A Revocation Scheme with Minimal Storage at Receivers

Tomoyuki Asano[⋆]

Sony Corporation
6-7-35 Kitashinagawa, Shinagawa-ku, Tokyo 141-0001, Japan
tomo@arch.sony.co.jp

Abstract. A revocation or a broadcast encryption technology allows a sender to transmit information securely over a broadcast channel to a select group of receivers excluding some revoked receivers. In this paper we propose two efficient revocation methods which are suitable for stateless receivers. The proposed methods use an a-ary key tree structure and require at most $r\left(\frac{\log(N/r)}{\log a}+1\right)$ ciphertexts broadcast. Our Method 1 requires only one key to be stored and $O\left(\frac{2^a \log^5 N}{\log a}\right)$ computational overhead at a receiver, whereas Method 2 requires $\frac{\log N}{\log a}$ keys and $O\left(2^a\right)$ computational overhead, where N and r respectively denote the total number of receivers and the number of revoked receivers. Our methods are very efficient with respect to the number of keys each receiver stores, especially Method 1 minimizes it.

1 Introduction

Recent advances in technology give us a lot of ways to distribute digital data without loss of quality. We can easily use, modify and exchange many kinds of digital data such as digital pictures or music. However, those advances have caused serious challenges related to *copyright protection* or *digital rights management* issues. Though copyright-protected data (e. g. music, movies or TV programs) should be treated under conditions to which its copyright holder agrees, various kinds of such content can be recorded, copied or exchanged in an illegal manner. One of the technologies that are being used to protect such data is called *revocation scheme* or *broadcast encryption scheme.* This technology allows a sender to transmit information securely over a broadcast channel to a select group of receivers. The sender may exclude some receivers (called *revoked receivers*) and enable only legitimate receivers to obtain the transmitted information.

Revocation schemes are used in many real world applications. For example, in a pay-TV system, users can watch TV programs if they subscribe to the service and pay the fee. If some users do not pay for the programs, they might be excluded, so that they will not be able to watch the program the following

[⋆] Most of this work was done while the author was visiting Stanford University.

month even if they own an appropriate receiver. Other examples are CPPM and CPRM [8] which are systems protecting copyrighted content stored on prerecorded or recordable media from unauthorized copying. In these systems only compliant receivers (i. e. players or recorders) that are manufactured under a certain license contract can retrieve secret information (called *session key*) from a medium using their receiver keys. The session key is required for decryption or encryption of the content file stored on the medium. If it is found that there is a receiver which does not obey the license contract this receiver will be revoked, and as a result it will no longer be able to retrieve the session keys distributed after the revocation.

For general receivers (e. g. consumer electronics devices), the easiest way to store secret information such as receiver keys is storing it as part of the initial configuration at manufacturing time. Giving a mechanism to receivers for changing keys they store increases the production cost and might also weaken their security. Therefore it is preferable in most cases to assume that receivers can not change their keys. Such receivers are called *stateless receivers.* As described in [17], typical examples of stateless receivers are off-line devices, such as CD or DVD players. In this paper we propose two efficient revocation methods that are suitable for stateless receivers.

The organization of this paper is as follows. We introduce related work and contributions of this paper in the rest of this section. Section 2 describes two revocation methods proposed in this paper. We discuss the security of our methods in section 3, some techniques and the properties of those methods in section 4. We present a modification of CPPM and CPRM in Section 5. Our results in this paper are summarized in section 6.

1.1 Related Work

As described in the previous section, a revocation scheme or broadcast encryption scheme allows a sender to transmit information securely over a broadcast channel to a select group of receivers. Let N and r be the total number of receivers in the system and the number of revoked receivers, respectively. A naive method to implement this scheme is as follows. Assume each receiver owns a unique key. A sender broadcasts secret information encrypted under each of the unique keys owned by the non-revoked receivers. This method requires each receiver to store only one key, but the sender must transmit $N - r$ ciphertexts. Since a large amount of bandwidth is necessary for large N, this method is not suitable for applications where the bandwidth for such data is restricted.

There exists another naive method where the size of the broadcast message is minimized. We call it the Power Set Method. The method defines a power set of N receivers, i. e. $\{S_{b_1 b_2 \ldots b_i \ldots b_N}\}$ where $b_i \in \{0, 1\}$. Each b_i indicates whether or not a receiver i belongs to a subset $S_{b_1 b_2 \ldots b_i \ldots b_N}$. It assigns a subset key for each subset and gives the subset key to receivers which belong to the subset. To send secret information to an arbitrary group of receivers, a sender chooses a subset where $b_i = 1$ only for selected receivers i, encrypts the information with a subset key corresponding to the subset, and broadcasts the ciphertext.

This method requires the sender to broadcast only one ciphertext, while each receiver needs to store 2^{N-1} keys. Hence this method is not suitable for receivers in many applications if N is large. However, we use this technique in conjunction with a key tree structure in order to reduce the number of ciphertexts which are broadcast in our methods.

The notion of *broadcast encryption* was introduced by Berkovits [3] and Fiat et al. [10] independently. The main criteria for this technology are the number of ciphertexts (the length of the message) to be broadcast, the number of keys each receiver stores, and the computational overhead at a receiver. Berkovits constructed a broadcast encryption method using a secret sharing scheme [22]. This method requires each receiver to store one key, however the length of the broadcast message is $O(N)$. Fiat et al. proposed an r-resilient method which is resistant to a collusion of up to r revoked receivers by combining their 1-resilient methods hierarchically. This method requires message length of $O\left(r^2 \log^2 r \log N\right)$ and the storage of $O(r \log r \log N)$ at each receiver.

Wallner et al. [24] and Wong et al. [25] independently proposed efficient methods using a logical key tree structure. Their methods define a logical tree and a node key for each node of the tree. Each receiver is assigned to a leaf of the tree and given a set of node keys defined for the nodes on the path from the leaf to the root of the tree. Therefore, each receiver stores $\log N + 1$ keys, assuming that the system uses a binary tree. All of these keys except one are shared by other receivers. This method revokes one receiver at a time, and updates all keys stored by non-revoked receivers, which have also been owned by the revoked receiver. A sender needs to broadcast $2 \log N$ ciphertexts and a receiver needs to perform at most $\log N$ decryptions for this single revocation. If the system needs to revoke r receivers by repeating the single revocation, the sender has to send $2r \log N$ ciphertexts.

