Internet Engineering Task Force (IETF)
Request for Comments: 5790
Category: Standards Track
ISSN: 2070-1721

H. Liu
W. Cao
Huawei Technologies
H. Asaeda
Keio University

February 2010

Lightweight Internet Group Management Protocol Version 3 (IGMPv3) and Multicast Listener Discovery Version 2 (MLDv2) Protocols

Abstract

This document describes lightweight IGMPv3 and MLDv2 protocols (LW-IGMPv3 and LW-MLDv2), which simplify the standard (full) versions of IGMPv3 and MLDv2. The interoperability with the full versions and the previous versions of IGMP and MLD is also taken into account.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc5790.

Table of Contents

1. Introduction

IGMP version 3 [2] and MLD version 2 [3] implement source filtering capabilities that are not supported by their earlier versions, IGMPv1 [4], IGMPv2 [5], and MLDv1 [6]. An IGMPv3- or MLDv2-capable host can tell its upstream router which group it would like to join by specifying which sources it does or does not intend to receive multicast traffic from. IGMPv3 and MLDv2 add the capability for a multicast router to learn sources that are of interest or that are not of interest for a particular multicast address. This information is used during forwarding of multicast data packets.

INCLUDE and EXCLUDE filter-modes are introduced to support the source filtering function. If a host wants to receive from specific sources, it sends an IGMPv3 or MLDv2 report with filter-mode set to INCLUDE. If the host does not want to receive from some sources, it sends a report with filter-mode set to EXCLUDE. A source-list for the given sources shall be included in the Report message.

INCLUDE and EXCLUDE filter-modes are also defined in a multicast router to process the IGMPv3 or MLDv2 reports. When a multicast router receives the Report messages from its downstream hosts, it forwards the corresponding multicast traffic by managing requested group and source addresses. Group timers and source timers are used to maintain the forwarding state of desired groups and sources under certain filter-modes. When a group report arrives or a certain timer expires, a multicast router may update the desired or undesired source-lists, reset related timer values, change filter-mode, or trigger group queries. With all of the above factors correlating with each other, the determination rules become relatively complex, as the interface states could be frequently changed.

The multicast filter-mode improves the ability of the multicast receiver to express its desires. It is useful to support Source-Specific Multicast (SSM) [7] by specifying interesting source addresses with INCLUDE mode. However, practical applications do not use EXCLUDE mode to block sources very often, because a user or application usually wants to specify desired source addresses, not undesired source addresses. Even if a user explicitly refuses traffic from some sources in a group, when other users in the same shared network have an interest in these sources, the corresponding multicast traffic will still be forwarded to the network. It is generally unnecessary to support the filtering function that blocks sources.

This document proposes simplified versions of IGMPv3 and MLDv2, named Lightweight IGMPv3 and Lightweight MLDv2 (or LW-IGMPv3 and LW-MLDv2). LW-IGMPv3 and LW-MLDv2 are subsets of the standard IGMPv3 and MLDv2.

They support both Any-Source Multicast (ASM) and SSM communications without a filtering function that blocks sources. Not only are they compatible with the standard IGMPv3 and MLDv2, but also the protocol operations made by hosts and routers (or switches performing IGMPv3/ MLDv2 snooping) are simplified to reduce the complicated operations. Since LW-IGMPv3 and LW-MLDv2 are fully compatible with IGMPv3 and MLDv2, hosts or routers that have implemented the full version do not need to implement or modify anything to cooperate with LW-IGMPv3/ LW-MLDv2 hosts or routers.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT","SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [1].

In addition, the following terms are used in this document.

(*,G) join:
An operation triggered by a host that wants to join a group G. In this case, the host receives from all sources sending to group G. This is typical in ASM communication.

(S,G) join:
An operation triggered by a host that wants to join a group G, specifying a desired source S. In this case, the host receives traffic only from source S sending to group G.

INCLUDE (S,G) join:
An operation triggered by a host that wants to join a group G under INCLUDE filter-mode, specifying a desired source S. Same meaning as (S,G) join.

EXCLUDE (*,G) join:
An operation triggered by a host that wants to join a group G under EXCLUDE filter-mode. Same meaning as (*,G) join.

EXCLUDE (S,G) join:
An operation triggered by a host that wants to join a group G under EXCLUDE filter-mode, specifying an undesired source S. This operation is not supported by LW-IGMPv3/LW-MLDv2.

