Paper

An Attribute-Based Encryption Method Using Outsourced Decryption and Hierarchical Access Structure

Tabassum N. Mujawar[1,2] and Lokesh B. Bhajantri[3]

[1] *Research Scholar, Department of CSE, Basaveshwar Engineering College, Bagalkot, Karnataka, India*
[2] *Department of Computer Engineering, Ramrao Adik Institute of Technology, D Y Patil deemed to be University, Navi Mumbai, Maharashtra, India*
[3] *Department of ISE, Basaveshwar Engineering College, Bagalkot, Karntaka, India*

https://doi.org/10.26636/jtit.2022.158421

Abstract—Cloud computing is being rapidly adopted by many organizations from different domains and large amounts of data is stored in the cloud. In order to ensure data security, the attribute-based access control mechanism has been emerging recently as a fine-grained access control model that grants access based on the data user's attributes. In this model, the data owner builds the access policy using the attributes of the data users and access to the data is granted only if the requirements of such an access policy are satisfied. Ciphertext policy-based attribute-based encryption (CPABE) is one of the most widely used methods for providing encrypted access control. Complex, time consuming and costly paring operations are the major issue with the CPABE method. Hence, another efficient method is needed to reduce the data user's overhead while decrypting data. This paper presents an efficient method consisting in outsourcing decryption operations to a third-party server, so that complex operations may be performed by that machine with only some simple calculations left on the data user's side. The concept of a hierarchical access structure is also integrated with the traditional CPABE technique. The hierarchical approach enables the data owner to encrypt multiple data using a single common hierarchical access structure. This allows the user to decrypt only the relevant part of ciphertext, depending on which fragment of the hierarchical access structure is satisfied. The paper evaluates also the performance of the proposed model in terms of time and storage cost.

Keywords*—*cloud computing, CPABE, hierarchical access structure.

1. Introduction

In order to maintain authenticity of the data stored on cloud servers, it is necessary to apply appropriate security mechanisms. The data must be accessible to authorized users only and access must be denied to other parties. In traditional models, access permissions will be granted by the server on which the data is stored. However, the server itself is an untrusted entity and if it is compromised, the data may be accessed by any unauthorized person. Also, in this approach, the data owner is completely dependent on a third-party server for enforcing the data access policies. Hence, another approach in which the data owner is capable of controlling the access policies and deciding who can access the data is required. The modern attribute-based access control model provides access to data based on considering attributes of the data users and allows data owners to establish access policies by combining attributes and specifying which data user is allowed to access specific data.

Attribute-based encryption (ABE) is a scheme that offers encrypted access control by considering the user's attributes [1]. The access policy is built using different logic gates and data user's attributes. The scheme provides fine-grained access control and relies on a one-to-many encryption mechanism. Two different categories of the ABE scheme may be distinguished: ciphertext policy-based attribute-based encryption (CPABE) [2] and key policy-based attribute-based encryption (KPABE) [3]. In the case of CPABE, the user's attributes are bound with a private key used for decryption and the access structure is associated with the ciphertext. The ciphertext is decrypted only when the attributes associated with the decryption key satisfy the access structure. In the case of KPABE, the ciphertext is combined with the attributes and the access structure is integrated with the decryption key.

CPABE is the most widely adopted scheme providing access control based on specific attributes. The traditional CPABE scheme proposed in [2] includes a rather costly decryption process. The degree of complexity is increased with the complexity of the access policy. This is a significant the drawback of the scheme, as the data user incurs a lot of overhead. In order to deal with this issue, an outsourcing mechanism is applied, allowing to hand over the costly operations to a third-party server. This machine generates a transformed ciphertext which can be further decrypted by the data user. While decrypting the transformed ciphertext, the user has to apply simple computations and this reduces the burden on the data user's side. In the traditional scheme, each message is encrypted separately, using the relevant access structure, and the ciphertext is generated. Sometimes, the different access structures are hierarchically related and that can be combined to form one common access structure which, in turn, may be relied upon to encrypt multiple pieces of data together, instead of encrypting them separately. Hence, the time required for encryption and the storage cost related to the generated ciphertext will be reduced by applying such hierarchical access structures.

