

IDR Workgroup  
Internet-Draft  
Updates: 4271 (if approved)  
Intended status: Standards Track  
Expires: 31 December 2025

M. Zheng  
Ciena  
A. Lindem  
Arrcus, Inc  
J. Haas  
Juniper Networks, Inc.  
A. Fu  
Bloomberg L.P.  
29 June 2025

BGP BFD Strict-Mode  
draft-ietf-idr-bgp-bfd-strict-mode-15

## Abstract

This document specifies extensions to RFC4271 BGP-4 that enable a BGP speaker to negotiate additional Bidirectional Forwarding Detection (BFD) extensions using a BGP capability. This BFD Strict-Mode Capability enables a BGP speaker to prevent a BGP session from being established until a BFD session is established. This is referred to as BFD "strict-mode".

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 31 December 2025.

## Copyright Notice

Copyright (c) 2025 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document.

Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Requirements Language . . . . .	4
3. BGP Session Attributes for BFD Strict-Mode . . . . .	4
4. BGP FSM Events for BFD Strict-Mode . . . . .	4
5. BFD Strict-Mode Capability Definition . . . . .	6
6. BFD Strict-Mode Capability Negotiation . . . . .	6
7. Starting and Stopping BFD Sessions Associated BGP BFD Strict-Mode . . . . .	7
8. BGP FSM State Changes . . . . .	7
8.1. Overview of BFD Strict FSM Changes . . . . .	7
8.2. Changes to the Idle State . . . . .	8
8.3. Changes to the Connect State . . . . .	8
8.3.1. Handling BfdAdminDown / Bfd_Disabled / BfdUp . . . . .	8
8.3.2. Handling BfdDown . . . . .	9
8.3.3. Handling BfdHoldTimer_Expires . . . . .	9
8.3.4. Handling BfdStrict_ConfigChanged . . . . .	9
8.3.5. Handling Event 20, BGPOpen with DelayOpenTimer running. . . . .	9
8.4. Changes to the Active State . . . . .	11
8.4.1. Handling BfdAdminDown / Bfd_Disabled / BfdUp . . . . .	12
8.4.2. Handling BfdDown . . . . .	12
8.4.3. Handling BfdHoldTimer_Expires . . . . .	12
8.4.4. Handling BfdStrict_ConfigChanged . . . . .	13
8.4.5. Handling Event 20, BGPOpen with DelayOpenTimer running. . . . .	13
8.5. Changes to the OpenSent State . . . . .	15
8.5.1. Handling BfdAdminDown / Bfd_Disabled / BfdUp . . . . .	15
8.5.2. Handling BfdDown . . . . .	15
8.5.3. Handling BfdHoldTimer_Expires . . . . .	16
8.5.4. Handling BfdStrict_ConfigChanged . . . . .	16
8.5.5. Handling Event 19, BGPOpen . . . . .	17
8.6. Changes to the OpenConfirm State . . . . .	18
8.6.1. Handling BfdAdminDown / Bfd_Disabled / BfdUp . . . . .	18
8.6.2. Handling BfdDown . . . . .	18
8.6.3. Handling BfdStrict_ConfigChanged . . . . .	19
8.7. Changes to the Established State . . . . .	19
8.7.1. Handling BfdAdminDown / Bfd_Disabled / BfdUp . . . . .	19
8.7.2. Handling BfdDown . . . . .	19
8.7.3. Handling BfdStrict_ConfigChanged / BfdHoldTimer_Expires . . . . .	20

9. Closing BGP Sessions . . . . .	20
10. Stability Considerations . . . . .	20
11. Manageability Considerations . . . . .	21
12. Security Considerations . . . . .	21
13. IANA Considerations . . . . .	21
14. Acknowledgement . . . . .	21
15. Normative References . . . . .	21
Authors' Addresses . . . . .	22

## 1. Introduction

Bidirectional Forwarding Detection BFD [RFC5880] enables routers to monitor data plane connectivity and to detect faults in the bidirectional forwarding path between them. This functionality is leveraged by routing protocols such as BGP [RFC4271] to rapidly react to topology changes in the face of path failures.

