

International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

J.126

(12/2007)

SERIES J: CABLE NETWORKS AND TRANSMISSION
OF TELEVISION, SOUND PROGRAMME AND OTHER
MULTIMEDIA SIGNALS

Interactive systems for digital television distribution

Embedded Cable Modem device specification

ITU-T Recommendation J.126



ITU-T Recommendation J.126

Embedded Cable Modem device specification

Summary

ITU-T Recommendation J.126 defines additional features that must be added to a Cable Modem for implementations that embed the Cable Modem with another application, such as an IPCablecom multimedia terminal adapter (MTA).

Source

ITU-T Recommendation J.126 was approved on 14 December 2007 by ITU-T Study Group 9 (2005-2008) under the ITU-T Recommendation A.8 procedure.

History

| Edition | | Approval |
|---------|-------|------------|
| 1.0 | J.126 | 2004-04-22 |
| 2.0 | J.126 | 2007-12-14 |

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g. interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2008

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

CONTENTS

| | Page |
|--|-------------|
| 1 Scope | 1 |
| 2 References..... | 1 |
| 2.1 Normative references..... | 1 |
| 2.2 Informative references..... | 2 |
| 3 Definitions | 3 |
| 4 Abbreviations and conventions..... | 4 |
| 4.1 Abbreviations | 4 |
| 4.2 Conventions..... | 6 |
| 5 Embedded cable modem..... | 6 |
| 5.1 Device interface reference model..... | 6 |
| 5.2 eDOCSIS requirements | 14 |
| Annex A – SLED MIB definition..... | 39 |
| Annex B – eSAFE MIB Definition..... | 44 |
| Annex C – Format and content for eCM/eSTB event, SYSLOG, and SNMP trap extensions | 55 |

ITU-T Recommendation J.126

Embedded Cable Modem device specification

1 Scope

This Recommendation defines additional features that must be added to a Cable Modem for implementations that embed the Cable Modem with another application, such as an IPCablecom MTA.

NOTE – The structure and content of this Recommendation have been organized for ease of use by those familiar with the original source material; as such, the usual style of ITU-T Recommendations has not been applied.

2 References

2.1 Normative references

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

- [BB] ITU-T Recommendation J.199 (2006), *Battery backup for cable-based devices*.
- [BPI+] ITU-T Recommendation J.125 (2004), *Link privacy for cable modem implementations*.
- [DOCSIS0-OSSI] ANSI/SCTE 22-3 2002, *Data-Over-Cable Service Interface Specification DOCSIS 1.0 Part 3: Operations Support System Interface (OSSI)*.
- [DOCSIS0-RFI] ANSI/SCTE 22-1 2002, *Data-Over-Cable Service Interface Specification DOCSIS 1.0 Radio Frequency Interface (RFI)*.
- [DOCSIS1-OSSI] ANSI/SCTE 23-3 2005, *DOCSIS 1.1 Part 3: Operations Support System Interface*.
- [DOCSIS1-RFI] ITU-T Recommendation J.112 Annex B (2004), *Data-over-cable service interface specifications: Radio-frequency interface specification*.
- [DOCSIS2-OSSI] ANSI/SCTE 79-2 2002, *DOCS 2.0 Operations Support System Interface*.
- [DOCSIS2-RFI] ITU-T Recommendation J.122 (2007), *Second-generation transmission systems for interactive cable television services – IP cable modems*.
- [DOCSIS3-MULPI] ITU-T Recommendation J.222.2 (2007), *Third-generation transmission systems for interactive cable television services – IP cable modems: MAC and upper layer protocols*.
- [DOCSIS3-OSSI] ANSI/SCTE 135-4 2007, *DOCSIS 3.0 Operations Support System Interface*.
- [DOCSIS-OSSI] Refers to [DOCSIS1-OSSI], [DOCSIS2-OSSI] and [DOCSIS3-OSSI].
- [DOCSIS-RFI/MULPI] Refers to [DOCSIS1-RFI], [DOCSIS2-RFI] and [DOCSIS3-MULPI].

- [DSG] ITU-T Recommendation J.128 (2005), *Set-top gateway specification for transmission systems for interactive cable television services.*
- [eRouter] ITU-T Recommendation J.218 (2007), *Cable modem IPv4 and IPv6 eRouter specification.*
- [IETF RFC 768] IETF RFC 768 (1980), *User Datagram Protocol.*
- [IETF RFC 791] IETF RFC 791 (1981), *Internet Protocol – DARPA Internet Program – Protocol Specification.*
- [IETF RFC 1493] IETF RFC 1493 (1993), *Definitions of Managed Objects for Bridges.*
- [IETF RFC 2011] IETF RFC 2011 (1996), *SNMPv2 Management Information Base for the Internet Protocol using SMIv2.*
- [IETF RFC 2131] IETF RFC 2131 (1997), *Dynamic Host Configuration Protocol.*
- [IETF RFC 2132] IETF RFC 2132 (1997), *DHCP Options and BOOTP Vendor Extensions.*
- [IETF RFC 2863] IETF RFC 2863 (2000), *The Interfaces Group MIB.*
- [IETF RFC 3396] IETF RFC 3396 (2002), *Encoding Long Options in the Dynamic Host Configuration Protocol (DHCPv4).*
- [IETF RFC 3418] IETF RFC 3418 (2002), *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP).*
- [IETF RFC 4293] IETF RFC 4293 (2006), *Management Information Base for the Internet Protocol (IP).*

2.2 Informative references

- [CableHome] CableLabs (<http://www.cablelabs.com/projects/cablehome/>).
- [CANN-DHCP] *CableLabs DHCP Options Registry*, CL-SP-CANN-DHCP-Reg-I01-070119, January 19, 2007, Cable Television Laboratories, Inc.
- [CCIF 2.0] *CableCARD™ Interface 2.0 Specification*, OC-SP-CCIF2.0-I09-070105, January 5, 2007, Cable Television Laboratories, Inc.
- [CH10] ITU-T Recommendation J.191 (2004), *IP feature package to enhance cable modems.*
- [CH11] ITU-T Recommendation J.192 (2004), *A residential gateway to support the delivery of cable data services.*
- [DOCSIS] CableLabs (<http://www.cablemodem.com/specifications/>).
- [HOST2.0] *OpenCable Host Device 2.0 Core Functional Requirements*, OC-SP-HOST2.0-CFR-I14-070615, June 15, 2007, Cable Television Laboratories, Inc.
- [IETF RFC 2578] IETF RFC 2578 (1999), *Structure of Management Information Version 2 (SMIv2).*
- [PacketCable] CableLabs (<http://www.packetcable.com/specifications/>).
- [PC10-MTA] ITU-T Recommendation J.167 (2001), *Media terminal adapter (MTA) device provisioning requirements for the delivery of real-time services over cable television networks using cable modems.*
- [PC10-MIB] ITU-T Recommendation J.166 (2001), *IPCablecom Management Information Base (MIB) framework.*

[TEI] ITU-T Recommendation J.214 (2007), *Cable modem TDM emulation interface*.

3 Definitions

This Recommendation defines the following terms:

3.1 CableHome: This is a Cable Television Laboratories, Inc. ("CableLabs") specification (see <http://www.cablelabs.com/projects/cablehome/>) for the interfaces necessary to extend high-quality cable-based services to network devices within the home. The CableHome project addresses issues such as device interoperability, QoS (Quality of Service), and network management. This term is also used for a system or device that is compliant with the CableHome specifications [CH10] or [CH11].

3.2 DOCSIS: The term for a system or device compliant with any one of the Cable Television Laboratories, Inc. ("CableLabs") series of specifications located at: <http://www.cablemodem.com/specifications/>.

3.3 CableModem base specifications: There are currently four versions of what are in this Recommendation referred to as the CableModem Base Specifications. The original CableModem is specified in SCTE 22-1 2002: *DOCSIS 1.0 Radio Frequency Interface* and SCTE 22-3. SCTE 22-1 is J.112 (1998) with some corrections and minor additions. The second specification is J.112 with SCTE 23-3, the third specification is J.122 with SCTE 79-2, and the fourth specification is J.222.

3.4 eCM: An eCM is an embedded Cable Modem, i.e., one that has been enhanced with the features of this Recommendation.

3.5 eDOCSIS: eDOCSIS is a CableLabs specification that defines the interface between the eCM and an eSAFE. The international version of the specification is this Recommendation.

3.6 eDOCSIS device: An eDOCSIS device is one that includes an eCM entity, one or more eSAFEs and supports a single software image using a CableModem secured software download mechanism.

3.7 embedded multimedia terminal adapter (eMTA): An embedded version of an MTA.

3.8 embedded MTA device (e-MTA): An eDOCSIS device that contains both an eMTA and an eCM.

3.9 embedded portal service element (ePS): An IPCable2Home-compliant eSAFE that provides management and network address translation functions between the cable data network and the home network.

3.10 logical CPE interface: A bidirectional, data-only 802.3/Ethernet MAC frame interface between eCM and an eSAFE.

3.11 multimedia terminal adapter (MTA): An IPCablecom device that contains the interface to a physical voice device, a network interface, CODECs, and all signalling and encapsulation functions required for VoIP transport, class features signalling and QoS signalling.

3.12 embedded service/application functional entity (eSAFE): An embedded version of a specified application, such as an IPCablecom multimedia terminal adapter (MTA), that provides a service using the CableModem IP platform, or a function or set of functions, such as the IPCable2Home Portal Service logical element, that supports the delivery of one or more services over an IP platform.

3.13 PacketCable: PacketCable is a CableLabs specification located at: <http://www.packetcable.com/specifications/>. The PacketCable specifications are interoperable interface specifications for delivering advanced, real-time multimedia services over a two-way cable plant. Built on top of the industry's highly successful cable modem infrastructure,

PacketCable networks uses Internet Protocol (IP) technology to enable a wide range of multimedia services, such as IP telephony, multimedia conferencing, interactive gaming, and general multimedia applications. This term is also applicable to a system or device that is compliant to the PacketCable specifications. The international version of PacketCable is standardized in ITU-T J-series Recommendations J.160 to J.179.

3.14 point of deployment (POD) module: A removable conditional access module which, when inserted into certain eSTB implementations, enables delivery of digital video programming and other services. POD module functionality includes copy protection and signal demodulation. The interface between the POD module and the eSTB is specified by [CCIF 2.0].

3.15 eRouter: DOCSIS Embedded Router. An eSAFE that is compliant with [eRouter], providing version 4 and/or version 6 Internet Protocol-formatted data forwarding, address configuration, and Domain Name services to Internet Protocol host devices connected to the cable modem in a customer's premises.

3.16 embedded set-top box (eSTB): An eSAFE that is compliant with [DSG], providing video, audio and data services. An example OpenCable-compliant eSTB is further specified in [HOST2.0].

3.17 embedded TDM emulator adapter (eTEA): An eSAFE that is compliant with [TEI], providing T1 and E1 Circuit transport over IP.

3.18 monolithic firmware image: A single firmware image containing one or more code images for the entire eDOCSIS device. For eDOCSIS devices, the Monolithic Firmware Image contains both the eCM code image as well as the applicable eSAFE code image. As an example for an eDOCSIS device containing an eSTB, the Monolithic Firmware Image contains the eCM code image as well as the eSTB code image (which may also be composed of multiple eSTB code images).

3.19 network virtual terminal (NVT): As defined in the Telnet Protocol. NVT was a bidirectional character device, representing characters as 7-bit ASCII codes, using an 8-bit field.

3.20 OpenCable host eSTB: An eSTB device built to CableLabs OpenCable Host specifications.

3.21 embedded security eSTB: An eSTB with integrated security functions.

3.22 secure microprocessor: The security element in a device that supports downloadable conditional access.

3.23 set-top device: An eDOCSIS device that contains an eSTB.

3.24 time division multiplexing (TDM): The means by which multiple digital signals can be carried on a single transmission path by interleaving portions of each signal in time.

4 Abbreviations and conventions

4.1 Abbreviations

This Recommendation uses the following abbreviations:

| | |
|-------|--|
| ASCII | American Standard Code for Information Interchange |
| BSoD | Business Services over DOCSIS |
| CATV | Community Access Television, Cable Television |
| CM | Cable Modem |
| CMCI | Cable Modem to Customer Premises Equipment Interface |
| CMTS | Cable Modem Termination System |

| | |
|---------|--|
| CVC | Code Verification Certificate |
| DHCP | Dynamic Host Configuration Protocol |
| DIX | Digital Intel Xerox |
| DNS | Domain Name Server |
| DOCSIS | Data-Over-Cable Service Interface Specifications |
| DSG | DOCSIS Set-top Gateway |
| eCM | embedded Cable Modem |
| eDOCSIS | embedded DOCSIS |
| eMTA | embedded Multimedia Terminal Adapter |
| ePS | embedded Portal Services Element |
| eSTB | embedded Set-Top Box |
| eSAFE | embedded Service/Application Functional Entity |
| eTEA | embedded T1/E1 TDM Emulation Adapter (TEA) |
| FCS | Frame Check Sequence |
| FQDN | Fully Qualified Domain Name |
| IP | Internet Protocol |
| LCI | Logical CPE Interface |
| LLC | Logical Link Control |
| MAC | Media Access Control |
| MIC | Message Integrity Check |
| MTA | Multimedia Terminal Adapter |
| OCAP | Open Cable Application Platform |
| PS | Portal Service |
| RF | Radio Frequency |
| ROM | Read-Only Memory |
| SAFE | Service/Application Functional Entity |
| SLED | Software Loopback for eDOCSIS |
| SNMP | Simple Network Management Protocol |
| SW | Software |
| SSD | Secure Software Download |
| TDM | Time Division Multiplexing |
| TFTP | Trivial File Transfer Protocol |
| TLV | Type/Length/Value |
| UDP | User Datagram Protocol |
| UPS | Uninterrupted Power Supply |
| USB | Universal Serial Bus |
| WAN | Wide Area Network |

4.2 Conventions

Throughout this Recommendation, the words that are used to define the significance of particular requirements are capitalized. These words are:

| | |
|------------|---|
| MUST | This word or the adjective "REQUIRED" means that the item is an absolute requirement of this Recommendation. |
| MUST NOT | This phrase means that the item is an absolute prohibition of this Recommendation. |
| SHOULD | This word or the adjective "RECOMMENDED" means that there may exist valid reasons in particular circumstances to ignore this item, but the full implications should be understood and the case carefully weighed before choosing a different course. |
| SHOULD NOT | This phrase means that there may exist valid reasons in particular circumstances when the listed behaviour is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behaviour described with this label. |
| MAY | This word or the adjective "OPTIONAL" means that this item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because it enhances the product; for example, another vendor may omit the same item. |

5 Embedded cable modem

The goals for this Recommendation are:

- to preserve functional separation of the Cable Modem entity from eSAFEs within the eDOCSIS device, so that existing cable plant integrity, cable modem configuration, management and provisioning security are not compromised;
- to isolate Cable Modem functionality so that compliance can be tested for the eCM component independent of eSAFEs;
- to enable the service provider to enable or disable forwarding traffic between each eSAFE and the eCM within the eDOCSIS Device;
- to maximize compatibility with existing back-office management/provisioning infrastructure so that new services enabled by eDOCSIS devices can be deployed rapidly;
- to architect eDOCSIS devices in such a way as to scale to new services and applications, and to take advantage of technology innovations to achieve low cost and high functionalities.

5.1 Device interface reference model

As per Figure 5-1, an eDOCSIS device consists of an embedded cable modem (eCM) and one or more embedded service/application functional entities (eSAFEs). An eDOCSIS device may also have one or more physically exposed interfaces. In addition, only a single secured software image download is used for the entire eDOCSIS device.

eSAFEs include:

- ePS: embedded IPCable2Home portal services logical element [CH10].
- eMTA: embedded IPCablecom multimedia terminal adapter [PC10-MTA] and [PC10-MIB].

- eSTB: embedded set-top box: An eSAFE that is compliant with [DSG], providing video, audio and data services. An example OpenCable-compliant eSTB is further specified in [HOST2.0].
- eTEA: embedded T1/E1 TDM Emulation Adapter (eTEA) [TEI].
- eRouter: An eSAFE that is compliant with [eRouter], providing Internet Protocol data forwarding, address configuration, and Domain Name services.

Within an eDOCSIS device, each eSAFE interfaces to the eCM via a point-to-point logical CPE interface.

Figure 5-2 presents a typical IPCable2Home Home Access eDOCSIS Device reference model.

Figure 5-3 presents a logical view of protocol stacks for an eCM to ePS interface.

Figure 5-4 presents a typical IPCablecom E-MTA eDOCSIS Device reference model.

Figure 5-5 presents a logical view of protocol stacks for an eCM to eMTA interface.

Figure 5-6 presents a typical OpenCable Host 2.0 eDOCSIS Device reference model where the Host provides DSG Flow, Socket Flow and IP Flow support.

NOTE 1 – The POD module may support Socket Flow or IP Flow or both, but neither is supported simultaneously by the Host.

Figure 5-7 presents a logical view of protocol stacks for an OpenCable Host 2.0 eCM to eSTB to POD module interface where the Host provides DSG Flow, Socket Flow and IP Flow support.

NOTE 2 – The POD module may support Socket Flow or IP Flow or both, but neither is supported simultaneously by the Host.

Figure 5-8 presents a typical embedded security STB eDOCSIS device reference model.

Figure 5-9 presents a logical view of protocol stacks for an eCM to eSTB interface (embedded security STB).

Figure 5-10 presents a typical T1/E1 TDM emulation adapter (TEA) eDOCSIS device reference model.

Figure 5-11 presents a logical view of protocol stacks for an eCM to eTEA interface (embedded TDM emulation adapter).

Figure 5-12 presents a typical DOCSIS eRouter eDOCSIS device reference model.

Figure 5-13 presents a logical view of protocol stacks for an eCM to IPv4 eRouter interface.

Figure 5-14 presents a logical view of protocol stacks for an eCM to IPv6 eRouter interface.