Since the key tree structure has good properties, modifications of the methods of [24, 25] have been proposed [5, 11, 16, 17]. Some of them reduce the messages for a single revocation to $\log N$ by combining the key tree structure with another technique. McGrew et al. [16] used a one-way function, Canetti et al. [5] used a pseudo random generator, and Kim et al. [11] used Diffie-Hellman key exchange scheme [9]. The number of keys a receiver stores remains $\log N + 1$, while their methods increase the computational overhead at a receiver, namely, each receiver needs to perform the computation of such a technique at most $\log N$ times. Similar to their original methods, they assume non-stateless receivers, i. e. receivers have a capability to change their keys.

If receivers are not stateless, they can store keys (e. g. shared keys established among the sender and a group of receivers) given at time t_1 and use them at time t_2 (where $t_1 < t_2$) to obtain the current session key. This may contribute to reduce the size of the broadcast. On the other hand, stateless receivers can store only the keys given at the initial stage such as manufacturing time. Hence every broadcast message must contain enough information to enable non-revoked receivers to obtain the current session key using their initial receiver keys.

Kumar et al. [13] proposed revocation methods using error correcting codes. In their methods only non-revoked receivers can correct the error in the broadcast message and retrieve the secret information. Their construction based on polynomials requires messages of $O\left(r \log N\right)$ broadcast and storage of $O\left(r \log N\right)$ at a receiver, and their construction based on algebraic-geometric codes requires message of $O\left(r^2\right)$ and $O\left(r \log N\right)$ storage overhead. The latter construction is interesting because the length of the message is independent of the number of total receivers.

Anzai et al. [2] and Naor et al. [18] independently proposed other methods using a secret sharing scheme. The main advantage of their methods is the size of storage at receivers. Their methods require receivers to store an element in a certain group. On the other hand, the methods require $O\left(w\right)$ messages broadcast and $O\left(w\right)$ exponentiations performed at a receiver, where w is the upper bound of the number of revoked receivers in the system which is fixed in advance. In other words, if we set the system resistant to a collusion of any number of revoked receivers then $O\left(N\right)$ messages and $O\left(N\right)$ exponentiations are required, regardless of the number of receivers actually revoked. Matsuzaki et al. [15] modified their method to reduce the computational overhead at a receiver to two modular exponentiations.

CPPM and CPRM [8] are methods for protection of copyrighted content stored on pre-recorded or recordable media (e. g. disks or semiconductor memories) that work with stateless receivers. Those methods require a prefixed number of ciphertexts being broadcast on the media and a relatively small number of keys being stored at a receiver. However, their revocation capability is restricted. We present detailed explanation of their properties as well as a modification of them in section 5.

Naor et al. [17] proposed two efficient methods suitable for stateless receivers using a binary key tree construction. The Complete Subset Method requires a sender to broadcast $r \log\left(N/r\right)$ ciphertexts and each receiver to store $\log N + 1$ keys, whereas the Subset Difference Method using a pseudo random sequence generator requires $2r - 1$ ciphertexts, $\frac{1}{2}\log^2 N + \frac{1}{2}\log N + 1$ keys and $O\left(\log N\right)$ computational overhead at a receiver.

Luby et al. [14] and Poovendran et al. [20] analyzed the criteria of broadcast encryption schemes under information theoretic concepts. Since the methods we propose in this paper are constructed upon a computational assumption, their bounds are not applicable to them.

Our methods use a key tree structure and are suitable for stateless receivers. They provide a good balance in the criteria for the revocation technology, and are more efficient with respect to the number of keys stored at each receiver compared to previously proposed methods with such a structure. Especially one of our methods requires receivers to store only one key.

Another topic related to a revocation technology is a *traitor tracing* technology introduced by Chor et al. [7]. This is used to find a receiver who contributed for production of a non-legitimate receiver device or software by giving its secret information (e. g. receiver keys). Many schemes with traitor tracing capability, such as [4, 7, 17, 18], have been proposed. We briefly explain the applicability of our methods to a traitor tracing scheme in section 4.4.

1.2 Our Results

In this paper we propose two efficient revocation methods suitable for stateless receivers. The Master Key technique due to Chick et al. [6] (a similar technique is also described in [10]) contributes to reduce the number of keys each receiver stores, and the Power Set Method used in conjunction with an a-ary logical key tree structure helps to reduce the number of ciphertexts to be broadcast, where the parameter a can be any positive integer satisfying $a > 1$. In turn, our methods require receivers to perform some computations.

The properties of our methods are shown in Table 1. For comparison, this table also contains the properties of the Complete Subset Method (CSM) and the Subset Difference Method (SDM) proposed in [17], which are considered to be most efficient among the methods proposed previously. In Method 1 we construct a revocation method which requires receivers to store only one key (a master key). Therefore, this method achieves minimal storage overhead for receivers. Method 2 is a variant of Method 1 which reduces the computational cost incurred by receivers to derive a key used for decryption of broadcast ciphertext from their master key in exchange for an increase in the number of master keys they store.

Table 1. The properties of methods in [17] and our methods

	CSM [17]	SDM [17]	Method 1	Method 2
Number of ciphertexts	$r \log (N/r)$	$2r - 1$	$r \left(\frac{\log (N/r)}{\log a} + 1 \right)$	$r \left(\frac{\log (N/r)}{\log a} + 1 \right)$
Number of keys @ receiver	$\log N$	$\frac{1}{2} \log^2 N$	1	$\frac{\log N}{\log a}$
Comp. cost for key derivation				
Pseudo-random generator	-	$O(\log N)$	-	-
Generation of primes	-	-	$O\left(\frac{2^a \log^5 N}{\log a} \right)$	-
Num. of multiplications	-	-	$\frac{\left(2^{a-1} - 1\right) \log N}{\log a}$	$2^{a-1} - 1$
Num. of modular exp.	-	-	1	1

Table 1 tells that although our methods require more computational overhead of receivers, they are more efficient than other methods with respect to the number of keys each receiver stores. Our methods are also more efficient than CSM with regard to the number of ciphertexts broadcast. Since the Master Key technique is based on the security of RSA cryptosystem [21], the size of each master key in our methods is the size of a secure RSA modulus. Note that as analyzed in [17] a receiver in each method in Table 1 needs $O(\log \log N)$ lookup operations (in order to find an appropriate ciphertext to decrypt) and a single decryption operation which are omitted from the table.

