Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®
Computer Networks 43 (2003) 539–555
COMPUTER NETWORKS
www.elsevier.com/locate/comnet

Scalable sender access control for bi-directional multicast routing

Ning Wang *, George Pavlou

Center for Communication Systems Research, University of Surrey, Surrey, United Kingdom

Received 10 October 2002; received in revised form 4 April 2003; accepted 4 April 2003

Responsible Editor: I.F. Akyildiz

Abstract

Bi-directional shared tree is an efficient routing scheme for interactive multicast applications with multiple sources. Given the open-group IP multicast service model, it is important to support sender access control in order to prevent group members from receiving irrelevant data, and also to protect the multicast tree from denial-of-service (DoS) attacks. Compared with source specific and uni-directional shared trees, where information sources can be authorized or authenticated at the single root or rendezvous point (RP), in bi-directional trees this problem is more pronounced since hosts can send data to the shared tree from any point in the network. In this paper we propose a scalable sender access control policy mechanism for bi-directional shared trees so that irrelevant data is policed and discarded once it arrives at an on-tree router. We consider both intra- and inter-domain routing, so that the mechanism can cope with large-scale multicast applications or many concurrent multicast sessions across multiple administrative domains.

Keywords: Multicast security; Denial of service; Bi-directional tree

1. Introduction

IP multicast [5] supports point to multi-point communication services for applications in which an information source sends data to a group of receivers simultaneously. Although some IP multicast applications have been available on the experimental multicast backbone (MBone) for several years, large-scale deployment has not been achieved until now. IP multicast is also known as "any source multicast (ASM)" since any information source, even outside a group, can send data to a multicast group without any control mechanism i.e. in the current service model group management is not stringent enough to control both senders and receivers. IGMPv2 [7] allows group members to join or leave a session but there are no control mechanisms to avoid receiving data from particular sources or prevent receivers from receiving sensitive information. It is common belief that the above characteristics have somehow prevented the successful deployment of IP multicast in a large scale on the Internet [6].

Realizing that many multicast applications are based on a one-to-many communication model, e.g.

* Corresponding author.
E-mail addresses: n.wang@eim.surrey.ac.uk, n.wang@surrey.ac.uk (N. Wang), g.pavlou@eim.surrey.ac.uk, g.pavlou@surrey.ac.uk (G. Pavlou).

1389-1286/$ - see front matter
doi:10.1016/S1389-1286(03)00289-5

Internet TV/radio, content distribution, etc., H.W. Holbrook et al. proposed the EXPRESS routing scheme [9], from which the source specific multicast (SSM) [10] service model subsequently evolved. In SSM each group is identified by an address tuple (S, G) where S is the unique address of the information source and G is the destination channel address. A single multicast tree is built, rooted at the well-known source for delivering data to all subscribers. In this situation, centralized group authorization and authentication can be achieved at the root of the single source through application level mechanisms. IGMPv3 [3] is currently being extended to support source specific joins in SSM.

On the other hand, there exist many other applications based on a many-to-many communication model, such as multi-party videoconferencing, distributed interactive simulation (DIS), multi-party Internet games, etc. For this type of applications, bi-directional shared trees, such as core based tree (CBT) [1], bi-directional PIM [8], and RAMA style simple multicast [14], are efficient routing schemes for delivering data between peering hosts. Fig. 1 illustrates the difference between uni- and bi-directional multicast routing: assuming that router C is the core/root of the multicast tree, in uni-directional routing each information source should unicast its data to C from where packets are forwarded to all receivers. In contrast, bi-directional routing allows traffic from sources to be delivered directly to group members without necessarily having to pass through the core; this results in smaller end-to-end delay and bandwidth conservation. However, since there is no single point for centralized group access control, sender authorization and authentication become difficult challenges. For example, a malicious host may attempt a denial-of-service (DoS) attack by flooding bogus data from any point of the bi-directional multicast tree. Sender access control for bi-directional trees in IP multicast is not catered for in the specification of any of the corresponding routing protocols [1,8]. In addition, it is not known if SSM can be extended to bi-directional multicast routing, hence the source filtering function of IGMPv3 may not apply to the underlying protocols. A potential solution that has been proposed in the literature is to periodically "push" the entire sender access list down to all the on-tree routers, so that only data from authorized senders can be accepted and forwarded to the bi-directional tree full policy maintenance (FPM) [2]). This simple access control mechanism has been adopted in RAMA-style simple multicast [14]. However, this approach is not scalable, especially when many multicast sessions are active and/or large group sizes with many senders are involved.

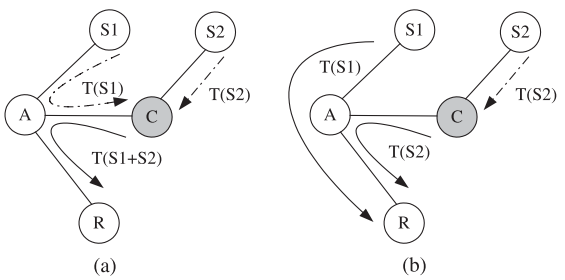


Fig. 1. (a) Uni-directional routing vs. (b) bi-directional routing.

In this paper we propose an efficient and scalable sender access control mechanism for bi-directional trees in IP multicast. The basic idea is to deploy access policy for external senders on the tree routers only where necessary, so that data from unauthorized senders is policed and discarded once it arrives on the bi-directional shared tree. Our scheme has little impact on the current bi-directional routing protocols and can be directly implemented without modifying the basic function of the current routing protocols. Moreover, the overhead introduced is much smaller than that proposed in [2,14].

The rest of the paper is organized as follows: Section 2 gives the overview of our dynamic maintenance of the access control policy. Sections 3 and 4 introduce sender authorization and authentication in both intra- and inter-domain routing. Considerations for multi-access networks are especially discussed in Section 5. In Section 6 we assess the scalability of our proposed scheme and we finally present a summary in Section 7.

2. Sender authorization and authentication overview

2.1. Sender access control deployment

Compared with source specific trees and even uni-directional shared trees such as PIM-SM [4], in

which external source filtering can be performed at the single source or rendezvous point (RP) where the registrations of all the senders are processed and authorized, in bi-directional trees this is much more difficult since data from any source is directly forwarded to the whole tree once it hits the first on-tree router. In fact, since there is no single point for centralized sender access control, source authorization and authentication has to be deployed at the routing level. As we have already mentioned, the simplest solution for this is to periodically broadcast the entire access control list down to all the routers on the bi-directional tree for deciding whether or not to accept data (e.g., [14]). However, this method is only feasible when a few small-sized groups with limited number of senders are considered. For large scale multicast applications, if we do not send the whole policy to all the on-tree routers for scalability reasons, three questions need to be answered as stated in [2]: (1) How to efficiently distribute the list where necessary? (2) How to find edge routers that act as the trust boundary? (3) How to avoid constant lookups for new sources? In fact if we try to statically mount the access control policy to an *existing* bi-directional multicast tree, none of the above three questions can be easily answered.

