

# Key Agreement Protocol For Group Data Sharing In Cloud Computing

Vaishali K. Sarkate<sup>1</sup>, Dr.Ranu Tuteja<sup>2</sup>

<sup>1</sup>Student, Dept. of Computer Science and Engineering, PRMIT&R, Badnera, Maharashtra, India

<sup>2</sup>Professor, Dept. of Computer Science and Engineering, PRMIT&R, Badnera, Maharashtra, India

\*\*\*

**Abstract** - Data sharing in cloud computing enables multiple participants to freely share the data in a group, which improves the efficiency of work in cooperative environment. Now a days it is a formidable challenge that how to ensure the security of the data sharing in a group, and how it efficiently share the data to other group members. So that we use a key agreement protocol is used for securely and efficiently share the data in a group. In this paper, The generation of a common conference key is performed in a public channel, suitable for cloud computing environments.

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. Compared with the one-to-many pattern, the many-to-many pattern in group data sharing provides higher efficiency in the environment of cooperative storage.

The key agreement protocol is based on a decentralized model, where a trusted third party is not required. This means that every data owner in a group fairly contributes and determines the common conference key such that the outsourced data are controlled by all the data owners within a group.

**Key Words:** Cloud Computing, Data Sharing, key agreement protocol

## 1. INTRODUCTION

Cloud computing and cloud storage have become hot topics in recent decades. Both are changing the way we live and greatly improving production efficiency in some areas. At present, due to limited storage resources and the requirement for convenient access, we prefer to store all types of data in cloud servers, which is also a good option for companies and organizations to avoid the overhead of deploying and maintaining equipment when data are stored locally. The cloud server provides an open and convenient storage platform for individuals and organizations, but it also introduces security problems. For instance, a cloud system may be subjected to attacks from both malicious users and cloud providers. In these scenarios, it is important to ensure the security of the stored data in the cloud. In [1], [2], [3], several schemes were proposed to preserve the privacy of the outsourced data. The above schemes only considered the security problems of a single data owner. However, in some applications, multiple data owners would like to securely share their data in a group manner. Therefore, a protocol that supports secure group data sharing under cloud computing is needed.

### 1.1 Motivation

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, 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. Compared with the one-to-many pattern, the many-to-many pattern in group data sharing provides higher efficiency in the environment of cooperative storage.

3. The key agreement protocol is based on a decentralized model, where a trusted third party is not required. This means that every data owner in a group fairly contributes and determines the common conference key such that the outsourced data are controlled by all the data owners within a group.

### 1.2 Objectives

The key agreement protocol is applicable to support data sharing in cloud computing to achieve the following objectives.

1. To generate a common conference key is in a public channel, suitable for cloud computing environments.

2. To provide secure data sharing for multiple data owners within a group, where the data sharing follows a many-to-many pattern.

3. To determines the common conference key such that the outsourced data are controlled by all the data owners within a group.

## 2. LITERATURE SURVEY

### 2.1 Related Topics

It is well known that data sharing in cloud computing can provide scalable and unlimited storage and computational resources to individuals and enterprises. However, cloud computing also leads to many security and privacy concerns, such as data integrity, confidentiality, reliability, fault tolerance, and so on. Note that the key agreement protocol is one of the fundamental cryptographic primitives, which can provide secure communication among multiple participants in cloud environments.

F. Chen, T. Xiang, Y. Yang, and S. S. M. Chow[2] had designed a general construction of a secure cloud storage protocol based on any secure network coding protocol. However, it is

not known if a secure network coding protocol can be constructed from a secure cloud storage protocol. It is an interesting future work to consider under what condition this can be done.

D. He, S. Zeadally, and L. Wu[3] they proposed a new CLPA scheme. Compared with previously proposed schemes, our CLPA scheme not only can address the security problems in TPKC-based public auditing schemes and ID-based public auditing schemes but also yields better performance.

W. Diffie and M. E. Hellman IEEE Transactions on Information Theory proposed The basic version of the Diffie-Hellman protocol provides an efficient solution to the problem of creating a common secret key between two participants.

L. Law, A. Menezes, M. Qu, J. Solinas, and S. Vanstone, [6] show the comparison includes the basic two-pass protocols. The computational requirement is indicated by counting the number of exponentiations computed by each principal in the protocol run and this is the complexity. In [14] and [15], based on symmetric-key cryptography, several schemes were proposed to enable efficient encryption of the outsourced data. However, encryption keys should be transmitted in a secure channel, which is not possible in practice, particularly in the open cloud environment.

Cloud storage auditing with verifiable outsourcing of key updates paradigm was proposed by Yu et al. in [10] 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 he will upload to the cloud. In contrast, the generation and distribution of the key are based on a centralized model in [10], which not only imparts a burden to the TPA but also introduces some security problems.

