CLOUD BASED ACCESS CONTROL MODEL FOR SELECTIVE ENCRYPTION OF DOCUMENTS WITH TRAITOR DETECTION

Punya Peethambaran[1] and Dr. Jayasudha J. S.[2]

[1]Department of Computer Science and Engineering, SCT College of Engineering, Trivandrum, Kerala
[2]Head of the Department, Department of Computer Science and Engineering, SCT College of Engineering, Trivandrum, Kerala

ABSTRACT

Cloud computing refers to a type of networked computing whereby an application can be run on connected servers instead of local servers. Cloud can be used to store data, share resources and also to provide services. Technically, there is very little difference between public and private cloud architecture. However, the security and privacy of the data is a very big issue when sensitive data is being entrusted to third party cloud service providers. Thus encryption with a fine grained access control is inevitable to enforce security in clouds. Several techniques implementing attribute based encryption for fine grained access control have been proposed. Under such approaches, the key management overhead is a little bit high in terms of computational complexity. Also, secret sharing mechanisms have added complexity. Moreover, they lack mechanisms to handle existence of traitors. Our proposed approach addresses these requirements and reduces the overhead of the key management as well as secret sharing by using efficient algorithms and protocols. Also, a traitor tracing technique is introduced into the cloud computing two layer encryption environment.

KEYWORDS

Cloud computing, attribute-based encryption, access control, group key management, fine grained access, cloud data security.

1. INTRODUCTION

Cloud computing is a widely used technology that aids in sharing data as well as resources and services through the internet [1] [2]. Cloud computing has lots of beneficial characteristics such as agility, reduced cost, device and location independence, easier maintenance, multitenancy, performance, broad network access etc. Today the computing world has attracted lots of organizations as well as individuals to store data on clouds for easily sharing data and thus to reduce the cost of sharing. It is well known that a coin will always have two sides. Even though the advantages of cloud data sharing are a boon, the security of the private data is a serious issue in case of really sensitive data. The private data should be made available only to the users who are authorized to use it.

Cloud computing requires the user to transfer their data to the cloud service provider for business as well as storage purposes. Cloud service providers cannot be fully trusted too [3]. Even though data sounds to be a simple thing, it is the most important asset for a business organization. If sensitive data is disclosed to the public or any other competitors of organizations, serious consequences may follow. Thus when cloud is used, priority goes to ensure that the data is kept

DOI : 10.5121/ijnsa.2014.6502

confidential and that not even the cloud service provider has access to the data that is transferred to the cloud [4]. The responsibility to keep the data safe from unauthorized access is to the organization itself in case of private clouds. But in case of public clouds, there are chances of data theft through internet. Therefore in public clouds, before uploading data to the clouds for sharing, the data owner will encrypt the data. By this method, the cloud service providers will not be able to access the data. Along with this, in order to avoid unauthorized users from accessing the data, the encryption should be done taking into consideration the access control policies (ACPs) of the organization [5].

The attributes specified in the ACPs reveal private information. So they should also be protected. It also helps face insider threats. Other attribute based encryption as well as proxy re- encryption based methods where proposed earlier but they couldn’t efficiently add or revoke users, attributes or policies. Other simple group key management techniques also lacked scalability as well as user attribute privacy. Recently proposed approach based on broadcast group key management address these issues [6].

We observe that, adding new users, revoking users and updating ACPs is performed by running the key generation algorithm again and thus producing a new key and public information. The key generation being done is the most computationally expensive operation in the scheme [6]. This is improvised in this paper by incorporating a newer version of the key management algorithm used. We also observed that the secret sharing protocol can be made more efficient by handling multiple conditions at the same time. Otherwise the communication and computation costs will increase in proportion to the number of attributes. The existence of traitors is also handled in the proposed scheme along with an audit log.

The remainder of the paper is organized as follows: Section 2 introduces the related works. Section 3 gives an overview of the overall system that utilizes the fast group key management and secret sharing. Section 4 presents the basic building blocks of this system. Section 5 gives a detailed description of the proposed system and shows how to trace the traitors and thus prevent pirates from accessing the data. Section 6 presents the experimental results and section 7 concludes the paper and outlines future research directions.

