Internet Engineering Task Force (IETF)
Request for Comments: 7420
Category: Standards Track
ISSN: 2070-1721

A. Koushik
Brocade Communications, Inc.
E. Stephan
Orange
Q. Zhao
Huawei Technology
D. King
Old Dog Consulting
J. Hardwick
Metaswitch

December 2014

# Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module

## Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes managed objects for modeling of the Path Computation Element Communication Protocol (PCEP) for communications between a Path Computation Client (PCC) and a Path Computation Element (PCE), or between two PCEs.

## Status of This Memo

This is an Internet Standards Track document.

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). Further information on Internet Standards is available in 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/rfc7420.

Table of Contents

## 1. Introduction

The PCE defined in [RFC4655] is an entity that is capable of computing a network path or route based on a network graph and applying computational constraints. A PCC may make requests to a PCE for paths to be computed.

PCEP is the communication protocol between a PCC and PCE and is defined in [RFC5440]. PCEP interactions include path computation requests and path computation replies as well as notifications of specific states related to the use of a PCE in the context of Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) Traffic Engineering (TE).

This memo defines a portion of the MIB for use with network management protocols in the Internet community. In particular, it defines a MIB module that can be used to monitor PCEP interactions between a PCC and a PCE, or between two PCEs.

The scope of this document is to provide a MIB module for the PCEP base protocol defined in [RFC5440]. Extensions to the PCEP base protocol are beyond the scope for this document.

### 1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY" and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119].

### 1.2. Terminology

This document uses the terminology defined in [RFC4655] and [RFC5440]. In particular, it uses the following acronyms.

- Path Computation Request (PCReq) message.
- Path Computation Reply (PCRep) message.
- Notification (PCNtf) message.
- Error (PCErr) message.
- Request Parameter (RP) object.
- Synchronization Vector (SVEC) object.
- Explicit Route Object (ERO).

This document uses the term "PCEP entity" to refer to a local PCEP speaker, "peer" to refer to a remote PCEP speaker, and "PCEP speaker" where it is not necessary to distinguish between local and remote.

## 2. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to section 7 of RFC 3410 [RFC3410].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIv2, which is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579], and STD 58, RFC 2580 [RFC2580].

## 3. PCEP MIB Module Architecture

The PCEP MIB module contains the following information:

a. PCE and PCC local entity status (see pcePcepEntityTable).

b. PCEP peer information (see pcePcepPeerTable).

c. PCEP session information (see pcePcepSessTable).

d. Notifications to indicate PCEP session changes.

The PCEP MIB module is limited to "read-only" access except for pcePcepNotificationsMaxRate, which is used to throttle the rate at which the implementation generates notifications.

### 3.1. pcePcepEntityTable

The PCEP MIB module may contain status information for multiple logical local PCEP entities. There are several scenarios in which there may be more than one local PCEP entity, including the following.

- A physical router, which is partitioned into multiple virtual routers, each with its own PCC.

- A PCE device that front ends a cluster of compute resources, each with a different set of capabilities that are accessed via different IP addresses.

The pcePcepEntityTable contains one row for each local PCEP entity. Each row is read-only and contains current status information, plus the PCEP entity's running configuration.

The pcePcepEntityTable is indexed by pcePcepEntityIndex, which also acts as the primary index for the other tables in this MIB module.

## 3.2. pcePcepPeerTable

The pcePcepPeerTable contains one row for each peer that the local PCEP entity knows about. Each row is read-only and contains information to identify the peer, the running configuration relating to that peer, and statistics that track the messages exchanged with that peer and its response times.

A PCEP speaker is identified by its IP address. If there is a PCEP speaker in the network that uses multiple IP addresses, then it looks like multiple distinct peers to the other PCEP speakers in the network.

The pcePcepPeerTable is indexed first by pcePcepEntityIndex, then by pcePcepPeerAddrType and pcePcepPeerAddr. This indexing structure allows each local PCEP entity to report its own set of peers.

Since PCEP sessions can be ephemeral, pcePcepPeerTable tracks a peer even when no PCEP session currently exists to that peer. The statistics contained in pcePcepPeerTable are an aggregate of the statistics for all successive sessions to that peer.

To limit the quantity of information that is stored, an implementation MAY choose to discard a row from pcePcepPeerTable if and only if no PCEP session exists to the corresponding peer.

## 3.3. pcePcepSessTable

The pcePcepSessTable contains one row for each PCEP session that the PCEP entity (PCE or PCC) is currently participating in. Each row is read-only and contains the running configuration that is applied to the session, plus identifiers and statistics for the session.

The statistics in pcePcepSessTable are semantically different from those in pcePcepPeerTable since the former applies to the current session only, whereas the latter is the aggregate for all sessions that have existed to that peer.

Although it is forbidden per [RFC5440] to have more than one active PCEP session between a given pair of PCEP entities at any one time, there is a window during session establishment where the

pcePcepSessTable may contain two rows for a given peer, one representing a session initiated by the local PCEP entity and one representing a session initiated by the peer. If either of these sessions reaches an active state, then the other is discarded.

The pcePcepSessTable is indexed first by pcePcepEntityIndex, then by pcePcepPeerAddrType and pcePcepPeerAddr, and finally by pcePcepSessInitiator. This indexing structure allows each local PCEP entity to report its own set of active sessions. The pcePcepSessInitiator index allows two rows to exist transiently for a given peer, as discussed above.

## 3.4. PCEP Notifications

The PCEP MIB module contains notifications for the following conditions.

a. pcePcepSessUp: PCEP session has gone up.

b. pcePcepSessDown: PCEP session has gone down.

c. pcePcepSessLocalOverload: Local PCEP entity has sent an overload PCNtf on this session.

d. pcePcepSessLocalOverloadClear: Local PCEP entity has sent an overload-cleared PCNtf on this session.

e. pcePcepSessPeerOverload: Peer has sent an overload PCNtf on this session.

f. pcePcepSessPeerOverloadClear: Peer has sent an overload-cleared PCNtf on this session.

## 3.5. Relationship to Other MIB Modules

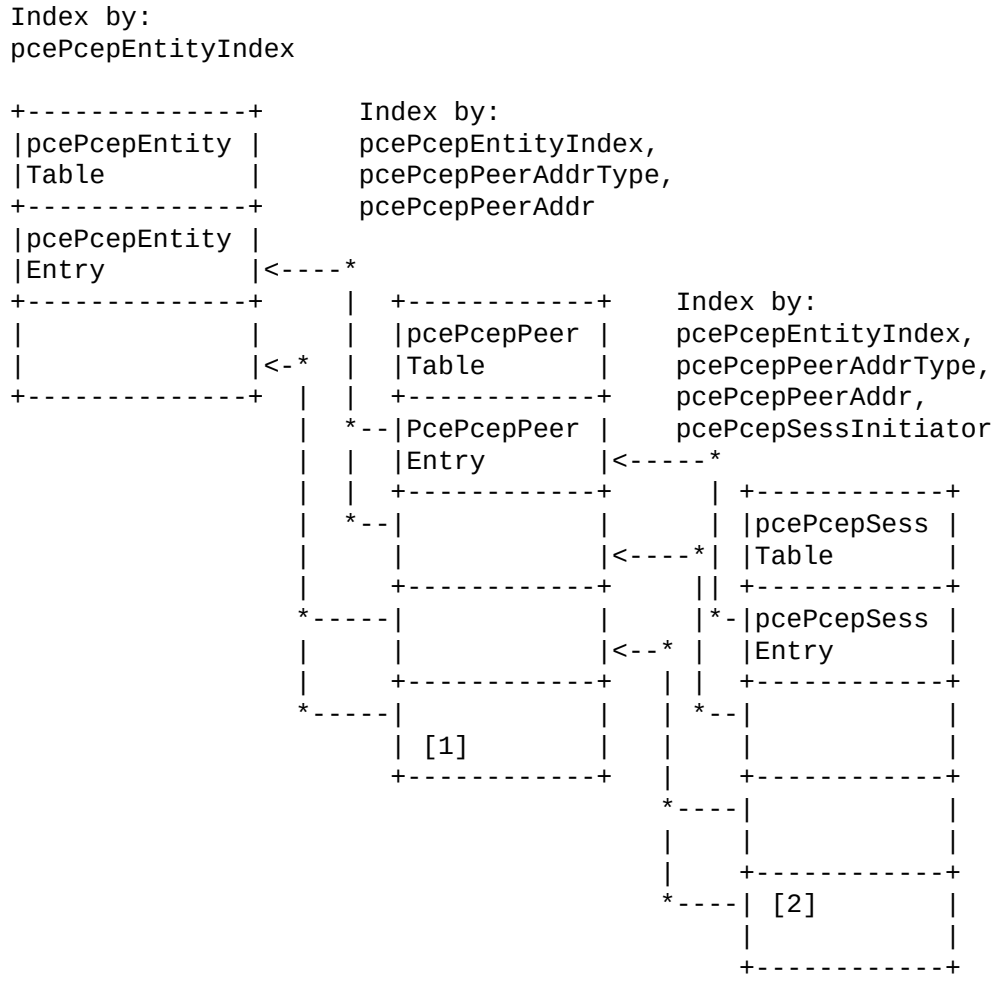
The PCEP MIB module imports the following textual conventions from the INET-ADDRESS-MIB defined in RFC 4001 [RFC4001]:

- InetAddressType
- InetAddress

PCEP relies on existing protocols that have specialized MIB objects to monitor their own activities. Consequently, this document considers that the monitoring of underlying protocols is out of scope of the PCEP MIB module.

## 3.6. Illustrative Example

The following diagram illustrates the relationships between pcePcepEntityTable, pcePcepPeerTable, and pcePcepSessTable.




[1]: A peer entry with no current session.
[2]: Two sessions exist during a window in session initialization.

