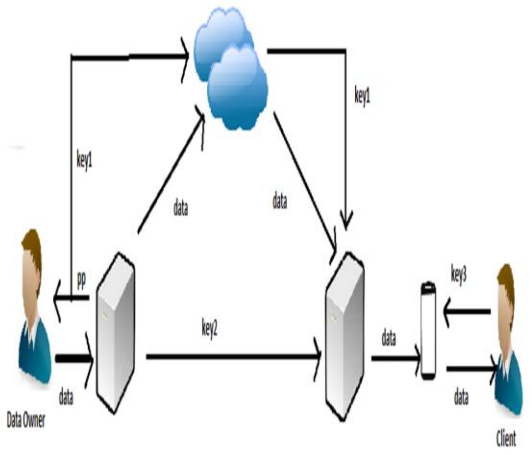
© April 2018 | IJIRT | Volume 4 Issue 11 | ISSN: 2349-6002

Key Management Protocol in Cipher text Policy Attribute

Mitta Karthik[1], M. Padmavathamma[2]
[1]*Student, Dept. of Computer Science, SVU CM & CS, Tirupathi*
[2]*Project Guide, M.Sc., M.Ed., M.S., M.Phil., Ph.D., Dept. of Computer Science, SVU CM & CS, Tirupati*

Abstract- **Cipher text policy attribute-based encryption (CP-ABE) is a promising cryptographic technique for fine-grained access control of outsourced data in the cloud. However, some drawbacks of key management hinder the popularity of its application. One drawback in urgent need of solution is the key escrow problem. We indicate that front-end devices of clients like smart phones generally have limited privacy protection, so if private keys are entirely held by them, clients risk key exposure that is hardly noticed but inherently existed in previous research. Furthermore, enormous client decryption overhead limits the practical use of ABE. In this work, we propose a collaborative key management protocol in CP-ABE (CKM-CP-ABE). Our construction realizes distributed generation, issue and storage of private keys without adding any extra infrastructure. A fine-grained and immediate attribute revocation is provided for key update. The proposed collaborative mechanism effectively solves not only key escrow problem but also key exposure. Meanwhile, it helps markedly reduce client decryption overhead. A comparison with other representative CP-ABE schemes demonstrates that our scheme has somewhat better performance in terms of cloud-based outsourced data sharing on mobile devices. Finally, we provide proof of security for the proposed protocol.**

ARCHITECTURE DIAGRAM



EXISTING SYSTEM

Attribute-based encryption (ABE) is a promising cryptographic primitive that offers an interesting solution to secure and flexible data sharing. ABE has an inherent one-to-many property, which means a single key can decrypt different cipher texts or different keys can decrypt the same cipher text. There are two types of ABE, called cipher text policy ABE (CP-ABE) and key policy ABE (KP-ABE). For CP-ABE, the access policy is embedded into a cipher text and the attribute set is embedded into a private key. For KP-ABE, the access policy is embedded into a private key and the attribute set is embedded into a cipher text. CP-ABE allows data owners to define their own access policy. As mentioned above, previous schemes of key management in attribute-based data sharing system mainly focuses on key update, proxy re-encryption and outsourced decryption. Some research demonstrated untrusted key authority may lead to key escrow problem and provided corresponding solutions. However, little research notices that if authority is untrusted, front-end devices especially mobile ones must be far more untrusted than it because they are inherently vulnerable to illegal access. If private keys are still entirely stored in front-end devices, a worse problem called key exposure occurs threatening confidentiality of private keys. In addition, most of attribute-based data sharing schemes enhanced security of key management at the cost of decryption overhead of data receivers. Therefore, we are not satisfied with previous Schemes of key management in terms of either security or efficiency.

PROPOSED SYSTEM

We propose a novel collaborative key management protocol in ciphertext policy attribute-based encryption (CKM-CP-ABE) aiming to enhance security and efficiency of key management in cloud data sharing system. A novel collaborative protocol is presented. With help of interaction among the key authority, a cloud server and a client who tends to access data, distributed generation, issue and storage of private keys are realized. Thus, secure key management is guaranteed without adding any extra physical infrastructure, which is more easy to deploy compared with previous multi-authority schemes. We introduce attribute groups to build the private key update algorithm. A unique attribute group key is allocated to each attribute group that contains clients who share the same attribute. Via updating attribute group key, a fine-grained and immediate attribute revocation is provided. That not only key escrow problem but also key exposure is threatening the confidentiality of private keys, which is hardly noticed in previous research. Compared to previous key management protocols for attribute-based data sharing system in cloud, our proposed protocol effectively addresses both two problems by its collaborative key management. Finally, we provide proof of security for the proposed protocol. The collaborative mechanism helps markedly reduce client decryption overhead by employing a decryption server to execute most of decryption while leave no knowledge about information to it.

