A secured cloud-medical data sharing with A-BRSA and Salp -Ant Lion Optimisation Algorithm

Binbusayyis, Adel; Alanazi, Abed; Alsubai, Shtwai; Alasiry, Areej; Marzougui, Mehrez; Alqahtani, Abdullah; Sha, Mohemmed; Aslam, Muhammad

Published in:
CAAI Transactions on Intelligence Technology

DOI:
10.1049/cit2.12305

Publication date:
2024

Citation for published version (APA):
Binbusayyis, A., Alanazi, A., Alsubai, S., Alasiry, A., Marzougui, M., Alqahtani, A., Sha, M., & Aslam, M. (2024). A secured cloud-medical data sharing with A-BRSA and Salp -Ant Lion Optimisation Algorithm. *CAAI Transactions on Intelligence Technology*. Advance online publication. https://doi.org/10.1049/cit2.12305

Received: 25 March 2023 | Revised: 20 June 2023 | Accepted: 5 July 2023

CAAI Transactions on Intelligence Technology

DOI: 10.1049/cit2.12305

ORIGINAL RESEARCH

A secured cloud-medical data sharing with A-BRSA and Salp - Ant Lion Optimisation Algorithm

Adel Binbusayyis[1] | Abed Alanazi[2] | Shtwai Alsubai[2] | Areej Alasiry[3] | Mehrez Marzougui[3] | Abdullah Alqahtani[1] | Mohemmed Sha[1] | Muhammad Aslam[4]

[1]Department of Software Engineering, College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al Kharj, Saudi Arabia

[2]Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al Kharj, Saudi Arabia

[3]College of Computer Science, King Khalid University, Abha, Saudi Arabia

[4]Department of Computer Science, Aberystwyth University, Penglais, Aberystwyth, UK

Correspondence

Muhammad Aslam and Adel Binbusayyis.
Email: mua19@aber.ac.uk and a.binbusayyis@psau.edu.sa

Abstract

Sharing medical data among healthcare providers, researchers, and patients is crucial for efficient healthcare services. Cloud-assisted smart healthcare (s-healthcare) systems have made it easier to store EHRs effectively. However, the traditional encryption algorithms used to secure this data can be vulnerable to attacks if the encryption key is compromised, posing a security threat. A secured cloud-based medical data-sharing system is proposed using a hybrid encryption model called A-BRSA, which combines attribute-based encryption (ABE) and B-RSA encryption. The system utilises the Salp-Ant Lion Optimisation Algorithm for optimal key selection. The encrypted data is stored in the cloud and transmitted to the recipient, where it is decrypted using A-BRSA-based decryption. The study measures turnaround time, encryption time, decryption time, and restoration efficiency to evaluate the system's performance. The results demonstrate the effectiveness of the A-BRSA model in ensuring secure medical data sharing in cloud-based s-healthcare systems.

KEYWORDS

big data, cloud computing, computational intelligence, computer vision, deep learning

1 | INTRODUCTION

Information is a valuable asset, and it is especially true in the era of cloud computing, big data, and IoTs. For data security and privacy, this unprecedented era of technological convergence presents enormous challenges [1]. Nowadays, the cloud enables the provision of medical services due to the enormous amount of pervasive computing power, broadband communications, and the availability of Internet resources as a common utility [2]. One type of internet-based computing called cloud computing makes it possible to quickly provision and immediately make available shared assets and data to computers and other devices [3]. Many organisations are starting to depend on cloud services. To guarantee that the data of their users is secure and confidential, cloud service providers must abide by security and privacy policies [4]. Because the electronic medical record system may have privacy and information security issues, some patients still have skepticism about it, even though they must sign an agreement and give their consent for the doctor to look through medical records [5].

Users are now able to store and share data securely in the cloud to the widespread adoption of cloud computing. This technology can be used for several different fields, including B2C, enterprise, public, and IoT services [6, 7]. IoMTs advancement in the field of medicine is the result of the convergence of IoT and the cloud (IoMT), where medical data

Abbreviations: AA, attribute authority; ABE, attribute-based encryption; A-BRSA, Attribute-based-Blowfish-Rivest–Shamir–Adleman; AC, access control; ALO, Ant Lion Optimiser Algorithm; BH-WABE, Blowfish Hybridised Weighted Attribute Based Encryption; B-RSA, Blowfish-Rivest–Shamir–Adleman; CP, ciphertext-policy; CP-ABE, ciphertext policy attribute-based encryption; CS, cloud storage; DO, data owner; EFH-CP-ABE, extended file hierarchy CP-ABE; EHR, electronic health records; IoMT, internet of medical things (IoMT); IoT, internet of things; KP, key-policy; KPABE, Key Policy Attribute Based Encryption; RD, resource discovery; SALO, Salp-Ant Lion Optimisation Algorithm; SSA, Salp Swarm Algorithm; TA, Trace Authority.

from wearable devices can be collected and stored in a hospital cloud environment for remote monitoring by healthcare professionals [8, 9]. However, there are security risks associated with data sharing in an IoMT cloud. Service providers are not always to be trusted, and stored data may be vulnerable to privacy breaches. To protect sensitive information, access control, and user authentication technologies are needed. Cloud environments frequently use ABE, a popular security technology that safeguards privacy by encrypting data with the user's attributes. CP-ABE and KP-ABE are the two types of ABE, with CP-ABE being the preferred technique.

One issue with CP-ABE is that the absence of identification of the user who generated the secret key raises the possibility of the secret key being inadvertently disclosed without the ability to trace the source of the leak. To address this, some researchers have proposed recording the attribute authority's issued key parameter's value for the user ID or the signature. However, this would compromise one of the main benefits of ABE is the cloud user's anonymity. Another issue is that the ciphertext expands as the number of attributes grows. Larger leading to increased storage space and computation time for decryption. Additionally, a verification procedure is required to guarantee the accuracy of the decryption results delivered by an outsourcing server. The data-sharing system is shown in Figure 1.

Based on a variety of attributes and policies, ABE enables users to decide whether to decrypt data [10]. Content-based, role-based, and multi-authority access policies are the three primary types of attribute-based encryption [11]. Providing granular control over encrypted data, ABE generalises public-key encryption. Log encryption is possible with ABE [12]. The use of ABE techniques in vector-driven search engine interfaces is also common [13]. Access control and privacy can both be achieved using the public key cryptographic technique known as ABE, which enables secure data sharing between many users [14]. Data is encrypted with the help of ABE attributes, and it is then a user's secret key that is connected to an access policy is used to decrypt the data. It provides revocation, collusion resistance, scalability, and fine-grained access control, and the user can only decrypt when the user credentials are compliant with the access policy. KPABE and Cipher Policy Attribute-Based Encryption are the two main categories for ABE [15].

The major contribution of this research work is:

- ✓ To introduce a new A-BRSA hybrid encryption model by combing the ABE and B-RS.
- ✓ The B-RSA model is formulated by amalgamating the blowfish algorithm and RSA, respectively.
- ✓ To choose the best key using the newly developed SALO, which combines the SSA and ALO, respectively.

The rest of the paper is arranged as follows:

Section 2: Existing Works and Drawbacks.

This section discusses the existing works that have been developed in recent years. It provides an overview of the research and techniques that have been previously employed in the field. Additionally, it highlights the drawbacks or limitations associated with these existing works, identifying areas where improvements are needed.

Section 3: Introducing a Novel Technique.

In this section, a novel technique is introduced as a solution to overcome the drawbacks mentioned in Section 2. The new technique is explained in detail, including its methodology, algorithms, or frameworks. The goal is to present a fresh approach that addresses the limitations of previous works and offers potential advancements in the field.

Section 4: Results and Analysis.

The results obtained from applying the proposed model or technique are presented and discussed in this section. It includes an evaluation of the performance, encryption time, decryption time and other relevant metrics associated with the proposed technique. The findings are analysed and compared with the results of existing works, demonstrating the effectiveness and improvements achieved by the novel approach.

Section 5: Conclusion.

The paper concludes with a strong conclusion in this section. It summarises the key findings and contributions of the research.

2 | RELATED WORKS

In 2019, Edemacu *et al.* [16] have examined the various ABE programmes for use in collaborative eHealth. Then went over some of the difficulties using ABE plans in group eHealth as well as some potential future areas. Then conducted a comparative analysis of the surveyed schemes' efficacy, revocation potential, and security.

In 2019, Al-Dahhan *et al.* [17] have proposed the advantages, necessities, difficulties, and drawbacks of outsourcing and cloud computing environments for data sharing. The revocation issue was one of the most important problems that the current single and multiple authorities CP-ABE schemes have not successfully addressed. Several contemporary revocation methods had examined in detail. Additionally, the access points used by the studies that were surveyed have been located. In addition to the survey papers already published, this paper has critically evaluated the pertinent work from several angles.

In 2017, Lin *et al.* [18] have suggested a protocol for joint key management for CP-ABE. Without requiring the addition of any additional infrastructure, our construction achieves distributed private key generation, issuance, and storage. A key update was provided with immediate and finely-grained attribute revocation. Key exposure issues are also successfully resolved by the collaborative mechanism that had proposed. It also significantly lowers the overhead associated with client decryption. The scheme performs a little bit better, when it comes to comparison with other representative CP-ABE schemes.

In 2016, Shabir *et al.* [19] have examined the functionality and constraints of several various ABE techniques and categories. Then expanded the survey to include more effective

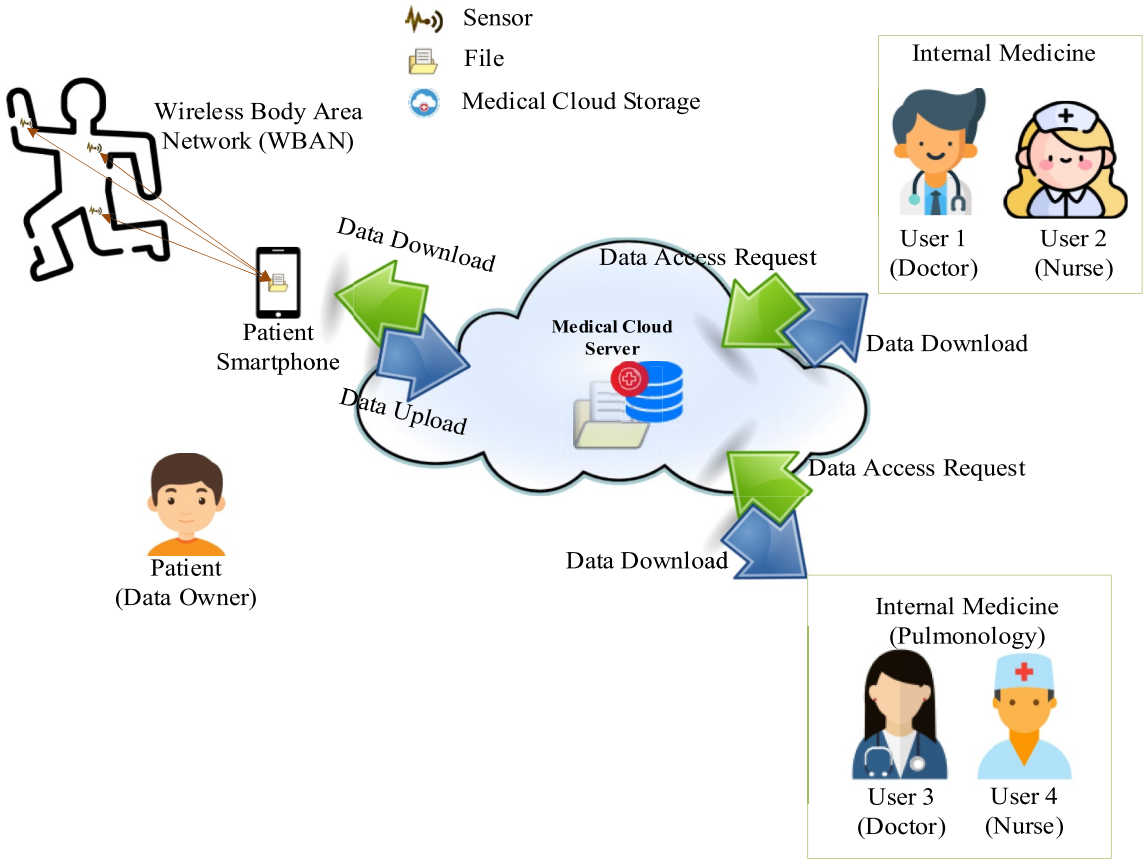


FIGURE 1 A cloud-based medical environment's data sharing system.

methods based on the granular access control, flexibility, and scalability of cloud computing, weighted attribute-based encryption with fine-grained access control.

