Proceedings 1998 IEEE International Conference on Parallel and Distributed Systems (Cat. No.98TB100250).
DOI: 10.1109/ICPADS.1998.741118

Secure and Scalable Inter-Domain Group Key Management for N-to-N Multicast

Thomas Hardjono and Brad Cain
Bay Networks
3 Federal Street, Mail Stop: BL3-03
Billerica, MA 01821, USA
Email: {thardjono,bcain}@baynetworks.com

Abstract

The current work contributes a new architecture for secure and scalable inter-domain group key management for N-to-N (conference) IP multicast. The architecture views the multicast routing infrastructure from the key management plane, and logically divides it into two general types of regions for key management to achieve scalability. The current work extends the centralized solution of Wong et al. (1998) into a distributed key management scheme suitable for inter-domain multicast group key management. Methods for initiating new multicast groups, as well as for members joining and leaving, are presented. The current paper also reasons two general types of IP-multicast that need to be made secure if multicast is to be one of the vehicles for future wide-scale delivery of voice, video and text over the Internet.

1. Introduction

The issue of the security of multicast has recently come to the foreground due to the increased interest in using multicast on the Internet as a vehicle of delivery of various data (eg. voice, image, text) to large numbers of users spread throughout the world. The MBone has provided a working example of multicast on a wide scale and has pointed to the enormous potential of multicast on the Internet. However, until an acceptable level of security can be assured for multicast transmissions, its use will remain limited from the commercial point of view.

The term “multicast” carries a variety of meanings to different people. In the current work the term multicast is taken to mean *IP multicast* as found in the definition of IPv4 and IPv6. This specific meaning of multicast implies that it is implemented through an infrastructure of network entities typically found on the Internet today, such as routers, switches and servers, together with their set of corresponding network protocols.

The IETF has provided good leadership in the effort to provide security standards for the Internet, by introducing the IPSEC standard for authentication and confidentiality and the ISAKMP/Oakley/IKE standards [1,2,3] for end-to-end key management. Although these technologies satisfy to a large extent the needs of secure communications in the Internet, they are aimed mainly at unicast transmissions between one sender and one receiver.

In the most basic form, the first step towards securing traffic within a multicast group is to provide a cryptographic key that is shared by the group members. Having such a key (and using IPSEC and its related technologies) allows group members to encipher the traffic within the multicast group. Thus, the group key also affords membership-enforcement by only allowing key holders to decipher the multicast traffic. Although group-authentication is implicitly provided through the possession of the key, sender-authentication must be provided through other means (eg. signature of individual sender).

In the next section some background is briefly provided, followed by a short discussion on the group key management requirements. This is then continued by the architecture and the overview of the protocols for key management in the face of membership changes.

2. Background

Group-oriented security, and more specifically the topic of its key management has been researched now for over two decades. Most of the earlier work have focused on cryptographic approaches to manage keys for hierarchic organizations and for conferences (eg. [4,5,6,7,8]). Others have sought different ways of

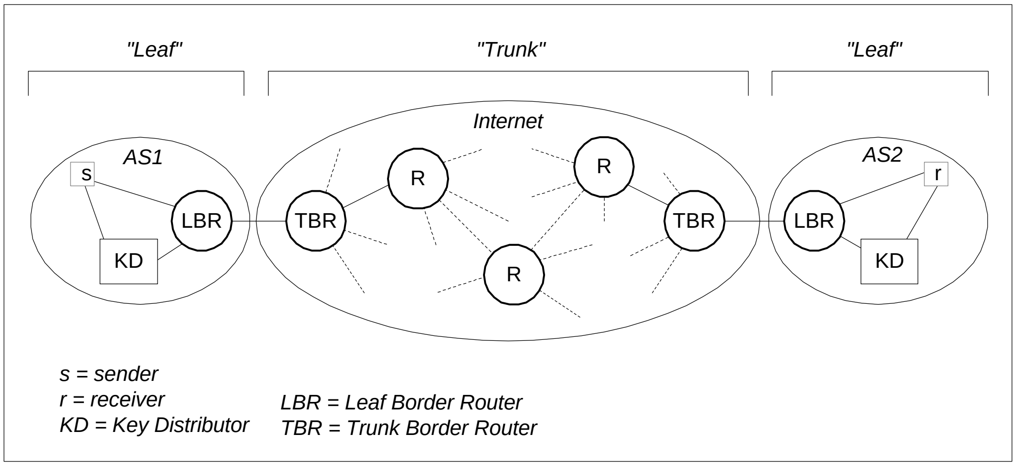

Figure 1: Basic model

sharing secrets within groups or to create digital multi-signature schemes (eg. [9,10]). Much of this research has been theoretical, and those which are more practical have not found an avenue of implementation on a wide scale.

