Key Management Schemes for Stateless Receivers Based on Time Varying Heterogeneous Logical Key Hierarchy

Miodrag J. Mihaljević

Mathematical Institute, Serbian Academy of Sciences and Arts
Kneza Mihaila 35, 11001 Belgrade, Serbia and Montenegro
Email: miodragm@turing.mi.sanu.ac.yu

Abstract. This paper proposes a family of key management schemes for broadcast encryption based on a novel underlying structure - Time Varying Heterogeneous Logical Key Hierarchy (TVH-LKH). Note that the main characteristics of the previously reported key management schemes include the following: employment of a static underlying structure for key management, and addressing the subset covering problem over the entire underlying structure. Oppositely, the main underlying ideas for developing of the novel key management schemes based on TVH-LKH include the following: (i) employment of a reconfigurable underlying structure; and (ii) employment of a divide-and-conquer approach related to the underlying structure and an appropriate communications-storage-processing trade-off (for example, a small increase of the communication overload and large reduction of the storage and processing overload) for addressing the subset covering problem and optimization of the overloads. The design is based on a set of "static" keys at a receiver (stateless receiver) which are used in all possible reconfiguration of the underlying structure for key management, and accordingly, in a general case, a key plays different roles depending on the employed underlying structure. A particular family of the components for developing TVH-LKH, is also proposed and discussed. The proposed technique is compared with the recently reported schemes, and the advantages of the novel one are pointed out.

Keywords: broadcast encryption, stateless receivers, key management, time varying schemes, heterogeneous structures, reconfigurability, tree graphs.

1 Introduction

Broadcasting encryption (BE) schemes define methods for encrypting content so that only privileged users are able to recover the content from the broadcast. Later on, this flagship BE application has been extended to another one - media content protection (see [17] or [12], for example). This application has the same one-way nature as an encrypted broadcast: A recorder makes an encrypted recording and, a player needs to play it back. This situation usually does not allow opportunity for the player and recorder to communicate. Accordingly, in this paper we are dealing with the stateless receivers - the devices in which the operations must be accomplished based only on the current transmission and its initial configuration because these receivers do not have a possibility to update their state from session to session.

When cryptography is used for securing communications, a session-encrypting key (SEK) is used to encrypt the data. Since the data are distributed to multiple receivers, in order to reduce the amount of encryption at the sender node and to minimize the required bandwidth, every intended receiver as well as the sender should share an identical SEK. In order to ensure that only the valid members of the group have access to the communications, SEK needs to be changed whenever the lifetime of it expires, or there is a change in membership of the group, or one or more members are compromised. SEK needs to be updated under membership change for the following reasons: (i) when a new member joins, to ensure that the new member has no access to the past communication of the group, and (ii) when a member departs or is deleted, to ensure that the departed or deleted member does not have access to future communications

Ensuring that only the valid members of the selected group have SEK at any given time instance is the key management problem in BE. On the other hand, for the SEK updating, a system needs another set of keys called the key-encrypting keys (KEKs) that can be used to encrypt and transmit the updated SEK to the valid members of the group. Hence, the key management problem reduces to the problem of distributing the KEKs to the members such that at any given time instant all the valid members can be securely reached and updated with the new SEK.

A number of sophisticated methods for BE key management have been reported in the literature employing the following approach: Provide in advance the receivers with a collection of the keys (KEKs) in such a manner that the communication overload is reduced.

The first breakthrough in BE key management is reported in [8] where the schemes in which each receiver has a fixed set of reusable keys were proposed. However, the complexity of these schemes was strongly dependent on the size of the adversarial coalition.

Later on, a number of different schemes as well as the system approaches, have been reported and analyzed - see [16], [20]-[21], [3], [1], [9], [17], [18], [19], [2] and [4], for example, and recently, certain results have been reported in [11], [13], [6], [5], [14] and [15], as well.

According to [11], the most interesting variant of BE deals with stateless receivers and has the following requirements:

- Each user is initially given a collection of symmetric encryption keys.
- The keys can be used to access any number of broadcasts.
- The keys can be used to define any subset of users as privileged.
- The keys are not affected by the user's "viewing history".
- The keys do not change when other users join or leave the system.
- Consecutive broadcasts can address unrelated privileged subsets.

- Each privileged user can decrypt the broadcast by himself.
- Even a coalition of all non-privileged users cannot decrypt the broadcast.

This paper addresses the problem of developing improved BE key management schemes assuming the above given requirements.

Contributions of the paper.
This paper proposes a family of key management schemes for broadcast encryption based on the Time Varying Heterogeneous Logical Key Hierarchy (TVH-LKH).

Note that the main characteristics of the previously reported key management schemes include the following ones: (i) employment of a static underlying structure for key management; (ii) addressing the subset covering problem considering the underlying structure as a whole.
Oppositely, the main underlying ideas for developing of the improved key management schemes based on TVH-LKH include the following:

– employment of a time varying (reconfigurable) heterogeneous underlying structure;
– employment of a divide-and-conquer approach related to the underlying structure and an appropriate communications-storage-processing trade-off (for example: a small increase of the communication overload and large reduction of the storage and processing overload) for addressing the subset covering problem and optimization of the overloads.

Note that the proposed design is based on a set of "static" keys at a receiver (stateless receivers) which are used for all possible reconfiguration of the underlying structure for key management. So, in a general case, a key plays different roles depending on the employed underlying structure.

