International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No.(02), March 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

Dynamic Groups In The Cloud For Secure Data Sharing

I.Vaseem Akram[1], Mohammad Ahamed Fakker[2], Mrs. D.Kavitha[3]

[1,2,3]M.E, Assistant Professor, Dept. of C.S.E, DACE, Chennai (India)

ABSTRACT

With the use of Cloud Computing technology, Users could be able to share the data among group members in the cloud effectively with lowermaintenance and little management effort. Data sharing in a cloud environment must provide security guarantees, sharing data where providing privacy-preserving is still achallenging issue due to frequent change of the membership due to collusion attack in an untrusted cloud. In Existing systems, the security of key distribution is based on the secure communication channel which is an assumption and difficult for practice in the cloud. In this paper a secure data sharing in the cloud for dynamic members by the way for key distribution without any secure communication channels and to achieve fine-grained access control as a member of the group can have the access to shared resource and once the group members are revoked then theycannot have access to resources again, and data sent over the network is authenticated and trust is established using Digital Signature. Finally, our scheme can achieve fine efficiency, which means previous users need not to change their own private keys for the situation where a new user joins in the group or the user is revoked from the group.

Key words:EC Key distribution, Cloud computing, Digital Signature

I.INTRODUCTION

In cloud computing, CSP offers an infinite storage space for clients to store and manage theirdata. They help Users to reduce their personal financial problem of data management by changing the area from local managements system to the cloud servers.

However, security became one of the main constraint and a possible danger, since we outsource the storage of our own confidential data, which is possibly sensitive to the CSP. To preserve Data privacy and its confidentiality, a common approach we undergo to encrypt data files before the members upload their encrypted data into the cloud storage.

Unfortunately, it is difficult to design a secure and effective data sharing scheme, more eventually for dynamic groups in the cloud.

A cryptographic system that enables secure data to be shared on untrustworthy servers based on the methods that divides files into file groups and encrypting each and everyfile group with a file-block key. Even though, the file-block keys need to be changed and distributed for a situation like user revocation, therefore, the system would have a heavy key distribution problem. Other schemes for data sharing on untrusted CSP have been proposed in.

However, the complex part of user participation and revocation in the scheme are continuously increasing with the number of data owners and the number of revoked users.

Lan Zhou,Vijay Varadharajan, and Michael Hitchenspresented a secure access control scheme where the encrypted data in cloud storage is accessed by invoking role-based encryption technique. It is claimed that the scheme can achieve effective user revocation where it combines role-based access control policies with encryption schemes to have a secure large data storage in the cloud. Unfortunately, the verifications between entities are not concerned, this scheme could be easily suffers from attacks.

Lu et al proposed a secure provenance scheme that actually leverage group signatures and cipher text-policy attribute based encryption techniques. Each and every user obtains two keys after the registration module where the attribute key is used in order to decrypt the data which is been encrypted by the attribute-based encryption methodologies and the group signature key is used for preserving the privacy of users and to trace the activities of that suspected members. Even though, the revocation is not supported in the scheme.

Liu et al presented a secure multi-owner data sharing scheme, named Mona. It is claimed that the scheme can achieve fine-grained access control and revoked users will not be able to access the sharing data again once they are revoked. However, this scheme will suffer from the collusion attack by the revoked user. The revoked user can use his private key to decrypt the encrypted data file and get the secret data after his revocation by conspiring with the cloud. The revoked user sends his request to the cloud, then the cloud responds the corresponding encrypted data file and revocation list to the revoked user without verifications.

Next, the revoked user can compute the decryption key with the help of the attack algorithm. Finally, this attack can lead to the revoked users getting the sharing data and disclosing other secrets of legitimate members.

II.LITERATURE SURVEY

In 2010, Shucheng Yu, Cong Wang, Kui Ren, and Weijing Lou proposed a scheme on "Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing".To keep sensitive user data confidential against untrusted servers, existing solutions usually apply cryptographic methods by disclosing data decryption keys only to authorized users.

