International Journal of Scientific Research in Science and Technology
Print ISSN: 2395-6011 | Online ISSN: 2395-602X (www.ijsrst.com)
doi : https://doi.org/10.32628/IJSRST22954

Development of A Novel Block Design Based Key Agreement Protocol for Cloud Environment to Improve Efficient Performance and Security

Dr. Gangolu Yedukondalu[1], Guna Santhoshi[2], Karnati Durga[3], Kotha Chandrakala[4], Dr. Mahesh Kotha[5]
[1]Professor, CSE (AI&ML) Department, Vignana Bharati Institute of Technology, Hyderabad, India
[2]Asst Professor, IT Dept, G. Narayanamma Institute of Technology and Science for Women, Hyderabad, India
[3]Assistant professor, Department of CSE (DS), CMR Engineering College Hyderabad, India
[4]Assistant Professor, Information Technology, CMR Technical Campus, Hyderabad, India
[5]Assistant professor, CSE (AI&ML) Department, CMR Technical Campus, Hyderabad, India

Article Info
Volume 9, Issue 5
Page Number : 52-63

Publication Issue
September-October-2022

Article History
Accepted : 01 Sep 2022
Published : 08 Sep 2022

ABSTRACT

Cloud computing is one of the recent emerging technologies. Heavy data sharing among multiple users is an open issue. 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. In this paper we focused on security issues with the help of key agreement protocols to perform efficient data sharing in cloud environment. We proposed a novel block design based key agreement method in support of symmetric balanced incomplete block design (SBIBD). The main objective of this method is to supports multiple participants, which can flexibly extend the number of participants in a cloud environment according to the structure of the block design. Based on the proposed group data sharing model, we present general formulas for generating the common conference key K for multiple participants. In addition, the fault tolerance property of our protocol enables the group data sharing in cloud computing to withstand different key attacks, which is similar to Yi's protocol.

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

I. 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, ut 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], several schemes were proposed to preserve the privacy of the outsourced data. The above schemes only considered 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. A key agreement protocol is used to generate a common conference key for multiple participants to ensure the security of their later communications, and this protocol can be

applied in cloud computing to support secure and efficient data sharing. Since it was introduced by Diffie-Hellman in their seminal paper [4], the key agreement protocol has become one of the fundamental cryptographic primitives. The basic version of the Diffie-Hellman protocol provides an efficient solution to the problem of creating a common secret key between two participants. In cryptography, a key agreement protocol is a protocol in which two or more parties can agree on a key in such a way that both influence the outcome. By employing the key agreement protocol, the conferees can securely send and receive messages from each other using the common conference key that they agree upon in advance. Specifically, a secure key agreement protocol ensures that the adversary cannot obtain the generated key by implementing malicious attacks, such as eavesdropping. Thus, the key agreement protocol can be widely used in interactive communication

environments with high security requirements (e.g., remote board meetings, teleconferences, collaborative workspaces, radio frequency identification [5], cloud computing and so on).

The Diffie-Hellman key agreement [4] provides a way to generate keys. However, it does not provide an authentication service, which makes it vulnerable to man-in-the-middle attacks. This situation can be addressed by adding some forms of authentication mechanisms to the protocol, as proposed by Law et al. in [6]. In addition, the Diffie-Hellman key agreement can only support two participants. Subsequently, to solve the different key attacks from malicious conferees, who attempt to deliberately delay or destroy the conference, Yi proposed an identity-based fault-tolerant conference key agreement in [7]. Currently, many researches have been devoted to improving the security and communication efficiency of the key agreement protocol, which is covered in the literature [8] Note that in Chung and Bae's paper [12] and Lee et al.'s paper [13], block design is utilized in the design of an efficient load balance algorithm to maintain load balancing in a distributed system. Inspired by [12] and [13], we introduce the symmetric balanced incomplete block design (SBIBD) in designing the key agreement protocol to reduce the complexity of communication and computation. As far as we know, the work to design the key agreement protocol with respect to the SBIBD is novel and original.

II. MAIN CONTRIBUTIONS

In this paper, we present an efficient and secure block design-based key agreement protocol by extending the structure of the SBIBD to support multiple participants, which enables multiple data owners to freely share the outsourced data with high security and efficiency. Note that the SBIBD is constructed as the group data sharing model to support group data sharing in cloud computing. Moreover, the protocol can provide authentication services and a fault tolerance property. The main contributions of this paper are summarized as follows.

