Data-Over-Cable Service Interface Specifications

IPv4 and IPv6 eRouter Specification

CM-SP-eRouter-I06-110623

ISSUED

Notice

This DOCSIS® specification is the result of a cooperative effort undertaken at the direction of Cable Television Laboratories, Inc. for the benefit of the cable industry and its customers. This document may contain references to other documents not owned or controlled by CableLabs®. Use and understanding of this document may require access to such other documents. Designing, manufacturing, distributing, using, selling, or servicing products, or providing services, based on this document may require intellectual property licenses from third parties for technology referenced in this document.

Neither CableLabs nor any member company is responsible to any party for any liability of any nature whatsoever resulting from or arising out of use or reliance upon this document, or any document referenced herein. This document is furnished on an "AS IS" basis and neither CableLabs nor its members provides any representation or warranty, express or implied, regarding the accuracy, completeness, noninfringement, or fitness for a particular purpose of this document, or any document referenced herein.

Document Status Sheet

Document Control Number:	CM-SP-eRouter-I06-110623			
Document Title:	IPv4 and IPv6 eRouter Specification			
Revision History:	I01 – 12/7/06 I02 – 2/23/07 I03 – 5/18/07 I04 – 6/11/10 I05 – 2/10/11 I06 – 6/23/11			
Date:	June 23, 2011			
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

This page intentionally left blank.

1 SCOPE

This specification is part of the DOCSIS family of specifications developed by Cable Television Laboratories, Inc. (CableLabs). This specification was developed for the benefit of the cable industry, and includes contributions by operators and vendors from North America, Europe, and other regions.

1.1 Introduction and Purpose

This specification defines a core set of features that enable multiple subscriber devices to gain access to operator provided high-speed data service using DOCSIS. This core set of features allows for both IPv4- and IPv6-enabled devices to gain connectivity to the Internet.

The eRouter is specified as an Embedded Service/Application Functional Entity (eSAFE) device as defined in [eDOCSIS] that is implemented in conjunction with a DOCSIS Cable Modem device.

The core set of features defined in this specification includes the ability to provision multiple CPE devices, a description of how to forward data to and from CPE devices, and also the ability to forward IP Multicast traffic to CPE devices.

1.2 Requirements

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

"MUST"	This word 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 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 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 Normative References [1]

In order to claim compliance with this specification, it is necessary to conform to all or part of 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.

[CANN DHCP]	CableLabs DHCP Options Registry Specification, CL-SP-CANN-DHCP-Reg-I07-110623, June 23, 2011, Cable Television Laboratories, Inc.
[eDOCSIS]	eDOCSIS™ Specification, CM-SP-eDOCSIS-I22-110623, June 23, 2011, Cable Television Laboratories, Inc.
[MULPI]	DOCSIS MAC and Upper Layer Protocol Interface Specification, CM-SP-MULPIv3.0-I16-110623, June 23, 2011, Cable Television Laboratories, Inc.
[OSSIv3.0]	DOCSIS Operations Support System Interface Specification. CM-SP-OSSIv3.0-I15-110623, June 23, 2011, Cable Television Laboratories, Inc.
[SECv3.0]	DOCSIS Security Specification, CM-SP-SECv3.0-I13-100611, June 11, 2010, Cable Television Laboratories, Inc.
[RFC 792]	IETF RFC 792, Internet Control Message Protocol, J. Postel, September 1981.
[RFC 826]	IETF RFC 826, An Ethernet Address Resolution Protocol, David C. Plummer, November, 1982.
[RFC 868]	IETF RFC 868, Time Protocol, J. Postel & K. Harrenstien, May 1983.
[RFC 1122]	IETF RFC 1122, Requirements for Internet Hosts - Communication Layers, R. Braden, October, 1989.
[RFC 1812]	IETF RFC 1812, Requirements for IP Version 4 Routers, F. Baker, June 1995.
[RFC 1918]	IETF RFC 1918, Address Allocation for Private Internets, Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, E. Lear, February 1996.
[RFC 2131]	IETF RFC 2131, Dynamic Host Configuration Protocol, R. Droms, March, 1997.
[RFC 2132]	IETF RFC 2132, DHCP Options and BOOTP Vendor Extensions, S. Alexander, R. Droms, March 1997.
[RFC 2461]	IETF RFC 2461, Neighbor Discovery for IP Version 6 (IPv6), T. Narten, E. Nordmark, W. Simpson, December, 1998.
[RFC 2462]	IETF RFC 2462, IPv6 Stateless Address Autoconfiguration, S. Thomson, T. Narten, December 1998.
[RFC 2463]	IETF RFC 2463, Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification, A. Conta, S. Deering, December 1998.
[RFC 2710]	IETF RFC 2710, Multicast Listener Discovery (MLD) for IPv6, S. Deering, W. Fenner, B. Haberman, October 1999.
[RFC 3022]	IETF RFC 3022, Traditional IP Network Address Translator (Traditional NAT), P. Srisuresh, K. Egevang, January 2001.

[1] Revised per eRouter-N-09.0877-2 on 5/11/10 by JB.

[RFC 3203] IETF RFC 3203, DHCP reconfigure extension, Y. T'Joens, C. Hublet, P. De Schrijver, December, 2001.

[RFC 3315] IETF RFC 3315, Dynamic Host Configuration Protocol for IPv6 (DHCPv6), R. Droms, Ed., J. Bound, B. Volz, T. Lemon, C. Perkins, M. Carney, July 2003.

[RFC 3319] IETF RFC 3319, Dynamic Host Configuration Protocol (DHCPv6) Options for Session Initiation Protocol (SIP) Servers, H. Schulzrinne, B. Volz, July 2003.

[RFC 3376] IETF RFC 3376, Internet Group Management Protocol, Version 3, B. Cain, S. Deering, I. Kouvelas, B. Fenner, A. Thyagarajan, October, 2002.

[RFC 3412] IETF RFC 3412, Configuring Networks and Devices with Simple Network Management Protocol (SNMP), M. MacFaden, D. Partain, J. Saperia, W. Tackabury, April 2003.

[RFC 3413] IETF RFC 3413, Internet Protocol Version 6 (IPv6) Addressing Architecture, R. Hinden, S. Deering, April 2003.

[RFC 3415] IETF RFC 3415, View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP), B. Wijnen, R. Presuhn, K. McCloghrie, December 2002.

[RFC 3417] IETF RFC 3417, A Conservative Selective Acknowledgment (SACK)-based Loss Recovery Algorithm for TCP, E. Blanton, M. Allman, K. Fall, L. Wang, April 2003.

[RFC 3419] IETF RFC 3419, Mobile IP Traversal of Network Address Translation (NAT) Devices, H. Levkowetz, S. Vaarala, May 2003.

[RFC 3489] IETF RFC 3489, STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs), J. Rosenberg, J. Weinberger, C. Huitema, R. Mahy, March 2003.

[RFC 3513] IETF RFC 3513, Internet Protocol Version 6 (IPv6) Addressing Architecture, R. Hinden, S. Deering, April, 2003.

[RFC 3584] IETF RFC 3584, Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework, R. Frye, D. Levi, S. Routhier, B. Wijnen, August 2003.

[RFC 3633] IETF RFC 3633, IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6, O. Troan, R. Droms, December 2003.

[RFC 3646] IETF RFC 3646, DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6), R. Droms, December 2003.

[RFC 3736] IETF RFC 3736, Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6, R. Droms, April 2004.

[RFC 3810] IETF RFC 3810, Multicast Listener Discovery Version 2 (MLDv2) for IPv6, R. Vida, Ed., L. Costa, Ed., June 2004.

[RFC 4075] IETF RFC 4075, Simple Network Time Protocol (SNTP) Configuration Option for DHCPv6, V. Kalusivalingam, Cisco Systems, May 2005.

[RFC 4242] IETF RFC 4242, Information Refresh Time Option for Dynamic Host Configuration Protocol for IPv6 (DHCPv6), S. Venaas, T. Chown, B. Volz, November 2005.

[RFC 4293] IETF RFC 4293, Management Information Base for the Internet Protocol (IP), S. Routhier, (Editor), Bill Fenner, Brian Haberman, Dave Thaler, April 2006.

[RFC 4361] IETF RFC 4361, Node-specific Client Identifiers for Dynamic Host Configuration Protocol Version Four (DHCPv4), T. Lemon, B. Sommerfeld, February 2006.

[RFC 4861] IETF RFC 4861, T. Narten, E. Nordmark, W. Simpson, H. Soliman, Neighbor Discovery for IP Version 6 (IPv6), September 2007.

[RFC 4862]	IETF RFC 4862, S. Thomson, T. Narten, T. Jinmei, IPv6 Stateless Address Autoconfiguration, September 2007.
[RFC 5006]	IETF RFC 5006, J. Jeong, S. Park, L. Beloeil, S. Madanapalli, IPv6 Router Advertisement Option for DNS Configuration, September 2007.
[I-D.DS-Lite]	IETF Internet-Draft draft-ietf-softwire-dual-stack-lite-02, A. Durand, Dual-stack lite broadband deployments post IPv4 exhaustion, October 2009.
[I-D.CPE-Router]	IETF Internet-Draft draft-ietf-v6ops-ipv6-cpe-router-03, H. Singh, W. Beebee, C. Donley, B. Stark, O. Troan, December 2009.

2.2 Informative References

This specification does not use any informative references.

2.3 Reference Acquisition

Cable Television Laboratories, Inc., 858 Coal Creek Circle, Louisville, CO 80027;
Phone 303-661-9100; Fax 303-661-9199; Internet: http://www.cablelabs.com

Internet Engineering Task Force (IETF), Internet: http://www.ietf.org/
Note: Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time.
The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.
Internet-Drafts may also be accessed at http://tools.ietf.org/html/

Internet Engineering Task Force (IETF) Secretariat, 48377 Fremont Blvd., Suite 117, Fremont, California 94538, USA, Phone: +1-510-492-4080, Fax: +1-510-492-4001. Internet: http://www/ietf.org

3 TERMS AND DEFINITIONS [2]

This specification uses the following terms:

Customer-Facing Interface	An eRouter interface used for connecting CPE devices. Defined in [I-D.CPE-Router] as a Local Area Network (LAN) Interface.
Customer-Facing IP Interface	An IP Interface connected to the eRouter that is not necessarily mapped one-to-one with the number of Customer-Facing ports on the eRouter. As defined in [I-D.CPE-Router], this is a LAN interface.
Customer-Facing Logical IP Interface	A logical Interface connected to the eRouter that is not necessarily mapped one-to-one with the number of Customer-Facing ports on the eRouter. As defined in [I-D.CPE-Router], this is a LAN interface.
eRouter	An eSAFE device that is implemented in conjunction with the DOCSIS Embedded Cable Modem.
Multicast Subscription Database	A simple table of entries for the IPv4 or IPv6 Multicast Group Membership information maintained by the eRouter on respective interfaces. Implementation details for storage of records are completely vendor-defined.
Operator-Facing Interface	The eRouter interface that is connected to the Embedded Cable Modem. As defined in [I-D.CPE-Router], this is a Wide Area Network (WAN) interface.
Operator-Facing IP Interface	IP Interface that is connected to the Embedded Cable Modem and is provisioning with an IP Address provided by the Operator. As defined in [I-D.CPE-Router], this is a WAN interface.

[2] Revised per eRouter-N-09.0877-2 on 5/11/10 by JB.

4 ABBREVIATIONS AND ACRONYMS

This specification uses the following abbreviations:

ALG	Application Layer Gateway
ARP	Address Resolution Protocol
ASN.1	Abstract Syntax Notation One
CPE	Customer Premises Equipment
CM	Cable Modem
DAD	Duplicate Address Detection
DNS	Domain Name Service
DUID	DHCP unique identifier
EAE	Early Authentication and Encryption
eSAFE	Embedded Service/Application Functional Entity
EUI	Extended Unique Identifier
FTP	File Transfer Protocol
ID	Identifier
IP	Internet Protocol
IRT	Initial Retransmission Times
LAN	Local Area Network
ND	Neighbor Discovery
MAC	Media Access Control
MIB	Management Information Base
MLD	Multicast Listener Discovery
MRC	Maximum Retransmission Count
MRD	Maximum Retransmission Duration
MRT	Maximum Retransmission Time
NAT	Network Address Translation
NAPT	Network Address Port Translation
OID	Object ID
OUI	Organization Unique Identifier
RFC	Request For Comment
RA	Router Advertisement
RD	Router Discovery
SIP	Session Initiation Protocol
TCP	Transmission Control Protocol
TLV	Type/Length/Value

TTL	Time To Live
UDP	User Datagram Protocol
WAN	Wide Area Network

5 THEORY OF OPERATION [3]

The eRouter device is intended to provide networking functionality in conjunction with an embedded DOCSIS eCM in an eDOCSIS device.

This specification defines a set of features for an IPv4 eRouter and a set of features for an IPv6 eRouter. Both sets of features can be implemented together as an IPv4 + IPv6 eRouter.

The figures below depict implementations of an eDOCSIS device with an IPv4 eRouter, an IPv6 eRouter, and an IPv4 + IPv6 eRouter.

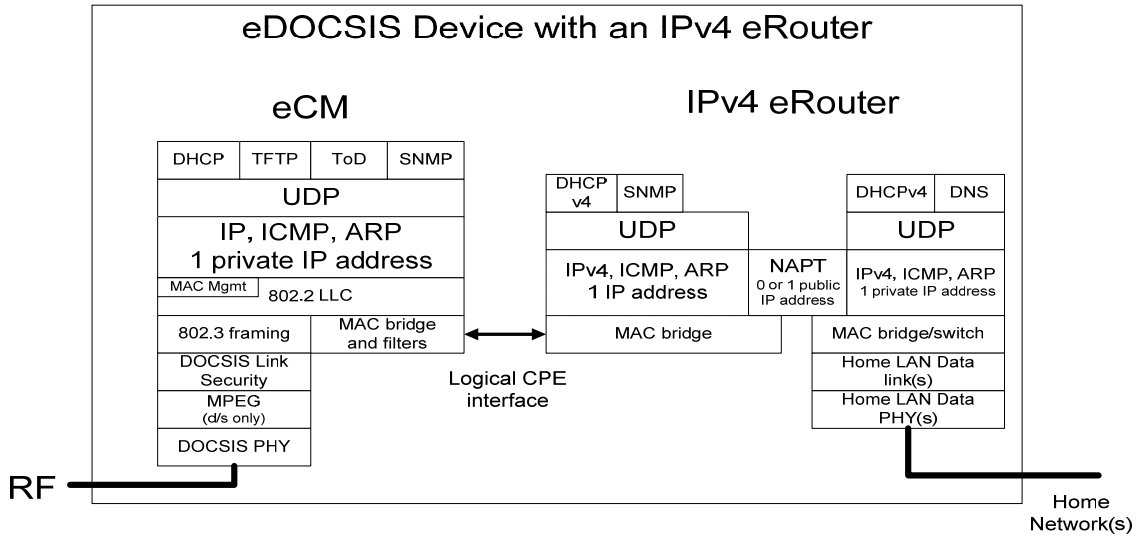

Figure 5-1 - Logical Components of an eDOCSIS device with an IPv4 eRouter

[3] Revised per eRouter-N-09.0877-2 on 5/11/10 by JB.

