ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 4, September 2014

An Efficient and Secured Storage Delegated Access Control to Maintain Confidentiality of Data

Syed Yasmeen[1], M. Naveen Kumar [2]

M.Tech Student Department of CSE, S. V. Engineering College for Women, Tirupati, Chittoor, Andhra Pradesh, India[1]

Assistant Professor, Department of CSE, S.V. Engineering College for Women, Tirupati, Chittoor, Andhra Pradesh, India[2]

ABSTRACT: Security and privacy represent major concerns in the adoption of cloud technologies for data storage. An approach to mitigate these concerns is the use of fine grained access control encryption. But in this approach Data owners thus incur high communication and computation costs. To overcome this problem data owner performs a coarse-grained broad cast encryption along with two layer process, whereas the cloud performs a fine-grained encryption on top of the owner encrypted data. A challenging issue is how to decompose access control policies (ACPs) such that the two layer encryption can be performed. A better approach should delegate the enforcement of fine-grained access control to the cloud, so to minimize the overhead at the data owners, while assuring data confidentiality from the cloud. Our proposed go a step further in the dispense process in two layer broadcast encryption schema, by removing the group manager initial setup of the group, as well as the addition of further members to the system, does not require any central authority. Under our approach, the data owner performs a coarse-grained encryption, whereas the cloud performs a fine-grained encryption on top of the owner encrypted data. Our considered is an extension to the subset-cover framework. It allows for efficient concrete instantiations, with parameter sizes that match those of the subset-cover constructions, while at the same time achieving the highest security level in the standard model. We utilize coherent group key management scheme that supports expressive ACPs. Our system assures the confidentiality of the data and preserves the privacy of users from the cloud while delegating most of the access control enforcement to the cloud.

KEYWORDS: Privacy, access control, security, policy, delegation Cloud Computing and Dynamic Encryption, Two layer encryption.

I. INTRODUCTION

In traditional access control models, the set of access rights a user can get is predetermined. Predetermining of access rights is equivalent to the possibility usage of the system by that user. These system admin acts every time a user need to access and then user gets the right from another user who already possesses it. This approach is known as delegation.

Delegation provides flexibility to access control model. Zhang[4] identify two cases when delegation is necessary. Initially, an individual is absent from their duty and so, someone else should carry out the tasks. Secondly, delegation is allowed to dispense the authority. Having one system admin who assigns access rights to all the user in the system would decrease efficiency.

For understanding privacy preserving delegation, we need an environment where data providers provide privacy policies for their data. The policies specify how to use the data and who can use the data. The access control model in such environments control data accesses based on the privacy policies. This is termed as privacy preserving access control model. There are number of models have been proposed. One of our contributions is to define a privacy model that allows a data provider to set privacy policies for different types of organizations accessing their data.

Broadcast encryption is a schema that allows a sender to send an encrypted text to some designated groups whose members of the group can decrypt it with his/her private key. And no one else from outside the group can decrypt the text. The broadcast encryption can be divided into two categories from a relation of receivers. Firstly, a sender can randomly designate several receivers. User in this category has no relation between each other. Secondly, a sender can encrypt a message to a designated group in which each user in the group can use his private key independently to decrypt the encrypted text. User can communicate with other users in the group and all users in the group are listening on a broadcast channel. The first category has lots of advantages. It is more flexible than the other category and sender can randomly designated a subset of receivers. However, these advantages make the first category much more complicated. It is very difficult to make the scheme satisfy so many advantages while keep the cipher text and keys constant size. For a network like a mobile ad hoc network, the complex in computation and the need for large memory make it inefficient.

Recently proposed approaches based on broadcast key management schemes address some of the above limitations. This refers to single layer encryption (SLE) approaches. This approach requires the data owner to implement access control through encryption performed at data owner. But like other previous approaches, SLE assures the privacy of the users and support fine-grained ACPs.

To overcome this limitation of SLE initiates a new model to address this. The approach is built on two layer of encryption applied to each and every data item uploaded to the cloud. It refers to two layer encryption (TLE), the data owner performs a coarse- grained encryption above the data in order to convince the confidentiality of the data from the cloud. Then the cloud performs a fine-grained encryption above the encrypted data provided by the data owner based on ACPs provided by the owner. But here the problem is with two layer encryption, the broadcast encryption is centralized group manager at the setup phase, this is static in nature. At any interval if a new member wants to enter the system then the schema must be dynamic. The dynamic situation is most realistic and requires security and availability. Our goal is to change the centralized system.

