An Implementation of IPv6 PIM-SSM in Linux Systems

Sangjin Jeong* and Hyoung-Jun Kim**
* Protocol Engineering Canter, Electronics and Telecommunications Research Institute, Korea
Tel : +82-42-860-1877 Fax : +82-42-861-5404 E-mail: sjjeong@etri.re.kr
** Protocol Engineering Canter, Electronics and Telecommunications Research Institute, Korea
Tel : +82-42-860-6576 Fax : +82-42-861-5404 E-mail: khj@etri.re.kr

Abstract: Currently, most IP multicasting applications are implemented based on Any-Source Multicast (ASM) model that supports many to many multicast services. However, it is known that current ASM-based multicast architecture has several deployment problems such as address allocation, lack of access control, and inefficient handling of well-known multicast sources. Source-Specific Multicast (SSM) working group in IETF proposed SSM architecture to overcome the weaknesses of ASM architecture. The architecture of SSM is based on one to many multicast services. Also, in order to provide SSM service, Multicast Listener Discovery Version 2 (MLDv2) protocol should be supported. In this paper, we introduce the architecture of SSM protocol and multicast group management protocol. After that, we present the architecture and implementation of IPv6 SSM and MLDv2 protocols in Linux systems.

Keywords: IPv6, Multicast, Source-Specific Multicast (SSM), Multicast Listener Discovery (MLD)

1. INTRODUCTION

IP multicasting architecture proposed by Steve Deering defines many to many multicast service model based on Any-Source Multicast (ASM) [1]. Also, many researches related to multicast have been done on the ASM architecture. However, after deploying multicast services in the network, the complexity and scalability problem of ASM limited the wide deployment of IP multicast services. In the ASM architecture, all multicast capable routers in the multicast trees should maintain state information of each multicast session, so this multicast state information consumes router resources and degrades the performance of unicast routing of the routers. Furthermore, in the ASM architecture, each multicast receiver can make multicast state information in the routers. Thus, malicious users can easily attack the multicast capable routers with Denial-of-Service (DoS) attack and ASM architecture uses Rendezvous Point (RP) to distribute multicast traffic to each receivers but the RP is an easy target of DoS attack. Also, the deployment problems of ASM-based IP multicast such as address allocation, lack of access control, and inefficient handling of well-known multicast sources became a serious issue [2].
Source-Specific Multicast (SSM) working group in IETF proposed SSM architecture to resolve the weakness of ASM architecture and the specification of SSM is almost finalized [3][4]. In SSM architecture, multicast channel is recognized by a pair of multicast traffic sender S and multicast group G, and multicast traffic from a sender to an empty group is dropped in the intermediate routers. Table 1 compares ASM with SSM architecture.

Table 1 Comparison of ASM and SSM

Service model	ASM	SSM
Network abstraction	Group	Channel
Identifier	G	S, G
Receiver operation	Join, Leave	(Un)Subscribe

In order to provide SSM service, Internet Group Management Protocol Version 3 (IGMPv3) is to be supported in IPv4 network and Multicast Listener Discovery Version 2 (MLDv2) protocol should be supported in IPv6 network [5].
In this paper, we introduce the architecture of SSM protocol and multicast group management protocol. After that, we present the architecture and implementation of IPv6 SSM and MLDv2 protocols in Linux systems.

2. OVERVIEW OF PIM-SSM AND MLDV2

In SSM architecture, a multicast receiver joins a SSM channel (S, G) that is identified by multicast source IP address S and multicast group address G. Multicast traffic from a sender is delivered only to multicast receivers that explicitly joined multicast channel (S, G) by MLDv2 protocol. Furthermore, there exists only one multicast sender for each SSM channel. Therefore, SSM architecture supports only one to many multicast service. Fig. 1 shows the architecture of SSM. The details of each model are as follows.