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

2. Fault detection and fault tolerance can be provided in the protocol. The presented protocol can perform fault detection to ensure that a common conference key is established among all participants without failure.

Moreover, in the fault detection phase, a volunteer will be used to replace a malicious participant to support the fault tolerance property. The volunteer enables the protocol to resist different key attacks [7], which makes the group data sharing in cloud computing more secure.

3. Secure group data sharing in cloud computing can be supported by the protocol. According to the data sharing model applying the SBIBD, multiple participants can form a group to efficiently share the outsourced data.

Subsequently, each group member performs the key agreement to derive a common conference key to ensure the security of the outsourced group 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 (i.e., they only obtain some unintelligible data). Therefore, the proposed key agreement protocol can support secure and efficient group data sharing in cloud computing. Notably, the above contributions substantially widen the field of applications of the key agreement protocol by applying an SBIBD with high security and flexibility. Moreover, the communication complexity is reduced without introducing
extra computational complexity.

III. LITERATURE SURVEY

F. Chen, T. Xiang, Y. Yang, and S. S. M. Chow [2] had designed a general construction of 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] had discussed Cloud-assisted WBANs, which are the integration of a cloud computing platform and WBANs, could bring major benefits (as we discussed earlier) over traditional WBANs. One of the major challenges of a cloud-assisted WBAN is to ensure the integrity of the medical data stored at a cloud server. The auditing technique is an efficient tool for checking the integrity of the data stored remotely. However, previous auditing schemes suffer from key management and key escrow problems. To address these challenges, 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. In addition, their proposed CLPA scheme is provably secure in a strong security model, making CLPA very suitable for use in cloud-assisted WBANs.

L. Law, A. Menezes, M. Qu, J. Solinas, and S. Vanstone, [6] shows the comparison includes the basic two-pass protocols. The computational requirements are indicated by counting the number of exponentiations computed by each principal in protocol run and this is the complexity. Also

H.Elkamchouchi, M.Eldefrawy works by computing and exchanging two vectors but the new one works and exchanges one value.

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.

In [16], it was introduced that resistance to compromised keys has been taken into consideration, which an important issue in the context of cloud is computing.

Cloud storage auditing with verifiable outsourcing of key updates paradigm was proposed by Yu et al. in [17] 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 is based on a centralized model in [17], which not only imparts a burden to the TPA but also introduces some security problems.

In [18], 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 [4], many works have attempted to provide authentication services in the key agreement protocol.
In [19], 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 [20]. 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.

To avoid the requirement of the public key certificate, in 1984, identity-based cryptography (IBC) was proposed by Shamir. However, it was not until 2001 that the first practical IBC scheme was proposed by Boneh and Franklin. Due to the strict security proof and high efficiency, this scheme has received widespread recognition in academic fields.

Motivated by the above observation, 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. 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.

Therefore, this research design a block design-based key agreement protocol for data sharing in cloud computing. First, proposed 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. Namely, the proposed protocol supports multiple participants

IV. 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. 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. 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. Since it was introduced in [16], resistance to compromised keys has been taken into consideration, which is an important issue in the context of cloud computing. Note that cloud storage auditing with verifiable outsourcing of key updates paradigm was proposed by Yu et al. in [17] 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 is based on a centralized model in [17], which not only imparts a burden to the TPA but also introduces some security problems.

In [18], 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 [4], many works have attempted to provide authentication services in the key agreement protocol. In [19], 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[21].

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. To avoid the requirement of the public key certificate, in 1984, identity-based cryptography (IBC) was proposed by Shamir. However, it was not until 2001 that the first practical IBC scheme [10] was proposed by Boneh and Franklin. Due to the strict security proof and high efficiency, this scheme has received widespread recognition in academic fields. In the same year, a popular proof model for group key establishment was proposed by Bresson et al. [23].

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 of the group Diffie-Hellman key exchange were addressed. However, some security properties are missing in [23], 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. Recently, 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 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. Motivated by the above observation, 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. Compared with the one-to many patterns, 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.

