International Journal on Future Revolution in Computer Science & Communication Engineering
Volume: 3 Issue: 9
ISSN: 2454-4248
204 – 209

Multiple Authorities Access under Public Cloud Storage

Bhagyashree D. Masatkar
M. Tech. IV Sem
TGPCET, Nagpur

Prof Rajesh Babu
CSE Department
TGPCET, Nagpur

Prof Roshani Talmale
CSE Department
TGPCET, Nagpur

Abstract **-** Public cloud storage is a cloud storage model that provide services to individuals and organizations to store, edit and manage data. Public cloud storage service is also known as storage service, utility storage and online storage. Cloud storage has many advantages, there is still remain various challenges among which privacy and security of users data have major problem in public cloud storage. Attribute Based Encryption(ABE) is a cryptographic technique which provides data owner direct control over their data in public cloud storage. In the traditional ABE scheme involve only single authority to maintain attribute set which can bring a single-point bottleneck on both security and performance. Now we use threshold multi-authority Cipher Text-Policy Attribute-Based Encryption (CP-ABE) access control scheme, name TMACS. TMACS is Threshold Multi-Authority Access Control System. In TMACS,multiple authority jointly manages the whole attribute set but no one has full control of any specific attribute. By combining threshold secret sharing (t,n) and multi-authority CP-ABE scheme, we developed efficient multi-authority access control system in public cloud storage.

Index Terms *-Access control, Attributes-Based Encryption, data storage, Multi-Authority*

*****

I. INTRODUCTION

Cloud storage is an important service of cloud computing, which provides services for data owners to outsource data to store in cloud via Internet. As cloud storage has many advantages ,there is still remains various challenges among which ,privacy and security of users' data have major issues in public cloud storage. Traditionally, a data owner stores his/her data in trusted servers, which are generally controlled by a fully trusted administrator[3].

Attribute-based Encryption (ABE) is regarded as one of the most suitable schemes to conduct data access control in public clouds for it can guarantee data owners direct control over their data and provide a fine-grained access control service. In most existing CP-ABE[1], [9] schemes there is only one authority responsible for attribute management and key distribution. This only-one-authority scenario can bring a single-point bottleneck on both security and performance.

Now in our scheme we use threshold multi-authorityCP-ABE [13],[14]access control scheme, to deal with the single-point bottleneck problem on security and performance in most existing schemes.. In TMACS, multiple authorities jointly manage the whole attribute set but no one has full control of any specific attribute.

The main contributions of this work can be summarized as fallows

- In existing cloud computing system there is problem of single-point bottleneck on security and performance against the single authority.
- To overcome this problem we are designing a multi-authority access control system.
- For designing multi-authority access control architecture we are introduced the combination of threshold secret sharing and multi-authority CP-ABE scheme.
- In this scheme we proposed a multi-authority access control system in public cloud storage, in which multiple authority jointly manages auniform attribute set.

1.1 Our contribution

In this paper weareintroduces multi-authority data access control for cloud storage system with Attributes-Based Encryption. Threshold multi-authority CP-ABE access control scheme called as TMACS .in this scheme multiple authority is responsible for secret key sharing, no one has full control of any specific attribute. In CP-ABE schemes, there is always a secret key(SK) used to generate attribute private keys, we introduce(t, n) threshold secret sharing into our scheme to share the secret key among authorities.

II. RELATED WORK

Cryptographic techniques are used to access control for cloud storage system. [3]The data owners encrypt files by using the symmetric encryption approach with content keys and then use every user's public key to encrypt the content keys. Attribute-based Encryption (ABE) is a promising technique that is very suitable for access control of encrypted data. In CP-ABE schemes,[1] there is always a secret key(SK) used to generate attribute private keys, we introduce(t, n) threshold secret sharing into our scheme to share the secret key among authorities. In existing access control systems for public cloud

 ISSN: 2454-4248
204 – 209

storage, there brings a single-point bottleneck on both security and performance against the single authority for any specific attribute.[1]By introducing the combining of (t;n) threshold secret sharing and multi-authority CP-ABE scheme we propose multi- authority access control system in public cloud storage, in which multiple authorities jointly manage a uniform attribute set. By combining the traditional multi-authority scheme with ours, we construct a hybrid one, which can satisfy the scenario of attributes coming from different authorities which can solve single point bottleneck problem and provide security.

