Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.html

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it defines a basic set of managed objects for Simple Network Management Protocol (SNMP)-based management of events that can be generated by PacketCable- and IPCablecom-compliant Multimedia Terminal Adapter devices.
1. The Internet-Standard Management Framework

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

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

2. Introduction
A multimedia terminal adapter (MTA) is used to deliver broadband Internet, data, and/or voice access jointly with telephony service to a subscriber’s or customer’s premises using a cable network infrastructure. A MTA is normally installed at the customer’s or subscriber’s premises, and it is coupled to a multiple system operator (MSO) using a hybrid fiber coax (HFC) access network.

A MTA is provisioned by the MSO for broadband Internet, data, and/or voice service. For more information on MTA provisioning refer to [PKT-SP-PROV] and [RFC4682]. MTA devices include one or more endpoints (e.g., telephone ports) which receive call signaling information to establish ring cadence, and codecs used for providing telephony service.

For more information on call signaling refer to [PKT-SP-MGCP] and [RFC3435].

For more information on codecs refer to [PKT-SP-CODEC].

Given the complexity of such systems it is important that a suitable event management mechanism be defined to allow for effective management. This MIB module provides objects suitable for generation and management of events on the MTA.

3. Terminology

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 RFC 2119 [RFC2119].

The terms "MIB module" and "information module" are used interchangeably in this memo. As used here, both terms refer to any of the three types of information modules defined in Section 3 of RFC 2578 [RFC2578]. Some of the terms used in this memo are defined below. Some additional terms are also defined in the PacketCable(TM) Management Event Mechanism Specification [PKT-SP-MEM1.5] and the PacketCable MTA Device Provisioning Specification [PKT-SP-PROV].

3.1 PacketCable

PacketCable is a CableLabs-led initiative that is aimed at developing interoperable interface specifications for delivering advanced, real-time multimedia services over two-way cable plants.

3.2 IPCableCom

IPCableCom is an ITU Telecommunication Standardization Sector (ITU-T) project that includes architecture and a series of recommendations that enable the delivery of real time services over the cable television networks using cable modems.
3.3 MTA

A Multimedia Terminal Adapter (MTA) is a PacketCable or IPCablecom compliant device providing telephony services over a cable or hybrid system used to deliver video signals to a community. It contains an interface to endpoints, a network interface, CODECs, and all signaling and encapsulation functions required for Voice over IP transport, call signaling, and Quality of Service signaling. A MTA can be an embedded or a standalone device. An Embedded MTA (E-MTA) is a MTA device containing an embedded Data Over Cable Service Interface Specifications (DOCSIS) Cable Modem. A Standalone MTA (S-MTA) is a MTA device separated from the DOCSIS Cable Modem by non-DOCSIS MAC interface (e.g., Ethernet, USB).

3.4 Endpoint

An endpoint or MTA endpoint is a standard RJ-11 telephony physical port located on the MTA and used for attaching the telephone device to the MTA.

3.5 MSO

Multi-System Operator. A cable company that operates many head-end locations in several cities.

3.6 UDP

User Datagram Protocol. A connectionless protocol built upon Internet Protocol (IP), as per RFC 768 [RFC768].

4. Overview

PacketCable, European Telecommunications Standards Institute (ETSI), and International Telecommunication Union Telecommunication Standardization Sector (ITU-T) IPCablecom compliant Multimedia Terminal Adaptors (MTAs) are required to generate management events upon the occurrence of certain operational conditions. For instance, "AC power failure, MTA operational on battery power". The complete set of conditions and the corresponding management events to be generated are specified in [PKT-SP-MEM1.5], [ETSITS101909-22] and [ITU-T-J176], respectively. In addition, the MTA manufacturer is allowed to specify vendor-specific management events. For example, vendor XYZ can specify "Memory read error, terminating process, code: XYZ123".

When management events are generated, they can either be stored in a local log on the MTA, or transmitted using two possible mechanisms: SNMP and syslog. This is required to be configurable and manageable.
by the management station for each management event (default values can be provided when the events are defined). This Internet-Draft proposes a MIB that can provide for configuration and management of such management events. A means to log the events is provided within the specified MIB module. For syslog as a transport, the necessary information (format, transport etc.) is also specified. For SNMP as a transport, the MIB Objects specified in the SNMP-TARGET-MIB and SNMP-NOTIFICATION-MIB as utilized, as specified in \[RFC3413\].

Further, each management event can be uniquely identified using the ‘Organization ID’ and an ‘Event ID’. The ‘Organization ID’ is the enterprise ID of the organization specifying the event (e.g., 4491 for CableLabs) and a unique identifier that identifies the event. The ‘Event ID’ is an identifier that uniquely identifies the event within the ‘Organization ID’ space. This Internet-Draft does not specify any management events. It only provides a mechanism to manage the storage and transmission of events.

The EVENT MIB module specified in this document is intended to update the EVENT MIB modules from which it is partly derived:
- the PacketCable 1.5 Management Event MIB Specification \[PKT-SP-EVEMIB1.5\],
- the ITU-T IPCablecom management event mechanism MIB requirements \[ITU-T-J176\]

Several normative and informative references are used to help define Management Event MIB objects. As a convention, wherever the requirements are equivalent at the time of the writing, the PacketCable reference is used. However, MTA implementations MUST refer to the corresponding specifications to ensure compliance.

4.1 Structure of the MIB

The Management Event MIB module is identified by pktcEventMib and is structured into six categories:

- pktcEventControl specifies the management information pertinent to control of the device’s event generation capabilities.

- pktcEventThrottle specifies the management information pertinent to throttling the transmission of management events using syslog or SNMP.

