ISSN (Online): 2349-7084
GLOBAL IMPACT FACTOR 0.238
ISRA JIF 0.351

INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 1, ISSUE 6, DECEMBER 2014, PP 428-435

Secure Data Sharing for Dynamic Groups in the Public Cloud

[1] S.L.SOWJANYA, [2] D.RAVIKIRAN

[1] M.Tech Research Scholar, Priyadarshini Institute of Technology and Science for Women
[2] Professor, Priyadarshini Institute of Technology and Science for Women

Abstract:-Cloud computing is an emerging computing paradigm in which resources of the computing infrastructure are provided as services over the Internet. Sharing group resource among cloud users is a major problem, so cloud computing provides an economical and efficient solution. Mona, secure data sharing in a multi-owner manner for dynamic groups preserves data, identity privacy from an unfrosted cloud and allows frequent change of the membership. In this project, we propose a secure multi owner data sharing scheme, for dynamic groups in the cloud. By leveraging group signature and dynamic broadcast encryption techniques, any cloud user can anonymously share data with others. Proposing a new model for Sharing Secure Data in the Cloud for the Dynamic Group.

Key Words: Cloud computing, data sharing, privacy-preserving, access control, dynamic groups.

1. INTRODUCTION

In cloud computing, the cloud service providers (CSPs), such as Amazon, are able to deliver various services to cloud users with the help of powerful data centers. By migrating the local data management systems into cloud servers, users can enjoy high-quality services and save significant investments on their local infrastructures. Cloud computing is one of the greatest platform which provides storage of data in very lower cost and available for all time over the internet Cloud computing is Internet-based computing, whereby shared resources, software and information are provided to computers and devices on demand.

Several trends are opening up the era of Cloud Computing, which is an Internet-based development and use of computer technology. Cloud Computing means more than simply saving on IT implementation costs. One of the most fundamental services offered by cloud providers is data storage. A company allows its staffs in the same group or department to store and share files in the cloud. By utilizing the cloud, the staffs can be completely released from the troublesome local data storage and maintenance. However, it also poses a significant risk to the confidentiality of those stored files. Cloud offers enormous opportunity for new innovation, and even disruption of entire industries. Cloud computing is the long dreamed vision of computing as a utility, where data owners can remotely store their data in the cloud to enjoy on demand high-quality applications and services from a shared pool of configurable computing resources. Identity privacy is one of the most significant obstacles for the wide deployment of cloud computing. Without the guarantee of identity privacy, users may be unwilling to join in cloud computing systems because their real identities could be easily disclosed to cloud providers and attackers. For example, a misbehaved staff can deceive others in the company by sharing false files without being traceable. Maintaining the integrity of data plays a vital role in the establishment of trust between data subject and service provider. Although envisioned as a promising service platform for the Internet, the new data storage paradigm in "Cloud" brings about many challenging design issues which have profound influence on the security and performance of the overall system. One of the biggest concerns with cloud data storage is that of data integrity verification at untrusted servers. What is more serious is that for saving money and storage space the service provider might neglect to keep or deliberately delete rarely accessed data files which belong to an ordinary client. CS2 provides security against the cloud provider, clients are still able not only to efficiently access their data through a search interface but also to add and delete files securely.

When preparing data to store in the cloud, the data processor begins by indexing it and encrypting it with a symmetric encryption scheme (e.g., AES) under a unique key refer to single writer/single reader (SWSR). It then encrypts the index using a searchable encryption scheme and encrypts the unique key with an attribute-based encryption scheme under an appropriate policy. Finally, it encodes the encrypted data and index in such a way that the data verifier can later verify their integrity using a proof of storage. Asymmetric searchable encryption (ASE) schemes where the party searching over the data is different from the party that generates and refer to many writer/single reader (MWSR).It is very inefficient. Attribute-based encryption scheme each user in the system is provided with a decryption key that has a set of attributes associated with it.

The main Objective of providing two levels of security is a unique and an esoteric study of implementation of an extremely secured system, employing 2 levels of security.

Level 1: Level 1 security provides a simple text based Password. Level 2: After the successful entry of the above level, the Level 2 Security System will then generate a one-time numeric password that would be valid just for that login session. The authentic user will be informed of this one time password on his e-mail.

