

Hierarchal attribute based cryptographic model to handle security services in cloud environment: a new model

Banavathu Rajarao¹, Meruva Sreenivasulu²

¹Department of Computer Science and Engineering, Jawaharlal Nehru Technological University Anantapur, Ananthapuramu, India

²Department of Computer Science and Engineering, K.S.R.M College of Engineering, Affiliated to Jawaharlal Nehru Technological University Anantapuramu, Ananthapuramu, India

Article Info

Article history:

Received Dec 9, 2022

Revised Mar 15, 2023

Accepted Apr 7, 2023

Keywords:

Access control policy
Authentication based privacy
Distributed computing
Personal identification based cryptographic approach
Privacy-based data sharing
Secure model

ABSTRACT

The sharing of information in the cloud is a unique element of the environment, but there is a risk that the information may land with the wrong people. To counterattack this problem, security-associated methodologies were used to secure the information that was readily available to clients. Despite the lack of benefits, this provides productive/adaptability and dependability in access control strategies between clients in the sharing of information. The novel hierarchal attribute-based cryptographic security model (NHACSM) is being proposed to provide adaptability, versatility, and access control in sharing information in the appropriate climate. This model allows clients to share information in a hierarchal way, allowing for a productive assessment of access control strategy and improved security. The NHACSM method is used to reduce the total time values for different user instances compared to conventional approaches, for example, attribute-set-based encryption (ASBE), key-policy attribute-based encryption (KP-ABE), and ciphertext-policy attribute-based encryption (CP-ABE). With respect to 10 instances existing methods achieve 2.7, 2.5, and 2.3 respectively, and also compared to 20, 30, 40, and 50 instances, our proposed method is low. The encryption and decryption time evaluation values and performance evaluation of different approaches, ASBE, CP-ABE, were taken into account when increasing the user instance.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Banavathu Rajarao

Department of Computer Science and Engineering, Jawaharlal Nehru Technological University Anantapur
Ananthapuramu, 515002, India

Email: b.rajarao1207@gmail.com

1. INTRODUCTION

Distributed computing is a science-based application that provides flexibility and flexibility to meet customer demands [1]. Through cloud clients, it reduces data owners' computational costs by using the cloud's robust, flexible resources [2]. A cloud service provider (CSP) provides cloud-based services such as software as a service (SAAS), platform as a service (PAAS), and infrastructure as a service (IAAS) for all cloud customers [3]. CSP is a cloud security company that helps clients control cloud-based productivity [4]. It saves cloud worker data and executes complex data management procedures to protect it from CSP or ridiculous [5]. Cloud insurance challenges revolve around data security [6].

Data protection is not a security concern, but good acquired access control is provided [7]. A portion of therapeutic benefits and other affiliated affiliations must be analyzed to create a plan or address distinct positions to a social event. System application and product (SAP) is used to access data comprehensively and work with and isolate data benefits to all bundle clients [8]. Different methods have

been used to examine individual customer assignments and pack customer arrangements for cloud data sharing.

Attribute-based encryption (ABE) is a method of secure verification using symmetric key-based cryptography [9]. It enables validation-based security saving to cloud clients, using dispersion of key methodology and supporting only single key correspondence. Yang *et al.* [10] decentralized confirmation method does not validate clients' cloud access, while Zhao and Wang [11] propose a decentralized access control technique that does not support verified clients, stores records as understandable documents, and does not allow record composition. All of these approaches have their own merits and drawbacks [12], and it is important to expand security aspects and empower access control strategy features in distributing information to all clients in distributed figures.

Helmy *et al.* [13] suggested a hybrid encryption framework using the Rubik's cube technique that merges chaotic encryption, advanced encryption standard (AES), and RC6, thereby yielding permutation-diffusion encrypted images. Later, orthogonal frequency-division multiplexing (OFDM) transmission showed improved encryption quality in comparison to classical methods. Zhang [14] presented a hybrid encryption approach applied to physical layer software control that mitigates the information leakage risk(s). Results concluded that utilizing the integrating Rivest-Shamir-Adleman (RAS) cryptosystem and block cipher ensured secure transmission with improved accuracy compared to traditional methods. Agarwal and Joshi [15] presented a hybrid cryptosystem for merging asymmetric RSA (ARSA) and symmetric DNA encryption (SDNAE). Further, the present method also addresses authentication, privacy, and efficiency issues, thereby providing proven sustainability for cloud-based internet of things (IoT) processing.

Novel hierarchal attribute-based cryptographic security model (NHACSM) is a suggested method for providing adjustable, flexible, and reliable access control for information sharing in conveyed climates. It allows clients to replace old documents with read-and-write jobs and re-license distributed computing tasks with updated activity, thus preventing hand-off attacks.