In this paper, an efficient CPABE scheme that utilizes the hierarchical access structure to encrypt data is proposed. Hierarchical access structures are built by combining multiple hierarchically related access structures. This approach helps to encrypt multiple pieces of data together, and generate a common ciphertext. If the access structure is satisfied fully, then the entire ciphertext is decrypted. If only a specific portion of the access structure is satisfied, then only the relevant portion of the ciphertext is decrypted. All major pairing operations are outsourced to an outsourcing server. This third-party outsourcing server returns the partially decrypted ciphertext to the data user. Then, the user can apply simple computations and can retrieve the original data. This scheme takes advantage of both the hierarchical access structure and the outsourcing approach. The proposed scheme is less costly in terms of storing the ciphertext and less time-consuming when it comes to performing the decryption process.

The remaining part of this paper is organized as follows. Section 2 describes the existing attribute-based encryption methods that rely on the outsourcing mechanism. Section 3 elaborates on the proposed outsourced decryption and hierarchical access structure-based CPABE scheme. The experimental analysis is described in Section 4. Conclusions are presented in Section 5.

2. Related Work

The complexity of the decryption operation in a traditional attribute-based encryption scheme depends primarily on how complex the access policy used for encryption is. The size of the generated ciphertext is quite large as well, meaning that the scheme requires more time for decryption. The overhead is incurred by the data user intending to access the encrypted data. An outsourcing-based ABE scheme is proposed in [4], eliminating the overhead stemming from decryption operations. In this scheme, the ABE ciphertext is converted into an ElGamal style ciphertext by the cloud. Then, the data user may transform this ElGamal style ciphertext to plaintext with less processing. The drawback of such an approach is that the correctness of the transformed ciphertext is not verified. It may be the case that a malicious cloud node may perform an incorrect transformation and, hence, the data user will not receive correct data. A scheme that performs an outsourced decryption along with a verifiable transformation is proposed in [5]. The correctness of transformation is verified by computing the checksum, which is a combination of one random message and the original message to be encrypted. This checksum and the random message are added to the ciphertext. The data user may verify correctness by computing the checksum again. An efficient outsourced ABE scheme with verifiability of transformation based on the key encapsulation mechanism is proposed in [6]. Here, the data is encrypted using symmetric encryption and the ABE approach is applied for encrypting the symmetric key. The hash value of the key is computed and is concatenated with ciphertext. The hash function is once again applied on this concatenated message and this hash is used to check whether the transformation has been performed correctly or not. The first hash value is used to check the integrity of the encryption key. The ABE scheme with outsourced decryption and verification that is CPA- and RCCA-secure is presented in [7]. This scheme utilizes a random value to check whether the cloud node has performed partial decryption correctly or not. The message and the random value are encrypted together. The scheme requires fewer computation resources and the ciphertext is small in size as well.

The standard model for a CPABE scheme with verifiable outsourced decryption and with a constant ciphertext length is presented in [8]. In this scheme the size of the ciphertext does not increase, despite the growing number of attributes or the complexity of access structure. The costly paring operations are outsourced to the cloud node and the data user can recover the message by applying very simple computations. The scheme proposed in [9] provides a fully verifiable outsourced decryption facility. The different access policies are designed for authorized and unauthorized users. The MAC is integrated with the ciphertext to ensure that the transformation performed by the cloud is correct. The scheme first verifies the correctness of the transformed ciphertext and then decrypts the ciphertext to obtain the plain text.

A scheme that outsources both encryption and decryption operations to a third-party untrusted server is proposed in [10]. Here, the exponentiation modulo computations are outsourced to a third-party encryption server, so that the overhead of the data owner can be reduced. An efficient method for generating the transformation key is proposed as well. The scheme verifies the correctness of the transformed messages and, hence, ensures their verifiability. An ABE scheme that supports hidden access policies is proposed in [11]. In this scheme, the attribute name is kept public, whereas the attribute value is kept hidden, so that privacy can be maintained. The scheme also outsources the costly paring operations to cloud nodes and data users can verify the calculations performed thereby.

An online-offline scheme for resource constrained devices operating in the cloud environment is proposed in [12]. It uses the Chameleon hash function to generate an immediate ciphertext and blind it with the offline ciphertext. In order to eliminate the overhead of decryption without authorization, a ciphertext test is performed before decryption. A verifiable and multiple authority-based ABE scheme is presented in [13]. In this scheme, the majority of encryption and decryption operations are transferred to fog devices. This will reduce the overhead on the part of the data owner and user. The scheme also presents a verification method to check the computations performed by fog devices. An efficient revocation method for users and attributes is also implemented. A fully outsourced ABE scheme is presented in [14]. The CPABE scheme is implemented by outsourcing all major tasks, such as encryption, decryption and key generation. In order to manage the additional communication overhead, key generation and encryption operations are performed offline. In [15], an efficient CPABE scheme that supports outsourced encryption and decryption is presented. The scheme utilizes the fog computing environment to outsource complex encryption and decryption operations. The fog nodes are responsible for partially encrypting the message and partially decrypting the ciphertext. The scheme also includes an efficient attribute revocation mechanism.

