Deniable Attribute-based Encryption using Audit-Free Cloud Storage

R.V.L.S.N Sastry[1], Dr.B. Giridhar[2], N.Viswanatha Reddy[3]
[1,2]Associate Professor,[3]PG Scholar, Dept of CSE, Sri Venkateswara College of Engineering and Technoology, Srikakullam

Abstract:Cloud storage services have become increasingly popular. Because of the importance of privacy, many cloud storage encryption schemes have been proposed to protect data from those who do not have access. All such schemes assumed that cloud storage providers are safe and cannot be hacked; however, in practice, some authorities (i.e., coercers) may force cloud storage providers to reveal user secrets or confidential data on the cloud, thus altogether circumventing storage encryption schemes. In this paper, we present our design for a new cloud storage encryption scheme that enables cloud storage providers to create convincing fake user secrets to protect user privacy. Since coercers cannot tell if obtained secrets are true or not, the cloud storage providers ensure that user privacy is still securely protected. Most of the proposed schemes assume cloud storage service providers or trusted third parties handling key management are trusted and cannot be hacked; however, in practice, some entities may intercept communications between users and cloud storage providers and then compel storage providers to release user secrets by using government power or other means. In this case, encrypted data are assumed to be known and storage providers are requested to release user secrets.

1.INTRODUCTION

Cloud storage services have rapidly become increasingly popular. Users can store their data on the cloud and access their data anywhere at any time. Because of user privacy, the data stored on the cloud is typically encrypted and protected from access by other users. Considering the collaborative property of the cloud data, attribute-based encryption (ABE) is regarded as one of the most suitable encryption schemes for cloud storage. There are numerous ABE schemes that have been proposed. Most of the proposed schemes assume cloud storage service providers or trusted third parties handling key management are trusted and cannot be hacked; however, in practice, some entities may intercept communications between users and cloud storage providers and then compel storage providers to release user secrets by using government power or other means. In this case, encrypted data are assumed to be known and storage providers are requested to release user secrets. Our scheme is based on Waters ciphertext policy-attribute based encryption (CP-ABE). We enhance the Waters scheme from prime order bilinear groups to composite order bilinear groups. By the subgroup decision problem assumption, our scheme enables users to be able to provide fake secrets that seem legitimate to outside coercers.

2.Problem Statement

Most deniable public key schemes are bitwise, which means these schemes can only process one bit a time; therefore, bitwise deniable encryption schemes are inefficient for real use, especially in the cloud storage service case. To solve this problem, designed a hybrid encryption scheme that simultaneously uses symmetric and asymmetric encryption. They use a deniably encrypted plan-ahead symmetric data encryption key, while real data are encrypted by a symmetric key encryption mechanism. Most deniable encryption schemes have decryption error problems. These errors come from the designed decryption mechanisms. uses the subset decision mechanism for decryption. The receiver determines the decrypted message according to the subset decision result. If the sender chooses an element from the universal set but unfortunately the element is located in the specific subset, then an error occurs. The same error occurs in all translucentset-based deniable encryption schemes.

3.Scope

The policy of a file may be denied under the request by the customer, when terminating the time of the agreement or totally move the files starting with one cloud then onto the next cloud nature's domain. The point when any of the above criteria exists the policy will be repudiated and the key director will totally evacuates the public key of the associated file. So no one can recover the control key of a repudiated file in future. For this reason we can say the file is certainly erased. To recover the file, the user must ask for the key supervisor to produce the public key. For that the user must be verified. The key policy attribute based encryption standard is utilized for file access which is verified by means of an attribute connected with the file. With file access control the file downloaded from the cloud will be in the arrangement of read just or write underpinned. Every client has connected with approaches for each one file. So the right client will access the right file. For making file access the key policy attribute based encryption.

3.1 Existing System

Most previous deniable encryption schemes, we do not use translucent sets or simulatable public key systems to implement deniability. Instead, we adopt the idea proposed in with some improvements. We construct our deniable encryption scheme through a multidimensional space. All data are encrypted into the multidimensional space. Only with the correct composition of dimensions is the original data obtainable. With false composition, ciphertexts will be decrypted to predetermined fake data. The information defining the dimensions is kept secret. We make use of composite order bilinear groups to construct the multidimensional space. We also use chameleon hash functions to make both true and fake messages convincing.

3.2 Proposed System

Techniques used in previous deniable encryption schemes, we build two encryption environments at the same time, much like the idea proposed in .We build our scheme with multiple dimensions while claiming there is only one dimension. This approach removes obvious redundant parts in . We apply this idea to an existing ABE scheme by replacing prime order groups with composite order groups. Since the base ABE scheme can encrypt one block each time, our deniable CPABE is certainly a blockwise deniable encryption scheme. Though the bilinear operation for the composite order group is slower than the prime order group, there are some techniques that can convert an encryption scheme from composite order groups to prime order groups for better computational performance.