The BFD interaction with BGP is specified in Section 10.2 of [RFC5882]. When BFD is enabled for a BGP neighbor, faults in the bidirectional forwarding detected by BFD result in BGP session termination. It is possible in some failure scenarios for the network to be in a state such that a BGP session may be established but a BFD session cannot be established. In some other scenarios, it may be possible to establish a BGP session, but a degraded or poor-quality link may result in the corresponding BFD session going up and down frequently.

To avoid situations that result in routing churn and to minimize the impact of network interruptions, it will be beneficial to disallow BGP to establish a session until BFD session is successfully established and has stabilized. We refer to this mode of operation as BFD "strict-mode". However, always using "strict-mode" would preclude BGP operation in an environment where not all routers support BFD strict-mode or have BFD enabled.

This document defines BFD "strict-mode" operation as preventing BGP session establishment until both the local and remote speakers have an established BFD session. The document also specifies a BGP capability [RFC5492] for announcing BFD parameters including a BGP speaker's support for "strict-mode"; i.e., requiring a BFD session for BGP session establishment.

## 2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

## 3. BGP Session Attributes for BFD Strict-Mode

Defined in this document:

### TBD-X) BfdEnabled:

A boolean value that is TRUE when BFD is configured and enabled for this BGP session.

### TBD-X) BfdStrictEnabled:

A boolean value that is TRUE when BGP BFD Strict-Mode procedures are to be used when BFD is enabled for this BGP session. If BfdEnabled is not TRUE for this BGP session, this attribute has no impact.

### TBD-X) BfdHoldTime:

Hold time value used for the BfdHoldTimer. The default value for this attribute is 30 seconds and is user configurable.

### TBD-X) BfdHoldTimer:

Hold timer used when the BGP HoldTime has been negotiated to zero to ensure the BGP session terminates if the associated BFD session does not enter the Up state.

### TBD-X) BfdStrictNegotiated:

A boolean value that is TRUE when the BFD strict-mode feature capability has been successfully negotiated for this BGP session. (See Section 6.)

Defined in RFC 5880:

### bfd.SessionState:

The BFD session state associated with this BGP session when BFD is configured and enabled for the session. (See Section 6.8.1 of [RFC5880].)

## 4. BGP FSM Events for BFD Strict-Mode

Event TBD-X: BfdAdminDown

## Definition:

The BFD session associated with this BGP session has transitioned to the AdminDown state.

## Status:

Optional

## Optional Attribute Status:

The BfdEnabled attribute for this BGP session SHOULD be set to TRUE.

## Event TBD-X: BfdDown

## Definition:

The BFD session associated with this BGP session has transitioned to the Down state.

## Status:

Optional

## Optional Attribute Status:

## Event TBD-X: BfdUp

## Definition:

The BFD session associated with this BGP session has transitioned to the Up state.

## Status:

Optional

## Optional Attribute Status:

The BfdEnabled attribute for this BGP session SHOULD be set to TRUE.

## Event TBD-X: Bfd\_Disabled

## Definition:

The BfdEnabled session attribute has been changed to FALSE.

## Status:

Optional

## Optional Attribute Status:

## Event TBD-X: BfdHoldTimer\_Expires

## Definition:

The BFD holdtimer, which is set when the negotiated BGP hold time is zero, has expired.

Status:  
Optional

Optional Attribute Status:

- \* The HoldTimer SHOULD NOT be running.
- \* The negotiated HoldTime SHOULD be zero.
- \* The BGP session state SHOULD be in Connect, Active, or OpenSent.

Event TBD-X: BfdStrict\_ConfigChanged

Definition:

The configuration for the BFD strict configuration for the BGP session has been changed.

Status:  
Optional

Optional Attribute Status:

If BfdEnabled is FALSE, this event MUST NOT occur. When BFD has been disabled, the local system will trigger a BfdAdminDown event instead.