It should be noted that most multicast applications are highly dynamic in nature, with frequent join/leaving of group members and even information senders. Hence the corresponding control policy should also be dynamically managed. Here we propose an efficient sender-initiated distribution mechanism of the access control list during the phase of multicast tree construction. The key idea is that each on-tree router only adds its *downstream* senders to the local sender access control list (SACL) during their join procedure, and the senders in the access list are activated by a notification from the core. In fact, only the core has the right to decide whether or not to accept the sources and it also maintains the entire SACL for all the authorized senders. Packets from an unauthorized host (even if it is actually on the tree) will be discarded once they reach any on-tree router. To achieve this, all senders must first register with the core before they can send data to the group. When a registration packet hits an on-tree router, the unicast address of the sender is added to the SACL of each router on the way. Under this scheme, the access policy for a particular sender is deployed on the branch from the first on-tree router where the registration is received along to the core router. We define the interface from which this registration packet is received as the downstream interface (DI) and the one used to deliver unicast data to the core as the upstream interface (UI). The format of each SACL entry is (G, S, I) where G indicates the group address, S identifies the sender and I is the downstream interface from which the corresponding registration packet was received. Once the registration reaches the core, the latter will contact a source authorization server (SAS) for deciding whether to accept the new sender. If the SAS has approved the join, the core will send an "activating packet" back to the source, and once each on-tree router receives this packet, it will activate the source in its SACL so that it is able to send data to the bi-directional tree from then on. In such a scenario, an activated source can only send group data to the tree via the path where its SACL entry has been recorded, i.e., even if a sender has been authorized, it cannot send data to the group from other branches. Source authentication entries kept in each SACL are maintained in soft state for flexibility, and this requires that information sources should periodically send "refresh" packets up to the core to keep their state alive in the upstream routers. This action is especially necessary when a source is temporarily not sending data for a period. When data packets are received from a particular sender, the on-tree router can assume that this source is still alive and will automatically refresh its state. If a particular link between the data source and the core fails, the corresponding state will time out and become obsolete. In this case, the host has to seek an alternative path to perform re-registration for continuing sending group data.

The difference between our dynamic access control scheme and the FPM mechanism is illustrated in Fig. 2(a) and (b) respectively. It should be noted that only the on-tree routers having received the sending request from the new source h (in gray color) need to maintain the policy for h. This is more scalable compared with the approach in

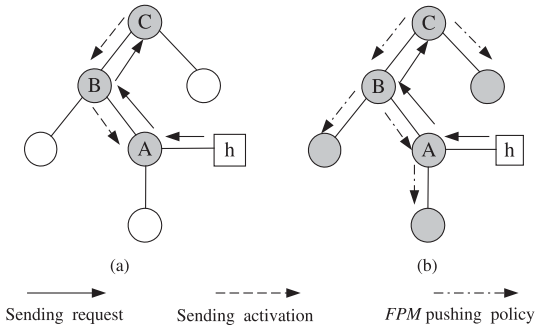


Fig. 2. Sender access control policy comparison.

which all on-tree routers keep the entire sender list. However, this requires that the sender should send data to the bi-directional tree *only* from the designated ingress router (router A in Fig. 2).

2.2. Data authentication and forwarding

When a router receives a data packet from one of its downstream interfaces, it will first check if there exists such an entry for the data source in its local SACL. If the router cannot find a matching entry that contains the unicast address of the source, the data packet is discarded. Otherwise if the corresponding entry has been found, the router will verify if this packet comes from the same interface as the one recorded in the SACL entry. Only if the data packet has passed these two authentication mechanisms, it will be forwarded to the upstream interface and the other interfaces with the group state, i.e., interfaces where receivers are attached. On the other hand, when a data packet comes from the upstream interface, the router will always forward it to all the other interfaces with group state without performing any authentication. Although the router cannot judge if this packet is from a registered sender since it comes from the upstream router, there exist only two possibilities: *either* the upstream router has the SACL entry for the data source or it has received the packet from its own parent router in the tree. The extreme case is that none of the intermediate ancestral routers have such an entry and then we have to backtrack to the core. Since the core has recorded entries for all the registered senders and it never forwards any unauthenticated packet on its downstream interfaces, we can safely conclude that each on-tree router can trust its parent, hence packets received from the upstream interface are always from valid senders. By maintaining such a trust chain residing on the bi-directional routing tree, sender access control for information sources can be achieved in a scalable fashion. However, this does not include the case of routers attached on multi-access networks such as LANs, we will discuss the relevant considerations and required additional operations in Section 5.

3. Intra-domain access control policy

3.1. Sender access control list construction and activation

In the IP Multicast architecture, information sources and receivers (referred to as group members) are treated separately. However, in many interactive applications, a host may act in both roles simultaneously. In this section, we classify senders of a multicast session as follows: if a host wants to be both sender and receiver, it must join the multicast group and become a send–receive capable member (SR-member, SRM). Otherwise if the host only wants to send data to the group without receiving any, it may choose to act either as a send-only member (SO-member, SOM) or a non-member sender (NMS). In the former case, the host must join the bi-directional tree in order to send data, and its designated router (DR) will forward the packets on the upstream interface as well as other interfaces with the group state. In the IP multicast model, information sources are allowed to send data to the group without becoming members. Hence, if the host is not interested in the information from the group, it may also choose to act as a NMS. In this case, the host must unicast its data packets towards the core. Once the data packet hits the first on-tree router and passes the corresponding source authentication, it will be forwarded to all the other interfaces with group state. We discuss below the exact mechanism for SACL construction and activation; the description is based on the CBT routing protocol but it can also apply to other bi-directional routing schemes

such as Bidir-PIM and border gateway multicast protocol (BGMP). Detailed flowchart for SACL construction and activation on each router is presented in Appendix A, while SACL-based data authentication and forwarding is shown in Appendix B.

(1) *SR-member join.* When the designated router (DR) receives a group G membership report from a SR-member S on the LAN, it will send a join request towards the core. Here we note that the group membership report cannot be suppressed by the DR if it is submitted by a send-capable member. Once a router receives this join request packet from one of its interfaces, say, A, then the (G, S, A) entry is added to its SACL. If the router is not been on the shared tree, a (*, G) state is created with the interface leading to the core as the upstream interface and A is set to the downstream interface. At the same time interface A is also added to the downstream interface (DI) list for group G, so that data from other sources can be forwarded to S via A. If the router already has (*, G) state, but A is not in the interface list with group state, then it is added to the DI list. Thereafter, the router just forwards the join request to the core via its upstream interface based on the underlying unicast routing table. On the other hand, if the router receives any join packet from its upstream interface, the packet will be dropped based on RPF check.

(2) *SO-member join.* Similar to SR-member joins, the DR of a SO-member also sends a join request up to the core and when the router receives this request from its interface A, the (G, S, A) entry is added to the local SACL. If the router is not yet on the tree, (*, G) state will be generated but interface A is not added to the DI list for group G. This guarantees that A will not forward group data to a send-only member S later on.

