2581-4575

International Journal For Recent Developments in Science & Technology

Crossref

A Peer Reviewed Research Journal

ENHANCED PRIVACY PROTECTION IN CLOUD SERVICES

[#1]SK.ALLABEE, [#2]V.GURU KUMAR

[1]M.TECH STUDENT, DEPARTMENT OF EEE, KAKINADA INSTITUTE OF TECHNOLOGICAL SCIENCES (KITS), RAMACHANDRAPURAM

[2]ASSISTANT PROFESSOR, DEPARTMENT OF EEE, KAKINADA INSTITUTE OF TECHNOLOGICAL SCIENCES (KITS), RAMACHANDRAPURAM.

Abstract: With the rapid development of computer technology, cloud-based services have become a hot topic. They not only provide users with convenience, but also bring many security issues, such as data sharing and privacy issue. In this paper, we present an access control system with privilege separation based on privacy protection (PS-ACS). In the PS-ACS scheme, we divide users into private domain (PRD) and public domain (PUD) logically. In PRD, to achieve read access permission and write access permission, we adopt the Key-Aggregate Encryption (KAE) and the Improved Attribute-based Signature (IABS) respectively. In PUD, we construct a new multi-authority cipher text policy attribute-based encryption (CP-ABE) scheme with efficient decryption to avoid the issues of single point of failure and complicated key distribution, and design an efficient attribute revocation method for it. The analysis and simulation result show that our scheme is feasible and superior to protect users' privacy in cloud-based services.

Keywords: access control; data sharing; privacy protection; cloud-based services

1. INTRODUCTION

With the rapid development of cloud computing, big data and public cloud services have been widely used. Users can store their data in the cloud service and rely on the cloud service provider to give data access to other users. However, the cloud service provider can no longer be fully trusted. Because it may give data access to some illegal users or attackers for profit gain. For users, it is necessary to take full advantage of cloud storage service, and also to ensure data privacy. Therefore, the study of access control scheme to protect users' privacy in cloud environment is of great significance. Since traditional access control strategy [1] cannot effectively solve the security problems that exist in data sharing, various schemes to achieve encryption and decryption of data sharing have been proposed. In 2007, Bethen court et al. [2] first proposed the cipher text policy attribute-based encryption (CP-ABE). However, this scheme does not consider the revocation of access permissions. Attrapadung et al. [3, 4] came up

with two user-revocable ABE scheme. However, they are not applicable in the outsourcing environment. In 2011, Hur et al. [5] put forward a fine grained revocation scheme, but it can easily cause key escrow issue. Lewko et al. [6] used multi-authority ABE (MA-ABE) to solve key escrow issue. But the access policy is not flexible. Later, Li et al. [7] presented a data sharing scheme based on systemic attribute encryption, which endows different access permissions to different users. However, it lacks of efficiency. Xie et al. [8] presented a revocable CPABE scheme. Compared with Hur's scheme, in the key update phase, the computation load of the data service manager will be reduced by half. Liang et al. [9] proposed a CP-ABE proxy encryption scheme which supports any monotonic access structures. However, their construction which is built in the composite order bilinear group cannot be

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

converted to the prime order bilinear group. In 2014, Chu et al. [10] proposed Key-Aggregate Encryption algorithm, which effectively shortens the length of the cipher text and the key, but only for the situation where the data owner knows user's identity. The above schemes only focus on one aspect of the research, and do not have a strict uniform standard either. In this paper, we present a more systematic, flexible and efficient access control scheme. To this end, we make the following main contributions:

1) We propose a novel access control system called PS-ACS, which is privilege separation based on privacy protection. To achieve read access permission, in PRD, the Key-Aggregate Encryption (KAE) scheme which greatly improves access efficiency is adopted. And in PUD, we construct a new multi-authority ciphertext policy attribute-based encryption (CP-ABE) scheme with efficient decryption to avoid the issues of single point of failure and complicated key distribution, and design an efficient attribute revocation method for it.

2) Compared with the MAH-ABE scheme which does not refer to the write access control, we exploit an Improved Attribute-based Signature (IABS) [11-13] scheme to enforce write access control in PRD. In this way, the user can pass the cloud server's signature verification without disclosing the identity, and successfully modify the file.