## 4. Object Definitions

### 4.1. PCE-PCEP-MIB

```
   PCE-PCEP-MIB DEFINITIONS ::= BEGIN

   IMPORTS
       MODULE-IDENTITY,
       OBJECT-TYPE,
       mib-2,
       NOTIFICATION-TYPE,
       Unsigned32,
       Counter32
              FROM SNMPv2-SMI             -- RFC 2578
       TruthValue,
       TimeStamp
              FROM SNMPv2-TC              -- RFC 2579
       MODULE-COMPLIANCE,
       OBJECT-GROUP,
       NOTIFICATION-GROUP
              FROM SNMPv2-CONF            -- RFC 2580
       InetAddressType,
       InetAddress
              FROM INET-ADDRESS-MIB;      -- RFC 4001

   pcePcepMIB MODULE-IDENTITY
       LAST-UPDATED
           "201412171200Z" -- 17 December 2014
       ORGANIZATION
           "IETF Path Computation Element (PCE) Working Group"
       CONTACT-INFO
           "Email: pce@ietf.org
            WG charter:
                     http://datatracker.ietf.org/wg/pce/charter/"

       DESCRIPTION
         "This MIB module defines a collection of objects for managing
          the Path Computation Element Communication Protocol (PCEP).

          Copyright (c) 2014 IETF Trust and the persons identified as
          authors of the code.  All rights reserved.

          Redistribution and use in source and binary forms, with or
          without modification, is permitted pursuant to, and subject
          to the license terms contained in, the Simplified BSD License
          set forth in Section 4.c of the IETF Trust's Legal Provisions
          Relating to IETF Documents
          (http://trustee.ietf.org/license-info)."
```

```
        REVISION
            "201412171200Z" -- 17 December 2014
        DESCRIPTION
            "Initial version, published as RFC 7420."
        ::= { mib-2 227 }

    pcePcepNotifications OBJECT IDENTIFIER ::= { pcePcepMIB 0 }
    pcePcepObjects       OBJECT IDENTIFIER ::= { pcePcepMIB 1 }
    pcePcepConformance   OBJECT IDENTIFIER ::= { pcePcepMIB 2 }

    --
    -- PCEP Entity Objects
    --

    pcePcepEntityTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF PcePcepEntityEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "This table contains information about local PCEP entities.
             The entries in this table are read-only."
        ::= { pcePcepObjects 1 }

    pcePcepEntityEntry OBJECT-TYPE
        SYNTAX      PcePcepEntityEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "This entry represents a local PCEP entity."
        INDEX       {  pcePcepEntityIndex  }
        ::= { pcePcepEntityTable 1 }

    PcePcepEntityEntry ::= SEQUENCE {
        pcePcepEntityIndex                Unsigned32,
        pcePcepEntityAdminStatus          INTEGER,
        pcePcepEntityOperStatus           INTEGER,
        pcePcepEntityAddrType             InetAddressType,
        pcePcepEntityAddr                 InetAddress,
        pcePcepEntityConnectTimer         Unsigned32,
        pcePcepEntityConnectMaxRetry      Unsigned32,
        pcePcepEntityInitBackoffTimer     Unsigned32,
        pcePcepEntityMaxBackoffTimer      Unsigned32,
        pcePcepEntityOpenWaitTimer        Unsigned32,
        pcePcepEntityKeepWaitTimer        Unsigned32,
        pcePcepEntityKeepAliveTimer       Unsigned32,
        pcePcepEntityDeadTimer            Unsigned32,
        pcePcepEntityAllowNegotiation     TruthValue,
        pcePcepEntityMaxKeepAliveTimer    Unsigned32,
```

```
        pcePcepEntityMaxDeadTimer          Unsigned32,
        pcePcepEntityMinKeepAliveTimer     Unsigned32,
        pcePcepEntityMinDeadTimer          Unsigned32,
        pcePcepEntitySyncTimer             Unsigned32,
        pcePcepEntityRequestTimer          Unsigned32,
        pcePcepEntityMaxSessions           Unsigned32,
        pcePcepEntityMaxUnknownReqs        Unsigned32,
        pcePcepEntityMaxUnknownMsgs        Unsigned32
    }

    pcePcepEntityIndex OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "This index is used to uniquely identify the PCEP entity."
        ::= { pcePcepEntityEntry 1 }

    pcePcepEntityAdminStatus OBJECT-TYPE
        SYNTAX      INTEGER {
                      adminStatusUp(1),
                      adminStatusDown(2)
                    }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The administrative status of this PCEP entity.

             This is the desired operational status as currently set by
             an operator or by default in the implementation.  The value
             of pcePcepEntityOperStatus represents the current status of
             an attempt to reach this desired status."
        ::= { pcePcepEntityEntry 2 }

    pcePcepEntityOperStatus OBJECT-TYPE
        SYNTAX      INTEGER {
                      operStatusUp(1),
                      operStatusDown(2),
                      operStatusGoingUp(3),
                      operStatusGoingDown(4),
                      operStatusFailed(5),
                      operStatusFailedPerm(6)
                    }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The operational status of the PCEP entity.  It takes one of
             the following values.
```

```
            - operStatusUp(1): the PCEP entity is active.
            - operStatusDown(2): the PCEP entity is inactive.
            - operStatusGoingUp(3): the PCEP entity is activating.
            - operStatusGoingDown(4): the PCEP entity is deactivating.
            - operStatusFailed(5): the PCEP entity has failed and will
              recover when possible.
            - operStatusFailedPerm(6): the PCEP entity has failed and
              will not recover without operator intervention."
        ::= { pcePcepEntityEntry 3 }

    pcePcepEntityAddrType OBJECT-TYPE
        SYNTAX      InetAddressType
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The type of the PCEP entity's Internet address.  This object
             specifies how the value of the pcePcepEntityAddr object
             should be interpreted.  Only values unknown(0), ipv4(1), or
             ipv6(2) are supported."
        ::= { pcePcepEntityEntry 4 }

    pcePcepEntityAddr OBJECT-TYPE
        SYNTAX      InetAddress
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The local Internet address of this PCEP entity.  The type is
             given by pcePcepEntityAddrType.

             If operating as a PCE server, the PCEP entity listens on
             this address.  If operating as a PCC, the PCEP entity binds
             outgoing TCP connections to this address.

             It is possible for the PCEP entity to operate both as a PCC
             and a PCE server, in which case it uses this address both to
             listen for incoming TCP connections and to bind outgoing
             TCP connections."
        ::= { pcePcepEntityEntry 5 }

    pcePcepEntityConnectTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (1..65535)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
```

```
        DESCRIPTION
            "The time that the PCEP entity will wait to establish a TCP
             connection with a peer.  If a TCP connection is not
             established within this time, then PCEP aborts the session
             setup attempt."
        ::= { pcePcepEntityEntry 6 }

    pcePcepEntityConnectMaxRetry OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The maximum number of times the system tries to establish
             a TCP connection to a peer before the session with the peer
             transitions to the idle state.

             When the session transitions to the idle state:
             - pcePcepPeerSessionExists transitions to false(2).
             - the associated PcePcepSessEntry is deleted.
             - a backoff timer runs before the session is tried again."
        ::= { pcePcepEntityEntry 7 }

    pcePcepEntityInitBackoffTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (1..65535)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The initial backoff time for retrying a failed session
             setup attempt to a peer.

             The backoff time increases for each failed session setup
             attempt, until a maximum backoff time is reached.  The
             maximum backoff time is pcePcepEntityMaxBackoffTimer."
        ::= { pcePcepEntityEntry 8 }

    pcePcepEntityMaxBackoffTimer OBJECT-TYPE
        SYNTAX      Unsigned32
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The maximum backoff time for retrying a failed session
             setup attempt to a peer.
```

```
            The backoff time increases for each failed session setup
            attempt, until this maximum value is reached.  Session
            setup attempts then repeats periodically without any
            further increase in backoff time."
        ::= { pcePcepEntityEntry 9 }

    pcePcepEntityOpenWaitTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (1..65535)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The time that the PCEP entity will wait to receive an Open
             message from a peer after the TCP connection has come up.
             If no Open message is received within this time, then PCEP
             terminates the TCP connection and deletes the associated
             PcePcepSessEntry."
        ::= { pcePcepEntityEntry 10 }

    pcePcepEntityKeepWaitTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (1..65535)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The time that the PCEP entity will wait to receive a
             Keepalive or PCErr message from a peer during session
             initialization after receiving an Open message.  If no
             Keepalive or PCErr message is received within this time,
             then PCEP terminates the TCP connection and deletes the
             associated PcePcepSessEntry."
        ::= { pcePcepEntityEntry 11 }

    pcePcepEntityKeepAliveTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The Keepalive transmission timer that this PCEP entity will
             propose in the initial OPEN message of each session it is
             involved in.  This is the maximum time between two
             consecutive messages sent to a peer.  Zero means that
             the PCEP entity prefers not to send Keepalives at all.

             Note that the actual Keepalive transmission intervals, in
             either direction of an active PCEP session, are determined
             by negotiation between the peers as specified by RFC
```

```
            5440, and so may differ from this configured value.  For
            the actually negotiated values (per session), see
            pcePcepSessKeepaliveTimer and
            pcePcepSessPeerKeepaliveTimer."
        ::= { pcePcepEntityEntry 12 }

    pcePcepEntityDeadTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The DeadTimer that this PCEP entity will propose in the
             initial OPEN message of each session it is involved in.
             This is the time after which a peer should declare a
             session down if it does not receive any PCEP messages.
             Zero suggests that the peer does not run a DeadTimer at
             all."
        ::= { pcePcepEntityEntry 13 }

    pcePcepEntityAllowNegotiation OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "Whether the PCEP entity will permit negotiation of session
             parameters."
        ::= { pcePcepEntityEntry 14 }

    pcePcepEntityMaxKeepAliveTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "In PCEP session parameter negotiation, the maximum value
             that this PCEP entity will accept from a peer for the
             interval between Keepalive transmissions.  Zero means that
             the PCEP entity will allow no Keepalive transmission at
             all."
        ::= { pcePcepEntityEntry 15 }

    pcePcepEntityMaxDeadTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
```

```
        DESCRIPTION
            "In PCEP session parameter negotiation, the maximum value
             that this PCEP entity will accept from a peer for the
             DeadTimer.  Zero means that the PCEP entity will allow not
             running a DeadTimer."
        ::= { pcePcepEntityEntry 16 }

    pcePcepEntityMinKeepAliveTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "In PCEP session parameter negotiation, the minimum value
             that this PCEP entity will accept for the interval between
             Keepalive transmissions.  Zero means that the PCEP entity
             insists on no Keepalive transmission at all."
        ::= { pcePcepEntityEntry 17 }

    pcePcepEntityMinDeadTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "In PCEP session parameter negotiation, the minimum value
             that this PCEP entity will accept for the DeadTimer.  Zero
             means that the PCEP entity insists on not running a
             DeadTimer."
        ::= { pcePcepEntityEntry 18 }

    pcePcepEntitySyncTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (0..65535)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The value of SyncTimer is used in the case of a synchronized
             path computation request using the SVEC object.

             Consider the case where a PCReq message is received by a PCE
             that contains the SVEC object referring to M synchronized
             path computation requests.  If after the expiration of the
             SyncTimer all the M path computation requests have not been
             received, a protocol error is triggered and the PCE MUST
             cancel the whole set of path computation requests.
```

```
            The aim of the SyncTimer is to avoid the storage of unused
            synchronized requests should one of them get lost for some
            reason (for example, a misbehaving PCC).

            A value of zero is returned if and only if the entity does
            not use the SyncTimer."
        ::= { pcePcepEntityEntry 19 }

    pcePcepEntityRequestTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (1..65535)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The maximum time that the PCEP entity will wait for a
             response to a PCReq message."
        ::= { pcePcepEntityEntry 20 }

    pcePcepEntityMaxSessions OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The maximum number of sessions involving this PCEP entity
             that can exist at any time."
        ::= { pcePcepEntityEntry 21 }

    pcePcepEntityMaxUnknownReqs OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The maximum number of unrecognized requests and replies that
             any session on this PCEP entity is willing to accept per
             minute before terminating the session.

             A PCRep message contains an unrecognized reply if it
             contains an RP object whose request ID does not correspond
             to any in-progress request sent by this PCEP entity.

             A PCReq message contains an unrecognized request if it
             contains an RP object whose request ID is zero."
        ::= { pcePcepEntityEntry 22 }

    pcePcepEntityMaxUnknownMsgs OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS  read-only
        STATUS      current
```

```
        DESCRIPTION
            "The maximum number of unknown messages that any session
             on this PCEP entity is willing to accept per minute before
             terminating the session."
        ::= { pcePcepEntityEntry 23 }

    --
    -- The PCEP Peer Table
    --

    pcePcepPeerTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF PcePcepPeerEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "This table contains information about peers known by
             the local PCEP entity.  The entries in this table are
             read-only.

             This table gives peer information that spans PCEP
             sessions.  Information about current PCEP sessions can be
             found in the pcePcepSessTable table."
        ::= { pcePcepObjects 2 }

    pcePcepPeerEntry OBJECT-TYPE
        SYNTAX      PcePcepPeerEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "Information about a single peer that spans all PCEP