- Address allocation: For IPv4, the address range of 232/8 has been assigned by IANA for SSM. In case of IPv6, [6] has defined an extension to the addressing architecture to allow for unicast prefix-based multicast addresses.
- Session description and channel discovery: An multicast application that wants to receive SSM traffic needs to know multicast sender S and multicast group G to join in advance. The discovery of SSM source and channel is done by looking up web sites that list existing SSM senders.
- SSM aware applications: An application that wants to receive an SSM session must first discover the channel address in use. A receiving application must be able to specify both a source address and a destination address to the network layer protocol module on the end-host.
- IGMPv3/MLDv2 host reporting and querier: In order to use SSM service, an end-host must be able to specify a

channel address, consisting of a source's unicast address and a multicast group address. The ability to specify an SSM channel address C is provided by MLDv2 in IPv6 networks. This protocol supports "source filtering", i.e., the ability of an end-system to express interest in receiving data packets sent only by specific sources. In fact, MLDv2 provides a superset of the capabilities required to realize the SSM service model [3][5].

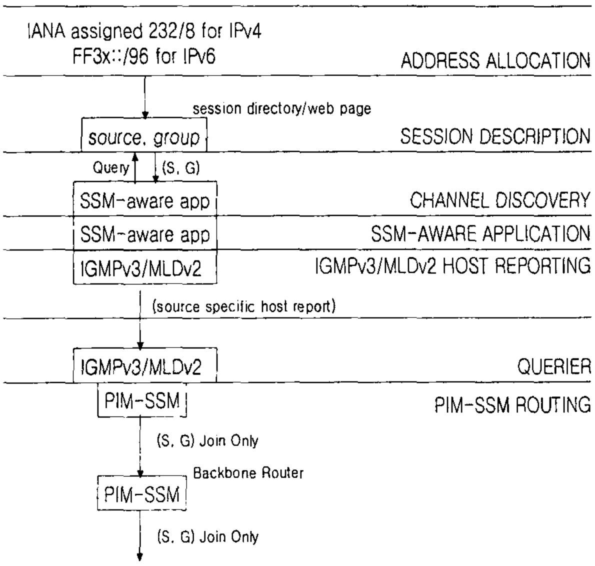

Fig. 1. SSM architecture

When the multicast service subscribed host receives a multicast packet, the host compares the source address of the packet with the source filtering table inside the kernel. Then, the host delivers the packet to the multicast applications only if the source of the packet is allowed to accept. When the host receives MLDv2 query messages from DR, the host reports current multicast group membership state through MLDv2 report messages.

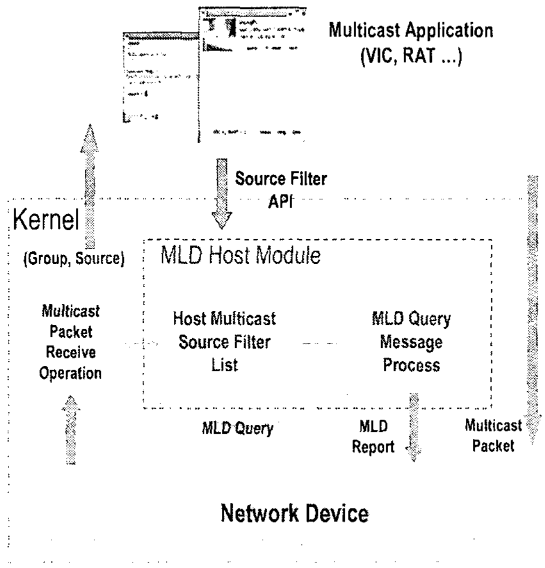

Fig. 2. Architecture of multicast host

3. ARCHITECTURE OF IPV6 PIM-SSM AND MLDV2 IN LINUX SYSTEMS

In order to receive PIMv6-SSM services in Linux systems, users need MLDv2 implemented multicast hosts. Furthermore, all intermediate routers between multicast senders and receivers should support PIMv6-SSM protocol unless multicast tunneling is used. In this section, we present the implementation of IPv6 multicast host that supports MLDv2 protocol. Then, we describe the implementation of IPv6 multicast router and its multicast packet forwarding architecture.

3.1. Architecture of MLDv2 based IPv6 Multicast Host

We introduce the implementation architecture of MLDv2 protocol in Linux system on the kernel version 2.4.18. MLDv2 host gets the request of changing subscribed multicast group from multicast applications through IPv6 socket API [7] and manages multicast source filtering table in Linux kernel. Also, according to the current multicast source filtering table, MLDv2 host responds to the MLDv2 Query sent from Designated Router (DR) to gather source filtering state in each host.