The problem of simultaneously achieving fine-grainedness, scalability, and data confidentiality of access control actually still remains unresolved. This paper addresses this challenging open issue by, on one hand, defining and enforcing access policies based on data attributes, and, on the other hand, allowing the data owner to delegate most of the computation tasks involved in fine- grained data access control to untrusted cloud servers without disclosing the underlying data contents.

In 2015,Wei Teng, Geng Yang, Member, IEEE, Yang Xiang, Senior Member, IEEE, Ting Zhang and Dongyang Wang, "Attribute-based Access Control with Constant-size Cipher text in Cloud Computing".is a promising scheme suitable for access control in cloud storage systems.

This paper proposes a hierarchical attribute-based access control scheme with constant-size ciphertext. The scheme is efficient because the length of ciphertext and the number of bilinear pairing evaluations to a constant are fixed. Its computation cost in encryption and decryption algorithms is low. Moreover, the hierarchical

authorization structure of our scheme reduces the burden and risk of a single authority **scenario**. We prove the scheme is of CCA2 security under the decisional q-Bilinear Diffie-Hellman Exponent assumption.

In 2015, Lan Zhou,Vijay Varadharajan, and Michael Hitchens, "Trust Enhanced Cryptographic Role-based Access Control for Secure Cloud Data Storag".To protect the privacy of data stored in the cloud, cryptographic role-based access control (RBAC) schemes have been developed to ensure that data can only be accessed by those who are allowed by access policies.

They have proposed trust models to reason about and improve the security for stored data in cloud storage systems that use cryptographic RBAC schemes. We present a design of a trust-based cloud storage system which shows how the trust models can be integrated into a system that uses cryptographic RBAC schemes.

In 2016, Xing Hu, Duncan S WONG and Chunming Tang,"Highly Efficient Proxy Re-Encryption Schemes for User-End Encrypted Cloud Data Sharing".In a proxy re-encryption (PRE) scheme, a semi- trusted proxy can convert a ciphertext under Alice's public key into another ciphertext that Bob can decrypt without accessing the underlying plaintext.

This property adds flexibility in various applications, such as cloud data sharing. In this paper, we study CCA-secure, single-hop unidirectional PRE schemes without pairings. We gain high efficiency and public verifiability which enables anyone to publicly verify the validity of the original ciphertexts and re-encrypted ciphertexts. With public verifiability, we can offload the integrity check of the wellformedness of ciphertexts from power-restrained clients to any honest-but-curious untrusted public cloud for improved efficiency.

In 2017,Ralf Kusters and Daniel Rausch," A Framework for Universally Compassable Diffie-Hellman Key Exchange".We would therefore like to get rid of reduction proofs for real-world key exchange protocols as much as possible and in many cases altogether, also for higher-level protocols which use the exchanged keys. So far some first steps have been taken in this direction. But existing work is still quite limited, and, for example, does not support Diffie-Hellman (DH) key exchange, a prevalent cryptographic primitive for real-world protocols. In this paper, building on work by K¨usters and Tuengerthal, we provide an ideal functionality in the universal composability setting which supports several common cryptographic primitives, including DH key exchange. This functionality helps to avoid reduction proofs in the analysis of real-world protocols and often eliminates them completely.

III.PROPOSED SYSTEM

In this paper, we propose a secure data sharing scheme, which can achieve secure key distribution and data sharing for dynamic group by establishing trust and authentication.

1. We provide a secure way for key distribution without any secure communication channels.

The users can securely obtain their private keys from group manager without any Certificate Authorities due to the verification for the public key of the user.

2. Our scheme can achieve fine-grained access control, with the help of the group user list, any user in the group can use the source in the cloud and revoked users cannot access the cloud again after they are revoked.

3. We establish trust and authentication among group members in the cloud using Digital Signature Scheme.

IV.THREAT MODEL

As the threat model, in this paper, we propose our Model based on the Dolev-Yao model, from which the adversary can overhear, intercept, and synthesis any message pointed at the communication channels.

With the Dolev-Yao model. The only way to protect the information from adversary by the passive eavesdroppers and active saboteurs is to design the effective security protocols.

That means there is not any kind of secure communication channel between the communication entities.