The proposed privacy model is integrated with the access control model to get privacy aware access policy. The policies containing constraints that specify the exact use of data along with two layers dynamic encryption schema, there is no need of specifying centralized authority each and every time to access the data in cloud computing group key managements process. Data users allotted to these policies use data as stated by the constraints. On the basis of these built, we proposed a delegated two layer dispense dynamic model where access policy to the data item can be delegated only if it satisfy the policies. This model checks the delegation operation to preserve the security

.

II. RELATED WORK

In this section we first introduce broadcast encryption schemes [9] [10] and. We present two layer encryption [5] [6][7]and an abstract view of the algorithms of policy decomposition[5].

III. PROPOSED SYSTEM

3.1 Dynamic Broadcast Encryption

Broadcast encryption (BE) was introduced to solve the problem of how to efficiently encrypt a message and broadcast it to a subset of the users in a system. The subset of users can change dynamically. In the broadcast encryption literature, these users are called *privileged* and the non-authorized users *revoked*. We denote the set of users by U, the set of revoked users R. The set of privileged users is thus U\R. We set N = |U| and r = |R|. While all users can get the encrypted message, only the privileged users can decrypt it. The simplest broadcast encryption scheme simply consists of encrypting a message for each privileged user separately and then broadcasting all the encrypted messages. Obviously, this scheme is very inefficient as the message length is prohibitively large (O(N − r)).

We use an algorithm based on *subset-cover* algorithm that supports broadcast encryption with stateless users. The algorithm builds a binary tree and assigns users to the leaf nodes and thus results in a predefined user grouping. Each such group is called a subset. A user can be a member of several subsets. The *cover*, denoted by C, is defined as the set of subsets that contains all the privileged users, that is, users in U/R. The subsets in the cover are disjoint and hence each privileged user belongs to only one subset.

We implement the two layer encryption scheme over the untrusted public cloud. The two layer implementation reduces the load over the owner and delegates the enforcement of access control over the cloud. The system provides a better way for various updates, user location and modification of data, and also includes an additional phase when compared to the existing system. The decomposition and dividing of the data to store across the different clouds and it is finally retrieved with the help of the keys provided by the provider to the users. In the proposed system we use symmetric key for both secure encryption and decryption. The group key management over the data owner and cloud service, the actual key are not given to the user. But it distributes one or more temporary keys that which allow retrieving the actual symmetric key for the decryption of data from the public cloud. We move to the dispense dynamic broadcast scheme to store data over cloud with the dynamic changes of access control mechanism policies, that the authorized user with the valid key only able encrypt and decrypt the data that which stored in the untrusted cloud environment.

3.2 TWO LAYER DYNAMIC ENCRYPTION

The two layer encryption approach implements six phases over secure encrypt and decrypt of data. The approach consists of four entities Data Owner, Authorized User, Identity Provider and Cloud.

A. *Identity token issuance*: IdPs issue identity tokens to Usrs based on their identity attributes.

B. *Policy decomposition*: The Owner decomposes each ACP into at most two sub ACPs such that the Owner enforces the minimum number of attributes to assure confidentiality of data from the Cloud. The decomposed ACPs must be consistent so that by integrating the sub ACPs we can generate the original ACPs. The Owner enforces the confidentiality related sub ACPs and the Cloud enforces the remaining sub ACPs.

C. *Identity token registration*: Usrs register their identity tokens in order to obtain secrets to decrypt the data that they are allowed to access. Usrs register can only those identity tokens related to the Owner's sub ACPs and register the remaining identity tokens with the Cloud in a privacy preserving manner. During this phase owner keeps one set and another set is provided to the cloud which prevent the cloud from decrypting the owner encrypted data.

D. *Data encryption and upload*: The Owner initially encrypts the data based on the Owner's sub ACPs in order to hide the content from the Cloud and then uploads them along with the public attribute and the cloud based on sub ACPs encrypt the uploaded encrypted data.

