European Telecommunications Standards Institute
TC Satellite Earth Stations & Systems (TC-SES)
BSM Working Group; Meeting #13
ETSI; Sophia Antipolis; France
24-26 February 2003

Source:	H. Cruickshank, S. Iyengar, M Howarth, Z. Sun University of Surrey, F. Zeppenfeldt ESA (ESTEC) and G. Kenny Logica UK, ESA project: "IP Based Security Systems Testbed for Satellite Broadcast and Multicast Networks". Email: h.cruickshank@eim.surrey.ac.uk, Tel: +44 (0)1483 68 6007, Fax: +44 (0)1483 68 6011
Date:	7th February 2003
Title:	Secure IP multicast over satellites.
Document for:	
Agenda item:	

1. Abstract

Satellites are also ideally suited for delivery of multicast applications. However secure multicast over satellites is a challenging problem. One important step toward the correct solution for end-to-end security is the integration of security architectures between satellites and IP terrestrial networks.

This paper presents a secure group management and key distribution architecture based on the current activity in the IETF on IPSEC and securing group communications. The paper presents an overview of IPSEC and its modes of operation, then it shows the Logical Key Hierarchy as a group key distribution system, which makes full use of the satellite broadcast capabilities. A secure management system is also presented. Finally a brief scenario of a subscription based satellite broadcasting is illustrated using these secure multicast techniques.

2. Introduction

Demand continues to grow for broadband networks capable of supporting applications such as multimedia and information distribution, and one important component of a communications architecture that can support these services is multicast. However, terrestrial IP multicast has only slowly been deployed, due to the complexities of wide scale networks that include large numbers of multicast-enabled routers. This situation is expected to continue at least for the foreseeable future, restricting accessibility to multicast content for most potential European users. In contrast, a satellite service could simplify multicast deployment and operations/maintenance, since a single satellite hop (using only a small number of multicast enabled routers) would provide uniform delivery across the whole footprint of the EC [1] [2].

Satellite revenues in the next few years are increasingly likely to come from the delivery of these IP-based applications and services, either to complement terrestrial broadband services, or to offer added-value services in some niche markets. The challenge for the next generation of satellite systems is therefore to define a common basis for efficient integration of satellites in IP-centric telecommunication networks. Satellite access, rather than long-distance transport, is seen as a particularly convenient element of the overall telecommunication infrastructure, since it provides ubiquitous broadband access to anyone deploying a satellite terminal, both for single residential users and SOHO / corporate networks.

Satellites are also ideally suited for delivery of multicast applications, including multimedia content distribution. The next generation of satellites will extend the coverage offered by satellite services, while increasingly utilizing standardized components to reduce terminal cost. GEO satellite systems are particularly well-suited to multicast, since a single transmission is able to be received by all terminals within a wide coverage area. If these systems can be optimised to simultaneously support multicast download of bulk content and streaming of real-time multicast content, they will provide a flexible and economic IP multicast delivery platform.

In particular, there is a large interest in the study of GEO (Geostationary Earth Orbit) system solutions, which aim at providing a functionally transparent integration of satellites in Internet networks. Satellite systems based on a variety of different technologies are currently being defined and developed, and are tailored to support specific multimedia services and user requirements. Current satellite platforms are often proprietary and rely on DVB (MPEG/DVB), ATM (Asynchronous Transfer Mode), or ATM-like technologies. They cover a more or less wide range of service provisioning over transparent or regenerative satellites and support multimedia in variable proportion, whether broadcast or multicast point-to-multi-point or multi-point-to-point, or bi-directional traffic (symmetrical or asymmetrical), or point-to-point.

As such the challenge for the next generation of satellite access systems is to define a common basis for efficient integration of satellites in IP-centric telecommunication networks. One important area of integration is the communication security system. Security can be provided at various such satellite

link level (such as DVB-S conditional access and DVB-RCS MPEG-TS security); or security can be provided at the network level such as IPSEC. Both link and network level security have their strong and weak points, and can work together to provide a stronger security system.

The document focuses in network level security and providing group security using IPSEC and the related group management architecture in the IETF group called MSEC (Multicast Security). This work is a contribution from an ESA (ESTEC) project with Logica (UK) and University of Surrey (UK). Section 3 provides an overview of IPSEC features; section 4 introduces the challenges in securing group communications. Section 5 proposes and architecture for key distribution to a multicast group and section 6 presents a secure group management framework, together with its interaction with IPSEC. Section 7 focuses on the relevance of IPSEC and group security to satellite networks and gives an example of managing satellite pay TV using such security techniques. Finally, section 8 presents a brief conclusion of this work.

3. Overview of IPSEC

The security architecture of the Internet Protocol known as IP security (IPSEC) is the most advanced effort in the standardization of Internet security. The IPSEC protocol suite is used to provide inter-operable cryptographically-based security services (i.e. confidentiality, authentication, integrity, and non-repudiation) at the IP layer [3]. It consists of an authentication protocol: Authentication Header (AH) [4], a confidentiality protocol: Encapsulated Security Payload (ESP) [5] and it also includes an Internet Security Association Establishment and Key Management Protocol (ISAKMP) [6]. These security protocols are designed for both IP version 4 (IPv4) and IP version 6 (IPv6) environments.