Therefore, this kind of threaten model can be more effective and practical to demonstrate the communication in the real world.

V.SYSTEM MODEL

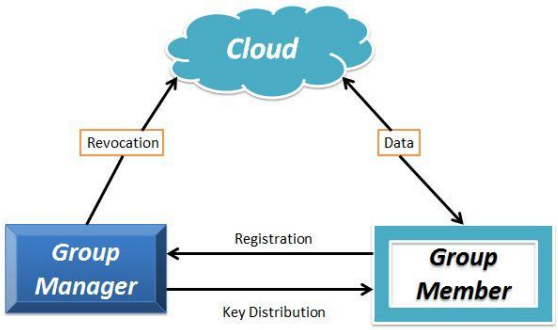


Figure 1 System model

As illustrated in figure 1, the system model consists of three different entities: the cloud, a group manager and a large number of group members.

The cloud, maintained by the cloud service providers, provides storage space for hosting data files in a pay-as-you-go manner. However, the cloud is untrusted since the cloud service providers are easily to become untrusted. Therefore, the cloud will try to learn the content of the stored data.

Group manager takes charge of system parameters generation, user registration, and user revocation. In the practical applications, the group manager usually is the leader of the group. Therefore, we assume that the group manager is fully trusted by the other parties.

Group members (users) are a set of registered users that will store their own data into the cloud and share them with others. In the scheme, the group membership is dynamically changed, due to the new user registration and user revocation.

VI.DESIGN GOALS

We describe the main design goals of the proposed scheme including key distribution, data confidentiality, access control and efficiency as follows:

Key Distribution: The requirement of key distribution is that users can securely obtain their private keys from the group manager without any Certificate Authorities. In existing schemes, this goal is achieved by assuming

that the communication channel is secure, however, in our scheme, we can achieve it without this strong assumption.

Access control: First, group members are able to use the cloud resource for data storage and data sharing. Second, unauthorized users cannot access the cloud resource at any time, and revoked users will be incapable of using the cloud resource again once they are revoked.

Finally Trust among cloud members is established, since every data shared between the group members is authenticated.

Group Manager: Group manager takes charge of system parameters generation, user registration, and user revocation. In the practical applications, the group manager usually is the leader of the group. Therefore, we assume that the group manager is fully trusted by the other parties.

Group members: Group members (users) are a set of registered users that will store their own data into the cloud and share them with others. In the scheme, the group membership is dynamically changed, due to the new user registration and user revocation.

Data confidentiality: Data confidentiality requires that unauthorized users including the cloud are incapable of learning the content of the stored data. To maintain the availability of data confidentiality for dynamic groups is still an important and challenging issue. Specifically, revoked users are unable to decrypt the stored data file after the revocation.

Efficiency: Any group member can store and share data files with others in the group by the cloud. User revocation can be achieved without involving the others, which means that the remaining users do not need to update their private keys.

VII.THE PROPOSED SCHEME

Preliminaries

Bilinear Maps: Let G1 and G2 be additive cyclic groups of the same Prime order. Let e: G1*G2->G2 denote a bilinear map constructed with the following properties:

1.Bilinear:For all a,b belongs to z*q and P,Q belongs to G1, e(aP,bQ)=e(P,Q)ab

2. Nondegenerate: There exists a point Q such that e (Q, Q) ≠1.

3. Computable: There is an efficient algorithm to compute e (P, Q) for any P, Q belongs to G1.

Complexity Assumptions

1. (Basic Diffie-Hellman Problem (BDHP)

2. (Decisional Diffie-Hellman Problem (DDHP) Assumption).

