DIGITAL GUARDIAN

S. Karmugilan[1], S. Sripradeep[2], M. Sumanraj[3], N. Thanigaivel[4]

[1 2 3] Student, [4]Assistant Professor, Department of Computer Science and Engineering, Krishnasamy College of Engineering and Technology, Cuddalore, Tamil Nadu, India.

ABSTRACT *The primary aim of this project is to present an efficient and secure key agreement approach which enables multiple users to freely share their data with high security and efficiency, which improve the efficiency of work in cooperative environments and has widespread potential application. Thus the data owners sign up with their registered basic information. Once the credentials match login is made successful. Data owners load files in the cloud. They share their files among the multiple participants in their group based on the key agreement protocol. The participants those who are assigned with that key to access that file can view the file. This is to improve the group sharing schema. Now, TPA (Key generator) send private key and public key to the concerned data owner and also generates unique key for the owner to upload files in the cloud. Based on the proposed group data sharing model, we present general formulas for generating the common conference key K for multiple participants. TPA transfers files to the owner. Finally, the auditor monitors the man in the middle attack this can be done by auditing the owner of the file. If there is any fault report the auditor blocks the attackers.*

Key Terms: Key agreement protocol, symmetric balanced incomplete block design (SBIBD), data sharing, cloud computing.

1. INTRODUCTION

In the recent years of internet computing, the rising reputation of cloud computing have attracted a large amount of internet users. The cloud computing is all about sharing of resources among users in real time. The cloud server provides convenient storage platform for individuals and organizations, but it also introduces security problems. Real-time refers to sharing of data to be visible instantly to other users who has authentication to see it. Cloud system may be subjected to attacks from both malicious users and cloud providers. It is important to ensure the security of the stored data in the cloud. However, in some applications, multiple data owners would like to securely share their data in a group manner. The protocol that supports secure group data sharing under cloud computing is needed. A key agreement protocol is used to generate a common conference key for multiple participants to ensure the security and this protocol can be applied in cloud computing to support secure and efficient data sharing.

Since it was introduced by taking a Diffie-Hellman protocol, it provides an efficient solution to the problem of creating a common secret key between two participants. In a key Agreement protocol is a protocol in which two or more participants can agree on a key. Thus, The Diffie-Hellman key agreement provides a way to generate keys one is public key and another one is private key. This situation can be addressed by adding some forms of authentication mechanisms to the protocol, as proposed by Diffie-Hellman key agreement can only support two or more participants. We introduce the symmetric balanced incomplete block design (SBIBD) in designing the key agreement protocol to reduce the complexity of communication and computation.

1.1 Main Contributions

In this project, we present an efficient and secure block design-based key agreement protocol by extending the structure of the SBIBD to support two or multiple participants, in which enables multiple data owners to freely share the data with high security and

efficiency. The SBIBD is constructed as the group of data sharing the model to the support group of data sharing in cloud computing. The protocol can be provides the authentication services and a fault tolerance property. The main contributions of this paper are summarized as follows.

Model of the group data sharing according to the structure:
In this paper, a group data sharing model is established based on the definition of the SBIBD, in which can be used to determine the way of communication among the multiple participants. The mathematical descriptions of the structure of the SBIBD, general formulas used for computing the common conference key for multiple participants are derived.

Fault detection and fault tolerance can be provided in the protocol:
We presented the protocol can be perform fault detection to ensure that a common conference key is established among all two or multiple participants without failure. The fault detection phase, a volunteer will be used to replace a malicious participant to support the fault tolerance property. The volunteer enables to the protocol to resist different key attacks which makes the group data sharing in cloud computing is more secure.

Secure group data sharing in cloud computing:
According to the data sharing model applying the SBIBD, multiple participants can form a group to efficiently share the outsourced data. Note that the common conference key is only produced by group members. Attackers or the semi-trusted cloud server has no access to the generated key. Thus, they cannot access the original outsourced data. Therefore, the proposed key agreement protocol can support secure and efficient group data sharing in cloud computing. The field of applications of the key agreement protocol by applying an SBIBD with high security and flexibility.