2. LITERATURE SURVEY

Many encryption techniques have been already proposed, that can achieve and provide security, prevent collusion attack and assure confidentiality of the data [7].

For fine grained access control of data, various models of access control exist such as Mandatory Access Control (MAC), Discretionary Access Control and Role Based Access Control. These are all identity based access control models because users and resources have unique names to identify them in all these models. These access control models are efficient in unchangeable systems where the set of users and services are known prior to. But nowadays users and resources are dynamic and ad-hoc. Also, the users can be identified by some attributes or features rather than identities that are predefined and fixed. Therefore the old approach of access control based on identity is not effective and access must be decided based on the attributes of the users. The access control technique must be flexible. Such access control techniques are called attribute based access control techniques. Attribute based access control grants access rights to users based on access control policies that are based on many attributes. The attributes can be either the attributes of the user or attributes of the resources. These attributes can be compared or cross checked against fixed values or against each other. This can also be called as relation based access control.

Attribute based encryption was first proposed by Amit Sahai and Brent Waters in 2005 [8]. It is actually a type of public key encryption in which the cipher text as well as the secret key of users is dependent upon attributes (E.g. age of a person, years of service of the user etc.). The authority of the system is responsible for generating keys for data owners and users for encryption and decryption of data. The list of attributes based on which the key is generated has to be predefined. If any new users with new attributes are admitted to the system, authority has to redefine the attributes and regenerate the keys and re encrypt. The decryption of a cipher text should only be possible if the attributes of the cipher matches the attributes of the user's key. Mathew Piretti [9] introduced the attribute based system in a distributed environment and has shown that it is an efficient and effective solution for managing data in a large loosely coupled distributed system in a secure manner [8].

In applications that involve groups of users, group key management is an important area of research [10] [11]. In early days, a trusted key server was responsible of sharing keys to users based on some secrets [12] [13]. But in those approaches, ensuring forward and backward secrecy was a tedious process as it involved sending new key information. To remove these limitations, hierarchical key management schemes were introduced [14] [15]. But only the key size while rekeying is reduced but each user have to maintain redundant keys that are hierarchically organized.

Yongdong wu et al. proposed an interesting attribute based access technique that moved the computationally intensive operations to cloud servers and thus supported portable devices that have limited resources without any compromise on the security aspects [16]. This scheme encrypts multiple data to a single cipher text. But this system allows only one privilege level and so is not much suitable for access control to scalable media. But the decryption is sometimes slow at low end devices because of a modular exponentiation operation.

3. OVERVIEW

In this section, an overview of our solution to the problem of complexity in update operation of ACV BGKM and existence of traitors in outsourced data in cloud environment is presented. A detailed description is provided in section 5.

Fine grained attribute based access control of documents with selective encryption can be achieved simply by encrypting each of the portion of the document that confirms to the same set of policies with the same key. This encryption process can be done while uploading to the cloud.

As shown in Fig. 1, the main steps of the scheme are identity token issuance, identity token registration and document management. Traitor tracing will be enabled during the whole process along with the audit log mechanism.

This scheme consists of mainly four entities, Owner, User, Cloud and the Identity Provider (IdP).

Owner defines the access control policies and uploads encrypted documents to the cloud.

- Cloud holds the encrypted data of the owner, public information indexed to the policy configurations and the audit log.
- IdP is a trusted third party that issues identity tokens to the users based on the identity attributes confirmed by the user. This is done based on a commitment scheme such as Pedersen commitment.
- User will register to the owner to get access to the encrypted data in the cloud after authentication.