In [16], the authors present two different CPABE schemes, where the encryption and decryption operations are outsourced. In the first approach, the untrusted service provider performs the complex operations related to encryption and decryption. On the other hand, the other approach includes a key encapsulation mechanism, with the encryption and decryption operations being outsourced to a semi-trusted service provider. A traceable and outsourced decryption-based ABE scheme is presented in [17]. The scheme provides an outsourced decryption facility to reduce burden of the data user. A mechanism for updating policies while encrypting data is also included. A traceability feature is also incorporated in this scheme, so that malicious users can be detected.

3. Proposed Work

The CPABE scheme that supports hierarchical access structure and outsourced decryption is presented in this paper. The hierarchically related access structures are combined together to form a common hierarchical access structure. This feature will allow encrypting multiple messages using one common hierarchical access structure. The major issue associated with traditional CPABE schemes is that the decryption operation is very costly computationally. The decryption of a CPABE ciphertext requires complex paring operations and this incurs computational overhead for the data user. Hence, an outsourcing mechanism is proposed permitting to transfer complex operations to a third-party server. This unit will generate a partially decrypted ciphertext without understanding any details about the original message. Therefore, privacy of the original message will be preserved. This outsourcing concept will reduce the burden of performing complex tasks on the data user's side. The model of the proposed scheme is presented in Fig. 1.

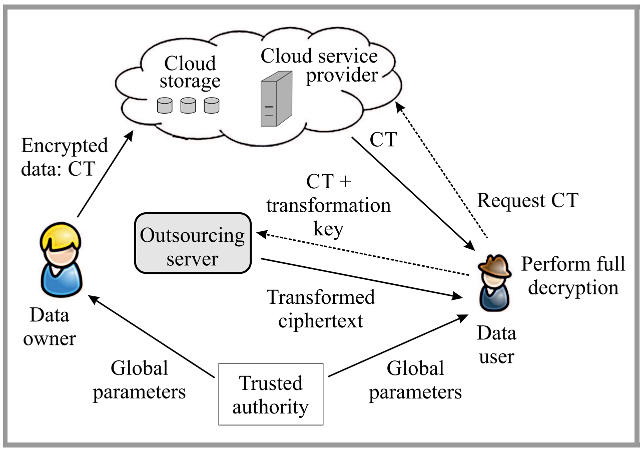

Fig. 1. System architecture.

The data owner generates access policies for the files that they want to store along with cloud storage. The access policies are prepared by combining attributes and logic gates. These decide who can access the data. A set of attributes of those users who are going to access the data is maintained and is based on the sensitivity of data. A tree-based access structure is generated to represent these conditions. Hierarchically related access structures are combined and one hierarchical access structure is formed. The data owner encrypts the data by using such a hierarchical access structure. This allows the data owner to encrypt multiple pieces of data together and generate a common ciphertext. This will reduce the efforts related to encrypting multiple pieces of data separately by using different access structures. This will reduce the encryption time and also the storage cost required to store separate ciphertexts. As the CPABE scheme is implemented, the access structure is integrated with the ciphertext and is then stored in the cloud. Global parameters of the system needed for encryption are issued by a trusted authority.

The cloud service provider is responsible for handling all requests of the data owner for storing the encrypted data in cloud storage and also deals with the users who are requesting access to the encrypted data. The data user must possess the necessary attributes to access the data from cloud storage. The service provider will transfer the ciphertext to the outsourcing server for partial decryption upon receiving a request from the data user.

The outsourcing server generates the transformed ciphertext if the attributes of data users satisfy the access structure. The hierarchical access structure is applied while encrypting the data. Hence, decryption is performed based on how big a portion of the access structure is satisfied. If the complete access structure is satisfied, then the user is allowed to access all of information. On the other hand, if only a specific portion of the access structure is satisfied, then only the data associated with that portion are accessible. Thus, the hierarchical access structure allows decrypting of all data or of its relevant portion. The outsourcing server only partially decrypts the data and generates the transformed ciphertext. The transformation algorithm will be applied in such a way that the outsourcing server cannot read or understand anything about the original data. The actual data can be retrieved only by the data user, by performing necessary computations over the transformed ciphertext. The transformation key is given to the outsourcing server to perform the partial decryption step. The partially decrypted ciphertext is sent to the data user. Now, the data user can decrypt it using the secret key and can recover the actual data.