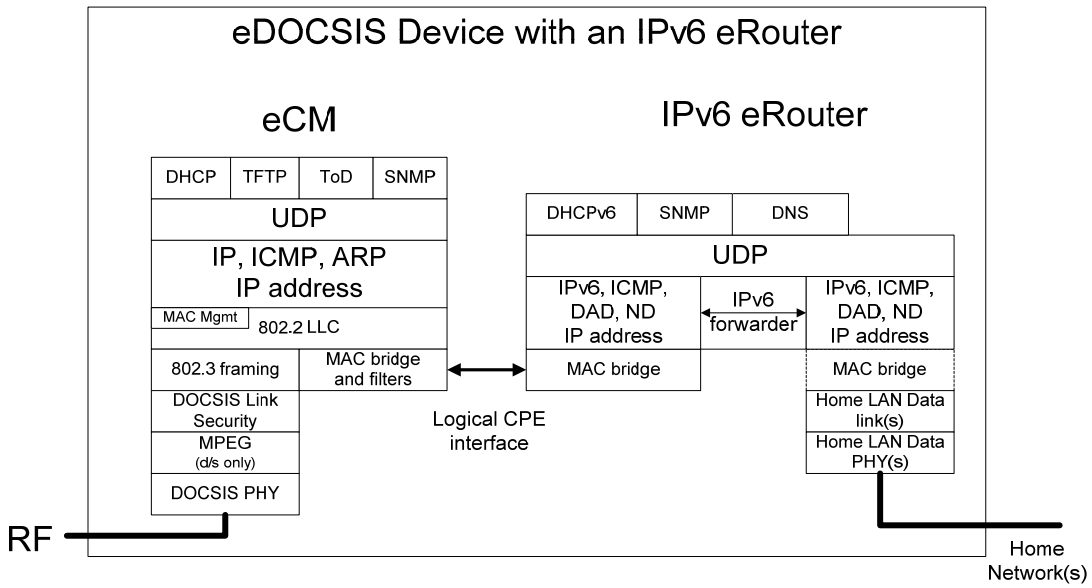

Figure 5-2 - Logical Components of an eDOCSIS device with an IPv6 eRouter

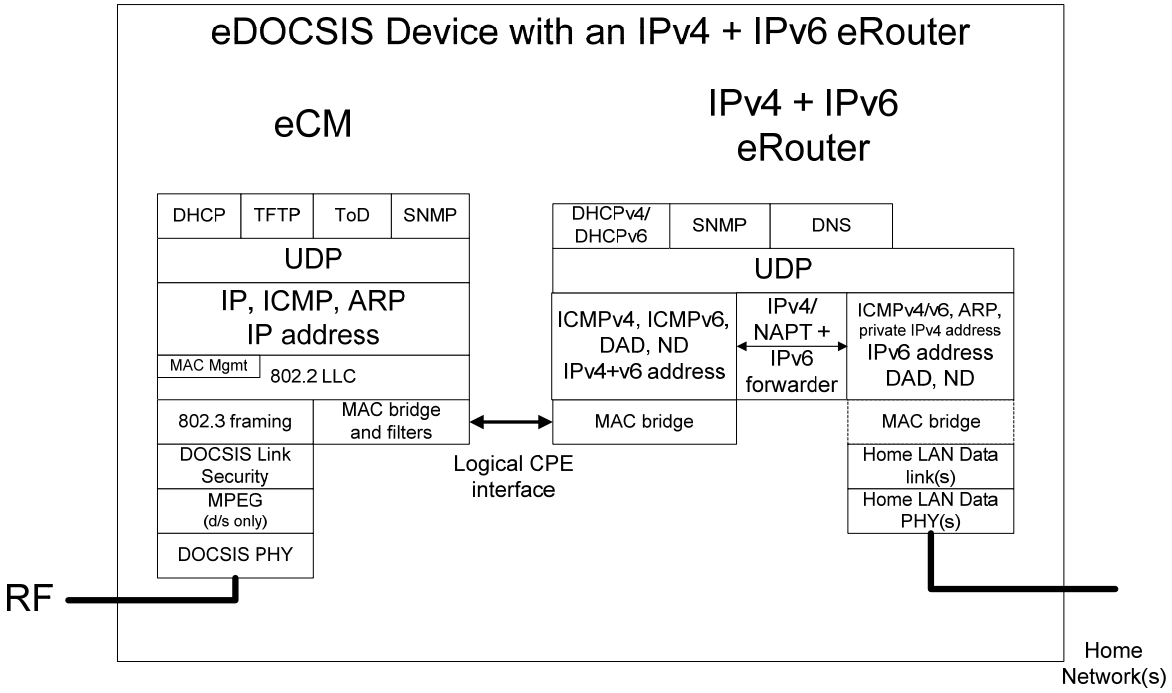

Figure 5-3 - Logical Components of an eDOCSIS device with an IPv4 + IPv6 eRouter

The primary function of the eRouter device is to allow subscribers to connect multiple CPE devices to the operator-provided DOCSIS high-speed Internet service. DOCSIS specifications allow subscribers to directly connect multiple CPE devices to the Cable Modem; however, that requires operators to provide provisioning to each of the

CPE devices. The eRouter is delegated the responsibility of provisioning multiple CPE devices at the subscriber end. Depending on which version of the eRouter is included with the eDOCSIS implementation, the eRouter allows provisioning of IPv4 CPEs, IPv6 CPEs, or both IPv4 and IPv6 CPEs simultaneously.

This specification defines the core set of functions that are performed by the eRouter; however, in most implementations, vendors would include additional features that would enhance the eRouter device. This may include additional networking features as well as the ability to provision and manage the eRouter from the subscriber side.

The specification defines: a) CPE provisioning with IPv4 and IPv6 addresses, b) IPv4 data forwarding with NAPT and IPv6 data forwarding, c) ability to forward IP Multicast traffic, and d) preserving IP QoS markings on IP data to and from the CPE devices.

This specification uses the terms Customer-Facing Interface and Operator-Facing Interface as defined in Section 3.

This specification defines requirements for an eRouter device with a single Operator-Facing IP Interface. This specification defines requirements for an eRouter device with a single Customer-Facing logical IP Interface that is not necessarily mapped one-to-one with the number of Customer-Facing ports on the eRouter. Though it is possible that an eRouter could have multiple Customer-Facing IP interfaces, it is vendor-specific how to forward traffic between Customer-Facing Interfaces and the Operator-Facing IP Interface when there are multiple Customer-Facing IP interfaces.

6 eROUTER INITIALIZATION [4]

Prior to its initialization, the eRouter is enabled or disabled through the TLVs in Annex B. The eRouter may operate in any one of four possible modes – Disabled, IPv4 Protocol Enabled, IPv6 Protocol-Enabled, or Dual IP Protocol-Enabled, as summarized in Table 6-1. The eRouter MUST support the Disabled Mode and at least one other Mode. The eRouter MAY support the IPv4 Protocol Enabled Mode. The eRouter MAY support the IPv6 Protocol Enabled Mode. The eRouter MAY support the Dual IP Protocol Enabled Mode. The eRouter MUST persist its initialization mode across reinitialization. The eRouter MAY reinitialize itself when its initialization mode changes.

Table 6-1 - eRouter Modes

Mode	IPv4	IPv6
Disabled	CM bridges all traffic per [MULPI] spec.	CM bridges all traffic per [MULPI] spec.
IPv4 Protocol Enabled	eRouter forwards IPv4 traffic with NAPT.	eRouter does not forward IPv6 traffic.
IPv6 Protocol Enabled	eRouter does not forward IPv4 traffic.	eRouter forwards IPv6 traffic.
Dual IP Protocol Enabled	eRouter forwards IPv4 packets using NAPT.	eRouter forwards IPv6 packets.

In Disabled Mode, the eRouter transparently forwards all traffic directly between its Customer-Facing Interface and its Operator-Facing Interface. In this mode, it appears as if there is no eRouter present. The CM bridges all traffic (regardless of IP protocol version) to the CPE ports that would have been behind the eRouter had it been enabled. In this mode, the eRouter specification is irrelevant – the interfaces become part of the cable modem. All behavior will occur according to the DOCSIS specifications.

When configured to operate in IPv4 Protocol Enabled Mode, the eRouter performs IPv4 provisioning and forwards IPv4 traffic between its Customer-Facing Interface and its Operator-Facing Interface. The eRouter operating in IPv4 Protocol Enabled Mode will not perform any IPv6 provisioning. An eRouter configured to operate in IPv4 Protocol Enabled Mode will not forward any IPv6 traffic. The IPv4 traffic is forwarded using NAPT according to Section 9.

When configured to operate in IPv6 Protocol Enabled Mode, the eRouter performs IPv6 provisioning and forwards IPv6 traffic between its Customer-Facing Interface and its Operator-Facing Interface. The eRouter operating in IPv6 Protocol Enabled Mode will not perform any IPv4 provisioning. An eRouter configured to operate in IPv6 Protocol Enabled Mode will not forward any IPv4 traffic.

In Dual IP Protocol Enabled Mode, the eRouter is configured to forward both IPv4 and IPv6 traffic between its Customer-Facing Interface and its Operator-Facing Interface. IPv4 traffic is forwarded using NAPT, according to Section 9. IPv6 traffic is forwarded according to Section 9.7.

[4] Revised per eRouter-N-09.0877-2 on 5/11/10 by JB.

7 IPV4 PROVISIONING

The normative requirements of this section are mandatory for an eRouter that implements the IPv4 Protocol Enabled Mode and/or the Dual IP Protocol Enabled Mode as defined in Section 6.

After the CM has completed provisioning, if the eRouter is configured to route IPv4 packets, the eRouter MUST use DHCPv4 [RFC 2131] via its Operator-Facing Interface in order to obtain an IP address and any other parameters needed to establish IP connectivity, as illustrated in Figure 7-1

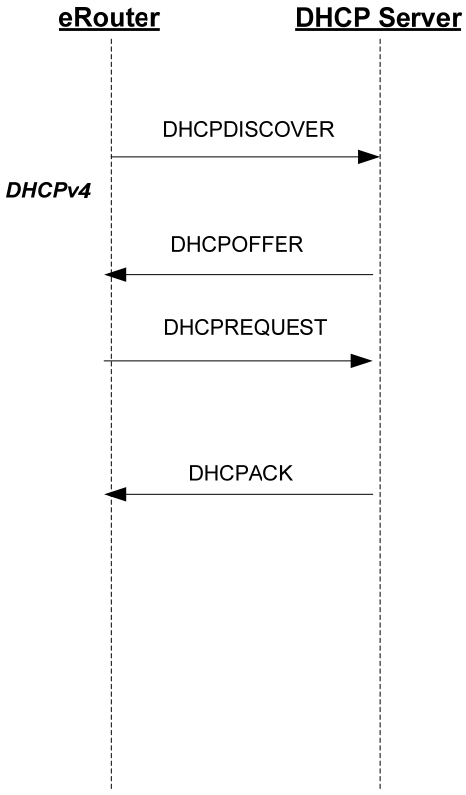

Figure 7-1 - IPv4 Provisioning Message Flow

The eRouter may receive multiple DHCPOFFER messages in response to its DHCPDISCOVER message. If a received DHCPOFFER message does not include all of the required DHCPv4 fields and options as described in Section 7.1, the eRouter MUST discard the DHCPOFFER message and wait for another DHCPOFFER message. If none of the received DHCPOFFER messages contain all the required DHCPv4 fields and options, the eRouter MUST retransmit the DHCPDISCOVER message.

The backoff values for retransmission of DHCPDISCOVER messages SHOULD be chosen according to a uniform distribution between the minimum and maximum values in the rows of Table 7-1.

Table 7-1 - eRouter DHCP Retransmission Interval

Backoff Number	Minimum (sec.)	Maximum (sec.)
1	3	5
2	7	9
3	15	17
4	31	33
5	63	65

The eRouter SHOULD also implement a different retransmission strategy for the RENEWING and REBINDING states, as recommended in [RFC 2131], which is based on one-half of the remaining lease time.

The eRouter MUST limit the number of retransmissions of the DHCPDISCOVER and DHCPREQUEST messages to five or fewer. The eRouter MUST NOT forward IPv4 traffic between its Customer-Facing Interface and its Operator-Facing Interface until it has completed IPv4 provisioning, including the successful receipt of a DHCPACK message. The eRouter MUST NOT forward IPv4 traffic if, at any time, it does not have an IPv4 address for its Operator-Facing Interface.

The eRouter MUST be able to accept a unicast response from the DHCP server/relay agent.

[RFC 3203] describes an extension to DHCPv4 that allows a server to send a FORCERENEW message that forces a client to renew its lease. The eRouter MUST ignore all received FORCERENEW messages.

7.1 DHCPv4 Fields Used by the eRouter

The eRouter MUST include the following fields in the DHCPDISCOVER and DHCPREQUEST messages:

- The hardware type (htype) MUST be set to 1.
- The hardware length (hlen) MUST be set to 6.
- The client hardware address (chaddr) MUST be set to the 48-bit MAC address associated with the IPv4 CM-facing interface of the eRouter.
- The Broadcast bit MUST NOT be set.
- The client-identifier option MUST be included, using the format defined in [RFC 4361].
- The parameter request list option MUST be included. The following option codes (defined in [RFC 2132] and [RFC 4361]) MUST be included in the list:
    - Option code 1 (Subnet Mask)
    - Option code 3 (Router Option)
    - Option code 6 (DNS Server Option)
    - Option code 60 (Vendor Class Identifier) [eRouter1.0]
    - Option Code 43 (see [eDOCSIS])
    - Option 55 Parameter Request List

The following fields are expected in the DHCPOFFER and DHCPACK messages returned to the eRouter. The eRouter MUST configure itself with the listed fields from the DHCPACK:

- The IP address to be used by the eRouter (yiaddr) (critical).

- The IP Address lease time, option 51 (critical).
- The Server identifier, option 54 (critical).
- The subnet mask to be used by the eRouter (Subnet Mask, option 1) (critical).

A list of addresses of one or more routers is to be used for forwarding eRouter-originated IP traffic (Router Option, option 3). The eRouter is not required to use more than one router IP address for forwarding (critical):

- A list of DNS Server addresses (critical).
- A list of options under the CL_V4EROUTER_CONTAINER_OPTION option which are passed on to CPE devices as defined in the [CANN DHCP].

If a critical field is missing or invalid in the DHCPACK received during initialization, the eRouter MUST restart the DHCP cycle, beginning with an initial DHCPDISCOVER.

If a non-critical field is missing or invalid in the DHCPACK received during initialization, the eRouter MUST ignore the field, and continue the provisioning process.

If the yiaddr, Server Address, or Lease Time field is missing or invalid in the DHCPACK received during a renew or rebind operation, the eRouter MUST retry the renew or rebind operation until either: (1) it receives a response containing valid values of the yiaddr, Server Address, and Lease Time fields; or (2) the lease expires. If the lease expires, the eRouter MUST restart the DHCP cycle, beginning with an initial DHCPDISCOVER.

If any field other than the yiaddr, Server Address or Lease Time is missing, or is invalid in the DHCPACK received during a renew or rebind operation, the eRouter MUST ignore the field if it is invalid and remain operational.

7.2 Router DHCPv4 Server Sub-element

The DHCP server is responsible for assigning network address leases to LAN IP devices associated with Customer-Facing Interfaces. It is also responsible for providing LAN IP devices with configuration information via DHCP Option codes, as specified in [RFC 2132].

7.2.1 DHCPv4 Server Function Goals

Goals for the DHCP server include the following:

- Assign network address leases to CPE devices according to [RFC 2131].
- Assign private CPE addresses according to [RFC 1918].
- Assign configuration information according to [RFC 2132].

7.2.2 DHCPv4 Server Function System Description

The eRouter DHCPv4 server responsibilities include:

- Assigning IP Addresses and delivering DHCP configuration parameters to CPE Devices. The server relies on built-in default values for initial IP Address pool configuration, lease parameter configuration, and DHCP options values.
- Optional logging of DHCPv4 server errors to a local event log.

7.2.3 DHCPv4 Server Function Requirements

The eRouter MUST include a DHCPv4 server compliant with [RFC 2131].

In addition, the following requirements apply to the DHCPv4 Server function:

- When the DHCP server assigns an active lease for an IP address to a CPE Device, the server MUST remove that IP address from the pool of IP addresses available for assignment.
- The DHCP server function of the eRouter MUST support the DHCP options indicated as mandatory in Table 7-2.
- The DHCP server function of the eRouter MUST NOT respond to DHCP messages that are received through the Operator-Facing Interface, nor originate DHCP messages from the Operator-Facing Interface.
- The DHCP server function of the eRouter MUST NOT deliver any DHCP option with null value to any CPE device.
- The DHCP server function SHOULD be operational independent of the eRouter Operator-Facing Interface connectivity state.
- If the eRouter Operator-Facing Interface is not successfully provisioned, the eRouter DHCP server function SHOULD assign a short lease time to CPE devices and may omit options it has not acquired.
- The DHCP server function MUST assign private IP address space as defined in [RFC 1918].
- The DHCP server function SHOULD log errors to a local event log.