3. Simplification Method Overview

The principle is to simplify the host’s and router’s behavior as much as possible to improve efficiency, while guaranteeing interoperability with the full versions, and introducing no side effects on applications.

For convenience, this document mainly discusses IGMPv3, since MLDv2 inherits the same source filtering mechanism, but this document additionally shows MLDv2's unique specifications when needed.

3.1. Behavior of Group Members

LW-IGMPv3 inherits the service interface model of IGMPv3.

```
IPMulticastListen ( socket, interface, multicast-address,
                    filter-mode, source-list )
```

In the lightweight protocol, INCLUDE mode on the host part has the same usage as the full version for INCLUDE (S,G) join, while EXCLUDE mode on the host part is preserved only for excluding null source-lists, which denotes a (*,G) join as used by IGMPv2/IGMPv1/MLDv1. The detailed host operation of LW-IGMPv3/LW-MLDv2 is described in Section 4.

3.2. Behavior of Multicast Routers

In IGMPv3, router filter-mode is defined to optimize the state description of a group membership [2]. As a rule, once a member report is in EXCLUDE mode, the router filter-mode for the group will be set to EXCLUDE. When all systems cease sending EXCLUDE mode reports, the filter-mode for that group may transit back to INCLUDE mode. The group timer is used to identify such a transition.

In LW-IGMPv3, hosts primarily send INCLUDE requests, and also can request an EXCLUDE (*,G) join, which can be interpreted by the router as a request to include all sources. Without the more general form of EXCLUDE requests, it is unnecessary for the router to maintain the EXCLUDE filter-mode, and the state model for multicast routers can be simplified as:

```
(multicast address, group timer, (source records))
```

Here a group timer is kept to represent a (*,G) join. Its basic behavior is: when a router receives a (*,G) join, it will set its group timer and keep the source-list for sources specified in the previously received source records. When the group timer expires, the router may change to reception of the listed sources only. The definition of the source record is the same as in the full version.

The elimination of the filter-mode will greatly simplify the router behavior. The details of router operation are described in Section 5.

4. LW-IGMPv3 Protocol for Group Members

4.1. Query and Report Messages

LW-IGMPv3 uses the same two sets of messages, Query and Report messages, as the full version protocols. There is no difference between the definition and usage of the Query message. But the report types in lightweight protocols are reduced because an operation that triggers EXCLUDE (S,G) join is omitted.

There are three Group Record Types defined in the full IGMPv3: the Current-State Record denoted by MODE_IS_INCLUDE (referred to as IS_IN) or MODE_IS_EXCLUDE (IS_EX), the Filter-Mode-Change Record denoted by CHANGE_TO_INCLUDE_MODE (TO_IN) or CHANGE_TO_EXCLUDE_MODE (TO_EX), and the Source-List-Change Record denoted by ALLOW_NEW_SOURCES (ALLOW) or BLOCK_OLD_SOURCES (BLOCK). LW-IGMPv3 inherits the actions on change of interface state and on reception of a query, but the IS_IN and IS_EX record types are eliminated and Current-State Records are replaced by other records. The following sections explain the details.

4.2. Action on Change of Interface State

When the state of an interface of a group member host is changed, a State-Change Report for that interface is immediately transmitted from that interface. The type and contents of the Group Record(s) in that report are determined by comparing the filter-mode and source-list for the affected multicast address before and after the change. While the requirements for the computation are the same as for the full version, in a lightweight version host the interface state change rules are simplified due to the reduction of message types. The contents of the new transmitted report are calculated as follows (Group Record Types are described in Section 4.4):

Old State	New State	State-Change Report Sent
INCLUDE (A)	INCLUDE (B)	ALLOW(B-A), BLOCK(A-B)
INCLUDE (A)	EXCLUDE ({})	TO_EX({})
INCLUDE ({})	EXCLUDE ({})	TO_EX({})
EXCLUDE ({})	INCLUDE (B)	TO_IN(B)

As in the full version, to cover the possibility of the State-Change Report being missed by one or more multicast routers, it is retransmitted [Robustness Variable]-1 more times, at intervals chosen at random from the range (0, [Unsolicited Report Interval]). (These values are defined in [2][3].)

