

Network Working Group
Internet Draft
Intended status: Standards Track
Expires: August 15, 2025

C. Lin
New H3C Technologies

February 15, 2025

Path Computation Element (PCE) Communication Protocol (PCEP) Send
Hold Timer
draft-lin-pcep-sendholdtimer-02

Abstract

This document defines the SendHoldTimer and SendHoldTimer_Expires events in the PCEP protocol. The implementation of SendHoldTimer helps address situations where a local system detects that a remote system has not terminated the PCEP session despite not processing PCEP messages. As per this document, when the SendHoldTimer expires, the local system should close the PCEP connection rather than solely relying on the remote system to terminate the session. This document provides updates to RFC5440.

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 August 15, 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 (<http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction.....	3
1.1. Conventions and Terminology.....	3
2. Problem Statement.....	3
3. SendHoldTimer - Changes to RFC 5440.....	4
3.1. Architectural.....	4
3.2. CLOSE Object.....	5
3.3. PCEP Finite State Machine (FSM).....	6
4. Security Considerations.....	7
5. IANA Considerations.....	7
6. Acknowledgements.....	7
7. References.....	7
7.1. Normative References.....	7
Authors' Addresses.....	8

1. Introduction

As described in [I-D.ietf-idr-bgp-sendholdtimer], any upper-layer protocol that uses TCP for transport can encounter similar situations where the remote system is unable to read TCP data due to a failure, leading to an inability to terminate the BGP connection. PCEP also requires a BGP-like send timeout mechanism [I-D.ietf-idr-bgp-sendholdtimer] to resolve this issue.

This document defines the SendHoldTimer and SendHoldTimer_Expires mechanisms in PCEP [RFC5440]. The failure to terminate a blocked PCEP session may result in a denial-of-service (DoS) attack, inhibiting the generation and delivery of essential PCEP messages such as PCReq, PCEPreplies, and PCNtf messages, consequently disrupting the normal PCE path computation function. This specification aims to address this issue by requiring the termination of the session at the local system when it detects that the remote system is unable to process any PCEP messages during the SendHoldTime period.

With the SendHoldTimer mechanism specified in this document, a blocked connection between PCC and PCE devices can be closed by the remote system, and it will also be closed locally if blocked.

1.1. Conventions and Terminology

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.

2. Problem Statement

An example of a network failure scenario: Since PCEP operates over TCP [RFC9293], a session is established between a PCC and a PCP. The PCC sends a PCReq message to the PCE to request path computation, and the PCC responds to the PCReq message with a PCRep message, used for traffic path selection.

In an established state between the PCC and PCE, there may be a situation where one party declares a zero-size TCP receive window (RCV.WND) to the other. This zero-size TCP receive window prevents the local system from sending crucial PCEP messages such as KEEPALIVE, PCReq, PCRep, PCErr, and PCNtf messages to the remote peer through the network socket.

In implementations lacking the `SendHoldTimer` concept, a failed or overloaded remote peer may result in a continuous accumulation of data on the local system's PCEP socket, affecting path selection, and potentially leading to forwarding failures and traffic loss.

In typical scenarios, PCEP implementations cannot observe the current receive window size of the underlying subsystem (such as TCP) or the peer, and there is no PCEP mechanism to terminate such a blocked session. Consequently, PCEP implementations are unable to consistently handle this situation.

This document provides a mechanism that enables PCEP implementations to detect whether the TCP socket between the PCC and PCE is active (data is being transmitted) or stalled. In the event of a stall, the PCEP session can be restarted to restore normal operation of the PCEP protocol.

3. `SendHoldTimer` - Changes to RFC 5440

3.1. Architectural

Chapter 4 introduces the framework of the PCEP protocol, including the `SendHoldTimer` and `SendHoldTimer_Expires` mechanisms. After the PCEP session state transitions to UP, the `SendHoldTimer` timer is initiated for this PCEP session, with `[SendHoldTime]` representing the timer's timeout value. This determines the duration for which the PCEP state is maintained when unable to transmit messages to its peer, and if this time is exceeded, the TCP connection will be terminated. PCC and PCE can configure the value of `SendHoldTime` independently for each peer.

