
Security Issues and Solutions in Multicast Content Distribution: A Survey

Paul Judge and Mostafa Ammar, Georgia Institute of Technology

Abstract

Multicast enables efficient large-scale content distribution by providing an efficient transport mechanism for one-to-many and many-to-many communication. The very properties that make multicast attractive, however, also make it a challenging environment in which to provide content security. We show how the fundamental properties of the multicast paradigm cause security issues and vulnerabilities. We focus on four areas of research in security for multicast content distribution: receiver access control, group key management, multicast source authentication, and multicast fingerprinting. For each we explain the vulnerabilities, discuss the objectives of solutions, and survey work in the area. Also, we briefly highlight other security issues in multicast content distribution including source access control, secure multicast routing, and group policy specification. We then outline several future research directions.

Multicast enables efficient large-scale content distribution by providing an efficient transport mechanism for one-to-many and many-to-many communication. Over the years, multicast has been the topic of many research, engineering, and deployment efforts. These efforts have continued to transform multicast into a technology that can be relied on by many applications. Work has been done in reliability, manageability, scalability, quality of service, and ease of deployment. As these areas become more mature, there is increased potential for multicast to be used as the underlying distribution mechanism for content distribution applications. Therefore, security in multicast content distribution is a concern. The maturity of multicast security solutions have the potential to enable the use of multicast for confidential and high-value content, and help spark the use of multicast by new applications.

There are a number of security issues in multicast content distribution directly related to the properties of multicast that make it efficient and attractive. There has been research that provides solutions to many of these security issues. Some of these solutions are ready for deployment, some are nearing maturity, and others are only in the early phases of research. The maturity and deployment of these solutions will help increase the ability of multicast technology to deliver new applications and more content. In this article we examine these various issues and solutions for providing secure multicast content distribution, and outline several future research directions.

Properties of Multicast

The definition of the host group model [1] provides a summary of the key properties of multicast: “a host group is a set of network entities sharing a common identifying multicast address, all receiving any data packets addressed to this multicast address by senders (sources) that may or may not be members of the same group and have no knowledge of the groups’ membership.” This definition highlights the three main properties of multicast:

- All members receive all packets sent to the address: Multi-

cast routing delivers all packets sent to the multicast address to all members of the multicast group.

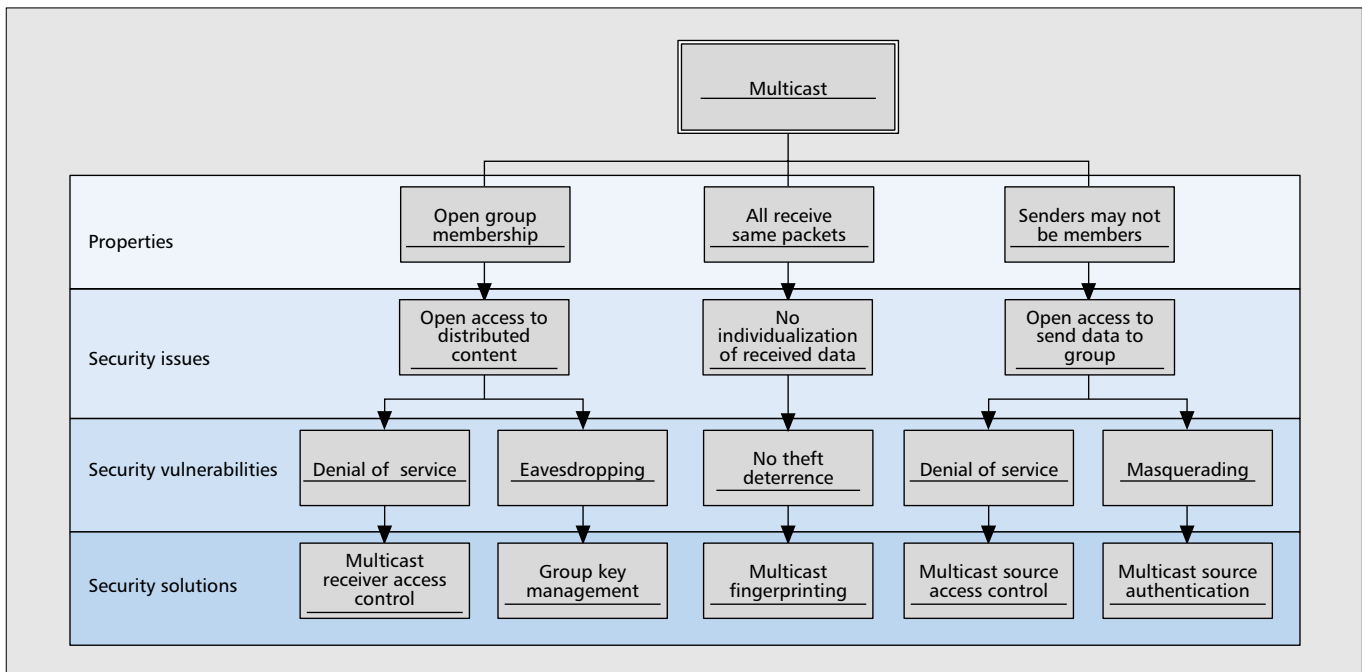
- Open group membership: Multicast provides an open group model and allows group membership to be transparent to the source.
- Open access to send packets to the group: Any host can send data to the multicast address, and it will be delivered to the multicast group without regard for the source of these packets.