4.3. Action on Reception of a Query

As in the full version, when a lightweight version host receives a query, it does not respond immediately. Instead, it delays its response by a random amount of time, bounded by the Max Resp Time value derived from the Max Resp Code in the received Query message [2][3]. The system may receive a variety of queries on different interfaces and of different kinds (e.g., General Queries, Group-Specific Queries, and Group-and-Source-Specific Queries), each of which may require its own delayed response.

Before scheduling a response to a query, the system must first consider previously scheduled pending responses and in many cases schedule a combined response. Therefore, the lightweight version host must be able to maintain the following state:

- A timer per interface for scheduling responses to General Queries.
- A per-group and interface timer for scheduling responses to Group-Specific and Group-and-Source-Specific Queries.
- A per-group and interface list of sources to be reported in the response to a Group-and-Source-Specific Query.

LW-IGMPv3 inherits the full version's rules that are used to determine if a report needs to be scheduled. The difference is regarding the simplification of EXCLUDE filter-mode and the type of report as detailed in Section 4.4.

4.4. LW-IGMPv3 Group Record Types

Among the Group Record Types defined in the full IGMPv3, several record types are not used in LW-IGMPv3 as some of the processes related to the filter-mode change to the EXCLUDE mode are eliminated and some of the Report messages are converged into a record having a null source address list. All of the record types of Report messages used by the full and lightweight version protocols are shown as follows:

IGMPv3	LW-IGMPv3	Comments
IS_EX({})	TO_EX({})	Query response for (*,G) join
IS_EX(x)	N/A	Query response for EXCLUDE (x,G) join
IS_IN(x)	ALLOW(x)	Query response for INCLUDE (x,G) join
ALLOW(x)	ALLOW(x)	INCLUDE (x,G) join
BLOCK(x)	BLOCK(x)	INCLUDE (x,G) leave
TO_IN(x)	TO_IN(x)	Change to INCLUDE (x,G) join
TO_IN({})	TO_IN({})	(*,G) leave
TO_EX(x)	N/A	Change to EXCLUDE (x,G) join
TO_EX({})	TO_EX({})	(*,G) join

where "x" represents a non-null source address list and "({})" represents a null source address list. For instance, IS_EX({}) means a report whose record type is IS_EX with a null source address list. "N/A" represents not applicable (or no use) because the corresponding operation should not occur in the lightweight version protocols.

LW-IGMPv3 does not use EXCLUDE filter-mode with a non-null source address list. A multicast router creates the same state when it receives a Report message containing either IS_EX({}) or TO_EX({}) record types. Therefore, LW-IGMPv3 integrates the IS_EX({}) operation with the TO_EX({}) operation.

When an LW-IGMPv3 host needs to make a query response for the state of INCLUDE (x,G) join, it makes a response whose message type is expressed with ALLOW(x), instead of using the IS_IN record type. Because the router’s processing of the two messages is exactly the same, the IS_IN(x) type is eliminated for simplification.

An LW-IGMPv3 host does not use EXCLUDE mode, while TO_IN and TO_EX records are used for example in the following situation: the host first launches an application (AP1) that requests INCLUDE (x,G) join, and sends ALLOW(x). Then the host launches another application (AP2) that joins (*,G), and it sends TO_EX({}). In this condition, when AP2 terminates but AP1 keeps working on the lightweight version host, the host sends a report with TO_IN(x) record type for [Robustness Variable] times.

Although an LW-IGMPv3 host adopts the four message types (ALLOW, BLOCK, TO_IN, and TO_EX) for simplification, using IS_EX({}) and IS_IN(x) (respectively, instead of TO_EX({}) and ALLOW(x)) in response to queries is not inhibited. This will not introduce the interoperation problem because the router process is, respectively, the same for the mentioned two message set, as long as the router implementation follows the rules given by full IGMPv3.

5. LW-IGMPv3 Protocol for Multicast Routers

The major difference between the full and lightweight version protocols on the router part is that in the lightweight version filter-mode is discarded and the function of the group timer is redefined. The states maintained by the lightweight router are reduced and the protocol operation is greatly simplified.

5.1. Group Timers and Source Timers in the Lightweight Version

In lightweight and full IGMPv3 routers, a source timer is kept for each source record and it is updated when the source is present in a received report. It indicates the validity of the source and needs to be referred to when the router takes its forwarding decision.