We show that our methods are secure under the assumption related to the RSA cryptosystem. Then we discuss some techniques which are used in our methods to reduce the size of the broadcast and the size of the storage at a receiver. We also provide a modification of CPPM and CPRM using the Master Key technique which reduces the size of the storage at receivers.

2 The Proposed Methods

In this section we introduce two efficient revocation methods. These methods use the Power Set Method' defined below as an elemental technique.

Power Set Method' with n elements. *Suppose there is a set of n elements i $(i = 1, \ldots, n)$. Define $2^n - 2$ subsets $S_{b_1 b_2 \ldots b_i \ldots b_n}$ where $b_i \in \{0, 1\}$, $\sum_{i=1}^{n} b_i \neq 0$ and $\sum_{i=1}^{n} b_i \neq n$ (which are elements of a power set except all b_i's are 0 and 1). Assign a subset key for each subset. Give subset keys corresponding to subsets where $b_i = 1$ to an element i. To send secret information to an arbitrary group of elements (except a group which consists of all elements), choose a subset where $b_i = 1$ only for selected elements i, encrypt the information with a subset key corresponding to the subset and send the encrypted information.*

Assume N is a power of a positive integer a. Our methods adopt a logical a-ary tree called the Hierarchical Key Tree (HKT) and the Power Set Method' with a elements for each internal node (including the Root[1]) in the HKT. Basically $2^a - 2$ subsets are defined for each internal node. A subset key is chosen for each subset and $2^{a-1} - 1$ subset keys are given to each child node of the internal node. A receiver is assigned to a leaf of the HKT. Let $path_j$ be a path from the leaf to which a receiver u_j is assigned to the Root. A receiver u_j is given master key(s) that can derive any subset key given to a node on $path_j$. Transmission of secret information including revocation of receivers is performed by broadcasting one or more ciphertexts encrypted under a subset key. This construction has a good property such that only one ciphertext needs to be sent for secure transmission to an arbitrary set of child nodes of a certain internal node in the HKT.

We have two ways of applying the Master Key technique to a revocation scheme. In Method 1 we adopt the Master Key system for the whole HKT. A receiver u_j is given a master key of $\left(2^{a-1} - 1\right) \log_a N + 1$ subset keys corresponding to the subsets to which the receiver belongs, i. e. those subset keys are given to the nodes on $path_j$. Note that those subsets contain a subset to which all of N receivers belong, and the corresponding subset key is used if no receivers are revoked. In Method 2 we apply the Master Key system to each internal node in the HKT. In this method $\log_a N$ master keys are given to a receiver u_j. Each master key can derive at most 2^{a-1} subset keys corresponding to subsets defined for a node on $path_j$, to which the receiver u_j belongs.

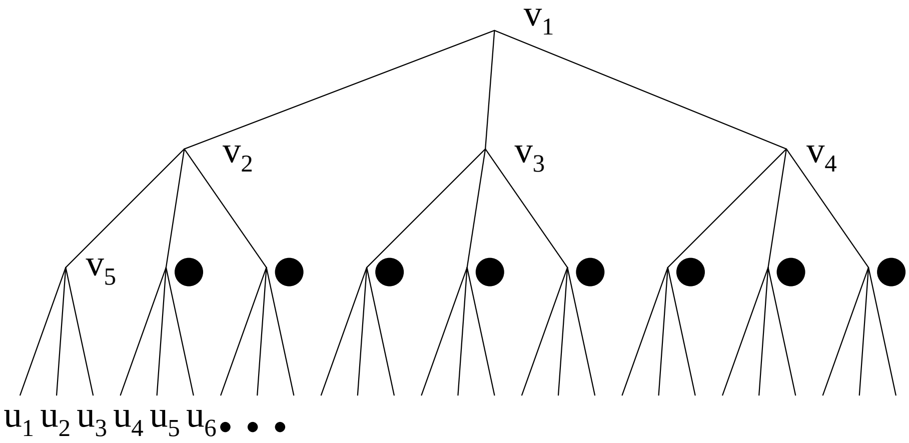
[1] For clarity, we write the root of the HKT as 'the Root'.

2.1 Method 1

Setup.

Step 1. Trusted Center (TC) which is a sender of secret information defines a rooted full a-ary HKT with N leaves. Each internal node in the HKT is named v_k $\left(k = 1, \ldots, \frac{N-1}{a-1}\right)$ where the Root is v_1 and other nodes are named with breadth first order. A receiver u_j $(j = 1, \ldots, N)$ is assigned to each leaf of the HKT. TC defines $2^a - 2$ subsets $S_{k,b_1 b_2 \ldots b_i \ldots b_a}$ where $b_i \in \{0, 1\}$, $\sum_{i=1}^{a} b_i \neq 0$ and $\sum_{i=1}^{a} b_i \neq a$ for an internal node v_k. TC also defines a subset $S_{1,11\ldots1}$ for the Root. TC selects large primes q_1 and q_2 and publishes M $(= q_1 q_2)$.

