex auto
 speed auto
!
interface Gigabtethernet0/0/1
 ip address 172.16.1.2 255.255.255.0
 bfd interval 50 min_rx 50 multiplier 3
 no shtdown
 duplex auto
 speed auto
!
router eigrp 11
 network 172.16.0.0
 bfd all-interfaces
 auto-summary
!
```

Configuration for RouterC

```
!
!
interface Gigabtethernet0/0/0
 no shutdown
 ip address 10.4.9.51 255.255.255.0
 duplex auto
 speed auto
!
interface Gigabtethernet0/0/1
 ip address 172.16.1.3 255.255.255.0
 bfd interval 50 min_rx 50 multiplier 3
 no shutdown
 duplex auto
 speed auto
!
router eigrp 11
 network 172.16.0.0
 bfd all-interfaces
 auto-summary
!
```

The output from the **show bfd neighbors details** command from RouterA verifies that BFD sessions have been created among all three routers and that EIGRP is registered for BFD support. The first group of output shows that RouterC with the IP address 172.16.1.3 runs BFD Version 0 and therefore does not use the echo mode. The second group of output shows that RouterB with the IP address 172.16.1.2 does run BFD Version 1, and the 50 millisecond BFD interval parameter had been adopted. The relevant command output is shown in bold in the output.

```
RouterA# show bfd neighbors details

OurAddr
       NeighAddr
      LD/RD  RH/RS    Holdown(mult)  State     Int
172.16.1.1    172.16.1.3
     5/3    1(RH)    150 (3 )       Up         Gig0/0/1
Session state is UP and not using echo function.
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holdown (hits): 150(0), Hello (hits): 50(1364284)
Rx Count: 1351813, Rx Interval (ms) min/max/avg: 28/64/49 last: 4 ms ago
Tx Count: 1364289, Tx Interval (ms) min/max/avg: 40/68/49 last: 32 ms ago
```

```
Registered protocols: EIGRP
Uptime: 18:42:45
Last packet: Version: 0
            - Diagnostic: 0
             I Hear You bit: 1     - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 3         - Length: 24
             My Discr.: 3          - Your Discr.: 5
             Min tx interval: 50000    - Min rx interval: 50000
             Min Echo interval: 0
OurAddr       NeighAddr
     LD/RD  RH/RS   Holdown(mult)  State     Int
172.16.1.1    172.16.1.2

    6/1    Up        0    (3 )   Up        Gig0/0/1
Session state is UP and using echo function with 50 ms interval.
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 3
Received MinRxInt: 1000000, Received Multiplier: 3
Holdown (hits): 3000(0), Hello (hits): 1000(317)
Rx Count: 305, Rx Interval (ms) min/max/avg: 1/1016/887 last: 448 ms ago
Tx Count: 319, Tx Interval (ms) min/max/avg: 1/1008/880 last: 532 ms ago
Registered protocols: EIGRP
Uptime: 00:04:30
Last packet: Version: 1

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

The output from the **show bfd neighbors details** command on Router B verifies that BFD sessions have been created and that EIGRP is registered for BFD support. As previously noted, RouterA runs BFD Version 1, therefore echo mode is running, and RouterC runs BFD Version 0, so echo mode does not run. The relevant command output is shown in bold in the output.

```
RouterB# show bfd neighbors details

OurAddr       NeighAddr
     LD/RD  RH/RS   Holdown(mult)  State     Int
172.16.1.2    172.16.1.1
    1/6    Up        0    (3 )   Up        Gig0/0/1
Session state is UP and using echo function with 50 ms interval.
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 3
Received MinRxInt: 1000000, Received Multiplier: 3
Holdown (hits): 3000(0), Hello (hits): 1000(337)
Rx Count: 341, Rx Interval (ms) min/max/avg: 1/1008/882 last: 364 ms ago
Tx Count: 339, Tx Interval (ms) min/max/avg: 1/1016/886 last: 632 ms ago
Registered protocols: EIGRP
Uptime: 00:05:00
Last packet: Version: 1
           - Diagnostic: 0
             State bit: Up         - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 3         - Length: 24
             My Discr.: 6          - Your Discr.: 1
             Min tx interval: 1000000    - Min rx interval: 1000000
             Min Echo interval: 50000
```

```
OurAddr       NeighAddr

 LD/RD  RH/RS   Holdown(mult)  State     Int
172.16.1.2    172.16.1.3
     3/6    1(RH)     118  (3 )   Up        Gig0/0/1
Session state is UP and not using echo function.
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holdown (hits): 150(0), Hello (hits): 50(5735)
Rx Count: 5731, Rx Interval (ms) min/max/avg: 32/72/49 last: 32 ms ago
Tx Count: 5740, Tx Interval (ms) min/max/avg: 40/64/50 last: 44 ms ago
Registered protocols: EIGRP
Uptime: 00:04:45
Last packet: Version: 0
            - Diagnostic: 0
             I Hear You bit: 1     - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 3         - Length: 24
             My Discr.: 6          - Your Discr.: 3
             Min tx interval: 50000    - Min rx interval: 50000
             Min Echo interval: 0
```

When Gigabit ethernet interface 0/0/1 on RouterB fails, BFD will no longer detect Router B as a BFD neighbor for RouterA or for RouterC. In this example, Giagbit Ethernet interface 0/0/1has been administratively shut down on RouterB.

The following output from the **show bfd neighbors** command on RouterA now shows only one BFD neighbor for RouterA in the EIGRP network. The relevant command output is shown in bold in the output.

```
RouterA# show bfd neighbors
OurAddr       NeighAddr

    LD/RD  RH/RS   Holdown(mult)  State     Int
172.16.1.1    172.16.1.3

    5/3    1(RH)     134  (3 )   Up        Gig0/0/1
```

The following output from the **show bfd neighbors** command on RouterC also now shows only one BFD neighbor for RouterC in the EIGRP network. The relevant command output is shown in bold in the output.

```
RouterC# show bfd neighbors

OurAddr       NeighAddr

   LD/RD RH  Holdown(mult)  State     Int
172.16.1.3    172.16.1.1

    3/5  1   114  (3 )      Up        Gig0/0/1
```

Example: Configuring BFD in an OSPF Network

In the following example, the simple OSPF network consists of Router A and Router B. Gigabit Ethernet interface 0/0/1 on Router A is connected to the same network as Gigabit Ethernet interface 0/0/1 in Router B. The example, starting in global configuration mode, shows the configuration of BFD. For both Routers A and B, BFD is configured globally for all interfaces associated with the OSPF process.

Note The following configurations are in legacy mode. Legacy mode is not supported on Cisco RSP3. To configure on Cisco RSP3, use the template mode configuration as shown in Configuring BFD Session Parameters on the Interface, on page 11 section.

Configuration for Router A

```
!
interface  Gigabitethernet 0/0/1
 ip address 172.16.10.1 255.255.255.0
 bfd interval 50 min_rx 50 multiplier 3
!
interface Gigabitethernet 0/0/0
ip address 172.17.0.1 255.255.255.0
!
router ospf 123
 log-adjacency-changes detail
 network 172.16.10.0 0.0.0.255 area 0
 network 172.17.10.0 0.0.0.255 area 0
 bfd all-interfaces
```

Configuration for Router B

```
!
interface Gigabitethernet 0/0/1
 ip address 172.16.10.2 255.255.255.0
 bfd interval 50 min_rx 50 multiplier 3
!
interface Gigabitethernet 0/0/0
 ip address 172.18.0.1 255.255.255.0
!
router ospf 123
 log-adjacency-changes detail
 network 172.16.10.0 0.0.0.255 area 0
 network 172.18.10.0 0.0.0.255 area 0
 bfd all-interfaces
```

The output from the **show bfd neighbors details** command verifies that a BFD session has been created and that OSPF is registered for BFD support. The relevant command output is shown in bold in the output.

Router A

```
RouterA# show bfd neighbors details
OurAddr       NeighAddr     LD/RD RH  Holdown(mult)  State     Int
172.16.10.1   172.16.10.2    1/2  1   532   (3 )      Up        Gig0/0/0
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 200000, MinRxInt: 200000, Multiplier: 5
Received MinRxInt: 1000, Received Multiplier: 3
Holdown (hits): 600(22), Hello (hits): 200(84453)
Rx Count: 49824, Rx Interval (ms) min/max/avg: 208/440/332 last: 68 ms ago
Tx Count: 84488, Tx Interval (ms) min/max/avg: 152/248/196 last: 192 ms ago
Registered protocols: OSPF

Uptime: 02:18:49
Last packet: Version: 0
             - Diagnostic: 0
```

```
             I Hear You bit: 1     - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 3         - Length: 24
             My Discr.: 2          - Your Discr.: 1
             Min tx interval: 50000    - Min rx interval: 1000
             Min Echo interval: 0
```

The output from the **show bfd neighbors details** command from Router B verifies that a BFD session has been created:

Router B

```
Router> show bfd neighbors details
Cleanup timer hits: 0
OurAddr       NeighAddr     LD/RD RH  Holdown(mult)  State     Int
172.16.10.2   172.16.10.1    8/1  1   1000 (5 )      Up        Gig0/0/0
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 1000, Multiplier: 3
Received MinRxInt: 200000, Received Multiplier: 5
Holdown (hits): 1000(0), Hello (hits): 200(5995)
Rx Count: 10126, Rx Interval (ms) min/max/avg: 152/248/196 last: 0 ms ago
Tx Count: 5998, Tx Interval (ms) min/max/avg: 204/440/332 last: 12 ms ago
Last packet: Version: 0            - Diagnostic: 0
             I Hear You bit: 1     - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 5         - Length: 24
             My Discr.: 1          - Your Discr.: 8
             Min tx interval: 200000    - Min rx interval: 200000
             Min Echo interval: 0
Uptime: 00:33:13
SSO Cleanup Timer called: 0
SSO Cleanup Action Taken: 0
Pseudo pre-emptive process count: 239103 min/max/avg: 8/16/8 last: 0 ms ago
 IPC Tx Failure Count: 0
 IPC Rx Failure Count: 0
 Total Adjs Found: 1
```

The output of the **show ip ospf** command verifies that BFD has been enabled for OSPF. The relevant command output is shown in bold in the output.

Router A

```
RouterA# show ip ospf

 Routing Process "ospf 123" with ID 172.16.10.1
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Incremental-SPF disabled
 Minimum LSA interval 5 secs
 Minimum LSA arrival 1000 msecs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 Number of external LSA 0. Checksum Sum 0x000000
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
```

```
 Number of areas in this router is 1. 1 normal 0 stub 0 nssa
 External flood list length 0
 BFD is enabled

    Area BACKBONE(0)
        Number of interfaces in this area is 2 (1 loopback)
        Area has no authentication
        SPF algorithm last executed 00:00:08.828 ago
        SPF algorithm executed 9 times
        Area ranges are
        Number of LSA 3. Checksum Sum 0x028417
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

Router B

```
RouterB# show ip ospf

 Routing Process "ospf 123" with ID 172.18.0.1
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Supports area transit capability
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Incremental-SPF disabled
 Minimum LSA interval 5 secs
 Minimum LSA arrival 1000 msecs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 Number of external LSA 0. Checksum Sum 0x0
 Number of opaque AS LSA 0. Checksum Sum 0x0
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 1. 1 normal 0 stub 0 nssa
 Number of areas transit capable is 0
 External flood list length 0
 BFD is enabled

    Area BACKBONE(0)
        Number of interfaces in this area is 2 (1 loopback)
        Area has no authentication
        SPF algorithm last executed 02:07:30.932 ago
        SPF algorithm executed 7 times
        Area ranges are
        Number of LSA 3. Checksum Sum 0x28417
        Number of opaque link LSA 0. Checksum Sum 0x0
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

The output of the **show ip ospf interface** command verifies that BFD has been enabled for OSPF on the interfaces connecting Router A and Router B. The relevant command output is shown in bold in the output.

Router A

```
RouterA# show ip ospf interface GigabitEthernet 0/0/1

GigabitEthernet0/0/1 is up, line protocol is up
  Internet Address 172.16.10.1/24, Area 0
  Process ID 123, Router ID 172.16.10.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State BDR, Priority 1, BFD enabled
  Designated Router (ID) 172.18.0.1, Interface address 172.16.10.2
  Backup Designated router (ID) 172.16.10.1, Interface address 172.16.10.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:03
  Supports Link-local Signaling (LLS)
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 172.18.0.1  (Designated Router)
  Suppress hello for 0 neighbor(s)
```

Router B

```
RouterB# show ip ospf interface Gigabit Ethernet 0/0/0
GigabitEthernet0/0/1 is up, line protocol is up
  Internet Address 172.18.0.1/24, Area 0
  Process ID 123, Router ID 172.18.0.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1, BFD enabled
  Designated Router (ID) 172.18.0.1, Interface address 172.18.0.1
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:01
  Supports Link-local Signaling (LLS)
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 0
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
```

Example: Configuring BFD in a BGP Network

In the following example, the simple BGP network consists of Router A and Router B. The Gigabit Ethernet interface 0/0/1 on Router A is connected to the same network as Gigabit Ethernet interface 0/0/1 in Router B. The example, starting in global configuration mode, shows the configuration of BFD.

Note The following configurations are in legacy mode. Legacy mode is not supported on Cisco RSP3. To configure on Cisco RSP3, use the template mode configuration as shown in Configuring BFD Session Parameters on the Interface, on page 11 section.

Configuration for Router A

```
!
interface GigabitEthernet 0/0/1
 ip address 172.16.10.1 255.255.255.0
 bfd interval 50 min_rx 50 multiplier 3
!
interface GigabitEthernet 0/0/0
ip address 172.17.0.1 255.255.255.0
!
!
router bgp 40000
 bgp log-neighbor-changes
 neighbor 172.16.10.2 remote-as 45000
 neighbor 172.16.10.2 fall-over bfd
 !
 address-family ipv4
 neighbor 172.16.10.2 activate
 no auto-summary
 no synchronization
 network 172.18.0.0 mask 255.255.255.0
 exit-address-family
!
```

Configuration for Router B

```
!
interface GigabitEthernet 0/0/1
 ip address 172.16.10.2 255.255.255.0
 bfd interval 50 min_rx 50 multiplier 3
!
interface GigabitEthernet 0/0/0
 ip address 172.18.0.1 255.255.255.0
!
router bgp 45000
 bgp log-neighbor-changes
 neighbor 172.16.10.1 remote-as 40000
 neighbor 172.16.10.1 fall-over bfd
 !
 address-family ipv4
 neighbor 172.16.10.1 activate
 no auto-summary
 no synchronization
 network 172.17.0.0 mask 255.255.255.0
 exit-address-family
!
```

The output from the **show bfd neighbors details** command from Router A verifies that a BFD session has been created and that BGP is registered for BFD support. The relevant command output is shown in bold in the output.