In [11], a key agreement algorithm was exploited by De Capitani di Vimercati et al. 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. Following the first pioneering work for key agreement .

In [12], a public key infrastructure (PKI) is used to circumvent man-in-the-middle attacks. However, 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) have been proposed in [14], [15]. These protocols are more efficient than the protocols that resort to the PKI because point additions or multiplications in elliptic curves are more efficient compared with the modular exponentiation. Moreover, based on the difficulty of solving the elliptic curve discrete logarithm problem (ECDLP), protocols that use ECC are more secure.

An identity-based authenticated key agreement protocol was proposed by Shen et al. in [9], which improves the efficiency of the conference key agreement and provides entity authentication services. However, there are some obstacles in Shen et al.'s protocol [9] in real applications. One is that the protocol only discusses a specific situation when the number of conferees is exactly 7. The other is that the protocol does not discuss the general situation and does not provide the key agreement process for multiple participants, which makes the protocol lack flexibility and practicability.

### 2.2 Summary & Discussion

As a development in the technology of the Internet and cryptography, group data sharing in cloud computing has opened up a new area of usefulness to computer networks. With the help of the conference key agreement protocol, the security and efficiency of group data sharing in cloud computing can be greatly improved. Specifically, the outsourced data of the data owners encrypted by the common conference key are protected from the attacks of adversaries. Compared with conference key distribution, the conference key agreement has qualities of higher safety and reliability.

## 3. PROPOSED SYSTEM ANALYSIS

### 3.1 Analysis

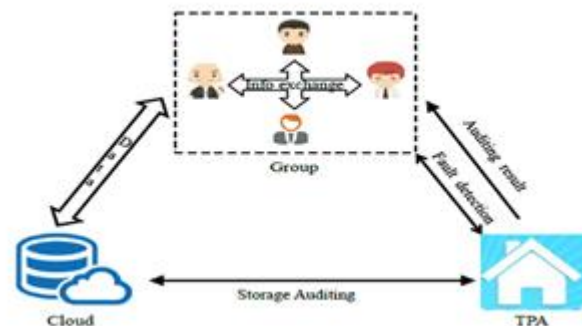


Figure 3.1: system model of data sharing in cloud computing

The system model of the group data sharing scheme in cloud computing is illustrated in Fig. 3.1. 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, which is a semi-trusted party, provides users with data storage services and download services. Users can be individuals or staff in a company. To work together, they form a group, upload data to the cloud server, and share the outsourced data with the group members. In practice, users can be mobile Android devices, mobile phones, laptops, nodes in underwater sensor networks, and so forth. Concerning this model, all the participants exchange messages from intended entities to determine a common conference key. In addition to participants, volunteers and adversaries are also included in the presented protocol, and all of them run as a probabilistic polynomial-time Turing machine. 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.

### 3.1.1 Problem Definition

Data sharing in cloud computing enables multiple participants to freely share the group data, which improves the efficiency of work in cooperative environments and has widespread potential applications. However, how to ensure the security of data sharing within a group and how to efficiently share the outsourced data in a group manner are formidable challenges. TO overcome this problem, key agreement protocols have played a very important role in secure and efficient group data sharing in cloud computing.

### 3.1.2 Requirement Analysis

- Operating System : Windows 7,8,10
- Technology Used: PHP
- Database Used : Mysql

## 3.2 Design

### 3.2.1 Proposed design

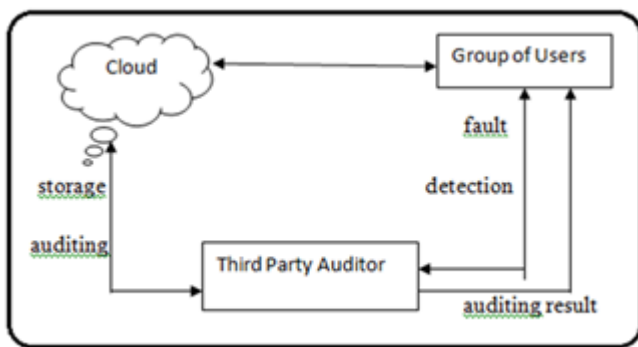


Figure 3.2: Architecture of Working system

Figure 3.2 shows that working system is divided into three parts:

I. User: Users can be individuals or staff in a company. To work together, they form a group, upload data to the cloud server, and share the outsourced data with the group members. In practice, users can be mobile Android devices, mobile phones, laptops, nodes in underwater sensor networks, and so forth. Concerning this model, all the participants exchange messages from intended entities to determine a common conference key. In addition to participants, volunteers and adversaries are also included in the presented protocol, and all of them run as a probabilistic polynomial-time Turing machine. 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.