1.2 ORGANIZATION
The remainder of this paper is organized as follows. Section 2 introduces related works. Section 3 briefly presents preliminaries and the system model. Section 4 describes the system architecture and adversary model. Section 5 describes the construction of group data sharing model. Section 6 shows the block design based key agreement protocol with the general formulas for calculating the common conference key for multiple participants. Section 7 and Section 8 present the security analysis and performance analyses, respectively. Finally, conclusions are drawn in Section 9. To understand our protocol well, the detailed process of the key agreement.

2. RELATED WORKS
It is well known that data sharing in cloud computing can provide scalable and unlimited storage and computational resources to individuals and enterprises. Cloud computing also leads to many security and privacy concerns, such as data integrity, confidentiality, reliability, fault tolerance and so on. The key agreement protocol is one of the fundamental cryptographic primitives, which can provide secure communication among multiple participants in cloud environments. However, encryption keys should be transmitted in a secure channel, particularly in the open cloud environment. Since it was introduced in resistance to compromised keys has been taken into consideration, which an important issue in the context of cloud is computing. Note that cloud storage auditing with verifiable outsourcing of key updates paradigm was proposed to achieve resistance to compromised keys. In this paradigm, the third party auditor (TPA) takes responsibility for the cloud storage auditing and key updates. In particular, the TPA is responsible for the selection and distribution of the key.
The key downloaded from the TPA can be used by the client to encrypt files that will be uploaded in the cloud. In a key agreement algorithm was exploited to achieve data access

when data are controlled by multiple owners. Therefore, the key agreement protocol can be applied in group data sharing to solve related security problems in cloud computing. In a Public Key Infrastructure (PKI) is used to circumvent man-in-the-middle attacks. In these protocols are not suitable for resource-constrained environments since they require executions of time-consuming modular exponentiation operations. Key agreement protocols that use elliptic curve cryptography (ECC). Moreover, based on the difficulty of solving the elliptic curve discrete logarithm problem (ECDLP), protocols that use ECC are more secure. To avoid the requirement of the public key certificate, in identity-based cryptography (IBC).

In this protocol, to manage the complexity of definitions and proofs for the authenticated group Diffie-Hellman key exchange, a formal model was presented, where two security goals. Diffie-Hellman key exchanges were addressed. However, some security properties are missing in which are essential for preventing malicious protocol participants. Note that all the above protocols have been proven and analyzed for security, but some of them can only be applied to the key agreement between two entities and need a large amount of resources to perform calculations. The key agreement protocol is applicable to support data sharing in cloud computing for the following reasons.

1. The generation of a common conference key is performed in a public channel, which is suitable for cloud computing environments.

2. The key agreement protocol can support and provide secure data sharing for multiple data owners within a group, where the data sharing follows a many-to-many pattern.

3. The key agreement protocol is based on a decentralized model, where a trusted third party is not required. We design a block design-based key agreement protocol for data sharing in cloud computing. First, we propose an algorithm to construct the (v; k + 1; 1)-design. Then, with respect to the mathematical description of the structure of the (v; k+1; 1)-design, general formulas for generating the common conference key K for multiple participants are derived. We believe that our contributions can widen the application scope of the key agreement protocol.

3. PRELIMINARIES AND SYSTEM MODEL

3.1 SECURITY ASSUMPTION

Security is one of the most essential conditions that a good cryptographic algorithm or protocol should first meet. Studies on safety issues can boil down to the security model.

The attacker's ability and the goal of security achieved can be well reflected by the correct and appropriate security model. Note that the security of our protocol relies on a variant of the computational Diffie-Hellman (CDH) .The bilinear Diffie-Hellman (BDH) assumption, which is defined as follows. According to the proof, the presented protocol can resist both passive attacks and active attacks. Many formal security analyses of the key agreement protocol can be found.

Block Design and (v; k + 1; 1) **Design**

In combinatorial mathematics, a block design is a set together with a family of subsets whose members are chosen to satisfy some set of properties that are deemed useful for a particular application.