2.1PRELIMINARY

In this section, we first give a brief review of background information on bilinear maps and the security assumption defined on it, then we briefly describe (t;n) threshold secret sharing introduced in TMACS.

2.1.1 Bilinear Maps

Let G; GT be two multiplicative cyclic groups with the same prime order p and g be a generator of G. A bilinear map e : G×G →GT defined on G has the following three properties:

1) Bilinearity**:** a,bϵZp and g1;g2ϵ G, we have e$(g_1, g_2) \epsilon$ G, we have e$(g_1^a g_2^b) = e(g_1, g_2)^{ab}$.

2) Non-degeneracy: There exists $g_1, g_2 \in G$ such that, $e(g_1, g_2) \neq G$which means the map does not send all pairs in G×G to the identity in G_T.

3) Computability: There is an efficient algorithm to compute$e(g_1, g_2)$for all$g_1, g_2 \in G$.

2.2. Threshold Secret Sharing

Secret sharing is a technique used to share a secret among group of participants. In this group of participant each of them is allocate a some share of secrete key. The secret can be reconstructed only when all share are combine together. There are various scheme are used for secret sharing .In this paper we are used (t,n) threshold secret sharing.

Here we give simple example of(t,n) threshold secret sharing. Assume that there n participants in the system which is denoted as P=$\{P_1, P_2 ....P_1\}$ And we set t $(t \leq n)$as the threshold value. Define a finite field $K = GF(q)$Each participant's identification in public can be denoted as $x_1, x_2 \ldots . x_n \epsilon GF(q), (x_i \neq x_j when i \neq j)$. First, each participant Pi selects a random element Si $\epsilon\, GF(q)$ as his/her sub-secret, then the master secret can be set as S=$\sum_{i=1}^{n} S_i$, which cannot be known by any participant. Each participant Pi separately generates a random polynomial $f_i(x)$ of degree t-1 that satisfies the formula $f_i(0) = S_i$. Subsequently, for each of the other n-1 participants p_j $(j = 1,2, \ldots \ldots i-1, i+1 \ldots, n)$, the participant Pi separately calculates sub-shares: $s_{ij} = f_i(x_j)$, and sends the sub-share $s_{ij}\, to\, P_j$ securely. Meanwhile, Pi generates $s_{ij} = f_i(x_i)$ for himself/herself. After receiving messages from all of the other n-1 participants, each participant Pi obtains n sub-shares$s_{ij} = (j = 1,2,3 \ldots n)$. Thus, Pi can calculate his/her own master share as$s_i = \sum_{j=1}^{n} s_{ij} = \sum_{j=1}^{n} f_i(x_i)$After that the sharing master secret S can be reconstructed by any t of the n participants.

III. SYSTEM MODEL AND SECURITY MODEL

In this section ,we give the definition of the system model and the security model in multi-authority public cloud storage system.

3.1 System Model

In public cloud storage system, there exits five entities, certificate authority(CA), attribute authority(AAs), data owners (Owners),data consumers(User), and the cloud server.

1. **CertificateAuthority :** Certificate Authority is responsible for the construction of the system by setting up system parameters and attribute public key(PK) of each attribute in whole attribute set[1].
2. **Attribute authority** :Attribute authority focuses on the attribute management and key generation. AA jointly manages the whole attribute set , any one of the AA can not assign users secrete key alone for the master key is shared by AA.
3. **Data Owner:** Owner encrypts his/her file and define access about who can get access to his/her data. Owner encrypts his/her data with a symmetric encryption algorithm .Then the owner formulates access policy over an attribute set and encrypts the symmetric key under the policy according to attribute public key gained from CA ..
4. **Data Consumer:**In this module, Users are having authentication and security to access the detail which is presented in the system. Before accessing or searching the details user should have the account in that otherwise they should register first. CA can assign user identity uid and password to data consumer.[1]
5. **Public Cloud Server:**An entity which is managed by cloud **server** provider to provide data storage services . In cloud data storage , a user store his data in cloud server .In cloud data storage s system , user store their data in clouds and no longer possess the data locally. Thus the correctness and availability of the data files being stored on the distributed cloud server must be guaranteed[4].

3.2 Security assumption and Security Model

3.2.1 Security Assumption

In multi-authority public cloud storage system,the security assumptions of the five roles is assumed as fallows.

