Multiple Video Channel Transmission using Secure IP Multicast

António Pinto
Escola Superior de Tecnologia e Gestão de Felgueiras, Instituto Politécnico do Porto, Portugal
Fac. Eng. Univ. Porto, Portugal
Phone: +351-25 531 4002, e-mail: apinto@estgf.ipp.pt

Manuel Ricardo
Fac. Eng. Univ. Porto, Portugal
INESC Porto, Portugal
Phone: +351-22 209 4267, e-mail: mricardo@inescporto.pt

Abstract — **Evolution is making telecom networks to converge towards all-IP communication scenarios where multiple services, including the broadcast of streaming video, are transported by IP packets. A possible solution for supporting multiple video channel transmissions with individual channel access control is by using secure IP multicast.**

In this paper the authors propose a solution for supporting the broadcast of multiple video channels which can be accessed only by authorized users; besides, and when a video channel is not visualized in the last mile, it may be temporarily stopped so that the cable can be used for other services, such as standard internet access.

I. INTRODUCTION

The technological evolution is leading telecommunications to all-IP network scenarios, where multiple services are transported as IP packets. Among these services is the broadcast of video, which is expected to reach users with quality and security better than those available today in analog cable TV. A solution for addressing the security of broadcast video channels in these communication scenarios is to use secure IP multicast [1], [2], [3] and [4]; in this solution, only the users having the correct cryptographic key can access and decode a video channel. The standard secure IP multicast solution uses symmetric encryption techniques, which generally lead to a single cryptographic key for all the group members; besides, it requires group key renewal when the group composition changes. Examples of using secure IP and the video broadcast context are given in [5], [6], and [7].

The new problems addressed in this paper are twofold: 1) we would like each user to have a set of keys, one for each subscribed channel, that are unique and not shared with the other members of a group; 2) if a given channel is not being viewed by any receiver in a network area, it shall not be transmitted; in this way, the bandwidth can be freed and used by other services. The solution proposed in this paper solves these problems and, to best of our knowledge, is new.

Section 2 defines some concepts. Section 3 provides the full set of requirements which characterize our problem and provides guidance for solving them. Section 4 presents the proposed solution, both from the architecture and the functional points of view. Section 5 concludes de paper.

II. DEFINITIONS

Channel - A channel is an audio and video composite stream, transported as Secure Real-time Transfer Protocol (SRTP) [1] over UDP/IP packets, and sent to an IP multicast address.

Group - A group is a set of receivers having a valid cryptographic key, which is used to decrypt the SRTP stream, that is, the channel.

Authorized User – A user is authorized if he has a valid contract with a video service provider. In this case, the latter, grants the user access to one or more video channels.

Permission – A clearance given by the video service provider to the user, which enables the user to access and decode a video channel.

III. REQUIREMENTS

The scenario considered assumes the transmission of multiple video channels. Each channel can be viewed only by authorized users. When the channel is not being visualized by any user in some last mile network area, it is not transmitted. For this scenario, the following requirements were identified:

1- Encrypted data transmission;
2- Unique user identification and authentication;
3- Unique channel identification;
4- Access control by user and channel;
5- Channel source authentication;
6- Group transmission.
7- Transmit only if required

The requirements 1 and 6 can be satisfied by adopting secure multicast; they intend to prevent video channel access by unauthorized users. Requirements 2 and 3 can be met by using symmetric cryptographic techniques. Requirement 5 suggests the adoption of asymmetric cryptography techniques, by letting the sources signing the transmitted data. The main problem seems to be the satisfaction of requirement 4; it aims at controlling individual user access in a video channel basis. The reduction of used bandwidth in a last mile network area is the objective of requirement 7.

In order to better understand these requirements and help the development of a solution, the following scenarios were considered during system specification: 1) an unauthorized user tries to access a video channel; 2) an authorized user tries to access a video channel for which he has no permission; 3) an authorized user tries to access a video channel for which he has permission.

The requirement 4 can be satisfied using an algorithm based on the multicast group access control mechanisms described in [8] and [9]; there, each user periodically receives a cryptographic key which enables him the access the video channel; this channel is encrypted with another key that was initially negotiated with the group controller or key server.

IV. PROPOSED SOLUTION

A. Framework

Figure 1 depicts the elements involved in the proposed solution: the Multicast Deflector, the Sender and the Receiver.

