Permanent Revocation in Attribute Based Broadcast Encryption

(Extended Abstract)

Shlomi Dolev*, Niv Gilboa† and Marina Kopeetsky‡
*Department of Computer Science
Ben-Gurion University of the Negev,Beer-Sheva, 84105, Israel
Email: dolev@cs.bgu.ac.il
†Department of Communication Systems Engineering
Ben-Gurion University of the Negev,Beer-Sheva, 84105, Israel
Email: niv.gilboa@gmail.com
‡Department of Software Engineering
Sami Shamoon College of Engineering
Beer-Sheva, 84100, Israel
Email: marinako@sce.ac.il

***Abstract*—We propose a new and efficient scheme for broadcast encryption. A broadcast encryption system allows a broadcaster to send an encrypted message to a dynamically chosen subset RS, $|RS| = n$, of a given set of users, such that only users in this subset can decrypt the message. An important component of broadcast encryption schemes is revocation of users by the broadcaster, thereby updating the subset RS. Revocation may be either temporary, for a specific ciphertext, or permanent.**

We present the first public key broadcast encryption scheme with permanent revocation of users, unlike all previous public key schemes that support temporary revocation. Our approach is especially appealing in applications in which once a user is revoked that user should not be able to decrypt any subsequent messages. Our scheme is fully collusion-resistant. In other words, even if all the revoked users collude, the revoked user cannot encrypt messages without receiving new keys from the broadcaster.

The overhead of revocation in our system is constant in all major performance measures including length of private and public keys, computational complexity, user's storage space, and computational complexity of encryption and decryption.

The scheme we construct improves on our original scheme in a poster presentation [7] by a factor of $O(\log n)$ in all major performance measures.

I. Introduction

The concept of broadcast encryption was first introduced in [9] and further developed in many works including [15], [12], [2], [10], [8] and [13]. Broadcast encryption systems allow a broadcaster to send encrypted data to a set of users such that only a subset RS of n *authorized* users can decrypt the data. A main challenge in constructing broadcast systems is ensuring that even when the users that are not in RS collude, it is computationally infeasible to decrypt a message.

A poster on a subset of the results appeared in the *ACM Conference on Computer and Communications Security*, pp.757-760, 2011. Partially supported by Rita Altura Trust Chair in Computer Sciences, Lynne and William Frankel Center for Computer Sciences, and the internal research program of the Shamoon College of Engineering.

Broadcast encryption systems support *temporary* revocation of r users if revoked users are excluded from the set RS for a single ciphertext. Typically, in such systems, the identities of the revoked users are parameters in the encryption mechanism. Hence, the broadcaster have to keep track of the revoked users forever, using additional memory and computations. Broadcast encryption systems support *permanent* revocation of r users if revoked users cannot decrypt any ciphertext after the revocation. Permanent user revocation is efficiently implemented in symmetric encryption schemes (e.g. the third scheme of [8]). Temporary revocation is achieved by various schemes including [13] and the first two schemes of [8].

Broadcast encryption systems are either stateful or stateless. A stateful scheme requires receivers to store a state and update it based on the ciphertexts they receive. Stateless receivers do not necessarily update a state. Stateless schemes are preferable in the sense that receivers do not have to be continuously online to correctly decrypt a transmission. However, stateful schemes open new avenues to achieve permanent revocation by basing decryption on the state and not enabling revoked users to correctly update a state. Furthermore, broadcast models in which the receivers can open a two-way channel to the broadcaster are becoming more prevalent, e.g., IPTV and Over-The-Top broadcasting. Given such two-way channels, receivers can update their state even if they go offline for a time. Current broadcast encryption schemes that are actually deployed, e.g. for DVB or IPTV rely on the broadcaster storing a different key for each user. The broadcaster sends a shared content key (called a "control word") to every authorized user, by encrypting it with the user's unique key. The content key is used to encrypt and decrypt content. Permanent revocation in this scheme is trivial; the broadcaster simply does not encrypt the content key with a revoked user's key. However, the ciphertext is always $O(n)$-bits long. In contrast, the ciphertext

978-0-7695-5014-5/12 $26.00 © 2012 IEEE
DOI 10.1109/CyberSecurity.2012.33

in our scheme is $O(r)$ bits, where r is the number of revoked users in the time interval since the last revocation.