Following the successful transmission of various messages, including `KeepAlive`, `PCReq`, and `PCRep`, the `SendHoldTimer` needs to be reset. In the event of a detected timeout of the `SendHoldTimer`, there is an option to immediately send a `Close` message with the reason "`SendHoldTimer_Expires`," which is an addition in this document.

The specific modification is as follows:

In the original document, it is required to ensure that any outstanding `PCReq` messages are sent before actively closing the session. However, when TCP is blocked and messages cannot be successfully transmitted, for the `SendHoldTimer_Expires` event, it is permissible to immediately send a `Close` message with the reason "`SendHoldTimer_Expires`."

The description in 4.2. Architectural Protocol Overview:

Old Text:

Each PCEP message is regarded as a single transmission unit and parts of messages MUST NOT be interleaved. So, for example, a PCC sending a PCReq and wishing to close the session, must complete sending the request message before starting to send a Close message.

Next Text:

Each PCEP message is regarded as a single transmission unit and parts of messages MUST NOT be interleaved. So, for example, a PCC sending a PCReq and wishing to close the session, must complete sending the request message before starting to send a Close message.

In special cases, if a timeout of the SendHoldTimer is detected, it is permissible to immediately send a Close message with the reason "SendHoldTimer_Expires."

3.2. CLOSE Object

In the Reason Value defined in the CLOSE Object in section 7.17, a new value "SendHoldTimer expired" is added.

Reasons

Value	Meaning
1	No explanation provided
2	DeadTimer expired
3	Reception of a malformed PCEP message
4	Reception of an unacceptable number of unknown requests/replies
5	Reception of an unacceptable number of unrecognized PCEP messages
TBD	SendHoldTimer expired (This Document)

3.3. PCEP Finite State Machine (FSM)

Appendix A. PCEP Finite State Machine (FSM), the handling in the UP state requires adding processing for SendHoldTimer.

Old Text:

If the TCP connection fails, the system releases the PCEP resources for that PCEP peer, closes the TCP connection, and moves to the Idle State.

Next Text:

If the TCP connection fails, or SendHoldTimer Expires, the system releases the PCEP resources for that PCEP peer, closes the TCP connection, and moves to the Idle State.

In "UP State", the handling of sending PCEP message succeed is revised as follows:

Old Text:

UP State:

In the UP state, the PCEP peer starts exchanging PCEP messages according to the session characteristics.

If the Keepalive timer expires, the system restarts the Keepalive timer and sends a Keepalive message.

UP State:

In the UP state, the PCEP peer starts exchanging PCEP messages according to the session characteristics.

If the Keepalive timer expires, the system restarts the Keepalive timer and sends a Keepalive message.

Restart the SendHoldTimer every time a PCEP message, including KeepAlive, PCReq, and PCRep, is successfully sent.

4. Security Considerations

This specification addresses the issue of potential attacks on PCEP routers, where PCEP peers pretend to be unable to process PCEP messages, thereby disrupting the normal operation of the PCEP protocol. In all other respects, this specification does not alter the security features of PCEP.

5. IANA Considerations

In the Reason Value defined in the CLOSE Object, a new value "SendHoldTimer expired" is added.

6. Acknowledgements

The authors wish to thank Job Snijders, Ben Cartwright-Cox, and Yingzhen Qu for their work on the BGP Send Hold Timer concept.

7. References

7.1. Normative References

- [I-D.ietf-idr-bgp-sendholdtimer] Snijders, J., Cartwright-Cox, B., and Y. Qu, "Border Gateway Protocol 4 (BGP-4) Send Hold Timer", Work in Progress, Internet-Draft, draft-ietf-idr-bgp-sendholdtimer, <<https://datatracker.ietf.org/doc/html/draft-ietf-idr-bgp-sendholdtimer>>
- [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>>.
- [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>>.
- [RFC5440] JP. Vasseur, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009, <<https://www.rfc-editor.org/info/rfc5440>>.

Authors' Addresses

Changwang Lin
New H3C Technologies
Beijing
China
Email: linchangwang.04414@h3c.com