3. (Weak Bilinear Diffie-Hellman Exponent (WBDHE)

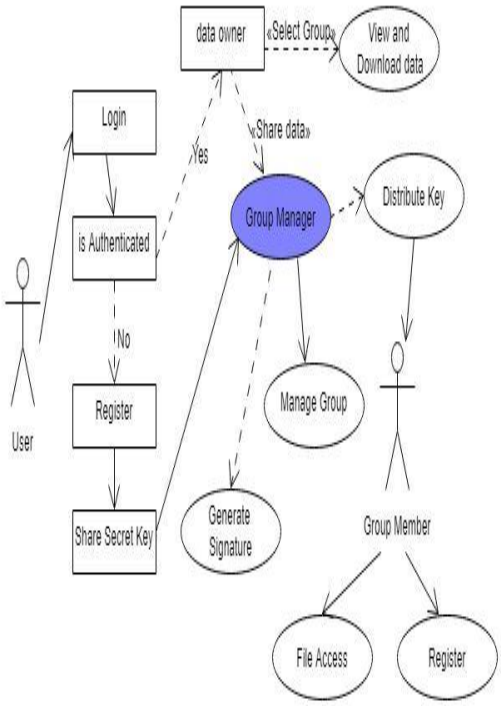
Notations

Each user has a pair of keys **(pk sk)** which is used in the asymmetric encryption algorithm, and pk needs to be negotiated with the group manager on the condition that no Certificate Authorities and security channels are involved in.

KEY is the private key of the user and is used for data sharing in the scheme. UL is the group user list which records part of the private keys of the legal group users.

DL is the data list which records the identity of the sharing data and the time that they are updated.

VIII.ARCHITECTURE DIAGRAM



Elliptic curve cryptography

Elliptic curve cryptography [ECC] is a public-keycryptosystem just like RSA, Rabin, and El Gamal.

Every user has a public key and a private key.

1. Public key is used for encryption/signature verification.
2. PrivateKey is used for decryption/signature generation.

The central part of any cryptosystem involving elliptic curves is the elliptic group.

All public-key cryptosystems have some underlying mathematical operation.

1. RSA has exponentiation (raising the message or cipher text to the public or private values)
2. ECC has point multiplication (repeated addition of two points).

Generic Procedures of ECC

Both parties agree to some publicly-known data items

1. Theelliptic curve equation
   - values of ***a*** and ***b***
   - prime, ***p***
2. Theelliptic group computed from the elliptic curve equation
3. Abasepoint, B, taken from the elliptic group
   - Similar to the generator used in current cryptosystems

Each user generates their public/private key pair

1. Private Key = an integer, x, selected from the interval [1, p-1]
2. Public Key = product, Q, of private key and base point

(Q = x*B)

Scenario – Elliptic Curve Cryptosystem Analog to El Gamal

Suppose Alice wants to send to Bob an encrypted message.

1.Both agree on a base point, B.

2. Alice and Bob create public/private keys.

- *Alice*

  Private Key = a

  Public Key = P_A = a* B
- *Bob*

  Private Key = b

  Public Key = P_B = b * B

3. Alice takes plaintext message, M, and encodes it onto a point, P_M, from the elliptic group

4. Alice chooses another random integer, k from the interval [1, p-1]

5. Thecipher text is a pair of points

- P_C = [(kB), (P_M+ kP_B)]

6**.** To decrypt, Bob computes the product of the first point from P_C and his private key, b

- b * (kB)

7. Bob then takes this product and subtracts it from the second point from P_C

- (P_M + kP_B) – [b(kB)] = P_M + k(bB) – b(kB) = P_M

8. Bob then decodes P_M to get the message, M.

Comparison With El Gamal Algorithm

1. The ciphertext is a pair of points

- P_C = [(kB), (P_M+ kP_B)]

2. The ciphertext in El Gamal is also a pair.

- C = (g^k mod p, mP_B^k mod p)

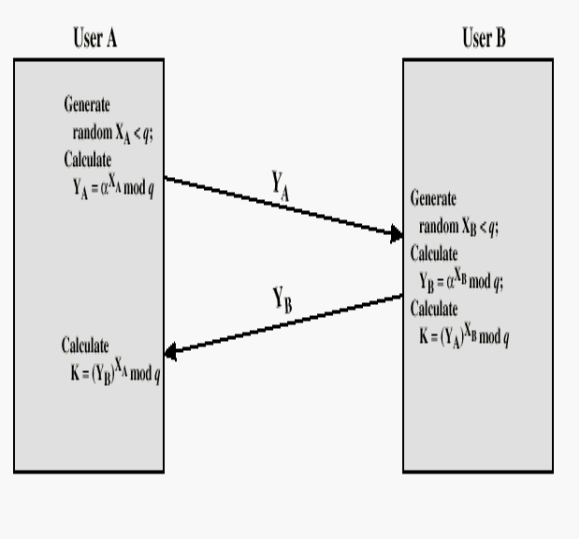
3. Bob then takes this product and subtracts it from the second point from P_C

