PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 23, 2013

A. Koushik
Cisco Systems, Inc.
S. Emile
France Telecom
Q. Zhao
Huawei Technology
D. King
Old Dog Consulting
J. Hardwick
Metaswitch

February 19, 2013

PCE communication protocol (PCEP) Management Information Base draft-ietf-pce-pcep-mib-04

Abstract

This memo defines an experimental portion of the Management Information Base for use with network management protocols in the Internet community. In particular, it describes managed objects for modeling of 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 Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

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

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

This Internet-Draft will expire on August 23, 2013.

Copyright Notice

Table of Contents

1. Introduction

The Path Computation Element (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 Path Computation Client (PCC) may make requests to a PCE for paths to be computed.

The PCE communication protocol (PCEP) is the communication protocol between a PCC and PCE for point-to-point (P2P) path computations and is defined in [RFC5440]. Such PCEP communication 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.

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

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 ATD 58, RFC 2580 [RFC2580].

3. 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, RFC 2119 [RFC2119].

4. Terminology

The terminology used in this document is built on notions introduced and discussed in PCE WG documents. The reader should be familiar with these documents.

Domain: any collection of network elements within a common sphere of address management or path computational responsibility.

IGP Area: OSPF Area or ISIS level/area.

This document also uses the terminology defined in [RFC4655] and [RFC5440].

5. PCEP MIB Module Architecture

The PCEP MIB contains the following information:

a. PCEP entity status.

b. PCEP peer information.

c. PCEP session information.

d. Notifications to indicate PCEP session changes.

5.1. Relations to other MIB modules

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

o InetAddressType

o InetAddress

o InetPortNumber

PCEP relies on existing protocols which 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.

6. Object Definitions

6.1. 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
            "201302191400Z" -- 19 February 2013
        ORGANIZATION
            "IETF Path Computation Element (PCE) Working Group"
        CONTACT-INFO
            "Email: pce@ietf.org
             WG charter:
                      http://www.ietf.org/html.charters/pce-charter.html"

        DESCRIPTION
           "This MIB module defines a collection of objects for managing
            PCE communication protocol (PCEP).

            Copyright (C) The IETF Trust (2013).  This version of this
            MIB module is part of RFC YYYY; see the RFC itself for full
            legal notices."
    -- RFC Ed,: replace YYYY with actual RFC number & remove this note
        REVISION
            "201302191400Z" -- 19 February 2013
        DESCRIPTION
            "Initial version, published as RFC YYYY."
    -- RFC Ed.: replace YYYY with actual RFC number & remove this note
        ::= { mib-2 XXX }
    -- RFC Ed.: replace XXX with IANA-assigned number & remove this note

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

    --
    -- PCE Entity Objects
    --
```

```
   pcePcepEntityTable OBJECT-TYPE
       SYNTAX      SEQUENCE OF PcePcepEntityEntry
       MAX-ACCESS  not-accessible
       STATUS      current
       DESCRIPTION
           "This table contains information about the PCEP Entity."
       ::= { pcePcepEntityObjects 1 }

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

   PcePcepEntityEntry ::= SEQUENCE {
       pcePcepEntityIndex               Unsigned32,
       pcePcepEntityAdminStatus         INTEGER,
       pcePcepEntityOperStatus          INTEGER,
       pcePcepEntityAddrType            InetAddressType,
       pcePcepEntityAddr                InetAddress,
       pcePcepEntityConnectTimer        Unsigned32,
       pcePcepEntityOpenWaitTimer       Unsigned32,
       pcePcepEntityKeepWaitTimer       Unsigned32,
       pcePcepEntityKeepAliveTimer      Unsigned32,
       pcePcepEntityDeadTimer           Unsigned32,
       pcePcepEntityMaxKeepAliveTimer   Unsigned32,
       pcePcepEntityMaxDeadTimer        Unsigned32,
       pcePcepEntityAllowNegotiation    TruthValue,
       pcePcepEntityMinKeepAliveTimer   Unsigned32,
       pcePcepEntityMinDeadTimer        Unsigned32,
       pcePcepEntitySyncTimer           Unsigned32,
       pcePcepEntityRequestTimer        Unsigned32,
       pcePcepEntityInitBackoffTimer    Unsigned32,
       pcePcepEntityMaxBackoffTimer     Unsigned32,
       pcePcepEntityMaxSessions         Unsigned32,
       pcePcepEntityMaxUnknownReqs      Unsigned32,
       pcePcepEntityMaxUnknownMsgs      Unsigned32
   }