As shown in Figure 1, the IP Authentication Header (AH) provides connectionless integrity and data origin authentication for IP datagrams. It can also provide protection against replays. The authentication header may be used, alone or in combination, with the ESP. AH authenticates slightly more information in the IP datagram than does the ESP authentication (the IP datagram header is not included in the computation of the cryptographic integrity checksum of ESP). The authentication header protocol has two modes: transport or tunnel.

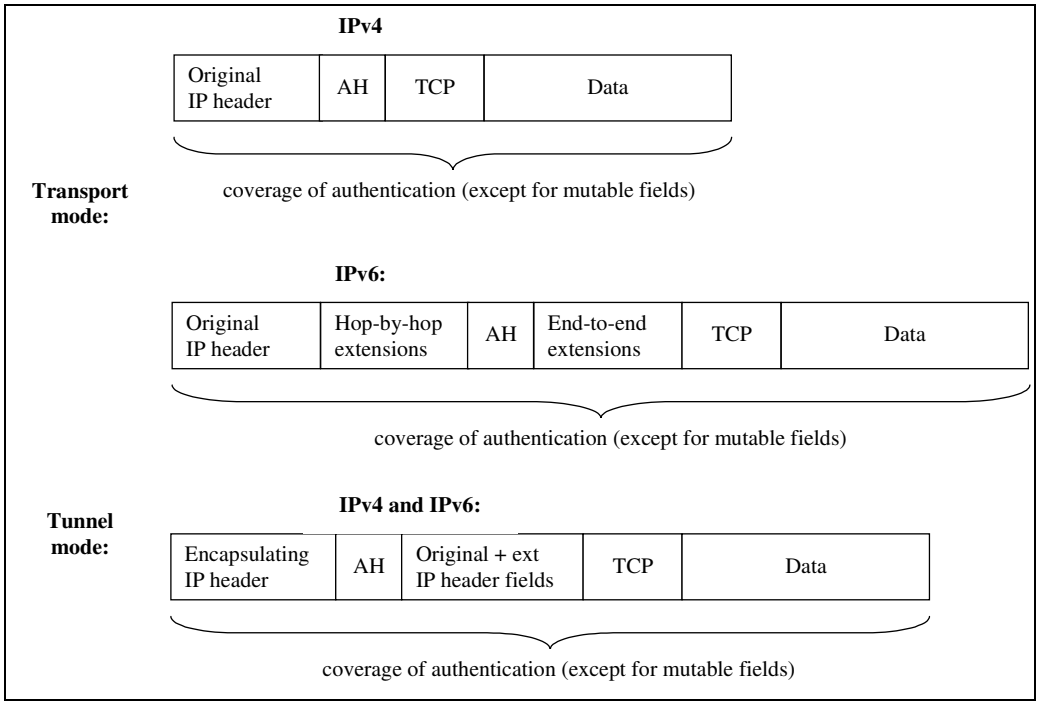

Figure 1 Authentication Header (AH) in transport and tunnel modes

Transport mode is used only in host-to-host authentication, while tunnel mode can be used between two hosts, either host-to-gateway or gateway-to-gateway. The tunnel allows the host to delegate the security service to the gateway. This is especially interesting for companies with two private distant

networks connected through the public Internet. In this mode, the IP header of the host/gateway responsible for computing/checking the AH is added while the old IP header is kept in the new IP datagram and moved after the AH. The AH does not protect mutable fields of IP datagrams (e.g., record route, timestamp, loose source routing and strict source routing options). These fields are specifically excluded from the authentication computation in order to prevent from the occurrence of authentication errors.

The Encapsulating Security Payload (ESP) protocol, shown in Figure 2, provides a mix of security services: data confidentiality, data origin authentication, connectionless integrity and anti-replay. As in the case of AH, the ESP uses a set of fields to identify the service being provided. Some of the fields are included in the ESP Header and others in the Trailer. The set of services depends on the options selected during security association establishment. ESP may be used alone or in combination with AH. It is designed to work in transport mode or in tunnel mode.