In the context of practical group key management for IP multicasting, recently a number of proposals have been put forward [11-15]. These are known as *Group Key Management* (GKM) protocols. Most of these suffer from one drawback or another (eg. lack of scalability or dependence on multicast routing protocol).

The current work approaches practical group key management from the inter-domain perspective, in the spirit of the grouping proposal of [13] and modifying the key management solution suggested by [15]. A two-level division of the multicast routing structure is introduced in combination with a distributed key management approach, one which lends itself to be employed in conjunction with localized GKM protocols.

The current work distinguishes between 1-to-N multicast and N-to-N multicast. In the first case, transmission is unidirectional from 1 sender (assumed to be the multicast initiator) to N receivers. Examples of this type of multicast includes Pay Per View (PPV) multicast, transmission of passive data (eg. stock market data) and others. Typically, a 1-to-N multicast such as PPV carries monetary value in the sense that receivers are subscribers who must pay for the data transmitted. Hence, PPV multicast would be of interest to commercial companies seeking to use multicast to reach a wider market. In the N-to-N case, transmission occurs among N members, in which every member is both a sender and receiver. Here the initiator of the multicast instance is assumed to be one of the group members. From the perspective of security the initiator is assumed to determine the membership list of the multicast group. A typical example of this second type of multicast would be a conference among N members.

Distinguishing between the two types of multicasts has subtle implications relating to the placement of trust to the entities participating in the multicast. More specifically, in the case of a 1-to-N multicast the initiator/sender (eg. PPV producer) cannot rely on other entities to be honest. This is true particularly with respect to entities in the same organization as a receiver or is under the control of the receiver (eg. router in the subnet of a receiver). The initiator/sender would thus tend to select entities in the network upon which it can exert control.

In the case of an N-to-N multicast (eg. conference) the role distribution is more democratic since each member can both send and receive. It is in the interest of a member to maintain the security of the multicast (unless, of course, the member is dishonest from the start). Hence, it is in the interest of each member to select the most trustworthy entity under its jurisdiction to participate in the multicast infrastructure.

In the remainder of the work, focus will be given to the N-to-N multicast application type.

3. GKM Requirements

For an inter-domain group key management (GKM) scheme/protocol to be useful, there a number of general requirements that it must satisfy, from the points of view of security and multicasting:

- *Scalability*: The scheme must be scalable, taking into consideration the possible distribution of receivers (dense or sparse). In the current context of the use of group keys to afford membership-enforcement, and traffic confidentiality and authentication, it must address in particular the effects of the re-keying of group keys.

- *Independence*: The protocol must independent of the underlying multicast routing protocol.

- *Practicality*: The protocol should be implementable using existing practical internetworking technologies and security technologies. It should rely on current cryptographic algorithms that are considered to be standard algorithms by the industry and by standardization bodies.

- *Security*: The protocol must be shown to be secure. It must be designed using good and solid protocol design techniques and it must be proven secure using protocol analysis techniques.

- *Interoperable*: The protocol must be flexible enough to allow it to be used in conjunction with various multicast routing protocols, and if possible, with various locally-scoped (intra-domain) group key management schemes.

4. Architecture for Conference Multicast

In this section the architecture for secure inter-domain group key management for conference (N-to-N) multicast is presented. The basic model upon which it is based is first discussed, followed by a description of the relationships among the entities in the architecture. A distributed group key management scheme based on the work of [15] is then presented for the conference multicast.

4.1 Basic Model and Assumptions

The basic model that underlies the proposed architecture is shown in Figure 1. The model can be viewed from the following two perspectives:

- *Key management plane*: From the perspective of the key management plane the multicast routing topology is divided into a “trunk” region (backbone) and one or more “leaf” regions. A leaf may contain more than one receiver. A leaf is also defined to contain one or more multicast-capable routers, which for simplicity will be called *leaf-routers*. The trunk (backbone) covers the topology between all the leaf regions within the multicast group. The trunk by definition contains no sender/producer or receiver/consumer of the multicast traffic. In the context of the current work on conference multicast, a leaf is defined to be the size of an autonomous system (although generally speaking a leaf region need not necessarily be the size of an AS). For convenience, the terms “leaf region” and “AS” will be used interchangeably in the ensuing discussions.