   pcePcepEntityIndex OBJECT-TYPE
       SYNTAX      Unsigned32 (1..2147483647)
       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."
        ::= { pcePcepEntityEntry 2 }

    pcePcepEntityOperStatus OBJECT-TYPE
        SYNTAX      INTEGER {
                      operStatusUp(1),          -- active
                      operStatusDown(2),        -- inactive
                      operStatusGoingUp(3),     -- activating
                      operStatusGoingDown(4),   -- deactivating
                      operStatusFailed(5),      -- failed, will recover
                                                -- when possible
                      operStatusFailedPerm(6)   -- operator intervention
                                                -- required
                    }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The operational status of the PCEP entity."
        ::= { 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 pcePcepPeerAddr object should
             be interpreted."
        ::= { pcePcepEntityEntry 4 }

    pcePcepEntityAddr OBJECT-TYPE
        SYNTAX      InetAddress
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The 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."
        ::= { 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 PCEP peer.  If a TCP connection is not
             established within this time then PCEP aborts the session
             setup attempt."
        ::= { pcePcepEntityEntry 6 }

    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 PCEP peer.  If no Open message is received
             within this time then PCEP aborts the session setup
             attempt."
        ::= { pcePcepEntityEntry 7 }

    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 PCEP peer during session
             initialization.  If no Keepalive or PCErr message is
             received within this time then PCEP aborts the session setup
             attempt."
        ::= { pcePcepEntityEntry 8 }

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

```
            "The keep alive 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 PCEP 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 PCEP 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 9 }

    pcePcepEntityDeadTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The dead timer 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 PCEP peer should declare a
             session down if it does not receive any PCEP messages.

             pcePcepEntityDeadTimer is recommended to be 4 times the
             pcePcepEntityKeepAliveTimer value.  Zero means suggesting
             that the peer does not run a dead timer at all; it is only
             allowed when pcePcepEntityKeepAliveTimer is also zero."
        ::= { pcePcepEntityEntry 10 }

    pcePcepEntityMaxKeepAliveTimer OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "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 11 }

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

```
        STATUS      current
        DESCRIPTION
            "The maximum value that this PCEP entity will accept from a
             peer for the Dead timer.  Zero means that the PCEP entity
             will allow not running a Dead timer.

             A Dead timer will not be accepted unless it is both greater
             than the session Keepalive timer and less than this field."
        ::= { pcePcepEntityEntry 12 }

    pcePcepEntityAllowNegotiation OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "Whether the PCEP entity will permit negotiation of session
             parameters."
        ::= { pcePcepEntityEntry 13 }

    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 14 }

    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 Dead timer.  Zero
             means that the PCEP entity insists on not running a Dead
             timer.

             A Dead timer will not be accepted unless it is both greater
             than the session Keepalive timer and greater than this
             field."
        ::= { pcePcepEntityEntry 15 }

    pcePcepEntitySyncTimer OBJECT-TYPE
```

```
        SYNTAX      Unsigned32 (1..65535)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The value of SYNC timer is used in the case of 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
             SYNC timer 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
             reasons (for example, a misbehaving PCC)."
        ::= { pcePcepEntityEntry 16 }

    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 17 }

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

             The back-off time increases for each failed session setup
             attempt, until a maximum back-off time is reached.  The
             maximum back-off time is pcePcepEntityMaxBackoffTimer."
        ::= { pcePcepEntityEntry 18 }

    pcePcepEntityMaxBackoffTimer OBJECT-TYPE
        SYNTAX      Unsigned32
        UNITS       "seconds"
        MAX-ACCESS  read-only
```

```
        STATUS      current
        DESCRIPTION
            "The maximum back-off time for retrying a failed session
             setup attempt to a peer.

             The back-off time increases for each failed session setup
             attempt, until this maximum value is reached.  Session
             setup attempts then repeat periodically without any
             further increase in back-off time."
        ::= { pcePcepEntityEntry 19 }

    pcePcepEntityMaxSessions OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "Maximum number of sessions involving this PCEP entity
             that can exist at any time."
        ::= { pcePcepEntityEntry 20 }

    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.