3) We provide security and performance analysis of our proposed PS-ACS scheme. The functionality and simulation results provide data security in acceptable performance impact, and prove the feasibility of the scheme.

2. SYSTEM MODEL

As shown in Fig.1, our system model consists of Data owner, users in PSD, and users in PUD, root authority CA, regional authority AA and cloud service provider, which are defined as follows.

1. The cloud service provider consists of two parts: data storage server and data service management. Data storage server is responsible for storing confidential data files, and data service management is in charge of controlling external users' access to secret data and returning the corresponding ciphertext.

2. In the actual cloud environment, CA manages multiple AA, and AA each manages attributes in their own field. The attributes owned by the user are issued by different authority.

3. Personal domain (PSD), in which users have special privileges, such as family, personal assistant, close friends and partners. This domain has a small number of users and small scale attributes, and the data owner knows the user's identity, which is easy to manage.

4. Public domain (PUD), which owns a huge number of users with unknown identity and a lot of attributes owned by the user.

5. Data Owner, based on the characteristics of users in public and personal domain to develop different access control strategy, encrypt uploaded files using the corresponding encryption method and then send to the cloud server.

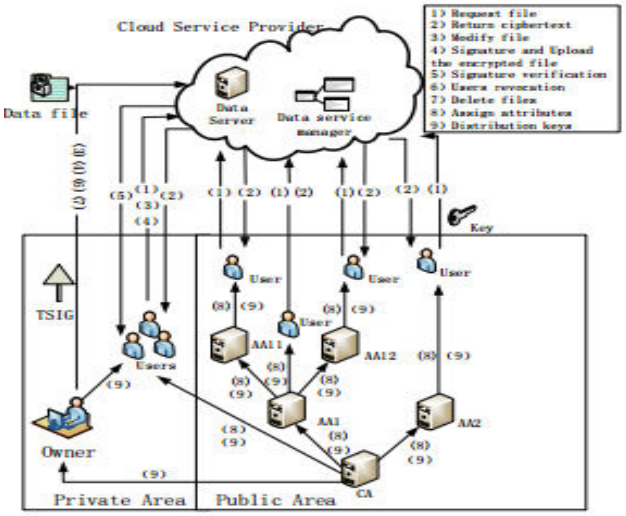

Figure 1. System framework

3. ACCESS CONTROL SCHEME IN PSD

A. Read Access Control

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

The PSD has a small number of users, and their identities are known to the owner. In general, the data owner only wants the users to access or modify parts of data files, and different users can access and modify different parts of the data. For example, the blogger can allow his friend to browse part of his private photos; enterprises can also authorize employees to access or modify part of sensitive data. This requires the data owner to grant users read or write access permission to some data. In Chen's MAH-ABE scheme, the CP-ABE is used to achieve the read access permission, but there are some defects to be considered. Firstly, since in the PSD, the users are all have a close relationship with the owner and the number is small, there is no need to use the CP-ABE which is applicable to the scenario which has a lot of users, and their identities are unknown to the owner, while the KAE scheme is set for the small users with certain identities. Besides, the distribution and management of keys and attributes, encryption and decryption process of CP-ABE are much more complex compared with the KAE scheme. Therefore, the KAE is exploited to implement the read access permission which improves the access efficiency.

Based on the above analysis, the paper uses the Aggregate Key Encryption scheme to encrypt the data files to realize different read access control. The specific application process of the KAE algorithm is as follows.

1) System setup and file encryption. The system first runs Setup of KAE to establish the public system parameter and master key. Each owner classified the file by its data attribute, such as "photo files", "blog files" and "game files". Fig.2 shows the way to classify the files. Choose and label the files, denoted by

$$i(i \in \{1,2,\ldots, n\})$$

note that a file class i cannot be the subset of another file class

$$j(j \in \{1,2,\ldots, n\}).$$

Then the owner's client application runs Encrypt of KAE using the public key and the number of classification file to encrypt the PHR files and sends them to the cloud.

2. Access and key distribution. When the user send access request to the cloud server, and his file index number is i , then the cloud server returns the corresponding encrypted classification file to the user. The owner authorized users access permission with the file index number denoted by j and sent the collection S of all the index number j to CA, CA generate an aggregate decryption key for a set of cipher text classes via Extract of KAE and sent it to the corresponding user, Finally, any user with an aggregate key can decrypt any cipher text whose class is contained in the aggregate key via Decrypt of KAE.