## 5. BFD Strict-Mode Capability Definition

The BFD Strict-Mode Capability is a BGP Capability [RFC5492] defined as follows:

Capability code: 74

Capability length: 0 octets

## 6. BFD Strict-Mode Capability Negotiation

A BGP speaker which supports capabilities advertisement and has BFD strict-mode enabled MUST include the BFD Strict-Mode Capability in its OPEN message.

A BGP speaker which supports the BFD Strict-Mode Capability examines the list of capabilities received from its peer. If both the local and remote BGP speakers include the BFD Strict-Mode Capability, the BfdStrictNegotiated session attribute (Section 3 below) is set to TRUE.

## 7. Starting and Stopping BFD Sessions Associated BGP BFD Strict-Mode

Implementations SHOULD start the BFD session associated with the BGP BFD strict-mode session prior to the BGP FSM starting. The motivation is to avoid delaying BGP FSM transitions while waiting for the BFD session reach the Up state.

Similarly, to support BFD hold-down requirements for detecting BFD session stability (see Section 10), implementations SHOULD NOT immediately destroy BFD sessions when associated BGP connections transition to Idle.

## 8. BGP FSM State Changes

### 8.1. Overview of BFD Strict FSM Changes

When BFD is enabled, and BFD strict-mode is enabled and negotiated, the BGP finite state machine is prevented from send a KEEPALIVE to the remote BGP speaker and advancing to the OpenConfirm state until the associated BFD session has reached the Up state.

In the FSM defined in [RFC4271], sending of a KEEPALIVE to the remote BGP speaker and advancement to the OpenConfirm state happens:

- \* In the Connect state upon receiving an OPEN message and the DelayOpenTimer is running.
- \* In the Active state upon receiving an OPEN message and the DelayOpenTimer is running.
- \* In the OpenSent upon receiving an OPEN message.

For each of these scenarios, when BFD is enabled, and BFD strict-mode is negotiated, a sub-state is introduced to track the pending BFD Up event:

- \* ConnectDelayOpenBfdUpPending
- \* ActiveDelayOpenBfdUpPending
- \* OpenSentBfdUpPending

If BFD strict-mode configuration is changed once the BGP FSM has started executing, but has not reached the Established state, the session is reset to the Idle state to ensure consistent behavior. I.e., no unexpected timers are running, and the BGP session's transition to Established is not lingering on a pending event. Once the BGP session has reached the Established state, changes to BFD strict-mode are irrelevant since the work of this feature has been completed.

The following changes are made to the BGP FSM defined in Section 8.2.2 of [RFC4271]:

## 8.2. Changes to the Idle State

In the "Idle State", the BfdAdminDown, BfdDown, BfdUp, Bfd\_Disabled, BfdStrict\_ConfigChanged events are ignored.

In the "Idle State", the BfdHoldTimer\_Expires event is ignored, but only would occur as an error in the FSM implementation.

## 8.3. Changes to the Connect State

The BfdHoldTimer is reset to zero and stopped on any transition to the Idle state.

### 8.3.1. Handling BfdAdminDown / Bfd\_Disabled / BfdUp

In response to the BfdAdminDown event (Event TBD-X), the Bfd\_Disabled event (Event TBD-X), or the BfdUp event (Event TBD-X) the the local system checks to see if it is in the ConnectDelayOpenBfdUpPending sub-state. If the FSM is in the ConnectDelayOpenBfdUpPending sub-state, the local system:

- \* sends a KEEPALIVE message,
- \* if the HoldTimer initial value is non-zero,
  - starts the KeepaliveTimer with the initial value and
  - resets the BfdHoldTimer value to zero,
- \* and changes its state to OpenConfirm (leaves ConnectDelayOpenBfdUpPending).

If the FSM is not in the ConnectDelayOpenBfdUpPending sub-state, the local system:

- \* stays in the Connect state.



