RESEARCH ARTICLE OPEN ACCESS

Privacy Preserving Two Layer Encryption Access Control In Public Clouds

A Reddy Prasad[1], Gudivada Lokesh[2], N Vikram[3]
MTech[1&3], Assistant Professor[2]
Department of Computer Science and Engineering,
Vemu College of Engineering and Technology,
Chittoor, Andhra Pradesh, India,

ABSTRACT
The motivation of this paper is to propose a secure Access control scheme, for public clouds. We proposed a Privacy Preserving Two layer Encryption Access control in public clouds, which provides more privacy and security compared to the traditional approaches. Current approaches to enforce ACPs on outsourced data using selective encryption require organizations to manage all keys and encryptions and upload the encrypted data to the remote storage. Such approaches incur high communication and computation cost to manage keys and encryptions whenever user credentials change. In this paper, we proposed a two layer encryption based approach to solve this problem by delegating as much of the access control enforcement responsibilities as possible to the Cloud while minimizing the information exposure risks due to colluding Users and Cloud.

Keywords: - Access Control, Anonymous Data, Cloud Computing, Privacy preserving, Two layer encryption.

I. INTRODUCTION

With the advent of technologies such as cloud computing, sharing data through a third-party cloud service provider has never been more economical and easier than now. However, such cloud providers cannot be trusted to protect the confidentiality of the data. In fact, data privacy and security issues have been major concerns for many organizations utilizing such services. Data often contains sensitive information and should be protected as mandated by various organizational policies and legal regulations. Encryption is a commonly adopted approach to assure data confidentiality. Encryption alone however is not sufficient as organizations often have also to enforce fine-grained access control on the data. Such control is often based on security-relevant properties of users, referred to as identity attributes, such as the roles of users in the organization, projects on which users are working, and so forth. These access control systems are referred to as attribute based access control (ABAC) systems [14]. Therefore, an important requirement is to support fine-grained access control, based on policies specified using identity attributes, over encrypted data with the involvement of the third-party cloud services, a crucial issue is that the identity attributes in the access control policies may reveal privacy-sensitive information about users and organizations and leak confidential information about the content. The confidentiality of the content and the privacy of the users are thus not assured if the identity attributes are not protected. It is well-known that privacy, both individual as well as organizational, is considered a key requirement in all solutions, including cloud services, for digital identity management [1]. Further, as insider threats are one of the major sources of data theft and privacy breaches, identity attributes must be strongly protected even from accesses within organizations. With initiatives such as cloud computing the scope of insider threats is no longer limited to the organizational perimeter. Therefore, protecting e identity attributes of the users while enforcing attribute-based access control both within the organization as well as in the cloud is crucial.

For example, let us consider a hospital that decides to use the cloud to manage their electronic health record (EHR)

system. Since EHRs are sensitive information, their confidentiality should be preserved from the cloud. Typical hospital stakeholders consist of employees playing different roles such as receptionist, cashier, doctor, nurse, pharmacist, system administrator, and so on. A cashier, for example, does not need have access to data in EHRs except the billing information in them while a doctor or a nurse does not need have access to billing information in EHRs. This requires the cloud based EHR system to support fine-grained access control. The typical identity attributes used by the stakeholders in our EHR system, such as role, location and position, can be used as good contextual information to connect with other publicly available information in order to learn sensitive information about individuals, leading to privacy violations. For example, if system administrators of the EHR system can see hospital employees' identity attributes, they can misuse the system to access EHRs and sell to outsiders without being caught. In order to address these issues, the cloud based EHR system should protect the identity attributes of users The goal of this article is to provide an overview of our approaches to enforce fine-grained access control on sensitive data stored in untrusted public clouds, while at the same assuring the confidentiality of the data from the cloud and preserving the privacy of users who are authorized to access the data. We compare these approaches and discuss about open issues.