- pktcEventStatus specifies the management information for the device to report status information related to the generated events.

- pktcEventDescr specifies the management information for the device to list all the events it is capable of generating.
- pktcEventLog specifies the management information for the device to store the generated events.

- pktcEventNotifications specifies the management information that defines the SNMP trap and inform messages.

4.2 pktcEventControl
The group of objects in this classification provide for three important controls: ability to reset the event logs and event descriptions, syslog configuration and event classes.

Some highlights are as follows:

pktcEventReset - this MIB Object allows a management station to reset the event logs, the event descriptions, or both.

pktcEventSyslog - this group of MIB Objects allow the management station to provide information for transmission of events to a syslog server, such as message formats and transport protocols.

pktcEventClassReportTable - this MIB table allows for MTAs to classify the management events into different categories, termed ‘event classes’. It then allows for common operations to be affected across all the events pertaining to a specific event class.

4.3 pktcEventThrottle
As indicated earlier, the generated events can be stored locally, or transmitted using SNMP, Syslog, or both. However, the management stations receiving such events may wish to control the rate of transmission of such events. This event throttling behavior is provided by the MIB Objects in this category.

Some highlights are as follows:

pktcEventThrottleAdminStatus - this MIB Object allows for transmissions to be unconstrained, maintained below threshold, stopped at the threshold or inhibited.

pktcEventThrottleThreshold - this MIB Object specifies the throttle, i.e., the number of events over an interval that is considered to be the threshold.
pktcEventThrottleInterval - this MIB Object specifies the interval over which the threshold is calculated.

4.4 pktcEventStatus
This category of MIB Objects is designed to provide status information related to event transmissions. It currently contains one MIB Object, ktcEventTransmissionStatus, that allows a client to report the status of event transmissions.

4.5 pktcEventDescr
This category is designed to provide a list of all the events that can be generated by an MTA and its associated descriptions. The MIB Objects are grouped under the MIB table pktcEventDescrTable.

4.6 pktcEventLog
This category is designed to allow the MTA to store all the events that are generated during its operation. The events are stored with information such as the time of the event, its description and related characteristics like severity levels.

4.7 pktcEventNotifications
This category specifies the notification information, when MTAs transmit messages using SNMP traps and informs. SNMP traps refer to the SNMPv2-Trap-PDU. SNMPv1 traps are explicitly disallowed.

5. Relationship to Other MIB Modules
Some management objects defined in other MIB modules are applicable to an entity implementing this MIB. In particular, it is assumed that an entity implementing the PKTC-IETF-EVENT-MIB module will also implement the 'interfaces' group of the IF-MIB [RFC2863].

5.1 MIB modules required for IMPORTS
The PKTC-IETF-EVENT-MIB MIB module IMPORTS objects from SNMPv2-SMI [RFC2578], SNMPv2-TC [RFC2579], SNMP-FRAMEWORK-MIB [RFC3411], SNMPv2-CONF [RFC2580], IF-MIB [RFC2863] and the INET-ADDRESS-MIB [RFC4001].

6. Definitions
PKTC-EVENT-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY,
    OBJECT-TYPE,
    Unsigned32,

NOTIFICATION-TYPE,
mib-2 FROM SNMPv2-SMI
TruthValue,
DateAndTime, TEXTUAL-CONVENTION
FROM SNMPv2-TC
SnmpAdminString FROM SNMP-FRAMEWORK-MIB
OBJECT-GROUP,
MODULE-COMPLIANCE,
NOTIFICATION-GROUP FROM SNMPv2-CONF
ifPhysAddress FROM IF-MIB
InetAddressType,
InetAddress,
InetPortNumber FROM INET-ADDRESS-MIB
snmpTargetBasicGroup, snmpTargetResponseGroup
FROM SNMP-TARGET-MIB
snmpNotifyGroup, snmpNotifyFilterGroup
FROM SNMP-NOTIFICATION-MIB
SyslogSeverity, SyslogFacility FROM SYSLOG-MIB;

pktcEventMib MODULE-IDENTITY
LAST-UPDATED "200707060000Z" -- July 06, 2007
ORGANIZATION "IETF IP over Cable Data Network Working Group"
CONTACT-INFO
"Sumanth Channabasappa
Cable Television Laboratories, Inc.
858 Coal Creek Circle,
Louisville, CO 80027, USA
+1 303-661-3307
Sumanth@cablelabs.com

Wim De Ketelaere
tComLabs
Gildestraat 8
9000 Gent, Belgium
+32 9 269 22 90
deketelaere@tComLabs.com

Eugene Nechamkin
Broadcom Corporation
200 - 13711 International Place
Richmond, BC, V6V 2Z8, Canada
+1 604 233 8500
eechamkin@broadcom.com

IETF IPCDN Working Group
General Discussion: ipcdn@ietf.org
Subscribe: http://www.ietf.org/mailman/listinfo/ipcdn
Archive: ftp://ftp.ietf.org/ietf-mail-archive/ipcdn
Co-Chair: Jean-Francois Mule, jf.mule@cablelabs.com
DESCRIPTION

"This MIB module specifies the basic management objects for managing events generated by the Multimedia Terminal Adapter devices compliant with the PacketCable and IPCablecom requirements.

Copyright (C) The IETF Trust (2007). This version of this MIB module is part of RFC nnnn; see the RFC itself for full legal notices."

-- RFC Ed: replace nnnn with actual RFC number and remove this note.

REVISION   "200707060000Z" -- July 06, 2007

DESCRIPTION

"Initial version, published as RFC yyyy."