Definition

Let V = f0; 1; 2:::v 1g be a set of v elements and B = f B0;B1;B2:::B1 be a set of b blocks, where Bi is a subset of

V and j Bi, j = k. it is a BIBD, which is called a (b; v; r; k;) design.

1. Each element of V appears in exactly r of the b blocks.

2. Every two elements of V appear simultaneously in exactly of the b blocks.

3. Parameters k and v of V meet the condition of k < v. Thus, no block contains all the elements of the set V.

4. Parameters b and v of V meet the condition of b < v. The case of equality is called a symmetric design.

Here, v is the number of elements of V, b denotes the number of blocks, k implies the number of elements in each block, and r and _ are the parameters of the design. It is also called a (v; k; _)-design.

In this paper, we require a (v; k + 1; 1)-design to construct our group data sharing decentralized model, where k is a prime number and = 1. Note that information exchange in our key agreement protocol is based on the (v; k + 1; 1)-design. Consequently, each participant can determine the intended message receivers or message senders based on the group data sharing model constructed by the (v; k + 1; 1)-design.

4. SYSTEM ARCHITECTURE AND ADVERSARY MODEL

SYSTEM ARCHITECTURE

The system model of our group data sharing scheme in cloud computing is illustrated in Fig. 1.the cloud to partially two types one is public cloud and another one is private cloud. A TPA, cloud and users are involved in the model, where the TPA is responsible for cloud storage authentication and authorization generating the system parameters. To create the three categories creator, reader and writer. Creator part is used to create the data or store the data in cloud.

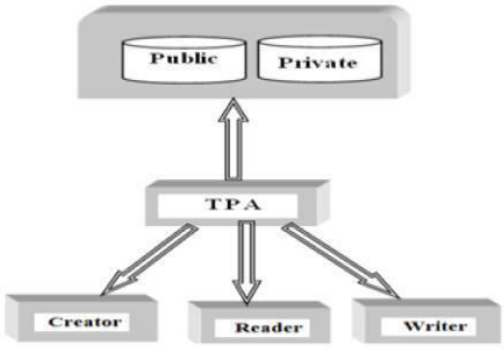
Reader part is to read the file and finally writer part is to write the data or store the data in cloud. The public cloud using all the members is seen without key but the private cloud fully secured, request to the admin. The admin to generate the key to provide the requesting user, and the user to open the file.

All Group members to upload the data in cloud. The cloud storage authentication will be provided, TPA to generate the conference key, and the admin to provide key to all users. The cloud, who is a semi trusted party, provides users with data storage services. Users can be individuals or multiple participants.

To work together, they form a group, upload data to the cloud server and share the outsourced data with the group members. Moreover, the group data sharing model is based on the SBIBD, where a trusted third party is not required. Two types of adversaries may be involved in the protocol, passive adversaries and active adversaries.

A passive adversary is a person who attempts to learn information about the conference key by eavesdropping on the multicast channel, whereas an active adversary is a person who attempts to impersonate a participant or disrupt a conference.

Note that the generation and update of the key are accomplished by the participants. Moreover, with the fault tolerance property of our protocol, the participants are able to ascertain the correctness of the common conference key.



ADVERSARY MODEL

The adversary model determines the capabilities and possible actions of the attacker.

1. The adversary reveals a long-term secret key of a participant in a conference and then impersonates others to this participant.
2. The adversary reveals some previous session keys and then learns the information about the session key of a fresh participant.
3. The adversary reveals the long-term keys of one or more participants in the current run.
4. A malicious participant chooses different sub keys, generates different signatures and broadcasts the messages to the corresponding participants.

WORKING METHODOLOGY

The working methodology can be best illustrated with architecture. Earlier in the

working principle we explained technical implementation.

Step 1: The participants have to sign up to become the participants of any group. In any social media the administrator only invites the participants. But in our project anyone who is willing to be the participant of that group can sign up with his mail id. This will be sent as a request to the administrator. At the time of sign up itself the participants have to mention their domains of interest on the topics which they want to share the data. Based on the domains of interest different keys will be assigned.