This article examines secure authentication for shared data. This study's main contribution: i) access control information should be shared with approved and unapproved clients to ensure secure access to cloud client boundaries; ii) the design of key management must be decentralized and integrated to ensure that no two clients can access the same information regardless of whether they agree independently; iii) suggested using blockchain for data storage and discussing smart computing data security; iv) cloud performance offers flexible and reliable information to various cloud metrics.

2. IMPLEMENTATION AND CONSTRUCTION OF NHACSM

2.1. Implementation procedure

The implementation procedure is initiated by the trustworthy user's security parameters connected to global public key (GPK), followed by key generation, encryption, trap-door generation, and finally the decryption process. The GPK plays a crucial role in the functioning of the cryptographic system. The key generation process serves as a foundation for secret key generation for both the data owner (DO) and data user (DU). The encryption process ensures the secure data format; trap-door generation provides specifically authorized searches on encrypted data and finally the decryption process. This section describes the implementation of security in NHACSM with the following calculation methods.

- a. The calculating technique is reviewed and organized by a trustworthy user, but only accepts security parameters connected to GPK.
- b. Generation of key: Let us assume as (1).

$$\left(attri\{ucpk_{id,k}|k \in \mathbb{R}\}, \{KeyVerif_k|k \in \mathbb{R}\}, AMSK_d, GPK \right) \quad (1)$$

- AA outputs secret key SK_{ud} of DU and pair corresponding to data owner's secret key (DO).
- c. Encryption of files: GPK is a cloud service provider that encrypts files and transforms them into CTfog_comm, saving all output as CT1 and CT2.
- d. GenTrap: input (ω, SK_{id}, PK_0) as trap door generator calculation technique is evaluated by DU with search word ω , secret key SK_{ud} , and key with public PK_0 then it generates output as Trapdoor (tw) and key related to re-encryption value SK_{ud}^ω .
- e. The decryption of files is done using cipher text, half-decrypted text, and key retrieval RK to produce plain text.

2.2. Proposed NHACSM schema

NHACSM is an algorithm implementation technique that uses attribute set-based encryption to analyze and handle the hierarchical structure of different users. It includes the procedure of domain authority, subordinate domain authorities, and distinct data users [16]. Figure 1 shows a hierarchical structure

representation of the user, and it has been proposed that the trustworthy user assesses privacy-associated master key-assisted parameters in the top-level scenario. Domain authority generates and distributes secure keys to subdomain authorities and next-level users, creating a key structure for each user with a decrypted key.

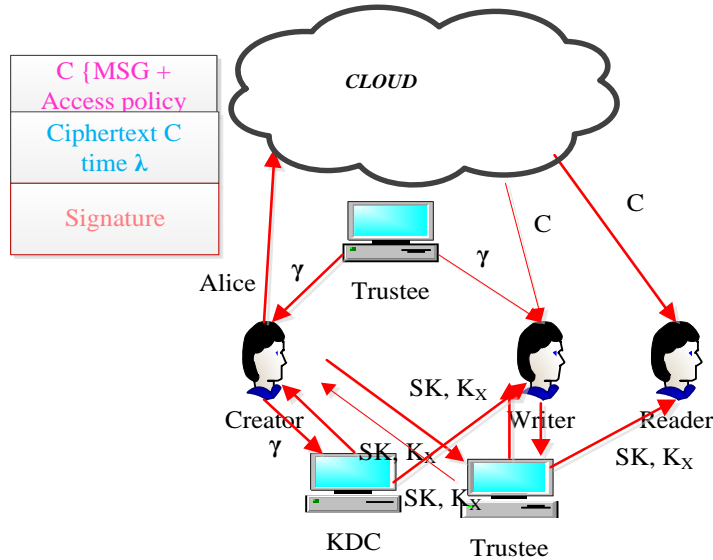


Figure 1. Hierarchical structure representation of use

We describe the basic properties of NHACSM i.e., it consists of the following methods.

- System_setup: The domain authority calculates PK and MK keys using setup calculation. In (3) bilinear map connection determines key_structure length, d . Be the prime order p with gen c and evaluate random parameters $\alpha, \beta \in Z_p, \forall i(1,2)$. Then master and public keys are generated based on key d length.

$$PK = \left(C, c, h_1 = c^{\beta_1}, f_1 = c^{\frac{1}{\beta_1}}, h_2 = c^{\beta_2}, f_2 = c^{\frac{1}{\beta_2}}, e(c, c)^\alpha \right) \tag{2}$$

$$MK = (\beta_1, \beta_2, c^\alpha) \tag{3}$$

- Domain authority creation and grant: Domain authority is organized and related to recursive attribute relations, i.e., $\{X_0, X_1, \dots, X_m\}, X_i = \{x_{i,1}, x_{i,2}, \dots, x_{i,n}\}$ with $a_{i,j}$ and associated attributes. Creates domain authority keys $r^{\{v\}}$, pick identity keys for each user $a_{i,j}, 0 \leq i \leq m, 1 \leq j \leq n_i$. Domain authority uses random user identity key selection (4).