-- RFC Ed: replace yyyy with actual RFC number and remove this note

::=  { mib-2 XXX }

-- RFC Ed: replace XXX with IANA-assigned number and remove this
-- note

SyslogSeverityMask ::= TEXTUAL-CONVENTION

STATUS       current

DESCRIPTION

"This textual convention represents a bit mask representing the severity of the events that can be generated by an MTA. It corresponds to the various severity levels associated with syslog messages, as specified in The syslog Protocol, [RFCAAA].

emergency(0)  - A condition that makes the system unusable.
alert(1)      - A service-affecting condition for which immediate action must be taken.
critical(2)   - A service-affecting critical condition.
error(3)      - An error condition.
warning(4)    - A warning condition.
notice(5)     - A normal but significant condition.
info(6)       - An informational message.
debug(7)      - A debug message."

SYNTAX  BITS { emergency(0),
                     alert(1),
                     critical(2),
                     error(3),
warning(4),
notice(5),
info(6),
dbg(7)
}

--

--
pktcEventNotifications OBJECT IDENTIFIER ::= { pktcEventMib 0 }
pktcEventMibObjects OBJECT IDENTIFIER ::= { pktcEventMib 1 }
pktcEventConformance OBJECT IDENTIFIER ::= { pktcEventMib 2 }
--

--
pktcEventControl OBJECT IDENTIFIER ::= { pktcEventMibObjects 1 }
pktcEventThrottle OBJECT IDENTIFIER ::= { pktcEventMibObjects 2 }
pktcEventStatus OBJECT IDENTIFIER ::= { pktcEventMibObjects 3 }
pktcEventDescr OBJECT IDENTIFIER ::= { pktcEventMibObjects 4 }
pktcEventLog OBJECT IDENTIFIER ::= { pktcEventMibObjects 5 }

---

-- Event Reporting control objects
---
pktcEventReset OBJECT-TYPE
SYNTAX  BITS {
    resetEvLogTable(0),
    resetEvDescrTable(1)
}
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION "This MIB object allows a management station to clear the local log of generated events, reset the management event descriptions, or both.

MTAs generate management events. These events are stored in the MIB table ‘pktcEventLogTable’. If a management station needs to clear all the current entries (e.g., after a troubleshooting operation is complete), it can do so by setting the resetEvLogTable(0) bit to a value of ‘1’.

The MTA is pre-configured with the events that it can generate. This is stored in the MIB table ‘pktcEventDescrTable’. This table also contains the descriptions associated with these events. These descriptions can be modified by a management station. However, if the management station wishes to reset the descriptions (to factory defaults), it can do so by
setting the resetEvDescrTable(1) bit to a value of ‘1’.

The MTA actions are summarized below:

Bit resetEvLogTable(0) set to a value of ‘1’
- delete all entries in pktcEventLogTable;
- reset the value of pktcEventLogIndex to ‘0’.

Bit resetEvDescrTable(1) set to a value of ‘1’
- reset the pktcEventDescrTable to the factory default values.

Bits resetEvLogTable(0) and resetEvDescrTable(1) set to a value of ‘1’
- perform the above actions as though they were performed individually (in any order).

Setting a reset bit to a value of ‘0’ MUST not result in any action.

The MTA MUST perform the above actions regardless of persistence (i.e. storage in non-volatile memory).

The MTA MUST always return a value of ‘00’ when this MIB Object is read.

::= { pktcEventControl 1 }

---
-- syslog-specific MIB Objects
---

pktcEventSyslog OBJECT IDENTIFIER ::= { pktcEventControl 2 }

pktcEventSyslogCapabilities OBJECT-TYPE
SYNTAX BITS {
    formatBSDSyslog(0),
    formatSyslogProtocol(1),
    transportUDP(2),
    transportTLS(3)
}
MAX-ACCESS read-only
STATUS current
DESCRIPTION "This MIB Object contains the MTA capabilities for supporting the Syslog protocol.

The MTA is allowed to support two Syslog message
formats, the BSD syslog message formats specified in [RFC3164] (formatBSDSyslog), and the IETF syslog protocol per [RFCAAA] (formatSyslogProtocol).

The MTA MUST support formatBSDSyslog. The MTA SHOULD support the IETF syslog standard.

BSD syslog protocol specifies UDP as the transport protocol, and the IETF syslog protocol allows for multiple transport protocols. This MIB module supports two transport protocols: UDP and TLS (the latter is only applicable when formatSyslogProtocol(2) is supported).

The MTA MUST support UDP transport. The MTA MAY support TLS for transport if it supports the IETF syslog protocol.

The MTA MUST set the appropriate bits, based on its implementation."

REFERENCE
"The BSD syslog Protocol, [RFC3164];
The syslog Protocol, [RFCAAA];
Transmission of syslog messages over UDP, [RFCBBB];
TLS Transport Mapping for Syslog, [RFCCCC]."

 ::= { pktcEventSyslog 1 }

pktcEventSyslogAddressType OBJECT-TYPE
SYNTAX       InetAddressType
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
"This MIB Object defines the Internet address type of
the syslog server specified by the MIB Object
'pktcEventSyslogAddress' ."

REFERENCE
"PacketCable MTA Device Provisioning Specification,
[PKT-SP-PROV]."

DEFVAL { ipv4 }  ::= { pktcEventSyslog 2 }

pktcEventSyslogAddress OBJECT-TYPE
SYNTAX       InetAddress
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
"This MIB Object contains the IP address of the

syslog server to which the MTA can transmit a syslog message upon the generation of a management event.

If this MIB Object is set to a value other than a routable unicast address (‘0.0.0.0’ or 255.255.255.255 for IPv4), the MTA MUST suspend the transmission of Syslog messages.