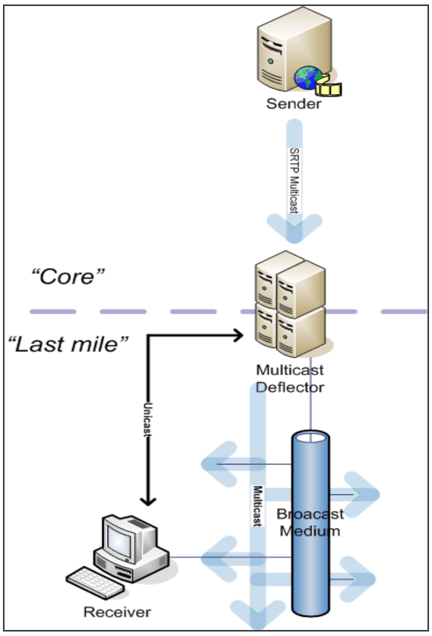

Figure 1 - Framework of the proposed solution

The Sender component is a standard SRTP server which has the responsibility of transforming an audio and video into an SRTP stream, and sending it to an IP multicast address. The Receiver component consists of two main functions: 1) receiving the SRTP multicast streams, and 2) performing the operations related to group joining and cryptographic key management.

The Multicast Deflector is the key component of our framework. It can authenticate the clients (Receivers), decide what multicast SRTP streams are forwarded to the last mile access network, and distribute cryptographic keys.

By placing the Multicast Deflector on the edge of the network, we enable the selection of the multicast SRTP streams to be retransmitted to the last mile network, according to the Receiver requests. When, for instance, a Receiver decides to view one of the video channels available, this will imply a channel request to the Multicast Deflector that, then, starts sending the channel. In order to avoid large delays, we assume the Multicast Deflector is always receiving all the streams.

B. Multicast Deflector Architecture

The Multicast Deflector Architecture is depicted in Figure 2; it consists of 4 main modules: Multicast Deflection Engine, SRTP, KDP, and IMGP.

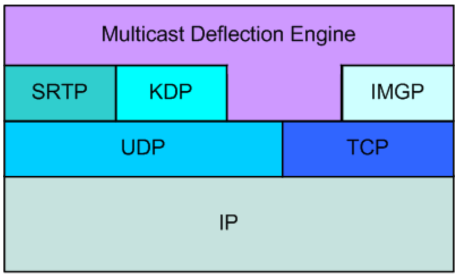

Figure 2 - Multicast Deflector Architecture

The Implicitly Managed Group Protocol (IMGP) enables the Receivers to join a group (i.e., to view a channel) and to leave a group implicitly. The IMGP protocol consists of a simple 2-message exchange: a group access request message, followed by either an acknowledgement or un-acknowledgment message. The acknowledgement message contains the channel access cryptographic key (the key encryption key), and a time-to-live (TTL) for that key. Prior to the expiration of the TTL and if the Receiver is interested in decoding the channel, it repeats the 2-message exchange procedure in order to obtain a new key, which is associated to a new TTL. If the request is not renewed, an implicit group leave is assumed for this Receiver and the channel transmission is ended. This protocol eases the management of situations where multiple channel access requests from the same Receiver are received, such as when a user starts *zapping* the channels. The IMGP is implemented over a unicast connection (see Figure 1), which is maintained until the Receiver is turned off. The information sent through this unicast connection is encrypted with a pre-shared symmetric cryptographic key. The IMGP module implicitly authenticates the Receivers, since their requests can be decrypted correctly only if they possess the Receiver pre-shared key.

The KDP module implements the Key Distribution Protocol, which is used in cryptographic key distribution operations and enables individual user access control to the distributed channel decryption keys. Individual Receiver access control is accomplished by distributing several copies of the cryptographic key, which allows the decryption of the SRTP stream. This stream is encrypted with the key encryption keys (obtained with the IMGP) of the existing Receivers. This protocol is based on the Iolus framework described in [8] and on the Secure Multicast Protocol with Copyright Protection described in [9]. The KDP messages are sent over IP multicast, as shown in Figure 1, and are signed digitally with the Multicast Deflector public key.