In 2020, Yan *et al.* [20] have provided a brief overview of RD, and the EHR are highlighted along with attribute-based encryption concepts, research issues, and ABE scheme implementations. Additionally, introduced the basic ideas of ABE and looked at the key abuse, revocation, and multi-authority research issues, and discussed how the issues were applied to resource discovery and e-health, particularly relating to the personal health record environment of cloud computing.

In 2021, Pavani *et al.* [21] have examined revocable attribute encryption techniques that helped the development of the E-Health System environment with storage and privacy modules. Also presented some important works that were cantered on secure cloud storage and privacy-preserving sharing of medical records.

In 2018, Ghosh *et al.* [22] have suggested BH-WABE a reliable and effective data-sharing system for reliable access control and secure data writing. Each attribute was given weight here according to how important it was, and by using access control policies, data was encrypted. An attribute authority assigns varying attributes according to weight and revokes or updates the attributes the outsourced data is kept on the cloud service provider's servers. To reduce the computational load, access the data file corresponding to its weight, that was available to the receiver. In terms of security, dependability, and effectiveness. The suggested BH-WABE offers multi-authority security, collusion resistance, and fine-grained access control.

In 2019, Li *et al.* [23] have suggested using several files could be encrypted using the EFH-CP-ABE protocol at the same access level. Their plan is very useful, especially for large organisations or businesses with a lot of hierarchical divisions because it significantly reduces their need for storage space and computing resources on cloud servers. Additionally, this solution provides users with secure and adaptable access control to cloud storage. Additionally, a novel scheme's security was established using a conventional approach. The associated experiment for the EFH-CP-ABE scheme was then completed, and the desired experimental results were obtained.

2.1 | Problem statement

The review of the existing works is shown in Table 1. Cloud computing is a modern way of accessing Information Technology resources that provides flexible, on-demand, and user-friendly access. However, security concerns arise with the transmission of information to cloud servers. Cloud storage has been suggested to be secured with property-based encryption, but the majority of the research only considers

TABLE 1 Review of the existing works.

Author	Aim/process	Research gaps
Barik *et al.* [1]	• Systems, architectures, and applications for cloud computing supported by fog in the era of big data and the Internet of things	• No suggestion is made to look into and deal with security problems.
Botta *et al.* [2]	• Combining cloud computing and the IoT	• More computational overhead
Rashid *et al.* [3]	• Attributes and services of cloud computing	• There is no solution for the security issue in cloud computing.
Kalaiprasath *et al.* [4]	• A semantic approach to end-to-end security for cloud compliance and security	• Doesn't constitute a negative rule
Kumar *et al.* [7]	• Attribute-based encryption survey, gap analysis, and future directions in cloud computing	• Security needs to be improved
Tembhare *et al.* [11]	• Role-based policies to protect patient health records' privacy in the cloud	• No renewal of attributes
Gafif *et al.* [12]	• For fine-grained access control, a simple ciphertext policy based attribute-based encryption	• The multiauthority user/attribute-revocable scheme is not expanded.
Wang *et al.* [14]	• A framework for data sharing with fine-grained access control in decentralised storage systems based on blockchain	• The revocation of user attributes and updating of access policies are not executed.
Li *et al.* [15]	• Cloud storage with revocation and encryption of searchable attribute-based ciphertext	• The performance of the system does not improve.
Al-Dahhan *et al.* [17]	• Ciphertext-policy attribute-based encryption revocation survey	• Numerous qualities are not emphasised.
Lin *et al.* [18]	• A collaborative key management protocol for cloud data sharing using ciphertext policy attributes	• The costs of decryption and encryption are higher
Yan *et al.* [20]	• The environment of cloud computing and attribute-based encryption	• Basic ABE does not have a mechanism for user accountability.

privilege management and identity privacy, when discussing data content security and access control. To address this issue, a safe data access control system that is enforced by the user is proposed to provide users the ability to outsource sensitive data for cloud storage securely. The suggested solution guarantees safe data transfer between the client and the target server, which is inaccessible to unauthorised users. Because cloud servers must Data collection and preparation: The share sensitive corporate data, it is essential to use an encryption system that is both efficient and has fine-grained access control. An effective and secure method of sharing data in the cloud is the main goal of the paper.

3 | PROPOSED METHODOLOGY

Cloud computing technology has been a game-changer for the healthcare industry, providing efficient and cost-effective services to users. However, when it comes to data sharing and the security of medical data when outsourcing to the cloud is a primary concern. Health information about patients must be kept private and confidential and must be guarded against unauthorised access. To address this security issue, this paper proposes a cloud-based secure healthcare framework that integrates an improved encryption algorithm and an optimal key selection model. This framework aims to offer a safe and reliable platform for exchanging medical data while protecting the privacy and confidentiality of patient information. The proposed solution enhances the protection of sensitive medical data, providing users with peace of mind when using the cloud for data outsourcing. The workflow of the proposed cloud-based medical data-sharing system can be described as follows:

- ✓ EHR data D^{inp} is collected and prepared for encryption.
- ✓ Hybrid encryption: The EHR data D^{inp} is encrypted using the A-BRSA encryption model which combines ABE and B-RSA. From the generated private keys $k_{key(i)}$, the optimal key k_{opt_key} is identified using the new hybrid deep learning model. The proposed SALO is created by combining the SSA and the ALO respectively. The decrypted data is denoted as D^e
- ✓ Data storage and transmission: The encrypted EHR data D^e is transmitted from the cloud to the receiver and cloud-stored data
- ✓ Decryption: At the receiver end, the encrypted data D^e is decrypted using the A-BRSA decryption. The decryption is undergone via the selected optimal key (generated using SALO) k_{opt_key}. The decrypted data is denoted as D^d, which is similar to D^{inp}.
- ✓ Evaluation: The system is analysed for its efficiency in providing secure medical data sharing in the cloud-based s-healthcare system in terms of turnaround time, encryption and decryption speed, and restoration effectiveness.

In more detail, the attribute-based encryption in A-BRSA allows for access control with fine granularity for the encrypted EHR data, where access to the data is determined

by the attributes of the user or receiver. The Blowfish-Rivest–Shamir–Adleman encryption provides strong encryption of the data. The hybrid optimisation model is used to select the optimal key for the encryption and decryption procedures, making sure that they are effective. The receiver receives the encrypted data that has been stored in the cloud, guaranteeing that the data's privacy is preserved throughout storage and transmission. The A-BRSA decryption at the receiver end allows the receiver to access the decrypted data. The evaluation of the system demonstrates that the A-BRSA encryption model is effective in providing secure medical data sharing in the cloud-based s-healthcare system. Figure 2 shows the proposed model's architecture.

3.1 | System model

In this part, the suggested model is explained. The general data flow and the system components will be described first. The general design of the suggested paradigm is displayed in Figure 3.

3.1.1 | System architecture

The proposed secured medical data sharing system includes four major components: (a) DO (sender-patient), (b) Encryption of Data, (c) CS and transmission, and (d) User (U) (receiver-doctor).

(a) DO: Healthcare providers maintain digital patient health records called electronic health records, or EHRs. The DO is the person or organisation that has control over the EHR data. This includes making decisions on who can access the data, and setting policies for access and storage. DO plays a crucial role in maintaining the privacy and security of their health information in an EHR system. The DO may also choose to give certain individuals the authority to add additional health information to their EHR on their behalf. The DO is responsible for creating metadata and encryption keys to secure the privacy of their EHR information. They also have the power to grant or revoke access to others and to add more EHR data to the record.

(b) Encryption of Data: The proposed architecture aims to ensure the privacy and security of EHR data by using encryption and optimisation methods. The real EHR data of the DO is encrypted using the A-BRSA encryption model, which allows for fine-grained access control to the encrypted data. To ensure optimal encryption, the optimal public key k_{opt_key} for the A-BRSA model is selected using a hybrid optimisation model, which is the combination of the ALO and the SSA. The encrypted EHR data, now represented as ciphertext is stored in the cloud and communicated to the receiver through a private blockchain. The metadata for the encrypted EHRs is also stored on the blockchain for search and tamper-resistant capabilities. This architecture helps ensure the privacy and security of the DO's EHR data.

(c) Cloud Storage: A cloud storage service allows individuals to store their data and access it from anywhere with an Internet connection. This service is provided by third-party providers, which consist of storage and AC servers. The storage component holds encrypted data, while the AC server manages user access and assists with decryption by verifying the attributes of the user requesting access to the data. This reduces the computational load on the user and increases the efficiency of the decryption process.

(d) Trace Authority (TA): The trusted server TA is in charge of managing user information. Before a user is issued a key, they must first register with the TA, who then generates an anonymous ID for the user. In the event of a leaked key, the TA can assist the AA in tracing the key back to the original user who was issued it.

(e) Data Decryption-The encrypted HER data is kept in cloud storage to reduce the expense and complexity of storing large amounts of data on local servers or physical storage devices. In our proposed architecture, cloud storage plays an important role in preserving the privacy as well as the confidentiality of the EHR data. The cloud storage component of our architecture includes the following components:

- ✓ Storage infrastructure: This component includes the physical or virtual servers, storage devices, and network components that make up the cloud storage infrastructure. These components provide the necessary hardware and software resources to store and manage the encrypted EHR data.
- ✓ Security measures: The security measures component of cloud storage ensures that the encrypted EHR data is protected from unauthorised access, tampering, and other security threats. This component includes firewalls, access control mechanisms, data encryption, and other security measures that prevent unauthorised access to the data.
- ✓ Data management tools: The data management tools component of cloud storage provides the necessary tools and interfaces for managing encrypted EHR data. This component includes tools for organising, searching, and retrieving the encrypted EHR data as well as tools for tracking the access and modification history of the data.
- ✓ Data replication and backup: This component ensures that the encrypted EHR data is backed up and can be recovered in case of data loss or corruption. Data replication and backup are critical components of cloud storage, ensuring that the encrypted EHR data is always available and protected against data loss.