The Internet Assigned Numbers Authority (IANA) has reserved the following three blocks of the IP address space for private internets:

```
10.0.0.0        -   10.255.255.255   (10/8 prefix)
172.16.0.0      -   172.31.255.255   (172.16/12 prefix)
192.168.0.0     -   192.168.255.255  (192.168/16 prefix)
```

Table 7-2 - DHCPv4 Server Options

Option Number	Option Function
0	Pad
255	End
1	Subnet Mask
3	Router Option
6	Domain Name Server
50	Requested IP Address
51	IP Address Lease Time
54	Server Identifier
55	Parameter Request List
X	Option(s) acquired under CL_V4EROUTER_CONTAINER_OPTION from the Operator

8 IPV6 PROVISIONING [5]

IPv4 address space is nearly exhausted. The IANA pool of free IPv4 address space is expected to be completely depleted before customers have been fully migrated to IPv6. The features necessary to facilitate transition to IPv6 are described in the following sections.

The normative requirements of this section are mandatory for an eRouter that implements the IPv6 Protocol Enabled Mode and/or the Dual IP Protocol Enabled Mode, as defined in Section 6.

After the CM has completed provisioning, if the eRouter is operating in the IPv6 Protocol Enabled Mode or the Dual IP Protocol Enabled Mode, the eRouter MUST use DHCPv6 [RFC 3315] in order to obtain an IP address for its Operator-Facing IP Interface and any other parameters needed to establish IP connectivity, as illustrated in Figure 8-1.

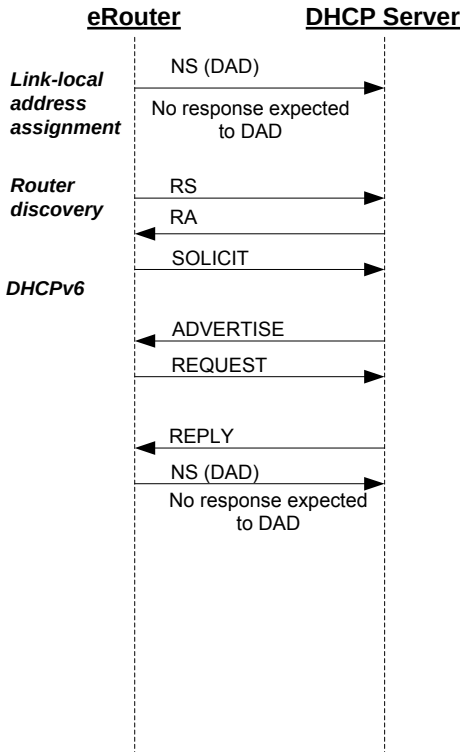

Figure 8-1 - IPv6 Provisioning Message Flow

The eRouter establishes IPv6 connectivity including assignment of:

- Link-local IPv6 address
- IPv6 address of a Default router
- Operator-Facing Interface IPv6 address (used for both management access to the eRouter and data forwarding)
- Other IPv6 configuration

These steps are described in the following sub-sections.

[5] Revised per eRouter-N-09.0877-2 on 5/11/10 by JB.

8.1 Obtain Link-Local Address [6]

The eRouter MUST construct a link-local address for its Operator-Facing Interface and each of its Customer-Facing Interface(s) according to the procedure in section 5.3 of [RFC 4862]. The eRouter MUST use the EUI-64 identifier as a link-local address for each of its interfaces as described in [RFC 3513]. For each of its interfaces, the eRouter MUST join the all-nodes multicast address and the solicited-node multicast address of the corresponding link-local address [RFC 4862], [RFC 2710]. The eRouter MUST use Duplicate Address Detection (DAD), as described in section 5.4 of [RFC 4862], to confirm that the constructed link-local addresses are not already in use. If the eRouter determines that the constructed link-local address is already in use, the eRouter MUST terminate IPv6 operation on that interface.

8.2 Perform router discovery [7]

The eRouter MUST perform router discovery as specified in section 6.3 of [RFC 4861] on its Operator-Facing Interface. The eRouter identifies neighboring routers and default routers from the received RAs.

8.3 Obtain IPv6 address and other configuration parameters [8]

An eRouter MUST examine the contents of RAs it receives and obey the following rules:

- If the M bit in the RA is set to 1, the eRouter MUST use DHCPv6 to obtain its IPv6 address for its Operator-Facing Interface and other configuration information (and ignore the 0 bit).
- If there are no prefix information options in the Router Advertisement, the eRouter MUST NOT perform SLAAC.
- If the RA contains a prefix advertisement with the A bit set to 0, the eRouter MUST NOT perform SLAAC on that prefix.

If an RA contains a prefix advertisement for an IPv6 network prefix on which the eRouter does not have an address and the M bit in the RA is set to 1, the eRouter MUST use DHCPv6 to obtain its IPv6 address for its Operator-Facing Interface and renew any current IA_PD lease(s).

The eRouter sends a DHCPv6 Solicit message as described in section 17.1.1 of [RFC 3315]. The Solicit message MUST include:

1. A Client Identifier option containing the DUID for this eRouter (as specified by [RFC 3315]).
2. An IA_NA option to obtain its IPv6 address.
3. An IA_PD option to obtain its delegated IPv6 prefix.
4. A Reconfigure Accept option to indicate the eRouter is willing to accept Reconfigure messages.
5. An Options Request option, which MUST include the DNS Recursive Name Server option [RFC 3646].
6. A Vendor Class option containing 32-bit number 4491 (the Cable Television Laboratories, Inc., enterprise number) and the string "eRouter1.0".
7. A Rapid Commit option indicating that the eRouter is willing to perform a 2-message DHCPv6 message exchange with the server. The Rapid Commit option is covered in more detail later in this section.
8. A DOCSIS Device Identifier Option, as defined in [CANN DHCP].

[6] Revised per eRouter-N-09.0877-2 on 5/11/10 by JB.
[7] Revised per eRouter-N-09.0877-2 on 5/11/10 by JB.
[8] Revised per eRouter-N-09.0877-2 on 5/11/10 and per eRouter-N-10.0974-2 on 1/24/11 by JB.

9. A Vendor-specific option, containing:
    a. The 32-bit number 4491 (the Cable Television Laboratories, Inc., enterprise number).
    b. A CableLabs Vendor Specific Option Request sub-option, which request CL_OPTION_ORO CableLabs Vendor-specific options as defined in [CANN DHCP].
    c. A list of options under the CL_EROUTER_CONTAINER_OPTION option which are passed on to CPE devices as defined in the [CANN DHCP].

The eRouter MUST use the following values for retransmission of the Solicit message (see section 14 of [RFC 3315] for details):

- IRT (Initial Retransmission Time) = SOL_TIMEOUT
- MRT (Maximum Retransmission Time) = SOL_MAX_TIMEOUT
- MRC (Maximum Retransmission Count) = 0
- MRD (Maximum Retransmission Duration) = 0

The DHCPv6 server may be configured to use a 2-message Rapid Commit sequence. The DHCP server and eRouter follow [RFC 3315] in the optional use of the Rapid Commit message exchange, as shown in Figure 8-2. In the Rapid Commit exchange between the DHCPv6 server and client, only the DHCPv6 "SOLICIT" and the DHCPv6 "REPLY" messages are used. The DHCPv6 “REPLY” message contains the address as well as any configuration parameters.

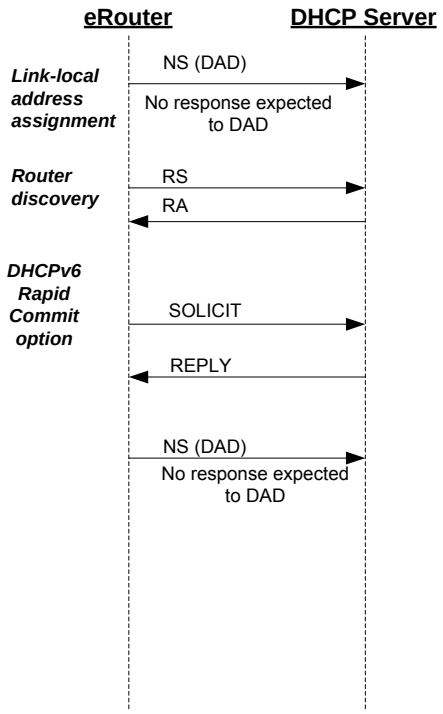

Figure 8-2 - IPv6 Provisioning Rapid Commit Message Flow

The DHCP server responds to Solicit messages and Request messages with Advertise and Reply messages (depending on the use of Rapid Commit option). The Advertise and Reply messages may include other configuration parameters, as requested by the eRouter, or as configured by the administrator, to be sent to the eRouter. If any of the following options is absent from the Advertise message, the eRouter MUST discard the message and wait for another Advertise message. If any of the following options is absent from the Reply message, the eRouter MUST consider IPv6 provisioning to have failed. In addition the eRouter MAY log an event.

1. The IA_NA option containing the eRouter's IPv6 address;
2. The IA_PD option containing the delegated IPv6 prefix for use by the eRouter;
3. Reconfigure Accept option.

The eRouter interface MUST join the all-nodes multicast address and the solicited-node multicast address of the IPv6 address acquired through DHCPv6. The eRouter MUST perform Duplicate Address Detection (DAD) with the IPv6 address acquired through DHCPv6.

If the eRouter determines through DAD that the IPv6 address assigned through DHCPv6 is already in use by another device, the eRouter MUST:

- Send a DHCP Decline message to the DHCP server, indicating that it has detected that a duplicate IP address exists on the link.
- Discontinue using the duplicate IP address.
- Consider the IPv6 provisioning process to have failed, and restart the IPv6 provisioning process.

If the eRouter determines through DAD that the IPv6 address that has been assigned through DHCPv6 is already in use by another device, the eRouter MAY log an event.

The eRouter MUST support the Reconfigure Key Authentication Protocol, as described in section 21.5 of [RFC 3315].

The eRouter MUST NOT forward any IPv6 traffic between its Customer-Facing Interface and its Operator-Facing Interface until it has successfully completed the IPv6 provisioning process. The eRouter MUST NOT forward any IPv6 traffic between its Customer-Facing Interface and its Operator-Facing Interface if, at any time, it does not have a Globally-assigned IPv6 address on its Operator-Facing Interface.

8.4 Use of T1 and T2 Timers [9]

The eRouter MUST initiate the lease renewal process when timer eRouter-T1 expires. The eRouter MUST initiate the lease rebinding process when timer eRouter-T2 expires. Timers eRouter-T1 and eRouter-T2 are called T1 and T2, respectively, in the DHCP specifications. If the DHCP server sends a value for eRouter-T1 to the eRouter in a DHCP message option, the eRouter MUST use that value. If the DHCP server does not send a value for eRouter-T1, the CM MUST set eRouter-T1 to 0.5 times the duration of the lease [RFC 3315]. If the DHCP server sends a value for eRouter-T2 to the eRouter in DHCP message options, the eRouter MUST use that value. If the DHCP server does not send a value for eRouter-T2, the eRouter MUST set eRouter-T2 to 0.875 times the duration of the lease [RFC 3315].

8.5 IPv6 Provisioning of CPE Devices [10]

The eRouter MUST assign a global IPv6 address to each Customer-Facing Interface based on a prefix derived from the prefix acquired under IA_PD in Section 8.3.

The eRouter SHOULD assign the EUI-64 IPv6 Interface Identifier to each Customer-Facing Interface based on the prefix acquired under IA_PD in Section 8.3. The EUI-64 IPv6 Interface Identifier is created by converting the IEEE 802 address assigned to each Customer-Facing Interface to an EUI-64 64-bit address, and complementing the U/L

[9] Revised per eRouter-N-09.0877-2 on 5/11/10 by JB.
[10] Revised per eRouter-N-09.0877-2 on 5/11/10 by JB.

bit; then prepending 64 bits of the prefix acquired under IA_PD in Section 8.3 to create the 128-bit IPv6 Interface Identifier address.[11]

The eRouter MUST generate Router Advertisements (RA) on its Customer-Facing Interfaces as per [RFC 4862]. Unless the eRouter is otherwise configured by an administrator, the RA MUST:

- have the M bit set to 0.
- have the O bit set to 1.
- contain prefix information with both the ICMPv6 options 'flags' On-link bit and A bit set to 1 derived from the prefix acquired under IA_PD in Section 8.3.[12]

These settings in the RA will cause CPE devices to use auto-configuration by default for assigning their global IPv6 address.

On the Customer-Facing Interface the eRouter MUST be able to pass the following set of DHCPv6 options received from the Cable Operator for the configuration of CPE devices.

- The OPTION_DNS_SERVERS option as specified in [RFC 3646]. This option carries a list of Domain Name Servers for the CPE devices.
- The list of options under the CL_V4EROUTER_CONTAINER_OPTION option which are passed to the eRouter by the operator.

The eRouter MAY include the Recursive DNS Server (RDNSS) option in its RA message as specified in [RFC 5006]. If the eRouter supports the RDNSS option, it MUST include the list of DNS servers included in the OPTION_DNS_SERVERS option.

8.6 DHCPv6 requirements for eRouter

The DHCPv6 requirements of this section are optional. The eRouter SHOULD provide DHCPv6 on CPE-facing interfaces, as described in the following RFCs:

- [RFC 3315] Dynamic Host Configuration Protocol for IPv6 (DHCPv6).
- [RFC 3319] Dynamic Host Configuration Protocol (DHCPv6) Options for Session Initiation Protocol (SIP) Servers.
- [RFC 3646] DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6).
- [RFC 3633] IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6.
- [RFC 3736] Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6.
- [RFC 4075] Simple Network Time Protocol (SNTP) Configuration Option for DHCPv6.
- [RFC 4242] Information Refresh Time Option for Dynamic Host Configuration Protocol for IPv6 (DHCPv6).

The DHCPv6 server SHOULD be able to manage at least one IA_NA for each client, and at least one address in each IA_NA. The DHCPv6 server SHOULD be able to assign an address to a client constructed from the client's EUI-64 identifier.

The DHCPv6 server SHOULD be able to manage at least one IA_PD for each client, and at least one delegated prefix in each IA_PD. The prefix delegated to the client is derived from the prefix delegated to the eRouter from the Cable Operator.

[11] Paragraph added per eRouter-N-07.0396-1 by kb 5/11/07.
[12] Paragraph modified per eRouter-N-07.0396-1 by kb 5/11/07.

The DHCPv6 server MAY implement vendor-defined default lifetimes for assigned addresses and delegated prefixes.

The eRouter SHOULD be able to pass a set of options received from the Cable Operator to the DHCPv6 server for configuration of CPEs.

The eRouter MAY relax the requirements on non-volatile storage of assigned addresses and delegated prefixes and MAY glean information about assigned addresses and delegated prefixes from Advertise, Renew, and Rebind messages received from clients.

The eRouter is not expected to implement:

- Rapid commit option and 2-message exchange for address assignment or prefix delegation.
- Support for transmission of Reconfigure messages.
- Any relay agent functions.
- Any specific mechanism for management access to assigned address, delegated prefixes, or lifetimes.
- Any authentication mechanisms.
- The server unicast option.
- Processing for user class or vendor class options received from clients.

9 IPV4 DATA FORWARDING AND NAPT OPERATION

9.1 Introduction

The normative requirements of this section are mandatory for an eRouter that implements the IPv4 Protocol Enabled Mode and/or the Dual IP Protocol Enabled Mode as defined in Section 6.

9.1.1 Assumptions

- There is only a single Operator-Facing IP Interface on the eRouter.
- There is typically a single Customer-Facing IP interface on the eRouter.
- At least one globally-routable IPv4 address is available to the eRouter's Operator-Facing IP Interface.
- The Operator-Facing IP Interface is Ethernet encapsulated.
- The Customer-Facing IP interface is Ethernet encapsulated.

9.1.2 Overview

IPv4 Forwarding in the eRouter consists of three logical sub-elements:

- IPv4 Router.
- NAPT (Network Address Port Translation).
- ARP (Address Resolution Protocol).

The IPv4 Router sub-element is responsible for forwarding packets between the Operator-Facing IP Interface and the Customer-Facing IP interfaces. This includes looking up the IPv4 Destination address to make a forwarding decision on whether to forward the packet from one of its interfaces to another one of its interface or to its internal stack.