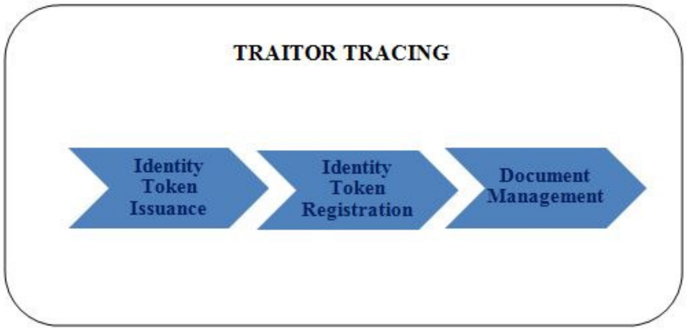

Fig. 1: Overall Architecture of the system

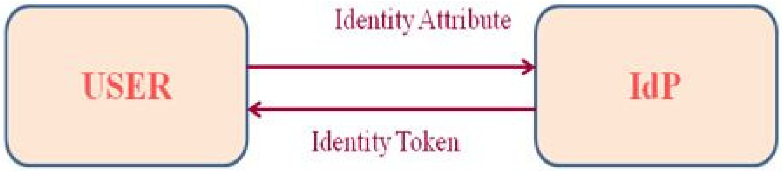

Fig. 2: Identity Token Issuance

Identity token issuance phase shown in Fig. 2 involves generation of Identity token based on the Pedersen commitment scheme which is explained in section IV. Even if there is multiple identity providers the same format is maintained for generating identity token by all of them to ensure proper working of the system.

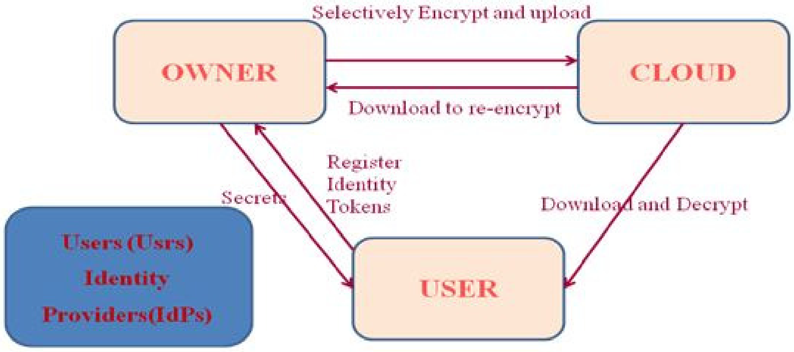

Fig. 3: Identity Token Registration and Secure upload of documents

Identity tokens are provided by the user to the owner before accessing the documents from the cloud as shown in Fig 3. Based on these data some secrets are generated and sent back to the user following the Aggregate EQ-OCBE protocol as explained in Section IV [19]. After this step FACV – BGKM algorithm is used for key management [24].

4. FUNDAMENTALS

In this section we first introduce Pedersen commitment; oblivious commitment based envelope protocol and broadcast encryption schemes which form the basis of this work.

4.1 Commitment Scheme – Pedersen Commitment

Pedersen commitment scheme is a computationally binding and unconditionally hiding commitment scheme introduced in [17] based on the discrete logarithm problem. The steps of Pedersen commitment is demonstrated in Fig 4.

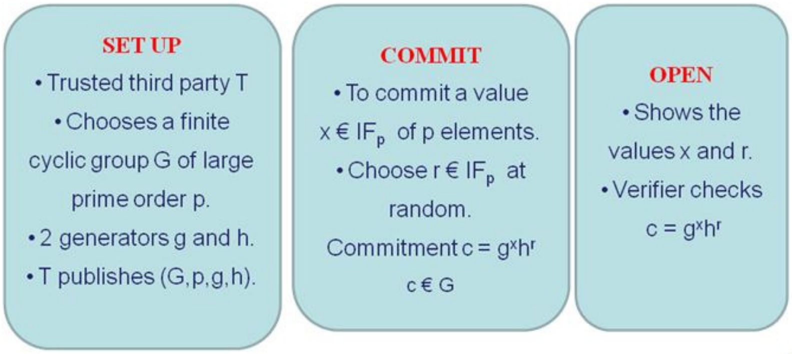
SET UP
• Trusted third party T
• Chooses a finite cyclic group G of large prime order p.
• 2 generators g and h.
• T publishes (G,p,g,h).
COMMIT
• To commit a value $x € IF_p$ of p elements.
• Choose $r € IF_p$ at random.
Commitment $c = g^x h^r$
$c € G$
OPEN
• Shows the values x and r.
• Verifier checks
$c = g^x h^r$

Fig. 4: Pedersen Commitment Scheme

4.2 Oblivious Commitment Based Envelope (OCBE) Protocol

Oblivious commitment based envelope (OCBE) protocols proposed by Li and Li [18] ensure obliviousness in transferring secrets to authorized users. This protocol is built on top of the Pedersen commitment scheme. It is a Diffie Hellman like protocol that will allow the user to get the original data if and only if committed value of the user matches the value specified by the owner of the data.

