Open Access **Baghdad Science Journal** **P-ISSN: 2078-8665**
Published Online First: Suppl. November 2022 **2022, 19(6): 1572-1580** **E-ISSN: 2411-7986**

DOI: https://dx.doi.org/10.21123/bsj.2022.7077

A Security and Privacy Aware Computing Approach on Data Sharing in Cloud Environment

***Mustafa Azeez AL Mayyahi*[1*]** ***Seyed Amin Hosseini Seno*[2]**

Department of Computer Engineering, Ferdowsi University of Mashhad, Iran.
*Corresponding author: mustafaal-mayyahi@alumni.um.ac.ir
E-mail addresses: hosseini@um.ac.ir

Received 19/2/2022, Accepted 26/6/2022, Published Online First 25/11/2022, Published 5/12/2022

Abstract:

Today, the role of cloud computing in our day-to-day lives is very prominent. The cloud computing paradigm makes it possible to provide demand-based resources. Cloud computing has changed the way that organizations manage resources due to their robustness, low cost, and pervasive nature. Data security is usually realized using different methods such as encryption. However, the privacy of data is another important challenge that should be considered when transporting, storing, and analyzing data in the public cloud. In this paper, a new method is proposed to track malicious users who use their private key to decrypt data in a system, share it with others and cause system information leakage. Security policies are also considered to be integrated with the texts encrypted to ensure system safety and to prevent the violation of data owners ' privacy. For this purpose, before sending the data to the cloud, it must be encrypted in such a way that operations such as max, min, etc. can be performed on it. The proposed method uses order-preserving symmetric encryption (OPES), which does not require decryption or re-encryption for mathematical operations. This process leads to a great improvement in delay. The OPES scheme allows comparison operations to be performed directly on encrypted data without decryption operands. According to the results, it is obvious that the proposed strategy is in a better position compared to the base paper in terms of the system's ability to find the malicious elements that cause the problem of leakage and in terms of system security to prevent the violation of privacy.

Keywords: Encryption, Cloud Computing, CP-ABE based encryption, Privacy, Security.

Introduction:

After the advent of parallel computing, cloud computing has become the latest modern computing technology[1]. Large computational tasks are split and then distributed in the cloud[2,3]. Because cloud computing contains users' personal information, data security has a particular importance. Attackers may intercept or even misuse data in the communication process. This process could compromise users' privacy. Therefore, privacy protection in the cloud has received a great deal of attention in recent years. Public key encryption is the most powerful mechanism for protecting privacy[4]. In addition, attribute-based encryption is considered to be one of the most promising cryptography methods for secure and flexible one-to-many access control, in large-scale distributed systems, especially with unknown participants. However, most attribute-based encryption schemes are not suitable for cloud computing because they involve pairing operations. This operation is a big challenge for mobile devices with limited resources[5].

Fan et al.[2] believe that the most commonly used solution to protect users' privacy is encryption, but imposes a large computational burden on users. One of the techniques reducing the computational burden is to outsource some computational operations. In most work, only the decryption operation is left to the cloud computing. This means that there is still a heavy computational burden on users. In the base paper, in addition to decryption operations, most of the cryptography operations are assigned to the public cloud. The base paper uses the CP-ABE encryption method to do this. It also adds a new method to the basic method to expire users' credentials so that users can be optimally

removed from the system after the work is completed and unauthorized access for expired users is not allowed. In the method of the base paper, it is not mentioned that there will be a possibility of information leakage in the system. This means that authorized users (users whose attributes comply with the data policy tree) can transfer this privilege to other users. In fact, any user whose attributes match the data policy tree can decrypt the data[6]. In fact, any user whose attributes match the data policy tree can decrypt the data. If a user sells their private key and violates data privacy and security, how can our system counteract this and protect data privacy? Also, in Fan et al.[2], similar to CP-ABE-based methods, the policy string is sent in plain text with cipher-text. Now, if unauthorized people receive this message, according to the policy, they will know which users can open this message. This process violates the privacy of data owners. To solve these challenges, this paper provides a method to track malicious users who, using their private keys and attributes, decrypt the data in the system, share the decrypted data with others, and lead to information leaks in the system. Also, the security of data transmission policies with cipher-text is considered to ensure the security of the system and prevent the violation of the privacy of data owners. In this study, a new method is proposed to track users who use their private key to decrypt the data in the system, share it with others and cause system information leakage. Security policies are also considered to be integrated with the texts encrypted to ensure system safety and to prevent the violation of data owners ' privacy.

