PacketCable™ 1.5 Specifications

MTA MIB

PKT-SP-MIB-MTA1.5-C01-191120

CLOSED

Notice

DISCLAIMER

This document is furnished on an "AS IS" basis and neither CableLabs nor its members provides any representation or warranty, express or implied, regarding the accuracy, completeness, noninfringement, or fitness for a particular purpose of this document, or any document referenced herein. Any use or reliance on the information or opinion in this document is at the risk of the user, and CableLabs and its members shall not be liable for any damage or injury incurred by any person arising out of the completeness, accuracy, or utility of any information or opinion contained in the document.

CableLabs reserves the right to revise this document for any reason including, but not limited to, changes in laws, regulations, or standards promulgated by various entities, technology advances, or changes in equipment design, manufacturing techniques, or operating procedures described, or referred to, herein.

This document is not to be construed to suggest that any company modify or change any of its products or procedures, nor does this document represent a commitment by CableLabs or any of its members to purchase any product whether or not it meets the characteristics described in the document. Unless granted in a separate written agreement from CableLabs, nothing contained herein shall be construed to confer any license or right to any intellectual property. This document is not to be construed as an endorsement of any product or company or as the adoption or promulgation of any guidelines, standards, or recommendations.

Document Status Sheet

Document Control Number:	PKT-SP-MIB-MTA1.5-C01-191120			
Document Title:	MTA MIB			
Revision History:	D01 — Draft Released September 30, 2004 I01 — January 28, 2005 C01 – November 20, 2019			
Date:	November 20, 2019			
Status:	~~Work in Progress~~	~~Draft~~	~~Issued~~	**Closed**
Distribution Restrictions:	~~Author Only~~	~~CL/Member~~	~~CL/ PacketCable/ Vendor~~	**Public**

Key to Document Status Codes:

Work in Progress	An incomplete document, designed to guide discussion and generate feedback that may include several alternative requirements for consideration.
Draft	A document in specification format considered largely complete, but lacking review by Members and vendors. Drafts are susceptible to substantial change during the review process.
Issued	A generally public document that has undergone Member and Technology Supplier review, cross-vendor interoperability, and is for Certification testing if applicable. Issued Specifications are subject to the Engineering Change Process.
Closed	A static document, reviewed, tested, validated, and closed to further engineering change requests to the specification through CableLabs.

Trademarks

CONTENTS

1 SCOPE

This specification describes the PacketCable 1.5 MTA MIB requirement.

2 REFERENCES

2.1 Normative References

In order to claim compliance with this specification, it is necessary to conform to the following standards and other works as indicated, in addition to the other requirements of this specification. Notwithstanding, intellectual property rights may be required to use or implement such normative references.

[1] PacketCable 1.5 MIB Framework, PKT-SP-MIBS1.5-C01-191120, November 20, 2019, Cable Television Laboratories, Inc.

[2] PacketCable 1.5 Network-Based Call Signaling Protocol Specification**,** PKT-SP-NCS1.5-C01-191120, November 20, 2019, Cable Television Laboratories, Inc.

[3] PacketCable 1.5 MTA Device Provisioning Specification**,** PKT-SP-PROV1.5-C01-191120, November 20, 2019, Cable Television Laboratories, Inc.

[4] IETF RFC 3413, Simple Network Management Protocol (SNMP) Applications.

[5] IETF STD 62, Simple Network Management Protocol Version 3 (SNMPv3), December 2002.

[6] IETF RFC 3412, Message Processing and Dispatching for the Simple Network Management Protocol (SNMP).

[7] PacketCable 1.5 Signaling MIB Specification, PKT-SP-MIB-SIG1.5-C01-191120, November 20, 2019, Cable Television Laboratories, Inc.

[8] IETF RFC 3414, User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3).

[9] IETF RFC 3415, View-based Access Control Model (VACM) for Simple Network Management Protocol (SNMP).

[10] PacketCable 1.5 Security Specification, PKT-SP-SEC1.5-C01-191120, November 20, 2019, Cable Television Laboratories, Inc.

[11] IETF RFC 1350/STD0033, The TFTP Protocol (Revision 2), July 1992.

[12] IETF RFC 3617, Uniform Resource Identifier (URI) Scheme and Applicability Statement for the Trivial File Transfer Protocol (TFTP), October 2003.

[13] IETF RFC 2616, Hypertext Transfer Protocol -- HTTP/1.1, June 1999.

2.2 Informative References

[14] PacketCable 1.5 Architecture Framework Technical Report, PKT-TR-ARCH1.5-C01-191120, November 20, 2019, Cable Television Laboratories Inc.

2.3 Reference Acquisition

- Cable Television Laboratories, Inc., 858 Coal Creek Circle, Louisville, CO 80027; Phone 303-661-9100; Fax 303-661-9199; Internet: https://www.cablelabs.com/
- Internet Engineering Task Force (IETF) Secretariat c/o Corporation for National Research Initiatives, 1895 Preston White Drive, Suite 100, Reston, VA 20191-5434, Phone 703-620-8990, Fax 703-620-9071, Internet http://www.ietf.org/

3 ABBREVIATIONS

There are no abbreviations used in this document.