### 8.3.2. Handling BfdDown

The BfdDown event (Event TBD-X) is ignored while in the Connect state.

A BFD session can transition to Down from the Init state, indicating the session has failed to come Up, or transition to Down from the AdminDown as part of starting the BFD state machine.

### 8.3.3. Handling BfdHoldTimer\_Expires

In response to the BfdHoldTimer\_Expires event (Event TBD-X), the local system:

- \* sends a NOTIFICATION message with the error code Cease (6) and error subcode BFD Down (10),
- \* drops the TCP connection,
- \* releases all BGP resources,
- \* increments the ConnectRetryCounter,
- \* (optionally) performs peer oscillation damping if the DampPeerOscillations attribute is set to TRUE, and
- \* changes its state to Idle.

### 8.3.4. Handling BfdStrict\_ConfigChanged

In response to the BfdStrict\_ConfigChanged event (Event TBD-X) the local system:

- \* drops the TCP connection,
- \* releases all BGP resources,
- \* sets ConnectRetryCounter to zero,
- \* stops the ConnectRetryTimer and sets ConnectRetryTimer to zero, and
- \* changes its state to Idle.

### 8.3.5. Handling Event 20, BGPOpen with DelayOpenTimer running.

In the "Connect State", the handling of Event 20, an OPEN message is received while the DelayOpenTimer is running, is revised as follows:

## Old Text:

- \* stops the ConnectRetryTimer (if running) and sets the ConnectRetryTimer to zero,
- \* completes the BGP initialization,
- \* stops and clears the DelayOpenTimer (sets the value to zero),
- \* sends an OPEN message,
- \* sends a KEEPALIVE message,
- \* if the HoldTimer initial value is non-zero,
  - starts the KeepaliveTimer with the initial value and
  - resets the HoldTimer to the negotiated value,
- \* else, if the HoldTimer initial value is zero,
  - resets the KeepaliveTimer and
  - resets the HoldTimer value to zero,
- \* and changes its state to OpenConfirm.

If the value of the autonomous system field is the same as the local Autonomous System number, set the connection status to an internal connection; otherwise it will be "external".

## New Text:

- \* stops the ConnectRetryTimer (if running) and sets the ConnectRetryTimer to zero,
- \* completes the BGP initialization,
- \* stops and clears the DelayOpenTimer (sets the value to zero),
- \* sends an OPEN message,
- \* If BfdEnabled is TRUE, and BfdStrictNegotiated is TRUE, and bfd.SessionState is neither Up nor AdminDown,
  - DOES NOT send a KEEPALIVE message,
  - if the HoldTimer initial value is non-zero,

- o DOES NOT start the KeepaliveTimer
  - o resets the HoldTimer to the negotiated value,
- else, if the HoldTimer initial value is zero,
  - o resets the KeepaliveTimer and
  - o resets the HoldTimer value to zero,
  - o starts the BfdHoldTimer with the value BfdHoldTime,
- stays in the Connect state (enters ConnectDelayOpenBfdUpPending).

\* else,

- sends a KEEPALIVE message,
- if the HoldTimer initial value is non-zero,
  - o starts the KeepaliveTimer with the initial value and
  - o resets the HoldTimer to the negotiated value,
- else, if the HoldTimer initial value is zero,
  - o resets the KeepaliveTimer and
  - o resets the HoldTimer value to zero,
- and changes its state to OpenConfirm.

If the value of the autonomous system field is the same as the local Autonomous System number, set the connection status to an internal connection; otherwise it will be "external".

#### 8.4. Changes to the Active State

The BfdHoldTimer is reset to zero and stopped for any transition to the Idle state.