$$DA(MK) = \left(\begin{array}{l} A, D = c^{\frac{\alpha+r^{\{u\}}}{\beta_1}}, D_{i,j} = c^{r_i^{\{u\}} \cdot H(a_{i,j})^{r_j^{\{u\}}}, \\ D_{i,j} = C^{r_{i,j}^{\{u\}}} \text{ for } (0 \leq i \leq m, 1 \leq j \leq n_i), \\ E_i = c^{\frac{r^{\{u\}}+r_i^{\{u\}}}{\beta_2}} \text{ for } (1 \leq i \leq m) \end{array} \right) \tag{4}$$

In the above domain authority master key generation, E_i is the form of translation unique rule formation $r^{\{u\}}$ relates to attribute set A_i to $r^{\{u\}}$ with associative translating components E_i and E_i may be used as E_i/E_i to translator to unique key generations, these details are employed again in calculating method. DDA++ is a cloud system that verifies every user, generates key structure, and grants to other users using authorizes derived from domain authority.

- User creation (DAMK, u, \aleph): The DA generates a master key using the key structure, then evaluates the key structure for the newly produced user \aleph , which is the combined key set structure of \aleph . Evaluates user's secret key using DA master key's unique identifier sequences.

$$(MK_{i+1}) = \begin{pmatrix} \ddot{\mathfrak{K}}, \ddot{D} = D \cdot f_i^{r\{u\}}, D_{i,j} \cdot c^{r_i\{u\}} H(a_{i,j})^{r_{i,j}\{u\}}, \\ \ddot{D}_{i,j} = D_{i,j}^1 \cdot c^{r_{ij}\{u\}} \text{ for } (a_{i,j} \in \ddot{\mathfrak{K}}), \\ \ddot{E}_i = E \cdot f_2^{r\{u\} + r_i\{u\}} \text{ for } (\mathfrak{N}_i \in \mathfrak{N}) \end{pmatrix} \quad (5)$$

The receiver key is directly taken from a trustworthy user's MK_{i+1} , the user structure's secret key.

- Encryption of file (PK, m , τ): plain message (m), Encrypt to M , the DEK file, and tree access structure. The τ encryption computation method (6),

$$Cipher_Text(CT) = \begin{pmatrix} \tau, \ddot{G} = Me(c, c)^{\alpha \cdot s} \cdot G = h_1^s, G = h_2^s, \forall b \in B: \\ G_b = c^{q^{(0)}}, G'_b = H(attr(b)^{q^{(0)}}), \\ \forall a \in A: \hat{G}_a = h_2^{q_a^{(0)}} \end{pmatrix} \quad (6)$$

where τ and A is the access tree structure and B is the parent node with sub-leaf users.

- Revocation of user: If a cloud user is revoked, they cannot access owner-shared data from any source. We overcome this problem by re-encrypting shared files read by users of revoked formats. NHACSM extends attribute set-based encryption to revoke users. Domain authority security privileges generate updated keys for revoked users if the data owner shares linked-shared files.
- File access operations: User decrypts updated data using $Dec(CT, SK_u)$ when cloud server makes encrypted request to user.
- The decryption of files: This calculating technique takes cipher text and key structure as input. The first decryption operation evaluates user key structure k regarding associative access tree structure and cipher text content accessed from data owner. If user u 's key structure τ , u and criteria are met, decrypt the entire material; otherwise, evaluate/perform decryption. Decryption is as (7).

$$Decr(CT, SK_u, i, t) = e(D_{i,j}, G'_t) / (D'_{i,j}, G'_t) = e(c, c)^{r_i\{u\}} \cdot q_t(0) \quad (7)$$

Decrypting all stored encrypted stuff with translated polynomial interpretation F_z is as (8).

$$F_z = e(\hat{G}_z, E_i/E_i) F'_z = e(c, c)^{r\{u\}} \cdot q_z(0) \quad (8)$$

The above decryption algorithm evaluates the message as $M = \hat{G} \cdot F / e(G, D)$

2.3. Novelty of the proposed work

The novelty of NHACSM relies on amicable handling techniques of hierarchical ABE in sub-processes such as individual key generation for data owners and distribution mechanism for authorization process, revocation of users, and encryption and decryption of data for cloud-based data handling. Moreover, incorporating a hierarchical attribute-based encryption approach enhances the data security, key generation based on the user's hierarchy, re-encryption of shared-files after the revocation process, and finally the decryption. The following section exemplifies the novelty of the proposed work.