In the following, the previous work is reviewed in Section 2. In Section 3, the problem model is presented. Section 4 presents the proposed method. Section 5 presents the results of the experiment. Finally, in the last section, the conclusion of the paper and the path of future work will be discussed.

Literature Review

This section discusses the following topics: privacy and security in cloud computing, cloud computing trust assessment model, attribute-based encryption in cloud computing, and access control mechanism for cloud computing.

1- Privacy-preserving System for Cloud Computing

Most relevant research works of this paper include the security solutions which cryptographically enforce fine-grained access control over attribute-based encryption (ABE) with special properties, e.g., attribute-based broadcast encryption (ABBE) and privacy-preserving ABE.

Xiong et al.[7] proposed an ABE-based cryptography method called the A^2 B^2 E system that provides a hidden access policy. The encrypted text is only visible to users whose identities are specified in the recipient collection. These users can simultaneously enforce the predefined access policy.

Li et al.[8] developed the privacy-preserving cloud-assisted mobile multimedia (PPCMM) technique, which is a cloud-based mobile multimedia data access control scheme. This approach has also taken into account the privacy policy and efficiency of online encryption and decryption. The PPCMM technique can also support any large-scale survey system (LSSS) uniform access policy.

Liu et al.[9] proposed lightweight access to medical services and privacy based on the multi-source attribute-based signature (ABS) technique for health care cloud called LPP-MSA. Under this scheme, MSR can access telemedicine services without fully disclosing its identity information. The result of implementing this technique is the realization of privacy.

Huang et al.[10] proposed a multidimensional media sharing and privacy scheme called SMACD for mobile cloud computing. The scheme encrypts each media layer with an attribute-based access policy. Consequently, the confidentiality of the media, as well as the fine-grained and coarse-grained access, will be guaranteed.

Xu et al.[11] proposed a verification information model and privacy-preserving based on lightweight data structures for Health-CPS. The design idea, architecture, formal definition, security definition, communication protocols of this model are presented in full detail. The main structural processes of the model (initialization, data addition, scaling, data query and verification) are also developed in this paper.

2- The Trust Assessment Model in Cloud Computing

Trust has been proved to be an efficient mechanism to solve the reputation and reliability problems in the distributed open environments.

Li et al.[12] proposed a reliable service combination model (TALMSC) for mobile cloud environments. The TALMSC model is a three-tier model involving mobile cloud users (customers), service providers, and service intermediaries. A new integrated trust model based on the FCE method is also proposed. This trust mechanism is a comprehensive, informed and appropriate method and is able to combine direct trust with users' recommendations. In addition, an improved two-

step FCM-based algorithm is designed to improve the learning ability of intermediaries.

Rathi et al.[13] considered various parameters such as data security, user management, permissions, authentication, and virtualization to provide security for cloud space. The proposed model can be used to evaluate different cloud services.

Wu[14] presented a cloud trust model based on a confidence level agreement. The proposed method can help cloud computing units to make good mutual decisions. The main contribution of this paper is to present a hierarchical trust modeling approach and improve the user's situational awareness in cloud computing.

Saeed et al.[15] proposed a new trust model called the ARICA Model, which helps the user increase the provider's trust and reduce third - party feedback. The ARICA model strengthens the dependence on the value of the user's trust. The proposed model calculates the trust based on five attributes: availability, reliability, integrity, confidentiality.

Mohsenzadeh et al.[16] proposed a new model of trust based on past interactions between mojitos. Relationships between entities are modeled on four types of previous interactions (ie, completely successful, completely unsuccessful, relatively successful, and relatively unsuccessful). Rewards and punishments are also provided to users to encourage honest behavior and prevent malicious behavior.

Sun[17] proposed a trust access control model for cloud services. First, they proposed a trust assessment method based on the degree of direct trust, trust risk, punishment, reward, and commitment to express the complexity and uncertainty of the trust relationship. Second, they developed a trust-based algorithm based on reliability weight, information entropy, and maximum scatter.

3- Attribute- based Encryption in Cloud Computing

In recent years, attribute-based encryption has become a hot spot in many ways such as access structures, security proofs, revocation and its efficiency.

Kumar et al.[18] examined the main encryption schemes and encryption policies based on access structure and multi-reference schemes. In addition, they evaluated various aspects of policy attribution-based cryptography, including hidden policy, proxy re-encryption, termination mechanism, and hierarchical attribute-based encryption.