It is widely known that data sharing in cloud computing can offer scalable and limitless storage and computational sources to people and enterprises. However, cloud computing additionally ends in many protections and privateers' concerns, together with records integrity, confidentiality, reliability, fault tolerance and so on. Note that the key agreement protocol is one of the essential cryptographic primitives, which could offer stable communication amongst a couple of members in cloud environments.

V. GROUP DATA SHARING

Cloud computing is said to be the service-oriented computing technology, which are affordable and flexible over the internet. In past few years the cloud has become more matured and provided many services, one of the primary services is data sharing in Group, where the data can be easily shared from one member to another. However, while sharing the data security is one of the primary concerns. In past several methodologies have been proposed. However, these methods lacked from the feasibility. Hence, in this paper we have propose methodology is based on the selection scheme. Here General Group Key is generated and moreover General Key agreement protocol is decentralized based model where the data are controlled by the owner within the same group. Moreover, the proposed methodology is evaluated by analyzing the comparative analysis based on the various number of parameter. Result Analysis suggest that our methodology simply outperforms the existing one.

In recent decades as the concept of cloud computing rises, cloud storage is said to be the one of the hotspots of the storage of information. It basically refers to a model, which refers to the model that provides the data storage. Here, CSP (cloud service provider) is directly responsible for making data available as well as accessible according to the requirement of use. Storage capacity is either bought or leased from provider to store the data by the individual or organization. This service can easily be accessed through the API or the application, which utilizes the API such as cloud storage gateway. Moreover, in the past few years, it has been observed that the demand of cloud storage has been phenomenal in accordance with the use of personal as well as business purpose, since it is highly based on the virtualized infrastructure and much more flexible in terms of multi-tenancy, scalability and availability.

They are typically knows as object storage such as Microsoft azure, amazon s3 and oracle cloud storage [4]. Since the cloud, computing gives us the feature of pay as you go service, the organization wants to pay only for the service they use, and cloud service provides exactly the same. Business using the CS can actually reduce up to 70% of energy consumption. CSP is totally responsible for the maintenance of the data and as well as the other tasks such as buying the additional storage capacity. Since the backup of the data are located in several places in the globe, it can also be applicable as the proof backup of natural disaster. Meanwhile, cloud storage is one service, which is not referred to the physical device, but it is the aggregation of many server and storage for its users.

The dynamic broadcast encryption technique allows data owners to securely share their data files with other users within the group including newly joined users. Unfortunately, each user has to compute revocation parameters to protect the confidentiality from the revoked users in the dynamic broadcast encryption scheme such that revoked users cannot access the data after their revocation
from the group. This results in both computations overhead of the encryption and the size of the cipher-text increases with the number of revoked users. Thus, the heavy overhead and large cipher-text size may hinder the adoption of the broadcast encryption to the limited users. The group manager is allowed to compute the revocation parameters, which includes the list of revoked users and make this revocation list available to public by migrating them into the cloud. Each time when users request for the data cloud service provider verifies the revocation list and provide access to data only to active users in the group. Such a design can significantly reduce the computation overhead of users to encrypt files and the cipher-text size.

VI. SYSTEM MODEL

The system model of our group data sharing scheme in cloud computing is illustrated. 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. 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.

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. 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. 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. Since the storage auditing can follow the state of the art auditing protocols, we only focus on the design of group data sharing scheme in cloud computing in the paper. The adversary model determines the capabilities and possible
actions of the attacker. Similar to [12], the adversary model is defined as follows.

The adversary reveals a long-term secret key of a participant in a conference and then impersonates
others to this participant.

1. The adversary reveals some previous session keys and then learns the information about the session key of a fresh participant. Consequently, the adversary can impersonate the fresh participant with the session key to others.

2. The adversary reveals the long-term keys of one or more participants in the current run. Then, the adversary attempts to learn the previous session key.

3. A malicious participant chooses different sub keys, generates different signatures and broadcasts the messages to the corresponding participants, which makes the conference key derived by different participants distinct.