#### 8.4.1. Handling BfdAdminDown / Bfd\_Disabled / BfdUp

In response to the BfdAdminDown event (Event TBD-X), the Bfd\_Disabled event (Event TBD-X), or the BfdUp event (Event TBD-X), the local system checks to see if it is in the ActiveDelayOpenBfdUpPending sub-state. If the FSM is in the ActiveDelayOpenBfdUpPending sub-state, the local system:

- \* sends a KEEPALIVE message,
- \* if the HoldTimer initial value is non-zero,
  - starts the KeepaliveTimer with the initial value and
  - resets the BfdHoldTimer value to zero,
- \* and changes its state to OpenConfirm (leaves ActiveDelayOpenBfdUpPending).

If the FSM is not in the ActiveDelayOpenBfdUpPending sub-state, the local system:

- \* stays in the Active state.

#### 8.4.2. Handling BfdDown

The BfdDown event (Event TBD-X) is ignored while in the Active state.

A BFD session can transition to Down from the Init state, indicating the session has failed to come Up, or transition to Down from the AdminDown as part of starting the BFD state machine.

#### 8.4.3. Handling BfdHoldTimer\_Expires

In response to the BfdHoldTimer\_Expires event (Event TBD-X), the local system:

- \* sends a NOTIFICATION message with the error code Cease (6) and error subcode BFD Down (10),
- \* drops the TCP connection,
- \* releases all BGP resources,
- \* increments the ConnectRetryCounter,
- \* (optionally) performs peer oscillation damping if the DampPeerOscillations attribute is set to TRUE, and

- \* changes its state to Idle.

#### 8.4.4. Handling BfdStrict\_ConfigChanged

In response to the BfdStrict\_ConfigChanged event (Event TBD-X), the local system:

- \* drops the TCP connection,
- \* releases all BGP resources,
- \* sets ConnectRetryCounter to zero,
- \* stops the ConnectRetryTimer and sets ConnectRetryTimer to zero, and
- \* changes its state to Idle.

#### 8.4.5. Handling Event 20, BGPOpen with DelayOpenTimer running.

In the "Active State", the handling of Event 20, an OPEN message is received while the DelayOpenTimer is running, is revised as follows:

Old Text:

- \* stops the ConnectRetryTimer (if running) and sets the ConnectRetryTimer to zero,
- \* completes the BGP initialization,
- \* stops and clears the DelayOpenTimer (sets the value to zero),
- \* sends an OPEN message,
- \* sends a KEEPALIVE message,
- \* if the HoldTimer initial value is non-zero,
  - starts the KeepaliveTimer with the initial value and
  - resets the HoldTimer to the negotiated value,
- \* else, if the HoldTimer initial value is zero,
  - resets the KeepaliveTimer and
  - resets the HoldTimer value to zero,

- \* and changes its state to OpenConfirm.

If the value of the autonomous system field is the same as the local Autonomous System number, set the connection status to an internal connection; otherwise it will be "external".

New Text:

- \* stops the ConnectRetryTimer (if running) and sets the ConnectRetryTimer to zero,
- \* completes the BGP initialization,
- \* stops and clears the DelayOpenTimer (sets the value to zero),
- \* sends an OPEN message,
- \* If BfdEnabled is TRUE, and BfdStrictNegotiated is TRUE, and bfd.SessionState is neither Up nor AdminDown,
  - DOES NOT send a KEEPALIVE message,
  - if the HoldTimer initial value is non-zero,
    - o DOES NOT start the KeepaliveTimer
    - o resets the HoldTimer to the negotiated value,
  - else, if the HoldTimer initial value is zero,
    - o resets the KeepaliveTimer and
    - o resets the HoldTimer value to zero,
    - o starts the BfdHoldTimer with the value BfdHoldTime,
  - stays in the Active state (enters ActiveDelayOpenBfdUpPending).
- \* else,
  - sends a KEEPALIVE message,
  - if the HoldTimer initial value is non-zero,
    - o starts the KeepaliveTimer with the initial value and
    - o resets the HoldTimer to the negotiated value,

- else, if the HoldTimer initial value is zero,
  - o resets the KeepaliveTimer and
  - o resets the HoldTimer value to zero,
- and changes its state to OpenConfirm.