- NHACSM introduces hierarchical attribute-based encryption in cloud environment for analyzing and managing the hierarchical relationships among users to lever high security. Adoption of such an encryption technique lubricates flexibility and stiffens access control on the basis of user attributes.
- The proposed work sketch approach for key generation. The key generation process encompasses separate secret key generation for data owners and data users including distributors as the authorization for data access differs depending on the hierarchy of users. Subsequently, secret keys are distributed to domain authorities, subdomain authorities, and users accordingly.
- Additionally, NHACSM provides a solution to the user revocation dispute arising in cloud. Once a user's access is revoked, re-encryption of shared files also commences. Furthermore, the domain authorities need to update by generating keys for the revoked users thus ensuring access restriction for the owner shared data.
- NHACSM model handles file encryption using access tree structures and encryption computations; whilst decryption using key structure of users and associated access tree structure.

3. EXPERIMENTAL EVALUATION

To test the proposed technique, configure a secure cloud with customers who can access several documents from diverse cloud data owners [17]. We used property-based encryption to outline NHACSM. Java and NetBeans build up the latest cloud environment using the latest CloudSim connectors [18] and [19]. Each host has 2.4 Hz, 4-8 GB RAM, and 1 TB storage for this execution. We analyze the following successions using these prerequisites:

- Setup_NHACSM: It generates public and expert keys for PK and MK presumptions [20].
- NHACSM_keyGen: It implements PK and MK key private chores with key designs. Design depth supports 1 or 2 support capacities [21].
- NHACSM_keyDeleg: This assigns DA's private key strategies for new clients based on PK and MK's space power. Space authority uses a private key [22].
- NHACSM_enc: Based on access tree strategy conditions, creates encoded document utilizing PK.
- NHACSM_Dec: Using private, it decodes the documents.
- NHACSM_rec: PK scrambles all records using private keys and re-encrypts both encoded and decoded documents. Encode document tasks decode the record using privately created [23].

Test following impacts of conducted technique on time taken by various activities and strategies to handle [24]. The customer cloud validation time for several cases with the proposed approach. To test the NHACSM technique using ASBE, KP-ABE, and CP-ABE [25]. The proposed approach produces effective security reaction time, document encryption, document unscrambling, access tree age time for various clients, and normal precision with memory usage for client tasks like transfer, demands, and download demands for proficient and secure information capacity in a distributed climate. Table 1 shows user instance processing times.

As demonstrated in Table 1 and Figure 2, classic approaches like ASBE and KP-ABE took nearly identical time to explore user instances on cloud, but when user instances rose, those approaches took longer to execute user services. The proposed approach took less time than existing approaches. Table 2 shows encryption times for different user instance requests with safe cloud upload of original material.

Table 1. Total time values for different user instances

Different users	CP-ABE	KP-ABE	ASBE	NHACSM
10	5.3	4.9	5.8	3.9
30	7.4	6.10	7.4	5.4
50	8.6	7.6	7.7	5.9
70	9.5	8.4	9.3	6.8
10	10.6	9.5	8.4	7.5

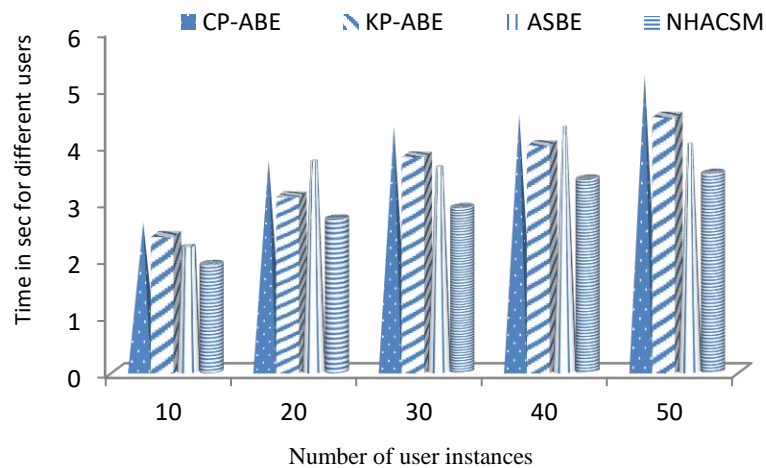


Figure 2. Performance evaluation for cloud setup environment to all the user operations

Table 2 and Figure 3 display the encryption time evaluation values and performance evaluation of different encryption methods. ASBE and CP-ABE took longer as user instances increased for different services. The proposed method encrypted user-uploaded files faster.