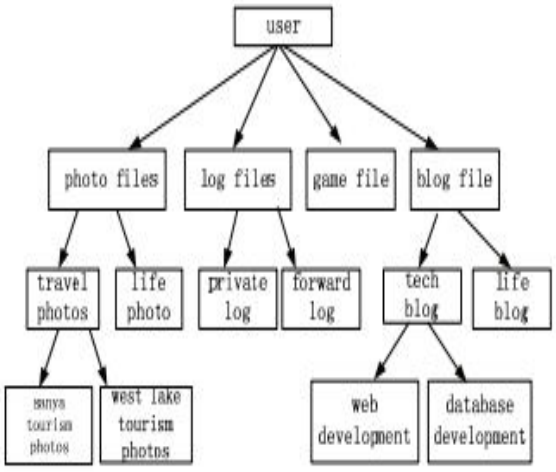

Figure 2. Data file classification

B. Write Access Control

As Chen's MAH-ABE scheme does not refer to the write access control, and in the PSD some cases exist, for example, the owner needs his friends to modify his file after he read it. So we proposed the write access permission in the PSD. For the user, the public key and file class label are all known, he can implement the algorithm to

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

encrypt the files after he modified, and then upload them to the cloud. But whether the cloud server saves the modified file is decided by the write access control policy. On the one hand, in the complex cloud environment, if a user's modification operations are very frequent, maybe he is very important to the user, so that the user may be stricken from outside attacks. Therefore, the user worries the leak of identity after the signature. On the other hand, in the data sharing scheme, the separate access of read and write to the file is extremely important. In PSD, not all users who have read permissions also have write permissions to the files. Whether the user has write permissions to the file is decided by the data owner. Therefore, this paper selects the improved attribute-based signature (IABS) to determine the user's write permission.

The main structure of the scheme includes five parts: an authentication center (CA), the data owner, users, mediator and cloud servers. The CA is responsible for generating master key which is sent to the owner and system parameters which are shared for all users. The mediator holds part components of the signature keys and is responsible for the validity check of attributes and users. The data owner produces the signature tree and sends it directly to the cloud server. The user encrypts the modified files and signs them using the attribute-based signature, then uploads them to the cloud server. The cloud server verifies the attribute-based signature, if the authentication is successful, the user has permission to modify files and the cloud server stores the file. Own to the limited space we will omit the specific description of the IABS scheme in PSD.

4 . ANALYSIS OF OUR SCHEME

In this section, we present the security analysis and performance analysis of our access control system (PS-ACS).

4.1 Security analysis

In PRD, users can only decrypt the files corresponding to the received aggregate keys and do not have access to other files, thus the data owner controls the users' access permissions. When the data file is modified, although CA is trusted, the system parameters and revocation instructions are generated by the CA. The signature policy is formulated by the data owner and is sent directly to the cloud server. The CA does not know the signature policy. Assuming that CA cannot give itself authorization, as long as the attributes of CA cannot satisfy the access policy, it is not valid to modify the file. Thus, the write access permissions still belong to the data owner. In the process of the users' signature, the signature key is only related to the users' attributes, so the user's identity is secure. On the whole, the IABS scheme can protect users' identity privacy.

In PUD, our construction achieves data confidentially. The outsourced data can be confidential against a user whose attributes do not satisfy the access policy. Since the attributes cannot satisfy the access structure in the cipher text, the user cannot receive the partially decrypted cipher text CT' during the transformation process. Thus, he cannot recover the original message. On the other hand, when a user is revoked, the cloud server will delete his transformation keys. Without the transformation keys, he cannot receive CT' either.

In addition, our scheme also achieves forward and backward security. When a user is revoked, his transformation keys will be deleted by the cloud server. Thus, he can no longer receive the partially decrypted ciphertext and cannot recover the original message. On the other hand, when a new user joins to share the outsourced data, the ciphertext will be re-encrypt by the cloud server so that he can also decrypt the cipthertext. Therefore, the forward and backward security of the outsourced data can be guaranteed.

