Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 5, May 2015, pg.662 – 669

RESEARCH ARTICLE

Multi –User Secure Data Sharing for Dynamic Groups in the Cloud Mona

Sharon.S.Anakal[1], Dr. Shubhangi.D.C[2]

[1]Computer science and Engineering & VTU, India

[2]Computer science and Engineering & VTU, India

[1] anakal.sharon@gmail.com; [2] shubhangidc@yahoo.co.in

Abstract— Cloud computing is an emerging computing paradigm in which resources of the computing are provided as services over the Internet and this also come out with many new challenges for data security and access control. Cloud computing is the developing technology, where data owners can remotely store and modify their data on the premise of pay-as-use manner and enjoy on demand high-quality applications. The fundamental service provided by the cloud is data Storage. Data sharing among the cloud user within a group is efficient. While sharing data in a multi-owner manner, preserving data and identity privacy from an untrusted cloud is a challenging issue, because of the frequent change of the membership, sharing data in multi-owner manner become a very difficult task. Therefore we propose secure multi-owner data sharing for dynamic groups by combining group signature and dynamic broadcast encryption techniques. At the same time we compute the number of revoked user and efficiency.

Keywords— cloud computing, privacy preserving, data sharing, dynamic groups

I. INTRODUCTION

Cloud Computing [1], the long-held vision of computing as a service, is likely to transform a large part of the IT industry, making software yet more attractive as to tune-up and shape the way IT hardware is designed and purchased, along with low maintenance feature. The cloud service providers (csp's) such as Microsoft's Azure storage service and Amazon's S3 provide consumers with scalable and dynamic storage. The primary service rendered by the cloud service providers is data storage, which provides user with scalable resources in the pay-as you-use approach at comparatively low prices. For example, Amazon's S3 information storage service simply charges $0.12 to $0.15 per gigabyte-month. As compared to building their own infrastructures and maintaining local data storage is again a problem, users are able to save their investments extensively by migrating businesses into the cloud. As the worker of the organization stores perceptive, confidential data, business plans and there may be information outflow in the untrusted cloud. A basic solution provided by existing system to ensure data privacy and security is encrypting the data files, prior uploading into the cloud server. But regrettably designing a secure and efficient cloud data sharing scheme for dynamic groups in the cloud is not simple job because of the some issues.

One is Identity privacy, cloud computing user's are unenthusiastic to join cloud computing service because, real identities of user are easily reviled by the intruders.

Second is No Multiple-owner Manner, single-owner manner is not flexible as multiple owner. Because multiple owner manner allow every member in the group should be able to alter their own part of data rather than reading too.

Third is Effect of Dynamic Groups: The joining of new staff and revocation of current employee makes the group dynamic in nature. It leads to a difficulty, a new granted user are not allowed to learn the content of data files stored before their participation by the anonymous system, because it impossible for new granted users to directly contact with unidentified data owners and get the corresponding decryption keys. To lessen the complication of key management it is desirable to obtain an efficient membership revocation mechanism without updating the secret keys of the remaining users. To solve the challenges recited above, we tend to propose mona, a secure multi-owner information sharing theme for dynamic groups within the cloud. The most contributions of this paper include: We tend to propose a secure multi-owner data sharing scheme. It implies that any user within the cluster can securely share data with others by the untrusted cloud. Untrusted cloud is, wherever there aren't any security measures taken their access control mechanisms are insecure and easily defeated. We provide secure and privacy-preserving access control to users, which guarantees any user in a group to anonymously utilize the cloud store. Additionally, the real identity of data owners can be revealed by the group admin when any kind of disputes occur. It also support dynamic groups efficiently and new granted users can directly decrypt data files uploaded before their involvement without contacting with data owners. User revocation is simply achieved through a unique revocation list while not changing the secret keys of the remaining users.

II. RELATED WORK

M. Kallahalla et al. [2] proposed cryptographic storage system which is known as Plutus. It enables secure file sharing without placing much trust on the file server or untrusted server. The main characteristic of Plutus is that all information is stored encrypted and all the key distribution is handled in a decentralized manner. It divide files into file groups and enable data owner to share the file groups with others by encrypting each file cluster with distinctive file-block key that can protect data. There are some restriction known, a significant key distribution overhead for large-scale file sharing, the file block key has to be updated and distributed again for a user revocation and provides end-to-end security for group sharing system with lazy revocation.