If the value of the autonomous system field is the same as the local Autonomous System number, set the connection status to an internal connection; otherwise it will be "external".

#### 8.5. Changes to the OpenSent State

The BfdHoldTimer is reset to zero and stopped for any transition to the Idle state.

##### 8.5.1. Handling BfdAdminDown / Bfd\_Disabled / BfdUp

In response to the the BfdAdminDown event (Event TBD-X), the Bfd\_Disabled event (Event TBD-X), or the BfdUp event (Event TBD-X), and the FSM is in the OpenSentBfdUpPending sub-state, the local system:

- \* sends a KEEPALIVE message, and
- \* sets a KeepaliveTimer (via the text below)
- \* resets the BfdHoldTimer value to zero,
- \* changes its state to OpenConfirm (leaves OpenSentBfdUpPending).

If the FSM is not in the OpenSentBfdUpPending sub-state, the local system:

- \* stays in the OpenSent state.

##### 8.5.2. Handling BfdDown

In response to the BfdDown event (Event TBD-X):

- \* if BfdEnabled is TRUE, and BfdStrictNegotiated is TRUE, the local system:
  - sends a NOTIFICATION message with the error code Cease (6) and error subcode BFD Down (10),
  - drops the TCP connection,

- releases all BGP resources,
  - sets ConnectRetryCounter to zero,
  - stops the ConnectRetryTimer and sets ConnectRetryTimer to zero,  
and
  - changes its state to Idle.
- \* else,
    - stays in the OpenSent State

#### 8.5.3. Handling BfdHoldTimer\_Expires

In response to the BfdHoldTimer\_Expires event (Event TBD-X), the local system:

- \* sends a NOTIFICATION message with the error code Cease (6) and error subcode BFD Down (10),
- \* drops the TCP connection,
- \* releases all BGP resources,
- \* increments the ConnectRetryCounter,
- \* (optionally) performs peer oscillation damping if the DampPeerOscillations attribute is set to TRUE, and
- \* changes its state to Idle.

#### 8.5.4. Handling BfdStrict\_ConfigChanged

In response to the BfdStrict\_ConfigChanged event (Event TBD-X), the local system:

- \* sends the NOTIFICATION with an error code Cease (6), error subcode Other Configuration Change (6),
- \* drops the TCP connection,
- \* releases all BGP resources,
- \* sets ConnectRetryCounter to zero,
- \* stops the ConnectRetryTimer and sets ConnectRetryTimer to zero,  
and



- \* changes its state to Idle.

#### 8.5.5. Handling Event 19, BGPOpen

##### Old Text:

When an OPEN message is received, all fields are checked for correctness. If there are no errors in the OPEN message (Event 19), the local system:

- \* resets the DelayOpenTimer to zero,
- \* sets the BGP ConnectRetryTimer to zero,
- \* sends a KEEPALIVE message, and
- \* sets a KeepaliveTimer (via the text below)
- \* sets the HoldTimer according to the negotiated value (see Section 4.2), - changes its state to OpenConfirm.

If the negotiated hold time value is zero, then the HoldTimer and KeepaliveTimer are not started. If the value of the Autonomous System field is the same as the local Autonomous System number, then the connection is an "internal" connection; otherwise, it is an "external" connection.

##### New Text:

When an OPEN message is received, all fields are checked for correctness. If there are no errors in the OPEN message (Event 19), the local system:

- \* resets the DelayOpenTimer to zero,
- \* sets the BGP ConnectRetryTimer to zero,
- \* sets the HoldTimer according to the negotiated value (see Section 4.2),
- \* If BfdEnabled is TRUE, and BfdStrictNegotiated is TRUE, and bfd.SessionState is neither Up nor AdminDown,
  - DOES NOT send a KEEPALIVE message, and
  - DOES NOT start the KeepaliveTimer
  - if the HoldTimer negotiated value is zero,

- o starts the BfdHoldTimer with the value BfdHoldTime,
  - stays in OpenSent state (OpenSentBfdUpPending)
- \* else,
- sends a KEEPALIVE message, and
  - sets a KeepaliveTimer (via the text below)
  - changes its state to OpenConfirm.