We note that we focus here on the host-group native-IP multicast model, which allows so-called any source multicast (ASM) as the most general multicast model available. As such it also represents the most challenging context in which to provide content distribution security functions. Other multicast models provide more restrictive frameworks that may make it easier to deal with some security aspects. For example, in the small group multicast model the source needs to know the identity of the multicast group members. Another example is the use of source-specific multicast (SSM) in which only one source can transmit to a specific multicast group. Another example is application-layer multicast that utilizes an overlay network to implement multicast functionality including group management and packet forwarding. These more restrictive models, however, while possibly alleviating some aspect of securing multicast distribution, continue to possess other multicast properties (e.g., the lack of distinction of received data among the receivers); therefore, the security techniques surveyed here continue to be relevant.

Security Issues and Solutions

These properties of multicast lead to security issues and vulnerabilities because of two reasons: the issues are multicast-specific or the issues also exist in unicast, but the unicast solutions do not apply. Figure 1 shows how each of the three multicast properties leads to vulnerabilities and the areas of research that provide solutions to these issues.

The open group model is beneficial in many environments because it provides a lightweight join operation, the source is



■ Figure 1. Multicast security issues and solutions. Basic multicast properties lead to security issues. Each issue leads to security vulnerabilities. There is a body of work proposed to deal with each of these vulnerabilities.

not required to maintain state for all group members, and it allows some anonymization for group members. However, this same property of multicast also causes security issues since it is not possible to restrict communication to a set of authorized hosts. In the IP multicast model, any host can use the Internet Group Membership Protocol (IGMP) to become a member of any IP multicast group, possibly causing eavesdropping, theft of service, or denial of service. The latter attack can be caused by a malicious host joining a number of multicast groups thereby utilizing large amounts of bandwidth or router resources. To defend against these threats, two classes of solutions have been proposed: group data encryption with *group key management* and *multicast receiver access control*.

The multicast model delivers any traffic sent to the multicast address to the entire group. This means that any host can send data to the multicast group. This leads to two problems. First, group members need to be able to verify that messages received are from the intended source. *Multicast source authentication* solutions have been proposed to provide this functionality. Second, there should be mechanisms to restrict unauthorized sources from sending data to multicast groups due to the potential for denial-of-service attacks. *Multicast sender access control* solutions are necessary to defend against this threat.

The fact that all members receive all packets sent to the group is a fundamental feature and benefit of multicast; however, this property also causes some security mechanisms used in unicast to not work in multicast environments. One reason for this is that there is no individualization of the received data. Traditionally, this individualization has sometimes been used to provide security. For example, *fingerprinting* is the embedding of receiver identifying information in content to deter unauthorized duplication and propagation. However, fingerprinting techniques used in unicast environments do not work in multicast environments because all users receive the same data. Therefore, *multicast fingerprinting* solutions have been proposed to achieve unique fingerprinting in a multicast environment while maintaining the efficiency of multicast.

As stated above, most of these issues exist across the different multicast models. However, some of the multicast schemes may be immune to some of these issues due to their design. For example, single-source multicast inherently provides some

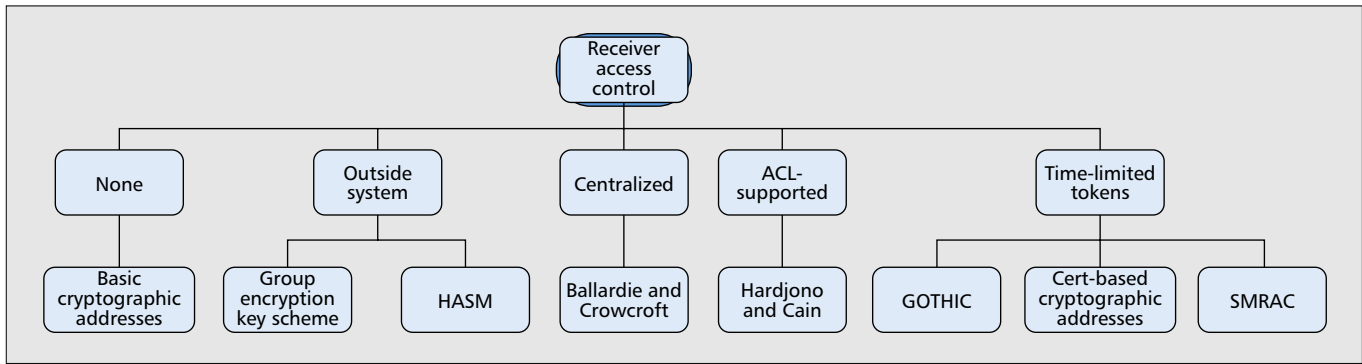
source access control since the group address is based on the source's unicast address. Small group multicast provides some receiver access control since the source knows the group membership. In application-layer multicast, the receiver access control problem differs since group management may not be based on IGMP.