II. RELATED WORK

Fine-grained access control (FGAC) allows one to enforce selective access to the content based on expressive policy specifications. Research in FGAC can be categorized into two dissemination models: *push-based* and *pull-based* models. Our work focuses on the pull-based model. In the push-based approaches [2], [3] subdocuments are encrypted with different keys, which are provided to users at the registration phase. The encrypted subdocuments are then broadcasted to all users. However, such approaches require that all [4] or some [3] keys be distributed in advance during user registration phase. This requirement makes it difficult to assure forward and backward key secrecy when user groups are dynamic. Further, the rekey process is not transparent, thus shifting the burden of acquiring new keys on users. Shang et al. [4] proposes approach to solve such problem. It lays the foundation to make rekey transparent to users and protect the privacy of the users who access the content. However, it does not support expressive access control policies as in our approach and also it is not directly applicable to pull based approaches. Under the pull-based model, the content publisher is required to be online in order to provide access to the content. Recent research efforts [10], [12], [5], [13] have proposed approaches to construct privacy preserving access control systems using a third-party storage service. In such approaches, the data owner has to enforce the ACPs and the privacy of the users from the content publisher is not protected. Further, in some approaches, multiple encryptions of the same document are required which is inefficient. A major drawback of all the above approaches is that they do not consider the management of encrypted data hosted in a third party when users are added or removed from the system or when the ACPs/subdocuments are updated. All the approaches require the data owner to handle encryption. Di Vimercati et al. [7] first identifies this problem and proposes an initial solution. While their solution improves over existing solutions, such solution does not support expressive attribute based policies and does not protect the privacy of the users.

The concept of attribute based encryption (ABE) has been introduced by Sahai and Waters [11]. The initial ABE system is limited only to threshold policies in which there are at least k out of n attributes common between the attributes used to encrypt the plaintext and the attributes users possess. Pirretti et al. [2] gave an implementation of such a threshold ABE system using a variant of the Sahai-Waters Large Universe construction [6]. Since this initial threshold scheme, a few variants have been introduced to provide more expressive ABE systems. Goyal et al. [8] introduced the idea of key policy ABE (KP-ABE) systems and Bethencourt et al. [9]

introduced the idea of cipher text-policy ABE (CP-ABE) systems. Even though these constructs are expressive and provably secure, they are not suitable for group management and especially in supporting forward security when a user leaves the group (i.e. attribute revocation) and in providing backward security when a new user joins the group. Some of the above schemes suggest using an expiration attribute along with other attributes. However, such a solution is not suitable for a dynamic group where joins and departures are frequent.

III. CLOUD ACCESS CONTROL CHARACTERISTICS

The identification and definition of Cloud access control characteristics and requirements, namely the access control policy, greatly amplifies the design of a model and the implementation of a mechanism regarding access control. In order to appoint a series of characteristics regarding access control we use the conceptual categorization for Cloud systems proposed in (Gouglidis and Mavridis, 2010). Figure 1 depicts the four layers of the conceptual categorization. The entropy layer identifies requirements from the dispersion of the objects in a system and the assets layer from the type of shared objects within the boundaries of the entropy layer. The management layer defines requirements from policy management and the logic layer incorporates requirements that are not handled by the former layers. A set of core requirements for access control systems that are considered important for the Cloud environment, follows. The identification of the requirements incorporates also characteristics that are exposed by the three levels of the information security infrastructure in the Cloud viz. application level, host level, and network level, where applicable. These characteristics may vary depending on the use cases that need to be supported by a specific system.

