밀집 멀티캐스트 그룹을 위한 도메인 Xcast 방안

정회원 김 병 순*

A Domain Xcast Scheme for a Dense Multicast Group

Byungsoon Kim* *Regular Member*

요 약

이 논문은 Explicit Multicast (Xcast) 방안에서 크기가 큰 밀집 멀티캐스트 그룹과 동적인 그룹 멤버쉽을 지원하는 방안을 제안한다. 이를 위해 도메인 에이전트가 호스트를 대신하여 그룹에 가입하는 계층적 가입 방법을 사용한다. 송신자가 먼저 도메인 에이전트에게 Xcast 패킷을 전송하면, 에이전트가 도메인에 있는 그룹 수신자에게 다시 전송한다. 그리고 호스트는 도메인 에이전트를 통해 언제든지 그룹에 조인하거나 탈퇴할 수 있다. 송신자가 전송하는 Xcast 헤더의 목적지 필드는 도메인 에이전트의 주소를 갖고, 도메인 에이전트가 전송하는 패킷의 목적지에는 도메인 안에 있는 수신자의 주소를 갖는다. 시뮬레이션 결과를 통해 밀집 멀티캐스트 그룹에 대해서 Xcast 헤더 크기화 패킷을 전달하는데 드는 비용을 기존의 방법보다 훨씬 줄일 수 있음을 보인다. 이 방안의 장점은 큰 밀집 그룹에 대해 헤더 오버헤더와 라우터에서의 패킷 처리 시간을 줄이고 동적인 그룹 멤버쉽을 지원한다.

Key Words : Xcast; multicast; hierarchical joining group; domain Xcast agent.

ABSTRACT

For the Explicit Multicast (Xcast) scheme, we propose a scheme that provides dynamic group membership and supports a large, dense multicast group. To do so, we use a hierarchical joining mechanism, where a domain agent joins a group on behalf of hosts. In our scheme, a sender transmits multicast packets to a domain agent, and then the agent forwards the packets to receivers in a domain. A host can join or leave a group at any time via the domain agent. Thus, a destination field of the Xcast header has addresses of domain agents instead of hosts. Simulation results show that our scheme can reduce more the Xcast header size and cost in forwarding packets than the Xcast, especially for a large multicast group. Like traditional multicast schemes, our scheme can support large multicast group members.

Ⅰ. Introduction

IP multicast allows transmission of an IP datagram to a set of hosts that form a multicast group, even though the group members might be spread across more than one distinct physical network. This has been becoming increasingly important for applications such as IP telephony, video-conferencing and network games [1].

Traditional multicast schemes [2-3] were

* Computer Education, Andong National University (bsgim@andong.ac.kr)
논문번호: 040120-0317, 접수일자 : 2004년 3월 17일
* This work was supported by a grant from 2003 Research Fund of Andong National University.

designed to handle very large multicast groups. The advantages of this traditional scheme are 1) dynamic group membership, where any nodes that know the multicast group address can join or leave the group at any time, and 2) on-demand data distribution, where any group members can send data to other members at any time. At the same time, any nodes can start receiving data anytime without disturbing the sender [4]. However, if there are a very large number of groups, those schemes have some problems: Multicast group addresses need a global address allocation mechanism. When a multicast packet arrives in a router, the router knows nothing about the destinations of the packet. Moreover, multicast routing protocols require inter-domain routing [4, 5].

The Xcast [5] is a newly proposed multicast scheme in order to support a very large number of small groups. Instead of using multicast group address, the Xcast encodes a list of destinations in the packets. In addition, the sender controls multicast group membership. The advantages of the Xcast are as follows: routers do not have to maintain state per session, it is not necessary to allocate multicast address and no control message is exchanged among routers. However, when a multicast group is very large, the size of the Xcast header is so large too. Thus, the processing time of an Xcast packet at Xcast-aware routers (XRs) will be high.

In this paper, we propose a scheme that supports dynamic group membership and a large, dense multicast group via the Xcast. To do so, our scheme uses a heirarchical joining group: a host joins a domain agent, and then that agent immediately joins a group on behalf of the host. In this scheme, a sender transmits Xcast packets to domain agents, and then each domain agent forwards the packets to group receivers in a domain. Simulation results show that the scheme can reduce overhead for the header processing.

The remainder of this paper is organized as follows: Section 2 reviews related works. Section 3 presents our scheme that supports a large multicast group. Our scheme is evaluated in Section 4 and finally, Section 5 concludes the paper.