(3) *Non-member sender (NMS) registration.* Here we use the terminology ''registration'' instead of ''join request'', since this host is not a group member and does not need to be on the tree in order to send group data. The registration packet from the non-member sender is unicast towards the core and when it hits the first router with (*, G) state, the (G, S, A) entry is created in the local SACL of all the on-tree routers on the way to the core. It should be noted that if a router is not on the tree, it does not maintain SACL for the group even if it has received the registration.

Finally, if a receive-only member (also known as the *group member* in the conventional multicast model) wants to join the group, the join request invokes a (*, G) state if the router is not on the tree, but no new SACL entries need to be created. Moreover, once the join request hits any on-tree router, a join-notification is immediately sent back without informing the core.

Once each on-tree router receives the activating notification from the core, the (G, S, A) entry is activated so that data from S can be forwarded on this router. Thereafter, the router will forward this notification packet exclusively on interface A (the interface from which the original join request was received) for activating the corresponding SACL on its downstream routers. It is noticed that route selection of notification packets is not shortest path routing, but is based on the interface where the pending join request is attached. Hence interface A might not be on the shorted path back to the sender S if the network link metric is asymmetric. This type of forwarding guarantees that the notification packet will finally follow the reversed path back to reach the source's DR. It is also worth mentioning that in the following two cases the notification is not valid and should be discarded: (i) There is not a matching (G, S, A) entry and (ii) the notification packet does not come from the upstream interface. The second requirement is in effect reversed path forwarding (RPF) checking, which is necessary because notification packets should exclusively originate from the core, and they must not appear on any interfaces except the upstream interface.

3.2. An example for intra-domain access policy

A simple network model is shown in Fig. 3(a). We assume that node A is the core router and all the designated routers (DR) of potential members of group G should send join request to this node. Hosts H1–H5 are attached to the individual routers as shown in the figure.

Initially suppose H1 wants to join the group. Its DR (router B) will create (*, G) state and send the

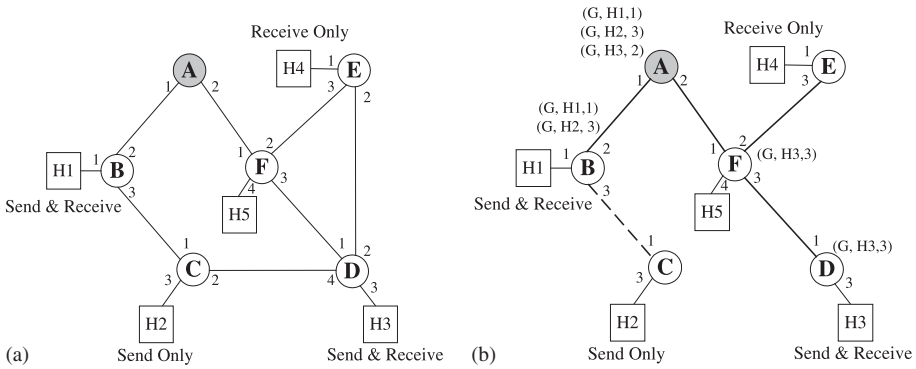


Fig. 3. Intra-domain SACL construction and activation.

join request to the core A. Since H1 is a SR-member that can both send and receive data, each of the routers the join request has passed will add this sender to its local SACL. Hence both routers B and A will have the SACL entry (G, H1, 1), since they both receive the join request from interface 1. Host H2 wants to only send data to group G, so it may choose to join as a SO-member or just act as a NMS. In the first case, its DR (router C) will create (*, G) state indicating that this router is on the tree and then add H2 to its SACL. Thereafter, router C will send a join request indicating H2 is a SO-member towards the core; when B receives this request, it will also add H2 to its local SACL and then forward the join request packet to A. Since H2 does not want to receive data from the group, the link BC becomes a send-only branch. To achieve this, router B will not add B3 to the interface list with group state. If H2 chooses to act as the non-member sender, router C will not create (*, G) state or SACL for the group but will send a registration packet towards A. When this packet hits an on-tree router, i.e. B in our example, H2 will be added to the local SACL of all the routers on the way. When sending group messages, router C just encapsulates the data destined to the core by setting the corresponding IP destination address to A. When the data reaches B and passes the SACL authentication, the IP destination address is changed to the group address originally contained in the option field of the data packet, and the message is forwarded to interfaces B1 and B2 to get to H1 and the core respectively. After H3 and H4 join the group, the resulting shared tree is shown in Fig. 3(b) with the SACLs of each on-tree router. It should be noted that H4 is a receive-only member, and hence routers E, F and A need not add it to their local SACLs. Suppose router F has received group data from H3 on interface F3, it will check in its local SACL if H3 is an authorized sender. When data passes the address and interface authentications, it is forwarded to both interfaces F1 and F2. When group data is received on the upstream interface F1, since its parent A is a trusted router (the data source should be either H1 or H2), it is forwarded to F2 and F3 immediately without authentication. However, if the non-registered host H5 wants to send data to the group, this will not be forwarded to the bi-directional tree due to the SACL authentication failure at router F.

4. Inter-domain access control policy

4.1. Basic descriptions

As we have mentioned above, on-tree routers only maintain the access policy for the downstream senders. However, if large-scale groups with many senders or many concurrent sessions are considered, the size of the SACL in the routers near the core will become a heavy burden for those routers. In this section we discuss how this situation can be improved with the aid of inter-domain IP multicast routing semantics.

Our idea is based on hierarchical access control policy to achieve scalability. Routers only maintain SACL for the downstream senders in the *local* domain and do not need to add sources from downstream domains to their local SACLs. In other words, all the senders for the group are only authenticated in the local domain. In the root domain, the core needs to keep entries only for its local senders; however in order to retain the function of authorizing and activating information sources from remote domains, on receiving their registrations the core router needs to contact the source authorization server residing in the local domain, which decides whether or not to accept the requests.

For each domain, a unique border router (BR) is elected as the "policy agent" and keeps the entire SACL for all the senders in the local domain, and we name this the designated border router (DBR) for the domain. In fact the DBR can be regarded as core of the sub-tree in the local domain. In this sense, all the data from an upstream domain can only be injected to the local domain from the unique DBR and all the senders in this domain can only use this DBR to send data up towards the core. This mechanism abides to the "third party independence" policy in that data from any sender must be internally delivered to all the local receivers without flowing out of the domain. This requires that joins from different hosts (including both senders and receivers) merge at a common point inside the domain. In BGP-4, all the BRs of a stub domain know for which unicast prefix(es) each of them is acting as the egress router, this satisfies the above requirement of "path convergence" of internal joins we just mentioned.

Since individual sender authentication is performed within each domain and invalid data never gets any chance to flow out of the local domain, the on-tree BR of the upstream domain will always trust its downstream DBR and will assume that all the data packets coming from it originate from authorized senders. Hence, when a packet leaves its local and enters remote domains, no further authentication is needed. This also avoids constant lookups when the authenticated data is traveling on the bi-directional tree.