Advantages:

❖ **Blockwise Deniable ABE.:**

This reduces the repeating number from the block size to the key size. Though bitwise deniable encryption is more flexible than blockwise deniable encryption in ”cooking” fake data, when considering cloud storage services, blockwise encryption is much more efficient in use.

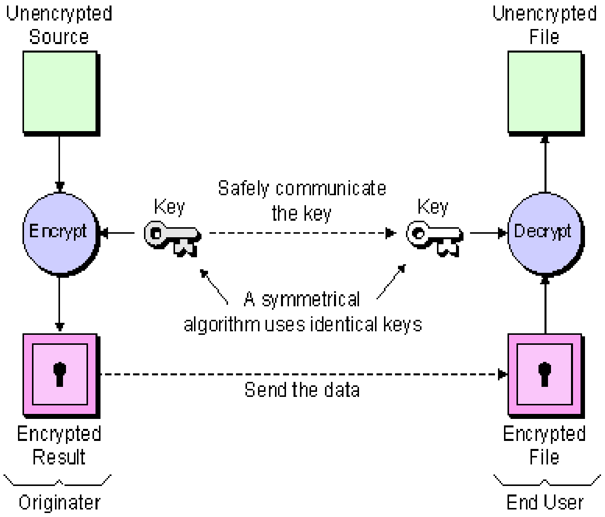
❖ **Consistent Environment:**

we build a consistent environment for our deniable encryption scheme. By consistent environment, we means that one encryption environment can be used for multiple encryption times without system updates.

❖ **Deterministic Decryption:**

The concept of our deniable scheme is different than these schemes described above. Our scheme extends a pairing ABE, which has a deterministic decryption algorithm

4.Implementation of modules



Deniable Encryption:

Deniable encryption involves senders and receivers creating convincing fake evidence of forged data in ciphertexts such that outside coercers are satisfied. Note that deniability comes from the fact that coercers cannot prove the proposed evidence is wrong and therefore have no reason to reject the given evidence. This approach tries to altogether block coercion efforts since coercers know that their efforts will be useless. We make use of this idea such that cloud storage providers can provide audit-free storage services. In the cloud storage scenario, data owners who store their data on the cloud are just like senders in the deniable encryption scheme. Those who can access the encrypted data play the role of receiver in the deniable encryption scheme, including the cloud storage providers themselves, who have system wide secrets and must be able to decrypt all encrypted data. We make use of ABE characteristics for securing stored data with a fine-grained access control mechanism and deniable encryption to prevent outside auditing.

Composite Order Bilinear Group:

Design a deniable CP-ABE scheme with composite order bilinear groups for building audit-free cloud storage services. Composite order bilinear groups have two attractive properties, namely projecting and cancelling. We make use of the cancelling property for building a consistent environment; however, Freeman also pointed out the important problem of computational cost in regard to the composite order bilinear group. The bilinear map operation of a composite order bilinear group is much slower than the operation of a prime order bilinear group with the same security level. That is, in our scheme, a user will spend too much time in decryption when accessing files on the cloud. To make composite order bilinear group schemes more practical, into prime order schemes. both projecting and cancelling cannot be simultaneously achieved in prime order groups in . For the same reason, we use a simulating tool proposed to convert our composite order bilinear group scheme to a prime order bilinear group scheme. This tool is based on dual orthonormal bases and the subspace assumption. Different subgroups are simulated as different orthonormal bases and therefore, by the orthogonal property, the bilinear operation will be cancelled between different subgroups. Our formal deniable CP-ABE construction method uses only the cancelling property of the composite order group.

Attribute-Based Encryption:

Cloud storage services have rapidly become increasingly popular. Users can store their data on the cloud and access their data anywhere at any time. Because of user privacy, the data stored on the cloud is typically encrypted and protected from access by other users. Considering the collaborative property of the cloud data, attribute-based encryption (ABE) is regarded as one of the most suitable encryption schemes for cloud storage. There are numerous ABE schemes that have been proposed, including . Most of the proposed schemes assume cloud storage service providers or trusted third parties handling key management are trusted and cannot be hacked; however, in practice, some entities may intercept communications between users and cloud storage providers and then compel storage providers to release user secrets by using government power or other means. In this case, encrypted data are assumed to be known and storage providers are requested to release user secrets. As an example, in 2010, without notifying its users, Google released user documents to the FBI after receiving a search warrant . In 2013, Edward Snowden disclosed the existence of global surveillance programs that collect such cloud data as emails, texts, and voice messages from some technology companies. Once cloud storage providers are compromised, all encryption schemes lose their effectiveness. Though we hope cloud storage providers can fight against such entities to maintain user privacy through legal avenues, it is seemingly more and more difficult.