Figure 1 shows an example of the HKT for $a = 3$ and $N = 27$, and subsets which are defined for internal nodes. The way to define the subsets is common to both of Method 1 and Method 2.



internal node v_k	subsets defined for v_k
v_1	$S_{1,100}$, $S_{1,010}$, $S_{1,001}$, $S_{1,110}$, $S_{1,101}$, $S_{1,011}$, $S_{1,111}$
$v_2, \ldots, v_{\frac{N-1}{a-1}}$	$S_{k,100}$, $S_{k,010}$, $S_{k,001}$, $S_{k,110}$, $S_{k,101}$, $S_{k,011}$

Fig. 1. Subsets defined for internal nodes of the tree

Step 2. TC chooses $(2^a - 2)\frac{N-1}{a-1} + 1$ primes $p_{k,b_1 b_2 \ldots b_i \ldots b_a}$ where $k = 1, \ldots, \frac{N-1}{a-1}$, $b_i \in \{0, 1\}$, $\sum_{i=1}^{a} b_i \neq 0$ for all k and $\sum_{i=1}^{a} b_i \neq a$ for $k \neq 1$. Let B denote $b_1 b_2 \ldots b_i \ldots b_a$. Then TC assigns $p_{k,B}$ to a subset $S_{k,B}$ and publishes this assignment. Let T be a product of all primes assigned to the subsets. TC randomly chooses an element $K \in \mathbb{Z}_M^*$ and sets a subset key $SK_{k,B}$ corresponding to a subset $S_{k,B}$ as

$$SK_{k,B} = K^{T/p_{k,B}} \bmod M$$

TC (imaginarily) gives subset keys $SK_{k,B}$ with $b_i = 1$ to i^{th} child node of the internal node v_k. Therefore, $2^{a-1} - 1$ subset keys are given to each of the child nodes of an internal node. In addition, a subset key $SK_{1,11\ldots1}$ is given to each of the child nodes of the Root.

Step 3. TC gives a receiver u_j a master key MK_j of $\left(2^{a-1} - 1\right) \log_a N + 1$ subset keys that are given to the nodes on $path_j$. Let w_j be a product of all primes assigned to the subsets to which the receiver u_j belongs (i. e. the corresponding subset keys are given to the nodes on $path_j$). The master key MK_j is defined as

$$MK_j = K^{T/w_j} \bmod M$$

In the example depicted in Fig. 1, the following master key MK_1 of subset keys is given to a receiver u_1.

master key	corresponding subset keys
MK_1	$SK_{1,100}$, $SK_{1,110}$, $SK_{1,101}$, $SK_{1,111}$
	$SK_{2,100}$, $SK_{2,110}$, $SK_{2,101}$
	$SK_{5,100}$, $SK_{5,110}$, $SK_{5,101}$

Revocation. Transmission of secret information (e. g. a session key used for encryption or decryption of a content file) including revocation of some receivers is performed by broadcasting one or more ciphertexts. Each ciphertext is an encryption of the secret information under a subset key. To find subset keys to be used for this encryption, TC abandons all subset keys which are known to revoked receivers. It can be considered as removing all edges from the leaves corresponding to the revoked receivers to the Root in the HKT. This removal leaves one or more disjoint subtrees. Each subtree corresponds to a subset defined for its root node which is an internal node in the HKT, and each leaf of them is assigned to a non-revoked receiver. TC encrypts the secret information under the subset keys corresponding to those subsets, then broadcasts the ciphertexts. We examine the upper bound of the number of ciphertexts broadcast in section 4.1 and describe a technique used to encrypt the secret information in section 4.2.

Decryption. A non-revoked receiver belongs to a subset corresponding to a subtree which is left in the revocation phase, namely, the subtree contains the leaf assigned to the receiver. Note that for a non-revoked receiver u_j, there is exactly one ciphertext among the broadcast message which is an encryption under a subset key which can be derived from its master key MK_j. Naor et al. [17] introduced some techniques for listing and searching the correspondence of subsets and receivers, which can be used in conjunction with our methods.

After finding an appropriate subset, a receiver u_j computes the corresponding subset key $SK_{k,B}$ from its master key MK_j and decrypts the ciphertext using the subset key in order to retrieve the secret information. The derivation of the subset key is performed as follows.

$$MK_j^{w_j/p_{k,B}} \bmod M = \left(K^{T/w_j}\right)^{w_j/p_{k,B}} \bmod M = K^{T/p_{k,B}} \bmod M = SK_{k,B}$$

Recall that $p_{k,B} \mid w_j$ and w_j is a product of $\left(2^{a-1}-1\right)\log_a N + 1$ primes. The computational overhead is roughly $O\left(\frac{2^a \log^5 N}{\log a}\right)$ for generation of primes as analyzed in section 4.3, and $\left(2^{a-1}-1\right)\log_a N$ multiplications and one modular exponentiation.

2.2 Method 2

Setup.

Step 1. The process in Step 1 is the same as in Method 1.

Step 2. TC chooses $2^a - 1$ primes $p_{b_1 b_2 \ldots b_i \ldots b_a}$ where $b_i \in \{0,1\}$ and $\sum_{i=1}^{a} b_i \neq 0$. Let B denote $b_1 b_2 \ldots b_i \ldots b_a$. TC assigns p_B to a subset $S_{k,B}$ defined for each internal node v_k $\left(k = 1, \ldots, \frac{N-1}{a-1}\right)$ and publishes this assignment. Let T be a product of all primes p_B. Then TC independently chooses $\frac{N-1}{a-1}$ elements $K_k \in \mathbb{Z}_M^*$ and sets a subset key $SK_{k,B}$ corresponding to a subset $S_{k,B}$ as

$$SK_{k,B} = K_k^{T/p_B} \bmod M$$

Similar to Method 1, TC (imaginarily) gives subset keys $SK_{k,B}$ with $b_i = 1$ to i^{th} child node of v_k.

Step 3. TC gives a receiver u_j a set of $\log_a N$ master keys $MK_{j,k}$, each of which is a master key of subset keys where the corresponding subsets are defined for a node v_k on $path_j$ and those subset keys are given to its child node which is also located on $path_j$. A master key $MK_{j,k}$ can derive $2^{a-1} - 1$ subset keys. In addition, a master key $MK_{j,1}$ can generate a subset key $SK_{1,11\ldots1}$. The master key $MK_{j,k}$ is defined as

$$MK_{j,k} = K_k^{T/w_{j,k}} \bmod M$$

where $w_{j,k}$ is a product of all primes assigned to the subsets satisfying (i) these subsets are defined for a node v_k and (ii) the corresponding subset keys are given to a child node of v_k where both v_k and the child node are located on $path_j$ (in other words, the leaf assigned to the receiver u_j is also a leaf of a subtree rooted at the child node).