Router A

```
RouterA# show bfd neighbors details

OurAddr       NeighAddr     LD/RD RH  Holdown(mult)  State     Int
172.16.10.1   172.16.10.2    1/8  1   332  (3 )      Up        Gig0/0/1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 200000, MinRxInt: 200000, Multiplier: 5
Received MinRxInt: 1000, Received Multiplier: 3
```

```
Holdown (hits): 600(0), Hello (hits): 200(15491)
Rx Count: 9160, Rx Interval (ms) min/max/avg: 200/440/332 last: 268 ms ago
Tx Count: 15494, Tx Interval (ms) min/max/avg: 152/248/197 last: 32 ms ago
Registered protocols: BGP
Uptime: 00:50:45
Last packet: Version: 0            - Diagnostic: 0
             I Hear You bit: 1     - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 3         - Length: 24
             My Discr.: 8          - Your Discr.: 1
             Min tx interval: 50000    - Min rx interval: 1000
             Min Echo interval: 0
```

The output from the **show bfd neighbors details** command on Router B verifies that a BFD session has been created:

Router B

```
Router> show bfd neighbors details
Cleanup timer hits: 0
OurAddr       NeighAddr     LD/RD RH  Holdown(mult)  State     Int
172.16.10.2   172.16.10.1    8/1  1   1000 (5 )      Up        Gig0/0/0
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 1000, Multiplier: 3
Received MinRxInt: 200000, Received Multiplier: 5
Holdown (hits): 1000(0), Hello (hits): 200(5995)
Rx Count: 10126, Rx Interval (ms) min/max/avg: 152/248/196 last: 0 ms ago
Tx Count: 5998, Tx Interval (ms) min/max/avg: 204/440/332 last: 12 ms ago
Last packet: Version: 0            - Diagnostic: 0
             I Hear You bit: 1     - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 5         - Length: 24
             My Discr.: 1          - Your Discr.: 8
             Min tx interval: 200000    - Min rx interval: 200000
             Min Echo interval: 0
Uptime: 00:33:13
SSO Cleanup Timer called: 0
SSO Cleanup Action Taken: 0
Pseudo pre-emptive process count: 239103 min/max/avg: 8/16/8 last: 0 ms ago
 IPC Tx Failure Count: 0
 IPC Rx Failure Count: 0
 Total Adjs Found: 1
```

The output of the **show ip bgp neighbors** command verifies that BFD has been enabled for the BGP neighbors:

Router A

```
RouterA# show ip bgp neighbors
BGP neighbor is 172.16.10.2,  remote AS 45000, external link
 Using BFD to detect fast fallover
.
.
.
```

Router B

```
RouterB# show ip bgp neighbors
BGP neighbor is 172.16.10.1,  remote AS 40000, external link
 Using BFD to detect fast fallover
.
```

.
.

Example: Configuring BFD in an IS-IS Network

The Cisco IOS XE implementation of IS-IS conforms to the specifications detailed in RFC 6213. The IS-IS router advertises its BFD-enabled interfaces by including type-length-value (TLV) in the IS-IS Hello (IIH) PDU. The BFD-enabled TLV includes the topologies and protocols that it supports along with the appropriate Multi-Topology Identifier (MTID) or Network Layer Protocol Identifier (NLPID) pairs. Note that when the router sends an IIH to a non-BFD-enabled interface, it omits the TLV.

In the following example, the simple IS-IS network consists of Router A and Router B. The Gigabit Ethernet interface 0/0/0 on Router A is connected to the same network as Gigabit Ethernet interface 0/0/0 for Router B. The example, starting in global configuration mode, shows the configuration of BFD.

Note The following configurations are in legacy mode. Legacy mode is not supported on the Cisco RSP3 module. To configure on Cisco RSP3, use the template mode configuration as shown in the Configuring BFD Session Parameters on the Interface, on page 11 section.

Configuration for Router A

```
!
interface GigabitEthernet 0/0/0
 ip address 209.165.200.226 255.255.255.224
ip router isis
 bfd interval 50 min_rx 50 multiplier 3
!
router isis
 net 49.0001.1720.1600.1001.00
 bfd all-interfaces
!
```

Configuration for Router B

```
!
interface GigabitEthernet 0/0/0
 ip address 209.165.200.227 255.255.255.224
ip router isis
 bfd interval 50 min_rx 50 multiplier 3
!
router isis
 net 49.0000.0000.0002.00
 bfd all-interfaces
!
```

Router A

The output from the **show bfd neighbors detail** command from Router A verifies that a BFD session has been created and that IS-IS is registered for BFD support:

```
RouterA# show bfd neighbors detail

OurAddr             NeighAddr         LD/RD RH  Holdown(mult)  State     Int
```

```
209.165.200.226   209.165.200.227    1/8  1   536  (3 )      Up        Gig0/0/0
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 200000, MinRxInt: 200000, Multiplier: 5
Received MinRxInt: 1000, Received Multiplier: 3
Holdown (hits): 600(0), Hello (hits): 200(23543)
Rx Count: 13877, Rx Interval (ms) min/max/avg: 200/448/335 last: 64 ms ago
Tx Count: 23546, Tx Interval (ms) min/max/avg: 152/248/196 last: 32 ms ago
Registered protocols: ISIS
Uptime: 01:17:09
Last packet: Version: 0            - Diagnostic: 0
             I Hear You bit: 1     - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 3         - Length: 24
             My Discr.: 8          - Your Discr.: 1
             Min tx interval: 50000    - Min rx interval: 1000
             Min Echo interval: 0
```

Router B

The output from the **show bfd neighbors detail** command from the line card on Router B verifies that a BFD session has been created:

```
Router> show bfd neighbors detail
Cleanup timer hits: 0
OurAddr             NeighAddr       LD/RD RH  Holdown(mult)  State     Int
209.165.200.227   209.165.200.226    8/1  1   1000 (5 )      Up        Gig0/0/0
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 1000, Multiplier: 3
Received MinRxInt: 200000, Received Multiplier: 5
Holdown (hits): 1000(0), Hello (hits): 200(5995)
Rx Count: 10126, Rx Interval (ms) min/max/avg: 152/248/196 last: 0 ms ago
Tx Count: 5998, Tx Interval (ms) min/max/avg: 204/440/332 last: 12 ms ago
Last packet: Version: 0            - Diagnostic: 0
             I Hear You bit: 1     - Demand bit: 0
             Poll bit: 0           - Final bit: 0
             Multiplier: 5         - Length: 24
             My Discr.: 1          - Your Discr.: 8
             Min tx interval: 200000    - Min rx interval: 200000
             Min Echo interval: 0
Uptime: 00:33:13
SSO Cleanup Timer called: 0
SSO Cleanup Action Taken: 0
Pseudo pre-emptive process count: 239103 min/max/avg: 8/16/8 last: 0 ms ago
 IPC Tx Failure Count: 0
 IPC Rx Failure Count: 0
 Total Adjs Found: 1
```

Router B

The output from the **show isis neighbors detail** command from the line card on Router B shows the TLV field in the IS-IS Hello (IIH) PDU through which IS-IS responds to a BFD-detected forwarding plane failure. See Intermediate System-to-Intermediate System (IS-IS) TLVs for more information on TLV.

```
Router> show isis neighbors detail
PE1 L1 interface GigabitEthernet 0/0/0 209.165.200.226 UP 20 04
Area Address(es): 49
SNPA: e8ed.f3b8.d115
State Changed: 00:03:17
Format: Phase V
Remote TID: 0
```

```
Local TID: 0
Interface name: GigabitEthernet 0/0/0
Neighbor Circuit Id: 4
Remote BFD Support:TLV (MTID:0, IPV4) -----Remote BFD TLV's received from neighbor
peer
BFD enabled: (MTID:0, ipv4) --------BFD TLV's sent from Router B port
Adj sync: Full
```

Example: Configuring BFD in an HSRP Network

In the following example, the HSRP network consists of Router A and Router B. The Gigabit Ethernet interface 2/0 on Router A is connected to the same network as Gigabit Ethernet interface 2/0 on Router B. The example, starting in global configuration mode, shows the configuration of BFD.



Note In the following example, the **standby bfd** and the **standby bfd all-interfaces** commands are not displayed. HSRP support for BFD peering is enabled by default when BFD is configured on the router or interface using the **bfd interval** command. The **standby bfd**and **standby bfd all-interfaces** commands are needed only if BFD has been manually disabled on a router or interface.



Note The following configurations are in legacy mode. Legacy mode is not supported on Cisco RSP3. To configure on Cisco RSP3, use the template mode configuration as shown in Configuring BFD Session Parameters on the Interface, on page 11 section.

Router A

```
interface GigabitEthernet0/0/1
 no shutdown
 ip address 10.0.0.1 255.0.0.0
 bfd interval 200 min_rx 200 multiplier 3
 standby 1 ip 10.0.0.11
 standby 1 preempt
 standby 1 priority 110

 standby 2 ip 10.0.0.12
 standby 2 preempt
 standby 2 priority 110
```

Router B

```
interface GigabitEthernet0/0/1
 ip address 10.1.0.22 255.255.0.0
 no shutdown
 bfd interval 200 min_rx 200 multiplier 3
 standby 1 ip 10.0.0.11
 standby 1 preempt
 standby 1 priority 90
 standby 2 ip 10.0.0.12
 standby 2 preempt
 standby 2 priority 80
```

The output from the **show standby neighbors** command verifies that a BFD session has been created:

```
RouterA#show standby neighbors

HSRP neighbors on GigabitEthernet0/0/0
  10.1.0.22
    No active groups
    Standby groups: 1
    BFD enabled !
RouterB# show standby neighbors

HSRP neighbors on GigabitEthernet0/0/1
  10.0.0.2
    Active groups: 1
    No standby groups
    BFD enabled !
```

Example: Configuring BFD Support for Static Routing

In the following example, the network consists of Device A and Device B. Serial interface 2/0 on Device A is connected to the same network as serial interface 2/0 on Device B. In order for the BFD session to come up, Device B must be configured.

Note The following configurations are in legacy mode. Legacy mode is not supported on Cisco RSP3. To configure on Cisco RSP3, use the template mode configuration as shown in Configuring BFD Session Parameters on the Interface, on page 11 section.

Device A

```
configure terminal
interface Serial 2/0
ip address 10.201.201.1 255.255.255.0
bfd interval 500 min_rx 500 multiplier 5
ip route static bfd Serial 2/0 10.201.201.2
ip route 10.0.0.0 255.0.0.0 Serial 2/0 10.201.201.2
```

Device B

```
configure terminal
interface Serial 2/0
ip address 10.201.201.2 255.255.255.0
bfd interval 500 min_rx 500 multiplier 5
ip route static bfd Serial 2/0 10.201.201.1
ip route 10.1.1.1 255.255.255.255 Serial 2/0 10.201.201.1
```

Note that the static route on Device B exists solely to enable the BFD session between 10.201.201.1 and 10.201.201.2. If there is no useful static route that needs to be configured, select a prefix that will not affect packet forwarding, for example, the address of a locally configured loopback interface.

In the following example, there is an active static BFD configuration to reach 209.165.200.225 through Gigabit Ethernet interface 0/0/0 in the BFD group testgroup. As soon as the static route is configured that is tracked by the configured static BFD, a single hop BFD session is initiated to 209.165.200.225 through Ethernet interface 0/0. The prefix 10.0.0.0/8 is added to the RIB if a BFD session is successfully established.

```
configure terminal
ip route static bfd GigabitEthernet 0/0/0 209.165.200.225 group testgroup
ip route 10.0.0.0 255.255.255.224 Ethernet 0/0 209.165.200.225
```

In the following example, a BFD session to 209.165.200.226 through Ethernet interface 0/0.1001 is marked to use the group testgroup. That is, this configuration is a passive static BFD. Though there are static routes to be tracked by the second static BFD configuration, a BFD session is not triggered for 209.165.200.226 through Gigabit Ethernet interface 0/0/0. The existence of the prefixes 10.1.1.1/8 and 10.2.2.2/8 is controlled by the active static BFD session (Gigabit Ethernet interface 0/0/0 209.165.200.225).

```
configure terminal
ip route static bfd GigabitEthernet 0/0/0 209.165.200.225 group testgroup
ip route 10.0.0.0 255.255.255.224 GigabitEthernet 0/0/0 209.165.200.225
ip route static bfd Ethernet 0/0.1001 209.165.200.226 group testgroup passive
ip route 10.1.1.1 255.255.255.224 GigabitEthernet 0/0/0 209.165.200.226
ip route 10.2.2.2 255.255.255.224 GigabitEthernet 0/0/0 209.165.200.226
```

BFD Dampening

Table 4: Feature History

Feature Name	Release	Description
BFD Dampening	Cisco IOS XE Amsterdam 17.3.1	Bidirectional Forwarding Detection (BFD) is a detection protocol that is designed to provide fast forwarding path failure detection for encapsulations, topologies, and routing protocols. BFD provides a consistent failure detection method. BFD detects forwarding path failures at a uniform rate, rather than the variable rates for different routing protocol. This feature is supported on the RSP2 module.

The BFD Dampening feature introduces a configurable exponential delay mechanism. This mechanism is designed to suppress the excessive effect of remote node reachability events flapping with BFD. The BFD Dampening feature allows the network operator to automatically dampen a given BFD session to prevent excessive notification to BFD clients, thus preventing unnecessary instability in the network. Dampening the notification to a BFD client suppresses BFD notification until the time the session under monitoring stops flapping and becomes stable.

Configuring the BFD Dampening feature, especially on a high-speed interface with routing clients, improves convergence time and stability throughout the network. (For routing client: minimize route recomputation, prefix flushing and learning due to stop or restart of routing protocol, translating in turn into reduced utilization of system processing resources by other devices in the network).