The use of FQDNs is syntactically allowed, but not recommended since a failure to resolve them in a timely manner may leave the device without access to the syslog daemon during critical network events.

The type of address this object represents is defined by the MIB Object pktDevEvsyslogAddressType.

The format of the syslog message is specified by the MIB Object 'pktcEventSyslogMsgFormat'.
pktcEventSyslogPort  OBJECT-TYPE
SYNTAX     InetPortNumber
MAX-ACCESS read-write
STATUS     current
DESCRIPTION
"This MIB Object contains the port number of the syslog Server to which the syslog messages are to be transmitted. The MTA MUST use the specified value to transmit to the syslog Server."

REFERENCE
"Transmission of syslog messages over UDP, [RFCBBBB]; TLS Transport Mapping for Syslog, [RFCCCC]."
DEFVAL { 514 }
::= { pktcEventSyslog 5 }

pktcEventSyslogMsgFormat  OBJECT-TYPE
SYNTAX INTEGER {
    formatBSDSyslog(1), -- The BSD syslog Protocol
    formatSyslogProtocol(2) -- The syslog Protocol
 }
MAX-ACCESS read-write
STATUS     current
DESCRIPTION
"This MIB Object contains the syslog message format to be used for transmitting syslog messages to the server contained in the MIB Object 'pktcEventSyslogServer'.

Two formats are supported: formatBSDSyslog(1) as specified in [RFC3164] and formatSyslogProtocol(2) as specified in [RFCAAA].

MTAs MUST support formatBSDSyslog(1). MTAs SHOULD support formatSyslogProtocol(2)."

REFERENCE
"The BSD syslog Protocol, [RFC3164];
The syslog Protocol, [RFCAAA]."
DEFVAL { formatBSDSyslog }
::= { pktcEventSyslog 6 }

---

-- Event classes
---
pktcEventClassReportTable OBJECT-TYPE
SYNTAX     SEQUENCE OF PktcEventClassReportEntry
MAX-ACCESS not-accessible
STATUS     current
DESCRIPTION

"This MIB table allows for management events that can be
generated by an MTA to be classified into categories,
or 'event classes'. For example, all the configuration
related events can be associated with an event class
titled 'configuration'. Such as a classification allows
for a management station to affect changes on a common
group of events, at once. Two operations are specified
on an event class: enabling or disabling of all the
events in an event class, and selective enabling or
disabling based on the severity level."
::= { pktcEventControl 3 }

pktcEventClassReportEntry OBJECT-TYPE
SYNTAX     PktcEventClassReportEntry
MAX-ACCESS not-accessible
STATUS     current
DESCRIPTION

"Each entry in this table specifies an event class as
identified by the MTA manufacturer, based on the events
supported.

The MTA MUST create one entry (index=100) for the event
class titled 'generic'. This event class MUST contain
all the events that do not belong to a vendor specified
event class.

In the case of events belonging to multiple categories,
event classes with lower indices take priority over
others."
INDEX { pktcEventClassIndex }
::= { pktcEventClassReportTable 1 }

PktcEventClassReportEntry ::= SEQUENCE {
   pktcEventClassIndex           Unsigned32,
   pktcEventClassName            SnmpAdminString,
   pktcEventClassReportStatus    TruthValue,
   pktcEventClassSeverityLevel   SyslogSeverityMask
}

pktcEventClassIndex OBJECT-TYPE
SYNTAX     Unsigned32 (1..100)
MAX-ACCESS not-accessible
STATUS     current
DESCRIPTION
"This MIB Object is an index into the event class table."
 ::= { pktcEventClassReportEntry 1 }

pktcEventClassName OBJECT-TYPE
SYNTAX SnmpAdminString (SIZE (1..100))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This MIB Object contains the event class name of an event class supported by the device.

Vendors MAY define different event classes (e.g. DHCP, SNMP, DEBUG) to group together management events of a particular category."
 ::= { pktcEventClassReportEntry 2 }

pktcEventClassReportStatus OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"This MIB Object indicates if events belonging to a particular event class are to be enabled or disabled for event reporting.

Setting this object to a value of ‘true’ enables reporting of all the events in the event class. When enabled, the means of reporting is specified by the MIB Object ‘pktcEventDescrReporting’.

Setting this object to a value of ‘false’ disables any reporting, irrespective of the value of the MIB Object ‘pktcEventDescrReporting’ for a specific event.

The default value of this MIB Object is vendor specific. However, the vendor SHOULD enable all event categories defined by PacketCable or IPCableCom, by default."
 ::= { pktcEventClassReportEntry 3 }

pktcEventClassSeverityLevel OBJECT-TYPE
SYNTAX SyslogSeverityMask
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"This MIB Object defines the severity level
of events belonging to a specific event class, that are enabled for event reporting.

This MIB Object has no effect on the event reporting unless the MIB Object 'pktcEventClassReportStatus' is set to a value of 'true', for the corresponding event class.

Setting a bit to a value of '1' implies that events corresponding to that severity level MUST be reported as defined by the corresponding value of 'pktcEventDescrReporting' for events in the event class.

Setting a bit to a value of '0' implies that events corresponding to that level MUST NOT be reported, irrespective of the corresponding value of 'pktcEventDescrReporting' for events in the event class.

It is recommended that the bits corresponding to emergency(0), alert(1), critical(2) and error(3) always be set to a value of '1'.

REFERENCE
"The syslog Protocol, [RFCAAA]; Syslog Management Information Base, [RFCABC]."
::= { pktcEventClassReportEntry 4 }

---
-- Event throttling control
---

pktcEventThrottleAdminStatus OBJECT-TYPE
SYNTAX INTEGER {
  unconstrained(1),
  maintainBelowThreshold(2),
  stopAtThreshold(3),
  inhibited(4)
}
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"This MIB Object controls the throttling of the transmitted messages upon generation of an event (SNMP/syslog).

