Secure with Annoy Control Cloud Data Access using Decisional Bilinear Daffier-Hellman Algorithm

Mr. Adapa Gopi [1], Mr.P.Sathish Reddy [2], Mr. Guguloth Venkatesh[3]
[1,2,3]*Department of Computer Science and Engineering, Kasireddy Narayan Reddy College Of Engineering & Research, Hayathnagar, Telangana, India*

Abstract— Cloud computing is a computing concepts, which enables when required and low maintenance usage of resources, but the data is shares to some cloud servers and various privacy related concerns emerge from it. Various schemes like based on the attribute based encryption have been developed to secure the cloud storage. Most work looking at the data privacy and the access control, while less attention is given to the privilege control and the privacy. In this paper, we present a privilege control scheme Anonymity Control to address and the user identity privacy in existing access control. Anonymity Control decentralizes the central authority to limit the identity leakage and thus achieves partial anonymity.It also generates the file access control to the privilege control, by which privileges of all operations on the cloud data can be managed in a proper manner. We present the Anonymity Control-F, which prevents the identity and achieve the anonymity. Our security analysis shows that both Anonymity Control and Anonymity Control-F are secure under the Diffie–Hellman assumption and our performance evaluation exhibits the feasibility of our schemes

Key words: Anonymity, multi-authority, attribute-based encryption.

1 INTRODUCTION

Computing is being transformed to a model consisting of services that are commoditized and delivered in a manner similar to utilities such as water, electricity, gas, and telephony. In such a model, users access services based on their requirements regardless of where the services are hosted. Several computing paradigms have promised to deliver this utility computing vision. Cloud computing is a radical computing paradigm, that enables on-demand, and low-cost usage of computing resources, but the data is outsourced to some cloud servers, and various privacy concerns emerge from it. A service offering computation resources is frequently referred to as Infrastructure as a Service (IaaS) and the applications as Software as a Service (SaaS)[1]. An environment used for construction, deployment, and management of applications is called PaaS (Platform as a Service).

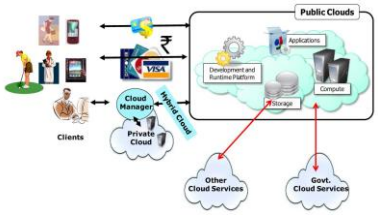

Fig.1: A bird's eye view of Cloud computing

Cloud computing delivers infrastructure, platform, and software (application) as services, which are made available as subscription-oriented services in a pay-as-you-go model to consumers. The price that CSPs (Cloud Service Providers) charge depends on the quality of service (QoS) expectations of CSCs (Cloud Service Consumers).Cloud computing fosters elasticity and seamless scalability of IT resources that are offered to end users as a service through the Internet. Cloud computing can help enterprises improve the creation and delivery of IT solutions by providing them with access to services in a cost-effective and flexible manner [2].

Clouds can be classified into three categories, depending on their accessibility restrictions and the deployment Model. They are:

- Public Cloud,
- Private Cloud, and
- Hybrid Cloud.

A public Cloud is made available in a pay-as-you-go manner to the general public users irrespective of their origin or affiliation. A private Cloud's usage is restricted to members, employees, and trusted partners of the organization. A hybrid Cloud enables the use of private and public Cloud in a seamless manner. Cloud computing applications span many domains, including business, technology, government, health care, smart grids, intelligent transportation networks, life sciences, disaster management, automation, data analytics, and consumer and social networks. Various models for the creation, deployment, and delivery of these applications as Cloud services have emerged. Access controls give organization the ability to control, restrict, monitor and protect resource availability, integrity and confidentiality. Observing all these obligatory things, various access control models have been projected. The disadvantage of those access control models is that, the data owners and service providers are not existing in the same trusted domain in cloud computing. Afterwards, a new access control system was projected by Yu called as Key-Policy Attribute-Based Encryption (KP-ABE) to enforce fine-grained access control. Due to absence of scalability and flexibility in

attribute management, it failed. Cipher text-Policy ABE (CP-ABE) plays a crucial part to impose access control of encrypted data [1]. AnonyControl and AnonyControl-F to allow cloud servers to control users' access privileges without perceptive of their identity information. Their main merits are: 1) the projected systems are able to protect user's privacy in contradiction of each only authority. Partial information is unveiled in AnonyControl and no information is revealed in AnonyControl-F. 2) The projected systems are tolerant against authority compromise, and compromising of up to (N −2) authorities does not bring the whole system down. 3) We provide detailed analysis on security and performance to show possibility of the system AnonyControl and AnonyControl-F. 4) We firstly contrivance the real toolkit of a multi authority based encryption system AnonyControl and AnonyControl-F.

