International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 4 | ISSN : 2456-3307

# Contributory Broadcast Coding with Efficient Coding and Short Ciphertext

[1]Yadavalli Gopi, [2]Surrisetty Chandrika

[1]Assistant Professor , Department Of MCA , St. Mary's Group of Institutions, Guntur, Andhra Pradesh, India.
[2]PG Student ,Department of MCA , St. Mary's Group of Institutions, Guntur, Andhra Pradesh, India

## ABSTRACT

Traditional broadcast encryption (BE) schemes allow a sender to securely broadcast to any subset of members but require a trusted party to distribute decryption keys. Group key agreement (GKA) protocols enable a group of members to negotiate a common encryption key via open networks so that only the group members can decrypt the ciphertexts encrypted under the shared encryption key, but a sender cannot exclude any particular member from decrypting the ciphertexts. In this paper, we bridge these two notions with a hybrid primitive referred to as contributory broadcast encryption (ConBE). In this new primitive, a group of members negotiates a common public encryption key while each member holds a decryption key. A sender seeing the public group encryption key can limit the decryption to a subset of members of his choice. Following this model, we propose a ConBE scheme with short ciphertexts. The scheme is proven to be fully collusion-resistant under the decision n-Bilinear Diffie-Hellman Exponentiation (BDHE) assumption in the standard model. Of independent interest, we present a new BE scheme that is aggregately. The aggregatability property is shown to be useful to construct advanced protocols.

**Keywords:** Broadcast Encryption, Group Key Agreement, Contributory Broadcast Encryption, Provable Security

## I. INTRODUCTION

With the increase in technology advancement in communication technologies, there is an increasing demand of versatile cryptographic primitives to protect group communications and computation platforms. These new platforms include instant-messaging tools, collaborative computing, mobile ad hoc networks and social networks. These new applications call for cryptographic primitives allowing asunder to securely encrypting to any subset of the users of the services without relying on a fully trusted dealer. Broadcast encryption (BE) is a well-studied primitive intended for secure group-oriented communications. It allows a sender to securely broadcast to any subset of the group members. Nevertheless, a BE system heavily relies on a fully trusted key server who generates secret decryption keys for the members and can read all the communications to any members. Group key agreement (GKA) is another well-understood cryptographic primitive to secure group-oriented communications. A conventional GKA allows a group of members to establish a common secret key via open networks. However, whenever a sender wants to send a message to a group, he must first join the group and run a GKA protocol to share a secret key with the intended members more recently, and to overcome this limitation, with the introduction of asymmetric GKA, in which only a common group public key is negotiated and each group member holds a different decryption key.

However, neither conventional symmetric GKA nor the newly introduced asymmetric GKA allow the sender to unilaterally exclude any particular member from reading the plaintext. Hence, it is essential to find more flexible cryptographic primitives allowing dynamic broadcasts without a fully trusted dealer. This paper investigates a close variation of the above mentioned problem of one-round group key agreement protocols and focuses on "how to establish a confidential channel from scratch for multiple parties in one round". We provide a short overview of some new ideas to solve this variation. Asymmetric GKA Observe that a major goal of GKAs for most applications is to establish a confidential broadcast channel among the group. We

investigate the potentiality to establish this channel in an asymmetric manner in the sense that the group members merely negotiate a common encryption key (accessible to attackers) but hold respective secret decryption keys. We introduce a new class of GKA protocols which we name asymmetric group key agreements (ASGKAs), in contrast to the conventional GKAs. A trivial solution is for each member to publish a public key and withhold the respective secret key, so that the final ciphertext is built as a concatenation of the underlying individual ones. However, this trivial solution is highly inefficient: the ciphertext increases linearly with the group size; furthermore, the sender has to keep all the public keys of the group members and separately encrypt for each membere. We are interested in nontrivial solutions that do not suffer from these limitations. Group key agreement (GKA) is another well-understood cryptographic primitive to secure group-oriented communications. A conventional GKA allows a group of members to establish a common secret key via open networks. However, whenever a sender wants to send a message to a group, he must first join the group and run a GKA protocol to share a secret key with the intended members. More recently introduced asymmetric GKA in which only a common group public key is negotiated and each group member holds a different decryption key.

However, neither conventional symmetric GKA nor the newly Introduced asymmetric GKA allow the sender to unilaterally exclude any particular member from reading the plaintext1. Hence, it is essential to find more flexible cryptographic primitives allowing dynamic broadcasts without a fully trusted dealer.

**A. Our Contributions**

We present the Contributory Broadcast Encryption (ConBE) primitive, which is a hybrid of GKA and BE.