E. *Data download and decryption:* The data can be downloaded from the cloud by the user by making use of the keys derived. Hence the user can get access to those data which have valid keys.

F. *Future Encryption evolution* :
When the user ACPs credentials change, the encrypted data need to be updated. In such case, re-encryption on data should be performed using new keys. This is possible as cloud implements access control encryption. It is simply to re-encrypt the affected data without interference of the owner.

The following condition is checked by the cloud in order to decide if re-encryption is required.

1. The new group of Usrs is a strict superset of the old group of Usrs, and backward secrecy is obtained.

2. For any ACP, the new group of Usrs is a strict subset of the old group of Usrs, and forward secrecy is enforced for the already encrypted data items.

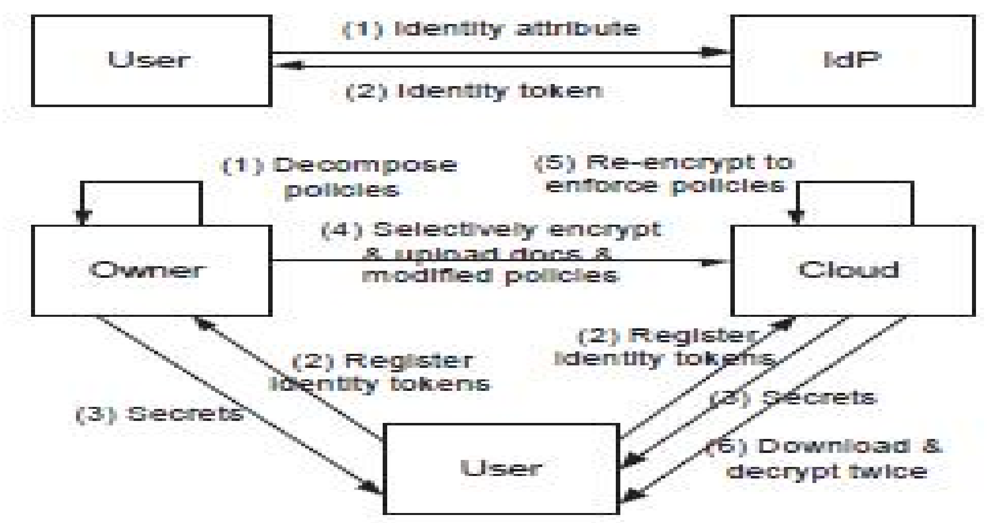

User
(1) Identity attribute
(2) Identity token
IdP
(1) Decompose policies
(5) Re-encrypt to enforce policies
Owner
(4) Selectively encrypt & upload docs & modified policies
Cloud
(2) Register identity tokens
(2) Register identity tokens
(3) Secrets
(3) Secrets
(6) Download & decrypt twice
User

Figure 1: Two Layer Encryption

IV. POLICY DECOMPOSITION

The Owner manages only those attribute conditions in ACC. The Cloud handles the remaining set of attribute conditions, ACB/ACC. The Owner re-writes its ACPs such that they cover ACC. In other words, the Owner enforces the parts of the ACPs related to the ACs in ACC and Cloud enforces the remaining ACs along with some ACs in ACC. The POLICY-DECOMPOSITION [5] algorithm 1 shows how the ACPs are decomposed into two sub ACPs based on the attribute conditions in ACC.

Algorithm 1 takes the ACP and ACC as input and produces the two sets of ACPs ACP_{owner} and ACP_{cloud} that are to be enforced at the Owner and the Cloud respectively. It first converts each policy into DNF and decompose each term into two conjunctive terms such that one conjunctive term has only those ACs in ACC and the other term may or may not have the ACs in ACC. It can be easily shown that the policy decomposition is consistent. That is, the conjunction of corresponding sub ACPs in $ACPB_{owner}$ and $ACPB_{Cloud}$ respectively produces an original ACP in ACPB.