A family of the components called sectioned heterogeneous LKH (SH-LKH) and its special form consisting of the sectioned key trees (SKTs) are considered for developing the reconfigurable logical key hierarchy, and TVH-LKH with two particular family members called SKT-A and SKT-B is discussed.

The approach employed for design of SH-LKH family could be formulated as follows: Before dealing with the set covering issues, perform an appropriate preprocessing over the underlying LKH in order to specify a more suitable underlying structure for the set covering.

The main underlying ideas for developing a novel family of key management schemes are based on employment of appropriate clustering of the keys and users, and employment of the heterogeneous time varying and cluster oriented local key management. Accordingly, the design rationale for the novel family includes the following: (i) specification of the appropriate partitions/sections over the employed LKH; (ii) performing key management on the section-by-section basis; (iii) in a general case, employment different key management schemes in different sections or in different time instants; (iv) in certain cases, employment of modified local (section related) key management schemes which employ a relaxed specification of the privileged set.

Let N be the number of receivers and R the number of revocations. Assuming that the parameters of a particular TVH-LKH scheme with SKT-A and SKT-B are H_{0A}, H_{0B}, H_{1B}, R_{0A}, R_{0B} and R_{1B}, such that $1 \le H_{0A} < log_2 N$, $2 \le H_{0B} + H_{1B} < log_2 N$, $1 \le R_{0A} \le R$, and $1 \le R_{1B} \le R_{0B} \le R$, its main characteristics are as follows. Dimension of the storage@receiver overload: $O(max\{ \ ((H_{0A})^{1.5} - H_{0A} + log_2 N), \ ((H_{0B})^{1.5} + (H_{1B})^{1.5} - H_{0B} - H_{1B} + log_2 N)\}$. Dimension of the communications overload: $O(min\{ \ (R + R_{0A}((log_2 N) - H_{0A}) - R_{0A} log_2 R_{0A}), \ (R + R_{0B} + R_{1B}((log_2 N) - H_{1B} - H_{0B}) - R_{1B} log_2 R_{1B})\}$. Maximum dimension of the processing@receiver overload: $O(max\{H_{0A}, max\{H_{0B}, H_{1B}\}\})$.

An illustrative comparison of the main characteristics of the proposed key management and the recently reported ones is given in Table 1, assuming a huge group with a heavy dynamics in order to demonstrate advantages of the proposal even in the considered scenario. Intentionally, the comparison is related to the most powerful recently reported schemes based on the binary tree approach to demonstrate advantages of the considered particular TVH-LKH which is also based on the binary tree approach.

Table 1. Illustrative numerical comparison of the main characteristics of the proposed TVH-LKH key management schemes and the Complete Sub-Tree (CST) [17], Subset Difference (SD) [17] and Layered Subset Difference (LSD) [11], assuming $N = 2^{27}$ receivers and $R = 2^{15}$ revocations, and that the parameters of the considered TVH-LKH technique are $H_{0A} = 10$, $H_{0B} = 7$, $H_{1B} = 7$, $R_{0A} = 2^{14}$, $R_{0B} = 2^{14}$ and $R_{1B} = 2^{11}$.

technique	storage@receiver	processing@receiver	communication
CST [17]	~ 27	~ 5	$\sim 12 \cdot 2^{15}$
SD [17]	~ 729	~ 27	$\sim 2^{15}$
basic LSD [11]	~ 140	~ 27	$\sim 2^{15}$
proposed TVH-LKH	~ 49	~ 7	$\sim 1.5 \cdot 2^{15}$

Table 1 illustrates how combining of the heterogeneous schemes in a time-varying manner appear as a powerful approach for developing improved key management schemes which yield a possibility for appropriate trade-offs between the main overloads of the system.

Organization of the paper.
Section 2 yields the underlying ideas for developing of the improved key management schemes, and a general framework for key management based on the

reconfigurable logical key hierarchy (TVH-LKH). Key management based on reconfigurable logical key hierarchy which employes a collection of the sectioned key trees is considered in Section 3 including a comparison of a particular TVH-LKH based technique and recently reported schemes targeting the same key management scenario. Finally, some concluding discussions are given in Section 4, and two proposition proofs are accommodated in Appendices A-B.

2 Underlying Ideas and General Framework for a Novel Design

This section points out the underlying ideas for the improved key management schemes proposed in this paper, and a general framework for development of these schemes.
Note that the *general static key management paradigm* is based on the following:
(a) BE center specify a set of all keys it will use, and assigns its subset to each receiver in such a manner that based on the keys stored at the receivers, BE center can split the set of all receivers into two arbitrary (usually) non overlapping parts.
(b) BE center adopts a method for covering an arbitrary subset of the receivers taking into account the keys assigned to the receivers.
(c)The established system is used for the session key distribution.

Unfortunately, in a general case, the above item (b) is a variation of the Set Cover problem (see [10] for example): It is known that no approximation algorithm exists for the Set Cover with a worst-case approximation ratio better than $ln(N)$ [7] (assuming that N is the number of receivers).

In order to deal with the covering problem in an efficient way and employing much smaller required set of keys and the reduced processing at a receiver in comparison with the reported schemes, this section proposes a novel approach based on the reconfigurable key management. The following main three issues are addressed: (i) underlying ideas for proposing reconfigurable logical key hierarchy; (ii) general framework for the reconfigurable key management; and (iii) a discussion on selection of the main components for the proposed framework.