Ⅱ. Related works

In the Xcast, a source node maintains a list of destinations to identify group members, encodes the list in the Xcast header, and then sends the packet to a router: The destination of the packet is specified by a list of unicast addresses instead of a multicast group address. Each XR along the delivery path parses the header, partitions the destinations based on each destination's next hop, and then forwards a packet to each of the next hops. When there is only one receiver on the list, the Xcast packet can be converted into a unicast packet.

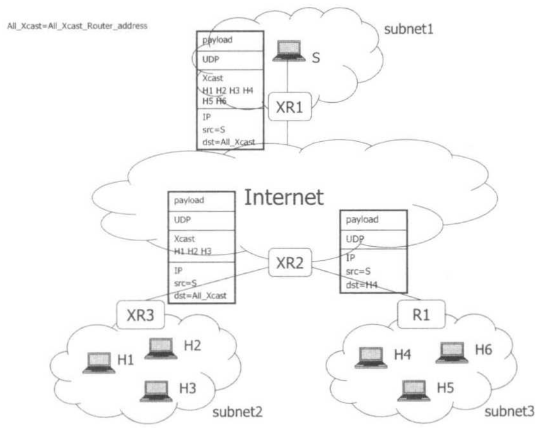

Figure 1. Xcast scheme

For example, suppose that a sender S is trying to send Xcast packets to destinations (H1, H2, ... , H6) in Figure 1, where router (R) is not an Xcast-aware router. The packet sent by S looks like this: [src=S | dest = H1 H2 H3 H4 H5 H6 | payload]. When XR2 receives the packet, it will send one copy of the packet to XR3, [src=S | dest=H1 H2 H3 | payload]. In addition, it will send 3 copies of the packet to destinations (H4, H5, H6) as an ordinary unicast packet. Each packet transmitted along the delivery path contains all remaining destinations. If a multicast group is very large, the header size of the Xcast may be so large. Thus, overhead for the header processing could be very large.

Ⅲ. Domain Xcast Agent Scheme

The destination of the Xcast header contains a list of group receivers. As a multicast group increases, the header size of the Xcast will also increase. The Xcast header might be so large for a large multicast group. In order to support a large, dense multicast group, we introduce a hierarchical joining group mechanism to the Xcast: hosts join a domain Xcast agent (DXA) that is a Xcast router, and then the DXA joins a group on behalf of the hosts in its domain. A source keeps a receiver list of DXAs to identify its group members. The source transmits Xcast packets to DXAs, and then each DXA forwards the packets to group receivers in a domain. Thus, the Xcast packet transmitted by the source will have a list of DXAs instead of hosts. Each DXA can maintain an Xcast group table (XGT) consisting of source address, multicast group address, and a receiver list, where the list is of host addresses in a domain. A DXA uses this table to convert the destination in the header into host address. Each router along the delivery path processes the packet in the same way as the Xcast scheme. When a DXA receives an Xcast packet whose destination is its own address, and if the next hop is an XR, it converts the destination to host address using its XGT. But, if the next hop is not an XR, a DXA converts the packet into one or more unicast packets using its XGT.

Figure 2 shows the proposed scheme architecture. There are 3 domains connected with the Internet, and each domain has a single DXA. We assume that a sender S tries to send multicast datagrams to group receivers (H1, H2, H3 and H4). The Xcast packet sent by the S looks like: [src=S | dest=DXA2 DXA3 | payload]. When XR1 receives the packet, it will divide into two packets as a normal Xcast packet: One is [src=S | DXA2 | payload]. The other is [src=S | DXA3 | payload]. When DXA2 receives the former packet, it will convert the destination into host address using its XGT since the next hop is an XR. Thus, the packet will be [src=S | H1 H2]. In case of DXA3, it will convert the latter packet into two unicast packets since the next hop is not an XR.

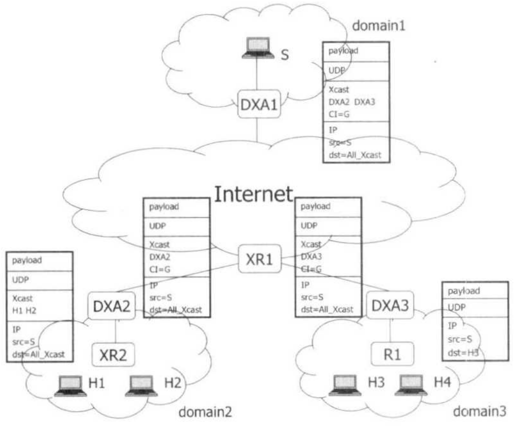

Figure 2. Domain Xcast agent scheme

For supporting dynamic group membership, it is necessary that a host can join and leave a multicast group at any time.

