A SECURE AND EFFICIENT ERASURE CODE-BASED CONTENT TRANSMISSION IN CLOUD USING CP-ABE APPROACH

G.Anitha, Mr.S.Arunkumar

M.Tech, SRM University, Department of Computer Science & Engineering
Assistant Professor, SRM University, Department of Computer Science & Engineering

1 gani3086@gmail.com
2 arunatr85@gmail.com

Abstract—In cloud the retention of retrieval data transmission which contain collection of storage servers to enable expanded storage services. Transmission of content in cloud causes data robustness and data confidentiality. For data robustness implementation of erasure code-based content with multiple key servers are concerned for the recovery process using redundancy. For data confidentiality implementation of homomorphic integrity based on new integrity tags can be computed from old integrity tags by storage servers without involvement of the user’s secret key or backup servers CP-ABE/Cipher text policy –Attribute based encryption approach where followed to prove the security of our integrity check scheme formally, and establish the parameters for achieving for data retrieval. A system for realizing complex access control on encrypted data that we call Ciphertext-Attribute-Based Encryption. By using our techniques encrypted data can be kept confidential even if the storage server is untrusted. Moreover, our methods are secure against collusion attacks. Previous Attribute Based Encryption systems used attributes to describe the encrypted data and built policies into user’s keys, while in our system attributes are used to describe a user’s credentials, and a party encrypting data determines a policy for who can decrypt.

Keywords: Cipher text policy-Attribute based Encryption, key based Encryption.

I. INTRODUCTION

CLOUD storage systems driven by high speed networks and large data centers provide reliable storage services over the Internet. A user can store his files in a cloud storage service, and access them via the Internet later. Because the user no longer possesses his files in his local depository, he is rightfully concerned about the security of the stored files[2]. Data confidentiality and data robustness are the main security issues. For data confidentiality, the user can first encrypt files, and then store the encrypted files in a cloud storage. Thus, the files in the cloud are confidential against not only outsiders, but also the cloud service provider. For data robustness, there are two concerns: service failure, and service corruption. Service failure is when the user does not get responses for his retrieval requests, and service corruption is when the user gets corrupted data. We are concerned about the data robustness issue for cloud services in this paper. There exist several expressive ABE schemes where the decryption algorithm only requires a constant number of pairing computations. Recently, Green et al. proposed a remedy to this problem by introducing the notion of ABE with outsourced decryption, which largely eliminates the decryption overhead for users. Based on the existing ABE schemes, Green et al. also presented concrete ABE schemes with outsourced decryption. In these existing schemes, a user provides an untrusted server, say a proxy operated by a cloud service provider, with a transformation key TK that allows the latter to translate any ABE ciphertext CT by that user’s attributes or access policy into a simple ciphertext CT’, and it only incurs a small overhead for the user to recover the plaintext from the transformed ciphertext CT’. The security property of the ABE scheme with outsourced decryption guarantees that an adversary (including the malicious cloud server) be not able to learn anything about the encrypted message; however, the scheme provides no guarantee on the correctness of the transformation done by the cloud server. In the cloud computing setting, cloud service providers may have strong financial incentives to return incorrect answers, if such answers require less work and are unlikely to be detected by users.

We considered the verifiability of the cloud’s transformation and provided a method to check the correctness of the transformation. However, the we did not formally define verifiability. But it is not feasible to construct ABE schemes with verifiable outsourced decryption following the model defined in the existing. Moreover, the method proposed in existing relies on random oracles (RO). Unfortunately, the RO model is heuristic, and a proof of security in the RO model does not directly imply anything about the security of an ABE scheme in the real world. It is well known that there exist cryptographic schemes which are secure in the RO model but are inherently insecure when the RO is instantiated with any real hash function.

In this work, firstly modify the original model of ABE with outsourced decryption in the existing to allow for verifiability of the transformations. After describing the formal definition of verifiability, we propose a new ABE model and based on this new model construct a concrete ABE scheme with verifiable outsourced decryption. Our scheme does not rely on random oracles. In this paper we only focus on CP-ABE with verifiable outsourced decryption. The same approach applies to KP-ABE with verifiable outsourced decryption. To assess the performance of our ABE scheme with verifiable outsourced decryption, we implement the CP-ABE scheme with verifiable outsourced decryption and conduct experiments on both an ARM-based mobile device and an...
Intel-core personal computer to model a mobile user and a proxy, respectively.

II. LITERATURE SURVEY

The literature survey specifies how Erasure code is applied in Cipher text policy and how ABE algorithm can be used for choosing the best mater to change the content.

- A Secure Erasure Code-Based Cloud Storage System with Secure Data Forwarding
  Data robustness is to replicate a message such that each storage server stores a copy of the message. It is very robust because the message can be retrieved as long as one storage server survives. Another way is to encode a message. servers act as storage nodes in a content addressable storage system for storing content addressable blocks. Our key servers act as access nodes for providing a front-end layer such as a traditional file system interface.

- Erasure Coding for Cloud Storage Systems: A Survey
  To save computational resources, the construction of exact regenerating codes has first been considered. exact repair helps to maintain the systematic erasure codes in the storage system, such that no decoding operations are required to recover the original file.

- Remote Data Checking Using Provable Data Possession
  The trade-offs in performance, security, and space overheads when adding robustness to a remote data checking scheme

- Ciphertext-Policy Attribute-Based Encryption
  To describe the encrypted data and built policies into user’s keys; while in our system attributes are used to describe a user’s credentials, and a party encrypting data determines a policy for who can decrypt.

- Secure Hash and Sign Signatures Without the Random Oracle
  A hash function which satisfies some strong (but well defined) computational assumptions. Finally, we demonstrate that these assumptions are reasonable, by proving that a function satisfying them exists under standard intractability assumptions.

