Received: September 8, 2021. Revised: November 16, 2021. 47

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Enhanced CP-ABE with RSA for Secure and Revocable Data Transmission of Big Data in Cloud

Rajashekar Mallajamma Basavarajegowda[1]* **Subramaniam Meenakshi Sundaram [1]**

[1]*Department of Computer Science & Engineering,*
GSSS Institute of Engineering & Technology for Women, Mysuru, Affiliated to VTU, Belagavi, Karnataka, India
* Corresponding author's Email: rajashekar@gsss.edu.in

Abstract: Cloud computing is a diversified domain and is a huge phenomenon that is a widely preferred technology in the current times. The users are unable to utilize cloud computing to the maximum, due to challenges in the security issues. To store sensitive or useful data in an insecure region is not an option for the user. Various techniques have been developed to provide security for the big data in the cloud. During voluminous data transmissions, it will not maintain the end-to-end security and the prevailing approaches do not solve the key complexities and not provides security for the keys. The Attribute-Based Encryption (ABE) can deliver a mechanism of fine-grained control by providing one-to-many encryption. But, the revocable process of accessing the privilege of encrypted data of users from cloud servers is challenging. Further, the complex process of ABE takes up huge computational time hence it becomes a burden for system users. To solve such issues, the proposed Enhanced Cipher-text Policy-based Attribute Encryption (CP-ABE) protect, broadcast and control the access of information, especially over the cloud server. In the proposed Enhanced CP-ABE method, proxy is added to securely transmit the key to the users. The Rivest Shamir Adleman (RSA) Algorithm provides security to outsourced big data in the cloud that enables public key encryption secure data when it is being sent over an insecure network such as the internet. The proposed Enhanced CP-ABE with RSA method effectively identifies the users who decrypt the ciphertexts during decryption. The experimental results show that the proposed Enhanced CP-ABE with RSA Scheme method acquires lesser encryption and decryption times of 1638 ms and 1102 ms respectively for a 256-bit key length. Whereas, the existing methods acquire encryption and decryption times of 1721ms and 1134 ms respectively.

Keywords: Attribute-based encryption, Ciphertext policy-based attribute encryption, Cloud computing, Rivest shamir adleman, Security.

1. Introduction

Cloud computing is widely utilized in many fields for the development of technology and science. Cloud computing estimates a larger scale of complex data and delivers powerful resources for internet users [1]. Due to the scaling in cloud computing, a large amount of sensitive and personal data that is stored in the cloud makes the users more concerned about the security of their data in the cloud. Cloud storage includes advantages such as low-cost and pay-as-you-use facilities [2]. Huge quantities of data are being outsourced to the cloud for preserving the storage of business and personal information. The cloud is easily accessible by the public, is not trustworthy and the outsourced data should not be leaked by the cloud service provider without the permission of the data owners [3]. Data privacy and security are vital issues and no user would like to share their documents which include sensitive details that have no guarantee of being secured [4]. The generality of the cloud will indirectly pose a threat to the confidentiality of data that are outsourced and cloud user's privacy. The problems that occur are to provide unauthorized users access to the data which is outsourced to the cloud, anytime and anywhere [5]. The solution is to utilize the encryption process on the data before uploading the data to the cloud. So, the solution reduces further data processing and sharing, because the data owners should download the data

that are encrypted from the cloud and must decrypt them before sharing [6].

Attribute-based encryption (ABE) is utilized for encryption that delivers access control and functionality of encryption together, which has attracted many researchers' attention. The ABE is a different type of public-key encryption method which implements flexible access controls to encrypted documents [7]. ABE technique is employed in many applications and serves as the base for implementing the "one to many" sharing of files and fine graining, whereas the traditional techniques provide only "one to one" encryption [8]. In ciphertext policy attribute-based encryption (CP-ABE) scheme, the size of ciphertexts increases linearly, as the number of attributes in the access structure improves the interaction of overheads in the receiver. Further, the number of pairing operations is linear with the number of attributes in accessing the policy, during the process of decryption, which increases the computational costs of the receiver. The privacy protection develops defects to a greater extent, the limitations of practical applications of the ABE approach are most important, where the bandwidth and computational resources will be lesser [9]. There are various research techniques for privacy protection based on access control, trust, and encryption. However, the existing cloud security methods are scattered to access the outsourced big data and not systematic. Hence it is required to develop an effective method to facilitate the privacy protection of cloud data [10]. To solve such issues, the proposed enhanced CP-ABE with RSA algorithm is implemented, for providing security to outsourced big data in the cloud. The ABE is applied to fine-grained control of encryption and enhanced cipher text policy is applied with ABE for privilege control of revocable process. The tree structure is applied to store the tag and encrypted data to provide efficient control in the enhanced CP-ABE with RSA. The tree structure applied in CP-ABE with RSA method helps to improves the efficiency of the encryption-decryption and access control.