Fig. 2 shows the architecture of multicast host. Users subscribe to multicast group address using multicast applications such as Video Conferencing Tool (VIC) and Robust Audio Tool (RAT). When multicast application joins to the multicast group address, it passes the (S, G) source filtering information to the kernel. The kernel merges source filtering information delivered from applications and manages source filtering table inside the kernel.

3.2. Architecture of IPv6 Multicast Router

In order to provide SSM IPv6 multicast services to users, MLDv2 protocol is to be implemented in the DR. As we introduced in Section 2, IPv6 multicast capable routers have to run MLDv2 protocol to gather multicast membership state of each receiver. In this section, we describe the architecture of MLDv2 daemon for IPv6 multicast router. Then, we present the multicast packet forwarding architecture inside the multicast capable router.

The operation of IPv6 multicast router can be divided into three parts, management of multicast group membership, update routing information, and forwarding of multicast packets. Fig. 3 depicts the overview of IPv6 multicast router. The router has three modules, PIMv6-SSM daemon, MLDv2 daemon, and multicast packet forwarder. PIMv6-SSM daemon receives routing information from multicast neighbor and builds multicast routing table by using routing information and link membership information received from MLDv2 daemon. After making the routing table, the PIMv6-SSM daemon puts the routing table into the Linux kernel through API.

The MLDv2 daemon receives MLDv2 report messages from each multicast receiver and calculates multicast group membership state of each link. After calculating the membership state, MLDv2 daemon delivers the membership information to the PIMv6-SSM daemon through Unix domain socket.

When the multicast router receivers a multicast packet, the multicast packet forwarder looks up the multicast forwarding table and forwards the packet to outgoing interface if the destination address is listed in the table. If the destination address is for multicast control message, the multicast forwarder forwards the packet to PIMv6-SSM daemon or MLDv2 daemon according to the address. The control messages for PIMv6-SSM use FF02::D as a destination address. Also, MLDv2 daemon uses FF02::16 as a destination address of MLDv2 messages [8].

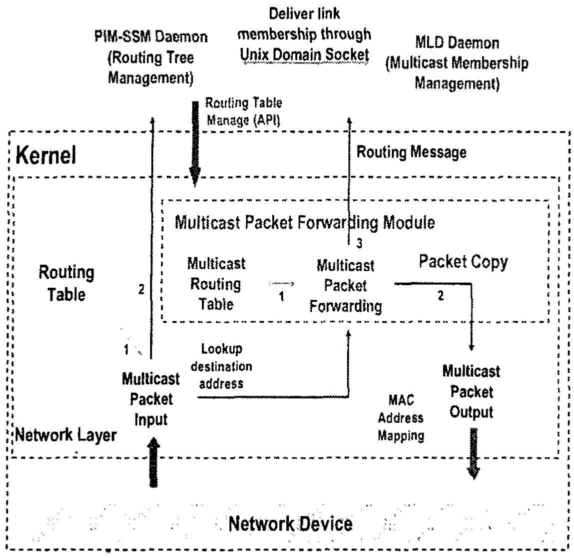

Fig. 3. Architecture of multicast router

3.3. Architecture of IPv6 Multicast Packet Forwarder

In order to provide IPv6 SSM multicast service, the multicast forwarder in Linux kernel should be extended to support to IPv6 multicast packet forwarding. In this section, we present the architecture of IPv6 multicast forwarder. We implemented IPv6 multicast forwarder based on IPv4 multicast forwarder. Also, the multicast forwarding table is defined based on IPv4 multicast forwarding table.

Fig. 4 shows the architecture of multicast packet forwarder. When multicast packet is received in the incoming interface, link layer function, net_rx_action, checks the packet type in MAC address. If the packet type is 0x86DD, namely IPv6, ipv6_rcv function process the received packet. The received packet is delivered to ip6_route_input function and the ip6_route_input function determines the next calling function based on IPv6 multicast forwarding table. If the destination address of received packet is multicast address and the router is working as a multicast capable mode, ip6_route_input_mc function sets ip6_mr_input function to skb→dst→input and ip6_mc_output function to skb→dst→output. Therefore, ip6_rcv_finish function will call ip6_mr_input function that is linked to skb→dst→input.