A trivial solution for constructing collusion resistant broadcast system works as follows. The broadcaster maintains n independent encryption keys, while each user is granted its personal decryption key. The broadcaster encrypts each message with all the encryption keys. Since the keys are independent, collusion resistance is satisfied for any number of revoked colluding users. Obviously, this scheme is not efficient in the number of encryption/decryption keys, size of broadcaster storage, and cost of encryption/decryption procedure.

Stateful symmetric encryption schemes. Protocols for stateful receivers have been introduced and analyzed in [11], [16], [3], [4], [19], [17] and in the third construction of [8]. The drawback of this approach is that only users that have the secret key, can receive and decrypt the broadcast messages.

Most of the stateful symmetric encryption schemes are based on graph theoretical constructions, and support permanent revocation of a single user or a group of users.

The best schemes of [17] require $\log n$ keys per update, linear server (broadcaster) storage of $2n - 1$ keys, and logarithmic user storage of $\log n$ keys. Moreover, in our scheme, the server stores $2 \log n + 1$ keys, which is a much smaller number of keys than the number of keys stored by stateful symmetric key schemes.

The most efficient scheme of [8], based on Generalized Decisional Diffie-Hellman Exponent (GDDHE) assumption (Construction 3) provides users' revocation with the symmetric encryption and decryption keys of constant size, and ciphertexts length of order $O(r)$, where r denotes the number of revoked users. Construction 3 of [8] supports users permanent revocation.

Stateless schemes. Stateless broadcast encryption schemes are based on either symmetric-key or public-key schemes, as we now detail.

Stateless Symmetric key schemes.

The protocols of [15] and [12] are based on a graph theoretic approach and provide temporary revocation of a single user or a group of users. The scheme of [12], based on the Layered Subset Difference technique, improves the results of [15], and shows that for any $\epsilon > 0$ one can create an efficient broadcast scheme (that supports users' revocation) with $O(\log^{1+\epsilon} n)$ keys, $O(r)$ messages, and $O(\log n)$ cryptographic operations. Here $r < n$ denotes a number of revoked users.

The use of symmetric key cryptosystems restricts the solutions presented in [8] in the sense that only the server (or central module) may broadcast the sensitive data.

Stateless public key schemes. The mostly used approach in creating collusion resistant broadcast or revocation systems is based on hardness of decisional algebraic problems in the groups of elliptic curves (for example Bilinear Decisional Diffie-Hellman (BDDH) problem). The broadcast encryption schemes for stateless receivers based on bilinear maps were proposed in [2], and further developed in [10]. The consequent constructions are compared regarding the efficiency parameters such as encryption/decryption keys and ciphertext sizes, and time complexity. Two constructions, based on bilinear maps, were introduced in [10]. In the first construction the ciphertext and private keys are of constant size, while the public key length is linear in the total number of receivers. The second construction achieves trade off between the ciphertext and the public key length when both of them are of order $O(\sqrt{n})$ for any subset of receivers from a system of n users. The system uses constant size ciphertexts.

A powerful technique for public-key, broadcast encryption systems, is Attribute Based Encryption (ABE) (e.g., [5], [14]). The purpose of ABE is to establish access policy for decrypted data among users of a given set.