The paper is organized as follows, the literature review is discussed in section 2, the proposed methodology is explained in section 3, the experimental results and discussion is described in section 4, the conclusion of the proposed method is highlighted in section 5.

2. Literature review

Premkamal [11] developed dynamic traceable CP-ABE for the storage of transformed big data in cloud platforms. The developed method included two schemes that were, dynamic revocation and dynamic tracing. In the tracing process, by using secret-key the proxy servers extract the id of users which is stored in the log files before computing the decrypted partial ciphertexts. In the revocation process, the auditing process provides security that revokes the attackers automatically. During the process of decryption, the ciphertext scheme dynamically traces the decrypt process. Somehow, during revocation the developed method did not reduce the computation overhead, as it does not modify the user's data.

Vengala [12] developed hybrid meerkat clan algorithm (HMCA) and optimized CP-ABE elliptic curves cryptography (ECC) for secured transmission of data, on the distributed cloud server. Initially, the user data features were extracted by utilizing a cloud server and deduplication of data. The HMCA selects the cloud server which was centered optimally by the features. Then, the SHA512 algorithm was utilized for deduplication of user data. Thus, in the future the developed model required improvement in compression and encryption by utilizing 2 level lempelziv algorithm. The developed method performed secure file uploads and downloads for only single data owners.

Xue [13] developed enhanced linear secret sharing scheme (LSSS) matrix expressions which were integrated with CP-ABE. In the LSSS method, a user can store the parameters about sub-policies by decrypting the user data, where it was reused in the further embedded access policies include the same sub-policy to significantly reduce the computation cost. Under chosen plaintext attacks the developed scheme was semantically secure and preserved the prudence of the data-sharing system. It reduced the decryption time and consumed lesser storage overhead, thus effectively it promotes efficiency. However, the computation cost of the user's future decryption can be largely reduced, as there is no need to re-decrypt the portion identified by the parameters

Wang [14] established the CP-ABE method for the privacy and security of data in mobile networks and healthcare. The developed CP-ABE method was provided a terminal of mobile devices with low computational and storage power. In the developed CP-ABE method delivered an expensive and computational process for the semi-trusted parties third by knowing the fixed number of local computations. The developed CP-ABE provided data privacy and security for sharing the data in an open environment which satisfied the utility of terminal mobile devices with lesser storage and computational power. The challenges in the developed method was that it reduced the size of text and not provided safety and privacy for healthcare information.

Zhao [15] established revocable and efficient storage based on the CP-ABE technique for outsourced data by using decryption as well as the fixed size of cipher and secret key. The theorem of the chinese remainder was used to achieve revocability for storage. To reduce the process of security operations, the ABE technique with a fixed size of a cipher as well as a secret key was utilized. The developed CP-ABE scheme was revocable and efficient for the storage of outsourced data decryption and constant-size cipher texts and secret keys. However, using the developed more stringent revocable storage schemes for updating the cipher texts was required for further enhancing the security and generality of the system. The developed model considered a secure outsourcing for the encryption algorithm that improved the system performance.

Banerjee [16] developed multiple authority CP-ABE methods for control of user access with a fixed-size key and ciphertext for the deployment of IoT. The developed method is a 3-factor access control of users that supports the multiple authority of ABE and was scalable in the smart card of users. The ciphertext size that was required for the request of authentication was fixed with specific attributes. The developed scheme needed lesser computational costs for the limited resources with smart devices but needed significant improvement in communication.

Ma [17] developed CP-ABE based secure and verifiable data deletion in cloud. The developed model performed secured data deletion and verification (SDVC) technique which was based on data verification and deletion. The developed model achieved fine grained secure data using the approach. However, future work is to design effective method to implement secure data deletion and verification for mobile devices under 5G environment.

Shen [18] developed a privacy-preserving attribute-based encryption system for data sharing in smart cities. The developed model performed fine-grained data access control scheme based on CP-ABE to implement access policies with a greater degree of expressiveness as well as hidden policies from curious cloud service providers. However, in the future the unequal attribute updating and revocation of all those operations influenced a large number of users and raise computation costs. For security and privacy parts, the problem of unequal attribute information leakage was required to be overcome.

3. Proposed methodology

The proposed and enhanced CP-ABE with RSA scheme performs operations in 5 phases named as: setup, user key generation, data encryption, data decryption and revocation. The proposed enhanced CP-ABE with RSA algorithm scheme is secure against the outsourced big data in the cloud which is briefly described in this section. The block diagram of the proposed method is shown in Fig. 1.