ALGORITHM 1:- POLICY-DECOMPOSITION

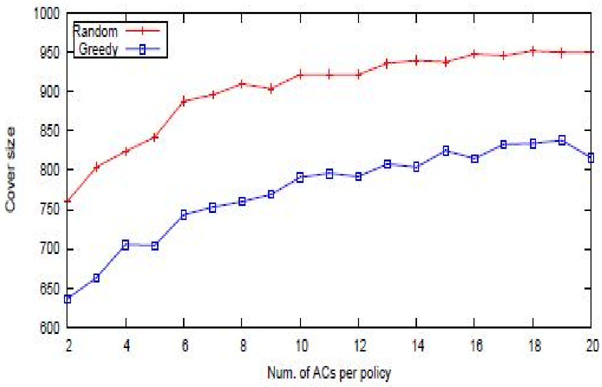
```
$ACPB_{Owner} = \varphi$
$ACPB_{Cloud} = \varphi$

for Each ACPi in ACPB do

Convert ACPi to DNF
```

```
ACPi(owner) = $\varphi$
ACPi(cloud)= $\varphi$
if Only one conjunctive term then
    Decompose the conjunctive term c into c1 and c2 such that ACs in c1 $\in$ ACC, ACs in c2 $\in$ ACC and $c = c1 \wedge c2$
    ACPi(owner)= c1
    ACPi(cloud)= c2
else if At most one term has more than one AC then for Each single AC term c of $ACP'_i$ do
    ACPi(owner)$\vee$= c
    ACPi(cloud)$\vee$= c
end for
    Decompose the multi AC term c into c1 and c2 such that ACs in c1 $\in$ ACC, ACs in c2 $\in$ ACC and $c = c1 \wedge c2$
    ACPi(owner)$\vee$= c1
    ACPi(cloud)$\vee$= c2
else
    for Each conjunctive term c of $ACP'$ do
        Decompose c into c1 and c2 such that ACs in c1 $\in$ ACC, ACs in c2 $\in$ ACC and $c = c1 \wedge c2$
        ACPi(owner)$\vee$= c1
    end for
    ACPi(cloud)= ACP
end if
    Add ACPi(owner) to $ACPB_{Owner}$
    Add ACPi(cloud)  to $ACPB_{Cloud}$
end for
Return $ACPB_{Owner}$ and $ACPB_{Cloud}$
```

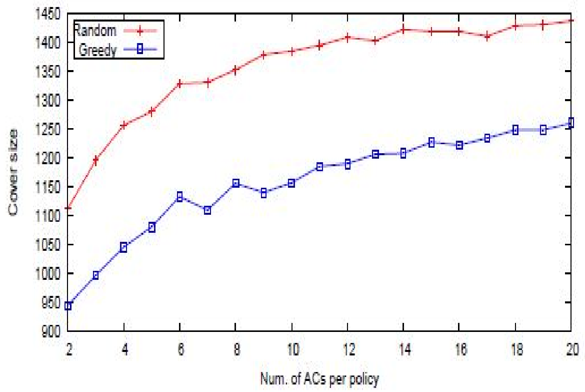
As shown in Algorithm 1, the Owner re-writes the ACPs that the Cloud should enforce such that the conjunction of the two decomposed sub ACPs yields an original ACP.

V. ANALYSIS

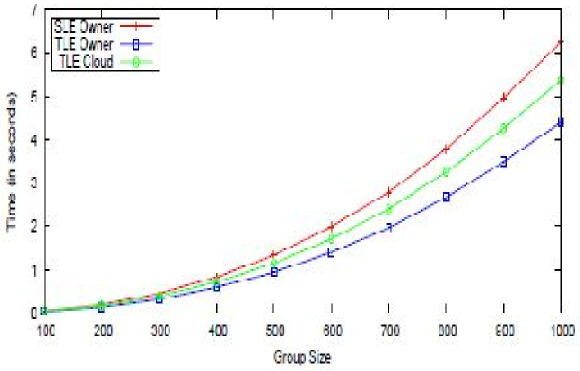
In this section, result concerning the policy decomposition algorithms. We then present an experimental comparison between the SLE and TLE approaches. Then give a high level analysis of the security and the privacy of both approaches.