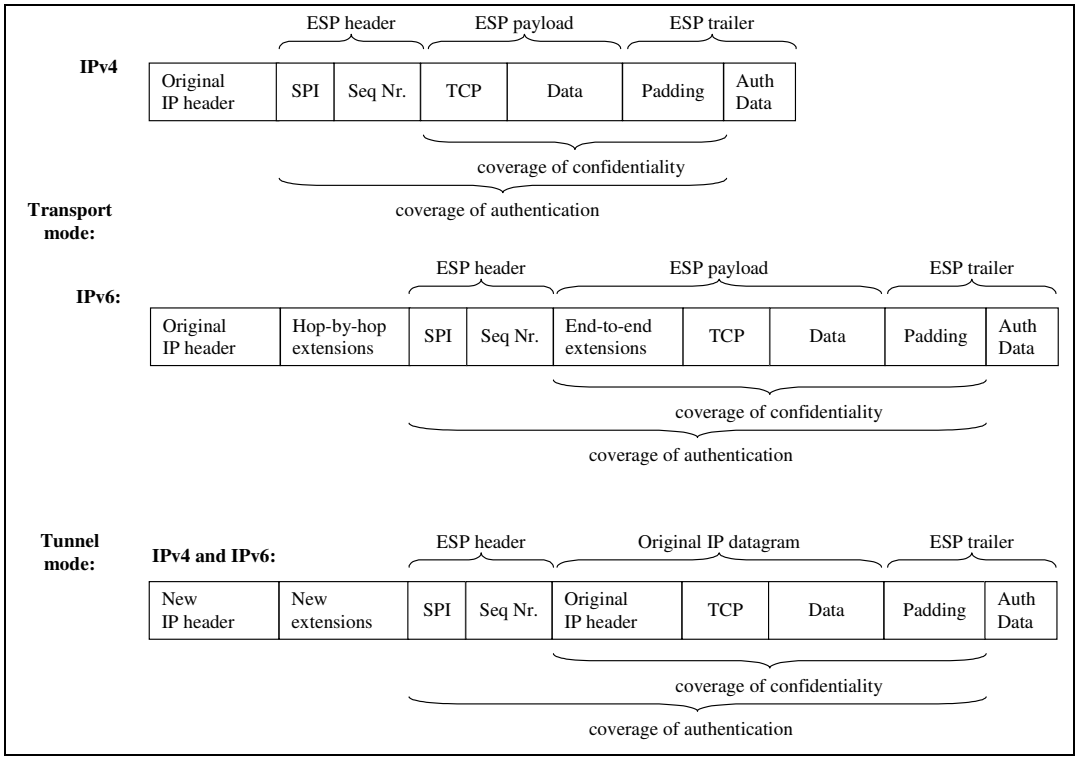

Figure 2 Encapsulated Security Payload (ESP) in transport and tunnel modes

The Security Parameter Index (SPI) field identifies the security association for this datagram (unique value for a given IP destination). SPI and destination uniquely identifies a security association. Finally, the sequence number is an optional field, and is included only if the anti-replay service is selected.

IPSEC with its various modes can be used to secure IP multicast communication in some limited scenarios. Section 4 and 5 provide a solution for secure group communications and the interaction with IPSEC.

4. Multicast security

The process of securing and performing key management for unicast connections is well understood [6], [7], [8], but multicast security is more complex. In principle, a multicast connection can be regarded as a set of unicast connections, but this approach does not scale well for large groups, especially at the scales expected in satellite systems. Protocols that manage the process of distributing keys in a multicast environment are under development [2], [11].

The principal actors in multicast key management are the group controller (GC) and group members (GMs). The former is responsible for creating and distributing keys and rekeying (to maintain security)

as appropriate; the group members are entities with access to the group keys. The GC need not be co-located with the multicast data source. Each group member has an initial one-to-one secure association with the group controller (using techniques such as Diffie-Hellman to create a shared secret known only to the two parties; or a pre-shared secret; or secret exchange using a public key system [18]). These secure associations are then used to create and share information about a group secure association between the group controller and all group members. The ultimate aim of the group secure association is to ensure that a single key, usually called the group traffic encryption key (GTEK), is known to the GC and all GMs, and to no entity outside the group: this key can then be used to encrypt the data multicast within the group.

The multicast group may need to be rekeyed for any of a number of reasons:

(1) The group key is usually updated regularly (typically every few seconds or minutes) to reduce the probability of successful cryptanalysis of the encrypted traffic.

(2) The group key may also need to be changed on demand if it is determined that the key has been compromised.

(3) Rekeying may be required when a new member joins the multicast group. This ensures that the member cannot decrypt encoded traffic sent prior to their joining (backward secrecy).

(4) Rekeying may be required when an existing member departs from the multicast group. This ensures that the member cannot decrypt encoded traffic sent after they leave (forward secrecy).

For large multicast groups that have frequent membership changes the cost of rekeying can be significant, since satellite resources are expensive. Scalable rekeying is therefore an important problem that needs to be considered in order to support secure communications for large dynamic groups. We now consider rekey techniques for each of the four functions listed above.

Several techniques exist for rekeying (1) and (3) above: two options are for the new group key to be encrypted with either (a) the old group key, or (b) a separate "control" key negotiated during session establishment. For (2) and (4) above a different rekeying approach is required since the old key is known by at least one user who is no longer to be a recipient of the multicast transmission. We now consider options for this rekeying.

A number of multicast key management approaches have been developed with the objective of improving the scalability of group secure associations, by ensuring that parameters grow more slowly than the group size, *N*. Parameters considered include group controller encryption effort, memory requirements, network traffic, and group members' decryption effort and memory requirements. Key management techniques include a flat system, Iolus [12], the logical key hierarchy, LKH [13], [14], and Kronos [19].

In the simplest system, a flat system, the GC shares a unique key with each individual group member. The GTEK can then be sent to the members by encrypting it *N* times with each of the *N* unique keys. Thus both the GC key encryption load and the rekey traffic increase linearly with *N*.

