Data-Over-Cable Service Interface Specifications

eDOCSIS™ Specification

CM-SP-eDOCSIS-I07-051209

ISSUED

Document Status Sheet

Document Control Number:	CM-SP-eDOCSIS-I07-051209			
Document Title:	Data-Over-Cable Service Interface Specifications eDOCSISP™P Specification			
Revision History:	I01 - Issued March 12, 2003 I02 - Issued November 17, 2003 I03 - Issued August 4, 2004 I04 - Issued November 24, 2004 I05 - Issued April 8, 2005 I06 - Issued August 12, 2005 I07 – Issued December 9, 2005			
Date:	December 9, 2005			
Status:	~~Work in Progress~~	~~Draft~~	Issued	~~Closed~~
Distribution Restrictions:	~~Author Only~~	~~CL/Member~~	~~CL/ Member/ Vendor~~	Public

Key to Document Status Codes:

Work in Progress	An incomplete document, designed to guide discussion and generate feedback, that may include several alternative requirements for consideration.
Draft	A document in specification format considered largely complete, but lacking review by Members and vendors. Drafts are susceptible to substantial change during the review process.
Issued	A stable document, which has undergone rigorous member and vendor review and is suitable for product design and development, cross-vendor interoperability, and for certification testing.
Closed	A static document, reviewed, tested, validated, and closed to further engineering change requests to the specification through CableLabs.

Trademarks:

Contents

Figures

Tables

1 INTRODUCTION

Existing DOCSIS specifications were created for stand-alone cable modems that provide high-speed broadband services using the hybrid-fiber-coaxial cable infrastructure. The emergence of a class of devices that embeds additional functionality with a Cable Modem, such as packet-telephony, home networking and video, has necessitated the creation of this specification to define additional requirements such as interfaces, management and provisioning models. This is necessary to ensure that the Cable Modem will function properly and interact properly with the embedded Service/Application Functional Entities (eSAFEs).

1.1 Scope

This specification defines additional features that must be added to a DOCSIS Cable Modem for implementations that embed the Cable Modem with another application, such as a PacketCable MTA.[1]

1.2 Goals

The goals for this specification are:

- To preserve functional separation of the DOCSIS cable modem entity from eSAFEs within the eDOCSIS Device, so that existing DOCSIS cable plant integrity, cable modem configuration, management and provisioning security are not compromised.
- To isolate DOCSIS cable modem functionality so that specification 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.[2]

[1] Edited per EDOCS-N-03099, 11/11/03, po
[2] Edited per EDOCS-N-03099, 11/11/03, po

1.3 DOCSIS Base Specifications

There are currently three versions of what are in this document referred to as the DOCSIS Base Specifications. These three versions are commonly referred to as DOCSIS 1.0, DOCSIS 1.1, and DOCSIS 2.0. A list of the document categories in the Data-Over-Cable Service Interface Specifications family is provided below. For updates, please refer to http://www.cablemodem.com/.

Designation			Title
DOCSIS 1.0	**DOCSIS 1.1**	**DOCSIS 2.0**	
SP-RFI	SP-RFIv1.1	SP-RFIv2.0	Radio Frequency Interface Specification
SP-OSSI	SP-OSSIv1.1	SP-OSSIv2.0	Operations Support System Interface Specification
SP-BPI	SP-BPI+	SP-BPI+	Baseline Privacy Interface Specification
SP-CMCI			Cable Modem to Customer Premises Equipment Interface Specification
SP-CMTS-NSI			Cable Modem Termination System Network Side Interface Specification

1.4 Requirements

Throughout this document, 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 specification.
MUST NOT	This phrase means that the item is an absolute prohibition of this specification.
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 behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior 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.

2 REFERENCES

2.1 References (normative)

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

[DOCSIS-CMCI] DOCSIS Cable Modem to Customer Premise Equipment Interface Specification, CM-SP-CMCI-I10-050408, April 8, 2005, Cable Television Laboratories, Inc.

[RFI 1.0] ANSI/SCTE 22-1 2002: DOCSIS 1.0 Radio Frequency Interface.

[RFI 1.1] DOCSIS Radio Frequency Interface Specification, SP-RFIv1.1-I10-030730, July 30, 2003, Cable Television Laboratories, Inc.

[RFI 2.0] DOCSIS Radio Frequency Interface Specification, CM-SP-RFIv2.0-I10-051209, December 9, 2005, Cable Television Laboratories, Inc.

[OSSI 1.0] ANSI/SCTE 22-3 2002: DOCSIS 1.0 Operations Support System Interface.

[OSSI 1.1] DOCSIS Operations Support System Interface Specification, SP-OSSIv1.1-I07-030730, July 30, 2003, Cable Television Laboratories, Inc.

[OSSI 2.0] DOCSIS Operations Support System Interface Specification, CM-SP-OSSIv2.0-I09-050812, August 12, 2005, Cable Television Laboratories, Inc.

[RFC 1493] IETF RFC 1493, Definitions of Managed Objects for Bridges, E. Decker, P. Langille, A. Rijsinghani & K. McCloghrie, July 1993.

[RFC 2011] IETF RFC 2011, SNMPv2 Management Information Base for the Internet Protocol using SMIv2, K. McCloghrie, November 1996.

[RFC 2863] IETF RFC 2863, The Interfaces Group MIB, K. McCloghrie, F. Kastenholz, June 2000.

[RFC 2131] IETF RFC 2131, Dynamic Host Configuration Protocol, Droms, R., March 1997.

[RFC 2132] IETF RFC 2132, DHCP Options and BOOTP Vendor Extensions, Alexander, S., and R. Droms, March 1997.

[RFC 3396] IETF RFC 3396, Encoding Long Options in the Dynamic Host Configuration Protocol (DHCPv4), Lemon, T., and S. Cheshire, November, 2002.[3]

[RFC 791] IETF STD5, RFC 791, Internet Protocol, J. Postel, September 1981.

[RFC 768] IETF STD6, RFC 768, User Datagram Protogram, J. Postel, September, 1980.

[3] Edited per EDOCS-N-03084/ 11/11/03, po

[DSG]	DOCSIS Set-Top Gateway (DSG) Specification, CM-SP-DSG-I06-051209, December 9, 2005, Cable Television Laboratories, Inc. [4]
[OC]	OpenCable Host 2.0 Core Functional Requirements, OC-SP-HOST2.0-CFR-I06-050708, July 8, 2005, Cable Television Laboratories, Inc.[5]
[CCIF 2.0]	CableCARD™ Interface 2.0 Specification, OC-SP-CCIF2.0-I03-051117, November 17, 2005, Cable Television Laboratories, Inc. 6
[BPI+]	DOCSIS Baseline Privacy Plus Interface Specification, CM-SP-BPI+-I12-050812, August 12, 2005, Cable Television Laboratories, Inc. [7]
[DOCSIS RFI]	Refers to both [RFI 1.1] and [RFI 2.0].
[DOCSIS OSS]	Refers to both [OSSI 1.1] and [OSSI 2.0].