VII.METHODS

The construction of the group data sharing model to support a group data sharing scheme for multiple participants applying an SBIBD, we design an algorithm to construct the (v, k + 1, 1)-design. Moreover, the constructed (v, k + 1, 1)-design requires some transformations to establish the group data sharing model such that v participants can perform the key agreement protocol. 4.1 Construct the (v, k + 1, 1)-design in our group data sharing model, the parameters of the SBIBD have some specific meanings. 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. Following papers [12] and [13], Algorithm 1 is designed to construct the structure of a (v, k + 1, 1)-design. 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 3, V = {0, 1, 2... v − 1} represents the set of v participants, whereas B = {B0, B1, B2... Bv−1} implies v blocks constituted by these v participants. Note that the block is defined as Bi = {Bi,0, Bi,1, Bi,2, ..., Bi,k}, which means each block embraces k + 1 participants, and Bi,j denotes which participant is contained in the j th column of the i th block.

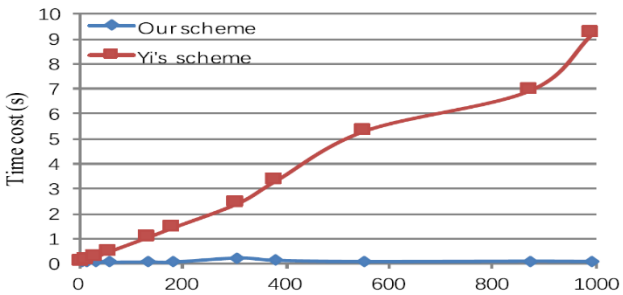

Fig 1. Efficiency comparison at initial phase

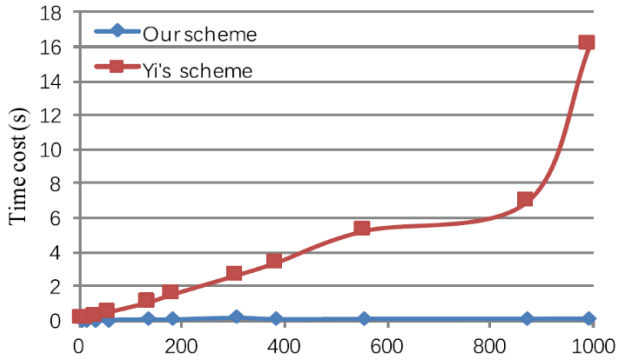

Fig 2. Efficiency comparison at key agreement phase

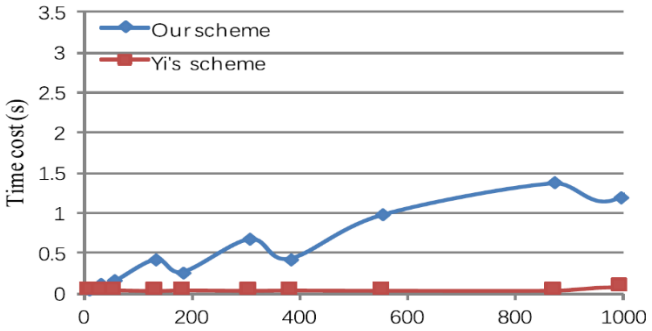

Fig 3. Efficiency comparison at authentication phase

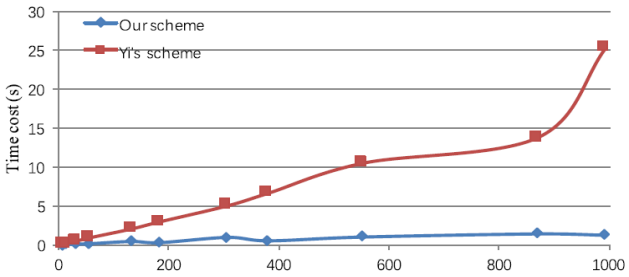

Fig 4. Efficiency comparison for multiple users

Sometimes we will consider blocks organized as a matrix in which column j is composed by elements Bi, j for i = 0, 1, 2... K and row i is composed by elements Bi, j for j = 0, 1, 2... k. The structure of the (v, k + 1, 1)-design is constructed by Algorithm 1, which outputs numbers Bi, j for i = 0, 1, ..., k2 + k and j = 0, 1, ..., k. In Algorithm 1, the notation MODk represents the modular operation that takes the class residue as an integer in the range 0, 1, 2... K – 1. Based on Algorithm 1, we can create the structure of a (v, k + 1, 1)-design that involves v participants. Moreover, Algorithm 1 can directly determine which participant should be involved in each block. For example, taking the (13, 4, 1)-design into consideration, where 13 participants are involved in this structure, we can decide which participant should be contained in the 3 rd column of the 8 th block by computing B7,2 = jk + 1 + MODk(i – j + (j – 1) b(i – 1)/kc) = 2 · 3 + 1 + MOD3(7 – 2 + (2 – 1) b(7 – 1)/3c) = 7 + MOD3(5 + 1 · 2) = 7 + 1 = 8. Therefore, from the above calculation, it is concluded that participant 8 is contained in the 3 rd column of the 8th block. Here, participant represents the I th participant.