(f) AA: This semi-trusted server is responsible for managing the attributes of a user and generating a decryption key for A-BRSA encryption. When a user requests a key, the AA generates it based on the user's current attributes and

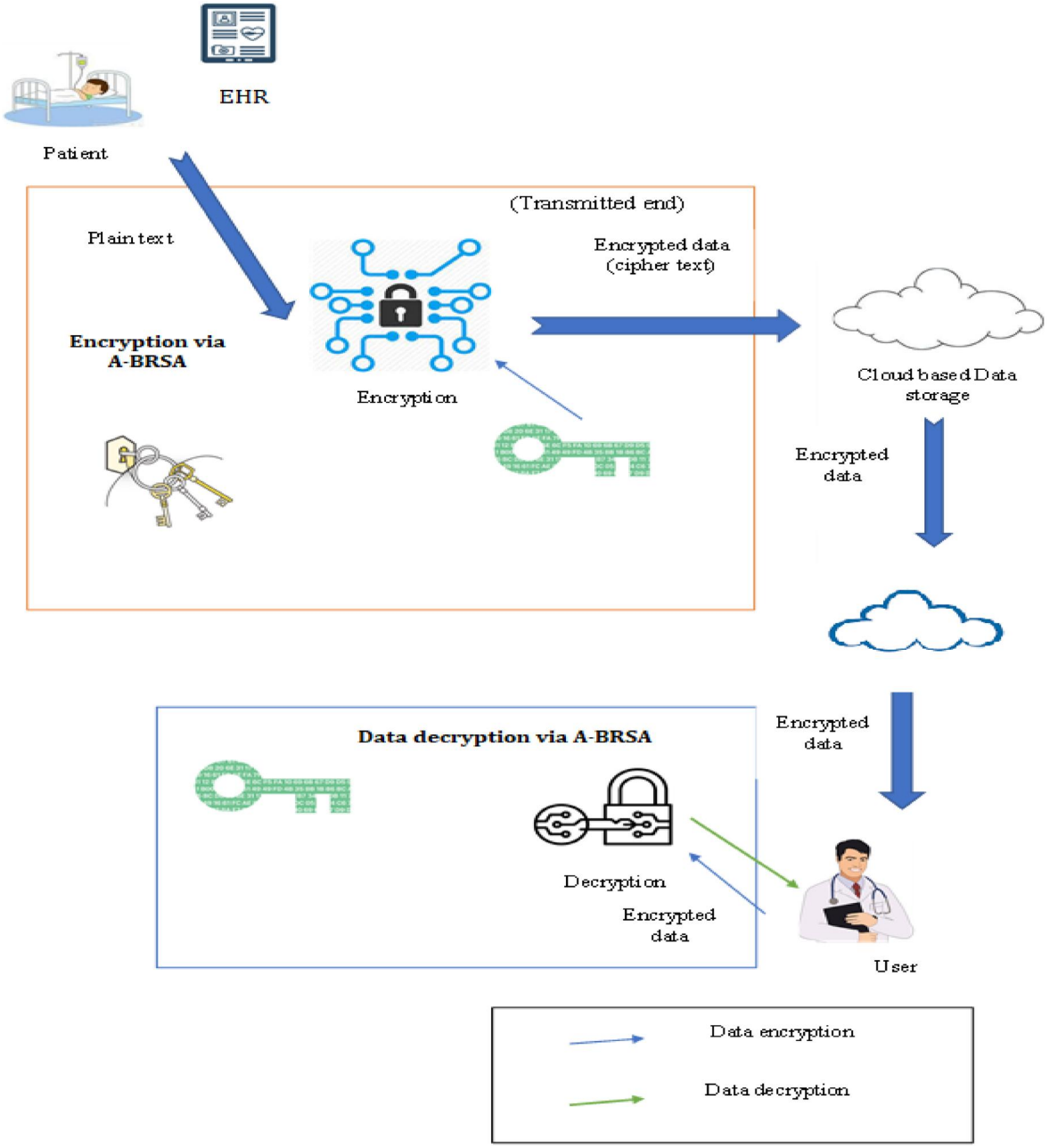
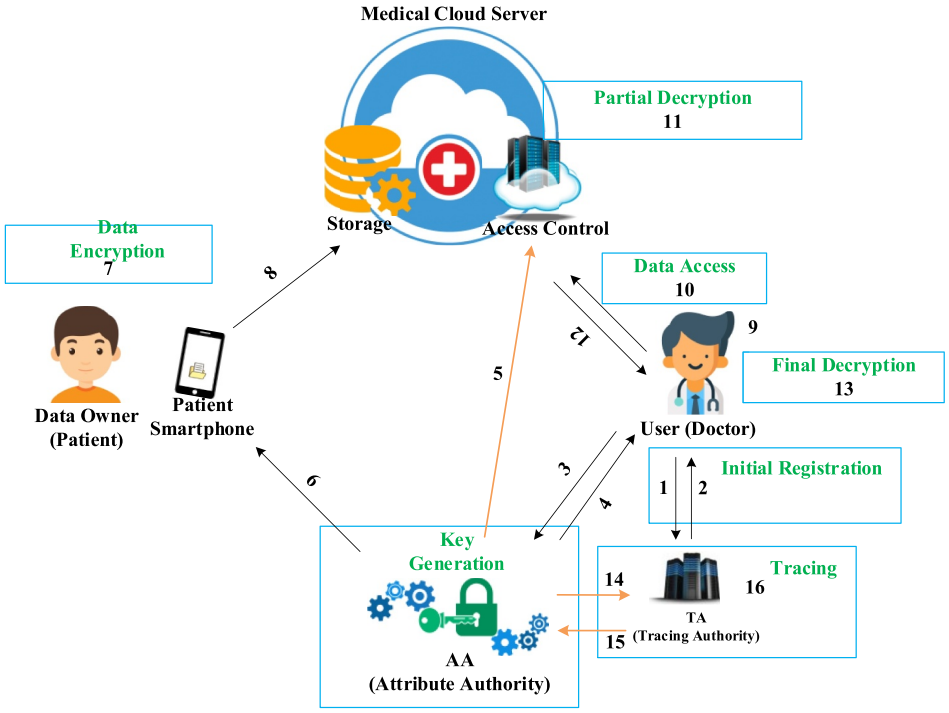


FIGURE 2 The architecture of the projected work.

sends it to the user. The key enables the user to decode the encrypted ciphertext. To ensure that only authorised individuals can access the EHR data, the decryption process is only carried out if the user uses the same optimal key k_{opt_key} that was generated using the hybrid optimisation model. This optimal key is a combination of the ALO and the SSA. The AA creates a key based on the user's attributes, which makes it impossible to identify the user directly. However, in case of a problem such as a leaked key, the AA can collaborate with the TA to determine which user was originally issued the key.

(g) User: In the proposed architecture, the user is someone who is seeking access to the data owner's (DO) electronic health records (EHRs). The user could be healthcare professionals, health insurance providers, caretakers, and others who have permission to access the EHR data. Users can use their attributes to access encryption data that is stored in the cloud. Using their attributes for partial decryption and keys provided by the AA for final decryption, the data is obtained. In the healthcare setting, this aids in maintaining the security and privacy of the EHR data.

This approach is a CP-ABE-based system for data sharing in a medical cloud environment that protects user privacy, forbids key abuse, and guarantees the integrity of the data uploaded by the data owner through a verification phase. Assuming that the patient in Figure 3 is the data owner, the



1: User registration 2: Anonymous ID issuance 3: Attribute registration and decryption key issuance request 4: Decryption key issuance 5: Sharing user information 6: Sending public parameters 7: Generating Ciphertext 8: Ciphertext Upload 9: Token generation to access the cloud 10: Request access to the cloud 11: User Authentication and partial decryption of ciphertext 12: Sending Partially decrypted ciphertext 13: After performing the final decryption, getting the message 14: Sending leaked key 15: Extracting and sending user anonymous ID from leaked key 16: User identity is verification by the user's unique ID corresponding to the anonymous ID

FIGURE 3 Overall scenario and key components of the proposed model.

patient creates an access structure with the traits of the user who has access to the data, encrypts it before uploading it, and uploads it. Only users who match the patient's specified attributes can access the medical cloud server and review patient data, including data users like doctors and nurses. This paper's scenario assumes that the doctor has given other users access to patient data (a credential or key) or shared permission to do so. However, since sharing or delegation of credential (key) authority to access is not taken into consideration, only users with access authority from AA can check the ciphertext that the data owner has uploaded. The user who was given the delegation key may in the future alter data or leak patient personal information that is stored on the server. It is possible to find out who shared or assigned the key first, and it is possible to verify the data's integrity in the outsourced system, according to the current characteristics of the thesis. In a six-phase process, data sharing is primarily carried out, and if a key has been leaked, the user who received the key initially is tracked down and identified during the tracking phase. The subsequent is the flow of each phase.

- **User registration and key issuance phase**: Before getting the key from the AA with the TA, the user registers. The TA generates an anonymous id_r sending it to the user after registering its special identifier and *id* values (as seen in Figure 3's steps 1–2).
- **Setup(p)**: By entering the security parameter p, the AA generates the master key *amk* and public key *pk* (shown in steps 3 in Figure 3).
- **KeyGen (*mk*, H, *pk*)**: In this stage, the hybrid optimisation model, which is a combination of ALO and SSA is used to create the best key possible for the A-BRSA encryption algorithm. To the AA, the user sends their ID number and attributes. A secret key SK that can decrypt the ciphertext is created by the AA using the user's values and sent to the user (shown in steps 3–4 in Figure 3).
- **ABE**: In this stage, the data is encrypted using the ABE technique, which encrypts the data using a set of attributes. The attributes may include the identity of the sender, the recipient, and the type of data being encrypted.
- **B-RSAEncryption**: The encrypted data from the ABE stage is then encrypted using the B-RSAencryption method.
- **Encrypt(pk, N, as)**: The user uses a typical symmetric key to encrypt the message. After that, the symmetric key is encrypted to produce a ciphertext cs, which is used to encrypt the message with the pk and as after the access structure (*as*) has been created. The key value for message verification is *vk*, access structure as, and the key used to

encrypt the key are all included in the ciphertext ct' (shown in steps 7–8 of Figure 3).