4.2. Inter-domain SACL construction and activation

Since BGMP [11] has been considered as the long-term solution to the Inter-domain multicast routing, in this section we will take BGMP as an example to illustrate how sender access control policy can be deployed in inter-domain applications.

First we discuss how the designated router for a group member sender submits its join request and how it is added to the SACL and activated. This applies to both SR-members and SO-members, the only difference between the two being whether or not to add the interface from which the join request was received to the interface list with the group state. Only if an on-tree router receives a join request from a sender in the local domain, it will add this sender to its SACL, otherwise the router will just forward the join request towards the core without updating its local SACL.

In Fig. 4, when host S wants to become a SO-member to send data, its DR (router A) sends a join request towards the DBR router B, which has the best exit to the root domain. All the internal routers receiving this request will add S to their local SACLs. Since B is the core of the sub-tree for the local domain, it also needs to create a SACL entry for host S once it receives the join request from its multicast interior gateway protocol (MIGP) component. Thereafter, B finds in its group routing information base (G-RIB) that the best route to the root domain is via its external peer C in the transit domain, so router B will send the BGMP join request towards C via its BGMP component. Once router C receives the join request, it creates (*, G) state (if it has not been on the tree), but will not create an entry for S in its local SACL. When C finds out that the best exit toward the root domain is D, it just forwards the join request to this internal BGMP peer, and hence router D becomes the DBR of the transit domain for group G. Suppose Bidir-PIM is the MIGP, the RP in this transit domain should be placed at D, and router C will use its MIGP component to send the join request towards D. When this join request travels through the transit domain, none of the internal routers along the way in the domain will

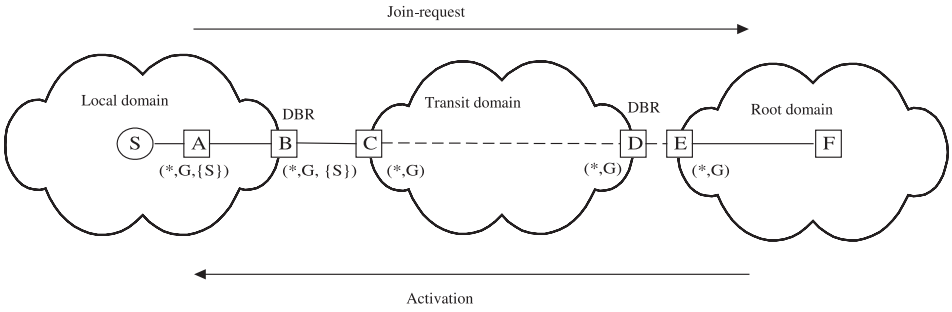


Fig. 4. Inter-domain join and activation.

add S to their local SACLs. After the join request reaches the root domain and the core router F authorizes the new sender by contacting the access control server and sends back the activating-notification, all the on-tree routers (including internal on-tree routers and the DBR) in the transit domain just forward it back towards the local domain where the new sender S is located. When the packet enters the local domain, all the on-tree routers (i.e. B and A in Fig. 4) will activate S in their SACLs.

As we have mentioned previously, a send-only host may also choose to act as a non-member sender (NMS). However there are some restrictions when inter-domain multicast routing is involved. If a send-only host is located in the domain where there are no receivers (we call this domain a *send-only domain*), then the host should join the bi-directional tree as a SO-member other than a non-member sender (NMS). Otherwise if the host acts as a NMS, its registration packet will not hit any on-tree router until it enters remote domains. This forces the on-tree router there to add a sender from another domain to its local SACL, which does not conform to the rule that on-tree routers only maintain access policy for senders in the local domain. On the other hand, if the host joins as a SO-member and since its DR will be on the tree, the authentication can be achieved by the on-tree routers in the local domain. It should be noted that for any on-tree routers in the send-only domain, the interface from which the join request for the SO-member is received is not added into the group's downstream interface list (which is always empty in a send-only domain for the group), and hence group traffic will not flow into the local domain at any time.

4.3. An example for inter-domain access policy

An example for inter-domain sender access control is given in Fig. 5. C is the core router and domains X, Y and Z are remote domains regarding the core C. Hosts a, b, c and d are attached to the routers in different domains. Also suppose that host a only wants to receive data from the group, hosts b and c want to both send and receive, while host d only wants to send data to the group. In this case, X is a receive-only domain and Z is a send-only domain. X1, Y1 and Z1 are BR that have been selected as the DBR for each domain. According to our inter-domain access control scheme, on-tree routers have the SACL entry for

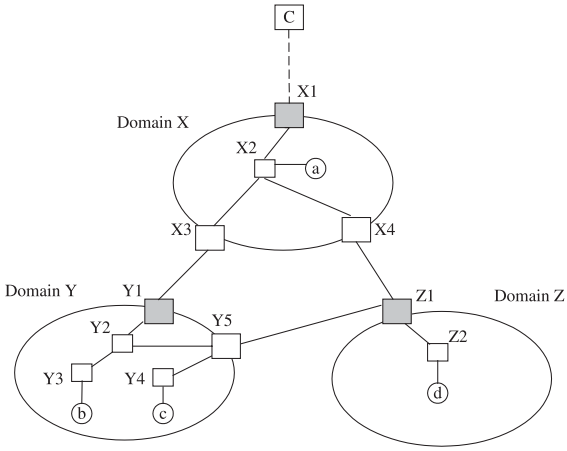


Fig. 5. Example for inter-domain sender access control.

downstream senders in the local domain, and each DBR has the policy for all the senders in the local domain. Hence, Y1 has the entry for hosts b and c in its SACL while the SACL of X1 contains no entries at all. Although X is the parent domain of Y and Z which both contain active senders, all the on-tree routers in X do not need to add these remote senders to their SACL. In fact data from Y and Z has already been authenticated by their own DBRs (i.e. Y1 and Z1) before it flows out of the local domains. Since host d only wants to send data to the group and there are no other receivers in domain Z, as we have mentioned, host d should join as a SOM. Otherwise if d acts as a NMS and sends its registration packet towards the core, this makes the first on-tree router (X2) add d to its SACL, but this is not scalable because on-tree routers are forced to add senders from remote domains. On the other hand, if host d joins as a SOM, the shared tree will span to its DR, i.e. Z2, and then the authentication can be performed at the routers in the local domain.

BGMP provides also the mechanism for building source-specific branches between BR. In Fig. 5, we assume that the current MIGP is PIM-SM. At certain time the DR in domain Y such as Y3 or Y4 may wish to receive data from host d in domain Z via the shortest path tree. Hence (S, G) state is originated and passed to the border router Y5, which is not the current DBR of domain Y. When Y5 receives the source specific join, it will create (S, G) state and then send the corresponding BGMP source specific join towards Z1. On the other hand, since Z1 is the DBR of domain Z, intra-domain sender authentication has been performed before the traffic is sent to Z1's BGMP component for delivery to remote domains. In fact Y5 will only receive and accept data originated from host d in domain Z due to its (S, G) state filtering. Once Y5 receives the data from host d, it can directly forward it to all the receivers in the local domain, since RPF check can be passed. When the DR receives the data from d via the shortest path, it will send a source specific prune message up towards the root domain to avoid data duplication. It should be noted that (*, G) state should only exist in the DBR for each domain/group, and internal nodes may only receive source specific traffic via alternative BR. From this example, we can also see that source specific tree can also interoperate with the proposed sender access control in the receiver's domain (note that the MIGP in domain Y is not a bi-directional routing protocol).