Li et al.[19] proved that multiple files cannot be encrypted at the same level of access. In order to create a more flexible method, they presented a EFH-CP-ABE scheme in which several files are encrypted at a level of access. Subsequently, they proved the security of the proposed scheme in the form of mathematical formulations.

Yang et al.[20] proposed a scheme without central authority for key and user management and reinforced privacy and security dynamically. The protocol evaluating the trust relationship between the attribute authorities is simple. In addition, the scalability conditions for the authorities are also supported by the proposed scheme.

Zhang et al.[21] proposed a new scheme to reduce information leakage. In the proposed design, the network-based trap extension technique is used to reach the maximize leakage rate of 1 degree. In addition, data is only sent to anonymous users who can protect the recipients' privacy.

Liao et al.[22] proposed a new and safe method for reducing the overflow of the cloud server to add the recyclability to cloud server resources. Many users are satisfied with the existence of only one access policy. In addition, decryption outsourcing has also decreased.

4- Access Control Mechanism for Cloud Computing

Roy et al.[23] proposed a new scheme which combines fine-grained access control and cloud-based multiple servers and authentication mechanisms for the health care industry of 4.0.

Zhou et al.[24] studied the application of quantum cryptography and quantum key distribution in the access control method. They presented the encryption scheme and key distribution protocol based on quantum mechanics (CQM). In this paper, the graphics language of CQM is used.

Yang et al.[25] developed a privacy-based blockchain-based access control framework called AuthPrivacyChain. The node account address in the blockchain is used as the node identity. This framework first encrypts the data access control license and then stores it in the blockchain. The authors then designed the access control and revocation processes in Auth Privacy Chain.

Xiong et al.[26] developed a new CP-ABE-based storage model for data storage and secure access for IoT applications. Then, they introduced a secure and efficient multi-purpose access control scheme for the IoT-based cloud storage system, SEM-ACSIT, which provides backward and forward security by revocation of user's attributes.

Problem Modeling, Assumptions and Constraints

Our system consists of five components: A Cloud Service Provider (CSP), Global Certification Authority (CA), Data Owner (DO), User Data (DU), Attribute Authorities (AA). CSP is responsible for storing information (encrypted text, keys and user lists). The data access service is supported by CSP. Once the user revocation and attribute revocation is achieved, CSP will be responsible for the relevant operations. In addition, encryption and decryption calculations are outsourced. Therefore, CSP is responsible for encrypting part of the text. If and only if the DU attributes meet the access control policy, CSP can use the available keys for pre-encryption. This part of the decoded text is then sent to the DU for final decoding. CA is the only entity that is fully trusted in our design. All AAs and DUs must register in the CA. CA is responsible for the global launch operation of the plan and issues a global identity to each user. DO is an entity that defines, encrypts, and defines access control policies on design attributes. DU is an entity that verifies DO confidential information. After decoding in CSP, DU can receive the decrypted text. It then retrieves the data with its secret key. Each AA has nothing to do with the other AA. They are responsible for user attributes that are independently exported, canceled and updated. In our scheme, each user needs private keys related to the attributes involved. AA is responsible for issuing these private keys.

The Proposed Method

This paper plan to address the security challenges posed by adding the possibility of detecting users who decode the data. In addition, his paper intends to provide a solution to the security of the sent policies and consequently allows a certain operation to be carried out. The innovation of this article is: Providing a way to track virtual users who decrypt data in the system and find malicious elements that sell their private keys due to the problem of secret key leakage, resulting in privacy and security of owners' data attack. In addition, modifications to the revocation system will be required to properly remove malicious elements detected in the system.

Our proposed solution consists of three sections that will be explained below. Given the challenges posed in the first section, in the following paper, the authors provide a way to both secure data transmission and detect malicious elements in the system that sell private keys and leak data owners' information. It also secures submitted policies and ultimately fully protects users' privacy. In addition, information must be stored on the cloud in such a way that authors can perform specific operations on it. This attribute allows the user to not only share data over the cloud but also perform specific calculations on the data while maintaining user privacy. Therefore, before sending to the cloud, the data must be encrypted in such a way that authors can perform operations such as max, min, etc. on them. To protect users' privacy, authors must be able to identify malicious nodes. To do this, the DU first sends its request for data access to the CSP. CSP, however, does not immediately provide the user with a simple access policy. CSP first encrypts it and then sends it to the user. Finally, it should check if the user can decrypt the access policy with their first secret key. If he does not have this ability, his request will be canceled altogether. Otherwise, the user can access the information if he can decrypt the access policy and if he can fulfill it and also the user trust level is higher than the threshold trust level. Otherwise, the user's request is canceled, and the extent of his trust decreases. If he can meet the policy but his trust level is less than the threshold value, the CA will again cancel the request. Because there is no good record of this user in the log file and the user has tried for unauthorized access, as a result, the system decides to cancel the user's request in order to provide better security. Because user trust is lower than the threshold. The user will have access to the data if the value of the trust exceeds the threshold. In fact, CSP sends partially encrypted data to the DU. If the CA detects a key leak, the user's trust is actually reduced and this violation is recorded in the log file and the user's trust is reduced. Next time, if this user submits their request again, authors can cancel the user immediately.

