



# 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 operators to easily identify and analyze issues with BFD sessions.

- [Prerequisites for BFD Debug Enhancement, on page 1](#)
- [Restrictions for BFD Debug Enhancement, on page 1](#)
- [Information About BFD Debug Enhancement, on page 1](#)
- [How to Configure BFD Debug Enhancement, on page 2](#)
- [Configuration Examples for BFD Debug Enhancement, on page 4](#)

## 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)
```