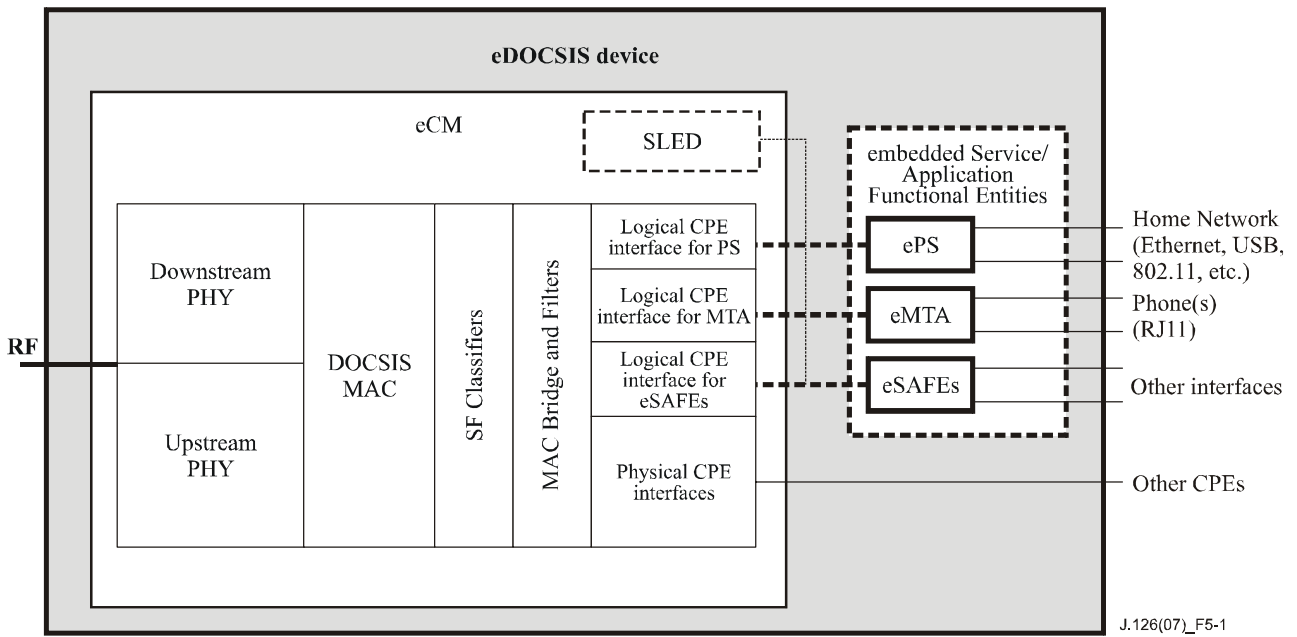


Figure 5-1 – eDOCSIS reference model

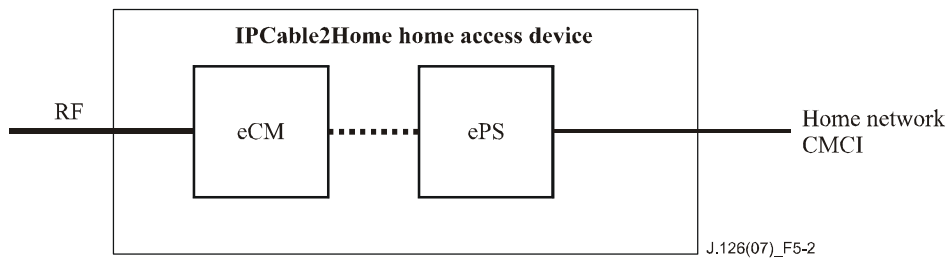


Figure 5-2 – IPCable2Home home access eDOCSIS device reference model

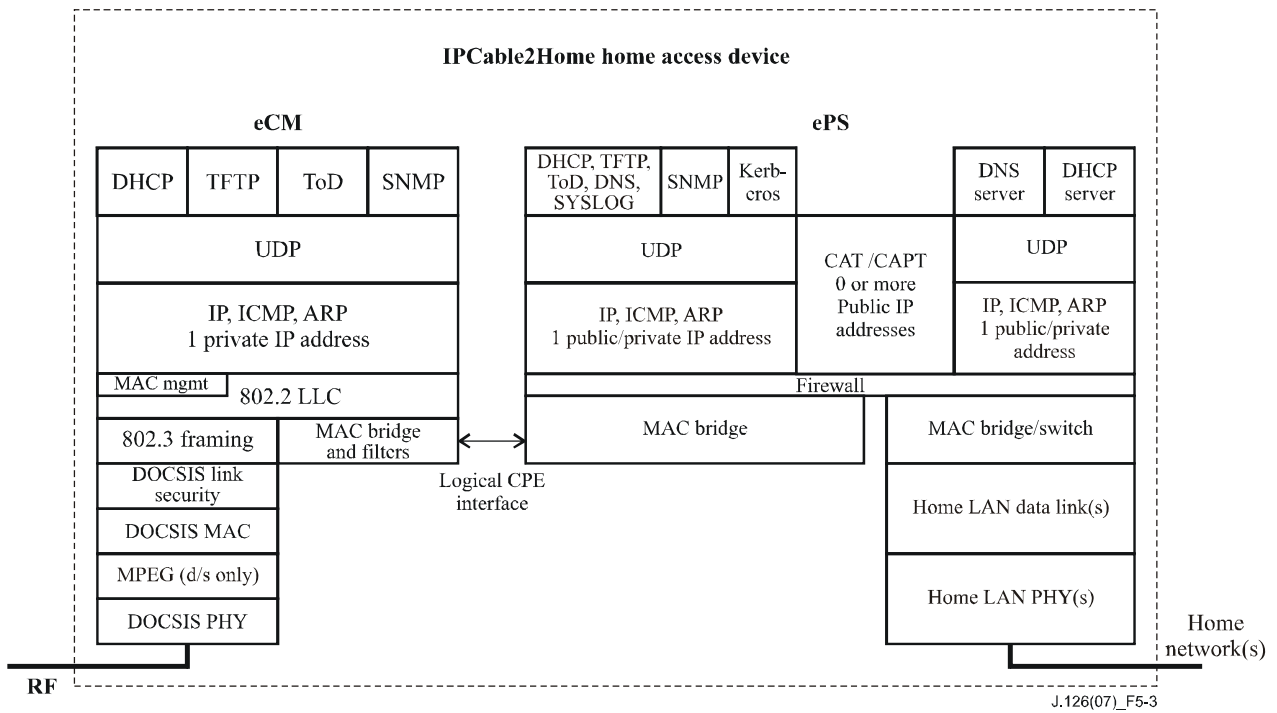
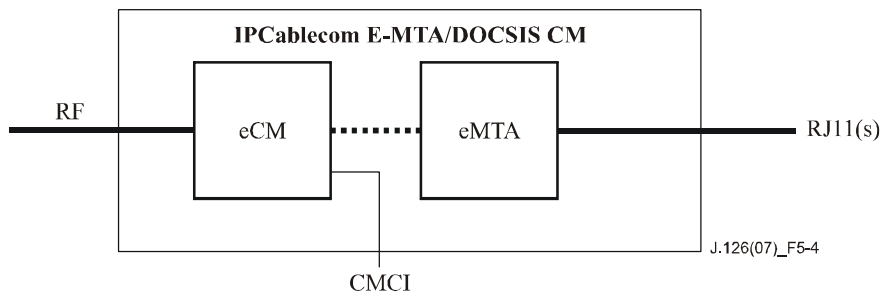


