An Efficient Bloom Filter to Evaluate the Access Policy and Locate an Attribute

[1*] Gujjula Ramireddy, [2] BSS Mounica(M.Tech Asst Prof)
[1,2]Dept of CSE, Eluru College of Engineering and Technology,
Duggirala(V),pedavegi(M),EULRU, Andhra Pradesh

ABSTRACT

How to control the access of the huge amount of big data becomes a very challenging issue, especially when big data are stored in the cloud. Ciphertext-Policy Attribute based Encryption (CP-ABE)[1] is a promising encryption technique that enables end-users to encrypt their data under the access policies defined over some attributes of data consumers and only allows data consumers whose attributes satisfy the access policies to decrypt the data. In CP-ABE [2], the access policy is attached to the ciphertext in plaintext form, which may also leak some private information about end-users. Existing methods only partially hide the attribute values in the access policies, while the attribute names are still unprotected. In this paper, we propose an efficient and fine-grained big data access control scheme with privacy-preserving policy. Specifically, we hide the whole attribute (rather than only its values) in the access policies. To assist data decryption, we also design a novel Attribute Bloom Filter to evaluate whether an attribute is in the access policy and locate the exact position in the access policy if it is in the access policy. Security analysis and performance evaluation show that our scheme can preserve the privacy from any LSSS access policy without employing much overhead.

INTRODUCTION

In the era of big data, a huge amount of data can be generated quickly from various sources (e.g., smart phones, sensors, machines, social networks, etc.)[4]. Towards these big data, conventional computer systems are not competent to store and process these data. Due to the flexible and elastic computing resources, cloud computing is a natural fit for storing and processing big data. With cloud computing, end-users store their data into the cloud, and rely on the cloud server to share their data to other users (data consumers) [5]. In order to only share end-users’ data to authorized users, it is necessary to design access control mechanisms according to the requirements of end-users.

When outsourcing data into the cloud, end-users lose the physical control of their data. Moreover, cloud service providers are not fully-trusted by end-users, which makes the

Access control more challenging. For example, if the traditional access control mechanisms (e.g., Access Control Lists) are

Applied, the cloud server becomes the judge to evaluate the access policy and make access decision. Thus, end-users may worry that the cloud server may make wrong access decision intentionally or unintentionally, and disclose their data to some unauthorized users. In order to enable end-users to control the access of their own data, some attribute-based access control

Schemes are proposed by leveraging attribute-based encryption. In attribute-based access control, end-users first define access policies for their data and encrypt the data [9].Under these access policies. Only the users whose attributes can satisfy the access policy are eligible to decrypt the data.

EXISTING SYSTEM

In order to enable end-users to control the access of their own data stored on untrusted remote servers (e.g., cloud servers), encryption-based access control [5] is an effective method, where data are encrypted by end-users and only authorized users are given decryption keys. This can also prevent the data security during the transmission over wireless networks which are vulnerable to many threats. However, traditional public key encryption methods are not suitable for data encryption because it may produce multiple copies of ciphertext for the same data when there are many data consumers in the system. In order to cope with this issue, some attribute-based access control schemes are proposed by leveraging attribute-based encryption which only produces one copy of ciphertext for each data and does not need to know how many intended data consumers during the data encryption. Moreover, once the cloud data are encrypted, some searchable encryption algorithms are proposed to support search on encrypted cloud data.

DISADVANTAGES OF EXISTING SYSTEM:

- Existing methods which only partially hide the attribute values in the access policies
- End-users may worry that the cloud server may make wrong access decision intentionally or unintentionally, and disclose their data to some unauthorized users.
- Attribute-based access control schemes can deal with the attribute revocation problem, they all suffer from one problem: the access policy may leak privacy. This is because the access policy is associated with the encrypted data in plaintext form.

PROPOSED SYSTEM:

- We propose an efficient and fine-gained big data access control scheme with privacy-preserving policy, where the whole attributes are hidden in the access policy rather than only the values of the attributes.

- We also design a novel Attribute Bloom Filter to evaluate whether an attribute is in the access policy and locate the exact position in the access policy if it is in the access policy.

- We further give the security proof and performance evaluation of our proposed scheme, which demonstrate that our scheme can preserve the privacy from any LSSS access policy without employing much overhead.

ADVANTAGES OF PROPOSED SYSTEM:

- We have proposed an efficient and fine-grained data access control scheme for big data, where the access policy will not leak any privacy information.
- our method can hide the whole attribute (rather than only its values) in the access policies
- Our scheme is selectively secure against chosen plaintext attacks.

IMPLEMENTATION

MODULES

1. Cloud Servers
2. Attribute Authority
3. End-users,
4. Data Consumers

MODULES DESCRIPTION

Cloud Servers Cloud Servers are employed to store, share and process big data in the system. The cloud servers are managed by cloud service providers, who are not in the same trust domain as end-users. Thus, cloud servers cannot be trusted by end-users to enforce the access policy and make access decisions. We also assume that the cloud server cannot collude with any End-users or Data Consumers.

Attribute Authority The attribute authority manages all the attributes in the system and assigns attributes chosen from the attribute space to end-users. It is also a key generation center, where the public parameters are generated. It also grants different access privileges to end-users by issuing secret

keys according to their attributes. The attribute authority is assumed to be fully trusted in the system.