2.1 Underlying Ideas for the Key Management Schemes Based on Reconfigurable Logical Key Hierarchy

Recall that the main characteristics of the reported key management schemes include the following:
- employment of a static underlying structure for the key management;
- addressing the subset covering problem considering the underlying structure as a whole.
Oppositely, the main underlying ideas for developing the improved TVH-LKH based key management schemes include the following:

– employment of a reconfigurable underlying structure;

– employment of a divide-and-conquer approach related to the underlying structure and an appropriate communications-storage-processing trade-off (for example, a small increase of the communication overload and large reduction of the storage and processing needed by a receiver) for addressing the subset covering problem and optimization of the system overloads.

Note that the design is based on a set of "static" keys at a receiver which are used for all possible reconfiguration of the underlying structure for key management. So, in a general case, a particular key plays different roles depending on the employed underlying structure.

Recently, very efficient key management schemes Complete SubTree (CST) and Subset Difference (SD) have been proposed in [17] and Layered Subset Difference (LSD) has been reported in [11]. These schemes have been developed by focusing on obtaining a solution for the underlying set covering problem using the tree based paradigm. The approach proposed in this paper, beside employment of the reconfigurability concept, is also different in comparison with the previously reported ones in a way which could be formulated as follows: Before dealing with the set covering issues, perform an appropriate preprocessing over the underlying LKH in order to specify a more suitable underlying structure for the set covering. The employed preprocessing could also be considered as a particular divide-and-conquer method for key management.

The main underlying ideas for developing a novel family of the key management schemes include the following ones.

– employment of time varying logical key hierarchy;
– specification of a set of different and appropriate partitions/sections of the logical key hierarchy (in a particular case based on appropriate clustering of the keys and users);
– performing key management on the section-by-section basis (heterogeneous cluster oriented local key management);
– in a general case, employment different key management schemes in different sections or the time instances;
– optionally, in certain cases, employment of modified local (section related) key management schemes which provide a relaxed specification of the privileged set.

The opportunity for employment of different key management schemes in different sections or the time instances opens a door for desired optimization of the key management overload characteristics. For example, recall that CST re-keying requires significantly smaller storage@receiver overload at the expense of increased communications overload in comparison with LSD based re-keying. Accordingly, employing the CST based technique in one subset of the tree sections and LSD based one in another subset, for example, yields an opportunity for obtaining the desired overall characteristics. Also note the following two characteristics of SD and LSD schemes: (i) communications overload is linear with R; (ii) storage@receiver overload is polynomial with $logN$. These characteristics open a door for the trade-off based on divide-and-conquer approach. Additionally, note that, for example, a relaxed version of SD or LSD, which does not perform the strict revocations but the relaxed ones in a manner similar to that reported in [1], could be employed as the appropriate one in certain cases.

Also note that, although the key management at the center's side is time varying and based on the section-by-section processing, this has no impact at the receivers side, and after all, a receiver should employ, in an appropriate manner, just one of its KEKs to recover the new SEK.

2.2 General Framework for the Key Management Based on Reconfigurable Logical Key Hierarchy

The Center's Framework

Pre-Processing.
Establishing the reconfigurable logical key hierarchy based key management requires the following main actions at the center side.

- Specification of a collection of the underlying structures to be used for the covering of privileged (non-revoked) receivers.
- Assigning a set of keys to each of the receivers in such a manner that the key management can be performed employing any of the underlying structures from the collection.

Processing.
For delivering a new SEK the center performs the following:

- According to the given list of revocations, the center select an appropriate underlying structure from the collection for key management.
- The center jointly broadcast encrypted forms of the new SEK obtained by employing different KEKs and information of the KEKs employed, as well as the mode of their use, determined by currently selected underlying structure.

The Receiver's Framework

The framework for the proposed TVH-LKH based key management at the receiver's side consists of the following components:

- Each receiver is provided with a set of the keys and information on modes of their use.
- If not revoked, during the key management communication, a receiver obtains the following information:
  - which of its KEKs should be employed for the new SEK recovering, and
  - in which mode the employed KEK should be used (depending on the currently employed underlying structure from the predefined set),

  and accordingly it is able to recover the new SEK.

2.3 On the Keys Employment and Selection of the Underlying Structures

Note that the design is based on a set of "static" keys at a receiver which are used for all possible reconfiguration of the underlying structure for key management, and accordingly, in a general case, a key plays different roles depending on the employed underlying structure.

A main component of the reconfigurable key management is a collection of the underlying structures, and regarding these structures note the following.

- The underlying structures could be very different but all of them should fulfil the following condition: They should be able to work with the same single set of keys (KEKs) assuming that a key can be employed in different modes.
- A large number of the reconfigurable schemes can be designed in an ad-hock manner. Selection of the underlying structures included in the collection depends on the functional requirements of the key management. An optimized design should particularly take into account the space and time distribution of the revocations.

Accordingly, for given number of keys at a receiver, the reconfigurable logical key hierarchy (TVH-LKH) based key management yields an opportunity for minimizing the communications overload or the processing@receiver overload. On the other hand, note that TVH-LKH based schemes do not require additional storage@receiver overload in comparison with corresponding static LKH schemes which can be employed for the same revocation scenario.

3 A Reconfigurable Key Management Based on a Collection of Sectioned Heterogeneous LKHs

3.1 General Design Issues

Recall that the first step for establishing a reconfigurable logical key hierarchy is selection of a collection of the appropriate underlying structures for key management. This section proposes a particular TVH-LKH based on a novel structure called sectioned heterogeneous LKH (SH-LKH) for developing the underlying collection for the reconfigurable key management.