2.2 References (informative)

[CH1.0]	CableHome 1.0 Specification, CH-SP-CH1.0-I05-030801, August 1, 2003, Cable Television Laboratories, Inc.
[PC10-MTA]	PacketCable MTA Device Provisioning Specification, PKT-SP-PROV-I11-050812, August 12, 2005, Cable Television Laboratories, Inc.
[PC10-MIB]	PacketCable MIBs Framework Specification, PKT-SP-MIBS-I10-050812, August 12, 2005, Cable Television Laboratories, Inc.
[CH1.1]	CableHome 1.1 Specification, CH-SP-CH1.1-I09-050812, August 12, 2005, Cable Television Laboratories, Inc. [8]

2.3 Reference Acquisition

CableLabs Specifications:

- Cable Television Laboratories, Inc., 858 Coal Creek Circle, Louisville, CO 80027; Phone 303-661-9100; Fax 303-661-9199; Internet: http://www.cablelabs.com /
- SCTE Standards, Society of Cable Telecommunication Engineers, 140 Philips Road, Exton PA 19341-1318, Phone 610-524-1725, www.scte.org
- IETF Standards, Internet Engineering Task Force (IETF) Secretariat c/o Corporation for National Research Initiatives, 1895 Preston White Drive, Suite 100, Reston, VA 20191-5434, Phone 703-620-8990, Fax 703-620-9071, Internet: www/ietf.org

[4] Added per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[5] Added per eDOCSIS-N-04.0151-3 by kb 7/18/04; made normative per eDOCSIS-N-05.0210-2 by kb 3/15/05.
[6] Added per eDOCSIS-N-04.0154-2 by kb 7/18/04; Revised per eDOCSIS-N-05.0260-2 by GO on 11/30/05.
[7] Added per eDOCSIS-N-04.0154-2 by kb 7/18/04.
[8] Edited per EDOCS-N-03084/ 11/11/03, po

3 GLOSSARY

CMCI Cable Modem (CM) to Customer Premise Equipment (CPE) Interface as defined in [DOCSIS-CMCI]

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

eDOCSIS eDOCSIS is the embedded DOCSIS specification that defines the interface between the eCM and an eSAFE.

eDOCSIS Device An eDOCSIS Device is one that includes an eCM entity, one or more eSAFEs and supports a single software image download that is used for the entire device.[9]

Embedded Security eSTB An eSTB with integrated security functions.

eMTA Embedded Multimedia Terminal Adaptor. An embedded version of an MTA.

E-MTA Embedded MTA device. An eDOCSIS device that contains both an eMTA and an eCM.

ePS Embedded Portal Service Element. A CableHome-compliant eSAFE that provides management and network address translation functions between the DOCSIS network and the home network.

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 [OC]. [10]

LCI Logical CPE Interface. A bi-directional or uni-directional data-only logical 802.3/Ethernet MAC frame interface between eCM and an eSAFE. [11]

MTA Multimedia Terminal Adapter as defined in [PC10-MTA]. Contains the interface to a physical voice device, a network interface, CODECs, and all signaling and encapsulation functions required for VoIP transport, class features signaling and QoS signaling.

eSAFE (embedded Service/Application Functional Entity) An embedded version of CableLabs-specified application, such as a PacketCable Multimedia Terminal Adapter (MTA), that provides a service using the DOCSIS IP platform, or a function or set of functions, such as the CableHome Portal Services logical element, that supports the delivery of one or more services over an IP platform.

OpenCable Host eSTB An eSTB device built to OpenCable Host specifications

[9] Edited per EDOCS-N-03099, 11/11/03, po; eDOCSIS-N-05.0223-3 by kb 7/18/05.
[10] Added per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[11] Revised per eDOCSIS-N-04.0160-1 by kb 7/28/04.

4 ABBREVIATIONS

This specification uses the following abbreviations:[12]

CM	Cable Modem
CMCI	Cable Modem to Customer Premises Equipment Interface
DOCSIS	Data-Over-Cable Service Interface Specifications
DSG	DOCSIS Set-top Gateway
eCM	Embedded Cable Modem
eDOCSIS	Embedded DOCSIS
eMTA	Embedded MTA
ePS	Embedded Portal Services Element
eSTB	Embedded Set-Top Box
eSAFE	Embedded Service/Application Functional Entity
LCI	Logical CPE Interface
SLED	Software Loopback for eDOCSIS

[12] Table revised per ECNs eDOCSIS-N-04.0151-3, eDOCSIS-N-04.0160-1

5 EMBEDDED DOCSIS CABLE MODEM

5.1 Device Interface Reference Model

Referring to Figure 5-1, an eDOCSIS device consists of an embedded DOCSIS 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 CableHome Portal Services Logical Element [CH1.0]
- eMTA: embedded PacketCable Multimedia Terminal Adapter [PC10-MTA], [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 [OC].[13]

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

Figure 5-2 presents a typical CableHome 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 PacketCable E-MTA (with DOCSIS cable modem) 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.[14]

Figure 5-7 presents a logical view of protocol stacks for an eCM to eSTB interface (OpenCable Host 2.0).[15]

Figure 5-8 presents a typical embedded security STB eDOCSIS Device reference model.[16]

Figure 5-9 presents a logical view of protocol stacks for an eCM to eSTB interface (embedded security STB).[17]

[13] Added per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[14] Figure added per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[15] Figure added per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[16] Figure added per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[17] Figure added per eDOCSIS-N-04.0151-3 by kb 7/18/04.