The group timer being used in the full version of IGMPv3 for transitioning the router's filter-mode from EXCLUDE to INCLUDE is redefined in the lightweight protocols to identify the non-source-specific receiving state maintained for (*,G) join. Once a group record of TO_EX({}) is received, the group timer is set to represent this (*,G) group join. The expiration of the group timer indicates that there are no more listeners on the attached network for this (*,G) group. Then if at this moment there are unexpired sources (whose source timers are greater than zero), the router will change to receiving traffic for those sources only. The role of the group timer can be summarized as follows:

Group Timer Value	Actions/Comments
G_Timer > 0	All members in this group.
G_Timer == 0	No more listeners to this (*,G) group. If all source timers have expired, then delete group record. If there are still source record timers running, use those source records with running timers as the source record state.

The operation related to the group and source timers has some differences compared to the full IGMPv3. In the full version, if a source timer expires under the EXCLUDE router filter-mode, its corresponding source record is not deleted until the group timer expires for indicating undesired sources. In the lightweight version, since there is no need to keep such records for blocking specific sources, if a source timer expires, its source record should be deleted immediately, not waiting for the time-out of the group timer.

5.2. Source-Specific Forwarding Rules

A full version multicast router needs to consult IGMPv3 state information when it makes decisions on forwarding a datagram from a source, based on the router filter-mode and source timer. In LW-IGMPv3, because of the absence of the router filter-mode, the group timer and source timer could be used for such decisions. The forwarding suggestion made by LW-IGMPv3 to the routing protocols is summarized as follows:

Group Timer	Source Timer	Action
G_Timer == 0	S_Timer > 0	Suggest forwarding traffic from source
G_Timer == 0	S_Timer == 0	Suggest stopping forwarding traffic from source and remove source record. If there are no more source records for the group, delete group record
G_Timer == 0	No Source Elements	Suggest not forwarding traffic from source
G_Timer > 0	S_Timer >= 0	Suggest forwarding traffic from source
G_Timer > 0	No Source Elements	Suggest forwarding traffic from source

5.3. Reception of Current-State Records

When receiving Current-State Records, the LW-IGMPv3 router resets its group or source timers and updates its source-list within the group. For source-specific group reception state (when G_Timer == 0 and

S_Timer > 0), the source-list contains sources whose traffic will be forwarded by the router, while in non-source-specific group reception (when G_Timer > 0), the source-list remembers the valid sources to receive traffic from after toggling to source-specific reception state.

Although the LW-IGMPv3 host only sends a subset of the messages of the full version, the LW-IGMPv3 router should be able to process as many messages as possible to be compatible with the full version host. Note that if the report type is IS_EX(x) with a non-empty source-list, the router will treat it as the same type of report with an empty source-list. The following table describes the action taken by a multicast router after receiving Current-State Records. The notations have the same meaning as those in the full IGMPv3 protocol.

Group Timer	Old Source-List	Report Rec'd	New Source-List	Actions
G_Timer == 0	A	IS_IN(B)	A+B	(B)=GMI
G_Timer == 0	A	IS_EX({})	A	G_Timer=GMI
G_Timer > 0	A	IS_IN(B)	A+B	(B)=GMI
G_Timer > 0	A	IS_EX({})	A	G_Timer=GMI

The above table could be further simplified since the processes are exactly the same for the two values of the G_Timer:

Old Source-List	Report Rec'd	New Source-List	Actions
A	IS_IN(B)	A+B	(B)=GMI
A	IS_EX({})	A	G_Timer=GMI

Without EXCLUDE filter-mode, a router's process on receiving a Current-State Record is simple: when a router receives an IS_IN report, it appends the reported source addresses to the previous source-list with their source timers set to GMI. Upon receiving an IS_EX({}) report, the router sets the non-source-specific receiving states by resetting the group timer value and keeps the previous source-list without modification.

5.4. Reception of Source-List-Change and Filter-Mode-Change Records

On receiving Source-List-Change and Filter-Mode-Change Records, the LW-IGMPv3 router needs to reset its group and source timers, update its source-list within the group, or trigger group queries. The queries are sent by the router for the sources that are requested to be no longer forwarded to a group. Note that if the report type is TO_EX(x) with a non-empty source-list, the router will treat it as the same type of report with an empty source-list. The table below describes the state change and the actions that should be taken.