The proposed enhanced CP-ABE method includes five performers that are data owners, users of data, trusted authority, cloud server and server proxy.

Data owner: The data owner is a trusted performer who manages the encryption of data and stores the privacy of data before uploading. The data owner forwards and shares the received outsourced information from users, according to the defined access policy. The data owners set up the limit of threshold access for data that is outsourced, to identify the malicious access.

Data users: The users of data get the plaintext from the obtained ciphertext only if the attributes are satisfiable with the access policies and trusted entities.

Trusted authority: TA creates the master-secret key (MSK), User-secret key (US), Public key (PK), and User identification (UID) number. The TA makes sure that the users of data are genuine people and TA also witnesses the proofs before forwarding the secret key.

Cloud server: CS includes a larger capacity for storage where the owners will store the details and provide data accessibility for non-revocable users. Further, the cloud server maintains the server proxy.

Proxy server: The PS is the entity that is dedicated to an organization in the environment of the cloud. The proxy server will correctly carry out the task assigned and gets the plain text. becomes curious to get the plain text. The PS updates the ciphertexts

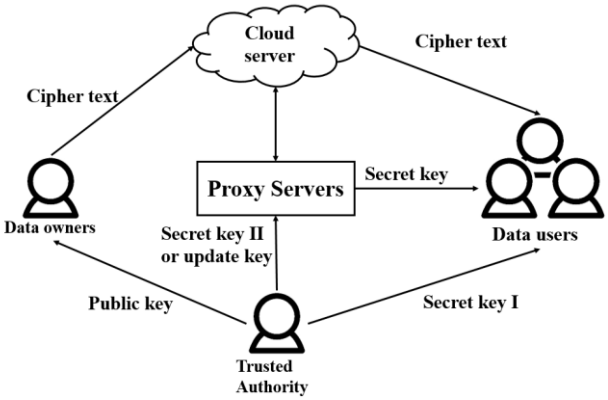

Figure. 1 The block diagram of the proposed enhanced CP-ABE with the RSA method.

as well as a secret key which needs to be done by the data owner and user. At the time of revocation, the proxy server stores one part of the user's secret key.

The proposed and enhanced CP-ABE with RSA algorithm performs the 5 phases named as setup, user key generation, data encryption, decryption, and revocation. The algorithm for secret key generation is given below:

3.1 Secret key generation

The trusted authority generates the MSK, PK, and numbers for UID by utilizing the algorithms of user setup. The TA obtains parameters for security which defines new PK and MSK that is considered as a set of universal attributes. After the PK and MSK generation, the trusted authority forwards PK to the owner of data and stores MSK. The trusted authority develops a generic UID number for every user who enters and adds their ID to the cloud. Initially, the trusted authority retrieves the information from the cloud and forwards the data as input for an algorithm that further will be stored in the cloud. The trusted authority generates the secret key for every user by utilizing the algorithm of key generation. After generating the secret key, the trusted authority forwards one part to the user and the other to the proxy server. The secret key's part will be stored in the server of the proxy, by which is not possible to decrypt the information due to the presence of the secret key with a user.

3.2 Encrypting the data

After obtaining the PK and MSK, the trusted authority TA stores the MSK for the generation of a secret key. Further, the trusted authority forwards the public key to owners for the process of encryption by utilizing the CP-ABE technique. The attribute of ciphertext policy according to encryption technique includes 4 kinds of algorithms known as setup, encryption, key generation, and decryption.

Set-up: The setup algorithm considers only security implicit parameters as input like PK and MSK

Encryption: Initially, the encryption algorithm considers the PK, message M and access structure A as input over the attributes. The algorithm encrypts the M and obtains ciphertext which can only be possessed by users that satisfy the structure of access and can perform message decryption. There is an assumption that ciphertext will contain A implicitly.

Key-generation: The algorithm of key generation selects the MK and attributes as input which describes the key and gives the output as SK.

Decryption: The decryption algorithm selects public parameter, ciphertext, A access policy and SK as input. If the access structure is satisfied with A then the decryption of ciphertext takes place and obtains M.

G_0 is considered as a prime order p for bilinear group and $e: G_0 \times G_0 \to G_1$ is the map of the bilinear technique. The security parameters such as κ identify the group size. The range of logarithmic coefficient is described as $\Delta I, S$ for $i \in Zp$ and element S is considered in such a way that $Z_p = \Delta \mathrm{I}, \mathrm{S(x)} = \mathrm{Q_j} \in S, j_6 \neq i\frac{x-j}{i-j}$ which further utilizes the hash in $H: \{0,1\} * \to G_0$ functions which is the random oracle. The hash function maps the attributes which are considered as binary strings with the random set of elements.