4 REQUIREMENTS

The PacketCable™ MTA MIB MUST be implemented as defined below.

```
PKTC-MTA-MIB DEFINITIONS ::= BEGIN
      IMPORTS
      MODULE-IDENTITY,
      OBJECT-TYPE,
      Integer32, Counter32,
      BITS,IpAddress, NOTIFICATION-TYPE         FROM SNMPv2-SMI
      TruthValue, RowStatus, DisplayString,
      MacAddress, TEXTUAL-CONVENTION            FROM SNMPv2-TC
      OBJECT-GROUP, MODULE-COMPLIANCE,
      NOTIFICATION-GROUP                        FROM SNMPv2-CONF
      clabProjPacketCable                       FROM CLAB-DEF-MIB
      ifIndex                                   FROM IF-MIB
      SnmpAdminString                           FROM SNMP-FRAMEWORK-MIB
      sysDescr                                  FROM SNMPv2-MIB;

pktcMtaMib MODULE-IDENTITY
    LAST-UPDATED    "200501280000Z" -- January 28, 2005
    ORGANIZATION    "Packet Cable OSS Group"
CONTACT-INFO
            "Sumanth Channabasappa
            Postal: Cable Television Laboratories, Inc.
            858 Coal Creek Circle
            Louisville, Colorado 80027-9750
            U.S.A.
            Phone:  +1 303-661-9100
            Fax:    +1 303-661-9199
            E-mail: mibs@cablelabs.com"
DESCRIPTION
            "This MIB module supplies the basic management objects
            for the MTA Device
            Acknowledgements:
            Angela Lyda           -      Arris Interactive
            Chris Melle           -      AT&T Broadband Labs
            Sasha Medvinsky       -      Motorola
            Roy Spitzer           -      Telogy Networks, Inc.
            Rick Vetter           -      Motorola
            Eugene Nechamkin      -      BroadCom Corp.
            Satish Kumar          -      Texas Instruments
            Copyright 1999-2005 Cable Television Laboratories, Inc.
             All rights reserved."
      REVISION "200501280000Z "
      DESCRIPTION
            "This revision, published as part of the PacketCable 1.5
            MIB MTA Specification I01."
      ::=  { clabProjPacketCable 1 }

-- Textual conventions
      X509Certificate ::= TEXTUAL-CONVENTION
      STATUS current
      DESCRIPTION
          "An X509 digital certificate encoded as an ASN.1 DER object."
      SYNTAX OCTET STRING (SIZE (0..4096))
--

--  PacketCable 1.5 only supports Embedded MTAs
--
--====================================================================
--
```

```
--  The MTA MIB only supports a single provisioning server.
--
--============================================================================

pktcMtaMibObjects       OBJECT IDENTIFIER ::= { pktcMtaMib 1 }
pktcMtaDevBase          OBJECT IDENTIFIER ::= { pktcMtaMibObjects 1 }
pktcMtaDevServer        OBJECT IDENTIFIER ::= { pktcMtaMibObjects 2 }
pktcMtaDevSecurity      OBJECT IDENTIFIER ::= { pktcMtaMibObjects 3 }

--
--  The following group describes the base objects in the MTA
--
pktcMtaDevResetNow  OBJECT-TYPE
      SYNTAX      TruthValue
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
            "Setting this object to true(1) causes the device to reset.
            Reading this object always returns false(2).  When
            pktcMtaDevResetNow is set to true, the following actions
            occur:
            1.  All connections (if present) are flushed locally
            2.  All current actions such as ringing immediately
            terminate
            3.  Requests for notifications such as notification based
            on digit map recognition are flushed
            4.  All endpoints are disabled.
            5.  The provisioning flow is started at step MTA - 1."
      ::= { pktcMtaDevBase 1 }

pktcMtaDevSerialNumber  OBJECT-TYPE
      SYNTAX      SnmpAdminString(SIZE (0..128))
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "This object specifies the manufacturer's serial number
             for this MTA. The value of this object MUST be identical
             to the value specified in DHCP option 43 sub-option 4. "
       REFERENCE
            "PacketCable MTA Device Provisioning Specification;
             RFC 2132, DHCP Options and BOOTP Vendor Extensions"
      ::= { pktcMtaDevBase 2 }

pktcMtaDevHardwareVersion  OBJECT-TYPE
      SYNTAX      SnmpAdminString(SIZE (0..48))
      MAX-ACCESS  read-only
      STATUS      obsolete
      DESCRIPTION
            "The manufacturer's hardware version for this MTA."
      ::= { pktcMtaDevBase 3 }

pktcMtaDevMacAddress   OBJECT-TYPE
      SYNTAX      MacAddress
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "This object specifies the telephony MAC address for
             this device. The value of this object MUST be identical
             to the value specified in DHCP option 43 sub-option 11. "
       REFERENCE
            "PacketCable MTA Device Provisioning Specification;
              RFC 2132, DHCP Options and BOOTP Vendor Extensions"
      ::= { pktcMtaDevBase 4 }
```

```
pktcMtaDevFQDN       OBJECT-TYPE
      SYNTAX      SnmpAdminString
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "The Fully Qualified Domain Name for this MTA."
      ::= { pktcMtaDevBase 5 }

pktcMtaDevEndPntCount     OBJECT-TYPE
      SYNTAX       Integer32 (1..255)
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "The physical end points for this MTA."
      ::= { pktcMtaDevBase 6 }

pktcMtaDevEnabled     OBJECT-TYPE
      SYNTAX      TruthValue
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
            "This object contains the MTA Admin Status of this device.
             If this object is set to 'true', the MTA is
             administratively enabled and the MTA MUST be able to
             interact with PacketCable entities such as CMS,
             Provisioning Server, KDC, other MTAs and MGs on all
             PacketCable interfaces.
             If this object is set to 'false', the MTA is
             administratively disabled and the MTA MUST perform the
             following actions for all endpoints:
               -  Shutdown all media sessions if present,
               -  Shutdown NCS signaling by following the Restart in
                  Progress procedures in the PacketCable NCS
                  specification.
             Additionally, the MTA MUST maintain the SNMP Interface for
             management. Also, the MTA MUST NOT continue Kerberized Key
             Management with CMSes until this object is set to 'true'.
             Note: MTAs MUST renew the CMS kerberos tickets according
             to the PacketCable Security Specification"
      REFERENCE
            "PacketCable Security Specification;
             PacketCable MTA Device Provisioning Specification"
      ::= { pktcMtaDevBase 7 }

pktcMtaDevTypeIdentifier     OBJECT-TYPE
      SYNTAX      SnmpAdminString
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "This is a copy of the device type identifier used in the
            DHCP option 60 exchanged between the MTA and the DHCP
            server."
      ::= { pktcMtaDevBase 8 }

pktcMtaDevProvisioningState     OBJECT-TYPE
      SYNTAX      INTEGER {
             pass                       (1),
             inProgress                 (2),
             failConfigFileError        (3),
             passWithWarnings           (4),
             passWithIncompleteParsing  (5),
             failureInternalError       (6),
             failOtherReason            (7)
```

```
      }
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "This object indicates the completion state of the
            MTA device provisioning process.

            pass:
            If the configuration file could be parsed successfully
            and the MTA is able to reflect the same in its
            MIB, the MTA MUST return the value 'pass'.

            inProgress:
            If the MTA is in the process of being provisioned,
            the MTA MUST return the value 'inProgress'.

            failConfigFileError:
            If the configuration file was in error due to incorrect
            values in the mandatory parameters, the MTA MUST reject
            the configuration file and the MTA MUST return the value
            'failConfigFileError'.

            passWithWarnings:
            If the configuration file had proper values for all the
            mandatory parameters but has errors in any of the optional
            parameters (this includes any vendor specific OIDs which
            are incorrect or not known to the MTA), the MTA MUST
            return the value 'passWithWarnings'.

            passWithIncompleteParsing:
            If the configuration file is valid, but the MTA cannot
            reflect the same in its configuration (for example, too
            many entries caused memory exhaustion), it must accept
            the CMS configuration entries related and the MTA MUST
            return the value 'passWithIncompleteParsing'.

            failureInternalError:
            If the configuration file cannot be parsed due to an
            internal error, the MTA MUST return the value
            'failureInternalError'.

            failureOtherReason:
            If the MTA cannot accept the configuration file for any
            other reason than the ones stated above, the MTA MUST
            return the value 'failureOtherReason'.

            When a final SNMP INFORM is sent as part of Step 25 of
            the MTA Provisioning process, this parameter is also
            included in the final INFORM message."
       REFERENCE
            "PacketCable MTA Device Provisioning Specification"
      ::= { pktcMtaDevBase 9 }

pktcMtaDevHttpAccess     OBJECT-TYPE
      SYNTAX      TruthValue
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "This indicates whether HTTP file access is supported for
            MTA configuration file transfer."
      ::= { pktcMtaDevBase 10 }

pktcMtaDevProvisioningTimer  OBJECT-TYPE
```

```
      SYNTAX      Integer32 (0..30)
      UNITS       "minutes"
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
            "This object defines the time interval for the
            provisioning flow to complete. The MTA MUST finish
            all provisioning operations starting from the moment when
            an MTA receives its DHCP ACK and ending at the moment when
            the MTA downloads its configuration file (e.g., MTA5 to
            MTA23 for Secure Flow) within the period of time set by
            this object.  Failure to comply with this condition
            constitutes the provisioning flow failure. If the
