Internet Engineering Task Force (IETF)
Request for Comments: 7419
Updates: 5880
Category: Informational
ISSN: 2070-1721

N. Akiya
M. Binderberger
Cisco Systems
G. Mirsky
Ericsson

December 2014

Common Interval Support in Bidirectional Forwarding Detection

Abstract

Bidirectional Forwarding Detection (BFD) requires that messages be transmitted at regular intervals and provides a way to negotiate the interval used by BFD peers. Some BFD implementations may be restricted to only support several interval values. When such BFD implementations speak to each other, there is a possibility of two sides not being able to find a common value for the interval to run BFD sessions.

This document updates RFC 5880 by defining a small set of interval values for BFD that we call "Common Intervals" and recommends implementations to support the defined intervals. This solves the problem of finding an interval value that both BFD speakers can support while allowing a simplified implementation as seen for hardware-based BFD. It does not restrict an implementation from supporting more intervals in addition to the Common Intervals.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7419.

Table of Contents

1. Introduction

The Bidirectional Forwarding Detection (BFD) standard [RFC5880] describes how to calculate the transmission interval and the detection time. However, it does not make any statement about how to solve a situation where one BFD speaker cannot support the calculated value. In practice, this may not have been a problem as long as software-implemented timers were used and as long as the granularity of such timers was small compared to the interval values being supported, i.e. as long as the error in the timer interval was small compared to 25 percent jitter.

In the meantime, requests exist for very fast interval values, down to 3.3 msec for the MPLS Transport Profile (MPLS-TP). At the same time, the requested scale for the number of BFD sessions is increasing. Both requirements have driven vendors to use Network Processors (NP), Field Programmable Gate Arrays (FPGAs), or other hardware-based solutions to offload the periodic packet transmission and the timeout detection in the receive direction. A potential

problem with this hardware-based BFD is the granularity of the interval timers. Depending on the implementation, only a few intervals may be supported, which can cause interoperability problems. This document proposes a set of interval values that should be supported by all implementations. Details are laid out in the following sections.

2. The Problem with Few Supported Intervals

Let's assume vendor "A" supports 10 msec, 100 msec, and 1 sec interval timers in hardware, and vendor "B" supports every value from 20 msec onward, with a granularity of 1 msec. For a BFD session, "A" tries to set up the session with 10 msec while "B" uses 20 msec as the value for RequiredMinRxInterval and DesiredMinTxInterval. Rx and Tx are negotiated as described in [RFC5880], which is 20 msec in this case. However, system "A" is not able to support the 20 msec interval timer. Multiple ways exist to resolve the dilemma, but none of them is without problems.

a. Realizing that it cannot support 20 msec, system "A" sends out a new BFD packet advertising the next larger interval of 100 msec with RequiredMinRxInterval and DesiredMinTxInterval. The new negotiated interval between "A" and "B" is then 100 msec, which is supported by both systems. However, the problem is that we moved from the 10/20 msec range to 100 msec, which has far deviated from operator expectations.

b. System "A" could violate [RFC5880] and use the 10 msec interval for the Tx direction. In the receive direction, it could use an adjusted multiplier value M' = 2 * M to match the correct detection time. Now, in addition to the fact that we explicitly violate [RFC5880], there may be the problem that system "B" drops up to 50% of the packets; this could be the case when "B" uses an ingress rate policer to protect itself and the policer would be programmed with an expectation of 20 msec receive intervals.

The example above could be worse when we assume that system "B" can only support a few timer values itself. Let's assume "B" supports 20 msec, 300 msec, and 1 sec. If both systems would adjust their advertised intervals, then the adjustment ends at 1 sec. The example above could even be worse when we assume that system "B" can only support 50 msec, 500 msec, and 2 sec. Even if both systems walk through all of their supported intervals, the two systems will never be able to agree on an interval to run any BFD sessions.

3. Well-Defined, Common Intervals

The problem can be reduced by defining interval values that are supported by all implementations. Then, the adjustment mechanism could find a commonly supported interval without deviating too much from the original request.

In technical terms, the requirement is as follows: a BFD implementation should support all values in the set of Common Interval values that are equal to or larger than the fastest (i.e., lowest) interval the particular BFD implementation supports.

This document defines the set of Common Interval values to be: 3.3 msec, 10 msec, 20 msec, 50 msec, 100 msec, and 1 sec.

In addition, both a 10 sec interval and multiplier values up to 255 are recommended to support graceful restart.

The adjustment is always towards larger (i.e., slower) interval values when the initial interval proposed by the peer is not supported.

This document is not adding new requirements with respect to the precision with which a timer value must be implemented. Supporting an interval value means advertising this value in the DesiredMinTxInterval and/or RequiredMinRxInterval field of the BFD packets and providing timers that are reasonably close. [RFC5880] defines safety margins for the timers by defining a jitter range.

How is the Common Interval set used exactly? In the example above, vendor "A" has a fastest interval of 10 msec and thus would be required to support all intervals in the Common Interval set that are equal or larger than 10 msec, i.e., it would support 10 msec, 20 msec, 50 msec, 100 msec, and 1 sec. Vendor "B" has a fastest interval of 20 msec and thus would need to support 20 msec, 50 msec, 100 msec, and 1 sec. As long as this requirement is met for the common set of values, then both vendor "A" and "B" are free to support additional values outside of the Common Interval set.

4. Security Considerations

This document does not introduce any additional security concerns. The security considerations described in the BFD documents, [RFC5880] and others, apply to devices implementing the BFD protocol, regardless of whether or not the Common Interval set is implemented.