Setup: The setup algorithm selects G_0 of a bilinear group with a p prime order and g generator. Then, it selects exponents randomly such as $\alpha, \beta \in Zp$ and MK is β, g and hash value h which is explained as shown in Eq. (1).

$$PK = G_0, g, h = g\,\beta, f = g\,1/\beta, e(g,g)^{\alpha} \quad (1)$$

Encryption: The message M is encrypted by an encryption algorithm by utilizing the access of the tree structure. Initially, the algorithm selects the q_x as polynomial for every x node with the leaves. The polynomials are selected in a top-down approach from the root. In every node, the degree d_x is set with polynomials lesser than the threshold value k_x in a particular node such as $d_x = k_x - 1$ and H is hash function. Beginning with the root, the algorithm of encryption selects $s \in Zp$ randomly and considers $qR(0) = s$. Further, $d_R$is selected with a different polynomial point which is random to define. For a different node, the $qx(0) = qparent(x)(index(x))$ is set and d_x is selected as the different point in order define q_x completely. The T is the message and y is considered as leaf node in the tree structure. At last, ciphertext CT is developed by giving the access structure to the tree and ciphertext is estimated as shown in Eq. (2).

$$CT = (T, \check{C} = M(g,g)^{\alpha,s}, C = h^s, \forall_y \in Y: C_y = g^{q_y(0)}, C_y^{|} = H\big(att(y)^{qy(0)}\big). \quad (2)$$

Key Gen: The generation of key algorithm considers input as an attribute and gives keys which identifies the group as output and child node is denoted as D. Initially, the random $r \in Zp$ is

selected with $r_j \in Zp$ for all the attributes $j \in S$ which estimates the key as shown in Eq. (3).

$$SK = \left(D = g^{\frac{\alpha+r}{\beta}}, \forall_j \in S: D_j = g^r . H(j)^{r_j}, D_j^l = g^{rj}\right) \quad (3)$$

Decrypt(CT,SK): The decrypt node(CT, SK, x) is defined initially which considers $CT = (T, C, \tilde{C}, \forall_y \in Y : C_y, C'_y)$ cipher text, SK, S attributes, and node from the tree as input. When the node is a leaf node in the tree then $i = att(x)$ is defined in such a way that $i \in S$ which is employed in Eq. (4-6).

$$DecryptNode(CT, SK, x) = \frac{e(Di, Cx)}{e(D'i, C'x)} \quad (4)$$

$$DecryptNode(CT, SK, x) = \frac{e\ g\ r \cdot H(i)\ ri, hqx(0)}{e(g\ ri, H(i)\ qx(0)} \quad (5)$$

$$DecryptNode(CT, SK, x) = e(g, g)\ rqx(0) \quad (6)$$

The decryption algorithm proceeds for every node having children nodes D_i to call the process of decryption and save the output. The arbitrary-sized child nodes (Fz) in such a way that $Fz = \perp$ will not exist it does not satisfy the decryption process and returns the function $\perp$.

3.3 Decrypting the data

After encrypting the data, the data is decrypted by using the RSA algorithm. In the proposed RSA algorithm, the generated keys are preserved in the file by utilizing the base-64 encoding approach and the values are stored securely. The proposed RSA key generation includes the utilization of two random numbers and two prime numbers. The encryption values are dependent on the value of public key components (N) and the decryption values are dependent on private key components (M). The value of N is similar to the product of 2 prime numbers and two random numbers. The value of M is similar to the product of 2 prime numbers. The encryption and decryption are carried out by using modulus value such as (e, N). . The value of M will not depend on the value of N, so the attackers who know the details of N will not be able to identify the prime number and cannot determine the value of decryption (d, M). The size of prime numbers should be of similar sizes and the generated keys are stored in the files by utilizing base-64 encoding techniques and the values are stored securely. The encryption is undergone by utilizing components of the public key and decryption is undergone by utilizing components of a private key. The proposed RSA algorithm for key generation, encryption and decryption is explained below.