Scalable and fine-grained data access control scheme by defining access polices based on data attributes and KP-ABE [3].This technique allows that data owner can send user secret key updates and data files to cloud servers while not revealing data contents or user access privilege information. This may be done using, key policy attribute-based encryption (KP-ABE) and uniquely combine it with the technique of proxy re-encryption (PRE) and lazy re-encryption. Data files are encrypted using random key by data owner. Using KP-ABE the random key is further encrypted with a set of attributes. Then the authorized users are given an access structure and corresponding secret key by the group manager. This system has some limitation such as multiple-owner manner is not supported by this system so that those single owner manners make it less flexible.

SiRiUS proposed by E. Goh et al [4], Securing Remote Untrusted Storage**.** SiRiUS assumes the network storage is untrusted and provides its own read-write cryptanalytic access control for file level sharing. Key management and revocation is easy and minimal with out-of–bond communication. File system novelty guarantees are supported by SiRiUS using hash tree constructions. SiRiUS contains a new technique of performing file random access in a cryptographic file system without the use of a block server. SiRiUS includes large scale group sharing using the NNL key revocation construction. It provides secure NFS without changing the file server. SiRiUS has some limitation in case of user revocation and dynamic groups. The user revocation is complex for large scale sharing. Private Key of every group member should be updated while joining of new user in the group.

Ateniese et al. [5] proposed proxy re-encryptions in which a semi-trusted proxy converts a ciphertext into a plaintext without seeing the plaintext. Blocks of content are encrypted with unique and symmetric content keys by the data owner. The resulting encrypted content keys are further encrypted under a master public key. The implementation of proxy re-encryption used in a secure file system uses a centralized access control server to manage access to encrypted content stored on distributed, untrusted replicas. The advantage of these schemes is that they are unidirectional and only a limited amount of trust is placed in the proxy. However, a collusion attack can occur between any revoked malicious user and untrusted server allowing them to find out the decryption Keys of all the encrypted blocks of content.

Secure provenance scheme which records ownerships and process history of data object. This scheme is based on the bilinear coupling techniques and it's characterized by providing the data confidentiality on sensitive documents hold on cloud, anonymous authentication on user access, and provenance tracking on unclear documents. These schemes depend upon group signatures and cipher text-policy attributes based encryption (CP-ABE) techniques [6]. Mainly, the method consists of individual attribute. After the registration, every user in this method obtains two keys: a group signature key and an attribute key. Using attribute-base encryption (ABE) any user will encipher a data file. For decryption of the encrypted data, an attribute keys is employed by others in the group. To achieve privacy conserving and traceability options, the client signs encrypted information with group signature key. Unfortunately, the disadvantage of this system is that user revocation isn't supported.

To overcome these problems, we design secure data sharing scheme for dynamic groups in an untrusted cloud by combining group signature and broadcast encryption techniques. In this model we can securely share the data among multiple group

members and user revocation is achieved. By comparing the existing work, our model offers the unique features as: Any group member able to store and share data files with others within a group. This system support dynamic group efficiently. It implies that new user joining and user revocation are easily achieved without involving remaining users. This system provides rigorous security using AES encryption technique.

III. METHODOLOGY

3.1 System Architecture

We contemplate a cloud computing design by combining with an example that an organization uses a cloud to allow its staffs within the same cluster or department to share files as illustrated in below figure1,

The system architecture consists of three entities:

- Group manager (admin).
- Group member.
- Cloud server.

Group manager: Group manager takes charge of system parameters generation, user registration, user revocation, and revealing the real identity of a staff when dispute information owner. Within the given example, the group manager is acted by the administrator of the corporate. Therefore, we have a tendency to assume that the group manager is fully trusted by the opposite parties and maintain revocation list and migrate this list into cloud for public use, and traceability.

Group Members: Group members are a collection of registered users that may store their private information into the cloud server and share them with others within the cluster.