The cloud server is always online and managed by the cloud provider.CA is assumed to be trusted , but it can also be

 ISSN: 2454-4248
204 – 209

compromised by an adversary, so as to AAs.Although a user can freely get ciphertext from the cloud server, we can't decrypt the encrypted data only unless the user's attributes satisfy the access policy hidden inside the encrypted data. Therefore , some malicious user are assumed to be dishonest and curious , who may collude with other entities excepts data owners to obtain the access permission beyond their privileges. As a comparisons , owners can be fully trusted.

3.2.2 Security model

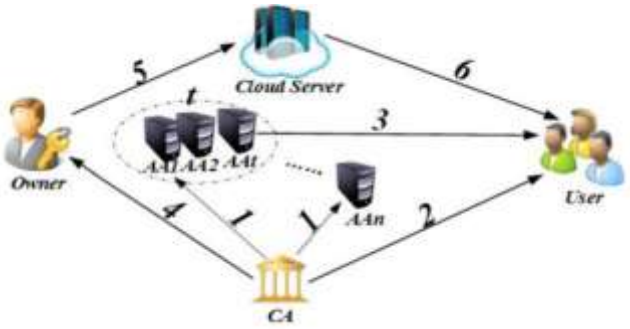
The security model in multi-authority public cloud storage ,which can be divided into two phases. In the first phase, the malicious user compromises AAs to gain AAs' master key. In the second phase, the adversary attempts to decrypt a ciphertext with the secret keys that can't satisfy the access policy inside the ciphertext. In this security model , there is an adversary and a challenger. The adversary can query for any attribute keys as long as they can not be applied directly to decrypt the ciphertext. The ciphertext is provided by the challenger and encrypted under an access structure with attribute public key.

IV. PROPOSED SCHEME FOR MULTI-AUTHORITY SYSTEM

In this section , we first give an overview of multi-authority access control system. In the following detailed we describe secret key generation ,Encryption , and Decryption.

4.1Overview of Our Scheme

To overcome the problem of single-point bottleneck, we introduced (t, n) threshold secret sharing , based on redundant multiple AAs, then propose a threshold multi-authority CP-ABE and the relevant access control scheme multiple access in public cloud storage.



(1) *AA* registers to *CA* to gain $(aid, aid.cert)$;
(2) User registers to *CA* to gain $(uid, uid.cert)$;
(3) User gains his/her *SK* from any t out of n *AAs*;
(4) Owners gain *PK* from *CA*;
(5) Owners upload (*CT*) to the cloud server;
(6) Users download (*CT*) from the cloud server.

5.2 Details of our Data Access Control Scheme

5.2.1 System Initialization

System initialization divided into three phases:CASetup1,AASetup,CASetup2.

1.CASetup1: CASetup1 is responsible for establishment of System parameters and accepting registration of user and AAs. The CASetup1 operation is run by CA.

- AAs Registration: Each AA sends a registration request to CA during the system initialization. For each legal registration CA assigns a unique identity.
- User Registration: Each user sends a registration request to CA during the system initialization .CA authenticates the user ,then assign an identification uid to him/her.

2. AASetup: AA cooperate with each other to share the master key in AAsetup. The operation of AASetup is run by each one of all n AAs. Then all AAs cooperate with each other to call (t, n) threshold secret sharing.and after receiving the share of all AA master keys share,AA calculates its relevant public keys.

3. CASetup2:.The operation CA setup2 is run by CA. Thecorresponding public key is generated by CA in CAsetup2

4.2.3Secret Key Generation

The secret key generation operation is run by one user and any t out of n AAs, user's secret key cannot be generated. In this operation , there is no interaction between any two of t AAs, so the user can select t AAs according to his/her own preference , and then separatelycontact with each of these t AAs to get the secret key share. After getting t secret key share separately from t AAs , the user can generate his/her secret key.

V. PROPOSED METHODOLOGY

5.1 TECHNIQUES / ALGORITHMS

ECC(Elliptic Curve Cryptography)

ECC is an asymmetric cryptography algorithm which involves some high level calculation using mathematical curves to Encrypt and Decrypt data. It is similar to RSA as its asymmetric but it uses a very small length key as compared to RSA.