II. Cloud: cloud, which is a semi trusted party, provides users with data storage services and download services.

III. TPA: TPA is responsible for cloud storage auditing, fault detection, and generating the system parameters.

### 3.2.2 Detail designed

#### Algorithms

#### 3-KEY Triple DES

Before using 3TDES, the user first generates and distributes a 3TDES key K, which consists of three different DES keys K1, K2, and K3. This means that the actual 3TDES key has a length of  $3 \times 56 = 168$  bits. The encryption scheme is illustrated as follows –

The encryption-decryption process is as follows –

- Encrypt the plaintext blocks using single DES with key K1.
- Now decrypt the output of step 1 using a single DES with key K2.
- Finally, encrypt the output of step 2 using a single DES with key K3.
- The output of step 3 is the ciphertext.
- Decryption of a ciphertext is a reverse process. The user first decrypts using K3, then encrypts with K2, and finally decrypts with K1.

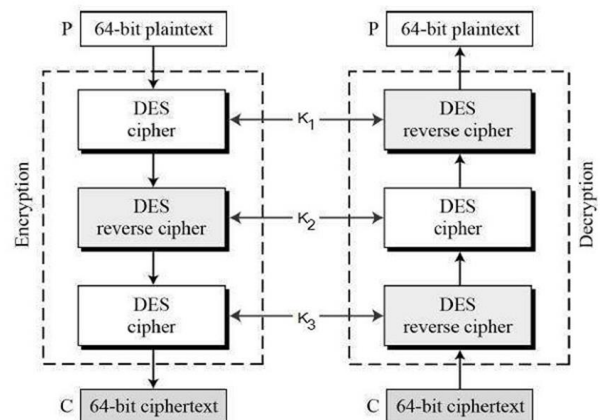


Figure 3.3 : Encryption Decryption Process

Due to this design of Triple DES as an encrypt–decrypt–encrypt process, it is possible to use a 3TDES (hardware) implementation for a single DES by setting K1, K2, and K3 to be the same value. This provides backward compatibility with DES.

The second variant of Triple DES (2TDES) is identical to 3TDES except that K3 is replaced by K1. In other words, the user encrypts plaintext blocks with key K1, then decrypt

with key K2, and finally encrypt with K1 again. Therefore, 2TDES has a key length of 112 bits.

Triple DES systems are significantly more secure than single DES, but these are clearly a much slower process than encryption using single DES.

#### Key Agreement Protocol

In cryptography, a key-agreement protocol is a protocol whereby two or more parties can agree on a key in such a way that both influence the outcome. If properly done, this precludes undesired third parties from forcing a key choice on the agreeing parties. Protocols that are useful in practice also do not reveal to any eavesdropping party what key has been agreed upon.

Many key exchange systems have one party generate the key, and simply send that key to the other party does not influence the key. Using a key-agreement protocol avoids some of the key distribution problems associated with such systems.

Protocols, where both parties influence the final derived key, are the only way to implement perfect forward secrecy.

Flowchart:

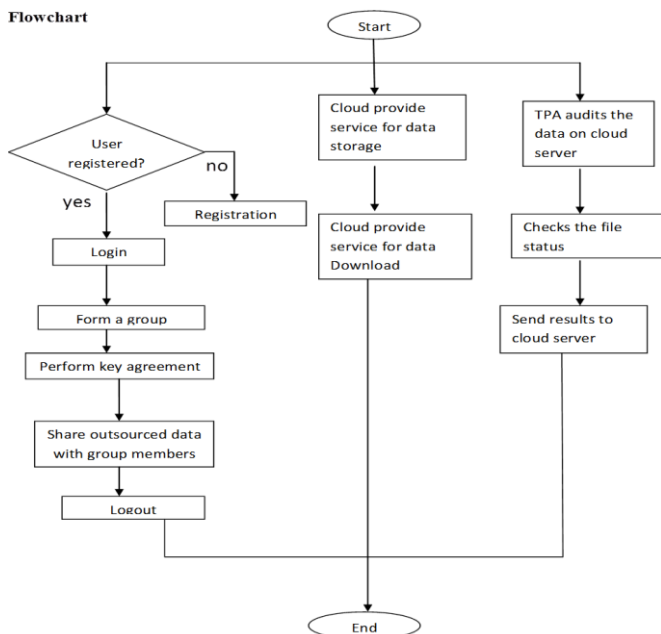


Figure 3.4: Flowchart

## 4. System Implementation & Testing

### 4.1 SETTING ENVIRONMENT

Client side and server side environment

Client side and server side are web development terms that describe where application code runs. Web developers will also refer to this distinction as the frontend vs. the backend.

client-server model

Much of the Internet is based on the client-server model. In this model, user devices communicate via a network with centrally located servers to get the data they need, instead of communicating with each other. End user devices such as laptops, smartphones, and desktop computers are considered to be 'clients' of the servers, as if they were customers obtaining services from a company. Client devices send requests to the servers for webpages or applications, and the servers serve up responses.