- *Internet infrastructure plane*: The Internet is seen a collection of autonomous systems (AS), some being Stub ASs and others Transit ASs, connected to each other via Internet Service Providers (ISPs) and other backbone connections. Corresponding to the key management plane, for each leaf region several multicast-aware routers may exist. However, we assume that for any multicast instance only one multicast-capable router at the internal border of a leaf region handles all multicast traffic belonging to that multicast instance. We call such a router the *Leaf Border Router* (LBR). We also assume that an LBR in a leaf region is directly connected to a multicast-aware router in the trunk region. We call such an external border router as the *Trunk Border Router* (TBR). The TBRs may be owned, for example, by the Internet Service Providers (ISP). The LBR-TBR pair associated with a multicast instance is tightly-coupled and together represents the entry (departure) point for multicast traffic into (out of) the corresponding leaf region.

Each leaf region is associated with a *Key Distributor* (KD) which is implemented by one or more secure servers. The Key Distributor (KD) plays an important role in generating and distributing keys to the members of the multicast group. Hence, each leaf area is defined to have (at least) one KD which is assumed to be trusted by the entities in that leaf (AS). Several Key Distributors may reside within a leaf region, dynamically called-up for service as the needs arise. However, an important security requirement in this situation is that the available (trusted) Key Distributors be known in advance in order to reduce the possibility of masquerading.

In the remainder of the current work the term “secure channel” or “secure” communications means that it must provide sender authentication, data confidentiality and data integrity. Also implied are security measures against replay-attacks, such as message freshness mechanisms, timestamps and others. Each entity involved in any multicast instance is assumed to participate in a certification infrastructure which contains a *Certification Authority* (CA) that signs the individual certificates of the entities. The certificates must contain parameters necessary for authenticating an entity. Such parameters include the *Distinguished Name* (DN) of the entity, public-key of the entity (if it has one), life-time of the certificate, the organization of the entity, and others. Certificates represents an crucial element in creating security associations and in performing key exchanges that lead to the formation of secure channels. They are, therefore, an important part of the secure implementation of IPSEC and its related technologies (ie. ISAKMP/Oakley/IKE).

For the N-to-N multicast the multicast group advertisement (eg. via session directory) is defined to carry the identity and certificate of the IKD associated with the initiator. This allows other entities to communicate directly to the IKD.

4. 2 Key Generation and Distribution

When an initiator of a conference multicast wishes to start a multicast group, it must first select a KD to start-off the multicast. The initiator can either select the KD within its own leaf region or it can select another (eg. belonging to a trusted service provider or belonging to a trusted third party). For simplicity, the KD selected by the initiator will be called the *Initiator KD* (IKD). The leaf where the initiator resides will correspondingly be called the *Initiator AS*, while the other leafs will be referred to as *Remote ASs*. The Key Distributor in a Remote AS will be referred to as the *Remote KD* (RKD).

When an initiator host requests the creation of a new multicast instance, it notifies its IKD via a secure channel. The IKD then generates the multicast-key *Km* for that multicast instance and returns a copy of it securely to the initiator/sender. The initiator also requests its router to advertise the new multicast group (eg. via SDP) so that the multicast can be propagated throughout the Internet via the multicast routing protocol.

The multicast advertisement must carry not only the Distinguished Name (DN) [16] and IP address of the initiator/sender, but also that of the IKD (in addition to the usual parameters, such as the multicast group identity). This allows remote entities -- such as KDs and group members in remote ASs -- to create an IPSEC *Security Association* (SA) [2] with the IKD. The Security Association is the basis for secure channels, through which keys and other parameters can be exchange with confidentiality and authentication.

For each remote AS which contains a multicast group member, the IKD shares a subgroup-key with the RKD of that remote AS. Either the IKD or the RKD can generate and deliver the key to one another. The RKD will then securely deliver a copy of the subgroup-key to the group members in that remote AS.