- **Cloud Storage**: After that, encrypted data is kept in the cloud for safe sharing and retrieval.
- **User data access and decryption phase**: When the data is required by the recipient, it is retrieved from the cloud and decrypted using the A-BRSA decryption method.
- **Partial decrypt**(*as*): Once the AC server has determined that the user's attribute set and the attribute set in the ciphertext match, a partial decryption is carried out. The user receives the result X and the ciphertext ct' from the AC server after a successful partial decryption.
- **Final decrypt** (X, pk, sk, ct'): The final decryption of the received ciphertext X, ct' is carried out by the user using their secret key SK. The user gets the key that was used to encrypt the message after successful decryption. After performing a step to ensure the integrity of the message, the user uses this key to decrypt the ciphertext.
- **Monitoring the user who initially issued the key**: By tracking the key's distribution, it is possible to identify the user who first acquired it. This aids in addressing the key abuse issue mentioned in steps 14 through 16 of Figure 3.

3.1.2 | User registration and key issuance phase

Step 1 The process of registration is initially started by the user sending a registration message that includes his unique identifier id_r and id_r value (rid_r, $id_{r,1}$) to the TA (Trust Authority).

The TA validates the value of the user's unique identifier, rid_r before computing $id_{r,2}$. The anonymous identification value for the user is displayed after that, $id_{r,1}$, is created and sent to the user.

The hash function used is h1: {0, 1}* → Z * q.

The calculation of $id_{r,2}$ is shown in Equation (1)

$$F = [att1, att2, att3...attn] \tag{1}$$

Step 2 The user sends the AA his anonymous id_r and his attribute set au^* (Authority Agent). Through the setup process, for data owners and users, the AA creates a master key and a public key. During setup, the following assumptions are made:

The universal set of attributes is represented as $F = [att1, att2, att3...attn]$. Each attribute has multiple values represented as $E_r = [e_{r,1}, e_{r,2}, e_{r,mr}]$ and can be expressed as the set as $D = [D1, D2, D3, ..., Dn]$ is an access policy, where D_r is a subset of E_r. Using the bilinear group's prime order as k, AA produces random values for the constructor α, β, g_r in A_k Both the public key as well as the master key are generated by calculating $B = v(t,t)\alpha, s = t.\beta$ in T_0, and $G_r = t.g_r$, respectively $< Setup = pk, amk >$. This is shown in Equations (2)–(4), respectively.

$$pk = \{v, t, \{G_r = t_{g_r}\} r \in [1, m], s = t.\beta, v(t,t)\alpha\} \tag{2}$$

$$amk = \{\alpha, \{g_r\} r \in [1, m]\} \tag{3}$$

$$Gpub\_AA = \alpha.K, msk = \{\alpha\} \tag{4}$$

Afterwards, the secret key sk is generated using the Keygen method using the anonymous id_r and the transmitted attribute au^* from the user. The steps for generating sk are

$$<keyGen(pk, id_r, amk, au^*) = sk>$$

- A random number w_r belongs to c, and qid_r is equal to di multiplied by K.
- q is an element of H (The symbol H stands for the set of attributes, and q represents the number of each attribute). $i1$ to iq belong to Ak (The symbol iq represents the value randomly chosen for each property).
- Hash functions: $s2: \{0,1\} * \rightarrow \{0,1\}\hat{i}m$, where im indicates the message's bit length. This is mathematically shown in Equations (5)–(7), respectively.

$$pskid_r = w_r + s_2(id_r, qid_r).\alpha \tag{5}$$

$$utr = id_r \oplus s2(id_r, qid_r) \tag{6}$$

$$i = \sum q = 1mim \tag{7}$$

Then, calculate $W' = t\alpha - i$ It generates the secret key as per Equation (8).

$$sk = H, pskid_r, utr, W' = t\alpha - t, \{Wr, 1 = t_{ir}\} r \in [1, m]\} \tag{8}$$

The user receives the AC server and the secret key SK receives the user (id_r, qid_r) (as depicted in Figures 4 and 5).

3.1.3 | Data encryption phase

The data owner uploads the encrypted data to the cloud storage at this point. To create a ciphertext, the data owner creates access structures based on the user's attributes and selects the multi-values of those attributes ct. Ciphertext ct' consists of ct, which encrypts the key for decrypting the message, cs, which encrypts both the verification value for the key ct and the message. The size of the ciphertext ct' is decreased as a result of computing values for each attribute, better utilising the unused cloud storage space.

Encryption $(pk, N, as) = ct'$. Here, N belongs to tg and the access policy D is comprised of [D1, D2, D3, Dm]. Generation

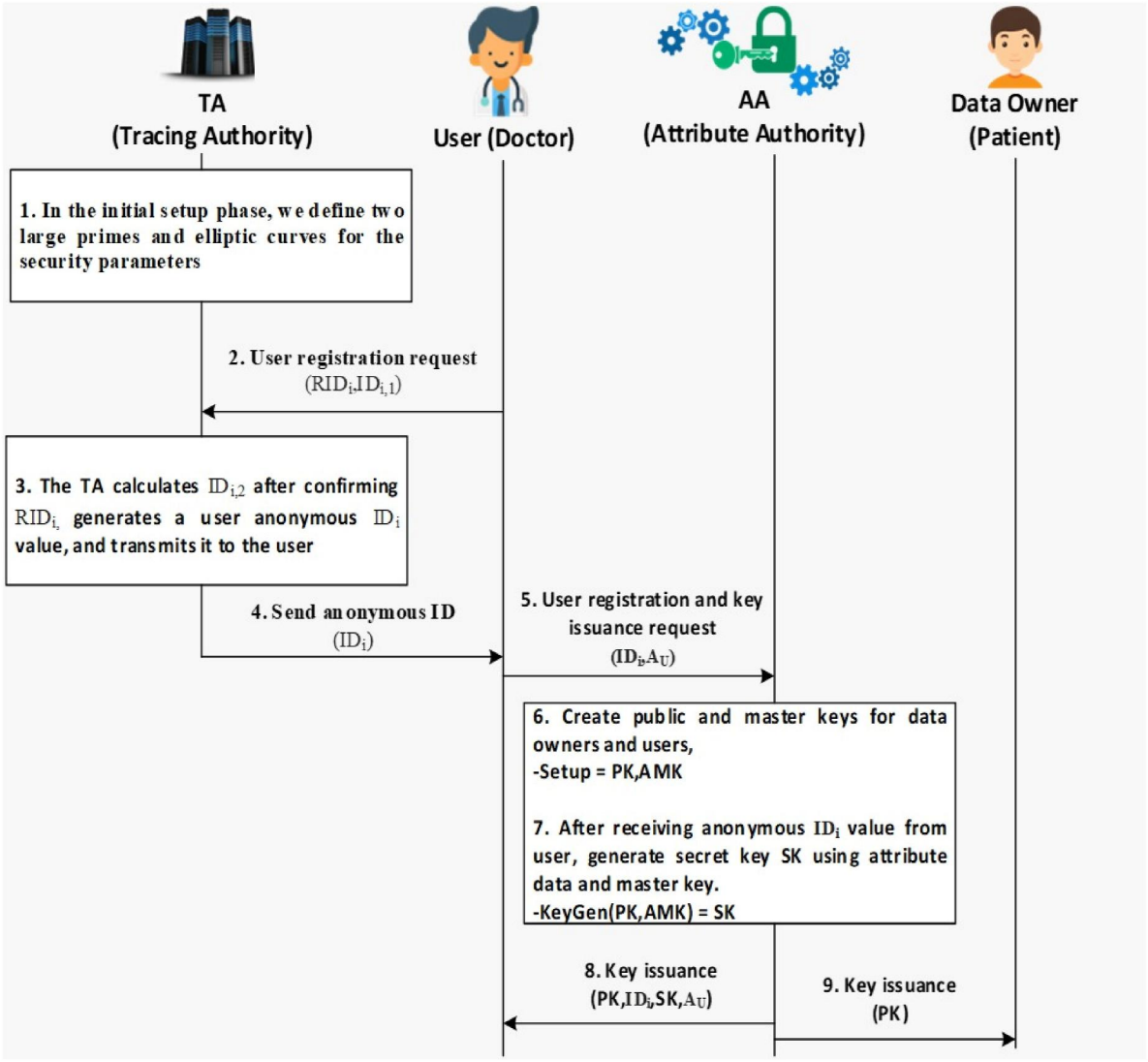


FIGURE 4 Phase of key issuance and user registration.

of a random value s from Zp (as per Equations (9)–(17), respectively.

$$h = \sum q = 1mHq \tag{9}$$

Then, the calculation is performed as per Equation (10).

$$\widehat{X} : Bh = key.v(t,t)\alpha h, \widetilde{X} : th, \overline{X} : sh \tag{10}$$

$$if\ er, 1 \in Dr, computes, X_r = tgmh \tag{11}$$

$$f\ er, 1 \in Dr, computes, X_{rtgm} + 1h \tag{12}$$

$$ct = <\widetilde{X}, \overline{X}, \widehat{X}, X'> \tag{13}$$

$$X' = \left(s. \prod r \in ASCr\right)h = \left(s. \prod r \in ashg_r\right)h, \text{tg} = \prod r = 1mtg_r \tag{14}$$

$$cs = EncKEY(N), vk = (ts(key), ts(N)) \tag{15}$$

$$ct = <\widetilde{X}, \overline{X}, \widehat{X}, X'> \tag{16}$$

$$ct' = <ct, cs, vk> \tag{17}$$

The ciphertext CT that was provided by the user is safely kept in the cloud storage.

3.1.4 | Data decryption and user data access phase

The user connects to the cloud at this point to retrieve the ciphertext and carry out the decryption. Two steps make up the decryption process: partial decryption by the user's final decryption, and the AC server.

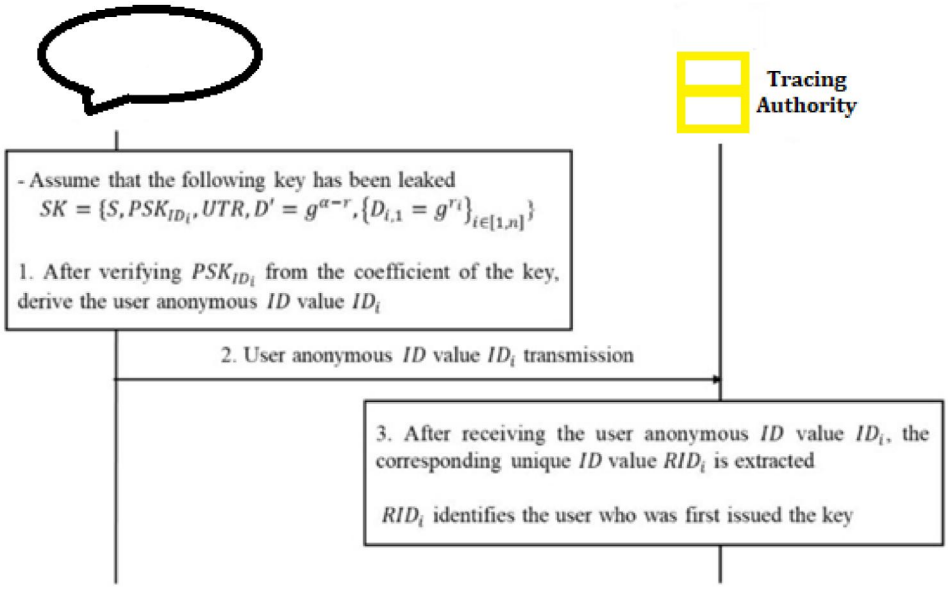


FIGURE 5 Tracing phase.