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

We compare our scheme with several existing multi-authority CP-ABE schemes in Table 2 in terms of the access structure type, user domains, the security against collusion attack and the support of revocation.

From Table 1, we can conclude that our scheme achieves high efficiency of privacy protection. Compared with other schemes, we adopt outsourcing decryption method to reduce the overhead in user decryption phase. In our system, the transformation keys are stored on the cloud server, the collusion among users and between users and authorities both need to be considered. In our scheme, each TK contains a unique parameter z which is bind with the user. As each z is different and it is kept secret to other users, different users cannot share their TKs to make collusion attack. Thus, our scheme can prevent users' collusion attack. On the other hand, although the cloud server transforms the ciphertext into an ElGamal-style ciphertext, as the ciphertext is associated with z which he does not know, he still learns nothing about the original message. Therefore, our scheme can prevent the server attack. On the whole, our scheme can achieve privacy protection in PUD.

Schemes	Access Structure	User domains	Against collusion attack	Revocation
Narayan [16]	Attribute policy	PUD	Single authority	User
Ruj [17]	Access tree	PUD	Against N-1 authorities collusion	User
Our scheme	LSSS	PRD and PUD	Against N-1 authorities collusion and user collusion	Attribute and user

Table I Comparison of security

Notation	Description
T_p	Time for one pairing operation
T_m	Time for one exponentiation operation
n	Number of attributes
n^2	The size of matrix
$\|M\|$	The size of the encrypted file
$\|C\|$	The size of C
$\|C'\|$	The size of C'
$2\|C'\|$	The size of (C_i, D_i)

Table 2 Notations for performance analysis

4.2 Performance analysis

We first make a statement for the notations used in the performance analysis, which are listed in Table 2. In our KAE scheme in the PRD, the system parameters are generated by the trusted authority, which is not within our consideration. Moreover, can be calculated in the system setup phase. In addition, the aggregate key only needs

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

one pairing operation, and to calculate a pairing operation is very fast, the specific comparison can be seen in Fig.3.

In Fig.3, the attribute-based encryption algorithm of the MAH-ABE scheme spent much more time than the KAE algorithm used in our scheme. If the attribute revocation occurs, the ABE algorithm will be more time-consuming. More importantly, the growth rate of time spent with the number of file attributes is much higher than KAE algorithm. The simulation results show the high efficiency of our scheme.

In Fig.4, the user only needs a very short time to sign the modified files. While, the authentication time only makes up a small part, so the process of signature and authentication consume a very small time. Therefore, from the client's perspective, the program is efficient.

In PUD, we adopt outsourcing decryption method. We compare our scheme with Ruj's scheme, and the results are shown in Table 3.

Combined with Table 4, the performance analysis is given in the following aspects.

• Since the cloud server uses a set of transformation keys to transform the ABE ciphertext into a constant-size ElGamal-style ciphertext which is much smaller and easier to decrypt than an ABE ciphertext, the user only needs one exponential operation to recover the message. From Table 3, it is obvious that the method can greatly reduce the overhead of users.

We implemented our construction in Charm [18], a framework developed to facilitate the rapid prototyping of cryptographic schemes and protocols. It is based on the Python language which allows the programmer to write code similar to the theoretical implementations. However, the routines that implement the dominant group operations use the PBC library [19] (written natively in C) and the time overhead imposed by the use of Python is usually less than 1%. Charm also provides routines for applying and using LSSS schemes needed for Attribute-Based systems. For more information on Charm we refer the reader to [18, 20]. All our implementations are executed on an Intel® Pentium® CPU G630@270GHz with 4.00GB RAM running Ubuntu14.04 and Python2.7.

We compared the computing time incurred in encryption and decryption. In Fig.5, the number of authorities is set to 10. It is obvious that our scheme requires less time for encryption and decryption than Ruj's scheme, especially for decryption. Since in the decryption phase, major computation overhead is delegated to the • Our scheme supports efficient user and attribute revocation without updating users' private keys. For user revocation, we do not need to re-encrypt the ciphertext and update all non-revoked users' private keys. Instead, we only need to delete the user's transformation keys. Without the transformation key, he can no longer decrypt the ciphertext. On the other hand, when attribute revocation occurs, private keys of all non-revoked users will not be updated, only the transformation keys which are stored in the cloud server and the involved ciphertext need to be updated. Thus, the efficiency of revocation can be greatly improved.

cloud, user only needs one exponentiation operation to recover the original message. Therefore, the decryption time for users can be greatly reduced. Computing cost for transformation is shown in Fig.6. On the whole, it can be concluded that our scheme's computation effi ciency is much better than Ruj's scheme.