In the example depicted in Fig. 1, the following three master keys are given to a receiver u_1.

master key	corresponding subset keys
$MK_{1,1}$	$SK_{1,100}$, $SK_{1,110}$, $SK_{1,101}$, $SK_{1,111}$
$MK_{1,2}$	$SK_{2,100}$, $SK_{2,110}$, $SK_{2,101}$
$MK_{1,5}$	$SK_{5,100}$, $SK_{5,110}$, $SK_{5,101}$

Revocation. The way of transmitting secret information including revocation is the same as in Method 1.

Decryption. The process in this phase is basically the same as in Method 1. The only difference is the way of deriving a subset key. A receiver u_j derives a subset key $SK_{k,B}$ from its master key $MK_{j,k}$ as follows.

$$MK_{j,k}^{w_{j,k}/p_B} \bmod M = \left(K_k^{T/w_{j,k}}\right)^{w_{j,k}/p_B} \bmod M = K_k^{T/p_B} \bmod M = SK_{k,B}$$

Since $w_{j,k}$ is a product of at most 2^{a-1} primes, this computation requires at most $2^{a-1}-1$ multiplications and one modular exponentiation.[2] The computational overhead for generation of primes is negligible as analyzed in section 4.3.

3 Security of the Proposed Methods

To study the security of our methods, we investigate attacks for the Master Key technique used in these methods. The Master Key system is adopted for the whole HKT in Method 1 whereas it is applied to each internal node in the HKT in Method 2. Suppose that some attackers colluding with each other try to compromise the Master Key system in order to obtain subset keys defined in the targeted system. We consider two cases:

Case I. None of the attackers knows any of subset keys defined in the targeted Master Key system.

Case II. The attackers know at least one subset key defined in the targeted system.

A situation that all attackers are outside of N receivers is regarded as Case I in both of our methods. It is another situation regarded as Case I in Method 2 that at least one of the attackers is a receiver of this revocation scheme but no one of them is assigned to a leaf of a subtree rooted at the node where the targeted Master Key system is applied. Since K in Method 1 and K_k in Method 2 are independent of other systems, subset keys are considered to be indistinguishable from random numbers of length $|M|$ for those attackers in Case I. Therefore we focus on Case II.

In Case II, the attackers know at least one subset key of the targeted Master Key system. Such attackers may include revoked receivers in the targeted system who attempt to obtain subset keys they do not have by breaking the system. We show that our methods are secure against any collusion of revoked receivers under the following assumption related to RSA cryptosystem.

Assumption. *If factors q_1, q_2 of a large composite $M = q_1 q_2$ are unknown then computing p^{th} roots $(\bmod M)$ for integral $p > 1$ is difficult.*

We introduce a theorem and a corollary proven in the appendix to [1].

[2] If we define a subset key $SK_{1,11\ldots1}$ without using the Master Key technique, we have a revocation method with $\frac{\log N}{\log a}+1$ keys and at most 2^{a-2} multiplications at a receiver.

Theorem. *Let t and $t_1, \ldots, t_n$ be given integers and suppose there is a computable function F for which $K^t = F(K^{t_1}, K^{t_2}, \ldots, K^{t_n}) \bmod M$ for every $K \in \mathbb{Z}_M^*$, the group of units $\bmod M$. Let $d = \gcd\{t_i\}$, $e = \gcd(t, d)$ and $p = d/e$. Then we can compute p^{th} roots in $\mathbb{Z}_M^*$.*

Suppose that such a function F exists for $p > 1$, then we can compute nontrivial p^{th} roots in $\mathbb{Z}_M^*$. This is contradictory to the assumption. Therefore the following corollary holds.

Corollary. *Under the assumption above, such function exists only for $p = 1$, namely $\gcd\{t_i\} \mid t$.*

Now we consider the case that some revoked receivers in the targeted Master Key system collude with each other and try to compromise a subset key which is not included in any of the master keys owned by the colluding receivers. Let $\mathbf{G}$ be a set of master keys, which are integer power of $K \bmod M$, $\{MK_1 = K^{t_1} \bmod M, MK_2 = K^{t_2} \bmod M, \ldots, MK_k = K^{t_k} \bmod M\}$. Suppose (i) there exists a function F computing a subset key $SK_m = K^{T/p_m} \bmod M = K^{t_m} \bmod M$ from the set $\mathbf{G}$ and (ii) SK_m is not included in the any master key of $\mathbf{G}$. On one hand, from (i) and the corollary,

$$\gcd\{t_i : MK_i \in \mathbf{G}\} \mid t_m \tag{1}$$

must hold. By definition of a master key, t_i is a product of primes corresponding to the subsets which are not included in the master key MK_i. Since we can write $t_i = \alpha_i p_m$ where $\gcd(\alpha_i, p_m) = 1$ from (ii), we have $\gcd\{t_i : MK_i \in \mathbf{G}\} = \alpha p_m$ where $\gcd(\alpha, p_m) = 1$. On the other hand, since T is a product of all primes corresponding to all subsets, we can write $T = \beta p_m$ where $\gcd(\beta, p_m) = 1$ (i. e. $\beta = t_m$). From (1), we have $\alpha p_m \mid \beta$. However, $\gcd(\alpha, p_m) = \gcd(\beta, p_m) = 1$. This is contradictory, so the assumption that SK_m which is not included in any master key in $\mathbf{G}$ can be derived from $\mathbf{G}$ is wrong. This proves that our methods are secure against any conspiracy of revoked receivers under the assumption described above.

4 Discussions on the Proposed Methods

4.1 The Number of Ciphertexts

Let $\#CT$ denote the number of ciphertexts broadcast in our methods. $\#CT$ is equal to the number of subsets corresponding to subtrees which are left in the revocation phase. In this section we examine its upper bound.