             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 21 }

    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."
        ::= { pcePcepEntityEntry 22 }

    --
    -- The PCEP Peer Table
```

```
   --

   pcePcepPeerObjects OBJECT IDENTIFIER ::= { pcePcepMIBObjects 2 }

   pcePcepPeerTable OBJECT-TYPE
       SYNTAX      SEQUENCE OF PcePcepPeerEntry
       MAX-ACCESS  not-accessible
       STATUS      current
       DESCRIPTION
           "Information about PCEP peers known by the local PCEP
            speaker.

            This MIB table gives PCEP peer information that spans PCEP
            sessions.  Information about current PCEP sessions can be
            found in the pcePcepSessTable MIB table."
       ::= { pcePcepPeerObjects 1 }

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

   PcePcepPeerEntry ::= SEQUENCE {
       pcePcepPeerAddrType                InetAddressType,
       pcePcepPeerAddr                    InetAddress,
       pcePcepPeerDiscontinuityTime       TimeStamp,
       pcePcepPeerInitiateSession         TruthValue,
       pcePcepPeerSessionExists           TruthValue,
       pcePcepPeerNumSessSetupOK          Counter32,
       pcePcepPeerNumSessSetupFail        Counter32,
       pcePcepPeerSessionUpTime           TimeStamp,
       pcePcepPeerSessionFailTime         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,
        pcePcepPeerNumReqSent               Counter32,
        pcePcepPeerNumSvecSent              Counter32,
        pcePcepPeerNumReqSentPendRep        Counter32,
        pcePcepPeerNumReqSentEroRcvd        Counter32,
        pcePcepPeerNumReqSentNoPathRcvd     Counter32,
        pcePcepPeerNumReqSentCancelRcvd     Counter32,
        pcePcepPeerNumReqSentErrorRcvd      Counter32,
        pcePcepPeerNumReqSentTimeout        Counter32,
        pcePcepPeerNumReqSentCancelSent     Counter32,
        pcePcepPeerNumReqSentClosed         Counter32,
        pcePcepPeerNumReqRcvd               Counter32,
        pcePcepPeerNumSvecRcvd              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 peer Internet address type (IPv4 or IPv6).

             This specifies how pcePcepPeerAddr should be interpreted."
        ::= { pcePcepPeerEntry 1 }

    pcePcepPeerAddr OBJECT-TYPE
        SYNTAX      InetAddress (SIZE (4..32))
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "The Internet address of the peer.

             The type of this address is specified by
             pcePcepPeerAddrType. "
```

```
        ::= { pcePcepPeerEntry 2 }

    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 3 }

    pcePcepPeerInitiateSession OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "Indicates whether the PCEP Entity initiates sessions to this
             peer, or waits for the peer to initiate a session."
        ::= { pcePcepPeerEntry 4 }

    pcePcepPeerSessionExists OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "Indicates whether a session with this peer currently
             exists."
        ::= { pcePcepPeerEntry 5 }

    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."
        ::= { pcePcepPeerEntry 6 }

    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 reaching session state
             pceSessionUp."
        ::= { pcePcepPeerEntry 7 }
```

```
   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 8 }

   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 9 }

   pcePcepPeerAvgRspTime OBJECT-TYPE
       SYNTAX      Unsigned32 (1..65535)
       UNITS       "seconds"
       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."
       ::= { pcePcepPeerEntry 10 }

   pcePcepPeerLWMRspTime OBJECT-TYPE
       SYNTAX      Unsigned32 (1..65535)
       UNITS       "seconds"
       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."
       ::= { pcePcepPeerEntry 11 }
```

```
   pcePcepPeerHWMRspTime OBJECT-TYPE
       SYNTAX      Unsigned32 (1..65535)
       UNITS       "seconds"
       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."
       ::= { pcePcepPeerEntry 12 }

   pcePcepPeerNumPCReqSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCReq messages sent to this peer."
       ::= { pcePcepPeerEntry 13 }

   pcePcepPeerNumPCReqRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of PCReq messages received from this peer."
       ::= { pcePcepPeerEntry 14 }

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

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

   pcePcepPeerNumPCErrSent OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
```

```
        STATUS      current
        DESCRIPTION
            "The number of PCErr messages sent to this peer."
        ::= { pcePcepPeerEntry 17 }

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

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

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

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

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

    pcePcepPeerNumUnknownRcvd OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
```

```
        STATUS      current
        DESCRIPTION
            "The number of unknown messages received from this peer."
        ::= { pcePcepPeerEntry 23 }

