Scientific Journal Impact Factor (SJIF): 3.518

International Journal of Modern Trends in Engineering and Research

ISSN (ONLINE) : 2349 - 9745 ISSN (PRINT) : 2393 - 8161

Improved Server Storage Security Using Mobiflage with Verifiable Outsourced Decryption

Saritha Xavier[1]And Soumi C.G[2]

[1,2]Department of Computer Science, IIET,Nellikuzhi

Abstract-Attribute-based encryption (ABE) is a new vision for public key encryption that allows users to encrypt and decrypt messages based on user attributes. An ABE scheme along with outsourced decryption is a better choice since it allows a third party to transform an ABE ciphertext into a short ciphertext using a public transformation key. Even this improved version of ABE does not guarantee the correctness of the transformation. In the proposed system, a security model is described which includes ABE with verifiable outsourced decryption with a highly improved server storage security through Mobiflage method. Mobiflage is a method of developing hidden volumes in the space within server storage. That is, an additional file system is created at an offset within the server storage and encrypted. With this method, the adversary can examine the storage but will not find any record of the hidden files, apps, or activities. Thus the storage security which is highly required in the case of a distributed environment can be maintained effectively to create a secure infrastructure.

Keywords-Access control, Outsourced Decryption, Verification, Confidentiality.

I. INTRODUCTION

In earlier days, the servers provide access control based on the assumption that, the data servers can be trusted in order to keep data confidential and provide access control correctly. But today this assumption is no longer true, because services are increasing today and they store their data across many servers and many data owners are sharing those data. An example for this is cloud data storage where the cloud service providers are not in same trusted as in the case of end users. The cloud service providers are in different domain and the hardware platforms are not in direct control of the data owners. To reduce the privacy concerns of user about their data, a common solution is to encrypt the data and store the encrypted form of data. Thus the data will remain private even if the data server or data owner are not trusted or compromised. The ciphertext or encrypted data is suitable for sharing and access control.In older days, encryption can be viewed as a method for one user to encrypt data to another target party, so that the target recipient can decrypt data and read the message. But in some application, users wish to encrypt data based on certain policies as opposed to specified set of users. So for encrypting the data, the user need a mechanism that allow user to look all parties that have the access structure or attributes that match the user policy for decryption. Therefore data encryption using symmetric or public key encryption is not a better solution for providing scalable access control.

To address these issues a new concept of encryption is proposed by Sahai and Waters [2]. Attribute based encryption is a one type of data encryption standard in which the user secret key and the ciphertext are dependent upon the attributes. In such a standard the decryption of the ciphertext is possible if and only if the attributes specified in the user secret key matches the attributes of ciphertext. Attribute Based Encryption scheme have two categories: Ciphertext PolicyAttribute Based Encryption (CP-ABE) and Key Policy Attribute Based Encryption (KP-ABE). These two categorization are based on whether the access policy are embedded in ciphertext or user private key.

In a CP-ABE scheme [6] each cipher text is associated with an access policy according to the attributes. The user's private key is also associated with a set of attributes. The user is able to decrypt a cipher textif and only if the set of attributes associated with the user's private key satisfies the access policy, which in turn associated with set of attributes. In KP-ABE scheme [3] each and every ciphertext is associated with a set of attributes. The every user's private key is associated with an access policy according to the attributes. A user is able to decrypt a ciphertext if and only if the set of attributes associated with the ciphertext satisfies the access policy associated with the user's private key, which in turn associated with set of attributes. Both CP-ABE and KP-ABE scheme can prevent any unauthorized users from accessing data.

The most existing Attribute Based Encryption scheme has a drawback which is related to ciphertext size. The size of attributes is directly proportional to the size of attributes. So, as the size of attributes the ciphertext size increases which in turn increases the decryption cost. This is drawback for resource limited devices such as mobile phones etc. And also the existing implementation of ABE scheme require more pairing groups and more pairing operations for decryption which is more expensive than the exponentiations. And also existing ABE scheme does not guarantee the correctness of the transformation and it does not provide storage security. In this proposed framework include verification of the transformation and provide improved storage security which is a crucial factor for distributed system.