The RSA algorithm for the generation of key

Input

Two prime numbers such as p, q and $p \neq q, p, q > 3$

Output

The components of public key $\{e, N\}$
The components of private key $\{d, M\}$

Procedure

$N \leftarrow p \times q \times (p-1) \times (q-1)$
$M \leftarrow p \times q$

The random integers are selected $r, p > 2^r < q$

Compute private key generation $\emptyset$ value of N
$\emptyset(N) \leftarrow \frac{(p-1)\times(q-1)\times(p-2^r)\times(q-2^r)}{2^r}$
Select a random number e, such that
$1 < e < \emptyset(N)$ and $\gcd\left(e, \emptyset(N)\right) = 1$
Calculates the random number d in such a way that

$$d * e = 1 * mod(\emptyset(N))$$

The RSA algorithm for encryption

Input

The plain text message, $P < N$
The public key component :$\{e, N\}$

Output

The cipher text, C

Procedure

The cipher text, $C \leftarrow P^e\ mod\ N$

The RSA algorithm for decryption

Input

The cipher text message, C
The private key components: $\{d, M\}$

Output

Decrypted plain text,P

Procedure

$P \leftarrow C^d \ mod \ M$

The cloud environment checks whether the users are malicious or not by utilizing the id of users. The malicious users will not obtain the ciphertexts from a cloud environment. The user gets the part of the secret key after obtaining ciphertext from the cloud environment and computes decryption. In the revocation process, the ciphertext will get the update from PK and MSK. When a new user joins the secret key will be generated from the master key and users will be able to access the published data and achieve the secrecy in reverse order.

4. Experimental setup and results

In this section, the experimental result of the proposed enhanced CP-ABE with the RSA method, used for providing security for outsourced big data in the cloud, is discussed. The validation of the proposed enhanced CP-ABE with the RSA method is carried out by using larger data files that need to be stored in the servers of the cloud. In this research, the proposed work is compared with the benchmark models for secure public auditing of data in the cloud. The proposed enhanced CP-ABE with RSA method is simulated using Java with system requirements of Windows 10 operating system, RAM of 128GB, Intel Core i9 processor and hard disk of 4TB. The attributes considered in the proposed CP-ABE with the RSA method are location coordinates, department, time, user id, secret code, random number. The proposed CP-ABE with the RSA method is authenticated in terms of an attribute, properly signed by the trusted authority.

The parameters considered to evaluate the proposed and enhanced CP-ABE with the RSA method is explained in this section. The quantitative and comparative analysis of the proposed and enhanced CP-ABE with the RSA method with the existing methods is also described in this section.

4.1 Performance metrics

The proposed enhanced CP-ABE with RSA scheme method for dynamic and secure public auditing of user data in the cloud, is evaluated and compared with the existing methods. The parameters such as encryption, decryption and execution time are utilized to evaluate the effectiveness of the proposed enhanced CP-ABE with RSA method and the existing methods which are discussed as follows:

- **Encryption time:** The encryption time is utilized to calculate the throughput of a model which gives the speed details of encryption. The throughput of the encryption model is calculated by using encryption of overall plain text in bytes divided by the encryption time.
- **Decryption time:** The time consumed to convert the encrypted data into its original format is known as decryption time. It is generally the reverse process of encryption that decodes the encrypted data so that only authorized users will be able to decrypt using a secret key.
- **Execution time:** The execution time is also called as central processing unit (CPU) time that is spent by the system to execute the task or it is considered as the time taken for the program to complete or terminate.

4.2 Quantitative analysis

The results obtained by the proposed and enhanced CP-ABE with RSA method that is used for providing security against selected plain text and collision attacks of users in the cloud are tabulated in terms of encryption, decryption and execution time as shown in table 1. It depicts the amount of time required (in milliseconds) to perform the task, considering the amount of data

Table 1 shows the quantitative analysis of the proposed CP-ABE method in terms of encryption, decryption, and execution time for secured user data in the cloud. The outcomes are obtained for 10 executions with different key lengths: 8, 16, 32, 64,128 and 256 in bits. The time taken by the proposed enhanced CP-ABE method for encrypting the data files is 1638ms for 256 key length sizes in bits. The time taken by the proposed enhanced CP-ABE with the RSA method for decrypting the encrypted data is 1102 ms for 256 key length size in bits. The time taken by the proposed enhanced CP-ABE method for executing the complete task is 3437 ms for 256 data sizes in bits. If we use the key length beyond 256 bits (means 512 bits) it consumes more time for encryption and decryption and also

Table 1. The quantitative analysis of proposed enhanced CP-ABE with RSA method in terms of encryption, decryption and execution time.