The traditional EQ-OCBE (Equality OCBE) protocol is used in [6]. But the efficiency issue of EQ-OCBE protocol can be addressed by incorporating Aggregate EQ-OCBE protocol into this system [19]. This will work when several equality conditions are to be satisfied without increasing the cost of computation. For e.g., “Give access to a subdocument if you are a doctor of Trust Hospital in USA”. Traditional EQ-OCBE protocol can also be used by applying it many times. But this approach will consume more bandwidth and will require more communication. Say, if there are n equality conditions to be satisfied in order to access a data section, the number of messages sent in between the owner and the user and cost of computation will increase n times. The more expensive exponentiation operations are replaced by less costly addition and multiplication operations. The simple aggregate EQ-OCBE protocol is illustrated in Fig 5.

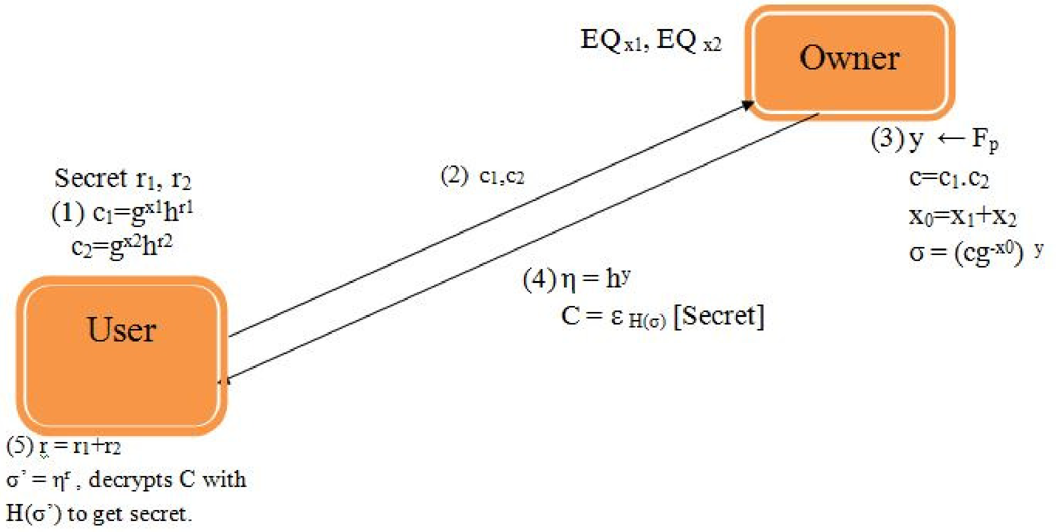

Fig. 5: Aggregate EQ-OCBE protocol

But collision may arise in the simple aggregate EQ-OCBE protocol. To avoid it hashing is introduced to aggregate EQ-OCBE which is illustrated in Fig 6.