Packet handling in the eRouter for NAPT includes:

- Providing a form of IPv4 address translation that allows for multiple IPv4 hosts on the Customer-Facing IP interfaces while presenting a small number of IPv4 addresses on the Operator-Facing IP Interface.
- Preventing unnecessary traffic on the Customer-Facing IP interfaces.
- Preventing traffic from one CPE device to another CPE device from traversing to the Operator-Facing Interface.

The ARP protocol on the eRouter provides a mechanism for converting IPv4 network addresses to Ethernet MAC addresses on both Customer-Facing IP interfaces and the Operator-Facing IP Interface.

9.2 System Description

9.2.1 Overview

Some eRouters may have multiple customer ports that are connected to the same logical IP router interface. One scenario would be when the eRouter has an 802.11 wireless port and an 802.3 Ethernet port on the single Customer-Facing logical IP interface. The text in this section uses the term "Customer-Facing IP interface" to refer to a single Customer-Facing logical IP router interface connected to the eRouter that is not necessarily mapped one-to-one with the number of Customer-Facing ports on the eRouter. This text documents the behavior of a single Customer-Facing

IP interface, though it is possible that an eRouter could have multiple Customer-Facing IP interfaces. It is vendor-specific how to route between Customer-Facing Interfaces and the Operator-Facing IP Interface when there are multiple Customer-Facing IP interfaces.

Packets need to be processed by each of the three sub-elements in a very specific order (see Figure 9-1). The order is different depending on whether packets are received from a Customer-Facing IP interface or the Operator-Facing IP Interface.

When receiving packets from the Customer-Facing IP interface, the eRouter first attempts to route the packet through the router sub-element. If the router sub-element forwards the packet to the Operator-Facing Interface, the packet is passed to the NAPT sub-element to see if the packet requires NAPT translation. Once the NAPT sub-element has completed its work, the packet is sent to the ARP sub-element to resolve the IPv4 network address to Ethernet MAC. Then the packet is encapsulated in an Ethernet header and sent out the operator interface. If the router sub-element forwards the packet back out the Customer-Facing IP interface (perhaps because the client is on a different private subnet), the packet is sent to the ARP sub-element to resolve the IPv4 network address to Ethernet MAC. Then the packet is encapsulated in an Ethernet header and sent out the appropriate interface. No NAPT processing is necessary for packets routed back out the Customer-Facing IP interface.

When packets are received from the Operator-Facing Interface, they are immediately sent to the NAPT sub-element to translate the IPv4 network addresses back to addresses within the domain of the router sub-element. Once the NAPT has been performed on the packet, it is then sent to the router sub-element. If the router sub-element forwards the packet to the Customer-Facing IP interface, it sends the packet to the ARP sub-element to resolve the IPv4 network address to Ethernet MAC, encapsulates the packet in an Ethernet header, and sends the packet out the appropriate interface. If the router sub-element forwards the packet back to the Operator-Facing IP Interface, it is vendor-specific how to deal with the packet. Some implementations may choose to forward the packet back to the operator network; some may choose to drop the packet. Regardless, traffic should not be sent to a given eRouter from the operator network unless it is destined for a subnet known to the Customer-Facing IP interface.

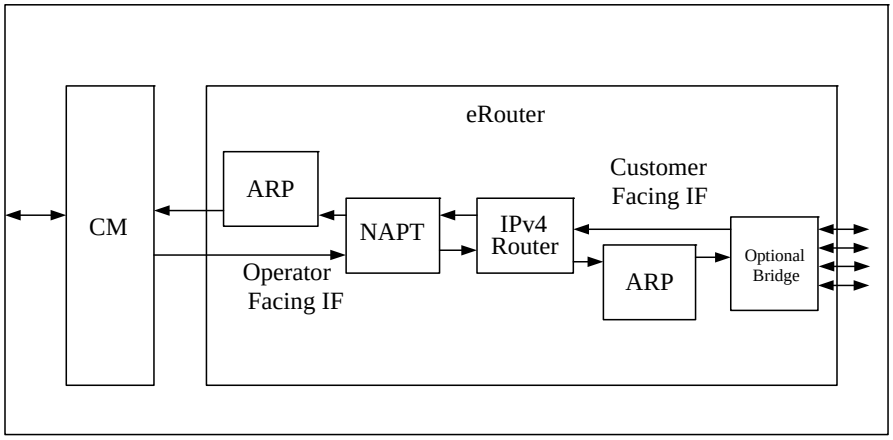

Figure 9-1 - eRouter IPv4 Forwarding Block Diagram

9.3 IPv4 Router

When the eRouter's IPv4 Router sub-element receives a packet from its NAPT sub-element (received initially by its Operator-Facing IP Interface), it validates the IPv4 header in the packet. The eRouter MAY validate the IPv4 header in accordance with [RFC 1812], section 5.2.2. As defined in [RFC 1812], section 5.3.1, the eRouter MUST decrement the IP TTL field by at least one when forwarding the packet either back to the Customer-Facing IP interface, or out the Operator-Facing Interface. Packets forwarded to the eRouter's local IP stack for processing, MUST NOT decrement the TTL. Once the IPv4 header has been validated, the eRouter processes the destination IPv4 address of the packet. If the destination IPv4 address matches the eRouter's public address assigned to its

Operator-Facing IP Interface, the eRouter sends the packet to its local IP stack for processing. If the destination IPv4 address does not match this address, the eRouter determines the next-hop address of the destination in order to forward the packet. The next-hop can be another router or a client directly connected to its Customer-Facing IP interface. The next-hop is determined by comparing the destination IPv4 address to the subnets assigned to its Customer-Facing IP interface. If the destination IPv4 address matches any of the prefixes assigned to the Customer-Facing IP interface, the destination is considered directly connected, or "on-link", and the next-hop to use for ARP purposes is the destination IPv4 address. If it does not match, the destination is considered remote or "off-link", and the next-hop to use for ARP purposes is the address of the intermediate router. Discovering other routers on the Customer-Facing IP interface is vendor-specific. If the eRouter cannot determine the next-hop of the IPv4 destination, then it MUST drop the packet.

When the eRouter's IPv4 Router sub-element receives a packet from its Customer-Facing IP interface, it validates the IPv4 header in the packet. The eRouter MAY validate the IPv4 header in accordance with [RFC 1812], section 5.2.2. As defined in [RFC 1812], section 5.3.1, the eRouter MUST decrement the IP TTL field by at least one when forwarding the packet, either back to the Customer-Facing IP Interface, or out the Operator-Facing Interface. Packets forwarded to the eRouter's local IP stack for processing, MUST NOT decrement the TTL. Once the IPv4 header has been validated, the eRouter processes the destination IPv4 address of the packet. If the destination IPv4 address matches one of the private addresses assigned to the eRouter, it sends the packet to its local IP stack for processing. If the destination IPv4 address does not match one of these addresses, the eRouter determines the next-hop address of the destination in order to forward the packet. The next-hop can be another router or a client directly connected to either its Operator-Facing IP Interface, or back out its Customer-Facing IP Interface. The next-hop is determined by comparing the destination IPv4 address to the subnets assigned to the IP interface on which the eRouter is transmitting. If the destination IPv4 address matches a sub-net prefix, the destination is considered directly connected or "on-link", and the next-hop to use for ARP purposes is the destination IPv4 address. If it does not match, the destination is considered remote or "off-link", and the next-hop to use for ARP purposes is the address of the intermediate router. The typical scenario for packets routed to the Operator-Facing IP Interface is that the next-hop router will be the eRouter's default, learned via DHCP, Section 7.1, which will be the CMTS. Discovering other routers, aside from the CMTS (or routing delegate chosen by the DHCP server if the CMTS is a bridge) on the Operator-Facing IP Interface, is vendor-specific. Discovery of other directly connected devices on the Operator-Facing IP Interface is also vendor-specific. The typical scenario for packets routed back out the Customer-Facing IP Interface is that the next-hop is a local host on a different subnet than that of the source, but directly connected to the eRouter. If the eRouter cannot determine the next-hop of the IPv4 destination address, it MUST drop the packet.

Regardless of whether the packet was received from the Customer-Facing IP Interface or the Operator IP Interface, the eRouter MUST generate an appropriate ICMP error message as described in [RFC 792] to identify the reason for dropping an IPv4 datagram, except in the follow cases:

- The drop is due to congestion.
- The packet is itself an ICMPv4 error message.
- The packet is destined for an IPv4 broadcast or multicast address.
- The source IPv4 address of the packet is invalid as defined by [RFC 1812], section 5.3.7.
- The packet is a fragment and is not the first fragment (i.e., a packet for which the fragment offset in the IPv4 header is nonzero).
- The eRouter's IPv4 router sub-element MUST process and/or generate the following ICMPv4 messages when appropriate:

0	Echo Reply	[RFC 792]
3	Destination Unreachable	[RFC 792]
11	Time Exceeded	[RFC 792]

NOTE: It is considered inappropriate for the eRouter's IPv4 router sub-element to generate ICMPv4 Destination Unreachable messages on the operator-facing interface. [13]

The eRouter MUST have at least one MAC address for its Operator-Facing IP Interface and one MAC address for its Customer-Facing IP Interface. The eRouter MUST share these source MAC addresses for IPv4 and IPv6. The eRouter MUST use the MAC address assigned to its Operator-Facing IP Interface as the source MAC address for all packets that it sends out its Operator-Facing IP Interface. The eRouter MUST use the MAC address assigned to the Customer-Facing IP Interface as the source MAC address for all packets that it sends out its Customer-Facing IP Interfaces.

The eRouter MUST forward broadcast packets received on either interface only to the eRouter's IP stack. The eRouter MUST NOT forward broadcast packets received on either interface to any interface other than the eRouter's IP stack.

9.4 NAPT

The eRouter MUST implement an NAPT function compliant with traditional Network Address Port Translation (NAPT) [RFC 3022], section 2.2. Per [RFC 3022], NAPT "is a method by which many network addresses and their TCP/UDP (Transmission Control Protocol/User Datagram Protocol) ports, are translated into a single network address and its TCP/UDP ports". Also, per [RFC 3022], the purpose of NAPT functionality is to "provide a mechanism to connect a realm with private addresses to an external realm, with globally unique registered addresses". The text in the NAPT sections below uses the term "public address(es)" to refer to the addresses reachable by the eRouter on its Operator-Facing IP Interface, assuming that they are globally-unique registered addresses. Note that an IP address that the eRouter views as globally unique, may be private to the operator's network. However, from the eRouter's perspective, these addresses are unique enough to ensure proper delivery to the next router upstream, and assumed to be globally unique.

Traditional NAPT is the simplest and most straightforward version of NAPT. Other versions that allow for mixtures of public and private network addresses on the Customer-Facing IP interface, or that allow users from the Operator-Facing IP Interface to establish translations to the Customer-Facing IP Interface, are not required by the eRouter and not discussed in this specification. Traditional NAPT requires that addresses used within the private network on Customer-Facing IP Interfaces cannot overlap with any public addresses reachable by the Operator-Facing IP Interface. Therefore, the eRouter MUST use any of the private IPv4 network addresses described in [RFC 1918] for its Customer-Facing IP interface. The Customer-Facing IP Interface is considered to be a member of one private realm. A private realm is a single domain of private addresses. This means that an eRouter cannot connect to multiple private realms or private address domains.

The eRouter MAY advertise routes to destinations on the Operator-Facing IP Interface on the private network. The eRouter MUST NOT advertise routes to private destinations on the Customer-Facing IP Interface. Destinations on the Customer-Facing IP Interface MUST NOT be propagated onto the Operator-Facing IP Interface.

The eRouter MUST create NAPT translations dynamically based on receiving a packet from a private source on the Customer-Facing IP Interface attempting to access a public address on the Operator-Facing IP Interface, as described in Section 9.4.1.

[13] Revised per eRouter-N-11.0991-1 on 5/6/11 by JB.

For packets that traverse the NAPT function, the eRouter MUST always map a combination of private IPv4 address and port number to the same combination of public IPv4 address and port number. That is, the eRouter does not implement a symmetric Network Address Translation (NAT) as defined in [RFC 3489].

The eRouter MUST NOT create NAPT translations when public sources on the Operator-Facing IP Interface attempt to access private destinations on the Customer-Facing IP Interface. Connectivity between two devices that both live on the Customer-Facing IP Interface, but on different subnets, do not require NAPT translations, as they are required to be part of the same private realm. Therefore, the eRouter MUST NOT create NAPT translations to allow connectivity between CPEs that live on the Customer-Facing IP Interface.

In the following sections, the term Private Network Address Port (PNAP) refers to the network address and TCP/UDP port of a device on Customer-Facing IP interface that is using a private network address. The term Global Network Address Port (GNAP) refers to the network address and TCP/UDP port of that same device on Operator-Facing IP Interface after it has been translated by NAPT.

9.4.1 Dynamically Triggered NAPT Translations

Dynamically-triggered NAPT is invoked when a device on the Customer-Facing IP Interface with a private network address attempts to initiate one or more sessions to a public destination on the Operator-Facing IP Interface. In this case, the eRouter creates a mapping of source PNAP to GNAP and simultaneously creates a mapping of destination GNAP to PNAP for the return packets. The eRouter then replaces the source PNAP fields of the packet with its corresponding GNAP fields and forwards the packet out the Operator-Facing IP Interface. Once the external destination responds, the eRouter intercepts the reply and changes the previously inserted GNAP fields (now destination) back to the original PNAP values.

The eRouter MUST timeout dynamically-created NAPT translations to ensure that stale entries get removed. This timeout value MUST default to 300 seconds. This time value MAY be configurable. Other mechanisms can be used (like analyzing TCP session state) to time out the translations sooner, but the eRouter MUST still time out translations based on the timeout time in case the more advanced mechanism fails (e.g., because packet loss occurred and the eRouter did not see the final packets of a TCP flow).

9.4.2 Application Layer Gateways (ALGs)

Many applications are hampered by NAPT for various reasons. A common problem is the appearance of IPv4 address and/or port information inside the application payload that is too deep into the packet to be manipulated by NAPT, which operates at the network and transport layers. ALGs can be deployed to work around some of the problems encountered, but if the payload of such packets is secured, (by secure transport or application level security) the application cannot work. Another common reason NAPT causes problems is when applications exchange address/port information to establish new connections, creating interdependencies that NAPT cannot know about. The subsections following describe specific ALGs required by the eRouter.

9.4.2.1 ICMP Error Message ALG

ICMP error messages are required for the well-known trace-route network debugging tool to work across the eRouter. This ALG is described in detail in [RFC 3022], section 4.3. The ICMP error message ALG MUST be implemented by the eRouter. Briefly stated, the eRouter MUST translate both the outer and inner IPv4 headers in the ICMP error message in order for the protocol to work correctly, when packets traverse through the NAPT sub-element.

9.4.2.2 FTP ALG

FTP is a fairly widely-used protocol, so the FTP ALG is one of the most important ALGs. The issue with FTP is that it uses the body of the control session packets to signal the data session parameters, including the new TCP ports, to use for the data session. Since NAPT relies heavily on the TCP port field in order to translate between the

private and public realm, this ALG is necessary to understand the new ports to be used by the ensuing data session. This ALG is described in detail in [RFC 3022], section 4.4. The FTP ALG MUST be implemented by the eRouter.

9.4.3 Multicast NAPT

IPv4 Multicast packets are a special case for NAPT and will need special handling at the eRouter. One scenario where forwarding of IP Multicast packets at the eRouter will need special handling is when a video source is using a private network address on a Customer-Facing IP Interface. In general, for video sources on the Customer-Facing IP Interface to work, the eRouter would be required to run at least one industry-standard multicast routing protocol to advertise the flows.

Since the eRouter will support IGMP proxy for IGMP v2 and v3, there is no reason to support a special translation for multicast packets in the eRouter for IGMP messages from private network addresses arriving on the Customer-Facing IP interface, as they will be consumed by the eRouter and new IGMP messages will be sent by the proxy agent from a public source network address on the Operator-Facing IP Interface.

9.5 ARP

The ARP function in the eRouter MUST be compliant with the following RFCs:

- An Ethernet Address Resolution Protocol [RFC 826].
- Requirements for IP Version 4 Routers [RFC 1812], section 3.3.2.
- Requirements for Internet Hosts [RFC 1122], section 2.3.2.