Figure 5-3 – eCM-ePS protocol stacks



**Figure 5-4 – IP-Cablecom E-MTA (with DOCSIS CM)
eDOCSIS reference model**

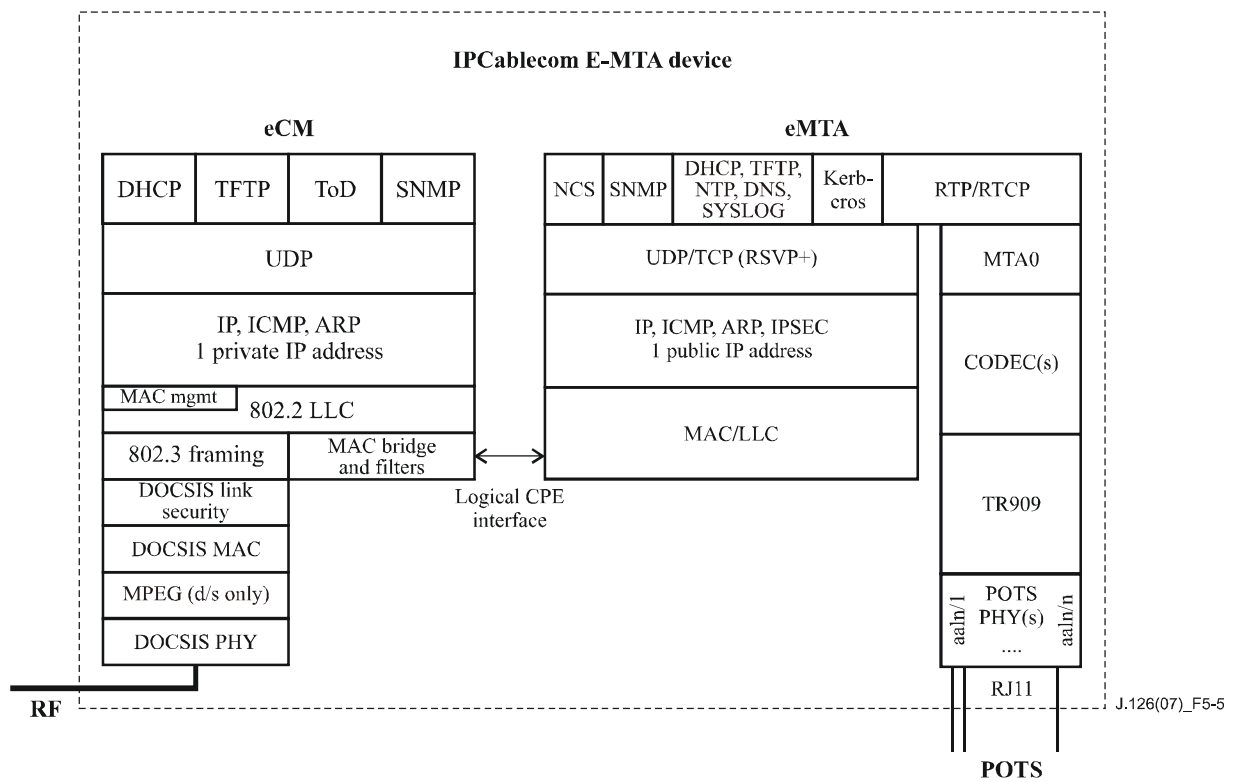


Figure 5-5 – eCM-eMTA protocol stacks

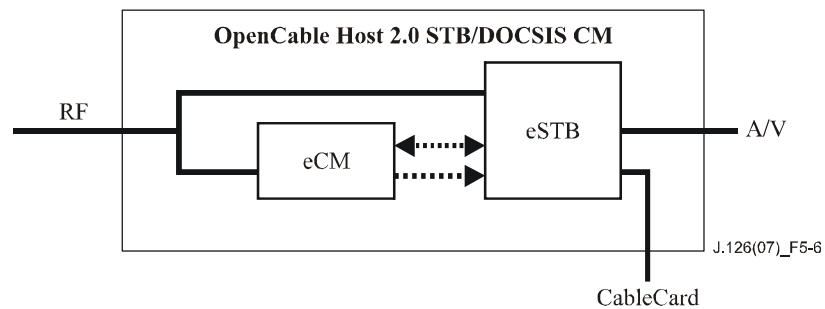


Figure 5-6 – OpenCable Host 2.0 eDOCSIS reference model

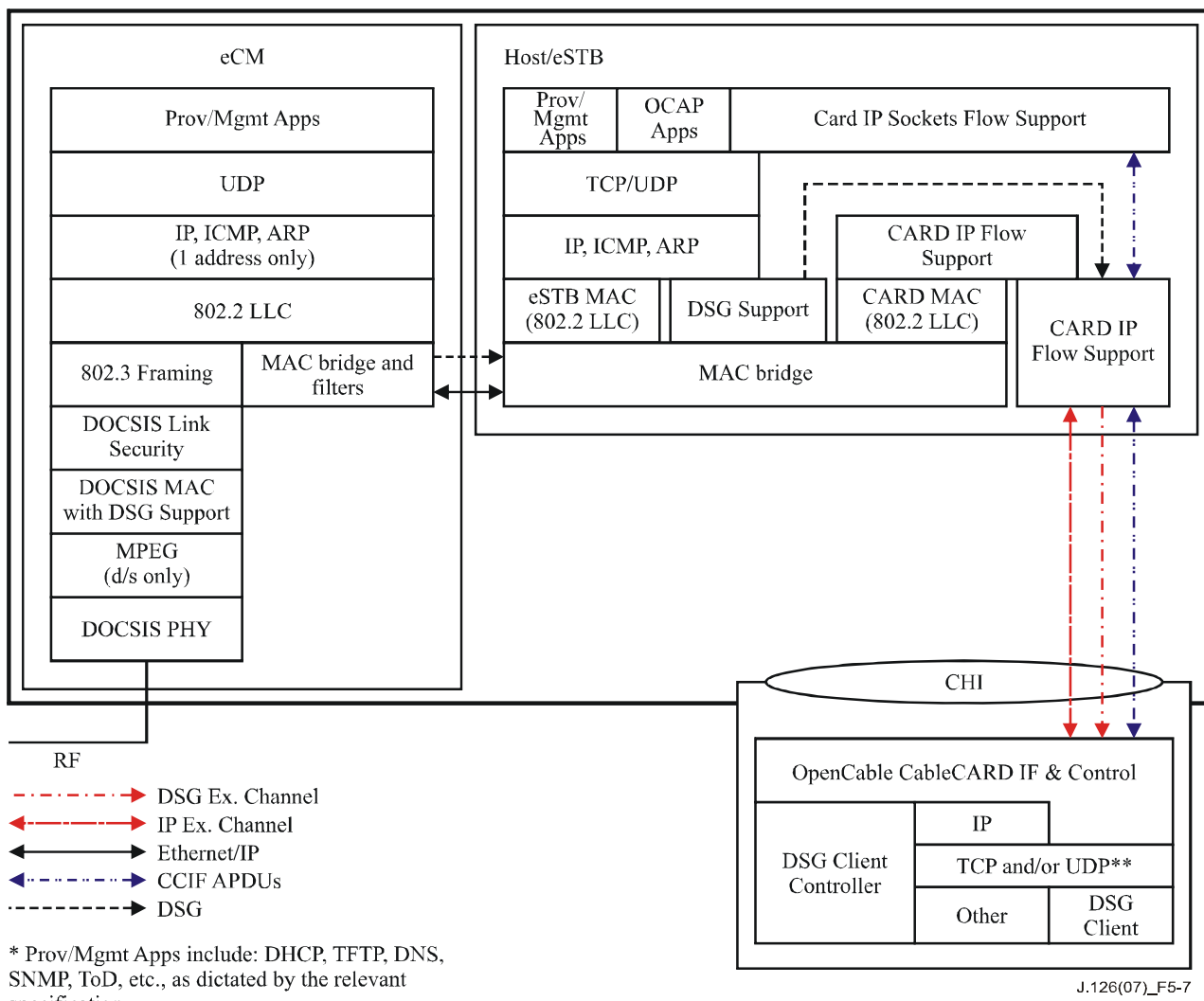


Figure 5-7 – eCM-eSTB protocol stacks – OpenCable Host 2.0 – Unicast IP flow

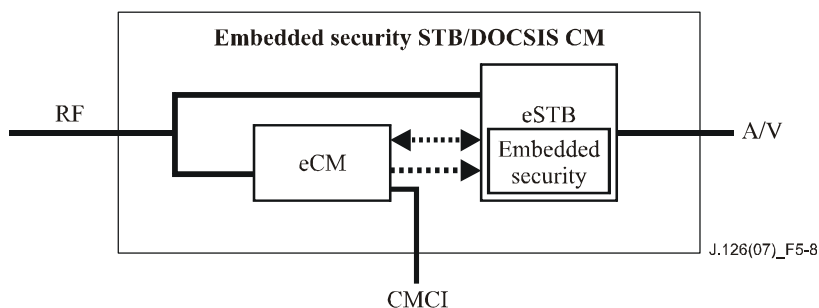


Figure 5-8 – Embedded security STB eDOCSIS reference model

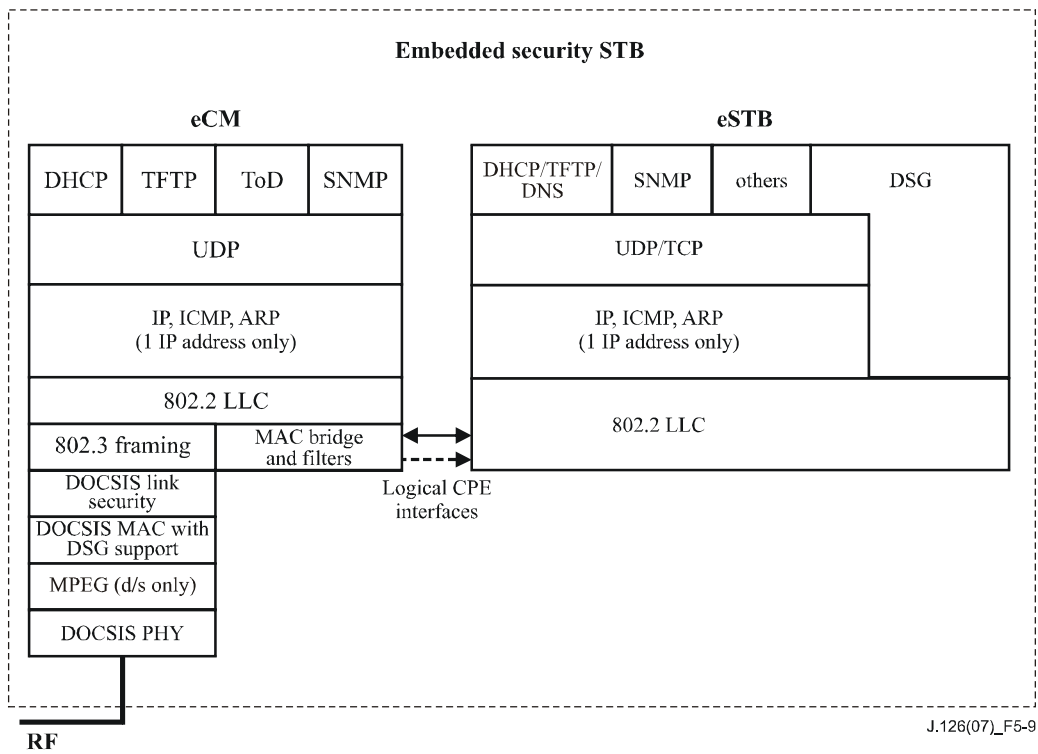


Figure 5-9 – eCM-eSTB protocol stacks – Embedded security STB

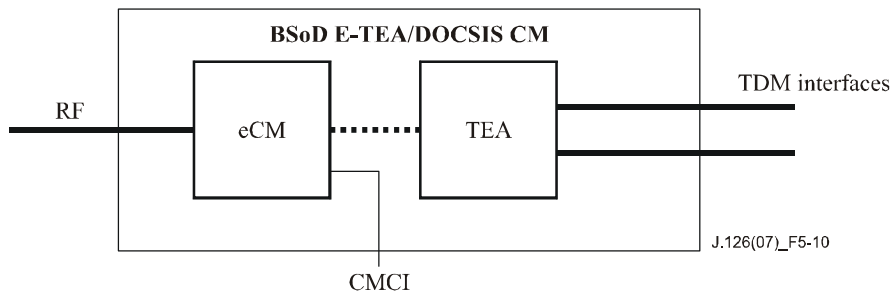
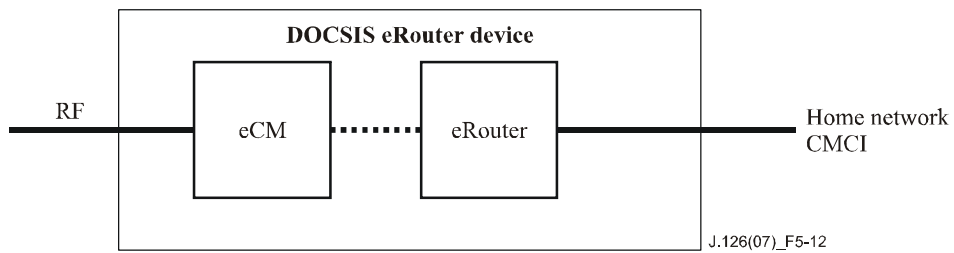
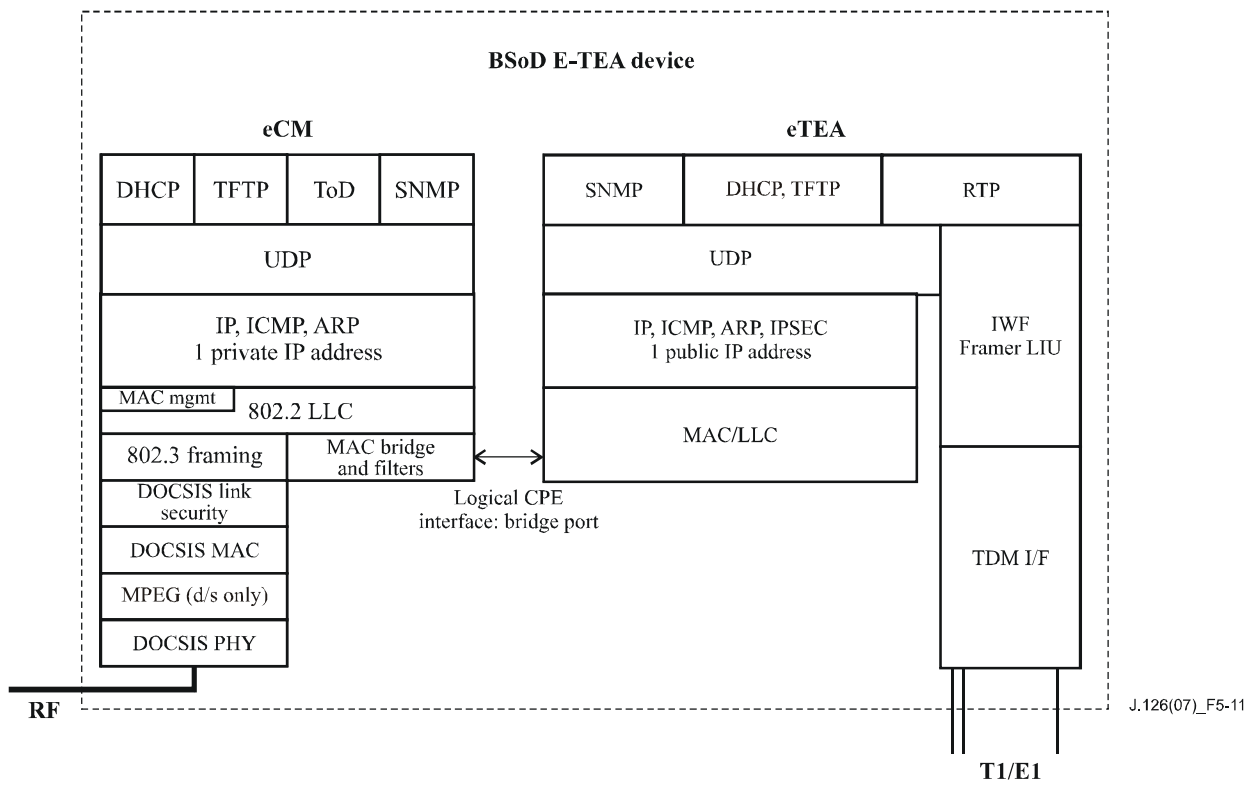


Figure 5-10 – BSoD eTEA (with DOCSIS CM) eDOCSIS reference model



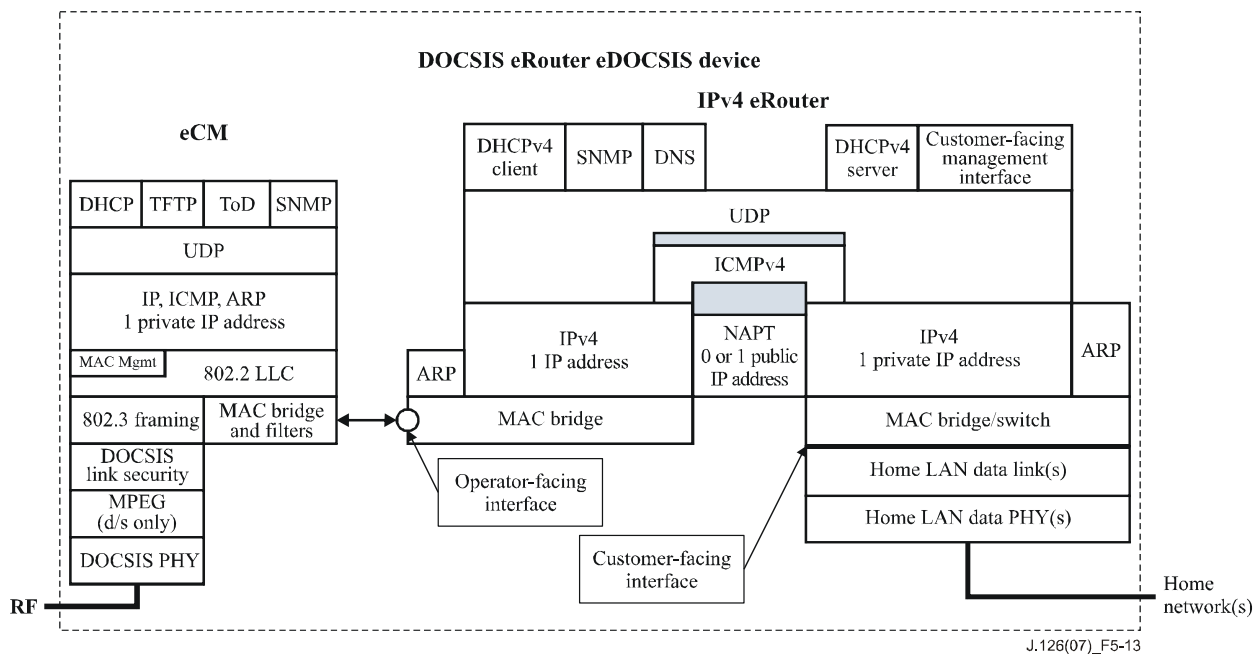


Figure 5-13 – DOCSIS IPv4 eRouter eDOCSIS device reference model

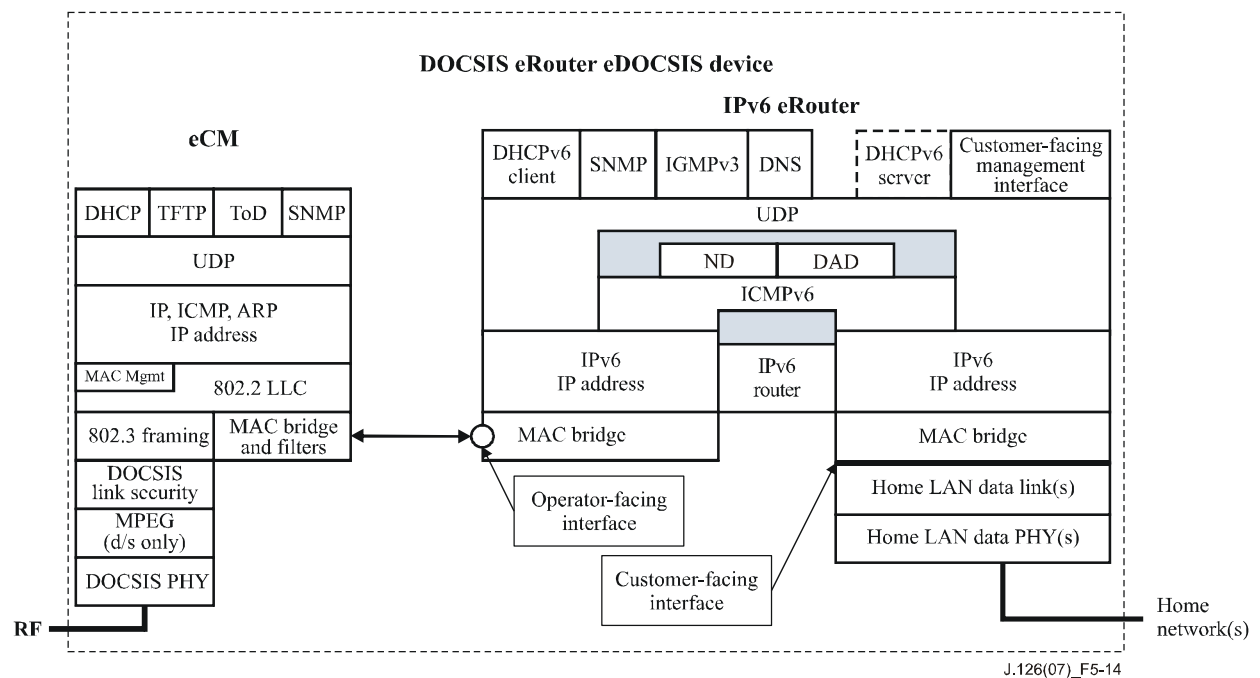


Figure 5-14 – DOCSIS IPv6 eRouter eDOCSIS device reference model

5.2 eDOCSIS requirements

5.2.1 General requirements

The eCM will provide an SNMP agent which is logically separated from any SNMP agent provided by an eSAFE.

Except for MIB objects that are explicitly allowed to be shared, the CableModem-specified MIBs MUST only be accessible through the Management IP address of the eCM. The eSAFE-specified MIBs MUST NOT be accessible through the Management IP address of the eCM.

The MIB objects that MAY be shared are:

- An eDOCSIS device MAY share MIB objects from [IETF RFC 3418]. As an exception, the eDOCSIS device MUST share the sysDescr MIB object¹ and report its value as defined in DOCSIS OSS.
- An eDOCSIS device MAY share MIB objects from UDP-MIB and IP-MIB. However, the eDOCSIS device MUST not share MIB objects from IP-MIB that define per-interface management information (e.g., ipNetToMediaTable or IpNetToPhysicalTable; see clause 5.2.3.2).
- An eDOCSIS device MAY share MIB objects rooted under snmpV2 [IETF RFC 2578].

The eCM MUST act as an entity distinct from, and MUST have logical CPE interfaces to, the eSAFEs.

All messages coming from the DOCSIS data network (labelled RF in the diagrams) destined for eSAFEs MUST be processed through the eCM first.

The eCM MUST be the only interface to the DOCSIS data network.

In addition, only a single secured software image download MUST be used for the entire eDOCSIS device and it MUST be controlled by the eCM.

For an eDOCSIS device containing an eSTB, the eCM MUST implement DSG client support functionality including one-way DOCSIS and DCD MAC message as specified in DSG.

An eDOCSIS device MUST NOT implement both an ePS and an eRouter simultaneously.

An eCM MUST meet the requirements of an equivalent stand-alone cable modem as specified in the applicable CableModem Base Specifications. In case any requirement in this Recommendation conflicts with a requirement in the CableModem Base Specifications, the requirement in this Recommendation takes precedence for any eDOCSIS device.

5.2.2 Interface requirements

The bridging function between RF port and the CPE interfaces (logical or physical) MUST be equivalent to that of a multi-port learning bridge. Each CPE interface MUST comply with the CM Forwarding Rules defined in [DOCSIS-RFI]. In particular:

- MAC addresses of each eSAFE MUST be counted toward the total allowed by the Maximum Number of CPEs configuration setting.
Note that an OpenCable Host STB, where the POD module opens an IP Flow, will have at least 2 MAC addresses, one for the eSTB and one for the POD module. In this case, both MAC Addresses are counted toward the total allowed by the Maximum Number of CPEs configuration setting.
- The DSG tunnel MAC addresses associated with ifIndex=18 MUST NOT be counted in the MaxCPE count of the eCM configuration file.
- Packet forwarding and filtering rules defined in [DOCSIS-RFI] specification MUST apply to both logical and physical interfaces to eSAFE as defined in this Recommendation and in [DOCSIS-OSSI].

¹ In order to associate one unique image to an eDOCSIS device, the sysDescr MIB value is shared among eCM and eSAFEs. This is defined with the purpose of providing a mechanism to properly associate a firmware image with the eDevice vendor name and hardware model.

- Data forwarding through the interfaces to all eSAFEs with the exception of the interface to the DSG Client (ifIndex 18) MUST obey the Network Access Control Object as defined in [DOCSIS-RFI]. NACO state does not affect the forwarding of DSG traffic (through ifIndex 18) in an eDOCSIS device containing an eSTB.

5.2.3 Operations support requirements

5.2.3.1 ifTable requirements

The eCM MUST represent the logical interface to each eSAFE with an entry in the ifTable with ifType other(1) as described in [DOCSIS-OSSI] and as detailed below.

If the eCM is embedded into a device which contains an active (i.e., not disabled) ePS, or an eRouter, then:

- the eCM MUST use ifIndex 1 (the Primary CPE interface) to represent the logical interface between the eCM and the ePS or between the eCM and the eRouter;
- the eCM MUST NOT report in the ifTable the physically exposed interfaces associated with the ePS or with the eRouter;
- the eCM MUST NOT report the MIB Module extensions associated with ePS or eRouter interfaces exposed to customer premises (e.g., EtherLike-MIB and USB-MIB).

If the eCM is embedded into a device that contains an eRouter that is disabled, the eCM MUST report in the eCM's ifTable the physical CPE interfaces that would be associated with the eRouter if the eRouter was not disabled.

NOTE – The ifIndex range allowed for CPE interfaces is described in Table 5-1.

If the eCM is embedded into a device which contains an eMTA, then:

- the eCM MUST use ifIndex 16 to represent the logical interface between the eCM and the eMTA;
- the eCM MUST NOT report in the ifTable the MTA endpoints (ifType = 198).

If the CM is embedded into a device which contains an eSTB, then:

- The eCM MUST use ifIndex 17 to represent the logical interface between the eCM and the eSTB for the interactive IP traffic.
- The eCM MUST use ifIndex 18 to represent the logical interface between the eCM and the eSTB for the one-way DSG tunnel traffic.
- The eCM MUST NOT report in the ifTable any other interfaces on the eSTB (such as POD module, DSG Clients, and A/V interfaces, etc.) which are not directly and physically connected to the eCM.

If the eCM is embedded into a device which contains an eTEA, then:

- The eCM MUST use ifIndex 19 to represent the logical interface between the eCM and the eTEA.
- The eCM MUST NOT report in the ifTable the eTEA interfaces (ifType = ds1(18), ds0Bundle(82), etc.).

The ifXTable MUST be supported in accordance with [IETF RFC 2863]. The default value of ifLinkUpDownTrapEnable MUST be enabled(1) for logical interfaces to eSAFEs.

The ifStackTable MUST be supported in accordance with [IETF RFC 2863]. The logical interface to an eSAFE MUST NOT contain any sub-layers.

Table 5-1 summarizes the eCM assignment of ifIndexes to its connected interfaces. Table 5-2 defines the details of the ifTable entries that MUST be supported by ePS, eRouter, eMTA, eSTB, and eTEA.

Table 5-1 – eDOCSIS ifTable interface designations

| Interface | Type |
|------------------|---|
| 1 | Primary CPE interface (IPCable2Home ePS WAN interface or eRouter Operator-Facing Interface, when eRouter is enable) |
| 2 | CATV-MAC |
| 3 | RF-downstream channel |
| 4 | RF-upstream channel |
| 5-15 | Other CPE interfaces |
| 16 | Reserved for IPCablecom/eMTA |
| 17 | Reserved for eSTB-IP |
| 18 | Reserved for eSTB-DSG |
| 19 | Reserved for eTEA |
| 20-31 | Reserved for other eDOCSIS interfaces |

An eDOCSIS compliant eCM can have zero, one or multiple CPE interfaces, as well as interfaces to one or multiple eSAFEs. When multiple CPE interfaces are present, if filter(s) (IP, LLC, or NmAccess) are applied to the eCM's "Primary CPE Interface" (ifIndex 1), the eCM MUST also apply the same filter(s) to its "Other CPE Interfaces" (ifIndexes 5 through 15). Moreover, filters are never used to limit traffic between the CPE interfaces ("Primary CPE Interface" and "Other CPE Interfaces") within the eCM. However, if IP, LLC, or NmAccess filters are applied to the eCM's "Primary CPE Interface" (ifIndex 1), the eCM MUST NOT apply these filters to ifIndex 16 through 31, which are reserved as interfaces to eSAFEs.

The above defined mechanism provides granular, independent control of filters applied to the CPE interfaces versus those applied to the interface to each eSAFE. In addition, the eCM MUST have the ability to filter traffic at a particular interface to an eSAFE, regardless of the origination point of that traffic. This granular filter control provides the ability for the eCM to filter traffic sourced by one eSAFE that is destined to another eSAFE within the same device.

Table 5-2 – [IETF RFC 2863] ifTable, MIB-Object details for eDOCSIS device interfaces

| RFC 2863 MIB-Object details for eCM-eSAFE Interfaces | ePS or eRouter | eMTA | eSTB | | eTEA |
|---|---|----------------------------------|-------------------------------------|--------------------------------------|-------------------------------|
| | | | eSTB-IP | eSTB-DSG | |
| ifIndex | 1 | 16 | 17 | 18 | 19 |
| ifDescr: MUST match the text | "CableHome Embedded Interface" for the ePS, or "eRouter Embedded Interface" for the eRouter | "PacketCable Embedded Interface" | "Set-Top Box Embedded IP Interface" | "Set-Top Box Embedded DSG Interface" | "BSoD Embedded TEA Interface" |
| ifType | other(1) | other(1) | other (1) | other (1) | other(1) |
| ifMtu | 0 | 0 | 0 | 0 | 0 |
| ifSpeed | 0 | 0 | 0 | 0 | 0 |
| ifPhysAddress | <empty-string> | <empty-string> | <empty-string> | <empty-string> | <empty-string> |
| ifAdminStatus: Only up/down controls are required for this interface. Other values are optional | up(1), down(2) | up(1), down(2) | up(1), down(2) | up(1), down(2) | up(1), down(2) |
| ifOperStatus: Only up/down controls are required for this interface. Other values are optional | up(1), down(2) | up(1), down(2) | up(1), down(2) | up(1), down(2) | up(1), down(2) |
| ifLastChange | <per RFC 2863> | <per RFC 2863> | <per RFC 2863> | <per RFC 2863> | <per RFC 2863> |
| ifInOctets | (n) | (n) | (n) | Deprecated | (n) |
| ifInUCastPkts | (n) | (n) | (n) | Deprecated | (n) |
| ifInNUCastPkts | Deprecated | Deprecated | Deprecated | Deprecated | Deprecated |
| ifInDiscards | 0 | 0 | 0 | 0 | 0 |
| ifInErrors | 0 | 0 | 0 | 0 | 0 |
| ifUnknownProtos | 0 | 0 | 0 | 0 | 0 |
| ifOutOctets | (n) | (n) | (n) | (n) | (n) |
| ifOutUCastPkts | (n) | (n) | (n) | (n) | (n) |
| ifOutNUCastPkts | Deprecated | Deprecated | Deprecated | Deprecated | Deprecated |
| ifOutDiscards | 0 | 0 | 0 | 0 | 0 |
| ifOutErrors | 0 | 0 | 0 | 0 | 0 |
| ifOutQlen | Deprecated | Deprecated | Deprecated | Deprecated | Deprecated |
| ifSpecific | Deprecated | Deprecated | Deprecated | Deprecated | Deprecated |

5.2.3.2 ipNetToMediaTable requirements

Table 5-3 shows the details of the ipNetToMediaTable entries that MUST be supported by an ePS and eMTA.