In addition to the various models, multicast content distribution involves a number of potential environments composed of different Internet Protocol (IP) versions, routing protocols, address allocation schemes, and interdomain requirements. The security issues we discuss are relevant to these many flavors of multicast, but may vary slightly across the particular environments.

In this article we discuss these areas of multicast security research: receiver access control, group key management, source authentication, and multicast fingerprinting. For each we further explain the vulnerabilities it introduces, outline the objectives of solutions, and survey work in the area. Also, we briefly highlight other security issues in multicast content distribution including source access control, secure multicast routing, and group policy specification.

Receiver Access Control

There are a number of available multicast routing protocols that provide the efficient transport mechanisms of multicast by routing packets with one group destination address to multiple recipients. The routing protocols must be aware of group members in the network in order to deliver packets to them. The mechanism provided for doing this is the Internet Group Membership Protocol (IGMP). A host uses this protocol to notify the routing system that it should deliver packets for a particular multicast group to this host. In the current model, any host can use IGMP to become a member of any IP multicast group, causing eavesdropping or theft of service. The traditional method used to protect the information is to encrypt the multicast data and provide decryption keys only to authorized members. In some cases, encrypted communication is not possible for any number of reasons including legal issues or technical reasons. Even if encryption is used, there are still risks involved with unauthorized users receiving encrypted



■ Figure 2. Multicast receiver authorization systems.

data such as traffic analysis and possibly cryptanalysis. The current model is also vulnerable to a denial-of-service attack in which malicious hosts join a number of multicast groups, utilizing large amounts of bandwidth or router resources.

Solving these problems requires controlling the ability of hosts to join the multicast group. We call this *multicast receiver access control*. The need for a solution to these problems is well known. The term *secure IGMP* has been used to refer to the protocol that would provide the solution.

Objectives

There are certain functionality and components required of all solutions for multicast receiver access control.

Required Functionality — The functions necessary to provide controlled access to a group are as follows:

1. Group policy specification functions: These involve a host requesting to specify a *group policy*, authenticating the host, and verifying that the host is the *group owner*. The group policy is an access control policy that specifies among other things which hosts have access rights to become members. The group owner is the entity that has been assigned ownership of the multicast group and is allowed to specify the group policy.
2. Access request functions: These involve a host notifying the system that it wishes to become a member of a certain group.
3. Access control functions: These involve receiving a host's request, authenticating the host and performing *authorization*. Authorization requires checking the group policy to determine if that host has the access rights to become a member of the requested group.

Components — A multicast receiver access control architecture is composed of two systems: the *group policy management system* and the *group member authorization system*. The group policy management system performs group policy specification functions. The group member authorization system involves access request functions and access control functions. A multicast receiver access control architecture also interacts with the routing system and any group key management system that may be in place.

Group Policy Management System — The group policy management system involves a group owner providing the list of authorized members and possibly other security policy for the group to the access control server (ACS).

Group Member Authorization System — The group member authorization system provides the core functionality of group access control architecture by controlling access to the group. The design goals of an authorization system are to maintain security and achieve scalability. The main scalability objective

is to reduce the computational load on network routers. The second objective is to reduce message overhead.

Proposed Solutions

Figure 2 shows that multicast receiver authorization solutions can be classified based on how they provide revocations. Some systems do not provide revocation, some systems leverage the authorization state maintained by some outside system, some systems must query a centralized server to maintain authorization state, other systems distribute access control lists to routers, and some systems efficiently provide revocation using time-limited authorizations.

Hardjono and Cain: In [2], Hardjono and Cain present a method for delivering keys to enable IGMP authentication and suggest a method of authorizing group members. The authorization server provides capability-like access tokens to group members and access control list (ACL)-like token lists to the routers. The host sends a join request including the access token to the router, which verifies that the access-token is in the token list.

Ballardie and Crowcroft: In [3], Ballardie and Crowcroft provide an early survey of multicast security threats and present some countermeasures. Within the discussion, they present a version of IGMP that allows receivers to be authorized before joining the group. The architecture includes authorization servers that possess ACLs distributed by an initiator. The host sends a request to an authorization server to obtain an authorization stamp that is included in the join request sent to the router. The router forwards the host's request to the authorization server for approval.