When reachability to the node under monitoring is flapping, penalty is assigned to that session. Penalty is gradually increased as long as the session state is not stable. Penalty is suppressed when next hop reachability becomes stable (during less flapping). There are different configuration parameters that are introduced in addressing this requirement namely Suppress Threshold, Half-Life period, Reuse Threshold and maximum suppress time.

BFD session dampening can be applied to all types of BFD sessions including IPv4 or IPv6 single-hop or multihop and so on. BFD session dampening is configured through bfd-template CLI at global level.

Restriction

- BFD Dampening is not supported on BFD multihop client and Micro BFD.
- BFD Dampening with down monitoring, does not work for multihop sessions.

Configuring BFD Dampening

Procedure

Step 1 enable

Example:

```
Router> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 configure terminal

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 3 bfd-template multi-hop *template-name*

Example:

```
Router(config)# bfd-template multi-hop doctemplate
```

Creates a Bidirectional Forwarding Detection (BFD) template and enters BFD configuration mode.

Step 4 interval min-tx *milliseconds* min-rx *milliseconds* multiplier *multiplier-value*

Example:

```
Router(config-bfd)# interval min-tx 120 min-rx 100 multiplier 3
```

Configures the transmit and receive intervals between BFD packets, and specifies the number of consecutive BFD control packets that must be missed before BFD declares that a peer is unavailable.

Step 5 dampening [*half-life-period reuse-threshold suppress-threshold max-suppress-time stability-threshold down-monitoring*]

Example:

```
Router(config-bfd)# dampening 2 1000 3000 8
Router(config-bfd)# dampening 4 200 1000 10 60 down-monitoring
```

Configures a device to dampen a flapping session.

Note

Enable *down monitoring* to implement the BFD dampening with down monitoring feature.

Step 6 end

Example:

```
Router(config-bfd)# end
```

Exits BFD configuration mode and returns to privileged EXEC mode.

Examples for BFD Dampening

show bfd neighbor detail

Following command shows the dampening specific details if enabled for the session.

```
IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
10.0.0.2                               1/0            Up         Up     Et0/0
Session Host: Software
OurAddr: 10.0.0.1
Handle: 1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 3
Received MinRxInt: 0, Received Multiplier: 0
Holddown (hits): 0(0), Hello (hits): 1000(0)
Rx Count: 0, Rx Interval (ms) min/max/avg: 0/0/0 last: 1257099 ms ago
Tx Count: 0, Tx Interval (ms) min/max/avg: 0/0/0 last: 1257099 ms ago
Elapsed time watermarks: 0 0 (last: 0)
Dampening:   penalty: 0                      - not dampened
             flap count: 0                   - halflife: 2 sec
             reuse threshold: 2000           - suppress threshold: 3000
             max suppress time: 6 sec
Registered protocols: BGP CEF
Last packet: Version: 1                 - Diagnostic: 0
             State bit: AdminDown       - Demand bit: 0
             Poll bit: 0                - Final bit: 0
             C bit: 0
             Multiplier: 0              - Length: 0
             My Discr.: 0               - Your Discr.: 0
             Min tx interval: 0         - Min rx interval: 0
             Min Echo interval: 0
```

show bfd neighbor dampening

Following command displays the dampening information about the BFD sessions configured with BFD dampening.

```
IPv4 Multihop Sessions
NeighAddr[vrf]                                LD/RD         RH/RS     State
20.0.0.1                                      1/0           Up          Up
Dampening:   penalty: 0                       - not dampened
             flap count: 0                    - halflife: 2 sec
             reuse threshold: 2000            - suppress threshold: 3000
                       max suppress time: 6 sec
```

show bfd neighbor dampened

Following command displays the dampening information about the BFD sessions that are dampened.

```
IPv4 Sessions
NeighAddr[vrf]                                  LD/RD         RH/RS     State
20.0.0.1                                        1/0           Down      Down
Dampening:   penalty: 4500                        - dampened
             flap count: 10                       - halflife: 2 sec
             reuse threshold: 2000                - suppress threshold: 3000
                       max suppress time: 6 sec
```

Use detail keyword to view the entire session information.

```
IPv4 Sessions
NeighAddr                                LD/RD         RH/RS     State     Int
1.1.1.2                                   1/1          Up        Up        Et0/0
Session state is UP and using echo function with 50 ms interval.
Session Host: Software
OurAddr: 10.0.0.1
Handle: 1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 3
Received MinRxInt: 1000000, Received Multiplier: 3
Holddown (hits): 0(0), Hello (hits): 1000(181)
Rx Count: 259, Rx Interval (ms) min/max/avg: 1/1000/663 last: 58 ms ago
Tx Count: 297, Tx Interval (ms) min/max/avg: 1/1000/578 last: 58 ms ago
Elapsed time watermarks: 0 0 (last: 0)
Dampening: Penalty: 4999             - dampened
           Flap Count: 38            - Half Life: 2 sec
           Reuse Threshold: 1000     - Suppress Threshold: 3000
           Max suppress time: 5 sec
Registered protocols: BGP CEF
Template: damp
Uptime: 00:00:00
Last packet: Version: 1                  - Diagnostic: 0
             State bit: Up               - Demand bit: 0
             Poll bit: 0                 - Final bit: 0
             C bit: 0
             Multiplier: 3               - Length: 24
             My Discr.: 1                - Your Discr.: 1
             Min tx interval: 1000000    - Min rx interval: 1000000
```

show bfd neighbor dampened (down-monitoring output)

```
Router# show bfd neigh dampened

IPv4 Sessions
NeighAddr                                LD/RD         RH/RS     State     Int
1.1.1.2                                   3/3          Down      Down      BD10
Dampening: Penalty: 353              - dampened(down monitoring)
           Flap Count: 3             - Half Life: 4 seconds
           Reuse Threshold: 200      - Suppress Threshold: 1000
           Max suppress time  : 10 seconds
           Time to unsupress  : 4 seconds
           Stability threshold: 60 seconds
           Exponential Backoff counter: 0
```

show bfd neighbor dampening (down-monitoring output)

```
Router# show bfd neigh dampening
```

```
IPv4 Sessions
NeighAddr                               LD/RD         RH/RS     State     Int
1.1.1.2                                  3/3          Up        Up        BD10
Dampening: Penalty: 1               - not dampened(down monitoring)
           Flap Count: 3            - Half Life: 4 seconds
           Reuse Threshold: 200     - Suppress Threshold: 1000
           Max suppress time  : 10 seconds
           Time to unsupress  : 0 seconds
           Stability threshold: 60 seconds
           Exponential Backoff counter: 1
```

Micro BFD

To enable fast detection of link failure in a port channel, BFD sessions are run on every member link of the port channel. The BFD session that is mapped to a member link is referred to as a micro-BFD session. The micro-BFD sessions on the member links are independent BFD sessions.

Table 5: Feature History

Feature Name	Release	Description
Micro BFD Support on Port Channel with EFPs	Cisco IOS XE Dublin 17.11.1	A Micro Bidirectional Forwarding Detection (Micro-BFD) session detects failures in member links of a port channel. You can now enable Micro-BFD sessions for a port channel on which Ethernet flow Point (EFP) or service instance is configured. This feature ensures that traffic is forwarded to a member link only when the micro-BFD session for that member link is in the UP state. As part of this feature, the **source-service-instance** *number* keyword has been added to the **port-channel bfd** command. The specified service instance provides the source IP address for the micro-BFD session. This feature is supported on the Cisco RSP3 module.
Micro BFD over LAG Convergence Optimization	Cisco IOS XE Bengaluru 17.6.1	Starting with 17.6.x release, the convergence for port-channel failures with Fast Reroute (FRR) is less than 50 milliseconds, when min-links is configured and equal to the total-links available under the port-channel. This feature is supported on the Cisco RSP3 module.

A port channel is an aggregation of multiple physical interfaces or member links that creates a logical interface. You can configure micro BFD sessions for fast detection of failure that occurs on any of the port-channel member links. When BFD creates a session corresponding to each member link in a port channel, it is called a micro-BFD session. You can enable micro-BFD sessions for port channels with or without EFPs or service instances. The functionality of micro-BFD remains the same in both cases. The difference lies in selecting the source IP address as explained below:

When you enable micro-BFD in port channels without EFPs, Micro-BFD establishes the session with the source and destination as follows:

- Source Address—The IP address assigned on the port channel interface.
- Destination Address—Address as configured in the CLI

For port channels with EFPs, Micro-BFD establishes the session with the source and destination as follows:

- Source Address—Assign the IP address of the Bridge Domain Interface (BDI) that is associated with the EFP configured under the port channel.
- Destination Address—Address as configured in the CLI

Features of Micro BFD

- From Cisco IOS XE Dublin 17.11.1, Micro BFD supports a port-channel with EFPs or service instances. This functionality is an addition to the existing micro-BFD support for a port-channel without EFPs or service instances at Layer 3.
- Micro BFD is supported both in hardware offloaded and software offloaded.
- Timers supported on micro BFD hardware offloaded session are from 3.3 ms to 200 ms.
- Timers supported on micro BFD software offloaded session are from 200 ms to 999 ms.
- Micro BFD supports Active/Active SDM template for port-channel for Cisco RSP3 module.
- Interior gateway protocol is not necessary for the micro BFD sessions to be up between the LAG.
- The client for micro BFD sessions is port-channel.
- Micro BFD is supported on port-channel with member links from different ASICs.
- Micro BFD is supported with LACP minimum links and LACP maximum links enabled on the port-channel.
- The convergence achieved with this feature is less than 50 ms with BFD timers configured to 3.3ms with multiplier value as 3.

Functions of Micro BFD

- Runs BFD session over each Link Aggregation Group (LAG) member link.
- Verifies link continuity for every member link.
- Makes BFD to control the LAG member link to be part of the Layer 2 load-balancing table of the LAG interface in the presence or the absence of Link Aggregation Control Protocol (LACP).
- Offloads the continuity check messages.
- Provides a sub 50 millisecond convergence for LAG failures for releases before 17.6.x. Starting with Cisco IOS XE Bengaluru 17.6.x release, the convergence for port-channel failures with Fast Reroute (FRR) is less than 50 milliseconds, when min-links is configured and equal to the total-links available under the port-channel.

Restrictions for Micro BFD

- Micro BFD supports single hop only.
- Micro BFD works only with IPv4.

- Micro BFD does not support ECHO mode.
- Maximum tested scale is 46 sessions .

Steps to Configure Micro BFD on Layer 3 Port Channel

Perform the following steps to configure micro BFD:

1. Configure port channel. See Configure Port-Channel, on page 44.
2. If Micro BFD needs to be enabled on an Active/Active SDM template, enable port channel, then enable the SDM template for the port channel ID before configuring the port-channel. See Configuring QoS Support on Port Channel Active Active and Selecting the SDM template.
3. Configure BFD template. See Configure Micro BFD on Port Channel, on page 45.
4. Configure Micro BFD over port-channel with the template configured. See Configure Micro BFD on Port Channel, on page 45.

Configure Port-Channel

Procedure

Step 1 enable

Example:

```
Router> enable
```

Enables privileged EXEC mode.

Enter your password if prompted.

Step 2 configure terminal

Example:

```
Router# configure terminal
```

Enters the global configuration mode.

Step 3 **Interface port channel** *number*

Example:

```
Router(config)#interface port-channel 48
```

Configures the port channel and enters the interface mode.

Step 4 **ip address** *ip-address mask*

Example:

```
Router(config-if)# ip address 10.201.201.1 255.255.255.0
```

Configures an IP address for the interface.

Step 5 **Interface** *interface typeinterface name*

Example:

```
Router(config)#interface gigabitethernet 0/0/0-6
```

Specifies the ethernet interface.

Step 6 **channel group** *channel group number* **mode** *mode name*

Example:

```
Router(config)#channel group 48 mode active
```

The following types of port channel modes can be configured:

- active
- passive
- on

Step 7 **end**

Exits the interface range configuration mode and returns to the configuration mode.

Verify Port Channel

Use the commands below to verify the port channel summary details:

```
Device#show etherchannel summary
 Flags:  D - down        P/bndl - bundled in port-channel
        I - stand-alone s/susp - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


 Number of channel-groups in use: 1
 Number of aggregators:           1

 Group  Port-channel  Protocol    Ports
 ------+-------------+-----------+-----------------------------------------------
 10      Po10(RU)         LACP     Te0/4/0(bndl)
```

Configure Micro BFD on Port Channel

Procedure

Step 1 **enable**

Example:

```
Router> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 3 **bfd-template single-hop** *MicroBFD_template name*

Example:

```
Router(config)#bfd-template single-hop MicroBFD_template
```

Configures the BFD template and enters BFD configuration mode.

Step 4 **interval min-tx** *milliseconds* **min-rx** *milliseconds* **multiplier** *multiplier-value*

Example:

```
Router(bfd-config)#interval min-tx 100 min-rx 100 multiplier 3
```

Configures the transmit and receive intervals between BFD packets, and specifies the number of consecutive BFD control packets that must be missed before BFD declares that a peer is unavailable.

Step 5 **end**

Example:

```
Router(bfd-config)# end
```

Exits BFD configuration mode and returns the router to privileged EXEC mode.

Step 6 **configure terminal**

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 7 **Interface port channel** *number*

Example:

```
Router(config)#interface port channel 48
```

Configures the port channel and enters the interface mode.

Step 8 **port channel bfd destination ipv4** *ip address* **MicroBFD_template**

Example:

```
Router(config-if)#port-channel bfd destination ipv4 48.48.48.2 MicroBFD_template
```

Configures micro BFD under the port channel.

Use the no form of the command to delete the Micro BFD session from the port channel.

Step 9 **End**

Exits interface configuration mode and returns to global configuration mode.