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 ≤ k 2 + k; i + + do
for j = 0; j ≤ k; j + + do
if j == 0 then
Bi, j = b(i – 1) /kc;
else Bi,
j = jk + 1 +MODk (i–j + (j –1) b (i – 1)/kc);
end if
end for
end for

Algorithm is an optimization of the algorithm in [12] and the proof of the correctness follows the same lines than the proof in [12] and [13]. 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.

VIII. CONCLUSION

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. However, the conference key agreement asks for a large amount of information interaction in the system and more computational cost. To combat the problems in the conference key agreement, the SBIBD is employed in the protocol design. In this paper, 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 of a ðv; k þ 1; 1Þ-design,

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 practical and secure. In our future work, we would like to extend our protocol to provide more properties (e.g., anonymity, traceability, and so on) to make it appliable for a variety of environments.

IX. REFERENCES

[1]. L. Zhou, V. Varadharajan, and M. Hitchens, "Cryptographic rolebased access control for secure cloud data storage systems," IEEE Trans. Inf. Forensics Secur., vol. 10, no. 11, pp. 2381–2395, Nov. 2015.

[2]. F. Chen, T. Xiang, Y. Yang, and S. S. M. Chow, "Secure cloud storage meets with secure network coding," in Proc. IEEE Conf. Comput. Commun., 2014, pp. 673–681.

[3]. D. He, S. Zeadally, and L. Wu, "Certificateless public auditing scheme for cloud-assisted wireless body area networks," IEEE Syst. J., vol. PP, no. 99, pp. 1–10, 2015, doi: 10.1109/JSYST.2015.2428620.

[4]. Q. Huang, Y. Yang, and J. Fu, "Secure data group sharing and dissemination with attribute and time conditions in Public Clouds," IEEE Transactions on Services Computing, 2018.

[5]. J. M. Such and N. Criado, "Resolving multi-party privacy conflicts in social media," IEEE Trans. on Knowledge and Data Engine, vol. 28, no. 7, pp. 1851-1863, 2016.

[6]. K. Seol, Y. Kim, E. Lee, Y. Seo, and D. Baik, "Privacy-preserving at- tribute-based access control model for XML-based electronic health record system," IEEE Access, vol. 6, pp. 9114-9128, 2018.

[7]. X. Yi, "Identity-based fault-tolerant conference key agreement," IEEE Trans. Depend. Secure Computation., vol. 1, no. 3, pp. 170–178, Jul.–Sep. 2004.

[8]. Ravindra Changala, "Development of Predictive Model for Medical Domains to Predict Chronic Diseases (Diabetes) Using Machine Learning Algorithms and Classification Techniques", ARPN Journal of Engineering and Applied Sciences, VOL. 14, NO. 6, MARCH 2019, ISSN 1819-6608.

[9]. J. Shen, S. Moh, and I. Chung, "Identity-based key agreement protocol employing a symmetric balanced incomplete block design," J. Commun. Netw., vol. 14, no. 6, pp. 682–691, 2012.

[10]. B. Dan and M. Franklin, "Identity-based encryption from the weil pairing," SIAM J. Comput., vol. 32, no. 3, pp. 213–229, 2003.

[11]. K. Xue, W. Chen, W. Li, J. Hong, and P. Hong, "Combining data owner-side and cloud-side access control for encrypted cloud storage," IEEE Transactions on Information Forensics

and Security, vol. 13, no. 8, pp. 2062–2074, 2018.

[12]. I. Chung and Y. Bae, "The design of an efficient load balancing algorithm employing block design," J. Appl. Mathematics Comput., vol. 14, no. 1, pp. 343–351, 2004.