3.1. Proposed Scheme

The proposed model comprises different phases, such as system setup, encryption, key generation, transformation key generation, partial decryption and decryption. The traditional CPABE system [2] is relied upon to implement these phases.

The system setup phase is responsible for generating the public key (PK) and the master key (MK). The trusted authority generates these keys by taking global system parameters as input. Let $\mathbb{G}$ be the bilinear group of prime order p with generator g and there is a bilinear map as e : $\mathbb{G} \times \mathbb{G}$. Then, the two elements α and β are selected over $\mathbb{Z}$p. PK is computed as PK $= \left[\mathbb{G}_0, g, g^{\beta}, e\left(g,g\right)^{\alpha}\right]$. MK is computed as MK $= g^{\alpha}$.

The hierarchical access structure $\mathbb{A}$ serves as the input for encryption, in the form of access tree, as well as PK and the messages to be encrypted $\{m_1, \ldots m_n\}$. Multiple messages are encrypted by applying the hierarchical access structure and the ciphertext is generated. The polynomial q_{node} and threshold value k are associated with each node of the access tree. In the access tree, the attributes are present at leaf nodes and their threshold value is 1. The threshold values for AND and OR gate are set to 2 and 1, respectively. The degree of the polynomial associated with each node is set to one less than the corresponding threshold value.

Let us assume there are n messages to be encrypted and m attributes. The secret value s for the root node of the hierarchical access structure is selected randomly, such that $s \in \mathbb{Z}p$. Let s_i be the secret associated with the root node of the access structure meant for message m_i. The ciphertext components for i-th message are computed as $CT_i' = m_i . e\left(g,g\right)^{\alpha s_i}$ and $CT_i^* = h^{s_i}$, where $h = g^{\beta}$.

In access tree the attributes are present at leaf nodes. For $k = 1, \ldots, m$ attributes and for each such node we compute $C_{i,k} = g^{q_k(0)}$ and $C'_{i,k} = H\left[\text{attribute}(k)\right]^{q_k(0)}$, where H is the hash function.

Finally, the generated ciphertext is represented as:

$$
\begin{aligned}
CT = \Big\{ & \mathbb{A}, \{m_1, \ldots, m_n\} \ , \\
& CT_i', CT_i^*, C_{i,k}, C'_{i,k} \ , \\
& \forall i \in (1,n) \ , \\
& \forall k \in (1,m) \Big\} \ .
\end{aligned}
$$

In the key generation phase, SK is sent to the user by taking MK and set of attributes A as input. Let there be m attributes in set $A = \{a_1, a_2, \ldots, a_m\}$. The random element x over $\mathbb{Z}p$ is selected and for each attribute a_j, a random element y_j over $\mathbb{Z}p$ is also selected. The secret key component is computed as $SK' = \left(g^{\alpha+x}\right)^{\frac{1}{\beta}}$. For each attribute a_j, $S_j = g^x . H\left(a_j\right)^{y_j}$ and $S_j' = g^{y_j}$ is computed. The secret key is represented as $SK = \left\{SK', S_j, S_j', \forall j \in (1,m)\right\}$.

Next, the transformation key (TK) is needed to support the outsourced decryption. It is used by the outsourcing server to partially decrypt the ciphertext. PK and SK are taken as input and TK is generated as output by this phase. The random element z over $\mathbb{Z}p$ is selected and for each attribute a_j random element y_j over $\mathbb{Z}p$ is selected. Then, $D = \left(SK'\right)^{\frac{1}{z}}$ is computed and for each attribute a_j, $D_j = g^{\frac{x}{z}} . H\left(a_j\right)^{\frac{y_j}{z}}$ and $D_j' = g^{\frac{y_j}{z}}$.