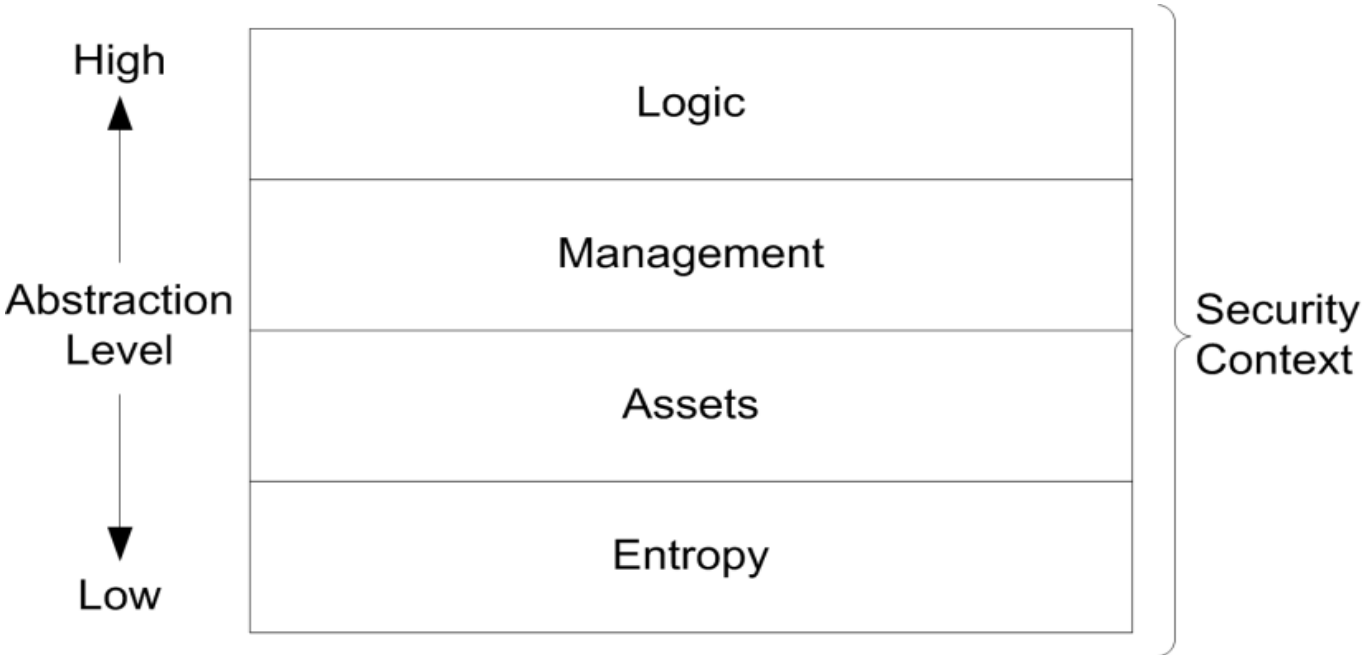

Figure 1. Conceptual categorization layer

.

Entropy layer: Applications are provided to consumers as a set of services via the SaaS service model. Each application is in most cases accessible through a web interface. Usually the services are deployed under the same organization and thus under the same domain. However, the use of the public or hybrid deployment models requires the collaboration of services among the participating organizations. Therefore, the application's entropy level can be relative high, depending on the used deployment model. Additionally, the hosts that are used to provide the assets of the Cloud can also be characterized by their high dispersion when the public or hybrid model is deployed and low when the private deployment model is applied.

Assets layer: The assets in Cloud computing systems are of two type viz. software and hardware. The software is exposed as a set of services that can be realized by technologies such as the web services. Collaboration among services is applicable. Hardware resources can be CPU, storage space, network bandwidth and so on. Specifically, we recognize that the fine-grained sharing of any resource in a Cloud system includes a resource requestor and a provider. When a user requests access to an asset, access must be granted only if the requestor is a legitimate user and also authorized to access the specified asset. Furthermore, as described in the definition of the Cloud, multi-tenancy must be supported. Thus, when multiple consumers from different companies are using a Cloud service model, consumers and their data must be guaranteed that are protected from each other throughout the collaboration.

Management layer: The management of policies in a Cloud computing system is required to be centralized in most cases. However, if collaboration is required as in the public and hybrid Cloud deployment models, the management of policies requires be distributing and applying among participating organizations. Moreover, each administrative user of an organization should administer the local policies of the organization. Additionally, administrators should run the policies in the collaboration that refer to resources of the administrator's organization. Furthermore, it must be guaranteed that no conflicts should exist among the policies of the individual organizations at the higher corporate level. Last but not least, the process of identifying policy violations should be automated, in all deployment models.

Logic layer: The main characteristic of the Cloud is the support of the business model that allows the provision of usage based pricing. Thus, quality of service policies (QoS) along with service level agreements (SLAs) must be supported, in order to provide to the consumers the agreed levels of quality. Resource providers should be able to define quality factors on their shareable resources. The quality factors concern the level of resource usage and can also be characterized as obligations that must be met from a provider when granting access to a resource requestor. For instance, quality factors could apply for setting disk quotas, memory or CPU utilization levels and so on and so forth. Furthermore, we identify the enforcement of the autonomy and security principle (Shafiq et al., 2005). The autonomy principle refers to the permission of an access under secure interoperation, if it is also permitted within the individual domain. The security principle pertains to the denial of an access under secure interoperation, if it is also denied within the individual domain. Furthermore, the principle of containment (Ravi Sandhu, 2008) that subsumes the principles of the separation of duties, least privilege and so forth, should be supported in each and among domains. The latter requirement greatly enhances the adoption of Cloud technologies in business organizations, where the existence of conflict of interest policies is presumed.