The ARP function in the eRouter is limited to IPv4 network addresses (pln= 4) and Ethernet hardware addresses (hln=6). When the eRouter needs to forward an IPv4 packet to a given IP address on either the Operator-Facing IP Interface or the Customer-Facing IP Interface, it consults a table of IPv4 network addresses that each map to Ethernet addresses. If the corresponding IPv4 network address is found in the table, its corresponding Ethernet address MUST be used as the Ethernet destination address of the packet. If the corresponding IPv4 network address is not found, the eRouter MUST start the ARP protocol in hopes that it will learn the IPv4 network address to Ethernet address association. The eRouter MUST use its own MAC address, as described in Section 9.3, as the source MAC address and source hardware address of all ARP packets.

The eRouter creates ARP translations, dynamically based on the eRouter, receiving an ARP reply destined for one of the eRouter's IPv4 network addresses. The eRouter also creates ARP translations, dynamically based on the eRouter, receiving an ARP request destined for one of the eRouter's IPv4 network addresses.

ARP entries maintained by the eRouter need careful examination before being aged. Both voice and video present humanly noticeable negative affects when ARP entries are removed in the middle of a session. [RFC 1122] suggests several different ways to age ARP entries in section 2.3.2.1. The eRouter SHOULD use option 2 – "Unicast polling", which allows for the ARP entry to stay fresh and in the ARP table as long as possible. This option is well-suited for routers that expect to have fairly small ARP tables and want long-term uninterrupted connectivity.

9.6 IPv4 Multicast

The eRouter learns IP multicast group membership information received on the Customer-Facing Interfaces and proxies it on the Operator-Facing Interface towards the next upstream multicast router. The eRouter forwards IPv4 multicast packets downstream based on the information learned at each Customer-Facing Interface.

The eRouter proxies IGMP information upstream actively by implementing mutually-independent IGMPv3 router functionality on Customer-Facing Interfaces, and IGMPv3 group member functionality on the Operator-Facing

Interface. On each IP interface, and independently of other IP interfaces, the eRouter generates, terminates, and processes IGMP messages according to IGMPv3 requirements. For example, the version of IGMP used on the cable network or the local area network will be defined locally at each network. The eRouter may send IGMPv2 reports on the Operator-Facing Interface while generating IGMPv3 queries on Customer-Facing Interfaces.[14]

The following elements define the eRouter IPv4 multicast behavior (also shown in Figure 9-2):

- An IGMPv3 Group Member that implements the group member part of IGMPv3[RFC 3376] on the Operator-Facing Interface.
- An IGMPv3 Router that implements the router portion of IGMPv3 [RFC 3376] on each Customer-Facing Interface.
- A subscription database per Customer-Facing Interface with multicast reception state of connected CPEs.
- An IPv4 Group Membership Database that merges subscription information from all the Customer-Facing Interfaces.

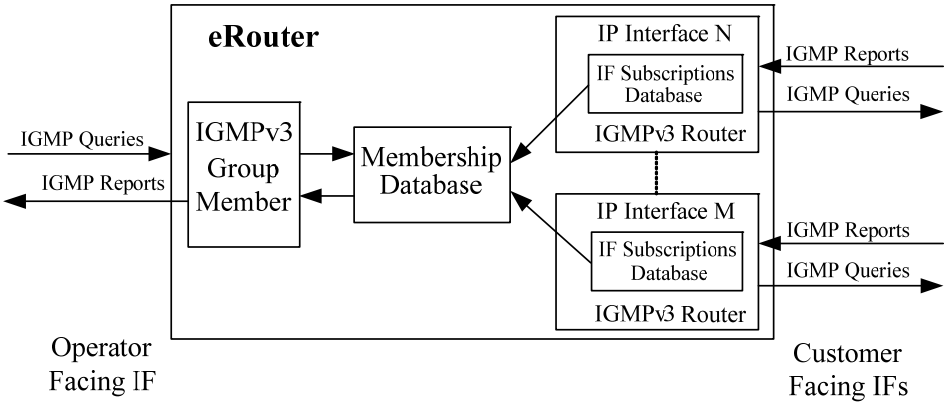

Figure 9-2 - eRouter IPv4 Multicast Forwarding Block Diagram

Central to the operation of the IGMPv3 Router(s) and IGMPv3 Group Member is the IPv4 Group Membership Database, through which the IGMPv3 Router(s) and IGMPv3 Group Member indirectly relate. This database condenses multicast reception state collected by the IGMPv3 Router(s) from connected CPEs. This information is used by the IGMPv3 Group Member on the Operator-Facing Interface as its own multicast reception interface state.

9.6.1 IGMP Proxying

The eRouter maintains the multicast reception state of CPEs on each Customer-Facing Interface in the interface's multicast subscription database. The eRouter obtains multicast reception state information of CPEs through the implementation of an IGMPv3 Router on each Customer-Facing Interface. Multicast reception state arrives at the eRouter in the form of IGMP Report messages transmitted by CPEs. The eRouter MUST implement the router portion of IGMPv3 [RFC 3376] on each Customer-Facing Interface. The eRouter MUST maintain, for each Customer-Facing Interface, the IPv4 multicast reception state of connected CPEs.

In the event of multiple queriers on one subnet, IGMPv3 elects a single querier based on the querier IP address. However, the querier election rules defined for IGMPv3 do not apply to the eRouter. The eRouter MUST always act as an IGMP querier on its Customer-Facing Interfaces.

On the Operator-Facing Interface, the eRouter MUST implement the group member portion of IGMPv3 [RFC 3376]. The eRouter MUST merge the multicast reception state of connected CPEs into an IPv4 group

[14] Sentence modified by eRouter-N-06.0352-1 by kb 2/2/07.

membership database as described in Section 9.6.1.1. The eRouter MUST use the IPv4 group membership database as multicast reception interface state per [RFC 3376], section 3.2, on the Operator-Facing Interface. Thus, when the composition of the group membership database changes, the eRouter reports the change with an unsolicited report sent on the Operator-Facing Interface. When queried by an upstream multicast router, the eRouter also responds with information from the group membership database.

The eRouter MUST NOT perform the router portion of IGMPv3 on the Operator-Facing Interface.

9.6.1.1 IPv4 Group Membership Database

The eRouter's Membership Database is formed by merging the multicast reception state records of Customer-Facing Interfaces. In compliance with [RFC 3376], the eRouter keeps per Customer-Facing Interface and per multicast address joined one record of the form:

(multicast address, group timer, filter-mode, (source records))

With source records of the form:

(source address, source timer)

The eRouter keeps an IPv4 Group Membership Database with records of the form:

(multicast-address, filter-mode, source-list)

The eRouter uses the IPv4 Group Membership Database records as the interface state for the IGMPv3 Group Member implementation on the Operator-Facing Interface. Each record of the IPv4 Group Membership Database is the result of merging all subscriptions for that record's multicast-address on Customer-Facing Interfaces. For each IPv4 multicast group joined on any Customer-Facing Interface, the eRouter MUST abide by the following process to merge all customer interface records for the group, into one Group Membership Database record:

- First, the eRouter pre-processes all customer interface group records by:

  Converting IGMPv1 and IGMPv2 records into IGMPv3 records.

  Removing group and source timers from IGMPv3 and converted records.

  Removing every source whose source timer is greater than zero from records with a filter mode value of EXCLUDE.

- Then the eRouter creates an IPv4 Group Membership Database record by merging the pre-processed records, using the merging rules for multiple memberships on a single interface specified in section 3.2 of the IGMPv3 specification [RFC 3376].

9.6.2 IPv4 Multicast Forwarding

The forwarding of IPv4 multicast packets received on any interface onto a Customer-Facing Interface is determined by the known multicast reception state of the CPEs connected to the Customer-Facing Interface. The eRouter MUST replicate an IPv4 multicast session on a Customer-Facing Interface, if at least one CPE device connected to the interface has joined the session. The eRouter MUST NOT replicate an IPv4 multicast session on a Customer-Facing Interface, if no CPE device connected to the interface has joined the session.

The eRouter MUST NOT forward IPv4 multicast packets received on any interface, i.e., any Customer-Facing or the Operator-Facing Interface, back to the same interface.

The eRouter MUST NOT forward IGMP messages received on any IP interface onto another IP interface. Except for IGMP packets, the eRouter MUST forward all IPv4 multicast traffic received on Customer-Facing Interfaces

onto the Operator-Facing Interface. Operator control of multicast traffic forwarding onto the cable network, if desired, can be done through the implementation of filters at the eCM.

9.6.3 IPv4 Multicast Forwarding Example

The eRouter in this example has two Customer-Facing Interfaces; CFIA and CFIB, connected to one LAN segment each. On CFIA, there are two CPEs connected; CPE1 and CPE2. CPE1 is IGMPv2 capable and will attempt to join group 224.0.100.1. CPE2 is IGMPv3 capable and will attempt to join group 224.128.100.1 from all sources. On CFIB, there is one CPE connected, CPE3, which is IGMPv3 capable and that will attempt to join group 224.128.100.1, except from source 198.200.200.200.

The router upstream of the eRouter (e.g., the CMTS) supports and is configured to operate in IGMPv3 mode, and thus the eRouter works in IGMPv3 mode on the Operator-Facing Interface.[15]

The setup is shown in Figure 9-3:

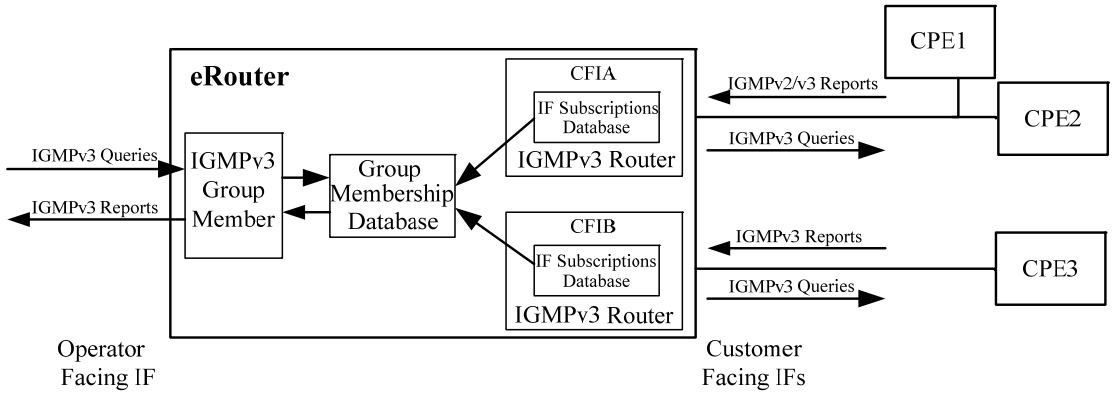

Figure 9-3 - IPv4 Multicast Forwarding Example[16]

The CPEs send reports as follows:[17]

Report From	Report Version	Multicast Address	Record Type	Source Address
CPE1	IGMPv2	224.0.100.1	N/A	N/A
CPE2	IGMPv3	224.128.100.1	EXCLUDE	Null
CPE3	IGMPv3	224.128.100.1	EXCLUDE	198.200.200.200

Because CPE1 sends an IGMPv2 report for group 224.0.100.1, CFIA operates in IGMPv2 compatibility mode for this group. On the other hand, CFIA and CFIB operate in IGMPv3 mode for group 224.128.100.1, because they receive IGMPv3 reports for this group from CPE2 and CPE3, respectively. The eRouter multicast reception state at each Customer-Facing Interface is the following:[18]

Interface	Multicast Address	Group Timer	Filter-Mode	Source Address	Source Timer
CFIA	224.0.100.1	A	EXCLUDE	Null	0

[15] Paragraph added per eRouter-N-06.0352-1 by kb 2/2/07.
[16] Figure revised per eRouter-N-06.0352-1 by kb 2/2/07.
[17] This paragraph and table below modified per eRouter-N-06.0352-1 by kb 2/2/07.
[18] This paragraph and table below modified per eRouter-N-06.0352-1 by kb 2/2/07.

Interface	Multicast Address	Group Timer	Filter-Mode	Source Address	Source Timer
CFIA	224.128.100.1	B	EXCLUDE	Null	0
CFIB	224.128.100.1	C	EXCLUDE	198.200.200.200	0

The interface state at the eRouter's Operator-Facing Interface, stored in the Group Membership Database, is the following:

Multicast Address	Filter-Mode	Source Address
224.0.100.1	EXCLUDE	Null
224.128.100.1	EXCLUDE	Null

The eRouter uses the information in the table above as multicast reception state at the Operator-Facing Interface. For example, in response to an IGMPv3 general query, the eRouter sends an IGMPv3 report for the two records shown.

Assuming that the CMTS is transmitting downstream four multicast streams, the eRouter forwards them as follows:[19]

Stream #	Multicast Address	Source Address	eRouter forwards on interfaces	
			CFIA	CFIB
1	224.0.200.2	198.100.100.100	NO	NO
2	224.0.100.1	198.100.100.100	YES	NO
3	224.128.100.1	198.100.100.100	YES	YES
4	224.128.100.1	198.200.200.200	YES	NO

9.7 Dual-Stack Lite Operation [20]

Even as operators migrate customers from IPv4 to IPv6 addressing, a significant percentage of Internet resources and content will remain accessible only through IPv4. As a consequence of the slow transition to IPv6 on the part of content providers, operators require mechanisms to allow customers to continue to access content and resources using IPv4 even after the last IPv4 allocations have been fully depleted. This necessitates multiplexing several customers behind a single IPv4 address. One technology that satisfies operator requirements for IPv4 address extension is dual-stack lite.

Dual-stack lite enables an operator to share IPv4 addresses among multiple customers by combining two well-known technologies: IP in IP (IPv4-in-IPv6) tunneling and NAT. More specifically, dual-stack lite encapsulates IPv4 traffic inside an IPv6 tunnel and sends it to an operator NAT device.

To facilitate IPv4 extension over an IPv6 network, the eRouter MAY implement dual-stack lite B4 (client) functionality, as specified in section 5 of [I-D.DS-Lite]. When dual-stack lite is enabled, the eRouter acquires an IPv6 address on its Operator-Facing Interface and learns the address of the operator NAT device via DHCPv6. It encapsulates IPv4 traffic inside IPv6 sourced from its Operator-Facing Interface and destined for the operator NAT device.

[19] Table below modified per eRouter-N-06.0352-1 by kb 2/2/07.
[20] Section added per eRouter-N-09.0877-2 on 5/11/10 by JB.

10 IPV6 DATA FORWARDING

The normative requirements of this section are mandatory for an eRouter that implements the IPv6 Protocol Enabled Mode and/or the Dual IP Protocol Enabled Mode, as defined in Section 6.

Assumptions

- There is only a single Operator-Facing IP Interface on the eRouter.
- There is typically a single Customer-Facing IP Interface on the eRouter.
- The Operator-Facing IP Interface is Ethernet encapsulated.
- The Customer-Facing IP Interface is Ethernet encapsulated.
- The eRouter advertises itself as a router (using ND) on both the Operator-Facing and Customer-Facing Interfaces so clients and routers on either interface learn about the eRouter.
- All the eRouters are on separate links and therefore will not see each other's RAs.

10.1 Overview

The IPv6 eRouter is responsible for implementing IPv6 routing. This includes looking up the IPv6 Destination address to decide which of the eRouter interfaces to send the packet.

The ND protocol is required on the eRouter. Like ARP in IPv4, it provides a mechanism for converting IPv6 network addresses to Ethernet MAC addresses on both the Customer-Facing IP interfaces and the Operator-Facing IP Interface. It also provides a mechanism for the eRouter to advertise its presence, host configuration parameters, routes, and on-link preferences.

Figure 10-1 shows a block diagram of the IPv6 eRouter with an IPv6 Router block and an ND block. The IPv6 functionality, however, does not have the clean separation indicated by these blocks. The IPv6 Routing and Neighbor Discovery blocks are closely intertwined and, therefore, are discussed together under the same subsection.