In Iolus [12], a multicast group is partitioned into several sub-groups. The group controller manages a tree of group sub-controllers, each of which manages a subset of the group membership. The advantage of this mechanism is that the rekey effort is shared between the sub-controllers, but the drawbacks of this approach are the large number of sub-controllers required in large groups, the need to trust the sub-controllers, and the delay caused by the need to rekey traffic as it passes through each sub-controller.

Logical Key Hierarchy (LKH), described in more detail below, uses a set of keys arranged in a tree structure to reduce the cost of rekeying. For a tree of outdegree k and depth d, the number of rekeys transmitted on a member compromise is reduced from $N = k^d$ (for a flat system) to $k \log_k N - 1$. The system is also robust against collusion, in that no set of users together can read any message unless one of them could have read it individually. Improvements to LKH for the specific case of binary trees (k=2) have also been proposed in one-way function trees [15] [16], and by [17]: both these approaches reduce the number of rekeys required in the event of compromise of a user from $2\log_2 N - 1$ to $\log_2 N$.

Kronos is a further approach to reducing rekey traffic in large dynamic multicast groups. This approach recognizes that if two users depart and cause two rekey events to occur, some of the keys that change will be common to the two rekey events; rekey traffic can therefore be saved by bundling changes together and rekeying, perhaps every few seconds.

5. Logical Key Hierarchy (LKH)

Logical Key Hierarchy (LKH) is a mechanism for security key management within a group of entities, providing the ability to initialise the group with a common key and then to rekey the group as required [RFC2627]. It is thus of particular application in secure multicast communications.

LKH requires two types of entity as shown in Figure 3: a group controller (GC) and one or more group members (GMs). The former is responsible for creating and distributing keys and rekeying (to maintain security) as appropriate; the group members are entities with access to the group keys. To support LKH, the GC communicates with each GM, but GMs do not need to communicate with each other. In a secure multicast communication it is not necessary for the GC to be co-located with any multicast data source.

LKH provides the following advantages:

- Scalable: the resources required to manage keys within a group grow more slowly than the group size, *N*; these resources include network transmission requirements, GC storage, GM storage, GC encryption effort, and GM decryption effort;
- Collusion-proof: no set of entities together can obtain any key unless one or more of them could have obtained it individually.

The benefit of LKH is particularly apparent when a group needs to be rekeyed.

LKH does not specify mechanisms for transmitting keys between a group controller and group members: this is the function of a group key management protocol such as GSAKMP [11]. LKH is independent of any particular encryption or decryption algorithms.

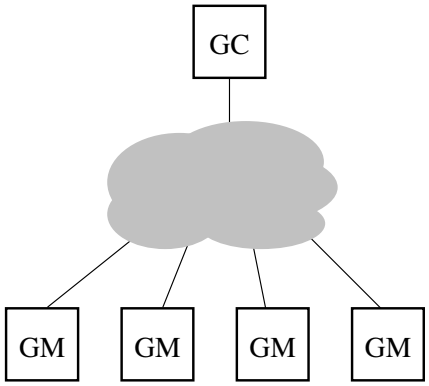

Figure 3: LKH entity logical structure

5.1. Overview of Logical Key Hierarchy

Each GM is assumed to have an initial pairwise secure association with the GC, that is to say the GM and the GC share a secret key known only to the two entities. The mechanism by which this secure association occurs is outside the scope of LKH, but typically it could involve using a technique such as Diffie-Hellman to create a shared secret known only to the two parties; or a pre-shared secret; or a secret exchange using a public key system.

We introduce LKH by considering a simple flat key management system that can be used to share a single key, 'A', so that it is known to the GC and all GMs but to no other entities. The flat key management system consists of *N* pairwise keys each shared between the group controller and one of the *N* group members (Figure 4). Each of these pairwise secure associations is represented by a circle and the group key is represented by the box labelled 'A'. If the group key is changed the new group key has to be encrypted with each user's unique pairwise key and then unicast to that user; each of these encrypted keys is represented by one of the lines drawn in Figure 4. Thus for *N* users a total of *N* encrypted keys are generated and transmitted across a network.

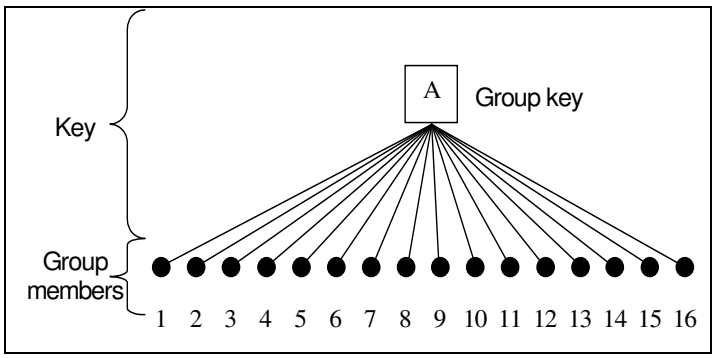

Figure 4: Key hierarchy: *N* pairwise keys

We contrast this with LKH, where a tree of keys is used to share a single key 'O' so that it is known to the GC and all GMs but to no other entities. In Figure 5 the keys are labelled A through O, the circles again represent the pairwise keys, and the lines each represent encrypted keys sent across the network, as we shall now see. Suppose now that User 11 needs to be deleted from the multicast group. Then all of the keys held by User 11 (keys F, K, N, O) must be changed and distributed to the users who need them, without permitting User 11 to obtain them or anyone else who is not entitled to them. To do this, we must replace the keys held by User 11, proceeding from the bottom up.