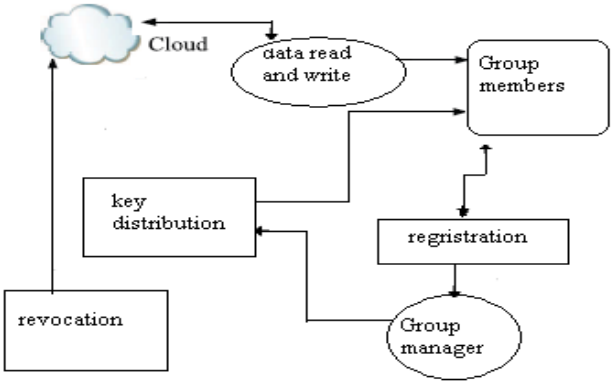


Fig: 1.system architecture

Cloud Server: Cloud is that the massive repository of resources. Cloud is liable for storing all users data and granting access to the file within a group to other group members based on publically accessible revocation list which is supported by group manager. We assume that the cloud server is honest but curious. That is, the cloud server won't maliciously delete or modify user data, however can try to learn the content of the stored data

. The preludes used in the scheme are:

Bilinear Maps: Let G1 and G2 be an additive cyclic group and a multiplicative cyclic group of the same prime order q, respectively [7]. Let e: G1 ×G1 → G2 denote a bilinear map constructed with the following properties:

1. Bilinear: We say that a map ê: G1 x G1 →G2 is bilinear if ê (xS, yT) = ê(S;T) xy for all S;T €G1 and all x;y € Z.
2. Non-degenerate: The map does not send all pairs in G1× G1 to the individuality in G2. Observe that since G1, G2 are groups of prime order this implies that if S is a generator of G1 then ê(S; S) is a generator of G2.
3. Computable: There is an efficient algorithm to compute ê (S; T) for any S;T € G1.

Group Signature

Group signatures, introduced by Chaum and van Heyst [8], provide anonymity for signers. Each group member has a private key that enables him to sign messages three properties that a group signature scheme must satisfy: • Correctness, which ensures that honestly-generated signatures verify and trace correctly;

• Full-anonymity, which ensures that signatures do not reveal their signer's identity; and

• Full-traceability, which ensures that all signatures, even those created by the collusion of multiple users and the group manager, trace to a member of the forging coalition.

Dynamic Broadcast Encryption

In dynamic broadcast encryption [9], [10]group manager dynamically include the new user without altering the antecedently computed information along with that user decryption keys need not to be changed or modified.

3.2 Proposed Scheme

In this scheme we are using the combination of dynamic broadcast encryption and group signature technique. The group signature allow user to use cloud resources and the dynamic broadcast encryption technique allows data owners to firmly share their data files with others including new user and revocation list is also presented in this scheme.

3.2.1 Scheme Description

This section describes the details of Mona including system initialization, user registration, user revocation, file generation, access controlling, and traceability.

System initialization: The admin is responsible for initialization of system as follows: Generating a bilinear map group system A=(t,G1,G2,e(‘,’)).Selecting two random numbers P,P0 € G1 along with two random numbers $1$2 €G1.Randomly choosing two elements S.G€G1and a number γ €z* and computing F.=γ.S,Y= γ.G.The system parameters including (A, S, P, P0 ,P1 ,P2, D, E, F, H, K, v, v1,Enc()), where v is a one-way hash function: {0,1}* —> Z*q ; v1 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 ui a number belong to Z*q and computes Wi, Xi as the following equation:

$$Wi = \frac{1}{\gamma + ui} S \in G1$$

$$Xi = \frac{1}{\gamma + u} G \in G1$$

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

User Revocation: User revocation is performed by the administrator or 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 b1,b2,…br. Let IDgroup denote the group identity. The tuple (Wi; ui; bi) represents that user i with the partial private key (Wi; ui) is revoked at time bi as shown in table 1. A1; A2;:::; Ar and Zr are calculated by the group manager with the private secret key as follws:

$$A_1 = \frac{1}{\gamma + u_1} . S \in G1.$$

$$A_2 = \frac{1}{(\gamma + u_1)(\gamma + u_2)} . S \in G1$$

$$A_r = \frac{1}{(\gamma + u_1)(\gamma + u_2) \ldots . (\gamma + u_r)} . S \in G1$$

$$Z_r = \frac{1}{Z(\gamma + u_1)(\gamma + u_2) \ldots . (\gamma + u_r) \in} G2$$

Motivated by the verifiable reply mechanism [11],to guarantee that users get the most recent version of the revocation list, we tend to let the group manger update the revocation list everyday even no user has being revoked in the day. In additionally, the others will verify the freshness of the revocation list from the contained current date tRL. In addition, the revocation list is enclosed by a signature sig(RL) to declare its validity. The signature is generated by the group owner i.e sig (RL)=γf1(RL). Finally, the group manager transmigrates the revocation list into the cloud for public usage.additionally; the others will verify the freshness of the revocation list from the contained current date tRL. In addition, the revocation list is enclosed by a signature sig(RL) to declare its validity. The signature is generated by the group owner i.e sig (RL)=γf1(RL). Finally, the group manager transmigrates the revocation list into the cloud for public usage.

TABLE 1

REVOCATION LIST

IDgroup	W_1	u_1	b_1	A_1			
	W_2	u_2	b_2	A_2			
	.	.	.	.			
	W r	u r	b r	A r	Z r	b RL	Sig(RL)

File Generation: To store and share a information file in the cloud, a group owner performs the subsequent operations: Obtaining the revocation list from the cloud. During 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 or not the marked data is recent. Second, validating the contained signature sig (RL) by the equation e (F, f1 (RL)) = e (P, sig (RL)). If the revocation list is invalid, the data owner stops this method. In additionally, the data owner adds (IDdata, T) into his local storage. Constructing the uploaded data file as shown in Table2, where tdata denotes the current time on the member, and a group signature on (IDdata, C1, C2, C, f(T); tdata) computed by the data owner through private key (W, u),upload the information within the cloud as shown in table 2. Algorithm2 is employed to used to verify the validity and algorithm 3 is employed to perform user revocation.

TABLE 2
MESSAGE FORMAT FOR UPLOADING DATA

Group ID	Data ID	Ciphertext	hash	time	Signature
IDgroup	IDdata	C_1,C_2,C	f(T)	tdata	σ

Algorithm (1): signature verification
Input: private key (Q, y), system parameters (S,D,E,P,F)and data K
Output: generate a valid group signature on K
begin
select random numbers α,β,rα,rβ,rx,rδ1,rδ2 ϵ Zq*
set $\delta 1 = y\alpha$ and $\delta 2 = y\beta$
compute as follow

$$\begin{cases} X_1 = \alpha \,.\, D \\ X_2 = \beta \,.\, E \\ X_3 = Qi + (\alpha + \beta).P \\ M_1 = r\alpha \,.\, D \\ M_2 = r\beta \,.\, E \\ M_3 = e(X_3,S)^{r_x}\, e(p,F)^{-r_\alpha - r_\beta}\, e(P,S)^{-r_{\delta_1} - r_{\delta_2}} \\ M_4 = r_x \,.\, X_1 - r_{\delta_1} \,.\, D \\ M_5 = r_x \,.\, X_2 - r_{\delta_1} \,.\, E \end{cases}$$

Set c =f (K, X_1,X_2,X_3,M_1,M_2,M_3,M_4,M_5)
Construct the following numbers

$$\begin{cases} s_\alpha = r_\alpha + c\alpha \\ s_\beta = r_\beta + c\beta \\ s_x = r_x + cx \\ s_{\delta_1} = r_{\delta_1} + c\delta_1 \\ s_{\delta_2} = r_{\delta_2} + c\delta_2 \end{cases}$$