             sessions to that peer."
        INDEX { pcePcepEntityIndex,
                pcePcepPeerAddrType,
                pcePcepPeerAddr }
        ::= { pcePcepPeerTable 1 }

    PcePcepPeerEntry ::= SEQUENCE {
        pcePcepPeerAddrType                 InetAddressType,
        pcePcepPeerAddr                     InetAddress,
        pcePcepPeerRole                     INTEGER,
        pcePcepPeerDiscontinuityTime        TimeStamp,
        pcePcepPeerInitiateSession          TruthValue,
        pcePcepPeerSessionExists            TruthValue,
        pcePcepPeerNumSessSetupOK           Counter32,
        pcePcepPeerNumSessSetupFail         Counter32,
        pcePcepPeerSessionUpTime            TimeStamp,
        pcePcepPeerSessionFailTime          TimeStamp,
        pcePcepPeerSessionFailUpTime        TimeStamp,
```

```
        pcePcepPeerAvgRspTime               Unsigned32,
        pcePcepPeerLWMRspTime               Unsigned32,
        pcePcepPeerHWMRspTime               Unsigned32,
        pcePcepPeerNumPCReqSent             Counter32,
        pcePcepPeerNumPCReqRcvd             Counter32,
        pcePcepPeerNumPCRepSent             Counter32,
        pcePcepPeerNumPCRepRcvd             Counter32,
        pcePcepPeerNumPCErrSent             Counter32,
        pcePcepPeerNumPCErrRcvd             Counter32,
        pcePcepPeerNumPCNtfSent             Counter32,
        pcePcepPeerNumPCNtfRcvd             Counter32,
        pcePcepPeerNumKeepaliveSent         Counter32,
        pcePcepPeerNumKeepaliveRcvd         Counter32,
        pcePcepPeerNumUnknownRcvd           Counter32,
        pcePcepPeerNumCorruptRcvd           Counter32,
        pcePcepPeerNumReqSent               Counter32,
        pcePcepPeerNumSvecSent              Counter32,
        pcePcepPeerNumSvecReqSent           Counter32,
        pcePcepPeerNumReqSentPendRep        Counter32,
        pcePcepPeerNumReqSentEroRcvd        Counter32,
        pcePcepPeerNumReqSentNoPathRcvd     Counter32,
        pcePcepPeerNumReqSentCancelRcvd     Counter32,
        pcePcepPeerNumReqSentErrorRcvd      Counter32,
        pcePcepPeerNumReqSentTimeout        Counter32,
        pcePcepPeerNumReqSentCancelSent     Counter32,
        pcePcepPeerNumReqSentClosed         Counter32,
        pcePcepPeerNumReqRcvd               Counter32,
        pcePcepPeerNumSvecRcvd              Counter32,
        pcePcepPeerNumSvecReqRcvd           Counter32,
        pcePcepPeerNumReqRcvdPendRep        Counter32,
        pcePcepPeerNumReqRcvdEroSent        Counter32,
        pcePcepPeerNumReqRcvdNoPathSent     Counter32,
        pcePcepPeerNumReqRcvdCancelSent     Counter32,
        pcePcepPeerNumReqRcvdErrorSent      Counter32,
        pcePcepPeerNumReqRcvdCancelRcvd     Counter32,
        pcePcepPeerNumReqRcvdClosed         Counter32,
        pcePcepPeerNumRepRcvdUnknown        Counter32,
        pcePcepPeerNumReqRcvdUnknown        Counter32
    }

    pcePcepPeerAddrType OBJECT-TYPE
        SYNTAX      InetAddressType
        MAX-ACCESS  not-accessible
        STATUS      current
```

```
        DESCRIPTION
            "The type of the peer's Internet address.  This object
             specifies how the value of the pcePcepPeerAddr object should
             be interpreted.  Only values unknown(0), ipv4(1), or
             ipv6(2) are supported."
        ::= { pcePcepPeerEntry 1 }

    pcePcepPeerAddr OBJECT-TYPE
        SYNTAX      InetAddress
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "The Internet address of the peer.  The type is given by
             pcePcepPeerAddrType."
        ::= { pcePcepPeerEntry 2 }

    pcePcepPeerRole OBJECT-TYPE
        SYNTAX      INTEGER {
                      unknown(0),
                      pcc(1),
                      pce(2),
                      pccAndPce(3)
                    }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The role that this peer took the last time a session was
             established.  It takes one of the following values.
             - unknown(0): this peer's role is not known.
             - pcc(1): this peer is a Path Computation Client (PCC).
             - pce(2): this peer is a Path Computation Element (PCE).
             - pccAndPce(3): this peer is both a PCC and a PCE."
        ::= { pcePcepPeerEntry 3 }

    pcePcepPeerDiscontinuityTime OBJECT-TYPE
        SYNTAX      TimeStamp
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The value of sysUpTime at the time that the information and
             statistics in this row were last reset."
        ::= { pcePcepPeerEntry 4 }

    pcePcepPeerInitiateSession OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
```

```
        DESCRIPTION
            "Indicates whether the local PCEP entity initiates sessions
             to this peer or waits for the peer to initiate a session."
        ::= { pcePcepPeerEntry 5 }

    pcePcepPeerSessionExists OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "Indicates whether a session with this peer currently
             exists."
        ::= { pcePcepPeerEntry 6 }

    pcePcepPeerNumSessSetupOK OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of PCEP sessions successfully established with
             the peer, including any current session.  This counter is
             incremented each time a session with this peer is
             successfully established."
        ::= { pcePcepPeerEntry 7 }

    pcePcepPeerNumSessSetupFail OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of PCEP sessions with the peer that have been
             attempted but failed before being fully established.
             This counter is incremented each time a session retry to
             this peer fails."
        ::= { pcePcepPeerEntry 8 }

    pcePcepPeerSessionUpTime OBJECT-TYPE
        SYNTAX      TimeStamp
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The value of sysUpTime the last time a session with this
             peer was successfully established.

             If pcePcepPeerNumSessSetupOK is zero, then this object
             contains zero."
        ::= { pcePcepPeerEntry 9 }
```

```
   pcePcepPeerSessionFailTime OBJECT-TYPE
       SYNTAX      TimeStamp
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The value of sysUpTime the last time a session with this
            peer failed to be established.

            If pcePcepPeerNumSessSetupFail is zero, then this object
            contains zero."
       ::= { pcePcepPeerEntry 10 }

   pcePcepPeerSessionFailUpTime OBJECT-TYPE
       SYNTAX      TimeStamp
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The value of sysUpTime the last time a session with this
            peer failed from active.

            If pcePcepPeerNumSessSetupOK is zero, then this object
            contains zero."
       ::= { pcePcepPeerEntry 11 }

   pcePcepPeerAvgRspTime OBJECT-TYPE
       SYNTAX      Unsigned32
       UNITS       "milliseconds"
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The average response time for this peer.

            If an average response time has not been calculated for this
            peer, then this object has the value zero.

            If pcePcepPeerRole is pcc, then this field is meaningless
            and is set to zero."
       ::= { pcePcepPeerEntry 12 }

   pcePcepPeerLWMRspTime OBJECT-TYPE
       SYNTAX      Unsigned32
       UNITS       "milliseconds"
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The smallest (low-water mark) response time seen from this
            peer.
```

```
            If no responses have been received from this peer, then this
            object has the value zero.

            If pcePcepPeerRole is pcc, then this field is meaningless
            and is set to zero."
        ::= { pcePcepPeerEntry 13 }

    pcePcepPeerHWMRspTime OBJECT-TYPE
        SYNTAX      Unsigned32
        UNITS       "milliseconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The greatest (high-water mark) response time seen from this
             peer.

             If no responses have been received from this peer, then this
             object has the value zero.

             If pcePcepPeerRole is pcc, then this field is meaningless
             and is set to zero."
        ::= { pcePcepPeerEntry 14 }

    pcePcepPeerNumPCReqSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of PCReq messages sent to this peer."
        ::= { pcePcepPeerEntry 15 }

    pcePcepPeerNumPCReqRcvd OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of PCReq messages received from this peer."
        ::= { pcePcepPeerEntry 16 }

    pcePcepPeerNumPCRepSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of PCRep messages sent to this peer."
        ::= { pcePcepPeerEntry 17 }
```

```
   pcePcepPeerNumPCRepRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCRep messages received from this peer."
       ::= { pcePcepPeerEntry 18 }

   pcePcepPeerNumPCErrSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCErr messages sent to this peer."
       ::= { pcePcepPeerEntry 19 }

   pcePcepPeerNumPCErrRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCErr messages received from this peer."
       ::= { pcePcepPeerEntry 20 }

   pcePcepPeerNumPCNtfSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCNtf messages sent to this peer."
       ::= { pcePcepPeerEntry 21 }

   pcePcepPeerNumPCNtfRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCNtf messages received from this peer."
       ::= { pcePcepPeerEntry 22 }

   pcePcepPeerNumKeepaliveSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of Keepalive messages sent to this peer."
       ::= { pcePcepPeerEntry 23 }
```

```
   pcePcepPeerNumKeepaliveRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of Keepalive messages received from this peer."
       ::= { pcePcepPeerEntry 24 }

   pcePcepPeerNumUnknownRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of unknown messages received from this peer."
       ::= { pcePcepPeerEntry 25 }

   pcePcepPeerNumCorruptRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of corrupted PCEP messages received from this
            peer."
       ::= { pcePcepPeerEntry 26 }

   pcePcepPeerNumReqSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests sent to this peer.  A request
            corresponds 1:1 with an RP object in a PCReq message.

            This might be greater than pcePcepPeerNumPCReqSent because
            multiple requests can be batched into a single PCReq
            message."
       ::= { pcePcepPeerEntry 27 }

   pcePcepPeerNumSvecSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of SVEC objects sent to this peer in PCReq
            messages.  An SVEC object represents a set of synchronized
            requests."
       ::= { pcePcepPeerEntry 28 }
```

```
   pcePcepPeerNumSvecReqSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests sent to this peer that appeared in
            one or more SVEC objects."
       ::= { pcePcepPeerEntry 29 }

   pcePcepPeerNumReqSentPendRep OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests that have been sent to this peer for
            which a response is still pending."
       ::= { pcePcepPeerEntry 30 }

   pcePcepPeerNumReqSentEroRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests that have been sent to this peer for
            which a response with an ERO was
            received.  Such responses indicate that a path was
            successfully computed by the peer."
       ::= { pcePcepPeerEntry 31 }

   pcePcepPeerNumReqSentNoPathRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests that have been sent to this peer for
            which a response with a NO-PATH object was received.  Such
            responses indicate that the peer could not find a path to
            satisfy the request."
       ::= { pcePcepPeerEntry 32 }

   pcePcepPeerNumReqSentCancelRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests that were canceled by the peer with
            a PCNtf message.
```

```
            This might be different than pcePcepPeerNumPCNtfRcvd because
            not all PCNtf messages are used to cancel requests, and a
            single PCNtf message can cancel multiple requests."
        ::= { pcePcepPeerEntry 33 }

    pcePcepPeerNumReqSentErrorRcvd OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests that were rejected by the peer with a
             PCErr message.

             This might be different than pcePcepPeerNumPCErrRcvd because
             not all PCErr messages are used to reject requests, and a
             single PCErr message can reject multiple requests."
        ::= { pcePcepPeerEntry 34 }

    pcePcepPeerNumReqSentTimeout OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests that have been sent to a peer and
             have been abandoned because the peer has taken too long to
             respond to them."
        ::= { pcePcepPeerEntry 35 }

    pcePcepPeerNumReqSentCancelSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests that were sent to the peer and
             explicitly canceled by the local PCEP entity sending a
             PCNtf."
        ::= { pcePcepPeerEntry 36 }

    pcePcepPeerNumReqSentClosed OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests that were sent to the peer and
             implicitly canceled when the session they were sent over was
             closed."
        ::= { pcePcepPeerEntry 37 }
```

```
   pcePcepPeerNumReqRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests received from this peer.  A request
            corresponds 1:1 with an RP object in a PCReq message.

            This might be greater than pcePcepPeerNumPCReqRcvd because
            multiple requests can be batched into a single PCReq
            message."
       ::= { pcePcepPeerEntry 38 }

   pcePcepPeerNumSvecRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of SVEC objects received from this peer in PCReq
            messages.  An SVEC object represents a set of synchronized
            requests."
       ::= { pcePcepPeerEntry 39 }

   pcePcepPeerNumSvecReqRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests received from this peer that appeared
            in one or more SVEC objects."
       ::= { pcePcepPeerEntry 40 }

   pcePcepPeerNumReqRcvdPendRep OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests that have been received from this
            peer for which a response is still pending."
       ::= { pcePcepPeerEntry 41 }

   pcePcepPeerNumReqRcvdEroSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
```

```
        DESCRIPTION
            "The number of requests that have been received from this
             peer for which a response with an ERO was sent.  Such
             responses indicate that a path was successfully computed by
             the local PCEP entity."
        ::= { pcePcepPeerEntry 42 }

    pcePcepPeerNumReqRcvdNoPathSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests that have been received from this
             peer for which a response with a NO-PATH object was sent.
             Such responses indicate that the local PCEP entity could
             not find a path to satisfy the request."
        ::= { pcePcepPeerEntry 43 }

    pcePcepPeerNumReqRcvdCancelSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests received from this peer that were
             canceled by the local PCEP entity sending a PCNtf message.

             This might be different than pcePcepPeerNumPCNtfSent because
             not all PCNtf messages are used to cancel requests, and a
             single PCNtf message can cancel multiple requests."
        ::= { pcePcepPeerEntry 44 }

    pcePcepPeerNumReqRcvdErrorSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests received from this peer that were
             rejected by the local PCEP entity sending a PCErr message.

             This might be different than pcePcepPeerNumPCErrSent because
             not all PCErr messages are used to reject requests, and a
             single PCErr message can reject multiple requests."