    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 24 }

    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 25 }

    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 26 }

    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 object was received.  Such
             responses indicate that a path was successfully computed by
             the peer."
        ::= { pcePcepPeerEntry 27 }
```

```
   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 28 }

   pcePcepPeerNumReqSentCancelRcvd OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The number of requests that were cancelled 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 29 }

   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 30 }

   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 31 }

   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 speaker sending a
             PCNtf."
        ::= { pcePcepPeerEntry 32 }

    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 33 }

    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 34 }

    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 35 }

    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 36 }

    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 object was sent.  Such
             responses indicate that a path was successfully computed by
             the local PCEP speaker."
        ::= { pcePcepPeerEntry 37 }

    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 speaker could
             not find a path to satisfy the request."
        ::= { pcePcepPeerEntry 38 }

    pcePcepPeerNumReqRcvdCancelSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests received from this peer that were
             cancelled by the local PCEP speaker 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 39 }

    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 speaker 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 40 }

    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 41 }

    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 42 }

    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 43 }

    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 44 }

    --
    -- The PCEP Sessions Table
    --

    pcePcepSessObjects OBJECT IDENTIFIER ::= { pcePcepMIBObjects 3 }
```

```
   pcePcepSessTable OBJECT-TYPE
       SYNTAX      SEQUENCE OF PcePcepSessEntry
       MAX-ACCESS  not-accessible
       STATUS      current
       DESCRIPTION
           "A table of PCEP sessions that involve the local PCEP
            speaker.  Each row in this table represents a single
            session."
       ::= { pcePcepSessObjects 1 }

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

            An entry in this table exists only if the corresponding PCEP
            session has been initialized by some event, such as manual
            user configuration, autodiscovery of a peer, or an incoming
            TCP connection.

            An entry appears in this table when the corresponding PCEP
            session transitions out of idle state.  If the PCEP session
            transitions back into idle state then the corresponding
            entry in this table is removed."
       INDEX { pcePcepEntityIndex,
               pcePcepPeerAddrType,
               pcePcepPeerAddr,
               pcePcepSessInitiator }
       ::= { pcePcepSessTable 1 }

   PcePcepSessEntry ::= SEQUENCE {
       pcePcepSessInitiator               INTEGER,
       pcePcepSessStateLastChange         TimeStamp,
       pcePcepSessState                   INTEGER,
       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,
      pcePcepSessNumReqSent               Counter32,
      pcePcepSessNumSvecSent              Counter32,
      pcePcepSessNumReqSentPendRep        Counter32,
      pcePcepSessNumReqSentEroRcvd        Counter32,
      pcePcepSessNumReqSentNoPathRcvd     Counter32,
      pcePcepSessNumReqSentCancelRcvd     Counter32,
      pcePcepSessNumReqSentErrorRcvd      Counter32,
      pcePcepSessNumReqSentTimeout        Counter32,
      pcePcepSessNumReqSentCancelSent     Counter32,
      pcePcepSessNumReqRcvd               Counter32,
      pcePcepSessNumSvecRcvd              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 speaker or the
           remote PCEP speaker.
```

```
            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 the MIB.  The sessions are
            distinguished by who initiated them, and so this field is an
            index for the 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
             entiries do not exist in this table in the idle state."
        ::= { pcePcepSessEntry 3 }

    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
             speaker 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 4 }
```

```
   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 5 }

   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 speaker
            transmits PCEP messages on this PCEP session.  Zero means
            that the local PCEP speaker 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 6 }