            object is set to 0, the MTA MUST ignore the provisioning
            timer condition."
      REFERENCE
        "PacketCable MTA Device Provisioning Specification."
      DEFVAL { 10 }
      ::=  { pktcMtaDevBase 11 }

pktcMtaDevProvisioningCounter  OBJECT-TYPE
      SYNTAX      Counter32
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "This object is the count of the number of times the
            provisioning cycle has looped through step MTA-1 since
            the last reboot."
      ::=  { pktcMtaDevBase 12 }
--
pktcMtaDevErrorOidsTable  OBJECT-TYPE
      SYNTAX SEQUENCE OF PktcMtaDevErrorOidsEntry
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION
            "If pktcMtaDevProvisioningState is reported with anything
            other than a pass(1) then this table is populated with the
            necessary information, each pertaining to observations of
            the configuration file. Even if different parameters
            share the same error ( Ex: All Realm Names are invalid ),
            all recognized errors must be reported as different
            instances."
      ::= { pktcMtaDevBase 13 }

pktcMtaDevErrorOidsEntry  OBJECT-TYPE
      SYNTAX PktcMtaDevErrorOidsEntry
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION
            "This contains the necessary information an MTA must
             attempt to provide in case the configuration file
             is not parsed and/or accepted in its entirety."
      INDEX { pktcMtaDevErrorOidIndex }
      ::= { pktcMtaDevErrorOidsTable 1 }

PktcMtaDevErrorOidsEntry ::= SEQUENCE {
      pktcMtaDevErrorOidIndex           Integer32,
      pktcMtaDevErrorOid                SnmpAdminString,
      pktcMtaDevErrorGiven              SnmpAdminString,
      pktcMtaDevErrorReason             SnmpAdminString
}

pktcMtaDevErrorOidIndex  OBJECT-TYPE
```

```
      SYNTAX Integer32(1..1024)
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION
            "This is the index to pktcMtaDevErrorOidsEntry.
            This is an integer value and will start from the value 1
            and be incremented for each error encountered in the
            configuration file. The indices need not necessarily
            reflect the order of error occurrences in the
            configuration file."
      ::= { pktcMtaDevErrorOidsEntry  1 }

pktcMtaDevErrorOid  OBJECT-TYPE
      SYNTAX     SnmpAdminString
      MAX-ACCESS read-only
      STATUS     current
      DESCRIPTION
            "This is the OID associated with the particular error. If
            the error was not due to an identifiable OID, then this
            can be populated with impartial identifiers, in hexadecimal
             or numeric format."
      ::= { pktcMtaDevErrorOidsEntry  2 }

pktcMtaDevErrorGiven                OBJECT-TYPE
      SYNTAX      SnmpAdminString
      MAX-ACCESS  read-only
      STATUS      current
DESCRIPTION
            "If the error was due to the value associated
            with the corresponding pktcMtaDevErrorOid, then this
            contains the value of the OID as interpreted by the MTA in
            the configuration file provided. If the error was not due
            to the value of an OID this must be set to an  empty
            string. This is provided to eliminate errors due to
            misrepresentation/misinterpretation of data."
      ::= { pktcMtaDevErrorOidsEntry  3 }

pktcMtaDevErrorReason     OBJECT-TYPE
      SYNTAX SnmpAdminString
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION
            "This indicates the reason for the error,
            as per the MTAs interpretation, in human readable form.
            Example include:
            VALUE NOT IN RANGE,
            VALUE DOES NOT MATCH TYPE
            UNSUPPORTED VALUE
            LAST 4 BITS MUST BE SET TO ZERO,
            OUT OF MEMORY, CANNOT STORE etc.
            This MAY also contain vendor specific errors
            for vendor specific OIDS and any proprietary error
            codes/messages which can help diagnose errors
            better, in a manner the vendor deems fit."
      ::= { pktcMtaDevErrorOidsEntry  4 }

pktcMtaDevSwCurrentVers    OBJECT-TYPE
      SYNTAX SnmpAdminString
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION
           "This object identifies the software version currently
            operating in the MTA.
```

```
            The MTA MUST return a string descriptive of the current
            software load.  This object should use the syntax defined
            by the individual vendor to identify the software version.
            The data presented in this object MUST be identical with
            the software version information contained in the sysDescr
            MIB Object of the MTA.
            The value of this object MUST be identical to the value
            specified in DHCP option 43 sub-option 6."
       REFERENCE
            "PacketCable MTA Device Provisioning Specification;
              RFC 2132, DHCP Options and BOOTP Vendor Extensions"
      ::= { pktcMtaDevBase 14 }

-- The following group describes server access and parameters used for
-- initial provisioning and bootstrapping.
--
--*********************************************************************
--*************************This object is obsolete*********************
--*********************************************************************

pktcMtaDevServerBootState OBJECT-TYPE
      SYNTAX INTEGER {
            operational                 (1),
            disabled                    (2),
            waitingForDhcpOffer         (3),
            waitingForDhcpResponse      (4),
            waitingForConfig            (5),
            refusedByCmts               (6),
            other                       (7),
            unknown                     (8)
      }
      MAX-ACCESS  read-only
      STATUS      obsolete
      DESCRIPTION
            "If operational(1), the device has completed loading and
            processing of configuration parameters and the CMTS has
            completed the Registration exchange.
            If disabled(2) then the device was administratively
            disabled, possibly by being refused network access in the
            configuration file.
            If waitingForDhcpOffer(3) then a DHCP Discover has been
            transmitted and no offer has yet been received.
            If waitingForDhcpResponse(4) then a DHCP Request has been
            transmitted and no response has yet been received.
            If waitingForConfig(5) then a request to the config
            parameter server has been made and no response received.
            If refusedByCmts(6) then the Registration Request/Response
            exchange with the CMTS failed. "
      REFERENCE
            "DOCSIS Radio Frequency Interface Specification"
      ::= { pktcMtaDevServer 1 }

--*********************************************************************
--*************************This object is obsolete*********************
--*********************************************************************

pktcMtaDevServerDhcp OBJECT-TYPE
      SYNTAX      IpAddress
      MAX-ACCESS  read-only
      STATUS      obsolete
      DESCRIPTION
            "The IP address of the DHCP server that assigned an IP
            address to this device. Returns 0.0.0.0 if DHCP was not
```

```
            used for IP address assignment."
      ::= { pktcMtaDevServer 2 }
--
pktcMtaDevServerDns1  OBJECT-TYPE
      SYNTAX      IpAddress
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
            "The IP address of the primary DNS server to be used by the
            MTA to resolve the FQDNs and IP addresses."
      ::= { pktcMtaDevServer 3 }

pktcMtaDevServerDns2  OBJECT-TYPE
      SYNTAX      IpAddress
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
           "The IP address of the Secondary DNS server to be used by
            the MTA to resolve the FQDNs and IP addresses. Contains
            0.0.0.0 if there is no Secondary DNS server specified for the MTA
            for the MTA under consideration."
      ::= { pktcMtaDevServer 4 }

pktcMtaDevConfigFile  OBJECT-TYPE
      SYNTAX      SnmpAdminString
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
            "This object specifies the MTA device configuration file
             information, including the access method, the server
             name and the configuration file name. The value of this
             object is the Uniform Resource Locator (URL) of the
             configuration file for TFTP or HTTP download.
             If this object value is a TFTP URL, it must be formatted
             as defined in RFC 3617.
             If this object value is an HTTP URL, it must be formatted
             as defined in RFC 2616.
             If the MTA SNMP Enrollment mechanism is used, then the MTA
             must download the file provided by the Provisioning Server
             during provisioning via an SNMP SET on this object.
             If the MTA SNMP Enrollment mechanism is not used, this
             object MUST contain the URL value corresponding to the
             'siaddr' and 'file' fields received in the DHCP ACK to
             locate the configuration file:   the 'siaddr' & 'file'
             fields represents the host and file of the TFTP URL.
             In this case, the MTA MUST return an
             'inconsistentValue' error in response to SNMP SET
             operations.  The MTA MUST return a zero-length string if
             the server address (host part of the URL) is unknown."
      REFERENCE
             "RFC 3617, URI Scheme for TFTP; RFC 2616, HTTP 1.1"
      ::= { pktcMtaDevServer 5 }

pktcMtaDevSnmpEntity  OBJECT-TYPE
      SYNTAX      SnmpAdminString
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "This object contains the FQDN of the SNMP entity of the
            Provisioning Server. When the MTA SNMP Enrollment
            Mechanism is used, this object represents the server the
            MTA communicates with, to receive the configuration file
            URL from, and, to send the enrollment notification to.
```

```
            The SNMP entity is also the destination entity for all
            the provisioning notifications. It may be also used for
            post-provisioning SNMP operations.
            During the provisioning phase, this SNMP
            entity FQDN is supplied to the MTA via the DHCP option 122