1- Adding the Possibility of Tracking System Users

This paper, first, intends to identify malicious elements in the system based on user tracking. In CPABE methods, the problem of information leakage is less addressed. Because multiple users can decrypt data using their attributes, it is difficult to identify authorized or unauthorized individuals. It is also difficult to identify the private keys used for decryption. Because the private key is generated based on user attributes (userId, requestId, physicalAddress and reputation), users with similar attributes have the same private key. As a result, during decryption authors will not know which user has decrypted. This security problem is called "secret key leakage". Once detected, the attacker must be removed from the system. So in the proposed mechanism, the revocation part must also be modified.

2- Increasing the Security of Sent Policies Along with Encrypted Texts

Second, the authors intend to secure sent policies with encrypted text messages. These policies specify authorized users. The solution is to have a "name and value" pair for each attribute, enter only the names in the policy, and enter the values for each attribute in the encrypted text. If a malicious node accesses the message and its accompanying policy, it will not be able to access important information according to the attribute name field, and as a result, users' privacy will not be compromised.

3- Possibility of Performing Special Operations on Data Encrypted in the Cloud

One of the most important issues in cloud services is the security of stored data. Before a user can trust the cloud service, he must ensure its privacy and minimize the risk of unauthorized access. The Order-Preserving Encryption Symmetric (OPES) technique is a definitive encryption scheme that preserves the numerical order of plain text. In the OPES scheme, the comparison operation is performed directly and without decoding the operands on the encrypted data. Therefore, query operations as well as MAX, MIN and COUNT can be performed directly on encrypted data. Similarly, GROUP BY and ORDER BY operations can also be used. Values should be decoded only when SUM or AVG is applied to the group. Therefore, the data is encrypted before it is sent to the cloud in such a way that it is possible to execute the operation like max and min and so on. In the proposed method, the OPES for arithmetic operation is not required to decrypt and then not re-encrypted. This trend will lead to an improvement in the delay. In the OPES scheme, the comparison operation is performed directly and without decoding the operands on the encrypted data. Therefore, query operations as well as MAX, MIN and COUNT can be performed directly on encrypted data. This scheme is used in cloud computing as a way to increase security, as it allows applications to execute sequence requests on encrypted data efficiently (without the need to decrypt the data). The OPES technique can be easily integrated with existing database systems, as it easily works with existing indexing structures such as B trees. The fact that the database is encrypted can be obvious to applications. Below, authors show the different steps of the proposed method in the form of an algorithm 1.

Algorithm 1 - Process Processing Algorithm

1. **Input:** RequestFromDUs,
2. requests=Receive_RequestFromDUs ();
3. **while** (requests ≠ null) **do for each** request
4. Send_ EncryptedAccessPolicy(request.DU);
5. success= request.DU.Decrypt_accessPolicy (DU.secrect key);
6. **if** (success==true) **then**
7. **if** (request.DU.Satisfy_Policy(request.DU.attributes)) **then**
8. **if** (request.DU.trust>= trust_thresh)) **then**
9. request.DU.Receive_PartiallyEncryptedDataOPES()
10. request.DU.Decrypt_PartiallyEncryptedDataOPES(request.DU.secondSecretKey);
11. addToLog file(request);
12. **if** (Detection_SecretKeyLeakage()==true) **then**
13. DecreaseTrust(request.DU);
14. **end if**
15. **else**
16. Revoke(DU);
17. **end if**
18. **else**
19. Request_ignore(request);
20. DecreaseTrust (request.DU);
21. **end if**
22. **else**
23. Request_ignore (request);
24. **end if**
25. **end while**

In line 2, all requests sent from DUs are entered in the request array. In CSP, that is, in the third line, as long as there is a request, that is, the array is not equal to Null, the mentioned tasks will be performed for each request. Line 4 sends the access policy data to DU. Subsequently, in line 5, if DU can decrypt the access policy with its secret key, ie "success == true", then in line 7, if DU can satisfy the access policy with its own attributes, and then in line 8, if DU trust value is greater than threshold trust, then the operation is done.