The server chooses a new key for the lowest node (not the leaf, for which a unicast secure association exists between the GC and the GM), and then transmits it encrypted with the appropriate daughter keys. Thus for this example, the first key replaced is Key F, and this new key will be sent encrypted with User 12's unique pairwise key. The second key replaced is Key K, which is sent encrypted with the newly replaced Key F (for User 12) and also sent encrypted with key E (for Users 9 and 10). Key N is then sent encrypted in the newly replaced Key K (for Users 9, 10, and 12) and also encrypted in key L (shared by Users 13 through 16). Finally, Key O is replaced, and this new key is sent encrypted in the newly replaced Key N (for Users 9, 10, and 12 through 16) and also separately is encrypted in key M (shared by Users 1 to 8). Since we are proceeding from the bottom up, each of the replacement keys will have been replaced before it is used to encrypt another key.

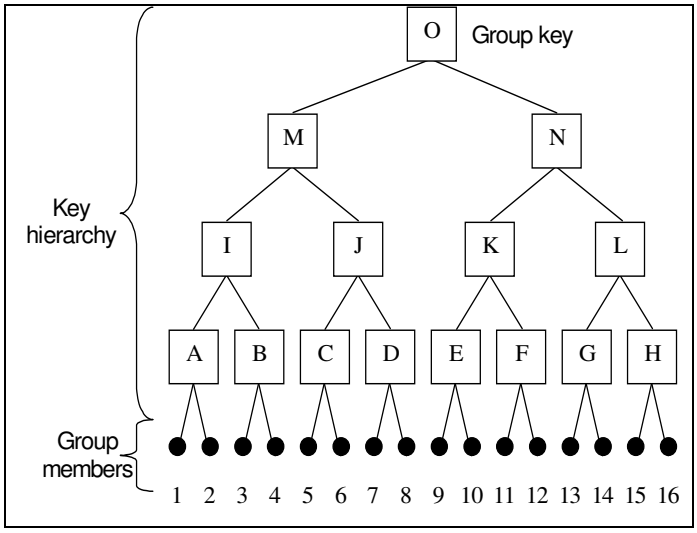

Figure 5: Logical Key Hierarchy

The seven keys sent represent a significant saving on the 16 keys that would need to be transmitted using the flat key system of Figure 4. We briefly write these keys as $\{F\}_{12}$ $\{K\}_E$ $\{K\}_F$ $\{N\}_K$ $\{N\}_L$ $\{O\}_M$ $\{O\}_N$. In general, the number of transmissions required is the sum of the degrees of the replaced nodes. In a *k*-ary tree of depth *d*, this is a total of $kd - 1 = k \log_k N - 1$ transmissions.

The group traffic encrypting key (GTEK), used to encrypt data traffic, may, depending on the group security policy, either be key O (Figure 5), or it may be separately encrypted using key O and transmitted to all group members.

5.2. Extension of LKH to multiple rekeying and user joins

LKH can be extended to allow for simultaneous rekeying following the departure of one or more GMs, and the arrival of one or more GMs, with a further saving in key encryption effort and key transmission. Figure 6 illustrates the case where GMs 11 and 15 are departing. To rekey this scenario requires 10 keys: $\{F\}_{12}$ $\{H\}_{16}$ $\{K\}_E$ $\{K\}_F$ $\{L\}_G$ $\{L\}_H$ $\{N\}_K$ $\{N\}_L$ $\{O\}_M$ $\{O\}_N$.

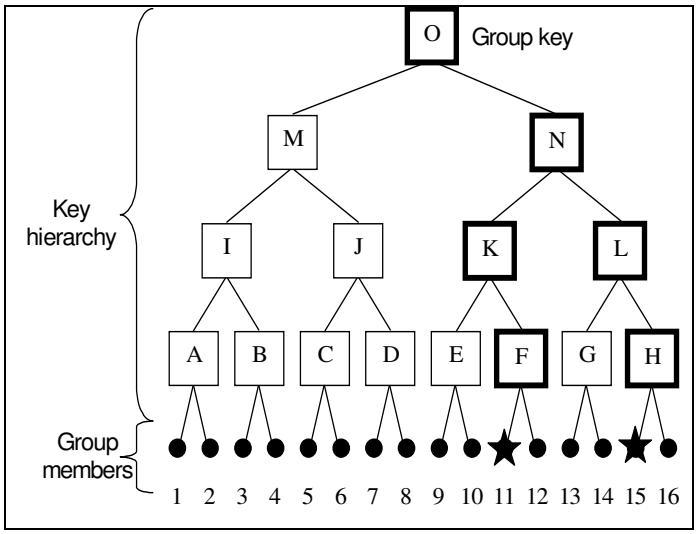

Figure 6: LKH simultaneous rekeying: two departures

For a joining GM there are two options: either the group key O has to be changed so that the new member can not retrospectively decrypt traffic that was transmitted prior to its joining, or the group key is not changed. The former is known as perfect backwards secrecy (PBS). Consider as an example when GM 1 joins (assuming all other GMs 2 through 16 are already joined): if the group key is to be rekeyed (to ensure PBS), then all keys on the path from the GM to the group key have to be changed, and 8 keys have to be transmitted: $\{A\}_1$ $\{A\}_2$ $\{I\}_A$ $\{I\}_B$ $\{M\}_I$ $\{M\}_J$ $\{O\}_M$ $\{O\}_N$. On the other hand, if the group key is not rekeyed then the joining GM simply needs to be given the set of keys from the leaf back to the tree root: these are keys: $\{A\}_1$ $\{I\}_A$ $\{M\}_I$ $\{O\}_M$.