Table 2. Encryption time values for different user instances

Different users	CP-ABE	KP-ABE	ASBE	NHACSM
10	5.3	4.7	4.6	4.5
20	6.4	5.8	6.3	3.1
30	7.4	7.7	6.3	3.3
40	8.3	7.2	8.2	4.7
50	9.6	8.4	9.4	7.6

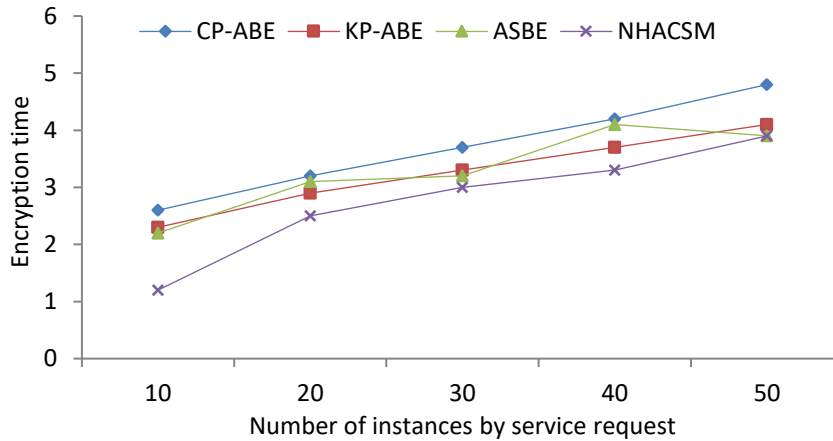


Figure 3. Performance evaluation different approaches with encryption time

In secure cloud storage, users who want to receive shared files from data owners evaluate the decryption time values in Table 3. Table 3 and Figure 4 display the decryption time evaluation values and performance evaluation of different decryption algorithms. ASBE, KP-ABE, and CP-ABE took longer as user instances increased for different services. The proposed method decrypted files uploaded by different instant service users faster. Table 4 and Figure 5 compare memory utilization methods for user instance services.

Table 3. Description time values

No. of User Instances	CP-ABE	KP-ABE	ASBE	NHACSM
10	4.8	5.8	5.3	4.7
20	5.3	6.7	5.9	3.7
30	4.8	8.5	6.4	3.3
40	7.4	6.9	5.7	3.9
50	6.8	7.5	7.9	6.4

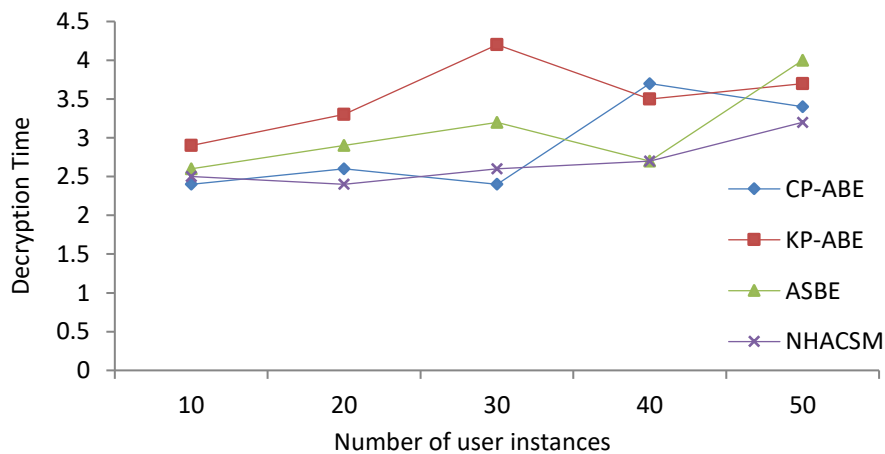


Figure 4. Performance evaluation of decryption time of different approaches

Table 4. Utilization of memory values in processing user operations in secure cloud storage

Users	CP-ABE	KP-ABE	ASBE	NHACSM
100	4,652	3,642	4,887	4,760
200	5,327	4,326	5,226	5,626
300	7,356	6,974	4,745	4,026
400	22,132	5,796	6,354	5,356
500	24,553	8,964	6,785	4,324

ASBE and KP-ABE used plenty of memory when user instance services increased. The proposed solution uses the least RAM to process user services due to its lower time complexity. Figure 5 compares access tree structure performance for different user tree constructions to store data securely.