GOTHIC: In [4], Judge and Ammar proposed GOTHIC, a comprehensive architecture for providing group access control. A host first requests a capability from the access control server and forwards it to the router along with the join request. The capabilities are identity-based and time-limited. The router host authenticates the host and verifies the capability before allowing the host to join the group. The authors discuss leveraging the state maintained by a group key management system by using the group encryption key as the access token. The authors also propose group access control aware group key management (GACA-GKM) that leverages the trust built into a group access control system to reduce the requirements of GKM and obtain substantial overhead reductions.

Source Authentication

Source authentication is the ability of group members to verify the identity of the sender of a received packet. In unicast, a shared secret key message authentication code (MAC) is used to provide authentication. In multicast, the group key provides a shared secret key; however, performing message authentication with this key only verifies that the sender is a member of the group, but not necessarily the intended source. Many applica-

tions require a level of authentication that allows a receiver to identify the individual sender of a message. There has been work that aims to efficiently provide this level of source authentication.

Objectives

The design objectives of a source authentication scheme should include the following:

- **Authenticity:** The receiver must be able to verify the identity of the data's source. One level of functionality is for the receiver to be able to verify that the data is from a group member. The next level of functionality is for the receiver to be able to verify that it is from an authorized sender. The most precise functionality is for the receiver to be able to determine the exact identity of the sender.
- **Integrity:** The receiver should be able to verify that the received data has not been modified. Some schemes provide only authentication without integrity checking.
- **Nonrepudiation:** Nonrepudiation requires the ability to prove that a host sent a particular message. This prevents the sender from later denying transmission of the message.
- **Efficiency:** The efficiency of the solution is based on communication, storage, and computation overhead at the source and receivers.
- **Collusion resistance:** The scheme should provide protection against collusion or at least be able to state in a provable manner the level of protection against collusion.
- **Minimal latency:** Some schemes require a certain number of packets to be stored before they can be signed or verified. For some real-time applications, this can introduce an intolerable delay.
- **Robustness against unreliable communication:** Some designs are based on an assumption of reliable communication. Some multicast environments do not provide reliable multicast communications; therefore, such schemes are unsuitable for these environments.

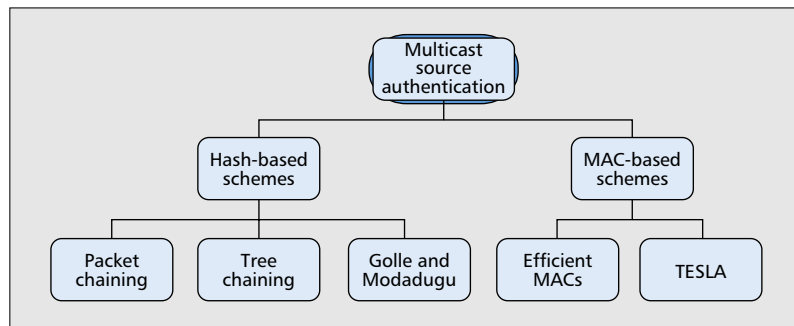
Proposed Solutions

As shown in Fig. 3, there have been two approaches in multicast source authentication schemes: hash-based and MAC-based.

Hash-Based Schemes — Digital signatures provide a simple method of individual authentication. However, due to the computational costs of computing and verifying digital signatures, signing each packet is not a practical solution.

Packet chaining: Gennaro and Rohatgi proposed packet chaining, a solution to efficiently authenticate digital streams [5] that allows only the first block to be signed and contain an association with subsequent packets. The stream of data packets is partitioned into chains, and each packet in the chain contains a hash of the next packet in the chain. Thus, only the first packet in the chain must be signed. This works for streams that are finite and in which the data is known in advance. For infinite streams, multiple one-time signatures are used.

Tree chaining: Wong and Lam [6] proposed tree chaining, a technique that partitions the stream of data packets into blocks and forms a tree structure to perform authentication. Each block of n messages can be authenticated with one signature. Each leaf node is a message digest of a data packet, and the parent nodes are message digests of the two children nodes. The root node is the message digest for the block, which is signed once for the entire group. To verify the packets the receiver recreates the path from the received packet up to the root, computes the digest of each node, and com-



■ Figure 3. Source authentication schemes.

pares the computed root to the signed received root.

Golle and Modadugu: Due to the association between packets, the above approaches are sensitive to data loss. Golle and Modadugu [7] proposed a hash-based scheme that aims to be robust against bursty packet loss. It achieves robustness by replicating packet signatures across multiple packets in the stream. The final packet also includes a signature. The authors provide results that show the burst tolerance of the scheme based on the efficiency resources.

Hybrid signatures: Rohatgi later proposed a scheme that uses public key digital signatures as well as faster one-way function-based k -time signatures [8]. The scheme creates sets of k -time key pairs offline and uses the normal digital signature to certify the public k -time keys. Message signatures are created online using a k -time private key and the certified k -time public key. The scheme avoids the need for reliable communication by sending the k -time keys more than once.