The partial decryption phase takes the ciphertext (CT) and TK as input and generates the partially decrypted ciphertext CT''. Complex paring computations are performed in this phase by the outsourcing server. Partial decryption is performed only when the user's attributes satisfy the necessary access structure. If the access structure is not satisfied by the user's attributes, then the phase returns null. If the access structure is satisfied, then this phase recovers value $e\left(g,g\right)^{\frac{xs_i}{z}}$ for each message. The transformed ciphertext for each ciphertext CT_i is computed as $CT_i'' = \left(CT1_i, CT2_i\right)$. Here, $CT1_i = CT_i'$ and

$$
CT2_i = \frac{e\left(CT_i^*, D\right)}{e\left(g,g\right)^{\frac{xs_i}{z}}} \ .
$$

In decryption phase the transformed ciphertexts CT_i'', CT and the TK are taken as input and the actual data is recovered. Now, for complete decryption, a simple operation is required as:

$$
m_i = \frac{CT1_i}{\left(CT2_i\right)^z} \ .
$$

4. Performance Analysis

The proposed hierarchical access structure and the outsourced decryption-based CPABE scheme are implemented using the Java JPBC library. Table 1 presents a comparison of the proposed scheme and some existing solutions with respect to the features adopted in each of the approaches. As per this comparison, only the proposed scheme supports a hierarchical access structure and, hence, it is more efficient in terms of encryption time and storage cost.

Table 1
Comparison of features

Scheme	Access structure	Encryption time	Outsourced operation	Storage cost
[14]	LSSS	Standard	Encryption, key generation and decryption	Standard
[10], [15], [16]			Encryption and decryption	
Proposed	Hierarchical	Less	Decryption	Less

Evaluation of the performance of the proposed scheme consists in comparing it with schemes with and without outsourcing. The symmetric encryption technique is used to encrypt the messages and the key used for encryption is encrypted using the proposed CPABE scheme. The data owner generates different hierarchical access structures, as required. These access structures are used to encrypt multiple files.

Various experiments are carried out by varying the number of attributes and the time needed to encrypt and decrypt is measured. Also, the number of messages to be encrypted is varied and the performance is measured with respect to time required for encryption and decryption. The hierarchical access structure eliminates the need for generating multiple ciphertexts and, hence, considerably improves storage cost as well. Storage cost is also compared by varying the number of messages.

Comparison of the decryption time needed by the proposed scheme, the traditional CPABE approach and CPABE with a hierarchical access structure, with a varying the number of attributes, is carried out and is shown in Fig. 2. As the proposed scheme outsources the decryption to a third-party server, the time required by the user to recover the original message is much shorter when compared with the traditional system. Therefore, the computation overhead of the data user is also reduced. The number of attributes is varied from 2 to 20. It can be observed from the Fig. 2 that the proposed scheme needs less time for decryption than the two remaining methods, for any number of attributes.

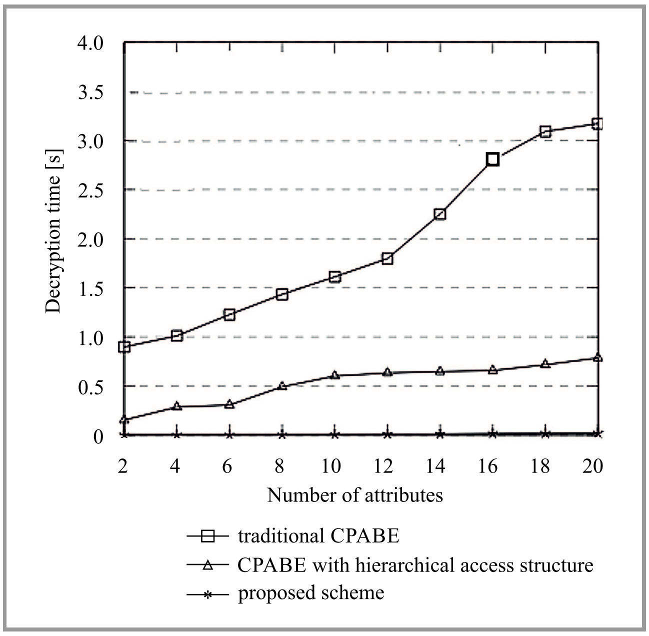

Fig. 2. Comparison of decryption time with respect to the number of attributes.

The encryption time required by the proposed method is also analyzed by varying the number of attributes. Performance is compared with the traditional CPABE approach without a hierarchical access structure (Fig. 3). It can be observed that the proposed method outperforms the other approach. This is possible because one common hierarchical access structure is used to encrypt multiple messages, instead of encrypting them separately.