            sub-option 3 as defined in RFC 3495."
      REFERENCE
             "PacketCable MTA Device Provisioning Specification;
             RFC 3495, DHCP Option for CableLabs Client Configuration."
      ::= { pktcMtaDevServer 6 }

pktcMtaDevProvConfigHash  OBJECT-TYPE
      SYNTAX      OCTET STRING (SIZE(16|20))
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
            "This object contains the hash value of the contents of
             the config file.
             If the authentication algorithm is MD5, the length is 128
             bits. If the authentication algorithm is SHA-1, the length
             is 160 bits. The hash calculation MUST follow
             the requirements defined in the PacketCable Security
             specification.
             When the MTA SNMP Enrollment mechanism is used, this
             hash value is calculated and sent to the MTA prior
             to sending the config file. This object value is then
             provided by the Provisioning server via an SNMP
             SET operation.
             When the MTA SNMP Enrollment mechanism is not in use, the
             hash value is provided in the configuration file itself
             and it is also calculated by the MTA. This object value
             MUST represent the hash value calculated by the MTA.
             When the MTA SNMP Enrollment mechanism is not in use, the
             MTA must reject all SNMP SET operations on this object and
             return an 'inconsistentValue' error."
       REFERENCE
           "PacketCable MTA Device Provisioning Specification;
            PacketCable Security Specification."
      ::= { pktcMtaDevServer 7 }

pktcMtaDevProvConfigKey  OBJECT-TYPE
      SYNTAX      OCTET STRING (SIZE(0|8))
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
            "This object contains the key used to encrypt/decrypt
             the configuration file when secure SNMPv3 provisioning
             is used.
             It is sent to the MTA prior to sending the config file.
             If the privacy algorithm is null, the length is 0.  If
             the privacy algorithm is DES, the length is 64 bits.
             This object must not be used in non secure provisioning
             mode.
             In non secure provisioning modes, the MTA MUST return an
             'inconsistentValue' in response to SNMP SET operations,
             and, the MTA MUST return a 'genErr' error in response to
             SNMP GET operations."
      ::= { pktcMtaDevServer 8 }

pktcMtaDevProvSolicitedKeyTimeout  OBJECT-TYPE
      SYNTAX      Integer32 (15..600)
      UNITS       "seconds"
      MAX-ACCESS  read-write
```

```
      STATUS      current
      DESCRIPTION
            "This object defines a Kerberos Key Management timer on the
             MTA. It is the time period during which the MTA saves the
             nonce and Server Kerberos Principal Identifier to match an
             AP Request and its associated AP Reply response from the
             Provisioning Server.
             After the timeout has been exceeded, the client discards
             this (nonce, Server Kerberos Principal Identifier) pair,
             after which it will no longer accept a matching AP Reply.
             This timer only applies when the Provisioning Server
             initiated key management for SNMPv3 (with a
             Wake Up message). This object should not be used in non
             secure provisioning modes. In non secure provisioning
             modes, the MTA MUST return an 'inconsistentValue' in
             response to SNMP SET operations,  and the MTA MUST
             return a 'genErr' error in response to SNMP GET
             operations."
      DEFVAL { 120 }
      ::= { pktcMtaDevServer 9 }


--=========================================================================
--
--  Unsolicited Key Updates are based on an exponential backoff
--  mechanism with two timers for AS replies.  The fast timers have a
--  maximum timer (pktcMtaDevProvUnsolicitedKeyMaxTimeout seconds) and
--  a nominal timer pktcMtaDevProvUnsolicitedKeyNomTimeout seconds)
--  from which the backoff timer determinations are made.
--
--=========================================================================

--=========================================================================
--
--  Timeouts for unsolicited key management updates are only pertinent
--  before the first SNMPv3 message is sent between the MTA and the
--  Provisioning server and before the configuration file is loaded.
--
--=========================================================================


pktcMtaDevProvUnsolicitedKeyMaxTimeout  OBJECT-TYPE

      SYNTAX       Integer32 (15..600)
      UNITS       "seconds"
      MAX-ACCESS   read-only
      STATUS       current
      DESCRIPTION
            "This object defines the timeout value that applies to
             an MTA-initiated AP-REQ/REP key management exchange with
             the Provisioning Server in SNMPv3 provisioning.
             It is the maximum timeout value and it may not be exceeded
             in the exponential back-off algorithm. If the DHCP option
             code 122 sub-option 5 is provided to the MTA, it overwrites
             this value.
             In non secure provisioning mode, the MTA MUST return
             a 'genErr' error in response to SNMP GET operations."
      REFERENCE
            "PacketCable Security Specification"
      DEFVAL  {600}
      ::= { pktcMtaDevServer 10 }

pktcMtaDevProvUnsolicitedKeyNomTimeout  OBJECT-TYPE
```

```
      SYNTAX       Integer32 (15..600)
      UNITS       "seconds"
      MAX-ACCESS   read-only
      STATUS       current
      DESCRIPTION
            "This object defines the starting value of the timeout
             for the AP-REQ/REP Backoff and Retry mechanism
             with exponential timeout in SNMPv3 provisioning.
             If the DHCP option code 122 sub-option 5 is provided
             the MTA, it overwrites this value.
             In non secure provisioning mode, the MTA MUST return
             a 'genErr' error in response to SNMP GET operations."
      REFERENCE
            "PacketCable Security Specification"
      DEFVAL  {30}
      ::= { pktcMtaDevServer 11 }
pktcMtaDevProvUnsolicitedKeyMeanDev OBJECT-TYPE
      SYNTAX      Integer32 (15..600)
      UNITS       "seconds"
      MAX-ACCESS  read-only
      STATUS      obsolete
      DESCRIPTION
            "This is the mean deviation for the round trip delay
             timings."
     REFERENCE
            "PacketCable Security Specification"
     ::= { pktcMtaDevServer 12 }

pktcMtaDevProvUnsolicitedKeyMaxRetries  OBJECT-TYPE

      SYNTAX      Integer32 (1..32)
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "This object contains a retry counter that applies to
            an MTA-initiated AP-REQ/REP key management exchange with
            the Provisioning Server in secure SNMPv3 provisioning.
            It is the maximum number of retries before the MTA stops
            attempting to establish a Security Association with
            Provisioning Server.
            If the DHCP option code 122 sub-option 5 is provided to
            the MTA, it overwrites this value.
            In non secure provisioning mode, the MTA MUST return
            a 'genErr' error in response to SNMP GET operations."
      REFERENCE
            "PacketCable Security Specification"
      DEFVAL  {8}
      ::= { pktcMtaDevServer 13 }

pktcMtaDevProvKerbRealmName  OBJECT-TYPE
      SYNTAX      SnmpAdminString (SIZE(1..255))
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
           "For Secure provisioning this object contains the name of
            the associated provisioning Kerberos realm acquired during
            the MTA4 provisioning step (DHCP Ack).
            Additionally this object value is used as an index
            into the pktcMtaDevRealmTable. In which case, the upper
            case ASCII representation of the associated Kerberos realm
            name MUST be used by both the Manager (SNMP entity)
            and the MTA.  The Kerberos realm name for the Provisioning
            Server is supplied to the MTA via DHCP option code 122
```

```
            sub-option 6 as defined in RFC 3495.
            For non secure provisioning modes, the value of
            this object MUST contain the value supplied in the DHCP
             ACK message (option code 122 sub-option 6)."
        REFERENCE
            "PacketCable MTA Device Provisioning Specification;
            RFC 3495, DHCP Option for CableLabs Client Configuration."
      ::= { pktcMtaDevServer 14 }

pktcMtaDevProvState  OBJECT-TYPE
      SYNTAX INTEGER {
            operational                 (1),
            waitingForSnmpSetInfo       (2),
            waitingForTftpAddrResponse  (3),
            waitingForConfigFile        (4)
      }
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION
           " This object defines the MTA provisioning state.
             If the state is:
               'operational(1)', the device has completed the loading
                and processing of the initialization parameters.

               'waitingForSnmpSetInfo(2)', the device is waiting on
                its configuration file download access information.
                Note that this state is only reported when the MTA
                SNMP enrollment mechanism is used.

               'waitingForTftpAddrResponse(3)', the device has sent a
                DNS request to resolve the server providing the
                configuration file and it is awaiting for a response.
                Note that this state is only reported when the MTA
                SNMP enrollment mechanism is used.

               'waitingForConfigFile(4)', the device has sent a
               request via TFTP or HTTP for the download of its
               configuration file and it is awaiting for a response or
               the file download is in progress."
      REFERENCE
            "PacketCable MTA Device Provisioning Specification,
             PacketCable Security Specification"
      ::= { pktcMtaDevServer 15 }

pktcMtaDevServerDhcp1   OBJECT-TYPE
      SYNTAX      IpAddress
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
"The IP address of the primary DHCP server which would cater to the
             MTA during its provisioning. Contains 255.255.255.255 if
             there was no preference given with respect to the DHCP