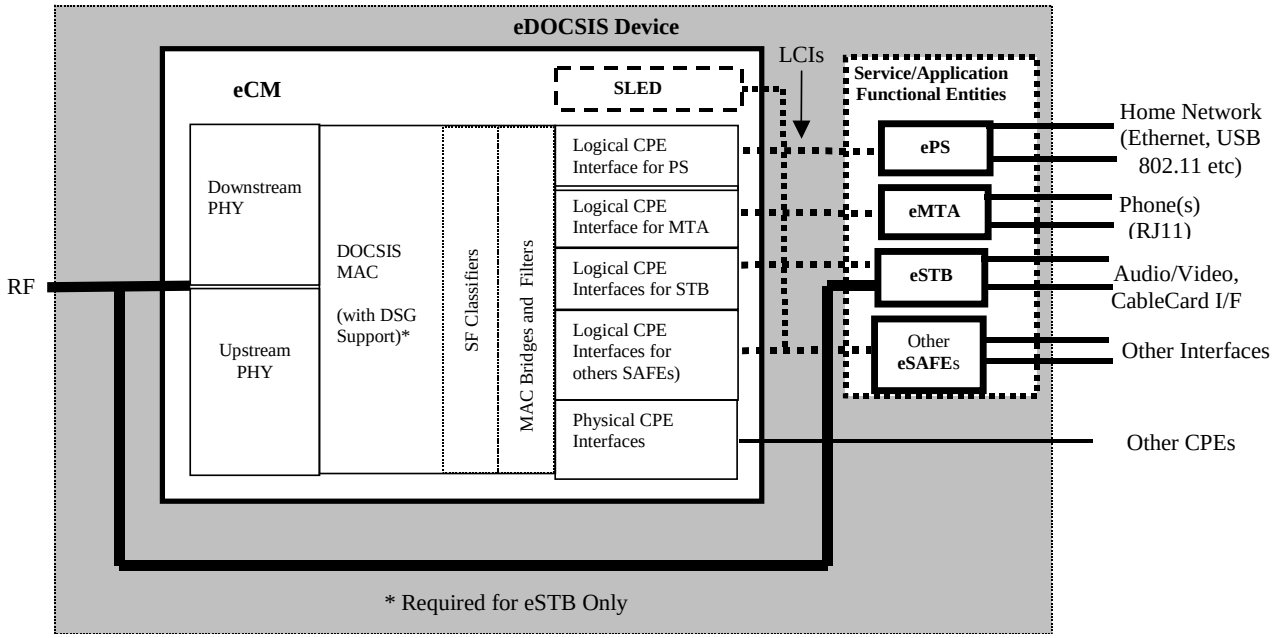

Figure 5-1 - eDOCSIS Reference Model

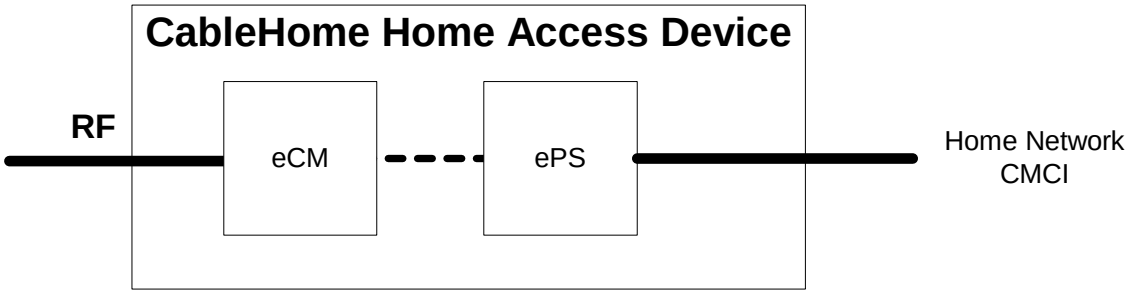

Figure 5-2 - CableHome Home Access eDOCSIS Device Reference Model

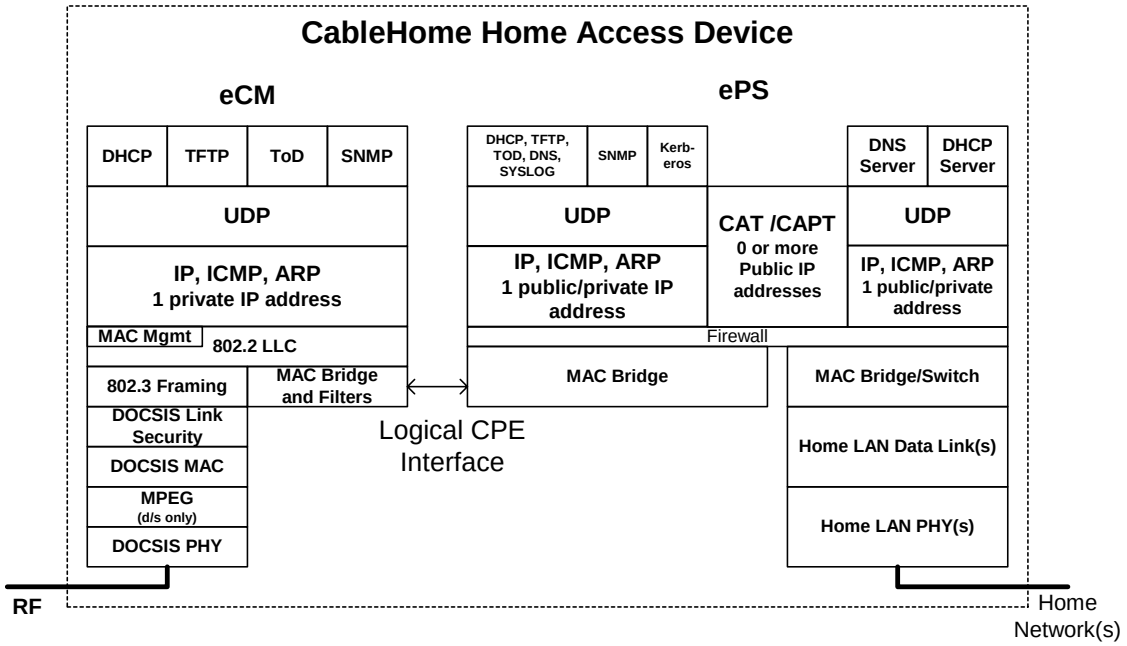

Figure 5-3 - eCM - ePS Protocol Stacks

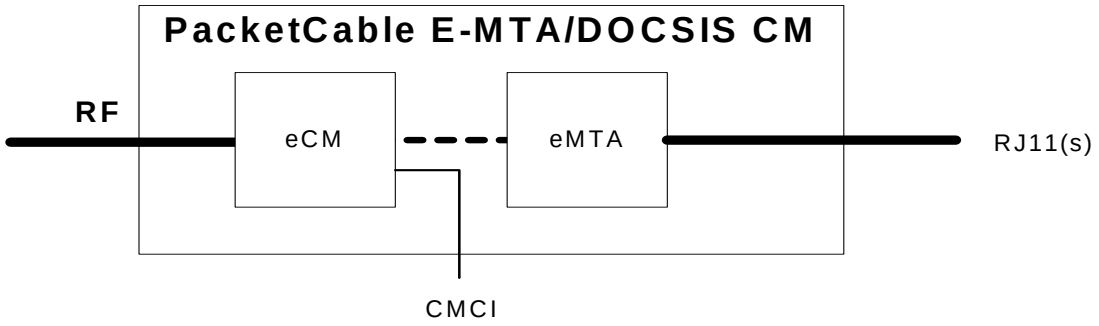

Figure 5-4 - PacketCable E-MTA (with DOCSIS CM) eDOCSIS Reference Model

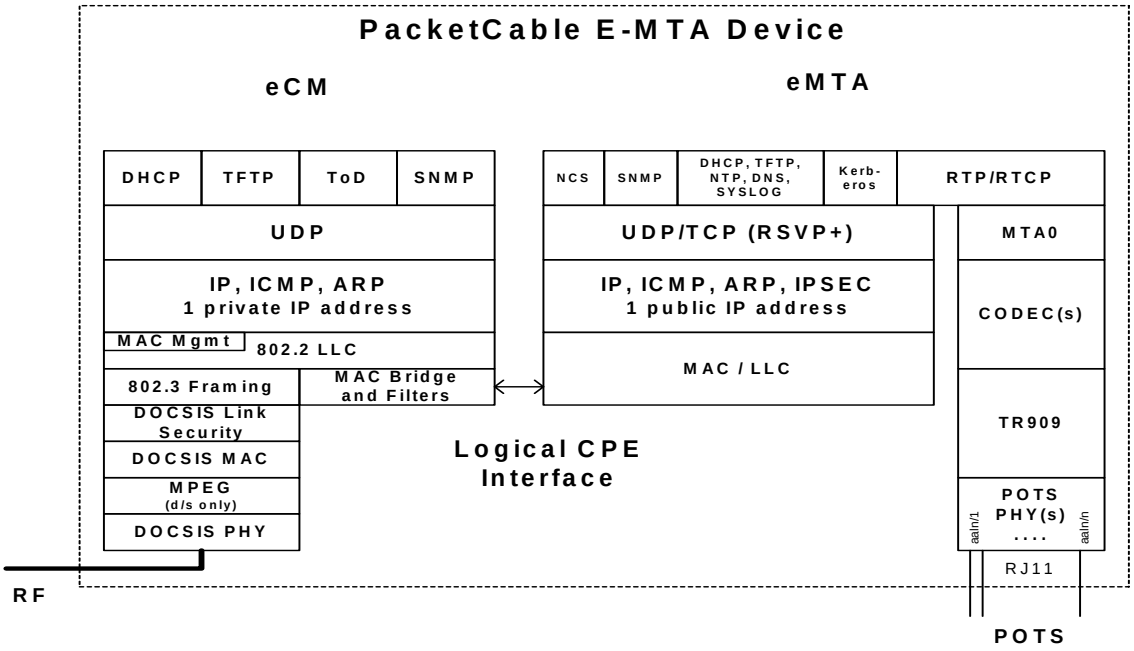

Figure 5-5 - eCM - eMTA Protocol Stacks

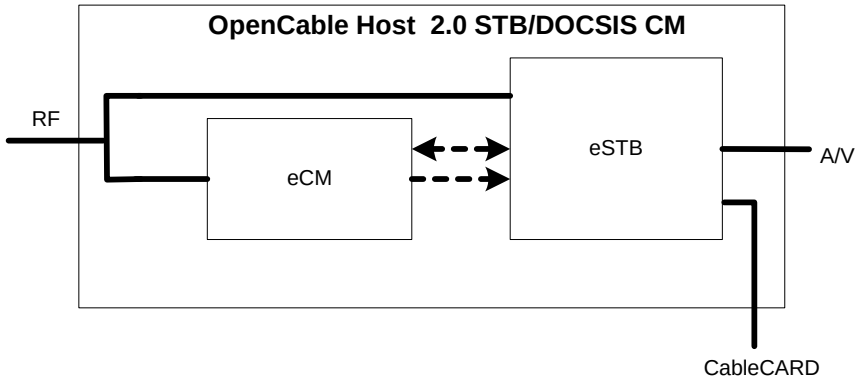

Figure 5-6 - OpenCable Host 2.0 eDOCSIS Reference Model

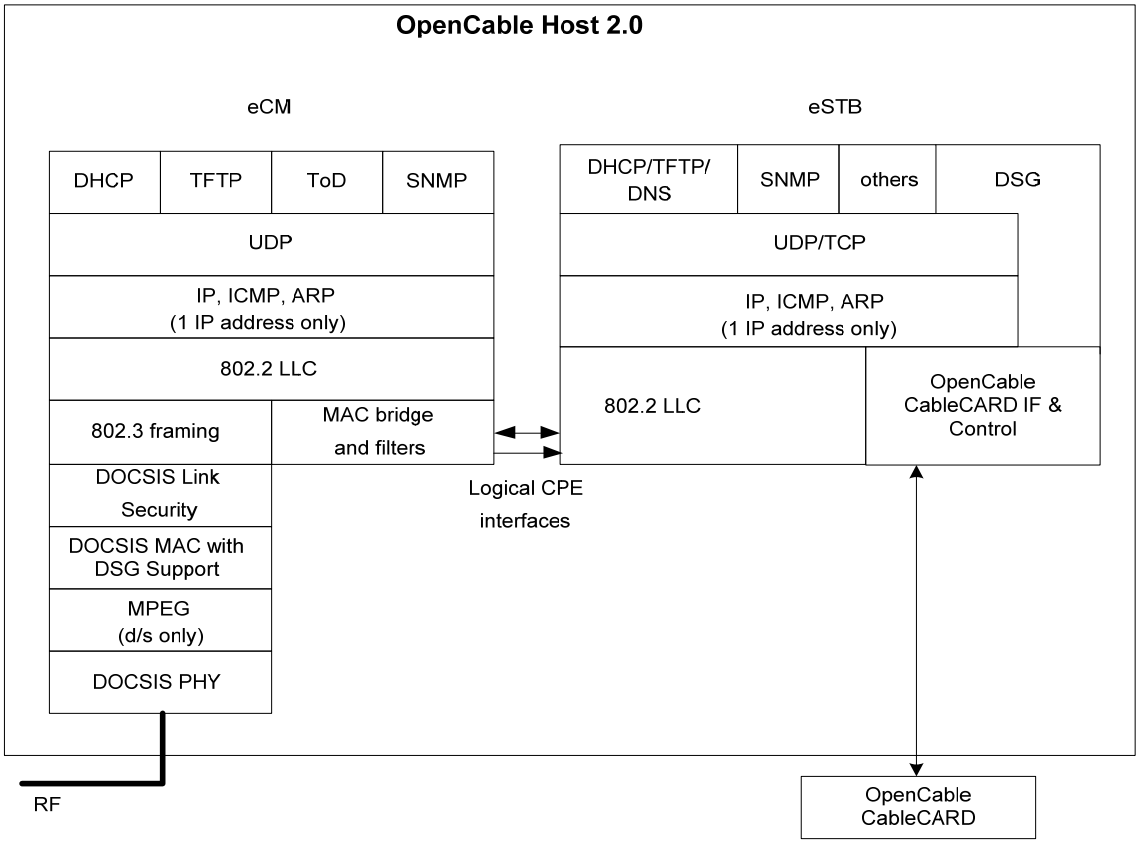