Cloud Storage:

Cloud storage services have become increasingly popular. Because of the importance of privacy, many cloud storage encryption schemes have been proposed to protect data from those who do not have access. All such schemes assumed that cloud storage providers are safe and cannot be hacked; however, in practice, some authorities (i.e., coercers) may force cloud storage providers to reveal user secrets or confidential data on the cloud, thus altogether circumventing storage encryption schemes. In this paper, we present our design for a new cloud storage encryption scheme that enables cloud storage providers to create convincing fake user secrets to protect user privacy. Since coercers cannot tell if obtained secrets are true or not, the cloud storage providers ensure that user privacy is still securely protected. Most of the proposed schemes assume cloud storage service providers or trusted third parties handling key management are trusted and cannot be hacked; however, in practice, some entities may intercept communications between users and cloud storage providers and then compel storage providers to release user secrets by using government power or other means. In this case, encrypted data are assumed to be known and storage providers are requested to release user secrets. we aimed to build an encryption scheme that could help cloud storage providers avoid this predicament. In our approach, we offer cloud storage providers means to create fake user secrets. Given such fake user secrets, outside coercers can only obtained forged data from a user's stored ciphertext. Once coercers think the received secrets are real, they will be satisfied and more importantly cloud storage providers will not have revealed any real secrets. Therefore, user privacy is still protected. This concept comes from a special kind of encryption scheme called deniable encryption.

Owner Module:

Owner module is to upload their files using some access policy. First they get the public key for particular upload file after getting this public key owner request the secret key for particular upload file. Using that secret key owner upload their file.

User Module:

This module is used to help the client to search the file using the file id and file name .If the file id and name is incorrect means we do not get the file, otherwise server ask the public key and get the encryption file.If u want the the decryption file means user have the secret key.

Distributed Key Policy Attribute Based Encryption:

KP-ABE is a public key cryptography primitive for one-to-many correspondences. In KP-ABE, information is associated with attributes for each of which a public key part is characterized. The encryptor associates the set of attributes to the message by scrambling it with the comparing public key parts. Every client is assigned an access structure which is normally characterized as an access tree over information attributes, i.e., inside hubs of the access tree are limit doors and leaf hubs are connected with attributes. Client secret key is characterized to reflect the access structure so the client has the ability to decode a cipher-text if and just if the information attributes fulfill his access structure. The proposed scheme consists of four algorithms which is defined as follows

Setup:

This algorithm takes as input security parameters and attribute universe of cardinality N. It then defines a bilinear group of prime number. It returns a public key and the master key which is kept secret by the authority party.

Encryption:

It takes a message, public key and set of attributes. It outputs a cipher text.

Key Generation:

It takes as input an access tree, master key and public key. It outputs user secret key.

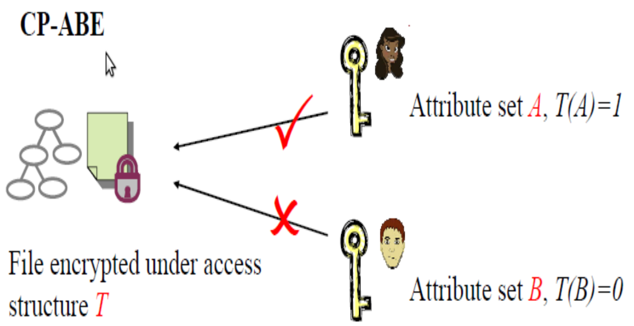
Decryption:

It takes as input cipher text, user secret key and public key. It first computes a key for each leaf node. Then it aggregates the results using polynomial interpolation technique and returns the message.

Algorithm:

Deniable (CP-ABE): Our plan-ahead, bideniable, and multi-distributional CP-ABE scheme is composed of the following **algorithms:**