   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 7 }

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

```
        DESCRIPTION
            "The local PCEP speaker's DeadTimer interval for this PCEP
             session."
        ::= { pcePcepSessEntry 8 }

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

             If pcePcepSessState is tcpPending or openWait then this
             field is not used and MUST be set to zero."
        ::= { pcePcepSessEntry 9 }

    pcePcepSessKAHoldTimeRem OBJECT-TYPE
        SYNTAX      Unsigned32 (0..255)
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The keep alive 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 10 }

    pcePcepSessOverloaded OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "If the local PCEP speaker has informed the peer that it is
             currently overloaded, then this is set to true.  Otherwise,
             it is set to false."
        ::= { pcePcepSessEntry 11 }

    pcePcepSessOverloadTime OBJECT-TYPE
        SYNTAX      Unsigned32
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The interval of time until the local PCEP speaker 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 12 }

    pcePcepSessPeerOverloaded OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "If the peer has informed the local PCEP speaker that it is
            currently overloaded, then this is set to true.  Otherwise,
            it is set to false."
        ::= { pcePcepSessEntry 13 }

    pcePcepSessPeerOverloadTime OBJECT-TYPE
        SYNTAX      Unsigned32
        UNITS       "seconds"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The interval of time 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 14 }

    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 15 }

    pcePcepSessAvgRspTime OBJECT-TYPE
        SYNTAX      Unsigned32 (1..65535)
        UNITS       "seconds"
        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 16 }

    pcePcepSessLWMRspTime OBJECT-TYPE
        SYNTAX      Unsigned32 (1..65535)
        UNITS       "seconds"
        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 17 }

    pcePcepSessHWMRspTime OBJECT-TYPE
        SYNTAX      Unsigned32 (1..65535)
        UNITS       "seconds"
        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 18 }

    pcePcepSessNumPCReqSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of PCReq messages sent on this session."
        ::= { pcePcepSessEntry 19 }

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

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

```
        DESCRIPTION
            "The number of PCRep messages sent on this session."
        ::= { pcePcepSessEntry 21 }

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

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

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

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

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

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

```
        DESCRIPTION
            "The number of Keepalive messages sent on this session."
        ::= { pcePcepSessEntry 27 }

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

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

    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 30 }

    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 31 }

    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 32 }

    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 33 }

    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 34 }

    pcePcepSessNumReqSentCancelRcvd OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests sent on this session that were
             cancelled 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 35 }

    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 36 }

    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 37 }

    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
            speaker sending a PCNtf."
        ::= { pcePcepSessEntry 38 }

    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 39 }

    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 40 }

    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 41 }

    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 42 }

    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 43 }

    pcePcepSessNumReqRcvdCancelSent OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of requests received on this session that were
             cancelled by the local PCEP speaker 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 44 }

    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 speaker 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 45 }

    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 46 }

    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 47 }

    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 48 }

    ---
    --- 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 }

    --
    -- 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
                                }
        ::= { pcePcepCompliances 1 }

    -- units of conformance

    pcePcepGeneralGroup OBJECT-GROUP
        OBJECTS { pcePcepEntityAdminStatus,
```

```
                pcePcepEntityOperStatus,
                pcePcepEntityAddrType,
                pcePcepEntityAddr,
                pcePcepEntityConnectTimer,
                pcePcepEntityOpenWaitTimer,
                pcePcepEntityKeepWaitTimer,
                pcePcepEntityKeepAliveTimer,
                pcePcepEntityDeadTimer,
                pcePcepEntityMaxKeepAliveTimer,
                pcePcepEntityMaxDeadTimer,
                pcePcepEntityAllowNegotiation,
                pcePcepEntityMinKeepAliveTimer,
                pcePcepEntityMinDeadTimer,
                pcePcepEntitySyncTimer,
                pcePcepEntityRequestTimer,
                pcePcepEntityInitBackoffTimer,
                pcePcepEntityMaxBackoffTimer,
                pcePcepEntityMaxSessions,
                pcePcepEntityMaxUnknownReqs,
                pcePcepEntityMaxUnknownMsgs,
                pcePcepPeerDiscontinuityTime,
                pcePcepPeerInitiateSession,
                pcePcepPeerSessionExists,
                pcePcepPeerNumSessSetupOK,
                pcePcepPeerNumSessSetupFail,
                pcePcepPeerSessionUpTime,
                pcePcepPeerSessionFailTime,
                pcePcepPeerAvgRspTime,
                pcePcepPeerLWMRspTime,
                pcePcepPeerHWMRspTime,
                pcePcepPeerNumPCReqSent,
                pcePcepPeerNumPCReqRcvd,
                pcePcepPeerNumPCRepSent,
                pcePcepPeerNumPCRepRcvd,
                pcePcepPeerNumPCErrSent,
                pcePcepPeerNumPCErrRcvd,
                pcePcepPeerNumPCNtfSent,
                pcePcepPeerNumPCNtfRcvd,
                pcePcepPeerNumKeepaliveSent,
                pcePcepPeerNumKeepaliveRcvd,
                pcePcepPeerNumUnknownRcvd,
                pcePcepPeerNumReqSent,
                pcePcepPeerNumSvecSent,
                pcePcepPeerNumReqSentPendRep,
                pcePcepPeerNumReqSentEroRcvd,
                pcePcepPeerNumReqSentNoPathRcvd,
                pcePcepPeerNumReqSentCancelRcvd,
                pcePcepPeerNumReqSentErrorRcvd,
```

```
                pcePcepPeerNumReqSentTimeout,
                pcePcepPeerNumReqSentCancelSent,
                pcePcepPeerNumReqSentClosed,
                pcePcepPeerNumReqRcvd,
                pcePcepPeerNumSvecRcvd,
                pcePcepPeerNumReqRcvdPendRep,
                pcePcepPeerNumReqRcvdEroSent,
                pcePcepPeerNumReqRcvdNoPathSent,
                pcePcepPeerNumReqRcvdCancelSent,
                pcePcepPeerNumReqRcvdErrorSent,
                pcePcepPeerNumReqRcvdCancelRcvd,
                pcePcepPeerNumReqRcvdClosed,
                pcePcepPeerNumRepRcvdUnknown,
                pcePcepPeerNumReqRcvdUnknown,
                pcePcepSessStateLastChange,
                pcePcepSessState,
                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,
                pcePcepSessNumReqSent,
                pcePcepSessNumSvecSent,
                pcePcepSessNumReqSentPendRep,
                pcePcepSessNumReqSentEroRcvd,
                pcePcepSessNumReqSentNoPathRcvd,
                pcePcepSessNumReqSentCancelRcvd,
```

```
                pcePcepSessNumReqSentErrorRcvd,
                pcePcepSessNumReqSentTimeout,
                pcePcepSessNumReqSentCancelSent,
                pcePcepSessNumReqRcvd,
                pcePcepSessNumSvecRcvd,
                pcePcepSessNumReqRcvdPendRep,
                pcePcepSessNumReqRcvdEroSent,
                pcePcepSessNumReqRcvdNoPathSent,
                pcePcepSessNumReqRcvdCancelSent,
                pcePcepSessNumReqRcvdErrorSent,
                pcePcepSessNumReqRcvdCancelRcvd,
                pcePcepSessNumRepRcvdUnknown,
                pcePcepSessNumReqRcvdUnknown
              }
    STATUS current
    DESCRIPTION
        "Objects that apply to all PCEP MIB implementations."
    ::= { pcePcepGroups 1 }

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

END
```