SH-LKH structure is displayed in Fig. 1. The triangles play roles of certain substructures: In a particular case they are the subtrees with the root at the triangle up and the leaves at the triangle bottom. These subtrees (embedded into triangles) could be very different including the following ones, (i) binary balanced tree, (ii) a tree consisting just of the root and a number of leaves, or (iii) other suitable trees.

¿From the center point of view, the key management scheme consists, as in an usual case, of the following two main components: (i) underlying structure for the keys and receivers assigning; (ii) methods employed for distributing a session key (SEK) to the stateless receivers. After this conceptual similarity, the

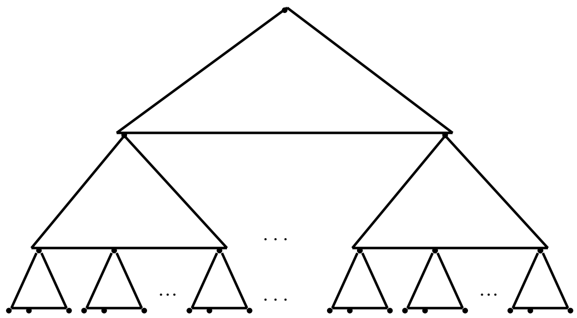

Fig. 1. A general form of the sectioned heterogeneous logical key hierarchy (SH-LKH). The triangles play roles of certain substructures, and in a particular case they are the subtrees with the root at the triangle up and the leaves at the triangle bottom.

proposed scheme differs from the reported ones as follows:
- instead of a single underlying structure the center ”possesses” a collection of different underlying structures
- each element of the collection is an SH-LKH;
- the distribution of SEK is based not on a single technique but on employment a number of different ones.

Accordingly, TVH-LKH employing SH-LKH is based on the following.

– The center selects an appropriate collection of SH-LKH to be used for key management.
– A set of keys is assigned to each of the receivers in such a manner that it can support any of SH-LKH key management schemes from the collection.
– In the case of SEK rekeying, the center broadcast SEK encrypted under different KEKs, and the related information on the employed keys and the mode of theirs use.
– At the receiver's side the processing is adjusted according to the obtained information on the employed keys.

Note that a special case of SH-LKH is the sectioned key tree (SKT) introduced in [15].

3.2 Key Management Based on Sectioned Key Trees (SKTs)

This section, following [15], yields a background for developing and analyzing a particular TVH-LKH based on a collection of the underlying structures called sectioned key trees (SKTs). Note that SKTs are just a particular family of the binary tree structures which could be employed for design of certain TVH-LKH.

Family of SKTs

An SKT is the sectioned key tree displayed in Fig. 2 and obtained by the following horizontal and vertical partitioning:
- a number of the horizontal layers is specified;
- each layer is partitioned into a number of sections and each section contains a sub-tree which root is identical to a leaf of the upper layer section.

In a special case, the following can be enforced: each of the layers has the same height, and each layer's section contains the same number of nodes. Accordingly, each section contains the same subtree.

In a general case, the tree is partitioned into L horizontal layers with the heights H_ℓ, $\ell = 0, 1, .., L-1$, respectively, assuming that $\ell = 0$ corresponds to the bottom layer and $\ell = L-1$ to the top one. Then, the top layer contains a sub-tree with $2^{H_{L-1}}$ leaves, and a layer ℓ consists of

$$\prod_{i=\ell+1}^{L-1} 2^{H_i} = 2^{\sum_{i=\ell+1}^{L-1} H_i}$$

sections, each containing a sub-tree with 2^{H_ℓ} leaves.

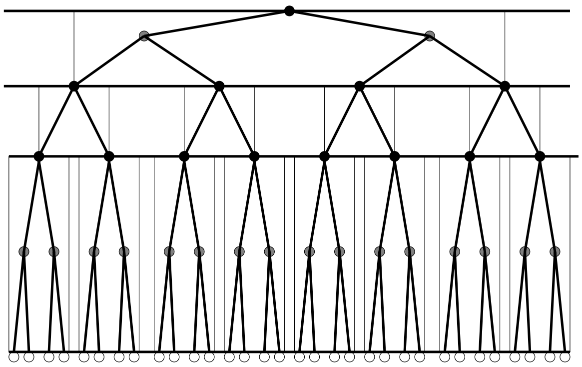

Fig. 2. An illustration of the sectioned key tree (SKT). As usually, the center is associated to the tree root, a receiver is at a leaf, and the keys are related to the tree nodes.

Accordingly, we assume the following basic scenario for the key management based on the above underlying structure: N receivers grouped into M clusters, R revocations in total, assuming R_m revocations from a cluster with index m, $m = 1, 2, ..., M$, and the parameter M is an integer such that $\sum_{m=1}^{M} R_m = R$ and N/M is an integer, $M \leq N$.

Section-by-Section Key Management

The proposed key management scheme assumes the section-by-section key management, and in a general case, it yields the opportunity for employment different local key management schemes in different sections.