2. RELATED WORK

E. Goh, H. Shacham, N. Modadugu, and D. Boneh [4] the use of Sirius is compelling in situations where users have no control over the file server (such as Yahoo! Briefcase or the P2P file storage provided by Farsite). They believe that SiRiUS is the most that can be done to secure an existing network file system without changing the file server or file system protocol. Key management and revocation is simple with minimal out-of-band communication. File system freshness guarantees are supported by SiRiUS using hash tree constructions. SiRiUS contains a novel method of performing file random access in a cryptographic file system without the use of a block server. Extensions to SiRiUS include large scale group sharing using the NNL key revocation construction. B. Wang, B. Li, and H. Li, [5] in this paper, we propose Knox, a privacy-preserving auditing scheme for shared data with large groups in the cloud. They utilize group signatures to compute verification information on shared data, so that the TPA is able to audit the correctness of shared data, but cannot reveal the identity o f the signer on each block. With the group manager's private key, the original user can efficiently add new users to the group and disclose the identities of signers on all blocks. The efficiency of Knox is not affected by the number of users in the group.

The data centers hardware and software is what we will call a cloud. When a cloud is made available in a pay-as-you-go manner to the general public, they call it a public cloud; the service being sold is utility computing. They use the term private cloud to refer to internal data centers of a business or other organization, not made available to the general public, when they are large enough to benefit from the advantages of cloud computing that we discuss here. Thus, cloud computing is the sum of SaaS and utility computing, but does not include small or medium-sized data centers, even if these rely on virtualization for management. People can be users or providers of SaaS, or users or providers of utility computing. They focus on SaaS providers (cloud users) cloud providers, which have received less attention than SaaS users.

In this paper consider the problem of building a secure cloud storage service on top of a public cloud infrastructure where the service provider is not completely trusted by the customer. They describe, at a high level, several architectures that combine recent and non-standard cryptographic primitives in order to achieve our goal. Survey the benefits such architecture would provide to both customers and service providers and give an overview of recent advances in cryptography motivated specifically by cloud storage.

They introduce new theoretical measures for the qualitative and quantitative assessment of encryption schemes designed for broadcast transmissions. The goal is to allow a central broadcast site to broadcast secure transmissions to an arbitrary set of recipients while minimizing key management related transmissions. They present several schemes that allow centers to broadcast a secret to any subset of privileged users out of a universe of size so that coalitions of users not in the privileged set cannot learn the secret.

They develop a new cryptosystem for One-grained sharing of encrypted data that call Key-Policy Attribute-

Based Encryption (KP-ABE). In cryptosystem, cipher texts are labelled with sets of attributes and private keys are associated with access structures that control which cipher texts a user is able to decrypt. They demonstrate the applicability of our construction to sharing of audit-log information and broadcast encryption. Our construction supports delegation of private keys which subsumes Hierarchical Identity-Based Encryption (HIBE). The data owner uses a random key to encrypt a file, where the random key is further encrypted with a set of attributes using KP-ABE. Then, the group manager assigns an access structure and the corresponding secret key to authorized users, such that a user can only decrypt a ciphertext if and only if the data file attributes satisfy the access structure. To achieve user revocation, the manager delegates tasks of data file reencryption and user secret key update to cloud servers. However, the single owner manner may hinder the implementation of applications with the scenario, where any member in a group should be allowed to store and share data files with others.

3. PRELIMINARIES

3.1 Group Signature

Chaum and van Heyst first introduced the concept of group signatures. In general, a group signature scheme allows any member of the group to sign messages while keeping the identity secret from verifiers. The variant of the short group signature scheme [1] will be used to achieve anonymous access control, as it supports efficient member-ship revocation.

In this described short signatures in the scheme are approximately the size of a standard RSA signature with the same security. Security of the group signature is based on the Strong Diffie-Hellman assumption and a new assumption in bilinear groups called the Decision Linear assumption.

To recover the message from an encryption, the user computes. By a natural extension of the proof of security of ElGamal, LE is semantically secure against a chosen-plaintext attack.

A number of revocation mechan isms for group signatures have been described. All these mechanisms can be applied to the system. The Revocation Authority (RA) publishes a Revocation List (RL) containing the private keys of all revoked users. Consequently the Revocation List can be derived directly from the private keys of revoked users. The list RL is given to all signers and verifiers in the system. It is used to update the group public key used to verify signatures.

The given RL, anyone can compute this new public key, and any unrevoked user can update her private key locally so that it is well formed with respect to this new public key. Revoked users are unable to do so.