Verify Micro BFD Session

Use the following commands to verify that the BFD sessions established are in UP state.

```
router#show etherchannel summary
Flags:  D - down        P/bndl - bundled in port-channel
        I - stand-alone s/susp - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator

        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port


Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
------+-------------+-----------+-----------------------------------------
------+-------------+-----------+--------
1       Po1(RU)         LACP     Gi0/3/2(bndl) Gi0/3/3(bndl)

RU - L3 port-channel UP State
SU - L2 port-channel UP state
P/bndl -  Bundled
S/susp  - Suspended


router#show bfd summary
Session          Up          Down

Total             8           8             0


router#show bfd neighbors
IPv4 Sessions
NeighAddr           LD/RD         RH/RS     State       Int
209.165.202.129   8/2148073631      Up        Up        Gi0/2/0
209.165.202.138   1028/2148073764   Up        Up        Gi0/2/1
209.165.202.139   45/108            Up        Up        Gi0/2/2
209.165.202.142   48/3              Up        Up        Te0/1/0
209.165.202.156   50/5              Up        Up        BD1000

IPv6 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
FE80::F278:16FF:FE85:CA91              46/1          Up        Up        Te0/1/0

Port Channel IPv4 Sessions
NeighAddr                                LD/RD         RH/RS     State     Int
 Parent Int
209.165.202.132                            53/112        Up        Up        Gi0/3/2
  Po1
209.165.202.132                            54/111        Up        Up        Gi0/3/3
  Po1
```

Configure Micro BFD on Port Channel with EFPs

Perform the following steps to enable Micro BFD sessions on port channels with Ethernet flow Point (EFP) or service instance configurations. For more information on EFPs, see Carrier Ethernet Configuration Guide. A pair of routing devices exchange micro BFD packets.

Note

- When the trunk EFP is configured on the port channel, micro BFD is supported only with the untagged EFP ID as source-service-instance.
- For static port channel, ensure that you must configure EFP with untagged encapsulation on the port channel interface.

This example shows the configuration on one router. Ensure that you configure the same steps on its peer.

Procedure

Step 1 **enable**

Example:

```
Router> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 3 **bfd-template single-hop** *MicroBFD_template name*

Example:

```
Router(config)#bfd-template single-hop MicroBFD_template
```

Configures the BFD template and enters BFD configuration mode.

Step 4 **interval min-tx** *milliseconds* **min-rx** *milliseconds* **multiplier** *multiplier-value*

Example:

```
Router(bfd-config)#interval min-tx 100 min-rx 100 multiplier 3
```

Defines the following values in the template: Transmit and receive intervals between BFD packets, and specifies the number of consecutive BFD control packets that must be missed before BFD declares that a peer is unavailable.

Step 5 **end**

Example:

```
Router(bfd-config)# end
```

Exits BFD configuration mode and returns the router to privileged EXEC mode.

Step 6 **configure terminal**

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 7 **interface port channel** *number*

Example:

```
Router(config)#interface port channel 10
```

Configures the port channel and enters the interface mode.

Step 8 **port channel bfd destination ipv4** *ip address* **MicroBFD_template source-service-instance** *number*

Example:

```
Router(config-if)#port-channel bfd destination ipv4 209.165.200.225 MicroBFD_template
source-service-instance 10
```

By configuring the service instance ID along with the Micro BFD configuration, you can enable Micro BFD sessions to handle Layer 2 packets. The specified IP address is the destination of the micro-BFD packet.

Use the no form of the command to delete the Micro BFD session from the port channel.

Step 9 **service instance** *number* **ethernet**

Example:

```
Router(config-if)#service instance 10 ethernet
```

Configures Ethernet for service instance 10.

Step 10 **encapsulation dot1q** *number*

Example:

```
Router(config-if-srv)#encapsulation untagged
```

Step 11 **l2protocol peer lacp**

Example:

```
Router(config-if-srv)#l2protocol peer lacp
```

Configures LACP to prevent route flapping of the port channel interface when configured with EVCs or service instances. LACP helps in maintaining network connectivity and does not degrade user experience.

Step 12 **bridge-domain** *number*

Example:

```
Router(config-if-srv)#bridge-domain 10
```

Configures the bridge domain.

Step 13 **End**

Exits interface configuration mode and returns to global configuration mode.

Step 14 **interface bdi** *number*

```
Router(config)#interface bdi 10
```

Specifies the bridge domain interface that is associated with the port-channel.

Step 15 **ip address** *ipv4 ipv6*

```
Router(config-if)#ip address 209.165.200.226 255.255.255.224
Router(config-if)#no shutdown
Router(config-if)#end
```

Assigning the IP address for the bridge domain interface. This IP address is the source of the BFD packet.

Verify Micro BFD on Port Channel with EFPs

Run the command to verify that the service instance is configured.

```
Router#show ethernet service instance id 10 interface port-channel 10 detail
Service Instance ID: 10
Service Instance Type: Static
Associated Interface: Port-channel10
Associated EVC:
Port-channel load-balance interface: None
L2protocol drop
CE-Vlans:

Encapsulation: dot1q 10 vlan protocol type 0x8100
Rewrite: ingress tag pop 1 symmetric
Interface Dot1q Tunnel Ethertype: 0x8100
State: Up
EFP Statistics:
Pkts In Bytes In Pkts Out Bytes Out
18         1980     46      3640
EFP Microblocks:

Microblock type: Bridge-domain
Bridge-domain: 10
Microblock type: L2Mcast
L2 Multicast GID: 1
Microblock type: dhcp_snoop
L2 Multicast GID: 1
```

Observe that the Layer 2 port-channel is in the UP state.

```
Router#show etherchannel summary
Flags: D - down P/bndl - bundled in port-channel
I - stand-alone s/susp - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use f - failed to allocate aggregator
    M - not in use, minimum links not met
    u - unsuitable for bundling
    w - waiting to be aggregated
    d - default port

Number of channel-groups in use: 2
Number of aggregators: 2
Group Port-channel Protocol Ports
------+-------------+-----------+-----------------------------------------------
10 Po10(SU) LACP Gi0/5/0(bndl) Gi0/5/1(bndl)
20 Po20(RU) LACP Gi0/5/3(bndl)
RU - L3 port-channel UP State
```

```
SU - L2 port-channel UP state
P/bndl - Bundled
S/susp - Suspended
```

Observe that the source and destination IP addresses of the routers that exchange micro BFD packets are displayed. The source IP address is the IP address of the BDI that is configured in the EFP where you configured the micro-BFD session.

```
Router#show bfd neighbors details

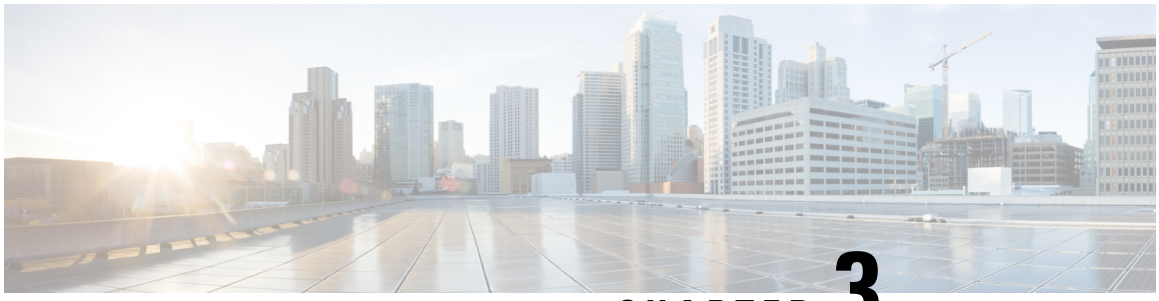
Port Channel IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int        Parent
Int
209.165.200.225                       1025/1025      Up        Up      Gi0/0/2         Po11

Session state is UP and not using echo function.
Session Host: Software
OurAddr: 209.165.200.226
Handle: 1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 200000, MinRxInt: 200000, Multiplier: 3
Received MinRxInt: 200000, Received Multiplier: 3
Holddown (hits): 0(0), Hello (hits): 200(3)
Rx Count: 11179007
Tx Count: 11179041
Elapsed time watermarks: 0 0 (last: 0)
Registered protocols: PochIPv4
Template: MicroBFD_template
Uptime: 2w5d
Last packet: Version: 1                  - Diagnostic: 0
             State bit: Up               - Demand bit: 0
             Poll bit: 0                 - Final bit: 0
             C bit: 1
             Multiplier: 3               - Length: 24
             My Discr.: 1                - Your Discr.: 1
             Min tx interval: 200000     - Min rx interval: 200000
             Min Echo interval: 0
```

The running configuration shows the configured values for the bfd destination and the source service instance.

```
Router#show running-config interface port-channel10
Building configuration...

Current configuration : 256 bytes
!
interface Port-channel10
 no ip address negotiation auto
 port-channnel bfd destination ipv4 209.165.200.225 source-service-instance 10
 service instance 10 ethernet
  encapsulation untagged
  l2protocol peer lacp
  bridge-domain 10
  end
```



CHAPTER 3

Static Route Support for BFD over IPv6

- Restrictions for Static Route for BFD over IPv6, on page 53
- Information About Static Route Support for BFD over IPv6, on page 53
- How to Configure Bidirectional Forwarding Detection for IPv6, on page 54
- Configuration Examples for Static Route Support for BFD over IPv6, on page 56

Restrictions for Static Route for BFD over IPv6

Multi-hop is *not* supported for static route.

Information About Static Route Support for BFD over IPv6

Using the BFDv6 protocol to reach the static route next hop ensures that an IPv6 static route is inserted only in the IPv6 Routing Information Base (RIB) when the next-hop neighbor is reachable. Using the BFDv6 protocol also can remove the IPv6 static route from the IPv6 RIB when the next hop becomes unreachable.

A user can configure IPv6 static BFDv6 neighbors. These neighbor can operate in one of two modes: associated (which is the default) and unassociated. A neighbor can be transitioned between the two modes without interrupting the BFDv6 session associated with the neighbor.

BFDv6 Associated Mode

In Bidirectional Forwarding Detection for IPv6 (BFDv6) associated mode, an IPv6 static route is automatically associated with an IPv6 static BFDv6 neighbor if the static route next hop exactly matches the static BFDv6 neighbor.

An IPv6 static route requests a BFDv6 session for each static BFDv6 neighbor that has one or more associated IPv6 static routes and is configured over an interface on which BFD has been configured. The state of the BFDv6 session will be used to determine whether the associated IPv6 static routes are inserted in the IPv6 RIB. For example, static routes are inserted in the IPv6 RIB only if the BFDv6 neighbor is reachable, and the static route is removed from the IPv6 RIB if the BFDv6 neighbor subsequently becomes unreachable.

BFDv6 associated mode requires you to configure a BFD neighbor and static route on both the device on which the BFD-monitored static route is required and on the neighboring device.

BFDv6 Unassociated Mode

An IPv6 static BFD neighbor may be configured as unassociated. In this mode, the neighbor is not associated with static routes, and the neighbor always requests a BFDv6 session if the interface has been configured for BFDv6.

Unassociated mode is useful in the following situations:

- Bringing up a BFDv6 session in the absence of an IPv6 static route—This case occurs when a static route is on router A, with router B as the next hop. Associated mode requires you to create both a static BFD neighbor and static route on both routers in order to bring up the BFDv6 session from B to A. Specifying the static BFD neighbor in unassociated mode on router B avoids the need to configure an unwanted static route.
- Transition to BFD monitoring of a static route—This case occurs when existing IPv6 static routes are inserted in the IPv6 RIB. Here, you want to enable BFD monitoring for these static routes without any interruption to traffic. If you configure an attached IPv6 static BFD neighbor, then the static routes will immediately be associated with the new static BFD neighbor. However, because a static BFD neighbor starts in a down state, the associated static routes are then removed from the IPv6 RIB and are reinserted when the BFDv6 session comes up. Therefore, you will see an interruption in traffic. This interruption can be avoided by configuring the static BFD neighbor as unassociated, waiting until the BFDv6 session has come up, and then reconfiguring the static BFD neighbor as associated.
- Transition from BFD monitoring of a static route—In this case, IPv6 static routes are monitored by BFD and inserted in the RIB. Here, you want to disable BFD monitoring of the static routes without interrupting traffic flow. This scenario can be achieved by first reconfiguring the static BFD neighbor as detached (thus disassociating the neighbor from the static routes) and then deconfiguring the static BFD neighbor.

How to Configure Bidirectional Forwarding Detection for IPv6

Specifying a Static BFDv6 Neighbor

An IPv6 static BFDv6 neighbor is specified separately from an IPv6 static route. An IPv6 static BFDv6 neighbor must be fully configured with the interface and neighbor address and must be directly attached to the local router.

Procedure

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 **ipv6 route static bfd** [**vrf** *vrf-name*] *interface-type interface-number ipv6-address* [**unassociated**]

Example:

```
Device(config)# ipv6 route static bfd gigabitethernet 0/0/0 2001:DB8::1
```

Specifies static route IPv6 BFDv6 neighbors.

Associating an IPv6 Static Route with a BFDv6 Neighbor

IPv6 static routes are automatically associated with a static BFDv6 neighbor. A static neighbor is associated with a BFDv6 neighbor if the static next-hop explicitly matches the BFDv6 neighbor.

Procedure

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 **ipv6 route static bfd** [**vrf** *vrf-name*] *interface-type interface-number ipv6-address* [**unassociated**]

Example:

```
Device(config)# ipv6 route static bfd gigabitethernet 0/0/0 2001:DB8::1
```

Specifies static route BFDv6 neighbors.

Step 4 **ipv6 route** [**vrf** *vrf-name*] *ipv6-prefix / prefix-length* {*ipv6-address* | *interface-type interface-number ipv6-address*} [**nexthop-vrf** [*vrf-name1* | **default**]] [*administrative-distance*] [*administrative-multicast-distance* | **unicast** | **multicast**] [*next-hop-address*] [**tag** *tag*]

Example:

```
Device(config)# ipv6 route 2001:DB8::/64 gigabitethernet 0/0/0 2001:DB8::1
```

Establishes static IPv6 routes.

Configuration Examples for Static Route Support for BFD over IPv6

Example: Specifying an IPv6 Static BFDv6 Neighbor

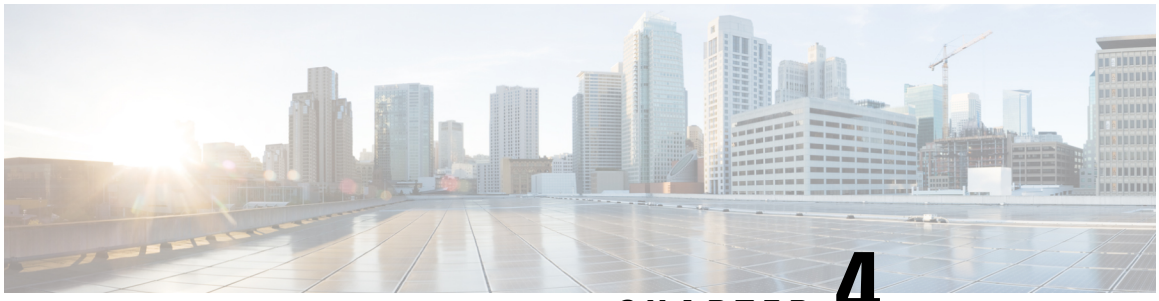
The following example specifies a fully configured IPv6 static BFDv6 neighbor. The interface is GigabitEthernet 0/0/0 and the neighbor address is 2001:DB8::1.