5. Operations on multi-access networks

We need special consideration for protecting group members from unauthorized sources attached to multi-access networks such as LANs. As we have mentioned, if an on-tree router receives data packets from its upstream interface, it will always forward them to all the other interfaces with group state, assuming that these come from an authorized information source. However this may not be the case if the upstream interface of an on-tree router is attached to a broadcast network. When an unauthorized host wants to send data with group address to the multi-access LAN, a corresponding mechanism must be provided to prevent these packets from being delivered to all the downstream group members. To achieve this, once the designated router (DR) on the LAN receives such a packet from its downstream interface, if it cannot find a matching access entry for the data source in its SACL, it will discard the packet, and at the same time it will send a "forbidding" control packet containing the unicast address of the unauthorized host to the LAN from its downstream interface. Taking the CBT routing protocol as an example, the IP destination address of this forbidding packet should be "all-CBT-router address (224.0.0.15)" and the value of TTL is set to 1. Once the downstream router receives this packet on its upstream interface, it will stop forwarding the data with this unicast address that originates from an unregistered host attached to the LAN. Hence all the downstream session members will only receive little amount of unauthorized data for a short period of time. In terms of implementation, the downstream on-tree routers should maintain a "forbid list" of unauthorized hosts recorded. Since all the possible unauthorized hosts can only originate from the local LAN, this list should not introduce much overhead to the routers. In Fig. 6, let's assume the

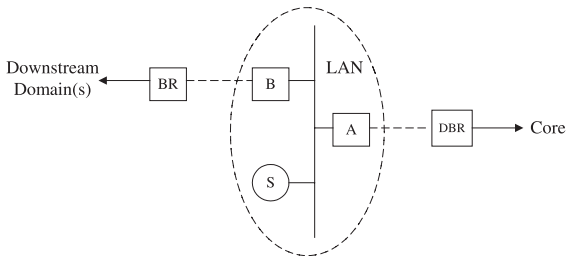


Fig. 6. Access control operation on LANs.

unauthorized host S sends data to the group. When the DR (router A) cannot find the corresponding entry in its local SACL, it immediately discards the packet and sends a "forbidding" packet containing the address of S to the LAN. Once the downstream router B receives the forbidding packet, it will stop forwarding data coming from S. If S sends data by maliciously using a different network prefix, both routers A and B will notice that the source address is not contained in their SACL list, or the data does not come from the correct interface; as such, they will not forward relevant packets which means that this type of malicious flooding will be only restricted to the local network. Nevertheless, the SACL mechanism is still not able to prevent flooding attacks by spoofing. For example, if S forges and uses the IP address of an authorized source on the same LAN, neither A nor B will be able to identify this type of IP spoofing. It should be noted though that even the FPM mechanism will not solve this problem either: relevant solutions should be implemented at lower levels and are outside the scope of this paper.

When inter-domain routing is concerned, further consideration is necessary for data traveling towards the core. This is because routers in transit domains do not have entries for remote senders in their SACLs. Also take Fig. 6 as an example, suppose that the LAN is located in a transit domain where there are no local authorized senders, and hence router A's SACL is empty. If there is data appearing on the LAN destined to the group address, there are only two possibilities: (1) the data came from a downstream domain and was forwarded to the LAN by router B; (2) a local unregistered host attached to the LAN (e.g., host S) sent the data. It is obvious that in the former case router A should pick up the packet and forward it towards the core, and for the latter, it should just discard the packet and send the corresponding "forbidding" packet to the LAN. This requires that the router is able to distinguish between packets coming from remote domains and packets coming from hosts directly attached to the LAN, which can be easily done by checking the source address prefix.

6. SACL scalability analysis

6.1. Simulation scenario

In this section we discuss scalability issues regarding router memory consumption. For simplicity we only discuss the situation of intra-domain routing here. Nevertheless, in inter-domain hierarchical sender access control the situation can be further improved. It is obvious that the *maximum* memory space needed in maintaining a SACL is $O(ks)$ where k is the number of multicast groups and s is the number of senders in the group. In fact, this is exactly the size of SACL in the core router. However, since on-tree routers need not keep the access policy for all sources but only for downstream senders, the average size of SACL in each on-tree router is significantly smaller.

We can regard the bi-directional shared tree as a hierarchical structure with the core at the top level, i.e., level 0. Since each of the on-tree routers adds its downstream senders to its local SACL, then the SACL size S of router i in the shared tree T can be expressed as follows:

$$S_i = \sum_{(i,j)\in T} S_j$$

and the average SACL size per on-tree router is

$$\overline{S} = \frac{\sum_{i=0}^{H} \sum_{j=1}^{L_i} S_j}{\sum_{i=1}^{n} Y_i},$$

where H is the number of hops from the farthest on-tree router (or maximum level) and L_i is the number of routers on level i, while

$$Y_i = \begin{cases} 1 & \text{if router } i \text{ is included in the shared tree,} \\ 0 & \text{otherwise.} \end{cases}$$

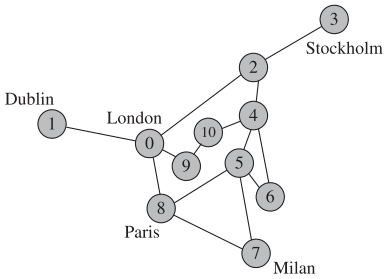


Fig. 7. Sprint IP backbone in Europe.

To ensure that the scalability was fairly evaluated through our simulation, random graphs with low average degrees, which represent the topologies of common point-to-point networks, e.g., NSFNET, were constructed. We adopted the commonly used Waxman's random graph generation algorithm [15] that has been implemented in GT-ITM for constructing our intra-domain network models. In addition, in order to evaluate the performance of the proposed mechanism at a larger scale, e.g., at Autonomous System level, we also conducted the simulation based on Sprint's European backbone topology (SprintLink [16]), which is an ideal AS-level simulation model. Within each AS, 100 routers are attached to each of the tier-1 node in the backbone (currently containing 11 data centers) and hence altogether 1100 nodes were involved in the AS-level simulation. The topology of the model is presented in Fig. 7 and in our simulation we assumed that the core router of the inter-domain bi-directional tree is located in London.

6.2. Intra-domain scalability performance

First we study the relationship between average SACL size and total number of senders. In the simulation we generate a random network with 100 routers with the core router also being randomly selected. The number of senders varies from 10 to 50 in steps of 10 while the group size is fixed at 50. We study three typical situations regarding the sending host type:

1. All senders are also receivers (AM);
2. 50% senders are also receivers (HM);
3. None of the senders are receivers (NM).

All send-only hosts choose to act as NMS without joining the bi-directional tree.