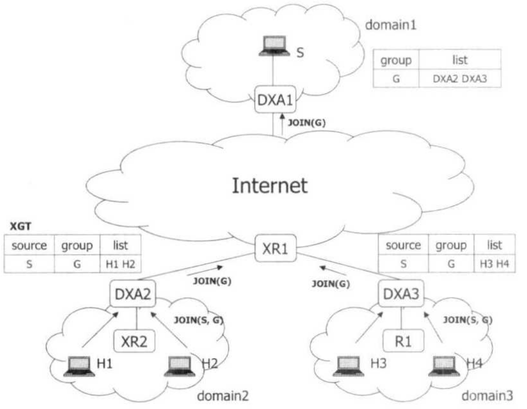

Figure 3. Joining a multicast group

Figure 3 shows the behaviors of joining a multicast group. When a host wants to join a multicast group, the host should send a JOIN(S, G), where S is the sender address, and G is the multicast group address, to the DXA in its domain. When DXA2 receives the JOIN, it stores information about the host in its XGT. If the host is the first group member, DXA2 immediately sends a JOIN(G) to the sender (S1) in order to join the group on behalf of the host. Otherwise, it is not necessary to send the message. Thus, when H2 joins the multicast group after H1, DXA2 need not send the JOIN message to the sender, because the sender already has sent multicast datagrams to DXA2. When the sender receives the JOIN, it adds the receiver who sent the JOIN to its receiver list.

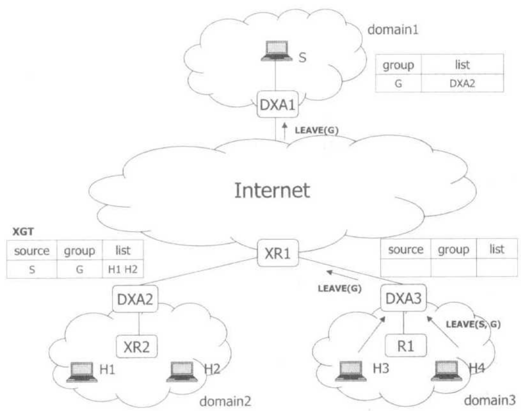

Figure 4. Leaving a multicast group

When a group member wants to leave a multicast group, the member should send a LEAVE(S, G) to the DXA in its domain as shown in Figure 4. When DXA3 receives the LEAVE, it deletes an entry of the host from its XGT. If the host is the last group member, DXA3 sends a LEAVE(G) to the sender (S1). Otherwise, there is no need to send the message. When the sender receives the LEAVE, it deletes the receiver from its receiver list.

Ⅳ. Performance analysis

The Xcast header for IPv4 is composed of a fixed part and two variable parts. The fixed part size is 12 octets and every destination has a corresponding bit in the bitmap field of the variable part, where the bitmap is 4-octet aligned. If P bit in the header is not set, only each destination address (four octets) is stored in the variable part that is a destination list field. Thus, according to the number of destinations, the length of the Xcast header for IPv4 is different.

We have evaluated the performance of the proposed scheme using SMPL(Simulation and Modeling Programming Language) [6]. For the performance evaluation of our scheme, we used a mesh network, where the network size was 36 x 36 subnets, and all routers were XRs. To compare our scheme with the Xcast scheme, it was assumed that the service area of the DXA is 3 x 3 square with 9 subnets, where the coordination of the DXA is (0,1) in each domain. In our simulation, a single multicast group was considered, and several simulation runs were performed with various sizes in the multicast group ranging from 1 to 64. After a number of simulations were performed, the averages were computed.