ABE was proposed in [18] as means for encrypted access control. The main idea of the ABE system is that ciphertexts are not necessarily encrypted for one particular user but encrypted for sets of users. Unlike traditional public-private key cryptography, user's private keys and ciphertexts are associated with a set of attributes that a user possesses. A user can decrypt a ciphertext if and only if he/she has a corresponding set of attributes associated with a security policy. In the Ciphertext Policy Attribute Based Encryption (CP-ABE) a user have to posses a certain set of attributes in order to access the data. We use CP-ABE techniques introduced and analyzed in [2] as a building block. The purpose of ABE is to establish access policy on who can decrypt data among users of a given set. The number of keys used in ABE is logarithmic in the number of users, which provides the smallest possible number of keys [6]. ABE ensures collusion resistance for any number of the colluding revoked users. The main idea of the CP-ABE is that a user's private key is associated with (an arbitrary number of) attributes. A user is able to decrypt a ciphertext if there is a match between his/her attributes and ciphertext's access structure.

The scheme closest to ours is introduced in [5]. As in [5] we deal with the simplified access structure that has only **AND** and **OR** gates where only depth-1 access trees are considered.

The proof of the basic schemes of [5] appears in [6]. In addition the basic scheme is optimized in [6] by introducing a hierarchical structure of the attributes. Unlike our scheme and like other ABE based revocation systems, the scheme of [5] provides only temporary revocation of users.

Efficiency of the broadcast encryption scheme. The efficiency is measured in terms of required server/user storage space, computational complexity of key update procedure and the number of messages sent upon join or revocation events.

Optimal efficiency is achieved for public key with temporary revocation in [13] and for symmetric key with permanent revocation in [8]. In both works, the encryption/decryption keys are of constant size, ciphertext size is of $O(r)$, where r is the number of revoked users, and the computational complexity of a key update procedure is $O(r)$.

The permanent revocation achieved in our best scheme, a transformation of [13] to a scheme with permanent revocation, requires public and private keys of length $O(1)$, while the length of a ciphertext to revoke r users is $O(r)$. The compu-

Revocation	Public key	Symmetric key
Temporary	[2], [5], [14], [10], [13]	[15], [12]
Permanent (and stateful)	Our scheme	[11], [16], [3], [4], [19], [17] [8]

TABLE I
CLASSIFICATION OF BROADCAST ENCRYPTION RESULTS

tational complexity of a key update is also $O(r)$.

Our contributions. We propose the first efficient public-key encryption scheme that supports permanent user revocation. We use Ciphertext Policy ABE (CP-ABE) techniques introduced and analyzed in [1] as a building block and extend it to support permanent revocation. Any user in [1] is assigned a set of attributes and can decrypt any ciphertext that embeds a policy, which satisfies the user's attributes. Furthermore, any coalition of users cannot decrypt a ciphertext if none of the user's attributes satisfies the policy.

A previous broadcast encryption work [5] bases broadcast encryption on CP-ABE. However, each revocation is temporary since sequentially revoked users (identified with different sets of attributes) can share their attribute keys and reconstruct the keys updated after their revocation. We eliminate this problem in such a way that any revoked user/users cannot collude to decrypt any ciphertext broadcast after the revocation.

The main advantages of our scheme are:

• We propose the first efficient public-key encryption scheme that supports permanent users' revocation. The identities of the revoked users are permanently excluded (upon key update procedure) from the encryption mechanism. Unlike existing public-key based schemes, the broadcaster in our scheme does not have to keep track of revoked users. Previous schemes that enabled permanent revocation were all based on symmetric keys: e.g., scheme 3 of [8] and [11].

• By providing permanent users' revocation, we treat the more complex notion of collusion when a previously revoked user U_i can get private information (including secret keys) from a later revoked user U_j (or set of such revoked users). Hence, our schemes cope with stronger adversary, compared with the previous public key schemes e.g., [2], [5]. The penalty we pay is that our scheme is stateful and hence all the participating users must be permanently on-line (or updated about the sessions they missed).

• There is no change in the public key upon executing the *Join* procedure, and *Join* may be efficiently implemented in $O(\log n)$ time complexity (it should be noted that the best implementation is introduced in [8] that requires $O(1)$ time complexity). We use an efficient key update based on the CP-ABE techniques, that is executed by the server (broadcaster).