A value of unconstrained(1) causes event messages to be transmitted without regard to the threshold settings."
A value of maintainBelowThreshold(2) causes event messages to be suppressed if the number of transmissions would otherwise exceed the threshold. A value of stopAtThreshold(3) causes event message transmission to cease at the threshold, and not resume until directed to do so.

A value of inhibited(4) causes all event message transmission to be suppressed.

An event causing both an SNMP and a syslog message is still treated as a single event.

Writing to this object resets the thresholding state.

Refer to MIB Objects pktcEventThrottleThreshold and pktcEventThrottleInterval for information on throttling.

DEFVAL { unconstrained }
 ::= { pktcEventThrottle 1 }

pktcEventThrottleThreshold OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
 "This MIB Object contains the number of events per pktcEventThrottleInterval to be transmitted before throttling.

An event causing both an SNMP and a syslog message is still treated as a single event."
DEFVAL { 2 }
 ::= { pktcEventThrottle 2 }

pktcEventThrottleInterval OBJECT-TYPE
SYNTAX      Unsigned32
UNITS      "seconds"
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
 "This MIB Object contains the interval over which the throttle threshold applies."
DEFVAL { 1 }
 ::= { pktcEventThrottle 3 }
---
-- Reporting of transmission status
---

pktcEventTransmissionStatus OBJECT-TYPE
SYNTAX      BITS {
                   syslogThrottled(0),
                   snmpThrottled(1),
                   validsyslogServerAbsent(2),
                   validSnmpManagerAbsent(3),
                   syslogTransmitError(4),
                   snmpTransmitError(5)
                }  
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION  
    "This MIB Object reflects the status of the event
transmissions.

    If a bit corresponding to a state is set to a value
    of:
        '1', it indicates that the state is true
        '0', it indicates that the state is false

    'Event throttling' is based on thresholds and the
    current setting of pktcEventThrottleAdminStatus.

    'Server/Manager' indicators must be based on the
    availability of valid syslog server/SNMP managers.

    'Transmit Errors' must only be used in cases where the
    MTA can identify unavailable servers."

 ::= { pktcEventStatus 1 }

---
-- Description of events
---

pktcEventDescrTable OBJECT-TYPE
SYNTAX      SEQUENCE OF PktcEventDescrEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION  
    "This MIB table contains all possible management events"
that can be generated by the device. This includes both PacketCable and IPCableCom defined, and vendor-specific events."

::= { pktcEventDescr 1 }

pktcEventDescrEntry OBJECT-TYPE
SYNTAX      PktcEventDescrEntry
MAX-ACCESS  not-accessible
STATUS      current

DESCRIPTION
"An entry in this table is created for each event the MTA implementing this MIB is capable of reporting."

INDEX { pktcEventDescrOrg, pktcEventDescrId }

::= { pktcEventDescrTable 1 }

PktcEventDescrEntry ::= SEQUENCE {
  pktcEventDescrOrg             Unsigned32,
pktcEventDescrId              Unsigned32,
pktcEventDescrFacility        INTEGER,
pktcEventDescrSeverityLevel   SyslogSeverity,
pktcEventDescrReporting       BITS,
pktcEventDescrText            SnmpAdminString,
pktcEventDescrClass           SnmpAdminString,
}

pktcEventDescrOrg OBJECT-TYPE
SYNTAX      Unsigned32(1..4294967295)
MAX-ACCESS  not-accessible
STATUS      current

DESCRIPTION
"This MIB Object provides the IANA enterprise number of the Organization defining the event. Thus, all PacketCable or IPCableCom defined events will contain the PacketCable or IPCableCom IANA enterprise number and for vendor-specific events it will contain the IANA enterprise number of the defining organization."

::= { pktcEventDescrEntry 1 }

pktcEventDescrId OBJECT-TYPE
SYNTAX      Unsigned32(1..4294967295)
MAX-ACCESS  not-accessible
STATUS      current

DESCRIPTION
"This MIB Object contains the event identifier for the specific event to which the priority and display strings belong."

REFERENCE
"PacketCable Management Event Mechanism Specification, [PKT-SP-MEM1.5];
PacketCable MTA Device Provisioning Specification, [PKT-SP-PROV]."
::= { pktcEventDescrEntry 2 }

pktcEventDescrFacility OBJECT-TYPE
SYNTAX SyslogFacility
MAX-ACCESS read-only
STATUS current
DESCRIPTION "This MIB Object contains the facility for the event. For PacketCable, IPCableCom or ETSI events this MUST be set to a value of local0(16)."
REFERENCE "The syslog Protocol, [RFCAAA]."
::= { pktcEventDescrEntry 3 }

pktcEventDescrSeverityLevel OBJECT-TYPE
SYNTAX SyslogSeverity
MAX-ACCESS read-write
STATUS current
DESCRIPTION "This MIB Object contains the severity level that is applicable to the specified event."
REFERENCE "The syslog Protocol, [RFCAAA]; Syslog Management Information Base, [RFCABC]."
::= { pktcEventDescrEntry 4 }

pktcEventDescrReporting OBJECT-TYPE
SYNTAX BITS {
  local(0),
  syslog(1),
  snmpTrap(2),
  snmpInform(3)
}
MAX-ACCESS read-write
STATUS current
DESCRIPTION "This MIB Object defines the action to be taken on occurrence of this event. Bit local(0) refers to local logging of events, bit syslog(1) refers to the transmission of events using syslog, bit snmpTrap(2) refers to the transmission of events using SNMP Traps (SNMPv2-Trap-PDU) and bit snmpInform(3) refers to the transmission of events using SNMP INFORMs."
Setting a bit to a value of '1' indicates that the corresponding action will be taken upon occurrence of this event. If none of the bits are set then no action is taken upon occurrence of the event. The success of transmission using syslog and SNMP depends on the MTA configuration. For example, a valid syslog server address is required for syslog message transmission.