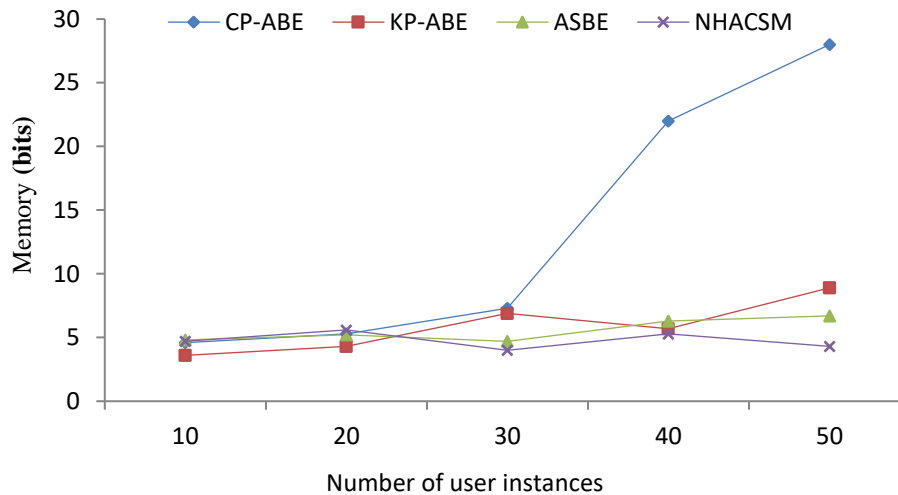


Figure 5. Performance evaluation of memory concerning different approaches

Table 5 displays the average precision of accuracy values of the suggested approach with different user instance files kept securely in the cloud. Based on the foregoing data, traditional methodologies yield less accurate results as user instances rise compared to the secure cloud storage solution. Support for Table 5 user instances browsing protected cloud data.

Table 5. Average accuracy values for different user instances

Users	CP-ABE	KP-ABE	ASBE	NHACSM
100	4.5	5.9	5.1	3.6
200	5.3	6.6	5.8	4.4
300	4.8	6.3	6.3	3.5
400	7.4	6.8	5.4	4.9
500	6.9	7.2	7.6	5.4

Figure 6 shows users' accuracy in dispersed data security procedures. Figures 2 to 6 indicate total user instances, encryption and decryption times, and memory utilization. The suggested method outperforms ASBE, KP-ABE, and CP-ABE in multi-file sharing cloud systems.

3.1. Validation of the proposed work

The validation of the proposed work is represented in terms of categorizing the users, decrypted keys to authorize the user, revocation mechanism ensuring the restriction over the shared data, and finally the enhanced encryption and decryption process in the cloud. Thus, NHACSM adopts a hierarchical attribute-based encryption model for efficient functioning such as key generation and management techniques, user revocation, and lastly efficient file encryption and decryption in cloud environment. The following section depicts the validation of the proposed work in terms of various parameters.

- a. Unlike the conventional attribute-based encryption which permits data access on the basis of associated attributes of either user or data; the hierarchical attribute-based encryption allows data access on the basis of the hierarchy of users. In the proposed model, the users are categorized under a hierarchical structure such as individual users, domain authorities, and sub-domain authorities. This categorization brings out different levels of authority and unique relationships among them. On hierarchical grounds, NHACSM limits data access by considering the attributes of users as well as the hierarchical position of users. This aids secure data access in cloud.
- b. NHACSM handles key generation and distribution efficiently within the hierarchical structure. As the roles of domain authorities, and individual users are all well-defined, the key generation process undergoes severe scrutiny in terms of security parameters by the trusted users. Here, the domain authorities create a key structure for subdomain authorities and individual users in such a way that every user has a decrypted key that authorizes or limits data access in a hierarchical structure.
- c. The user revocation dispute in cloud is addressed by NHACSM. When a user's access benefits are revoked, the model prevents access to owner-shared data by re-encrypting the shared files that were accessible earlier. The process of generating the updated keys for the revoked users by the authorities of the domain limits the access of shared files as decryption cannot be proceeded. Thus, the revocation mechanism adopted in the model increased overall security and restricted the control over shared data in cloud.
- d. NHACSM grants encryption and decryption processes in cloud. The encryption is performed utilizing access tree structures and encryption computations. On the other hand, secured decryption is carried forward through key structure of users and the associated access tree structure. The global public key of cloud service provider aids efficient encryption whilst in decryption the user key structure acts as a pivotal factor. Additionally, the decryption criteria must be necessarily satisfied by the user.