Group Timer	Old Source-List	Report Rec'd	New Source-List	Actions
G_Timer == 0	A	ALLOW(B)	A+B	(B)=GMI
G_Timer == 0	A	BLOCK(B)	A	Send Q(G,A*B)
G_Timer == 0	A	TO_IN(B)	A+B	(B)=GMI Send Q(G,A-B)
G_Timer == 0	A	TO_EX({})	A	G_Timer=GMI
G_Timer > 0	A	ALLOW(B)	A+B	(B)=GMI
G_Timer > 0	A	BLOCK(B)	A	Send Q(G,A*B)
G_Timer > 0	A	TO_IN(B)	A+B	(B)=GMI SendQ(G,A-B) Send Q(G)
G_Timer > 0	A	TO_EX({})	A	G_Timer=GMI

The table could be further simplified by merging duplicate lines:

Old Source-List	Report Rec'd	New Source-List	Actions
A	ALLOW(B)	A+B	(B)=GMI
A	BLOCK(B)	A	Send Q(G,A*B)
A	TO_IN(B)	A+B	(B)=GMI Send Q(G,A-B) If G_Timer>0 Send Q(G)
A	TO_EX({})	A	G_Timer=GMI

6. Interoperability

LW-IGMPv3/LW-MLDv2 hosts and routers must interoperate with hosts and routers of the full version [2][3]. Also, LW-IGMPv3/LW-MLDv2 hosts and routers must interoperate gracefully with hosts and routers running IGMPv1/v2 or MLDv1.

6.1. Interoperation with the Full Version of IGMPv3/MLDv2

LW-IGMPv3/LW-MLDv2 do not introduce any change on the message formats of the group Query and Report messages that the full version protocols use.

6.1.1. Behavior of Group Members

An LW-IGMPv3 host's compatibility mode is determined from the Host Compatibility Mode variable, which can be in one of three states: IGMPv1, IGMPv2, or IGMPv3. When a lightweight host behaves on its interface as LW-IGMPv3, its Host Compatibility Mode of that interface is set to IGMPv3, and the host sends a subset of IGMPv3 Report messages, which can be recognized by a multicast router running the full or the lightweight IGMPv3 protocol on the same LAN.

6.1.2. Behavior of Multicast Routers

An LW-IGMPv3 or LW-MLDv2 router does not process directly IS_EX(x) and TO_EX(x) reports that are used by the full version. When an LW-IGMPv3/LW-MLDv2 router receives these Report messages from full version hosts, it MUST translate them internally to IS_EX({}) and TO_EX({}) respectively and behave accordingly.

6.2. Interoperation with IGMPv1/IGMPv2

Since the lightweight protocols can be treated as a parallel version of the full version of IGMPv3/MLDv2, its compatibility principle and method with the older version are generally the same as that of full IGMPv3/MLDv2.

6.2.1. Behavior of Group Members

The Host Compatibility Mode of an interface is set to IGMPv2 and its IGMPv2 Querier Present timer is set to Older Version Querier Present Timeout seconds (defined in [2]) whenever an IGMPv2 General Query is received on that interface. The Host Compatibility Mode of an interface is set to IGMPv1 and its IGMPv1 Querier Present timer is set to Older Version Querier Present Timeout seconds whenever an IGMPv1 Membership Query is received on that interface.

In the presence of older version group members, LW-IGMPv3 hosts may allow its Report message to be suppressed by either an IGMPv1 or IGMPv2 membership report. However, because the transmission of IGMPv1 or v2 packets reduces the capability of the LW-IGMPv3 system, as a potential protection mechanism, the choice to enable or disable the use of backward compatibility may be configurable.

6.2.2. Behavior of Multicast Routers

The behavior of an LW-IGMPv3 router when placed on a network where there are routers that have not been upgraded to IGMPv3 is exactly the same as for a full IGMPv3 router in this situation [2].

A full IGMPv3 router uses Group Compatibility Mode (whose value is either of IGMPv1, IGMPv2, or IGMPv3) per group record to indicate which version of IGMP protocol it applies to the group. This value is set according to the version of the received IGMP reports. When Group Compatibility Mode is IGMPv3, the lightweight router performs the LW-IGMPv3 protocol for that group.