Compared to its preliminary Asiacrypt 2011 version, this full paper provides complete security proofs, illustrates the necessity of the aggregatability of the underlying BE building block and shows the practicality of our ConBE scheme with experiments. Specifically, our main contributions are as follows. First, we model the ConBE primitive and formalize its security definitions. ConBE incorporates the underlying ideas of GKA and BE. A group of members interact via open networks to negotiate a public encryption key while each member holds a different secret decryption key. Using the public encryption key, anyone can encrypt any message to any subset of the group members and only the intended receivers can decrypt. Unlike GKA, ConBE allows the sender to exclude some members from reading the ciphertexts. Compared to BE, ConBE does not need a fully trusted third party to set up the system. We formalize collusion resistance by defining an attacker who can fully control all the members outside the intended receivers but cannot extract useful information from the ciphertext.

Second, we present the notion of aggregately broadcast encryption (AggBE). Coarsely speaking, a BE scheme is aggregately if its secure instances can be aggregated into a new secure instance of the BE scheme. Specifically, only the aggregated decryption keys of the same user are valid decryption keys corresponding to the aggregated public keys of the underlying BE instances. We observe that the aggregatability of AggBE schemes is necessary in the construction of our ConBE scheme and the BE schemes in the literature are not aggregatable. We construct a concrete AggBE scheme tightly proven to be fully collusion-resistant under the decision BDHE assumption. The proposed AggBE scheme offers efficient encryption/decryption and short ciphertexts. Finally, we construct an efficient ConBE scheme with our AggBE scheme as a building block. The ConBE construction is proven to be semi-adaptively secure under the decision BDHE assumption in the standard model.

Only one round is required to establish the public group encryption key and set up the ConBE system. After the system set-up, the storage cost of both the sender and the group members is O(n), where n is the number of group members participating in the setup stage. However, the online complexity (which dominates the practicality of a ConBE scheme) is very low. We also illustrate a trade-off between the set-up complexity and the online performance. After a trade-off, the variant has O(n2=3) complexity in communication, computation and storage. This is comparable to up-to-date regular BE schemes which have O(n1=2) complexity in the same performance metrics, but our scheme does not require a trusted key dealer. We conduct a series of experiments and the experimental results validate the practicality of our scheme.

**B. Potential Applications**

A potential application of our ConBE is to secure data exchanged among friends via social networks. Since the Prism scandal , people are increasingly concerned about the protection of their personal data shared with their friends over social networks. Our ConBE can provide a feasible solution to this problem. Indeed, Phan et al. underlined the applications of our ConBE to social networks. In this scenario, if a group of users want to share their data without letting the social network operator know it, they can use our ConBE scheme. Since the setup procedure of our ConBE only requires one round of communication, each member of the group just needs to broadcast one message to other intended members in a send-and-leave way, without the synchronization requirement. After receiving the messages from the other members, all the members share the encryption key that allows any user to selectively share his/her data to any subgroup of the members. Furthermore, it also allows sensitive data to be shared among different groups. Other applications may include instant messaging among family members, secure scientific research tasks jointly conducted by scientists from different places, and disaster rescue using a mobile ad hoc network

A common feature of these scenarios is that a group of users would like to exchange sensitive data but a fully trusted third party is unavailable. Our ConBE provides an efficient solution to these applications.

## II. EXISTING SYSTEM

Group key agreement (GKA) is another well-understood cryptographic primitive to secure group-oriented communications. A conventional GKA allows a group of members to establish a common secret key via open networks. However, whenever a sender wants to send a message to a group, he must first join the group and run a GKA protocol to share a secret key with the intended members.

More recently, and to overcome this limitation, Wu et al. introduced asymmetric GKA, in which only a common group public key is negotiated and each group member holds a different decryption key.

However, neither conventional symmetric GKA nor the newly introduced asymmetric GKA allow the sender to unilaterally exclude any particular member from reading the plaintext. Hence, it is essential to find more flexible cryptographic primitives allowing dynamic broadcasts without a fully trusted dealer.

**DISADVANTAGES OF EXISTING SYSTEM:**

Need a fully trusted third party to set up the system.

Existing GKA protocols cannot handle sender/member changes efficiently.

## III. PROPOSED SYSTEM

Wepresent the Contributory Broadcast Encryption (ConBE) primitive, which is a hybrid of GKA and BE.

This full paper provides complete security proofs, illustrates the necessity of the aggregatability of the underlying BE building block and shows the practicality of our ConBE scheme with experiments.

First, we model the ConBE primitive and formalize its security definitions. ConBE incorporates the underlying ideas of GKA and BE. A group of members interact via open networks to negotiate a public encryption key while each member holds a different secret decryption key. Using the public encryption key, anyone can encrypt any message to any subset of the group members and only the intended receivers can decrypt.