- $(P_M + kP_B) – [b(kB)] = P_M + k(bB) – b(kB) = P_M$

4. In El Gamal, Bob takes the quotient of the second value and the first value raised to Bob's private value

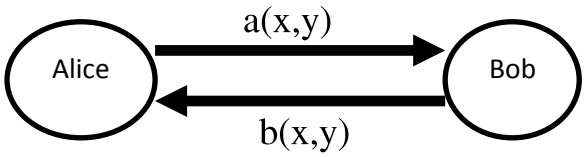
- $m = mP_B^k / (g^k)^b = mg^{k*b} / g^{k*b} = m$

Diffie-Hellman (DH) Key Exchange



ECC Diffie-Hellman

- **Public:** Elliptic curve and point B=(x,y) on curve
- **Secret:** Alice's a and Bob's b



- Alice computes **a(b(x,y))**
- Bob computes **b(a(x,y))**
- These are the same since **ab = ba**

Working of Elliptic Curve

Diffie-Hellman Exchange

- Alice and Bob want to agree on a shared key.

1. Alice and Bob compute their public and private keys.

- Alice

» Private Key = a

» Public Key = P_A = a* B

- Bob

» Private Key = b

» Public Key = P_B = b * B

2. Alice and Bob send each other their public keys.

3. Both take the product of their private key and the other user's public key.

- Alice → K_{AB} = a(bB)
- Bob → K_{AB} = b(aB)
- Shared Secret Key = K_{AB} = abB

Security of ECC

To **protect** a 128 bit AES key it would take a:

1. RSA Key Size: 3072 bits

2. ECC Key Size: 256 bits

NIST guidelines for public key sizes for AES

ECC KEY SIZE (Bits)	RSA KEY SIZE (Bits)	KEY SIZE RATIO	AES KEY SIZE (Bits)
163	1024	1 : 6	
256	3072	1 : 12	128
384	7680	1 : 20	192
512	15 360	1 : 30	256

Supplied by NIST to ANSI X9F1

Hard problem analogous to discrete log

1.Q=kP, where Q,P belong to a prime curve

given k,P → "easy" to compute Q

given Q,P → "hard" to find k

known as the elliptic curve logarithm problem

2.k must be large enough

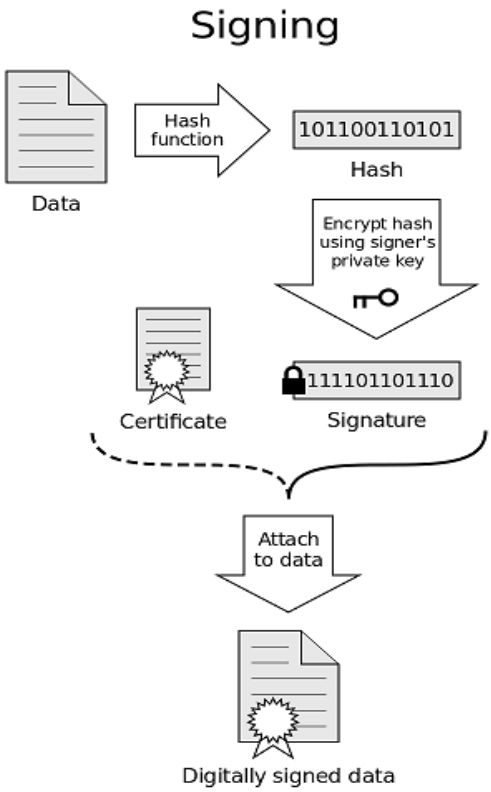
ECC security relies on elliptic curve logarithm problem

Compared to factoring, can use much smaller key sizes than with RSA etc.