Step 2: On seeing the request from the participants, the Administrator (Certifying Authority) assigns a unique key to the participant based on his/her domains of interest. The participants can select multiple domains at a time. In our project we limited the domain interest to 2.

Step 3: Once the participant is authorized by the admin with a key, he can login into his domain within the group and can upload and download the data relevant to his domain.

Step 4: All the participants can access the files located in the common cloud without the need of key. This area is called common cloud. We segregate the server into 2 types as public cloud and private cloud. Common cloud is otherwise called public cloud. The secured information are stored in the private cloud. The participants with relevant domain key alone can access the data from their private cloud shared by other participants of their domain.

Step 5: If any participant wants to access the data which is not relevant to their domain, then he has to make a request to the Admin. On seeing the credentials, the admin authorize him with a OTP which is a time bound. He can view the data but he cannot modify or upload the data in that private cloud. If he wants to access the data without the knowledge of the Admin, then an alert message will be sent to the Admin. The Admin can decide on whether to retain the participant or delink him from the group. This provides security to the data in the cloud based on the key agreement protocol.

WORKING PRINCIPLE

Role of Certifying Authority

A TPA, cloud and users are involved in the model, where the TPA is responsible for cloud storage auditing, fault detection and generating the system parameters. The cloud, who is a semi-trusted party, provides users with data storage services and download services. To work together, they form a group, upload data to the cloud server and share the outsourced data with the group members.

Moreover, the group data sharing model is based on the SBIBD, where a trusted third party is not required. With respect to this model, all the participants exchange messages from intended entities according to the structure of the SBIBD to determine a common conference key. Two types of adversaries may be involved in the protocol, a person who attempts to impersonate a participant or disrupt a conference.

5. THE CONSTRUCTION OF THE GROUP DATA SHARING MODEL Design of data sharing in a group

Step 1: Step 1 describes transformations of the first k + 1 blocks of $\{S_0, S1, S2,..., S_k\}$ in B to the first k + 1 blocks in E.

Step 2: Transformations of step 2 are based on Step 1; in S_0 with (k + 1) (k + 1) elements, element 0 appears k+1 times in the first column of S_0 and the remaining k_2 + k elements $\{1,2,...,k_2 + k\}$ appear exactly once in S_0 in order.

Step 3: The transformations of step 3 are based on Step 2; in sector S_x (x 6= 0) with k blocks, the set of the k elements of the x^{th} column is equal to the index set of the k blocks in S_x.

DESIGN OF KEY

Generation of a (v, k + 1, 1)-design

In a (v, k+1, 1)-design, v denotes the number of participants and the number of blocks. Every block embraces k + 1 participant, and

every participant appear k + 1 times in these v blocks. Furthermore, every two participants appear simultaneously in exactly one of the v blocks. First, a prime number k is selected. Then, the number of participants is determined by the value of k, which is computed as $v = k^2 + k + 1$. Finally, according to Definition , $V = \{0,1,2,...,v-1\}$ represents the set of v participants, whereas $B = \{B_0,B_1,B_2,...,B_{v-1}\}$ implies v blocks constituted by these v participants. Note that the block is defined as $B_i = \{B_i,0,B_i,1,B_i,2,...,B_i, k\}$, which means each block embraces k+1 participants, and $B_{i,j}$ denotes which participant is contained in the j^{th} column of the i^{th} block. Sometimes we will consider blocks organized as a matrix in which column j is composed by elements $B_{i,j}$ for i = 0,1,2,...,k and row i is composed by elements $B_{i,j}$ for j = 0,1,2,...,k.

The structure of the (v,k + 1,1)-design is constructed by Algorithm 1, which outputs numbers $B_{i,j}$ for i = 0,1,...,k_2 +k and j = 0,1,...,k. Algorithm 1 can directly determine which participant should be involved in each block.