The trust threshold in the simulation is set to 0.5. Also, DU can only receive partial encrypted data with OPES and decrypt it with its second secret key. Finally, the processed request is recorded in the log file. In line 12, if the CSP detects a key leak, it reduces the DU trust value, and if the DU value is less than the threshold trust, then the CSP cancels the request (line 16). If the DU fails to comply with the policy, then its request will be rejected and its confidence level will decrease (lines 19 and 20), ie the submitted request will not be allowed. If the DU fails to decrypt the access policy, the request will still be rejected (line 23).

Simulation of the Proposed Method

To simulate the proposed method, authors use the iFogSim[27]. The iFogSim library is based on the Java programming language and has several modules and classes for simulating fog computing. This library is an extended version of the Cloudsim Simulator and uses many Cloudsim classes. The system in which the simulation is run has an Intel

Core i7 processor with a speed of 2 GHz, 12 GB of RAM and a 64-bit Microsoft Windows 10 operating system.

1- Simulation Parameters

Simulation parameters are shown in Table 1. Data owner is the one who encrypts and defines the access control policy over attributes for the scheme. Data user is the one that wants to acknowledge the secret data from the data owner. The number of data owners is 10 and the number of data users is 100 and, ultimately, the number of attacks is between 2 and 10.

Table 1. Simulation parameters

Parameters	Values
Number of data owners	10
Number of data users	100
Number of attacks	2-10
The amount of violator users	0.1, 0.05, 0.01

2- Evaluation of Results

The results of the proposed method are compared with paper[2] (Because the goal is to improve this paper[2]). Our evaluation metrics are similar to the metrics of this paper. The most important factors that indicate improvement of the proposed method are:

- The ability of the system to locate the malicious elements causing secret leakage.
- System security in prevention of violation of user's privacy.
- The cost of decoding.

To evaluate the proposed solution, a data access scenario is used in the cloud. Data owners plan to save their data in a secure cloud. On the other hand, the system users also plan to access the available data on using their attributes.

The scenario authors considered for this study is defined as:

Method have several users and several data owners. A CA is responsible for detecting secret keys. All processes of sending, receiving and accessing data are controlled by this entity. The user requests access to the data. If the user can successfully decrypt the access policy, then run the policy decryption process. If any of these steps fail, the user's trust will decrease and the CA will record it as a record associated with an unauthorized user. All accesses in CA are recorded, so if an unauthorized access is discovered or a secret key leakage by CA, the amount of trust of the authorized users to access the data will also be changed.

3- The System's Ability to Locate the **Malicious** Elements

According to the results in Fig. 1, it is obvious that the proposed strategy is in a better position compared to the base paper (both in terms of the system's ability to find the malicious elements that cause the problem of leakage and in terms of system security to prevent the violation of privacy). To test this criterion, several attacks were simulated. The purpose of the attacks was to obtain attributes as well as users' private keys. In each attack, the attacker tries to obtain the user's private key according to the user's characteristics. In the base paper, the user attributes are in the form of clear text. An attacker can easily identify these attributes by listening to the network. Then, based on these attributes, it extracts its desired private key. Therefore, extracting private keys and attributes is easily possible. In our method, attributes are encrypted. So the attacker (due to attribute encryption) cannot easily extract private attributes and keys. This criterion is based on "the number of times a secret key is exposed divided by the total number of available secret keys". The results of the number of violating users are presented in Table 2.

Table 2. The number of violating users in the proposed method and the base paper

Number of violating users (violating users / total users)	Base paper[2]	Proposed method
0.01	13	4
0.05	36	15
0.1	42	27

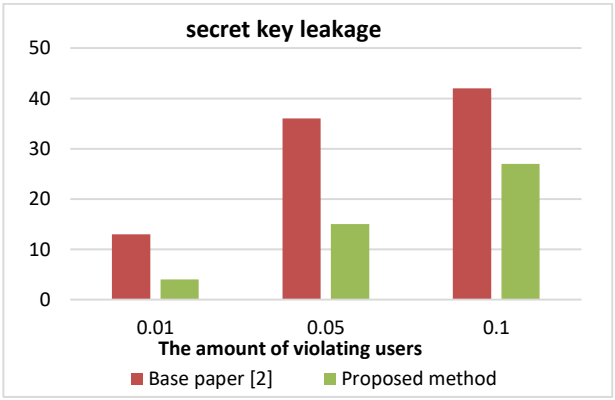

Figure 1. Comparison of the number of violating users in the proposed method and the basic paper