        ::= { pcePcepPeerEntry 45 }

    pcePcepPeerNumReqRcvdCancelRcvd OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
```

```
        DESCRIPTION
            "The number of requests that were received from the peer and
             explicitly canceled by the peer sending a PCNtf."
        ::= { pcePcepPeerEntry 46 }

    pcePcepPeerNumReqRcvdClosed OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests that were received from the peer and
             implicitly canceled when the session they were received over
             was closed."
        ::= { pcePcepPeerEntry 47 }

    pcePcepPeerNumRepRcvdUnknown OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of responses to unknown requests received from
             this peer.  A response to an unknown request is a response
             whose RP object does not contain the request ID of any
             request that is currently outstanding on the session."
        ::= { pcePcepPeerEntry 48 }

    pcePcepPeerNumReqRcvdUnknown OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of unknown requests that have been received from
             a peer.  An unknown request is a request whose RP object
             contains a request ID of zero."
        ::= { pcePcepPeerEntry 49 }

    --
    -- The PCEP Sessions Table
    --

    pcePcepSessTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF PcePcepSessEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "A table of PCEP sessions that involve the local PCEP
             entity.  Each entry in this table represents a single
             session.  The entries in this table are read-only.
```

```
            An entry appears in this table when the corresponding PCEP
            session transitions out of idle state.  If the PCEP session
            transitions back into an idle state, then the corresponding
            entry in this table is removed."
        ::= { pcePcepObjects 3 }

    pcePcepSessEntry OBJECT-TYPE
        SYNTAX      PcePcepSessEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "This entry represents a single PCEP session in which the
             local PCEP entity participates.

             This entry exists only if the corresponding PCEP session has
             been initialized by some event, such as manual user
             configuration, auto-discovery of a peer, or an incoming TCP
             connection."
        INDEX { pcePcepEntityIndex,
                pcePcepPeerAddrType,
                pcePcepPeerAddr,
                pcePcepSessInitiator }
        ::= { pcePcepSessTable 1 }

    PcePcepSessEntry ::= SEQUENCE {
        pcePcepSessInitiator                INTEGER,
        pcePcepSessStateLastChange          TimeStamp,
        pcePcepSessState                    INTEGER,
        pcePcepSessConnectRetry             Counter32,
        pcePcepSessLocalID                  Unsigned32,
        pcePcepSessRemoteID                 Unsigned32,
        pcePcepSessKeepaliveTimer           Unsigned32,
        pcePcepSessPeerKeepaliveTimer       Unsigned32,
        pcePcepSessDeadTimer                Unsigned32,
        pcePcepSessPeerDeadTimer            Unsigned32,
        pcePcepSessKAHoldTimeRem            Unsigned32,
        pcePcepSessOverloaded               TruthValue,
        pcePcepSessOverloadTime             Unsigned32,
        pcePcepSessPeerOverloaded           TruthValue,
        pcePcepSessPeerOverloadTime         Unsigned32,
        pcePcepSessDiscontinuityTime        TimeStamp,
        pcePcepSessAvgRspTime               Unsigned32,
        pcePcepSessLWMRspTime               Unsigned32,
        pcePcepSessHWMRspTime               Unsigned32,
        pcePcepSessNumPCReqSent             Counter32,
        pcePcepSessNumPCReqRcvd             Counter32,
        pcePcepSessNumPCRepSent             Counter32,
        pcePcepSessNumPCRepRcvd             Counter32,
```

```
      pcePcepSessNumPCErrSent             Counter32,
      pcePcepSessNumPCErrRcvd             Counter32,
      pcePcepSessNumPCNtfSent             Counter32,
      pcePcepSessNumPCNtfRcvd             Counter32,
      pcePcepSessNumKeepaliveSent         Counter32,
      pcePcepSessNumKeepaliveRcvd         Counter32,
      pcePcepSessNumUnknownRcvd           Counter32,
      pcePcepSessNumCorruptRcvd           Counter32,
      pcePcepSessNumReqSent               Counter32,
      pcePcepSessNumSvecSent              Counter32,
      pcePcepSessNumSvecReqSent           Counter32,
      pcePcepSessNumReqSentPendRep        Counter32,
      pcePcepSessNumReqSentEroRcvd        Counter32,
      pcePcepSessNumReqSentNoPathRcvd     Counter32,
      pcePcepSessNumReqSentCancelRcvd     Counter32,
      pcePcepSessNumReqSentErrorRcvd      Counter32,
      pcePcepSessNumReqSentTimeout        Counter32,
      pcePcepSessNumReqSentCancelSent     Counter32,
      pcePcepSessNumReqRcvd               Counter32,
      pcePcepSessNumSvecRcvd              Counter32,
      pcePcepSessNumSvecReqRcvd           Counter32,
      pcePcepSessNumReqRcvdPendRep        Counter32,
      pcePcepSessNumReqRcvdEroSent        Counter32,
      pcePcepSessNumReqRcvdNoPathSent     Counter32,
      pcePcepSessNumReqRcvdCancelSent     Counter32,
      pcePcepSessNumReqRcvdErrorSent      Counter32,
      pcePcepSessNumReqRcvdCancelRcvd     Counter32,
      pcePcepSessNumRepRcvdUnknown        Counter32,
      pcePcepSessNumReqRcvdUnknown        Counter32
  }

  pcePcepSessInitiator OBJECT-TYPE
      SYNTAX      INTEGER {
                     local(1),
                     remote(2)
                  }
      MAX-ACCESS  not-accessible
      STATUS      current
      DESCRIPTION
          "The initiator of the session; that is, whether the TCP
           connection was initiated by the local PCEP entity or the
           peer.

           There is a window during session initialization where two
           sessions can exist between a pair of PCEP speakers, each
           initiated by one of the speakers.  One of these sessions is
           always discarded before it leaves OpenWait state.  However,
           before it is discarded, two sessions to the given peer
```

```
            appear transiently in this MIB module.  The sessions are
            distinguished by who initiated them, and so this field is an
            index for pcePcepSessTable."
        ::= { pcePcepSessEntry 1 }

    pcePcepSessStateLastChange OBJECT-TYPE
        SYNTAX      TimeStamp
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The value of sysUpTime at the time this session entered its
             current state as denoted by the pcePcepSessState object."
        ::= { pcePcepSessEntry 2 }

    pcePcepSessState OBJECT-TYPE
        SYNTAX      INTEGER {
                       tcpPending(1),
                       openWait(2),
                       keepWait(3),
                       sessionUp(4)
                    }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The current state of the session.

             The set of possible states excludes the idle state since
             entries do not exist in this table in the idle state."
        ::= { pcePcepSessEntry 3 }

    pcePcepSessConnectRetry OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of times that the local PCEP entity has
             attempted to establish a TCP connection for this session
             without success.  The PCEP entity gives up when this
             reaches pcePcepEntityConnectMaxRetry."
        ::= { pcePcepSessEntry 4 }

    pcePcepSessLocalID OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The value of the PCEP session ID used by the local PCEP
             entity in the Open message for this session.
```

```
            If pcePcepSessState is tcpPending, then this is the session
            ID that will be used in the Open message.  Otherwise, this
            is the session ID that was sent in the Open message."
        ::= { pcePcepSessEntry 5 }

    pcePcepSessRemoteID OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The value of the PCEP session ID used by the peer in its
             Open message for this session.

             If pcePcepSessState is tcpPending or openWait, then this
             field is not used and MUST be set to zero."
        ::= { pcePcepSessEntry 6 }

    pcePcepSessKeepaliveTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The agreed maximum interval at which the local PCEP entity
             transmits PCEP messages on this PCEP session.  Zero means
             that the local PCEP entity never sends Keepalives on this
             session.

             This field is used if and only if pcePcepSessState is
             sessionUp.  Otherwise, it is not used and MUST be set to
             zero."
        ::= { pcePcepSessEntry 7 }

    pcePcepSessPeerKeepaliveTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The agreed maximum interval at which the peer transmits PCEP
             messages on this PCEP session.  Zero means that the peer
             never sends Keepalives on this session.

             This field is used if and only if pcePcepSessState is
             sessionUp.  Otherwise, it is not used and MUST be set to
             zero."
        ::= { pcePcepSessEntry 8 }
```

```
   pcePcepSessDeadTimer OBJECT-TYPE
       SYNTAX      Unsigned32 (0..255)
       UNITS       "seconds"
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The DeadTimer interval for this PCEP session."
       ::= { pcePcepSessEntry 9 }

   pcePcepSessPeerDeadTimer OBJECT-TYPE
       SYNTAX      Unsigned32 (0..255)
       UNITS       "seconds"
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The peer's DeadTimer interval for this PCEP session.

            If pcePcepSessState is tcpPending or openWait, then this
            field is not used and MUST be set to zero."
       ::= { pcePcepSessEntry 10 }

   pcePcepSessKAHoldTimeRem OBJECT-TYPE
       SYNTAX      Unsigned32 (0..255)
       UNITS       "seconds"
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The Keepalive hold time remaining for this session.

            If pcePcepSessState is tcpPending or openWait, then this
            field is not used and MUST be set to zero."
       ::= { pcePcepSessEntry 11 }

   pcePcepSessOverloaded OBJECT-TYPE
       SYNTAX      TruthValue
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "If the local PCEP entity has informed the peer that it is
            currently overloaded, then this is set to true.  Otherwise,
            it is set to false."
       ::= { pcePcepSessEntry 12 }

   pcePcepSessOverloadTime OBJECT-TYPE
       SYNTAX      Unsigned32
       UNITS       "seconds"
       MAX-ACCESS  read-only
       STATUS      current
```

```
        DESCRIPTION
            "The interval of time that is remaining until the local PCEP
             entity will cease to be overloaded on this session.

             This field is only used if pcePcepSessOverloaded is set to
             true.  Otherwise, it is not used and MUST be set to zero."
        ::= { pcePcepSessEntry 13 }

    pcePcepSessPeerOverloaded OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "If the peer has informed the local PCEP entity that it is
             currently overloaded, then this is set to true.  Otherwise,
             it is set to false."
        ::= { pcePcepSessEntry 14 }

    pcePcepSessPeerOverloadTime OBJECT-TYPE
        SYNTAX      Unsigned32
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The interval of time that is remaining until the peer will
             cease to be overloaded.  If it is not known how long the
             peer will stay in overloaded state, this field is set to
             zero.

             This field is only used if pcePcepSessPeerOverloaded is set
             to true.  Otherwise, it is not used and MUST be set to
             zero."
        ::= { pcePcepSessEntry 15 }

    pcePcepSessDiscontinuityTime OBJECT-TYPE
        SYNTAX      TimeStamp
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The value of sysUpTime at the time that the statistics in
             this row were last reset."
        ::= { pcePcepSessEntry 16 }

    pcePcepSessAvgRspTime OBJECT-TYPE
        SYNTAX      Unsigned32
        UNITS       "milliseconds"
        MAX-ACCESS  read-only
        STATUS      current
```

```
        DESCRIPTION
            "The average response time for this peer on this session.

             If an average response time has not been calculated for this
             peer, then this object has the value zero."
        ::= { pcePcepSessEntry 17 }

    pcePcepSessLWMRspTime OBJECT-TYPE
        SYNTAX      Unsigned32
        UNITS       "milliseconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The smallest (low-water mark) response time seen from this
             peer on this session.

             If no responses have been received from this peer, then this
             object has the value zero."
        ::= { pcePcepSessEntry 18 }

    pcePcepSessHWMRspTime OBJECT-TYPE
        SYNTAX      Unsigned32
        UNITS       "milliseconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The greatest (high-water mark) response time seen from this
             peer on this session.

             If no responses have been received from this peer, then this
             object has the value zero."
        ::= { pcePcepSessEntry 19 }

    pcePcepSessNumPCReqSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of PCReq messages sent on this session."
        ::= { pcePcepSessEntry 20 }

    pcePcepSessNumPCReqRcvd OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of PCReq messages received on this session."
        ::= { pcePcepSessEntry 21 }
```

```
   pcePcepSessNumPCRepSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCRep messages sent on this session."
       ::= { pcePcepSessEntry 22 }

   pcePcepSessNumPCRepRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCRep messages received on this session."
       ::= { pcePcepSessEntry 23 }

   pcePcepSessNumPCErrSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCErr messages sent on this session."
       ::= { pcePcepSessEntry 24 }

   pcePcepSessNumPCErrRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCErr messages received on this session."
       ::= { pcePcepSessEntry 25 }

   pcePcepSessNumPCNtfSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCNtf messages sent on this session."
       ::= { pcePcepSessEntry 26 }

   pcePcepSessNumPCNtfRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCNtf messages received on this session."
       ::= { pcePcepSessEntry 27 }
```

```
   pcePcepSessNumKeepaliveSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of Keepalive messages sent on this session."
       ::= { pcePcepSessEntry 28 }

   pcePcepSessNumKeepaliveRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of Keepalive messages received on this session."
       ::= { pcePcepSessEntry 29 }

   pcePcepSessNumUnknownRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of unknown messages received on this session."
       ::= { pcePcepSessEntry 30 }

   pcePcepSessNumCorruptRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of corrupted PCEP messages received on this
            session."
       ::= { pcePcepSessEntry 31 }

   pcePcepSessNumReqSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests sent on this session.  A request
            corresponds 1:1 with an RP object in a PCReq message.

            This might be greater than pcePcepSessNumPCReqSent because
            multiple requests can be batched into a single PCReq
            message."
       ::= { pcePcepSessEntry 32 }
```

```
   pcePcepSessNumSvecSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of SVEC objects sent on this session in PCReq
            messages.  An SVEC object represents a set of synchronized
            requests."
       ::= { pcePcepSessEntry 33 }

   pcePcepSessNumSvecReqSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests sent on this session that appeared in
            one or more SVEC objects."