Figure 5-7 - eCM – eSTB Protocol Stacks – OpenCable Host 2.0

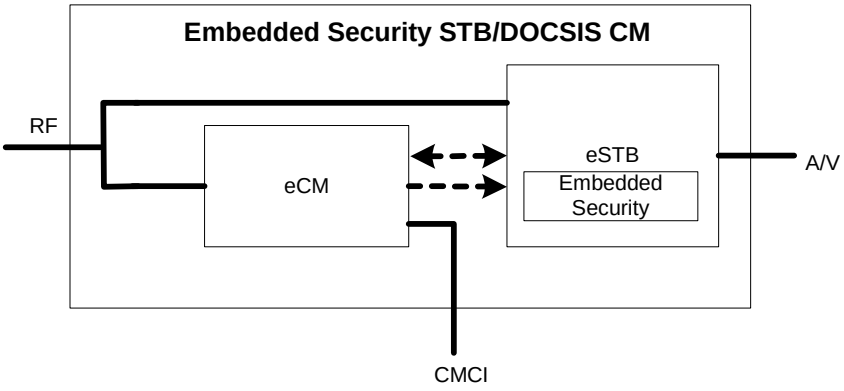

Figure 5-8 - Embedded Security STB eDOCSIS Reference Model

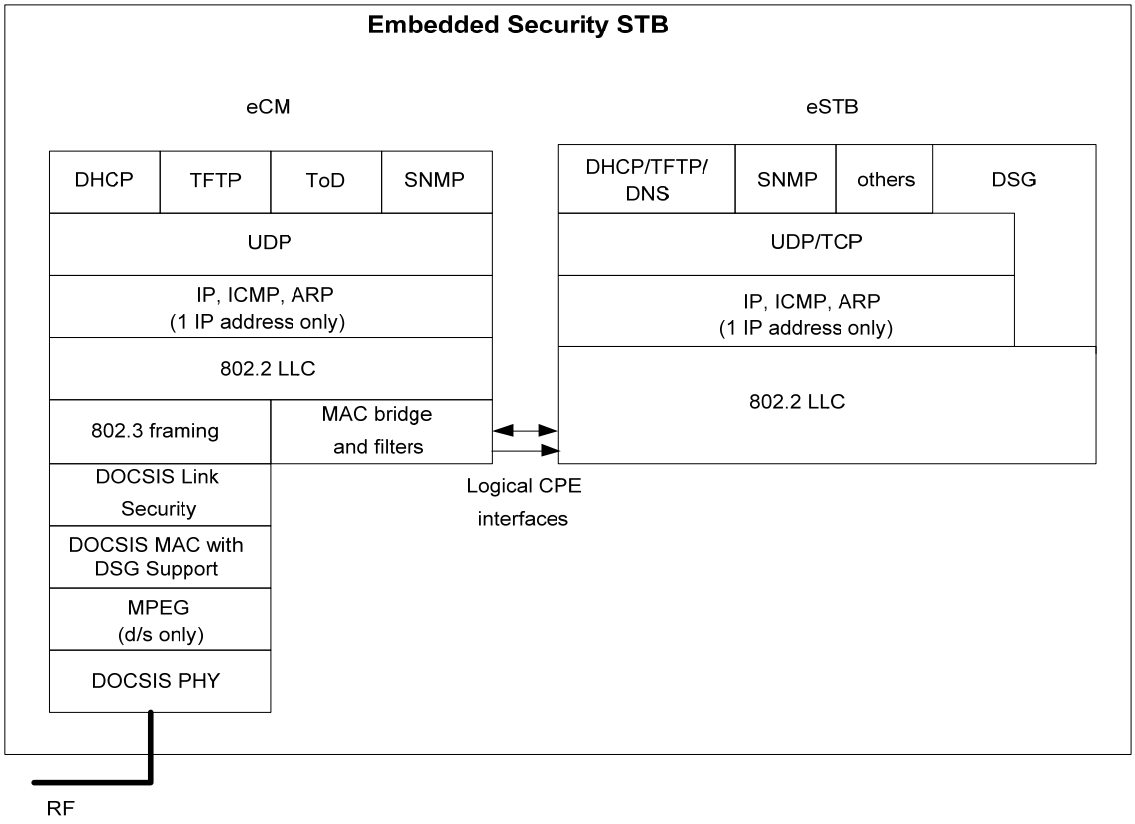

Figure 5-9 - eCM – eSTB Protocol Stacks – Embedded Security STB

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 DOCSIS-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:

- the snmpGroup, systemGroup, udpGroup, icmpGroup and ipGroup (with the exception of ipNetToMediaTable and ipAddrTable within the ipGroup which MUST NOT be shared).
- the mibs rooted under snmpV2 (1.3.6.1.6).