ALGORITHM

Generation of a (v; k + 1; 1)-design

```
for (i = 0; i < k; i + +) do for( j
      = 0; j < k; j + +) do
        if (j == 0 )then
         Bi,j = 0;
     else
        Bi,j = ik + j;
      end if
    end for
end for
   for (i = k + 1; i <k2 + k; i + +)
     do for (j = 0; j _ k; j + +) do
       if (j == 0) then
           Bi,j = b(i   1) =kc;
  else
Bi,j = j(k+1+MODk(i  j+(j  1) b(i    1)=kc);
       end if
     end for
  end for
```

Algorithm is an optimization of the algorithm in and the proof of the correctness follows the same lines than the structure created by Algorithm 1 can be proven to satisfy the conditions of the (v; k + 1; 1)-design, which means that each participant of V appears exactly k + 1 times in B and that each pair of participants of V appears exactly once in B.

These properties can be utilized to design the group data sharing model, which can diminish the communication cost of the proposed protocol. The detailed process of the protocol and the corresponding performance analysis based on the model can be found in Section respectively.

Design of the Group Data Sharing Model

Algorithm, the structure B of the (v; k + 1; 1)-design is constructed for v participants, which satisfies the properties of an SBIBD. However, to generate a common conference key for the v participants, the structure of the (v; k+1; 1) design should have the property that each block Bi embraces participant i. Here the block of the structure of the (v; k + 1; 1)-design, and the order of the appearance of these v blocks is represented by i.

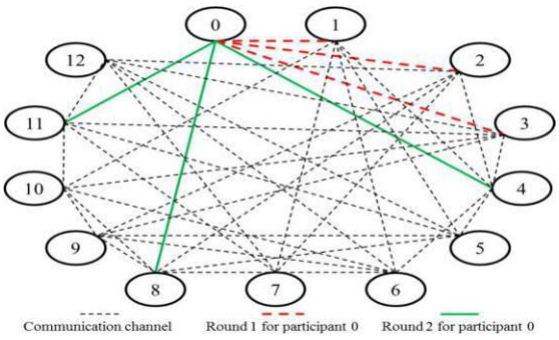


Fig.2: design group data sharing model can be determined by the mathematical descriptions in the four different cases. A concrete example can be found in Appendix, where the structure of design is constructed. In our protocol, two rounds are required to generate a common conference key. The group data sharing model can determine which participants are the intended message senders of participant i.

The group data sharing model is established as follows. If j 2 Ei, participant j is the intended message sender of participant i in Round 1.

If i 2 Ej, participant j is the intended message sender of participant i in Round 2.

Based on the group data sharing model, every participant can receive messages from their intended message senders after two rounds of the key agreement. After the construction of the group data sharing model, the block design-based key agreement protocol is designed for data sharing in cloud computing.

7. A BLOCK DESIGN-BASED KEY AGREEMENT PROTOCOL

Initial Phase

In the protocol, a TPA takes responsibility for generating some system parameters and distributing the private key for all participants.

Definition

In The structure of E of a (v; k + 1; 1)-design are two hash functions, which map its arbitrary length to a non zero point of G1 and nonzero integer, respectively. In our block design-based key agreement protocol, participant is a public key and private key are mapped as H1(IDi) and Si = sH1(IDi), respectively. Moreover, to provide authentication, based on the RSA cryptographic algorithm, the TPA selects a public key ei and a private key di for each participant and distributes (ei; n) to all the participants, where n is the product of two large prime numbers.

Subsequently, participant i computes Yi = H2(IDi), and Xi = (Yi)di and keeps (di;Xi) secret.

Key Agreement Phase

In the key agreement phase, two rounds are required for generating a common conference key for multiple participants, and the way of message exchanges is with respect to the group data sharing model established by the structure E of the (v; k + 1; 1)-design.

Case1: Participant 0 needs to receive messages from participant j(1< j < k).

Case2: For participant i(i k), they need to receive messages from participant

J (j = mk + 1 +MODk(i - 1)(m - 1); j 6= i).

Case3: For participant i(i = Em,m), they need to receive messages from participant 0 and participant j([j = b(i-|1)/k] k + m,j = i).

8. SECURITY ANALYSIS

The security of our protocol is based on the BDH assumption. In this section, we prove that our protocol is secure against passive attacks and active attacks.