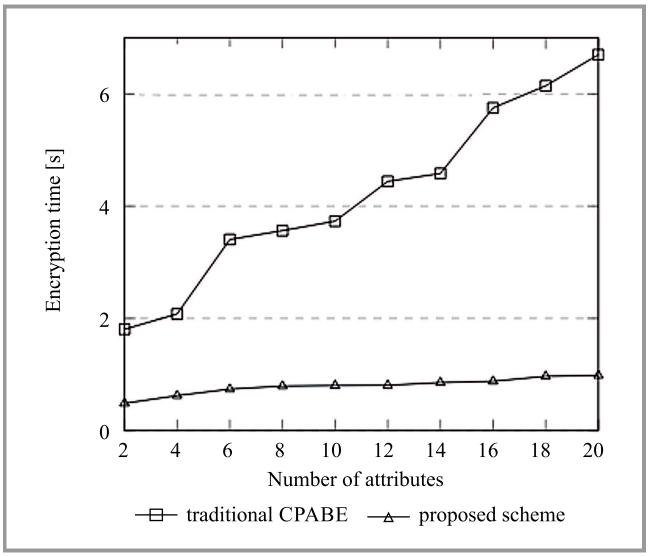

Fig. 3. Comparison of encryption time with respect to the number of attributes.

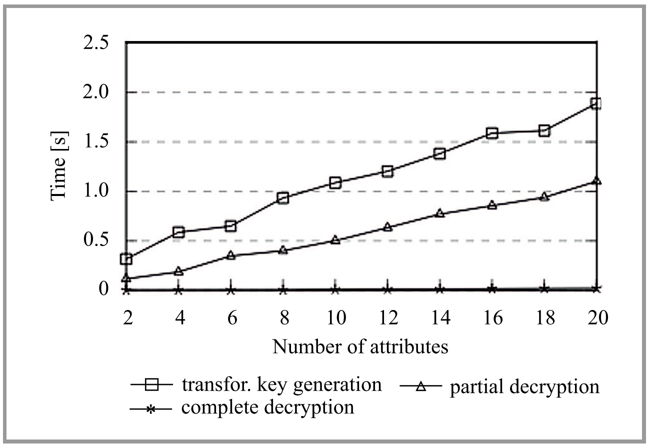

Fig. 4. Comparison of time required for decryption components with respect to the number of attributes.

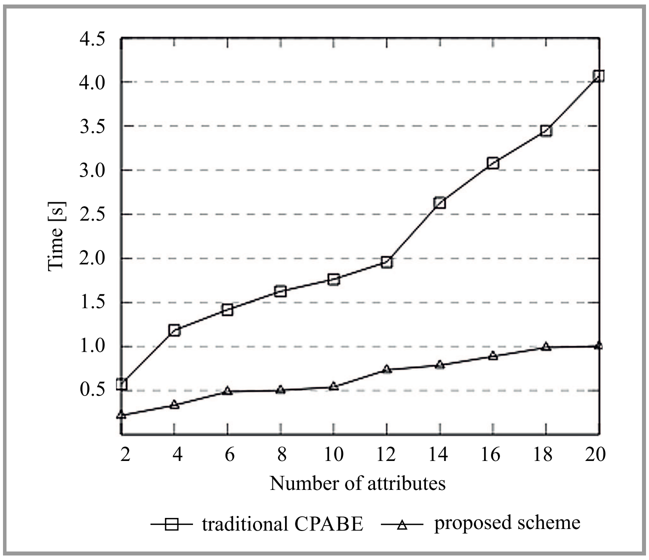

Fig. 5. Comparison of time required for private key generation with respect to the number of attributes.

In the proposed system, decryption is outsourced to an outsourcing server. The decryption process involves various components, such as transformation key generation, partial decryption and complete decryption. The comparison of time required for all these components, by varying the number of attributes, is shown in Fig. 4. It can be observed that the time required for transformation key generation and partial decryption is linearly dependent on the number of attributes. The time needed for complete decryption is almost constant, as very simple computations are performed by the data user for decrypting the partially decrypted ciphertext.

Figure 5 shows the comparison of time required for secret key generation, with different numbers of attributes, by the traditional approach and the proposed scheme. The proposed scheme takes less time to complete the operation compared with the traditional approach. The time required by the proposed scheme increases at a slower rate, compared to other methods.