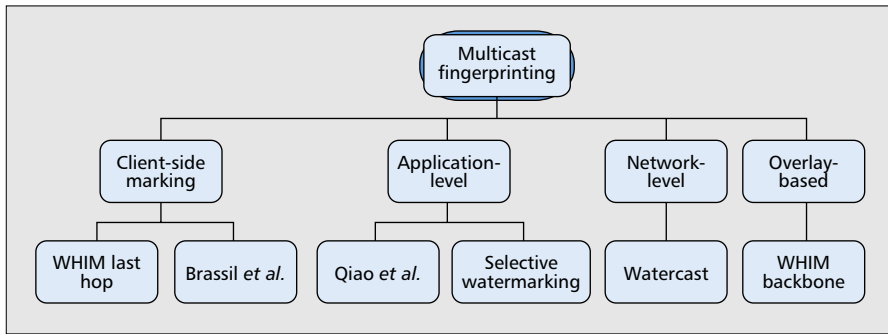
MAC-Based Schemes — There have been schemes proposed that use message authentication codes to provide authentication rather than digital signatures to increase efficiency.

Efficient MACs: Cannetti *et al.* proposed a scheme that makes use of efficient MACs [9]. In this scheme, the sender holds a set of l MAC keys, and each group member holds a subset of the l keys. Each message is then MACed with each of the l keys, and the recipient verifies the MAC with the keys it holds. The authors show that appropriate choice of subsets provides a high probability of protection against collusion.

TESLA: Perrig *et al.* proposed TESLA, a MAC-based scheme that provides authentication without regard for packet loss rate [10]. The scheme involves the source signing the first packet and providing notification of a chain of MAC keys. Each packet P_i is authenticated with a MAC using a key K_i . Later packets reveal each K_i . The scheme requires some time synchronization between the sender and the receivers since each packet must be received before the next packet is sent.

Multicast Fingerprinting

Encryption is generally used to safeguard content while it is being transmitted so that unauthorized persons cannot read the stream from the network, but this offers no protection after the intended receiver receives the data. There is no protection against unauthorized duplication and propagation by the intended receiver. Watermarking can provide protection in the form of *theft deterrence*. *Watermarking* is the embedding of some identifying information into the content in such a way that it cannot be removed by the user but can be extracted or read by the appropriate party. Watermarks can be used for copyright protection or for identification of the receiver. Copyright protection watermarks embed some information in the data to identify the copyright holder or content provider, while receiver-identifying watermarking, commonly referred to as *fingerprinting*, embeds information to identify the receiver of that copy of the content. Thus, if an unauthorized copy



■ Figure 4. Fingerprinting solutions.

of the content is recovered, extracting the fingerprint will show who the initial receiver was.

In multicast environments, traditional fingerprinting or embedding the receiver's identification as the watermark at the source will not work since all the receivers will share the same watermark. It is necessary to watermark content with unique information for distinct receivers of the same multicast stream. A simple method to achieve unique watermarks for each receiver would be to watermark the stream differently for each receiver and to unicast the watermarked streams. Of course, the inefficiency of such a scheme calls for a better solution. The goal is to maintain the security of this approach while achieving scalability.

Objectives

The design objectives of a system to fingerprint multicast content should be security and scalability. We outline the concepts involved in achieving these goals. The features and components of the system necessary to accomplish these goals should be designed into the solution.

Security:

Robustness of the fingerprinting method: The fingerprint is what distinguishes one user from another. This can be a particular pattern of frames or a particular pattern embedded in a frame. The method used must be robust to efforts of a user to remove this distinguishing information. There has been significant work in multimedia watermarking. A scheme extending these efforts into fingerprinting multicast content is desirable since it ensures a robust fingerprinting method.

Collusion problem: Collusion is when a set of group members work together to use the set of differently watermarked streams to create a copy of the content that cannot be determined to contain the fingerprint of any of those receivers. The solution must be based on a fingerprinting scheme that is not susceptible to collusion.

Asymmetric fingerprinting: Schemes should be able to provide asymmetric fingerprinting. This allows the sender to identify the receiver of a recovered copy of data without previously knowing the fingerprinted data. Thus, the sender is not capable of distributing the data and accusing an innocent receiver.

Protection granularity: The granularity of protection is the amount of content needed for the protocol to be able to determine the receiver of the content. Schemes should be able to provide the smallest possible protection granularity but also be flexible so that this can be changed depending on the needs of the application.

Scalability:

Logging requirements: Logging is necessary because once the content is recovered and the fingerprint is extracted, there must be some record of which receiver was represented by the ID recovered from the watermark at that instant in time. The storage and processing overhead of logging should be minimal.

Efficiency: The efficiency of the solution is based on the amount of data the source must transmit and encrypt, and the amount of data introduced into the network.

Proposed Solutions

Figure 4 shows that there have been four classes of multicast fingerprinting solutions depending on where the watermarking takes place. Client-side marking schemes involve some client software that watermarks the content.