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 (labeled 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].[18]

An eCM, MUST meet the requirements of an equivalent standalone cable modem as specified in the applicable DOCSIS Base Specifications In case any requirement in this document conflicts with a requirement in the DOCSIS Base Specifications (i.e., DOCSIS 1.0, 1.1 and DOCSIS 2.0), the requirement in this document takes precedence for any eDOCSIS Device.[19]

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 [DOCSISx-RFI][20]. In particular:[21]

- MAC addresses of each eSAFE MUST be 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.[22]
- Packet forwarding and filtering rules defined in [DOCSISx-RFI] specification MUST apply to both logical and physical interfaces to eSAFEs as defined in this specification and in [DOCSISx-OSSI]23.
- 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 [DOCSISx-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 [DOCSISx-OSSI] and as detailed below.

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

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

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

[18] Paragraph added per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[19] Edited per EDOCS-N-03099, 11/11/03, po
[20][DOCSISx-RFI] is a shorthand notation for [RFI 1.0], [RFI 1.1], and [RFI 2.0]
[21] Section modified per eDOCSIS-N-04.0150-2, 7/12/04, kb
[22] Bullet added per eDOCSIS-N-04.0182-2 by kb 11/2/04.
[23][DOCSISx-OSSI] is a shorthand notation for [OSSI 1.0], [OSSI 1.1], and [OSSI 2.0]

- 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:[24]

- 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 CableCARD, DSG Clients, and A/V interfaces, etc) which are not directly and physically connected to the eCM.

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

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

Table 5-1 summarizes the ifIndex assignments in the eCM. Table 5-2 defines the details of the ifTable entries that MUST be supported by ePS, eMTA and eSTB.[25]

Table 5-1 - eDOCSIS ifTable Interface Designations

Interface	Type
1	Primary CPE interface (CableHome ePS WAN interface)
2	CATV-MAC
3	RF-downstream channel
4	RF-Upstream channel
5 – 15	Other CPE Interfaces
16	Reserved for PacketCable/eMTA
17	Reserved for eSTB-IP[26]
18	Reserved for eSTB-DSG[27]
19 – 31	Reserved for Other eDOCSIS Interfaces

An eDOCSIS compliant eCM can have multiple CPE interfaces, as well as interfaces to multiple eSAFEs. 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

[24] Paragraph and bullets added per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[25] Sentence modified per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[26] Row added per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[27] Row added per eDOCSIS-N-04.0151-3 by kb 7/18/04.

CPE Interface" (ifIndex 1), the eCM MUST NOT apply these filters to ifIndex 16 through 31, which are reserved as interfaces to eSAFEs. [28]

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.[29]

Table 5-2 - [RFC 2863] ifTable, MIB-Object Details for eDOCSIS Device Interfaces

[RFC 2863] MIB-Object details for eCM-eSAFE Interfaces	ePS	eMTA	eSTB[30]	
			eSTB-IP	eSTB-DSG
ifIndex	1	16	17	18
ifDescr: MUST match the text	"CableHome Embedded Interface"	"PacketCable Embedded Interface[31]"	"Set-Top Box Embedded IP Interface"	"Set-Top Box Embedded DSG Interface"
ifType	other(1)	other(1)	other (1)	other (1)
ifMtu	0	0	0	0
ifSpeed	0	0	0	0
ifPhysAddress	<empty-string>	<empty-string>[32]	<empty-string>	<empty-string>
ifAdminStatus: Only up/own control 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)
ifOperStatus: Only up/own control 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)
ifLastChange	<per [RFC 2863]>	<per [RFC 2863]>[33]	<per[RFC 2863]>	<per [RFC 2863]>
ifInOctets	(n)	(n)	(n)	Deprecated
ifInUCastPkts	(n)	(n)	(n)	Deprecated
ifInNUCastPkts	Deprecated	Deprecated	Deprecated	Deprecated
ifInDiscards	0	0	0	0
ifInErrors	0	0	0	0
ifUnknownProtos	0	0	0	0
ifOutOctets	(n)	(n)	(n)	(n)

[28] Paragraph added by eDOCSIS-N-04.0153-2 by kb
[29] Paragraph added by eDOCSIS-N-04.0153-2 by kb
[30] Columns added per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[31] Edited per EDOCS-N-03099, 11/11/03, po
[32] Edited per EDOCS-N-03099, 11/11/03, po
[33] Edited per EDOCS-N-03099, 11/11/03, po

[RFC 2863] MIB-Object details for eCM-eSAFE Interfaces	ePS	eMTA	eSTB[30]	
			eSTB-IP	eSTB-DSG
ifOutUCastPkts	(n)	(n)	(n)	(n)
ifOutNUCastPkts	Deprecated	Deprecated	Deprecated	Deprecated
ifOutDiscards	0	0	0	0
ifOUtErrors	0	0	0	0
ifOutQlen	Deprecated	Deprecated	Deprecated	Deprecated
ifSpecific	Deprecated	Deprecated	Deprecated	Deprecated

5.2.3.2 [RFC 2011] ipNetToMediaTable Requirements

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

Table 5-3 - [RFC 2011] ipNetToMedia MIB-Object Details for eDOCSIS Device Interfaces

[RFC 2011] MIB-Object details for eCM-eSAFE Interfaces	ePS	eMTA	eSTB-IP[34]
ipNetToMediaIfIndex	1	16	17
ipNetToMediaPhysAddress	WAN-Man MAC Address	MTA MAC Address	STB 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
ipNetToMediaType	static(4)	static(4)	static(4)

5.2.3.3 [RFC 1493] Requirements[35]

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

The eCM MUST support all bridge statistics of the [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.

Within each dot1dTpFdbTable entry for an eSAFE, the dot1dTpFdbAddress value MUST be the 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.

[34] Column added per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[35] Section replaced per eDOCSIS-N-04.0158-2 by kb 7/18/04.

An eDOCSIS device containing an eSTB MUST implement the objects from the [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-dependant 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 5-4 - 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.[36]

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 addresses already established via DSG operation.[37]

5.2.4 DHCP Option 43 Syntax Requirements

In order to facilitate device provisioning, all eDOCSIS devices will use DHCP Option 43 during registration process for providing vendor class identification, embedded component and vendor specific capability enumerations.

[36] Paragraph added per eDOCSIS-N-04.0182-2 by kb 11/2/04.
[37] Paragraph added per eDOCSIS-N-04.0182-2 by kb 11/2/04.

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 5-5.

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 CableLabs, sub-options 11-127 are reserved for future CableLabs use, 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 5-5. 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 [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 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 5-5 for the intended purpose of each sub-option.

The eCM MAY include Option 43 sub-option 1 in DHCPDISCOVER and DHCPREQUEST 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.[38]

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).

[38] Paragraph modified per eDOCSIS-N-04.0123-2 and eDOCSIS-N-04.0134-2 by kb 7/18/04.

Defined eSAFEs are: "EPS" for CableHome embedded Portal Services Element, "EMTA" for PacketCable embedded MTA, and "ESTB" for embedded set-top box.[39]

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[40]-octet, hexidecimally-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.