Return σ=(X_1,X_2,X_3,c,sα,sβ,sx,sδ_1,sδ_2)
end

Algorithm (2) Signature verification
Input: system parameters (S, D, E, P, F), K and signature
σ=(X_1,X_2,X_3,c,sα,sβ,sx,sδ_1,sδ_2)

$$\sigma = (X_1, X_2, X_3, c, s_\alpha, s_\beta,$$

Output: valid or invalid
begin
Compute the following values

$$\begin{cases} \tilde{N_1} = s_\alpha . U - c.X_1 \\ \tilde{N_2} = s_\beta . V - c.X_2, \\ \tilde{N_3} = \left(\frac{e(X_3, F)}{e(S, S)}\right)^c e(X_3, S)^{s_x} e(P, F)^{-s_\alpha - s_\beta}, \\ \quad e(P, S)^{-s_{\delta_1}, -s_{\delta_2}} \\ \tilde{N_4} = s_x . X_1 - s_{\delta_1} . U \\ \tilde{N_5} = s_x . X_2, - s_{\delta^2} . V \end{cases}$$

if c= f(K, X_1,X_2,X_3,$\check{N}_1$,$\check{N}_2$,$\check{N}_3$,$\check{N}_4$,$\check{N}_5$)
Return valid
Else
Return invalid

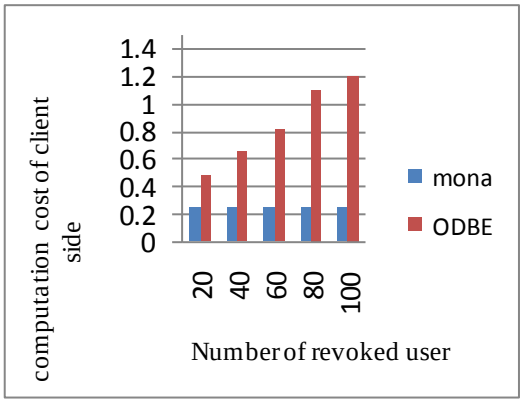
Algorithm (3). Revocation verification
Input: system parameters (P_o,P_1,P_2), a group signature σ, and a set of revocation keys W_1…….Wr
Output: true or false
begin
set temp = e(X_1,P_1)e(X_2,P_2)
for i = 1 to n
if e(X_3_Wi,P_o) = temp
return true
end if
end for
return false
end

File Deletion: File stored in the cloud are often deleted by either the group manager or the data owner to delete a file IDdata, the group manager computes a signature and sends the signature beside IDdata to the cloud.

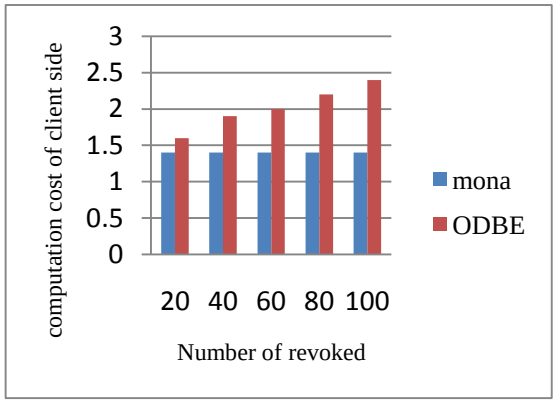
File Access : To be taught the content of a shared file, obtaining the data file and also the revocation list from the cloud server. During this operation, the user first adopts its private key(W, u) to compute a signature σu on the message (IDgroup, IDdata, t, σu) by exploitation Algorithm 1, where t denote the current time, and the IDdata are often obtained from the native shared file list maintained by the manager, then, the user sends a data request containing (IDgroup, IDdata,t) to the cloud server. Upon receiving the request, the cloud server employs Algorithm 2 to envision the validity of the signature. After a successful verification, the cloud server responds the corresponding data file and also the revocation list to the user.

IV. RESULT AND DISCUSSION

The result is evaluated in three ways: user side, administrator side and cloud side. The computation cost between these three sides is compared with file generation and file access using mona and ODBE (original dynamic broadcast encryption)[10]. Computation cost of mona remains idle while file generation when compared with ODBE which varies as shown in fig 2. The reason is that the parameters (Ar; 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. 2a and 2b, we can find out that sharing a 10-Mbyte file and a 100-Mbyte.