Step 1 The user generates a tk, using $pskid_r$ access to the cloud is made possible by the secret key k_{opt_key} that was given to AA. When a user makes a cloud access request via $id_r \| tk \| G$ the AC server uses the $pskid_r$ values provided by the user held by AA to verify (id_r, qid_r) of the TK value and determine the user's legitimacy. This is mathematically shown in Equation (18).

$$Token(tk) = pskid_r \| au^* \tag{18}$$

The AC server, knowing the value of (id_r, qid_r) for the user, performs verification by conducting the following verification process: This is mathematically shown in Equation (19).

$$pskid_r.K = qid_r + S_2.(id_r, qid_r).G_{pub\_AA} \tag{19}$$

Upon verification of the user's legitimacy, the desired ciphertext is requested from values specified in the ciphertext for the storage, the access structure, and the user attributes are compared. This leads to the performance of partial decryption. This is mathematically shown in Equation (20).

$$< partial\ decrypt(ct, au^*, c) = X > \tag{20}$$

Using ciphertext and user attributes, the partial decryption process is described as per Equation (21).

$$X = v(t_r, X')v\left(\check{X}, \left(\prod q \in Ht_{gr}\right)_h\right) = v(t,t)_{ih} \tag{21}$$

Step 2 The user receives the result X from the AC server along with the partially decrypted ciphertext ct'.

Step 3 The user utilises sk, X, pk to perform ct' obtained from the AC server is fully decrypted, producing a key that can decrypt the message. The ciphertext ct is then decrypted, following verification that the original message can indeed be decrypted.

$$< Finaldecrypt(pk, sk, X, ct') = N >$$

The user carries a key that can access the message N for the final decryption. This is mathematically shown in Equation (22).

$$key' = Xv(X, W').X = key.v(t,t)\alpha_h .v(t_h, t_\alpha - i).v(t,t)ih \tag{22}$$

$$N' = Deckey'(cs) \tag{23}$$

Next, a new verification key (vk') is created using N' and key'. Then, by comparing it to the current vk. The verification process is carried out to ensure the message's integrity. This is mathematically shown in Equation (24).

$$vk = vk' = ts(key), ts(N)) = ts(key'), ts(N') \tag{24}$$

3.1.5 | Tracing phase

This step aims to secure the cloud against unauthorised access by third parties. While the agency investigating cannot determine the method of distributing the secret key k_{opt_key}, they can determine the individual who received the key first and distribute it through tracking. Given that $sk = H$, $pskid_r$, utr, $W' = t\alpha - i$, $\{W_{r,1} = gri\} r \in [1, m]\}$ has already been distributed, and the first step (as depicted in Figure 6) will now be taken.

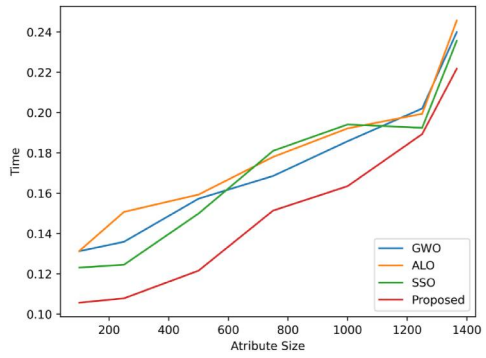


FIGURE 6 Analysis of attribute size for various existing techniques.

Step 1 Finding the User Who Originally Issued the Key. The Authentication Authority (AA) verifies $pskid_r$ in the key distribution parameters and retrieves the anonymous ID of the user value, id_r.This ID value is then sent to the Tracing Authority (TA).

$$pskid_r.k = Jid_r + s2(id_r.Jid_r).G_{pub\_AA} \text{ (Verification complete)}$$

$$id_r = ut_r \oplus pskid_r \tag{25}$$

Step 2 The Tracing Authority (TA) first receives the user's anonymous *id* value, id_r, and then extracts the user's unique *id* value rid_r. The TA then verifies the user's identity who originally issued the key for rid_r. This is mathematically shown in Equations (26) and (27), respectively.

$$id_r = \{id_{r,1}, id_{r,2}, G_r\} \tag{26}$$

$$rid_r = id_{r,2} \oplus s_1.(\beta.id_{r,1}, G_r, G_{pubTA}) \tag{27}$$

3.2 | A-BRSA

3.2.1 | Attribute-based encryption

k attribute authorities and one central authority make up a Multi Authority ABE system. Also given a value is each attribute authority, which is dk. The system employs the subsequent algorithms:

Setup: A randomised algorithm that needs to be handled by a reliable entity takes a security input parameter. Along with a public key and secret key pair for each of the attribute authorities, the output also includes a system public key and master secret key that will be used by the central authority.

Attribute Key generation: In this stage, the data is encrypted using the ABE method, which uses a set of attributes used to encrypt data. The attributes may include the identity of the sender, the recipient, and the type of data being encrypted. An attribute authority will use a random algorithm to generate attribute keys the GID of the user, the authority's secret key, a set of attributes in the domain Z_X^p, and the value of the authority w_p are all inputs. (We'll assume that before this algorithm is applied, these attributes have been verified by the user's claim.) give the user a secret key.

B-RSAEncryption: The encrypted data from the ABE stage is then encrypted using the B-RSAencryption method. As per the proposed.

Decryption: A user-run deterministic algorithm. Ciphertext that was encrypted using an attribute set Z_xX and attribute set decryption keys.

z_f are both accepted as inputs. Outputs a message for all authorities p if $\left|Z_X^p \cap Z_f^p\right| > w_p$. This decryption is accomplished by using the A-BRSA decryption method.

3.2.2 | B-RSA encryption model

The proposed B-RSA Encryption model is the combination of the Blowfish algorithm and Rivest Shamir Adleman Algorithm. The steps followed in the proposed B-RSAEncryption model is manifested below:

Key Generation: Generate a Blowfish key using a secure key generation technique. Generate the key size both encryption and decryption utilise the same keys, requiring a total of 18 sub-keys. These 18 sub-keys are kept in an S-array with a 32-bit input for each element. It is set up with the $H_r[c]$ digits. Each sub-key is altered with the input keys. Each sub-key is altered with the input keys.

Data encryption using Blowfish: Encrypt the data using the generated Blowfish key. Both the encryption and decryption processes require the use of four substitution boxes, designated as $\{H^y[1], H^y[2], H^y[3], H^y[4]\}$, which contains 32-bit $\{H^y[r][0], H^y[r][1], .., H^y[r][255]\}$ entities, totalling 255. The input contains c in a 64-bit data element and blowfish has 16 rounds. $c0$ and cI divide c into 64 equal halves. Next, for $r = 1$ to 16, Change cI and cO. Swap cO and cI once more after the sixteenth round to undo the previous swap. Once this is done, cI = cI XOR H_{17} and cO = cO XOR H_{18}. Lastly, combine cO and cI once more to produce the cipher text x. After encryption, we get the Cipher text.

Key encryption using RSA: Encrypt the Blowfish cipher text x using RSA encryption. In the blowfish as well as the RSA model, the encryption, as well as decryption, is undergone using the new Salp-Ant Lion Optimisation Algorithm (SALO). The sender computes the ciphertext x after padding the message N (unpadded plaintext) into an integer n (the padded plaintext). This is mathematically shown in Equations (28).

$$x \equiv n^v (mod\ m) \tag{28}$$

Transmission of encrypted data and key: Send the encrypted data and RSA-encrypted Blowfish key to the recipient.

Key decryption using RSA: The recipient decrypts the Blowfish key using their private RSA key. This is mathematically shown in Equation (29).

$$\begin{array}{l}\text{For any } n\epsilon\mathbb{Z}_m \\ n^{v\times w} \equiv n\ (mod\ m)\end{array} \tag{29}$$

Where $m = k \times j$ and $w^{-1} \equiv v(mod\ \Phi(m))$. Proof. Euler's Theorem already establishes this fact in the case where $gcd(n; m) = 1$, but it also holds when $gcd(n; m)\ 6 = 1$ and m is the product of two different prime numbers.

Data decryption using Blowfish: The recipient uses the decrypted Blowfish key to decrypt the data.

This hybridisation combines the fast encryption and decryption of Blowfish with the security of RSA encryption, providing a secure method for transmitting sensitive data.

3.2.3 | SALO

The best key for both encrypting and decrypting is generated using the new SALO. The ALO is a metaheuristic optimisation algorithm that draws inspiration from the predatory behaviour of ant lions. A population-based optimisation algorithm called SSA makes use of the behaviour of swarms in nature to find the best answers to challenging issues. The algorithm simulates how a group of animals, like birds or fish, would move in unison to accomplish a common objective, like locating food or avoiding predators. As per the proposed approach, the slap swarm optimisation model is introduced within the ALO model. The proposed model encloses the following phases:

Step 1 **Initialisation**: Initially, the m-count of search agents (salps as well as ant lions) are initialised. The input to the proposed model is the generated public keys. The ant positions are the generated keys as per the proposed model.

Step 2 **Fitness evaluation**: The proposed model's objective performance is to minimise the key length.

Step 3 **Random walk-based Ant Lion Movement**: Since ants in nature move stochastically in search of food, the following random walk is chosen to model ants' movement: This is mathematically shown in Equation (30).

$$\begin{array}{l}C(g) = [\,0, cumsum(2i(g_1) - 1), cumsum \\ \qquad (2i(g_2) - 1), \ldots\ldots cumsum(2i(g_m) - 1)\,]\end{array} \tag{30}$$

where $i(g)$ is an equation for a stochastic function, g stands for the random walk's step, and m is the maximum number of iterations: For modelling the movement of ants in their natural foraging behaviour, a random walk is selected as per Equation (31).

$$i(g) = \begin{cases} 1\ if\ rand\ > 0.5 \\ 0\ if\ rand \leq 0.5 \end{cases} \tag{31}$$

where g denotes the random walk step and rand denotes a [0, 1]-dimensional random number with uniform distribution. The ant's location is noted in the following matrix and is used for optimisation:

$$N_{ant} = \begin{bmatrix} Z_{1,1} & Z_{1,2} & \ldots & \ldots & Z_{1,w} \\ Z_{2,1} & Z_{2,2} & \ldots & \ldots & Z_{2,w} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ Z_{m,1} & Z_{m,2} & \ldots & \ldots & Z_{m,w} \end{bmatrix} \tag{32}$$

where N_{ant} is the matrix used to store each ant's position, $Z_{r,\,q}$ displays the value of the rth ant's qth variable (dimension), Both $mandw$ represent the total number of ants and variables, respectively. Each ant is evaluated using a fitness (objective) function during optimisation, and its fitness value is recorded in the following matrix:

$$N_{OA} = \begin{bmatrix} u\left(\left[X_{1,1}, X_{1,2}, \ldots\ldots X_{1,d}\right]\right) \\ u\left(\left[X_{2,1}, X_{2,2}, \ldots\ldots X_{2,d}\right]\right) \\ \vdots \\ \vdots \\ u\left(\left[X_{m,1}, X_{1,2}, \ldots\ldots X_{m,d}\right]\right) \end{bmatrix} \tag{33}$$

where m is the total number of ants, Xr, q displays the value of the qth dimension of the rth ant, MOA is the matrix for saving each ant's fitness, and u is the objective function. The following matrices are used to save their positions and fitness values:

$$N_{antlion} = \begin{bmatrix} ZO_{1,1} & ZO_{1,2} & \ldots & \ldots & ZO_{1,w} \\ ZO_{2,1} & ZO_{2,2} & \ldots & \ldots & ZO_{2,w} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ ZO_{m,1} & ZO_{m,2} & \ldots & \ldots & ZO_{m,w} \end{bmatrix} \tag{34}$$

where $N_{antlion}$ is the matrix used to store each antlion's position, $ZOr,\ q$ displays the value of the rth antlion in the qth dimension, w is the total number of variables, and m is the total number of antlions.

$$N_{OAL} = \begin{bmatrix} u\left(\left[XO_{1,1}, XO_{1,2}, \ldots\ldots XO_{1,d}\right]\right) \\ u\left(\left[XO_{2,1}, XO_{2,2}, \ldots\ldots XO_{2,d}\right]\right) \\ \vdots \\ \vdots \\ u\left(\left[XO_{m,1}, XO_{1,2}, \ldots\ldots XO_{m,d}\right]\right) \end{bmatrix} \tag{35}$$

The total number of antlions would be m, $N_{\,OAL}$ is the matrix used to save each antlion's fitness, XOr, q displays the value of the rth antlion in the qth dimension, and u is the objective function?