The client-server model is used because servers are typically more powerful and more reliable than user devices. They also are constantly maintained and kept in controlled environments to make sure they're always on and available; although individual servers may go down, there are usually other servers backing them up. Meanwhile, users can turn their devices on and off, or lose or break their devices, and it should not impact Internet service for other users.

Servers can serve multiple client devices at once, and each client device sends requests to multiple servers in the course of accessing and browsing the Internet.

#### 4.2 Implementation details

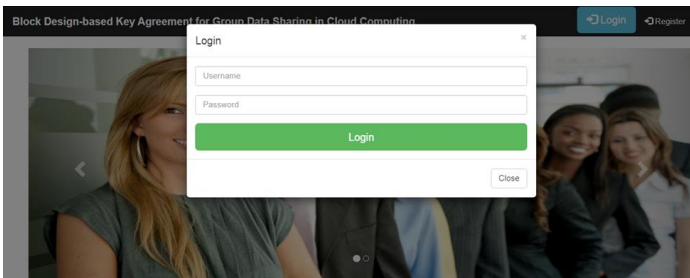
The working system of this dissertation is a web base application. It is implemented by using HTML, CSS, JAVA SCRIPT, jQuery, PHP, and Bootstrap.

#### 4.3 System Execution details



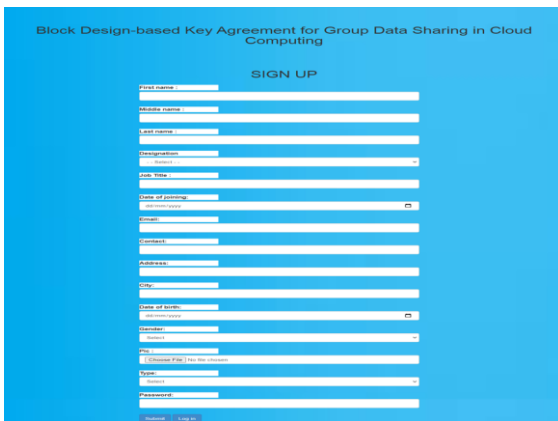
Screenshot 4.1: Home Page

- Screenshot 4.1 shows the home page which is the index page of the existing system.
- It shows links for login and registration on the upper right side in the navigation bar.



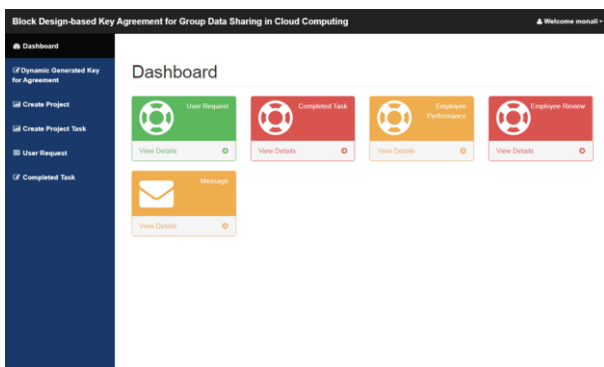
screenshot 4.2: Login form

- When the user clicks on the login link on home page, this page gets opened.
- Login form required of two fields named username and password.
- Only registered users are allowed to fill login form.



Screenshot 4.3: Registration Page

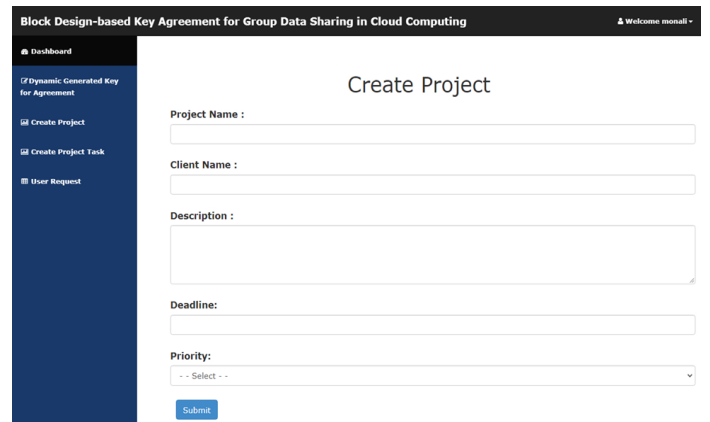
- When the user clicks on the registration link on home page, this page gets opened.
- Registration form required information of the user such as first name, last name, email, password, contact number, address, and so on.
- If there is a new user to the system then first of all they have to register themselves.