The ip6_mr_input function looks up the destination address in the multicast forwarding table. If the destination address is found in the table, ip6_mr_forward function forward the received packet. If the received packet is for router itself, i.e. PIMv6-SSM and MLDv2 messages, the packet is delivered to ip6_mc_input function and processed. The ip6_mr_forward function calls ip6mr_queue_xmit function as many times as the number of outgoing interfaces and passes the received packet to it. The ip6mr_queue_xmit function sets ip6_mc_output function to skb→dst→output and send the packet through link layer. Thus, ip6_mc_output function builds MAC frame by using IPv6 multicast destination address and transmits the packet. Therefore, the received packet is forwarded to correct outgoing interface.

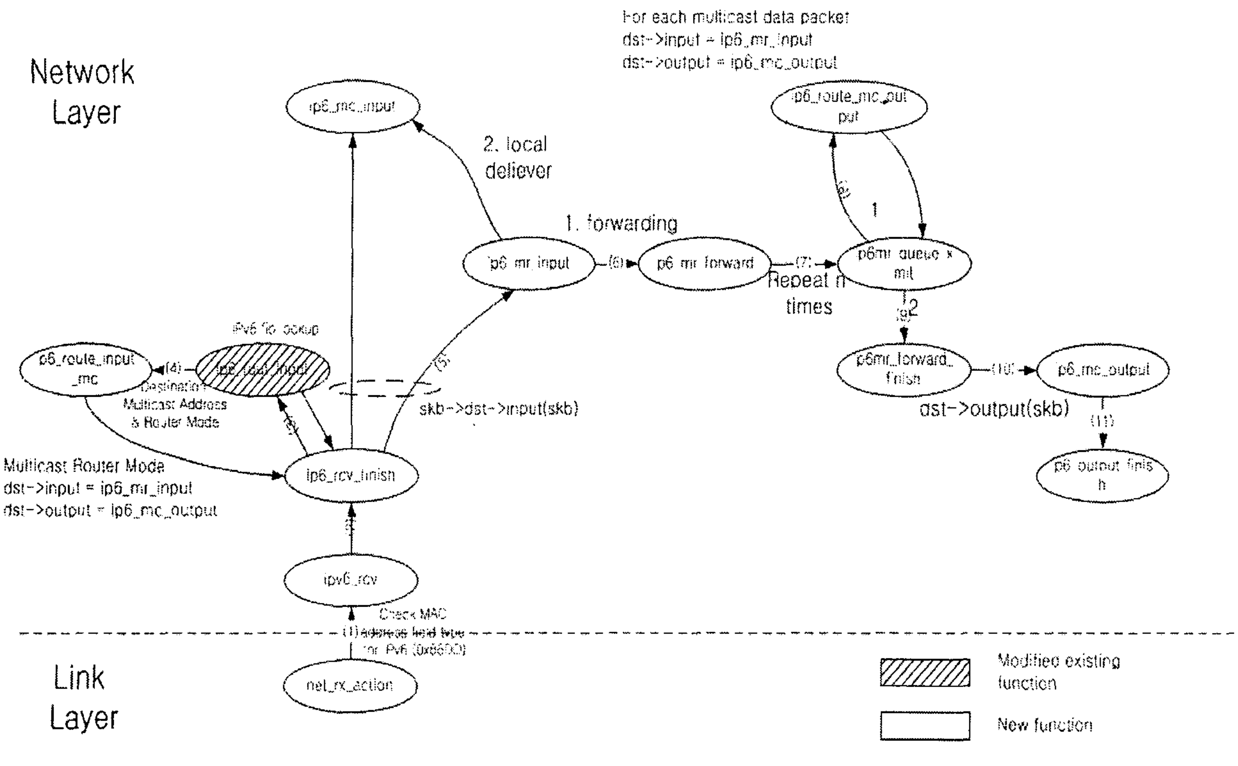

Fig. 4. Architecture of multicast packet forwarder

4. EXPERIMENTAL RESULTS

Fig. 5 shows the testbed to check the operation of IPv6 PIMv6-SSM multicast router and host. All the systems in our testbed are implemented based on Linux kernel 2.4.18. The testbed consists of two multicast capable routers and three multicast hosts. Two of the hosts are work as a multicast source. Both multicast capable routers run PIMv6-SSM routing protocol and MLDv2 multicast group management protocol. Also, both routers work as DR. Both Sender A and Sender B send multicast traffic to multicast group address FF04::10. After that, Receiver C joins the multicast group and performs block/unblock source operation to allow and to block multicast traffic from specific source. Fig. 6 shows block/unblock source operation.