Key Length	Encryption Time	Decryption Time	Execution Time (ms)
8	21070	20299	41977
16	14994	14382	29990
32	7795	7394	15822
64	4436	3813	8856
128	2544	2019	5198
256	1638	1102	3437

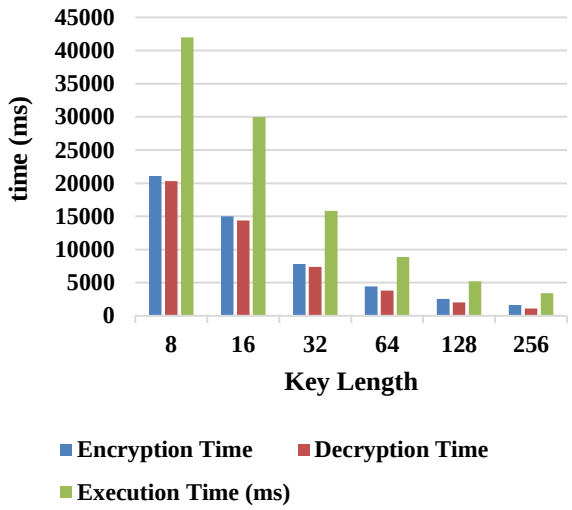

Figure. 2 Proposed enhanced CP-ABE with RSA method for encryption, decryption and execution time

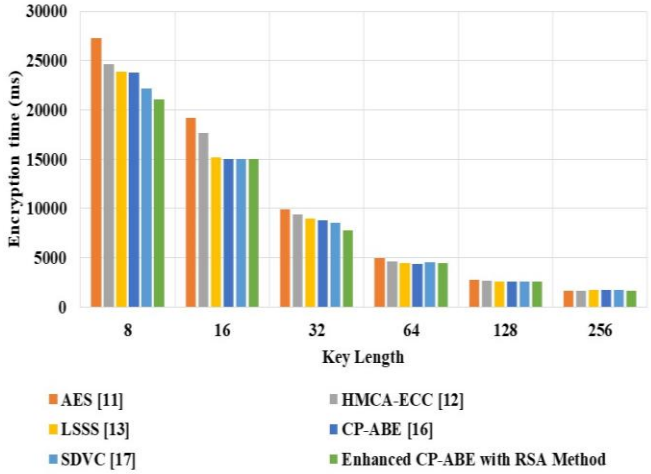

Figure. 3 Comparison of proposed enhanced CP-ABE with RSA method and existing methods for encryption time

communication costs would invariably increase. The graphical representation of the proposed CP-ABE with the RSA method in terms of encryption, decryption and execution time is shown in Fig. 2.

4.3 Comparative analysis

The obtained results of the proposed CP-ABE with RSA scheme are compared with the results obtained for previous methods such as AES and CP-ABE with respect to dynamic and secure public auditing of user data in the cloud. The comparison is done for different sizes from 8 to 256 bits of key length in terms of encryption and decryption time. The encryption time comparison with existing methods is shown in table 2.

Table 2 shows the encryption time of the proposed enhanced CP-ABE with RSA method with the existing AES and CP-ABE methods. The encryption time is taken by approaches used to secure public auditing of user data files in the cloud increases as the data bits' size increase. The proposed method acquired lesser time to encrypt the data than the existing methods and is suitable for providing the confidentiality of user's sensitive outsourced data. The proposed method acquired an encryption time of 1638 ms for the key length of 256 sizes in bits. Whereas, the existing AES and CP-ABE acquired 1648 ms and 1721 ms for the key length of in bits. The graphical representation of a comparison of encryption time for proposed CP-ABE with RSA scheme with existing methods is shown in Fig. 3.

Table 2. The comparison of encryption time with existing methods for different key lengths

Key Length	AES [11]	HMCA-ECC [12]	LSSS [13]	CP-ABE [16]	SDVC [17]	Enhanced CP-ABE with RSA Method
8	27241	24631	23834	23783	22151	21070
16	19158	17643	15203	15010	15006	14994
32	9920	9370	8963	8794	8526	7795
64	4972	4631	4421	4365	4512	4436
128	2742	2621	2567	2542	2556	2544
256	1648	1647	1742	1721	1713	1638

Table 3. The comparison of decryption time with existing methods for different key length.

Key Length	AES [11]	HMCA-ECC [12]	LSSS [13]	CP-ABE [16]	SDVC [17]	Proposed Enhanced CP-ABE with RSA Method
8	24747	24316	23247	23005	22378	20299
16	17873	16423	15642	14512	14453	14382
32	9042	8061	7783	7592	7431	7394
64	4593	4326	3921	3836	3821	3813
128	2265	2167	2126	2053	2035	2019
256	1175	1163	1137	1134	1117	1102

Table 4. The comparison of execution time with existing methods for different key length.