Recall that a sender needs to broadcast one ciphertext encrypted under $SK_{1,11\ldots1}$ if no receivers are revoked. Now we increment the number of revoked receivers one by one. To maximize $\#CT$ we should choose a new revoked receiver such that it shares minimum paths with receivers that have been already revoked, because the shared paths do not contribute to increase the number of the subtrees. Using this strategy, we choose up to $a-1$ revoked receivers as a leaf of subtrees which are rooted at distinct child nodes of the Root. Each picking of revoked receiver increases $\#CT$ by $\log_a N - 1$. When we choose a^{th} revoked receiver, $\#CT$ is increased by $\log_a N - 2$ by choosing a leaf of a subtree rooted at the remaining child node of the Root. Similarly, each addition of $\left[a^{j-1}+1\right]^{\text{th}}$ to $\left[a^{j-1}(a-1)\right]^{\text{th}}$ revoked receiver increases $\#CT$ by $\log_a N - j$, and each addition of $\left[a^{j-1}(a-1)+1\right]^{\text{th}}$ to $\left[a^{j}\right]^{\text{th}}$ revoked receiver increases it by $\log_a N - j - 1$. Therefore we have the upper bound of the number of ciphertexts as follows. For $r < a$ clearly we have $r(\log_a N - 1) + 1$, and for $r \geq a$ we have

$$\begin{aligned}
&1 + \sum_{i=1}^{r} \left(\log_a N - \lceil \log_a i \rceil\right) - \sum_{j=1}^{\lfloor \log_a r \rfloor} \left\{a^j - a^{j-1}(a-1)\right\} \\
&= 1 + r \log_a N - \sum_{i=1}^{r} \lceil \log_a i \rceil - \sum_{j=1}^{\lfloor \log_a r \rfloor} a^{j-1} \\
&= r \log_a N - \left(\lfloor \log_a r \rfloor + 1\right) r + a^{\lfloor \log_a r \rfloor} \\
&< r \log_a N - r\left(\log_a r - 1 + 1\right) + a^{\log_a r} \\
&= r\left(\log_a (N/r) + 1\right) \\
&= r\left(\frac{\log (N/r)}{\log a} + 1\right)
\end{aligned}$$

Since $r(\log_a N - 1) + 1 < r(\log_a (N/r) + 1)$ when $1 \leq r < a$, this upper bound holds for any $r \geq 1$.

4.2 Encryption of Secret Information

Each ciphertext broadcast in our methods is an encryption of secret information I (e.g. a session key) under a subset key. Any encryption algorithm which is considered to be secure can be used for this encryption. For example, we can use a secure block cipher algorithm with the block size $|I|$.

However, the length of a subset key, $|M|$, is equal to the length of a secure modulus of RSA cryptosystem and generally $|M| > |BK|$, where $|BK|$ is the key size of the block cipher algorithm. As introduced in [17], we can use a one-way function $h : \mathbb{Z}_M^* \to \{0,1\}^{|BK|}$ that maps elements which are randomly distributed over $\mathbb{Z}_M^*$ to randomly distributed strings of the desired length [19]. Namely, we can write the encryption of the secret information as $E_{h(SK)}(I)$ where SK and $E_{BK}(m)$ respectively denote a subset key and an encryption of message m under a block cipher algorithm using an encryption key BK. This gives that the size of each ciphertext is reduced to the size of the secret information which is transmitted, regardless of the size of a subset key.

4.3 Representation of Primes

In our methods a receiver needs to use some primes for derivation of a subset key. In this section we present techniques to store or find those primes and evaluate their storage and computational overhead.

Method 1. The total number of primes assigned to the subsets is $(2^a - 2)\frac{N-1}{a-1} + 1$ in Method 1. Since the size of the n^{th} prime is $O(n \log n)$ [12], we roughly estimate that the size of each prime is at most $O(2^a N \log 2^a N)$. In order to derive a subset key $SK_{k,B}$ in the decryption phase, a receiver needs to compute $w_j/p_{k,B}$ which is a product of $\left(2^{a-1} - 1\right)\log_a N$ primes corresponding to the subsets to which the receiver belongs except $p_{k,B}$. If receivers strictly store the primes which are required for derivation of subset keys, they need the storage of $O\left(\left(\frac{2^{a-1}-1}{\log a}\log N + 1\right)(\log N + a + \log(\log N + a))\right)$ bits. Note that since those primes are public, receivers do not need to store them in a confidential manner.

On the other hand, this amount of storage overhead might be too large for some type of receivers. In order to reduce the size of such non-secret storage, we can define the assignment of the primes to the subsets as follows. A prime $p_{k,B}$ corresponding to a subset $S_{k,B}$ is $(B)_2{}^{\text{th}}$ smallest prime larger than $(k-1)L$, where $(B)_2$ denotes a binary number represented by a bit string B and L is a positive integer. Since at most $2^a - 1$ subsets are defined for an internal node in the HKT, L should be large so that an interval $((k-1)L, kL]$ contains at least $2^a - 1$ primes. If a number p is chosen at random, the probability that it is prime is about $1/\ln p$ [23]. Recall that the size of a prime used in the method is at most $O(2^a N \log 2^a N)$. Therefore, if we use L satisfying $L > (2^a - 1)\ln(2^a N \log 2^a N)$, it is expected that the interval $((k-1)L, kL]$ contains at least $2^a - 1$ primes.

Each receiver can compute $p_{k,B}$ from k and B in an on-the-fly manner as follows. From $(k-1)L+1$, a receiver tests each number using a primality testing algorithm until it finds $(B)_2{}^{\text{th}}$ smallest prime. An example of a probabilistic primality testing algorithm is the Miller-Rabin algorithm. Since the complexity of the algorithm for testing a number p is $O\left(\log^3 p\right)$ [23], it is expected that the computational overhead for finding a prime is $O\left(\log^4 p\right)$. A receiver u_j needs to find at most $2^a - 1$ primes (including primes u_j does not use) for each of $\log_a N$ internal nodes on $path_j$, therefore the total computational overhead for generation of primes is roughly

$$O\left(\frac{2^a - 1}{\log a}\log N\left(\log N + a + \log\left(\log N + a + \log\left(\log N + a\right)\right)\right)^4\right)$$

Note that we assume receivers can not store the primes strictly in order to evaluate Method 1.

Method 2. The total number of primes assigned to the subsets in Method 2 is $2^a - 1$. Note that this is much smaller than in Method 1. Since the bit length of the largest prime is roughly $O(a + \log a)$, the size of the storage which is required to store those primes is $O((2^a - 1)(a + \log a))$ bits. It may be possible for receivers to store those primes strictly if a parameter a is chosen reasonably, and we assume it for evaluation of Method 2. Note that since those primes are system-wide universal and public, receivers do not have to store them in a secure non-volatile memory such as a storage for master keys, but they can store them in a usual mask ROM which is used to store program codes.