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

Scheme	Size of ciphertext stored in the cloud server	Size of ciphertext user received	Decryption time for user
Ruj [17]	n^2+ $n\cdot(\lvert C\rvert+2\lvert C'\rvert)$ $+\lvert C\rvert+\lvert M\rvert$	n^2+ $n\cdot(\lvert C\rvert+2\lvert C'\rvert)$ $+\lvert C\rvert+\lvert M\rvert$	$2n\cdot T_p+n\cdot T_m$
Our scheme	n^2+ $(2n+1)\cdot\lvert C'\rvert$ $+\lvert C\rvert+\lvert M\rvert$	$2\lvert C\rvert+\lvert M\rvert$	T_m

Table 3 Comparison of communication cost and computing time

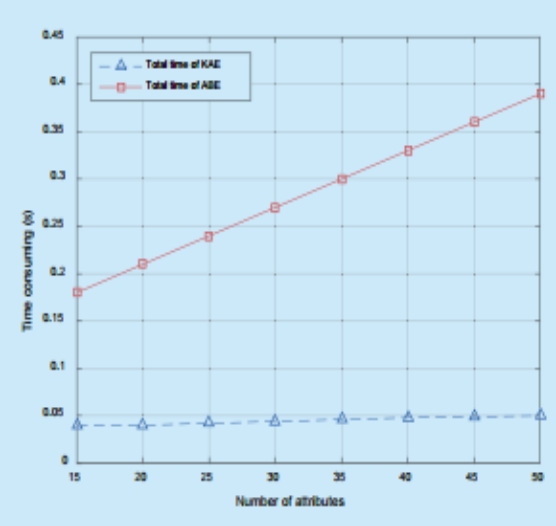

Fig.3 Total time of KAE and ABE

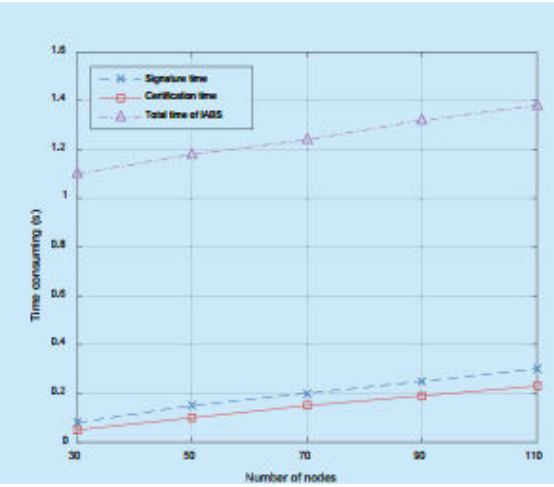

Fig.4 The signature and authentication time of IABS

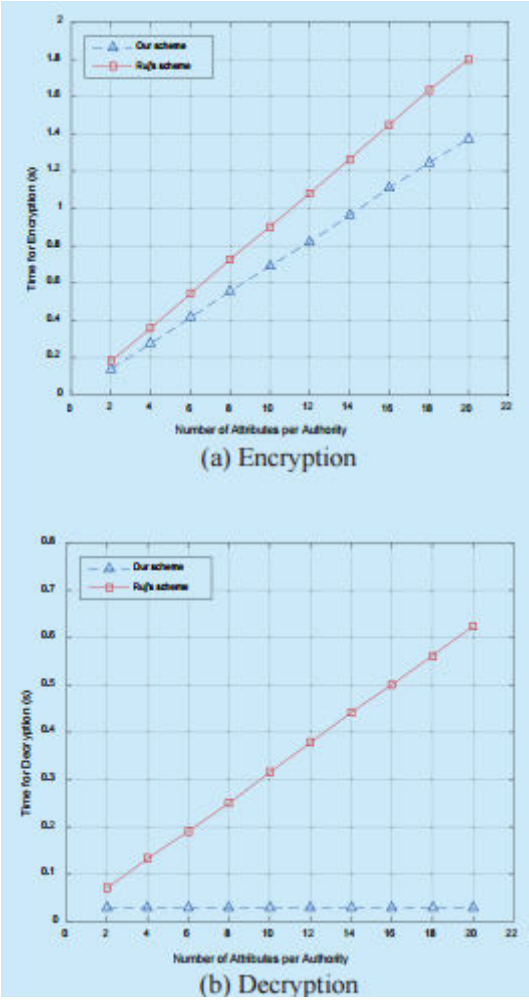

Fig.5 Comparison of Encryption and Decryption Time

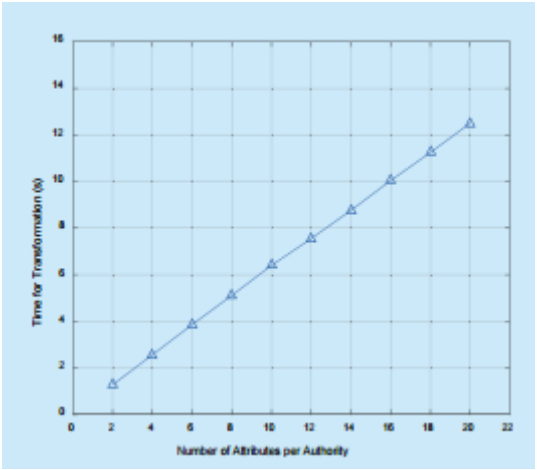

Fig.6 Computing cost for transformation

5. CONCLUSION

In this paper, we proposed an access control system (PS-ACS), which is privilege separation based on privacy protection. Through

International Journal For Recent Developments in Science & Technology
2581-4575
Crossref
A Peer Reviewed Research Journal

the analysis of cloud environment and the characteristics of the user, we divide users into personal domain (PRD) and public domain (PUD) logically. In PRD, we set read and write access permissions for users respectively. To achieve read access permission, the KAE scheme which can improve the access efficiency is adopted. A high degree of patient privacy is guaranteed simultaneously by using IABS scheme which can determine users' write access permission. For users in PUD, we constructed a new multi-authority ciphertext policy attribute-based encryption (CP-ABE) scheme with efficient decryption to avoid the issues