Key Generation

Key generation is an important part where we have to generate both public key and private key. The sender will be encrypting the message with receiver's public key and the receiver will decrypt its private key.

Now, we have to select a number **'d'** within the range of **'n'**.

Using the following equation we can generate the public key

Q=d* P

d = The random number that we have selected within the range of (**1 to n-1**). **P** is the point on the curve.

'Q' is the public key and **'d' is the private key.**

ISSN: 2454-4248
204 – 209

5.1.1.Encryption

The process of converting information or data into a code, especially to prevent unauthorized access.Encryption is the process of encoding a message or information in such a way that only authorized person can access it. Encryption does not itself prevent interference, but denies the intelligible content to a would-be interceptor. In an encryption scheme, the intended information or message, referred to as plaintext, is encrypted using an encryption algorithm, generating ciphertext that can only be read if decrypted.

Step1: Define curve
Step2: Generate public private key pair using that curve , for both Generate shared secrete key from the key pair.
Step:. From that shared secrete key , generate an encryption key.
Step5: Using that encryption key and symmetric encryption algorithm, encrypt the data to send.

5.1.2.Decryption

Decryption is the process of taking encoded or encrypted text or other data and converting it back into text that you or the computer can read and understand. This term could be used to describe a method of un-encrypting the data manually or with un-encrypting the data using the proper codes or keys.
Step1: Generate public private key pair using the same curve for that curve. For Receiver
Step2: Regenerate a shared secrete key using private key of receiver and public key of sender.
Step3: From that shared secrete key, generate an encryption key.
Step4: Using that encryption key and symmetric encryption algorithm decrypt the data.

VI. IMPLEMENTATION OF PROPOSED TECHNOLOGY

Fig2:CA login page

This is login page for CA.

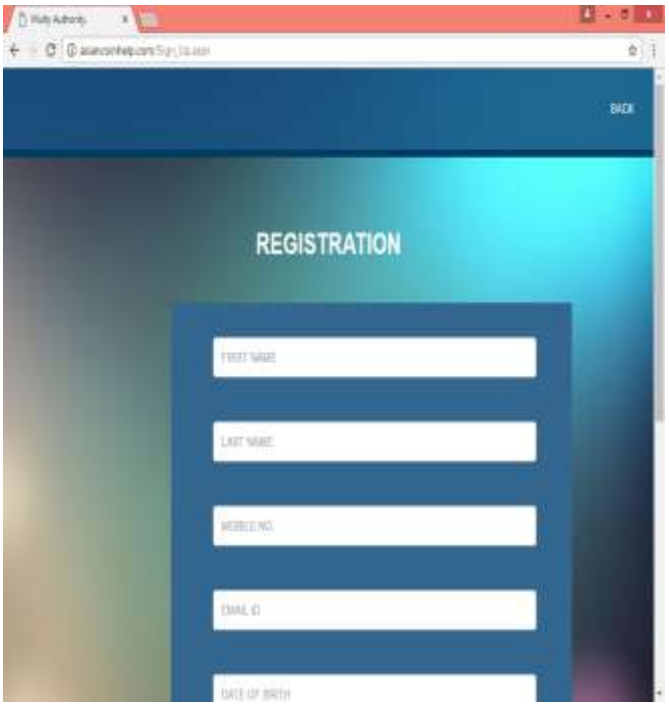


Fig3: Registration

In the registration CA will accepts users ,owners and AAs registration request by assing a unique userid and password for each legal user.