If the negotiated hold time value is zero, then the HoldTimer and KeepaliveTimer are not started. If the value of the Autonomous System field is the same as the local Autonomous System number, then the connection is an "internal" connection; otherwise, it is an "external" connection.

## 8.6. Changes to the OpenConfirm State

### 8.6.1. Handling BfdAdminDown / Bfd\_Disabled / BfdUp

The BfdAdminDown, Bfd\_Disabled, and BfdUp events are ignored in the OpenConfirm state.

### 8.6.2. Handling BfdDown

In response to the BfdDown event (Event TBD-X):

- \* if BfdEnabled is TRUE, and BfdStrictNegotiated is TRUE, the local system:
- sends a NOTIFICATION message with the error code Cease (6) and error subcode BFD Down (10),
  - drops the TCP connection,
  - releases all BGP resources,
  - sets ConnectRetryCounter to zero,
  - stops the ConnectRetryTimer and sets ConnectRetryTimer to zero, and
  - changes its state to Idle.
- \* else,

- stays in the OpenConfirm State

#### 8.6.3. Handling BfdStrict\_ConfigChanged

In response to the BfdStrict\_ConfigChanged event (Event TBD-X), the local system:

- \* sends a NOTIFICATION message with the error code Cease (6) and error subcode Other Configuration Change (6),
- \* drops the TCP connection,
- \* releases all BGP resources,
- \* sets ConnectRetryCounter to zero,
- \* stops the ConnectRetryTimer and sets ConnectRetryTimer to zero, and
- \* changes its state to Idle.

#### 8.7. Changes to the Established State

##### 8.7.1. Handling BfdAdminDown / Bfd\_Disabled / BfdUp

The BfdAdminDown, Bfd\_Disabled, and BfdUp events are ignored in the Established state.

##### 8.7.2. Handling BfdDown

In response to the BfdDown event (Event TBD-X), the local system:

- \* sends a NOTIFICATION message with the error code Cease (6) and error subcode BFD Down (10),
- \* drops the TCP connection,
- \* deletes all routes associated with this connection,
- \* releases all BGP resources,
- \* increments the ConnectRetryCounter by 1,
- \* (optionally) performs peer oscillation damping if the DampPeerOscillations attribute is set to TRUE, and
- \* changes its state to Idle.

### 8.7.3. Handling BfdStrict\_ConfigChanged / BfdHoldTimer\_Expires

The BfdStrict\_ConfigChange event is ignored in the Established state.

The BfdHoldTimer\_Expires event in the Established state is a FSM error, and is ignored.

## 9. Closing BGP Sessions

When BGP sessions are closed according to the procedures in this document, the session SHOULD be terminated with a NOTIFICATION message with the Cease Code (6) and the "BFD Down" Subcode (10); see [RFC9384]. This informs the operator that interaction with BFD is the root cause of the BGP session being unable to move to the Established state.

## 10. Stability Considerations

The use of BFD strict-mode along with mechanisms such as hold-down (a delay in the initial BGP Establishment state following BFD session establishment) and/or dampening (a delay in the BGP Establishment state following failure detected by BFD) may help reduce the frequency of BGP session flaps and therefore reduce the associated routing churn.

To avoid deadlock when utilizing both BFD hold-down and BFD strict-mode, when strict-mode is enabled for a peer, the BGP FSM MUST be enabled. That is, BFD hold-down procedures MUST NOT prevent BGP from establishing a connection with the remote BGP speaker.

If both the local and remote BGP speakers include the BFD Strict-Mode Capability, the BGP state machine is permitted to transition to the Established state from the OpenConfirm state after the locally configured BFD hold-down interval is observed. That is, the BFD session has been Up for the desired amount of time.