For simplicity, we will refer to the subgroup-keys as *Kas1, Kas2, ..., KasN* corresponding to the *N* subgroups AS1, AS2, ..., ASN. For convenience we assume that the IKD resides in AS1. The multicast-key for a given multicast group is denoted by *Km* and is generated by the IKD. The distribution of keys is as follows:

- A member *i* within a subgroup ASj (*1 £ j £ N*) will have a private-key *Kji* which is shared with its local RKD (namely *KDj* in ASj). This key can be generated by the RKD and delivered securely to the member (or vice versa). In addition, the member *i* in ASj will be given by its key distributor KDj a copy of the multicast-key *Km* and a copy of the subgroup-key *Kasj* (*1 £ j £ N*).

- The Key Distributor KDj in ASj holds the subgroup-key *Kasj* and the multicast-key *Km*. It also shares the private-key *Kji* with each of its members *i* in ASj.

- The IKD holds a copy of the multicast-key *Km* (which it generates) and it holds a copy of all the subgroup-keys, namely keys *Kas1, Kas2, ..., KasN* corresponding to the *N* subgroups AS1, AS2, ..., ASN. Note, however, that the IKD does not hold copies of the private-keys of members outside its AS. Like other Key Distributors, the IKD only holds the private-keys of members strictly within its own AS1 (namely keys *K1i* of member *i*). This is in contrast to the approach in [15] in which a central server holds the private-keys of all members in the multicast group, which is somewhat burdensome on the server and makes that single server the main point of attack for intruders.

In the following, in all 1-to-1 communications we assume that a secure channel is employed, established through a *Security Association* (SA) between the two communicating parties and through the use of the necessary authentication and confidentiality mechanisms. The RKDs are assumed to have a pair-wise Security Association with the IKD set up before the commencement of any multicast.

4.3 New Multicast Group Initiation

The process of multicast initiation, and key generation and distribution is outlined in the following.

Step N1: The initiator in the Initiator AS notifies its LBR about the creation of a new multicast group. This is

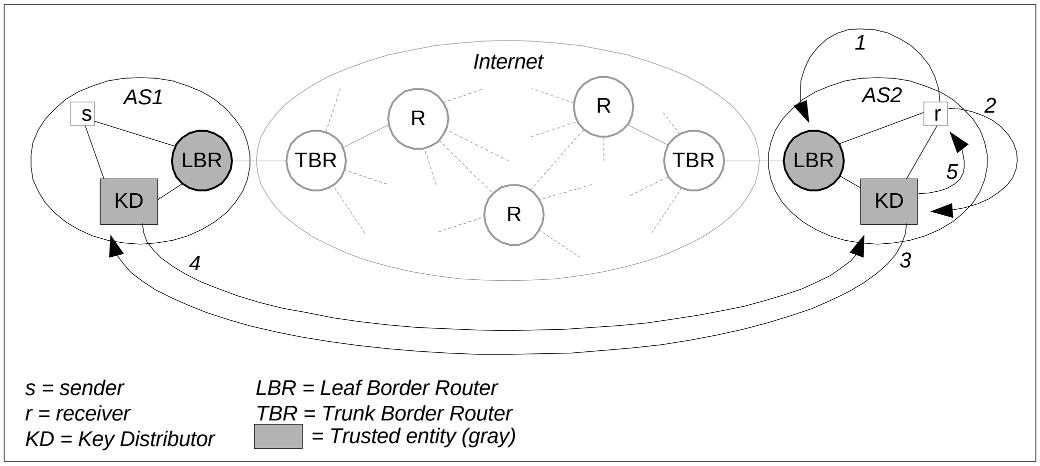

Figure 2: First member in a Remote AS

done typically through the underlying multicast routing protocol in conjunction with membership protocols, such as IGMP [17,18].

Step N2: The initiator notifies the IKD about the new multicast group and requests it to generate keys for the new multicast group. The initiator and IKD also establish a Security Association and a shared private-key known only to the two. The request from the initiator may carry an Access Control List (ACL) containing the Distinguished Name (DN) of all entities that can (cannot) join the multicast group.

Step N3: The IKD generates a multicast-key *Km* for that new group and also generates a subgroup-key *Kas1* for the Initiator AS (namely AS1).

4.4 First Member in a Remote AS

The first member in a given remote AS who joins a multicast group holds a special role in its AS in the sense that it triggers its local RKD to join the key structure emanating from the IKD at the Initiator AS. This candidate member is assumed to have found-out about the new multicast group through other means (eg. session directory as in MBone). In the following we assume that the candidate member is in AS2 while the source AS is AS1 (Figure 2):

Step F1: The host in an AS2 requests its LBR to join the multicast group and to "feed" it with the traffic from the multicast group. This may involve the use of a group membership protocol (ie. IGMP).