II. RELATED WORKS

There are works that are closely related to the system. Some of the recent works [8] for verification for outsourced decryption are used to construct ABE with verifiable outsourced decryption. But today they are impractical for the ABE approach. And the solution are mainly based on Gentry's fully homomorphic encryption system [9] and [8] also one bootstrapping operation of homomorphic encryption takes 30 minutes for security. Since one operation would count for a small constant number of gates in the overall computation, this would be repeated many times toevaluateanABEdecryption. The solutions in [10] allow a client to outsource the pairing operations to a server.And these are technique of outsourcing the pairings. The solutions in the system requiretheclienttocomputemultipleexponentiationsin thetargetgroupforeverypairing operation. These exponentiation operations are more expensive and workload will be increased.

Recently, Lai et al. [5] implemented a security model for checking the modification in an outsourced ABE system that is propose a verification scheme and a concrete construction with verifiable outsourced decryption. In their construction they appends a redundant ciphertext of a random message and a tag which is computed from the real message and the random message to each ciphertext, and provide the original untransformed ciphertext as an auxiliary input in the final decryption step by the user. And the scheme produced by Lai et al. [5] introduces significant overhead in both ciphertext size and decryption operation.And also Green et al. [4] proposed a simple method to adapt their RCCA (replayable chosen-ciphertext attack) systems in the ABE scheme for providing verifiable outsourced decryption. In this method they append hash value of some randomness of ciphertext. In this method the user is not able to detect the dishonest character of the server in the transformation process. And it does not provide verifiability in the schema and the system works only in the heuristic model that is random oracle model. And this method is infeasible and does not guarantee the recovery of ciphertext in the randomness.

III.PROPOSED WORK

The proposed system, include efficient method to verify the correctness of the transformation done by the server. Instead of using session key, the system first compress the message into shorter one using hash function. Then this hash value is used for checking the correctness of the transformation. The correctness ensures the integrity of the recovered message. The verification key resists the modification if any occur in the transformation and provide as an auxiliary input for decryption of message.

Also present an efficient method to improve the server storage security. The server contains all the details of different users and also the messages that are transmitted between the users.To provide more security for those data and the access policy according to the attributes from the adversary, we need to provide a new scheme called Mobiflage in which an additional file system is created at an offset. From the new file, if anyone accesses the server, but he is unable to see the files included in the new file.

In following subsections we will discuss in detail the Attribute Based Encryption with verifiable outsourced decryption and also Mobiflage in which a new file is created at an offset which is obtained by the server.

A. Attribute Based Encryption with Verifiable Outsourced Decryption

Attribute Based Encryption is public key encryption in which encryption and decryption of data takes place on the basis of attributes. Our work mainly focuses on Cipher text Policy Attribute Based Encryption (CP-ABE) [6]. In the CP-ABE the cipher text is associated with access structure. The access structure is based on set of attributes. An access structure can be defined as collection of elements or parties. Here we represent parties as attributes. So access structure contains complete collection of attributes. The attributes in the access structure is authorized attributes. And the attributes not defined in the access structure are called unauthorized attributes. The access structure we defined is a monotone access structure.

To eliminate the decryption overhead of the user the system allow dividing the user private key as transformation key and the elgamal secret key. The transformation key is shared with third person or a proxy server. This proxy server will decrypt the original cipher text and produce a simple cipher text using the transformation key. This is termed as outsourced decryption. Then the simple ciphertext can be decrypted with minimal overhead than the original one using the user secret key. But we cannot guarantee that the outsourced decryption is a perfect one or correct. So we use verification method in order to ensure the correctness of the transformation done by the proxy server. We cannot trust the proxy server. If any modification is happened to the data it can be determined by the verification scheme. If any person view or make changes in the message, then the verification key or the second hash value will be different from the first hash value.

