ADVANCE SECURITY FOR CLOUD USING DYNAMIC KEY.

Provide security using dynamic key.

Mrs. S.A.Ubale.
Atul Narute, Tejas Kalambe, Sarika Nalawade, Kiran Ingulkar

atulnarute9@gmail.com, tejaskalambe2@gmail.com, sarikanalawade199@gmail.com, kiranigulamkar09@gmail.com
Computer Engineering, ZCOER, Narhe, Pune-411041.

Abstract —*Cloud computing is a kind of Internet-based computing that provides shared processing resources and data to computers and other devices on demand. It is a model for enabling ubiquitous, on-demand access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) which can be rapidly provisioned and released with minimal management effort. Cloud computing and storage solutions provide users and enterprises with various capabilities to store and process their data in third-party data centers. Encryption is the process of transforming information so it is unintelligible to anyone but the intended recipient. Decryption is the process of transforming encrypted information so that it is intelligible again. Decryption with the correct key is simple. Decryption without the correct key is very difficult, and in some cases impossible for all practical purposes. In our approach we are going to unite the key and Session id for encryption and decryption purpose. So that we can have a stronger security for our storage because as per the session id changes dynamically our key also change dynamically. By getting the references from the above components we are developing strong security to the cloud.*

Keywords- *cloud security, encryption, decryption, advance security, encryption key, etc.*

INTRODUCTION

Public cloud storage is a service in cloud computing where data owners and storage service providers are typically in Separate domains . A data owner may store his data files in a service provider's domain and relies on the service provider to provide users with access to the data files. Without proper security controls, the service provider may abuse the trust of the data owner. For example, without the consent of the data owner, the service provider may view or edit the content of the stored data files, or may give such access to users who do not have permission to use the data. Therefore, some form of cryptographic access control needs to be implemented in the Cloud. On one hand, it enables data owners to protect the confidentiality of their own data by storing them on the cloud servers in an encrypted form. On the other hand, ABE facilities granting access rights to the data by allowing access decision to the data to be made based on a users attributes such as a doctor or a nurse. There are basically two forms of ABE: key-policy ABE and ciphertext - policy Attribute-based encryption . In KP-ABE, each user is given an access policy which could be a set of attributes that are linked with logical operators, such as AND and OR. A user can decrypt the encrypted data file if and only if the file's attributes Satisfies the access policy attached to the key of the user. In the CP-ABE, on the other hand, each user is given a set of attributes, and each data file is associated with an access policy. A user can decrypt the encrypted data file if and only if the user's attributes satisfy the access policy of the file. In most existing ABE schemes, a centralized master authority is used to issue secret keys for all users in a system. However, this may cause a performance bottleneck on the centralized authority, especially when the number of users is prohibitively large. Current researches address this issue by proposing Multi-Authority ABE and Hierarchical ABE schemes. Their proposals have considered various scenarios. However, their proposals are either expensive computationally or not designed for large-scale cloud environment. In this paper, we are consider a scenario where a company operates many departments distributed in different places, and wishes to share its data with the departments through a public cloud provider. Company A pays the cloud provider for sharing corporate data in cloud servers, and has many branches distributed in different locations. The data owner, which could be any one in the company, may upload data files onto the cloud along with their access policy. The head office of the company, which acts as a master authority, may delegate a set of attributes to each of its branches. These branches, which act as multi-attribute authority, may issue secret keys for users in the next level. The user may then access any data file stored in the cloud servers if and only if his attributes in the secret key satisfy the data files access policy.

EXISTING SYSTEM

There are required two keys that is public & private, and these keys are not change. Cryptographic keys for both encryption of plaintext and decryption of ciphertext. The keys may be identical or there may be a simple transformation to go between the two keys The keys, in practice, represent a shared secret between two or more parties that can be used to maintain a private information link.[2] This requirement that both parties have access to the secret key

is one of the main drawbacks of symmetric key encryption, in comparison to public-key encryption (also known as asymmetric key encryption).