Screenshot 4.4: Admin Dashboard

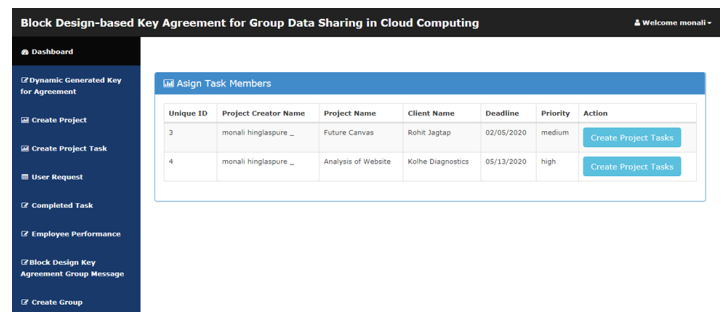
- Screenshot 4.4 shows the admin dashboard.
- Once the admin completed their login, this admin dashboard gets opened.

- Admin dashboard consists of five panels and they are user request, completed task, employee performance, message.



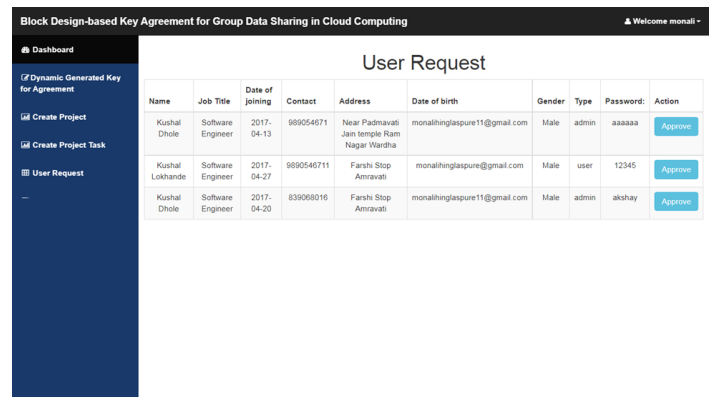
Screenshot 4.5: Create Project

- When the admin clicks on to create the project on the left panel of the dashboard, this form gets opened.
- When there is a new project for an organization, then the first admin fills tasks and information about the new project in this form.



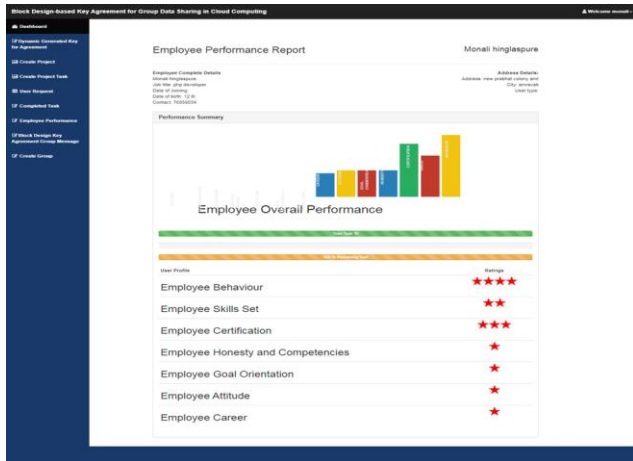
Screenshot 4.6: Project List

- When the admin clicks on create project task, this list gets opened.
- This page consists of a list of the entire project in the organization with creating project task button.



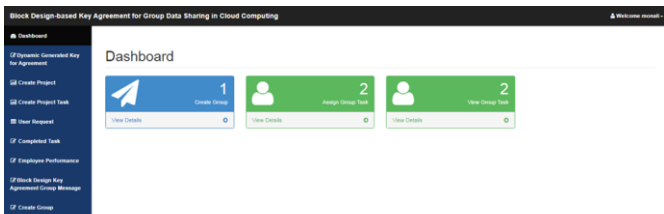
Screenshot 4.8: Employee User List

- When the admin clicks on the user request panel, this list gets opened.
- Admin approves the user request by click on approve button.
- When admin clicks on create Completed Task panel this list gets opened.
- It consists of project completion details.



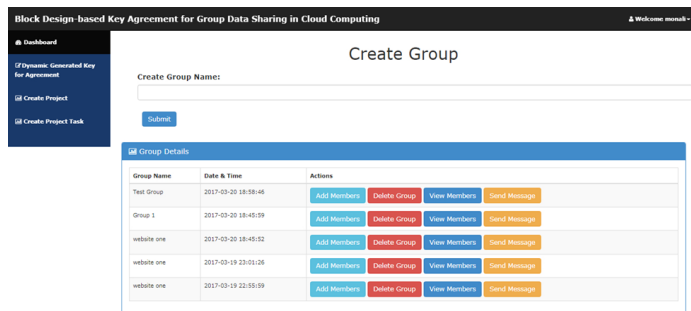
Screenshot 4.10: Employee Performance

- When the admin clicks on the employee performance panel, this graph gets opened.
- This page displayed the overall performance of an employee.