The algorithms used for the construction of the Attribute Based Encryption with Verifiable Outsourced Decryption (VO-ABE) are:

Setup: The Setup algorithm takes input as attribute collection and the security parameter. The Setup algorithm produce output as master public key and the master secret key. The master secret key is used for generation of user secret key. The master public key defines the message space.

Encrypt: The Encrypt algorithm takes input as master public key, the message that we are sending which is element of message space, and the input to encryption algorithm. The Encrypt algorithm produce output as ciphertext that is encrypted form of message and the verification key which represents the first hash value.

KeyGen: The KeyGen algorithm takes input as the master secret key and the input for key generation algorithm. The algorithm produce output as transformation key which is used for transformation and the decryption key or user secret key which is used for decryption.

Transform: The Transform algorithm takes input as the transformation key and the ciphertext which is generated in the Encrypt algorithm. The Transform algorithm produces simple ciphertext which can decrypt by user with minimum overhead.

Decrypt: The Decrypt algorithm takes input as user secret key and the simple ciphertext which is generated from Transform algorithm. It produces the original message and also generates second hash value.

The hash values are compared after decryption. If it matches then the message is secured and if it is not matches then the message is corrupted and the user can request for a message again.

B. MOBIFLAGE

The main purpose of Mobiflage is to improve the server storage security by hiding the data in some other file within the storage. This technique [7] is mainly used for resource limited devices such as mobile phones etc. Even though the data confidentiality can be improved by encryption, in some applications or scenario this is inadequate. As users provide their keys to some others the sensitive data become viral and it reduces the confidentiality of data. This is because users are forced to disclose their decryption keys or private keys. Mobiflage is a method of encrypting a file and store the encrypted file in some offset which is determined by the server in another encrypted. Thus the server security can be increased. If there is any chance for further attacks and break the Attribute Based Encryption, the attacker can view only the file. He doesn't know about the hidden file that is located in particular location.

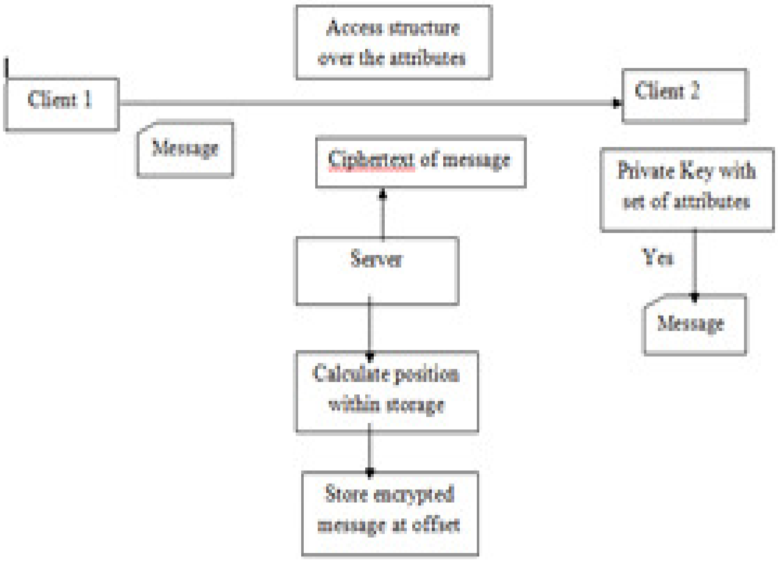
Access structure over the attributes
Client 1
Client 2
Message
Ciphertext of message
Private Key with set of attributes
Server
Yes
Message
Calculate position within storage
Store encrypted message at offset

Figure 1:The block diagram of system

Figure 1 shows the basic flow of the system. When client1 send message to client2 the server first encrypt the message with the access structure which is associated with universe of attributes.The attributes are collected from the client during registration. Then invoke the cp-abe module for further operation. That is, for generation of master secret key, master public key, transformation key and the user private key. Then server encrypt message using cpabe encrypt algorithm. The server also stores the message in the storage system. At the receiver side the client2 decrypt if and only if set of attributes in his private. If it matches then he can decrypt and view the message. In order to protect the data from any adversary, the server calculates a position inside the encrypted file and inserts the encrypted message into the file. Thus server can protect the sensitive messages from any adversary by hiding the volume within the storage system.