2 RELATED WORK

In this section, we discuss all the different access control systems which provide a facility like availing of data to the user even during the fault occurrence situation in the cloud. To attain flexibility and fine-grained access control, many access control systems have been projected. The main drawback of these systems is they are applicable to the system in which the data owners and the service providers present within the same trusted domain. Later, to overcome this drawback, a new system called as Attribute-Based Encryption [ABE] projected by Yu [7]. Expressibility lacking is the main drawback of ABE system. ABE systems are classified in to Key- Policy Attribute-Based Encryption [KP-ABE] and Cipher text-Policy Attribute Based Encryption [CP-ABE] based on the association of attributes and access policy with cipher texts and user decryption keys. The main problem with KP-ABE system is, here the encrypt or is only able to choose descriptive attribute for the data and has no choice other than to trust the key issuer. The drawback with CP-ABE system is, the users here can only use all possible combination of attributes that are organized logically as single set. This results in lacking of flexibility and fine-grained access. To overcome all these drawbacks and to achieve scalability, flexibility and fine grained access control; Zhinguo has projected a Hierarchical Attribute-Set-Based Encryption [HASBE] system [1]. In [5] and [6], a multi-authority system is presented in which each user has an ID and they can interact with each key generator (authority) using different pseudonyms. One user's different pseudonyms are tied to his private key, but key generators never know about the private keys, and thus they are not able to link multiple pseudonyms belonging to the same user. Also, the whole attributes set is divided into N disjoint sets and managed by N attributes authorities. In this setting, each authority knows only a part of any user's attributes, which are not enough to figure out the user's identity. However, the system projected by Chase et al. [6] considered the basic threshold-based KP-ABE, which lacks generality in the encryption policy expression. Many attribute based encryption systems having multiple authorities have been projected afterwards [7]–[10], but they either also employ a threshold-based ABE [7], or have a semi-honest central authority [8]–[10], or cannot tolerate arbitrarily many users' collusion attack [7]. The work by Lewko et al. [11] and Muller et al. [12] are the most similar ones to ours in that they also tried to decentralize the central authority in the CP-ABE into multiple ones. Lewko et al. use a LSSS matrix as an access structure, but their system only converts the AND, OR gates to the LSSS matrix, which limits their encryption policy to boolean formula, while we inherit the flexibility of the access tree having threshold gates. Muller et al. also supports only Disjunctive Normal Form (DNF) in their encryption policy. Besides the fact that we can express arbitrarily general encryption policy, our system also tolerates the compromise attack towards attributes authorities, which is not covered in many existing works. Recently, there also appeared traceable multi-authority ABE [13] and [14], which are on the opposite direction of ours. Those systems introduce accountability such that malicious users' keys can be traced. On the other hand, similar direction as ours can be found in [15] [17], who try to hide encryption policy in the cipher texts, but their solutions do not prevent the attribute disclosure in the key generation phase. To some extent, these three works and ours complement each other in the sense that the combination of these two types protection will lead to a completely anonymous ABE.

3 SYSTEM DESIGN

Here the objective is to attain a multi-authority CP-ABE which attains the security defined and guarantees the confidentiality of Data Consumers' identity information and tolerates compromise attacks on the authorities or the collusion attacks by the authorities. There are four types of entities: A) N Attribute Authorities: Cloud Server, Data Owners and Data Consumers. A user can be a Data Owner and a Data Consumer concurrently. Authorities are supposed to have dominant computation capabilities, and they are managed by government offices for the reason that some attributes partly contain users' individually identifiable information. The whole attribute set is divided into N disjoint sets and organized by each authority, therefore each authority is aware of only part of attributes.[6],[16] A Data Owner is the entity who wishes to outsource encrypted data file to the Cloud Servers. The Cloud Server, who is supposed to have sufficient storage capacity, does nothing but store them. Newly merged Data Consumers request private keys from all of the authorities, and they do not know which attributes are organized by which authorities. When the Data Consumers appeal their private keys from the authorities, authorities jointly create conforming private key and send it to them. All Data Consumers are capable to download any of the encrypted data files, but only those whose private keys satiate the privilege tree Tp can implement the operation associated with privilege p. The server is delegated to execute an operation p if and only if the user's identifications are proved through the privilege tree Tp.[8] B. Threats Model: We suppose that the Cloud Servers are semi-honest, who behave accurately in most of time but may conspire with mischievous Data Consumers or Data Owners to return others' file contents to gain illegal profits. But they are also presumed to gain legal