             servers for MTAprovisioning."
      ::= { pktcMtaDevServer 16 }

pktcMtaDevServerDhcp2  OBJECT-TYPE
      SYNTAX      IpAddress
      MAX-ACCESS  read-only
          STATUS      current
      DESCRIPTION      "The IP address of the Secondary DHCP server
             which could cater to the MTA during its provisioning.
             Contains 0.0.0.0 if there is no specific secondary DHCP
             server to be considered during MTA       provisioning."
```

```
    ::= { pktcMtaDevServer 17 }

pktcMtaDevTimeServer       OBJECT-TYPE
      SYNTAX      IpAddress
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
            "IP address of the Time Server from which to obtain the
            time. Contains 0.0.0.0 if the Time Protocol is not used for
            time synchronization."
      ::= { pktcMtaDevServer 18 }

--
-- The following group describes the security objects in the MTA
--

pktcMtaDevManufacturerCertificate  OBJECT-TYPE
      SYNTAX      X509Certificate
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
           " This object contains the MTA Manufacturer Certificate.
             The object value must be the ASN.1 DER encoding of the MTA
             manufacturer's X.509 public key certificate. The MTA
             Manufacturer Certificate is issued to each MTA
             manufacturer and is installed into each MTA at the time of
             manufacture or with a secure code download. The specific
             requirements related to this certificate are defined in
             the PacketCable Security specification."
       REFERENCE
           "PacketCable Security Specification."
      ::= {pktcMtaDevSecurity 1 }

pktcMtaDevCertificate  OBJECT-TYPE
      SYNTAX      X509Certificate
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "ASN.1 DER encoding of the MTA's X.509 public-key
            certificate issued by the manufacturer and installed
            into the embedded-MTA in the factory. This certificate,
            called MTA Device Certificate, contains the MTA's MAC
            address. It cannot be updated by the provisioning server."
      ::= { pktcMtaDevSecurity 2 }

--*********************************************************************
--************************* THIS OBJECT IS OBSOLETE *******************
--*********************************************************************
pktcMtaDevSignature  OBJECT-TYPE
      SYNTAX      OCTET STRING (SIZE (0..256))
      MAX-ACCESS  read-only
      STATUS      obsolete
      DESCRIPTION
            "A unique signature created by the MTA for each SNMP
            Inform or SNMP Trap or SNMP GetResponse message exchanged
            prior to enabling SNMPv3 security ASN.1 encoded Digital
            signature in the Cryptographic message syntax (includes
            nonce). "
      ::= { pktcMtaDevSecurity 3 }

pktcMtaDevCorrelationId  OBJECT-TYPE
      SYNTAX      Integer32
```

```
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "Random value generated by the MTA for use in registration
            authorization.  It is for use only in the MTA initialization
            messages and for MTA configuration file download "
      ::= { pktcMtaDevSecurity 4 }

--============================================================================
--
--  pktcMtaDevSecurityTable
--
--  The pktcMtaDevSecurityTable shows security association information
--  relating to a particular MTA endpoint.  The MTA endpoint is indexed
--  with ifIndex.
--
--============================================================================
--****************************************************************************
--************************** THIS TABLE IS OBSOLETE ******************
--****************************************************************************

pktcMtaDevSecurityTable OBJECT-TYPE
      SYNTAX      SEQUENCE OF PktcMtaDevSecurityEntry
      MAX-ACCESS  not-accessible
      STATUS      obsolete
          DESCRIPTION
            "Contains per endpoint security information."
      ::= { pktcMtaDevSecurity 5 }

pktcMtaDevSecurityEntry OBJECT-TYPE
      SYNTAX      PktcMtaDevSecurityEntry
      MAX-ACCESS  not-accessible
      STATUS      obsolete
      DESCRIPTION
            "List of security attributes for a single PacketCable
            endpoint interface."
      INDEX { ifIndex }
      ::= { pktcMtaDevSecurityTable 1 }

PktcMtaDevSecurityEntry ::= SEQUENCE {
      pktcMtaDevServProviderCertificate    X509Certificate,
      pktcMtaDevTelephonyCertificate       X509Certificate,
      pktcMtaDevKerberosRealm              OCTET STRING,
      pktcMtaDevKerbPrincipalName          DisplayString,
      pktcMtaDevServGracePeriod            Integer32,
      pktcMtaDevLocalSystemCertificate     X509Certificate,
      pktcMtaDevKeyMgmtTimeout1            Integer32,
      pktcMtaDevKeyMgmtTimeout2            Integer32
      }

pktcMtaDevServProviderCertificate OBJECT-TYPE
      SYNTAX      X509Certificate
      MAX-ACCESS  read-write
      STATUS      obsolete
      DESCRIPTION
            "ASN.1 DER encoding of the Telephony Service
            Provider's X.509 public-key certificate, called
            Telephony Service Provider Certificate. It serves
            as the root of the intra-domain trust hierarchy.
            Each MTA is configured with this certificate so
            that it can authenticate TGSs owned by the same
            service provider. The provisioning server needs
```

```
            the ability to update this certificate in the MTAs
            via both SNMP and configuration files"
      ::= { pktcMtaDevSecurityEntry 1 }

pktcMtaDevTelephonyCertificate OBJECT-TYPE
      SYNTAX      X509Certificate
      MAX-ACCESS  read-write
      STATUS      obsolete
      DESCRIPTION
            "ASN.1 DER encoding of the MTA's X.509 public-key
            certificate issued by the Service Provider with either
            the Service Provider CA or a Local System CA. This
            certificate, called MTA Telephony Certificate, contains
            the same public key as the MTA Device Certificate issued
            by the manufacturer. It is used to authenticate the
            identity of the MTA to the TGS (during PKINIT exchanges).
            The provisioning server needs the ability to update this
            certificate in the MTAs via both SNMP and configuration
            files"
      ::= { pktcMtaDevSecurityEntry 2 }

pktcMtaDevKerberosRealm OBJECT-TYPE
      SYNTAX      OCTET STRING (SIZE (0..1280))
      MAX-ACCESS  read-write
      STATUS      obsolete -- moved to realm table
      DESCRIPTION
            "Specifies a Kerberos realm (i.e. administrative domain),
            required for Packet Cable key management."
      ::= { pktcMtaDevSecurityEntry 3 }

pktcMtaDevKerbPrincipalName OBJECT-TYPE
      SYNTAX      DisplayString (SIZE(0..40))
      MAX-ACCESS  read-write
      STATUS      obsolete
      DESCRIPTION
            "Kerberos principal name for the Call Agent.  This
            information is required in order for the MTA to obtain
            Call Agent Kerberos tickets.  This principal name does not
            include the realm, which is specified as a separate field
            in this configuration file.  A Single Kerberos principal
            name MAY be shared among several Call Agents."
      ::= { pktcMtaDevSecurityEntry 4 }

pktcMtaDevServGracePeriod OBJECT-TYPE
      SYNTAX      Integer32 (15..600)
      UNITS       "minutes"
      MAX-ACCESS  read-write
      STATUS      obsolete       -- moved to realm table
      DESCRIPTION
            "The MTA MUST obtain a new Kerberos ticket (with a PKINIT
            exchange) this many minutes before the old ticket expires.
            The minimum allowable value is 15 mins.  The default is 30
            mins."
      DEFVAL { 30 }
      ::= { pktcMtaDevSecurityEntry 5 }

pktcMtaDevLocalSystemCertificate OBJECT-TYPE
      SYNTAX      X509Certificate
      MAX-ACCESS  read-write
      STATUS      obsolete
      DESCRIPTION
            "The Telephony Service Provider CA may delegate the
            issuance of certificates to a regional Certification
```

```
            Authority called Local System CA (with the corresponding
            Local System Certificate).  This parameter is the ASN.1
            DER encoding of the Local System Certificate.  It MUST have
            a non-empty value when the MTA Telephony certificate is
            signed by a Local System CA.  Otherwise, the value MUST
            be of length 0."
::= { pktcMtaDevSecurityEntry 6 }

pktcMtaDevKeyMgmtTimeout1 OBJECT-TYPE
      SYNTAX      Integer32 (15..600)
      UNITS       "seconds"
      MAX-ACCESS  read-write
      STATUS      obsolete -- moved to cms table
      DESCRIPTION
            "This timeout applies only when the MTA initiated key
            management.  It is the period during which the MTA will
            save a nonce (inside the sequence number field) from the
            sent out AP Request and wait for the matching AP Reply
            from the CMS."
    REFERENCE
            "PacketCable Security Specification"
      ::= { pktcMtaDevSecurityEntry 7 }

pktcMtaDevKeyMgmtTimeout2 OBJECT-TYPE
      SYNTAX      Integer32 (15..600)
      UNITS       "seconds"
      MAX-ACCESS  read-write
      STATUS      obsolete -- changed to adaptive backoff and moved to
                  -- cms table
      DESCRIPTION
            "This timeout applies only when the CMS initiated key
             management (with a Wake Up or Rekey message).
             It is the period during which the MTA will
             save a nonce (inside the sequence number  field) from
             the sent out AP Request and wait for the matching AP
             Reply from the CMS."
    REFERENCE
            "PacketCable Security Specification"
      ::= { pktcMtaDevSecurityEntry 8 }

--
--            Ticket Granting Server information
--
--*********************************************************************
--************************* THIS TABLE IS OBSOLETE ******************
--*********************************************************************

pktcMtaDevTgsTable OBJECT-TYPE
      SYNTAX      SEQUENCE OF PktcMtaDevTgsEntry
      MAX-ACCESS  not-accessible
      STATUS      obsolete -- Secure Provisioning ECR
      DESCRIPTION
            "Contains per endpoint Ticket Granting Server information."
      ::= {  pktcMtaDevSecurity 8 }
pktcMtaDevTgsEntry OBJECT-TYPE