Step F2: The host requests its RKD to provide it with a copy of the multicast-key. The host and the RKD also establish a Security Association and a shared private-key.

Step F3: Not having a copy of the multicast-key associated with the multicast group, this RKD requests the IKD in AS1 (of that multicast group) to provide it with a copy of the multicast-key *Km* through a secure channel. Before doing so, the RKD generates a subgroup-key *Kas2* for its own AS2. The RKD then sends the request and a copy of the subgroup-key to the IKD in AS1 through a secure channel.

Step F4: The IKD receives the request accompanied with the new subgroup-key *Kas2*. The IKD stores the new subgroup-key and checks the candidate member against the access control list. If the candidate is allowed to join the multicast group the IKD provides a copy of the multicast-key *Km* to the requesting RKD through a secure channel.

Step F5: Upon receiving the multicast-key from the IKD, the RKD stores the key *Km* and provides the candidate member with a copy of the multicast-key *Km* and the subgroup-key *Kas2*.

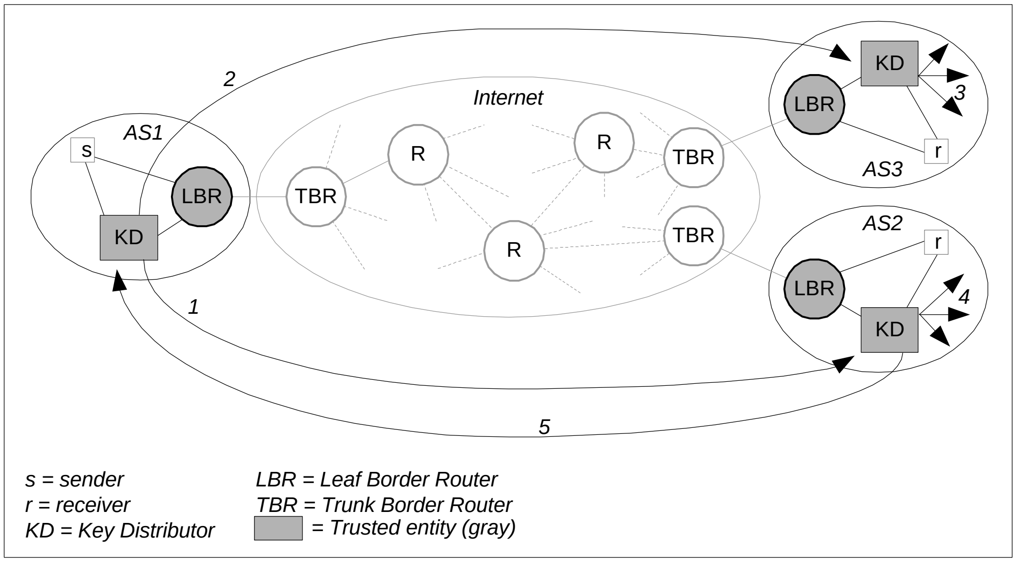

Figure 3: Member leaving

4.5 New Member Joining

There are a number of ways that a host in an AS can join an existing multicast group in that AS (which already has at least one member).

The most direct approach would be for the RKD in that AS (or IKD in AS1) to simply provide a copy of the necessary parameters to the host after access control checking has been carried-out. That is, the KD and the host first establish a Security Association and secure channel through which both can establish a shared private-key. Through the same secure channel the KD can then simply provide a copy of the multicast-key and subgroup-key to the host. This approach, however, may allow the host to decipher previous traffic (assuming the host is dishonest and has been intercepting previous multicast traffic in its AS).

In order to ensure that a host can only receive traffic generated from the time the host joins the multicast group, nothing short of a total re-key of the multicast-key must be carried-out. This is briefly outlined in the following scenario in which a host in an AS2 requests to join the multicast group:

Step J1: The host requests to receive the multicast traffic (eg. via IGMP) to its LBR in AS2.

Step J2: The host requests membership to its RKD and requests a copy of the multicast key. The RKD performs the necessary access control check, and the host and the RKD also establish a Security Association and a shared private-key.

Step J3: The RKD issues a join-request (on behalf of the new member) to the IKD and provides the IKD with the necessary parameters.