```
Device(config)# ipv6 route static bfd gigabitethernet 0/0/0 2001:DB8::1
```

Example: Associating an IPv6 Static Route with a BFDv6 Neighbor

In this example, the IPv6 static route 2001:DB8::/32 is associated with the BFDv6 neighbor 2001:DB8::1 over the GigabitEthernet 0/0/0 interface:

```
Device(config)# ipv6 route static bfd gigabitethernet 0/0/0 2001:DB8::1
Device(config)# ipv6 route 2001:DB8::/32 gigabitethernet 0/0/0 2001:DB8::1
```



CHAPTER 4

BFD on BDI Interfaces

The Cisco BFD on BDI Interfaces feature alleviates limitations on the maximum number of interfaces per system that switched virtual interfaces (SVI) impose. This document describes how to configure the Bidirectional Forwarding Detection (BFD) protocol on bridge domain interfaces (BDIs).

- Information About BFD on Bridge Domain Interfaces, on page 57
- How to Configure BFD on BDI Interfaces, on page 57
- Configuration Examples for BFD on BDI Interfaces, on page 61

Information About BFD on Bridge Domain Interfaces

BFD on Bridge Domain Interfaces

Each BDI is associated with a bridge domain on which traffic is mapped using criteria defined and configured on the associated Ethernet flow points (EFPs). You can associate either single or multiple EFPs with a given bridge domain. Thus you can establish a BFD single-hop session over BDI interfaces that are defined in either a global table or a VPN routing and forwarding (VRF) table, and all existing single-hop BFD clients will be supported for BFD over BDI.

The Cisco BFD on BDI feature does not affect BFD stateful switchover (SSO) on platforms that are SSO capable.

How to Configure BFD on BDI Interfaces

Enabling BFD on a Bridge Domain Interface

Perform these steps to enable single hop BFD on an individual BDI interface.

Note Multihop BFD is not interface specific so you do not need BDI interface-level configuration to establish multihop BFD sessions.

Before you begin

Two or more nodes must be connected.

Procedure

Step 1 **enable**

Example:

```
Router> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 3 **bfd-template single-hop** *template-name*

Example:

```
Router(config)# bfd-template single-hop bfdtemplate1
```

Creates a single-hop BFD template and enters BFD configuration mode.

Step 4 **interval min-tx** *milliseconds* **min-rx** *milliseconds* **multiplier** *multiplier-value*

Example:

```
Router(bfd-config)# interval min-tx 120 min-rx 100 multiplier 3
```

Configures the transmit and receive intervals between BFD packets, and specifies the number of consecutive BFD control packets that must be missed before BFD declares that a peer is unavailable.

Step 5 **interface** *type number*

Example:

```
Router(config)# interface bdi 100
```

Configures a bridge domain interface and enters interface configuration mode.

Step 6 **ip address** *ip-address mask*

Example:

```
Router(config-if)# ip address 10.201.201.1 255.255.255.0
```

Configures an IP address for the interface.

Step 7 **bfd template** *template name*

Enables the BFD template.

Step 8 **exit**

Example:

```
Router(config-if)# exit
```

Exits interface configuration mode and returns to global configuration mode.

Associating an Ethernet Flow Point with a Bridge Domain

Before you begin

BFD must be enabled on both nodes.

Procedure

Step 1 **enable**

Example:

```
Router> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 3 **interface** *type slot/subslot/port*

Example:

```
Router(config)# interface GigabitEthernet0/0/3
```

Configures an interface type and enters interface configuration mode.

Step 4 **no ip address**

Example:

```
Router(config-if)# no ip address
```

Disables IP processing.

Step 5 **negotiation auto**

Example:

```
Router(config-if)# negotiation auto
```

Enables the autonegotiation protocol to configure the speed, duplex, and automatic flow control of the interface.

Step 6 **cdp enable**

Example:

```
Router(config-if)# cdp enable
```

Enables Cisco Discovery Protocol on the interface.

Step 7 **service instance** *id service-type*

Example:

```
Router(config-if)# service instance 2 ethernet
```

Configures an Ethernet service instance and enters service instance configuration mode.

Step 8 **encapsulation dot1q** *vlan-id*

Example:

```
Router(config-if-srv)# encapsulation dot1q 2
```

Enables IEEE 802.1Q encapsulation of traffic on the subinterface.

Step 9 **rewrite ingress tag pop 1 symmetric**

Example:

```
Router(config-if-srv)# rewrite ingress tag pop 1 symmetric
```

Specifies removal of the outermost tag from the frame ingressing the service instance and the addition of a tag in the egress direction.

Step 10 **exit**

Example:

```
Router(config-if)# exit
```

Exits service instance configuration mode and returns to interface configuration mode.

Step 11 **exit**

Example:

```
Router(config-if)# exit
```

Exits interface configuration mode and returns to global configuration mode.

Step 12 **bridge-domain** *vlan-id*

Example:

```
Router(config)# bridge-domain 2
```

Associates the bridge domain with the Ethernet flow point.

Example:

What to do next

Configuration Examples for BFD on BDI Interfaces

Examples for BFD on BDI Interfaces

The following example shows how to configure BFD on a BDI.

```
Router#show bfd neighbors

IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
10.1.1.2                              2049/1         Up        Up        BD2
Router#
Router#show running interface gi0/0/3
Building configuration...

Current configuration : 230 bytes
!
interface GigabitEthernet0/0/3
no ip address
ip pim passive
ip igmp version 3
negotiation auto
cdp enable
service instance 2 ethernet
  encapsulation dot1q 2
  rewrite ingress tag pop 1 symmetric
  bridge-domain 2
!
end

Router#show running interface bdi2

Building configuration...

Current configuration : 127 bytes
!
interface BDI2
ip address 10.1.1.3 255.255.255.0
bfd interval 100 min_rx 100 multiplier 3
bfd neighbor ipv4 10.1.1.2
end
```

And similarly for the other node:

```
Router2#show running interface bdi2
```

```
Building configuration...

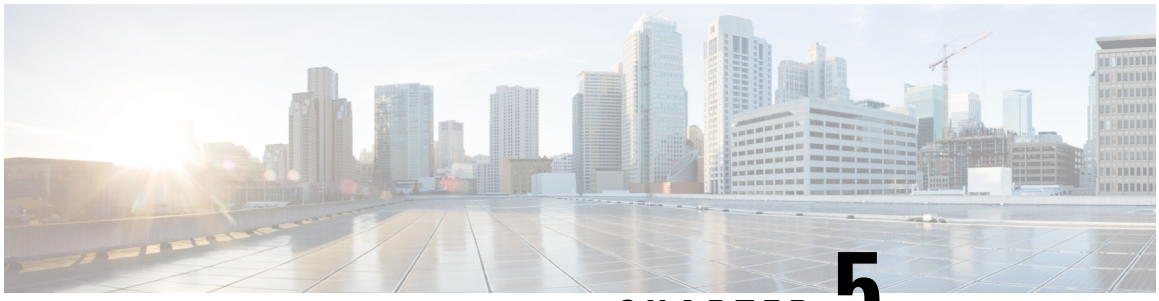
Current configuration : 127 bytes
!
interface BDI2
ip address 10.1.1.2 255.255.255.0
bfd interval 100 min_rx 100 multiplier 3
bfd neighbor ipv4 10.1.1.3
end

ED3#show run int gig0/0/3
Building configuration...

Current configuration : 195 bytes
!
interface GigabitEthernet0/0/3
no ip address
negotiation auto
cdp enable
service instance 2 ethernet
  encapsulation dot1q 2
  rewrite ingress tag pop 1 symmetric
  bridge-domain 2
!
end

Router2#show bfd neighbors

IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
10.1.1.3                                1/2049       Up        Up        BD2
ED3#
```



CHAPTER 5

BFD on IP Unnumbered Interfaces

This feature module describes how to enable BFD on IP unnumbered interfaces.

Note This feature is applicable only for Cisco ASR 900 RSP2 Module.

- Information about BFD on IP Unnumbered Interfaces, on page 63
- Prerequisites for BFD on IP Unnumbered Interfaces, on page 63
- Restrictions for BFD on IP Unnumbered Interfaces, on page 64
- Configuring BFD on IP Unnumbered Interfaces, on page 64
- Verifying BFD on IP Unnumbered Interface Configuration, on page 67
- Example: Configuration of BFD on IP Unnumbered Interfaces, on page 68
- Additional References, on page 73
- Feature Information for BFD on IP Unnumbered Interfaces, on page 73

Information about BFD on IP Unnumbered Interfaces

Cisco ASR 900 RSP2 Module supports BFD to run on IP unnumbered interfaces, which take the IP address from the loopback address. You can use the same loopback address on multiple interfaces. Physical, BDI, and Port-channel interfaces are supported for BFD on IP unnumbered interfaces. Single hop with authentication and Segment-routing (SR) are supported with BFD on IP unnumbered interfaces.

Prerequisites for BFD on IP Unnumbered Interfaces

- point-to-point keyword is mandatory for BFD on IP unnumbered interfaces:

  ip unnumbered Loopback <> poll point-to-point

- point-to-point keyword should be enabled on client protocols for ISIS and OSPF:

  ip ospf network point-to-point

  isis network point-to-point

Restrictions for BFD on IP Unnumbered Interfaces

- BFD on IP unnumbered interface is supported only with ISIS and OSPF.
- Echo mode is not supported.
- BFD on IP unnumbered interface is not supported for IPv6.
- BFD dampening is not supported.
- MHBFD is not supported for the clients of OSPF and ISIS.

Configuring BFD on IP Unnumbered Interfaces

Procedure

Step 1 **enable**

Example:

```
Device> enable
```

Enables privileged EXEC mode.

Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
router#configure terminal
```

Enters global configuration mode.

Step 3 **bfd-template single-hop** *template-name*

Example:

```
Router(config)# bfd-template single-hop bfdtemplate1
```

Creates a single-hop BFD template and enters BFD configuration mode.

Step 4 **interval min-tx** *milliseconds* **min-rx** *milliseconds* **multiplier** *multiplier-value*

Example:

```
Router(bfd-config)# interval min-tx 120 min-rx 100 multiplier 3
```

Configures the transmit and receive intervals between BFD packets, and specifies the number of consecutive BFD control packets that must be missed before BFD declares that a peer is unavailable.

Step 5 **interface loopback** *instance*

Example:

```
router#interface loopback 0
```

Creates a loopback interface and enter the interface config mode.

Step 6 **ip address** *address subnet mask*

Example:

```
router(conig-if)#ip address 21.21.21.21 255.255.255.255
```

Configures an IP address for this loopback interface.

Step 7 **ip router isis** *area-tag*

Example:

```
router(config-if)#ip router isis 1
```

Configures an IS-IS routing process for the IP on the configured interface and attaches an area designator to the routing process.

Step 8 **ip ospf** *process id* **area** *area id*

Example:

```
router(config-if)# ip ospf 1 area 0
```

Enables OSPFv2 on an interface.

Step 9 **interface gigabitethernet***slot/subslot/port*

Example:

```
router(config-if)#interface gigabitethernet0/0/3
```

Configures the interface type as gigabitethernet.

Step 10 **ip unnumbered Loopback** *interface* **poll point-to-point**

Example:

```
router(config-if)# ip unnumbered Loopback 0 poll point-to-point
```

Enables IP unnumbered processing on the loopback interface with point-to-point host polling.

Step 11 **no ip redirects**

Example:

```
router(config-if)# no ip redirects
```

Disables ICMP redirect messages.

Step 12 **ip router isis** *process tag*

Example:

```
router  (config-if)#ip router isis 1
```

Enable IS-IS as an IP routing protocol on the routers, and assign a tag to the process.

Step 13 **ip ospf network point-to-point**

Example:

```
router(config-if)#ip ospf network point-to-point
```

Configures an OSPFv2 point-to-point link on the interface.

Step 14 **ip ospf bfd**

Example:

```
router(config-if)#ip ospf bfd
```

Enables BFD on the OSPFv2 interface.

Step 15 **ip ospf** *process id* **area** *area id*

Example:

```
router(config-if)# ip ospf 1 area 0
```

Enables OSPFv2 on an interface.

Step 16 **isis network point-to-point**

Example:

```
router(config-if)#isis network point-to-point
```

Configures an IS-IS network into a point-to-point network.

Step 17 **bfd template** *template name*

Enables the BFD template.

Step 18 **router ospf** *process id*

Example:

```
router(config-if)#router ospf 1
```

Enables OSPF routing protocol on the router and attaches a process ID.

Step 19 **router-id** *router id*

Example:

```
router(config-router) router-id 209.165.201.0
```

Assigns the router ID on the OSPF router.

Step 20 **bfd all-interfaces strict-mode**

Example:

```
router(config-router)#bfd all-interfaces strict-mode
```

Enables BFD strict-mode for OSPF on all interfaces.

Step 21 **router isis** *area tag*

Example:

```
router(config-router)#router isis 1
```

Assigns a tag to an IS-IS process. Enters router configuration mode.

Step 22 **net** *network entity title*

Example:

```
router(config-router)#net 49.0001.0000.0000.0000.000a.00
```

Configures the NET on the device. The NET identifies the device for IS-IS.

Step 23 **is-type level-2-only**

Example:

```
router(config-router)#is-type level-2-only
```

Defines the routing level of the IS-IS routing process as level 2.

Step 24 **metric-style wide**

Example:

```
router(config-router)#metric-style wide
```

Configures a router running IS-IS so that it generates and accepts only new-style TLVs.

Step 25 **log-adjacency-changes**

Example:

```
router(config-router)#log-adjacency-changes
```

Configures the router to send a syslog message when the state of an OSPF neighbor changes.