      SYNTAX      PktcMtaDevTgsEntry
      MAX-ACCESS  not-accessible
      STATUS      obsolete -- Secure Provisioning ECR
      DESCRIPTION
            "List of Tgs attributes for a single packet cable
            endpoint interface."
      INDEX { ifIndex, pktcMtaDevTgsIndex }
      ::= { pktcMtaDevTgsTable 1 }
```

```
PktcMtaDevTgsEntry ::= SEQUENCE {
      pktcMtaDevTgsIndex      Integer32,
      pktcMtaDevTgsLocation   DisplayString,
      pktcMtaDevTgsStatus     RowStatus
      }

pktcMtaDevTgsIndex OBJECT-TYPE
      SYNTAX      Integer32 (1..2147483647)
      MAX-ACCESS  not-accessible
      STATUS      obsolete -- Secure Provisioning ECR
      DESCRIPTION
            "Index into the TGS table for TGS locations.
            IfType specifies the endpoint, TgsIndex specifies a TGS."
       ::= { pktcMtaDevTgsEntry 1 }
pktcMtaDevTgsLocation OBJECT-TYPE
      SYNTAX      DisplayString (SIZE (0..255))
      MAX-ACCESS  read-create
      STATUS      obsolete -- Secure Provisioning ECR
      DESCRIPTION
            "Name of the TGS Ticket Granting Server, which is the
            Kerberos Server.  This parameter is a FQDN or Ipv4 address.
            There may be multiple entries of this type.  The order
            in which these entries are listed is the priority order
            in which the MTA will attempt to contact them for this
            endpoint."
      ::= { pktcMtaDevTgsEntry 2 }

pktcMtaDevTgsStatus    OBJECT-TYPE
      SYNTAX      RowStatus
      MAX-ACCESS  read-create
      STATUS      obsolete       -- Secure Provisioning ECR
      DESCRIPTION
            "This object contains the Row Status associated with
            the pktcMtaDevTgsTable."
::= { pktcMtaDevTgsEntry 3 }

pktcMtaDevTelephonyRootCertificate  OBJECT-TYPE
      SYNTAX      X509Certificate
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
            "ASN.1 DER encoding of the IP Telephony Root X.509
            public-key certificate stored in the MTA non-volatile
            memory and updateable with a code download.  This
            certificate is used to validate the initial AS Reply
            from the KDC received during the MTA initialization."
      ::= { pktcMtaDevSecurity 9 }

--=====================================================================
--
--     Procedures for setting up security associations:
--
--     A security association may be setup either via configuration or via
--     NCS signaling.
--
--       I.     Security association setup via configuration.
--
--           The realm must be configured first.  Associated with the
--           realm is a KDC.  The realm table (pktcMtaDevRealmTable)
--           indicates information about realm (e.g., name,
--           organization name) and parameters associated with KDC
--           communications (e.g., grace periods,  AS request/AS
```

```
--           reply adaptive backoff parameters).

--           Once the realm is established, one or more servers may be
--           defined in the realm.  For PacketCable, these are
--           Call Management Servers (CMSs). Associated with each CMS
--           entry in the pktcMtaDevCmsTable is an explicit reference
--           to a Realm via the realm index
--           (pktcMtaDevCmsKerbRealmName), the FQDN of the CMS,
--           and parameters associated with IPSec management with the
--           CMS (e.g., clock skew, AP request/
--           AP reply adaptive backoff parameters).
--


--
--
--       II.    Security association setup via NCS signaling
--

--           Note:  The following process is done automatically by the
--           MTA.  The NCS is not involved in creating signaled entries.
--           The current CMS signaling association being used by an
--           endpoint is marked as active in CMS MAP table.  If NCS
--           signaling requests a change of signaling association to
--           a different FQDN, the MTA checks the current CMS MAP
--           table entries for the affected endpoint.  If the entry
--           exists in the CMS MAP table, the current CMS MAP table
--           entry is marked inactive and the newly chosen CMS MAP
--           table entry is marked active.
--
--           If the entry does not exist in the CMS MAP table, the
--           CMS table is checked to determine whether or not it
--           contains the CMS specified by CMS signaling (possibly
--           a redirection).  If the desired CMS entry is defined,
--           then a corresponding entry is created and an entry in
--           the CMS MAP table is created.  If the MTA does not
--           have current associations with that CMS, it will now
--           perform key management to establish required security
--           associations. Once the desired CMS entry is established,
--           the current CMS MAP table entry is marked inactive and
--           the newly created CMS MAP table entry is marked active.
--           Otherwise the current CMS MAP table entry remains
--           active and the newly created CMS MAP table entry is marked
--           in active.
--
--           If the entry does not exist in the CMS MAP table and the
--           CMS entry does not exist in the CMS table, a new CMS
--           table entry should be created.  This CMS entry should use
--           the same realm as used by this endpoint. The default
--           values for the clock skew and AP request/AP reply adaptive
--           backoff parameters should be used.  The MTA will now
--           perform key management to establish required security
--           associations. Once the desired CMS entry is established,
--           the current CMS MAP table entry is marked inactive and
--           the newly created CMS MAP table entry is marked active.
--           Otherwise the current CMS MAP table entry remains
--           active and the newly created CMS MAP table entry is
--           marked inactive.
--
--      III.   When the MTA receives wake-up or rekey messages from a CMS,
--           it performs key management based on the corresponding entry
--           in the CMS table.  If the matching CMS entry does not exist,
--           it must ignore the wake-up or rekey messages.
--
```

```
--========================================================================



--========================================================================
--
--     pktcMtaDevRealmTable
--
--  The pktcMtaDevRealmTable shows the KDC realms.  The table is
--  indexed withpktcMtaDevRealmName.  The Realm Table is used in with
--  conjunction any server which needs a security association with an
--  server MTA.  The table (today the CMS) has a security association.
--  Each server-MTA security association is associated with a
--  single Realm.  This allows for multiple realms, each
--  with its own security association.
--
--========================================================================

pktcMtaDevRealmTable  OBJECT-TYPE
      SYNTAX      SEQUENCE OF PktcMtaDevRealmEntry
      MAX-ACCESS  not-accessible
      STATUS      current
      DESCRIPTION
            "Contains per Kerberos realm security parameters."
      ::= { pktcMtaDevSecurity 16 }

pktcMtaDevRealmEntry  OBJECT-TYPE
      SYNTAX      PktcMtaDevRealmEntry
      MAX-ACCESS  not-accessible
      STATUS      current
      DESCRIPTION
            "List of security parameters for a single Kerberos realm."
      INDEX { IMPLIED pktcMtaDevRealmName }
      ::= { pktcMtaDevRealmTable 1 }

PktcMtaDevRealmEntry ::= SEQUENCE {
      pktcMtaDevRealmName                        SnmpAdminString,
      pktcMtaDevRealmPkinitGracePeriod           Integer32,
      pktcMtaDevRealmTgsGracePeriod              Integer32,
      pktcMtaDevRealmOrgName                     OCTET STRING,
      pktcMtaDevRealmUnsolicitedKeyMaxTimeout    Integer32,
      pktcMtaDevRealmUnsolicitedKeyNomTimeout    Integer32,
      pktcMtaDevRealmUnsolicitedKeyMeanDev       Integer32,
      pktcMtaDevRealmUnsolicitedKeyMaxRetries    Integer32,
      pktcMtaDevRealmStatus                      RowStatus
      }

pktcMtaDevRealmName  OBJECT-TYPE
      SYNTAX      SnmpAdminString(SIZE(1..255))
      MAX-ACCESS  not-accessible
      STATUS      current
      DESCRIPTION
            "The corresponding  Kerberos Realm name. This is used as
            an index into pktcMtaDevRealmTable. When used as an index,
            used by both the Manager(SNMPv3 Entity)  and the MTA."
      ::= { pktcMtaDevRealmEntry 1 }

pktcMtaDevRealmPkinitGracePeriod  OBJECT-TYPE
      SYNTAX      Integer32 (15..600)
      UNITS       "minutes"
      MAX-ACCESS  read-create
      STATUS      current
```

```
      DESCRIPTION
            "For the purposes of the key management with an Application
            Server (CMS or Provisioning Server), the MTA MUST obtain a
            new Kerberos ticket (with a PKINIT exchange) this many
            minutes before the old ticket expires. The minimum
            allowable value is 15 mins. The default is 30 mins. This
            parameter MAY also be used with other Kerberized
            applications."
      DEFVAL { 30 }
      ::= { pktcMtaDevRealmEntry 2 }

pktcMtaDevRealmTgsGracePeriod  OBJECT-TYPE
      SYNTAX      Integer32 (1..600)
      UNITS       "minutes"
      MAX-ACCESS  read-create
      STATUS      current
      DESCRIPTION
            "When the MTA implementation uses TGS Request/TGS Reply
            Kerberos messages for the purpose of the key management
            with an Application Server (CMS or Provisioning Server),
            the MTA MUST obtain a new service ticket for the
            Application Server (with a TGS Request) this many minutes
            before the old ticket expires. The minimum allowable value
            is 1 min. The default is 10 mins. This parameter MAY also
            be used with other Kerberized applications."
      DEFVAL { 10 }
      ::= { pktcMtaDevRealmEntry 3 }

pktcMtaDevRealmOrgName  OBJECT-TYPE
      SYNTAX      OCTET STRING (SIZE (1..64))
      MAX-ACCESS  read-create
      STATUS      current
      DESCRIPTION
            "The value of the X.500 organization name attribute in the
            subject name of the Service provider certificate"
      ::= { pktcMtaDevRealmEntry 4 }

--============================================================================
--
--  Unsolicited Key Updates are based on an exponential backoff
--  mechanism with two timers for AS replies.  The backoff timers has a