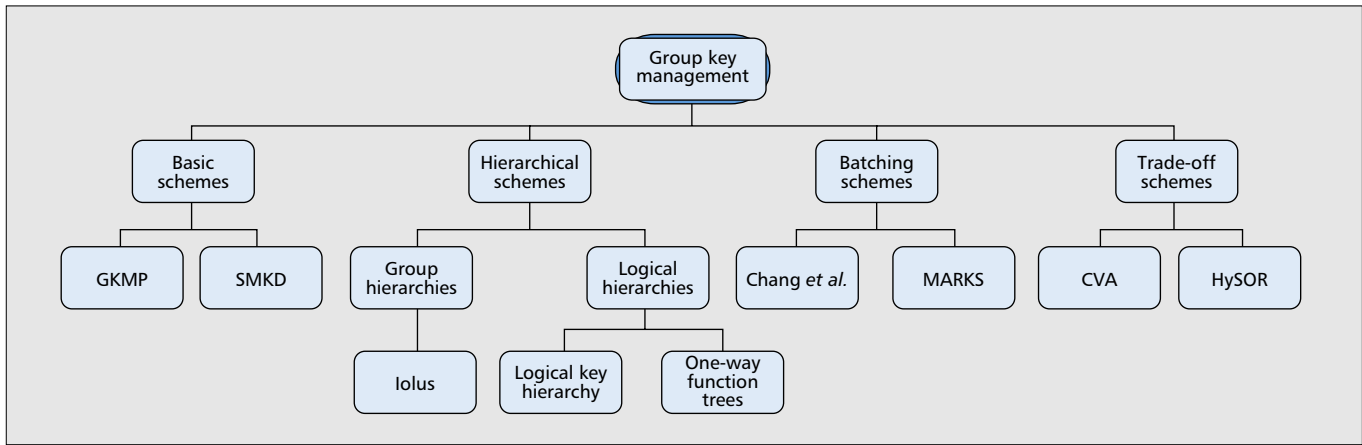
Application-level schemes add logic to the application to deliver unique versions of the content. Network-level schemes involve computation in the network that causes each user to receive a unique version of the content. Overlay-based schemes involve intermediaries in the content distribution path that uniquely watermark the content for receivers.

Application-based approach: Chu, Qiao, and Nahrstedt [11] proposed a protocol to provide a different version of a multicast video stream to each group member. The protocol creates two watermarked MPEG streams, assigns a unique random binary sequence to each user, and uses this sequence to arbitrate between those two watermarked streams. For the i th watermarked frame in stream j ($j = 0, 1$), a different key KEY_i^j is used to encrypt it. Then user n is given either KEY_i^0 or KEY_i^1 depending on the random bit sequence of user n .

Watercast: Brown, Perkins, and Crowcroft [12] proposed a technique that has each group member receive a slightly different version of the multicast video stream. For a multicast group with a tree of depth d , the source creates n differently watermarked copies of each packet such that $n > d$. On receiving a transmission group of packets, each router forwards all but one of the packets. The last hop router then forwards exactly one packet to the subnet with the receiver(s). The goal is that each receiver then receives a stream that consists of a unique combination of watermarked packets. The original receiver of a recovered stream can be determined by simulating the operation of various network components during the time the clip was originally transmitted.

Selective watermarking: Wu and Wu [13] proposed a technique that multicasts most of the video and uniquely watermarks and unicasts a portion of the video. Depending on the specific selection scheme used, the chosen segments could be from 90% to less than 1 percent of the original video. There is a trade-off between efficiency and security. As smaller amounts of the video are chosen for encryption and watermarking, the security of the scheme decreases. As larger percentages of the video are chosen to be watermarked, encrypted, and unicast, security increases, but the efficiency of the protocol begins to resemble that of the simple unicast model.

WHIM: Judge and Ammar proposed WHIM, an architecture for watermarking multicast content with a hierarchy of intermediaries [14]. The system has two components, WHIM backbone (WHIM-BB) and WHIM last hop (WHIM-LH). WHIM-BB introduces a hierarchy of intermediaries into the network and forms an overlay network between them. The unique path between the source and each intermediary on this overlay network is used to distinguish between intermediaries. Each path is identified by the IDs of the intermediaries on the path. Each intermediary embeds its portion of the path ID into the content as a watermark as it forwards the content through the network. WHIM-LH allows applications to identify the individual receivers by having intermediaries mark the content distinctly for any children receivers they might have. WHIM-LH uses a secure client-side fingerprint insertion pro-



■ Figure 5. Group key management solutions.

gram to form a domain-wide secure distribution and fingerprinting system including key distribution and logging.