Key Length	AES [11]	HMCA-ECC [12]	LSSS [13]	CP-ABE [16]	SDVC [17]	Proposed Enhanced CP-ABE with RSA Method
8	52617	50341	47234	46717	45436	41977
16	37509	34216	32016	30112	30106	29990
32	19468	17482	16995	16993	16473	15822
64	10061	9234	8921	8859	8858	8856
128	5494	5361	5206	5196	5196	5198
256	3449	3452	3450	3459	3453	3437

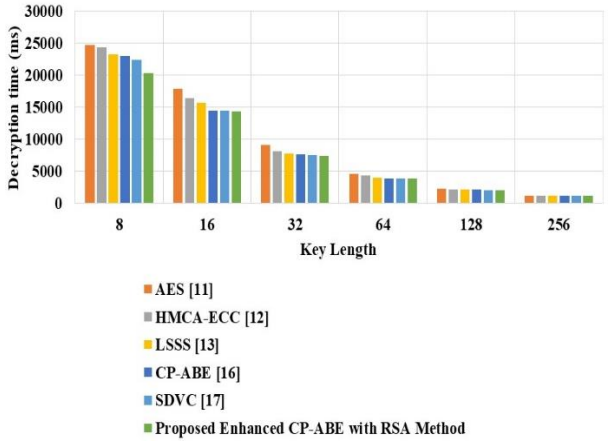

Figure. 4 Proposed enhanced CP-ABE with RSA method for decryption time with existing methods.

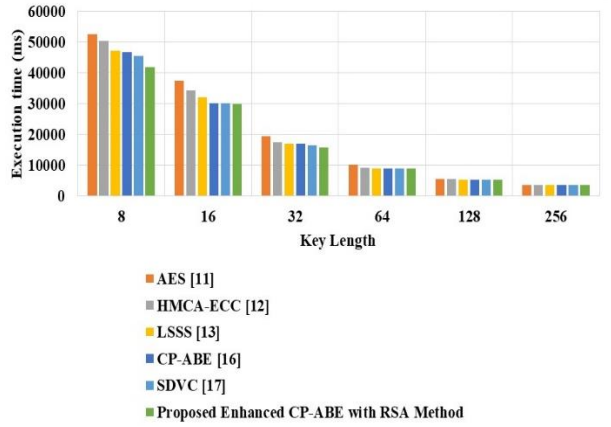

Figure. 5 Proposed enhanced CP-ABE with RSA method for execution time with existing methods.

Table 3 shows the decryption time of the proposed enhanced CP-ABE with the RSA method with the existing AES and CP-ABE. The decryption time is taken by approaches used to secure user data files in the cloud decreases as the size of data bits' increases. The proposed method acquired lesser time to decrypt the encrypted data than the existing methods and is suitable for providing the confidentiality of user's outsourced sensitive data. The proposed method acquired a decryption time of 1102 ms for the key length of 256 sizes in bits. Whereas, the existing AES and CP-ABE acquired 1175 ms and 1134 ms for the key length of 256 in bits. The graphical representation of a comparison of decryption time for the proposed Enhanced CP-ABE with RSA scheme method with the existing method is shown in Fig. 4.

Table 4 shows the execution time of the proposed enhanced CP-ABE with the RSA method and the existing AES and CP-ABE. The execution time taken by approaches used to secure user data files in cloud, reduces as the size of data bits' increases. The proposed method acquired lesser time for executing the task than the existing methods and is suitable for providing the confidentiality of user's outsourced sensitive data. The proposed method acquired an execution time of 3437 ms for the key length of 256 sizes in bits. Whereas, the existing AES [11], HMCA-ECC [12], LSSS [13], CP-ABE [16] and SVDC [17] achieved 3449 ms, 3452 ms, 3450 ms, 3459 ms and 3453 ms. for the key length of 256 in bits. The graphical representation of a comparison of execution time for proposed enhanced CP-ABE with RSA method with existing methods is shown in Fig. 5.

5. Conclusion

To provide security for outsourced big data in the cloud, an enhanced CP-ABE with the RSA method was proposed in this research paper. The proposed enhanced CP-ABE with the RSA algorithm effectively identifies the user who decrypts the ciphertexts during decryption. The TA creates PK, MSK and UID by utilizing the setup algorithm. The trusted authority forwards the secret key to the user and saves it in the server proxy. The data owner encrypts the information and forwards the ciphertext to the cloud environment by utilizing the CP-ABE algorithm. The user's sensitive encrypted data is decrypted by using the RSA algorithm. The cloud environment checks whether the users are malicious or not by utilizing their IDs. The malicious users could not obtain the ciphertexts from the cloud environment. The user gets the part of the secret key fter obtaining ciphertext from the cloud environment and then computes the decryption. In the revocation process, the cipher text will get the update from PK and MSK. The experimental results show that the

proposed and enhanced CP-ABE with RSA scheme acquired lesser implementation time for the encryption and decryption processes. The enhanced CP-ABE with RSA method has execution time of 3437 ms and existing CP-ABE method has execution time of 3459 ms. The proposed method has effective performance due to the tree structure and proposed method privilege control. In future work, the RSA algorithm can be improved to increase the result performance by providing more security for outsourced data.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

The paper conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, writing—original draft preparation, writing—review and editing, visualization, have been done by 1st author. The supervision and project administration, have been done by 2nd author.