For similar security ECC offers significant computational advantages

Signing

The user generates his own digital signature using his private key and attaches it to the data file



The Digital Signature is generated using a secure hashing algorithm like SHA-256

The hashing algorithm generates a constant size hash key which could be act as a Signature of the data file that need to be sent across the Group

IX.VERIFICATION

The verification of the Cloud members is done to authenticate the shared data, so we implement the concept of Digital signature

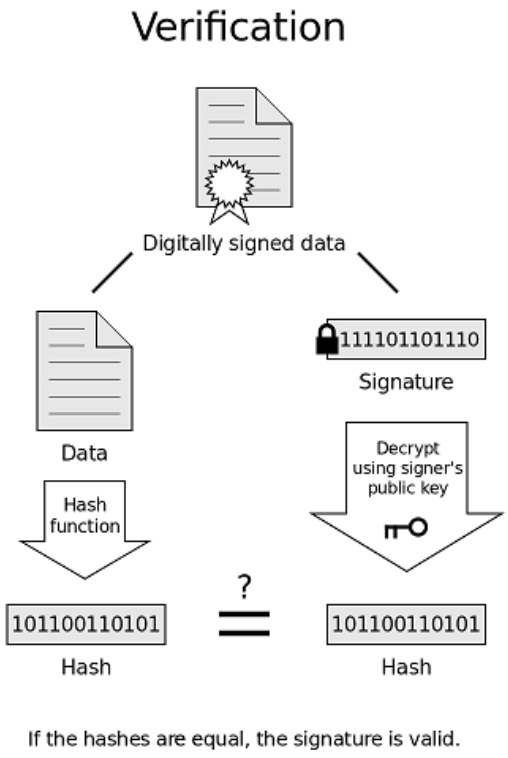
When one of member in the group receive the shared data, he verifies the identity of the sender by separating the data file from the signature.

Now, he decrypt the signature using the sender's public key and obtains a hash value for the data file

Using the hashing algorithm.

The Group member can now verify the authenticity by comparing the signature obtained from the data file along with the attached signature which decrypted by the sender's public key.

If both the signature matched then he could be able ensure that the data file is from a trusted and authenticated user in the Group.



X.CONCLUSION

In this paper, we design a secure data sharing scheme for dynamic groups in the cloud environment. In our scheme, the users can securely obtain their private keys from group manager Certificate Authorities (CA) and secure communication channels. Also, our scheme is able to support dynamic groups efficiently, whenever a new user joins in the group or a registered user is revoked from the group, the private keys of that other users do not need to be recomputed and updated in List. Moreover, our scheme can achieve secure user revocation.Once the users are revoked, they cannot be able to get the original data files even if they contact and conspire with the untrusted cloud. In our scheme the group members are authenticated using Digital Signatures, hence trust among the group members is established to securely share the data. With the use of our Algorithm the key to transfer will be significantly reduced, increasing the speed of Computation and efficiency of key Distribution and Computation using Digital Signatures will be acheived.

REFERENCES

[1] Shucheng Yu, Cong Wang, Kui Ren, and Weijing Lou, "Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing," Proc. ACM Symp. Information, Computer and Comm. Security, pp. 282-292, 2010.

[2] Wei Teng, Geng Yang, Member, IEEE, Yang Xiang, Senior Member, IEEE, Ting Zhang and Dongyang Wang, "Attribute-based Access Control with Constant-size Cipher text in Cloud Computing", IEEE TRANSACTIONS ON CLOUD COMPUTING,2015

[3] Lan Zhou,Vijay Varadharajan,and Michael Hitchens, ”Trust Enhanced Cryptographic Role-based Access Control for Secure Cloud Data Storag” IEEE Transactions on Information Forensics and Security,2015

[4] Xing Hu, Duncan S WONG and Chunming Tang,”HighlyEfficient ProxyRe-Encryption Schemes for User-End Encrypted Cloud Data Sharing”, International Symposium on Parallel and Distributed Computing ,2016

[5] Ralf Kusters and Daniel Rausch,” AFrameworkforUniversally Compassable Diffie-HellmanKeyExchange", IEEE Symposium on Security and Privacy,2017