IV. PROPOSED SYSTEM

Our basic approach follows the conventional data outsourcing scenario where the Owner enforces all the access control policies through selective encryption and uploads encrypted

data to the untrusted Cloud. We refer to this approach as single layer encryption (SLE). The SLE approach supports fine-grained attribute-based access control policies and preserves the privacy of users from the Cloud. However, in such an approach, the Owner is in charge of encrypting the data before uploading it to the third-party server as well as re-encrypting the data whenever user credentials or authorization policies change and managing the encryption keys. The Owner has to download all affected data before performing the selective encryption.

The Owner thus incurs high communication and computation costs, which then negate the benefits of using a third party service. A better approach should delegate the enforcement of fine-grained access control to the Cloud, so to minimize the overhead at the Owner, whereas at the same time assuring data confidentiality from the third-party server. In this section, we provide an overview of an approach, based on two layers of encryption, which addresses such requirement. Under such approach, referred to as two-layer encryption (TLE), the Owner performs a coarse grained encryption, whereas the Cloud performs a fine grained encryption on top of the data encrypted by the coarse grained encryption. A challenging issue in this approach is how to decompose the ABAC policies such that the two-layer encryption can be performed. In order to delegate as much access control enforcement as possible to the Cloud, one needs to decompose the ABAC policies so that the Owner only needs to manage the minimum number of attribute conditions in these policies that assures the confidentiality of data from the Cloud. Each policy should be decomposed into two sub policies such that the conjunctions of the two sub policies result in the original policy. The two-layer encryption should be performed such that the Owner first encrypts the data based on one set of sub policies and the Cloud re-encrypts the encrypted data using the other set of policies. The two encryptions together enforce the original policies as users should perform two decryptions in order to access the data. For example, consider the policy $(C1 \wedge C2) \vee (C1 \wedge C3)$. This policy can be decomposed as two sub policies C1 and $C2 \vee C3$. Notice that the decomposition is consistent; that is, $(C1 \wedge C2) \vee (C1 \wedge C3) = C1 \wedge (C2 \vee C3)$. The Owner enforces the former by encrypting the data for the users satisfying the former and the Cloud enforces the latter by re-encrypting the Owner encrypted data for the users satisfying the latter. Since the Cloud does not handle C1, it cannot decrypt the Owner encrypted data and thus confidentiality is preserved. Notice that users should satisfy the original policy to access the data by performing two decryptions. An analysis of this approach suggests that the problem of decomposing for coarse and fine grained encryption while assuring the confidentiality of data from the third party and the two encryptions together enforcing the policies is NP-complete.

We have thus investigated optimization algorithms to construct near optimal solutions to this problem. Under our TLE approach, the third party server supports two services: the storage service, which stores encrypted data, and the access control service, which performs the fine grained encryption. As shown in Figure 4, we utilize the same AB-GKM scheme that allows users whose attributes satisfy a certain policy to derive the group key and decrypt the content they are allowed to access from the Cloud. Our proposed approach assures the confidentiality of the data and preserves the privacy of users from the access control service as well as the cloud storage service while delegating as much of the access control enforcement as possible to the third party through the two- layer encryption technique. The TLE approach has many advantages. When the policy or user dynamics changes, only the outer layer of the encryption needs to be updated. Since the outer layer encryption is performed at the third party, no data transmission is required between the Owner and the third party. Further, both the Owner and the third party service utilize the AB-GKM scheme for key management whereby the actual keys do not need to be distributed to the users. Instead, users are given one

or more secrets which allow them to derive the actual symmetric keys for decrypting the data.

V. CONCLUSION

Current approaches to enforce ACPs on outsourced data using selective encryption require organizations to manage all keys and encryptions and upload the encrypted data to the remote storage. Such approaches incur high communication and computation cost to manage keys and encryptions whenever user credentials change. In this paper, we proposed a two layer encryption based approach to solve this problem by delegating as much of the access control enforcement responsibilities as possible to the Cloud while minimizing the information exposure risks due to colluding Users and Cloud. A key problem in this regard 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 problem is NP-Complete and provided approximation algorithms. Based on the decomposed ACPs, we proposed a novel approach to privacy preserving fine-grained delegated access control to data in public clouds. Our approach is based on a privacy preserving attribute based key management scheme that protects the privacy of users while enforcing attribute based ACPs. As the experimental results show, decomposing the ACPs and utilizing the two layer of encryption reduce the overhead at the Owner. As 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] G.Ateniese, R. Burns, R.urtmola, J.Herring, L. Kissner, Z. Peterson, and D.Song, "Deduplication in cloud storage using side channels in cloud services," Oct 2008.