6. Group Secure Association Key Management Protocol (GSAKMP)

The Group Secure Association Key Management Protocol (GSAKMP)[11] is a general protocol for creating and managing cryptographic groups on a network. A cryptographic group is a logical association of users or hosts that support a common security policy using shared cryptographic keying material. The GSAKMP-Light (GL) profile of GSAKMP is a three messages version of GSAKMP that does not require an underlying secure unicast security association. GL achieves this simplification by assuming that the group creation process includes the transmission to prospective members of an acceptable security suite for group establishment. While GSAKMP provides mechanisms for cryptographic group creation, other protocols may be used in conjunction with GSAKMP to allow various applications to create groups according to their application-specific requirements.

For example, in a small-scale video conference the organizer might use a session invitation protocol like Session Initiation Protocol (SIP) to transmit information about the time of the conference, the address of the session, and the formats to be used. For a large-scale video transmission, the organizer might use a multicast announcement protocol like Session Announcement Protocol (SAP). In both of these cases, non-sensitive information about the security mechanisms to be used in the cryptographic group establishment could easily be transmitted to the prospective group members. This allows 2 messages to be removed from the group establishment portion of GSAKMP, producing the GSAKMP-Light (GL) profile. GSAKMP-Light provides a profile for the group establishment case where group members have been previously notified of a set of security mechanisms during the group announcement or invitation.

GSAKMP Light provides mechanisms to perform the following key management tasks:

- Disseminate group policy;
- Distribute group keys;
- Rekey the group (e.g. if a member is compromised).

To facilitate the transmission of security mechanism settings during session invitation or announcement, this document also describes a useful default set of security algorithms and configurations called Security Suite 1. Future security suites can be defined as needed. Full specification of this suite allows an entire set of algorithms and settings to be described to prospective group members in a concise manner. Future security suites can be defined as needed.GL uses the policy definition stated in Internet Draft [11] as the policy input to the enforcement process.

The GL profile uses the LKH rekey protocol as defined in [13]. Further rekey protocols will be added in the future.

Here is the definition of some important the entities and terms used in GSAKMP documentations:

1. Group Controller: The Group Controller (GC) is a group member with authority to perform any critical protocol actions including:

- Creating and distributing keys
- Maintain the Rekey infrastructure

- Building and maintaining the Rekey arrays

2. Group Member: A group member (GM) is any entity with access to the group keys. Regardless of how a member becomes a part of the group or how the group is structured, GMs will perform the following actions:

- Validate the authorizations for security relevant actions;
- Accept group keys from the GC;
- Request group keys from the GC;
- Maintain local Certificate Revocation Lists (CRLs);
- Enforce the cooperative group policies as stated in the group policy token;
- Perform peer review of key management actions; and
- Manage their local key.

3. Group Secure Association (GSA): A cryptographic group is a logical association of users or hosts that share cryptographic key(s). This group may be established to support associations between applications or communication protocols.

4. Group Policy: The group policy completely describes the protection mechanisms and security relevant behaviours of the group. This policy must be commonly understood and enforced by the group for coherent secure operations.

5. Group Traffic Encryption Key (GTEK): The key or keys created for encrypting the group data.

6.1. GSAKMP Sequence of events

The sequence of events for GSAKMP-Light (GL) is straightforward. The sequence is:

- Security suite definition is transmitted outside the protocol.
- Group establishment phase which consists of
  - Light Request to Join (L-RTJ)
  - Light Key Download (L-KD)
  - Light Acknowledgement (L-ACK)
- Group SA up and running
- Group management

The announcement will contain at a minimum the security suite for GL. GL is designed to support architectures with a capability to announce or otherwise prepublish the group establishment parameters. These group establishment parameters must contain enough information for a group member to configure GL. Once a member has joined the group the GSA is considered up.

Group management messages are multicast in nature and include rekey, policy changes, and group deletion.

The life cycle of a GSAKMP group secure association can be divided into three phases, and these are now briefly discussed. The discussion is illustrated with the message flows shown in Figure 7; the left side of the diagram represents the actions of the GC, and the right side of the diagram represents the actions of the GMs.