PROPOSED SYSTEM

In our system , we are using key and Session id along with encryption and Decryption purpose. We get a new session id every time we start our session, key will be same throughout the session. We are going to embed the session id and old key to change the key for every transaction dynamically. So that security will be more strong and Reliable

SYSTEM ARCHITCTURE

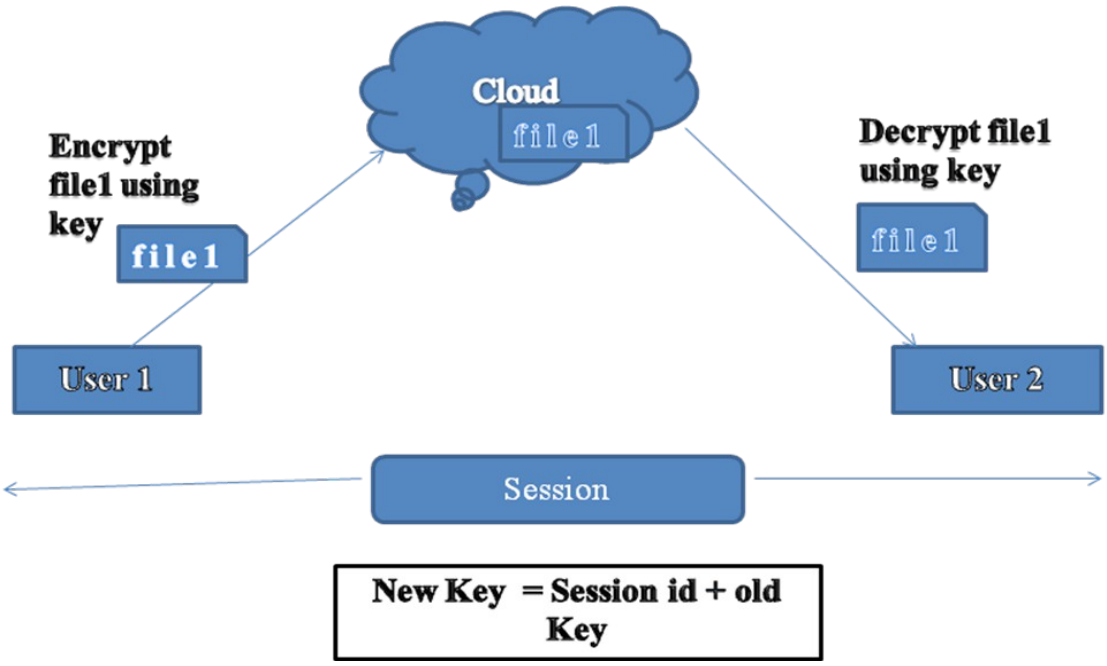

Figure : System Architecture

1. A protocol for outsourcing data storage to a cloud provider in secure fashion is provided. The provider is unable to read stored data; authorized users may do so based on qualification through possession of the right attributes without arbitration by the dataowner. The protocol is designed to be efficient for resource-constrained mobile users by delegating computation and requests to a cloud provider or trusted authority, where appropriate, without compromising security.

2. An improvement is made over a traditional attribute based encryption scheme, such that responsibility over key generation is divided between a mobile data owner and a trusted authority; the owner is relieved of the highest computational and messaging burdens.

3. Additional security is provided through a group keying mechanism; the data owner controls access based on the distribution of an additional secret key, beyond possession of the required attributes. This additional security measure is an optional variant applicable to highly sensitive data subject to frequent access.

4. Re-encryption, as a process of transforming the stored ciphertext, permits efficient revocation of users; it does not require removal of attributes and subsequent key regeneration, and may be administered by a trusted authority without involvement of the data owner.

5. The real-world performance of the proposed protocol is demonstrated on commercially popular mobile and cloud platforms. In addition, simulations show the scalability of the protocol in terms of computational workloads within a very large mobile user population.

REFERANCES

[1] Adel Binbusayyis and Ning Zhang, "School of Computer Science, The University of Manchester, Manchester", 2015.

[2] W. Stallings, "Network Security, Prentice Hall Press, Uppeer Saddle River, NJ", 2015.

[3] J. Hur and D.K. Noh, "Attribute-Based Access Control with Efficient Revocation in Data Outsourcing Systems", 2015.

[4] J. Bethencourt, A. Sahai and B. Waters, "Ciphertext-Policy Attribute-Based Encryption", 2014.

[5]P.K. Tysowski and M.A. Hasan, "Hybrid Attribute-Based Encryp-tion and Re-Encryption for Scalable Mobile Applications in Clouds", 2013.