       ::= { pcePcepSessEntry 34 }

   pcePcepSessNumReqSentPendRep OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests that have been sent on this session
            for which a response is still pending."
       ::= { pcePcepSessEntry 35 }

   pcePcepSessNumReqSentEroRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of successful responses received on this session.
            A response corresponds 1:1 with an RP object in a PCRep
            message.  A successful response is a response for which an
            ERO was successfully computed."
       ::= { pcePcepSessEntry 36 }

   pcePcepSessNumReqSentNoPathRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of unsuccessful responses received on this
            session.  A response corresponds 1:1 with an RP object in a
            PCRep message.  An unsuccessful response is a response with
            a NO-PATH object."
```

```
        ::= { pcePcepSessEntry 37 }

    pcePcepSessNumReqSentCancelRcvd OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests sent on this session that were
             canceled by the peer with a PCNtf message.

             This might be different than pcePcepSessNumPCNtfRcvd because
             not all PCNtf messages are used to cancel requests, and a
             single PCNtf message can cancel multiple requests."
        ::= { pcePcepSessEntry 38 }

    pcePcepSessNumReqSentErrorRcvd OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests sent on this session that were
             rejected by the peer with a PCErr message.

             This might be different than pcePcepSessNumPCErrRcvd because
             not all PCErr messages are used to reject requests, and a
             single PCErr message can reject multiple requests."
        ::= { pcePcepSessEntry 39 }

    pcePcepSessNumReqSentTimeout OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests sent on this session that have been
             sent to a peer and have been abandoned because the peer has
             taken too long to respond to them."
        ::= { pcePcepSessEntry 40 }

    pcePcepSessNumReqSentCancelSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests sent on this session that were sent
             to the peer and explicitly canceled by the local PCEP
             entity sending a PCNtf."
        ::= { pcePcepSessEntry 41 }
```

```
   pcePcepSessNumReqRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests received on this session.  A request
            corresponds 1:1 with an RP object in a PCReq message.

            This might be greater than pcePcepSessNumPCReqRcvd because
            multiple requests can be batched into a single PCReq
            message."
       ::= { pcePcepSessEntry 42 }

   pcePcepSessNumSvecRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of SVEC objects received on this session in PCReq
            messages.  An SVEC object represents a set of synchronized
            requests."
       ::= { pcePcepSessEntry 43 }

   pcePcepSessNumSvecReqRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests received on this session that
            appeared in one or more SVEC objects."
       ::= { pcePcepSessEntry 44 }

   pcePcepSessNumReqRcvdPendRep OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests that have been received on this
            session for which a response is still pending."
       ::= { pcePcepSessEntry 45 }

   pcePcepSessNumReqRcvdEroSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of successful responses sent on this session.  A
            response corresponds 1:1 with an RP object in a PCRep
```

```
            message.  A successful response is a response for which an
            ERO was successfully computed."
        ::= { pcePcepSessEntry 46 }

    pcePcepSessNumReqRcvdNoPathSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of unsuccessful responses sent on this session.
            A response corresponds 1:1 with an RP object in a PCRep
            message.  An unsuccessful response is a response with a
            NO-PATH object."
        ::= { pcePcepSessEntry 47 }

    pcePcepSessNumReqRcvdCancelSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests received on this session that were
            canceled by the local PCEP entity sending a PCNtf message.

            This might be different than pcePcepSessNumPCNtfSent because
            not all PCNtf messages are used to cancel requests, and a
            single PCNtf message can cancel multiple requests."
        ::= { pcePcepSessEntry 48 }

    pcePcepSessNumReqRcvdErrorSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests received on this session that were
            rejected by the local PCEP entity sending a PCErr message.

            This might be different than pcePcepSessNumPCErrSent because
            not all PCErr messages are used to reject requests, and a
            single PCErr message can reject multiple requests."
        ::= { pcePcepSessEntry 49 }

    pcePcepSessNumReqRcvdCancelRcvd OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests that were received on this session
            and explicitly canceled by the peer sending a PCNtf."
```

```
        ::= { pcePcepSessEntry 50 }

    pcePcepSessNumRepRcvdUnknown OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of responses to unknown requests received on this
             session.  A response to an unknown request is a response
             whose RP object does not contain the request ID of any
             request that is currently outstanding on the session."
        ::= { pcePcepSessEntry 51 }

    pcePcepSessNumReqRcvdUnknown OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of unknown requests that have been received on
             this session.  An unknown request is a request whose RP
             object contains a request ID of zero."
        ::= { pcePcepSessEntry 52 }

    ---
    --- Notifications Configuration
    ---

    pcePcepNotificationsMaxRate OBJECT-TYPE
        SYNTAX       Unsigned32
        MAX-ACCESS   read-write
        STATUS       current
        DESCRIPTION
            "This variable indicates the maximum number of
             notifications issued per second.  If events occur
             more rapidly, the implementation may simply fail to
             emit these notifications during that period or may
             queue them until an appropriate time.  A value of zero
             means no notifications are emitted and all should be
             discarded (that is, not queued)."
        ::= { pcePcepObjects 4 }

    ---
    --- Notifications
    ---

    pcePcepSessUp NOTIFICATION-TYPE
        OBJECTS     {
                       pcePcepSessState,
```

```
                    pcePcepSessStateLastChange
                }
    STATUS      current
    DESCRIPTION
        "This notification is sent when the value of
         pcePcepSessState enters the sessionUp state."
    ::= { pcePcepNotifications 1 }

pcePcepSessDown NOTIFICATION-TYPE
    OBJECTS     {
                    pcePcepSessState,
                    pcePcepSessStateLastChange
                }
    STATUS      current
    DESCRIPTION
        "This notification is sent when the value of
         pcePcepSessState leaves the sessionUp state."
    ::= { pcePcepNotifications 2 }

pcePcepSessLocalOverload NOTIFICATION-TYPE
    OBJECTS     {
                    pcePcepSessOverloaded,
                    pcePcepSessOverloadTime
                }
    STATUS      current
    DESCRIPTION
        "This notification is sent when the local PCEP entity enters
         overload state for a peer."
    ::= { pcePcepNotifications 3 }

pcePcepSessLocalOverloadClear NOTIFICATION-TYPE
    OBJECTS     {
                    pcePcepSessOverloaded
                }
    STATUS      current
    DESCRIPTION
        "This notification is sent when the local PCEP entity leaves
         overload state for a peer."
    ::= { pcePcepNotifications 4 }

pcePcepSessPeerOverload NOTIFICATION-TYPE
    OBJECTS     {
                    pcePcepSessPeerOverloaded,
                    pcePcepSessPeerOverloadTime
                }
    STATUS      current
```

```
        DESCRIPTION
            "This notification is sent when a peer enters overload
             state."
        ::= { pcePcepNotifications 5 }

    pcePcepSessPeerOverloadClear NOTIFICATION-TYPE
        OBJECTS     {
                       pcePcepSessPeerOverloaded
                    }
        STATUS      current
        DESCRIPTION
            "This notification is sent when a peer leaves overload
             state."
        ::= { pcePcepNotifications 6 }

    --
    -- Module Conformance Statement
    --

    pcePcepCompliances
        OBJECT IDENTIFIER ::= { pcePcepConformance 1 }

    pcePcepGroups
        OBJECT IDENTIFIER ::= { pcePcepConformance 2 }

    --
    -- Read-Only Compliance
    --

    pcePcepModuleReadOnlyCompliance MODULE-COMPLIANCE
        STATUS current
        DESCRIPTION
            "The module is implemented with support for read-only.  In
             other words, only monitoring is available by implementing
             this MODULE-COMPLIANCE."

        MODULE -- this module
            MANDATORY-GROUPS    {
                                  pcePcepGeneralGroup,
                                  pcePcepNotificationsGroup
                                }

        OBJECT       pcePcepEntityAddrType
        SYNTAX       InetAddressType { unknown(0), ipv4(1), ipv6(2) }
        DESCRIPTION  "Only unknown(0), ipv4(1), and ipv6(2) support
                      is required."
```

```
-- The following restriction is commented out because of a limitation
-- in SMIv2 which does not allow index objects to be restricted in
-- scope.  Nevertheless, this object is intended to be restricted in
-- scope, as follows.
--
--     OBJECT       pcePcepPeerAddrType
--     SYNTAX       InetAddressType { unknown(0), ipv4(1), ipv6(2) }
--     DESCRIPTION  "Only unknown(0), ipv4(1), and ipv6(2) support
--                   is required."

       ::= { pcePcepCompliances 1 }

   -- units of conformance

   pcePcepGeneralGroup OBJECT-GROUP
       OBJECTS { pcePcepEntityAdminStatus,
                 pcePcepEntityOperStatus,
                 pcePcepEntityAddrType,
                 pcePcepEntityAddr,
                 pcePcepEntityConnectTimer,
                 pcePcepEntityConnectMaxRetry,
                 pcePcepEntityInitBackoffTimer,
                 pcePcepEntityMaxBackoffTimer,
                 pcePcepEntityOpenWaitTimer,
                 pcePcepEntityKeepWaitTimer,
                 pcePcepEntityKeepAliveTimer,
                 pcePcepEntityDeadTimer,
                 pcePcepEntityAllowNegotiation,
                 pcePcepEntityMaxKeepAliveTimer,
                 pcePcepEntityMaxDeadTimer,
                 pcePcepEntityMinKeepAliveTimer,
                 pcePcepEntityMinDeadTimer,
                 pcePcepEntitySyncTimer,
                 pcePcepEntityRequestTimer,
                 pcePcepEntityMaxSessions,
                 pcePcepEntityMaxUnknownReqs,
                 pcePcepEntityMaxUnknownMsgs,
                 pcePcepPeerRole,
                 pcePcepPeerDiscontinuityTime,
                 pcePcepPeerInitiateSession,
                 pcePcepPeerSessionExists,
                 pcePcepPeerNumSessSetupOK,
                 pcePcepPeerNumSessSetupFail,
                 pcePcepPeerSessionUpTime,
                 pcePcepPeerSessionFailTime,
                 pcePcepPeerSessionFailUpTime,
                 pcePcepPeerAvgRspTime,
                 pcePcepPeerLWMRspTime,
```

```
                pcePcepPeerHWMRspTime,
                pcePcepPeerNumPCReqSent,
                pcePcepPeerNumPCReqRcvd,
                pcePcepPeerNumPCRepSent,
                pcePcepPeerNumPCRepRcvd,
                pcePcepPeerNumPCErrSent,
                pcePcepPeerNumPCErrRcvd,
                pcePcepPeerNumPCNtfSent,
                pcePcepPeerNumPCNtfRcvd,
                pcePcepPeerNumKeepaliveSent,
                pcePcepPeerNumKeepaliveRcvd,
                pcePcepPeerNumUnknownRcvd,
                pcePcepPeerNumCorruptRcvd,
                pcePcepPeerNumReqSent,
                pcePcepPeerNumSvecSent,
                pcePcepPeerNumSvecReqSent,
                pcePcepPeerNumReqSentPendRep,
                pcePcepPeerNumReqSentEroRcvd,
                pcePcepPeerNumReqSentNoPathRcvd,
                pcePcepPeerNumReqSentCancelRcvd,
                pcePcepPeerNumReqSentErrorRcvd,
                pcePcepPeerNumReqSentTimeout,
                pcePcepPeerNumReqSentCancelSent,
                pcePcepPeerNumReqSentClosed,
                pcePcepPeerNumReqRcvd,
                pcePcepPeerNumSvecRcvd,
                pcePcepPeerNumSvecReqRcvd,
                pcePcepPeerNumReqRcvdPendRep,
                pcePcepPeerNumReqRcvdEroSent,
                pcePcepPeerNumReqRcvdNoPathSent,
                pcePcepPeerNumReqRcvdCancelSent,
                pcePcepPeerNumReqRcvdErrorSent,
                pcePcepPeerNumReqRcvdCancelRcvd,
                pcePcepPeerNumReqRcvdClosed,
                pcePcepPeerNumRepRcvdUnknown,
                pcePcepPeerNumReqRcvdUnknown,
                pcePcepSessStateLastChange,
                pcePcepSessState,
                pcePcepSessConnectRetry,
                pcePcepSessLocalID,
                pcePcepSessRemoteID,
                pcePcepSessKeepaliveTimer,
                pcePcepSessPeerKeepaliveTimer,
                pcePcepSessDeadTimer,
                pcePcepSessPeerDeadTimer,
                pcePcepSessKAHoldTimeRem,
                pcePcepSessOverloaded,
                pcePcepSessOverloadTime,
```

```
                pcePcepSessPeerOverloaded,
                pcePcepSessPeerOverloadTime,
                pcePcepSessDiscontinuityTime,
                pcePcepSessAvgRspTime,
                pcePcepSessLWMRspTime,
                pcePcepSessHWMRspTime,
                pcePcepSessNumPCReqSent,
                pcePcepSessNumPCReqRcvd,
                pcePcepSessNumPCRepSent,
                pcePcepSessNumPCRepRcvd,
                pcePcepSessNumPCErrSent,
                pcePcepSessNumPCErrRcvd,
                pcePcepSessNumPCNtfSent,
                pcePcepSessNumPCNtfRcvd,
                pcePcepSessNumKeepaliveSent,
                pcePcepSessNumKeepaliveRcvd,
                pcePcepSessNumUnknownRcvd,
                pcePcepSessNumCorruptRcvd,
                pcePcepSessNumReqSent,
                pcePcepSessNumSvecSent,
                pcePcepSessNumSvecReqSent,
                pcePcepSessNumReqSentPendRep,
                pcePcepSessNumReqSentEroRcvd,
                pcePcepSessNumReqSentNoPathRcvd,
                pcePcepSessNumReqSentCancelRcvd,
                pcePcepSessNumReqSentErrorRcvd,
                pcePcepSessNumReqSentTimeout,
                pcePcepSessNumReqSentCancelSent,
                pcePcepSessNumReqRcvd,
                pcePcepSessNumSvecRcvd,
                pcePcepSessNumSvecReqRcvd,
                pcePcepSessNumReqRcvdPendRep,
                pcePcepSessNumReqRcvdEroSent,
                pcePcepSessNumReqRcvdNoPathSent,
                pcePcepSessNumReqRcvdCancelSent,
                pcePcepSessNumReqRcvdErrorSent,
                pcePcepSessNumReqRcvdCancelRcvd,
                pcePcepSessNumRepRcvdUnknown,
                pcePcepSessNumReqRcvdUnknown,
                pcePcepNotificationsMaxRate
            }
    STATUS current
    DESCRIPTION
        "Objects that apply to all PCEP MIB module implementations."
    ::= { pcePcepGroups 1 }
```

```
   pcePcepNotificationsGroup NOTIFICATION-GROUP
       NOTIFICATIONS { pcePcepSessUp,
                       pcePcepSessDown,
                       pcePcepSessLocalOverload,
                       pcePcepSessLocalOverloadClear,
                       pcePcepSessPeerOverload,
                       pcePcepSessPeerOverloadClear
                     }
       STATUS   current
       DESCRIPTION
           "The notifications for a PCEP MIB module implementation."
       ::= { pcePcepGroups 2 }

   END
```