It is RECOMMENDED that the BFD hold-down intervals used with BFD strict-mode, when configured, use similar values. Similarly, the negotiated BGP holdtime SHOULD be long enough to account for the time between the BGP FSM reaching the OpenConfirm state, the BFD hold-down interval, and any delay for the BFD session being initiated. Failure to do so can result in the BGP speaker that has transitioned to the Established state expiring its BGP holdtime and closing the connection. This is because the remote BGP speaker hasn't transitioned to Established and begun sending KEEPALIVE messages.

A BGP speaker SHOULD log a message if it closes its session due to hold timer expiration while waiting for the BFD hold-down interval.

The behavior of BGP speakers implementing BFD hold-down without negotiating the BFD strict-mode feature is out of scope of this document. However, the authors are aware that inconsistent behaviors in BGP implementations for BFD hold-down without BFD strict-mode may result in BGP session deadlock.

## 11. Manageability Considerations

Auto-configuration is possible for enabling BFD strict-mode. However, the configuration automation is out of the scope of this document.

To simplify troubleshooting and avoid inconsistencies, it is RECOMMENDED that BFD strict-mode configuration be consistent for both BGP peers.

This draft introduces sub-states in the existing BGP finite state machine for tracking BFD session status inputs for strict mode operation. Implementations SHOULD provide visibility for these sub-states in its display of the BGP finite state machine.

## 12. Security Considerations

The mechanism defined in this document interacts with the BGP finite state machine when so configured. The security considerations for BFD thus, become BGP-4 considerations [RFC4271] when so used. Given that a BFD session is required for a BGP session, a Denial-of-Service (DoS) attack on BGP can now be mounted by preventing a BFD session between the BGP peers from reaching the Up state, or interrupting an existing BFD session. The use of a BFD Authentication mechanism, some of which are defined in [RFC5880], is thus RECOMMENDED when used to protect BGP-4 [RFC4271].

## 13. IANA Considerations

This document defines the BFD Strict-Mode Capability. The Capability Code 74 has been assigned from the First-Come-First-Served range (64-238) of the Capability Codes registry.

## 14. Acknowledgement

The authors would like to acknowledge the review and inputs from Shyam Sethuram, Mohammed Mirza, Bruno Decraene, Carlos Pignataro, Enke Chen, Anup Kumar, and Ketan Talalukar.

## 15. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <<https://www.rfc-editor.org/info/rfc4271>>.
- [RFC5492] Scudder, J. and R. Chandra, "Capabilities Advertisement with BGP-4", RFC 5492, DOI 10.17487/RFC5492, February 2009, <<https://www.rfc-editor.org/info/rfc5492>>.
- [RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <<https://www.rfc-editor.org/info/rfc5880>>.
- [RFC5882] Katz, D. and D. Ward, "Generic Application of Bidirectional Forwarding Detection (BFD)", RFC 5882, DOI 10.17487/RFC5882, June 2010, <<https://www.rfc-editor.org/info/rfc5882>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC9384] Haas, J., "A BGP Cease NOTIFICATION Subcode for Bidirectional Forwarding Detection (BFD)", RFC 9384, DOI 10.17487/RFC9384, March 2023, <<https://www.rfc-editor.org/info/rfc9384>>.

#### Authors' Addresses

Mercia Zheng  
Ciena  
3939 N. 1st Street  
San Jose, CA 95134  
United States  
Email: [merciaz.ietf@gmail.com](mailto:merciaz.ietf@gmail.com)

Acee Lindem  
Arrcus, Inc  
301 Midenhall Way  
Cary, NC 27513  
United States  
Email: [acee.ietf@gmail.com](mailto:acee.ietf@gmail.com)

Jeffrey Haas  
Juniper Networks, Inc.  
1133 Innovation Way  
Sunnyvale, CA 94089  
United States of America  
Email: jhaas@juniper.net

Albert Fu  
Bloomberg L.P.  
731 Lexington Avenue  
New York, NY 10022  
United States of America  
Email: aful4@bloomberg.net