The random walks are normalised using the following equation to keep them inside the search space (min-max normalisation). This is mathematically shown in Equation (36).

$$C_r^g = \frac{(C_r^g - z_r) \times (w_r - x_r^g)}{(w_r^g - z_r)} + x_r \tag{36}$$

where z_r is the minimum value of the random walk in the rth variable and y_r is the maximum value of the random walk in the rth variable, and x_r^g is the minimum value of the rth variable at the gth iteration and w_r^g the maximum value of the rth variable at the gth iteration.

Step 4 Trapping in antlion's pits

Antlion traps affect the ants' random movements. To represent this supposition mathematically, the following equations are suggested:

$$x_r^g = Antlion_q^g + x^g \tag{37}$$

$$w_r^g = Antlion_q^g + w^g \tag{38}$$

where x^g represents a variable's minimum value at iteration g, w^g represents all variable's maximum value at that iteration g, x_r^g represents the ant's minimum value for all the variables, w_r^g represents the highest value for all of the variables for an ant and $Antlion_q^g$ represents the position of the chosen antlion q at iteration g.

Step 5 Building trap

When choosing ants based on fitness the ALO algorithm must employ a roulette wheel operator during optimisation. This mechanism increases the likelihood of catching ants for the fitter antlions.

3.2.4 | Sliding ants towards antlion

Ants must move randomly, and depending on their fitness, antlions can construct traps. However, Antlions shoot sand out of the pit's centre when they see an ant in the trap. Trying to escape, the trapped ant slides down due to this behaviour. To mathematically model this behaviour results in an adaptive reduction in the hyper-sphere radius of the ants' random walks. In this regard, the following equations are suggested:

$$x^g = \frac{x^g}{R} \tag{39}$$

$$w^g = \frac{w^g}{R} \tag{40}$$

R is a ratio, x^g is the variable with the lowest value at iteration g, and w^g denotes the vector with the highest value at iteration g.

- **Catching prey and re-building the pit based on Salp Swarm Algorithm (proposed)**

When an ant is snatched by the antlion's jaws after falling into the pit, the hunt has reached its conclusion. Following this phase, the antlion drags the ant into the sand, where it eats the insect's body. It is thought that prey is caught when an ant grows fitter than its corresponding antlion and enters the sand because ants imitate this process. Then, an antlion must change its position to coincide with the hunted ant's most recent location to raise its chances of capturing fresh prey. In this phase, the SSA-based leader update is implied.

In SSA, the leader's position is modified, wherein the Ant_r^g the parameter is newly introduced. This phase is determined by Equation (41).

$$Antlion_p^g = \begin{cases} U_q + Ant_r^g\left((ub_q - lb_q)x_2 + lb_q\right) & u\left(Ant_r^g\right) > u\left(Antlion_p^g\right) \\ U_q - Ant_r^g\left((ub_q - lb_q)x_2 + lb_q\right) & u\left(Ant_r^g\right) \le u\left(Antlion_p^g\right) \end{cases} \tag{41}$$

where ub_q displays the qth dimension's upper limit. and lb_q represents the qth dimension's inferior limit, c_q^1 shows the location of the leader, U_q reveals the food source's position vector in the qth dimension, x_3 The key parameter of the algorithm is x_1, and then it contains random values inside of the range [0, 1] as a function in Eqn. (49). where g indicates the iteration being displayed, Antliont q displays the position of the chosen qth antlion at the current iteration, and Ant_r^g displays the position of the rth ant at the current iteration.

Step 6 **Elitism**: Evolutionary algorithms possess elitism, which enables them to retain the best solution(s) found at any stage of the optimisation process. Therefore, it is assumed that each circle has both the roulette wheel and an antlion chosen by the elite at the same time. This is mathematically shown in Equation (42).

$$Ant_r^g = \frac{I_Z^g + I_V^g}{2} \tag{42}$$

Where Ant_r^g denotes the position of the rth ant at the gth iteration, I_V^g is the random walk around the elite at the gth iteration, I_Z^g is the random walk around the antlion selected by the roulette wheel at the gth iteration.

With the help of the suggested operators, it is now possible to define the ALO optimisation algorithm from the previous subsections. A three-tuple function, the ALO algorithm is by definition close to the global optimum for optimisation problems. This is demonstrated mathematically in Equation (42).

$$ALO(Z, Y, X) \tag{43}$$

where X returns the value true when the end criterion is satisfied and Y modifies the initial population provided by function Z. The function Z is responsible for producing the

initial, random solutions. These are the definitions for the functions Z, Y, and X.

$$\varnothing \xrightarrow{X} \{N_{Ant}, N_{OA}, N_{Antlion}, N_{OAL}\} \tag{44}$$

$$\left\{ N_{Ant}, N_{Antlion} \right\} \rightarrow \left\{ N_{Ant}, N_{Antlion} \right\} \tag{45}$$

$$\{ N_{Ant}, N_{Antlion} \} \rightarrow \{ true, false \} \tag{46}$$

where N_{OA} contains the ants corresponding fitness, N_{OAL} has the antlions fitness, and N_{Ant} is a matrix chosen to represent the ant's position. Algorithm 1 shows the pseudocode of SALO model.

Algorithm 1 SALO model

```
Start
Initialise ant population N_Ant
Initialise antlion population N_Antlion
Initialise fitness matrices N_OA and N_OAL
Set termination condition
While termination condition is not met do:
  Evaluate fitness of ants in N_Ant
  Perform random walk-based antlion
movement:
    For each ant in N_Ant do:
        Generate a random walk path using
Equation (31)
        Update ant's position using
Equations (37) and (38)
        Normalise random walk using
Equation (36)
  Evaluate fitness of antlions in N_Antlion
  Build traps and catch prey:
    For each ant in N_Ant do:
        Select antlions based on fitness
        Slide ants towards antlions using
Equations (39) and (40)
    End for
  Update antlions' positions:
    For each antlion in N_Antlion do:
        Select an ant to update the antlion's
position (Equation (41))
        End for
    End for
  Apply elitism:
    For each antlion in N_Antlion do:
        Select a random walk around the elite
and the antlion selected by roulette wheel
        End for
    End for
  return the best solution found in
N_Antlion
End While
Return the best solution found in N_Antlion
Terminate
```

4 | RESULT AND DISCUSSION

The suggested model has been utilised in MATLAB. The proposed model has been validated using three different databases, such as Database-1 [24], Database-2 [25] and Database-3 [26]. Among the collected data, 40% of the remaining data was used for testing, and the remaining 60% was used for training. The graphical analysis and statically analysis are compared with various existing techniques such as Grey Wolf Optimisation (GWO), Ant Lion Optimisation (ALO), SSA, collaborative key management protocol [18] (pap1), Lightweight ciphertext-policy attribute-based encryption [12] (pap2) and Blowfish hybridised weighted attribute-based encryption [22] (pap3). The evaluation has been made in terms of the turnaround, encryption, and decryption times, restoration effectiveness, and KPA and CPA analysis.

4.1 | Analysis of attribute size

The analysis of attribute size using various existing techniques is shown in Figure 6. The comparison is made based on the time required for encryption using different algorithms, including Grey Wolf Optimisation (GWO), Ant Lion Optimisation (ALO), Social Spider Optimisation (SSO), and the proposed A-BRSA model. When the attribute size is 200, the encryption time for GWO, ALO, SSO, and the proposed A-BRSA model is measured as 0.131, 0.129, 0.122, and 0.105, respectively. This demonstrates that the proposed A-BRSA model achieves a significantly lower encryption time compared to the existing techniques. Similarly, as the attribute size increases to 400, 600, 800, 1000, 1200, and 1400, the encryption times for GWO, ALO, SSO, and the proposed A-BRSA model are compared. In all cases, the proposed A-BRSA model consistently outperforms the existing techniques, showcasing its superiority in terms of efficiency. The attained results can be attributed to several factors. Firstly, the hybrid encryption model of A-BRSA combines Attribute-Based Encryption (ABE) and B-RSA, leveraging the strengths of both techniques to enhance security and efficiency in medical data sharing. This combination allows for more robust encryption and decryption processes. Furthermore, the selection of the optimal key using the Salp-Ant Lion Optimisation Algorithm (SALO) plays a crucial role in the improved performance of the proposed approach. SALO combines the advantages of the Salp Swarm Algorithm (SSA) and the Ant Lion Optimisation (ALO) algorithm, resulting in better optimisation and key selection. Overall, the results clearly demonstrate that the proposed A-BRSA model, utilising SALO for key selection, offers significant improvements in terms of turnaround time, encryption time, decryption time, and restoration efficiency. These findings support the effectiveness and suitability of the A-BRSA model for secure medical data sharing in cloud-based s-healthcare systems.