• The efficiency of revoking r users in our best scheme is $O(r)$ in communication and time and is therefore worse by at most a constant factor from optimal revocation.

• Our scheme is based on maintaining a state for each participant. The technique we propose may support transformation of other ABE based broadcast encryption schemes into schemes that support permanent revocation, in particular the transformation of the (simple construction) scheme of [13].

Organization In section II we present a first construction for public-key broadcast encryption with permanent revocation. This first scheme has overhead $O(\log n)$ in ciphertext size, secret key size and computation time. In section III we construct the second and improved construction for public-key broadcast encryption with permanent revocation. This second scheme has overhead $O(1)$ in all performance measures. In section IV we discuss a generic transformation of ABE-based schemes for broadcast encryption with temporary revocation into broadcast encryption with permanent revocation.

II. BROADCAST ENCRYPTION WITH PERMANENT REVOCATION

Our first scheme uses CP-ABE [1] in a way that support users' permanent revocation. The main idea is to change the state of each non revoked user by updating the master key MK and the secret key SK_i of each user in a way that all the users except the revoked user U_j can decrypt the ciphertext and no coalition of users (that record the messages after the exclusion of U_j) can assist in updating SK_j and computing the new secret master key.

The scheme proceeds as follows:

• Each user is defined by a unique combination of attributes, e.g. the bits in a binary representation of the user's ID. Each user receives attribute keys that enable sending a public-key encrypted message to be decrypted by any subset of users, see [5] for details. The broadcaster authorizes a subset of receivers RS by broadcasting the global secret decryption key K_{global}. This key is encrypted by the appropriate attribute keys for RS (according to the ABE system). The broadcaster may then encrypt bulk data using K_{global}.

• Each user from the receiver set RS maintains the state $State_i$ that is defined as a value of a certain function over a secret counter variable CTR: $State_i = f_i(CTR)$.

• When a user U_j is revoked from the receivers set RS, the broadcaster updates the counter variable CTR to a new secret value $\widetilde{CTR}$, and broadcasts its encrypted value to all non revoked users. As a result, the state of each user U_i, $U_i \in RS - \{U_j\}$ is updated to $State_i = f_i(\widetilde{CTR})$. Thus, the encryption key and ciphertext generated by the broadcaster, and appropriate secret encryption key K_{global} are updated.

• Each joined user receives fresh previously unused attribute keys from the broadcaster. The initiated by the broadcaster **Setup** procedure is, in essence, the random algorithm that involves a random string. Due to the randomization, performed during **Setup**, a previously revoked and joined after the revocation user, gets completely fresh attribute keys. These keys may be the attribute keys correspondingly to the same (before revocation and after join) access structure.

This update is performed in such a way that even a coalition of all users from the new set of receivers RS cannot collude in order to reveal the updates after U_j's revocation $State_j = f_j(\widetilde{CTR})$.

Consider the basic construction of the CP-ABE system ([1]). Let G_0 be a bilinear group of prime order p, and let g be a random generator of G_0. Let $e : G_0 \times G_0 \rightarrow G_1$ be a proper bilinear map. The security parameter k denotes the size of the groups. Let M be a secret message that should be encrypted and sent by the broadcaster to the users from the set $RS - \{U_j\}$.

The order of the performed actions is as follows. Firstly, the broadcaster runs the **Setup** algorithm that generates the public key PK and the master key MK. Next, the **Key generation** procedure outputs the attribute secret keys for the set of attributes that identifies the corresponding access structure T. The attribute secret key SK is unique for each user (from the receiver set RS) whose attributes satisfy T. In essence, the encryption of a message (global key in our case) is a certain one way function of the set of attributes and a user. The uniqueness of the SK for each user is satisfied by the randomness that the broadcaster inserts in the secret key for each user during the **Key generation** procedure, and the random updating of SK by each user upon revocation event (that changes the access structure). Finally, the broadcaster encrypts a message (global key) via the **Encrypt** procedure. The constructed ciphertext is, in essence, a certain one way function of the attributes which satisfy a given access structure T (for a given receiver set RS). It should be noted that ciphertext CT is unique for each user from RS, and it does not depend on a specific user. Our modifications of the basic scheme of [1] are as follows:

Setup. Choose G_0, g, and two random elements $\alpha, \beta \in Z_p$. The public key is published exactly as in [1]: $PK = G_0,\ g,\ h = g^{\beta}, e(g,g)^{\alpha}$. The master key MK includes our new random component $CTR \in Z_p$: $MK = \beta,\ g^{\alpha}, CTR$.

Key generation (MK, S). The input of the algorithm is a set of attributes S, and the output is a secret key that identifies the set. Two random numbers r_i and r_{ij} are chosen from Z_p for each user U_i and attribute $j \in S$, respectively. The component E_i encodes the state of U_i, which is a function of CTR.

It should be noted that the users maintain distinct states. The private key of U_i is:

$$\left\{ D = g^{\frac{\alpha + r_i}{\beta}},\ E_i = e(g,g)^{r_i \cdot CTR}, \forall\ j \in S : D_j = g^{r_i} H(j)^{r_{ij}},\ D'_j = g^{r_{ij}} \right\}$$

Encrypt. The encryption procedure encrypts a message M under the access structure (AS) $T = RS - \{U_j\}$ (see [1] and [5] for a simplification of AS). For each node x (including the leaves) a polynomial q_x is properly defined (see [1] for the encryption details). Starting with the root node R, a random secret for sharing $s \in Z_p$ is chosen and the root polynomial is defined in 0 as $q_R(0) = s$. It should be noted that the secret s and its corresponding shares are changed (decremented by assigning CTR) in our modification.

Set $s_2 = -s - CTR \bmod p$ and construct the ciphertext CT as:

$$CT = (T = RS - \{U_j\},\ \widetilde{C} = Me(g,g)^{\alpha s_2}$$

$$C = h^{s_2}, \forall\ y \in Y : C_y = g^{q(0)},\ C'_y = H(att(y))^{q_y(0)}).$$

Here Y denotes the set of leaf nodes in T and H is a cryptographic proper hash function.

Decryption. The decryption procedure performed by each user that possess a set of attributes corresponding to T is as follows: First, the user computes $A_i = e(g,g)^{r_i s}$, by using the DecryptNode procedure of [1]. Then,

$$M = \widetilde{C}/(e(C,D) \cdot A_i \cdot E_i)$$

since

$$e(C,D) = e\left(g^{\beta s_2}, g^{\frac{\alpha + r_i}{\beta}}\right) = e(g,g)^{(\alpha + r_i)s_2} = e(g,g)^{\alpha s_2} \cdot e(g,g)^{r_i s_2} = e(g,g)^{\alpha s_2} \cdot e(g,g)^{r_i(-s - CTR)}.$$

Hence,

$$e(C,D) \cdot E_i = e(g,g)^{\alpha s_2} \cdot e(g,g)^{-r_i s}.$$

As a result,

$$e(C,D) \cdot E_i \cdot A_i = e(g,g)^{\alpha s_2}.$$

Finally,

$$M = \widetilde{C}/(e(C,D) \cdot A_i \cdot E_i).$$

The broadcaster updates CTR in MK by $CTR \leftarrow CTR + s \bmod p$. The user updates E_i in its private key by

$$E_i \leftarrow E_i \cdot A_i = e(g,g)^{r_i CTR} e(g,g)^{r_i s} = e(g,g)^{r_i(CTR+s)}.$$

Unlike previous CP-ABE based schemes (e.g., [5]), the users' attribute keys in our scheme remain constant regardless of the possible revocations, whereas only a global state CTR and corresponding functions of CTR are updated.

Once a user U_j is revoked, it cannot compute its function of CTR, $e(g,g)^{r_i \cdot CTR}$ even with the collusion of every other user. Thus, the revocation is permanent.