Assuming SKT with L layers, and that a layer ℓ contains $M^{(\ell)}$ sections, $\ell = 0, 1, ..., L-1$, we propose the following section-by-section key management:

- layer 0 processing
  - For the subtree corresponding to section j, identify a set $\mathcal{R}_j^{(0)}$ of the leaves (receivers) which should be revoked, $j = 1, 2, ..., M^{(0)}$.
  - Perform section-by-section processing: for the revocations over the subtree in section j employ a desired key management scheme for revocation of elements in $\mathcal{R}_j^{(0)}$, $j = 1, 2, ..., M^{(0)}$.
- layer ℓ processing, $\ell = 1, 2, ..., L-1$
  - For the subtree corresponding to section j, identify a set $\mathcal{R}_j^{(\ell)}$ of the leaves which correspond to the sections in layer $\ell - 1$ affected by the revocations, and accordingly which should be revoked, $j = 1, 2, ..., M^{(\ell)}$.
  - Perform section-by-section processing: for the revocations over the subtree in section j employ a desired key management scheme for revocation of elements in $\mathcal{R}_j^{(\ell)}$, $j = 1, 2, ..., M^{(\ell)}$.

Center

At the center side, the procedure for revocation of a number of receivers consists of the following main steps:
(a) the center specifies a set of receivers which should be revoked;
(b) employing the section-by-section processing, the center decides on KEKs (nodes of the tree) which should be used for the new SEK delivery (encryption);
(c) center broadcast the following message: (i) an implicit information on the employed KEKs; and (ii) the new SEK encrypted by each of the employed KEKs.

Let $E(\cdot)$ denotes the algorithm employed for encryption of the new SEK ($newSEK$), I_m defines the information on a KEK with index m, KEK_m, employed for encryption of the new SEK, $m = 1, 2, ..., M$, where M is total number of KEKs employed for covering the desired subset of receivers, and $\mathcal{F}_{newSEK}(\cdot)$ denotes the algorithm employed for the payload encryption. Accordingly, BE center broadcast the following:

$$
\begin{aligned}
&[[I_1, I_2, ..., I_M, E_{KEK_1}(newSEK), E_{KEK_2}(newSEK), ..., \\
&\qquad E_{KEK_M}(newSEK)], \mathcal{F}_{newSEK}(Payload)] \\
= &\quad [[I_1, I_2, ..., I_M, C_1, C_2, ..., C_M], PayloadCiphertext] \ .
\end{aligned}
$$

Receivers

At a receiver side the situation is equivalent as, for example, to the one when CST, SD, or LSD based approaches are employed. A receiver should store a number of cryptographic keys, monitor the communication channel to see whether its current SEK should be exchanged, and if "yes" extract the new SEK based on certain processing employing a memorized key. Actually, a receiver can not be aware of the employed underlying structure at the center's side.

At a receiver's side the re-keying is performed as follows. Each receiver monitors the communications channel looking for the re-keying message broadcasted by the center. In this message, a non-revoked receiver will find an information on a KEK it posses which should be used for the new SEK recovering. Based on this information and the encrypted form of the new SEK, the non-revoked receiver will recover the new SEK.

Accordingly, upon receiving a broadcast message, the receiver performs the following operations:
- Finding I_m which is related to the receiver: If the receiver is revoked, no such information will be found;
- Employing I_m and the keys stored at the receiver, perform a processing in order to recover KEK_m employed for $newSEK$ encryption.
- Recovering the new SEK performing the decryption $E^{-1}_{KEK_m}(C_m)$.

Finally, after recovering the new SEK, the payload is obtained by $\mathcal{F}^{-1}_{newSEK}(PayloadCiphertext)$.

Two Particular Key Management Schemes: SKT-A and SKT-B

As the illustrative examples, this section specify two particular key management schemes called SKT-A and SKT-B where SKT stands for Sectioned Key Tree.

SKT-A. SKT-A is a particular key management scheme based on the following partitioning of the key tree and the local re-keying:
• There are two horizontal layers and height of the bottom one is equal to H_{0A}, and accordingly the upper layer has height equal to $log_2N - H_{0A}$;
• Basic LSD [11] revocation method is employed in each section of the bottom layer and CST [17] revocation method is employed in the upper layer-section.

SKT-B. SKT-B is a particular key management scheme based on the following partitioning of the key tree and the local re-keying:
• There are three horizontal layers and heights of the bottom and middle ones are equal to H_{0B} and H_{1B}, respectively; accordingly the top layer has height equal to $log_2N - H_{0B} - H_{1B}$;
• Basic LSD [11] revocation method is employed in each section of the two lower layers and CST [17] revocation method is employed in the upper layer-section.

Analysis of SKT Based Key Management Schemes

This section is focused on the following issues of the considered key management schemes: (i) communications - dimension of the messages overload to be sent for the re-keying; (ii) storage@receiver - dimension of keys which should be stored at a receiver; (iii) processing@receiver - processing overload due to the keys updating at receiver.

Main Characteristics of SKT-A. Taking into account the results reported in [17] and [11], it can be shown that SKT-A key management has the following main characteristics.

Proposition 1. SKT-A key management requires the following overload for R revocations in total which affect R_{0A} different sections, assuming R/R_{0A} revocations per section:

- dimension of the storage@receiver overload: $O((H_{0A})^{1.5} - H_{0A} + log_2 N)$;
- dimension of the communications overload: $O(R + R_{0A}((log_2 N) - H_{0A}) - R_{0A} log_2 R_{0A})$;
- dimension of the processing@receiver overload: $O(H_{0A})$.

The proposition proof is given in Appendix A.

Main Characteristics of SKT-B. Taking into account the results reported in [17] and [11], it can be shown that SKT-B key management has the following main characteristics.