4- The Degree of Resistance of the System against Privacy Attacks

Examining the simulation results, it is obvious in Fig. 2 that the proposed solution is better than the base paper method in case of decoding costs. The reason for the superiority of the proposed method is that the proposed method attempts to hide the submitted policy along with the encrypted text. This concealment prevents the disclosure of information and the violation of the privacy of data owners. The results of the number of successful

attacks are presented in Table 3. This criterion is calculated based on the number of attributes disclosed divided by the whole number of available attributes.

Table 3. Results of the number of successful attacks in the proposed method and the basic paper

Number of attacks	Base paper [2]	Proposed method
2	0.1	0.01
4	0.2	0.04
6	0.36	0.25
8	0.48	0.33
10	0.54	0.37

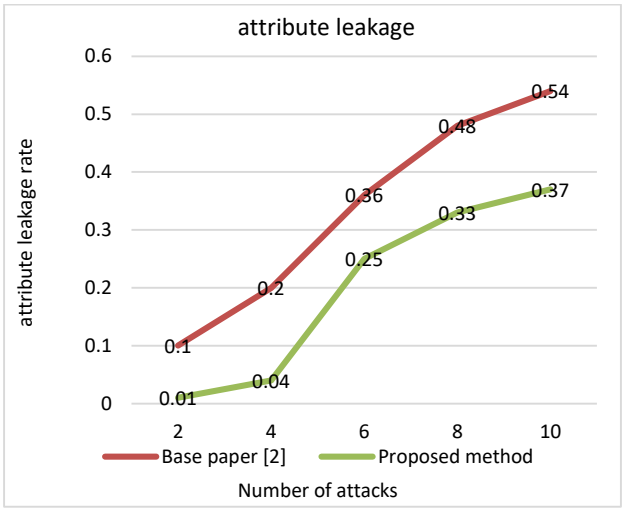

Figure 2. Comparison of the number of successful attacks in the proposed method and the basic paper

Security of a method of information sharing based on its vulnerability is dealing with attacks and information leakage rates. For this reason, one of the criteria for assessment of the proposed method is the amount of data leakage. Since attributes are encrypted, no one can access user attributes. In addition to data content, access to user attributes is also impossible. Lack of access to user attributes makes it almost impossible to access the original key to decode the data. These two protocols, namely the detection of the key leakage challenge, and the cryptography of attributes, make it difficult to access attributes.

5- Decryption Cost

Figure 3 shows the cost of calculations in the user decryption operation. The cost of user decryption varies according to the user's attribute set. In Fig. 3, it is clear that in 20 consecutive implementations, the proposed method is less expensive than the other designs and the base paper method. The reason for the superiority of the proposed method is the use of the OPES scheme for mathematical operations that do not need to be decrypted and re-encrypted. This has significantly improved the reduction of latency. The OPES technique allows comparison operations to be performed directly on encrypted data without decoding operands. The decoding cost results are presented in Table 4.

Table 4. Results of decryption cost in the proposed method and the base paper

User attribute	Base paper[2]	Proposed method
10	18	13
20	32	25
30	44	32
40	63	41
50	80	53

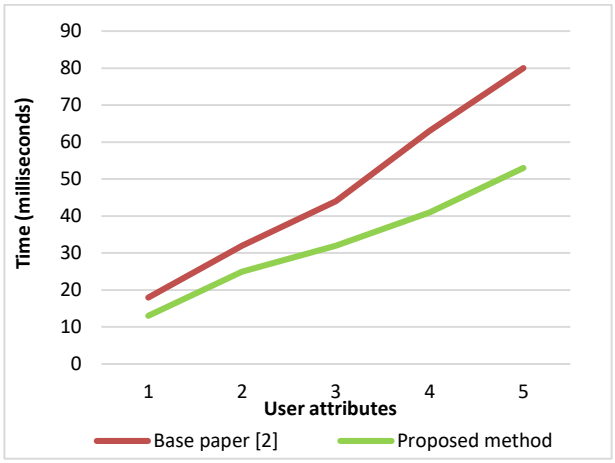

Figure 3. Comparison of decoding costs in the proposed method and the basic paper

Conclusion:

In this study, a new method is proposed to track users who use their private key to decrypt the data in the system, share it with others and cause system information leakage. Security policies are also considered to be integrated with the texts encrypted to ensure system safety and to prevent the violation of data owners ' privacy. In the proposed method, the OPES technique is used which does not require decryption and re-encryption mathematical operations. This has greatly improved the reduction of latency. The OPES technique allows comparison operations to be performed directly on encrypted data without decoding operands. According to the results, it is clear that the proposed strategy is in a better position compared to the base paper (both in terms of the system's ability to find the malicious elements that cause the problem of leakage and in terms of system security to prevent the violation of privacy). The reason for the superiority of the proposed method is the use of the OPES scheme for mathematical operations that do not need to be

decrypted and re-encrypted. This has greatly improved the reduction of latency.

Authors' Declaration:

- Conflicts of Interest: None.
- We hereby confirm that all the manuscript's figures and tables are ours. besides, the Figures and images, which are not ours, have been given permission for re-publication attached with the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee at the Ferdowsi University of Mashhad.

Authors' Contributions Statement:

M. A. A. proposed the work, designed the figures and the tables, and analyzed and compared the results. He also collected the data related to the previous studies. S. A. H. Seno discussed and compared the results. Both authors read the manuscript carefully and approve the final manuscript.

References:

1. Hajibaba M, Gorgin S. A review on modern distributed computing paradigms: Cloud computing, jungle computing and fog computing. J Comput. Inf. Technol. 2014;22(2):69-84.
2. Fan K, Liu T, Zhang K, Li H, Yang Y. A secure and efficient outsourced computation on data sharing scheme for privacy computing. J Parallel Distrib Comput. 2020; 135: 169-76.
3. Abed Marwa M, Manal F Younis. Developing load balancing for IoT-cloud computing based on advanced firefly and weighted round robin algorithms. Baghdad Sci J. 2019; 16(1): 130-139.
4. Waters B. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realizatio. In International Workshop on Public Key Cryptography Springer, Berlin, Heidelberg. 2011: pp. 53-70.
5. Li H, Lan C, Fu X, Wang C, Li F, Guo H. A secure and lightweight fine-grained data sharing scheme for mobile cloud computing. Sensors. 2020; 20(17): 4720.
6. Albu-Salih AT, Seno SA, Mohammed SJ. Dynamic routing method over hybrid SDN for flying ad hoc network. Baghdad Sci J. 2018;15(3): 361-368.
7. Xiong H, Zhang H, Sun J. Attribute-based privacy-preserving data sharing for dynamic groups in cloud computing. IEEE Syst J. 2018; 13(3): 2739-50.
8. Li Q, Tian Y, Zhang Y, Shen L, Guo J. Efficient privacy-preserving access control of mobile multimedia data in cloud computing. IEEE Access. 2019; 7: 131534-42.
9. Liu J, Tang H, Sun R, Du X, Guizani M. Lightweight and privacy-preserving medical services access for healthcare cloud. IEEE Access. 2019; 7: 106951-61.
10. Huang Q, Zhang Z, Yang Y. Privacy-preserving media sharing with scalable access control and secure deduplication in mobile cloud computing. IEEE Trans Mob Comput. 2020; 20(5): 1951-64.
11. Xu J, Wei L, Wu W, Wang A, Zhang Y, Zhou F. Privacy-preserving data integrity verification by using lightweight streaming authenticated data structures for healthcare cyber–physical system. Future Gener Comput. Syst. 2020; 108: 1287-96.
12. Li W, Cao J, Hu K, Xu J, Buyya R. A trust-based agent learning model for service composition in mobile cloud computing environments. IEEE Access. 2019; 7: 34207-26.
13. Rathi SR, Kolekar VK. Trust model for computing security of cloud. In 2018 Fourth international conference on computing communication control and automation (ICCUBEA). IEEE. 2018: pp. 1-5.
14. Wu X. Study on Trust Model for Multi-users in Cloud Computing. Int J Netw Secur. 2018; 20(4): 674-8.
15. Saeed O, Shaikh RA. A user-based trust model for cloud computing environment. Int J Adv Comput. 2018;9(3): 337-46.
16. Mohsenzadeh A, Bidgoly AJ, Farjami Y. A novel reward and penalty trust evaluation model based on confidence interval using Petri Net. J Netw Comput Appl. 2020; 154: 102533.
17. Sun P. Research on cloud computing service based on trust access control. Int J Eng. 2020; 12: 1847979019897444.
18. Kumar P, Alphonse PJ. Attribute based encryption in cloud computing: A survey, gap analysis, and future directions. J Netw Comput Appl. 2018; 108: 37-52.
19. Li J, Chen N, Zhang Y. Extended file hierarchy access control scheme with attribute based encryption in cloud computing. IEEE Trans Emerg Topics Comput. 2019.
20. Yang Y, Chen X, Chen H, Du X. Improving privacy and security in decentralizing multi-authority attribute-based encryption in cloud computing. IEEE Access. 2018; 6: 18009-21.
21. Zhang L, Gao X, Guo F, Hu G. Improving the Leakage Rate of Ciphertext-Policy Attribute-Based Encryption for Cloud Computing. IEEE Access. 2020; 8: 94033-42.
22. Liao Y, Zhang G, Chen H. Cost-efficient outsourced decryption of attribute-based encryption schemes for both users and cloud server in green cloud computing. IEEE Access. 2020; 8: 20862-9.
23. Roy S, Das AK, Chatterjee S, Kumar N, Chattopadhyay S, Rodrigues JJ. Provably secure fine-grained data access control over multiple cloud servers in mobile cloud computing-based healthcare applications. IEEE Trans Industr Inform. 2018; 15(1): 457-68.
24. Zhou L, Wang Q, Sun X, Kulicki P, Castiglione A. Quantum technique for access control in cloud computing II: Encryption and key distribution. J Netw Comput Appl. 2018; 103: 178-84.
25. Yang C, Tan L, Shi N, Xu B, Cao Y, Yu K. AuthPrivacyChain: A blockchain-based access