- ❖ Setup(1) → (PP,MSK): This algorithm takes security parameter as input and returns public parameter PP and system master key MSK.
- ❖ KeyGen(MSK, S) → SK: Given set of attributes S and MSK, this algorithm outputs private key SK.
- ❖ Enc(PP,M,A) → C: This encryption algorithm takes as input public parameter PP, message M, and LSSS access structure A = (M,) over the universe of attributes. This algorithm encrypts M and outputs a ciphertext C, which can be decrypted by those who possess an attribute set that satisfies access structure A. Note that A is contained in C.
- ❖ Dec(PP, SK,C) → {M,⊥}: This decryption algorithm takes as input public parameter PP, private key SK with its attribute set S, and ciphertext C with its access structure A. If S satisfies A, then this algorithm returns M; otherwise, this algorithm returns ⊥.
- ❖ OpenEnc(PP,C,M) → PE: This algorithm is for the sender to release encryption proof PE for (M,C).OpenDec(PP, SK,C,M) → PD: This algorithm is for the receiver to release decryption proof PD for (M,C).
- ❖ Verify(PP,C,M, PE, PD) → {T, F}: This algorithm is used to verify the correctness of PE and PD


CP-ABE
Attribute set *A*, *T(A)=1*
File encrypted under access structure *T*
Attribute set *B*, *T(B)=0*

Conclusion:

In this work, we proposed a deniable CP-ABE scheme to build an audit-free cloud storage service. The deniability feature makes coercion invalid, and the ABE property ensures secure cloud data sharing with a fine-grained access control mechanism. Our proposed scheme provides a possible way to fight against immoral interference with the right of privacy. We hope more schemes can be created to protect cloud user privacyIn this work, we proposed a deniable CP-ABE scheme to build an audit-free cloud storage service. The deniability feature makes coercion invalid, and the ABE property ensures secure cloud data sharing with a fine-grained access control mechanism. Our proposed scheme provides a possible way to fight against immoral interference with the right of privacy. We hope more schemes can be created to protect cloud user privacy

REFERENCES

[1]A. Sahai and B. Waters, “Fuzzy identity-based encryption,” in Eurocrypt, 2005, pp. 457–473.

[2]V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-based encryption for fine-grained access control of encrypted data,” in ACM Conference on Computer and Communications Security, 2006, pp. 89–98.

[3]J. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-policy attribute-based encryption,” in IEEE Symposium on Security and Privacy, 2007, pp. 321–334.

[4] B. Waters, “Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization,” in Public Key Cryptography, 2011, pp. 53–70.

[5]A. Sahai, H. Seyalioglu, and B. Waters, “Dynamic credentials and ciphertext delegation for attribute-based encryption,” in Crypto, 2012, pp. 199–217.

[6]S. Hohenberger and B. Waters, “Attribute-based encryption with fast decryption,” in Public Key Cryptography, 2013, pp. 162–179.

[7]P. K. Tysowski and M. A. Hasan, “Hybrid attribute- and reencryption- based key management for secure and scalable mobile applications in clouds.” IEEE T. Cloud Computing, pp. 172–186, 2013.

[8]Wired. (20 4) Spam suspect uses google docs; fbi happy. [Online].Available: http://www.wired.com/2010/04/cloud-warrant/

[9]Wikipedia. (2014) Global surveillance disclosures (2013present).[Online].Available:http://en.wikipedia.org/wiki/ Global surveillance disclosures (2013-present)

[10](2014) Edward snowden. [Online]. Available: http://en. wikipedia.org/wiki/Edward Snowden

[11]——. (2014) Lavabit.[Online]. Available: http://en.wikipedia. org/wiki/Lavabit

[12]R. Canetti, C. Dwork, M. Naor, and R. Ostrovsky, “Deniable encryption,” in Crypto, 1997, pp. 90–104.

[13]A. B. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters, “Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption,” in Eurocrypt, 2010,pp. 62–91.

[14]N. Attrapadung, J. Herranz, F. Laguillaumie, B. Libert, E. de Panafieu, and C. R`afols, “Attribute-based encryption chemes with constant-size ciphertexts,” Theor.Comput.Sci., vol. 422, pp. 15–38, 2012.

[15]M. D¨urmuth and D. M. Freeman, “Deniable encryption with negligible detection probability: An interactive construction,” inEurocrypt, 2011, pp. 610–626.

[16]A. O’Neill, C. Peikert, and B. Waters, “Bi-deniable public-key encryption,” in Crypto, 2011, pp. 525–542.

[17]P. Gasti, G. Ateniese, and M. Blanton, “Deniable cloud storage: sharing files via public-key deniability,” in WPES, 2010, pp. 31– 42.

[18]M. Klonowski, P. Kubiak, and M. Kutylowski, “Practical deniable encryption,” in SOFSEM, 2008, pp. 599–609.

[19]M. H. Ibrahim, “A method for obtaining deniable public-key encryption,” I. J. Network Security, vol. 8, no. 1, pp. 1–9, 2009.

[20]J. B. Nielsen, “Separating random oracle proofs from complexity theoretic proofs: The non-committing encryption case,” in Crypto, 2002, pp. 111–126