Proposition 2. SKT-B key management requires the following overload for R revocations in total which affect R_{0B} and R_{1B} different sections in the lower two layers, the bottom (0-th) and the middle (1-st) ones, respectively:

- dimension of the storage@receiver overload: $O((H_{0B})^{1.5} + (H_{1B})^{1.5} - H_{0B} - H_{1B} + log_2 N)$;
- dimension of the communications overload: $O(R + R_{0B} + R_{1B}((log_2 N) - H_{1B} - H_{0B}) - R_{1B} log_2 R_{1B})$;
- dimension of the processing@receiver overload: $O(max\{H_{0B}, H_{1B}\})$.

Proposition 2 proof is given in Appendix B.

3.3 Illustrative Example of TVH-LKH Employing SKTs

As an illustration of the proposed TVH-LKH based on a collection of SKTs, we consider the following toy example:
• TVH-LKH underlying collection consists of only SKT-A and SKT-B, and there are R revocations in total.
• In SKT-A case, R revocation affect R_{0A} clusters of receivers (sections).
• In SKT-B case, R revocation affect R_{0B} sections in the bottom layer and R_{1B} sections in the middle layer.

Proposition 3. The above specified TVH-LKH key management over a group of N receivers requires the following overload for R revocations in total which affect R_{0A} or R_{0B} and R_{1B} different sections in the lower layers, of SKT-A and SKT-B, respectively:

- dimension of the storage@receiver overload: $O(max\{\ ((H_{0A})^{1.5} - H_{0A} + log_2N), ((H_{0B})^{1.5} + (H_{1B})^{1.5} - H_{0B} - H_{1B} + log_2N)\}$;
- dimension of the communications overload: $O(min\{\ (R + R_{0A}((log_2N) - H_{0A}) - R_{0A}log_2R_{0A}), (R + R_{0B} + R_{1B}((log_2N) - H_{1B} - H_{0B}) - R_{1B}log_2R_{1B})\}$;
- dimension of the processing@receiver overload: $O(H_{0A},$ or $max\{H_{0B}, H_{1B}\})$.

Proof Remarks. The proposition statement is a direct consequence of Propositions 1-2, and the selection strategy related to TVH-LKH which assumes employment of a scheme from the available collection which yields minimization of the communications overload. Particularly note that storage@receiver overload is determined by the maximum storage@receiver overload required by the schemes in the collection.

Accordingly, based on the results on CST, SD and LSD reported in [17] and [11], respectively, a comparison of these schemes and the considered TVH-LKH is summarized in Tables 2 and 3. Note that intentionally, the comparison is related to the most powerful recently reported schemes based on the binary tree approach to demonstrate advantages of considered particular TVH-LKH which is also based on the binary tree approach.

Table 2 yields a comparison of the storage and processing overloads, and Table 3 displays a comparison of the communications overloads.

Table 2. Comparison of the storage@receiver and processing@receiver overloads of the proposed TVH-LKH key management scheme and the Complete Sub-Tree (CST)[17], Subset Difference (SD) [17] and Layered Subset Difference (LSD) [11], assuming N receivers, R revocations, and the parameters of the considered TVH-LKH technique are H_{0A}, H_{0B}, H_{1B}, R_{0A}, R_{0B} and R_{1B}, such that $1 \leq H_{0A} < log_2N$, $2 \leq H_{0B} + H_{1B} < log_2N$, $1 \leq R_{0A} \leq R$, and $1 \leq R_{1B} \leq R_{0B} \leq R$.

technique	storage@receiver	processing@receiver
CST [17]	$O(log_2N)$	$O(log_2log_2N)$
SD [17]	$O((log_2N)^2)$	$O(log_2N)$
basic LSD [11]	$O((log_2N)^{1.5}$	$O(log_2N)$
proposed TVH-LKH	$O(max\{\ ((H_{0A})^{1.5} - H_{0A} + log_2N),$ $((H_{0B})^{1.5} + (H_{1B})^{1.5} - H_{0B} - H_{1B} + log_2N)\}$	$O(H_{0A})$ or $O(max\{H_{0B}, H_{1B}\})$

Table 3. Comparison of the communications overload of the proposed TVH-LKH key management scheme and the Complete Sub-Tree (CST)[17], Subset Difference (SD) [17] and Layered Subset Difference (LSD) [11], assuming N receivers, R revocations, and the parameters of the considered TVH-LKH technique are H_{0A}, H_{0B}, H_{1B}, R_{0A}, R_{0B} and R_{1B}, such that $1 \leq H_{0A} < log_2 N$, $2 \leq H_{0B} + H_{1B} < log_2 N$, $1 \leq R_{0A} \leq R$, and $1 \leq R_{1B} \leq R_{0B} \leq R$.

technique	communication overload
CST [17]	$O(Rlog_2 \frac{N}{R})$
SD [17]	$O(R)$
basic LSD [11]	$O(R)$
proposed TVH-LKH	$O(min\{\ (R + R_{0A}(log_2 N - H_{0A}) - R_{0A} log_2 R_{0A}),$ $(R + R_{0B} + R_{1B}((log_2 N) - H_{1B} - H_{0B}) - R_{1B} log_2 R_{1B})\}$

4 Discussion

A novel and flexible paradigm for developing BE key management schemes is proposed. The proposal is based on the reconfigurability concept, and it yields the improved overall characteristics in comparison with the previously reported techniques. Tables 1-3 show that combining of the heterogeneous schemes in a time-varying manner appear as a powerful approach for developing improved key management schemes which yield a possibility for desired trade-offs between the main overloads related to the key management system. The design is based on a set of "static" keys at a receiver which are used for all possible reconfiguration of the underlying structure for key management, and accordingly, in a general case, a key plays different roles depending on the employed underlying structure.