The IPv6 eRouter uses a local IPv6 routing table to forward packets. The eRouter creates the IPv6 routing table upon initialization of the IPv6 portion of the eRouter and adds entries according to the receipt of Router Advertisement messages containing on-link prefixes and routes.

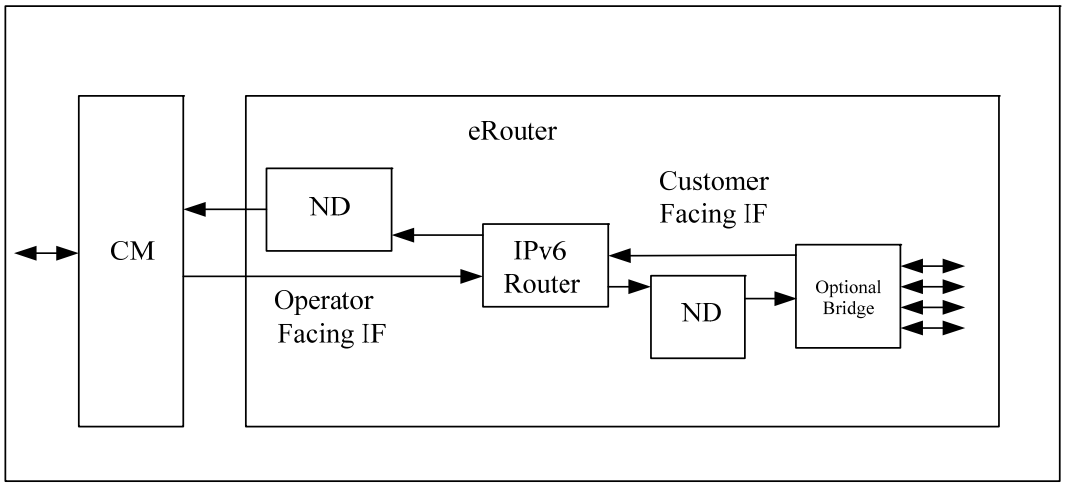

Figure 10-1 - eRouter IPv6 Forwarding Block Diagram

10.2 System Description [21]

Except when noted, the ND function in the eRouter MUST comply with the Neighbor Discovery for IPv6 [RFC 2461]. Per [RFC 4861], ND is used "to determine the link-layer addresses for neighbors known to reside on attached links and to quickly purge cached values that become invalid."

Several sections of [RFC 4861] do not apply to the eRouter. These sections include:

section 6.2.7 - RA Consistency

section 6.2.8 - Link-local Address Change

section 7.2.8 - Proxy Neighbor Advertisements

section 8 - Redirect Function

section 11 - Security Considerations

section 12 - Renumbering Considerations

The eRouter MUST support the following ND messages per [RFC 4861]: Router Solicitation, Router Advertisement, Neighbor Solicitation, and Neighbor Advertisement.

The eRouter receives a packet and checks the destination address of the packet. If the destination IPv6 address matches the address assigned to the eRouter's IP interface, the eRouter forwards the packet to its local IP stack for processing. If the destination IPv6 address does not match the eRouter's address, the eRouter determines the next-hop address of the destination in order to forward the packet. The next-hop can be a router, or the destination itself. The next-hop is determined by comparing the destination IPv6 address to the subnets assigned to the IP interface on which the eRouter is transmitting out. If the destination IPv6 address matches a sub-net prefix, the destination is considered directly connected or "on-link", and the next-hop to use for ND purposes is the destination IPv6 address. If it does not match, the destination is considered remote or "off-link", and the next-hop to use for ND purposes is the address of the intermediate router.

The typical scenario for packets routed to the Operator-Facing IP Interface is that the next-hop router will be the eRouter's default, learned via Router Advertisement [RFC 3315], from the CMTS. Discovering other routers, aside from the CMTS (or routing delegate chosen if the CMTS is a bridge), on the Operator-Facing IP Interface is vendor-specific. Discovery of other directly-connected devices on the Operator-Facing IP Interface is also vendor-specific. The typical scenario for packets routed back out the Customer-Facing IP Interface is that the next-hop is a local host on a different subnet than that of the source, but directly connected to the eRouter. If the eRouter cannot determine the next-hop of the IPv6 destination address, then it MUST drop the packet.

Once a next-hop is determined, the eRouter's Neighbor Cache is consulted for the link-layer address of the next-hop address. If necessary, address resolution is performed. Address resolution is accomplished by multicasting a Neighbor Solicitation that prompts the addressed neighbor to return its link-layer address in a Neighbor Advertisement. The neighbor cache entry is then updated with this link-layer address, and the eRouter then forwards the packet to the link-layer address contained in this cache entry. If an error occurs at any point in the process, the eRouter discards the packet. Regardless of whether the packet was received from the Customer-Facing IP Interface or the Operator IP Interface, the eRouter MUST generate an appropriate ICMP error message, as described in [RFC 2463], to identify the reason for dropping an IPv6 datagram, except in the follow cases:

[21] Revised per eRouter-N-09.0877-2 on 5/11/10 by JB.

- The drop is due to congestion.
- The packet is itself an ICMPv6 error message.
- The packet is destined for an IPv6 multicast address (except if the packet is the "Packet Too Big Message" or the "Parameter Problem Message", as explained in [RFC 2463], section 2.4, paragraph (e)).
- The packet is destined for a link-layer broadcast address, except as noted above.
- The packet is destined for a link-layer multicast address, except as noted above.
- The source IPv6 address of the packet does not uniquely identify a single node, as explained in detail in [RFC 2463], section 2.4, paragraph (e).
- The eRouter MUST process and/or generate the following ICMPv6 messages when appropriate:

1	Destination Unreachable	[RFC 2463]
3	Time Exceeded	[RFC 2463]
129	Echo Reply	[RFC 2463]
130	Multicast Listener Query	[RFC 3810]
131	Multicast Listener Report	[RFC 3810]
132	Multicast Listener Done	[RFC 3810]
133	Router Solicitation	[RFC 4861]
134	Router Advertisement	[RFC 4861]
135	Neighbor Solicitation	[RFC 4861]
136	Neighbor Advertisement	[RFC 4861]
143	Version 2 Multicast Listener Report	[RFC 3810]

NOTE: It is considered inappropriate for the eRouter to generate ICMPv6 Destination Unreachable messages on the operator-facing interface. [22]

The eRouter is responsible for decrementing the Hop Limit field in the IPv6 packet that it is going to forward. If the eRouter receives an IPv6 packet with a Hop Limit of zero, or the eRouter decrements an IPv6 packet's Hop Limit to zero, it MUST discard that packet and send an ICMPv6 Time Exceeded message with Code 0 to the source of that IPv6 packet.

The eRouter is also responsible for reinserting the Ethernet header of IPv6 packets. The eRouter has at least one MAC address for its Operator-Facing IP Interface and one MAC address for its Customer-Facing IP Interface that are shared for IPv4 and IPv6 (see Section 8.3). The eRouter MUST use the MAC address assigned to its Operator-Facing IP Interface as the source MAC address for all IPv6 packets that it sends out its Operator-Facing IP Interface. The eRouter MUST use the MAC address assigned to the Customer-Facing IP Interface as the source MAC address for all IPv6 packets that it sends out its Customer-Facing IP Interfaces. Per [RFC 4861], the eRouter uses the MAC address of the next-hop address learned via Neighbor Discovery as the destination MAC address for the IPv6 packet.

The eRouter MUST forward link-local multicast packets received on either interface only to the eRouter's IP stack. The eRouter MUST NOT forward link-local multicast packets received on either interface to any interface other than the eRouter's IP stack.

[22] Revised per eRouter-N-11.0991-1 on 5/6/11 by JB.

10.3 IPv6 Multicast

The eRouter learns IP multicast group membership information received on the Customer-Facing Interfaces and proxies it on the Operator-Facing Interface towards the next upstream multicast router. The eRouter forwards IPv6 multicast packets downstream based upon the information learned at each Customer-Facing Interface.

The eRouter proxies MLD information upstream actively by implementing mutually-independent MLDv2 router functionality on Customer-Facing Interfaces and MLDv2 multicast listener functionality on the Operator-Facing Interface. On each IP interface, and independently of other IP interfaces, the eRouter generates, terminates, and processes MLD messages according to MLDv2 requirements. For example, the version of MLD used on the cable network or the local area network will be defined locally at each network. The eRouter may send MLDv1 reports on the Operator-Facing Interface while generating MLDv2 queries on Customer-Facing Interfaces.[23]

The following elements define the eRouter IPv6 multicast behavior (also shown in Figure 10-2):

- An MLDv2 Multicast Listener that implements the multicast listener part of MLDv2 [RFC 3810] on the Operator-Facing Interface;
- An MLDv2 Router that implements the router part of MLDv2 [RFC 3810] on each Customer-Facing Interface;
- A Subscription Database per Customer-Facing Interface with multicast reception state of connected CPEs;
- An IPv6 Group Membership Database that merges subscription information from all the Customer-Facing Interfaces.

These logical sub-elements are shown in Figure 10-2.

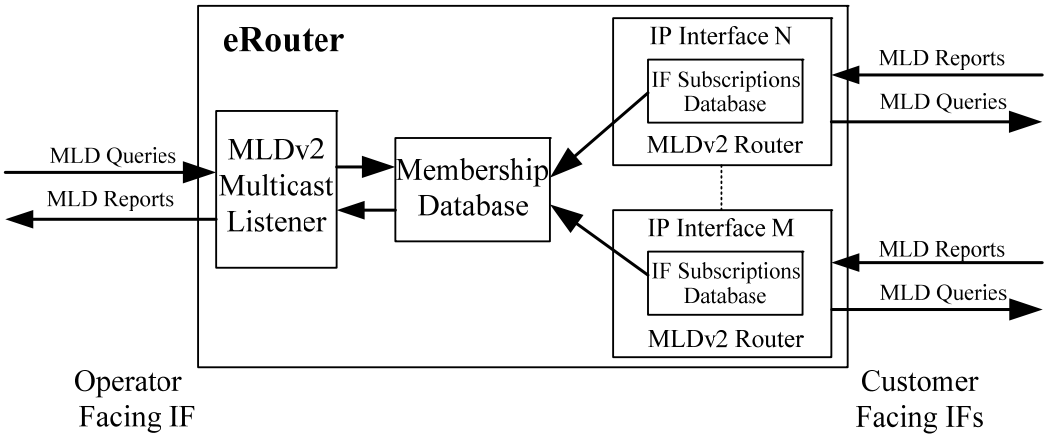

Figure 10-2 - eRouter IPv6 Multicast Forwarding Block Diagram

10.3.1 MLD Proxying

The eRouter maintains the multicast reception state of CPEs on each Customer-Facing Interface in the interface's multicast subscription database. The eRouter obtains CPE's multicast reception state information through the implementation of an MLDv2 Router on each Customer-Facing interface. Multicast reception state arrives at the eRouter in the form of MLD Report messages transmitted by CPEs. The eRouter MUST implement the router portion of the MLDv2 protocol, [RFC 3810], on each Customer-Facing Interface. The eRouter MUST maintain, for each Customer-Facing Interface, the IPv6 multicast reception state of connected CPEs.

[23] Sentence modified per eRouter-N-06.0352-1 by kb 2/2/07.

In the event of multiple queriers on one subnet, MLDv2 elects a single querier based on the querier IP address. However, the querier election rules defined for MLDv2 do not apply to the eRouter. The eRouter MUST always act as an MLD querier on its Customer-Facing Interfaces.

On the Operator-Facing Interface, the eRouter MUST implement the multicast listener portion of the MLDv2 protocol, [RFC 3810]. The eRouter MUST merge the multicast reception state of connected CPEs into an IPv6 group membership database, as described in Section 10.3.2, IPv6 Group Membership Database. The eRouter MUST use the membership database as multicast reception interface state per [RFC 3810], section 4.2, for the Operator-Facing Interface. Thus, when the composition of the IPv6 multicast membership database changes, the eRouter reports the change with an unsolicited report sent on the Operator-Facing Interface. When queried by an upstream multicast router, the eRouter also responds with information from the membership database.

The eRouter MUST NOT perform the router portion of MLDv2 on the Operator-Facing Interface.

10.3.2 IPv6 Group Membership Database

The eRouter's Membership Database is formed by merging the multicast reception state records of Customer-Facing Interfaces. In compliance with [RFC 3810], the eRouter keeps per Customer-Facing Interface and per multicast address joined one record of the form:

(multicast address, group timer, filter-mode, (source records))

With source records of the form:

(source address, source timer)

The eRouter keeps an IPv6 Group Membership Database with records of the form:

(multicast-address, filter-mode, source-list)

The eRouter uses the IPv6 Group Membership Database records as interface state for the MLDv2 Multicast Listener implementation on the Operator-Facing Interface. Each record of the IPv6 Group Membership Database is the result of merging all subscriptions for that record's IPv6 multicast-address on Customer-Facing Interfaces. For each IPv6 multicast group joined on any Customer-Facing Interface, the eRouter MUST abide by the following process to merge all customer interface records for the group into one Group Membership Database record:

- First, the eRouter pre-processes all customer interface group records by:

  Converting MLDv1 records into MLDv2 records.

  Removing group and source timers from MLDv2 and converted records.

  Removing every source whose source timer is greater than zero from records with a filter mode value of EXCLUDE.

- Then the eRouter creates an IPv6 Group Membership Database record by merging the pre-processed records, using the merging rules for multiple memberships on a single interface specified in section 4.2 of the MLDv2 specification [RFC 3810].

10.3.3 IPv6 Multicast Forwarding

The forwarding of IPv6 multicast packets received on any interface onto a Customer-Facing Interface is determined by the known multicast reception state of the CPEs connected to the Customer-Facing Interface. The eRouter MUST replicate an IPv6 multicast session on a Customer-Facing Interface if at least one CPE device connected to the interface has joined the session. The eRouter MUST NOT replicate an IPv6 multicast session on a Customer-Facing Interface if no CPE device connected to the interface has joined the session.

The eRouter MUST NOT forward IPv6 multicast packets received on any interface, i.e., any Customer-Facing or the Operator-Facing Interface, back to the same interface.

In compliance with IPv6 link-scope packet forwarding rules, the eRouter MUST NOT forward MLD messages received on an IP interface onto another IP interface. Also, the eRouter MUST NOT forward link-scoped IPv6 multicast packets received on an IP interface onto another IP interface. The eRouter MUST forward all non-link-scoped IPv6 multicast traffic received on Customer-Facing Interfaces onto the Operator-Facing Interface. Operator control of multicast traffic forwarding onto the cable network, if desired, can be done through the implementation of filters at the eCM.

10.3.4 IPv6 Multicast Forwarding Example

The eRouter in this example has two Customer-Facing Interfaces; CFIA and CFIB, connected to one LAN segment each. On CFIA, there are two CPEs connected; CPE1 and CPE2. CPE1 is MLDv1-capable and will attempt to join group FF1E::100. CPE2 is MLDv2-capable and will attempt to join group FF1E::128 from all sources. On CFIB, there is one CPE connected, CPE3, which is MLDv2 capable and that will attempt to join group FF1E::128, except from source 3FFE:2900::200.

The router upstream of the eRouter (e.g., the CMTS) supports and is configured to operate in MLDv2 mode, and thus the eRouter works in MLDv2 mode on the Operator-Facing Interface.