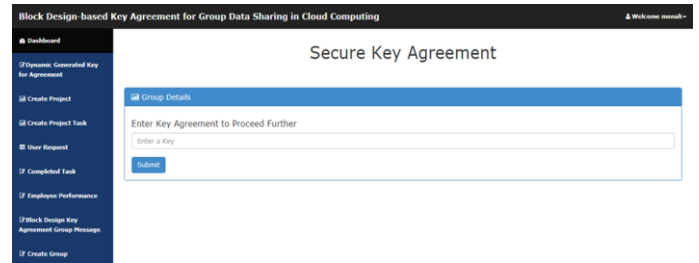
Screenshot 4.11: Block Design Group Dashboard

- When the admin needs to perform block design-related tasks this dashboard gets opened.
- It consists of three panels, create group, arrange group task and view group task.



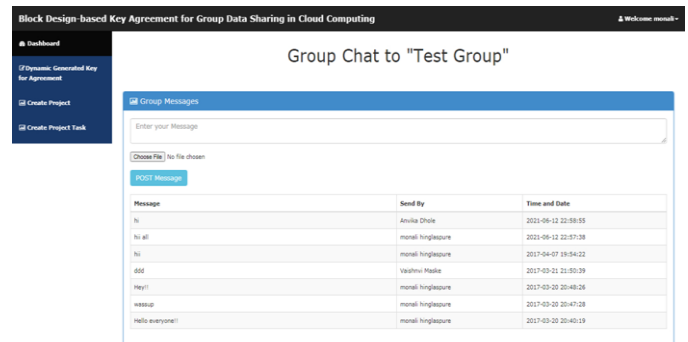
Screenshot 4.12: Group Customization and Message Send

- When the admin needs to create a group for conversation, then they create a new group by click on create group button.
- This page consists of a list of all the groups in an organization.



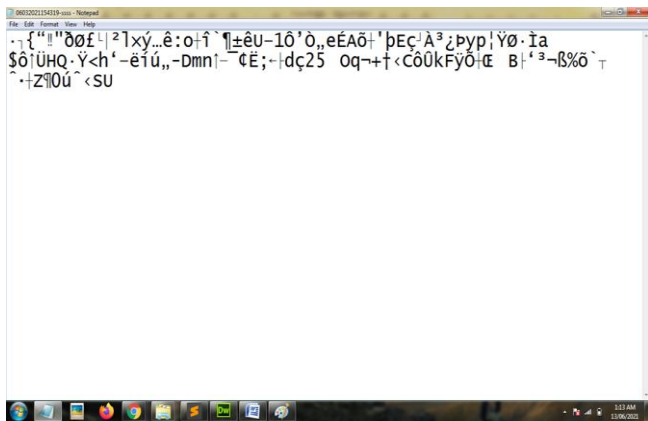
Screenshot 4.13: Key Agreement to chat further

- Once the group is created, this page gets opened.
- All the members of the group have to do the key agreement before entering the group.
- Admin can add members by click on add member button and remove members by click on the remove from group button.
- Screenshot 4.14 shows the list of members in the group.
- Admin can view the list of all the members of a particular group on this page.



Screenshot 4.16: Message in the group

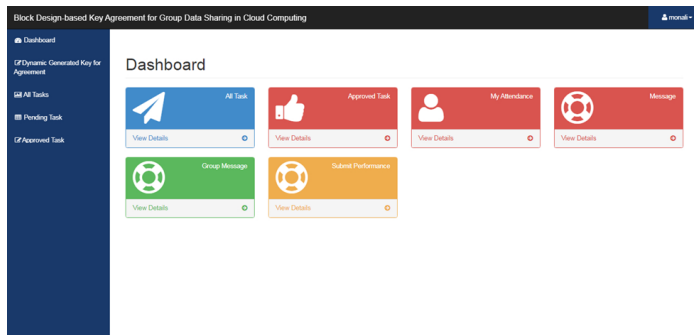
- User can type their message in the message box, and the messages of another user in the group are given in the figure.
- User can send the message by click on the post button.
- If Users want to send any document then they first select the document by click on choose file and click on post.



Screenshot 4.17: Encrypted Message in the file

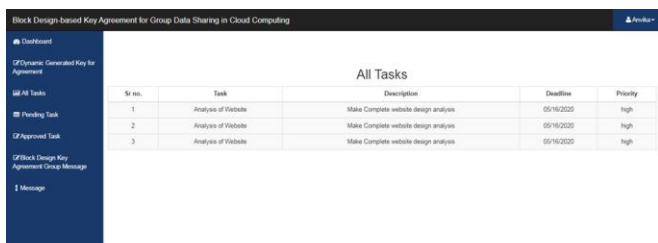
- Once the message is sent, the message gets encrypted as shown above.