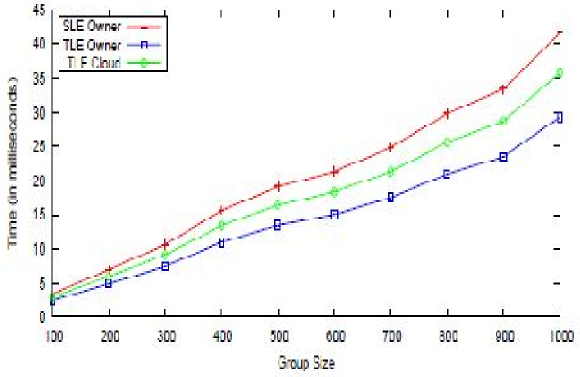
(a) Size of ACCs for 1000 attributes



(b) Size of ACCs for 1500 attributes



(c) Average time to generate keys for SLE vs. TLE



(d) Average time to derive keys for SLE vs. TLE.

VI. CONCULSION AND FUTURE WORK

Present approaches to implement ACPs on data outsourcing using selective encryption which requires organizations to maintain all keys, encryption, and upload encrypted data to remote storage. Such approach leads to high computational cost to manage keys and encryption whenever user credentials changes. In this paper, we proposed a two layer dynamic encryption based approach to solve this problem by delegating the access control enforcement responsibilities to the Cloud while minimizing the information exposure risks at Usrs and Cloud. A key problem in this model is how to decompose ACPs so that the Owner has to handle a minimum number of attribute conditions while hiding the content from the Cloud. We showed that the policy decomposition algorithms. Based on the decomposed ACPs, we proposed delegated access control to maintain confidential of data. Our future work, we plan to investigate the alternative choices for the TLE approach further. We also plan to further reduce the computational cost by exploiting partial relationships among ACPs.

REFERENCES

[1]M. Nabeel and E. Bertino, “Privacy preserving delegated access control in the storage as a service model,” in *EEE International Conference on Information Reuse and Integration (IRI)*, 2012.

[2]S. D. C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, “Over-encryption: Management of access control evolution on outsourced data,” in *Proceedings of the 33rd International Conference on Very Large Data Bases*, ser. VLDB ’07. VLDB Endowment, 2007, pp. 123–134.

[3]E. Bertino and E. Ferrari, “Secure and selective dissemination of XML documents,” *ACM Trans. Inf. Syst. Secur.*, vol. 5, no. 3,pp. 290–331, 2002.

[4]X. Zhang, S. Oh and R. Sandhu, “PBDM: a flexible delegation model n RBAC,” In proceedings of the eighth ACM symposium on Access control models and technologies (SACMAT), New York, NY, USA, pp. 149–157, 2003.

[5]N. Shang, M. Nabeel, F. Paci, and E. Bertino, “A privacy preserving approach to policy-based content dissemination,” in *ICDE ’10: Proceedings of the 2010 IEEE 26th International Conference on Data Engineering*, 2010.

[6] M. Nabeel, E. Bertino, M. Kantarcioglu, and B. M. Thuraisingham, “Towards privacy preserving access control in the cloud,” in *Proceedings of the 7th International Conference on Collaborative Computing: Networking, Applications and Worksharing*, ser. CollaborateCom ’11, 2011, pp. 172–180.

[7] M. Nabeel, N. Shang, and E. Bertino, “Privacy preserving policy based content sharing in public clouds,” *IEEE Transactions on Knowledge and Data Engineering*, 2012.

[8]M. Nabeel and E. Bertino, “Towards attribute based group key management,” in *Proceedings of the 18th ACM conference on Computer and communications security*, Chicago, Illinois, USA,2011.

[9]A. Fiat and M. Naor, “Broadcast encryption,” in *Proceedings of the 13th Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO ’93. London, UK: Springer-Verlag, 1994, pp. 480–491.

[10]D. Naor, M. Naor, and J. B. Lotspiech, “Revocation and tracing schemes for stateless receivers,” in *Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO ’01. London, UK: Springer-Verlag,2001, pp. 41–62.

BIOGRAPHY

Syed Yasmeen is a student in Master of Technology in the Department of Computer Science and Engineering, S. V Engineering College for Women, Tirupati, Andhra Pradesh, India.

M. Naveen Kumar is an Assistant Professor in the Department of Computer Science and Engineering, S.V. Engineering College for Women, Tirupati, Andhra Pradesh, India.