IV.RESULTS

The proposed system use verification technique to ensure the data confidentiality during transmission. Verification scheme guarantees the correctness of transmission and also it resists any modification. Experiments show that if any adversary attacks the encrypted during transformation, the verification key or the hash value during decryption will be different from the first hash value. If there is no change happen in the data then the hash value matches the first hash value and the message is secured. And also the method is nearly optimal and it brings minimum overhead during the verification process.

Also introduce a method Mobiflage which will improve the server storage security. By using Mobiflage we can hide the data in the position within storage area. When one client sends some message to another, the server encrypts data using Attribute Based Encryption and stores the data in server side. In order to provide more security to those we first calculate a position in the rand0m volume and provide the encrypted data to these position. If anyone attacks the server storage, the adversary is unknown about the existence and location of hidden volume. Thus data become more secure from attacker using Mobiflage.

V.CONCLUSION

In several distributed system or applications, the user is able to access data if the user possesses some set of attributes or credentials. We present an efficient system that enforce complex access control over the ciphertext or encrypted data where we can maintain the data confidentiality. We present a verification scheme which will resist modification to encrypted data. The verification key guarantees the correctness of the transformation. Also introduce an effective method Mobiflage to improve the server storage security. With Mobiflage we can hide the encrypted volume within the storage system. Thus the data in the server side can be protectedfrom the attacker because the encrypted data is inside another encrypted volume.

REFERENCES

[1] Baodong Qin, Robert H. Deng, Shengli Liu, and Siqi Ma,"Attribute-Based Encryption With Efficient Verifiable Outsourced Decryption"IEEE Transactions on Information Forensics And Security, Vol. 10, No. 7, July 2015

[2] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in Advances in Cryptology (Lecture Notes in Computer Science), vol. 3494, R. Cramer, Ed. 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. ACM Conf. Comput. Commun. Secur., 2006, pp. 89–98.

[4] M. Green, S. Hohenberger, and B. Waters, "Outsourcing the decryption of ABE ciphertexts," in Proc. 20th USENIX Secur. Symp., 2011, p. 34.

[5] J. Lai, R. H. Deng, C. Guan, and J. Weng, "Attribute-based encryption with verifiable outsourced decryption," IEEE Trans. Inf. Forensics Security, vol. 8, no. 8, pp. 1343–1354, Aug. 2013.

[6] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute- based encryption," in Proc. IEEE Symp. Secur. Privacy, May 2007, pp. 321–334

[7] Adam Skillen and Mohammad Mannan, "Mobiflage: Deniable Storage Encryption for Mobile Devices," IEEE Transactions on Dependable And Secure Computing, Vol. 11, NO. 3, May-June 2014

[8] B. Chevallier-Mames, J.-S. Coron, N. McCullagh, D. Naccache, and M. Scott, "Secure delegation of elliptic-curve pairing," in Smart Card Research and Advanced Application (Lecture Notes in Computer Science), vol. 6035, D. Gollmann, J.-L. Lanet, and J. Iguchi-Cartigny, Eds. Berlin, Germany: pringer-Verlag, 2010, pp. 24–35.

[9] C. Gentry, "Fully homomorphic encryption using ideal lattices," in Proc. STOC, 2009, pp. 169–178.

[10]R. Gennaro, C. Gentry, and B. Parno, "Non-interactive verifiable computing: Outsourcing computation to untrusted workers," in Advances in Cryptology (Lecture Notes in Computer Science), vol. 6223, T. Rabin, Ed. Berlin, Germany: Springer-Verlag, 2010, pp. 465–482.