**Table 5-3 – [IETF RFC 2011] ipNetToMedia MIB-Object details
for eDOCSIS device interfaces**

| RFC 2011 MIB-Object details for eCM-eSAFE interfaces | ePS | eMTA | eSTB-IP | eTEA |
|--|---|---|--|--|
| ipNetToMediaIfIndex | 1 | 16 | 17 | 19 |
| ipNetToMediaPhysAddress | WAN-Man MAC Address | MTA MAC Address | STB MAC Address | TEA MAC Address |
| ipNetToMediaNetAddress | WAN-Man Address, if acquired; otherwise 0.0.0.0 | MTA Address, if acquired; otherwise 0.0.0.0 | STB IP Address, if acquired; otherwise 0.0.0.0 | TEA IP Address, if acquired; otherwise 0.0.0.0 |
| ipNetToMediaType | static(4) | static(4) | static(4) | static(4) |

If implemented in an eDOCSIS device, eSAFE elements ePS, eMTA, eSTB, and/or eTEA SHOULD support the ipNetToPhysicalTable [IETF RFC 4293] as described in Table 5-4. The eRouter MUST support the ipNetToPhysicalTable [IETF RFC 4293] as described in Table 5-4.

Table 5-4 – ipNetToPhysical MIB-Object details for eDOCSIS device interfaces

| MIB Object Name | ePS or eRouter | eMTA | eSTB-IP | eTEA |
|-------------------------------|--|---|---|---|
| ipNetToPhysicalIfIndex | 1 | 16 | 17 | 19 |
| ipNetToPhysicalPhysAddress | WAN-Man MAC Address for the ePS or operator-facing interface MAC Address for the eRouter | MTA MAC Address | STB MAC Address | TEA MAC Address |
| ipNetToPhysicalNetAddressType | ipv4(1) or ipv6(2), as applicable | ipv4(1) or ipv6(2), as applicable | ipv4(1) or ipv6(2), as applicable | ipv4(1) or ipv6(2), as applicable |
| ipNetToPhysicalNetAddress | WAN-Man IP Address for the ePS, or eRouter operator-facing interface IP address, if acquired; otherwise a zero-length string | MTA IP Address, if acquired; otherwise a zero-length string | STB IP Address, if acquired; otherwise a zero-length string | TEA IP Address, if acquired; otherwise a zero-length string |
| ipNetToPhysicalLastUpdated | <refer to RFC 4293> | <refer to RFC 4293> | <refer to RFC 4293> | <refer to RFC 4293> |
| ipNetToPhysicalType | static(4) | static(4) | static(4) | static(4) |
| ipNetToPhysicalState | <refer to RFC 4293> | <refer to RFC 4293> | <refer to RFC 4293> | <refer to RFC 4293> |
| ipNetToPhysicalRowStatus | 'active' | 'active' | 'active' | 'active' |

5.2.3.3 IETF RFC 1493 requirements

The eCM MUST add ports associated with eSAFEs to its [IETF RFC 1493] dot1dBasePortTable.

The eCM MUST support all bridge statistics of the [IETF RFC 1493] dot1dTpPortTable for all ports associated with eSAFEs.

The eCM MUST create a row entry in its dot1dTpFdbTable for each active eSAFE MAC address within the eDOCSIS device. Note that an eSAFE could have more than one active MAC address. In the case of an eCM embedded into a device that contains an OpenCable Host, both the Host and the POD module could have active MAC addresses.

Within each dot1dTpFdbTable entry for an eSAFE, the dot1dTpFdbAddress value MUST be the corresponding eSAFE MAC address, the dot1dTpFdbPort value MUST be the port associated with the ifIndex of that eSAFE from the dot1dBasePortTable, and the dot1dTpFdbStatus value MUST be mgmt(5). The eCM MUST prevent row entries for eSAFEs from being aged-out or overwritten.

5.2.3.3.1 IETF RFC 1493 requirements for the eSTB

An eDOCSIS device containing an eSTB MUST implement the objects from the [IETF RFC 1493] dot1dStaticTable with a MAX-ACCESS of read-only.

In an eDOCSIS device containing an eSTB, the eCM will acquire the DSG MAC addresses from the eSTB in an implementation-dependent manner. For each DSG MAC address acquired by the eCM, the eCM MUST create a corresponding row entry in the dot1dStaticTable.

For a dot1dStaticTable row entry for a DSG MAC address, the dot1dStaticAddress value MUST be the DSG MAC address, the dot1dStaticReceivePort value MUST be the port associated with ifIndex 2 (CATV-MAC interface) from the dot1dBasePortTable, the dot1dStaticAllowedToGoTo value MUST be the port associated with the ifIndex 18 (interface to the DSG Client of the eSTB) from the dot1dBasePortTable, and the dot1dStaticStatus value MUST be deleteOnReset(4).

Table 5-5 – dot1dStaticTable entries for DSG MAC addresses

| MIB object | Value |
|--------------------------|---|
| dot1dStaticAddress | DSG MAC Address |
| dot1dStaticReceivePort | port associated with ifIndex 2 in dot1dBasePortTable |
| dot1dStaticAllowedToGoTo | port associated with ifIndex 18 in dot1dBasePortTable |
| dot1dStaticStatus | deleteOnReset(4) |

Each row entry in the dot1dStaticTable describes a bridging rule in which the eCM only bridges a frame with a destination address of the value of dot1dStaticAddress – that is received from the port represented by the value of dot1dStaticReceivePort – if the frame is to be bridged out a port included in the value of dot1dStaticAllowedToGoTo.

An eCM MUST NOT bridge a frame having a destination address equal to an active DSG MAC address if that frame was received from any port other than the one associated with ifIndex 2 (CATV-MAC).

An eCM MUST NOT bridge a frame having a destination address equal to an active DSG MAC address to any port other than the one associated with ifIndex 18 (the interface to the DSG Client of the eSTB).

If a CPE MAC address is acquired by the eCM via the eCM Configuration File or via the address learning process, and the eCM is later informed that the same address is a DSG tunnel MAC address, the eCM SHOULD remove the DSG tunnel MAC address from its list of acquired CPE MAC addresses.

Also, the eCM SHOULD NOT populate a CPE MAC address from the eCM Configuration File into its list of acquired CPE MAC addresses if that MAC address matches that of a DSG tunnel MAC address already established via DSG operation.

5.2.3.3.2 IETF RFC 1493 requirements for the eTEA

An eCM MUST NOT bridge a frame having a destination address equal to an active eTEA MAC address if that frame was received from any port other than the one associated with ifIndex 19.

An eCM MUST NOT bridge a frame having a destination address equal to an active eTEA MAC address to any port other than the one associated with ifIndex 19 (the interface to the eTEA).

If a CPE MAC address is acquired by the eCM via the eCM Configuration File or via the address learning process, and the eCM is later informed that the same address is an eTEA MAC address, the eCM SHOULD remove the eTEA MAC address from its list of acquired CPE MAC addresses.

5.2.3.4 Battery backup UPS MIB requirements

eSAFE specifications can require support for the Battery Backup UPS MIB, if Battery Backup is supported in the containing eDOCSIS device. For more information, please refer to the Battery Backup MIB specification [BB].

For an eDOCSIS device meeting the requirements specified in [BB], the eCM MUST implement the Battery Backup UPS MIB specified in [BB].

5.2.4 DHCP option 43 syntax requirements

In order to facilitate device provisioning, all eDOCSIS devices operating with IPv4 will use DHCP [IETF RFC 2131] Option 43 during registration process for providing vendor class identification, embedded component and vendor specific capability enumerations. Requirements in this clause apply only to cable modems operating with IPv4 and do not apply to cable modems not operating with an IPv4 protocol stack, such as a DOCSIS 3.0 or greater cable modem operating with only an IPv6 protocol stack.

5.2.4.1 General requirements

Each eCM DHCP DISCOVER and DHCP REQUEST MUST use option 43 and its sub-options 2 through 10 for vendor-specific information to identify embedded components as specified in Table 5-6.

Similarly, each eSAFE MAY issue its own DHCP DISCOVER and DHCP REQUEST with Option 43 after eCM has been successfully registered and operational; details are specified in each eSAFE's specification.

5.2.4.2 DHCP option 43 syntax

DHCP option 43 provides device specific information through the use of sub-options. Sub-options 1 through 10 are specified by this Recommendation, sub-options 11-127 are reserved for future definition, and sub-options 128 and above are reserved for vendor use.

The eCM MUST implement the vendor-specific information option (DHCP option 43) as specified in Table 5-6. Details of DHCP option 43 and its sub-options for eDOCSIS are further defined below. The definitions of DHCP option 43 sub-options MUST conform to requirements imposed by [IETF RFC 2132].

The option begins with a type octet with the value of number 43, followed by a length octet. The length octet is followed by the number of octets of data equal to the value of the length octet. The value of the length octet does not include the two octets specifying the tag and length.

DHCP option 43 in eDOCSIS is a compound option. The content of option 43 is composed of one or more sub-options. Supported DHCP option 43 sub-options in eDOCSIS is in the range 1-254. A sub-option begins with a tag octet containing the sub-option code, followed by a length octet which indicates the total number of octets of data. The value of the length octet does not include itself or the tag octet. The length octet is followed by "length" octets of sub-option data.

5.2.4.3 DHCP option 43 sub-option encoding

The encoding of each option 43 sub-option is defined below. See Table 5-6 for the intended purpose of each sub-option.

The eCM MAY include option 43 sub-option 1 in DHCP DISCOVER and DHCP REQUEST messages. If DHCP option 43 sub-option 1 is included in these DHCP client messages, the eCM MUST encode this sub-option by the number of octets equal to the value of the length octet of this

sub-option, with each octet codifying a requested sub-option. If the length octet of this sub-option is 0 (because there are no requested sub-options), this sub-option SHOULD be omitted from DHCP option 43. The eCM MUST encode each of the DHCP option 43 sub-options 2, 3, 4, 5, 6, 7, 8, 9 and 10 as a character string consisting of characters from the NVT ASCII character set, with no terminating NULL.

An eCM MUST send DHCP option 43 sub-option 2 containing the character string "ECM" (without the quotation marks).

An eCM MUST send DHCP option 43 sub-option 3 containing a colon-separated list of all eSAFE types in the eDOCSIS device, including at a minimum the colon-separated character string "ECM:<eSAFE>" (without the quotation marks).

Defined eSAFEs are: "EPS" for IPCable2Home embedded Portal Services Element, "EMTA" for IPCablecom embedded MTA, "ESTB" for embedded set-top box, and "ETEA" for embedded TDM emulation adapter.

An eCM MUST send DHCP option 43 sub-option 4 containing the device serial number as in MIB object docsDevSerialNumber.

An eCM MUST send DHCP option 43 sub-option 5 containing the Hardware version number, identical to the value as reported in <Hardware version> field in MIB object sysDescr.

An eCM MUST send DHCP option 43 sub-option 6 containing the Software version number, identical to the value as reported in <Software version> field in MIB object sysDescr.

An eCM MUST send DHCP option 43 sub-option 7 containing the Boot ROM version number, identical to the value as reported in <Boot ROM version> field in MIB object sysDescr.

An eCM MUST send DHCP option 43 sub-option 8 containing a 6-octet, hexadecimally-encoded, vendor-specific organization unique identifier (OUI) that uniquely identifies the eCM manufacturer. A vendor MAY use the same OUI as in the eCM's MAC address, and MAY use a single OUI to identify all its eDOCSIS products.

An eCM MUST send DHCP option 43 sub-option 9 containing the Model number, identical to the value as reported in <Model number> field in MIB object sysDescr.

An eCM MUST send DHCP option 43 sub-option 10 containing the Vendor name, identical to the value as reported in <Vendor name> field in MIB object sysDescr.

If an eCM is embedded with one or more eSAFEs that utilize eCM Config File Encapsulation, the eCM MUST send DHCP option 43 sub-option 15 containing the list of eSAFEs that support this feature.

If an eCM is embedded in a device containing an eSTB, the eCM MUST send DHCP option 43 sub-option 18 containing the type of video security element in the device.

In eDOCSIS, DHCP option 43 sub-options 11-127 are reserved for future definition and MUST NOT be used for other purposes.

In eDOCSIS, DHCP option 43 sub-options 128-254 are reserved for vendors and MAY be used by vendors.

If the total number of octets in all DHCP option 43 sub-options exceeds 255 octets, the eCM MUST follow [IETF RFC 3396] to split the option into multiple smaller options.

Table 5-6 – DHCP option 43 syntax

| eCM DHCP Option 43 | Value | Description |
|--------------------|--|---|
| Sub-option 1 | | The request sub-option vector is a list of sub-options (within option 43) to be returned to client by the server upon reply to the request. None defined. |
| Sub-option 2 | "ECM" | Device type of the component making the DHCP request. For DOCSIS, this is: "ECM"= embedded Cable Modem (as specified by DOCSIS 1.0, 1.1, 2.0, or 3.0 Base Specifications) |
| Sub-option 3 | "ECM: <eSAFE ₁ :eSAFE ₂ ... SAFE _n >" | Colon-separated list of eCM and eSAFE(s) contained in the complete eDOCSIS device. First on the list MUST be "ECM" for eCM. <eSAFE _x > can be "EMTA", "EPS", "EROUTER", "ESTB", or "ETEA" corresponding to embedded MTA, embedded Portal Services Element, embedded Router, embedded STB, and embedded TDM Emulation Adapter, respectively. For example: "ECM:EMTA" = A PacketCable/Embedded MTA "ECM:EPS" = A CableHome/Embedded Portal Services Element "ECM:EROUTER" = A DOCSIS Embedded Router "ECM:ESTB" = An Embedded STB "ECM:ETEA" = An Embedded TEA "ECM:EMTA:EPS" = An Embedded MTA and Embedded Portal Services Element |
| Sub-option 4 | "<device serial number>" | Device serial number as in MIB object docsDevSerialNumber e.g., "123456" |
| Sub-option 5 | "<Hardware version>" | Hardware version number. Identical to value as reported in the <Hardware version> field in MIB object sysDescr. e.g., "v.3.2.1" |
| Sub-option 6 | "<Software version>" | Software version number. Identical to value as reported in the <Software version> field in MIB object sysDescr. e.g., "v.1.0.2" |
| Sub-option 7 | "<Boot ROM version>" | Boot ROM version. Identical to value as reported in the <Boot ROM version> field in MIB object sysDescr. e.g., "Bv4.5.6" |
| Sub-option 8 | "<OUI>" | A 6-octet, hexadecimally-encoded, vendor-specific organization unique identifier (OUI) that may match the OUI in eCM's MAC address. |
| Sub-option 9 | "<Model number>" | Device model number. Identical to value as reported in the <Model number> field in MIB object sysDescr. e.g., "T3000" |
| Sub-option 10 | "<Vendor name>" | Vendor name or ID. Identical to value as reported in the <Vendor name> field in MIB object sysDescr. e.g., "XYZ Corp" |
| Sub-option 15 | "<eSAFE ₁ :eSAFE ₂ ... SAFE _n >" | Colon-separated list of eSAFEs contained in the eDOCSIS device that supports eCM Config File Encapsulation. The list could contain "EMTA", "EPS", "ESTB", "ETEA" or "EROUTER". The requirements for support of eCM Config File Encapsulation by each eSAFE type (and hence the requirements for the contents of this sub-option) can be found in the relevant eSAFE specification. If no eSAFE supports eCM Config File Encapsulation, then the eCM MUST either not populate this sub-option, or set the sub-option length to zero. For example: "ESTB" = An Embedded STB with eCM eSAFE config file TLV support (or absence of sub-option 15) = no eSAFE supports the eCM eSAFE config file TLV |

Table 5-6 – DHCP option 43 syntax

| eCM DHCP Option 43 | Value | Description |
|--------------------|--------------------------|--|
| Sub-option 18 | "<VideoSecurityElement>" | Video Security Type Only present if the device contains an eSTB. This sub-option takes one of the following values: "CARD" corresponds to an eSTB with a POD module as its security element. "SM" corresponds to an eSTB with a Secure Microprocessor as its security element. "EMBED" corresponds to an embedded security eSTB. |

5.2.4.4 Examples (informative)

5.2.4.4.1 DOCSIS cable modem with embedded IPCablecom 1.0 MTA example

Table 5-7 shows the syntax for DHCP option 43 for the eCM in an E-MTA.

Table 5-7 – Example eDOCSIS device: E-MTA

| DHCP DISCOVER options | Value | Description |
|-----------------------------|---|---|
| eCM DHCP DISCOVER | | |
| eCM Option 43 sub-option 1 | N/A (omitted because the list is empty) | List of sub-options (within option 43) to be returned by server |
| eCM Option 43 sub-option 2 | "ECM" | Embedded cable modem |
| eCM Option 43 sub-option 3 | "ECM:EMTA" | ECM followed by a list of embedded components (eSAFEs) |
| eCM Option 43 sub-option 4 | "123456" | Device serial number as in MIB object docsDevSerialNumber |
| eCM Option 43 sub-option 5 | "V1.2.3" | Hardware version number as in <Hardware version> field in MIB object sysDescr |
| eCM Option 43 sub-option 6 | "V3.2.1" | Software version number as in <Software version> field in MIB object sysDescr |
| eCM Option 43 sub-option 7 | "Boot 4.5.6" | Boot ROM version number as in <Boot ROM version> field in MIB object sysDescr |
| eCM Option 43 sub-option 8 | "0204DF" | 6-octet OUI as Vendor ID |
| eCM Option 43 sub-option 9 | "Xman200" | Device model number as in <Model number> field in MIB object sysDescr |
| eCM Option 43 sub-option 10 | "XYZ Broadband" | Vendor name as in <Vendor name> field in MIB object sysDescr |

5.2.4.4.2 DOCSIS cable modem with embedded IPCable2Home 1.1 PS example

Table 5-8 shows the syntax for DHCP options 43 for the eCM in an IPCable2Home 1.1 ePS device.