3.2 Dynamic Broadcast Encryption

Broadcast encryption [5] enables a broadcaster to transmit encrypted data to a set of users so that only a privileged subset of users can decrypt the data. A. Fiat [5] described a broadcaster encrypts messages and transmits these to a group of users who are listening to a broadcast channel and use their private keys to decrypt transmissions.

Cecile described dynamic broadcast encryption scheme involves two authorities: a group manager and a broadcaster. The group manager grants new members access to the group by providing to each new member a public label lab and a decryption key dk. The generation of (lab, dk) is performed using a secret manager key. The broadcaster encrypts messages and transmits these to the whole group of users through the broadcast channel.

In a public-key broadcast encryption scheme, the broadcaster does not hold any private information and encryption is performed with the help of a public group encryption key ek containing. When the broadcaster encrypts a message, some group members can be revoked temporarily from decrypting the broadcast content thanks to a one-time revocation mechanism. The KEM-

DEM methodology, broadcast encryption is viewed as the combination of a specific key encapsulation mechanism (a Broadcast-KEM) with a symmetric encryption (DEM) that remains implicit. It leaves as an open problem to realize dynamic public-key broadcast encryption with an encryption key substantially. Finally, expect our trapdoor mechanism to find other cryptographic applications in the future.

4. SYSTEM MODEL AND DESIGN GOALS

4.1 System Model

We consider a cloud computing architecture by combining with an example that an organisation uses a cloud to enable its employees in the same group or department to share files. The system model consists of three different entities: the cloud server, a group manager, and a large number of group members (i.e., the employees) as illustrated in Fig. 1

Cloud server is operated by cloud service providers and the fundamental service provides by them as storage as a service (SaaS). However, the cloud is not fully trusted by the group members. We assume that the cloud server is honest and trust them.

So that cloud server will not maliciously delete or modify user data, by achieving data auditing schemes.

Group manager is responsible for system parameters generation, registering the user, revocating the group member and revealing the real identity incase of any dispute occur. In the

given example, the group manager is acted by the administrator of the organisation and group manager is fully trusted by the other parties.

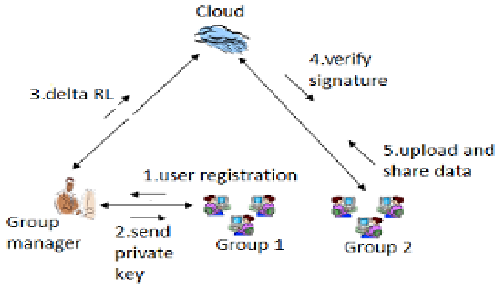


Fig.1 System model

Group members are the registered users they will stockpile their private data into the cloud server and share the data among the group members. In our example, the employee plays the role of group members. It allows the group members to be dynamically changed, due to the staff resignation and the participation of new employee in the organization.

4.2 Design Goals

Access control: Cloud Server allows only the authorized group member to store their private data in the cloud offered by cloud service providers as SaaS and it won't allow unauthorized group member to store their data in the cloud.

Data confidentiality: Data owner will store their data in the cloud and share the data among the group members. Who upload the data have rights to modify and delete their data in the cloud.

Traceability: In case of any dispute occurs it can easily traceable. If other group member delete the other group members data can be easily noticeable.

5. PROPOSED SCHEME

To achieve the reliable and scalable in MONA, in this paper we are presenting the new framework for MONA. In this method we are further presenting how we are managing the risks like failure of group manager by increasing the number of backup group manager, hanging of group manager in case number of requests more by sharing the workload in multiple group managers. This method claims required efficiency, scalability and most importantly reliability.

Advantage

To overcome the disadvantage of existing system MONA, in the proposed MONA is if the group manager stop working due to large number of requests coming from different groups of owners, then backup group manager will remains available. Here user get extra time for accessing data after the time out by sending request to the cloud.

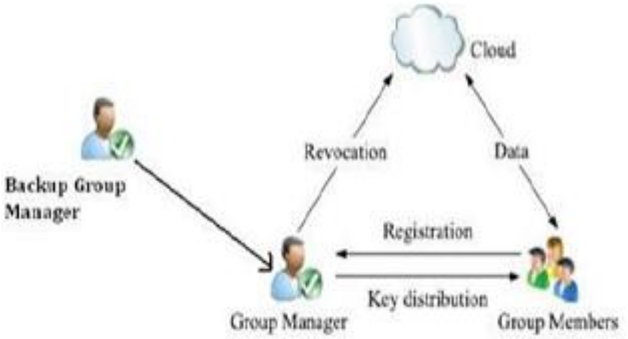


Fig 2 Proposed System Model

Scheme Description

This section describes system, initialization, user registration, user revocation, file generation, file deletion and file access.