The setup is shown in Figure 10-3:

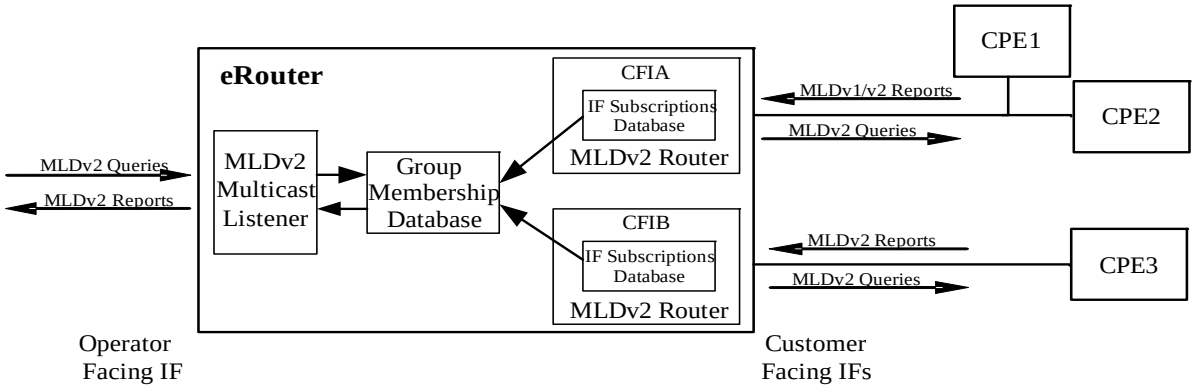

Figure 10-3 - IPv6 Multicast Forwarding Example[24]

The CPEs send reports as follows:[25]

Report From	Report Version	Multicast Address	Record Type	Source Address
CPE1	MLDv1	FF1E::100	N/A	N/A
CPE2	MLDv2	FF1E::128	EXCLUDE	Null
CPE3	MLDv2	FF1E::128	EXCLUDE	3FFE:2900::200

[24] Figure modified per eRouter-N-06.0352-1 by kb 2/2/07.
[25] This paragraph and table below modified per eRouter-N-06.0352-1 by kb 2/2/07.

Because CPE1 sends an MLDv1 report for group FF1E::100, CFIA operates in MLDv1 compatibility mode for this group. On the other hand, CFIA and CFIB operate in MLDv2 mode for group FF1E::128, because they receive MLDv2 reports for this group from CPE2 and CPE3, respectively. The eRouter multicast reception state at each Customer-Facing Interface is the following:[26]

Interface	Multicast Address	Group Timer	Filter-Mode	Source Address	Source Timer
CFIA	FF1E::100	A	EXCLUDE	Null	0
CFIA	FF1E::128	B	EXCLUDE	Null	0
CFIB	FF1E::128	C	EXCLUDE	3FFE:2900::200	0

The eRouter merges the multicast reception state of connected CPEs shown above into the Group Membership Database as follows:

Multicast Address	Filter-Mode	Source Address
FF1E::100	EXCLUDE	Null
FF1E::128	EXCLUDE	Null

The eRouter uses the information in the Group Membership Database as multicast reception state at the Operator-Facing Interface. For example, in response to an MLDv2 general query, the eRouter sends an MLDv2 report for the two records shown.

Assuming that the CMTS is transmitting four multicast streams downstream, the eRouter forwards them as follows:[27]

Stream #	Multicast Address	Source Address	eRouter forwards on interfaces	
			CFIA	CFIB
1	FF1E::200	3FFE:2900::100	NO	NO
2	FF1E::100	3FFE:2900::100	YES	NO
3	FF1E::128	3FFE:2900::100	YES	YES
4	FF1E::128	3FFE:2900::200	YES	NO

[26] This paragraph and table below modified per eRouter-N-06.0352-1 by kb 2/2/07.
[27] Table below modified per eRouter-N-06.0352-1 by kb 2/2/07.

11 QUALITY OF SERVICE

QoS on the eRouter is optional. The eRouter SHOULD support Layer 2 and Layer 3 QoS, as defined in this section. The QoS functionality described herein allows the operator to selectively provide a level of differentiation among the various data streams destined for CPE behind the eRouter. Typical applications could include Internet Protocol Television services (IPTV) and other enhanced data services, though it is anticipated that overall packet counts will still be dominated by largely undifferentiated best-effort data traffic.

Because Layer 2 (e.g., 802.1 p/Q Ethernet) headers will be stripped off as packets traverse the eRouter, the eRouter MUST prioritize the forwarding of IP packets based on the values marked in the IPv4 ToS byte or IPv6 Traffic Class field.

11.1 Downstream Quality of Service Operation

This section deals with the requirements regarding traffic going to CPEs, through the eRouter, from the Cable network.

The eRouter MUST provide two or more priority queues on each Customer-Facing Interface for traffic going to CPEs. The eRouter MAY provide a configuration mechanism to map ToS/Traffic Class field priority values to the high- and low-priority queues. As a default setting, the eRouter might use the most significant bit of the ToS/Traffic Class field to determine priority to queue mappings.

11.2 Upstream Quality of Service Operation

This section deals with traffic coming from the CPEs attached to the eRouter to the cable network.

For the purposes of QoS of upstream traffic from CPE devices, the interface between the eRouter and the embedded CM must be considered to be of infinite bandwidth, and thus no congestion should be expected to occur on this interface. Thus, the eRouter does not need to provide any queues in the upstream direction. The eRouter MAY provide a configuration mechanism to determine whether the eRouter allows CPE devices to pass QoS-tagged packets with the IP ToS/Traffic Class field intact, or whether the eRouter resets the IP ToS/Traffic Class field to 0. The eRouter MAY use the IP ToS/Traffic Class field to populate Layer 2 QoS headers to ensure upstream QoS treatment. Although other implementations are possible, one such implementation is to directly map the three most significant bits of the IP ToS/Traffic Class field into the 802.1Q priority field.

In the case where multiple Customer-Facing Interfaces are implemented, the eRouter may support additional QoS mechanisms to prioritize upstream traffic based on ingress interface.

Annex A SNMP MIB Objects supported by the eRouter

This Annex defines the SNMP MIBs that the eRouter is required to implement.

The eRouter MUST support the following MIB objects:

- ipNetToPhysicalTable [RFC 4293];
- vacmAccessTable [RFC 3415];
- vacmSecurityToGroupTable [RFC 3415];
- vacmViewTreeFamilyTable [RFC 3415];
- vacmAccessReadViewName [RFC 3415];
- vacmAccessWriteViewName [RFC 3415];
- snmpCommunityTable [RFC 3584];
- snmpTargetAddrTable [RFC 3413] ;
- snmpTargetAddrTAddress [RFC 3413];
- snmpTargetAddrTMask [RFC 3584];
- snmpTargetAddrExtTable [RFC 3584].

Additional information for the configuration and use of the above MIB objects is defined in Annex B.

A.1 eRouter Interface Numbering

The eRouter MUST use in its MIB tables when appropriate an ifIndex number of '1' for the Operator-Facing Interface and an ifIndex number of '2' for the first Customer-Facing Interface. Any additional Customer-Facing Interfaces MUST be numbered sequentially from '3' onwards.

Annex B Configuration of eRouter Operational Parameters

This Annex defines the configuration TLVs used by the eRouter and describes how the configuration parameters are transferred from the eCM to the eRouter.

B.1 eRouter SNMP Configuration

This Annex sub-section defines the configuration of SNMP access to the eRouter.

B.1.1 eRouter SNMP Modes of Operation

The eRouter MUST support SNMPv1, SNMPv2c, in SNMP-coexistence mode as defined in [RFC 3584]. The eRouter MAY support SNMPv3 as defined in [OSSIv3.0].

B.1.2 eRouter SNMP Access Control Configuration

The eRouter uses the View-based Access Control Model (VACM) for configuration of SNMPv1v2c co-existence as defined in [RFC 3584].

B.1.2.1 *View-based Access Control Model (VACM) Profile*

This section addresses the default VACM profile for the eRouter.

The eRouter MUST support a pre-installed entry in the vacmViewTreeFamilyTable [RFC 3415] as follows:

Table B-1 - vacmViewTreeFamilyTable

Column Name (* = Part of Index)	Column Value
* vacmViewTreeFamilyViewName	eRouterManagerView
* vacmViewTreeFamilySubtree	*<1.3.6.1>*
vacmViewTreeFamilyMask	Zero-length String
vacmViewTreeFamilyType	'included'
vacmViewTreeFamilyStorageType	volatile (2) or nonvolatile (3)
vacmViewTreeFamilyStatus	active (1)

The eRouter MAY also support additional views to be configured by the operator during the provisioning process, as defined in the SNMPv1v2c Access View Name encoding Annex B.3.3.4 and the SNMPv3 Access View Configuration encoding.

B.1.3 SNMPv1v2c Coexistence Configuration

This section specifies eRouter handling of the SNMPv1v2c Coexistence Configuration encodings as defined in Annex B.3.3 when included in the eRouter configuration information. The SNMPv1v2c Coexistence Configuration encoding is used to configure SNMPv3 framework tables for SNMPv1 and v2c access.

The eRouter uses the SNMPv1v2c Coexistence Configuration encodings to create entries in the following tables:

- snmpCommunityTable;
- snmpTargetAddrTable;
- vacmSecurityToGroupTable;

- vacmAccessTable;
- snmpTargetAddrExtTable.

B.1.3.1 *Mapping SNMPv1v2c Coexistence Configuration*

This section describes the mapping of SNMPv1v2c Coexistence Configuration into SNMPv3 entries.

Table B-2 provides a Variable Name as a short-hand reference to be used in the SNMPv3 tables defined in sub-sections below for each of the SNMPv1v2c Coexistence Configuration encodings. The table also defines the mapping between each of the SNMPv1v2c Coexistence Configuration encodings and the associated SNMP MIB objects.

Table B-2 - SNMPv1v2c Coexistence Configuration Mapping

Encodings	Variable Name	Associated MIB Object
SNMPv1v2c Community Name	*CommunityName*	snmpCommunityName [RFC 3584]
SNMPv1v2c Transport Address Access		
SNMPv1v2c Transport Address	*TAddress*	snmpTargetAddrTAddress [RFC 3413]
SNMPv1v2c Transport Address Mask	*TMask*	snmpTargetAddrTMask [RFC 3584]
SNMPv1v2c Access View Type	*AccessViewType*	
SNMPv1v2c Access View Name (optional, see B.3.3.4)	*AccessViewName or eRouterManagerView*	vacmAccessReadViewName and vacmAccessWriteViewName [RFC 3415]

The eRouter is not required to verify the consistency across tables.

Table B-3 through Table B-7 describe the eRouter procedures to populate the SNMPv3 framework tables to conform to the "SNMP Management Framework Message Processing and Access Control Subsystems" [RFC 3412].

When configuring entries in these SNMPv3 tables:

- The ReadViewName and WriteViewName may correspond to default entries as defined in Annex B.1.3.1 or entries created using SNMPv3 Access View Configuration (see Annex B.3.4).
- Multiple columnar objects can be configured with indexes containing the string "@eRouterRouterconfig". If these tables are configured through other mechanisms, network operators should not use values beginning with "@eRouterconfig", to avoid conflicts.

B.1.3.1.1 snmpCommunityTable

The snmpCommunityTable is defined in the "SNMP Community MIB Module" section of [RFC 3584].

The eRouter MUST create one row in snmpCommunityTable for each SNMPv1v2c Coexistence Configuration TLV as follows:

- The eRouter sets the value of snmpCommunityIndex to "@eRouterconfig_n" where ‘n’ is a sequential number starting at 0 for each TLV processed (e.g., "@eRouterconfig_0", "@eRouterconfig_1", etc.)
- The eRouter creates space separated tags in snmpCommunityTransportTag for each SNMPv1v2c Community Name sub-TLV of the SNMPv1v2c Coexistence Configuration encoding.

Table B-3 - snmpCommunityTable

Column Name (* = Part of Index)	Column Value
* snmpCommunityIndex	"@eRouterconfig_n" where n is 0..m-1 and m is the number of SNMPv1v2c Community Name TLVs
snmpCommunityName	<*CommunityName*>
snmpCommunitySecurityName	"@eRouterconfig_n"
snmpCommunityContextEngineID	<the engineID populated by the SNMP>
snmpCommunityContextName	<Zero-length OCTET STRING>
snmpCommunityTransportTag	"@eRouterconfigTag_n" where n is 0..m-1 and m is the number of SNMPv1v2c Coexistence Configuration TLVs
snmpCommunityStorageType	volatile (2)
snmpCommunityStatus	active (1)

B.1.3.1.2 snmpTargetAddrTable

The snmpTargetAddrTable is defined in the "Definitions" section of [RFC 3413].

The eRouter MUST create one row in snmpTargetAddrTable for each SNMPv1v2c Transport Address Access sub-TLV of the SNMPv1v2c Coexistence Configuration encoding.

Table B-4 - snmpTargetAddrTable

Column Name (* = Part of Index)	Column Value
* snmpTargetAddrName	"@eRouterconfigTag_n_i" where 'n' is 0..m-1 and 'm' is the number of SNMPv1v2c Coexistence Configuration TLVs. Where 'I' is 0..p-1 and p is the number of SNMPv1v2c Transport Address Access sub-TLV within the SNMPv1v2c Coexistence Configuration TLV n
snmpTargetAddrTDomain	IPv4: snmpUDPDomain [RFC 3417] IPv6: transportDomainUdpIpv6 [RFC 3419]
snmpTargetAddrTAddress (IP Address and UDP Port)	IPv4: SnmpUDPAddress [RFC 3417] OCTET STRING (6) Octets 1-4: <*TAddress*> Octets 5-6: <*TAddress*> IPv6: TransportAddressIPv6 [RFC 3419] OCTET STRING (18) Octets 1-16: <*TAddress*> Octets 17-18: <*TAddress*>
snmpTargetAddrTimeout	Default from MIB
snmpTargetAddrRetryCount	Default from MIB
snmpTargetAddrTagList	"@eRouterconfigTag_n" where n is 0..m-1 and m is the number of SNMPv1v2c Coexistence Configuration TLVs
snmpTargetAddrParams	'00'h (null character)
snmpTargetAddrStorageType	volatile (2)
snmpTargetAddrRowStatus	active (1)

B.1.3.1.3 snmpTargetAddrExtTable

The snmpTargetAddrExtTable is defined in the "SNMP Community MIB Module" section of [RFC 3584].

The eRouter MUST create one row in snmpTargetAddrExtTable for each SNMPv1v2c Transport Address Access sub-TLV of the SNMPv1v2c Coexistence Configuration encoding.

Table B-5 - snmpTargetAddrExtTable

Column Name (* = Part of Index)	Column Value
* snmpTargetAddrName	"@eRouterconfigTag_n_i" where 'n' is 0..m-1 and 'm' is the number of SNMPv1v2c Coexistence Configuration TLVs. Where 'i' is 0..p-1 and p is the number of SNMPv1v2c Transport Address Access sub-TLV within the SNMPv1v2c Coexistence Configuration TLV n
snmpTargetAddrTMask	<Zero-length OCTET STRING> when <*TMask*> is not provided in the i-th SNMPv1v2c Transport Address Access sub-TLV IPv4: SnmpUDPAddress [RFC 3417] OCTET STRING (6) Octets 1-4: <*TMask*> Octets 5-6: <UDP Port> IPv6: TransportAddressIPv6 [RFC 3419] OCTET STRING (18) Octets 1-16: <*TMask*> Octets 17-18: <UDP Port>
snmpTargetAddrMMS	Maximum Message Size

B.1.3.1.4 vacmSecurityToGroupTable

The vacmSecurityToGroupTable is defined in the "Definitions" section of [RFC 3415].

The eRouter MUST create two rows in vacmSecurityGroupTable for each SNMPv1v2c Coexistence Configuration TLV as follows:

- The eRouter sets the value of vacmSecurityName to "@eRouterconfig_n" where 'n' is a sequential number starting at 0 for each SNMPv1v2c Coexistence Configuration TLV processed (e.g., "@eRouterconfig_0", "@eRouterconfig_1", etc.);
- The eRouter sets the value of vacmGroupName to "@eRouterconfigV1_n" for the first row and "@eRouterconfigV2_n" for the second row where 'n' is a sequential number starting at 0 for each SNMPv1v2c Coexistence Configuration TLV processed (e.g., "@eRouterconfigV1_0", "@eRouterconfigV1_1", etc.).

Table B-6 - vacmSecurityToGroupTable

Column Name (* = Part of Index)	First Row Column Value	Second Row Column Value
* vacmSecurityModel	SNMPV1 (1)	SNMPV2c (2)
* vacmSecurityName	"@eRouterconfig_n"	"@eRouterconfig_n"
vacmGroupName	"@eRouterconfigV1_n"	"@eRouterconfigV2_n"
vacmSecurityToGroupStorageType	volatile (2)	volatile (2)
vacmSecurityToGroupStatus	active (1)	active (1)

B.1.3.1.5 vacmAccessTable

The vacmAccessTable is defined in the "Definitions" section of [RFC 3415].