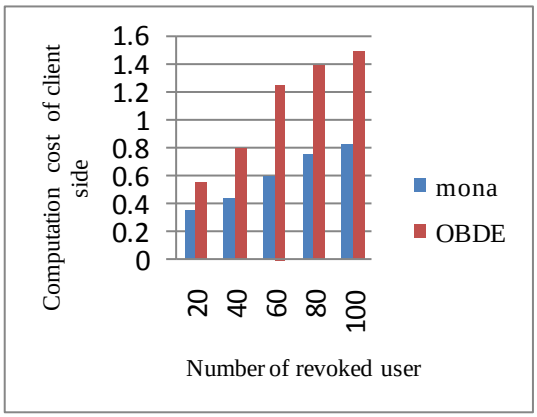
(a) Generating 10MB file



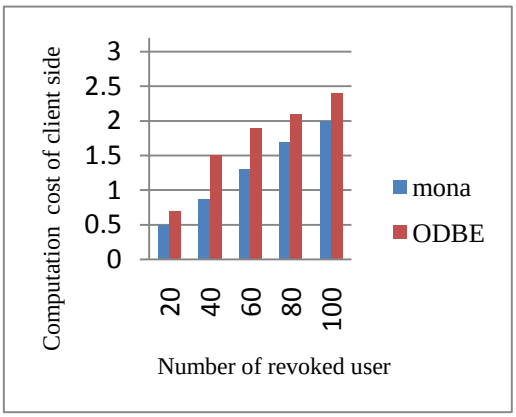
(b) Generating 100MB file

Fig 2: comparing computation cost for file generation

The computation cost during the file access of the size 10 and 100MB is shown in the below fig3. The cost of mona increases as with the number of revoked users, where as in ODBE the (A1; A2;:::; Ar) to be calculated if the accessed file is large is checked in figs.2a and 2b.



(a) Generating 10MB file



(b) Generating 100MB file

Fig 3: comparing computation cost for file access.

V. CONCLUSIONS

We designed a secure data sharing in untrusted cloud using dynamic encryption and group signature additionally we have also implemented a provision for user revocation and new user joining. User revocation is achieved by publically displaying the updated list and storage overhead and computation cost are constant. In future we can provide an option where the authorized users IP address can be used to recognize the real identity by the administrator if there is any misuse of data.

ACKNOWLEDGEMENT

I would like to thank my guide Dr.Shubhangi.D.C for assisting me in this paper work.

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] 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.

[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] 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.

[5] 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.
[6] 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.
[7] 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.
[8] D. Chaum and E. van Heyst, "Group Signatures," Proc. Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT), pp. 257-265, 1991.
[9] A. Fiat and M. Naor, "Broadcast Encryption," Proc. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 480-491, 1993.
[10] C. Delerablee, P. Paillier, and D. Pointcheval, "Fully Collusion Secure Dynamic Broadcast Encryption with Constant-Size Cipher texts or Decryption Keys," Proc. First Int'l Conf. Pairing-Based Cryptography, pp. 39-59, 2007.
[11] B. Sheng and Q. Li, "Verifiable Privacy-Preserving Range Query in Two-Tiered Sensor Networks," Proc. IEEE INFOCOM, pp. 46- 50, 2008.
[12] 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 crptography,http://eprint.iacr.org/2008/290.pdf, 2008.
[13] 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.
[14] Yong CHENG, Jun MA and Zhi-ying "Efficient revocation in ciphertext-policy attribute-based encryption based cryptographic cloud storage" Zhejiang University and Springer-Verlag Berlin 2013.
[15] D. Boneh, X. Boyen, and H. Shacham, "Short Group Signature," Proc. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 41-55, 2004.
[16] 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.
[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.
[18] D. Boneh, B. Lynn, and H. Shacham, "Short Signature from the Weil Pairing," Proc. Int'l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology, pp. 514-532, 2001.
[19] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing," Proc. IEEE INFOCOM, pp. 525-533, 2010.
[20] D. Pointcheval and J. Stern, "Security Arguments for Digital Signatures and Blind Signatures," J. Cryptology, vol. 13, no. 3, pp. 361-396, 2000.
[21] The GNU Multiple Precision Arithmetic Library (GMP), http:// gmplib.org/, 2013.