We formalize collusion resistance by defining an attacker who can fully control all the members outside the intended receivers but cannot extract useful information from the ciphertext.

Second, we present the notion of aggregatable broadcast encryption (AggBE). Coarsely speaking, a BE scheme is aggregatable if its secure instances can be aggregated into a new secure instance of the BE scheme. Specifically, only the aggregated decryption keys of the same user are valid decryption keys corresponding to the aggregated public keys of the underlying BE instances.

Finally, we construct an efficient ConBE scheme with our AggBE scheme as a building block. The ConBE construction is proven to be semi-adaptively secure under the decision BDHE assumption in the standard model.

**ADVANTAGES OF PROPOSED SYSTEM:**

1. We construct a concrete AggBE scheme tightly proven to be fully collusion-resistant under the decision BDHE assumption.
2. The proposed AggBE scheme offers efficient encryption/decryption and short ciphertexts.
3. Only one round is required to establish the public group encryption key and set up the ConBE system.

## IV. SYSTEM ARCHITECTURE:

**IMPLEMENTATION**

**MODULES:**

1. **Data owner**

In this module, the data owner should register by providing user name, password, email and group, after registering owner has to Login by using valid user name and password. The Data owner browses and uploads their data to the cloud server. For the security purpose the data provider encrypts the data file and then stores in the web server.

2. **Group Authority**

The group authority is responsible for registering and login authorization for the end users if they are in the same group and also 1. View Group Users 2. View Group Signs 3. View Registered User.

3. **Storage Server**

The Storage server is responsible for data storage and file authorization for an end user. The data file will be stored in cloud server with their tags such as Owner, file name, secret key, mac and private key, can also view the registered Owners and End-users in the cloud server. The data file will be sending based on the privileges. If the privilege is correct then the data will be sent to the corresponding user and also will check the file name, end user name and secret key. If all are true then it will send to the corresponding user or he will be captured as attacker.

4. **Data Consumer(End User)**

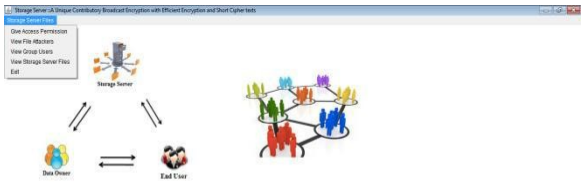
The data consumer is nothing but the end user who will request and gets file contents response from the corresponding cloud servers. If the file name and secret key, access permission like Search and download is correct then the end is getting the file response from the cloud or else he will be considered as an attacker and also he will be blocked in corresponding cloud. If he wants to access the file after blocking he wants to UN block from the cloud.

5. **Attacker**

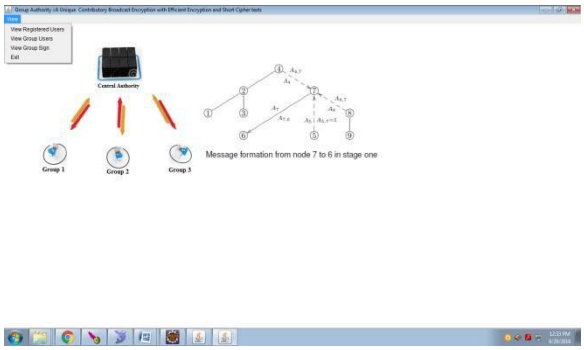
Threat model is one who is trying to receive files by giving fake Skey to the file in the Storage Server. The attacker may be within a Network or from outside the network. If attacker is from inside the network then those attackers are called as internal attackers. If the attacker is from outside the network then those attackers are called as external attackers.