From Fig. 8 we can see that the average SACL size grows as the number of senders increases. However, we observe that even when the number of senders reaches a size as large as 50, the average SACL size is still very small (less than four on average). This is in significant contrast with the strategy of FPM on each router [2,14]. Further comparison between the two methods is presented in Table 1. From the figure we can also see that if all the senders are also receivers on the bi-directional tree (case AM), this results in a larger average SACL size. On the other hand, if none of the senders is a receiver (case NM), the corresponding SACL size is smaller. This phenomenon is expected because given the fixed number of receivers on the bi-directional tree as well as the sender group, the larger the proportion of senders coming from receiver set, the larger the resulting average SACL size. However this gap decreases with larger sender group size.

Next we study the effect on SACL size resulting from the senders' choice of acting as a send-only member (SOM) or a non-member sender (NMS). As we have mentioned, a host only wishing to send data to the group can decide to act as a SOM or NMS. Fig. 9 illustrates the relationship between the SACL size and total number of senders. The group

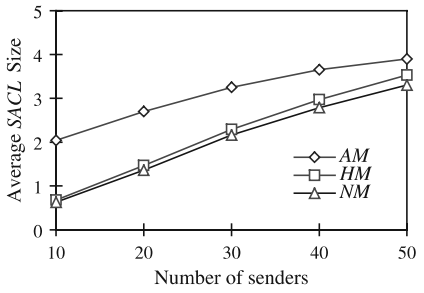


Fig. 8. SACL size vs. number of senders.

Table 1
Comparison with FPM (average SACL size)

S	10	20	30	40
FPM	10	20	30	40
SOM	0.65	1.27	1.82	2.3
NMS	0.73	1.4	2.09	2.73

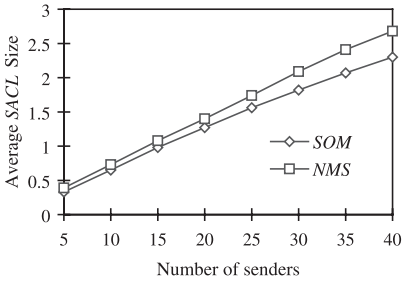


Fig. 9. SACL size vs. number of send-only hosts.

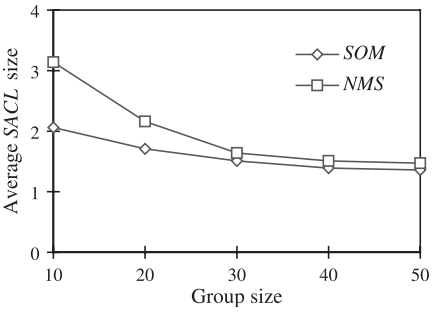


Fig. 10. Average SACL Size vs. group size.

size is fixed at 50 and the number of senders varies from 5 to 40 in steps of 5. It should be noted that in this simulation all group members are receive-only hosts and do not send any data to the group. From the figure we can see that the SACL size also grows with the increase of the number of senders. Moreover, if all the hosts join the bi-directional tree and act as Send-only member (SOM), the average SACL size is smaller. The reason for this is obvious: If the hosts choose to take the role of SOM, this will make the bi-directional tree expand for including the DRs of these senders. Since the number of on-tree routers grows while the total number of senders remains the same, the resulting average SACL size becomes smaller. On the other hand, if all of the hosts act as non-member sender (NMS), the figure of the shared tree does not change and no more on-tree routers are involved.

We continue to study the relationship between the average SACL size and the group size (number of receivers) with number of senders fixed at 20. We still let these senders choose to act as a SOM or NMS respectively. From Fig. 10 we can see that the SACL size decreases with the growth of the group size in both cases. On the other hand, a SOM join results in smaller average SACL size compared with a NMS one. The gap is more significant with fewer receivers. This is because if senders choose to act as SOM, they have to join the tree and generate many send-only branches, i.e., more routers are involved in the bi-directional tree. If the hosts just send data without becoming group members, the shared tree will not span to any of these senders, so the number of on-tree routers is independent of the number of senders. When the group size is small (e.g., 10 receivers), the size of the bi-directional tree will increase significantly to include all the senders if they join as SOMs, hence the gap is bigger for a small set of receivers.

We also present a comparison between our method and the "full policy maintainance" (FPM) strategy regarding a router's memory consumption. Table 1 gives the relationship of SACL size and total number of senders (S). From the table we can see that the length of the access list recorded in each on-tree router in the FPM mechanism is exactly the number of active senders. This imposes very big overhead on routers compared with our proposed scheme. It is inferred that the maximum number of SACL size of our proposed scheme is also the number of active senders (e.g., the core itself). However the number of such heavily burdened routers is significantly smaller than in FPM. To verify this, we conducted the simulation focusing on the SACL entry distribution in the whole network. In Fig. 11, the X-axis indicates the SACL size and the Y-axis shows the number of the routers with that number of SACL entries. We still take peer-to-peer applications as our example and the group size is fixed at 30. From the figure we can see that the proposed scheme imposes a very light memory burden on-tree routers, e.g., on average 31.7 out of 100 routers have a SACL size equal to 1, and only one router (the core) has the SACL size equal to the group size of 30; this is in consistent with the analysis presented above.

6.3. Inter-domain scalability performance

In this section, we present our SACL size analysis in the Inter-domain scenario based on the

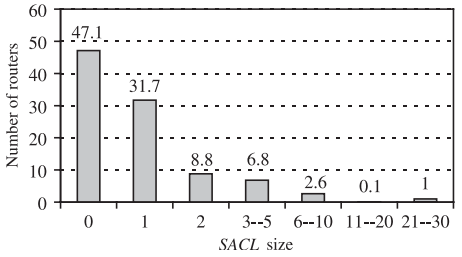


Fig. 11. SACL size distribution.

AS-level topology shown in Fig. 7. We compare the performance between the hierarchical approach we presented in Section 4 with the non-hierarchical one that forces each on-tree router to maintain SACL entries for all downstream senders including those in foreign domains. In this simulation we assume that all receivers are also sources, and we evaluate the SACL size performance with the variation of number of senders from 10 to 50 in each domain (altogether 110–550 senders). Fig. 12(a) illustrates the overall average SACL size for all the 11 domains. We notice that by using the hierarchical approach the average SACL size is reduced by 25.8%, since on-tree routers need only to maintain SACL entries for their local downstream sources. On the other hand, the difference between the two schemes can be most significantly reflected at the root domain, as it is shown in Fig. 12(b). If the hierarchical solution is adopted, the SACL size performance in the root domain is very similar to that in any other domain (shown in Fig. 12(a)). In contrast, we can observe that the SACL size in the non-hierarchical approach is much larger than that of the hierarchical one, e.g., when the number of senders in each domain is 50, the average size in the root domain is almost three times that of the hierarchical solution. This result is expected because all the inter-domain join requests will enter the root domain to reach the core, thus imposing a heavy burden to the routers near the core. Moreover, we can infer that the core router itself has to maintain 550 entries for all the sources in the non-hierarchical approach, while in the hierarchical one only 50 SACL entries are needed for the local sources.