Table 5-8 – Example eDOCSIS device: DOCSIS 1.1 IPCable2Home/ePS

| DHCP DISCOVER options | Value | Description |
|-----------------------------|-----------------|---|
| eCM DHCP DISCOVER | | |
| eCM Option 43 sub-option 1 | "<null>" | List of sub-options (within option 43) to be returned by server |
| eCM Option 43 sub-option 2 | "ECM" | Embedded cable modem |
| eCM Option 43 sub-option 3 | "ECM:EPS" | ECM followed by a list of embedded components (eSAFEs) |
| eCM Option 43 sub-option 4 | "123456" | Device serial number as in MIB object docsDevSerialNumber |
| eCM Option 43 sub-option 5 | "V1.2.3" | Hardware version number as in <Hardware version> field in MIB object sysDescr |
| eCM Option 43 sub-option 6 | "V3.2.1" | Software version number as in <Software version> field in MIB object sysDescr |
| eCM Option 43 sub-option 7 | "Boot 4.5.6" | Boot ROM version number as in <Boot ROM version> field in MIB object sysDescr |
| eCM Option 43 sub-option 8 | "0204DF" | 6-octet OUI as Vendor ID |
| eCM Option 43 sub-option 9 | "Xman200" | Device model number as in <Model number> field in MIB object sysDescr |
| eCM Option 43 sub-option 10 | "XYZ Broadband" | Vendor name as in <Vendor name> field in MIB object sysDescr |

5.2.4.4.3 DOCSIS cable modem with embedded STB example

Table 5-9 illustrates the syntax for DHCP options 43 for the eCM with an embedded STB.

Table 5-9 – Example eDOCSIS device: DOCSIS 1.1 embedded STB

| DHCP DISCOVER & REQUEST Options | Options Value | Description |
|--|---|---|
| eCM DHCP DISCOVER & REQUEST | | |
| eCM Option 43 sub-option 1 | N/A (omitted because the list is empty) | List of sub-options (within option 43) to be returned by server. |
| eCM Option 43 sub-option 2 | "ECM" | Device type of the entity making the DHCP request. For DOCSIS, this is: "ECM" = embedded Cable Modem (as specified by DOCSIS 1.0, 1.1, or 2.0 Base Specifications) |
| eCM Option 43 sub-option 3 | "ECM:ESTB" | Colon-separated list of eCM and eSAFE(s) contained in the device. First on the list MUST be "ECM". "ECM" = embedded CM "ESTB" = embedded STB |
| eCM Option 43 sub-option 4 | e.g., "123456" | Device serial number from MIB object docsDevSerialNumber |
| eCM Option 43 sub-option 5 | e.g., "V1.2.3" | Hardware version number from <Hardware version> field in MIB object sysDescr |
| eCM Option 43 sub-option 6 | e.g., "V3.2.1" | Software version number from <Software version> field in MIB object sysDescr |
| eCM Option 43 sub-option 7 | "Boot 4.5.6" | Boot ROM version number from <Boot ROM version> field in MIB object sysDescr |
| eCM Option 43 sub-option 8 | "0204DF" | A 6-octet, hexadecimally-encoded, vendor-specific organization unique identifier (OUI) that may match the OUI in the eCM's MAC address |

Table 5-9 – Example eDOCSIS device: DOCSIS 1.1 embedded STB

| DHCP DISCOVER & REQUEST Options | Options Value | Description |
|--|-----------------|---|
| eCM DHCP DISCOVER & REQUEST | | |
| eCM Option 43 sub-option 9 | "Xman200" | Device model number from <Model number> field in MIB object <code>sysDescr</code> |
| eCM Option 43 sub-option 10 | "XYZ Broadband" | Vendor name as in <Vendor name> field in MIB object <code>sysDescr</code> |
| eCM Option 43 sub-option 15 | "ESTB" | eSTB supports eCM eSAFE Config file TLVs |
| eCM Option 43 sub-option 18 | "EMBED" | This device contains an embedded security eSTB. |

5.2.4.4.4 DOCSIS Cable Modem with embedded IPCablecom MTA and IPCable2Home PS

Table 5-10 shows the syntax for DHCP option 43 for the eCM in an eDOCSIS device that contains the following embedded components: a DOCSIS 2.0 eCM, an IPCablecom 1.0 eMTA and an IPCable2Home 1.1 ePS.

Table 5-10 – Example eDOCSIS device containing DOCSIS 2.0 eCM, eMTA and ePS

| DHCP DISCOVER options | Value | Description |
|-----------------------------|-----------------|--|
| eCM DHCP DISCOVER | | |
| eCM Option 43 sub-option 1 | "<null>" | List of sub-options (within option 43) to be returned by server |
| eCM Option 43 sub-option 2 | "ECM" | Embedded cable modem |
| eCM Option 43 sub-option 3 | "ECM:EMTA:EPS" | ECM followed by a list of embedded components (eSAFEs) |
| eCM Option 43 sub-option 4 | "123456" | Device serial number as in MIB object <code>docsDevSerialNumber</code> |
| eCM Option 43 sub-option 5 | "V1.2.3" | Hardware version number as in <Hardware version> field in MIB object <code>sysDescr</code> |
| eCM Option 43 sub-option 6 | "V3.2.1" | Software version number as in <Software version> field in MIB object <code>sysDescr</code> |
| eCM Option 43 sub-option 7 | "Boot 4.5.6" | Boot ROM version number as in <Boot ROM version> field in MIB object <code>sysDescr</code> |
| eCM Option 43 sub-option 8 | "0204DF" | 6-octet OUI as Vendor ID |
| eCM Option 43 sub-option 9 | "Xman200" | Device model number as in <Model number> field in MIB object <code>sysDescr</code> |
| eCM Option 43 sub-option 10 | "XYZ Broadband" | Vendor name as in <Vendor name> field in MIB object <code>sysDescr</code> |

5.2.4.4.5 DOCSIS cable modem with embedded TEA

Table 5-11 shows the syntax for DHCP option 43 for the eCM with an embedded TEA.

Table 5-11 – Example eDOCSIS device: DOCSIS 1.1 TEA

| DHCP DISCOVER Options | Value | Description |
|-----------------------------|---|---|
| eCM DHCP DISCOVER | | |
| eCM Option 43 sub-option 1 | N/A (omitted because the list is empty) | List of sub-options (within option 43) to be returned by server |
| eCM Option 43 sub-option 2 | "ECM" | Embedded cable modem |
| eCM Option 43 sub-option 3 | "ECM:ETEA" | ECM followed by a list of embedded components (eSAFEs) |
| eCM Option 43 sub-option 4 | "123456" | Device serial number as in MIB object docsDevSerialNumber |
| eCM Option 43 sub-option 5 | "V1.2.3" | Hardware version number as in <Hardware version> field in MIB object sysDescr |
| eCM Option 43 sub-option 6 | "V3.2.1" | Software version number as in <Software version> field in MIB object sysDescr |
| eCM Option 43 sub-option 7 | "Boot 4.5.6" | Boot ROM version number as in <Boot ROM version> field in MIB object sysDescr |
| eCM Option 43 sub-option 8 | "0204DF" | 6-octet OUI as Vendor ID |
| eCM Option 43 sub-option 9 | "Xman200" | Device model number as in <Model number> field in MIB object sysDescr |
| eCM Option 43 sub-option 10 | "XYZ Broadband" | Vendor name as in <Vendor name> field in MIB object sysDescr |

5.2.4.4.6 DOCSIS cable modem with embedded router

Table 5-12 shows the syntax for DHCP Option 43 for the eCM with an embedded router.

Table 5-12 – DOCSIS cable modem with embedded router

| DHCP DISCOVER Options | Value | Description |
|-----------------------------|---|---|
| eCM DHCP DISCOVER | | |
| eCM Option 43 sub-option 1 | N/A (omitted because the list is empty) | List of sub-options (within option 43) to be returned by server |
| eCM Option 43 sub-option 2 | "ECM" | Embedded cable modem |
| eCM Option 43 sub-option 3 | "ECM:EROUTER" | ECM followed by a list of embedded components (eSAFEs) |
| eCM Option 43 sub-option 4 | "123456" | Device serial number as in MIB object docsDevSerialNumber |
| eCM Option 43 sub-option 5 | "V1.2.3" | Hardware version number as in <Hardware version> field in MIB object sysDescr |
| eCM Option 43 sub-option 6 | "V3.2.1" | Software version number as in <Software version> field in MIB object sysDescr |
| eCM Option 43 sub-option 7 | "Boot 4.5.6" | Boot ROM version number as in <Boot ROM version> field in MIB object sysDescr |
| eCM Option 43 sub-option 8 | "0204DF" | 6-octet OUI as Vendor ID |
| eCM Option 43 sub-option 9 | "Xman200" | Device model number as in <Model number> field in MIB object sysDescr |
| eCM Option 43 sub-option 10 | "XYZ Broadband" | Vendor name as in <Vendor name> field in MIB object sysDescr |
| eCM Option 43 sub-option 15 | "EROUTER" | eRouter supports eCM eSAFE Config file TLVs |

5.2.5 DHCPv6 vendor-specific option syntax requirements

In order to facilitate provisioning, all eDOCSIS devices implementing a DOCSIS 3.0 or greater CM and operating with IPv6 will include vendor-specific information options during the CM IPv6 registration process [DOCSIS3-MULPI]. Vendor-specific information options include configuration file location and name information, syslog server information, device identifier information, and cable modem capabilities. Refer to [CANN-DHCP].

5.2.5.1 eDOCSIS device information in DHCPv6 vendor-specific options

An eCM operating with IPv6 provides device-specific information through the use of the DHCPv6 vendor-specific information options. This information, which is carried in option 43 sub-options in the case of an eCM or eSAFE using IPv4, provides the provisioning system with details about the eDOCSIS device, including implemented CableLabs specification, hardware revision, software revision, and number and type of implemented eSAFEs. Refer to clause 5.2.4.3 for more information about DHCPv4 option 43 requirements.

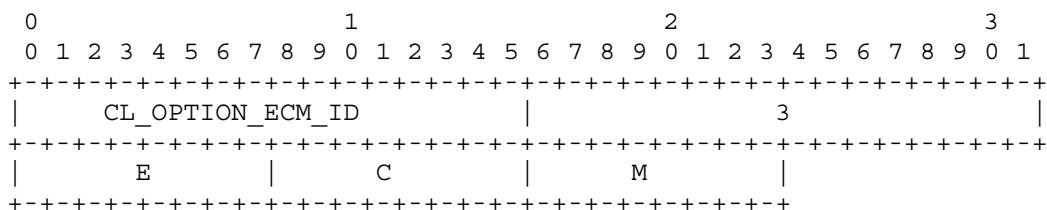
An eCM operating with IPv6 MUST include the DHCPv6 vendor-specific option codes listed below in DHCPv6 solicit messages:

- Option Code 2: Embedded Cable Modem Identifier
- Option Code 3: List of Implemented eSAFEs
- Option Code 4: Device Serial Number
- Option Code 5: Hardware Version Number
- Option Code 6: Software Version Number
- Option Code 9: Model Number
- Option Code 10: Vendor Identifier

The format of the DHCPv6 vendor specific option codes 2-6 and 9-10 listed above generally follow the format of other vendor-specific options described in [CANN-DHCP]. The specific format of each of the required Option Codes listed above is described below.

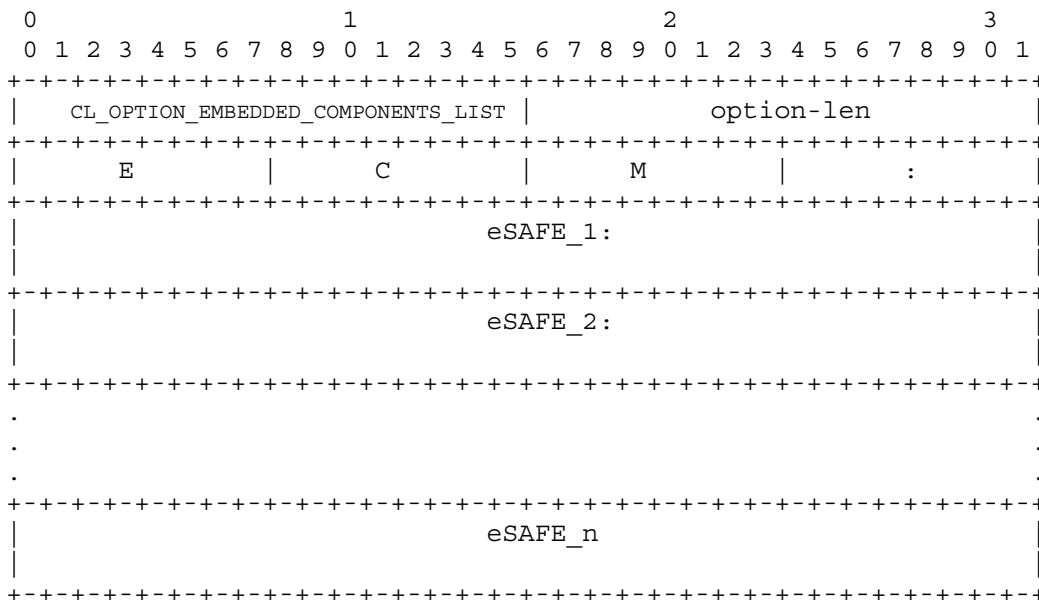
Option Code 2: Embedded Cable Modem Identifier

Option code 2 contains the string ECM, indicating to the provisioning system that the DHCP client is implemented on an embedded cable modem.

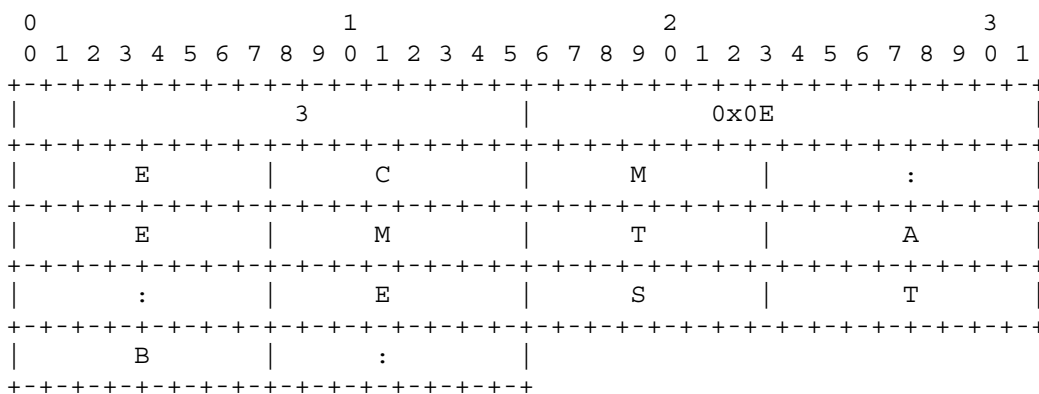


Option Code 3: List of Implemented eSAFEs

This option is a colon-separated list of eSAFEs implemented in the eDOCSIS device. The CM MUST list in the List of Implemented eSAFEs option the abbreviation for each eSAFE implemented in the device, beginning with ECM and separated with a colon. Defined eSAFE abbreviations are listed in the Description field for DHCPv4 Option 43, sub-option 3 in Table 5-6.

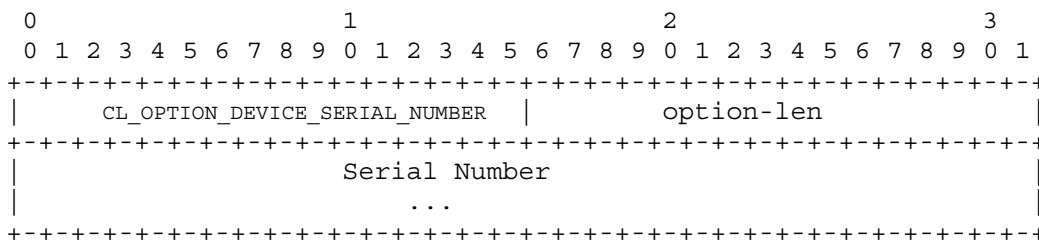


An example of the List of Implemented eSAFEs for a hypothetical product implementing an eCM, eMTA, and eSTB follows:



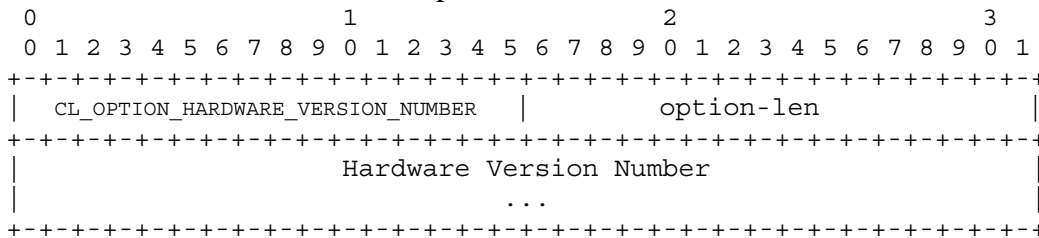
Option Code 4: Device Serial Number

The Device Serial Number option contains the eDOCSIS device's serial number.



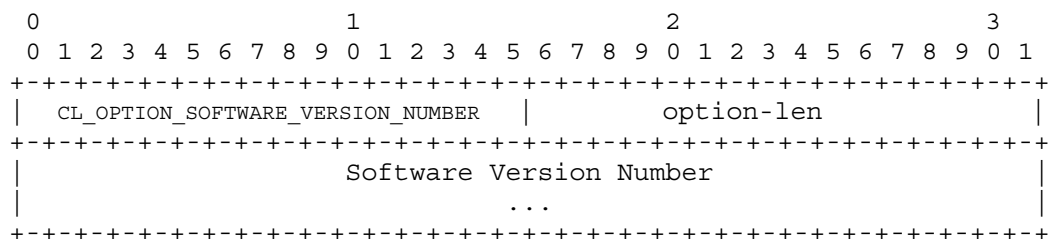
Option Code 5: Hardware Version Number

The Hardware Version Number option contains the eDOCSIS device's hardware version number.



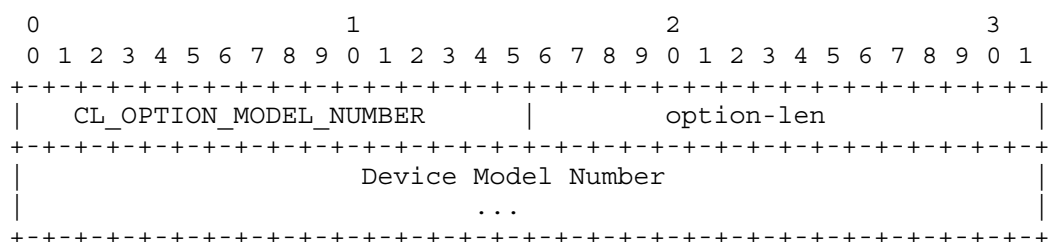
Option Code 6: Software Version Number

The Software Version Number option contains the vendor-assigned version number of the software implemented in the device.