Step J4: The IKD performs the access control check, and if the candidate member passes the check the IKD initiates a re-keying of the multicast-key.

Step J5: After all the existing members (including those in AS2) have been re-keyed with the new multicast-key, the IKD issues a confirmation message to the requesting RKD in AS2. Note that in the event of a total re-keying of the multicast-key belonging to a given multicast group, all Key Distributors involved in the multicast group (ie. KDs in ASs that contain a group member) also obtain a copy of the new multicast-key.

Step J6: Upon receiving the confirmation message, the RKD in AS2 delivers a copy of the new multicast-key to the host (candidate member) through a secure channel. Only then does the host become a full member, since it can now decrypt all the traffic of the multicast group.

For total re-keying (Step J4), in the case of a new member joining the group, the IKD can simply multicast the new multicast-key to the existing members, after which the new member is admitted and is given a copy of the new multicast-key. Note that in multicasting the new multicast-key, the new key in enciphered under the existing ("old") multicast-key. This may or may not be acceptable to security practitioners as this may allow an attacker who has "broken" (compromised) the existing key (and is therefore passively listening to the multicast traffic) to illegally obtain the new multicast-key and to continue his/her interception of the traffic.

4.6 Decentralized Re-Keying

When a member of a multicast group leaves or is ejected from the group then a method must be employed to prevent that ex-member from listening (deciphering) further to the multicast-traffic. This in effect requires the total re-keying of the multicast key. However, unlike the case of a member joining, the IKD cannot encipher the new multicast-key under the existing multicast-key since the ex-member will still possesses that old key.

Another aspect related to re-keying the possibility that existing keys have been compromised, either by cryptanalysis or by other means (eg. keys stolen).

In the context of the current architecture the decentralized approach is the preferred method as it shares the task of re-keying among all the Key Distributors. This approach is a distributed version of [15] and is described in the following, in which we assume that the member is leaving AS2 (Figure 3):

Step L1: The IKD instructs the RKD in AS2 to terminate membership of a given host (or that host requests it directly to its RKD). The IKD also generates a new multicast-key and delivers a copy to the RKD in AS2 via a secure channel (unicast).

Step L2: The IKD re-keys the other ASs by delivering the new multicast-key to each of the other RKDs through a 1-to-1 (unicast) secure channel. This involves the new multicast-key being enciphered under a private-key which is pair-wise shared only by the IKD and a RKD.

Step L3: Each RKD (with the exception of AS2) would then provide a copy of the multicast-key to the members in its AS. This is achieved by the RKD encrypting the new multicast-key under the subgroup-key (of that AS) and multicasting the result to only its AS.

Step L4: In the case of AS2 (from which the ex-member left) the RKD must also generate a new subgroup-key. The RKD in AS2 must distribute the newly received multicast-key *and* a new subgroup-key to each remaining member through a 1-to-1 (unicast) secure channel to that member.

Step L5: The RKD in AS2 reports the completion of the re-keying in AS2 to the IKD and at the same time provides the IKD with a copy of new subgroup-key of AS2.

5. Brief Remarks

The current approach to inter-domain group key management relies on IPSEC (and its related technologies) to achieve a secure unicast between a sender and receiver. The choice of IPSEC is driven by the practical aspects of the problem of multicast security and from the fact of the wide availability of IPSEC implementations in the networking industry. The reliance on IPSEC is also driven by the belief that as demand for security (in IP multicast) increases, IPSEC will be the inevitable building block that the networking industry will employ to provide multicast security.

Another issue central to multicast security is that of trust-relationships among entities involved in the multicast. This issue is closely tied to the type of application in question (which in this case is a conference) and the control over entities, such the KDs and the routers. A conference based on an N-to-N multicast within a single enterprise or organization will differ in its trust-relationship specification from a conference across differing organizations. In the current work, we assume that each host-member trusts its KD and its LBR, and that the KDs collectively trust one another, reflecting more the single-enterprise application of the conference. We believe that the issue of conferencing among distrusting participants across organizations (eg. competitors) can only be solved at the Application layer using cryptographic schemes for conferencing (eg. based on smartcards [4,5]), in which our current group key management scheme is relegated to the position of only providing a host-to-host level security at the Network layer [19]. Solutions to reduce the risks involved in replicated KDs (eg. [20]) may also be employed to enhance our current approach.