Employee Section



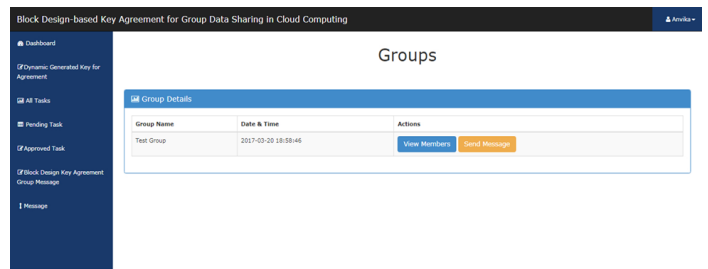
Screenshot 4.18: Employee Dashboard

- Once an employee is authenticated, this dashboard gets opened.
- It consists of six panels on the right side and some links on the left side.
- The panels are all task, approved task, my attendance, message, group message, and student performance.



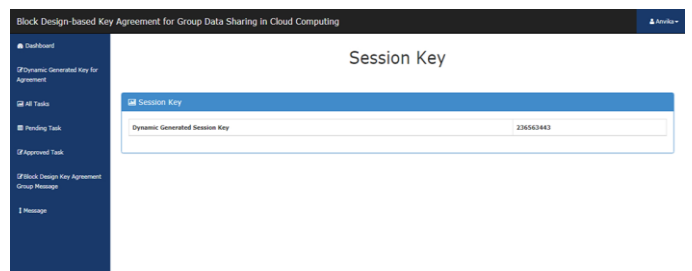
Screenshot 4.19: Employee allotted task

- When an employee clicks on approve task, this page gets open.
- This page consists of all the allotted tasks of a single employee.



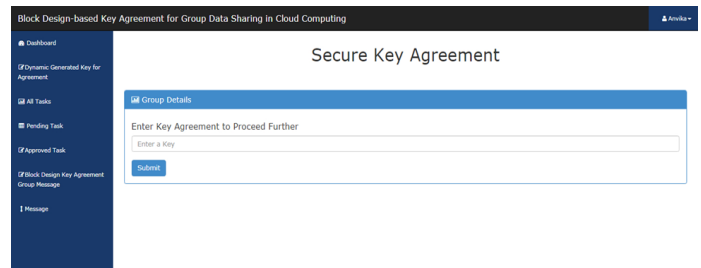
Screenshot 4.21: Employee Group Message

- When employee clicks on group messages, then the list of groups in which the employee is involved gets displayed.



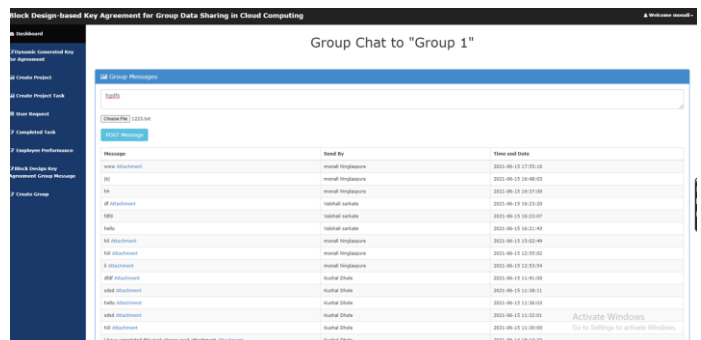
Screenshot 4.22: Dynamic Generated Key Agreement

- When employee clicks on send message button of group message page, then the key is generated for key agreement.



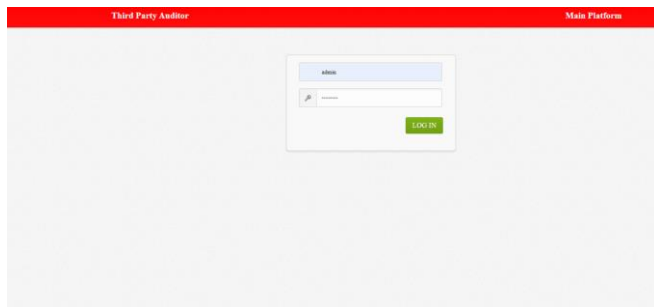
Screenshot 4.23: Secure Agreement Key Interchange

- Before sending the message to another user, the employee has to do the key agreement.



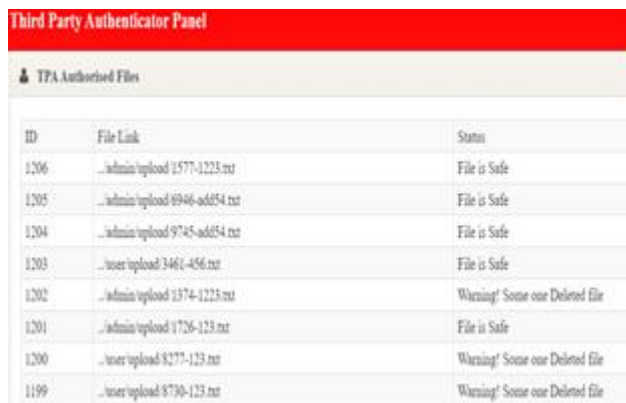
Screenshot 4.24: Message chat communication

- Once employees had done a key agreement then they are allowed for group chat.
- Employee can type a message in the textbox and clicks on the post button to send the message.
- If an employee wants to share a file then they can select the file by click on choose file and then click on post message to send the file.
- If employees want to send a personal message to another user, they can send the message by clicking on the message button.