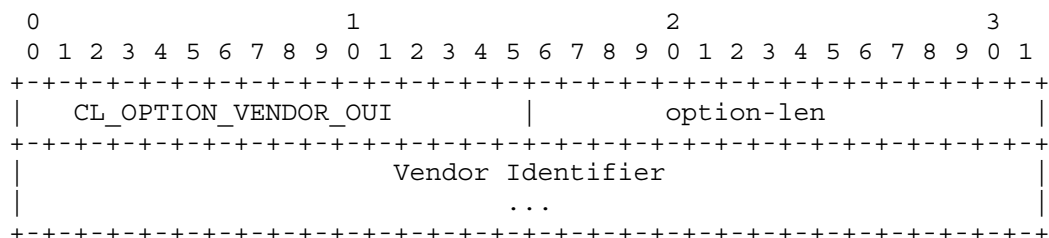
Option Code 9: Device Model Number

The Device Model Number option contains the model number of the eDOCSIS device.



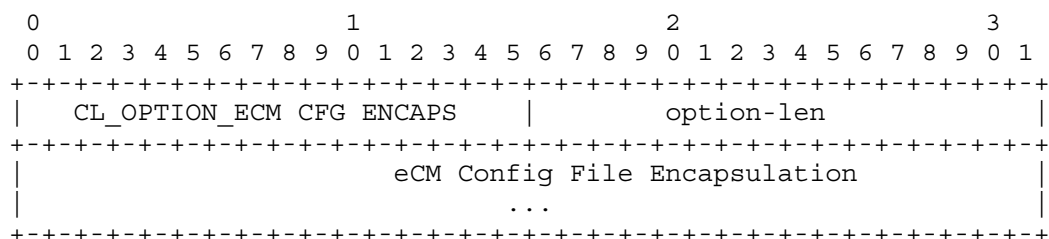
Option Code 10: Vendor Identifier

The Vendor Identifier option contains the name or other identifier of the vendor who manufactures and/or sells the eDOCSIS device.



Option Code 15: eCM Config File Encapsulation

The eCM Config File Encapsulation option indicates the list of eSAFE devices that support this feature.



5.2.6 Testability requirements

In order to verify conformance to this Recommendation and to the DOCSIS Base Specifications, a mechanism to generate and receive traffic bridged through the eCM is required. eDOCSIS devices that have a physically exposed CMCI (e.g., 10BaseT or USB) can be tested by using external packet generation equipment connected to that interface.

For cost, security, or other reasons, however, certain eDOCSIS devices may not have an exposed CMCI, necessitating an alternative mechanism.

Additionally, an eDOCSIS device MAY have multiple eSAFEs, each with a logical CPE interface (LCI) to the eCM. This Recommendation places requirements on the LCIs as well as the bridging of traffic among eCM and eSAFEs.

To this end, a software loopback for eDOCSIS (SLED) is specified below.

5.2.6.1 General requirements

An eCM SHOULD implement SLED. An eCM without an externally accessible CMCI port, or a physical interface configured to be equivalent to a CMCI port, MUST implement SLED.

5.2.6.2 SLED protocol description

5.2.6.2.1 General description

SLED is an embedded test function residing in an eCM enabling DOCSIS and eDOCSIS conformance testing coverage, particularly when an exposed CMCI is not available.

The SLED test functions are controlled via SLED MIB objects as specified in Annex A. The SLED MIB objects MUST be associated with the SNMP stack of the eCM. The SLED MIB objects MUST NOT be accessible through the CMCI.

To prevent unintended activation of SLED, all SLED functions MUST be disabled by default. SLED functions MUST only be enabled if the MIB object *sledGlobalEnable* is set to 'true' prior to eCM registration; *sledGlobalEnable* MAY be set to 'true' via inclusion in TLV-11 of the eCM's configuration file.

The SLED MIB values revert to power-on values when the CM de-registers or loses Operational state; the *sledGlobalEnable* will revert to 'false' and in-progress packet generation or loopback will be stopped.

Figure 5-15 illustrates the SLED reference model.

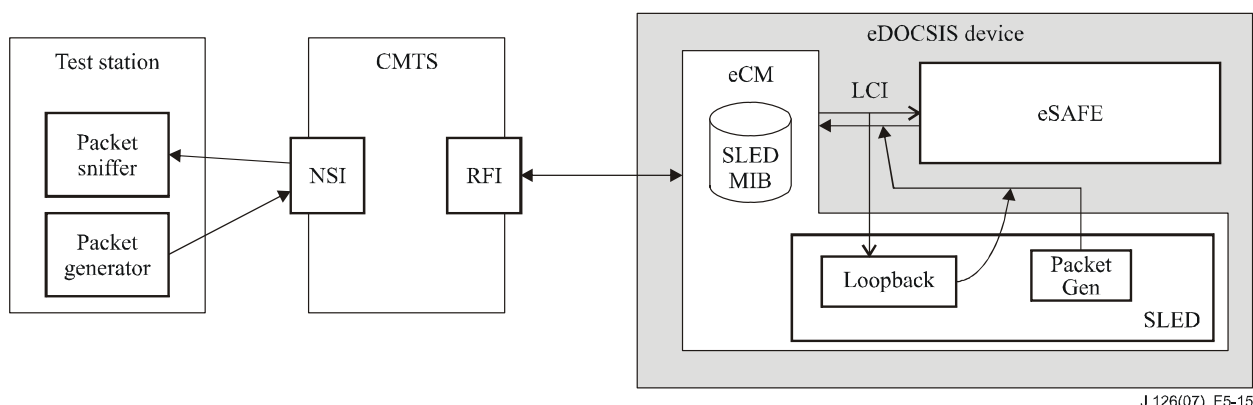


Figure 5-15 – SLED reference model

The SLED functionality supports:

- 1) *Packet loopback*

The primary purpose of the Packet loopback protocol is to enable verification of the receipt of packets across the LCI by the eSAFE. Once enabled by the SLED MIB object, all packets that are forwarded to the indicated LCI are encapsulated into a predefined packet header (Ethernet DIX frame header + IP header + UDP header) and reflected back across the LCI to the eCM for forwarding to the final destination. Typically, the looped-back packets will be addressed to, and captured by, a test station residing in the network side interface (NSI) of the CMTS.
- 2) *Packet generation*

SLED MIB objects are defined to enable setting up of Ethernet framing and payload transmission for packet generation and transmission through the LCI to the eCM. The SLED MIB objects described below control the packet transmission with parameters such as Ethernet packet header, packet rate, and the number of packets.
- 3) Packet loopback and packet generation SLED functions MUST be able to be controlled independently.
- 4) The packet loopback and packet generation SLED functions MUST NOT disrupt network connectivity to or from the eSAFE. When SLED loopback is enabled, a packet transmitted across the LCI in the eCM-to-eSAFE direction, MUST be forwarded to both the eSAFE and the SLED loopback function. When SLED functions are enabled, packets to/from the eSAFE MUST continue to be bridged across the LCI.

5.2.6.2.2 Loopback protocol

An eCM implementing SLED MUST implement the following loopback protocol:

- 1) The SLED packet loopback function is attached to the LCI associated with the eSAFE by setting SLED MIB *sledLoopbackInterface* to the eCM's ifIndex number associated with the LCI (per Table 5-1).
- 2) The SLED MIB object *sledLoopbackPktHdr* is configured with the 42-byte loopback Ethernet packet/IP/UDP headers (14-byte Ethernet header + 20-byte IPv4 header + 8-byte UDP header).
- 3) As an example, the following loopback header parameters could be used:
 - a) Ethernet MAC source address = eSAFE MAC address;
 - b) Ethernet MAC destination address = test station MAC address;
 - c) IP source address = eSAFE Management IP address;
 - d) IP destination address = test station IP address;
 - e) UDP source port number = 7;
 - f) UDP destination port number = 7.
- 4) When the SLED MIB object *sledLoopbackInterface* is set to an ifIndex associated with an LCI which supports SLED, *sledLoopbackPktHdr* contains a 42-byte octet string, and *sledLoopbackEnable* is set to 'true', the SLED MUST operate in a loopback mode.
- 5) When operating in loopback mode, all Ethernet packets forwarded across the indicated LCI by the eCM will be processed as follows:
 - a) If the received Ethernet packet is greater than 1472 octets, the Ethernet packet MUST be split into two fragments according to IP fragmentation scheme as described in [IETF RFC 791], the first consisting of the first 1472 octets of the Ethernet packet and the second containing the remaining octets, resulting in two payloads that are processed as described below.

- b) If the received Ethernet packet is less than or equal to 1472 octets, the entire packet MUST be processed as a single payload.
 - c) Each payload generated in step 5a or 5b MUST be prepended with the contents of *sledLoopbackPktHdr*.
 - d) The mutable fields within *sledLoopbackPktHdr* MUST be recomputed. The mutable fields are IP Header Checksum, IP Total Length per [IETF RFC 791], and UDP Checksum, UDP Length per [IETF RFC 768].
 - e) If the Ethernet packet is fragmented as defined in step 5a, the appropriate IP header fields MUST be updated to indicate IP fragmentation. The IP fragmentation header values will differ depending on if this is the first or second fragment being processed (per [IETF RFC 791]). Further, the final 8-bytes of *sledLoopbackPktHdr* (the UDP header) MUST NOT be prepended to the second fragment.
 - f) The Ethernet FCS MUST be computed and appended.
 - g) The resulting Ethernet packet MUST be transmitted to the LCI toward the eCM.
- 6) When the SLED MIB object *sledLoopbackEnable* is set to 'false', the SLED loopback function MUST be disabled.
- 7) While the SLED loopback function is enabled, the eCM MUST reject changes to *sledLoopbackInterface* or *sledLoopbackPktHdr*.

Figure 5-16 illustrates the SLED packet loopback encapsulation.

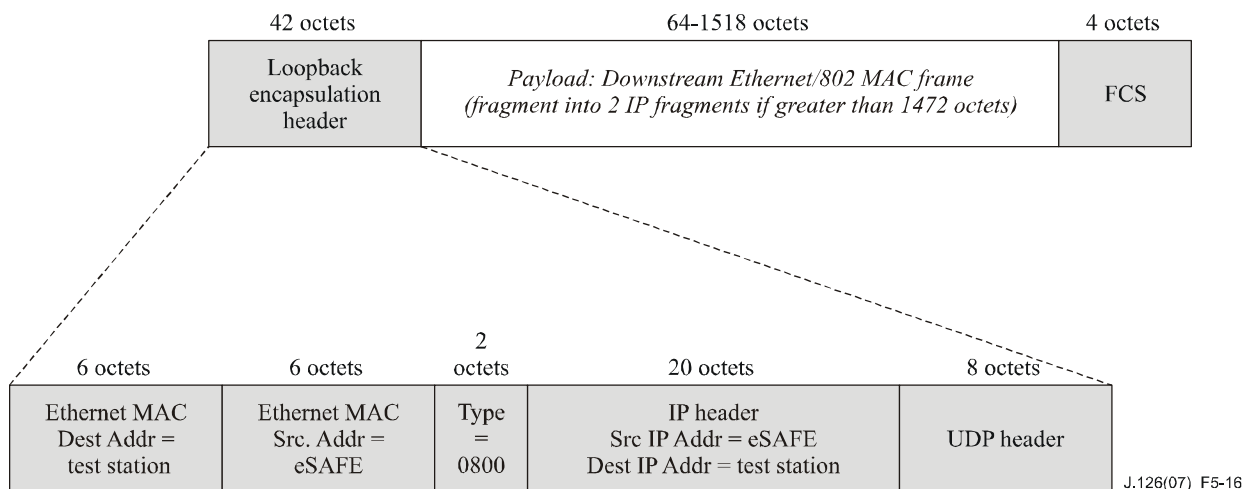


Figure 5-16 – SLED packet loopback encapsulation

Figure 5-17 illustrates an example of the SLED loopback sequence.

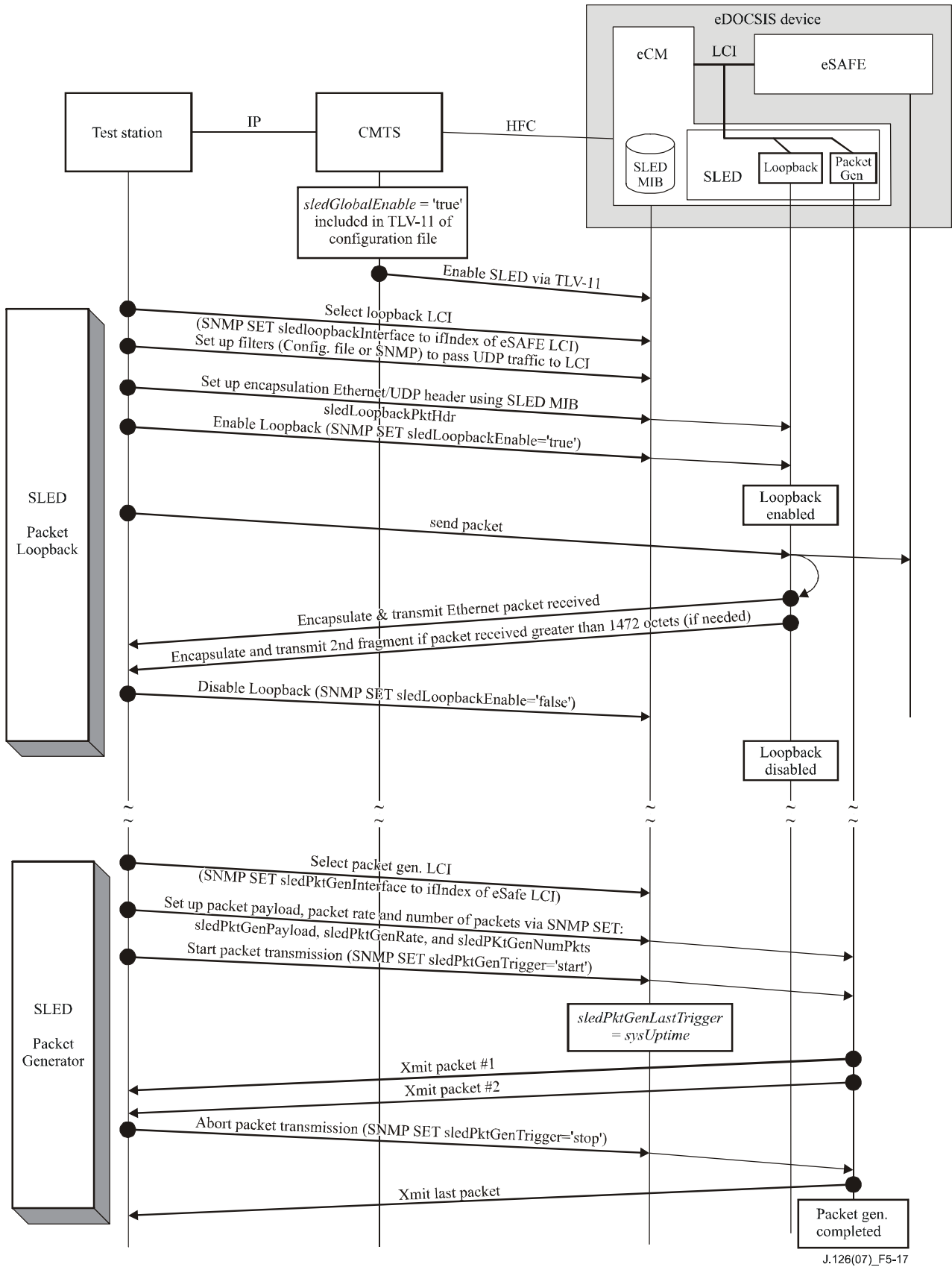


Figure 5-17 – SLED packet loopback and generation sequences

5.2.6.2.3 Packet generation protocol

An eDOCSIS device implementing SLED MUST implement the following packet generator protocol:

- 1) The SLED packet generation function is attached to the eCM's LCI associated with the eSAFE by setting SLED MIB *sledPktGenInterface* to the ifIndex number associated with the LCI (per Table 5-1).
- 2) The SLED MIB object *sledPktGenPayload* is set up to be a complete Ethernet (DIX/802 MAC) packet, including FCS trailer, for transmission across the LCI. The FCS is set to be correct for the packet as specified, and MAY be recalculated by the eCM as required for upstream processing; the SLED is not required to validate the FCS, and a packet with an invalid FCS MAY be transmitted with a corrected FCS.
- 3) The SLED MIB objects *sledPktGenRate* and *sledPktGenNumPkts* are set to non-zero values.
- 4) When *sledPktGenInterface* is set to an ifIndex associated with an LCI which supports SLED, *sledPktGenRate* and *sledPktGenNumPkts* are both set to non-zero values, the SLED Packet Generator MUST start to send generated Ethernet packets to the LCI in within 250 ms after *sledPktGenTrigger* is set to 'start'; the SLED SHOULD start to transmit packets to the LCI as soon as possible in order to minimize the amount of time it takes to run tests that use the SLED Packet Generator.
- 5) When *sledPktGenTrigger* is set to 'start', the SLED Packet Generator MUST set the SLED MIB *sledPktGenLastTrigger* to the current value of the system MIB *sysUptime*.
- 6) The packets generated by the SLED Packet Generator MUST be the exact copies of the Ethernet packet specified by the SLED MIB *sledPktGenPayload*. The average rate of generated packets MUST be as specified by the SLED MIB *sledPktGenRate*.
- 7) The packet generation MUST be continued until the total number of generated packets reaches the limit as specified by the SLED MIB *sledPktGenNumPkts*, unless terminated by setting *sledPktGenTrigger* to 'stop'. If *sledPktGenTrigger* is set to 'stop' while packets are being generated, the SLED SHOULD stop packet generation within 1 second.
- 8) While the previous sequence of SLED packets is still in progress, the eCM MUST reject changes to *sledPktGenInterface*, *sledPktGenPayload*, *sledPktGenNumPkts* or *sledPktGenRate*.

Refer to Figure 5-17 for an illustration of the SLED packet loopback and generation sequences.

5.2.7 Firmware download

All eDOCSIS devices MUST support a single secured firmware image download that is used for the entire eDOCSIS device.

With the exception of eDOCSIS devices that contain eSTBs, the download MUST be controlled by the eCM using the DOCSIS secure software download (SSD) mechanisms as specified in [BPI+], [DOCSIS2-RFI] and DOCSIS OSS.

Firmware download for an eDOCSIS device that contains an eSTB (referred to as a Set-top Device) is defined in the following subclauses.

5.2.7.1 Set-top device firmware download

There are two firmware download methods for an eDOCSIS device that contains an eSTB:

- 1) DSM-CC data carousel methods as defined in [CCIF 2.0],
- 2) DOCSIS secure software download (SSD) mechanisms as defined in [DOCSIS-RFI], [BPI+] and [DOCSIS-OSSI].

Note that OpenCable host 2.0 Set-top devices support the above firmware download methods as specified in [HOST2.0] and [CCIF 2.0].