References

[1] S. Xue and C. Ren, "Security Protection of System Sharing Data with Improved CP-ABE Encryption Algorithm under Cloud Computing Environment", *Automatic Control and Computer Sciences*, Vol. 53, No. 4, pp. 342-350, 2019.

[2] 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.

[3] S. Han, K. Han, and S. Zhang, "A data sharing protocol to minimize security and privacy risks of cloud storage in big data era", *IEEE Access*, Vol. 7, pp. 60290-60298, 2019.

[4] J. Ning, Z. Cao, X. Dong, K. Liang, L. Wei, and K. K. R. Choo, "Cryptcloud+: secure and expressive data access control for cloud storage", *IEEE Transactions on Services Computing*, Vol. 14, No. 1, pp. 111-124, 2018.

[5] H. Tian, F. Nan, H. Jiang, C. C. Chang, J. Ning, and Y. Huang, "Public auditing for shared cloud data with efficient and secure group management", *Information Sciences*, Vol. 472, pp. 107-125, 2019.

[6] J. R. Gudeme, S. K. Pasupuleti, and R. Kandukuri, "Attribute-based public integrity auditing for shared data with efficient user revocation in cloud storage", *Journal of Ambient Intelligence and Humanized Computing*, Vol. 12, pp. 2019-2032, 2020.

[7] R. Mishra, D. Ramesh, and D. R. Edla, "BB-tree based secure and dynamic public auditing convergence for cloud storage", *The Journal of Supercomputing*, Vol. 77, No. 5, pp. 1-40, 2020

[8] G. Wu, Y. Mu, W. Susilo, F. Guo, and F. Zhang, "Threshold privacy-preserving cloud auditing with multiple uploaders", *International Journal of Information Security*, Vol. 18, No. 3, pp. 321-331, 2019.

[9] Y. Zhang, J. Li, and H. Yan, "Constant size ciphertext distributed CP-ABE scheme with privacy protection and fully hiding access structure", *IEEE Access*, Vol. 7, pp. 47982-47990, 2019.

[10] P. J. Sun, "Privacy protection and data security in cloud computing: a survey, challenges, and solutions", *IEEE Access*, Vol. 7, pp. 147420-147452, 2019.

[11] P. K. Premkamal, S. K. Pasupuleti, and P. J. A. Alphonse, "Dynamic traceable CP-ABE with revocation for outsourced big data in cloud storage", *International Journal of Communication Systems*, Vol. 34, No. 2, p. e4351, 2020.

[12] D. V. K. Vengala, D. Kavitha, and A. P. Kumar, "Secure data transmission on a distributed cloud server with the help of HMCA and data encryption using optimized CP-ABE-ECC", *Cluster Computing-The Journal of Networks Software Tools and Applications*. Vol. 23, No. 3, pp. 1683-96. 2020.

[13] K. Xue, N. Gai, J. Hong, D. Wei, P. Hong, and N. Yu, "Efficient and Secure Attribute-based Access Control with Identical Sub-Policies Frequently Used in Cloud Storage", *IEEE Transactions on Dependable and Secure Computing*, 2020.

[14] S. Wang, H. Wang, J. Li, H. Wang, J. Chaudhry, M. Alazab, and H. Song, "A Fast CP-ABE System for Cyber-Physical Security and Privacy in Mobile Healthcare Network", *IEEE Transactions on Industry Applications*, Vol. 56, No. 4, pp. 4467-4477, 2020.

[15] Y. Zhao, M. Ren, S. Jiang, G. Zhu, and H. Xiong, "An efficient and revocable storage CP-ABE scheme in the cloud computing", *Computing*, Vol. 101, No. 8, pp. 1041-1065, 2019.

[16] S. Banerjee, S. Roy, V. Odelu, A. K. Das, S. Chattopadhyay, J. J. Rodrigues, and Y. Park, "Multi-Authority CP-ABE-Based user access control scheme with constant-size key and ciphertext for IoT deployment", *Journal of Information Security and Applications*, Vol. 53, p. 102503, 2020.

[17] J. Ma, M. Wang, J. Xiong, and Y. Hu, "CP-ABE-Based Secure and Verifiable Data Deletion in Cloud", *Security and Communication Networks*, Vol. 2021, 2021.
[18] X. Shen, C. Huang, D. Wang, and J. Shi, "A Privacy-Preserving Attribute-Based Encryption System for Data Sharing in Smart Cities", *Wireless Communications and Mobile Computing*, Vol. 2021, 2021.