benefit when users' requests are appropriately processed, which means they will follow the protocol in general. N authorities are presumed to be un-trusted. That is, they will follow our projected protocol in general, but try to [9] find out as much information as possible discretely. More precisely, we assume they are concerned in users' attributes to achieve the identities, but they will not conspire with users or other authorities. This conjecture is similar to many prior researches on security issue in cloud computing and it is also reasonable since these authorities will be audited by government offices. Nevertheless, we will further relax this assumption and allow the complicity between the authorities. Data Consumers are untrusted since they are random users comprising attackers. They may connive with other Data Consumers to illegally access what they are not allowed to.[10] Besides, we do not consider the identity leakage from the underlying network since this can be slightly prohibited by retaining anonymized network protocols. The time complexity of the setup computation is O(N2) since every authority computes N − 1 pieces. This can be more reduced to O(N) by smearing the simple trick. We first group the authorities into C groups, and exchanges the parameters within the group only. Then, the time complexity is reduced to O(CN) = O(N) since C is a constant.

3.1 KEY GENERATION

This is executed by considering for each T_P, the algorithm first selects a polynomial q_x for each node x in it. For each node x, sets the degree dx of the polynomial q_x as one less than the threshold value k_x . Starting from the root node R_P, the algorithm arbitrarily picks $s_P \in Z_P$ and sets $q_{RP}(0) := s_P$ and arbitrarily selects other coefficients for q_{RP} . Then, for any other node x, the coefficients [11],[15]are chosen randomly and the constant term is set as $q_{parent}(x)(index(x))$ such that $q_x(0) = q_{parent}(x)(index(x))$ (index(x) is the index of the x's child nodes, and parent (x) is node x's parent node. Finally, picks a arbitrary element $h \in Z_P$ such that h−1 mod p exists, and calculates gh·s p, Dh−1, and the cipher text.

Algorithm 1 1-Out-of-2 Oblivious Transfer

1: Bob randomly picks a secret s and publishes g^s to Alice.
2: Alice creates an encryption/decryption key pair:$\{g^r, r\}$
3: Alice chooses i and calculates $EK_i = g^r$, $EK_{i-1} = \frac{g^s}{g^r}$ and sends EK_0 to Bob.
4: Bob calculates $EK_1 = \frac{g^s}{EK_0}$ and encrypts M_0 using EK_0 and M_1 using EK_1 and sends two cipher texts $E_{EK_0}(M_0), E_{EK_1}(M_1)$ to Alice.
5: Alice can use r to decrypt the desired cipher text $E_{EK_i}(M_i)$, but she cannot decrypt the other one. Meanwhile, Bob does not know which cipher text is decrypted.

Algorithm 2 1-Out-of-n Oblivious Transfer

1: Bob randomly picks n secrets $s_1, \ldots, s_n$ and calculates t_i as follows:

$$\forall i \in \{1, \ldots, n\} : t_i = s_1 \oplus \cdots \oplus s_{i-1} \oplus M_i$$

2: For each $i \in \{1, \ldots, n\}$, Bob and Alice are engaged in a 1-out-of-2 OT where Bob's first message is t_i and the second message is s_i. Alice picks t_i to receive if she wants M_i and s_i otherwise.
3: After Alice receives n components, she has $t_i = s_1 \oplus \cdots \oplus s_{i-1} \oplus M_i$ for the i she wants and s_k for $k \neq i$, she can recover the M_i by