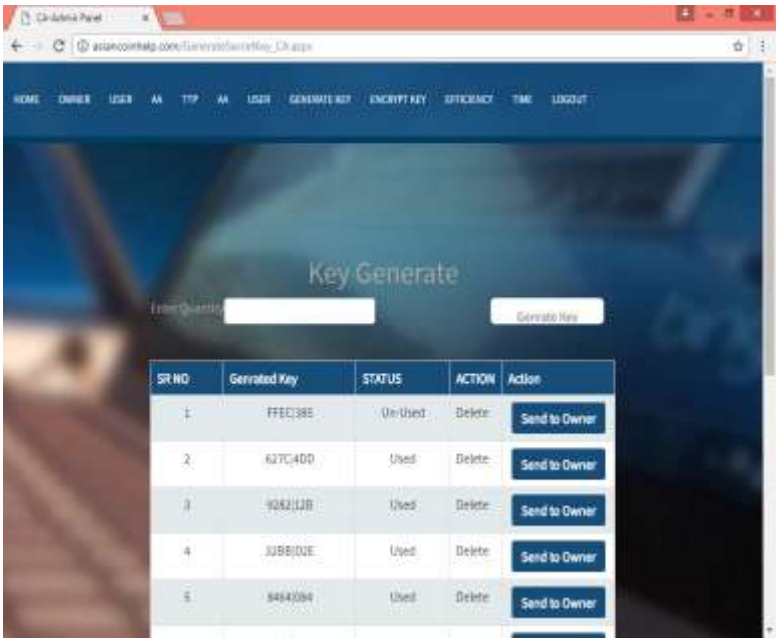


Fig4 :Key generation

In the key generation phase CA can generate the key and sends to Owner.

 ISSN: 2454-4248
204 – 209

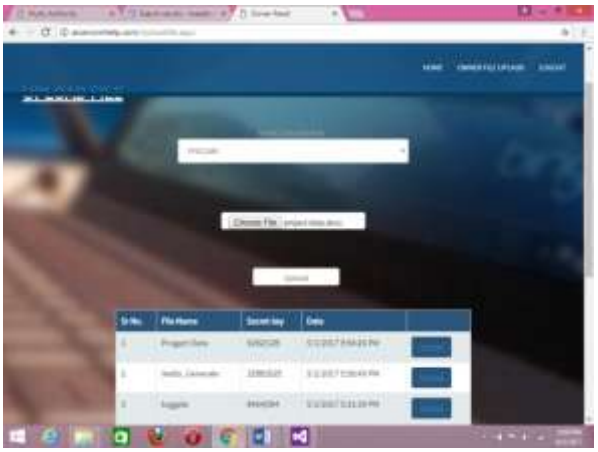


Fig 5: Upload file

After receiving the secret key , owner can upload the file to the cloud server.

Fig6: Key Encryption

In the key encryption AAs can get the encrypted keys and the request from user for the keys.AA send this encrypted key to user for donloding the file from cloud server file.

VII. RESULT

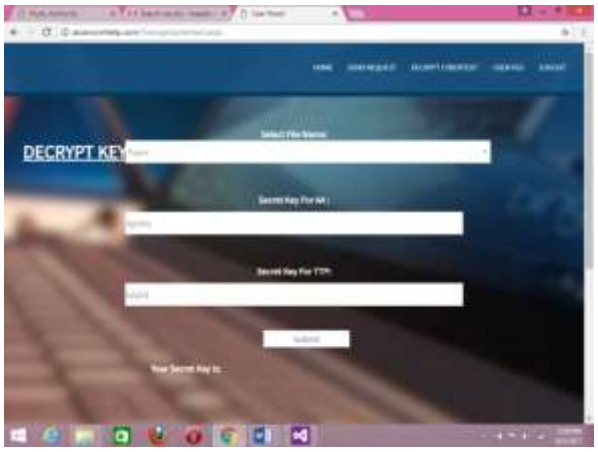


Fig7: Decrypt key

User can decrypt the key using all share of key which is obtained from AA .After performing decryption user can get the original key.

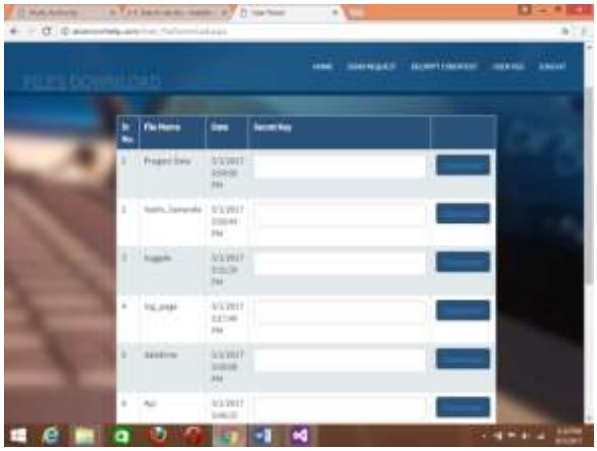


Fig8: User file download

Using original key user download the file from cloud server.