4.2 | Analysis the of decryption time

The investigation of decryption time using various techniques is displayed in Figure 7. The comparison is made based on the time required for decryption using different algorithms, including Grey Wolf Optimisation (GWO), Ant Lion Optimisation (ALO), Social Spider Optimisation (SSO), Collaborative Key Management Protocol [18], Lightweight Ciphertext-Policy Attribute-Based Encryption [12], Blowfish Hybridised Weighted Attribute-Based Encryption [22], and the proposed A-BRSA model. In database1, the decryption time (s) for GWO, ALO, SSO, Collaborative Key Management Protocol, Lightweight Ciphertext-Policy Attribute-Based Encryption, Blowfish Hybridised Weighted Attribute-Based Encryption, and the proposed A-BRSA model are measured as 0.23, 0.24, 0.21, 0.22, 0.202, 0.249, and 0.19, respectively. It is observed that the proposed A-BRSA model achieves a significantly lower decryption time compared to the existing techniques. Similarly, in database 2, the decryption time (s) for GWO, ALO, SSO, Collaborative Key Management Protocol, Lightweight Ciphertext-Policy Attribute-Based Encryption, Weighted Attribute-Based Encryption using Blowfish Hybridisation are measured as 0.23, 0.21, 0.24, 0.20, 0.21, 0.22, and 0.19. Once again, the proposed A-BRSA model demonstrates superior performance with a lower decryption time compared to the comparative techniques. In database 3, the decryption time (s) for GWO, ALO, SSO, Collaborative Key Management Protocol, Lightweight Ciphertext-Policy Attribute-Based Encryption, Blowfish Hybridised Weighted Attribute-Based Encryption, and the proposed A-BRSA model are measured as 0.21, 0.23, 0.248, 0.24, 0.212, 0.214, and 0.19. The results consistently indicate that the proposed A-BRSA model achieves lower decryption time, showcasing its effectiveness in enhancing efficiency in the decryption process. The attained results can be attributed to several factors. Firstly, the hybrid encryption model of A-BRSA combines Attribute-Based Encryption (ABE) and B-RSA, which allows for a more efficient and effective decryption process. The combination of these two techniques ensures a robust and secure decryption of the encrypted medical data. Furthermore, the optimisation and key selection process using the Salp-Ant Lion Optimisation Algorithm (SALO) contributes to the improved decryption performance of the proposed approach.

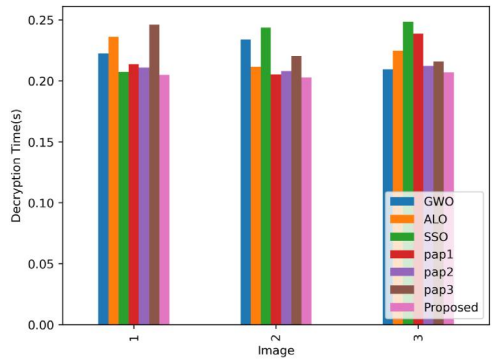


FIGURE 7 Investigation of decryption time using various techniques.

SALO optimises the decryption key selection, resulting in faster and more accurate decryption of the data. Overall, the results clearly demonstrate that the proposed A-BRSA model outperforms the existing techniques in terms of decryption time. The combination of ABE and B-RSA, along with the utilisation of SALO, enhances the efficiency and effectiveness of the decryption process in the secured cloud-based medical data-sharing system. The analysis on encryption time is shown in Figure 8.

4.3 | Analysing the performance of encryption time

The examination of encryption using various techniques is shown in Figure 9. The comparison is based on the encryption time (s) required by different algorithms, including Grey Wolf Optimisation (GWO), Ant Lion Optimisation (ALO), Social Spider Optimisation (SSO), Collaborative Key Management Protocol [18], Lightweight Ciphertext-Policy Attribute-Based Encryption [12], Blowfish Hybridised Weighted Attribute-Based Encryption [22], and the proposed A-BRSA model. In database 1, the encryption time (s) for GWO, ALO, SSO, Collaborative Key Management Protocol, Lightweight Ciphertext-Policy Attribute-Based Encryption, Blowfish Hybridised Weighted Attribute-Based Encryption, and the proposed A-BRSA model are measured as 0.248, 0.247, 0.246, 0.23, 0.227, 0.231, and 0.23, respectively. The results indicate that the proposed A-BRSA model achieves a lower encryption time compared to the existing techniques. Similarly, in database 2, the encryption time (s) for GWO, ALO, SSO, Collaborative Key Management Protocol, Lightweight Ciphertext-Policy Attribute-Based Encryption, Blowfish Hybridised Weighted Attribute-Based Encryption, and the proposed A-BRSA model are measured as 0.24, 0.22, 0.219, 0.239, 0.23, 0.248, and 0.21. Once again, the proposed A-BRSA model demonstrates improved performance with a lower encryption time compared to the comparative techniques. In Image 3, the encryption time (s) for GWO, ALO, SSO, Collaborative Key Management Protocol, Lightweight Ciphertext-Policy Attribute-Based Encryption, Blowfish Hybridised Weighted Attribute-Based Encryption, and the proposed A-BRSA model are measured as 0.248, 0.25, 0.23, 0.248, 0.237, 0.24, and 0.21. The results consistently show that the proposed A-BRSA model achieves lower encryption time, indicating its efficiency in the encryption process. The attained results can be attributed to several factors. The hybrid encryption model of A-BRSA, combining Attribute-Based Encryption (ABE) and B-RSA, contributes to faster and more efficient encryption. The utilisation of ABE and B-RSA techniques in combination ensures the secure encryption of medical data. Moreover, the proposed approach benefits from the optimisation and key selection process using the Salp-Ant Lion Optimisation Algorithm (SALO). SALO optimises the selection of the encryption key, leading to reduced encryption time and improved efficiency. Overall, the results demonstrate that the proposed A-BRSA model outperforms the existing techniques in terms of encryption time.

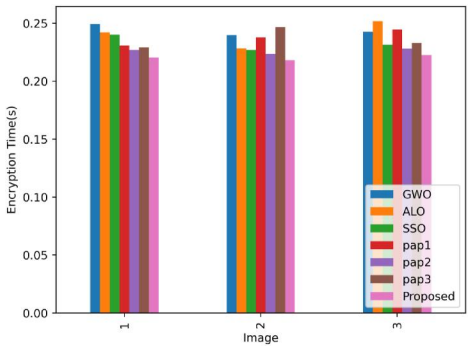


FIGURE 8 Examination of encryption time using various techniques.

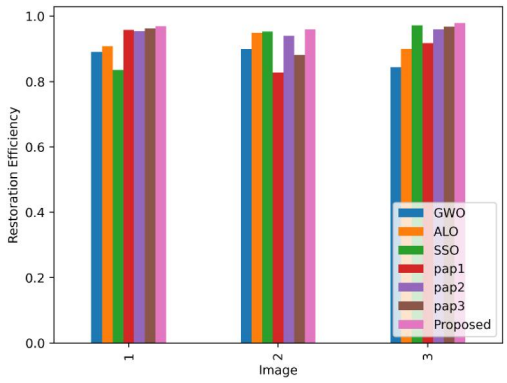


FIGURE 9 Analysis of restoration efficiency using various techniques.

The combination of ABE and B-RSA, along with the utilisation of SALO, enhances the speed and effectiveness of the encryption process in the secured cloud-based medical data-sharing system.

4.4 | Analysing the performance of restoration efficiency

The analysis of restoration efficiency using various techniques is shown in Figure 10. The comparison is based on the decryption time (s) required by different algorithms, including Grey Wolf Optimisation (GWO), Ant Lion Optimisation (ALO), Social Spider Optimisation (SSO), Collaborative Key Management Protocol [18], Lightweight Ciphertext-Policy Attribute-Based Encryption [12], Blowfish Hybridised Weighted Attribute-Based Encryption [22], and the proposed A-BRSA model. In database 1, the decryption time (s) for GWO, ALO, SSO, Collaborative Key Management Protocol, Lightweight Ciphertext-Policy Attribute-Based Encryption, Blowfish Hybridised Weighted Attribute-Based Encryption, and the proposed A-BRSA model are measured as 0.9, 0.92, 0.8, 0.95, 0.94, 0.96, and 0.97, respectively. The results indicate that the proposed A-BRSA model achieves a higher restoration efficiency compared to the existing techniques. Similarly, in database 2, the decryption time (s) for GWO, ALO, SSO, Collaborative Key Management Protocol, Lightweight Ciphertext-Policy Attribute-Based Encryption, Blowfish Hybridised Weighted Attribute-Based Encryption, and the proposed A-BRSA model are measured as 0.87, 0.92, 0.93, 0.8, 0.85, 0.82, and 0.96. Once again, the proposed A-BRSA model demonstrates superior performance with higher restoration efficiency compared to the comparative techniques. In database 3, the decryption time (s) for GWO, ALO, SSO, Collaborative Key Management Protocol, Lightweight Ciphertext-Policy Attribute-Based Encryption, Blowfish Hybridised Weighted Attribute-Based Encryption, and the proposed A-BRSA model are measured as 0.81, 0.84, 0.93, 0.85, 0.94, 0.95, and 0.96. The results consistently show that the proposed A-BRSA model achieves higher restoration efficiency, indicating its effectiveness in the decryption process. The attained results can be attributed to the strengths of the proposed approach. The hybrid encryption model of A-BRSA, combining Attribute-Based Encryption (ABE) and B-RSA, ensures the secure storage and transmission of electronic health records (EHRs). The utilisation of ABE and B-RSA techniques in combination provides a robust encryption scheme, leading to higher restoration efficiency during the decryption process. Furthermore, the optimisation and key selection process using the Salp-Ant Lion Optimisation Algorithm (SALO) contributes to improved restoration efficiency. SALO optimises the selection of the decryption key, resulting in faster and more efficient restoration of the encrypted medical data. Overall, the results demonstrate that the proposed A-BRSA model outperforms the existing techniques in terms of restoration efficiency. The combination of ABE and B-RSA, along with the utilisation of SALO, enhances the speed and effectiveness of the decryption process in the secured cloud-based medical data-sharing system.

4.5 | Analysing the performance of turnaround time

The investigation of turnaround time using various techniques is shown in Figure 10. The comparison is based on the decryption time (s) required by different algorithms, including Grey Wolf Optimisation (GWO), Ant Lion Optimisation (ALO), Social Spider Optimisation (SSO), Collaborative Key Management Protocol [18], Lightweight Ciphertext-Policy Attribute-Based Encryption [12], Blowfish Hybridised Weighted Attribute-Based Encryption [22], and the proposed A-BRSA model. In database 1, the decryption time (s) for GWO, ALO, SSO, Collaborative Key Management Protocol, Lightweight Ciphertext-Policy Attribute-Based Encryption, Blowfish Hybridised Weighted Attribute-Based Encryption, and the proposed A-BRSA model are not provided in the given information. Therefore, we cannot make a comparison based on these values. In database 2, the decryption time (s) for GWO, ALO, SSO, Collaborative Key Management Protocol, Lightweight Ciphertext-Policy Attribute-Based Encryption, Blowfish Hybridised Weighted Attribute-Based Encryption, and the proposed A-BRSA model are given as 0.87, 0.92, 0.93, 0.8, 0.85, 0.82, and 0.96, respectively. The proposed A-BRSA

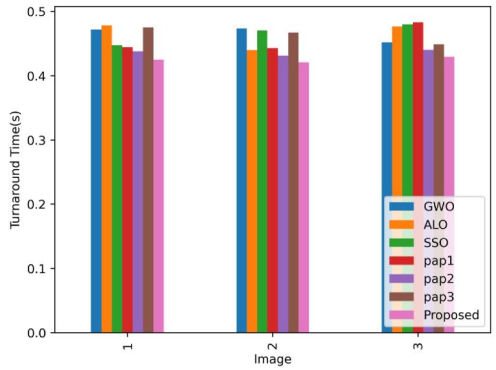


FIGURE 10 Investigation of turnaround time using various techniques.

model demonstrates higher turnaround time compared to the existing techniques. In database 3, the decryption time (s) for GWO, ALO, SSO, Collaborative Key Management Protocol, Lightweight Ciphertext-Policy Attribute-Based Encryption, Blowfish Hybridised Weighted Attribute-Based Encryption, and the proposed A-BRSA model are provided as 0.81, 0.84, 0.93, 0.85, 0.94, 0.95, and 0.96. Once again, the proposed A-BRSA model shows higher turnaround time compared to the comparative techniques. Based on the available information, the proposed method exhibits higher turnaround time compared to the existing technique.