Screenshot 4.26: TPA Scan

- Fig shows login form for TPA to enter the system, First TPA has to authenticate themselves.



ID	File Link	Status
1206	...admin/upload/12577-1223.txt	File is Safe
1205	...admin/upload/6946-ad654.txt	File is Safe
1204	...admin/upload/9745-ad654.txt	File is Safe
1203	...user/upload/3461-456.txt	File is Safe
1202	...admin/upload/1374-1223.txt	Warning! Some our Deleted File
1201	...admin/upload/1726-123.txt	File is Safe
1200	...user/upload/8277-123.txt	Warning! Some our Deleted File
1199	...user/upload/8730-123.txt	Warning! Some our Deleted File

Screenshot 4.27: TPA Scan

- Fig shows list of files audited by TPA. It includes file Id , Link of File, And status of file Whether it is safe or deleted of some changes have made in file then alert message has send.

## 5. Conclusion & Future scope

As a development in the technology of the Internet and cryptography, group data sharing in cloud computing has opened up a new area of usefulness to computer networks. With the help of the conference key agreement protocol, the security and efficiency of group data sharing in cloud computing can be greatly improved. Specifically, the outsourced data of the data owners encrypted by the common conference key are protected from the attacks of adversaries. Compared with conference key distribution, the conference key agreement has qualities of higher safety and reliability.

The Diffie Hellman Key and Encryption Techniques will be used with all types of media. Diffie Hellman Key will be used with Elliptical Curve Cryptography using Encryption and Decryption.

## REFERENCES

- [1] F. Chen, T. Xiang, Y. Yang, and S. S. M. Chow, "Secure cloud storage meets with secure network coding," in IEEE INFOCOM, 2014, pp. 673–681.
- [2] D. He, S. Zeadally, and L. Wu, "Certificateless public auditing scheme for cloud-assisted wireless body area networks," IEEE Systems Journal, pp. 1–10, 2015.
- [3] W. Diffie and M. E. Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, vol. 22, no. 6, pp. 644–654, 1976.
- [4] J. Shen, H. Tan, S. Moh, I. Chung, and J. Wang, "An efficient rfid authentication protocol providing strong privacy and security," Journal of Internet Technology, vol. 17, no. 3, p. 2, 2016.
- [5] L. Law, A. Menezes, M. Qu, J. Solinas, and S. Vanstone, "An efficient protocol for authenticated key agreement," Designs Codes and Cryptography, vol. 28, no. 2, pp. 119–134, 2010.
- [6] J. Shen, S. Moh, and I. Chung, "Identity-based key agreement protocol employing a symmetric balanced incomplete block design," Journal of Communications and Networks, vol. 14, no. 6, pp. 682–691, 2012.
- [7] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: Improved definitions and efficient constructions," Journal of Computer Security, vol. 19, no. 5, pp. 79–88, 2011.
- [8] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 1, pp. 222–233, 2014.
- [9] J. Yu, K. Ren, C. Wang, and V. Varadharajan, "Enabling cloud storage auditing with key-exposure resistance," IEEE Transactions on Information Forensics and Security, vol. 10, no. 6, pp. 1–1, 2015.
- [10] J. Yu, K. Ren, and C. Wang, "Enabling cloud storage auditing with verifiable outsourcing of key updates," IEEE Transactions on Information Forensics and Security, vol. 11, no. 6, pp. 1–1, 2016.
- [11] S. D. C. D. Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, "Encryption policies for regulating access to outsourced data," Acm Transactions on Database Systems, vol. 35, no. 2, pp. 78–78, 2010.
- [12] H. Guo, Z. Li, Y. Mu, and X. Zhang, "Cryptanalysis of simple three-party key exchange protocol," Computers and Security, vol. 27, no. 1-2, pp. 16–21, 2008.
- [13] Z. Tan, "An enhanced three-party authentication key exchange protocol for mobile commerce environments," Journal of Communications, vol. 5, no. 5, pp. 436–443, 2010.



[14] Y. M. Tseng, "An efficient two-party identity-based key exchange protocol." *Informatica*, vol. 18, no. 1, pp. 125–136, 2007.

[15] A. Shamir, "Identity-based cryptosystems and signature schemes," *Lecture Notes in Computer Science*, vol. 21, no. 2, pp. 47–53, 1985.