Specification of a management event does not necessarily include the actions to be taken upon its generation, i.e., it does not need to specify if a generated event needs to be transmitted via SNMP, syslog or locally stored.

Thus, certain default values of this MIB object is are specified, based on the event’s severity level (refer to the MIB Object ‘pktcEventDescrSeverityLevel’.)

If the severity level of an event is emergency(0), alert(1), critical(2) or error(3), the MTA MUST set the bits for local(0), syslog(1) and snmpInform(3) to a value of '1', and the remaining bits to a value of '0'. For an event with any other severity level the MTA MUST set the bits for local(0) and syslog(1) to a value of '1' and the rest of the bits to a value of '0'.

::= { pktcEventDescrEntry 5 }

pktcEventDescrText OBJECT-TYPE
  SYNTAX SnmpAdminString(SIZE (0..127))
  MAX-ACCESS read-write
  STATUS current
  DESCRIPTION
  "This MIB Object provides a human-readable description of the event."
  ::= { pktcEventDescrEntry 6 }

pktcEventDescrClass OBJECT-TYPE
  SYNTAX SnmpAdminString(SIZE (1..100))
  MAX-ACCESS read-only
  STATUS current
  DESCRIPTION
  "This MIB Object represents an event class associated with the corresponding event."

  See the MIB table ‘pktcEventClassReportTable’ for a description of event classes and usage.

  If the event is not categorized into a specific
vendor-specified event class, it will be associated with the event class 'generic'."

::= { pktcEventDescrEntry 7 }

---
-- Log of generated events
---

pktcEventLogTable OBJECT-TYPE
SYNTAX SEQUENCE OF PktcEventLogEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"This MIB table contains a log of the events generated by the MTA.
A description of all the events that can be generated by the device can be obtained from the
MIB table 'pktcEventDescrTable'. The contents of this table MAY persist in non-volatile memory."
::= { pktcEventLog 1 }

pktcEventLogEntry OBJECT-TYPE
SYNTAX PktcEventLogEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Each entry in this table describes an event that has occurred, indexed in the chronological order of generation. The details of the event are borrowed from the parameters associated with the corresponding event entry in 'pktcEventDescrTable', at the time of the event generation. While all entries created as such can be cleared using the MIB Object pktcEventReset, the Event entries themselves cannot be individually deleted."

INDEX { pktcEventLogIndex }
::= { pktcEventLogTable 1 }

PktcEventLogEntry ::= SEQUENCE {
pktcEventLogIndex             Unsigned32,
pktcEventLogTime              DateAndTime,
pktcEventLogOrg               Unsigned32,
pktcEventLogId                Unsigned32,
pktcEventLogText              SnmpAdminString,
pktcEventLogEndpointName      SnmpAdminString,
pktcEventLogType              BITS,
pktcEventLogTargetInfo        SnmpAdminString,
pktcEventLogCorrelationId OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "This MIB Object provides relative ordering of the objects in the event log."

pktcEventLogAdditionalInfo OBJECT-TYPE
SYNTAX SnmpAdminString
MAX-ACCESS read-only
STATUS current
DESCRIPTION "This MIB Object provides a human-readable description of the time at which the event occurred. The actual time used SHOULD be defined by the architecture using this MIB definition. If unspecified, this MUST be the local time."

REFERENCE
"The BSD syslog Protocol, [RFC3164];
The syslog Protocol, [RFCAAA]."

pktcEventLogOrg OBJECT-TYPE
SYNTAX Unsigned32(1..4294967295)
MAX-ACCESS read-only
STATUS current
DESCRIPTION "This MIB Object provides the IANA enterprise number of the Organization defining the event. Thus, all PacketCable or IPCableCom defined events will contain the CableLabs or IPCableCom IANA enterprise number and for vendor-specific events it will contain the IANA
enterprise number of the defining organization.

::= { pktcEventLogEntry 3 }

pktcEventLogId OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION  "This MIB Object contains the event identifier for the
specific event to which the priority and display strings
belong."

::= { pktcEventLogEntry 4 }

pktcEventLogText OBJECT-TYPE
SYNTAX      SnmpAdminString
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION  "This MIB Object contains the contents of
the MIB Object ‘pktcEventDescrText’, corresponding
to the event, at the moment of generation."

::= { pktcEventLogEntry 5 }

pktcEventLogEndpointName OBJECT-TYPE
SYNTAX      SnmpAdminString
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION  "This MIB Object provides the endpoint identifier
followed by the PacketCable MTA’s Fully Qualified
Domain Name (FQDN) and the IP Address (IP)
of the PacketCable MTA device.

This will be denoted as follows:
aaln/n:<FQDN>/*IP>, where ‘n’ is the Endpoint number.
or
<FQDN>/IP> if it is not specific to an endpoint."

::= { pktcEventLogEntry 6 }

pktcEventLogType OBJECT-TYPE
SYNTAX      BITS {
  local (0),
  syslog (1),
  trap (2),
  inform (3)
}
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"This MIB Object contains the kind of actions taken by
the MTA when the event occurred.

A bit with a value of '1' indicates the corresponding
action was taken. Setting it to a value of '0'
indicates that the corresponding action was not taken.