--  maximum value of pktcMtaDevRealmUnsolicitedKeyMaxTimeout seconds
--  and a nominal timer has a pktcMtaDevRealmUnsolicitedKeyNomTimeout
--  seconds from which the backoff timer determinations are made.
--  After pktcMatDevRealmUnsolicitedMaxRetries have occurred no more
--  attempts are made.
--
--============================================================================

pktcMtaDevRealmUnsolicitedKeyMaxTimeout  OBJECT-TYPE
      SYNTAX Integer32 (1..600)
      UNITS "seconds"
      MAX-ACCESS read-create
      STATUS current
      DESCRIPTION
            "This timeout applies only when the MTA initiated key
            management. The maximum timeout is the value which may not
            be exceeded in the exponential backoff algorithm. If
             provided, DHCP-Option-122-Sub-option 4 overrides this value."
      REFERENCE
            "PacketCable Security Specification"
      DEFVAL { 30 }
      ::= { pktcMtaDevRealmEntry 5 }
```

```
pktcMtaDevRealmUnsolicitedKeyNomTimeout  OBJECT-TYPE
      SYNTAX       Integer32 (100..600000)
      UNITS        "milliseconds"
      MAX-ACCESS   read-create
      STATUS       current
      DESCRIPTION
"Defines the starting value of the timeout for the AS-REQ/REP Backoff
            and  Retry mechanism with exponential timeout. If
           provided, DHCP-Option-122-Sub-option 4 override this
           value."
      REFERENCE
            "PacketCable Security Specification,
            PacketCable Provisioning Specification"
      DEFVAL { 10000 }
      ::= { pktcMtaDevRealmEntry 6 }
pktcMtaDevRealmUnsolicitedKeyMeanDev OBJECT-TYPE
      SYNTAX      Integer32 (1..600)
      UNITS       "seconds"
      MAX-ACCESS  read-only
      STATUS      obsolete
      DESCRIPTION
            "This is measurement of the mean deviation for the round
            trip delay timings."
      REFERENCE
            "PacketCable Security Specification"
      DEFVAL { 2 }
      ::= { pktcMtaDevRealmEntry 7 }

pktcMtaDevRealmUnsolicitedKeyMaxRetries  OBJECT-TYPE
      SYNTAX Integer32 (0..1024)
      MAX-ACCESS read-create
      STATUS current
      DESCRIPTION
            "This is the maximum number of retries before the MTA
            gives up attempting to establish a security association.
            If provided,DHCP-Option-122-Sub-option 4 overrides this
            value."
      REFERENCE
            "PacketCable Security Specification"
      DEFVAL { 5 }
      ::= { pktcMtaDevRealmEntry 8 }

pktcMtaDevRealmStatus     OBJECT-TYPE
      SYNTAX      RowStatus
      MAX-ACCESS  read-create
      STATUS      current
      DESCRIPTION
            "This object contains the Row Status associated with
            the pktcMtaDevRealmTable."
      ::= { pktcMtaDevRealmEntry 9 }

--============================================================================
--
--  pktcMtaDevCmsTable
--
-- The pktcMtaDevCmsTable shows the IPSec key management policy
-- relating to a particular CMS.  The table is indexed with
-- pktcMtaDevCmsFQDN.
--
--============================================================================

pktcMtaDevCmsTable  OBJECT-TYPE
      SYNTAX      SEQUENCE OF PktcMtaDevCmsEntry
```

```
      MAX-ACCESS  not-accessible
      STATUS      current
      DESCRIPTION
            "Contains per CMS key management policy."
      ::= { pktcMtaDevSecurity 17 }

pktcMtaDevCmsEntry  OBJECT-TYPE
    SYNTAX      PktcMtaDevCmsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
"List of key management parameters for a single MTA-CMS
interface."
    INDEX { IMPLIED pktcMtaDevCmsFqdn }
    ::= { pktcMtaDevCmsTable 1 }

PktcMtaDevCmsEntry ::= SEQUENCE {
      pktcMtaDevCmsFqdn                       SnmpAdminString,
      pktcMtaDevCmsKerbRealmName              SnmpAdminString,
      pktcMtaDevCmsSolicitedKeyTimeout        Integer32,
      pktcMtaDevCmsMaxClockSkew               Integer32,
      pktcMtaDevCmsUnsolicitedKeyMaxTimeout   Integer32,
      pktcMtaDevCmsUnsolicitedKeyNomTimeout   Integer32,
      pktcMtaDevCmsUnsolicitedKeyMeanDev      Integer32,
      pktcMtaDevCmsUnsolicitedKeyMaxRetries   Integer32,
      pktcMtaDevCmsStatus                     RowStatus,
      pktcMtaDevCmsIpsecCtrl                  TruthValue
      }

pktcMtaDevCmsFqdn  OBJECT-TYPE
      SYNTAX      SnmpAdminString (SIZE(1..255))
      MAX-ACCESS  not-accessible
      STATUS      current
      DESCRIPTION
            "The fully qualified domain name of the CMS.
            is the index into the pktcMtaDevCmsTable.
            When used as an index, the upper case ASCII
            representation of the associated CMS FQDN
            MUST be used by both the SNMP Manager and the MTA."
      ::= { pktcMtaDevCmsEntry 1 }

pktcMtaDevCmsKerbRealmName  OBJECT-TYPE
      SYNTAX      SnmpAdminString (SIZE(1..255))
      MAX-ACCESS  read-create
      STATUS      current
      DESCRIPTION
            "The Kerberos Realm Name of the associated CMS. This is
            the index into the pktcMtaDevRealmTable.
            When used as an index, the upper case ASCII
            representation of the associated CMS FQDN
            must be used by both the SNMP Manager and the MTA "
      ::= { pktcMtaDevCmsEntry 2 }

pktcMtaDevCmsMaxClockSkew  OBJECT-TYPE
      SYNTAX     Integer32 (1..1800)
      UNITS      "seconds"
      MAX-ACCESS  read-create
      STATUS      current
      DESCRIPTION
            "This is the maximum allowable clock skew between the
            MTA and CMS"
      DEFVAL { 300 }
      ::= { pktcMtaDevCmsEntry 3 }
```

```
pktcMtaDevCmsSolicitedKeyTimeout  OBJECT-TYPE
      SYNTAX Integer32 (100..30000)
      UNITS "milliseconds"
      MAX-ACCESS read-create
      STATUS current
      DESCRIPTION
            "This timeout applies only when the CMS initiated key
             management(with a Wake Up or Rekey message). It is the
             period during which the MTA will save a nonce (inside the
             sequence number field) from the sent out AP Request and
             wait for the matching AP Reply from the CMS."
      REFERENCE
            "PacketCable Security Specification"
      DEFVAL { 1000 }
      ::= { pktcMtaDevCmsEntry 4 }
--=============================================================================
--
--  Unsolicited Key Updates are based on an exponential backoff
--  mechanism with mechanism with two timers for AP replies.  The
--  backoff timers have a maximum value of
--  pktcMtaDevCmsUnsolicitedKeyMaxTimeout
--  seconds and a nominal timer has
--  pktcMtaDevCmsUnsolicitedKeyNomTimeout seconds from which the
--  backoff timer determinations are made.  After
--  pktcMatDevCmsUnsolicitedMaxRetries have occurred no more
--  attempts are made.
--
--=============================================================================

pktcMtaDevCmsUnsolicitedKeyMaxTimeout  OBJECT-TYPE

      SYNTAX      Integer32 (1..600)
      UNITS       "seconds"
      MAX-ACCESS  read-create
      STATUS current
      DESCRIPTION
            "This timeout applies only when the MTA initiated key.
             The maximum management timeout is the value which may not
             be exceeded in the exponential backoff algorithm."
      REFERENCE
            "PacketCable Security Specification"
      DEFVAL { 8 }
      ::= { pktcMtaDevCmsEntry 5 }


pktcMtaDevCmsUnsolicitedKeyNomTimeout  OBJECT-TYPE
      SYNTAX Integer32 (100..30000)
      UNITS "milliseconds"
      MAX-ACCESS read-create
      STATUS current
      DESCRIPTION
            "Defines the starting value of the timeout for the
            AP-REQ/REP Backoff and Retry mechanism with exponential
            timeout for CMS."
      REFERENCE
            "PacketCable Security Specification"
      DEFVAL { 500 }
      ::= { pktcMtaDevCmsEntry 6 }

pktcMtaDevCmsUnsolicitedKeyMeanDev OBJECT-TYPE
      SYNTAX      Integer32 (1..600)
      UNITS       "seconds"
```

```
      MAX-ACCESS  read-only
      STATUS      obsolete
      DESCRIPTION
           "This is the measurement of the mean deviation for the
           round trip delay timings."
      REFERENCE
            "PacketCable Security Specification"
      ::= { pktcMtaDevCmsEntry 7 }

pktcMtaDevCmsUnsolicitedKeyMaxRetries  OBJECT-TYPE

      SYNTAX Integer32 (0..1024)
      MAX-ACCESS read-create
      STATUS current
      DESCRIPTION
            "This is the maximum number of retries before the MTA
            gives up attempting to establish a security association."
      REFERENCE
            "PacketCable Security Specification"
      DEFVAL { 5 }
      ::= { pktcMtaDevCmsEntry 8 }

pktcMtaDevCmsStatus     OBJECT-TYPE

      SYNTAX      RowStatus
      MAX-ACCESS  read-create
      STATUS      current
      DESCRIPTION
            "This object contains the Row Status associated with the
            pktcMtaDevCmsTable."
      ::= { pktcMtaDevCmsEntry 9 }

pktcMtaDevCmsIpsecCtrl   OBJECT-TYPE
      SYNTAX            TruthValue
      MAX-ACCESS        read-only
      STATUS current
      DESCRIPTION
            "The value of 'true(1)' indicates that IPSEC and IPSEC
            KeyManagement MUST be used to communicate with the CMS.
            The value of 'false(2)' indicates that IPSEC Signaling
            Security is disabled for both the IPSEC Key Management and
            IPSECprotocol (for the specific CMS)."
      DEFVAL { true }
      ::= { pktcMtaDevCmsEntry 10 }



--============================================================================
--
--     pktcMtaCmsMapTable
--*** this table is obsolete ***
--
--
--  The pktcMtaCmsMapTable contains the signaling associations
--  between MTA endpoints and CMSs.  It maps the endpoint to
--  zero or more entries in pktcMtaDevCmsTable.
--
--  The table contains the following indexes and rows:
--
--  ifIndex -the index of the physical port
--
--  pktcMtaCmsMapCmsIndex - the index of the CMS entry in the
--  pktcMtaDevCmsTable.  Valid indices are equal to current
```

```
--  pktcMtaDevCmsIndex values.
--
--  pktcMtaCmsMapOperStatus -    this value indicates which signaling
--  association the endpoint is actively using
--
--  pktcMtaCmsMapAdminStatus -   this flag indicates whether or not
--  an endpoint should use a particular CMS and its security
--  association.  By setting this flag to inhibit, this associated
--  CMS cannot provide signaling to the referenced endpoint.
--
--  pktcMtaCmsMapRowStatus - allows for the creation and deletion of
--  endpoint mappings via the NMS
--
--
--============================================================================