MODULE IMPLEMENTATION

- Client
- Key Authority
- Encryption
- Decryption
- Data Owner

1. Client:

A client is a user who intends to access data in cloud storage via front-end devices. With the potential trend of mobile cloud services, mobile devices are the majority of front-end devices. If the client's attribute set satisfies an access policy associated with ciphertext, the client will be allowed to correct plaintext. We assume that most mobile devices are performance-restrained, so clients may be in danger of suffering key exposure.

2. Key Authority:

The key authority is a important component in the system. The key authority is responsible for most calculating tasks, including key generation, key update, etc. We assume that the key authority is semi-trusted in our system, meaning it is interest about the value of plaintext but has no intention of tampering with it.

3. Encryption:

In this paper, we propose a novel collaborative key management protocol in cipher text policy attribute-based encryption (CKM-CP-ABE) aiming to enhance security and efficiency of key management in cloud data sharing system. A novel collaborative protocol is presented. With help of interaction among the key authority, a cloud server and a client who tends to access data, distributed generation, issue and storage of private keys are realized. Thus, secure key management is guaranteed without adding any extra physical infrastructure, which is easier to deploy compared with previous multi-authority schemes. We introduce attribute groups to build the private key update algorithm. We indicate that not only key escrow problem but also key exposure is threatening the confidentiality of private keys, which is hardly noticed in previous research.

4. Decryption:

The decryption server has powerful computing capabilities. It undertakes and isolates the most, but not all task of decryption. We assume that the decryption server is semi-trusted and the decryption server access channel is insecure, because it is sufficient for CKM-CP-ABE to guarantee data security. The collaborative mechanism helps markedly reduce client decryption overhead by employing a decryption server to execute most of decryption while leave no knowledge about information to it.

5. Data Owners:

A data owner is an authorized user in the system who possesses data to be uploaded. Data owner define their own explicit access policies so that only desirable clients are granted permission to obtain plaintext. Data owner first sent request to multiple key authorities for upload our file with high security. After sending the request get keys from multiple key authority for upload file. This process is until going where numbers of data owner upload our file to cloud sever with encryption with data privacy.

ALGORITHMS

- Key Generation Algorithm

Key generation is the process of generating keys in cryptography. A key is used to encrypt and decrypt whatever data is being encrypted/decrypted. A device or program used to generate keys is called a key generator or key gen. An unpredictable (typically large and random) number is used to begin generation of an acceptable pair of keys suitable for use by an asymmetric key algorithm. In an asymmetric key encryption scheme, anyone can encrypt messages using the public key, but only the holder of the paired private key can decrypt.

- Encryption Algorithm

We propose a novel collaborative key management protocol in cipher text policy attribute-based encryption (CKM-CP-ABE) aiming to enhance security and efficiency of key management in cloud data sharing system.

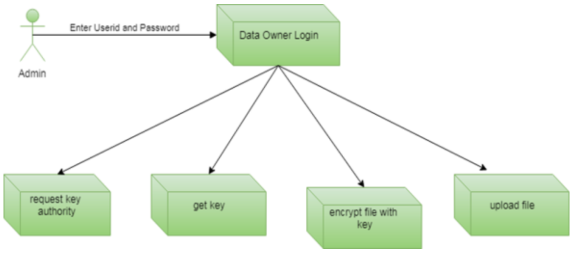
SYSTEM DESIGN

Component Diagram / Use Case Diagram / Flow Diagram

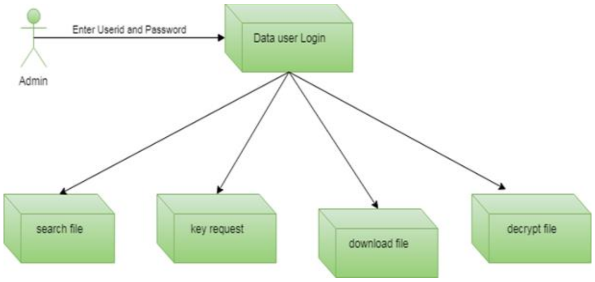
The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of the input data to the system, various processing carried out on these data, and the output data is generated by the system.

Component Diagram

Data Owner



Data User



CONCLUSION

Cipher text policy attribute-based encryption is a promising cryptographic technique to realize fine-grained access control in secure cloud storage. In this paper, we propose a novel collaborative key management protocol to enhance both security and efficiency of key management in cipher text policy attribute-based encryption for cloud data sharing system. Distributed key generation, issue and storage of private keys are realized without adding any extra physical infrastructure. We introduce attribute groups to build a private key update algorithm for fine-grained and immediate attribute revocation. The proposed collaborative mechanism perfectly addresses not only key escrow problem but also a worse problem called key exposure that previous research hardly noticed. Meanwhile it helps to optimize clients' user experience since only a small amount of responsibility is taken by them for decryption. Thus, the proposed scheme performs better in cloud data sharing system serving massive performance-restrained front-end devices with respect to either security or efficiency. Our future work will build on the preliminary findings in this work to develop the proposed scheme by reducing cipher text size, encryption cost and decryption cost, which are still open problems that hinder practical application of attribute-data sharing.