End-user End-users are the data owners/producers who outsource their data into the cloud. They also would like to control the access of their data by encrypting the data with CP-ABE. End-users are assumed to be honest in the system.

Data Consumers Data consumers request the data from cloud servers. Only when their attributes can satisfy the access policies of the data, data consumers can decrypt the data. However, data consumers may try to collude together to access some data that are not accessible individually.

SCREENS

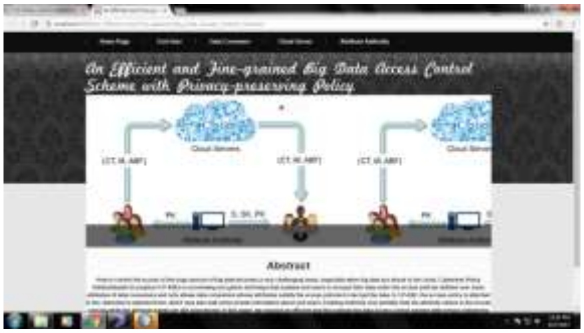
An Efficient and Fine-grained Big Data Access Control Scheme with Privacy-preserving Policy
Cloud Servers
Abstract

Fig: Home Page

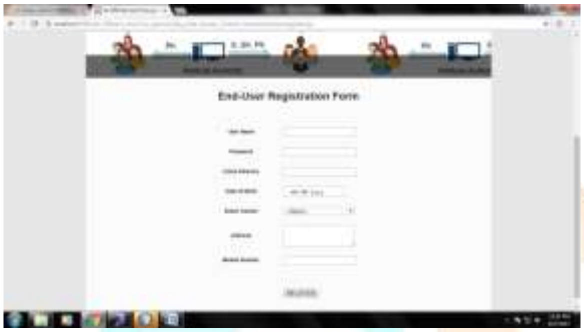

Fig: End User Registration

An Efficient and Fine-grained Big Data Access Control Scheme with Privacy-preserving Policy
Cloud Servers
Attribute Authority
Abstract

Fig: End User Home

An Efficient and Fine-grained Big Data Access Control Scheme with Privacy-preserving Policy
Cloud Servers
Abstract

Fig: Data Consumer Home

An Efficient and Fine-grained Big Data Access Control Scheme with Privacy-preserving Policy
Cloud Servers
Abstract

Fig: Cloud Server Home

CONCLUSION

We have proposed an efficient and fine-grained data access control scheme for big data, where the access policy will not leak any privacy information. Different from the existing methods which only partially hide the attribute values in the access policies, our method can hide the whole attribute (rather than only its values) in the access policies. However, this may lead to great challenges and difficulties for legal data consumers to decrypt data. To cope with this problem, we have also designed an attribute localization algorithm to evaluate whether an attribute is in the access policy. In order to improve the efficiency, a novel Attribute Bloom Filter has been designed to locate the precise row numbers of attributes in the access matrix. We have also demonstrated that our scheme is selectively secure against chosen plaintext attacks. Moreover, we have implemented the ABF by using Murmur Hash and the access control scheme to show that our scheme can preserve the privacy from any LSSS access policy without employing much overhead. In our future work, we will focus on how to deal with the offline attribute guessing attack that check the guessing “attribute strings” by continually querying the ABF.

REFERENCES

[1] C. Dong, L. Chen, and Z. Wen, “When private set intersection meets big data: an efficient and scalable protocol,” in Proc. of CCS’13. ACM, 2013, pp. 789–800.

[2] K. Yang, X. Jia, and K. Ren, “Secure and verifiable policy update outsourcing for big data access control in the cloud,” IEEE Trans.

[3] 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, 2013.

[4] P. Mell and T. Grance, “The NIST definition of cloud computing,” [Recommendationsof the National Institute of Standards and Technology-Special Publication 800-145], 2011.

[5] R. Lu, H. Zhu, X. Liu, J. K. Liu, and J. Shao, “Toward efficient andprivacy-preserving computing in big data era,” IEEE Network, vol. 28,no. 4, pp. 46–50, 2014.

[6] K. Yang and X. Jia, “Expressive, efficient, and revocable data accesscontrol for multi-authority cloud storage,” IEEE Trans. Parallel Distrib.Syst., vol. 25, no. 7, pp. 1735–1744, July 2014.

[7] H. Li, D. Liu, K. Alharbi, S. Zhang, and X. Lin, “Enabling fine-grainedaccess control with efficient attribute revocation and policy updatingin smart grid,” KSII Transactions on Internet and Information Systems(TIIS), vol. 9, no. 4, pp. 1404–1423, 2015.

[8] K. Yang, Z. Liu, X. Jia, and X. S. Shen, “Time-domain attribute-basedaccess control for cloud-based video content sharing: A cryptographicapproach,” IEEE Trans. on Multimedia (to appear), February 2016.

[9] B. Waters, “Ciphertext-policy attribute-based encryption: An expressive,efficient, and provably secure realization,” in Proc. of PKC’11. Berlin,Heidelberg: Springer-Verlag, 2011, pp. 53–70.

.