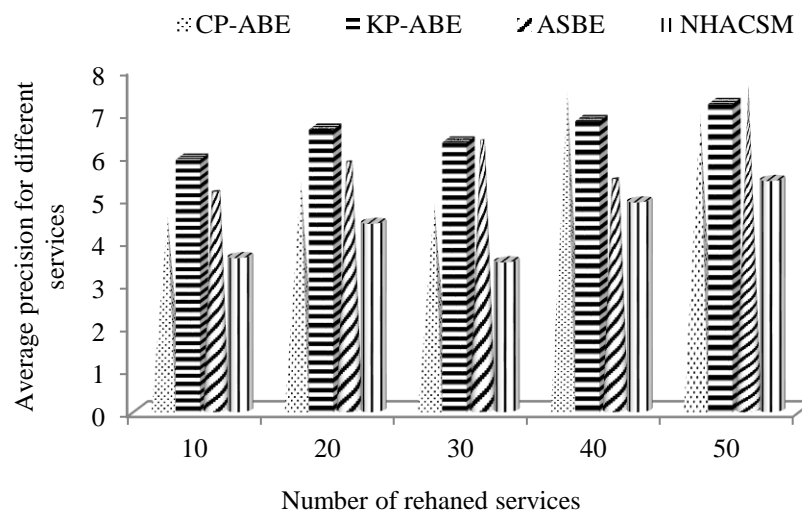


Figure 6. Performance of accuracy with different user secure operations

4. CONCLUSION




In this report, NHACSM is a secure confirmation approach to give green, adaptable, adaptable client supply access to distributed computing. It uses an impact hierarchal approach to approach individual documents, which is available in trademark set-based encryption. NHACSM does not best support client security and it accomplishes high-level concepts like disavowal purchaser in measurements sharing presuming more than one venture credit is a present. NHACSM security execution system with explicit thought-level estimation methodologies. The applied trials showed that green comfort in general execution assessment can be used to assess well-being efficiencies in distributed computing. Further expansion is needed to control the organization's keys and examine how they can help team-oriented measurements impart to cloud servers. The NHACSM method reduces total time values for different user instances compared to conventional approaches, with ASBE, KP-ABE, CP-ABE, and NHACSM being the lowest for 10, 20, 30, 40, and 50 instances. ASBE and CP-ABE were taken more time to evaluate different encryption approaches when increasing the user instance. ASBE, KP-ABE, and CP-ABE took longer to decode when user instances for different services increased. The proposed approach took less time than existing approaches. Encryption

time and performance evaluations. The proposed method encrypted user-uploaded files faster. Decryption time and performance evaluation of alternative techniques. The proposed method decrypted files uploaded by different instant service users faster. Memory use performance evaluation for user instance services. Data security accuracy of scattered users. Memory evaluation using various methods is also effective. Memory use, total user instances, and encryption and decryption times. The suggested method outperforms ASBE, KP-ABE, and CP-ABE in multi-file sharing cloud systems. To regulate the company's keys and analyze how they would support team-oriented loosen-up measures imparting to trusted cloud servers in dispersed computing.