5. References

5.1. Normative References

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, June 2010, <http://www.rfc-editor.org/info/rfc5880>.

5.2. Informative References

[G.8013_Y.1731]
International Telecommunications Union, "OAM functions and mechanisms for Ethernet based networks", ITU-T Recommendation G.8013/Y.1731, November 2013.

[GR-253-CORE]
Telcordia Technologies, Inc., "Synchronous Optical Network (SONET) Transport Systems: Common Generic Criteria", GR-253-CORE Issue 05, October 2009.

Appendix A. Why Some Values Are in the Common Interval Set

The list of Common Interval values is trying to balance various objectives. The list should not contain too many values, as more timers may increase the implementation costs. On the other hand, fewer values produces larger gaps and adjustment jumps. More values in the lower interval range are thus seen as critical to support customer needs for fast detection in setups with multiple vendors.

- 3.3 msec: required by MPLS-TP, to support the defect detection time of 10 msec from [GR-253-CORE].
- 10 msec: general consensus is to support 10 msec. Multiple vendors plan to or do already implement 10 msec.
- 20 msec: basically avoids a larger gap in this critical interval region. Still allows 50-60 msec detect and restore (with multiplier of 2) and covers existing software-based implementations.
- 50 msec: widely deployed interval. Supporting this value reflects the reality of many BFD implementations today.
- 100 msec: similar to 10 msec, this value allows the reuse of [G.8013_Y.1731] implementations, especially hardware. It supports a large number of 100 msec sessions with multiplier 9 (9 x 100 msec), which could be replacing of 3 x 300 msec configurations used by customers to have a detection time slightly below 1 sec for VoIP setups.
- 1 sec: as mentioned in [RFC5880]. While the interval for Down packets can be 1 sec or larger, this document recommends use of exactly 1 sec to avoid interoperability issues.

The recommended value for large intervals is 10 sec, allowing for a timeout of 42.5 minutes with a multiplier of 255. This value is kept outside the Common Interval set, as it is not required for normal BFD operations that occur in the sub-second range. Instead, the expected usage is for graceful restart, if needed.

Appendix B. Timer Adjustment with Non-identical Interval Sets

[RFC5880] implicitly assumes that a BFD implementation can support any timer value equal to or above the advertised value. When a BFD speaker starts a Poll Sequence, then the peer must reply with the Final (F) bit set and adjust the transmit and detection timers accordingly. With contiguous software-based timers, this is a valid assumption. Even in the case of a small number of supported interval

values, this assumption holds when both BFD speakers support exactly the same interval values.

But what happens when both speakers support intervals that are not supported by the peer? An example is router "A" supporting the Common Interval set plus 200 msec, while router "B" supports the Common Intervals plus 300 msec. Assume both routers are configured and run at 50 msec. Now, router A is configured for 200 msec. We know the result must be that both BFD speakers use 1 sec timers, but how do they reach this endpoint?

First, router A sends a packet with 200 msec. The P bit is set according to [RFC5880]. The Tx timer stays at 50 msec, the detection timer is 3 * 200 msec:

(A) DesiredTx: 200 msec, MinimumRx: 200 msec, P-bit
Tx: 50 msec, Detect: 3 * 200 msec

Router B now must reply with an F bit. The problem is B is confirming timer values that it cannot support. The only setting to avoid a session flap would be

(B) DesiredTx: 300 msec, MinimumRx: 300 msec, F-bit
Tx: 50 msec, Detect: 3 * 300 msec

immediately followed by a P-bit packet, as the advertised timer values have been changed:

(B) DesiredTx: 300 msec, MinimumRx: 300 msec, P-bit
Tx: 50 msec, Detect: 3 * 300 msec

This is not exactly what Section 6.8.7 of [RFC5880] states about the transmission rate. On the other hand, as we will see, this state does not last for long. Router A would adjust its timers based on the received Final bit:

(A) Tx: 200 msec, Detect: 3 * 1 sec

Router A is not supporting the proposed 300 msec and would use 1 sec instead for the detection time. It would then respond to the received Poll Sequence from router B using 1 sec, as router A does not support the Max(200 msec, 300 msec):

(A) DesiredTx: 1 sec, MinimumRx: 1 sec, F-bit
Tx: 200 msec, Detect: 3 * 1 sec

followed by its own Poll Sequence, as the advertised timer values have been changed:

(A) DesiredTx: 1 sec, MinimumRx: 1 sec, P-bit
Tx: 200 msec, Detect: 3 * 1 sec

Router B would adjust its timers based on the received Final bit

(B) Tx: 300 msec , Detect: 3 * 1 sec

and would then reply to the Poll Sequence from router A:

(B) DesiredTx: 300 msec, MinimumRx: 300 msec, F-bit
Tx: 1 sec, Detect: 3 * 1 sec

which finally makes router A adjust its timers:

(A) Tx: 1 sec, Detect: 3 * 1 sec

In other words, router A and B go through multiple Poll Sequences until they reach a commonly supported interval value. Reaching such a value is guaranteed by this document.

Acknowledgments

We would like to thank Sylvain Masse and Anca Zamfir for bringing up the discussion about the Poll Sequence, and Jeffrey Haas for helping find the fine line between "exact" and "pedantic".

Authors' Addresses

Nobo Akiya
Cisco Systems

EMail: nobo@cisco.com

Marc Binderberger
Cisco Systems

EMail: mbinderb@cisco.com

Greg Mirsky
Ericsson

EMail: gregory.mirsky@ericsson.com