In eDOCSIS, DHCP Option 43 sub-option 11-127 are reserved for eSAFEs and CableLabs, and MUST NOT be used for other purpose. The partitions are as follows:

- Sub-options 11 to 30: reserved for CableHome use,
- Sub-options 31 to 50: reserved for PacketCable use,
- Sub-options 51 to 127: reserved for future CableLabs use.

In eDOCSIS, DHCP Option 43 sub-option 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 [RFC 3396] to split the option into multiple smaller options.

Table 5-5 - 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.

[39] Paragraph modified per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[40] Changed from 3 to 6 by eDOCSIS-N-04.0123-2 by kb 7/30/04.

eCM DHCP Option 43	Value	Description
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, or 2.0 Base Specifications)
Sub-option 3[41]	"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", or "ESTB" corresponding to embedded MTA, embedded Portal Service Element, and embedded STB, respectively. For example: • "ECM:EMTA" = A PacketCable/Embedded MTA • "ECM:EPS" = A CableHome/Embedded Portal Services Element • "ECM:ESTB" = An Embedded STB • "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, hexidecimally-encoded, vendor-specific Organization Unique Identifier (OUI) that may match the OUI in eCM's MAC address.[42]
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-options 11 to 127		Reserved for CableLabs.
Sub-options 128 to 254		Reserved for vendors.

[41] Modified per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[42] Modified per eDOCSIS-N-04.0123-2 by kb 7/18/04.

5.2.4.4 Examples (Informative)

5.2.4.4.1 DOCSIS Cable Modem with Embedded PacketCable 1.0 MTA Example

Table 5-6 shows the syntax for DHCP Option 43 for the eCM in an E-MTA.

Table 5-6 - 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)[43]	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[44]
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 CableHome 1.1 PS Example

Table 5-7 shows the syntax for DHCP Options 43 for the eCM in a CableHome 1.1 E-PS device.

Table 5-7 - Example eDOCSIS Device: DOCSIS 1.1 CableHome/E-PS

DHCP DISCOVER Options	Value	Description
eCM DHCP DISCOVER		
eCM Option 43 sub-option 1	N/A (omitted because the list is empty)[45]	List of sub-options (within option 43) to be returned by server
eCM Option 43 sub-option 2	"ECM"	Embedded cable modem

[43] Modified per eDOCSIS-N-04.0123-2 by kb 7/18/04.
[44] Modified per eDOCSIS-N-04.0123-2 by kb 7/18/04.
[45] Modified per eDOCSIS-N-04.0123-2 by kb 7/18/04.

DHCP DISCOVER Options	Value	Description
eCM DHCP DISCOVER		
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[46]
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

[46] Modified per eDOCSIS-N-04.0123-2 by kb 7/18/04.

5.2.4.4.3 DOCSIS Cable Modem with Embedded STB Example[47]

Table 5-8 illustrates the syntax for DHCP Options 43 for the eCM with an embedded STB.

Table 5-8 - 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)[48]	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, hexidecimally-encoded, vendor-specific Organization Unique Identifier (OUI) that may match the OUI in the eCM's MAC address.
eCM Option 43 sub-option 9	"Xman200"	Device model number from <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

5.2.4.4.4 DOCSIS Cable Modem with Embedded PacketCable MTA and CableHome PS

Table 5-9 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, a PacketCable 1.0 eMTA and a CableHome 1.1 ePS.

[47] Section added per eDOCSIS-N-04.0151-3 by kb 7/18/04.
[48] Options Value and Description revised per eDOCSIS-N-04.0169-1 by kb 11/1/04.

Table 5-9 - 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	N/A (omitted because the list is empty)[49]	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 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[50]
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 [51]

5.2.5 Testability Requirements[52]

In order to verify conformance to this specification 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 specification 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.

5.2.5.1 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.

[49] Modified per eDOCSIS-N-04.0123-2 by kb 7/18/04.
[50] Modified per eDOCSIS-N-04.0123-2 by kb 7/18/04.
[51] Edited per EDOCS-N-03084, 11/11/03, po
[52] Section added per EDOCS-N-03088 11/17/03

5.2.5.2 SLED Protocol Description

5.2.5.2.1 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.[53]

Figure 5-10 illustrates the SLED reference model.

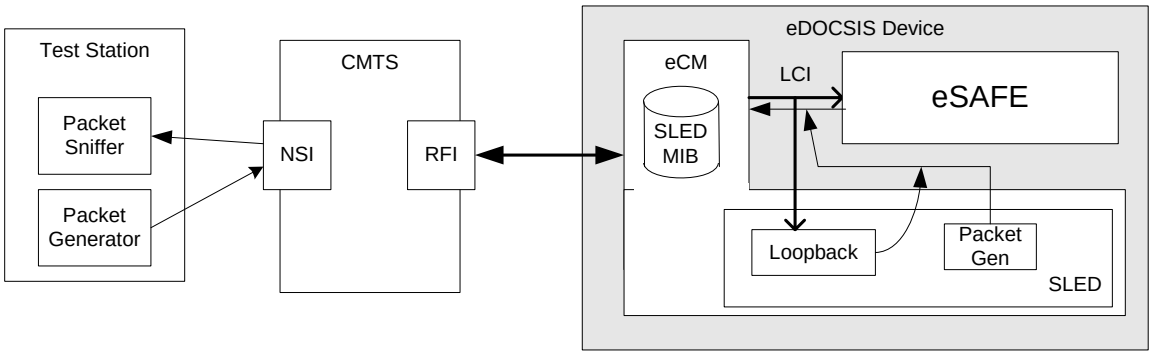

Figure 5-10 - SLED Reference Model

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 pre-defined 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.

[53] Paragraph added per eDOCSIS-N-04.0122-2 by kb 7/18/04.

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.

5.2.5.2.2 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[54]:
    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 [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 to 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 [RFC 791], and UDP Checksum, UDP Length per [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 [RFC 791]). Further, the final 8-bytes of *sledLoobpackPktHdr* (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.

[54] Note: The eCM MUST remove the Ethernet FCS/CRC32 before encapsulating and processing the packet for loopback. Because the CRC32 may not be present, may be incorrect, and is not relevant to the SLED loopback tests, it is omitted from the packet that is looped back. Note added per eDOCSIS-N-04.0122-2 by kb 7/18/04.

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 5-11 illustrates the SLED packet loopback encapsulation.[55]

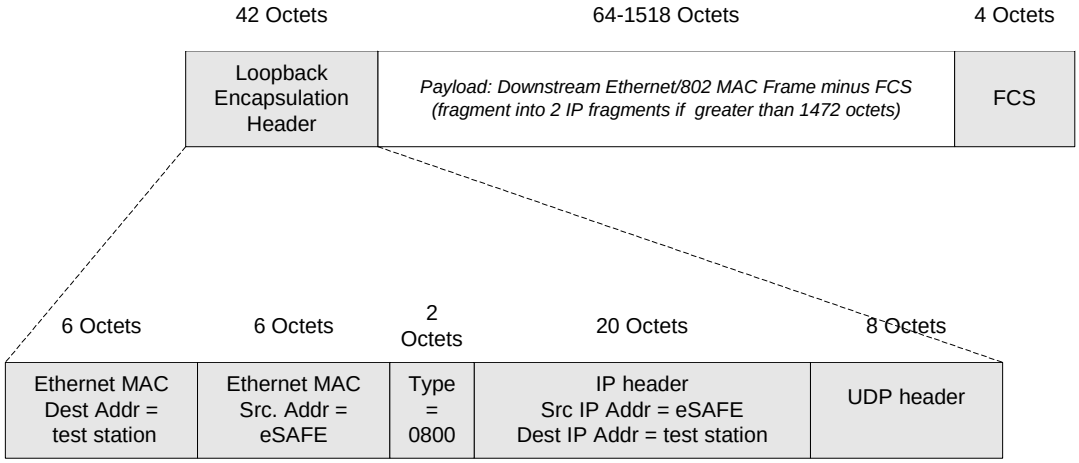

Figure 5-11 - SLED Packet Loopback Encapsulation

Figure 5-12 illustrates an example of the SLED loopback sequence.