REFERENCES

- [1] R. Ahuja and S. K. Mohanty, "A scalable attribute-based access control scheme with flexible delegation cum sharing of access privileges for cloud storage," *IEEE Transactions on Cloud Computing*, vol. 8, no. 1, pp. 32–44, Jan. 2020, doi: 10.1109/TCC.2017.2751471.
- [2] G. Viswanath and P. V. Krishna, "Hybrid encryption framework for securing big data storage in multi-cloud environment," *Evolutionary Intelligence*, vol. 14, no. 2, pp. 691–698, Jun. 2021, doi: 10.1007/s12065-020-00404-w.
- [3] S. Belguith, N. Kaaniche, and M. Hammoudeh, "Analysis of attribute-based cryptographic techniques and their application to protect cloud services," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 3, Mar. 2022, doi: 10.1002/ett.3667.
- [4] M. Tahir, M. Sardaraz, Z. Mehmood, and S. Muhammad, "CryptoGA: a cryptosystem based on genetic algorithm for cloud data security," *Cluster Computing*, vol. 24, no. 2, pp. 739–752, Jun. 2021, doi: 10.1007/s10586-020-03157-4.
- [5] P. Chinnasamy, S. Padmavathi, R. Swathy, and S. Rakesh, "Efficient data security using hybrid cryptography on cloud computing," in *Inventive Communication and Computational Technologies: Proceedings of ICICCT 2020*, 2021, pp. 537–547, doi: 10.1007/978-981-15-7345-3_46.
- [6] R. Banavathu and S. Meruva, "Efficient secure data storage based on novel blockchain model over IoT-based smart computing systems," *Measurement: Sensors*, vol. 27, Jun. 2023, doi: 10.1016/j.measen.2023.100741.
- [7] R. Adee and H. Mouratidis, "A dynamic four-step data security model for data in cloud computing based on cryptography and steganography," *Sensors*, vol. 22, no. 3, Feb. 2022, doi: 10.3390/s22031109.
- [8] J. Li, F. Sha, Y. Zhang, X. Huang, and J. Shen, "Verifiable outsourced decryption of attribute-based encryption with constant ciphertext length," *Security and Communication Networks*, pp. 1–11, 2017, doi: 10.1155/2017/3596205.
- [9] B. Yergaliyeva, Y. Seitkulov, D. Satybaldina, and R. Ospanov, "On some methods of storing data in the cloud for a given time," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 20, no. 2, pp. 366–372, 2022, doi: 10.12928/telkomnika.v20i2.21887.
- [10] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: concept and applications," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, pp. 1–19, Mar. 2019, doi: 10.1145/3298981.
- [11] Z. Zhao and J. Wang, "Verifiable outsourced ciphertext-policy attribute-based encryption for mobile cloud computing," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 6, pp. 3254–3272, Jun. 2017, doi: 10.3837/tiis.2017.06.024.
- [12] B. K. Samanthula, Y. Elmehdwi, G. Howser, and S. Madria, "A secure data sharing and query processing framework via federation of cloud computing," *Information Systems*, vol. 48, pp. 196–212, Mar. 2015, doi: 10.1016/j.is.2013.08.004.
- [13] M. Helmy, E.-S. M. El-Rabaie, I. Eldokany, and F. E. Abd El-Samie, "Proposed hybrid encryption framework for robust 3D image communication over wireless channels," *Optik*, vol. 273, Feb. 2023, doi: 10.1016/j.jpleo.2022.170205.
- [14] J. Zhang, "Application of hybrid encryption algorithm in physical layer software control," *Results in Physics*, vol. 51, Aug. 2023, doi: 10.1016/j.rinp.2023.106665.
- [15] S. Agarwal and G. Joshi, "Hybrid encryption of cloud processing with IoT devices using DNA and RSA cryptography," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 11, no. 6, pp. 21–27, 2023, doi: 10.17762/ijritcc.v11i6.6767.
- [16] A. I. Abdulsada, D. G. Honi, and S. Al-Darraj, "Efficient multi-keyword similarity search over encrypted cloud documents," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 23, no. 1, pp. 510–518, 2021, doi: 10.11591/ijeecs.v23.i1.pp510-518.
- [17] W. Wang *et al.*, "Leaky cauldron on the dark land: understanding memory side-channel hazards in SGX," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 2421–2434, doi: 10.1145/3133956.3134038.
- [18] Q. Huang, Y. Yang, and L. Wang, "Secure data access control with ciphertext update and computation outsourcing in fog computing for internet of things," *IEEE Access*, vol. 5, pp. 12941–12950, 2017, doi: 10.1109/ACCESS.2017.2727054.
- [19] A. A. Fairosebanu and A. C. N. Jebaseeli, "Data security in cloud environment using cryptographic mechanism," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 12, no. 1, pp. 462–471, 2023, doi: 10.11591/eei.v12i1.4590.
- [20] W. Sun, R. Zhang, W. Lou, and Y. T. Hou, "Rearguard: secure keyword search using trusted hardware," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, Apr. 2018, pp. 801–809, doi: 10.1109/INFOCOM.2018.8485838.
- [21] P. Sun, "Security and privacy protection in cloud computing: discussions and challenges," *Journal of Network and Computer Applications*, vol. 160, Jun. 2020, doi: 10.1016/j.jnca.2020.102642.
- [22] Y. Zheng, X. Yuan, X. Wang, J. Jiang, C. Wang, and X. Gui, "Toward encrypted cloud media center with secure deduplication," *IEEE Transactions on Multimedia*, vol. 19, no. 2, pp. 251–265, Feb. 2017, doi: 10.1109/TMM.2016.2612760.
- [23] M. Elhoseny, K. Shankar, S. K. Lakshmanaprabu, A. Maselena, and N. Arunkumar, "Hybrid optimization with cryptography encryption for medical image security in internet of things," *Neural Computing and Applications*, vol. 32, no. 15, pp. 10979–10993, Aug. 2020, doi: 10.1007/s00521-018-3801-x.
- [24] H. Ma and Z. Zhang, "A new private information encryption method in internet of things under cloud computing environment," *Wireless Communications and Mobile Computing*, pp. 1–9, Sep. 2020, doi: 10.1155/2020/8810987.
- [25] K. M. Nagaraju and R. Boraiah, "Key-cipher policy attribute-based encryption mechanism for access control of multimedia data in cloud storages," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 28, no. 1, pp. 545–550, 2022, doi: 10.11591/ijeecs.v28.i1.pp545-550.

BIOGRAPHIES OF AUTHORS

Banavathu Rajarao    received a B.Tech. degree in IT from Abdul Kalam Institute of Technological Sciences, Kothagudem, TS, India. He holds an M.Tech. degree in computer science and engineering. He is a research scholar at Jawaharlal Nehru Technological University, Anantapur, Andhra Pradesh, India. His research areas are cloud computing and IoT. He published various papers in international/national journals and also published book, book chapters. He had three copyrights. He can be contacted at b.rajarao1207@gmail.com.



Mervu Sreenivasulu    received a B.Tech. degree in computer science and engineering, from K.L. College of Engineering (K. L. University), Vaddeswam, A.P. India, in 1990. He completed his M.E. in computer science and engineering, from Jadavpur University, Calcutta, India, in 1999. He received a Ph.D. in computer science and engineering, from JNTUA, Ananthapuramu, in 2013. He has 28 years of teaching experience. He is currently working as a professor in the Department of Computer Science and Engineering Department, K.S.R.M. College of Engineering (Autonomous), Kadapa. He is a life member of ISTE and a member of IE(I). He published 19 papers in international journals and presented 20 papers at national and international conferences. His research interests include data mining, machine learning, natural language processing, cloud computing, and computer networks. He can be contacted at mesrinu@rediffmail.com.