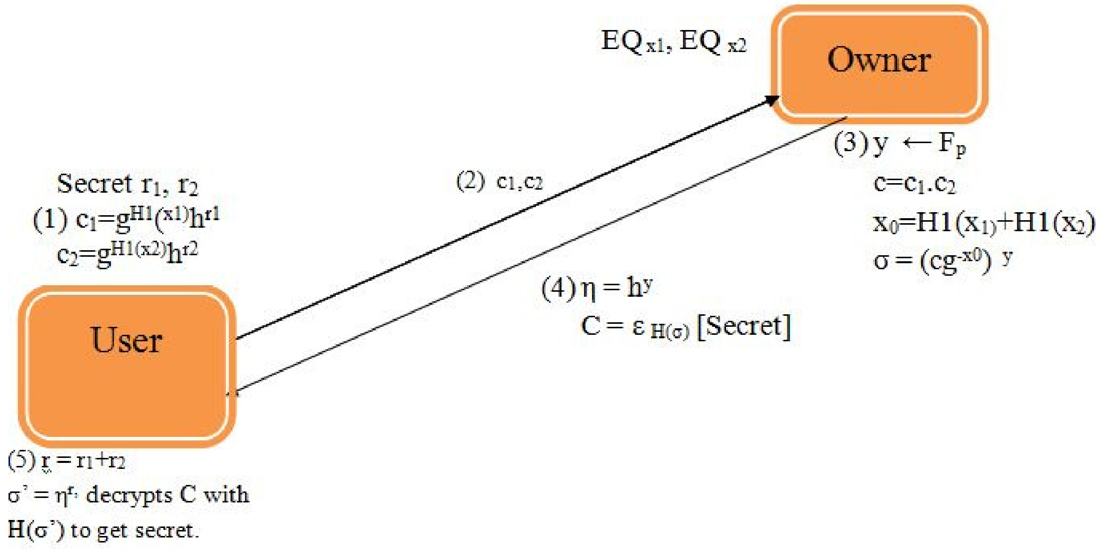

Fig. 6: Aggregate EQ-OCBE protocol with hashing

4.3 Broadcast Group Key Management

In this section, the requirements of an efficient group key management scheme are listed and then an overview of the BGKM scheme used in this system is presented.

Challel and Seba [20] observed and listed certain requirements for an efficient GKM scheme such as minimal trust, key hiding, key independence, collusion resistance, forward and backward secrecy, low bandwidth overhead, ease of maintenance and minimal storage requirements with acceptable computational costs.

The Access Control Vector Broadcast Group Key (ACV-BGKM) scheme makes rekeying a one-off function [21] [22] [23]. A single broadcast is only needed for the rekey operation without the need of any secure channels. Here the private key is not given directly to the users; instead, secrets are combined with the public information to get the private keys [8]. But the update operation is quite expensive in ACV-BGKM.

Fast ACV-BGKM is an extension of ACV-BGKM that minimizes the computational cost of update operations [24]. Pre- computation can be used to improve the efficiency of this scheme. It follows a baby-step-giant-step (BSGS) rekeying method. But there is no need to run the complete key generation step again for updating new users or policies.

FACV-BGKM only takes O (n) time when compared to the O (n3) of ACV-BGKM for the key generation phase [24]. Thus the key generation computation cost is minimized to minimize the overall cost of ACV-BGKM scheme to form the FACV-BGKM scheme.

5. PROPOSED SCHEME

Here, we propose a system based on attribute based encryption in clouds that is more efficient by utilizing aggregate EQ-OCBE protocol, FACV – BGKM, traitor tracing and audit logging.

5.1 Fast ACV-BGKM

For sharing the secrets with the users, the existing system used the basic EQ-OCBE protocol. But it lacked the capacity to handle multiple equalities efficiently. This is overcome in our approach by using Aggregate EQ-OCBE protocol which is explained in Section IV.

In FACV-BGKM, the parameter generation phase is the same as that in ACV-BGKM [24]. But along with the number of users N, a random value M≤N is added and is selected as N. An index i is assigned to each of the user uniformly at random (1≤i≤N) to each of the n currently registered users. Along with that, the conditional subscription secrets are also assigned. The remaining N-n pre-computed secrets are used for rekeying when new users join the group.
For key generation an N × (N+M) finite field matrix A is created for i based on equation 1.

$$a_{i,j} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } 1 \le j \le N \text{ and } i \ne j \\ H(S_i||z_j) & \text{if } N < j \le N + M \end{cases} \qquad (1)$$

Next step is to calculate a null space of A and select a vector Y from the null space. And store the other basis vectors and mark Y as used. Then a (N+M) dimensional finite field vector X is calculated based on equation 2. The public information is the same as ACV-BGKM scheme that comprises of X and the z values for each user.

$$X = (\textstyle\sum_{i=1}^{n} K.e_i^T) + Y \qquad (2)$$

For deriving the key, a user who knows his own index i and his identity token has to derive the (N+M) dimensional row vector which is a row of matrix A. The key can be derived by equation 3.

$$K = v_i.\mathrm{X} \quad (3)$$

The main advantage of this approach comes during the update phase. Whenever a new user joins, a new index i is chosen from the list of unused indexes and a new identity token is provided to him based on the identity attributes of that user. Equation 2 is used once more with a new key in order to compute a new X value. Also, if a user leaves the organization, a new key is chosen and the X value is recomputed. New Y can also be used from the null space of matrix A. So only the concerned public information has to be updated with the new X value and the new key can be used to selectively re-encrypt the documents. So there is no need to run the key generation phase entirely like in ACV-BGKM.