Security against Passive Attacks

In our protocol with v participants, a participant and a volunteer in the protocol are a probabilistic polynomial-time turing machine, as is an adversary. A passive adversary is the person who attempts to learn information about the conference key by eavesdropping on the multicast channel.

Security against Active Attacks

In an active attack, an adversary not only learns information about the conference key but also replays, forges and delays the messages. To resist active attacks, desired properties for a practical key agreement protocol.

Key comprise impersonation

Our protocol can withstand the key comprise impersonation attack, in which the adversary impersonates a legal conferee of participant i. In our protocol, long term secret keys of participants are independent of each other with respect to real identities of participants. Therefore, with the long-term secret key (Si), the adversary still cannot learn any information about long-term secret keys of other participants.

Known session key

The known session key prevents the session key held by a fresh participant from being compromised by an adversary, even if the adversary has learned some previous session keys.

Perfect forward security

A protocol offers perfect forward security if the compromising of long-term keys (Si) during the communication among multiple participants cannot result in the compromising of the previous session key (K). In our protocol, the previous session key.

Key confirmation

If a participant is assured that its counterparts actually have possession of a particular secret key, the protocol provides key confirmation.

Each participant can ensure that its counterparts actually have possession of a common conference key K. Therefore, the presented protocol can provide key confirmation. Moreover, the presented protocol can resist denial of service attacks. Note that the presented protocol is contributory. Participant in our protocol equally contributes to the common conference key and guarantees the freshness of the key

9. PERFORMANCE ANALYSIS AND EVALUATION

Performance Analysis

Generally, the performance of a key agreement protocol consists of communicational and computational efficiency. In each round of our protocol, each participant has to receive k messages from the intended k participants according to a (v; k +1; 1)-design of the SBIBD. Computational complexity is composed of pairing computations, point multiplications and modular exponentiations, whereas communication complexity is composed of the number of participants.

10. CONCLUSION

With the swift development in the technology of the World Wide Web and crypt techniques, data sharing in the group in cloud computing in a secured and efficient method has opened up many scopes in the area of networking. Using the conference key agreement protocol, the security and efficiency of group data sharing in cloud computing is improved in our project. To be more precise, the outsourced data of the data owners are encrypted by the common conference keys which are protected from the attacks of adversaries. Compared with conference key distribution, the conference key agreement has qualities of higher safety and reliability. However, the conference key agreement asks for a large amount of information interaction in the system and more computational cost.

To overcome the problems in the conference key agreement, the SBIBD is employed in the protocol design. In this project, we present a novel block design-based key agreement protocol that supports group data sharing in cloud computing. Due to the definition and the mathematical descriptions of the structure multiple participants can be involved in the protocol and general formulas of the common conference key for participants are derived. Moreover, the introduction of volunteers enables the presented protocol to support the fault tolerance property, thereby making the protocol more efficient and secure.

11. FUTURE ENHANCEMENTS

In our future work, we would like to expand our protocol to provide more properties (e.g., anonymity, traceability) to make it applicable for a variety of environments.

12. REFERENCES

[1] Barua.R, Dutta.R, and Sarkar.P, “Extending joux’s protocol to multi party key agreement (extended abstract).” Lecture Notes in Computer Science, vol. 2003, pp. 205–217, 2003.

[2] Blakewilson.S, Johnson.D, and Menezes.A, “Key agreement protocols and their security analysis,” in IMA International Conference on Cryptography and Coding, 1997, pp. 30–45.

[3] Chen.F, Xiang.T, Yang.Y, and Chow.S.S.M, “Secure cloud storage meets with secure network coding” in IEEE INFOCOM, 2014, pp. 673–681.

[4] Chung.Ib and Bae.Y, “The design of an efficient load balancing algorithm employing blockdesign,”JournalofApplied Mathematics and Computing,vol.14, no. 1, pp. 343–351, 2004.

[5] Diffie.W and Hellman.M.E, “New directions in cryptography,” IEEE Transactions on Information Theory, vol. 22, no. 6, pp. 644–654, 1976.