III. Permanent Revocation with $O(1)$ Overhead

The **Setup** is performed as in the basic scheme of [13] without modifications. A proper group G_0 of a prime order p, two random generators $g,\ h \in G_0$, and two random secret numbers $a,\ b \in Z_p$ are chosen. The bilinear transformation e is defined as in [13]. The public key is published as $PK = \left(g, g^b, g^{b^2}, h^b, e(g,g)^a\right)$. The secret master key of the broadcaster MK includes the new random component $CTR \in Z_p$: $MK = (a,\ b,\ CTR)$.

The **Key generation** algorithm chooses a random $t \in Z_p$ (as in [13]) and publishes the secret private key (that identifies the set of the corresponding attributes) as

$$SK = (D_0 = g^a g^{b^2 t};\ D_1 = (g^{bID} h)^t,$$

$$D_2 = g^{-t}; E = e(g,g)^{CTR \cdot b^2 t})$$

As it was mentioned above, the component E encodes the state, which is a function of CTR. Here ID denotes the identity of the non-revoked user.

The encryption procedure is modified in the following way. As in [13], the **Encrypt** algorithm first picks a random secret $s \in Z_p$. It should be mentioned that s will be updated by the broadcaster upon user's (or users') revocation. As in [13], s is split into t shares as $s = s^{(1)} + \ldots\ s^{(r)}$. Let ID_i denotes the $i-th$ identity in the revocation set $R = \{ID_1, \ldots,\ ID_r\}$ of r revoked users. Upon the revocation of r determined above users, the broadcaster updates secret s as $s_2 = s + CTR \bmod p$ and splits s_2 as $s_2 = s_2^{(1)} + \ldots\ s_2^{(r)}$. The constructed ciphertext CT has the following structure:

$$CT = (\widetilde{C} = e(g,g)^{as}M,\ C_0 = g^{s_2},$$

$$\forall i = 1, \ldots, r, C_{i,1} = g^{bs_2^i}, C_{i,2} = (g^{b^2 ID_i} h^b)^{s_2^i}).$$

The **Decrypt** procedure, provided by each non-revoked user U_i is performed as in [13]. The major difference is that the secret s is updated (by adding the CTR variable) per each revocation event. The computation is correctly defined $\forall i\ ID \neq ID_i$.

$$\frac{e(C_0, D_0)}{e(D_1, \prod_{i=1}^r C_{i,1}^{1/(ID-ID_i)}) \cdot e(D_2, \prod_{i=1}^r C_{i,2}^{1/(ID-ID_i)})} =$$

$$\frac{e(g^{s_2}, g^a g^{b^2 t})}{e(g,g)^{b^2 t}} = \frac{e(g^{s+CTR}, g^a g^{b^2 t})}{e(g,g)^{b^2 t}} =$$

$$\frac{e(g,g)^{(s+CTR)a} \cdot e(g,g)^{(s+CTR)b^2 t}}{e(g,g)^{sb^2 t}} =$$

$$e(g,g)^{(s+CTR)a} \cdot e(g,g)^{CTRb^2 t}$$

The product $e(D_1, \prod_{i=1}^r C_{i,1}^{1/(ID-ID_i)}) \cdot e(D_2, \prod_{i=1}^r C_{i,2}^{1/(ID-ID_i)})$ is equal to $A = e(g,g)^{b^2 t}$, and it is is computed by each non-revoked user (defined by the identity ID). As a result of the decryption procedure, we get the entire secret message M is revealed as follows:

$$M = \frac{\widetilde{C} \cdot E \cdot A}{e(C_0, D_0)}.$$

The broadcaster updates CTR in MK by $CTR \leftarrow CTR + s \bmod p$. The user updates E in its private key by

$$E \leftarrow E \cdot A = e(g,g)^{CTR \cdot b^2 t} e(g,g)^{s \cdot b^2 t} = e(g,g)^{(CTR+s)b^2 t}.$$

As in the scheme presented in Section II, any revoked user, say U_j, cannot compute its function of CTR, $e(g,g)^{CTR \cdot b^2 t}$. Hence, the revocation is permanent.