When Group Compatibility Mode is IGMPv2, an LW-IGMPv3 router inherits this compatibility mechanism with the following rules:

IGMP Message	LW-IGMPv3 Equivalent
v2 Report	TO_EX({})
v2 Leave	TO_IN({})

When Group Compatibility Mode is IGMPv1, an LW-IGMPv3 router internally translates the following IGMPv1 and IGMPv2 messages for that group to their LW-IGMPv3 equivalents:

IGMP Message	LW-IGMPv3 Equivalent
v1 Report	TO_EX({})
v2 Report	TO_EX({})

6.3. Interoperation with MLDv1

LW-MLDv2 hosts and routers MUST interoperate with hosts and routers running MLDv1. The method is the same as described in Section 6.2. The difference is that when an LW-MLDv2 router has a MLDv1 listener on its network, it translates the following MLDv1 messages to their LW-MLDv2 equivalents:

MLDv1 Message	LW-MLDv2 Equivalent
Report	TO_EX({})
Done	TO_IN({})

7. Implementation Considerations

The lightweight protocols require no additional procedure for the implementation of the related protocols or systems, e.g., IGMP/MLD snooping, multicast routing protocol, and operation of application sockets, while the processing loads on the switches and routers that run IGMPv3/MLDv2 (snooping) and multicast routing protocols may be greatly decreased.

7.1. Implementation of Source-Specific Multicast

[8] specifies the requirements for the implementation of Source-Specific Multicast (SSM) on IGMPv3/MLDv2 hosts and routers. The lightweight protocol follows the same rules as given in [8] except for the change of the message types due to the simplification.

An LW-IGMPv3/LW-MLDv2 host should not invoke (*,G) join (i.e., TO_EX({})) and (*,G) leave (i.e., TO_IN({})) for applications whose multicast addresses are in the SSM address range. An upstream LW-IGMPv3/LW-MLDv2 router MUST NOT establish forwarding state and MAY log an error on reception of them as described in [7].

7.2. Implementation of Multicast Source Filter (MSF) APIs

[9] defines the following Multicast Source Filter (MSF) APIs: (1) IPv4 Basic MSF APIs, (2) IPv4 Advanced MSF APIs, (3) Protocol-Independent Basic MSF APIs, and (4) Protocol-Independent Advanced MSF APIs.

According to the MSF API definition, an LW-IGMPv3 host should implement either the IPv4 Basic MSF API or the Protocol-Independent Basic MSF API, and an LW-MLDv2 host should implement the Protocol-Independent Basic MSF API. Other APIs, IPv4 Advanced MSF API and Protocol-Independent Advanced MSF API, are optional to implement in an LW-IGMPv3/LW-MLDv2 host.

8. Security Considerations

The security considerations are the same as that of the full version of IGMPv3/MLDv2.

9. Acknowledgements

The authors would like to thank MBONED and MAGMA working group members. Special thanks is given to Marshall Eubanks, Guo Feng, Mark Fine, Alfred Hoenes, Prashant Jhingran, Bharat Joshi, Guo Tao, Wang Wendong, and Gong Xiangyang for their valuable suggestions and comments on this document.

10. References

10.1. Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

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

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

[4] Deering, S., "Host extensions for IP multicasting", STD 5, RFC 1112, August 1989.

[5] Fenner, W., "Internet Group Management Protocol, Version 2", RFC 2236, November 1997.

[6] Deering, S., Fenner, W., and B. Haberman, "Multicast Listener Discovery (MLD) for IPv6", RFC 2710, October 1999.

[7] Holbrook, H. and B. Cain, "Source-Specific Multicast for IP", RFC 4607, August 2006.

[8] Holbrook, H., Cain, B., and B. Haberman, "Using Internet Group Management Protocol Version 3 (IGMPv3) and Multicast Listener Discovery Protocol Version 2 (MLDv2) for Source-Specific Multicast", RFC 4604, August 2006.

10.2. Informative References

[9] Thaler, D., Fenner, B., and B. Quinn, "Socket Interface Extensions for Multicast Source Filters", RFC 3678, January 2004.

Authors' Addresses

Hui Liu
Huawei Technologies Co., Ltd.
Huawei Bld., No.3 Xinxi Rd.
Shang-Di Information Industry Base
Hai-Dian Distinct, Beijing 100085
China

EMail: Liuhui47967@huawei.com

Wei Cao
Huawei Technologies Co., Ltd.
Huawei Bld., No.3 Xinxi Rd.
Shang-Di Information Industry Base
Hai-Dian Distinct, Beijing 100085
China

EMail: caowayne@huawei.com

Hitoshi Asaeda
Keio University
Graduate School of Media and Governance
5322 Endo
Fujisawa, Kanagawa 252-8520
Japan

EMail: asaeda@wide.ad.jp