[2] D. Boneh, X. Boyen, and E. Goh, "Hierarchical Identity Based Encryption with Constant Size Ciphertext," Proc. Ann. Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT), pp. 440-456, 2005.

[3] K. D. Bowers, A. Juels, and A. Oprea, "Hail: A high-availability and integrity layer for cloud storage," in Proc. Of CCS'09, 2009, pp. 187-198.

[4] A. Fiat and M. Naor, "Broadcast Encryption," Proc. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 480-491, 1993.

[5] S. Kamara and K. Lauter, "Cryptographic Cloud Storage," Proc. Int"l Conf. Financial Cryptography and Data Security (FC), pp. 136-149, Jan. 2010.

[6] E. Goh, H. Shacham, N. Modadugu, and D. Boneh, "Sirius: Securing Remote Untrusted Storage," Proc. Network and Distributed Systems Security Symp. (NDSS), pp. 131-145, 2003.

[7]] A. Fiat and M. Naor, "Broadcast Encryption," Proc. Int"lCryptology Conf. Advances in Cryptology (CRYPTO), pp. 480-491, 1993.

[8] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," Comm. ACM, vol. 53, no. 4, pp. 50-58, Apr. 2010.

[9] X.Liu,B.Wang,Y.Zhang, and J.Yan,"Mona: Secure Multi-Owner Data Sharing for Dynamic Groups in the Cloud,"IEEE Computer Society,vol. 24,no. 6,June. 2013.

[10] D. Boneh, X. Boyen, and E. Goh, "Hierarchical Identity Based Encryption with Constant Size Ciphertext," Proc. Ann. Int"l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT), pp. 440-456, 2005.

[11] B. Waters, "Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization," Proc. Int"l Conf. Practice and Theory in Public Key Cryptography Conf. Public KeyCryptography, http://eprint.iacr.org/2008/290.pdf, 2008.

[12] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing," Proc. IEEE INFOCOM, pp. 534-542, 2010.

[13] First Author (Year), "Manuscript Title", *Proceedings / Conference Name*, Vol. 1, No. 1, Pp. 10–15.

[14] D.A. First Author & B.S. Second Author (Year), "Book Title", Chapter No. (If any), Editors: First Editor Name & Second Editor Name, *Publisher Name*, Edition, Press, Place, Pp. 10-15.

Authors

P Reddy Prasad Pursuing M.Tech., CSE (II Year), Vemu Institute of Technology, Chittoor, Andhra Pradesh. Completed M C A from Sri Vidyanikethan Engineering College in 2009, persuing M.Tech CSE in VEMU, Chittoor, after MCA having 3 Years teaching Experience in Vemu College of Engineering & Technology, Chittoor . Areas of Interest Cloud Computing , Cryptography, Algorithms Design and Analysis Process . Attended one National Conference at SVCET Chittoor in the topic of Recent Trends in Computing.

Gudivada Lokesh, Asst. Prof., Vemu College of Engineering and Technology, Chittoor. B.Tech from JNTU in 2010, M.E from SATYABHAMA in 2012. Having 2 Years of experience in teaching in Vemu from 2011 to till date, Area of Interesting Data Mining, Cloud Computing, Big Data,Computer Networks. Attended 2 National & International Conferences, 2 journals.

Vikram Neerugatti Pursuing M.Tech., CSE (II Year), Global College of Engineering and Technology, Kadapa. B.Tech from JNTU in 2009, M.S from BRAINWELLS UNIVERSITY, UK. in 2010. MSc Psychology from SVU Thirupathi. Having 4 Years of experience in teaching in SVCE from 2010 to till date, Area of Interesting Data Mining, Computer Networks, Android Operating systems. Attended 3 National & International Conferences. 3 international Journals, attended 4 workshops, organized 3 workshops. Guided 5 UG level projects and 3 PG level projects.