pktcMtaCmsMapTable OBJECT-TYPE
      SYNTAX      SEQUENCE OF PktcMtaCmsMapEntry
      MAX-ACCESS  not-accessible
      STATUS      obsolete
      DESCRIPTION
            "Contains per endpoint CMS signaling associations."
      ::= { pktcMtaDevSecurity 18 }

pktcMtaCmsMapEntry OBJECT-TYPE
      SYNTAX      PktcMtaCmsMapEntry
      MAX-ACCESS  not-accessible
      STATUS      obsolete
      DESCRIPTION
            "List of signaling associations."
      INDEX { ifIndex, pktcMtaCmsMapCmsFqdn }
      ::= { pktcMtaCmsMapTable 1 }

PktcMtaCmsMapEntry ::= SEQUENCE {
      pktcMtaCmsMapCmsFqdn DisplayString,
      pktcMtaCmsMapOperStatus     INTEGER,
      pktcMtaCmsMapAdminStatus  INTEGER,
      pktcMtaCmsMapRowStatus  RowStatus
      }

pktcMtaCmsMapCmsFqdn OBJECT-TYPE
      SYNTAX      DisplayString (SIZE(1..255))
      MAX-ACCESS  not-accessible
      STATUS      obsolete
      DESCRIPTION
            "The index for the associated CMS.  Valid indices
            are equal to current pktcMtaDevCmsFqdn values."
      ::= { pktcMtaCmsMapEntry 1 }
pktcMtaCmsMapOperStatus OBJECT-TYPE
      SYNTAX  INTEGER {
            inactive      (1),
            active        (2)
      }
      MAX-ACCESS  read-only
      STATUS      obsolete
      DESCRIPTION
            "The operational status of signaling association.  The
            meaning of the  status is as follows:
            inactive - signaling is not currently active
            active - signaling is active."
::= { pktcMtaCmsMapEntry 2 }
pktcMtaCmsMapAdminStatus OBJECT-TYPE
      SYNTAX      INTEGER {
```

```
      inhibit  (1),
      allow    (2)
      }
      MAX-ACCESS  read-create
      STATUS      obsolete
      DESCRIPTION
            "The administrative status for signaling over the indicated
            security association.  The meaning of the status is as
            follows:
            inhibit -signaling is not currently allowed
            allow - signaling is allowed."
      ::= { pktcMtaCmsMapEntry 3 }
pktcMtaCmsMapRowStatus OBJECT-TYPE
      SYNTAX      RowStatus
      MAX-ACCESS  read-create
      STATUS      obsolete
      DESCRIPTION
            "This object is used for creating and deleting an entry
            in this table via an element manager."
      ::= { pktcMtaCmsMapEntry 4 }

pktcMtaDevResetKrbTickets  OBJECT-TYPE
      SYNTAX   BITS {
            invalidateProvOnReboot (0),
            invalidateAllCmsOnReboot (1)
      }
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
            "This object defines a Kerberos Ticket Control Mask that
             instructs the MTA to invalidate the specific Application
             Server Kerberos Ticket(s) that are stored locally in the
             MTA NVRAM (non-volatile or persistent memory).
             If the MTA does not store Kerberos tickets in NVRAM, it
             MUST ignore setting of this object, and MUST report a BITS
             value of zero when the object is read.
             If the MTA supports Kerberos tickets storage in NVRAM, the
             object value is encoded as follows:
             - Setting the invalidateProvOnReboot bit (bit 0) to 1
               means that the MTA MUST invalidate the Kerberos
               Application Ticket(s) for the Provisioning Application
               at the next MTA reboot (if secure SNMP provisioning mode
               is used).  In non secure provisioning modes, the MTA MUST
               return an 'inconsistentValue' in response to SNMP SET
               operations with a bit 0 set to 1.
             - Setting the invalidateAllCmsOnReboot bit (bit 1) to 1
               means that the MTA MUST invalidate the Kerberos
               Application Ticket(s) for all CMSes currently assigned
               to the MTA endpoints."
     REFERENCE
              "PacketCable Security Specification"
DEFVAL {{  }}
::= { pktcMtaDevSecurity 19 }

 --
--  notification group is for future extension.
--
pktcMtaNotificationPrefix OBJECT IDENTIFIER ::= { pktcMtaMib 2 }
pktcMtaNotification OBJECT IDENTIFIER ::= {
 pktcMtaNotificationPrefix 0 }
pktcMtaConformance  OBJECT IDENTIFIER ::= { pktcMtaMib 3 }
pktcMtaCompliances  OBJECT IDENTIFIER ::= { pktcMtaConformance 1 }
pktcMtaGroups       OBJECT IDENTIFIER ::= { pktcMtaConformance 2 }
```

```
--
--    Notification Group
--
pktcMtaDevProvisioningEnrollment   NOTIFICATION-TYPE
      OBJECTS {
             sysDescr,
             pktcMtaDevSwCurrentVers,
             pktcMtaDevTypeIdentifier,
             pktcMtaDevMacAddress,
             pktcMtaDevCorrelationId
      }
      STATUS   current
      DESCRIPTION
            "This INFORM notification is issued by the MTA to initiate
             the PacketCable provisioning process when the MTA SNMP
             enrollment mechanism is used.
             It contains the system description, the current software
             version, the MTA device type identifier, the MTA MAC
             address (obtained in the MTA ifTable in the ifPhysAddress
             object that corresponds to the ifIndex 1) and a
             correlation ID."
     ::= { pktcMtaNotification 1 }
pktcMtaDevProvisioningStatus   NOTIFICATION-TYPE
      OBJECTS {
        pktcMtaDevMacAddress,
        pktcMtaDevCorrelationId,
        pktcMtaDevProvisioningState
}
      STATUS      current
      DESCRIPTION
            "This INFORM notification may be issued by the MTA to
             confirm the completion of the PacketCable provisioning
             process, and to report its provisioning completion
             status.
             It contains the MTA MAC address (obtained in the MTA
             ifTable in the ifPhysAddress object that corresponds
             to the ifIndex 1), a correlation ID and the MTA
             provisioning state as defined in
             pktcMtaDevProvisioningState."
      ::= { pktcMtaNotification 2 }

--  compliance statements
pktcMtaBasicCompliance MODULE-COMPLIANCE
      STATUS     current
      DESCRIPTION
            "The compliance statement for devices that implement
            MTA feature."
      MODULE   --pktcMtaMib
--  unconditionally mandatory groups
      MANDATORY-GROUPS {
      pktcMtaGroup,
      pktcMtaNotificationGroup
          }
      ::= { pktcMtaCompliances 3 }
pktcMtaGroup OBJECT-GROUP
      OBJECTS {            pktcMtaDevResetNow,
            pktcMtaDevSerialNumber,
            pktcMtaDevMacAddress,
            pktcMtaDevFQDN,
            pktcMtaDevEndPntCount,
            pktcMtaDevEnabled,
            pktcMtaDevTypeIdentifier,
            pktcMtaDevProvisioningState,
```

```
            pktcMtaDevHttpAccess,
            pktcMtaDevCertificate,
            pktcMtaDevCorrelationId,
            pktcMtaDevManufacturerCertificate,
            pktcMtaDevServerDhcp1,
            pktcMtaDevServerDhcp2,
            pktcMtaDevServerDns1,
            pktcMtaDevServerDns2,
            pktcMtaDevTimeServer,
            pktcMtaDevConfigFile,
            pktcMtaDevSnmpEntity,
            pktcMtaDevRealmPkinitGracePeriod,
            pktcMtaDevRealmTgsGracePeriod,
            pktcMtaDevRealmOrgName,
            pktcMtaDevRealmUnsolicitedKeyMaxTimeout,
            pktcMtaDevRealmUnsolicitedKeyNomTimeout,
            pktcMtaDevRealmUnsolicitedKeyMaxRetries,
            pktcMtaDevRealmStatus,
            pktcMtaDevCmsKerbRealmName,
            pktcMtaDevCmsUnsolicitedKeyMaxTimeout,
            pktcMtaDevCmsUnsolicitedKeyNomTimeout,
            pktcMtaDevCmsUnsolicitedKeyMaxRetries,
            pktcMtaDevCmsSolicitedKeyTimeout,
            pktcMtaDevCmsMaxClockSkew,
            pktcMtaDevCmsStatus,
            pktcMtaDevProvUnsolicitedKeyMaxTimeout,
            pktcMtaDevProvUnsolicitedKeyNomTimeout,
            pktcMtaDevProvUnsolicitedKeyMaxRetries,
            pktcMtaDevProvKerbRealmName,
            pktcMtaDevProvSolicitedKeyTimeout,
            pktcMtaDevProvConfigHash,
            pktcMtaDevProvConfigKey,
            pktcMtaDevProvState,
            pktcMtaDevProvisioningTimer,
            pktcMtaDevTelephonyRootCertificate,
            pktcMtaDevErrorOid,
            pktcMtaDevErrorGiven,
            pktcMtaDevErrorReason,
            pktcMtaDevSwCurrentVers,
            pktcMtaDevResetKrbTickets,
            pktcMtaDevCmsIpsecCtrl,
            pktcMtaDevProvisioningCounter
      }
      STATUS    current
      DESCRIPTION
            "Group of objects for PacketCable MTA MIB."
      ::= { pktcMtaGroups 1 }
pktcMtaNotificationGroup NOTIFICATION-GROUP
      NOTIFICATIONS {
            pktcMtaDevProvisioningStatus,
            pktcMtaDevProvisioningEnrollment
      }
      STATUS current
      DESCRIPTION
            "These notifications deal with change in status of
            MTA Device."
      ::= { pktcMtaGroups 2 }

pktcMtaObsoleteGroup       OBJECT-GROUP
      OBJECTS {
            pktcMtaDevHardwareVersion,
            pktcMtaDevSignature,
            pktcMtaDevServProviderCertificate,
```

```
            pktcMtaDevTelephonyCertificate,
            pktcMtaDevKerberosRealm,
            pktcMtaDevKerbPrincipalName,
            pktcMtaDevServGracePeriod,
            pktcMtaDevLocalSystemCertificate,
            pktcMtaDevKeyMgmtTimeout1,
            pktcMtaDevTgsLocation,
            pktcMtaDevTgsStatus,
            pktcMtaDevServerBootState,
            pktcMtaCmsMapOperStatus,
            pktcMtaCmsMapAdminStatus,
            pktcMtaCmsMapRowStatus,
            pktcMtaDevRealmUnsolicitedKeyMeanDev,
            pktcMtaDevCmsUnsolicitedKeyMeanDev,
            pktcMtaDevProvUnsolicitedKeyMeanDev,
            pktcMtaDevServerDhcp,
            pktcMtaDevKeyMgmtTimeout2
      }
      STATUS  obsolete
      DESCRIPTION
            "Group of obsolete objects for PacketCable MTA MIB."
      ::= { pktcMtaGroups 3}

END
```

Appendix A. Acknowledgements

On behalf of CableLabs and its participating member companies, we would like to extend our thanks to all those who contributed to the development of this specification. Certainly all the participants of the provisioning focus team have added value to this effort by participating in the review and weekly conference calls. Particular thanks are given to:

Angela Lyda (Arris Interactive)
Chris Melle (AT&T Broadband Labs)
Sasha Medvinsky (Motorola, Inc.)
Roy Spitzer (Telogy Networks, Inc.)
Rick Vetter (Motorola, Inc.)
Eugene Nechamkin (BroadCom Corp.)
Satish Kumar (Texas Instruments)

Jean-Francois Mule, Sumanth Channabasappa, Venkatesh Sunkad (CableLabs, Inc.)