In Section 4.2 we mentioned that if a sender comes from a domain without any group members, it should join the tree as a SOM. In this case a router needs not maintain SACL entries for remote sources, and hence the average router overhead can also be reduced. We consider the following scenario: suppose domains 3, 6 and 9 are sender-only domains that contain no group members, and there are 50 receivers in each of the rest domains respectively. Now we let the number of senders that are not receivers in each domain vary from 5 to 40 in steps of 5 (there are no SR-members in any domain), and we evaluate the performance between the choice of acting as SOMs and NMSs for these sources. In a similar fashion to the previous experiment, we consider both the average performance of all domains and the typical performance of the root domain. From Fig. 13(a) we can observe that the average performance of all the 11 domains is very similar to that of the intra-domain scenario. On the other hand, we also observe that the gap between SOM and NMS is much more obvious in the root

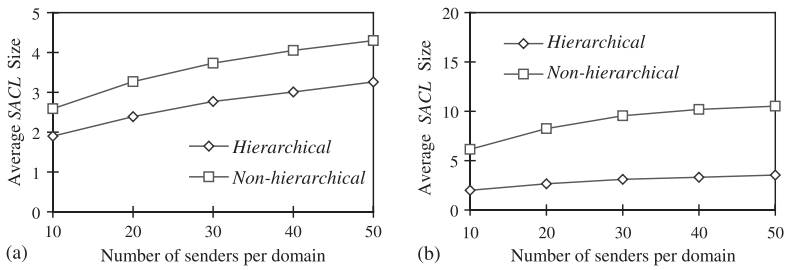


Fig. 12. SACL size vs. number of senders (inter-domain). (a) Average performance; (b) Root domain performance.

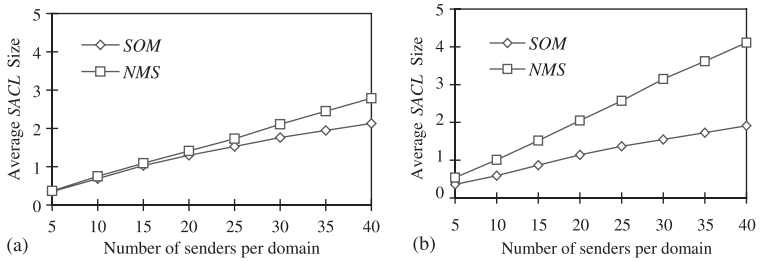


Fig. 13. SACL size vs. number of send-only hosts (inter-domain). (a) Average performance; (b) Root domain performance.

domain as it is shown in Fig. 13(b). When the total number of sending hosts reaches 40 per domain, the SACL size of SOM is only 47% that of NMS in domain 0. This result is also expected because the NMS schemes forces inter-domain SACL entry maintenance, and typically the on-tree routers need only to record SACL entries for all the hosts in remote send-only domains (domains 3, 6 and 9 in our simulation scenario).

Finally we evaluate the effect of the multicast group size on inter-domain SACL scalability, in a similar fashion to the intra-domain scenario. We fix the number of send-only hosts in each domain to be 20 and also assume that there are no SR-members, and the per-domain group size varies from 10 to 50. We still compare between the SOM and NMS cases. By comparing Figs. 10 and 14, we can observe that the performance of SOM and NMS in the inter-domain scenario are very similar to the intra-domain one: the gap between the two becomes less obvious as the group size grows in each domain. On the other hand, the inter-domain SACL size does not even increase in comparison to the intra-domain case presented in Fig. 10. Hence we can draw the conclusion that the proposed scheme scales well for inter-domain bi-directional trees due to the fact that sender access control is restricted within individual domains.

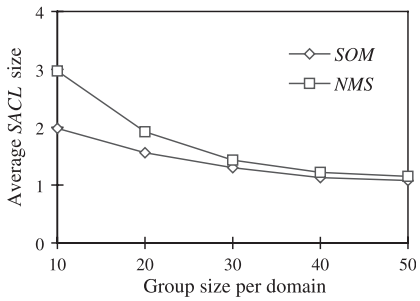


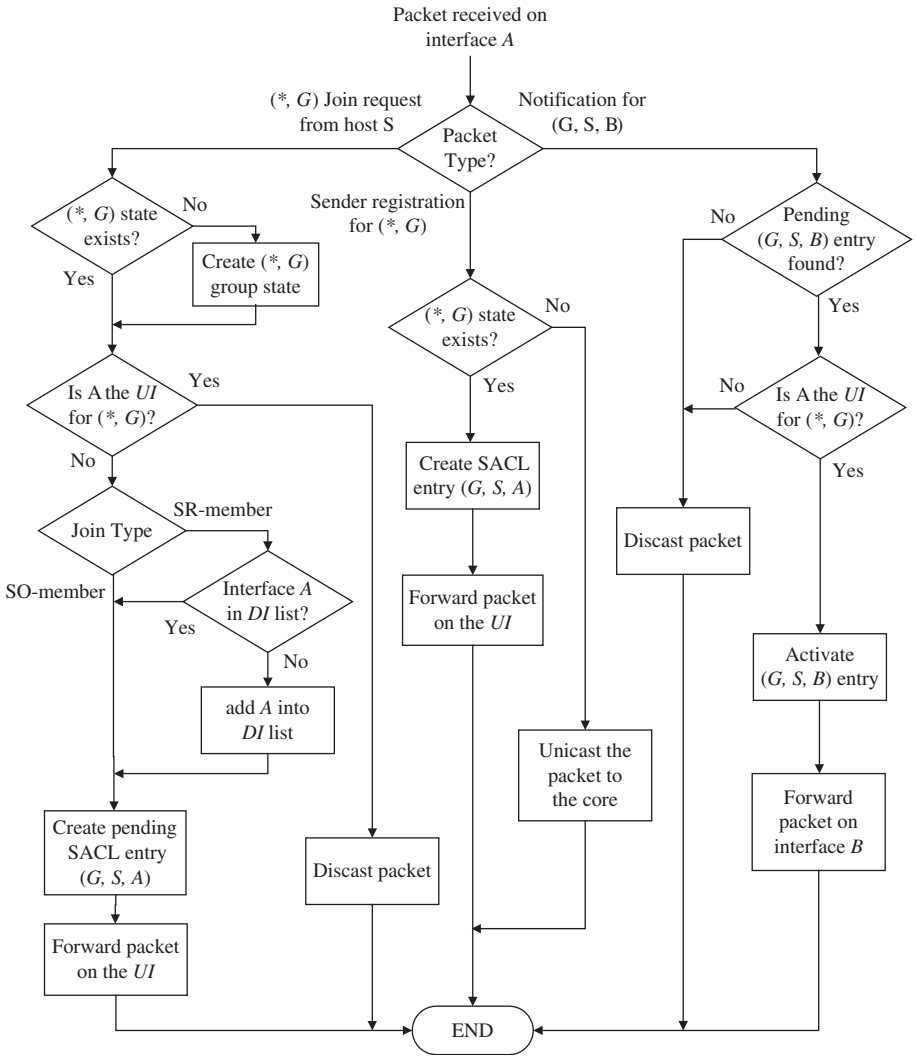
Fig. 14. Average SACL size vs. group size (Inter-domain).