$$M_i = t_i \oplus s_{i-1} \oplus s_{i-2} \oplus \cdots \oplus s_1$$

4 PROPOSED SYSTEM: ACHIEVING FULL ANONYMITY

It is assumed that semi-honest authorities in AnonyControl and they will not conspire with each other. This is a essential supposition in AnonyControl because each authority is in control of a subset of the whole attributes set, and for the attributes that it is in charge of, it knows the exact information of the key requester. If the information from all authorities is collected totally, the complete attribute set of the key requester is recovered and thus his identity is disclosed to the authorities. In this sense, AnonyControl is semi anonymous since partial identity information (represented as some attributes) is released to each authority, but we can achieve a full-anonymity and also permit the collusion of the authorities.[12] The key point of the identity information leakage we had in earlier system as well as every present attribute based encryption systems is that key generator (or attribute authorities in our system) issues attribute key based on the[13] stated attribute, and the generator has to know the user's attribute to do so. We need to host a new technique to let key generators issue the correct attribute key without knowing what attributes the users have. A naive solution is to give all the attribute keys of all the attributes to the key requester and let him pick whatsoever he wants. By this procedure, the key generator does not know which attribute keys the key requester chosen, but we have to completely trust the key requester that he will not pick any attribute key not allowed to him. [14]

5 PERFORMANCE EVALUATION

It is presented here regarding the performance evaluation based on the assessment about executing prototype system of AnonyControl-F. This is the first implementation of a multi-authority attribute based encryption system. This prototype system offers five command line tools.[12]
(i)anonycontrol-setup: Jointly generates a public key and N master keys.
(ii)anonycontrol-keygen: Generates a part of private key for

the attribute set it is responsible for.
(iii)anonycontrol-enc: Encrypts a file under r privilege trees.
(iv)anonycontrol-dec: Decrypts a file if possible.
(v)anonycontrol-rec: Decrypts a file and re-encrypts it under different privilege trees.[17]
This toolkit is based on the CP-ABE toolkit [4] which is available online and the whole system is implemented on a Linux system with Intel i7 2nd Gen @ 2.7GHz and 2GB RAM. It is furthermore employed three similar works under the same condition for the comparison purpose. Particularly, it is set only one privilege for the file access, and measured the time to create one privilege tree and calculate its verification parameter. In general, the computation overhead of is much higher than others because their system involves many more exponentiations and bilinear mappings due to the accountability [15],[18]. The encryption/decryption under different file sizes did not show big differences when file sizes are large (≥20MB), because the run times are dominated by the symmetric encryption (AES-256). Finally, only run times are plotted because the privilege creation is the Unique process in the system.

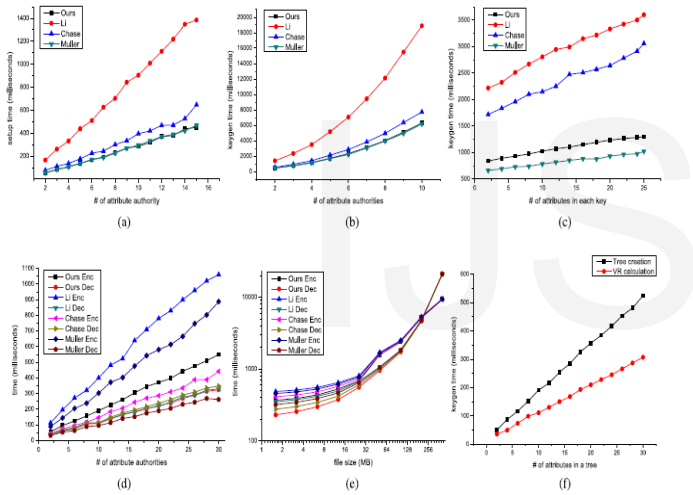

Fig. 2. Experiment result on our implemented prototype system. (a) Setup time. (b) Keygen time with different authorities' #. 20 attributes per key. (c) Keygen time with different attributes #. 4 authorities. (d) Encryption and decryption time with different attributes number. File size is 100KB. (e) Encryption and decryption time with different file size. 20 attributes in T0. (f) Time to create a privilege tree and decrypt a verification parameter from it.