of single point of failure and complicated key distribution, and design an efficient attribute revocation method for it. The analysis and the simulation result show that the PSACS scheme is feasible and superior to protect the privacy of data in cloud-based services.

REFERENCES

[1] S. Yu, C. Wang, K. Ren, "Achieving secure, scalable, and fine-grained data access control in cloud computing," Proc. IEEE INFOCOM, pp. 1-9, 2010.

[2] J. Bethencourt, A. Sahai, B. Waters, "Ciphertext-policy attribute-based encryption," Proc. Security and Privacy, pp. 321-334, 2007.

[3] J. Hur, D.K. Noh, "Attribute-based access control with efficient revocation in data outsourcing systems," IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 7 pp. 1214-1221, 2011.

[4] A. Lewko, B. Waters, "Decentralizing attribute-Based encryption," Proc. Advances in Cryptology-EUROCRYPT, pp. 568-588, 2011.

[5] M. Li, S. Yu, Y. Zheng, "Scalable and secure sharing of personal health records in cloud computing using attribute-Based Encryption," IEEE Transactions on Parallel and Distributed System, vol. 24, no. 1, pp. 131- 143, 2013.

[6] C.K. Chu, S.S.M. Chow, W.G. Tzeng, "Key-aggregate cryptosystem for scalable data sharing in cloud storage," IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 2, pp.468-477, 2014.

[7] J. Li, K. Kim, "Hidden attribute-based signatures without anonymity revocation," Information Sciences, vol. 180, no. 9, pp. 1681-1689, 2010.

[8] H.K. Maji, M. Prabhakaran, M. Rosulek, "Attribute-Based Signatures," Proc. Topics in Cryptology - CT-RSA, pp. 376-392, 2011.

[9] S. Kumar, S. Agrawal, S. Balaraman, "Attribute based signatures for bounded multi-level threshold circuits," Proc. Public Key Infrastructures, Services and Applications, pp. 141-154, 2011.