System Initialization

The group manager takes charge of system initialization as follows: Generating a bilinear map group system S=(q, G1, G2,e(.,.)). The system parameters including (S, P, H, H0 ,H1 ,H2, U, V , W , Y , Z, f, f1, Enc()), where f is a one-way hash function: {0,1}* —> Z*q ; f1 is hash function: {0,1}* —> G1; and Enck() is a secure symmetric encryption algorithm with secret key k.

User Registration

For the registration of user i with identity IDi, the group manager randomly selects a number xi belong to Z*q and computes Ai, Bi as the following equation:

$$\begin{cases} A_i = \dfrac{1}{\gamma + x_i} \cdot P \in G_1 \\ B_i = \dfrac{x_i}{\gamma + x_i} \cdot G \in G_1. \end{cases}$$

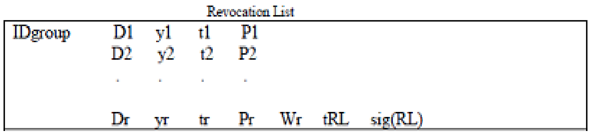
Then, the group manager adds (Ai, xi, IDi) into the group user list, which will be used in the traceability phase. After the registration, user i obtains a private key (xi, Ai, Bi), which will be used for group signature generation and file decryption.

Revocation List

User revocation is performed by the group manager via a public available revocation list (RL), based on which group members can encrypt their data files and ensure the confidentiality against the revoked users. The

list is characterized by time stamp t1,t2,…tr. In the proposed system once the user time stamp over does not wait for the group manager to update the time stamp or revocation list here once the time over the user immediately send request for extra time for access the data to the cloud. Then the cloud will send that request to the group manager once the see it and give permission then the cloud will time to access the data but if the group manager did not give permission then the cloud will not give permission for access of the data.

Table1



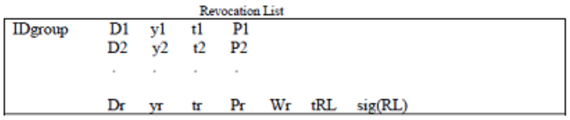
Revocation List

IDgroup	D1	y1	t1	P1			
	D2	y2	t2	P2			
	.	.	.	.			
	Dr	yr	tr	Pr	Wr	tRL	sig(RL)

File Generation

To store and share a data file in the cloud, a group member performs the following operations: Getting the revocation list from the cloud. In this step, the member sends the group identity IDgroup as a request to the cloud. Then, the cloud responds the revocation list RL to the member. Verifying the validity of the received revocation list. First, checking whether the marked date is fresh. Second, verifying the contained signature sig(RL) by the equation e(W, f1 (RL)) = e(P, sig(RL)). If the revocation list is invalid, the data owner stops this scheme. Encrypting the data file M. Selecting a random number T and computing fT. The hash value will be used for data file deletion operation. In addition, the data owner adds (IDdata, T) into his local storage. Constructing the uploaded data file as shown in Table 2, where tdata denotes the current time onthe member, and a group signature on (IDdata, C1, C2, C, f(T); tdata) computed by the data owner through private key (A, x).

Table 2: Message Format



Revocation List

IDgroup	D1	y1	t1	P1			
	D2	y2	t2	P2			
	.	.	.	.			
	Dr	yr	tr	Pr	Wr	tRL	sig(RL)

Uploading the data shown in Table 2 into the cloud server and adding the ID data into the local shared data list maintained by the manager. On receiving the data, the clouds first check its validity. If the algorithm returns true, the group signature is valid; otherwise, the cloud abandons the data. In addition, if several users have been revoked by the group manager, the cloud also performs revocation verification. Finally, the data file will be stored in the cloud after successful group signature and revocation verifications.

File Deletion

File stored in the cloud can be deleted by either the group manager or the data owner (i.e., the member who uploaded the file into the server). To delete a file ID data, the group manager computes a signature and sends the signature along with ID data to the cloud.

6. PERFORMANCE EVALUATION

In this section, we first analyze the storage cost of Mona, and then perform experiments to test its computation cost.