The Gain Origins. The main origin of the gain obtained by the proposed key management in comparison with the previously reported techniques is due to the employed concept of reconfigurability and a dedicated divide-and-conquer approach. Particularly, certain gain origins include the following: (i) partition of the underlying LKH structure into the sections which appears as a very powerful technique for obtaining improved characteristics; (ii) performing overall key management based on a number of local (the section oriented) key managements; in a general case these key managements can be different and time varying.

Some Further Work Directions. TVH-LKH yields a generic framework for developing efficient key management, and besides the underlying structures discussed in this paper, it is an open problem to find novel constructions and particularly ones dedicated to certain applications. Also recall that (as in other schemes) there are three main overloads related to the proposed key management: storage@receiver, processing@receiver and communications overload. Taking into account certain constraints on these parameters, the proposed schemes can be optimized following the approaches reported in [3] and [19]. For example, for given constraints on storage@receiver and processing@receiver, the schemes can be optimized regarding the communications overload, or for the given communications budget, the schemes can be optimized regarding storage@receiver and processing@receiver. On the other hand, in certain cases (where this is appropriate), further reduction of the overloads can be obtained employing a relaxed specification of the targeting receivers subset in a manner similar to that reported in [1] where certain receivers which should be revoked will not be excluded during the re-keying, assuming that the rate of this free-riders is within desired limits.

References

1. M. Abdalla, Y. Shavitt and A. Wool, "Key management for restricted multicast using broadcast encryption", *IEEE/ACM Trans. Networking*, vol. 8, pp. 443-454, Aug. 2000.
2. S. Banerjee and B. Bhattacharjee, "Scalable secure group communication over IP multicast", *IEEE Journal on Selected Areas in Communications*, vol. 20, pp. 1511-1527, Oct. 2002.
3. R. Canetti, T. Malkin and K. Nissim, "Efficient communication-storage tradeoffs for multicast encryption", EUROCRYPT'99, *Lecture Notes in Computer Science*, vol. 1592, pp. 459-474, 1999.
4. K.-C. Chan and S.-H. Gary Chan, "Distributed server networks for secure multicast", *IEEE Journal on Selected Areas in Communications*, vol. 20, pp. 1500-1510, Oct. 2002.
5. P. D'Arco and D.R. Stinson, "Fault tolerant and distributed broadcast encryption", CT-RSA 2003, *Lecture Notes in Computer Science*, vol. 2612, pp. 263-280, 2003.
6. G. Di Crescenzo and O. Kornievskaia, "Efficient re-keying protocols for multicast encryption", SCN 2002, *Lecture Notes in Computer Science*, vol. 2576, pp. 119-132, 2003.
7. U. Feige, "A threshold of $ln(n)$ for approximating set cover", *Jour. ACM*, vol. 45, pp. 634-652, July 1998.
8. A. Fiat and M. Naor, "Broadcast encryption", Advances in Cryptology - CRYPTO93, *Lecture Notes in Computer Science*, vol. 773, pp. 480491, 1994.
9. J.A. Garay, J. Staddon and A. Wool, "Long-lived broadcast encryption", CRYPTO 2000, *Lecture Notes in Computer Science*, vol. 1880, pp. 333-352, 2000.
10. M.R. Garey and D.S. Jonson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco, CA: Freeman, 1979.
11. D. Halevy and A. Shamir, "The LCD broadcast encryption scheme", CRYPTO 2002, *Lecture Notes in Computer Science*, vol. 2442, pp. 47-60, 2002.
12. J. Lotspiech, S. Nusser and F. Prestoni, "Broadcast encryption's bright future", *IEEE Computer*, (7 pages) August 2002.
13. J.H. Ki, H.J. Kim, D.H. Lee and C.S. Park, "Efficient multicast key management for stateless receivers", ICISC 2002, *Lecture Notes in Computer Science*, vol. 2587, pp. 497-509, 2003.
14. N. Matsuzaki, T. Nakano and T. Matsumoto, "A flexible tree-based key management framework", *IEICE Trans. Fundamentals*, vol. E86-A, pp. 129-135, 2003.

15. M.J. Mihaljević, "Broadcast encryption schemes based on the sectioned key tree", ICICS2003, *Lecture Notes in Computer Science*, vol. 2836, Oct. 2003.
16. S. Mittra, "Iolus: A framework for scalable secure multicasting", Proc. ACM SIGGCOM97, pp. 277-288, Sept. 1997.
17. D. Naor, M. Naor and J. Lotspiech, "Revocation and tracing schemes for stateless receivers", CRYPTO 2001, *Lecture Notes in Computer Science*, vol. 2139, pp. 41-62, 2001.
18. R. Poovendran and J. S. Baras, "An information theoretic approach for design and analysis of rooted-tree-based multicast key management schemes", *IEEE Trans. Inform. Theory*, vol. 47, pp. 2824-2834, Nov. 2001.
19. R. Poovendran and C. Bernstein, "Design of secure multicast key management schemes with communication budget constraint", *IEEE Communications Letters*, vol. 6, pp. 108-110, March 2002.
20. D. Wallner, E. Harder and R. Agee, "Key management for multicast: Issues and architectures", *RFC 2627*, http://www.ietf.org/rfc/rfc2627.txt
21. C.K. Wong, M. Gouda, and S.S. Lam, "Secure group communications using key graphs", *IEEE/ACM Trans. Networking*, vol. 8, pp. 16-31, Feb. 2000.