- Provably Data Possession at Untrusted Stores
  We introduce a model for provable data possession (PDP) that allows a client that has stored data at an untrusted server to verify that the server possesses the original data without retrieving it. The model generates probabilistic proofs of possession by sampling random sets of blocks from the server, which drastically reduces I/O costs. The client maintains a constant amount of metadata to verify the proof. The challenge/response protocol transmits a small, constant amount of data, which minimizes network communication. Thus, the PDP model for remote data checking supports large data sets in widely-distributed storage systems.

- Remote Data Checking Using Provable Data Possession
  We introduce a model for provable data possession (PDP) that can be used for remote data checking: A client that has stored data at an untrusted server can verify that the server possesses the original data without retrieving it. The model generates probabilistic proofs of possession by sampling random sets of blocks from the server, which drastically reduces I/O costs. The client maintains a constant amount of metadata to verify the proof. The challenge/response protocol transmits a small, constant amount of data, which minimizes network communication. Thus, the PDP model for remote data checking is lightweight and supports large data sets in distributed storage systems. The model is also robust in that it incorporates mechanisms for mitigating arbitrary amounts of data corruption. We present two provably-secure PDP schemes that are more efficient than previous solutions. In particular, the overhead at the server is low (or even constant), as opposed to linear in the size of the data. We then propose a generic transformation that adds robustness to any remote data checking scheme based on spot checking. Experiments using our implementation verify the practicality of PDP and reveal that the performance of PDP is bounded by disk I/O and not by cryptographic computation. Finally, we conduct an in-depth experimental evaluation to study the trade-offs in performance, security, and space overheads when adding robustness to a remote data checking scheme.

1. Problem Identification
  CP-ABE decryption were imported. The method of verifiable of the cloud transformation provide the correctness of the transformation. Redundancy were avoided by erasure code during the transmission of data in cloud.

III. PROPOSED SYSTEM

We considered the verifiability of the cloud’s transformation and provided a method to check the correctness of the transformation. However, the we did not formally define verifiability. But it is not feasible to construct ABE schemes with verifiable outsourced decryption following the model defined in the existing. Moreover, the method proposed in existing relies on random oracles (RO). Unfortunately, the RO model is heuristic, and a proof of security in the RO model does not directly imply anything about the security of an ABE scheme in the real world. It is well known that there exist cryptographic schemes which are secure in the RO model but are inherently insecure when the RO is instantiated with any real hash function.

In this work, firstly modify the original model of ABE with outsourced decryption in the existing to allow for verifiability of the transformations. After describing the formal definition of verifiability, we propose a new ABE model and based on this new model construct a concrete ABE scheme with
verifiable outsourced decryption. Our scheme does not rely on random oracles.

In this paper we only focus on CP-ABE with verifiable outsourced decryption. The same approach applies to KP-ABE with verifiable outsourced decryption. To assess the performance of our ABE scheme with verifiable outsourced decryption, we implement the CP-ABE scheme with verifiable outsourced decryption and conduct experiments on both an ARM-based mobile device and an Intel-core personal computer to model a mobile user and a proxy, respectively, in fig 4.1.

![Fig 4.1 Architecture diagram for cipher policy](image)

IV. ALGORITHM SPECIFICATION

ABE Algorithms

Attribute-based encryption (ABE) is a relatively recent approach that reconsiders the concept of public-key cryptography. In traditional public-key cryptography, a message is encrypted for a specific receiver using the receiver’s public-key. Identity-based cryptography and in particular identity-based encryption (IBE) changed the traditional understanding of public-key cryptography by allowing the public-key to be an arbitrary string, e.g., the email address of the receiver[3]. ABE goes one step further and defines the identity not atomic but as a set of attributes, e.g., roles, and messages can be encrypted with respect to subsets of attributes (key-policy ABE - KP-ABE) or policies defined over a set of attributes (ciphertext-policy ABE - CP-ABE). The key issue is, that someone should only be able to decrypt a ciphertext if the person holds a key for “matching attributes” (more below) where user keys are always issued by some trusted party. Attribute-based encryption (ABE) is a vision of public key encryption that allows users to encrypt and decrypt messages based on user attributes. This functionality comes at a cost. In a typical implementation, the size of the ciphertext is proportional to the number of attributes associated with it and the decryption time is proportional to the number of attributes used during decryption. Specifially, many practical ABE implementations require one pairing operation per attribute used during decryption.

2. RSA Algorithm

3. The RSA algorithm is the mostly used for key generations and key distributions in zone. RSA is an algorithm for public-key cryptography that is based on the presumed difficulty of factoring large integers, the factoring problem. The RSA algorithm involves three steps: key generation, encryption and decryption. Here, safe zone required during key generation phase can be created using pair-wise secret keys exchanged between each of the 3 member which are involved over an infrared zone. RSA keys are often of length a power of two, like 512, 1024, or 2048 bits. The RSA algorithm operates as follows as:

- Encryption of plaintext
- Decryption of ciphertext

RSA as described above suffers from several problems given our definitions. First, it is deterministic, since a given message always encrypts to the same ciphertext. Further, it does not satisfy Non-Malleability, since two encryptions, can, for example, be multiplied to get a new encryption. In some contexts, this kind of malleability can be useful, but it should be taken into account when designing systems that use RSA.

V. CONCLUSION

we only focus on CP-ABE with verifiable outsourced decryption. To justify the performance of our ABE scheme with verifiable outsourced decryption, we implement the CP-ABE scheme with verifiable outsourced decryption and conduct experiments on both an ARM-based mobile device and an Intel-core personal computer to model a mobile user and a proxy. The content based data transmission taken place with the help of erasure code.

REFERENCES