Storage

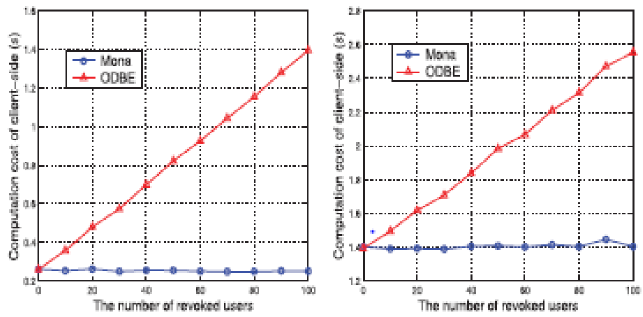
Without loss of generality, we set q=160 and the elements in G1 and G2 to be 161 and 1,024 bit, respectively. In addition, we assume the size of the data identity is 16 bits, which yield

a group capacity of data files. Similarly, the size of user and group identity are also set as 16 bits.

Group manager. In Mona, the master private key of the group manager Additionally, the user list and the shared data list should be stored at the group manager. Considering an actual system with 200 users and assuming that each user share 50 files in average, the total storage of the group manager is (80.125+42.125*200+2*10,000)* Kbytes, which is very acceptable.

Group members. Essentially, each user in our scheme only needs to store its private key (Ai, Bi, xi) which is about 60 bytes. It is worth noting that there is a tradeoff between the storage and the computation overhead. For example, the four pairing operations including (e(H, W), e(H, P), e(P, P), e(Ai, P)) can be precomputed once and stored for the group signature generation and verification. Therefore, the total storage of each users is about 572 bytes.

The extra storage overhead in the cloud. In Mona, the format of files stored in the cloud is shown in Table 2. Since C3 is the ciphertext of the file under the symmetrical encryption, the extra storage overhead to store the file is about 248 bytes, which includes (IDGroup, IDData, C1, C2, C3, f(T), tdata, σ).



(a) Generating a 10 MB file (b) Generating a 100 MB file

Fig.3.1. Comparison on computation cost for file generation between Mona and ODBE.

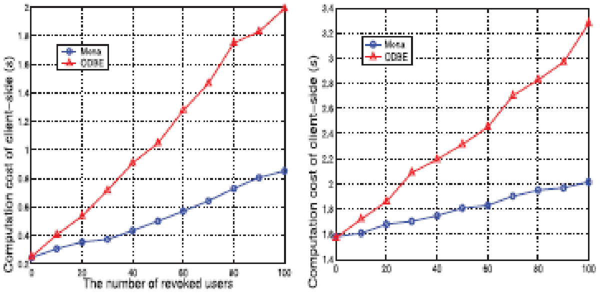
Simulation

The simulation consists of three components: client side, manager side as well as cloud side. Both client-side and manager-side processes are conducted on a laptop with Core 2 T7250 2.0 GHz, DDR2 800 2G, Ubuntu 10.04 X86. The cloud-side process is implemented on a machine that equipped with Core 2 i3-2350 2.3 GHz, DDR3 1066 2G,Ubuntu 12.04 X64. In the simulation, we choose an elliptic curve with 160-bit group order, which provides a competitive security level with 1,024-bit RSA.

Client Computation Cost

In Fig. 6.1, we list the comparison on computation cost of clients for data generation operations between Mona and the way that directly using the original dynamic broadcast encryption. It is easily observed that the computation cost in Mona is irrelevant to the number of revoked users. On the contrary, the computation cost increases with the number of revoked users in ODBE. The reason is that the parameters (Pr, Zr) can be obtained from the revocation list without sacrificing the security in Mona, while several time-consuming operations including point multiplications in G1 and exponentiations in G2 have to be performed by clients to compute the parameters in ODBE. From Figs. 5.1a and 5.1b, we can find out that sharing a 10 Mbyte file and a 100-Mbyte one, cost a client about 0.2 and 1.4 seconds in our scheme, respectively, which implies that the symmetrical encryption operation domains the computation cost when the file is large. The computation cost of clients for file access operation with the size of 10 and 100 Mbytes are illustrated in Fig. 5.2. The computation cost in Mona increases with the number of revoked users,Besides the above operations, P1, P2, …, Pr need to be computed by clients in ODBE.