IV. Generic Transformation

We informally discuss a generic procedure for transforming an ABE based broadcast encryption scheme with temporary revocation into a schemes that supports permanent revocation of users. Each non-revoked user possesses a state, which is changed upon revocation of a certain user or a group of users. The change of a state of any non revoked user is performed by updating the secret master key MK by the broadcaster, and a corresponding update of the secret key SK_i of each non-revoked user U_i (based on $U_i - th$ state). As a result of this procedure, all users except the revoked U_j can decrypt the ciphertext and no coalition of users (that record the messages after the exclusion of U_j) can assist in updating SK_j and computing the new secret master key MK.

The generic scheme for integration of the permanent revocation into any ABE based scheme includes the following steps into the defined above encryption procedure:

Setup. This algorithm chooses a bilinear group G_0 of prime order p, a proper bilinear map $e : G_0 \times G_0 \rightarrow G_1$, a random generator g (or generators g and h), and random exponents $a,\ b \in Z_p$ (see [2], [5], [13]). The output of *Setup* is the public key PK. PK securely encapsulates the random secrets a and b. In all schemes with temporary revocation the secret master key MK includes the random secrets used for the PK generation. For example, $MK = (b,\ g^a)$ in [2] and [5], and $MK = (a, b)$ in [13]. In order to perform generic transformation from temporary to permanent revocation, the additional secret random component CTR is added to MK. The encoding of a user's state is based on the new counter variable CTR.

Key generation (MK, S). The key generation algorithm takes as input a set S of predefined attributes, and outputs a secret SK, known to all non-revoked users (users that posses the attributes set S) . It should be noted that S may be defined differently, based on the considerations of network management system. In order to construct a scheme with permanent users' revocation, the state encoding component E is included into SK. E securely encapsulates the state variable CTR. Due to the randomness used for the generating of E, the non-revoked users, that posses the same attribute set S, have distinct states ($E = e(g,g)^{rCTR}$ in our generic scheme applied to the schemes of [2] and [5], and $E = e(g,g)^{b^2 tCTR}$ in [13]). Here r and t are randomly chosen by each user in [2], [5] and [13], respectively.

Encrypt. The input of this algorithm is the public key PK, a message $M \in G_1$, and a corresponding access structure AS. The output of the *Encrypt* procedure is a ciphertext CT. According to our modification, the secret s shared among the non-revoked users, is updated upon a revocation event as $s_2 = -s - CTR$. The general encryption procedure of [2], [5], and [13] is not modified. The main point of our modification is that a new secret value (modified by a broadcaster) is shared between the non-revoked users from the updated set of attributes S.

Decrypt. After the decryption, performed by each user (which

possess a set of attributes corresponding to the $AS\ T$), the broadcaster updates CTR in MK by $CTR \leftarrow CTR + s \bmod p$. As a result, each user updates E in its private key. Due to the fact that the random exponent is generated by each user in an independent way, the state, encoded by E, is distinct for each user.

Once a user U_j is revoked, it cannot compute its function of CTR and, even with the collusion of every other user. Thus, the revocation is permanent.

V. Conclusions

We present the first public key CP-ABE based broadcast encryption scheme that supports permanent revocation of users. Our scheme is fully collusion-resistant for any number of the colluding users.

The overhead of our first system is $O(1)$ in all major performance measures including length of private and public keys, computational complexity, user's storage space, and computational complexity of encryption and decryption.

References

[1] J. Bethencourt, A. Sahai, B. Waters,"Ciphertext-Policy Attribute Based Encryption", *IEEE Symposium on Security and Privacy* (SP '07), pp. 321-334, 2007.