An event may trigger one or more actions (e.g.: syslog
and SNMP) or may remain as a local event since
transmissions could be disabled or inhibited as defined
by the Throttle MIB Objects."

::= { pktcEventLogEntry 7 }

pktcEventLogTargetInfo OBJECT-TYPE
SYNTAX     SnmpAdminString
MAX-ACCESS read-only
STATUS     current
DESCRIPTION
"This MIB Object contains a comma separated list of the
actions taken for external notifications, along with the
target IP address for the generated events. Locally
stored events must not be recorded in this MIB Object.

The syntax is as:
$action-1/IP>,<$action-2/IP>,<$action-3/IP>

Where $action-n/IP is to be denoted as follows:
For syslog events:
    syslog/<IP address of the syslog Server>
For SNMP traps:
    snmpTrap/<IP address of the SNMP Server>
For SNMP INFORMS:
    snmpInform/<IP address of the SNMP Server>

If there are multiple targets for the same type (SNMP
Traps sent to multiple IP addresses) or if there are
multiple messages sent to the same IP (syslog and SNMP
sent to the same IP address) they need to be reported
individually."

::= { pktcEventLogEntry 8 }

pktcEventLogCorrelationId OBJECT-TYPE
SYNTAX     Unsigned32
MAX-ACCESS read-only
STATUS     current
DESCRIPTION
"This MIB Object contains the correlation ID generated by the MTA during the initiation of the last provisioning flow, within or following which the event occurred.

Although a Correlation-ID, once generated after MTA reset, does not change until next MTA reset, the value of this object will differ for the events preserved across MTA resets in case of a persistent pktcEventLogTable.

For more information on the generation of correlation ids, refer to the corresponding PacketCable/IPCableCom Device Provisioning specifications."

REFERENCE
"PacketCable MTA Device Provisioning Specification, [PKT-SP-PROV]."

::= { pktcEventLogEntry 9 }

pktcEventLogAdditionalInfo OBJECT-TYPE
SYNTAX    SnmpAdminString
MAX-ACCESS read-only
STATUS     current
DESCRIPTION
"This MIB Object contains additional, useful information in relation to the corresponding event that an MTA might wish to report (for example: parameterized data or debugging information). The format is vendor-specific.
If MTA cannot provide any additional information for the particular event generated, it MUST populate this MIB Object with a string of zero-length."

::= { pktcEventLogEntry 10 }

---
-- Notifications
---

pktcEventInform NOTIFICATION-TYPE
OBJECTS {
    pktcEventLogTime,
    pktcEventLogOrg,
    pktcEventLogId,
    pktcEventLogEndpointName,
    pktcEventLogCorrelationId,
    ifPhysAddress
}
STATUS     current
DESCRIPTION
"This Notification MIB Objects contains the SNMP Inform contents for event reporting.

It contains the event log time, the organization ID, the event identifier, the endpoint identifier, the correlation id, and the MTA MAC address."
::= { pktcEventNotifications 1 }

pktcEventTrap NOTIFICATION-TYPE
OBJECTS {
pktcEventLogTime,
pktcEventLogOrg,
pktcEventLogId,
pktcEventLogEndpointName,
pktcEventLogCorrelationId,
ifPhysAddress}
STATUS current
DESCRIPTION
"This Notification MIB Objects contains the SNMP Trap (SNMPv2-Trap-PDU) contents for event reporting.

It contains the event log time, the organization ID, the event identifier, the endpoint identifier, the correlation id, and the MTA MAC address."
::= { pktcEventNotifications 2 }

---
-- Conformance/Compliance
---

pktcEventCompliances  OBJECT IDENTIFIER ::= { pktcEventConformance  1 }
pktcEventGroups  OBJECT IDENTIFIER ::= { pktcEventConformance  2 }

pktcEventBasicCompliance MODULE-COMPLIANCE
STATUS current
DESCRIPTION
"The compliance statement for devices that implement Event reporting feature."
MODULE --pktcEventMib

MANDATORY-GROUPS {
pktcEventGroup,  
pktcEventNotificationGroup  
}  

MODULE SNMP-TARGET-MIB  
MANDATORY-GROUPS {  
    snmpTargetBasicGroup,  
    snmpTargetResponseGroup  
}  

MODULE SNMP-NOTIFICATION-MIB  
MANDATORY-GROUPS {  
    snmpNotifyGroup,  
    snmpNotifyFilterGroup  
}  

::= { pktcEventCompliances 3 }  

pktcEventGroup OBJECT-GROUP  
OBJECTS {  
    pktcEventReset,  
    pktcEventSyslogCapabilities,  
    pktcEventSyslogAddressType,  
    pktcEventSyslogAddress,  
    pktcEventSyslogTransport,  
    pktcEventSyslogPort,  
    pktcEventSyslogMsgFormat,  
    pktcEventThrottleAdminStatus,  
    pktcEventThrottleThreshold,  
    pktcEventThrottleInterval,  
    pktcEventTransmissionStatus,  
    pktcEventDescrFacility,  
    pktcEventDescrSeverityLevel,  
    pktcEventDescrReporting,  
    pktcEventDescrText,  
    pktcEventLogTime,  
    pktcEventLogOrg,  
    pktcEventLogId,  
    pktcEventLogText,  
    pktcEventLogEndpointName,  
    pktcEventLogType,  
    pktcEventLogTargetInfo,  
    pktcEventLogCorrelationId,  
    pktcEventLogAdditionalInfo,  
    pktcEventDescrClass,  
    pktcEventClassName,  
    pktcEventClassReportStatus,  
    pktcEventClassSeverityLevel  
}
STATUS       current
DESCRIPTION
"Group of MIB objects for PacketCable Management Event
MIB."
::= { pktcEventGroups 1 }

pktcEventNotificationGroup NOTIFICATION-GROUP
   NOTIFICATIONS { pktcEventInform, pktcEventTrap }
STATUS       current
DESCRIPTION
"Group of MIB objects for notifications related to
change in status of the MTA Device."
::= { pktcEventGroups 2 }
END

7. Acknowledgments

The current editors would like to thank the members of the IETF
IPCDN working group and the CableLabs PacketCable Provisioning focus
team for their contributions, comments and suggestions.