7. Summary

In this paper we propose an efficient mechanism of sender access control for bi-directional multicast trees in the IP multicast service model. Each on-tree router maintains dynamically the access policy for its downstream senders. With this scheme, data packets from unauthorized hosts are discarded once they hit any on-tree router. As such, group members do not receive irrelevant data, and network service availability is guaranteed since the multicast tree is protected from deniel-of-service attacks such as data flooding from malicious hosts. In order to achieve scalability for large-scale multicast applications with many information sources and in order to accommodate more concurrent multicast sessions, we also extend our control mechanism to inter-domain routing where a hierarchical access policy is maintained on the bi-directional tree. Simulation results show that the memory overhead of our scheme is quite lightweight, resulting in good scalability even for inter-domain bi-directional multicast routing schemes.

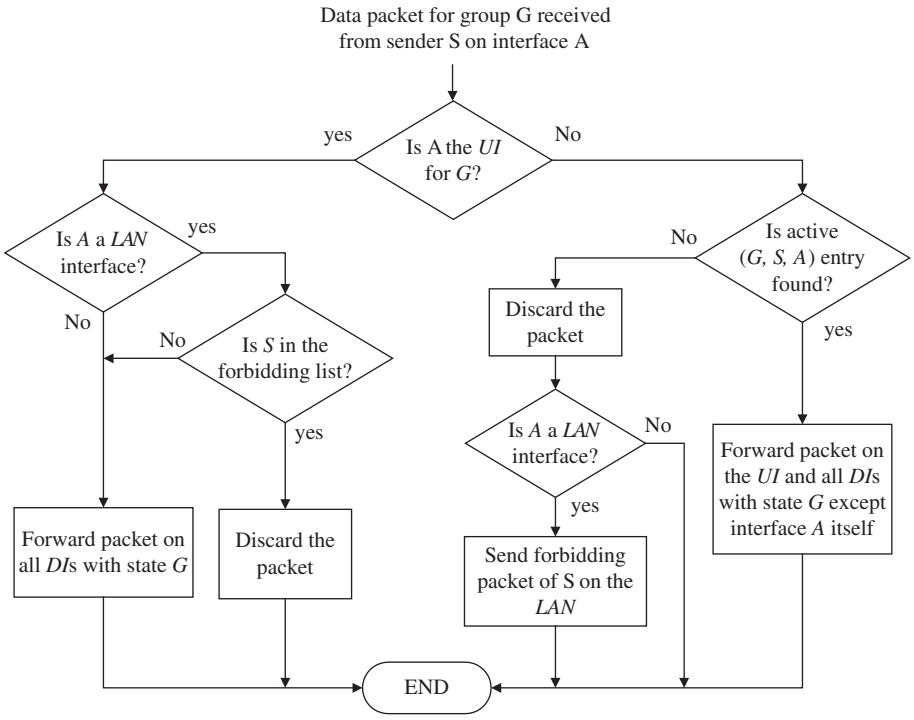
Nevertheless, this paper only provides a general paradigm for sender access control but does not present a solution for the restriction of sources based on the specific interest from individual receivers. Related works include [12,13] and this will be one of our future research directions.

Appendix A. Router actions for SACL management



- In the flow branch of handling notification packets, we use A to identify the interface from which the notification packet is received whereas B to identify a particular downstream interface with pending downstream SR/SO members.
- UI and DI stand for upstream/downstream interface respectively.

Appendix B. SACL-based data authentication and forwarding



References

[1] T. Ballardie, P. Francis, J. Crowcroft, Core based trees (CBT): an architecture for scalable multicast routing, in: Proc. SIGCOMM'93, pp. 85–95.
[2] B. Cain, Source access control for bidirectional trees, 43rd IETF Meeting, December 1998.
[3] B. Cain et al., Internet Group Management Protocol, Version 3, RFC 3376, October 2002.
[4] S. Deering et al., The PIM architecture for wide-area multicast routing, IEEE/ACM Transactions on Networking 4 (2) (1996) 153–162.
[5] S. Deering, Multicast routing in Internetworks and extended LANs, in: Proc. ACM SIGCOMM 1988, pp. 55–64.
[6] C. Diot et al., Deployment issues for the IP multicast service and architecture, IEEE Network, Jan./Feb. 2000, pp. 78–88.
[7] W. Fenner, Internet Group management Protocol, version 2, RFC 2236, November 1997.
[8] M. Handley et al., Bi-directional protocol independent multicast (BIDIR-PIM), Internet Draft, draft-ietf-pim-bidir-*.txt, June 2002, Work in progress.
[9] H.W. Holbrook, D.R. Cheriton, IP multicast channels: EXPRESS support for large-scale single-source applications, in: Proc. ACM SIGCOMM'99.
[10] H.W. Holbrook, B. Cain, Source-specific multicast for IP, Internet Draft, draft-holbrook-ssm-arch-*.txt, November 2002, Work in progress.
[11] S. Kummar et al., The MASC/BGMP architecture for inter-domain multicast routing, in: Proc. ACM SIGCOMM'99.
[12] B.N. Levine et al., Consideration of receiver interest for IP multicast delivery, in: Proc. IEEE INFOCOM, vol. 2, 2000, pp. 470–479.
[13] M. Oliveira et al., Router level filtering for receiver interest delivery, in: Proc. NGC' 2000.
[14] R. Perlman et al., Simple multicast: a design for simple, low-overhead multicast, Internet Draft, draft-perlman-simple-mulitcast-*.txt, October 1999.

[15] B.M. Waxman, Routing of multipoint connections, IEEE Journal on Selected Areas in Communications 6 (9) (1988) 1617–1622.
[16] SprintLink in Europe, http://www.sprintworldwide.com/maps/network/europe.html.

Ning Wang received his B.Eng degree from Changchun University of Science and Technology, China in 1996 and his M.Eng degree from Nanyang Technological University, Singapore in 2001 respectively. He is currently a PhD student in the Centre for Communication Systems Research (CCSR), Department of Electronic Engineering, University of Surrey, UK. His research interests include multicast communication, QoS provisioning, and Internet traffic engineering.

George Pavlou is a Professor of Communication and Information Systems at the Centre for Communication Systems Research, Department of Electronic Engineering, University of Surrey, UK, where he leads the activities of the Networks Research Group. He holds a BEng in Electrical and Mechanical Engineering from the National Technical University of Athens, Greece, and MSc and PhD degrees in Computer Science from University College London, UK. He is responsible for a number of European and UK research projects and industrial collaborations. His research interests include network planning and dimensioning, traffic engineering, multicast, mobile ad hoc networks, multimedia service control, programmable networks and communications middleware. He has published about 100 papers in refereed international conferences and journals and has contributed to standardization activities in ISO, ITU-T, TMF, OMG and IETF.