control framework with privacy protection in cloud. IEEE Access. 2020; 8: 70604-15.
26. Xiong S, Ni Q, Wang L, Wang Q. SEM-ACSIT: secure and efficient multiauthority access control for IoT coud storage. IEEE Internet Things J. 2020;7(4):2914-27.
27. Gupta H, Vahid Dastjerdi A, Ghosh SK, Buyya R. iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments. Software Pract Expe. 2017;47(9):1275-96.

طريقة الحوسبة المراعية للأمان والخصوصية في مشاركة البيانات في بيئة السحابة

مصطفى عزيز المياحي **سيد أمين حسيني سينو**

قسم هندسة شبكات الحاسوب، جامعة فردوسي، مشهد، إيران.

الخلاصة:

حالياً، تلعب الحوسبة السحابية في حياتنا اليومية دور بارز جدًا. يتيح نموذج الحوسبة السحابية توفير موارد قائمة ومطلوبة. لقد غيرت الحوسبة السحابية الطريقة التي تدير بها المؤسسات الموارد نظرًا لقوتها وتكلفتها المنخفضة وطبيعتها المنتشرة. عادة ما يتم تحقيق أمن البيانات باستخدام طرق مختلفة مثل التشفير. ومع ذلك، تعد خصوصية البيانات تحديًا مهمًا آخر يجب مراعاته عند نقل البيانات وتخزينها وتحليلها في السحابة العامة. في هذا البحث، تم اقتراح طريقة جديدة لتتبع المستخدمين الخبثاء الذين يستخدمون مفتاحهم الخاص لفك تشفير البيانات في النظام ومشاركتها مع الآخرين والتسبب في تسرب معلومات النظام. تعتبر سياسات الأمان أيضًا متكاملة مع النصوص المشفرة لضمان سلامة النظام ومنع انتهاك خصوصية أصحاب البيانات. لهذا الغرض، قبل إرسال البيانات إلى السحابة، يجب تشفيرها بطريقة يمكن من خلالها إجراء عمليات مثل max، و min، وما إلى ذلك عليها. تستخدم الطريقة المقترحة التشفير المتماثل للحفاظ على النظام (OPES)، والذي لا يتطلب فك التشفير أو إعادة التشفير للعمليات الحسابية. هذه العملية تؤدي إلى تحسن كبير في التأخير. يسمح مخطط OPES بإجراء عمليات المقارنة مباشرة على البيانات المشفرة بدون معاملات فك التشفير. وبحسب النتائج، فمن الواضح أن الإستراتيجية المقترحة في وضع أفضل مقارنة بالورقة الأساسية من حيث قدرة النظام على إيجاد العناصر الخبيثة التي تسبب مشكلة التسرب ومن ناحية أمن النظام لمنع انتهاك.

الكلمات المفتاحية: التشفير، الحوسبة السحابية، التشفير القائم على CP-ABE ، الخصوصية، الأمان.