6 CONCLUSION

This paper proposes a semi anonymous attribute -based privilege control scheme AnonyControl and a fully anonymous attribute based privilege control scheme AnonyControl F to address the user privacy problem in a cloud storage server. Using multiple authorities in the cloud computing system, our proposed schemes achieve not only fine-grained privilege control but also identity anonymity while conducting privilege control based on users' identity information. More importantly, our system can tolerate up to N − 2 authority compromise, which is highly preferable especially in Internet
-based cloud computing environment. We also conducted detailed security and performance analysis which shows that Anony-Control both secure and efficient for cloud storage system. The AnonyControl-F directly inherits the security of the AnonyControl and thus is equivalently secure as it, but extra communication overhead is incurred during the 1-out-of-n oblivious transfer.One of the promising future works is to introduce the efficient user revocation mechanism on top of our anonymous ABE. Supporting user revocation is an important issue in the real application, and this is a great challenge in the application of ABE schemes. Making our schemes compatible with existing ABE schemes [39]–[41] who support efficient user revocation is one of our future works.

REFERENCES

[1] A. Shamir, "Identity-based cryptosystems and signature systems," in Advances in Cryptology. Berlin, Germany: Springer-Verlag, 1985, pp. 47–53.
[2] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in Advances in Cryptology. Berlin, Germany: Springer-Verlag, 2005, pp. 457–473.
[3] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proc. 13th CCS, 2006, pp. 89–98.
[4] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute based encryption," in Proc. IEEE SP, May 2007, pp. 321–334.
[5] M. Chase, "Multi-authority attribute based encryption," in Theory of Cryptography. Berlin, Germany: Springer-Verlag, 2007, pp. 515–534.
[6] M. Chase and S. S. M. Chow, "Improving privacy and security in multi-authority attribute-based encryption," in Proc. 16th CCS, 2009, pp. 121–130.
[7] 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.
[8] V. Božovi´c, D. Socek, R. Steinwandt, and V. I. Villányi, "Multi-authority attribute-based encryption with honest-but-curious central authority," Int. J. Comput. Math., vol. 89, no. 3, pp. 268–283, 2012.
[9] F. Li, Y. Rahulamathavan, M. Rajarajan, and R. C.-W. Phan, "Low complexity multi-authority attribute based encryption system for mobile cloud computing," in Proc. IEEE 7th SOSE, Mar. 2013, pp. 573–577.
[10] K. Yang, X. Jia, K. Ren, and B. Zhang, "DAC-MACS: Effective data access control for multi-authority cloud storage systems," in Proc. IEEE INFOCOM, Apr. 2013, pp. 2895–2903.
[11] A. Lewko and B. Waters, "Decentralizing attribute-based encryption," in Advances in Cryptology. Berlin, Germany: Springer-Verlag, 2011, pp. 568–588.
[12] S. Müller, S. Katzenbeisser, and C. Eckert, "On multi-authority ciphertext-policy attribute-based encryption," Bull. Korean Math. Soc., vol. 46, no. 4, pp. 803–

819, 2009.
[13] J. Li, Q. Huang, X. Chen, S. S. Chow, D. S. Wong, and D. Xie, "Multiauthority ciphertext-policy attribute-based encryption with accountability," in Proc. 6th ASIACCS, 2011, pp. 386–390.
[14] H. Ma, G. Zeng, Z. Wang, and J. Xu, "Fully secure multi-authority attribute-based traitor tracing," J. Comput. Inf. Syst., vol. 9, no. 7, pp. 2793–2800, 2013.
[15] S. Hohenberger and B. Waters, "Attribute-based encryption with fast decryption," in Public-Key Cryptography. Berlin, Germany: Springer-Verlag, 2013, pp. 162–179.
[16] J. Hur, "Attribute-based secure data sharing with hidden policies in smart grid," IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 11, pp. 2171–2180, Nov. 2013.[17] Y. Zhang, X. Chen, J. Li, D. S. Wong, and H. Li, "Anonymous attribute based encryption supporting efficient decryption test," in Proc. 8th ASIACCS, 2013, pp. 511–516.

AUTHORS


[1] **Adapa Gopi** PG Scholar, Dept of CSE, Kasireddy Narayan Reddy College of Engineering & Research India.


[2] **P. Sathish Reddy** Associate professor and Head of Department In Department of CSE Kasireddy Narayan Reddy College of Engineering & Research India.


[3] **Venkatesh Guguloth**, Assoc Prof, Dept of CSE, Kasireddy Narayan Reddy College of Engineering & Research, Hayathnagar, RR(Dt), TS, India.