The SRTP module acts as both SRTP server and client. It receives an SRTP stream from the original server, decrypts it, encrypts it again with a new local symmetric cryptographic key, and sends the stream to the last mile network using IP multicast. The new local cryptographic key used for each channel is periodically obtained from the Multicast Deflector Engine. The cryptographic keys used to decrypt the original SRTP streams are pre-shared cryptographic keys.

The Multicast Deflection Engine can be seen has the module that controls the modules presented above (IMGP, KDP, and SRTP). For instance, the Multicast Deflection Engine is responsible for instructing the SRTP module to start or stop transmitting a certain channel, depending on the channel access requests sent by the Receivers. It is also responsible for the management of the TTL for the several channels and to set the cryptographic key used by the SRTP module in the last mile network.

The modular specification of the Multicast Deflector enables the addition of new functionalities. For instance, an interesting output from the Multicast Deflector could be the channel viewing profile of the users. This profile could lead to an extra optimization of core network usage, by allowing the Multicast Deflector not to subscribe the multicast groups associated to the channels that are never viewed by its local Receivers. Another function could be to adapt the IMGP module in order to check for Receiver credentials in an Authentication, Authorization and Accounting (AAA) server, also enabling usage accounting. A third possibility could be to support pre-paid user access by adapting the IMGP module, in order to interact with a credit control server.

The authentication model of the presented framework resides on the introduction of the Multicast Deflector element between the usual subscriber-channel provider models. With this, a reduction of group size is achieved, because these groups are limited to the Receivers in the same last mile network area, and improves the global scalability of the framework. The acceptation of a Receiver credentials can even be limited to one Multicast Deflector element, i.e., to the Receivers network area, and by this, preventing other Receivers in other network areas from using that credentials. The reuse of the same Receiver credentials in the same last mile network area can easily be restrained by tracking Receiver IP addresses.

V. CONCLUSIONS

The solution proposed solves the problem of efficiently using the bandwidth available at the last mile network and the secure user access to individual video broadcast channels.

The Multicast Deflector arises as the key element of this solution. It enables core optimization through the use of IP multicast group transmission, and enables the last mile network optimization by retransmitting only the user requested video channels. The user access control to individual video channels is achieved by using the IMGP and the KDP protocol; using them, it is even possible to suspend access to a certain video channel, for a certain period of time, to one or more authenticated users.

The modular architecture of Multicast Deflector enables the addition of other functionalities in the future, such as the support of AAA servers, pre-paid video channels access, and user activity monitoring.

REFERENCES

[1] M. Baugher, D. McGrew, M. Naslund, E. Carrara, and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)," in *RFC 3711*, 2004.

[2] T. Hardjono and B. Weis, "The Multicast Group Security Architecture," in *RFC 3740*, 2004.

[3] J. Arkko, E. Carrara, F. Lindholm, M. Naslund, and K. Norrman, "MIKEY: Multimedia Internet KEYing," in *RFC 3830*, 2004.

[4] M. Baugher, R. Canetti, L. Dondeti, and F. Lindholm, "MSEC Group Key Management Architecture," in *Internet Draft, draft-ietf-msec-gkmarch-08.txt*, 2004.

[5] S. Setia, S. Koussih, S. Jajodia, and E. Harder, "Kronos: A Scalable Group Re-Keying Approach for Secure Multicast," presented at 2000 IEEE Symposium on Security and Privacy (S&P 2000), Berkeley, CA, 2000.

[6] L. Mingyan, R. Poovendran, and C. Berenstein, "Design of Secure Multicast Key Management Schemes With Communication Budget Constraint," in *Communications Letters, IEEE*, vol. VI, 2002, pp. 108-110.

[7] J. Huang and S. Mishra, "Mykil: A Highly Scalable and Efficient Key Distribution Protocol for Large Group Multicast," presented at IEEE 2003 Global Communications Conference (GLOBECOM 2003), San Francisco, CA, 2003.

[8] S. Mittra, "Iolus: a framework for scalable secure multicast," presented at ACM SIGCOMM '97, Cannes, France, 1997.

[9] H. Chu, L. Qiao, K. Nahrstedt, and H. Wang, "A Secure Multicast Protocol with Copyright Protection," *ACM Computer Communication Review Journal (ACM CCR)*, vol. XXXII, pp. 42-60, 2002.