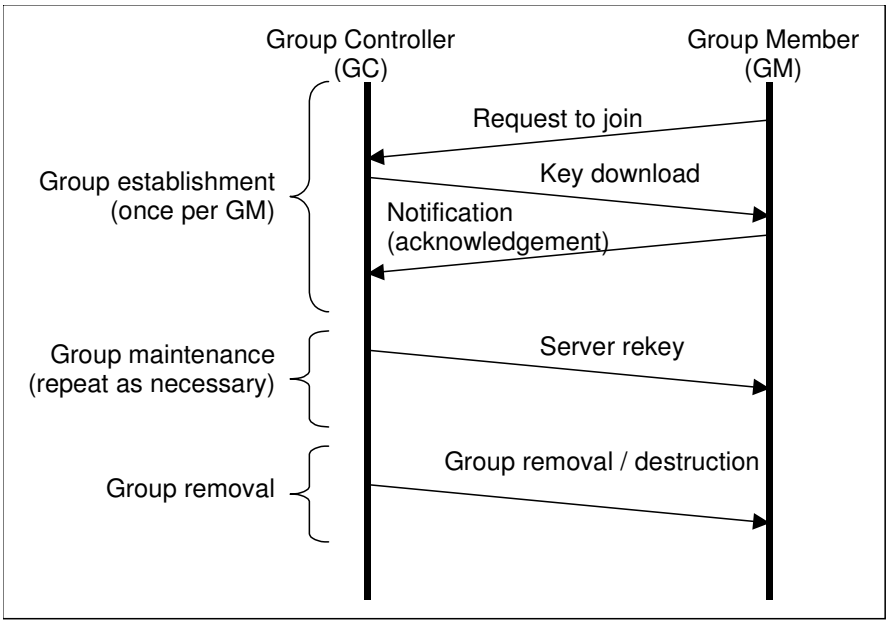

Figure 7: GSAKMP Light message exchange

6.2. Interactions with IPSEC

GSAKMP and LKH can work with IPSEC by passing the key and the associated SPI that are received in the GSAKMP client (on the end system) to the IPSEC engine. IPSEC can be used to secure multicast traffic as long as the SPI is uniquely identified [20]. This is not a problem if there is a single group controller (that issues keys and its related SPIs), which is envisaged to be the case for satellite oriented secure group communications. However having multiple key controllers (e.g. different security domains) will be a potential problem for the future.

Another potential problem area relates to anti-replay protection in case of multiple senders in one multicast group. This is a generic problem to all secure group communications and not specific to IPSEC. Normally sequence numbers are used to protect against replay attacks, where an attacker might store and replay some old messages to the group. In the case of multiple senders, the sequence numbers of messages must be co-ordinated between various senders. This is a difficult problem and an open research issue. However in IPSEC, this problem can be avoided (not sloved) in two ways:

1. Stop the anti-replay mechanism (and checking the IPSEC message sequence number) in the multicast receivers. This of course weakens the security system.

2. The use of separate sequence number sets per each sender and this implies using a separate keys and SPIs per sender. The receivers will keep track of sequence numbers for each sender.

The second solution can satisfy the requirements for many multicast applications, with the disadvantage of increased complexity in receivers for having multiple keys for the same group and keeping sets of sequence numbers. This can be a serious problem if the number of senders is very large. Restriction on the number of sender can be clearly defined in group policy that group controller (GSSAKMP) can disseminate to the group members. Of course the LKH distribution tree can be the same for that group and used to distribute all senders keys. As such, there will be multiple keys at the root of the LKH tree (see key “O” in figure 5).

7. Secure group communications relevance to satellites

Multicast represent an important class of satellite applications, where security and group key management is very important issues to be solved. A secure multicast overview and techniques has been presented in section 4, 5 and 6, namely the LKH architecture for key distribution and GSAKMP for key management application, where group controller is in charge of admitting and removing multicast group members and distributing security keys using LKH.

This security management system is a standalone system that can be interfaced to a security system (that provides data privacy and integrity) at the network level (for example IPSEC) or application level. This system can be used for secure satellite broadcasting services (securing MPEG streams in TV programmes and Internet multimedia streams). However easy interworking with IPSEC makes the satellite security system more integrated with terrestrial Internet, where IPSEC is becoming widely available in routers and end systems such as Linux and Windows based workstation.

One envisaged scenario is a subscription based satellite broadcasting service can offer TV programmes such as movie and sports services as two separate secure multicast groups. All customers that subscribe to the movie service will be managed by the GSAKMP group controller and the movie decryption key will be distributed using LKH tree architecture. When a subscriber leaves this services a new encryption/decryption key will be distributed through the same satellite multicast channel (for the movie service) to all remaining member except the leaving subscriber. The necessary LKH keys can distributed in a single multicast message as described in section 5. So the overheads of rekeying are low.

Using IPSEC or other IETF conformant system allows such movie and sports TV service to be widely accessible through the Internet or any medium such as future 3G systems that support IP multicast, as well as the satellite network. Furthermore by converting to IP based systems, future revenue streams such as broadband Internet access, Interactive TV and email can be run alongside the existing offerings over the satellite, giving subscribers a tightly coupled “one stop shop” for all their multimedia needs. Standardising on IP allows a simplification of the transmission and receiving equipment producing potential cost savings and also allows new ideas to be brought to market quickly and inexpensively.