5.2.5.2.3 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 msec 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.

[55] Figure modified per eDOCSIS-N-04.0122-2 by kb 7/18/04.

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 5-12 for an illustration of the SLED packet loopback and generation sequences.

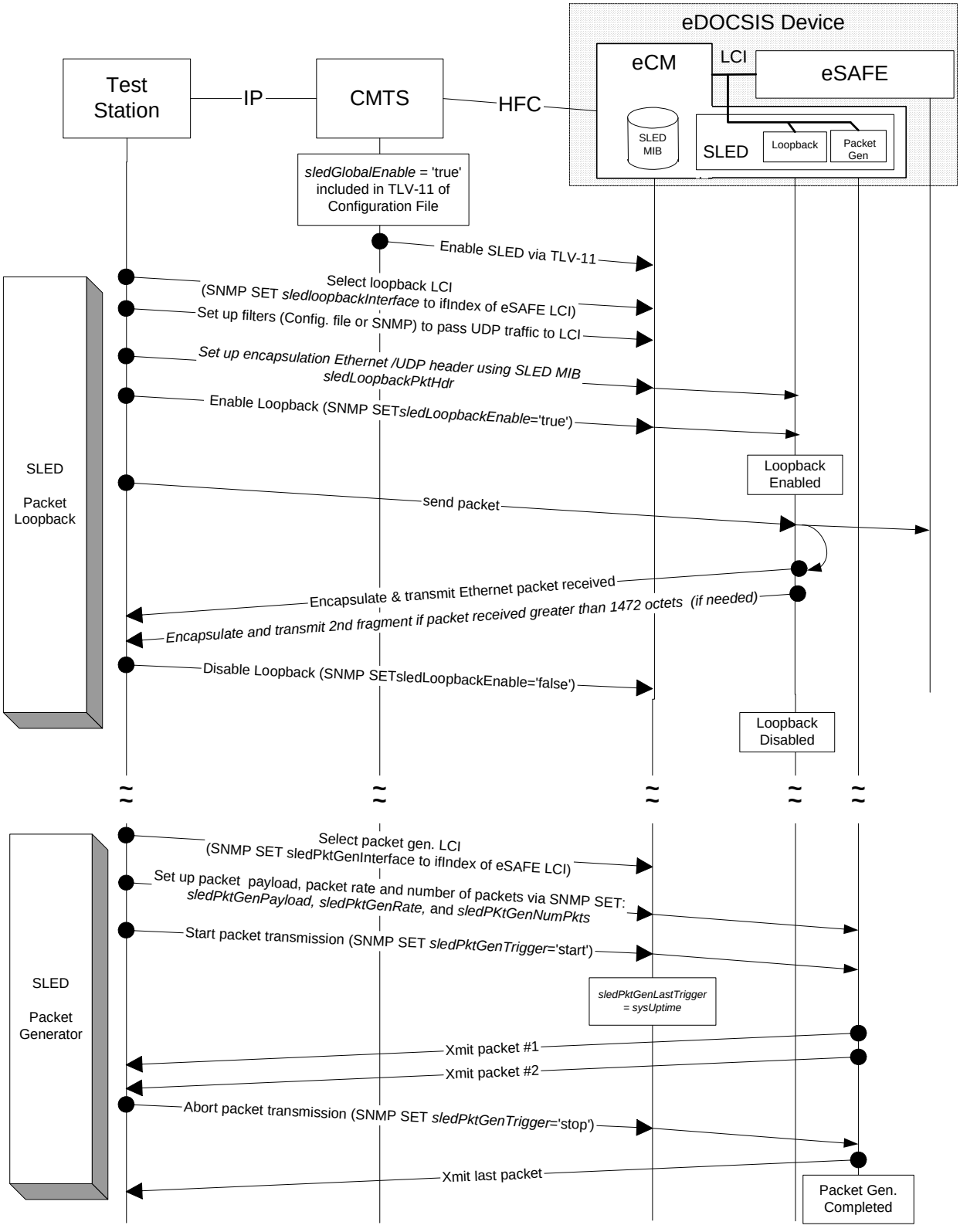

Figure 5-12 - SLED Packet Loopback And Generation Sequences

5.2.6 Firmware Download[56]

All eDOCSIS devices MUST support a single secured firmware image download that MUST be 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 (SSDL) mechanisms as specified in [BPI+], [DOCSIS RFI], and [DOCSIS OSS].

Firmware download for eDOCSIS devices that contain eSTBs (referred to as a Set-top Device) is defined in the following subsection.

5.2.6.1 eSTB-eDOCSIS Device Firmware Download[57]

5.2.6.1.1 Firmware Download for an eDOCSIS Device Implementing eSTB[58]

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

1. DSM-CC data carousel methods as defined in the CableCARD Interface 2.0 specification [CCIF 2.0],
2. DOCSIS Secure Software Download (SSDL) mechanisms as defined in [BPI+], [DOCSIS RFI], and [DOCSIS OSS].

OpenCable Host 2.0 Set-top devices support the above firmware download methods as specified in [OC] and [CCIF 2.0].

All other Set-top Devices that do not comply with [OC] MUST support at least one of 1) the DOCSIS SSDL method, and 2) all the DSM-CC Data Carousel methods [CCIF 2.0].

5.2.6.1.2 OSS Requirements for Set-top Devices that Do Not Support DOCSIS Secure Software Download

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

- The SNMP Set or Config File TLV MUST be ignored and the Device MUST remain capable of accepting new software through the non-DOCSIS firmware download mechanism.
- The eCM MUST report the attempt to trigger a TFTP firmware upgrade by logging the appropriate event (via an entry in the docsDevEvTable). The text of this event MUST explicitly indicate that the TFTP upgrade is not applicable for this Set-top Device.
- The docsDevSwOperStatus MUST remain other (5).

If the Set-top Device does not implement DOCSIS SSDL 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 Secure Software Download not supported"

[56] Section added per eDOCSIS-N-04.0154-2 by kb 7/18/04.
[57] Section modified per eDOCSIS-N-05.0223-3 by kb 7/18/05.
[58] Section modified per eDOCSIS-N-05.0210-2 by kb 3/15/05; Revised per ECN eDOCSIS-N-05.0260-2 by GO on 11/30/05.

If the Set-top Device does not implement DOCSIS SSDL mechanisms, its eCM MUST ignore any firmware upgrade related information in the CM Configuration File and MUST log the appropriate event (via an entry in the docsDevEvTable). The text of this event MUST explicitly indicate that the firmware upgrade related information is not applicable. The message MUST be sent at level “4 Error”.