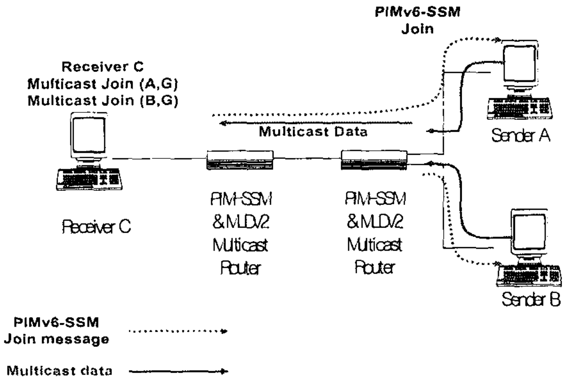

Fig. 5. PIMv6-SSM testbed

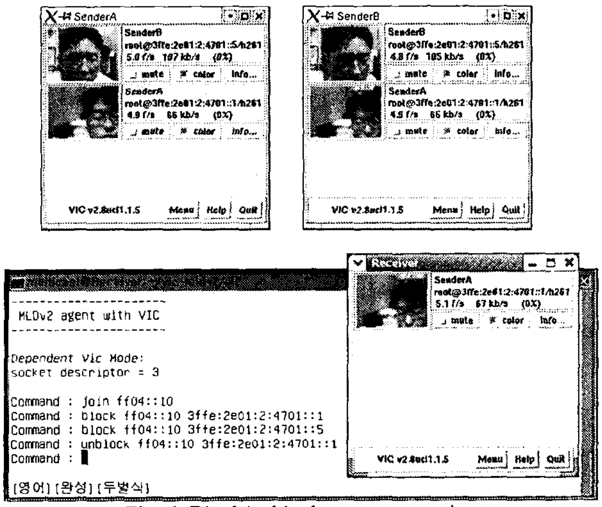

Fig. 6. Block/unblock source operation

5. CONCLUSIONS

In this paper, we introduced the overview of SSM and MLDv2 protocols that are intended for overcoming the weaknesses of conventional ASM architecture. We also, provided the requirements of MLDv2 to provide SSM in IPv6 networks. Then, we presented the architecture and implementation of IPv6 SSM and MLDv2 protocols in Linux systems. Since most of IP multicast services are based on one to many multicast services such as VOD and AOD, SSM service model can be apply to most of conventional multicast services.

References

[1] S. Deering, "Host Extensions for IP Multicasting", IETF RFC 1112, Aug. 1989.

[2] C. Diot, B. Levine, B. Lyles, H. Kassem, and D. Balensiefer, "Deployment Issues for the IP Multicast Service and Architecture", IEEE Network, vol. 14, no. 1, Jan. 2000.

[3] S. Bhattacharyya, et al., "An Overview of Source-Specific Multicast (SSM)", IETF RFC 3569, July 2003.

[4] H. Holbrook, et al., "Source-Specific Multicast for IP", Work in progress, draft-ietf-ssm-arch-05.txt, July 2004.

[5] R. Vida, et al., "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", IETF RFC 3810, June 2004.

[6] B. Haberman, D. Thaler, "Unicast-Prefix-based IPv6 Multicast Addresses", IETF RFC 3306, Aug. 2002.

[7] D. Thaler, et al., "Socket Interface Extensions for Multicast Source Filters", IETF RFC 3678, Jan. 2004.

[8] http://www.iana.org/assignments/ipv6-multicast-addresses