VIII. CONCLUSION

In this paper we proposed multiauthority access control system. This schemea revocable decentralized data access control system can support efficient attribute revocation for multi-authority cloud storage systems. It eliminates decryption overhead of users according to attributes. This secure attribute based encryption technique for robust data security that is being shared in the cloud. This revocable multi-authority data access scheme with verifiable outsourced decryption and it is secure and verifiable. This scheme will be a promising technique, which can be applied in any remote storage systems and online social networks etc.

ACKNOWLEDGMENT

The authors would really like to CSE Department and Principal of TGPCET TulsiramGaiyakwadPatil college of Engineering & Technology providing their support and facilities like labs, software’s etc. needed to carry out this work.

REFERENCES

[1] Wei Li, KaipingXue, YingjieXue, and Jianan Hong. “TMACS: A Robust and Verifiable Threshold Multi-Authority Access Control System in Public Cloud Storage” VOL. 27, NO. 5, MAY 2016

[2] B. Waters, “Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization,” in Proc.14th Int. Conf. Practice Theory Public Key Cryptography, 2011, pp. 53–70

[3] K. Yang and X. Jia, “Attributed-based access control for multi-authority systems in cloud storage,” in Proc. IEEE 32nd Int. Conf.Distrib. Comput. Syst., 2012, pp. 536–545.

[4] G.Rajesh Babu, Ananth Kumar ,”Security In Inter Cloud Data Transfer” International Journal of Innovative Research in Computer Science & Technology (IJIRCST) ISSN: 2347-5552, Volume-2, Issue-5, September-2014.

ISSN: 2454-4248
204 – 209

[5] T. Jung, X. Li, Z. Wan, and M. Wan, "Privacy preserving cloud data access with multi-authorities," in Proc. 32nd IEEE Int. ConfComput. Commun., 2013, pp. 2625–2633.

[6] S. Patil, P. Vhatkar, and J. Gajwani, "Towards secure and dependable storage services in cloud computing," Int. J. Innovative Res.Adv. Eng., vol. 1, no. 9, pp. 57–64, 2014.

[7] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in Proc.29th IEEE Int. Conf. Comput. Commun., 2010, pp. 1–9.

[8] J. Li, X. Huang, J. Li, X. Chen, and Y. Xiang, "Securely outsourcing attribute-based encryption with checkability," IEEE Trans. ParallelDistrib. Syst., vol. 25, no. 8, pp. 2201–2210, Aug. 2014.

[9] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, "Scalable and secure sharing of personal health records in cloud computing using attribute- based encryption," IEEE Trans. Parallel Distrib. Syst.,

[10] G.Rajesh Babu, G.Ananth Kumar, Vishal Tiwari," Security Risks Associated with the Cloud Computing, International Journal of Research (IJR) ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 06, June 2015.

[11] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attri- bute-based encryption," in Proc. IEEE Symp. Security Privacy, 2007, pp. 321–334ol. 24, no. 1, pp. 131–143, Jan. 2013.

[12] J. Hur, "Improving security and efficiency in attribute-based data sharing," IEEE Trans. Knowl. Data Eng., vol. 25, no. 10, pp. 2271– 2282, Oct. 2013.

[13] Z. Wan, J. Liu, and R. Deng, "Hasbe: A hierarchical attribute based solution for flexible and scalable access control in cloud computing," IEEE Trans. Inf. Forensics Security, vol. 7, no. 2, pp. 743–754, Apr. 2012.

[14] Y. Wu, Z. Wei, and H. Deng, "Attribute-based access to scalable media in cloud-assisted content sharing," IEEE Trans. Multimedia, vol. 15, no. 4, pp. 778–788, Jun. 2013

[15] M. Chase, "Multi-authority attribute based encryption," in Proc. 4th Theory Cryptography Conf., 2007, pp. 515–534.

[16] A. Lewko and B. Waters, "Decentralizing attribute-based encryption," in Proc. 30th Annu. Int. Conf. Theory Appl. Crypto- graphic Techn., 2011, pp. 568–588

[17] H. Lin, Z. Cao, X. Liang, and J. Shao, "Secure threshold multi- authority attribute based encryption without a central authority," Inf. Sci., vol. 180, no. 13, pp. 2618–2632, 2010.

[18] T. Pedersen, "A threshold cryptosystem without a trusted party," in Proc. 10th Annu. Int. Conf. Theory Appl. Cryptographic Techn., 1991, pp. 522–526.