7. Security Considerations

The readable objects in the PCE-PCEP-MIB module (i.e., those with MAX-ACCESS other than not-accessible) may be considered sensitive in some environments since, collectively, they provide information about the amount and frequency of path computation requests and responses within the network and can reveal some aspects of their configuration.

In such environments it is important to control also GET and NOTIFY access to these objects and possibly even to encrypt their values when sending them over the network via SNMP.

SNMP versions prior to SNMPv3 did not include adequate security. Even if the network itself is secure (for example by using IPsec), even then, 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.

It is RECOMMENDED that implementers consider the security features as provided by the SNMPv3 framework (see [RFC3410], section 8), including full support for the SNMPv3 cryptographic mechanisms (for authentication and privacy).

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.

8. 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 XXX }

Editor's Note (to be removed prior to publication): the IANA is requested to assign a value for "XXX" under the 'mib-2' subtree and to record the assignment in the SMI Numbers registry. When the assignment has been made, the RFC Editor is asked to replace "XXX" (here and in the MIB module) with the assigned value and to remove this note.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2578] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.

[RFC2579] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.

[RFC2580] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.

[RFC4001] Daniele, M., Haberman, B., Routhier, S., and J. Schoenwaelder, "Textual Conventions for Internet Network Addresses", RFC 4001, February 2005.

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

9.2. Normative References

[RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.

Appendix A. Acknowledgement

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

Funding for the RFC Editor function is currently provided by the Internet Society.

Authors' Addresses

A S Kiran Koushik
Cisco Systems, Inc.

EMail: kkoushik@cisco.com

Stephan Emile
France Telecom
2 avenue Pierre Marzin
Lannion F-22307
France

EMail: emile.stephan@orange-ftgroup.com

Quintin Zhao
Huawei Technology
125 Nagog Technology Park
Acton, MA 01719
US

EMail: qzhao@huawei.com

Daniel King
Old Dog Consulting
UK

EMail: daniel@olddog.co.uk

Jonathan Hardwick
Metaswitch
100 Church Street
Enfield EN2 6BQ
UK

EMail: jon.hardwick@metaswitch.com