Finally secure distributions (such as TV multicasting) using LKH and GSAKMP can avoid the current fraud problems related to DVB smart cards, which has an inherent weakness because the smart cards play an active role in decrypting the DVB broadcasts. The role of smart cards should be confined to subscriber authentication and storing the lowest level keys in LKH system as shown in Figure 5 (individual subscriber keys with group controller). In this way, breaking a smart card keys will only affect that subscriber and not the whole broadcasting service. The compromised subscribers can be easily removed from the broadcast (multicast) service by a single LKH rekeying message by the group controller.

Currently, Logica (UK) and University of Surrey, with the sponsorship of ESA (ESTEC), are working together to build a demo for secure multicast over satellites. The LKH and GSAKMP software is currently being developed with a special attention paid to compatibility issues with the original GSAKMP system that was proposed by SPARTA in [11]. Our team hope to push GSAKMP-Light into standard level at the MSEC group in the IETF by providing a second independent implementation of this system.

It is planned to submit to the BSM group an updated version of this work and detailed conclusions and recommendations after the end of the project (April 2003).

8. Conclusion

This paper presents a security solution for secure group management over satellites. This solution is based on IPSEC and related IETF activity in the MSEC working group on topics such as IPSEC, LKH

key distribution architecture and GSAKMP-Light key management system. Also the paper presents some research open issues regarding IPSEC and multicast security interaction.

Finally the paper shows a simple example of satellite TV broadcasting on subscription basis, using LKH and GSAKMP techniques. This solution can avoid the current fraud problems related to DVB smart cards, which is an inherent weakness because the smart cards play an active role in decrypting the DVB broadcasts.

This work is in progress within an ESA project between Logica (UK) and University of Surrey (UK) to provide a solution for secure IP multicast over satellites, with aim to build a demo to illustrate the secure multicast concept. It is planned to submit to the BSM group an updated version of this work and firmer conclusions after the end of the project (April 2003)

9. Reference:

[1] M. Annoni, etal (TILAB); H. Cruickshank, M. Howarth and Z. Sun (CCSR, University of Surrey, UK). "Interworking between Multi-Layer IPSEC and secure multicast services over GEO satellites". COST-272 meeting, paper number TD-02-016-P. June 2002

[2] Z. Sun, H. Cruickshank, S. Iyengar and M. Howarth, IP Multicast over Satellites – Technology Challenges, Proceedings of the 20th AIAA International Communication Satellite Systems Conference and Exhibit, Montreal, Canada, 12-15 May 2002

[3] S. Kent and R. Atkinson, "Security Architecture for the Internet Protocol", IETF RFC2401, Nov 1998.

[4] S. Kent and R. Atkinson, "IP Authentication Header", IETF RFC2402, Nov.1998.

[5] S. Kent and R. Atkinson, "IP Encapsulating Security Payload (ESP)", IETF RFC2406, Nov.1998.

[6] D. Maughan et al., "Internet Security Association and Key Management Protocol (ISAKMP)", IETF RFC2408, Nov. 1998

[7] D. Harkins and D. Carrel, "The Internet Key Exchange," IETF RFC2409, Nov. 1998.

[8] H. Orman, "The OAKLEY key determination protocol," IETF RFC2412, Nov. 1998.

[9] J. Arrko et al., "MIKEY: Multimedia Internet KEYing,", IETF Internet Draft, work-in-progress, draft-ietf-msec-mikey-05.txt, Oct 29 2002, expires Apr 2003.

[10] M. Baugher et al., "The group domain of interpretation," IETF Internet Draft, work-in-progress, draft-ietf-msec-gdoi-06.txt, Oct 2002, expires April 2003.

[11] H. Harney, A. Schuett and A. Colegrove, "GSAKMP Light," IETF Internet Draft, work-in-progress, draft-ietf-msec-gsakmp-light-sec-01.txt, Jul 2002, expires Dec 2002.

[12] S. Mittra, "Iolus: a framework for scalable secure multicasting," *Proc. SIGCOMM* 1997, pp.277-288.

[13] D. Wallner, E. Harder and R. Agee, "Key management for multicast: issues and architectures," IETF RFC2627, June 1999.

[14] C.K. Wong, M. Gouda and S.S. Lam, "Secure group communications using key graphs," *IEEE/ACM Trans. Networking*, Vol. 8 No. 1, 2000, pp.16-30.

[15] D. Balenson et al., "Key management for large dynamic groups: one-way function trees and amortized initialization", IETF Draft, work-in-progress, draft-balenson-groupkeymgmt-oft-00.txt, Feb 1999.

[16] M.J. Moyer et al., "A survey of security issues in multicast communications," *IEEE Network*, Nov 1999, pp.12-23.

[17] R. Canetti et al., “Multicast security: a taxonomy and some efficient constructions,” *Proc. IEEE INFOCOM* 1999, pp. 708-716.

[18] B. Schneier, “Applied cryptography,” John Wiley & Sons, 1996.

[19] S. Setia et al., “Kronos: a scalable group re-keying approach for secure multicast,” *Proc. IEEE Symp. on Research in Security and Privacy 2000*, pp.215-228.

[20] M. Baugher etal., “IP Multicast issues with IPsec” draft-ietf-msec-ipsec-multicast-issues-01.txt. December 2002.