Group Key Management

In unicast, two users can provide confidentiality by encrypting data with a shared key. In multicast, *group key encryption* is used in which the multicast traffic is encrypted with a symmetric key and every authorized member of the group is given the decryption key. This becomes complicated when group membership is dynamic. Upon a change in membership, it is often necessary to change the group key so that the leaving member cannot access new broadcasts or a new member cannot access old broadcasts. The term *leave* is used to describe the act of a voluntary or forced leave. It is necessary to reduce the cost of updating the group key in these situations. When a new member joins, the new group key can be sent to the original group members using the old group key. However, when a member leaves, the solution involves more work. The simplest approach is, upon each leave, to compute a new group key and send it to each user encrypted with its individual key. This is not acceptable because it requires n separate encryptions and transmissions for each join or leave. A simple improvement of this is to encrypt the new key with each user's individual key (resulting in n encryptions), but send all of the keys in one message to the entire group. This reduces transmission costs, but still requires n encryptions and causes the users to be able to detect their key among the group of keys in the received message. Work in *group key management* aims to provide efficient rekeying schemes for dynamic group memberships.

Work in GKM includes basic, hierarchical, batching, and trade-off schemes as shown in Fig. 5. Basic schemes include the early work in GKM and did not focus on efficient rekeying. Hierarchical schemes include the first attempts at reducing rekeying overhead. Batching schemes attempt to further reduce rekeying overhead by not changing the key on every join or leave, but instead batching a number of joins or leaves before rekeying. It has been generally accepted and recently proven that $O(\log(n))$ is the lowest overhead achievable by a GKM scheme if strict nonmember confidentiality and noncollusion are required. Trade-off schemes attempt to provide lower than $O(\log(n))$ overhead by trading off some collusion resistance. Due to space constraints, we do not discuss each of the proposed schemes here, but such a discussion can be found in other works [15, 16].

Related Areas of Research

Secure multicast routing: Shields and Garcia-Luna-Aceves [17] proposed keyed HIP (KHIP), a secure hierarchical multi-

cast routing protocol. The authors show that multicast routing protocols are vulnerable to attacks against the routing infrastructure and can cause denial of service by creating routing loops or blackholes. KHIP provides authentication mechanisms that allow only trusted routers to join the multicast tree.

There has been work that aims to add security mechanisms to the PIM-SM multicast routing protocol.

Sender access control: The problem of controlling which hosts can send data to a group is a separate problem from receiver access control. This is because IGMP is not used to register multicast senders.

In [3] a scheme to detect and prevent unauthorized multicast traffic is proposed. This scheme requires each packet to include a timestamp and an authorization stamp. Upon noticing multicast traffic from a new source, a router forwards a copy of the packet to the authorization service, which verifies that the authorization stamp was created by a host with the rights to send data for that particular multicast group and that the timestamp is current. If the verifications fail, the router is notified and is required to send an alert upstream toward the source in order to have all routers block traffic from the unauthorized source.

One viewpoint is that sender access control is becoming less of a problem with recent multicast schemes such as SSM that inherently provide sender access control. Recent discussions within the Internet Engineering Task Force (IETF) have maintained that receiver and sender access control should be solved separately, but have considered a scheme similar to secure IGMP for sender access control.

Group security policy: Multicast group policy is an important element of securing multicast content distribution. It deals with specifying the parameters and mechanisms involved with securing the group. There has been work that presented requirements for policy management in secure groups. This work explains that requirements include the specification, distribution, evaluation, and enforcement of policy.

Another problem in group security policy is verifying the entity that is allowed to specify the group's policy. This entity is usually the group owner, but determining who this is and authenticating an entity to be the group owner can be a complex task. In [4] the authors examine this problem and propose two solutions for a *group owner determination and authentication system* (GODAS).

Conclusions

In this article we outline the various security and protection issues in multicast content distribution. We focus on four areas of work, explain the issues and vulnerabilities that exist, and discuss the research that has been done to provide solu-

tions. We also briefly discuss some related areas.

Security in multicast content distribution has matured over the years, but there remain open problems in the area that must be resolved to help multicast enable more applications.

Group key management and multicast receiver access control: Although significant work has been done on GKM, the proposed schemes are based on environments that lack multicast receiver access control. It has been shown that group access control-aware GKM schemes can greatly improve efficiency [4]. Further research is needed to develop such schemes and to better analyze the performance improvements based on real-world conditions.

Group key management for unreliable communications environments: Most work in GKM has focused on reducing the overhead for rekeying, and these schemes have assumed the presence of reliable multicast. While there have been advances in reliable multicast, in many environments it is not used. Development of efficient GKM schemes that can operate without reliable multicast is an important problem for future research.

Sender access control: The issue of unauthorized hosts sending data to the multicast group leads to a number of vulnerabilities, as we have shown. However, there have been few solutions proposed for this problem. Thus, this remains an area for further research.

Multicast routing security: There have been security mechanisms proposed for certain multicast routing protocols, but some of the most widely deployed protocols lack security mechanisms. Further research is needed to develop security mechanisms for common multicast routing protocols or a general scheme that is portable across protocols. There has been significant work in secure unicast routing protocols that can be leveraged.