Therefore, Mona is still superior than ODBE in terms of computation cost. Similar to the data generation operation, the total computation cost is mainly determined by the symmetrical decryption operation if the accessed file is large, which can be verified from Figs. 5.2a and 5.2b. In addition, the file deletion for clients is about 0.075 seconds, because it only costs a group signature on a message (IDdata, T) where T is a 160-bit number in Z*q.



(a) Accessing a 10 MB file (b) Accessing a 100 MB file

Fig. 3.2. Comparison on computation cost for file access between Mona and ODBE.

7. CONCLUSION

In this paper, securely share the data file among the dynamic groups. Without revealing their identity members in the same group can share the data efficiently. Elliptic curve cryptography is used for over all security. When compared to other algorithm key size is very small, it is not able to hack easily. Delta RL is used for efficient revocation without updating private keys of remaining users. In future, concentrate on key management, how to revoke the private keys from the group members. Extensive analyses show that our proposed scheme satisfies the desired security requirements and guarantees efficiency as well. Here we also show that how user gets extra time even after the time out this also one of the advantage of proposed schema.

REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A.Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M.Zaharia, "A View of Cloud Computing," Comm. ACM, vol. 53,no. 4, pp. 50-58, Apr. 2010.

[2] S. Kamara and K. Lauter, "Cryptographic Cloud Storage," Proc.Int'l Conf. Financial Cryptography and Data Security (FC), pp. 136-149, Jan. 2010.

[3] S. Yu, C. Wang, K. Ren, and W. Lou,Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing,"Proc. IEEE INFOCOM, pp. 534-542, 2010.

[4] M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu,"Plutus: Scalable Secure File Sharing on Untrusted Storage," Proc. USENIX Conf. File and Storage Technologies, pp. 29-42, 2003.

[5] E. Goh, H. Shacham, N. Modadugu, and D. Boneh, "Sirius:Securing Remote Untrusted Storage," Proc. Network and Distributed
Systems Security Symp. (NDSS), pp. 131-145, 2003.

[6] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved Proxy Re-Encryption Schemes with Applications to Secure Distributed Storage," Proc. Network and Distributed Systems Security Symp. (NDSS), pp. 29-43, 2005.

[7] R. Lu, X. Lin, X. Liang, and X. Shen, "Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing," Proc. ACM Symp. Information, Computer and Comm. Security, pp. 282-292, 2010.

[8] B. Waters, "Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization," Proc. Int'l Conf. Practice and Theory in Public Key Cryptography Conf. Public Key Cryptography, http://eprint.iacr.org/2008/290.pdf, 2008.

[9] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data," Proc. ACM Conf. Computer and Comm. Security (CCS),pp. 89-98, 2006.

[10] D. Naor, M. Naor, and J.B. Lotspiech, "Revocation and Tracing Schemes for Stateless Receivers," Proc. Ann. Int'l Cryptology Conf.Advances in Cryptology (CRYPTO), pp. 41-62, 2001.

[11] D. Boneh and M. Franklin, "Identity-Based Encryption from the Weil Pairing," Proc. Int'l Cryptology Conf. Advances in Cryptology
(CRYPTO), pp. 213-229, 2001.

[12] D. Boneh, X. Boyen, and H. Shacham, "Short Group Signature," Proc. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 41-55, 2004.

[13] D. Boneh, X. Boyen, and E. Goh, "Hierarchical Identity Based Encryption with Constant Size Ciphertext," Proc. Ann. Int'l Conf.Theory and Applications of Cryptographic Techniques (EUROCRYPT), pp. 440-456, 2005.

[14] C. Delerablee, P. Paillier, and D. Pointcheval, "Fully Collusion Secure Dynamic Broadcast Encryption with Constant-Size Ciphertexts or Decryption Keys," Proc. First Int'l Conf. Pairing-Based Cryptography, pp. 39-59, 2007.

[15] D. Chaum and E. van Heyst, "Group Signatures," Proc. Int'l Conf.Theory and Applications of

Cryptographic Techniques (EUROCRYPT),pp. 257-265, 1991.

[16] A. Fiat and M. Naor, "Broadcast Encryption," Proc. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 480-491, 1993.

[17] B. Wang, B. Li, and H. Li, "Knox: Privacy-Preserving Auditing for Shared Data with Large Groups in the Cloud," Proc. 10th Int'l Conf. Applied Cryptography and Network Security, pp. 507-525, 2012.