The eRouter MUST create two rows in vacmAccessTable for each SNMPv1v2c Coexistence Configuration encoding as follows:

- The eRouter sets the value of vacmGroupName to "@eRouterconfigV1_n" for the first row and "@eRouterconfigV2_n" for the second row where 'n' is a sequential number starting at 0 for each SNMPv1v2c Coexistence Configuration encoding processed (e.g., "@eRouterconfigV1_0", "@eRouterconfigV1_1", etc.);

  In case the eRouter does not support the SNMPv3 Access View Name encoding in Annex B.3, the eRouter MUST use the default view defined in Annex B.1.2.1 and ignore the Sub-TLV SNMPv1v2c Access View Name.

Table B-7 - vacmAccessTable

Column Name (* = Part of Index)	Column Value	Column Value
* vacmGroupName	"@eRouterconfigV1_n"	"@eRouterconfigV2_n"
* vacmAccessContextPrefix	<zero-length string>	<zero-length string>
* vacmAccessSecurityModel	SNMPV1 (1)	SNMPV2c (2)
* vacmAccessSecurityLevel	noAuthNoPriv (1)	noAuthNoPriv (1)
vacmAccessContextMatch	exact (1)	exact (1)
vacmAccessReadViewName	Set < *AccessViewName*> *or eRouterManagerView*	Set < *AccessViewName*> *or eRouterManagerView*
vacmAccessWriteViewName	When <*AccessViewType*> == '2' Set < *AccessViewName*> *or eRouterManagerView* When <*AccessViewType*> != '2' Set <Zero-length OCTET STRING>	When <*AccessViewType*> == '2' Set < *AccessViewName* *or eRouterManagerView* When <*AccessViewType*> != '2' Set <Zero-length OCTET STRING>
vacmAccessNotifyViewName	<Zero-length OCTET STRING>	<Zero-length OCTET STRING>
vacmAccessStorageType	volatile (2)	volatile (2)
vacmAccessStatus	active (1)	active (1)

B.1.3.2 ***Mapping SNMPv3 Access View Configuration***

If SNMPv3 is supported by the eRouter, the SNMPv3 Access View Configuration encoding is used to configure the vacmViewTreeFamilyTable.

Table B-8 provides a Variable Name as a short-hand reference to be used in the SNMPv3 tables defined in the sub-sections below for each of the SNMPv3 Access View Configuration encodings. The table also defines the mapping between each of the SNMPv3 Coexistence Configuration encodings and the associated SNMP MIB objects.

Table B-8 - SNMPv3 Access View Configuration Encoding

Encodings	Variable Name	Associated MIB Object [RFC 3415]
SNMPv3 Access View Name	*AccessViewName*	vacmViewTreeFamilyViewName
SNMPv3 Access View Subtree	*AccessViewSubTree*	vacmViewTreeFamilySubtree
SNMPv3 Access View Mask	*AccessViewMask*	vacmViewTreeFamilyMask
SNMPv3 Access View Type	*AccessViewType*	vacmViewTreeFamilyType

The eRouter is not required to verify the consistency across tables.

Table B-9 describes the eRouter procedures to populate the vacmViewTreeFamilyTable to conform to the "SNMP Management Framework Message Processing and Access Control Subsystems" [RFC 3412].

When configuring entries in these SNMPv3 tables:

- One entry is created for each SNMPv3 Access View Configuration encoding. Some Access Views may have a number of included/excluded OID branches. Only Access View Name will be common for all these OID branches. To support such type of Access View, multiple SNMPv3 Access View Configuration encodings need to be defined.

B.1.3.2.1 vacmViewTreeFamilyTable

The vacmViewTreeFamilyTable is defined in the "Definitions" section of [RFC 3415].

If the SNMPv3 Access View Configuration encoding is supported by the eRouter, then the eRouter MUST:

- Create one row in vacmViewTreeFamilyTable for each SNMPv3 Access View Configuration TLV;
- Reject the configuration if two or more SNMPv3 Access View Configuration encodings have identical index components (*AccessViewName* and *AccessViewSubTree*);
- Set the object vacmViewTreeFamilySubtree to 1.3.6 when no sub-TLV SNMPv3 Access View Subtree is defined;
- Set the object vacmViewTreeFamilyMask to the default zero-length string when no sub-TLV SNMPv3 Access View Mask is defined;
- Set the object vacmViewTreeFamilyType to the default value 1 (included) when no sub-TLV SNMPv3 Access View Type is defined.

Table B-9 - vacmViewTreeFamilyTable

Column Name (* = Part of Index)	Column Value
* vacmViewTreeFamilyViewName	<*AccessViewName*>
* vacmViewTreeFamilySubtree	<*AccessViewSubTree*>
vacmViewTreeFamilyMask	<*AccessViewMask*>
vacmViewTreeFamilyType	<*AccessViewType*>
vacmViewTreeFamilyStorageType	volatile (2)
vacmViewTreeFamilyStatus	active (1)

B.2 eCM Proxy mechanism for configuration of eRouter

The eRouter configuration encodings are encapsulated in the 'eCM Config File Encapsulation' encoding defined in [eDOCSIS]. The eCM receives the configuration file and parses its contents. The encodings in the eCM configuration file encapsulated in Type 202 are for exclusive use of the eRouter, and these TLVs are transferred from the eCM to the eRouter in a vendor specific manner. After the eCM successfully completes registration, the eRouter uses these encapsulated TLVs for initialization.

The eRouter initializes per the 'eRouter Operation Mode' encoding, encapsulated under the TLV 202 in the eCMs configuration file. During the eRouter initialization process, the eCM reports the eRouter state with the Flow Step information and status in the esafeProvisioningStatusTable [eDOCSIS].

The eCM configuration download process includes certain security aspects; e.g., EAE and secure download which provide for confidentiality and authenticity of the information contained in the CM configuration file as defined in [MULPI] and [SECv3.0].

B.3 eRouter Configuration Encodings

This section defines the encodings required for eRouter configuration and how those are processed by the eRouter.

B.3.1 eRouter TLV Processing

The eRouter MUST disregard encodings that are not defined in this section.

The following sub-sections provide definitions of Configuration Encodings that are valid for eRouter use. The eRouter MUST reject invalid eRouter Configuration Encodings. When the eRouter Configuration Encodings are rejected, the eRouter MUST operate in 'Disabled Mode' per Section 6, ERouter Initialization.

The eRouter MUST reject the eRouter Configuration Encoding if that encoding results in an entry in the SNMP table that cannot be created because of a conflict with an existing entry.

B.3.2 eRouter Initialization Mode Encoding

This encoding defines the eRouter initialization mode (Section 6) configured by the Operator.

A valid eRouter Initialization Mode Encoding contains exactly one instance of this TLV.

Type	Length	Value
1	1	0: Disabled
		1: IPv4 Protocol Enabled
		2: IPv6 Protocol Enabled
		3: Dual IP Protocol Enabled
		4-255: Invalid

B.3.3 SNMPv1v2c Coexistence Configuration

This encoding specifies the SNMPv1v2c Coexistence Access Control configuration for the eRouter. This encoding creates entries in the SNMPv3 framework tables as specified in Annex B.1.3.1 above.

A valid SNMPv1v2c Coexistence Configuration (Type 53) encoding contains the SNMPv1v2c Community Name and one or more instance(s) of SNMPv1v2c Transport Address Access. A valid SNMPv1v2c Coexistence

Configuration (Type 53) encoding may also contain the SNMPv1v2c Access View Type and the SNMPv1v2c Access View Name.

The eRouter does not make persistent entries in the SNMP framework table.

The eRouter MUST support a minimum of 5 SNMPv1v2c Coexistence Configuration encodings.

Type	Length	Value
53	N	Composite

B.3.3.1 *SNMPv1v2c Community Name*

This sub-TLV specifies the Community Name (community string) used in SNMP requests to the eRouter.

Type	Length	Value
53.1	1..32	Text

B.3.3.2 *SNMPv1v2c Transport Address Access*

This sub-TLV specifies the Transport Address and Transport Address Mask pair used by the eRouter to grant access to the SNMP entity querying the eRouter.

Type	Length	Value
53.2	N	Variable

A valid SNMPv1v2c Transport Address Access encoding contains one instance of SNMPv1v2c Transport Address and may contain one instance of SNMPv1v2c Transport Address Mask.

The eRouter accepts one or more instances of sub-TLV 53.2 SNMPv1v2c Transport Address Access within a TLV 53.

B.3.3.2.1 SNMPv1v2c Transport Address

This sub-TLV specifies the Transport Address to use in conjunction with the Transport Address Mask used by the eRouter to grant access to the SNMP entity querying the eRouter.

Type	Length	Value
53.2.1	6 or 18	Transport Address

Transport addresses are 6 or 18 bytes in length for IPv4 and IPv6 type addresses respectively.

B.3.3.2.2 SNMPv1v2c Transport Address Mask

This sub-TLV specifies the Transport Address Mask to use in conjunction with the Transport Address used by the eRouter to grant access to the SNMP entity querying the eRouter. This sub-TLV is optional.

Type	Length	Value
53.2.2	6 or 18	Transport Address Mask

Transport addresses are 6 or 18 bytes in length for IPv4 and IPv6 type addresses respectively.

B.3.3.3 *SNMPv1v2c Access View Type*

The SNMPv1v2c Access View Type encoding specifies the type of access to grant to the community name specified in the SNMPv1v2c Community Name encoding. This TLV is optional. If this TLV is not present, the eRouter MUST set the value of the SNMPv1v2c Access View Type to Read-Only.

Type	Length	Value
53.3	1	1: Read-only 2: Read-write

B.3.3.4 *SNMPv1v2c Access View Name*

This sub-TLV specifies the name of the view that provides the access indicated in the SNMPv1v2c Access View Type. This sub-TLV is optional.

Type	Length	Value
53.4	1.32	String

B.3.4 SNMPv3 Access View Configuration

This encoding specifies the SNMPv3 Simplified Access View configuration of the eRouter. This TLV creates entries in SNMPv3 tables.

The eRouter supports SNMPv3 Access View Configuration encoding only if the eRouter supports SNMPv3.

A valid SNMPv3 Access View Configuration encoding contains one instance of SNMPv3 Access View Name. The eRouter does not make persistent entries in the SNMP framework table.

The eRouter MUST reject the eRouter Configuration Encoding if an eRouter created entry in an SNMP table is rejected due reaching the limit in the number of entries supported for that table.

Type	Length	Value
54	N	Composite

B.3.4.1 *SNMPv3 Access View Name*

This encoding specifies the administrative name of the view defined by the SNMPv3 Access View Configuration.

Type	Length	Value
54.1	1..32	Text

B.3.4.2 *SNMPv3 Access View Subtree*

This encoding specifies an ASN.1 formatted object identifier (OID) that represents the filter sub-tree included in the SNMPv3 Access View Configuration encoding.

A valid SNMPv3 Access View Subtree encoding starts with the ASN.1 Universal type 6 (OID) byte, followed by the ASN.1 length field, and then followed by the ASN.1 encoded object identifier components. For example, the sub-tree 1.3.6 is encoded as 0x06 0x03 0x01 0x03 0x06.

If this encoding is not included under the SNMPv3 Access View Name encoding, the eRouter MUST use the default OID sub-tree of 1.3.6.

Type	Length	Value
54.2	N	OID

B.3.4.3 *SNMPv3 Access View Mask*

This sub-TLV specifies the bit mask to apply to the Access View Subtree of the Access View TLV.

Type	Length	Value
54.3	0..16	Bits

This sub-TLV is optional. If this sub-TLV is not present, the eRouter MUST assign a zero-length string to SNMPv3 Access View Mask.

B.3.4.4 *SNMPv3 Access View Type*

This sub-TLV specifies the inclusion or exclusion of the sub-tree indicated by SNMPv3 Access View Subtree. The value of 1 indicates that the sub-tree of SNMPv3 Access View SubTree is included in the Access View. The value of 2 indicates that the sub-tree of SNMPv3 Access View Sub Tree is excluded from the Access View.

Type	Length	Value
54.4	1	1: included 2: excluded

This sub-TLV is optional. If this sub-TLV is not present, the eRouter MUST assign the value 'included' to SNMPv3 Access View Type.

B.3.5 Vendor Specific Information

The Vendor Specific Information encoding is used to extend the capabilities of the eRouter specification, through the use of vendor-specific features. A valid Vendor Specific Information encoding contains only one Vendor ID field (Annex B.3.5.1) to indicate that the settings apply to a specific vendor device.

The eRouter MUST ignore a Vendor Specific Information encoding that includes a Vendor ID different to the one of the eRouter.

Type	Length	Value
43	N	Variable

B.3.5.1 *Vendor ID Encoding*

The Vendor ID encoding contains the vendor identification specified by the three-byte vendor-specific Organization Unique Identifier of the eRouter's MAC addresses.

The Vendor ID 0xFFFFFF is reserved.

Type	Length	Value
43.8	3	OUI

Appendix I Acknowledgements (Informative)

On behalf of the cable industry and our member companies, CableLabs would like to thank the following individuals for their contributions to the development of this specification.

Contributor	Company Affiliation
Ben Bekele	Cox
Amol Bhagwat	CableLabs
Ralph Brown	CableLabs
John Brzozowski	Comcast
Eduardo Cardona	CableLabs
Margo Dolas	Broadcom
Chris Donley	CableLabs
Ralph Droms	Cisco Systems
Alain Durand	Comcast
Toerless Eckert	Cisco Systems
Kirk Erichsen	Time Warner Cable
Doc Evans	ARRIS
Roger Fish	Broadcom
Deepak Kharbanda	CableLabs
Michelle Kuska	CableLabs
Diego Mazzola	Texas Instruments
John McQueen	Broadcom
Jean-Francois Mule	CableLabs
Harsh Parandekar	Cisco Systems
Michael Patrick	Motorola
Saifur Rahman	Comcast
Lakshmi Raman	CableLabs
Ryan Ross	Juniper
Matt Schmitt	CableLabs
Ron da Silva	Time Warner Cable
Madhu Sudan	Cisco Systems
Dan Torbet	ARRIS
Greg White	CableLabs

We would particularly like to thank Ralph Droms (Cisco) and Doc Evans (ARRIS) for their extensive contributions to the specification and also for defining the eRouter Provisioning details for DHCPv4 and DHCPv6. We would like to thank Ryan Ross (Juniper) for his detailed work on the eRouter NAPT and IPv6 data-forwarding behavior. We would like to thank Margo Dolas (Broadcom) for her work in numerous areas of the specification, including the eRouter IPv4 data forwarding functionality. We would like to thank Diego Mazzola and Deepak Kharbanda for defining the eRouter IP Multicast functionality. We would like to thank John McQueen (Broadcom) for helping define the eRouter DHCPv4 server behavior. We would like to thank Dan Torbet (ARRIS), Chris Donley (CableLabs), and Kirk Erichsen (Time Warner Cable) for their work on developing the QoS functionality. We thank Deepak Kharbanda (CableLabs) who led the IPv6 Focus Team that developed this specification. And finally, many thanks go out to all the active members of the IPv6 Focus Team who contributed to this specification.

Appendix II Revision History

II.1 The following ECN was incorporated into CM-SP-eRouter-I02-070223:

ECN Identifier	Accepted Date	Title of EC
eRouter-N-06.0352-1	1/10/2007	eRouter Multicast Examples

II.2 The following ECN was incorporated into CM-SP-eRouter-I03-070518:

ECN Identifier	Accepted Date	Title of EC
eRouter-N-06.0396-1	3/14/2007	IPv6 Provision of CPE Devices Clarifications

II.3 The following ECN was incorporated into CM-SP-eRouter-I04-100611:

ECN Identifier	Accepted Date	Title of EC
eRouter-N-09-0877-2	12/30/2009	IPv6 Router Update

II.4 The following ECN was incorporated into CM-SP-eRouter-I05-110210:

ECN Identifier	Accepted Date	Title of EC
eRouter-N-10.0974-2	01/05/2011	eRouter Lease Renewal for IPv6 Prefix stability

II.5 The following ECN was incorporated into CM-SP-eRouter-I06-110623:

ECN Identifier	Accepted Date	Title of EC
eRouter-N-11.0991-1	05/04/2011	Clarification to ICMP Destination Unreachable