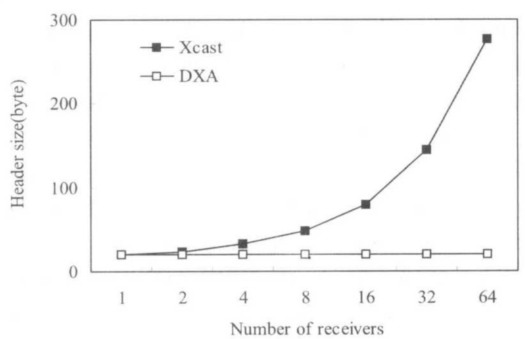

Figure 5. Cost of header size in a domain

Figure 5 shows the header size of both the Xcast and the DXA transmitted by the sender when the number of receivers varies in a domain. As the number of receivers in a domain increases, the header size of the Xcast steeply increases. If a group is very large, the header overhead will be so high in the Xcast scheme. Thus, the Xcast could not support a very large group. As shown in Figure 5, our scheme is not dependent on the number of receivers in a domain, because only a single DXA joins a group on behalf of receivers in a domain; therefore, for a dense multicast group, the number of destinations does not affect the overhead of header processing.

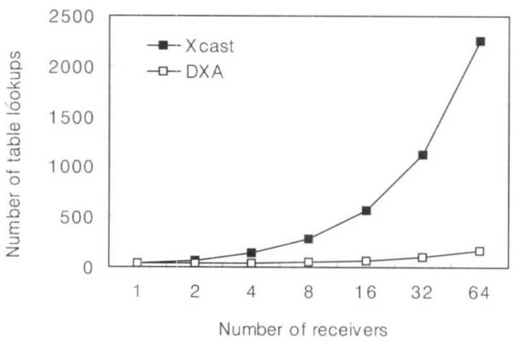

Figure 6. Cost of table lookups

Each XR performs a route table lookup in the Xcast routing table to determine the next Xcast hop for each of the destinations listed in the packet. Figure 6 shows the number of routing table lookups until a packet arrives at all destinations when the number of receivers varies in a domain. As the number of receivers in a domain increases, the number of table lookups in the Xcast scheme increases as well. This is because the number of destinations for the Xcast also increases. However, our scheme slowly increases, because the header destination only contains DXA addresses instead of hosts. Thus, each XR has less decision time for the next hop in our scheme. Thus, comparing with the Xcast scheme, the packet processing time at XRs will be very small for a dense group.

From both Figure 5 and 6, we can see that our scheme largely reduces the header size and the number of route table lookups for a dense, large group.

V. Conclusion

We have presented a heirarchical joining group called DXA for a large, dense multicast group via the Xcast. The DXA receives Xcast packets from a sender, and then forwards the packets to receivers in a domain. In the proposed scheme, a receiver can join or leave a multicast group dynamically, and a destination field of an Xcast packet has addresses of DXAs instead of hosts; therefore, the scheme can support both dynamic group membership, and a large, dense multicast group for the Xcast scheme. From the simulation results, our scheme can reduce the header size and the number of table lookups for a large, dense group. Thus, we can reduce much overhead for the header processing time and the transmission time. Like traditional multicast schemes, our scheme can support large multicast group members.

References

[1] R. Boivie, N. Feldman & C. Metz, "Small Group Multicast, A New Solutions for Multicasting on the Internet," *IEEE Internet Computing*, Vol. 4, No. 3, May/June 2000.

[2] D. Waitzman, C. Partridge & S. Deering, "Distance Vector Multicasting Routing Protocol," *RFC 1075*, Nov. 1988.

[3] Ballardie, "Core Based Trees (CBT) Multicast Routing Architecture," *RFC 2201*, Sep. 1997.

[4] V. Visoottiviseth, Y. Takahashi, Y. Kadobayashi & S. Yamaguchi, "SIM: Sender Initiated Multicast for small group communications," Proceeding of INET, 2001.

[5] R. Boivie, N. Feldman, Y. Imai, W. Livens, D. Ooms and O. Paridaens, "Explicit Multicast (Xcast) Basic Specification," *IETF Internet-Draft*, draft-ooms-xcast-basic-spec-04.txt, 2003.

[6] M.H. MacDougall, "Simulating Computer Systems: Techniques and Tools," *MIT Press*, 1987.

김 병 순(Byungsoon Kim) 정회원

1991년 2월 : 서강대학교 컴퓨터학과 졸업
1993년 2월 : 서강대학교 컴퓨터학과 석사
2003년 2월 : 경북대학교 컴퓨터공학과 공학박사
1993년 1월 ~ 1996년 3월 : 현대전자 정보시스템 사업본부
1996년 3월 ~ 2003년 8월 : 대구공업대학 컴퓨터정보계열 조교수
2003년 9월 ~ 현재 : 안동대학교 컴퓨터교육과 전임강사

<관심분야> 이동 통신, 멀티캐스트, 임베디드 시스템