5.2 Traitor Detection and Audit Log

It is always nice to maintain audit logs when accessing data. A log is maintained in the proposed system that does not let anyone identify the user based on the log entries. Only the encrypted identity tokens are stored in the audit logs. The decrypted identity is shown to the owner only when an adversary is detected.

An ideal solution to stop piracy is almost impossible to achieve in reality. It is practical to assume that some piracy will occur but counter measures can be taken to deal with the threat. Broadcast encryption schemes and traitor tracing schemes can be effectively combined in order to minimize the damage caused due to piracy.

It is assumed in this work that the identity tokens generated by the identity provider are not revealed to the user. The pirate can access the files if he gets the secrets assigned to the user, since all other parameters in the key derivation algorithm are public values. As shown in fig. 7, the set of secrets for each policy configuration i.e., each subdocument, are extracted during the initial stage. Thus the policy configurations are attached with a set of secrets. So whenever the user wants to access to a subdocument, the index of the secrets provided to the cloud is found and the corresponding registered identity token to which those secrets are assigned is extracted. If and only if there is a match, the subdocument can be decrypted with the original key from FACV-BGKM. If there is a mismatch, the id_tag is extracted from the identity token and the owner is notified and is allowed to view the audit log. The id_tag can be used by the owner to find out the details of the traitor from the identity provider.

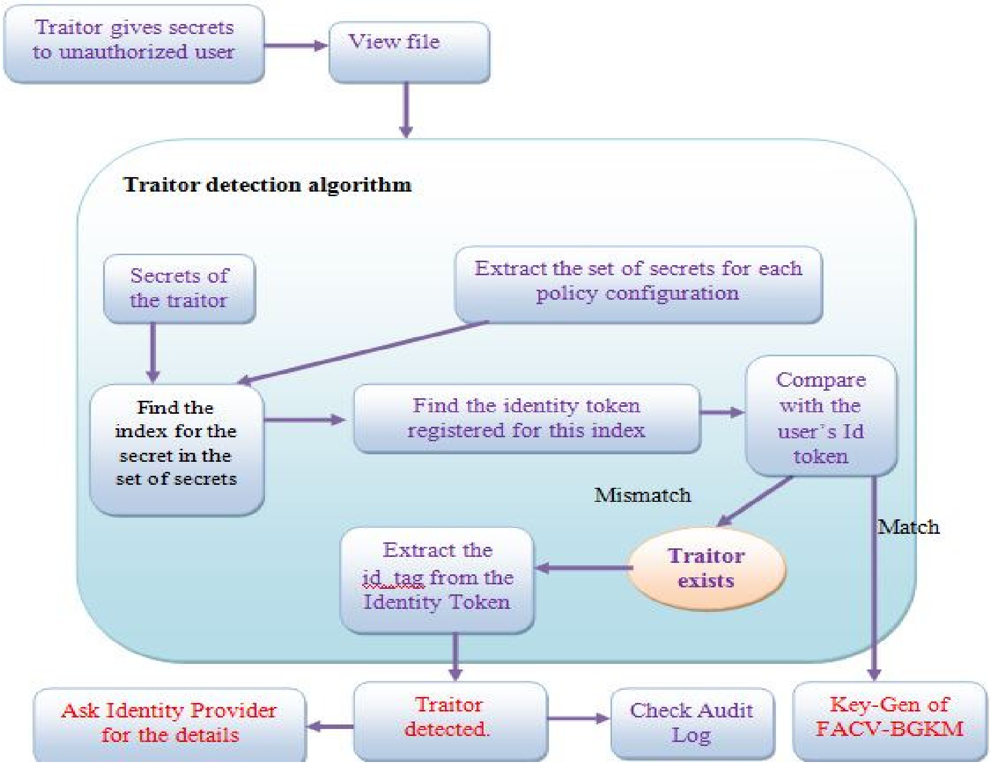
Traitor gives secrets to unauthorized user
View file
Traitor detection algorithm
Secrets of the traitor
Extract the set of secrets for each policy configuration
Find the index for the secret in the set of secrets
Find the identity token registered for this index
Compare with the user's Id token
Mismatch
Match
Traitor exists
Extract the id_tag from the Identity Token
Ask Identity Provider for the details
Traitor detected.
Check Audit Log
Key-Gen of FACV-BGKM