Appendix A: Sketch of Proposition 1 Proof

Recall that in SKT-A scheme there are $2^{log_2 N - H_{0A}}$ sections in the lower layer, and each of them is controlled via the basic LSD technique [11]; the upper layer consists of only one section where CST technique [17] is employed.

Note that the re-keying of a receiver is performed via the lower layer section or the upper layer one. Accordingly, a receiver should store the keys related to LSD and CST based re-keying. A section oriented basic LSD technique requires $(H_{0A})^{1.5}$ keys, and the upper section oriented CST requires $log_2 N - H_{0A}$ keys. So, dimension of storage@receiver overload is $O((H_{0A})^{1.5} - H_{0A} + log_2 N)$.

Regarding the processing@receiver overload note the following. A new SEK could be delivered to the receiver employing the LSD or CST related keys. If a LSD related key is employed, the new SEK recovering at the receiver requires the processing overload proportional to H_{0A}. If a CST related key is employed, the new SEK recovering requires processing@receiver overload proportional to $log_2 log_2 2^{log_2 N - H_{0A}} = log_2(log_2 N - H_{0A})$. So the maximum processing@receiver overload is: $O(max\{H_{0A}, log_2(log_2 N - H_{0A})\}) = O(H_{0A})$.

Finally, regarding the communications overload, suppose that there are r_m revocations in the mth section, $m = 1, 2, ..., 2^{log_2 N - H_{0A}}$, noting that $\sum_{m=1}^{2^{log_2 N - H_{0A}}} r_m = R$, and $\sum_{m=1}^{2^{log_2 N - H_{0A}}} (1 - \delta_{0,r_m}) = R_{0A}$, where $\delta_{a,b}$ is a function which takes value 1 if $a = b$, and 0 otherwise. LSD based revocation within a section m requires communication overload of dimension $O(r_m)$, assuming $r_m > 0$. So, revocation of all R receivers require a communications overload of dimension $O(R)$. Also, R_{0A} revocations should be performed over the upper section employing CST, which requires additional communication overload of dimension $O(R_{0A} log_2(2^{log_2 N - H_{0A}}) - R_{0A} log_2 R_{0A})$. Accordingly, dimension of the communications overload is given by $O(R + R_{0A}((log_2N) - H_{0A}) - R_{0A}log_2R_{0A})$.

Appendix B: Sketch of Proposition 2 Proof

Recall that in SKT-B scheme there are $2^{log_2N-H_{0B}}$ sections in the lower layer, and $2^{log_2N-H_{0B}-H_{1B}}$ in the middle layer: each of them is controlled via the basic LSD technique [11]; the upper layer consists of only one section where CST technique [17] is employed.

Note that the re-keying of a receiver is performed via a section within one of the tree layers, i.e., via a lower layer section or via a middle layer section or via the upper layer one. Accordingly, a receiver should store the keys related to LSD rekeying within the lower or middle layer, and CST related ones for the upper layer. Recall that the lower layer section oriented basic LSD technique requires $(H_{0B})^{1.5}$ keys, the middle layer section oriented basic LSD technique requires $(H_{1B})^{1.5}$ keys, and the upper section oriented CST requires $log_2N - H_{0B} - H_{1B}$ keys. So, dimension of storage@receiver overload is $O((H_{0B})^{1.5} + (H_{1B})^{1.5} - H_{0B} - H_{1B} + log_2N)$.

Regarding the processing@receiver overload note the following. A new SEK could be delivered to the receiver employing the LSD or CST related keys. If a LSD related key is employed, the new SEK recovering at the receiver requires the processing overload proportional to H_{0B} or H_{1B} depending weather a key from the lower or middle layer is employed. If a CST related key is employed, the new SEK recovering requires processing@receiver overload proportional to $log_2log_22^{log_2N-H_{0B}-H_{1B}} = log_2(log_2N - H_{0B} - H_{1B})$. So the maximum processing@receiver overload is: $O(max\{H_{0B}, H_{1B}, log_2(log_2N - H_{0B} - H_{1B})\}) = O(max\{H_{0B}, H_{1B}\})$.

Finally, regarding the communications overload, suppose that there are r_m revocations in the mth section, $m = 1, 2, ..., 2^{log_2N-H_{0B}}$, noting that $\sum_{m=1}^{2^{log_2N-H_{0B}}} r_m = R$, and $\sum_{m=1}^{2^{log_2N-H_{0B}}} (1 - \delta_{0,r_m}) = R_{0B}$, where $\delta_{a,b}$ is a function which takes value 1 if $a = b$, and 0 otherwise. LSD based revocation within a section m requires communication overload of dimension $O(r_m)$, assuming $r_m > 0$. So, revocation of all R receivers require a communications overload of dimension $O(R)$. Also, R_{0B} revocations of the sections from the lower layer should be performed within the middle layer employing middle sections oriented basic LSD approach. Employing an equivalent consideration to the above one related to the lower layer, we obtain that revocation of all R_{0B} sections in the middle layer require a communications overload of dimension $O(R_{0B})$. Additionally, R_{1B} revocations should be performed over the upper section employing CST, which requires additional communication overload of dimension $O(R_{1B}log_2(2^{log_2N-H_{0B}-H_{1B}}) - R_{1B}log_2R_{1B})$. Accordingly, dimension of the communications overload is given by $O(R + R_{0B} + R_{1B}((log_2N) - H_{1B} - H_{0B}) - R_{1B}log_2R_{1B})$.