## 5. Security Considerations

The pcePcepNotificationsMaxRate object defined in this MIB module has a MAX-ACCESS clause of read-write. Such objects may be considered sensitive or vulnerable in some network environments. The support for SET operations in a non-secure environment without proper protection opens devices to attack. In particular, pcePcepNotificationsMaxRate may be used improperly to stop notifications being issued or to permit a flood of notifications to be sent to the management agent at a high rate.

All the readable objects in this MIB module (i.e., objects with a MAX-ACCESS other than not-accessible) may be considered sensitive or vulnerable in some network environments. It is thus important to control even GET and/or NOTIFY access to these objects and possibly to even encrypt the values of these objects when sending them over the network via SNMP. The sensitivity/vulnerability arises because, collectively, these objects provide information about the amount and frequency of path computation requests and responses within the network and can reveal some aspects of its configuration.

SNMP versions prior to SNMPv3 did not include adequate security. Even if the network itself is secure (for example by using IPsec), there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in this MIB module.

Implementations SHOULD provide the security features described by the SNMPv3 framework (see [RFC3410]), and implementations claiming compliance to the SNMPv3 standard MUST include full support for authentication and privacy via the User-based Security Model (USM) [RFC3414] with the AES cipher algorithm [RFC3826]. Implementations MAY also provide support for the Transport Security Model (TSM)

[RFC5591] in combination with a secure transport such as SSH [RFC5592] or TLS/DTLS [RFC6353].

Further, deployment of SNMP versions prior to SNMPv3 is NOT RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to enable cryptographic security. It is then a customer/operator responsibility to ensure that the SNMP entity giving access to an instance of this MIB module is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

## 6. IANA Considerations

The MIB module in this document uses the following IANA-assigned OBJECT IDENTIFIER values recorded in the SMI Numbers registry:

| Descriptor | OBJECT IDENTIFIER value |
|---|---|
| pcePcepMIB | { mib-2 227 } |

## 7. References

### 7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC2578] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999, <http://www.rfc-editor.org/info/rfc2578>.

[RFC2579] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999, <http://www.rfc-editor.org/info/rfc2579>.

[RFC2580] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999, <http://www.rfc-editor.org/info/rfc2580>.

[RFC3414] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", STD 62, RFC 3414, December 2002, <http://www.rfc-editor.org/info/rfc3414>.

[RFC3826] Blumenthal, U., Maino, F., and K. McCloghrie, "The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model", RFC 3826, June 2004, <http://www.rfc-editor.org/info/rfc3826>.

[RFC4001] Daniele, M., Haberman, B., Routhier, S., and J. Schoenwaelder, "Textual Conventions for Internet Network Addresses", RFC 4001, February 2005, <http://www.rfc-editor.org/info/rfc4001>.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009, <http://www.rfc-editor.org/info/rfc5440>.

[RFC5591] Harrington, D. and W. Hardaker, "Transport Security Model for the Simple Network Management Protocol (SNMP)", STD 78, RFC 5591, June 2009, <http://www.rfc-editor.org/info/rfc5591>.

[RFC5592] Harrington, D., Salowey, J., and W. Hardaker, "Secure Shell Transport Model for the Simple Network Management Protocol (SNMP)", RFC 5592, June 2009, <http://www.rfc-editor.org/info/rfc5592>.

[RFC6353] Hardaker, W., "Transport Layer Security (TLS) Transport Model for the Simple Network Management Protocol (SNMP)", STD 78, RFC 6353, July 2011, <http://www.rfc-editor.org/info/rfc6353>.

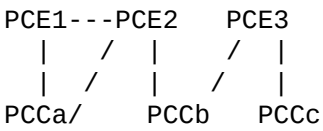
## 7.2. Informative References

[RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002, <http://www.rfc-editor.org/info/rfc3410>.

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006, <http://www.rfc-editor.org/info/rfc4655>.

## Appendix A. PCEP MIB Module Example

This example considers the set of PCC/PCE relationships shown in the following figure. The example shows the contents of the PCEP MIB module as read at PCE2 and PCCb.




The IP addresses of the PCE speakers in this diagram are given in the following table.

| | |
|---|---|
| PCE1 | 1.1.1.1 |
| PCE2 | 2.2.2.2 |
| PCE3 | 3.3.3.3 |
| PCCa | 11.11.11.11 |
| PCCb | 22.22.22.22 |
| PCCc | 33.33.33.33 |

In this example, the PCEP session between PCCb and PCE3 is currently down.

## A.1. Contents of PCEP MIB Module at PCE2

At PCE2, there is a single local PCEP entity that has three peers (PCCa, PCCb, and PCE1). There is a session active to all of these peers.