[2] D. Boneh, C. Gentry, B. Waters, "Collusion Resistant Broadcast Encryption With Short Ciphertexts and Private Keys", *25-th Annual International Cryptology Conference CRYPTO 2005*, USA, 2005. In *Lecture Notes in Computer Science*, volume 3621, pp. 258-275.

[3] R. Canetti, J. Garay, G. Itkis, D. Micciancio, M. Naor, B. Pinkas, "Multicast Security: A Taxonomy and Some Efficient Constructions", *INFOCOM'99*, Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Proceedings, volume 2, pp. 708-716, 1999.

[4] R. Canetti, T. Malkin, K. Nissim, "Efficient Communication-Storage Tradeoffs for Multicast Encryption", *EUROCRYPT'99, LNCS1592*, pp. 459-474, 1999.

[5] L. Cheung, J. A. Cooley, R. Khazan, C. Newport, "Collusion Resistant Group Key Management Using Attribute Based Encryption", *Cryptology ePrint Archive*, Report 2007/161, 2007. Presented at GOCP 07.

[6] L. Cheung, C. Newport, "Provably Secure Ciphertext Policy ABE", *Proceedings of the 14th ACM conference on Computer and communications security* (CCS), pp. 456-465, 2007.

[7] S. Dolev, N. Gilboa, M. Kopeetsky, "Attribute Based Encryption with Permanent Revocation", *CCC2011*, Poster presentation, Chicago, USA, 2011.

[8] C. Delerablee, P. Paillier, D. Pointcheval, "Fully Collusion Secure Dynamic Broadcast Encryption with Constant-Size Ciphertexts or Decryption Keys", *Proceedings of the first International Conference on Pairing-based Cryptography*, LNCS 4575, pp. 39-59, Springer-Verlag, July 2007, Tokyo, Japan.

[9] A. Fiat, M. Naor, "Broadcast Encryption". In Douglas R. Stinson, editor, *CRYPTO'93*, volume 773 of *LNCS*, pp. 480-491, CA, USA, 1994. Springer-Verlag, Berlin, Germany.

[10] C. Gentry, B. Waters, "Adaptive Security in Broadcast Encryption Systems", In *Eurocrypt*, 2009.

[11] H. Harney, E. Harder, "Logical Tree Hierarchy Protocol", *Internet Draft, Internet Engineering Task Force*, April, 1999.

[12] D. Halevy, A. Shamir, "The LSD Broadcast Encryption Scheme", *CRYPTO 2002*, LNCS 2442, pp. 47-60, 2002.

[13] A. Lewko, A. Sahai, B. Waters, "Revocation Systems with Very Small Private Keys", In *IEEE Symposium on Security and Privacy*, pp. 273-285, 2010.

[14] D. Lubicz, T. Sirvent, "Attribute-Based Broadcast Encryption Scheme Made Efficient", In *AFRICACRYPT*, LNCS, volume 5023, pp. 342-325, 2008.

[15] D. Naor, M. Naor, J. Lotspiech, "Revocation and Tracing Schemes for Stateless Receivers", *CRYPTO 2001*, LNCS, vol. 2139, pp. 41-62, 2001.

[16] A. Perrig, D. Song, J. D. Tygar, "ELK, a New protocol for Efficient Large-Group Key Distribution", *IEEE Symposium on Security and Privacy 2001*, Proceedings, pp. 247-262, 2001.

[17] A. T. Sherman, D. A. McGrew, "Key Establishment in Large Dynamic Groups using One-Way Function Trees", *IEEE Transactions on Software Engineering*, no. 29, volume 5, pp. 444-458, 2003.

[18] A. Sahai, B. Waters, "Fuzzy Identity Based Encryption", *Advances in Cryptology- Eurocrypt*, volume LNCS 3494, pp. 457-473, Springer, 2005.

[19] C. K. Wong, M. Gouda, S. Lam, "Secure Group Communications Using Key Graphs", *IEEE/ACM Transactions on Networking*, volume 8, no. 1, February, 2000.