The current editors wish to express their gratitude to the following
individuals (in alphabetical order): Dan Romanscu, David Harrington,
Greg Nakanishi, Jean-Francois Mule, John Berg, Kevin Marez, Paul
Duffy, Peter Bates, Randy Presuhn, Rich Woundy, Rick Vetter, Roy
Spitzer, and Satish Kumar.

Special thanks to the MIB doctors David Harrington and Dan Romanscu,
as well as Rich Woundy for expert feedback on the document.

8. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate

[PKT-SP-PROV] Packetcable MTA Device Provisioning Specification
PKT-SP-PROV-I11-050812.


(MTA) Management Information Base for PacketCable and IPCablecom

De Ketelaere/Nechamkin/Channabasappa Expires - January 2008  [Page 30]
[RFCABC] Glenn Mansfield Keeni, "Syslog Management Information Base", RFCABC, <Date>

Editor’s Note (to be removed prior to publication): This is the reference to ‘draft-ietf-syslog-device-mib’. When the draft is accepted, the RFC Editor is requested to replace ABC (and in the reference earlier) with the assigned value and to remove this note.

[RFCAAA] R. Gerhards, "The syslog Protocol", RFCAAA, <Date>

Editor’s Note (to be removed prior to publication): This is the reference to ‘draft-ietf-syslog-protocol’. When the draft is accepted, the RFC Editor is requested to replace AAA (and in the reference earlier) with the assigned value and to remove this note.

[RFCBBB] A. Okmianski, "Transmission of syslog messages over UDP", RFCBBB, <Date>

Editor’s Note (to be removed prior to publication): This is the reference to ‘draft-ietf-syslog-transport-udp’. When the draft is accepted, the RFC Editor is requested to replace BBB (and in the reference earlier) with the assigned value and to remove this note.

[RFCCCC] F. Miao, M. Yuzhi, "TLS Transport Mapping for Syslog", RFCCCC, <Date>

Editor’s Note (to be removed prior to publication): This is the reference to ‘draft-ietf-syslog-transport-tls’. When the draft is accepted, the RFC Editor is requested to replace CCC (and in the reference earlier) with the assigned value and to remove this note.


9. Informative References


10. IANA Considerations

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

<table>
<thead>
<tr>
<th>Descriptor</th>
<th>OBJECT IDENTIFIER Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>pktcEventMib</td>
<td>{ mib-2 XXX }</td>
</tr>
</tbody>
</table>

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.

11. Security Considerations

There are a number of management objects defined in this MIB module
with a MAX-ACCESS clause of read-write. Such objects may be considered sensitive or vulnerable in some network
environments. The support for SET operations in a non-secure
environment without proper protection can have a negative effect on
network operations. Security threats include events unreported on
errors, redirection of events (deliberately or otherwise) or
minimized reporting of errors. Such threats can mask certain
misconfiguration attempts and denial of service attacks that can be
recognized and thwarted via event reporting.

MIB objects of significance include:
- those that control the event generation, the target syslog address
for events and the reporting status, i.e.:
  pktcEventReset
  pktcEventSyslogAddressType
  pktcEventSyslogAddress
  pktcEventSyslogUdpPort
  pktcEventClassReportStatus
- those related to event classes, i.e.:
  pktcEventClassSeverityLevel
- those related to throttling, i.e.:
  pktcEventThrottleAdminStatus
  pktcEventThrottleThreshold
  pktcEventThrottleInterval
- those related to the event reporting capabilities of an MTA, i.e:
  pktcEventDescrSeverityLevel
  pktcEventDescrReporting
  pktcEventDescrText
  pktcEventDescrClass

Some of the readable objects in this MIB module (i.e., objects
with a MAX-ACCESS other than not-accessible) may be considered
sensitive or vulnerable in some network environments. It is thus
important to control even GET and/or NOTIFY access to these objects
and possibly to even encrypt the values of these objects when
sending them over the network via SNMP. These are the tables and
objects and their sensitivity/vulnerability:

pktcEventLogTable: This table contains the log of generated event
messages. Read access to this table might reveal some specific
information that should be kept confidential.

pktcEventTransmissionStatus: This MIB Object reveals the status of
event transmission and MAY be sensitive in some environments.

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 perform GET or SET (change/create/delete)
operations.

12. Authors’ Addresses
Sumanth Channabasappa
Cable Television Laboratories, Inc.
858 Coal Creek Circle,
Louisville, CO 80027, USA

De Ketelaere/Nechamkin/Channabasappa Expires - January 2008  [Page 34]
PacketCable/IPCablecom Event management MTA MIB

+1 303-661-3307  
Sumanth@cablelabs.com

Wim De Ketelaere  
tComLabs  
Gildestraat 8  
9000 Gent, Belgium  
+32 9 269 22 90  
deketelaere@tComLabs.com

Eugene Nechamkin  
Broadcom Corporation  
200 - 13711 International Place  
Richmond, BC, V6V 2Z8, Canada  
+1 604 233 8500  
enechamkin@broadcom.com

Full Copyright Statement

Copyright (C) The IETF Trust (2007).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this
specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).