The contents of the PCEP MIB module as read at PCE2 are as follows.

```
In pcePcepEntityTable  {
       pcePcepEntityIndex                1,
       pcePcepEntityAdminStatus          adminStatusUp(1),
       pcePcepEntityOperStatus           operStatusUp(1),
       pcePcepEntityAddrType             ipv4(1),
       pcePcepEntityAddr                 2.2.2.2, -- PCE2
       pcePcepEntityConnectTimer         60,
       pcePcepEntityConnectMaxRetry      5,
       pcePcepEntityInitBackoffTimer     30,
       pcePcepEntityMaxBackoffTimer      3600,
       pcePcepEntityOpenWaitTimer        60,
       pcePcepEntityKeepWaitTimer        60,
       pcePcepEntityKeepAliveTimer       1,
       pcePcepEntityDeadTimer            4,
       pcePcepEntityAllowNegotiation     true(1),
       pcePcepEntityMaxKeepAliveTimer    60,
       pcePcepEntityMaxDeadTimer         240,
       pcePcepEntityMinKeepAliveTimer    1,
       pcePcepEntityMinDeadTimer         4,
       pcePcepEntitySyncTimer            60,
       pcePcepEntityRequestTimer         120,
       pcePcepEntityMaxSessions          999,
       pcePcepEntityMaxUnknownReqs       5,
       pcePcepEntityMaxUnknownMsgs       5
   }

In pcePcepPeerTable {
       pcePcepPeerAddrType                ipv4(1), --PCE1
       pcePcepPeerAddr                    1.1.1.1,
       pcePcepPeerRole                    pccAndPce(3),
       pcePcepPeerDiscontinuityTime       TimeStamp,
       pcePcepPeerInitiateSession         true(1),
       pcePcepPeerSessionExists           true(1),
       pcePcepPeerNumSessSetupOK          1,
       pcePcepPeerNumSessSetupFail        0,
       pcePcepPeerSessionUpTime           TimeStamp,
       pcePcepPeerSessionFailTime         0,
       pcePcepPeerSessionFailUpTime       TimeStamp,
       pcePcepPeerAvgRspTime              0,
       pcePcepPeerLWMRspTime              0,
```

```
        pcePcepPeerHWMRspTime               0,
        pcePcepPeerNumPCReqSent             0,
        pcePcepPeerNumPCReqRcvd             0,
        pcePcepPeerNumPCRepSent             0,
        pcePcepPeerNumPCRepRcvd             0,
        pcePcepPeerNumPCErrSent             0,
        pcePcepPeerNumPCErrRcvd             0,
        pcePcepPeerNumPCNtfSent             0,
        pcePcepPeerNumPCNtfRcvd             0,
        pcePcepPeerNumKeepaliveSent         123,
        pcePcepPeerNumKeepaliveRcvd         123,
        pcePcepPeerNumUnknownRcvd           0,
        pcePcepPeerNumCorruptRcvd           0,
        pcePcepPeerNumReqSent               0,
        pcePcepPeerNumSvecSent              0,
        pcePcepPeerNumSvecReqSent           0,
        pcePcepPeerNumReqSentPendRep        0,
        pcePcepPeerNumReqSentEroRcvd        0,
        pcePcepPeerNumReqSentNoPathRcvd     0,
        pcePcepPeerNumReqSentCancelRcvd     0,
        pcePcepPeerNumReqSentErrorRcvd      0,
        pcePcepPeerNumReqSentTimeout        0,
        pcePcepPeerNumReqSentCancelSent     0,
        pcePcepPeerNumReqSentClosed         0,
        pcePcepPeerNumReqRcvd               0,
        pcePcepPeerNumSvecRcvd              0,
        pcePcepPeerNumSvecReqRcvd           0,
        pcePcepPeerNumReqRcvdPendRep        0,
        pcePcepPeerNumReqRcvdEroSent        0,
        pcePcepPeerNumReqRcvdNoPathSent     0,
        pcePcepPeerNumReqRcvdCancelSent     0,
        pcePcepPeerNumReqRcvdErrorSent      0,
        pcePcepPeerNumReqRcvdCancelRcvd     0,
        pcePcepPeerNumReqRcvdClosed         0,
        pcePcepPeerNumRepRcvdUnknown        0,
        pcePcepPeerNumReqRcvdUnknown        0
    },
    {
        pcePcepPeerAddrType                 ipv4(1),  --PCCa
        pcePcepPeerAddr                     11.11.11.11,
        pcePcepPeerRole                     pcc(1),
        pcePcepPeerDiscontinuityTime        TimeStamp,
        pcePcepPeerInitiateSession          false(0),
        pcePcepPeerSessionExists            true(1),
        pcePcepPeerNumSessSetupOK           1,
        pcePcepPeerNumSessSetupFail         0,
        pcePcepPeerSessionUpTime            TimeStamp,
        pcePcepPeerSessionFailTime          0,
```

```
        pcePcepPeerSessionFailUpTime        TimeStamp,
        pcePcepPeerAvgRspTime               200,
        pcePcepPeerLWMRspTime               100,
        pcePcepPeerHWMRspTime               300,
        pcePcepPeerNumPCReqSent             0,
        pcePcepPeerNumPCReqRcvd             3,
        pcePcepPeerNumPCRepSent             3,
        pcePcepPeerNumPCRepRcvd             0,
        pcePcepPeerNumPCErrSent             0,
        pcePcepPeerNumPCErrRcvd             0,
        pcePcepPeerNumPCNtfSent             0,
        pcePcepPeerNumPCNtfRcvd             0,
        pcePcepPeerNumKeepaliveSent         123,
        pcePcepPeerNumKeepaliveRcvd         123,
        pcePcepPeerNumUnknownRcvd           0,
        pcePcepPeerNumCorruptRcvd           0,
        pcePcepPeerNumReqSent               0,
        pcePcepPeerNumSvecSent              0,
        pcePcepPeerNumSvecReqSent           0,
        pcePcepPeerNumReqSentPendRep        0,
        pcePcepPeerNumReqSentEroRcvd        0,
        pcePcepPeerNumReqSentNoPathRcvd     0,
        pcePcepPeerNumReqSentCancelRcvd     0,
        pcePcepPeerNumReqSentErrorRcvd      0,
        pcePcepPeerNumReqSentTimeout        0,
        pcePcepPeerNumReqSentCancelSent     0,
        pcePcepPeerNumReqSentClosed         0,
        pcePcepPeerNumReqRcvd               3,
        pcePcepPeerNumSvecRcvd              0,
        pcePcepPeerNumSvecReqRcvd           0,
        pcePcepPeerNumReqRcvdPendRep        0,
        pcePcepPeerNumReqRcvdEroSent        3,
        pcePcepPeerNumReqRcvdNoPathSent     0,
        pcePcepPeerNumReqRcvdCancelSent     0,
        pcePcepPeerNumReqRcvdErrorSent      0,
        pcePcepPeerNumReqRcvdCancelRcvd     0,
        pcePcepPeerNumReqRcvdClosed         0,
        pcePcepPeerNumRepRcvdUnknown        0,
        pcePcepPeerNumReqRcvdUnknown        0
    },
    {
        pcePcepPeerAddrType                 ipv4(1), -- PCCb
        pcePcepPeerAddr                     22.22.22.22,
        pcePcepPeerRole                     pcc(1),
        pcePcepPeerDiscontinuityTime        TimeStamp,
        pcePcepPeerInitiateSession          true(1),
        pcePcepPeerSessionExists            true(1),
        pcePcepPeerNumSessSetupOK           1,
```

```
          pcePcepPeerNumSessSetupFail         0,
          pcePcepPeerSessionUpTime            TimeStamp,
          pcePcepPeerSessionFailTime          0,
          pcePcepPeerSessionFailUpTime        TimeStamp,
          pcePcepPeerAvgRspTime               200,
          pcePcepPeerLWMRspTime               100,
          pcePcepPeerHWMRspTime               300,
          pcePcepPeerNumPCReqSent             0,
          pcePcepPeerNumPCReqRcvd             4,
          pcePcepPeerNumPCRepSent             4,
          pcePcepPeerNumPCRepRcvd             0,
          pcePcepPeerNumPCErrSent             0,
          pcePcepPeerNumPCErrRcvd             0,
          pcePcepPeerNumPCNtfSent             0,
          pcePcepPeerNumPCNtfRcvd             0,
          pcePcepPeerNumKeepaliveSent         123,
          pcePcepPeerNumKeepaliveRcvd         123,
          pcePcepPeerNumUnknownRcvd           0,
          pcePcepPeerNumCorruptRcvd           0,
          pcePcepPeerNumReqSent               0,
          pcePcepPeerNumSvecSent              0,
          pcePcepPeerNumSvecReqSent           0,
          pcePcepPeerNumReqSentPendRep        0,
          pcePcepPeerNumReqSentEroRcvd        0,
          pcePcepPeerNumReqSentNoPathRcvd     0,
          pcePcepPeerNumReqSentCancelRcvd     0,
          pcePcepPeerNumReqSentErrorRcvd      0,
          pcePcepPeerNumReqSentTimeout        0,
          pcePcepPeerNumReqSentCancelSent     0,
          pcePcepPeerNumReqSentClosed         0,
          pcePcepPeerNumReqRcvd               4,
          pcePcepPeerNumSvecRcvd              0,
          pcePcepPeerNumSvecReqRcvd           0,
          pcePcepPeerNumReqRcvdPendRep        0,
          pcePcepPeerNumReqRcvdEroSent        3,
          pcePcepPeerNumReqRcvdNoPathSent     1,
          pcePcepPeerNumReqRcvdCancelSent     0,
          pcePcepPeerNumReqRcvdErrorSent      0,
          pcePcepPeerNumReqRcvdCancelRcvd     0,
          pcePcepPeerNumReqRcvdClosed         0,
          pcePcepPeerNumRepRcvdUnknown        0,
          pcePcepPeerNumReqRcvdUnknown        0
      }

   In pcePcepSessTable {
          pcePcepSessInitiator                local(1), --PCE1
          pcePcepSessStateLastChange          TimeStamp,
          pcePcepSessState                    sessionUp(4),
```

```
pcePcepSessConnectRetry             0,
pcePcepSessLocalID                  1,
pcePcepSessRemoteID                 2,
pcePcepSessKeepaliveTimer           1,
pcePcepSessPeerKeepaliveTimer       1,
pcePcepSessDeadTimer                4,
pcePcepSessPeerDeadTimer            4,
pcePcepSessKAHoldTimeRem            1,
pcePcepSessOverloaded               false(0),
pcePcepSessOverloadTime             0,
pcePcepSessPeerOverloaded           false(0),
pcePcepSessPeerOverloadTime         0,
pcePcepSessDiscontinuityTime        TimeStamp,
pcePcepSessAvgRspTime               0,
pcePcepSessLWMRspTime               0,
pcePcepSessHWMRspTime               0,
pcePcepSessNumPCReqSent             0,
pcePcepSessNumPCReqRcvd             0,
pcePcepSessNumPCRepSent             0,
pcePcepSessNumPCRepRcvd             0,
pcePcepSessNumPCErrSent             0,
pcePcepSessNumPCErrRcvd             0,
pcePcepSessNumPCNtfSent             0,
pcePcepSessNumPCNtfRcvd             0,
pcePcepSessNumKeepaliveSent         123,
pcePcepSessNumKeepaliveRcvd         123,
pcePcepSessNumUnknownRcvd           0,
pcePcepSessNumCorruptRcvd           0,
pcePcepSessNumReqSent               0,
pcePcepSessNumSvecSent              0,
pcePcepSessNumSvecReqSent           0,
pcePcepSessNumReqSentPendRep        0,
pcePcepSessNumReqSentEroRcvd        0,
pcePcepSessNumReqSentNoPathRcvd     0,
pcePcepSessNumReqSentCancelRcvd     0,
pcePcepSessNumReqSentErrorRcvd      0,
pcePcepSessNumReqSentTimeout        0,
pcePcepSessNumReqSentCancelSent     0,
pcePcepSessNumReqRcvd               0,
pcePcepSessNumSvecRcvd              0,
pcePcepSessNumSvecReqRcvd           0,
pcePcepSessNumReqRcvdPendRep        0,
pcePcepSessNumReqRcvdEroSent        0,
pcePcepSessNumReqRcvdNoPathSent     0,
pcePcepSessNumReqRcvdCancelSent     0,
pcePcepSessNumReqRcvdErrorSent      0,
pcePcepSessNumReqRcvdCancelRcvd     0,
pcePcepSessNumRepRcvdUnknown        0,
```

```
        pcePcepSessNumReqRcvdUnknown        0
    },
    {
        pcePcepSessInitiator                remote(2), --PCCa
        pcePcepSessStateLastChange          TimeStamp,
        pcePcepSessState                    sessionUp(4),
        pcePcepSessConnectRetry             0,
        pcePcepSessLocalID                  2,
        pcePcepSessRemoteID                 1,
        pcePcepSessKeepaliveTimer           1,
        pcePcepSessPeerKeepaliveTimer       1,
        pcePcepSessDeadTimer                4,
        pcePcepSessPeerDeadTimer            4,
        pcePcepSessKAHoldTimeRem            1,
        pcePcepSessOverloaded               false(0),
        pcePcepSessOverloadTime             0,
        pcePcepSessPeerOverloaded           false(0),
        pcePcepSessPeerOverloadTime         0,
        pcePcepSessDiscontinuityTime        TimeStamp,
        pcePcepSessAvgRspTime               200,
        pcePcepSessLWMRspTime               100,
        pcePcepSessHWMRspTime               300,
        pcePcepSessNumPCReqSent             0,
        pcePcepSessNumPCReqRcvd             1,
        pcePcepSessNumPCRepSent             1,
        pcePcepSessNumPCRepRcvd             0,
        pcePcepSessNumPCErrSent             0,
        pcePcepSessNumPCErrRcvd             0,
        pcePcepSessNumPCNtfSent             0,
        pcePcepSessNumPCNtfRcvd             0,
        pcePcepSessNumKeepaliveSent         123,
        pcePcepSessNumKeepaliveRcvd         123,
        pcePcepSessNumUnknownRcvd           0,
        pcePcepSessNumCorruptRcvd           0,
        pcePcepSessNumReqSent               0,
        pcePcepSessNumSvecSent              0,
        pcePcepSessNumSvecReqSent           0,
        pcePcepSessNumReqSentPendRep        0,
        pcePcepSessNumReqSentEroRcvd        0,
        pcePcepSessNumReqSentNoPathRcvd     0,
        pcePcepSessNumReqSentCancelRcvd     0,
        pcePcepSessNumReqSentErrorRcvd      0,
        pcePcepSessNumReqSentTimeout        0,
        pcePcepSessNumReqSentCancelSent     0,
        pcePcepSessNumReqRcvd               3,
        pcePcepSessNumSvecRcvd              0,
        pcePcepSessNumSvecReqRcvd           0,
        pcePcepSessNumReqRcvdPendRep        0,
```

```
        pcePcepSessNumReqRcvdEroSent        3,
        pcePcepSessNumReqRcvdNoPathSent     0,
        pcePcepSessNumReqRcvdCancelSent     0,
        pcePcepSessNumReqRcvdErrorSent      0,
        pcePcepSessNumReqRcvdCancelRcvd     0,
        pcePcepSessNumRepRcvdUnknown        0,
        pcePcepSessNumReqRcvdUnknown        0
    },
    {
        pcePcepSessInitiator                remote(2), --PCCb
        pcePcepSessStateLastChange          TimeStamp,
        pcePcepSessState                    sessionUp(4),
        pcePcepSessConnectRetry             0,
        pcePcepSessLocalID                  2,
        pcePcepSessRemoteID                 1,
        pcePcepSessKeepaliveTimer           1,
        pcePcepSessPeerKeepaliveTimer       1,
        pcePcepSessDeadTimer                4,
        pcePcepSessPeerDeadTimer            4,
        pcePcepSessKAHoldTimeRem            1,
        pcePcepSessOverloaded               false(0),
        pcePcepSessOverloadTime             0,
        pcePcepSessPeerOverloaded           false(0),
        pcePcepSessPeerOverloadTime         0,
        pcePcepSessDiscontinuityTime        TimeStamp,
        pcePcepSessAvgRspTime               200,
        pcePcepSessLWMRspTime               100,
        pcePcepSessHWMRspTime               300,
        pcePcepSessNumPCReqSent             0,
        pcePcepSessNumPCReqRcvd             4,
        pcePcepSessNumPCRepSent             4,
        pcePcepSessNumPCRepRcvd             0,
        pcePcepSessNumPCErrSent             0,
        pcePcepSessNumPCErrRcvd             0,
        pcePcepSessNumPCNtfSent             0,
        pcePcepSessNumPCNtfRcvd             0,
        pcePcepSessNumKeepaliveSent         123,
        pcePcepSessNumKeepaliveRcvd         123,
        pcePcepSessNumUnknownRcvd           0,
        pcePcepSessNumCorruptRcvd           0,
        pcePcepSessNumReqSent               0,
        pcePcepSessNumSvecSent              0,
        pcePcepSessNumSvecReqSent           0,
        pcePcepSessNumReqSentPendRep        0,
        pcePcepSessNumReqSentEroRcvd        0,
        pcePcepSessNumReqSentNoPathRcvd     0,
        pcePcepSessNumReqSentCancelRcvd     0,
        pcePcepSessNumReqSentErrorRcvd      0,
```

```
        pcePcepSessNumReqSentTimeout        0,
        pcePcepSessNumReqSentCancelSent     0,
        pcePcepSessNumReqRcvd               4,
        pcePcepSessNumSvecRcvd              0,
        pcePcepSessNumSvecReqRcvd           0,
        pcePcepSessNumReqRcvdPendRep        0,
        pcePcepSessNumReqRcvdEroSent        3,
        pcePcepSessNumReqRcvdNoPathSent     1,
        pcePcepSessNumReqRcvdCancelSent     0,
        pcePcepSessNumReqRcvdErrorSent      0,
        pcePcepSessNumReqRcvdCancelRcvd     0,
        pcePcepSessNumRepRcvdUnknown        0,
        pcePcepSessNumReqRcvdUnknown        0
    }
```

## A.2. Contents of PCEP MIB Module at PCCb

At PCCb, there is a single local PCEP entity that has two peers (PCE2 and PCE3). There is a session active to PCE2, but the session to PCE3 is currently down.