All other Set-top Devices that do not comply with HOST 2.0 MUST support a mechanism for firmware download. Examples are 1) all the DSM-CC Data Carousel methods [CCIF 2.0], and 2) the DOCSIS SSD method.

Firmware downloads can be triggered either via the eCM or the eSTB depending upon implementation. When a Set-top Device firmware download is triggered via the eCM logical element, the eCM MUST report the status of the firmware download as described in clause 5.2.7.2. When a set-top device firmware download is triggered via the eSTB logical element, the eCM MUST report the status of the firmware download as indicated in clause 5.2.7.3.

5.2.7.2 OSS requirements for firmware downloads initiated by the eCM

If the Set-top Device supports DOCSIS SSD, then it MUST report the status of a firmware download initiated by the eCM in accordance with the CM requirements in DOCSIS OSS.

If the Set-top Device does not implement DOCSIS SSD mechanisms, its eCM MUST set the docsDevSwOperStatus to other(5). The Set-top Device responds to an attempt to trigger a TFTP upgrade (initiated through SNMP or Configuration File) as follows:

- The Set-top Device MUST ignore DOCSIS SSD triggers through SNMP or Config File TLVs.
- The Set-top Device MUST remain capable of accepting new software through the non-DOCSIS firmware download mechanism.
- The Set-top Device MUST report the attempt to trigger a TFTP firmware upgrade by logging the appropriate event at the eCM (via an entry in the docsDevEvTable).
- The docsDevSwOperStatus MUST remain other(5).

If the Set-top Device does not implement DOCSIS SSD mechanisms, then all the download-related requirements as specified in [BPI+] do not apply. Additionally the eCM in this type of Set-top Device MUST support the BPI+ MIB docsBpi2CodeDownloadControl objects with the following constraints (other objects within docsBpi2CodeDownloadControl are as defined within the MIB):

- docsBpi2CodeDownloadStatusCode always reports other(7);
- docsBpi2CodeDownloadStatusString always returns the string "DOCSIS SSD not supported".

5.2.7.3 OSS requirements for firmware downloads initiated by the eSTB

During a firmware download for the Set-top Device which is initiated by the eSTB, the eCM MUST set its MIB objects as follows:

- docsDevSwServer to 0.0.0.0.
- docsDevSwFilename to the filename of the image the eSTB is downloading.
- docsDevSwAdminStatus to ignoreProvisioningUpgrade(3).
- docsDevSwOperStatus to inProgress(1).
- docsDevSwCurrentVers to the current version of the eDOCSIS device code.
- docsBpi2CodeDownloadStatusCode to other(7).
- docsBpi2CodeDownloadStatusString to the string "Set-top Device code file download initialized by the eSTB".

During the download of an image for the Set-top Device which is initiated by the eSTB, the eCM MUST ignore any change to the docsDevSwAdminStatus MIB object. Note that by setting docsDevSwAdminStatus to ignoreProvisioningUpgrade and by fixing this value, the eCM will

ignore any firmware download triggers through the eCM configuration file while a firmware download initiated by the eSTB is taking place. Note also that setting docsDevSwAdminStatus to ignoreProvisioningUpgrade and by fixing this value, the eCM effectively ignores firmware download triggers through SNMP while a firmware download initiated by the eSTB is taking place.

After the download process finishes, the eCM MUST set its MIB objects as follows:

- docsDevSwServer to 0.0.0.0.
- docsDevSwFilename to the filename of the image that the eSTB intended to download.
- docsDevSwAdminStatus to ignoreProvisioningUpgrade(3) if the eDOCSIS device firmware download initiated by the eSTB succeeded, or to the value present before the download was initiated if the eDOCSIS device firmware download initiated by the eSTB failed.
- docsDevSwOperStatus to failed(4) if the eDOCSIS device firmware download initiated by the eSTB failed, or to other(5) if it succeeded.
- docsDevSwCurrentVers to the current version of the eDOCSIS device code.
- docsBpi2CodeDownloadStatusCode to codeFileVerified(5) if the eDOCSIS device could verify the firmware download, or codeFileRejected(6) if the eDOCSIS device could not verify the firmware download and therefore rejected it, or to other(7) in any other case.
- docsBpi2CodeDownloadStatusString to the string "Firmware Download initiated by the eSTB successful" if the eDOCSIS device firmware download initiated by the eSTB succeeded, or "Firmware Download initiated by the eSTB failed".

In the case where the CVC distributed to the Set-top Device through the eSTB fails verification checks (e.g., those defined in OpenCable), the eCM MUST set its MIB objects as follows:

- docsBpi2CodeDownloadStatusCode to other(7).
- docsBpi2CodeDownloadStatusString to the string "Set-top Device CVC validation failure for CVC distributed through eSTB".

5.2.8 eSAFE configuration

It is within the scope of each eSAFE specification to define the configuration mechanisms for each type of eSAFE device. eDOCSIS provides two methods for the direct configuration of eSAFE features via the eCM. Either of these two methods may be used by an eSAFE, both may be used, or neither may be used, as defined in the relevant eSAFE specification. The two methods are:

- eSAFE-MIB Configuration – The eSAFE-MIB can provide eCM MIB objects that can be used to configure a particular type of eSAFE (see Annex B). Such MIB objects can be set via the CM configuration file, or by direct SNMP access to the eCM.
- eCM Config File Encapsulation – The eCM configuration file can contain eSAFE specific configuration parameters, the details of which are defined by the eSAFE specification. These configuration parameters are encapsulated in the eCM configuration file via the "eCM eSAFE Configuration File TLVs", and are passed to the eSAFE upon validation of the configuration file.

5.2.8.1 eCM config file encapsulation

If the eCM is embedded with one or more eSAFEs that utilize eCM Config File Encapsulation, the eCM MUST recognize the corresponding eCM eSAFE Configuration TLVs as listed in Table 5-13.

Table 5-13 – eCM eSAFE TLVs

| Type | Length | Applies to eSAFE type |
|----------|--------|-----------------------|
| 201 | n | ePS |
| 202 | n | eRouter |
| 203..215 | | <reserved> |
| 216 | n | eMTA |
| 217 | n | eSTB |
| 218 | | <reserved> |
| 219 | n | eTEA |
| 220..231 | | <reserved> |

Upon successful validation of the CM MIC, the eCM MUST pass the contents of the appropriate eCM eSAFE Configuration File TLVs to each eSAFE that supports eCM Config File Encapsulation.

The eCM MUST silently ignore eCM eSAFE Configuration File TLVs for eSAFEs that do not exist or that do not support eCM Config File Encapsulation.

The mechanism used to pass eCM eSAFE Configuration File TLVs to the eSAFE is vendor specific. It is in the scope of each eSAFE specification to define the encoding of configuration parameters within the corresponding eSAFE TLV and the rules in case the contents are longer than the 254-octet maximum length of each eCM configuration file TLV instance.

Annex A

SLED MIB definition

(This annex forms an integral part of this Recommendation)

An eCM implementing SLED MUST implement the following SLED MIB objects.

```
SLED-MIB DEFINITIONS ::= BEGIN
IMPORTS
    MODULE-IDENTITY,
        Integer32,
        OBJECT-TYPE          FROM SNMPv2-SMI
        TruthValue,
        TimeStamp           FROM SNMPv2-TC
        OBJECT-GROUP,
        MODULE-COMPLIANCE  FROM SNMPv2-CONF
        clabProjDocsis     FROM CLAB-DEF-MIB
        InterfaceIndex     FROM IF-MIB
    ;

sledMib MODULE-IDENTITY
    LAST-UPDATED      "200312300000Z" -- December 30, 2003
    ORGANIZATION      "eDOCSIS Working Group"
    CONTACT-INFO
        "John Eng
        Postal:      Cable Television Laboratories, Inc
        400 Centennial Parkway
        Louisville, CO 80027
        U.S.A.
        Phone:      +1 303-661-9100
        Fax:        +1 303-661-9199
        E-mail:     mibs@cablelabs.com"
    DESCRIPTION
        "This MIB module provides the management objects necessary
        to configure and invoke the Software Loopback Application
        for eDOCSIS (SLED) functionality."
    REVISION          "200312300000Z" -- December 30, 2003
        DESCRIPTION
            "Initial version of the ITU-T eDOCSIS SLED MIB module.

            Acknowledgement:
            Greg Nakanishi of Motorola, Inc.
            "

    ::= { clabProjDocsis 13 }
-- Administrative assignments
sledNotifications      OBJECT IDENTIFIER ::= { sledMib 0 }
sledMibObjects         OBJECT IDENTIFIER ::= { sledMib 1 }
sledMibNotificationsObjects OBJECT IDENTIFIER ::= { sledMib 2 }
sledMibConformance    OBJECT IDENTIFIER ::= { sledMib 3 }

-- Object Groups
sledGlobal             OBJECT IDENTIFIER ::= { sledMibObjects 1 }
sledLoopback          OBJECT IDENTIFIER ::= { sledMibObjects 2 }
sledPktGen            OBJECT IDENTIFIER ::= { sledMibObjects 3 }

--
-- The following group describes the objects that apply to
-- both loopback and packet generator SLED functionality
--
```

```

sledGlobalEnable OBJECT-TYPE
SYNTAX      TruthValue
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
    "This object allows the SLED functionality to be enabled/disabled.

    This object may only be updated prior to device registration. If
    the device has completed registration, any attempt to update
    the value of this object returns 'notWritable'.

    Prior to registration, if the value of this object is set to
    'true', the SLED functionality is enabled and access to this MIB is
    allowed.

    Prior to registration, if the value of this object is set to
    'false', the SLED functionality is disabled and any attempt to
    update other objects in this MIB returns 'noAccess'.
    "
    DEFVAL {false}
    ::= { sledGlobal 1 }

--
-- The following group describes the loopback objects
--
sledLoopbackInterface OBJECT-TYPE
SYNTAX      InterfaceIndex
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
    "The index of the logical CPE interface (LCI) that the SLED
    loopback function is attached to. If the index does not
    correspond to a LCI supported by this device, 'wrongValue'
    is returned.

    Any attempt to set this object while sledLoopbackEnable is
    set to 'true' returns 'notWritable'."
    ::= { sledLoopback 1 }

sledLoopbackEnable OBJECT-TYPE
SYNTAX      TruthValue
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
    "Setting this object to 'true' enables the loopback function.
    Setting this object to 'false' disables the loopback function.
    When enabled, all Ethernet packets received by the SLED from the
    LCI are processed as follows:

    1. If the received Ethernet packet is greater than 1472 octets,
       the Ethernet packet is split into two fragments, the first
       consisting of the first 1472 octets of the Ethernet packet
       and the second containing the remaining octets, resulting in
       two payloads that are processed as described below. If
       the received Ethernet packet is less than or equal to 1472
       octets, the entire packet will be processed as a single payload.
    2. For each payload generated in step 1, the payload is appended to
       the contents of sledLoopbackPktHdr.
    3. The mutable fields within sledLoopbackPktHdr MUST be recomputed.
       The mutable fields are IP Header Checksum, IP Total Length, UDP
       Checksum, and UDP Length.
    4. If the Ethernet packet was fragmented in step 1, the appropriate
       IP header fields (Flags and Fragment Offset) are updated to
       indicate IP fragmentation.
    "

```

- These IP fragmentation header values will differ depending on if this is the first or second fragment being processed.
5. The Ethernet FCS is computed and appended.
 6. The resulting Ethernet packet is transmitted to the LCI."

```
DEFVAL {false}
 ::= { sledLoopback 2 }
```

```
sledLoopbackPktHdr OBJECT-TYPE
```

```
SYNTAX      OCTET STRING (SIZE(42))
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
  "A properly formatted Ethernet(DIX)+IP+UDP header for use in
  SLED loopback processing as described in sledLoopbackEnable.
  The object value contains mutable fields that are recomputed:
  the IP Header Checksum, IP Total Length, UDP Length, and
  UDP Checksum. Any attempt to set this object while
  sledLoopbackEnable is set to 'true' returns 'notWritable'."
 ::= { sledLoopback 3 }
```

```
--
```

```
-- The following group describes the packet generation objects
```

```
--
```

```
sledPktGenInterface OBJECT-TYPE
```

```
SYNTAX      InterfaceIndex
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
  "The index of the logical CPE interface (LCI) that the SLED
  packet generation function is attached to. If the index does
  not correspond to a LCI supported by the device, 'wrongValue'
  is returned. Any attempt to set this object while
  sledPktGenTrigger is set to 'start' returns 'notWritable'."
 ::= { sledPktGen 1 }
```

```
sledPktGenPayload OBJECT-TYPE
```

```
SYNTAX      OCTET STRING (SIZE(64..1518))
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
  "The properly formatted Ethernet packet payload to be generated.
  Any attempt to set this object while sledPktGenTrigger is set
  to 'start' returns 'notWritable'."
 ::= { sledPktGen 2 }
```

```
sledPktGenRate OBJECT-TYPE
```

```
SYNTAX      Integer32
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
  "The packet rate (in packets per second) that the SLED is to
  transmit the packet specified in the sledPktGenPayload. Any
  attempt to set this object while sledPktGenTrigger is set to
  'start' returns 'notWritable'."
DEFVAL {10}
 ::= { sledPktGen 3 }
```

```
sledPktGenNumPkts OBJECT-TYPE
```

```
SYNTAX      Integer32
MAX-ACCESS  read-write
STATUS      current
```

```

DESCRIPTION
    "Number of packets to be generated at the rate specified by
    sledPktGenRate. Any attempt to set this object while
    sledPktGenTrigger has been set to 'start' will return
    'notWritable'."
    DEFVAL {1}
    ::= { sledPktGen 4 }

sledPktGenTrigger OBJECT-TYPE
    SYNTAX      INTEGER {
        start(1),
        stop(2)
    }
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "This object controls the packet generation.  Setting this object
        to 'start' causes the packet generation to begin.  Reading this
        object will return 'start' if a packet generation is in progress,
        otherwise it will return 'stop'.  Setting this object to 'stop'
        while packet generation is in progress aborts the packet
        generation.  Setting this object to 'start' while packet
        generation is in progress, 'wrongValue' is returned."
    DEFVAL {stop}
    ::= { sledPktGen 5 }

sledPktGenLastTrigger OBJECT-TYPE
    SYNTAX      TimeStamp
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Value of sysUptime when the packet generation was
        last triggered."
    ::= { sledPktGen 6 }

-- Conformance information *****

sledMibCompliances OBJECT IDENTIFIER ::= { sledMibConformance 1 }
sledMibGroups      OBJECT IDENTIFIER ::= { sledMibConformance 2 }

-- Compliance statements

sledMibCompliance MODULE-COMPLIANCE
    STATUS      current
    DESCRIPTION
        "The compliance statement for SLED."
    MODULE

-- unconditionally mandatory groups

    MANDATORY-GROUPS {
        sledMibBaseGroup
    }

::= { sledMibCompliances 1 }

sledMibBaseGroup OBJECT-GROUP
    OBJECTS {
        sledGlobalEnable,
        sledLoopbackInterface,
        sledLoopbackEnable,
        sledLoopbackPktHdr,
        sledPktGenInterface,

```



```
        sledPktGenPayload,  
        sledPktGenRate,  
        sledPktGenNumPkts,  
        sledPktGenTrigger,  
        sledPktGenLastTrigger  
    }  
    STATUS      current  
    DESCRIPTION  
        "Group of object in SLED MIB"  
        ::= { sledMibGroups 1 }  
END
```

Annex B

eSAFE MIB Definition

(This annex forms an integral part of this Recommendation)

An eCM in an eDOCSIS device implementing an embedded CableHome portal services logical element (ePS) and/or embedded PacketCable multimedia terminal adapter (eMTA) logical element as an eSAFE MUST implement the following eSAFE MIB.

```
ESAFE-MIB DEFINITIONS ::= BEGIN
IMPORTS
    MODULE-IDENTITY,
    Unsigned32,
    OBJECT-TYPE          FROM SNMPv2-SMI  --RFC 2578
    OBJECT-GROUP,
    MODULE-COMPLIANCE   FROM SNMPv2-CONF  -- RFC 2580

    TruthValue,
    DateAndTime,
    PhysAddress         FROM SNMPv2-TC    -- RFC 2579

    SnmpAdminString    FROM SNMP-FRAMEWORK-MIB --RFC 3411

    ifIndex            FROM IF-MIB  --RFC 2863

    clabProjDocsis    FROM CLAB-DEF-MIB
;

esafeMib MODULE-IDENTITY
    LAST-UPDATED "200702230000Z" -- February 23, 2007
    ORGANIZATION "Cable Television Laboratories, Inc."
    CONTACT-INFO
        "Postal: Cable Television Laboratories, Inc.
        858 Coal Creek Circle
        Louisville, CO 80027
        U.S.A.
        Phone: +1 303-661-9100
        Fax: +1 303-661-9199
        E-mail: mibs@cablelabs.com"
    DESCRIPTION
        "This MIB module provides the management objects necessary
        to configure functionality of eSAFE components of a device
        implementing an eDOCSIS compliant cable modem and one or
        more eSAFE elements.

        Copyright 1999-2007 Cable Television Laboratories, Inc.
        All rights reserved."
    REVISION "200702230000Z" -- February 23, 2007
    DESCRIPTION
        "This revision is published as CM-SP-eDOCSIS-I11."

    REVISION "200607280000Z" -- July 28, 2006
    DESCRIPTION
        "This revision is published as CM-SP-eDOCSIS-I09."
    ::= { clabProjDocsis 14 }

-- Administrative assignments

-- esafeNotifications OBJECT IDENTIFIER ::= { esafeMib 0 }
esafeMibObjects      OBJECT IDENTIFIER ::= { esafeMib 1 }
esafeBase            OBJECT IDENTIFIER ::= { esafeMibObjects 1 }
```

```

esafePsMibObjects OBJECT IDENTIFIER ::= { esafeMibObjects 2 }
esafeMtaMibObjects OBJECT IDENTIFIER ::= { esafeMibObjects 3 }
esafeStbMibObjects OBJECT IDENTIFIER ::= { esafeMibObjects 4 }
esafeErouterMibObjects OBJECT IDENTIFIER ::= { esafeMibObjects 5 }

```

```
-- Object Groups
```

```
--
```

```
-- eSAFE Base Objects
```

```
--
```

```

esafeProvisioningStatusTable OBJECT-TYPE
    SYNTAX SEQUENCE OF EsafeProvisioningStatusEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "This table contains the current provisioning
        status of each implemented eSAFE, and information
        about the last failure or exception condition in
        the eSAFE provisioning process, if applicable."
    ::= { esafeBase 1 }

```

```

esafeProvisioningStatusEntry OBJECT-TYPE
    SYNTAX EsafeProvisioningStatusEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "An entry in this table is created for
        each eSAFE implemented in the eDOCSIS
        compliant device."
    INDEX { ifIndex }
    ::= { esafeProvisioningStatusTable 1 }

```

```

EsafeProvisioningStatusEntry ::=SEQUENCE
{
    esafeProvisioningStatusProgress INTEGER,
    esafeProvisioningStatusFailureFound TruthValue,
    esafeProvisioningStatusFailureFlow SnmpAdminString,
    esafeProvisioningStatusFailureEventID Unsigned32,
    esafeProvisioningStatusFailureErrorText SnmpAdminString,
    esafeProvisioningStatusLastUpdate DateAndTime
}

```

```

esafeProvisioningStatusProgress OBJECT-TYPE
    SYNTAX INTEGER {
        notInitiated(1),
        inProgress(2),
        finished(3)
    }
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "The current state of the eSAFE provisioning process.

        notInitiated(1) indicates that the eSAFE has not yet
        begun its provisioning process.

        inProgress(2) indicates that the eSAFE is in the process
        of provisioning.

        finished(3) indicates that the eSAFE completed
        its provisioning process. Provisioning success
        or failure information is provided by other
        objects, including

```

```
    esafeProvisioningStatusFailureFound and
    cabhPsDevProvState."
 ::= { esafeProvisioningStatusEntry 1 }
```

esafeProvisioningStatusFailureFound OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"true(1) indicates that the eSAFE encountered an error condition during the provisioning process. An eSAFE could start a looping process from a previous flow step after a failure, therefore this value is retained until the flow step that initially failed eventually passes or is updated with another error condition.