Group rights management: Beyond protecting the infrastructure and delivering content to only authorized hosts, there is other control and protection functionality that may be desired in multicast content distribution. Rights management allows a group controller to go beyond defining authorized and unauthorized hosts to be able to define different types of access rights for different users. There has been work in digital rights management for unicast environments. Further research is needed to develop efficient and flexible group rights management schemes.

References

- [1] D. Cheriton and S. Deering, "Host Groups: A Multicast Extension for Datagram Internetworks," *Data Commun. Symp.*, Sept. 1985, pp. 172-79.
- [2] T. Hardjono and B. Cain, "Key Establishment for IGMP Authentication in IP Multicast," *IEEE ECUMN*, CREF, Colmar, France, 2000.
- [3] A. Ballardie and J. Crowcroft, "Multicast-Specific Security Threats and Countermeasures," *Proc. ISOC Symp. Net. and Distrib. Sys. Sec.*, San Diego, CA, Feb. 1995, pp. 2-16.
- [4] P. Q. Judge and M. H. Ammar, "Gothic: Group Access Control Architecture for Secure Multicast and Anycast," *IEEE INFOCOM*, July 2002.
- [5] R. Gennaro and P. Rohatgi, "How to Sign Digital Streams," *LNCS*, vol. 1294, 1997.
- [6] C. Wong and S. Lam, "Digital Signatures for Flows and Multicasts," *IEEE/ACM Trans. Net.*, vol. 7, 1999.
- [7] P. Golle and N. Modadugu, "Authenticating Streamed Data in the Presence of Random Packet Loss," *Net. and Distrib. Sys. Sec. Symp.*, 2001.
- [8] P. Rohatgi, "A Compact and Fast Hybrid Signature Scheme for Multicast Packet Authentication," *ACM Conf. Comp. and Commun. Sec.*, Nov. 1999.
- [9] R. Canetti et al., "Multicast Security: A Taxonomy and Efficient Constructions," *IEEE INFOCOM*, New York, NY, Mar. 1999.
- [10] A. Perrig et al., "Efficient and Secure Source Authentication for Multicast," *Net. and Distrib. Sys. Sec. Symp.*, Feb. 2001.
- [11] H. Chu, L. Qiao, and K. Nahrstedt, "A Secure Multicast Protocol with Copyright Protection," *Proc. IS&T/SPIE's Symp. Elect. Imaging: Sci. and Tech.*, Jan. 1999.
- [12] I. Brown, C. Perkins, and J. Crowcroft, "Watercasting: Distributed Watermarking of Multicast Media," *Networked Group Commun. '99*, Pisa, Italy, Nov. 1999, pp. 286-300.
- [13] T. Wu and S. Wu, "Selective Encryption and Watermarking of mpeg

Video," tech. rep., NC State Univ.

- [14] P. Q. Judge and M. H. Ammar, "WHIM: Watermarking Multicast Video with a Hierarchy of Intermediaries," *Proc. NOSSDAV*, Chapel Hill, NC, June 2000.
- [15] M. Moyer, J. Rao, and P. Rohatgi, "A Survey of Security Issues in Multicast Communications," *IEEE Network*, vol. 13, Nov.-Dec. 1999, pp. 12-23.
- [16] P. Judge, "Security and Protection Architectures for Large-Scale Content Distribution," Ph.D. thesis, Georgia Tech, Atlanta, GA, Dec. 2002.
- [17] C. Shields and J. J. Garcia-Luna-Aceves, "KHIP - A Scalable Protocol for Secure Multicast Routing," *SIGCOMM*, 1999, pp. 53-64.

Biographies

MOSTAFA AMMAR [F] (ammar@cc.gatech.edu) is currently a professor in the College of Computing at Georgia Tech. He received S. B. and S. M. degrees from MIT in 1978 and 1980, respectively, and a Ph. D. in electrical engineering from the University of Waterloo, Ontario, Canada, in 1985. His research interests are in the areas of computer network architectures and protocols, distributed computing systems, and performance evaluation. He is co-author of *Fundamentals of Telecommunication Networks* (Wiley, 1994). He was also co-guest editor of the April 1997 issue of *IEEE Journal on Selected Areas in Communications* on network support for multipoint communication. He was Technical Program Co-Chair for the 1997 IEEE International Conference on Network Protocols and the 2002 Networked Group Communication Workshop. He currently serves as Editor-in-Chief of *IEEE/ACM Transactions on Networking*. He is a member of ACM and of the Association of Professional Engineers of the Province of Ontario, Canada.

PAUL Q. JUDGE [M] (judge@cc.gatech.edu) received a B. S. degree in computer science from Morehouse College in 1998. He received his M. S. and Ph. D. in computer science from the Georgia Institute of Technology in 2000 and 2002, respectively. His research interests include computer networking and information security. He is a member of the ACM and Phi Beta Kappa.