Step 26 **bfd all-interfaces**

Example:

```
router(config-router)#bfd all-interfaces
```

Enables BFD for all of the interfaces for which IS-IS is routing.

Verifying BFD on IP Unnumbered Interface Configuration

To verify the BFD on IP unnumbered interface configuration, use the show commands given below:

```
Router#show bfd neighbors
IPv4 Sessions
 NeighAddr                LD/RD         RH/RS     State     Int
10.0.0.1                          1/3              Up        Up        Gi0/0/3
3.3.3.3                     1025/1025       Up        Up        Po10

Router#show bfd neighbors detail
IPv4 Sessions
NeighAddr         LD/RD         RH/RS     State     Int
10.0.0.1            1/3          Up        Up        Gi0/0/3
Session state is UP and not using echo function.
Session Host: Hardware
OurAddr: 21.21.21.21
Handle: 1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holddown (hits): 0(0), Hello (hits): 50(0)
Rx Count: 4304
Tx Count: 4306
Elapsed time watermarks: 0 0 (last: 0)
Registered protocols: OSPF CEF ISIS FRR
Uptime: 00:02:41
Last packet: Version: 1     - Diagnostic: 0
State bit: Up               - Demand bit: 0
Poll bit: 0                 - Final bit: 0
C bit: 1
Multiplier: 3               - Length: 24
My Discr.: 3                - Your Discr.: 1
```

```
Min tx interval: 50000      - Min rx interval: 50000
Min Echo interval: 0
```

Example: Configuration of BFD on IP Unnumbered Interfaces

Configuration on router 1:

```
key chain cisco
key 1
  key-string bfd
!
segment-routing mpls
!
set-attributes
  address-family ipv4
   sr-label-preferred
  exit-address-family
!
global-block 20000 30000
!
connected-prefix-sid-map
  address-family ipv4
   21.21.21.21/32 index 2222 range 1
   22.22.22.22/32 index 2223 range 1
   23.23.23.23/32 index 2224 range 1
  exit-address-family
!
bfd-template single-hop TEST
interval min-tx 500 min-rx 500 multiplier 3
authentication md5 keychain cisco

interface Loopback0
ip address 21.21.21.21 255.255.255.255
ip router isis 1
ip ospf 1 area 0
!
interface Loopback1
ip address 22.22.22.22 255.255.255.255
ip router isis 1
ip ospf 1 area 0
!
interface Loopback2
ip address 23.23.23.23 255.255.255.255
ip router isis 1
ip ospf 1 area 0
!
interface Port-channel10
ip unnumbered Loopback1 poll point-to-point
no ip redirects
ip router isis 1
ip ospf network point-to-point
ip ospf bfd
ip ospf 1 area 0
load-interval 30
negotiation auto
mpls ip
bfd template TEST
isis network point-to-point
!
interface GigabitEthernet0/0/3
ip unnumbered Loopback0 poll point-to-point
no ip redirects
```

```
ip router isis 1
ip ospf network point-to-point
ip ospf bfd
ip ospf 1 area 0
load-interval 30
negotiation auto
mpls ip
bfd interval 50 min_rx 50 multiplier 3
isis network point-to-point
!
router ospf 1
router-id 21.21.21.21
segment-routing mpls
segment-routing prefix-sid-map advertise-local
fast-reroute per-prefix enable prefix-priority low
fast-reroute per-prefix ti-lfa area 0
 fast-reroute keep-all-paths
bfd all-interfaces strict-mode
!
router isis 1
net 49.0001.0000.0000.0000.000a.00
is-type level-2-only
metric-style wide
log-adjacency-changes
segment-routing mpls
segment-routing prefix-sid-map advertise-local
fast-reroute per-prefix level-2 all
fast-reroute ti-lfa level-2
bfd all-interfaces
!
```

Configuration on router 2:

```
key chain cisco
key 1
  key-string bfd
!
segment-routing mpls
!
set-attributes
  address-family ipv4
   sr-label-preferred
  exit-address-family
!
global-block 20000 30000
!
connected-prefix-sid-map
  address-family ipv4
   10.0.0.1/32 index 1111 range 1
   2.2.2.2/32 index 1112 range 1
   3.3.3.3/32 index 1113 range 1
  exit-address-family
!
interface Loopback0
ip address 10.0.0.1 255.255.255.255
ip router isis 1
ip ospf 1 area 0
!
interface Loopback1
ip address 2.2.2.2 255.255.255.255
ip router isis 1
ip ospf 1 area 0
!
interface Loopback2
ip address 3.3.3.3 255.255.255.255
```

```
ip router isis 1
ip ospf 1 area 0
!
interface Port-channel10
ip unnumbered Loopback2 poll point-to-point
no ip redirects
ip router isis 1
ip ospf network point-to-point
ip ospf bfd
ip ospf 1 area 0
load-interval 30
negotiation auto
mpls ip
bfd template TEST
isis network point-to-point
!
interface GigabitEthernet0/0/0
ip unnumbered Loopback0 poll point-to-point
no ip redirects
ip router isis 1
ip ospf network point-to-point
ip ospf bfd
ip ospf 1 area 0
load-interval 30
negotiation auto
mpls ip
bfd interval 50 min_rx 50 multiplier 3
no bfd echo
isis network point-to-point
!
router ospf 1
router-id 10.0.0.1
segment-routing mpls
segment-routing prefix-sid-map advertise-local
fast-reroute per-prefix enable prefix-priority low
fast-reroute per-prefix ti-lfa area 0
 fast-reroute keep-all-paths
bfd all-interfaces strict-mode
!
router isis 1
net 49.0001.0000.0000.0000.000b.00
is-type level-2-only
metric-style wide
log-adjacency-changes
segment-routing mpls
segment-routing prefix-sid-map advertise-local
fast-reroute per-prefix level-2 all
fast-reroute ti-lfa level-2
bfd all-interfaces
!
sh bfd neighbors

IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
11.11.11.11                              1/2         Up        Up        Gi0/0/3
12.12.12.12                          1026/1025       Up        Up        Gi0/0/4
13.13.13.13                              2/1         Up        Up        BD111
21.21.21.21                              3/1         Up        Up        Gi0/0/0
22.22.22.22                          1025/1025       Up        Up        Po10
BGL-PEGASUS#
sh bfd neighbors detail

IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
```

```
11.11.11.11                               1/2          Up        Up        Gi0/0/3
Session state is UP and not using echo function.
Session Host: Hardware
OurAddr: 10.0.0.1
Handle: 3
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holddown (hits): 0(0), Hello (hits): 50(0)
Rx Count: 2392
Tx Count: 2413
Elapsed time watermarks: 0 0 (last: 0)
Registered protocols: OSPF CEF ISIS FRR
Uptime: 00:01:46
Last packet: Version: 1                  - Diagnostic: 0
             State bit: Up               - Demand bit: 0
             Poll bit: 0                 - Final bit: 0
             C bit: 1
             Multiplier: 3               - Length: 24
             My Discr.: 2                - Your Discr.: 1
             Min tx interval: 50000      - Min rx interval: 50000
             Min Echo interval: 50000

IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
12.12.12.12                          1026/1025       Up        Up        Gi0/0/4
Session state is UP and not using echo function.
Session Host: Software
OurAddr: 2.2.2.2
Handle: 4
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 500000, MinRxInt: 500000, Multiplier: 3
Received MinRxInt: 500000, Received Multiplier: 3
Holddown (hits): 1296(0), Hello (hits): 500(331)
Rx Count: 332, Rx Interval (ms) min/max/avg: 1/504/440 last: 204 ms ago
Tx Count: 332, Tx Interval (ms) min/max/avg: 1/504/439 last: 320 ms ago
Elapsed time watermarks: 0 0 (last: 0)
Registered protocols: OSPF CEF FRR ISIS
Template: TEST
Authentication(Type/Keychain): md5/cisco
last_tx_auth_seq: 4  last_rx_auth_seq 6
Uptime: 00:02:25
Last packet: Version: 1                  - Diagnostic: 0
             State bit: Up               - Demand bit: 0
             Poll bit: 0                 - Final bit: 0
             C bit: 0
             Multiplier: 3               - Length: 48
             My Discr.: 1025             - Your Discr.: 1026
             Min tx interval: 500000     - Min rx interval: 500000
             Min Echo interval: 0

IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
13.13.13.13                             2/1          Up        Up        BD111
Session state is UP and not using echo function.
Session Host: Hardware
OurAddr: 3.3.3.3
Handle: 5
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 300000, MinRxInt: 300000, Multiplier: 3
Received MinRxInt: 300000, Received Multiplier: 3
Holddown (hits): 0(0), Hello (hits): 300(0)
Rx Count: 502
Tx Count: 504
```

```
Elapsed time watermarks: 0 0 (last: 0)
Registered protocols: OSPF CEF FRR ISIS
Uptime: 00:02:22
Last packet: Version: 1                  - Diagnostic: 0
             State bit: Up               - Demand bit: 0
             Poll bit: 0                 - Final bit: 0
             C bit: 1
             Multiplier: 3               - Length: 24
             My Discr.: 1                - Your Discr.: 2
             Min tx interval: 300000     - Min rx interval: 300000
             Min Echo interval: 0

IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
21.21.21.21                             3/1          Up        Up        Gi0/0/0
Session state is UP and not using echo function.
Session Host: Hardware
OurAddr: 10.0.0.1
Handle: 1
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 50000, MinRxInt: 50000, Multiplier: 3
Received MinRxInt: 50000, Received Multiplier: 3
Holddown (hits): 0(0), Hello (hits): 50(0)
Rx Count: 4305
Tx Count: 4323
Elapsed time watermarks: 0 0 (last: 0)
Registered protocols: OSPF CEF ISIS FRR
Uptime: 00:02:59
Last packet: Version: 1                  - Diagnostic: 0
             State bit: Up               - Demand bit: 0
             Poll bit: 0                 - Final bit: 0
             C bit: 1
             Multiplier: 3               - Length: 24
             My Discr.: 1                - Your Discr.: 3
             Min tx interval: 50000      - Min rx interval: 50000
             Min Echo interval: 50000

IPv4 Sessions
NeighAddr                              LD/RD         RH/RS     State     Int
22.22.22.22                          1025/1025       Up        Up        Po10
Session state is UP and not using echo function.
Session Host: Software
OurAddr: 3.3.3.3
Handle: 2
Local Diag: 0, Demand mode: 0, Poll bit: 0
MinTxInt: 500000, MinRxInt: 500000, Multiplier: 3
Received MinRxInt: 500000, Received Multiplier: 3
Holddown (hits): 1232(0), Hello (hits): 500(393)
Rx Count: 393, Rx Interval (ms) min/max/avg: 4/504/440 last: 268 ms ago
Tx Count: 396, Tx Interval (ms) min/max/avg: 4/504/437 last: 268 ms ago
Elapsed time watermarks: 0 0 (last: 0)
Registered protocols: CEF OSPF FRR ISIS
Template: TEST
Authentication(Type/Keychain): md5/cisco
last_tx_auth_seq: 5  last_rx_auth_seq 4
Uptime: 00:02:53
Last packet: Version: 1                  - Diagnostic: 0
             State bit: Up               - Demand bit: 0
             Poll bit: 0                 - Final bit: 0
             C bit: 0
             Multiplier: 3               - Length: 48
             My Discr.: 1025             - Your Discr.: 1025
             Min tx interval: 500000     - Min rx interval: 500000
             Min Echo interval: 0
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS XE Command Reference	Cisco IOS IP Routing: Protocol-Independent Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 5880	Bidirectional Forwarding Detection (BFD)

MIBs

MIB	MIBs Link
NA	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

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

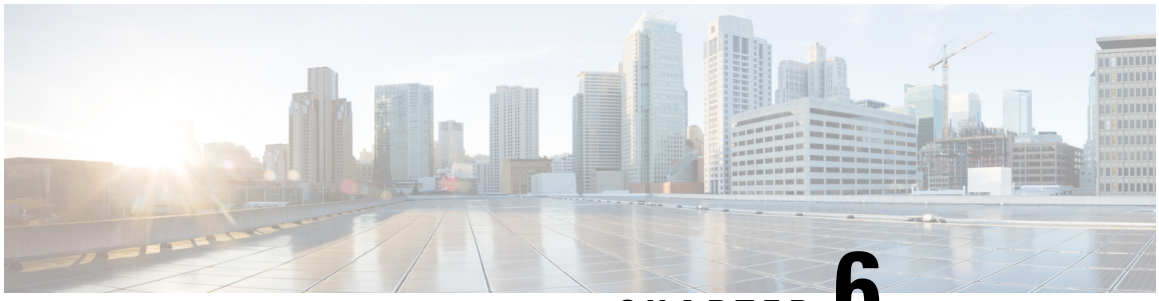
Feature Information for BFD on IP Unnumbered Interfaces

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 https://cfnng.cisco.com/. An account on Cisco.com is not required.

Table 6: Feature Information for BFD on IP Unnumbered Interfaces

Feature Name	Releases	Feature Information
BFD on IP Unnumbered Interfaces	Cisco IOS XE Everest 16.5.1	In Cisco IOS XE Everest 16.5.1, BFD is supported to run on IP unnumbered interfaces, which take the IP address from the loopback address. You can use the same loopback address on multiple interfaces. In Cisco IOS XE Everest 16.5.1, support was added for the Cisco ASR 903 router.



CHAPTER 6

IS-IS Client for BFD

When Bidirectional Forwarding Detection (BFD) support is configured with Intermediate System To Intermediate System (IS-IS) as a registered protocol with BFD, IS-IS receives forwarding path detection failure messages from BFD.

- Prerequisites for IS-IS Client for BFD, on page 75
- Restrictions for IS-IS Client for BFD, on page 75
- Information About IS-IS Client for BFD, on page 75
- How to Configure ISIS IPv6 Client for BFD, on page 76
- Configuration Examples for ISIS IPv6 Client for BFD, on page 79

Prerequisites for IS-IS Client for BFD

- IS-IS must be running on all participating devices.
- The baseline parameters for BFD sessions must be configured on the interfaces that run BFD sessions to BFD neighbors.

Restrictions for IS-IS Client for BFD

Only one IS-IS IPv6 session is supported.

Information About IS-IS Client for BFD

IS-IS BFD Topology

When BFD support is configured with IS-IS as a registered protocol with BFD, IS-IS receives forwarding path detection failure messages from BFD. BFD support for IS-IS can be configured in either router address-family configuration mode or interface configuration mode. IS-IS IPv6 can run in single-topology or in Multi-Topology (MT) mode.

IS-IS BFD supports both IPv4 and IPv6 on the same adjacency for single-topology or multi-topology mode. If BFD is enabled for both IPv4 and IPv6, IS-IS sends two BFD session creation requests to BFD. For single-topology mode, the IS-IS adjacency state can only be UP if both BFD sessions are UP. If either of the

BFD sessions is DOWN, the associated IS-IS adjacency state is also DOWN. For MT mode, the IS-IS adjacency state can be UP as long as one of topologies has a BFD session in an UP state.

IS-IS BFD Session Creation

IS-IS requests a BFD session for the interface and IP address of the neighboring device when all of the following conditions are met:

- An IS-IS adjacency entry exists.
- The Address Family Identifier (AFI) specific peer interface address is known.
- IS-IS BFD is enabled for that AFI on an interface.
- IS-IS is enabled for that AFI on the local interface.
- If the neighboring device supports RFC 6213, BFD must be enabled for the specified Multi-Topology Identifier (MTID) or Network Layer Protocol Identifier (NLPID).

IS-IS BFD IPv6 Session Deletion

When IS-IS BFD IPv6 is disabled on an interface, IS-IS removes related BFD sessions for IPv6 from the adjacent device. When the IS-IS adjacency entry is deleted, all BFD sessions are also deleted. IS-IS requests BFD to remove each BFD session that it has requested when any of the following events occur:

- The IS-IS instance is deleted or un-configured.
- The IS-IS adjacency entry is deleted.
- IS-IS BFD is disabled on the next hop interface for an address-family.
- The neighboring device supports RFC 6213 and indicates that it no longer supports BFD for the specified MTID or NLPID.

How to Configure ISIS IPv6 Client for BFD

Configuring IS-IS IPv6 Client Support for BFD on an Interface

Procedure

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 **interface** *type number*

Example:

```
Device(config)# interface gigabitethernet 6/0/0
```

Enters interface configuration mode.

Step 4 **ipv6 address** *ipv6-address/mask*

Example:

```
Device(config-if)# ipv6 address 2001:DB8::1/32
```

Configures IPv6.

Step 5 **isis ipv6 bfd**

Example:

```
Device(config-if)# isis ipv6 bfd
```

Enables IPv6 BFD on a specific interface that is configured for IS-IS.

Step 6 **end**

Example:

```
Device(config-if)# end
```

Exits interface configuration mode and returns to privileged EXEC mode.

Configuring IS-IS IPv6 Client Support for BFD on All Interfaces

Procedure

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 **router isis**

Example:

```
Device(config)# router isis
```

Enables the IS-IS routing protocol and enters router configuration mode.

Step 4 **metric-style wide**

Example:

```
Device(config-router)# metric-style wide
```

(Optional) Configures a device that is running IS-IS so that it generates and accepts only new-style type, length, value objects (TLVs).

Step 5 **address-family ipv6**

Example:

```
Device(config-router)# address-family ipv6
```

Enters address family configuration mode for configuring IS-IS routing sessions that use standard IPv6 address prefixes.

Step 6 **multi-topology**

Example:

```
Device(config-router-af)# multi-topology
```

(Optional) Enables multi-topology IS-IS for IPv6.

Step 7 **bfd all-interfaces**

Example:

```
Device(config-router-af)# bfd all-interfaces
```

Enables BFD for all interfaces participating in the routing process.

Step 8 **end**

Example:

```
Device(config-router-af)# end
```

Exits address family configuration mode and returns to privileged EXEC mode.

Configuration Examples for ISIS IPv6 Client for BFD

Example: IS-IS IPv6 Client Support for BFD on a Single Interface

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 6/0/0
Device(config-if)# ipv6 address 2001:DB8::1/32
Device(config-if)# isis ipv6 bfd
Device(config-if)# end
```

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 6/0
Device(config-if)# ipv6 address 2001:DB8::1/32
Device(config-if)# isis ipv6 bfd
Device(config-if)# end
```

Example: IS-IS IPv6 Client Support for BFD on All Interfaces

```
Device> enable
Device# configure terminal
Device(config)# router isis
Device(config-router)# metric-style wide
Device(config-router)# address-family ipv6
Device(config-router-af)# multi-topology
Device(config-router-af)# bfd all-interfaces
Device(config-router-af)# end
```

The following is a sample configuration where interface 0/0/7 of Router A is connected to interface 0/4/6 of router B.

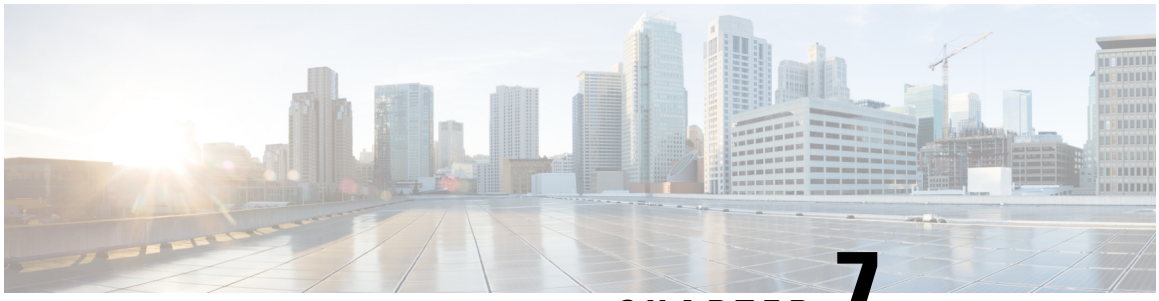
Configuration for Router A

```
bfd-template single-hop BFDM
 interval min-tx 50 min-rx 50 multiplier 3
!
interface TenGigabitEthernet0/0/7
 ipv6 address 2001:DB8::1/32
 ipv6 router isis
 bfd template BFDM
 isis ipv6 bfd
!
router isis
 net 49.0001.1720.1600.1001.00
!
```

Configuration on Router B

```
Router B
```

```
bfd-template single-hop BFDM
 interval min-tx 50 min-rx 50 multiplier 3
!
interface TenGigabitEthernet0/4/6
 ipv6 address 2001:DB8::1/32
 ipv6 router isis
 bfd template BFDM
 isis ipv6 bfd
!
router isis
 net 49.0000.0000.0002.00
!
!
```



CHAPTER 7

BFD IPv6 Encapsulation Support

Bidirectional Forwarding Detection for IPv6 encapsulations are described within a session information structure. These session information structures are defined by BFDv6 for the protocols supported. BFDv6 uses information from the session information structures to determine the correct encapsulation for BFDv6 packets on that session.

- Prerequisites for BFD IPv6 Encapsulation Support, on page 81
- Restrictions for BFD IPv6 Encapsulation Support, on page 81
- Information About BFD IPv6 Encapsulation Support, on page 82
- How to Configure BFD IPv6 Encapsulation Support, on page 83
- Configuration Examples for BFD IPv6 Encapsulation Support, on page 84

Prerequisites for BFD IPv6 Encapsulation Support

- When using Bidirectional Forwarding Detection over IPv6 (BFDv6), IPv6 Cisco Express Forwarding and IPv6 unicast routing must be enabled on all participating routers.
- When you configure BFD IPv6 software sessions, you should configure the following CLI command:

  no ipv6 nd nud igp

  If the peer is also an ASR device, the above command should be configured on the peer device too.

Restrictions for BFD IPv6 Encapsulation Support

- BFDv6 supports only global IPv6 neighbor addresses if a global IPv6 address is configured on the interface.
- Only asynchronous mode is supported. In asynchronous mode, either BFDv6 peer can initiate a BFDv6 session.

Information About BFD IPv6 Encapsulation Support

Overview of the BFDv6 Protocol

This section describes the BFDv6 protocol, how it is different from BFD for IPv4, and how it works with BFD for IPv4. BFD is a detection protocol designed to provide fast forwarding path failure detection times for all media types, encapsulations, topologies, and routing protocols. In addition to fast forwarding path failure detection, BFD provides a consistent failure detection method for network administrators. BFDv6 provides IPv6 support by accommodating IPv6 addresses and provides the ability to create BFDv6 sessions.

BFDv6 Registration

BFD clients register with BFD using a registry application program interface (API). The registry arguments include protocol type and the address and interface description block (IDB) of the route to be monitored. These APIs and arguments are all assumed by BFD to be IPv4.

BFDv6 has registries from which these arguments have been removed, and the protocol and encapsulation are described within a session information structure. These session information structures are defined by BFDv6 for the protocols supported. BFDv6 uses information from the session information structures to determine the correct encapsulation for BFDv6 packets on that session.

BFDv6 Global and Link-Local Addresses

BFDv6 supports both global and link-local IPv6 addresses for neighbor creation. BFDv6 sessions select source addresses to match the neighbor address types (for example, global IPv6 address neighbors must be paired with global IPv6 source addresses and link-local IPv6 address neighbors must be paired with link-local IPv6 source addresses). The table below shows the address pairings that BFDv6 supports.

Table 7: BFDv6 Address Pairings for Neighbor Creation

Source Address	Destination Address	Status
Global	Global	Supported
Global	Link local	Not supported
Link local	Global	Not supported
Link local	Link local	Supported

Because all IPv6-enabled interfaces have a link-local address and BFDv6 selects the source address, link-local address neighbors are always paired with a link-local interface address. The link-local source address with global destination address is not supported by Cisco Express Forwarding. Therefore, a global IPv6 address must be configured on an interface before a session with a global address neighbor may be established in BFDv6. BFDv6 rejects any sessions in which the neighbor address is global and no global address is configured on the interface.

Note The behavior of a unique local address (ULA) in BFDv6 is the same as a global address.

BFD for IPv4 and IPv6 on the Same Interface

BFD supports multiple IPv4 and IPv6 sessions per interface, with no restriction on the protocol of those sessions.

How to Configure BFD IPv6 Encapsulation Support

Configuring BFD Session Parameters on the Interface

The steps in this procedure show how to configure BFD on the interface by setting the baseline BFD session parameters on an interface. Repeat the steps in this procedure for each interface over which you want to run BFD sessions to BFD neighbors.

Note RSP3 Module eysupports only the following BFD interval timers:

3.3ms, 6.6ms, 10ms, 20ms, 50ms, 100ms, 200ms ,999ms. It is recommended that peer should also configure the same timer values.

Procedure

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 **bfd-template single-hop** *template-name*

Example:

```
Router(config)# bfd-template single-hop bfdtemplate1
```

Creates a single-hop BFD template and enters BFD configuration mode.

Step 4 **interval min-tx** *milliseconds* **min-rx** *milliseconds* **multiplier** *multiplier-value*

Example:

```
Router(bfd-config)# interval min-tx 120 min-rx 100 multiplier 3
```

Configures the transmit and receive intervals between BFD packets, and specifies the number of consecutive BFD control packets that must be missed before BFD declares that a peer is unavailable.

Step 5 **interface gigabitethernet** *number*

Example:

```
Device(config)# interface gigabitethernet 0/0/0
```

Specifies the Gigabit Ethernet interface and enters interface configuration mode.

Step 6 Perform one of the following steps:

- **ip address** *ipv4-address mask*
- **ipv6 address** *ipv6-address*/*mask*

Example:

Configuring an IPv4 address for the interface:

```
Device(config-if)# ip address 10.201.201.1 255.255.255.0
```

Configuring an IPv6 address for the interface:

```
Device(config-if)# ipv6 address 2001:DB8::/32
```

Configures an IP address for the interface.

Step 7 **bfd template** *template name*

Enables the BFD template.

Step 8 **end**

Example:

```
Device(config-if)# end
```

Exits interface configuration mode and returns to privileged EXEC mode.

Configuration Examples for BFD IPv6 Encapsulation Support

Example: Configuring BFD Session Parameters on the Interface

```
Device# show ipv6 ospf neighbor detail

 Neighbor 172.16.3.3
```

```
    In the area 1 via interface GigabitEthernet0/0/0
    Neighbor: interface-id 3, link-local address FE80::205:5FFF:FED3:5808
    Neighbor priority is 1, State is FULL, 6 state changes
    DR is 172.16.6.6 BDR is 172.16.3.3
    Options is 0x63F813E9
    Dead timer due in 00:00:33
    Neighbor is up for 00:09:00
    Index 1/1/2, retransmission queue length 0, number of retransmission 2
    First 0x0(0)/0x0(0)/0x0(0) Next 0x0(0)/0x0(0)/0x0(0)
    Last retransmission scan length is 1, maximum is 2
    Last retransmission scan time is 0 msec, maximum is 0 msec
```

The following is a sample configuration where interface 0/0/7 of Router A is connected to interface 0/4/6 of router B.

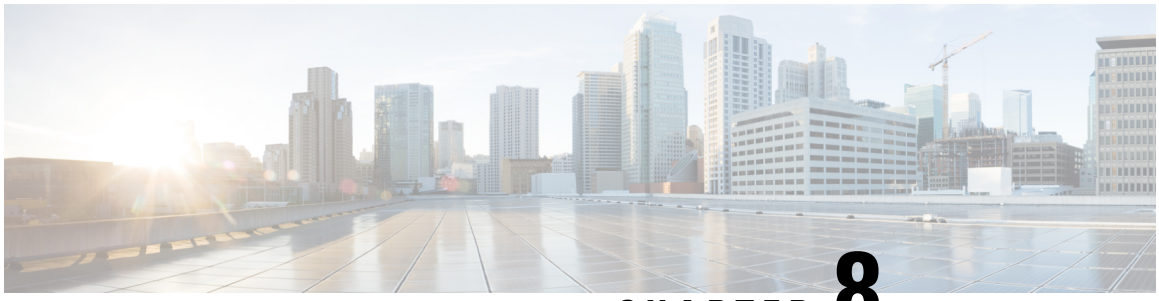
Configuration on Router A

```
bfd-template single-hop BFDM
interval min-tx 50 min-rx 50 multiplier 3

router ospfv3 1
router-id 2.2.2.2
!
address-family ipv6 unicast
  bfd all-interfaces
exit-address-family
!
!
interface TenGigabitEthernet0/0/7
ipv6 address 19:1:1::1/64
ospfv3 1 ipv6 area 0
bfd template BFDM
```

Configuration on Router B