The eSAFE device needs to reflect in the value of 'esafePsProvisioningStatusFailureFound' any recognized errors even if it is still in the process of provisioning, i.e., when esafeProvisioningStatusProgress has a value of inProgress(2)."

REFERENCE

"PacketCable Provisioning specification, Provisioning Overview section; CableHome specification, Provisioning Processes section, BSoD TDM Emulation Interface Specification, Provisioning section."

```
 ::= { esafeProvisioningStatusEntry 2 }
```

esafeProvisioningStatusFailureFlow OBJECT-TYPE

SYNTAX SnmpAdminString

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"If esafeProvisioningStatusFailureFound is true(1), this object contains the label for the provisioning flow step in which the error condition was encountered, otherwise an empty value is reported.

The value of this object corresponds to the provisioning sequence 'Flow Step' designator for the associated eSAFE, as defined in the eSAFE specification. For example, an ePS will report a value like CHPSWMD-1, an eMTA will report a value such as MTA-1, and a TEA will report a value such as eTEA-1."

REFERENCE

"PacketCable Provisioning specification, Provisioning Overview section; CableHome specification, Provisioning Processes section; BSoD TDM Emulation Interface Specification, provisioning section."

```
 ::= { esafeProvisioningStatusEntry 3 }
```

esafeProvisioningStatusFailureEventID OBJECT-TYPE

SYNTAX Unsigned32 (0..4294967295)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"If esafeProvisioningStatusFailureFound is true(1), this object contains the eSAFE log error Event Identifier defined in the eSAFE specification, otherwise it returns a value '0'. For an eMTA type eSAFE, this object reports the PacketCable EventID value from the Provisioning Events table, e.g., 65535. For an ePS type eSAFE, this object reports the EventID value from the Defined Events for CableHome table, e.g., 68000100. For an eTEA type eSAFE, this object reports the Event ID value from the 'Event ID' column of the DOCSIS Events Extensions table for BSoD TDM Emulation Interface Specification, e.g., 8000100."

REFERENCE

"PacketCable Provisioning specification, Appendix I Provisioning Events; CableHome specification, Appendix II Format and Content for Event, SYSLOG, and SNMP Trap; BSoD TDM Emulation Interface Specification, Annex A Event, SYSLOG, and SNMP Trap Extensions."

::={ esafeProvisioningStatusEntry 4 }

esafeProvisioningStatusFailureErrorText OBJECT-TYPE

SYNTAX SnmpAdminString

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"If esafeProvisioningStatusFailureFound is true(1), this object contains the eSAFE log error messages based on the eSAFE specification, otherwise it returns an empty value. For an eMTA type eSAFE, this object reports the value from the 'Comments' column of the Provisioning Events table, e.g., 'The DNS Response from the DNS server did not resolve the TFTP FQDN.' For an ePS type eSAFE this object reports the value from the 'Event Text' column of the Defined Events for CableHome table, e.g., 'DHCP Failed - Discover sent, no offer received. For an eTEA type eSAFE, this object reports the value from the 'Event Message' column of the DOCSIS Events Extensions table for BSoD TDM Emulation Interface Specification, e.g., 'TFTP config file download error (eTEA-6).'"

REFERENCE

"PacketCable Provisioning specification, PacketCable Management Event Mechanism specification; CableHome specification, Appendix II Format and Content for Event, SYSLOG, and SNMP Trap; BSoD TDM Emulation Interface Specification, DOCSIS Events Extensions Annex for Event, SYSLOG, and SNMP Trap Extensions."

::={ esafeProvisioningStatusEntry 5 }

esafeProvisioningStatusLastUpdate OBJECT-TYPE

SYNTAX DateAndTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION
 "The value of the eCM docsDevDateTime when
 this row entry was last updated."
 ::= { esafeProvisioningStatusEntry 6 }

esafeDevStatusTable OBJECT-TYPE
 SYNTAX SEQUENCE OF EsafeDevStatusEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "This table contains entries that provide the SNMP manager
 with status information pertaining to each implemented
 eSAFE device. While this table MUST be implemented by all
 eDOCSIS devices, the support for reporting such information
 and the status conditions supported will be determined by
 the corresponding eSAFE specifications. It is highly
 recommended that the eSAFE MIBs themselves have objects
 to specify more detailed information."
 ::= { esafeBase 2 }

esafeDevStatusEntry OBJECT-TYPE
 SYNTAX EsafeDevStatusEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "An entry in this table MUST be created for each eSAFE
 device behind the eCM. The index needs to be the
 corresponding index in the ifTable for the associated
 eSAFE device."
 INDEX { ifIndex }
 ::= { esafeDevStatusTable 1 }

esafeDevStatusEntry ::= SEQUENCE
 {
 esafeDevServiceIntImpact INTEGER,
 esafeDevServiceIntImpactInfo SnmpAdminString
 }

esafeDevServiceIntImpact OBJECT-TYPE
 SYNTAX INTEGER {
 significant(1),
 none(2),
 unsupported(3)
 }
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "The value of this MIB object indicates the service
 interruption impact assessment of the corresponding eSAFE
 device as determined by the current status of the eSAFE
 device, in accordance with the directives provided in the
 eSAFE specification.

 If esafeDevServiceIntImpact is set to significant (1), it
 indicates that the corresponding eSAFE device (as per the
 eSAFE specification) identifies a significant impact on the
 active services at the given point in time. This impact
 level is highly recommended for critical or real-time
 services, though the impact assessment is left to the
 directives provided by the associated eSAFE specification."

If esafeDevServiceIntImpact is set to none (2), it indicates that the corresponding eSAFE device (as per the eSAFE specification) identifies no significant impact on the services offered at the given point in time.

If esafeDevServiceIntImpact is unsupported(3), it indicates that the corresponding eSAFE device has no known interfaces to support this feature or the eSAFE specification does not recommend this feature.

If the eSAFE specification specifies the use of this mechanism, then it MUST define definitive states for the impacts (significant or none) and the value of unsupported(3) MUST not be used by the eDOCSIS device for that eSAFE interface.

However, if the corresponding eSAFE specification does not provide any directives, then the value MUST be set to unsupported(3).

If there exists multiple services being offered by an eSAFE device (Either multiple services or multiple instances of the same service), this MIB MUST indicate the highest possible impact and other impact information SHOULD be populated in the associated esafeDevServiceIntImpactInfo table."

```
::={ esafeDevStatusEntry 1 }
```

esafeDevServiceIntImpactInfo OBJECT-TYPE

SYNTAX SnmpAdminString

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object provides more information to the SNMP Managers regarding the condition reported in esafeDevServiceIntImpact. The eSAFE device vendor could use this to fill in specific vendor strings or values that could add value or provide more information related to the status.

Examples:

For eMTA devices:

Lines 1 and 3 have active connections, Line 2 is not provisioned.

<Value of the corresponding MIB object in the eMTA MIBs, if applicable>

For other eSAFE devices:

Critical video streaming in progress, please wait for 5.30 minutes

<Value of the corresponding MIB object in the eSAFE MIBs, if applicable>.

The device MUST report 'No Additional Information' in case the associated eSAFE vendor cannot obtain information from the eSAFE device."

```
::={ esafeDevStatusEntry 2 }
```

--

-- Objects that apply to an eCM with an ePS type eSAFE.

--

esafePsCableHomeModeControl OBJECT-TYPE

```
SYNTAX      INTEGER {
                disabledMode(1),
                provSystem(2),
                dormantCHMode(3)
            }
```

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"This object provides control over the mode of operation of the CableHome ePS eSAFE element of the eDOCSIS compliant device.

When this object is set to disabledMode(1), the ePS eSAFE element is instructed to switch to CableHome Disabled Mode operation.

When this object is set to provSystem(2), the ePS eSAFE element restarts its provisioning process.

When this object is set to dormantCHMode(3), the ePS eSAFE element is instructed to switch to CableHome Dormant Mode operation. In this mode, the ePS restarts its provisioning process omitting CableHome-specific DHCP Options 60 and 43 in the DHCP DISCOVER and DHCP REQUEST messages, acquires an IP address lease from the cable operator's DHCP server, and operates in unmanaged Dormant CableHome Mode regardless of the values of the file and siaddr fields or of the values of DHCP options that might otherwise configure the ePS to operate in DHCP Provisioning Mode or in SNMP Provisioning Mode.

The value of this object MUST persist across cable modem resets."

REFERENCE

"CableHome specifications, CableHome Operational Modes section."

DEFVAL { dormantCHMode }

::= { esafePsMibObjects 1 }

esafePsCableHomeModeStatus OBJECT-TYPE

```
SYNTAX      INTEGER {
                disabledMode(1),
                dormantCHMode(2),
                cableHomeMode(3)
            }
```

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object provides visibility to the current mode of operation of the CableHome ePS eSAFE element of the eDOCSIS compliant device.

If the value of this object is disabledMode(1), the ePS eSAFE element is currently operating in CableHome Disabled Mode.

If the value of this object is dormantCHMode(2), the ePS is currently operating in Dormant CableHome Mode.

If the value of this object is cableHomeMode(3), the ePS is currently operating in CableHome mode."

REFERENCE

"CableHome specification, CableHome Operational Models section."

::= { esafePsMibObjects 2 }

--

-- Objects that apply to an eCM with an eRouter type eSAFE.

--

esafeErouterAdminMode OBJECT-TYPE

SYNTAX INTEGER {
disabled(1),
ipv4Only(2),
ipv6Only(3),
ipv4AndIpv6(4)
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object provides visibility to the mode of operation to which the DOCSIS eRouter eSAFE element was set via TLV in the cable modem configuration file.

If the value of this object is disabled(1), the eRouter was configured via TLV in the cable modem configuration file to not initialize as described in the eRouter Initialization section of the CableLabs IPv4 and IPv6 eRouter Specification.

If the value of this object is ipv4Only(2), the eRouter was configured via TLV in the cable modem configuration file to operate with an IPv4 network address and with the IPv4 stack operational and to operate without an IPv6 network address and to not run an IPv6 protocol stack.

If the value of this object is ipv6Only(3), the eRouter was configured via TLV in the cable modem configuration file to operate with an IPv6 network address and with the IPv6 stack operational and to operate without an IPv4 network address and to not run an IPv4 protocol stack.

If the value of this object is ipv4AndIpv6(4), the eRouter was configured via TLV in the cable modem configuration file to operate with an IPv4 network address and an IPv6 network address and to run both IPv4 and IPv6 protocol stacks."

REFERENCE

"DOCSIS IPv4 and IPv6 eRouter Specification, CM-SP-eRouter-I02-070223."

::= { esafeErouterMibObjects 1 }

esafeErouterOperMode OBJECT-TYPE

SYNTAX INTEGER {
disabled(1),
ipv4OnlyFwding(2),
ipv6OnlyFwding(3),
ipv4AndIpv6Fwding(4),
noIpv4AndNoIpv6Fwding(5)
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object provides visibility to the current mode of operation of the DOCSIS eRouter eSAFE element of the eDOCSIS compliant device.

If the value of this object is disabled(1), the eRouter eSAFE element has been administratively Disabled. The eDOCSIS device will bridge traffic according to the configuration of the DOCSIS embedded cable modem (eCM)

If the value of this object is ipv4OnlyFwding(2), the eRouter eSAFE element is currently operating with the IPv4 protocol stack operational, is forwarding IPv4 traffic, and is not running an IPv6 protocol stack and not forwarding IPv6 traffic.

If the value of this object is ipv6OnlyFwding(3), the eRouter eSAFE element is currently operating with the IPv6 protocol stack operational, is forwarding IPv6 traffic, and is not running an IPv4 protocol stack and not forwarding IPv4 traffic.

If the value of this object is ipv4AndIpv6Fwding(4), the eRouter eSAFE element is currently operating with both the IPv4 protocol stack and IPv6 protocol stack operational, and is forwarding IPv4 and IPv6 traffic.

If the value of this object is noIpv4AndNoIpv6Fwding(5), the eRouter is currently operating with neither the IPv4 nor IPv6 protocol stack running. The eRouter is unable to pass traffic between the Operator-Facing Interface and the Customer-Facing Interface. "

REFERENCE

"DOCSIS IPv4 and IPv6 eRouter Specification, CM-SP-eRouter-I02-070223."

::= { esafeErouterMibObjects 2 }

esafeErouterPhysAddress OBJECT-TYPE

SYNTAX PhysAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The physical address of the operator-facing interface of the DOCSIS eRouter eSAFE element."

::= { esafeErouterMibObjects 3 }

-- Conformance information

esafeMibConformance OBJECT IDENTIFIER ::= { esafeMib 2 }

esafeMibCompliances OBJECT IDENTIFIER ::= { esafeMibConformance 1 }

esafeMibGroups OBJECT IDENTIFIER ::= { esafeMibConformance 2 }

-- Compliance statements

esafeMibBasicCompliance MODULE-COMPLIANCE

STATUS current

DESCRIPTION

"The compliance statement for eSAFE MIB objects."

MODULE -- eSAFE-MIB

```

-- unconditionally mandatory groups

MANDATORY-GROUPS {
    esafeBaseGroup
}

-- conditionally mandatory groups

GROUP esafePsMibGroup
    DESCRIPTION
        "This group is implemented only by eDOCSIS devices
        that implement an embedded Portal Services logical
        element (ePS) compliant with CableLabs
        CableHome specifications."

GROUP esafeErouterMibGroup
    DESCRIPTION
        "This group is implemented only by eDOCSIS devices
        that implement a DOCSIS embedded router (eRouter)
        element compliant with CableLabs DOCSIS eRouter
        specifications."
    ::= { esafeMibCompliances 1 }

-- eSAFE Base Group Declarations

esafeBaseGroup OBJECT-GROUP
    OBJECTS {
        esafeProvisioningStatusProgress,
        esafeProvisioningStatusFailureFound,
        esafeProvisioningStatusFailureFlow,
        esafeProvisioningStatusFailureEventID,
        esafeProvisioningStatusFailureErrorText,
        esafeProvisioningStatusLastUpdate,
        esafeDevServiceIntImpact,
        esafeDevServiceIntImpactInfo
    }
    STATUS current
    DESCRIPTION
        "Group of eSAFE Base objects in the eSAFE MIB."
    ::= { esafeMibGroups 1 }

-- PS MIB Group

esafePsMibGroup OBJECT-GROUP
    OBJECTS {
        esafePsCableHomeModeControl,
        esafePsCableHomeModeStatus
    }
    STATUS current
    DESCRIPTION
        "Group of embedded PS-specific objects
        in the eSAFE MIB."
    ::= { esafeMibGroups 2 }

-- eRouter MIB Group

esafeErouterMibGroup OBJECT-GROUP
    OBJECTS {
        esafeErouterAdminMode,
        esafeErouterOperMode,
        esafeErouterPhysAddress
    }

```

```
STATUS      current
DESCRIPTION
    "Group of embedded Router-specific objects
    in the eSAFE MIB."
:: = { esafeMibGroups 3 }
```

END

Annex C

Format and content for eCM/eSTB event, SYSLOG, and SNMP trap extensions

(This annex forms an integral part of this Recommendation)

To facilitate device provisioning and fault management, the eCM of a Set-top Device MUST support the DOCSIS Event extensions defined in this annex.

This annex is an extension of Annex D Format and Content for Event, SYSLOG, and SNMP Trap (normative) of [DOCSIS2-OSSI].

Table C.1 – eDOCSIS events extensions

| Set-top device software upgrade | | | | | | | |
|---------------------------------|----------------------------|-------------|--|---------------------------|----------------|----------|-------------------------------|
| Process | Sub-process | CM priority | Event message | Message notes and details | Error code set | Event ID | Trap name |
| SW Upgrade | SW Upgrade General Failure | Notice | DOCSIS SSD not supported | | H01.1 | 72000101 | |
| SW Upgrade | Verification of CVC | Error | Set-top Device CVC validation failure for CVC distributed through the eSTB | | H01.2 | 72000102 | docsDevCmSwUpgradeCVCFailTrap |
| SW Upgrade | SW Upgrade Init | Notice | Set-top Device code file download initialized through the eSTB | | H01.3 | 72000103 | docsDevCmSwUpgradeInitTrap |
| SW Upgrade | SW Upgrade General Failure | Error | Set-top Device code file download through the eSTB failed | | H01.4 | 72000104 | docsDevCmSwUpgradeFailTrap |
| SW Upgrade | SW Upgrade Success | Notice | Set-top Device code file success fully downloaded through the eSTB | | H01.5 | 72000105 | docsDevCmSwUpgradeSuccessTrap |

SERIES OF ITU-T RECOMMENDATIONS

| | |
|-----------------|--|
| Series A | Organization of the work of ITU-T |
| Series D | General tariff principles |
| Series E | Overall network operation, telephone service, service operation and human factors |
| Series F | Non-telephone telecommunication services |
| Series G | Transmission systems and media, digital systems and networks |
| Series H | Audiovisual and multimedia systems |
| Series I | Integrated services digital network |
| Series J | Cable networks and transmission of television, sound programme and other multimedia signals |
| Series K | Protection against interference |
| Series L | Construction, installation and protection of cables and other elements of outside plant |
| Series M | Telecommunication management, including TMN and network maintenance |
| Series N | Maintenance: international sound programme and television transmission circuits |
| Series O | Specifications of measuring equipment |
| Series P | Telephone transmission quality, telephone installations, local line networks |
| Series Q | Switching and signalling |
| Series R | Telegraph transmission |
| Series S | Telegraph services terminal equipment |
| Series T | Terminals for telematic services |
| Series U | Telegraph switching |
| Series V | Data communication over the telephone network |
| Series X | Data networks, open system communications and security |
| Series Y | Global information infrastructure, Internet protocol aspects and next-generation networks |
| Series Z | Languages and general software aspects for telecommunication systems |