Fig. 7: Traitor Detection Algorithm

6. RESULTS

The use of aggregate EQ-OCBE protocol reduced the time complexity for transferring the secrets between the owner and user. Execution of EQ-OCBE protocol for each equality condition was an overhead. The overhead is removed by using Aggregate EQ-OCBE protocol, which is required to run only once for all the equality conditions.

The use of FACV-BGKM scheme reduced the time complexity of key generation during new user addition or user revocation from O (n^3) to O (n) [24]. It has been proven that this scheme is key hiding as well as forward and backward key protecting. Traitor detection and audit logging enhanced the security of the system by providing added protection.

7. CONCLUSION AND FUTURE WORK

Cloud computing services have introduced a modern trend of outsourcing the data storage and manipulation functions to third party cloud service providers. But, serious security issues may arise due to the same. We have proposed an extension to the ACV-BGKM scheme for attribute based fine grained access control with better user addition and revocation computations, better evaluation of expressive access control policies and security.

FACV-BGKM scheme attains better computational complexity at the expense of higher space complexity and pre-computation. When the data is encrypted and stored in clouds, searching for a keyword and retrieving only those files is a difficult problem. Thus privacy preserving querying remains an open problem.

REFERENCES

[1] R. Buyya, C. ShinYeo, J. Broderg and I. Brandic, “Cloud computing and emerging it platforms: Vision, hype and reality for delivering computing as the 5th utility”, Future Generation computer systems 2009, pp. 599-616.

[2] M. Armbrust, A. Fox, R. Grifth, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica and M. Zaharia, “A view of cloud computing”, Communications of the ACM, vol. 53, pp. 50-58, 2010.

[3] C. C. Chang, I. C. Lin and C. T. Liao, “An access control system with time-constraint using support vector machines", International Journal of Network Security, vol. 2, no. 2, pp. 150-159, 2006.

[4] S. F. Tzeng, C. C. Lee and T. C. Lin, “A novel key management scheme for dynamic access control in a hierarchy", International Journal of Network Security, vol. 12, no. 3, pp. 178-180, 2011.

[5] S. Kamara and K. Lauter, “Cryptographic cloud storage", Proceedings of the 14th international conference on financial cryptography and data security, pp. 136-149, 2010.

[6] Mohamed Nabeel, Ning Shang and Elisa Bertino, “Privacy Preserving Policy-Based Content Sharing in Public Clouds”, IEEE Transactions on knowledge and data engineering, vol. 25, no. 11, November 2013

[7] J. S. Su, D. Cao, X. F. Wang, Y. P. Su and Q. L. Hu, “Attribute-based encryption schemes", Journal of Software, vol. 6, pp. 1299-1315, 2012.

[8] A. Sahai and B. Waters, “Fuzzy identity based encryption", Advances in Cryptology V EUROCRYPT, vol. 3494 of LNCS, pp. 457-473, 2005.

[9] M. Pirretti, P. Traynor, P. McDaniel and B. Waters, “Secure attribute based systems”, Journal of Computer Security, vol. 18, no. 5, pp. 799–837, 2010.

[10] Y. Challal and H. Seba, “Group key management protocols: A novel taxonomy,” International Journal of Information Technology, vol. 2, no. 2, pp. 105–118, 2006.

[11] X. Zou, Y. Dai and E. Bertino, “A practical and flexible key management mechanism for trusted collaborative computing,” INFOCOM 2008. The 27th Conference on Computer Communications. IEEE, pp. 538–546, April 2008.