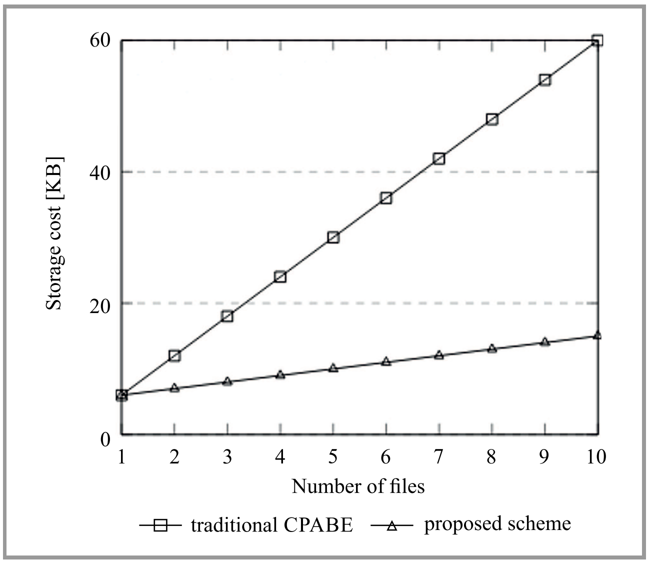

Fig. 6. Comparison of storage cost required for ciphertext with respect to the number of files.

The comparison of storage cost needed for storing ciphertext by the proposed outsourcing model and the traditional scheme, for a different number of files, is shown in Fig. 6. Ten different files with size of 1 KB each are considered for evaluating the performance. The storage cost of the proposed scheme is significantly lower compared with other methods.

5. Conclusion

The paper presents an attribute-based encryption method that utilizes the hierarchical access structure and the outsourcing mechanism. In the proposed scheme, the common hierarchical access structure is generated by combing hierarchically related access structures. Hence, multiple data can be encrypted with this common access structure and an appropriate portion of data is decrypted by satisfying the relevant portion of the access structure. This reduces the encryption time and storage space required for ciphertext. In order to deal with the complexity of the decryption operation, the outsourced decryption mechanism is presented. Here, the cloud server performs the partial decryption process and generates some intermediate ciphertext. Further, this transformed ciphertext is decrypted by the user completely. The outsourcing mechanism reduces the burden of complex operation on the data user's side and makes decryption process more efficient. Thus, the proposed scheme constitutes an efficient approach for implementing attribute-based access control for cloud data. The proposed method is efficient in terms of encryption time, decryption time and storage cost when compared with the traditional approach.

References

[1] A. Sahai and B. Waters, "Fuzzy identity based encryption", in *Advances in Cryptology – EUROCRYPT 2005. 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, R. Cramer, Ed. *LNCS*, vol. 3494, pp. 457–473. Berlin, Heidelberg: Springer, 2005 (DOI: 10.1007/11426639_27).

[2] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext policy attribute based encryption", in *Proc. IEEE Symp. on Secur. and Priv. SP'07*, Berkeley, CA, USA, 2007, pp. 321–334 (DOI: 10.1109/SP.2007.11).

[3] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute based encryption for fine-grained access control of encrypted data", in *Proc. of the 13th ACM Conf. on Comp. and Commun. Secur.*, Alexandria, VA, USA, 2006, pp. 89–98, 2006 (DOI: 10.1145/1180405.1180418).

[4] M. Green, S. Hohenberger, and B. Waters, "Outsourcing the decryption of ABE ciphertexts", in *Proc. of the 20th USENIX Conf. on Secur.*, San Francisco, CA, USA, 2011 [Online]. Available: https://www.usenix.org/legacy/event/sec11/tech/full_papers/Green.pdf

[5] M. Green, S. Hohenberger, and B. Waters, "Attribute-based encryption with verifiable outsourced decryption", *IEEE Trans. on Inform. Foren. and Secur.*, vol. 8, no. 8, pp. 1343–1354, 2013 (DOI: 10.1109/TIFS.2013.2271848).

[6] B. Qin, R. H. Deng, S. Liu, and S. Ma, "Attribute-based encryption with efficient verifiable outsourced decryption", *IEEE Trans. on Inform. Foren. and Secur.*, vol. 10, no. 7, pp. 1384–1393, 2015 (DOI: 10.1109/TIFS.2015.2410137).

[7] X. Mao, J. Lai, Q. Mei, K. Chen, and J. Weng, "Generic and efficient constructions of attribute-based encryption with verifiable outsourced decryption", *IEEE Trans on Depend. and Secure Comput.*, vol. 13, pp. 533–546, 2016 (DOI: 10.1109/TDSC.2015.2423669).