6. Conclusion

The current work has proposed a new architecture for secure and scalable inter-domain group key management for conference (N-to-N) IP multicast. The N-to-N multicast consist of N members, each of which act both senders and receivers. The basis of the architecture consists of logically dividing the key management plane into two types of regions corresponding to the types of keys held by entities in a region, namely a "trunk" region and one or more "leaf" regions. This logical division simplifies the multiple-level approach of [13] and removes the need of border routers of the regions to carry-out multicast traffic “translations'” (decryptions followed by re-encryptions). Furthermore, the current work improves the centralized solution of [15] by modifying it into a distributed key management scheme suitable for inter-domain group key management for multicast security.

Acknowledgements

We thank Jim Luciani and Naganand Doraswamy for comments, insights and support.

References

[1] R. Atkinson, "Security architecture for the internet protocol," RFC 1825, IETF, August 1995.

[2] D. Maughan and M. Schertler, "Internet security association and key management protocol (ISAKMP)," July 1997. draft-ietf-ipsec-isakmp-08.txt available at http://www.ietf.org.

[3] D. Harkins and D. Carrel, "The internet key exchange (IKE)," March 1998. draft-ietf-ipsec-isakmp-oakley-07.txt available at http://www.ietf.org.

[4] I. Ingemarsson, D. T. Tang, and C. K. Wong, "A conference key distribution system," *IEEE Transactions on Information Theory*, vol. IT-28, no. 5, pp. 714--720, 1982.

[5] K. Koyama and K. Ohta, "Identity-based conference key distribution systems," in *Advances in Cryptology - CRYPTO'87* (LNCS No. 293) (C. Pomerance, ed.), pp. 175--184, Springer-Verlag, 1987.

[6] M. Burmester and Y. Desmedt, "A secure and efficient conference key distribution system," in *Advances in Cryptology - Proceedings of Eurocrypt'94* (LNCS No. 950), pp. 275--286, Springer-Verlag, 1994.

[7] M. Steiner, G. Tsudik, and M. Waidner, "Diffie-Hellman key distribution extended to group communications," in *Proceedings of the 3rd ACM Conference on Computer and Communications Security*, (New Delhi), ACM, March 1996.

[8] M. Burmester and Y. Desmedt, "Efficient and secure conference key-distribution," in *Security Protocols* (LNCS No. 1189) (M. Lomas, ed.), pp. 119--129, Springer-Verlag, 1996.

[9] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612--613, 1979.

[10] G. J. Simmons, "An introduction to shared secret and/or shared control schemes and their application," in *Contemporary Cryptology* (G. J. Simmons, ed.), pp. 441--497, IEEE Press, 1992.

[11] H. Harney and C. Muckenhirn, "Group key management protocol (GKMP) specification," RFC 2093, IETF, July 1997.

[12] T. Ballardie, "Scalable multicast key distribution," RFC 1949, IETF, 1996.

[13] S. Mittra, "The Iolus framework for scalable secure multicasting," in *Proceedings of ACM SIGCOMM'97*, pp. 277--288, ACM, 1997.

[14] D. Harkins and N. Doraswamy, "A secure scalable multicast key management protocol," November 1997. draft-ietf-ipsecond-00.txt.

[15] C. K. Wong, M. Gouda, and S. Lam, "Secure group communications using key graphs," in *Proceedings of ACM SIGCOMM'98*, ACM, September, 1998.

[16] C. Adams and S. Farrell, "Internet X.509 public key infrastructure certificate management protocols," March 1998. draft-ietf-pkix-ipki3cmp-07.txt available at http://www.ietf.org.

[17] S. Deering, "Host extensions for IP multicasting," RFC 1112, IETF, 1989.

[18] B. Cain, S. Deering, and A. Thyagarajan, "Internet group management protocol version 3," tech. rep., IETF, November 1997. draft-ietf-idmr-igmp-v3-00.txt. available at http://www.ietf.org.

[19] T. Hardjono, B. Cain and N. Doraswamy, “An Architecture for Conference-Support using Secured Multicast”, *Proceedings of the 8th IFIP Conference on High Performance Networking HPN’98*, Vienna, September 1998.

[20] L. Gong, “Increasing Availability and Security of Authentication Service”, *IEEE Journal on Selected Areas in Communications*, Vol. 11, No. 5, pp. 657-662, 1993.