[12] H. Harney and C. Muckenhirn, “Group key management protocol (GKMP) specification,” Network Working Group, United States, Tech. Rep., 1997.

[13] H. Chu, L. Qiao, K. Nahrstedt, H. Wang and R. Jain, “A secure multicast protocol with copyright protection,” SIGCOMM Comput. Commun. Rev., vol. 32, no. 2, pp. 42–60, 2002.

[14] C. Wong and S. Lam, “Keystone: a group key management service,” in International Conference on Telecommunications, ICT, 2000.

[15] Sherman and D. McGrew, “Key establishment in large dynamic groups using one-way function trees,” Software Engineering, IEEE Transactions on, vol. 29, no. 5, pp. 444–458, May 2003.

[16] Yongdong Wu, Zhuo Wei and Robert H. Deng, “Attribute-Based Access to Scalable Media in Cloud-Assisted Content Sharing Networks”, IEEE transactions on multimedia, vol. 15, no. 4, June 2013.

[17] T. Pedersen, “Non-interactive and information-theoretic secure verifiable secret sharing,” in CRYPTO ’91: Proceedings of the 11th Annual International Cryptology Conference on Advances in Cryptology. London, UK: Springer-Verlag, 1992, pp. 129–140.

[18] J. Li and N. Li, “OACerts: Oblivious attribute certificates,” IEEE Transactions on Dependable and Secure Computing, vol. 3, no. 4, pp.340–352, 2006.

[19] Ning Shang, Federica Paci,Elisa Bertino, "Efficient and privacy-preserving enforcement of attribute-based access control", Proceedings of the 9th Symposium on Identity and Trust on the Internet, pp 63-68, New York,USA, 2010.

[20] Y. Challal and H. Seba, "Group key Management Protocols: A Novel Taxonomy," Int'l J. Information Technology, vol. 2, no. 2, pp. 105-118, 2006.

[21] G. Chiou and W. Chen, "Secure Broadcasting Using the Secure Lock," IEEE Trans. Software Eng., vol. 15, no. 8, pp. 929-934, Aug. 1989.

[22] S. Berkovits, "How to Broadcast a Secret," Proc. 10th Ann. Int'l Conf. Advances in Cryptology (EUROCRYPT '91), pp. 535-541, 1991.

[23] X. Zou, Y. Dai and E. Bertino, "A Practical and Flexible Key Management Mechanism for Trusted Collaborative Computing," Proc. IEEE INFOCOM, pp. 538-546, Apr. 2008.

[24] Ning Shang, Mohamed Nabeel, Elisa Bertino and Xukai Zou, "Broadcast Group Key Management with Access Control Vectors", CERIAS TR 2010.

[25] Benny chor, Amos Fiat, Moni Naor, "Tracing Traitors", Proceedings of the 14th Annual International Cryptology Conference on Advances in Cryptology, pp 257-270, Springer, UK,1994.

Authors

Punya Peethambaran is currently doing her M Tech in Computer Science and Engineering at Sree Chitra Thirunal College of Engineering under Kerala University, Trivandrum, Kerala, India. Punya received her B Tech Degree in Computer Science and Engineering from Cochin University College of Engineering, Kerala, India in 2011. Her interest domain is security, wireless network, intrusion tolerance and cloud computing. She has worked in the industry for one year. She has published papers on page replacement algorithms in flash aware swap systems and MANET misbehavior detection.

Dr. Jayasudha J S is working as professor and head of the department at the department of computer science and engineering, Sree Chitra Thirunal College of Engineering, Trivandrum, Kerala. She did her B. E. degree from Madurai Kamaraj University and M. E. degree from National Institute of Technology, Trichy and doctorate degree from University of Kerala. Her Ph.D. thesis title is "Web caching and Pre-fetching techniques for Web traffic/Latency reduction". She is recognized as approved research guide for PhD works in Computer Science and guiding Ph.D. students in Manonmaniam Sundaranar University and Noorul Islam University. Now she is also doing research in Computer Networks. She has published her research works in many national and international conferences and journals. She has 18 years of teaching experience and has organized many community development programs, short term courses and conferences.