We also have some ways to reduce the size of the storage. For example, we can define the assignment of those primes such that a prime p_B corresponding to a subset $S_{k,B}$ is $(B)_2$ $^{\text{th}}$ smallest odd prime. Receivers store $A/2$ bits table, with each bit telling them whether or not the corresponding odd number is prime, where A is large enough to consist of $2^a - 1$ primes, such that $A \approx 2^a a$. This table is also system-wide universal and non-secret. This technique is introduced in [12], as well as another way to cut down the size of the storage by listing the size of gaps between primes.

As another option, receivers can compute those primes in an on-the-fly manner. It is easy to find $2^a - 1$ smallest primes for reasonably chosen a, and it requires no storage space.

4.4 Other Properties

In this section we briefly explain other properties that our methods have.

The Number of Revoked Receivers. It is not necessary to fix r, the number of revoked receivers, in advance in our methods. A sender can select an arbitrary group of r $(0 \leq r < N)$ receivers that are revoked at each time of transmission of secret information. Conversely, the sender can choose any select group of receivers as actual recipients of each transmission. The number of ciphertexts broadcast is roughly proportional to r. This is an advantage of the revocation methods using a key tree structure [5, 11, 16, 17, 24, 25] including ours over the methods using a secret sharing scheme [2, 15, 18]. In the latter methods, w which is the upper bound of r must be fixed in advance and the length of broadcast message is $O(w)$ regardless of the number of receivers actually revoked.

Stateless Receivers. In our methods, no receivers need to change their master key(s) in order to revoke or re-entitle receivers. Therefore our methods are suitable for stateless receivers. Suppose a receiver u_j has been revoked during a certain period and is re-entitled to obtain the secret information which will be transmitted afterward. Even if u_j has recorded all messages broadcast during the period and colludes with other receivers which have been also revoked during the period, it can not obtain the secret information sent at that time unless the encryption scheme used to encrypt the secret information is compromised.

Traitor Tracing. As described in section 1.1, a traitor tracing technology is used to find a receiver who contributed for production of a non-legitimate receiver device by giving its private keys. The requirement for tracing traitors proposed in [17] is to find the identities of those that contributed their keys to a non-legitimate receiver and revoke them with still allowing broadcasting to the legitimate receivers. Since our methods have the same property as their methods with respect to the tracing capability such as a bifurcation property, their efficient tracing algorithm is effective in conjunction with our methods. The upper bound of the bifurcation value z is $2/3$ in our methods, therefore the algorithm can be performed with at most $t \log_{1/z} N$ iterations, where t denotes the number of traitors.

5 Modification of CPPM and CPRM

CPPM and CPRM [8] are mechanisms for protection of copyrighted content recorded on pre-recorded or recordable media from unauthorized copying. Those mechanisms contain revocation of receivers (i. e. recorders or players). In this section we show a modification of them using the Master Key technique which reduces the number of keys a receiver stores.[3]

5.1 Brief Description of CPPM and CPRM

In CPPM and CPRM, Trusted Center (TC) which is a sender defines a table with X rows and Y columns and chooses a key $K_{x,y}$ $(x = 1, \ldots, X,\ y = 1, \ldots, Y)$ for each element (x, y) in the table. A compliant receiver u_j is given a unique vector $V_j = (v_1 \ldots v_Y)$ where $v_y \in \{1, \ldots, X\}$ and a set of Y keys $K_{v_1,1}, \ldots, K_{v_Y,Y}$. On the other hand, each compliant pre-recorded or recordable medium is given a Media Key Block (MKB) on its pre-recorded area during its manufacturing process. The MKB is a collection of encryptions of a session key under a key $K_{x,y}$, where the session key is a key used for encryption or decryption of the content file stored on the medium. Note that the MKB does not contain the encryptions under keys $K_{x,y}$ which are given to the revoked receivers. In consequence, the revoked receivers will not be able to obtain the session key from the medium. A non-revoked receiver stores at least one key with high probability which can be used to decrypt a ciphertext in the MKB in order to obtain the session key. This construction gives a revocation method requiring a sender to broadcast at most XY ciphertexts on the medium and each receiver to store Y keys.

5.2 The Modification

Now we modify this method using the Master Key technique. Instead of choosing keys $K_{x,y}$ independently with each other, TC chooses and publishes distinct XY primes $p_{x,y}$ for each element (x, y) in the table, and defines each key $K_{x,y}$ as

$$K_{x,y} = K^{T/p_{x,y}} \bmod M$$

where M is a public value and a product of two large secret primes, K is a secret value chosen randomly from $\mathbb{Z}_M^*$, and T is a product of all primes $p_{x,y}$. Then TC gives a receiver u_j a master key MK_j of Y keys, one key from each column.

[3] Note that discussions on the properties of CPPM and CPRM in this paper are based on section 4.5 of [17].

Each of the keys corresponds to an element (v_y, y) in the table, where v_y is y^{th} element of the vector V_j. The master key MK_j is defined as

$$MK_j = K^{T/w_j} \bmod M$$

where $w_j = \prod_{y=1}^{Y} p_{v_y,y}$ and v_y is y^{th} element of V_j.

Given a master key MK_j, a receiver u_j can derive a key $K_{v_y,y}$ from MK_j as

$$MK_j^{w_j/p_{v_y,y}} \bmod M = \left(K^{T/w_j}\right)^{w_j/p_{v_y,y}} \bmod M = K^{T/p_{v_y,y}} \bmod M = K_{v_y,y}$$

Discussions on the security of the proposed revocation methods in section 3 are also directly suitable for this modification. We can also use techniques described in section 4.2 and 4.3 in order to reduce the size of each ciphertext in the MKB and the size of the storage at a receiver, respectively. This modification reduces the number of keys each receiver stores to only one in exchange for additional computational overhead, i. e. $Y - 1$ multiplications and one modular exponentiation, assuming that receivers store Y primes strictly. Other properties such that the size of the MKB and the upper bound of the number of revoked receivers still remain the same as in the original methods. The size of the MKB, namely, the number of ciphertexts in the MKB is at most XY, and the size of each ciphertext is the size of the session key. Note that as analyzed in [17], since the probability that a legitimate receiver is revoked increases non-negligibly when the number of revoked receivers becomes large, the revocation capability must be bounded by X.