```
bfd-template single-hop BFDM
interval min-tx 50 min-rx 50 multiplier 3
!
interface TenGigabitEthernet0/4/6
ipv6 address 19:1:1::2/64
ospfv3 1 ipv6 area 0
bfd template BFDM
end
!
router ospfv3 1
router-id 3.3.3.3
!
address-family ipv6 unicast
  bfd all-interfaces
exit-address-family
```



CHAPTER 8

Configuring BFD Debug Enhancement

The Bidirectional Forwarding Detection (BFD) Debug Enhancement feature enables logging of debugging information for critical BFD events, normal BFD events, and BFD packets. This feature enables BFD event traces and BFD event logs. This feature allows network engineers and operaters to easily identify and analyze issues with BFD sessions.

- Prerequisites for BFD Debug Enhancement, on page 87
- Restrictions for BFD Debug Enhancement, on page 87
- Information About BFD Debug Enhancement, on page 87
- How to Configure BFD Debug Enhancement, on page 88
- Configuration Examples for BFD Debug Enhancement, on page 90

Prerequisites for BFD Debug Enhancement

Bidirectional Forwarding Detection (BFD) feature must be running on the device.

Restrictions for BFD Debug Enhancement

The Bidirectional Forwarding Detection (BFD) event trace logs cannot store more than 65536 elements in a BFD instance. Once the log elements reach this limit, new log entries replace the old log entries.

RSP3 do not support the Bidirectional Forwarding Detection (BFD) debug enhancement.

Information About BFD Debug Enhancement

Overview of BFD Debug Enhancement

The Bidirectional Forwarding Detection (BFD) Debug Enhancement feature enables logging of debugging information for critical BFD events, normal BFD events, and BFD packets. This feature enables BFD event traces and BFD event logs.

BFD Debug Enhancement - Event Tracer subsystem

The Event Tracer subsystem in the BFD Debug Enhancement feature helps network engineers trace BFD events. This subsystem generates debugging information that can be extracted and analyzed. You can optionally store the traced data in a file for further analysis.

BFD Debug Enhancement - Event Log Architecture

The Event Log Architecture in the BFD Debug Enhancement feature is classified into three types:

- BFD Critical—Logs debugging information about all critical BFD event traces, whenever the BFD session attains the DOWN state.
- BFD Event—Logs debugging information about all BFD events that can be logged using the **debug bfd event** command.
- BFD Packet—Logs debugging information about all BFD packets that can be logged using the **debug bfd packet** command.

How to Configure BFD Debug Enhancement

Configuring BFD Debug Enhancement

Procedure

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 **bfd-template single-hop** *template-name*

Example:

```
Router(config)# bfd-template single-hop bfdtemplate1
```

Creates a single-hop BFD template and enters BFD configuration mode.

Step 4 **interval min-tx** *milliseconds* **min-rx** *milliseconds* **multiplier** *multiplier-value*

Example:

```
Router(bfd-config)# interval min-tx 120 min-rx 100 multiplier 3
```

Configures the transmit and receive intervals between BFD packets, and specifies the number of consecutive BFD control packets that must be missed before BFD declares that a peer is unavailable.

Step 5 **interface gigabitethernet** *number*

Example:

```
Device(config)# interface gigabitethernet 0/0/0
```

Specifies the Gigabit Ethernet interface and enters interface configuration mode.

Step 6 **bfd template** *template name*

Enables the BFD template.

Step 7 **end**

Example:

```
Device(config-if)# end
```

Exits interface configuration mode and returns to privileged EXEC mode.

Step 8 **debug bfd event**

Example:

```
Device# debug bfd event
```

Enables debugging information for BFD events.

Step 9 **debug bfd packet**

Example:

```
Device# debug bfd packet
```

Enables debugging information for BFD packets.

Step 10 **monitor event-trace bfd event enable**

Example:

```
Device# monitor event-trace bfd event enable
```

Enables the event traces for normal BFD events.

Step 11 **monitor event-trace bfd event all enable**

Example:

```
Device# monitor event-trace bfd event all enable
```

Enables the event trace filters for normal BFD events.

Step 12 **monitor event-trace bfd packet enable**

Example:

```
Device# monitor event-trace bfd packet enable
```

Enables the event traces for BFD packet events.

Step 13 **monitor event-trace bfd packet all enable**

Example:

```
Device# monitor event-trace bfd packet all enable
```

Enables the event trace filters for BFD packet events.

Step 14 **show monitor event-trace bfd**

Example:

```
Device# show monitor event-trace bfd
```

Displays event trace messages for BFD events. Use the **critical**, **event**, and **packet** keywords to display event trace messages for critical BFD events, normal BFD events, and BFD packet events, respectively.

Configuration Examples for BFD Debug Enhancement

Example: Configuring BFD Debug Enhancement

```
Device> enable
Device# configure terminal
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# bfd interval 50 min_rx 50 multiplier 5
Device(config-if)# end
Device# debug bfd event
Device# debug bfd packet
Device# monitor event-trace bfd event enable
Device# monitor event-trace bfd event all enable
Device# monitor event-trace bfd packet enable
Device# monitor event-trace bfd packet all enable
Device# show monitor event-trace bfd critical all
Device# show monitor event-trace bfd event all
Device# show monitor event-trace bfd packet all
```

The following is sample output from the **show monitor event-trace bfd** command for a critical BFD event:

```
Device# show monitor event-trace bfd critical latest

Oct  2 10:57:25.173: BFD-DEBUG CRITICAL: V1 FSM ld:1 handle:1
Event:DETECT TIMER EXPIRED state:UP Rx Count:123 Tx Count: 120
Control Packets Rx:[Oct2 10:57:25.173][Oct2 10:57:25.173][Oct2 10:57:25.173]
Control Packet Tx:[Oct2 10:57:26.173][Oct2 10:57:27.173][Oct2 10:57:28.173]
Echo Rx:[-][-][-]
Echo Tx:[-][-][-]

Oct  2 10:57:25.173: BFD-DEBUG CRITICAL: V1 FSM ld:1 handle:1
Event:ECHO FAILURE state:UP Rx Count:123 Tx Count: 120
Control Packet Rx:[Oct2 10:57:25.173][Oct2 10:57:25.173][Oct2 10:57:25.173]
Control Packet Tx:[Oct2 10:57:26.173][Oct2 10:57:27.173][Oct2 10:57:28.173]
Echo Rx:[Oct2 10:57:25.273][Oct2 10:57:25.373][Oct2 10:57:25.473]
Echo Tx:[Oct2 10:57:25.273][Oct2 10:57:25.373][Oct2 10:57:25.473]
```

The following is sample output from the **show monitor event-trace bfd** command for a normal BFD event:

Note Logs all events that are logged using the **debug bfd event** command.

```
Device# show monitor event-trace bfd event latest

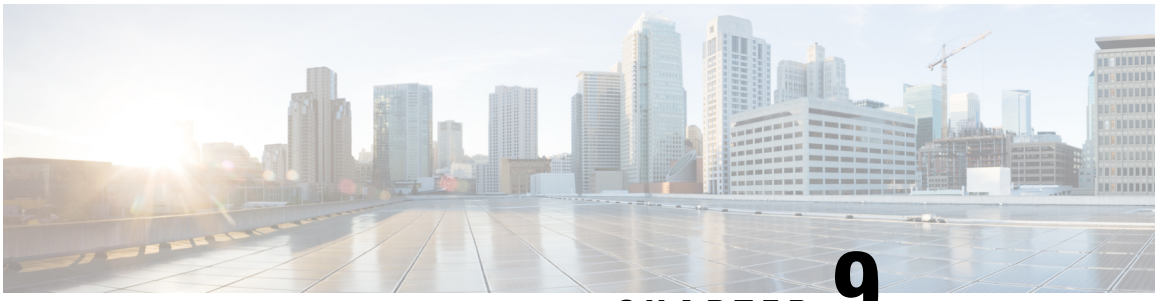
*Oct 13 20:31:17.043: BFD-DEBUG Event: V1 FSM ld:1 handle:1 event:RX UP state:UP (0)
*Oct 13 20:31:16.945: BFD-DEBUG Event: V1 FSM ld:1 handle:1 event:RX INIT state:DOWN (0)
*Oct 13 20:31:16.150: BFD-DEBUG EVENT: bfd_session_created, proc:EIGRP, idb:Ethernet0/0
handle:1 act
*Oct 13 20:31:14.633: BFD-DEBUG Event: V1 FSM ld:1 handle:1 event:DETECT TIMER EXPIRED
state:ADMIN DOWN (0)
```

The following is sample output from the **show monitor event-trace bfd** command for a BFD packet event:

Note Logs all packet events that are logged using the **debug bfd packet** command.

```
Device# show monitor event-trace bfd packet latest

*Oct 12 05:30:46.849: BFD-DEBUG Packet: Tx IP:10.1.1.2 ld/rd:1/0
diag:7(Administratively Down) AdminDown   cnt:21 (0)
```



CHAPTER 9

Bidirectional Forwarding Detection Configuration of Deterministic Hardware Offload

This chapter provides information about configuring the Bidirectional Forwarding Detection (BFD) deterministic hardware offload feature on the router.

Prerequisites for BFD Deterministic Hardware Offload

- This feature must be deployed only after identifying a downtime.

Restrictions for BFD Deterministic Hardware Offload

- For existing BFD sessions (before using a new configuration command) to be made deterministic, you must recreate the sessions using the **shutdown** or **no shutdown** command, or reconfigure them.
- BFD echo with uRPF is *not* supported on the router for both hardware and software based BFD sessions.
- Echo based sessions—Maximum supported in hardware is 255.
- Non-echo based sessions—Maximum supported in hardware is 511.
- For BFD over G8032, only BFD IPv4 and IPv6 Single Hop is supported. BFD Echo Mode is not supported.
- If the hardware offloaded BFD session reaches the maximum hardware limit per system, then a new BFD session with BFD interval of more than 50 msec is created as a software session. No new BFD sessions are created for less than a 50 msec BFD interval.

Hardware offload BFD Echo Sessions on RSP3 Module

- Packets may get punted to CPU for hardware offload per sessions, when the BFD template is removed and added. This issue occurs on the RSP3 module with 1K BFD sessions using the same template. Some software BFD sessions might flap due to this issue.
- Hardware offloaded BFD echo packets are not punted to the ICMP queue, hence ICMP redirects are not generated.

Table 8: Feature History

Feature Name	Feature Release	Description
BFD over G8032 and Multi EFP BDI	Cisco IOS XE Bengaluru 17.4.1	Hardware limit for BFD and hardware offload are enhanced for BFD over G8032.

Hardware Limits for Cisco RSP3 module for BFD over G8032

	Hardware Offload	
BFD	**RSP3**	**RSP2**
IPv4	64	1023
IPv6	63	1023
IPv4 + IPv6	127	NA

Hardware offload BFD Echo Sessions on RSP2 Module for BFD over G8032

BFD echo packet with the same source and destination IP address are punted to the ICMP queue. This generates ICMP redirects.

Table 9: Hardware Limit for the Cisco RSP 2 module

BFD	Hardware Offload
IPv4	1023
IPv6	1023

Table 10: Hardware Limit for the Cisco RSP 2 module for BFD over G8032

BFD	Hardware Offload
IPv4	63
IPv6	63

- Information About BFD Deterministic Hardware Offload, on page 95
- Configuring BFD Deterministic Hardware Offload, on page 95
- Example for Configuring BFD Deterministic Hardware Offload, on page 96

Information About BFD Deterministic Hardware Offload

The BFD deterministic hardware offload feature allows you to define the BFD transmission timer value. Any BFD sessions below or equal to the set value are sent to hardware. In the conventional BFD design, sessions can be hosted in hardware or software. However, sessions hosted in hardware have the advantage of running at more aggressive timers and not being impacted by CPU spikes.

Currently the determination of which session is hosted in hardware or software is done on a first come first serve basis. The sessions are hosted in hardware till the hardware limit is reached and then the sessions are either hosted in software (if supported) or rejected. In addition, after stateful switchover (SSO) or on reloading, sessions which are hosted in hardware do not necessarily remain in hardware. A similar condition applies for software sessions. The number of BFD sessions that can be supported in the hardware are also limited.

The BFD deterministic hardware offload feature allows you to select the destination of sessions to either hardware or software. This feature allows sessions hosted in hardware to remain in hardware after SSO or reload.

Using this feature you can configure a global command which defines the BFD transmission timer below or equal to the sessions are hosted in hardware.

Note On reaching hardware session limit, inactive BFD session do not come up when an active session is removed.

Note Starting with Cisco IOS XE Release 3.13, the global command defines the BFD transmission timer below which the sessions are hosted in hardware; those equal to or above this value are hosted in the software.

Configuring BFD Deterministic Hardware Offload

To set the timer value below which the BFD sessions are offloaded to hardware, complete the following steps:

Procedure

Step 1 enable

Example:

```
Router> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 configure terminal

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 3 platform bfd-offload-timer *timervalue*

Example:

```
Router(config)# platform bfd-offload-timer 2000
```

Sets the timer value.

- *timervalue*—The valid range is from 200000-999000 microseconds.

  Note
  There is no default value for this timer. The timer is enabled only after you specify a valid non-zero value. Starting with Cisco IOS Release 3.13, the timer value range is from 200000-999000 microseconds. For releases prior to Cisco IOS Release 3.13, the timer value range is 3300-999000 microseconds.

Step 4 end

Example:

```
Router(config)# end
```

Exits interface configuration mode and returns to privileged EXEC mode.

Example for Configuring BFD Deterministic Hardware Offload

The following example shows how to set the timer value to 400 microseconds:

Router> enable

Router# configure terminal

Router(config)# platform bfd-offload-timer 400000

Router(config)# end

Related Documents

Related Topic	Document Title
Bidirectional Forwarding Detection	*Cisco IOS Software Releases 12.2 SX Bidirectional Forwarding*

Standards

Standard	Title
IETF Draft	Bidirectional Forwarding Detection, January 2006 (http://www.ietf.org/internet-drafts/draft-ietf-bfd-base-03.txt)
IETF Draft	BFD for IPv4 and IPv6 (Single Hop), March 2005 (http://www.ietf.org/internet-drafts/draft-ietf-bfd-v4v6-1hop-0