**SCREENSHOT**

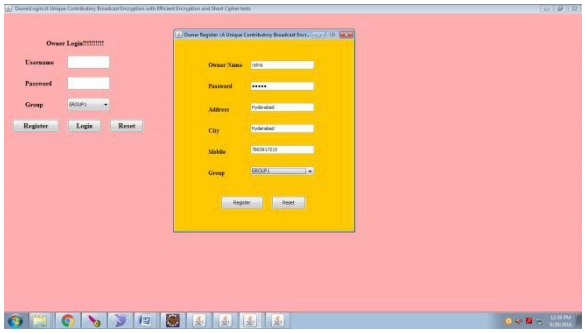
**Storage Server**



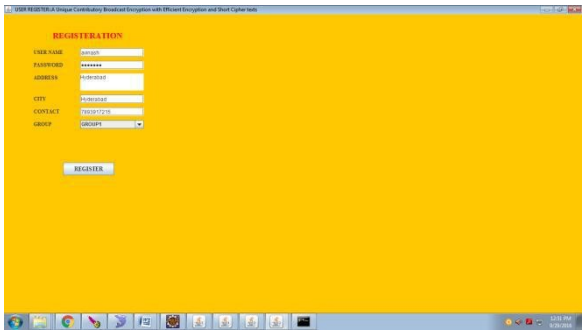
### Group Authority



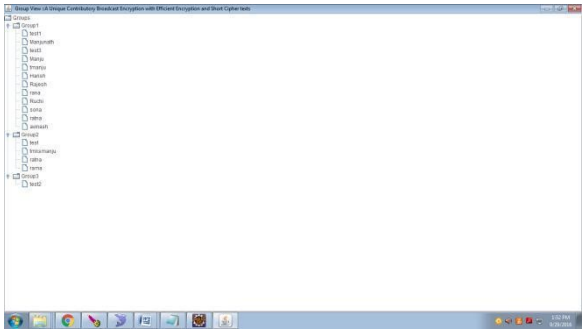

### Owner Registration



### End User Registration




### Group View



## V. CONCLUSION

In this paper, we formalized the ConBE primitive. In ConBE, anyone can send secret messages to any subset of the group members, and the system does not require a trusted key server. Neither the change of the sender nor the dynamic choice of the intended receiversrequire extra rounds to negotiate group encryption/decryption keys. Following the ConBE model, we instantiated an efficient ConBE scheme that is secure in the standard model. As a versatile cryptographic primitive, our novel ConBE notion opens a new avenue to establish secure broadcast channels and can be expected to secure numerous emerging distributed computation applications

## VI. REFERENCES

[1]. A. Fiat and M. Naor, "Broadcast Encryption," in Proc. Crypto 1993, 1993, vol. LNCS 773, Lecture Notes in Computer Science, pp. 480-491.

[2]. I. Ingemarsson, D.T. Tang and C.K. Wong, "A Conference Key Distribution System," IEEETransactions on Information Theory, vol. 28, no. 5, pp. 714-720, 1982.

[3]. Q. Wu, Y. Mu, W. Susilo, B. Qin and J. Domingo-Ferrer, "Asymmetric Group Key Agreement," in Proc.Eurocrypt 2009, 2009, vol. LNCS 5479, Lecture Notes in Computer Science, pp. 15370

[4]. Q. Wu, B. Qin, L. Zhang, J. Domingo-Ferrer and O. Farr`as, "Bridging Broadcast Encryption and Group Key Agreement," in Proc. Asiacrypt 2011, 2011, vol.LNCS 7073, Lecture Notes in Computer Science, pp. 143-160.

[5]. D. H. Phan, D. Pointcheval and M. Strefler, "Decentralized Dynamic Broadcast Encryption," in Proc. SCN 2012, 2011, vol. LNCS 7485, Lecture Notes in Computer Science, pp. 166- 183

[6]. A. Sherman and D. McGrew, "Key Establishment in Large Dynamic Groups Using One-way FunctionTrees," IEEE Transactions on Software Engineering, vol. 29, no. 5, pp. 444-458, 2003.

[7]. Y. Kim, A. Perrig and G. Tsudik, "Tree-Based Group Key Agreement," ACM Transactions on Information System Security, vol. 7, no. 1, pp. 60-96, 2004.

[8]. Y. Mao, Y. Sun, M. Wu and K.J.R. Liu, "JET:Dynamic Join-Exit- Tree Amortization and Scheduling for Contributory Key Management," IEEE/ACMTransactions on Networking, vol. 14, no. 5, pp. 1128-1140, 2006.

[9]. Boyd and J.M. Gonz'alez-Nieto,"Round-Optimal Contributory Conference Key Agreement,"

inProc. PKC 2003, 2003, vol. LNCS 2567, LectureNotes in Computer Science, pp. 161-174.
[10]. W.-G. Tzeng and Z.-J. Tzeng, "Round Efficient Conference Key Agreement Protocols with ProvableSecurity," in Proc. Asiacrypt 2000, 2000, vol. LNCS1976, Lecture Notes in Computer Science.
[11]. R. Dutta and R. Barua, "Provably Secure Constant Round Contributory Group Key Agreement in Dynamic Setting," IEEE Transactions on Information Theory, vol. 54, no. 5, 2007-2025, 2008.
[12]. W.-G. Tzeng, "A Secure Fault- Tolerant Conference- KeyAgreementProtocol,"IEEE Transactions on Computers, vol. 51, no.4, pp. 373-379, 2002.
[13]. X. Yi, "Identity-Based Fault-Tolerant Conferenc Key Agreement," IEEE Transactions Dependable Secure Computing vol. 1, no. 3, 170- 178, 2004.