[8] J. Li, F. Sha, Y. Zhang, X. Huang, and J. Shen, "Verifiable outsourced decryption of attribute-based encryption with constant ciphertext length", *Secur. Commun. Netw.*, vol. 2017, pp. 1–11, 2017 (DOI: 10.1155/2017/3596205).

[9] J. Li, Y. Wang, Y. Zhang, and J. Han, "Full verifiability for outsourced decryption in attribute based encryption", *IEEE Trans. on Serv. Comput.*, vol. 13, pp. 478–487, 2017 (DOI: 10.1109/TSC.2017.2710190).

[10] Z. Li, W. Li, Z. Jin, H. Zhang, and Q. Wen, "An efficient ABE scheme with verifiable outsourced encryption and decryption", *IEEE Access*, vol. 7, pp. 29023–29037, 2019 (DOI: 10.1109/ACCESS.2018.2890565).

[11] J. Yu, G. He, X. Yan, Y. Tang, and R. Qin, "Outsourced ciphertext-policy attribute-based encryption with partial policy hidden", *Int. J. of Distrib. Sensor Netw.*, vol. 16, pp. 1–14, 2020 (DOI: 10.1177/1550147720926368).

[12] L. Li, Y. Zhang, X. Chen, and Y. Xiang, "Secure attribute-based data sharing for resource-limited users in cloud computing", *Computers & Secur.*, vol. 72, pp. 1–12, 2018 (DOI: 10.1016/j.cose.2017.08.007).

[13] K. Fan, J. Wang, X. Wang, H. Li, and Y. Yang "A secure and verifiable outsourced access control scheme in fog-cloud computing", *Sensors*, vol. 17, no. 7, Article no. 1695, 2017 (DOI: 10.3390/s17071695).

[14] R. Zhang, H. Ma, and Y. Lu, "Fine-grained access control system based on fully outsourced attribute-based encryption", *J. of Syst. Softw.*, vol. 125, pp. 344–353, 2017 (DOI: 10.1016/j.jss.2016.12.018).

[15] J. Zhao, P. Zeng, and K. R. Choo, "An efficient access control scheme with outsourcing and attribute revocation for fog-enabled e-health", *IEEE Access*, vol. 9, pp. 13789–13799, 2021 (DOI: 10.1109/ACCESS.2021.3052247).

[16] H. E. Gafif and A. Toumanari, "Efficient ciphertext-policy attribute-based encryption constructions with outsourced encryption and decryption", *J. Secur. and Commun. Netw.*, vol. 2021, pp. 1–17, 2021 (DOI: 10.1155/2021/8834616).

[17] K. Sethi, A. Pradhan, and P. Bera, "Practical traceable multi-authority CP-ABE with outsourcing decryption and access policy updation", *J. of Inform. Secur. and Appl.*, vol. 51, pp. 1–16, 2020 (DOI: 10.1016/j.jisa.2019.102435).

Tabassum N. Mujawar earned her M.E. in Computer Engineering from the University of Mumbai, Maharashtra, India in 2012. Currently, she is pursuing a Ph.D. in Computer Science and Engineering from Basaveshwar Engineering College, Bagalkot, affiliated to Visvesvaraya Technological University (VTU), Belgaum, Karnataka, India. She is working as an Assistant Professor at the Ramrao Adik Institute of Technology, D Y Patil deemed to be University. She has total 15 years of teaching experience. Her areas of interests include cloud computing, security and machine learning.

E-mail: tabbu3002@gmail.com
Research Scholar,
Department of Computer Science
and Engineering
Basaveshwar Engineering College,
Bagalkot, Karntaka, India

Department of Computer Engineering
Ramrao Adik Institute of Technology,
D Y Patil deemed to be University
Navi Mumbai, Maharashtra, India

Lokesh B. Bhajantri received his Ph.D. degree in Computer Science and Engineering from the Visvesvaraya Technological University (VTU), Belgaum, Karnataka, in 2015. He has been working, for the past 16 years, as an Associate Professor at the Department of Information Science and Engineering, Basaveshwar Engineering College, Bagalkot, India. His areas of interests include distributed/wireless sensor networks, cognitive Internet of Things, mobile computing and communications, networking protocols, genetic algorithms, applications of agents, as well as real time systems.

https://orcid.org/0000-0002-3947-4292
E-mail: lokeshcse@yahoo.co.in
Department of ISE
Basaveshwar Engineering College
Bagalkot, India