6 Summary

In this paper we have proposed two efficient revocation methods which are suitable for stateless receivers. Our methods use the Master Key technique and the Power Set Method' with an a-ary key tree structure in order to reduce the number of keys each receiver stores and the number of ciphertexts broadcast, respectively. Method 1 requires receivers to store only one key. Method 2 is its variant which reduces the computational overhead of receivers in exchange for an increase in the number of master keys they store. We have discussed the security of our methods and some techniques used in those methods. We also have shown a modification of CPPM and CPRM using the Master Key technique where the number of keys each receiver stores is reduced to one.

Acknowledgments

I am deeply grateful to Dan Boneh for helpful discussions and suggestions. I thank Philippe Golle for many comments regarding the paper. This work would have never started without discussions with Ryuji Ishiguro, Yoshitomo Osawa and Tateo Oishi.

References

1. S. G. Akl and P. D. Taylor, “Cryptographic Solution to a Problem of Access Control in a Hierarchy,” ACM Transactions on Computer Systems, Vol. 1, No. 3, pp. 239-248, 1983.
2. J. Anzai, N. Matsuzaki and T. Matsumoto, “A Quick Group Key Distibution Scheme with Entity Revocation,” Advances in Cryptology - Asiacrypt ’99, Lecture Notes in Computer Science 1716, Springer, pp. 333-347, 1999.
3. S. Berkovits, “How to Broadcast a Secret,” Advances in Cryptology - Eurocrypt ’91, Lecture Notes in Computer Science 547, Springer, pp. 535-541, 1991.
4. D. Boneh and M. Franklin, “An Efficient Public Key Traitor Tracing Scheme,” Advances in Cryptology - Crypto ’99, Lecture Notes in Computer Science, Vol. 1666, Springer-Verlag, pp. 338-353, 1999.
5. R. Canetti, T. Malkin and K. Nissim, “Efficient Communication-Storage Tradeoffs for Multicast Encryption,” Advances in Cryptology - Eurocrypt ’99, Lecture Notes in Computer Science 1592, Springer, pp. 459-474, 1999.
6. G. C. Chick and S. E. Tavares, “Flexible Access Control with Master Keys,” Advances in Cryptology - Crypto ’89, Lecture Notes in Computer Science 435, Springer, pp. 316-322, 1990.
7. B. Chor, A. Fiat and M. Naor, “Tracing Traitors,” Advances in Cryptology - Crypto ’94, Lecture Notes in Computer Science, Vol. 839, Springer-Verlag, pp. 257-270, 1994.
8. “Content Protection for Pre-recorded Media Specification” and “Content Protection for Recordable Media Specification,” available from `http://www.4centity.com/tech/cprm/`.
9. W. Diffie and M. Hellman, “New Directions in Cryptography,” IEEE Transactions on Information Theory, IT-22 (6), 1976.
10. A. Fiat and M. Naor, “Broadcast Encryption,” Advances in Cryptology - Crypto ’93, Lecture Notes in Computer Science 773, Springer, pp. 480-491, 1994.
11. Y. Kim, A. Perrig and G. Tsudik, “Simple and Foult-Tolerant Key Agreement for Dynamic Collaborative Groups,” Proceedings of ACM Conference on Computer and Communication Security, CCS 2000.
12. D. E. Knuth, “The Art of Computer Programming,” vol. 2, Addison-Wesley, 1981.
13. R. Kumar, S. Rajagopalan and A. Sahai, “Coding Constructions for Blacklisting Problems without Computational Assumptions,” Advances in Cryptology - Crypto ’99, Lecture Notes in Computer Science 1666, Springer, pp. 609-623, 1999.
14. M. Luby and J. Staddon, “Combinatorial Bounds for Broadcast Encryptions,” Advances in Cryptology - Eurocrypt ’98, Lecture Notes in Computer Science 1403, Springer, pp. 512-526, 1998.
15. N. Matsuzaki, J. Anzai and T. Matsumoto, “Light Weight Broadcast Exclusion Using Secret Sharing,” Information Security and Privacy - 5th Australasian Conference, ACISP 2000, Lecture Notes in Computer Science 1841, Springer, pp. 313-327, 2000.
16. D. A. McGrew and A. T. Sherman, “Key Establishment in Large Dynamic Groups Using One-Way Function Trees,” Manuscript, available from `http://www.csee.umbc.edu/~sherman/Papers/itse.ps`, 1998.
17. D. Naor, M. Naor and J. Lotspiech, “Revocation and Tracing Schemes for Stateless Receivers,” Advances in Cryptology - Crypto 2001, Lecture Notes in Computer Science 2139, Springer, pp. 41-62, 2001.

18. M. Naor and B. Pinkas, "Efficient Trace and Revoke Schemes," Financial Cryptography '2000, Lecture Notes in Computer Science 1962, Springer, pp. 1-20, 2000.
19. M. Naor and O. Reingold, "Number-Theoretic Constructions of Efficient Pseudo-Random Functions," Proceedings of 38th IEEE Symposium on Foundations of Computer Science, pp. 458-467, 1997.
20. R. Poovendran and J. S. Baras, "An Information Theoretic Analysis of Rooted-Tree Based Secure Multicast Key Distribution Schemes," Advances in Cryptology - Crypto '99, Lecture Notes in Computer Science 1666, Springer, pp. 624-638, 1999.
21. R. L. Rivest, A. Shamir and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," Communications of the ACM, 21, pp. 120-126, 1978.
22. A. Shamir, "How to Share a Secret," Communications of the ACM, 22, pp. 612-613, 1979.
23. D. R. Stinson, "Cryptography: Theory and Practice," CRC Press, 1995.
24. D. Wallner, E. Harder and R. Agee, "Key Management for Multicast: Issues and Architectures," IETF Network Working Group, Request for Comments: 2627, available from `ftp://ftp.ietf.org/rfc/rfc2627.txt`, 1999.
25. C. K. Wong, M. Gouda and S. S. Lam, "Secure Group Communications Using Key Graphs," Proceedings of ACM SIGCOMM '98, 1998.