[13]. O. Lee, S. Yoo, B. Park, and I. Chung, "The design and analysis of an efficient load balancing algorithm employing the symmetric balanced incomplete block design," Inf. Sci., vol. 176, no. 15, pp. 2148–2160, 2006.

[14]. Ravindra Changala, "A Survey on Development of Pattern Evolving Model for Discovery of Patterns In Text Mining Using Data Mining Techniques" in Journal of Theoretical and Applied Information Technology in 31st August 2017. Vol.95. No.16, ISSN: 1817-3195, pp.3974-3987.

[15]. N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 1, pp. 222–233, Jan. 2014.

[16]. J. Yu, K. Ren, C. Wang, and V. Varadharajan, "Enabling cloud storage auditing with key-exposure resistance," IEEE Trans. Inf. Forensics Secur., vol. 10, no. 6, pp. 1167–1179, Jun. 2015.

[17]. L. Xu, C. Jiang, N. He, Z. Han, and A. Benslimane, "Trust-based col- laborative privacy management in online social networks," IEEE Transactions on Information Forensics and Security, vol. 14, no. 1, pp. 48- 60, 2019.

[18]. H. Cui, X. Yi, and S. Nepal, "Achieving scalable access control over encrypted data for edge computing networks," IEEE Access, vol. 6, pp. 30049–30059, 2018.

[19]. Ravindra Changala, "Retrieval of Valid Information from Clustered and Distributed Databases" in Journal of innovations in computer science and engineering (JICSE), Volume 6, Issue 1,Pages 21-25, September 2016.ISSN: 2455-3506.

[20]. Z. Tan, "An enhanced three-party authentication key exchange protocol for mobile commerce environments," J. Commun., vol. 5, no. 5, pp. 436–443, 2010.

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

[22]. A. Shamir, "Identity-based cryptosystems and signature schemes," in Proc. Workshop Theory Appl. Cryptographic Techn., 1985, vol. 21, no. 2, pp. 47–53.

[23]. Ravindra Changala, "Data Mining Techniques for Cloud Technology" in International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE),Volume 4, Issue 8, Pages 2319-5940, ISSN: 2278-1021, August 2015.

[24]. O. Hasan, L. Brunie, E. Bertino, and N. Shang, "A decentralized privacy preserving reputation protocol for the malicious adversarial model," IEEE Trans. Inf. Forensics Secure., vol. 8, no. 6, pp. 949– 962, Jun. 2013.

[25]. L.-K. Hua, Introduction to Number Theory. Berlin, Germany: Springer, 2012.

[26]. R. Barua, R. Dutta, and P. Sarkar, "Extending joux's protocol to multi party key agreement (extended abstract)," in Proc. 4th Int. Conf. Cryptology India, 2003, pp. 205–217.

[27]. R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: Improved definitions and efficient constructions," J. Comput. Secur., vol. 19, no. 5, pp. 79–88, 2011.

[28]. H. Guo, Z. Li, Y. Mu, and X. Zhang, "Cryptanalysis of simple three-party key exchange protocol," Comput. Secur., vol. 27, no. 1/2, pp. 16–21, 2008.

[29]. B. Lamacchia, K. Lauter, and A. Mityagin, "Stronger security of authenticated key exchange," in Proc. Int. Conf. Provable Secur., 2007, pp. 1–16.

[30]. Ravindra Changala, Evaluation And Analysis Of Discovered Patterns Using Pattern Classification Methods In Text Mining, ARPN Journal of Engineering and Applied Sciences, Vol3,Issue 11.

[31]. Dr.Mahesh K, "A Survey on Predicting Uncertainty of Cloud Service Provider Towards Data Integrity and Economic" 2019 IJSRST | Volume 6 | Issue 1 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X.

Cite this article as :

Dr. Gangolu Yedukondalu, Guna Santhoshi, Karnati Durga, Kotha Chandrakala, Dr. Mahesh Kotha "Development of A Novel Block Design Based Key Agreement Protocol for Cloud Environment to Improve Efficient Performance and Security", International Journal of Scientific Research in Science and Technology (IJSRST), Online ISSN : 2395-602X, Print ISSN : 2395-6011, Volume 9 Issue 5, pp. 52-63, September-October 2022. Available at doi : https://doi.org/10.32628/IJSRST22954
Journal URL : https://ijsrst.com/IJSRST22954