Annex A SLED MIB Definition[59]

```
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       "200504080000Z" -- April 8, 2005
    ORGANIZATION       "CableLabs eDOCSIS Working Group"
    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 and invoke the Software Loopback Application
            for eDOCSIS (SLED) functionality."
    REVISION "200504080000Z" -- April 8, 2005
    DESCRIPTION
            "This revision is published as part of the CableLabs
            eDOCSIS Specification I05.

            Copyright 1999-2005 Cable Television Laboratories, Inc.
            All rights reserved."
    ::=  { 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
```

[59] Annex added per EDOCS-N-03088 11/17/03; revised per eDOCSIS-N-05.0212-2 by kb 3/14/05.

```
            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, the eCM removes the
            Ethernet FCS/CRC32 from the original packets. All Ethernet
            packets received by the SLED from the LCI are then processed
            as follows:[60]

            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))
```

[60] Paragraph modified per eDOCSIS-N-04.0122-2 by kb 7/18/04.

```
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
            "A properly formatted Ethernet(DIX)+IP+UDP headers 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      Unsigned32[61]
    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      Unsigned32[62]
    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 }
```

[61] Changed per eDOCSIS-N-04.0122-2 by kb 7/18/04.
[62] Changed per eDOCSIS-N-04.0122-2 by kb 7/18/04.

```
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

An eCM in an eDOCSIS device implementing an embedded CableHome Portal Services logical element (ePS) or/and embedded PacketCable Multimedia Terminal Adapter logical element (eMTA) as an eSAFE MUST implement the following eSAFE MIB.[63, 64, 65, 66 67]

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

    TruthValue,
    DateAndTime
              FROM SNMPv2-TC  -- RFC 2579

    SnmpAdminString         FROM SNMP-FRAMEWORK-MIB --RFC 3411

    ifIndex                 FROM IF-MIB

    clabProjDocsis          FROM CLAB-DEF-MIB
;

esafeMib MODULE-IDENTITY
    LAST-UPDATED "200504080000Z" –- April 8, 2005
    ORGANIZATION "CableLabs eDOCSIS Working Group"
    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."
    REVISION "200504080000Z" -- April 8, 2005
    DESCRIPTION
            "This revision is published as part of the CableLabs
            eDOCSIS Specification I05.

            Copyright 1999-2005 Cable Television Laboratories, Inc.
            All rights reserved."
    ::= { clabProjDocsis 14 }

-- Administrative assignments

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

[63] Annex added by eDOCSIS-N-04.0152-5 by kb.
[64] Imports section modified per eDOCSIS-N-04.0172-6 by kb 11/1/04.
[65] Added esafeDevStatusTable, esafeDevStatusEntry, esafeDevServiceIntImpact, esafeDevServiceIntImpactInfo per eDOCSIS-N-04.0172-6 by kb 11/2/04.
[66] Revised esafePsCableHomeModeControl per eDOCSIS-N-04.0176-2 by kb 11/2/04.
[67] Revised ProvisioningStatusProgress, ProvisioningStatusFailureFound, and ProvisioningStatusFailureErrorText per eDOCSIS-N-05.0208-4 by kb 3/14/05.

```
esafePsMibObjects   OBJECT IDENTIFIER ::= { esafeMibObjects 2 }
-- Placeholders
-- esafeMtaMibObjects  OBJECT IDENTIFIER ::= { esafeMibObjects 3 }
-- esafeStbMibObjects  OBJECT IDENTIFIER ::= { esafeMibObjects 4 }

-- 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."
    ::={ 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, and an eMTA will report a value
            such as MTA-1."
    REFERENCE
            "PacketCable Provisioning specification,
            Provisioning Overview section; CableHome
            specification, Provisioning Processes
            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."
    REFERENCE
            "PacketCable Provisioning specification,
            Appendix I Provisioning Events; CableHome
            specification, Appendix II Format and Content
            for Event, SYSLOG, and SNMP Trap."
    ::={ 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."
    REFERENCE
            "PacketCable Provisioning specification,
            PacketCable Management Event Mechanism specification;
            CableHome specification, Appendix II Format and Content
            for Event, SYSLOG, and SNMP Trap."
    ::={ 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
```

```
            Models 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 }


-- 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."
    ::= { 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 }

END
```

Appendix I Acknowledgements

The inter-disciplinary nature of this specification involves contributions from many individuals. On behalf of CableLabs and its participating member companies, I would like to extend our sincere appreciation to all those have contributed to the development of this specification. Special thanks are given to:

Ralph Brown ..... CableLabs

Eduardo Cardona..... CableLabs

Kevin Luehrs..... CableLabs

Matt Osman..... CableLabs

Joe Weber..... CableLabs

Greg White..... CableLabs

Margo Dolas..... Broadcom

Gordon Li..... Conexant

Sridhar Sharma..... Imedia

Greg Nakanishi ..... Motorola

Mariano Schain ..... Texas Instruments

Nancy Davoust..... YAS

Doug Jones..... YAS

Special thanks are also given to Liz Weeks of YAS for her skillful editing work making the aggressive release deadlines possible.

John Eng, CableLabs.

Appendix II Revision History

The following Engineering Change Notices have been incorporated in SP-eDOCSIS I02-031117.

ECN	Date Ratified	Summary
EDOCS-N-03084	8/20/03	Incorporate DHCP Option 43 for eDOCSIS.
EDOCS-N-03099	10/29/03	Corrections and Clarifications; alignment of rearranged text with J.eCM submission.
EDOCS-N-03088	10/29/03	Incorporate Software Loopback for eDOCSIS (SLED).

The following Engineering Change Notices have been incorporated in CM-SP-eDOCSIS I03-040804.

ECN	Date Ratified	Summary
eDOCSIS-N-04.0122-2	2/11/04	eDOCSIS SLED clarifications.
eDOCSIS-N-04.0123-2	2/11/04	eDOCSIS DHCP Option 43 clarifications.
eDOCSIS-N-04.0134-2	3/17/04	eDOCSIS DHCP Option 43 suboption 1 clarifications.
eDOCSIS-N-04.0138-1	4/7/04	Reserved DHCP 43 suboptions for eSAFEs.
eDOCSIS-N-04.0150-2	6/30/04	eDOCSIS clarification on the effect of NACO state on eSAFE traffic.
eDOCSIS-N-04.0151-3	6/30/04	Adding eSTB as a new eDOCSIS eSAFE.
eDOCSIS-N-04.0152-5	7/7/04	eSAFE MIB definition for eDOCSIS.
eDOCSIS-N-04.0153-2	6/30/04	eDOCSIS ifIndex filter rules for interfaces to eSAFEs.
eDOCSIS-N-04.0154-2	6/30/04	Modify eDOCSIS firmware download requirements to account for eSTB.
eDOCSIS-N-04.0158-2	7/7/04	Additional RFC 1493 requirements for eDOCSIS devices.
eDOCSIS-N-04.0160-1	7/28/04	Miscellaneous editorial and typo corrections.
eDOCSIS-N-04.0161-1	7/28/04	Remove reference to non-existent eSAFE MIB object.

The following Engineering Change Notices have been incorporated in CM-SP-eDOCSIS I04-041124.

ECN	Date Ratified	Summary
eDOCSIS-N-04.0169-1	9/1/04	Correct eCM DHCP sub-option 43.1 entry in Table 5-8.
eDOCSIS-N-04.0172-6	10/6/04	Enhancements to eDOCSIS MIB for device status, specifically related to safe reboots.
eDOCSIS-N-04.0176-2	10/27/04	Add dormant mode option to eSAFE MIB object esafePsCableHomeModeControl.
eDOCSIS-N-04.0182-2	10/27/04	Clarification of MaxCPE count for DSG tunnel MAC addresses.

The following Engineering Change Notices have been incorporated in CM-SP-eDOCSIS-I05-050408.

ECN	Date Ratified	Summary
eDOCSIS-N-05.0208-4	2/16/05	esafeProvisioningStatusProgress description change.
eDOCSIS-N-05.0210-2	2/23/05	Software download for CMs with eSTBs.
eDOCSIS-N-05.0212-2	3/9/05	Discrepancy between the CLAB-DEF-MIB and the SLED-MIB.

The following Engineering Change Notices have been incorporated in CM-SP-eDOCSIS-I06-050812.

ECN	Date Ratified	Summary
eDOCSIS-N-05.0223-3	7/14/05	SW Download for Non-OC eDOCSIS Device

The following Engineering Change Notices have been incorporated in CM-SP-eDOCSIS-I07-051209.

ECN	Date Ratified	Summary
eDOCSIS-N-05.0260-2	11/9/05	Update of eDOCSIS for changes in OC downloading.