The contents of the PCEP MIB module as read at PCCb are as follows.

```
In pcePcepEntityTable  {
        pcePcepEntityIndex                1,
        pcePcepEntityAdminStatus          adminStatusUp(1),
        pcePcepEntityOperStatus           operStatusUp(1),
        pcePcepEntityAddrType             ipv4(1),
        pcePcepEntityAddr                 22.22.22.22, -- PCCb
        pcePcepEntityConnectTimer         60,
        pcePcepEntityConnectMaxRetry      5,
        pcePcepEntityInitBackoffTimer     30,
        pcePcepEntityMaxBackoffTimer      3600,
        pcePcepEntityOpenWaitTimer        60,
        pcePcepEntityKeepWaitTimer        60,
        pcePcepEntityKeepAliveTimer       1,
        pcePcepEntityDeadTimer            4,
        pcePcepEntityAllowNegotiation     true(1),
        pcePcepEntityMaxKeepAliveTimer    60,
        pcePcepEntityMaxDeadTimer         240,
        pcePcepEntityMinKeepAliveTimer    1,
        pcePcepEntityMinDeadTimer         4,
        pcePcepEntitySyncTimer            60,
        pcePcepEntityRequestTimer         120,
        pcePcepEntityMaxSessions          999,
        pcePcepEntityMaxUnknownReqs       5,
        pcePcepEntityMaxUnknownMsgs       5
    }
```

```
   In pcePcepPeerTable {
          pcePcepPeerAddrType                ipv4(1), --PCE2
          pcePcepPeerAddr                    2.2.2.2,
          pcePcepPeerRole                    pce(2),
          pcePcepPeerDiscontinuityTime       TimeStamp,
          pcePcepPeerInitiateSession         true(1),
          pcePcepPeerSessionExists           true(1)),
          pcePcepPeerNumSessSetupOK          0,
          pcePcepPeerNumSessSetupFail        1,
          pcePcepPeerSessionUpTime           TimeStamp,
          pcePcepPeerSessionFailTime         TimeStamp,
          pcePcepPeerSessionFailUpTime       TimeStamp,
          pcePcepPeerAvgRspTime              0,
          pcePcepPeerLWMRspTime              0,
          pcePcepPeerHWMRspTime              0,
          pcePcepPeerNumPCReqSent            4,
          pcePcepPeerNumPCReqRcvd            0,
          pcePcepPeerNumPCRepSent            0,
          pcePcepPeerNumPCRepRcvd            4,
          pcePcepPeerNumPCErrSent            0,
          pcePcepPeerNumPCErrRcvd            0,
          pcePcepPeerNumPCNtfSent            0,
          pcePcepPeerNumPCNtfRcvd            0,
          pcePcepPeerNumKeepaliveSent        0,
          pcePcepPeerNumKeepaliveRcvd        0,
          pcePcepPeerNumUnknownRcvd          0,
          pcePcepPeerNumCorruptRcvd          0,
          pcePcepPeerNumReqSent              4,
          pcePcepPeerNumSvecSent             0,
          pcePcepPeerNumSvecReqSent          0,
          pcePcepPeerNumReqSentPendRep       0,
          pcePcepPeerNumReqSentEroRcvd       3,
          pcePcepPeerNumReqSentNoPathRcvd    1,
          pcePcepPeerNumReqSentCancelRcvd    0,
          pcePcepPeerNumReqSentErrorRcvd     0,
          pcePcepPeerNumReqSentTimeout       0,
          pcePcepPeerNumReqSentCancelSent    0,
          pcePcepPeerNumReqSentClosed        0,
          pcePcepPeerNumReqRcvd              0,
          pcePcepPeerNumSvecRcvd             0,
          pcePcepPeerNumSvecReqRcvd          0,
          pcePcepPeerNumReqRcvdPendRep       0,
          pcePcepPeerNumReqRcvdEroSent       0,
          pcePcepPeerNumReqRcvdNoPathSent    0,
          pcePcepPeerNumReqRcvdCancelSent    0,
          pcePcepPeerNumReqRcvdErrorSent     0,
          pcePcepPeerNumReqRcvdCancelRcvd    0,
          pcePcepPeerNumReqRcvdClosed        0,
```

```
        pcePcepPeerNumRepRcvdUnknown        0,
        pcePcepPeerNumReqRcvdUnknown        0
    },
    {
        pcePcepPeerAddrType                 ipv4(1),  --PCE3
        pcePcepPeerAddr                     3.3.3.3,
        pcePcepPeerRole                     pce(2),
        pcePcepPeerDiscontinuityTime        TimeStamp,
        pcePcepPeerInitiateSession          true(1),
        pcePcepPeerSessionExists            false(0),
        pcePcepPeerNumSessSetupOK           1,
        pcePcepPeerNumSessSetupFail         0,
        pcePcepPeerSessionUpTime            TimeStamp,
        pcePcepPeerSessionFailTime          TimeStamp,
        pcePcepPeerSessionFailUpTime        TimeStamp,
        pcePcepPeerAvgRspTime               200,
        pcePcepPeerLWMRspTime               100,
        pcePcepPeerHWMRspTime               300,
        pcePcepPeerNumPCReqSent             4,
        pcePcepPeerNumPCReqRcvd             0,
        pcePcepPeerNumPCRepSent             0,
        pcePcepPeerNumPCRepRcvd             3,
        pcePcepPeerNumPCErrSent             0,
        pcePcepPeerNumPCErrRcvd             0,
        pcePcepPeerNumPCNtfSent             0,
        pcePcepPeerNumPCNtfRcvd             0,
        pcePcepPeerNumKeepaliveSent         123,
        pcePcepPeerNumKeepaliveRcvd         123,
        pcePcepPeerNumUnknownRcvd           0,
        pcePcepPeerNumCorruptRcvd           0,
        pcePcepPeerNumReqSent               4,
        pcePcepPeerNumSvecSent              0,
        pcePcepPeerNumSvecReqSent           0,
        pcePcepPeerNumReqSentPendRep        0,
        pcePcepPeerNumReqSentEroRcvd        3,
        pcePcepPeerNumReqSentNoPathRcvd     0,
        pcePcepPeerNumReqSentCancelRcvd     0,
        pcePcepPeerNumReqSentErrorRcvd      0,
        pcePcepPeerNumReqSentTimeout        0,
        pcePcepPeerNumReqSentCancelSent     0,
        pcePcepPeerNumReqSentClosed         1,
        pcePcepPeerNumReqRcvd               0,
        pcePcepPeerNumSvecRcvd              0,
        pcePcepPeerNumSvecReqRcvd           0,
        pcePcepPeerNumReqRcvdPendRep        0,
        pcePcepPeerNumReqRcvdEroSent        0,
        pcePcepPeerNumReqRcvdNoPathSent     0,
        pcePcepPeerNumReqRcvdCancelSent     0,
```

```
          pcePcepPeerNumReqRcvdErrorSent      0,
          pcePcepPeerNumReqRcvdCancelRcvd     0,
          pcePcepPeerNumReqRcvdClosed         0,
          pcePcepPeerNumRepRcvdUnknown        0,
          pcePcepPeerNumReqRcvdUnknown        0
      }

   In pcePcepSessTable {
          pcePcepSessInitiator                local(1), --PCE2
          pcePcepSessStateLastChange          TimeStamp,
          pcePcepSessState                    sessionUp(4),
          pcePcepSessConnectRetry             0,
          pcePcepSessLocalID                  1,
          pcePcepSessRemoteID                 1,
          pcePcepSessKeepaliveTimer           1,
          pcePcepSessPeerKeepaliveTimer       1,
          pcePcepSessDeadTimer                4,
          pcePcepSessPeerDeadTimer            4,
          pcePcepSessKAHoldTimeRem            1,
          pcePcepSessOverloaded               false(0),
          pcePcepSessOverloadTime             0,
          pcePcepSessPeerOverloaded           false(0),
          pcePcepSessPeerOverloadTime         0,
          pcePcepSessDiscontinuityTime        TimeStamp,
          pcePcepSessAvgRspTime               200,
          pcePcepSessLWMRspTime               100,
          pcePcepSessHWMRspTime               300,
          pcePcepSessNumPCReqSent             4,
          pcePcepSessNumPCReqRcvd             0,
          pcePcepSessNumPCRepSent             0,
          pcePcepSessNumPCRepRcvd             4,
          pcePcepSessNumPCErrSent             0,
          pcePcepSessNumPCErrRcvd             0,
          pcePcepSessNumPCNtfSent             0,
          pcePcepSessNumPCNtfRcvd             0,
          pcePcepSessNumKeepaliveSent         123,
          pcePcepSessNumKeepaliveRcvd         123,
          pcePcepSessNumUnknownRcvd           0,
          pcePcepSessNumCorruptRcvd           0,
          pcePcepSessNumReqSent               4,
          pcePcepSessNumSvecSent              0,
          pcePcepSessNumSvecReqSent           0,
          pcePcepSessNumReqSentPendRep        0,
          pcePcepSessNumReqSentEroRcvd        3,
          pcePcepSessNumReqSentNoPathRcvd     1,
          pcePcepSessNumReqSentCancelRcvd     0,
          pcePcepSessNumReqSentErrorRcvd      0,
          pcePcepSessNumReqSentTimeout        0,
```

```
          pcePcepSessNumReqSentCancelSent     0,
          pcePcepSessNumReqRcvd               0,
          pcePcepSessNumSvecRcvd              0,
          pcePcepSessNumSvecReqRcvd           0,
          pcePcepSessNumReqRcvdPendRep        0,
          pcePcepSessNumReqRcvdEroSent        0,
          pcePcepSessNumReqRcvdNoPathSent     0,
          pcePcepSessNumReqRcvdCancelSent     0,
          pcePcepSessNumReqRcvdErrorSent      0,
          pcePcepSessNumReqRcvdCancelRcvd     0,
          pcePcepSessNumRepRcvdUnknown        0,
          pcePcepSessNumReqRcvdUnknown        0
      }

      -- no session to PCE3
```

## Acknowledgements

The authors would like to thank Santanu Mazumder, Meral Shirazipour, and Adrian Farrel for their valuable input.

## Contributors

Dhruv Dhody
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
India

EMail: dhruv.ietf@gmail.com

Authors' Addresses

Agrahara Kiran Koushik
Brocade Communications, Inc.

EMail: kkoushik@brocade.com

Emile Stephan
Orange
2 Avenue Pierre Marzin
Lannion F-22307
France

EMail: emile.stephan@orange.com

Quintin Zhao
Huawei Technology
125 Nagog Technology Park
Acton, MA 01719
United States

EMail: qzhao@huawei.com

Daniel King
Old Dog Consulting

EMail: daniel@olddog.co.uk

Jonathan Hardwick
Metaswitch
100 Church Street
Enfield EN2 6BQ
United Kingdom

EMail: jonathan.hardwick@metaswitch.com