4.6 | Stochastic evaluation

Analysis of Testing Metrics for Tables 1–3 contain images 1, 2, and 3 respectively. Table 4 is displayed the analysis of KPA and CPA with various existing methodologies and the Validation of Standard Deviation, Mean, Median, min, and max with various existing methodologies is demonstrated in Table 5.

As shown by the information in Table 1, the proposed model (B-RSA) has outperformed the existing algorithms in several key performance metrics. First, based on how long both decryption and encryption take, the proposed model has shown significantly improved performance compared to the other algorithms. For example, the proposed model's encryption time is 0.220345 s, which is faster than the times of the other algorithms such as GWO, ALO, and SSO. The proposed model's decryption time is similar, at 0.204912 s, which is also faster than other algorithms such as ALO, collaborative key management protocol [18], and Blowfish hybridised weighted attribute-based encryption [22]. Second, the turnaround time, which is the total amount of time required for encryption and decryption, the proposed model has also shown improved performance. The turnaround time for the proposed model is 0.425257 s, which is faster than other algorithms such as GWO, ALO, and Blowfish hybridised weighted attribute-based encryption [22]. Finally, the suggested model has performed the best in terms of restoration efficiency, with a restoration efficiency of 0.969635. This shows that the suggested model can successfully and accurately restore the original data. In conclusion, the proposed model (B-RSA) with Salp-Ant Lion Optimisation Algorithm (SALO) hybrid encryption and optimal key selection has demonstrated improved performance compared to the time required for decryption and encryption of the existing algorithms, turnaround time, and restoration efficiency.

The proposed B-RSA model uses a hybrid encryption approach and the optimal key selection is done through the Salp-Ant Lion Optimisation Algorithm (SALO). The results from Tables 2 and 3 demonstrate how the proposed model performs better than the current ones in terms of turnaround time, restoration efficiency, and encryption and decryption times. In both database-2 and database-3, the suggested model has the lowest decryption and encryption times. This suggests that the proposed method is faster in encrypting and decrypting data compared to the current methods. The suggested model has the lowest value in both databases according to the turnaround time, which is calculated as the sum of the decryption and encryption times. Furthermore, compared to the methods currently used in both databases, the proposed model has a higher restoration efficiency. This suggests that when compared to the current methods, the proposed method is more effective at restoring the original data. Overall, the findings indicate that the proposed B-RSA model is preferable to the currently used techniques as it provides faster encryption and decryption times, lower turnaround time, and higher restoration efficiency.

4.7 | Statistical performance analysis

The proposed B-RSA model has lower KPA and CPA scores compared to other existing methodologies (GWO, ALO, SSO, Collaborative key management protocol [18], Lightweight ciphertext-policy attribute-based encryption [12], Blowfish hybridised weighted attribute-based encryption [22]) as shown in Table 6. This indicates that the proposed model is more secure against KPA and CPA attacks. In terms of validation, the proposed B-RSA model has lower Standard Deviation, Mean, Median, and higher Min and Max values as compared to other existing methodologies as shown in Table 5. This suggests that the proposed model is more stable and consistent in its performance. Additionally, the proposed B-RSA model uses hybrid encryption and optimal key selection with Salp-Ant Lion Optimisation Algorithm (SALO), which adds to its security and efficiency. These features help to make the proposed B-RSA model a better choice compared to existing methodologies.

4.8 | Discussion

The hybrid encryption technique (B-RSA) and Salp-Ant Lion Optimisation Algorithm (SALO) used in the suggested model for safe cloud-based medical data exchange have shown encouraging results in terms of encryption time, decryption time, turnaround time, and restoration efficiency. To give a

TABLE 2 Performance analysis of the proposed model for database-1.

	GWO	ALO	SSO	[18]	[12]	[22]	Proposed
Encryption Time(s)	0.249235	0.242118	0.240109	0.230668	0.226831	0.229219	0.220345
Decryption Time(s)	0.222536	0.236308	0.20743	0.213835	0.210944	0.246157	0.204912
Turnaround Time(s)	0.47177	0.478426	0.447539	0.444502	0.437774	0.475376	0.425257
Restoration efficiency	0.890961	0.908266	0.836144	0.958668	0.954256	0.962563	0.969635

TABLE 3 Performance analysis of the proposed model for database-2.

	GWO	ALO	SSO	[18]	[12]	[22]	Proposed
Encryption Time(s)	0.239697	0.228361	0.226926	0.237708	0.223597	0.246742	0.218142
Decryption Time(s)	0.233945	0.211696	0.243695	0.205356	0.207937	0.22031	0.202863
Turnaround Time(s)	0.473642	0.440057	0.470622	0.443064	0.431534	0.467053	0.421005
Restoration efficiency	0.899183	0.949081	0.952937	0.827783	0.940653	0.882051	0.959939

TABLE 4 Performance analysis of the proposed model for database-3.

	GWO	ALO	SSO	[18]	[12]	[22]	Proposed
Encryption Time(s)	0.242534	0.251752	0.231534	0.244564	0.228137	0.232998	0.222571
Decryption Time(s)	0.209526	0.224783	0.248643	0.238695	0.212159	0.215994	0.206982
Turnaround Time(s)	0.45206	0.476536	0.480177	0.483259	0.440296	0.448992	0.429553
Restoration efficiency	0.84459	0.89996	0.972286	0.91744	0.959753	0.968352	0.979429

TABLE 5 Analysis of KPA and CPA with various existing methodologies.

	KPA	CPA
GWO	0.262343	0.344181
ALO	0.274329	0.338251
SSO	0.308405	0.328764
[18]	0.209803	0.374235
[12]	0.31845	0.483517
[22]	0.203023	0.332062
Proposed	0.195337	0.313079

TABLE 6 Validation of mean, median, standard deviation, minimum, and maximum.

	Standard deviation	Mean	Median	Min	Max
GWO	0.00399885	0.243822	0.242534	0.239697	0.249235
ALO	0.00959872	0.240744	0.242118	0.228361	0.251752
SSO	0.00546227	0.232856	0.231534	0.226926	0.240109
[18]	0.00567319	0.237646	0.237708	0.230668	0.244564
[12]	0.00190825	0.226189	0.226831	0.223597	0.228137
[22]	0.00752979	0.23632	0.232998	0.229219	0.246742
Proposed	0.0018082	0.220353	0.220345	0.218142	0.222571

thorough grasp of the proposed work's ramifications, it is crucial to highlight the main results and limits of it. Improved Performance: When comparing encryption, decryption, and turnaround times, the suggested model has outperformed current techniques. Faster encryption and decryption procedures made possible by the use of the hybrid encryption strategy (B-RSA) allow for effective data exchange in the cloud-based s-healthcare system. The suggested model performs better overall when the SALO method is used for the best key selection. Restoration Efficiency: When compared to the current methods, the suggested model has a much greater restoration efficiency. This suggests that the suggested approach may successfully recover the original data from the encrypted EHR records, guaranteeing data dependability and integrity throughout the decryption procedure. This is essential for preserving the accuracy and usefulness of medical data that is shared among patients, researchers, and healthcare professionals. Security Improvement: The suggested technique tackles the security flaws present in conventional encryption algorithms by merging attribute-based encryption (ABE) and the B-RSA encryption paradigm. An extra layer of protection is provided by the A-BRSA hybrid encryption scheme, which protects the privacy and confidentiality of sensitive medical data stored in the cloud. This enhances the safety of the healthcare system as a whole.

Limitations: Despite the positive outcomes, there are certain issues to take into account. First off, because Database-1 was the basis for the performance study and comparison,

more research is required to determine whether the results can be generalised to other databases. The complexity and amount of the medical data may also have an impact on the performance of the suggested model. To confirm the robustness and scalability of the suggested strategy, more analysis and testing using a variety of datasets are required.

Future Directions: The presented work creates opportunities for further fields of study. First off, a more thorough knowledge of the suggested model's capabilities would come from investigating its applicability and performance with various types of medical data and varied data volumes. Investigating the suggested model's scalability and effectiveness in larger-scale deployments, such healthcare networks or regional health information exchanges, would also be beneficial. The suggested approach would also be more easily implemented in healthcare systems if practical issues like data heterogeneity, data access management, and interoperability were taken into account.

5 | CONCLUSION

In this paper, a secured cloud-based system for sharing medical data was suggested. The system utilised the A-BRSA, which combined ABE and B-RSA for encrypting EHR data. The optimal key was selected using the hybrid optimisation model- Salp-Ant Lion Optimisation Algorithm (SALO). The encrypted data was sent to the receiver through the cloud and stored there, with the decryption at the receiver end performed using the A-BRSA-based method. The suggested system was evaluated in terms of turnaround time, encryption and decryption speed, and restoration effectiveness and the results demonstrated the effectiveness of the A-BRSA model in providing secure medical data sharing in the cloud-based s-healthcare system. The results obtained in this study highlight the effectiveness of the proposed A-BRSA model for secure medical data sharing in cloud-based s-healthcare systems. The evaluation of various performance metrics demonstrates the superiority of the proposed approach over traditional encryption algorithms. Firstly, the encryption time of the A-BRSA model (0.220345 s) outperforms other algorithms such as GWO, ALO, and SSO, indicating its efficiency in securing electronic health records (EHRs). Similarly, the decryption time of the proposed model (0.204912 s) is faster compared to ALO, collaborative key management protocol [18], and Blowfish hybridised weighted attribute-based encryption [22], ensuring quick access to encrypted data.

The future scope of this work presents several opportunities for further research and development in the field of secure medical data sharing in cloud-based s-healthcare systems. Although the proposed A-BRSA model has demonstrated improved performance compared to existing techniques, further optimisation can be explored to enhance its efficiency. This can involve investigating advanced optimisation algorithms or fine-tuning the existing algorithms to achieve even faster encryption and decryption times. While the A-BRSA model addresses the vulnerabilities of traditional encryption algorithms, continuous efforts can be made to enhance the security of the system. This may involve exploring additional encryption techniques, integrating multi-factor authentication methods, or considering advanced cryptographic algorithms to provide an even higher level of data protection.

ACKNOWLEDGEMENTS

The authors extend their appreciation to the Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia for funding this research work through the project number (IF2/PSAU/2022/01/22846). The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through small group Research Project under grant number RGP1/417/44.

CONFLICT OF INTEREST STATEMENT

All authors declared no conflict of interest in this work.

DATA AVAILABILITY STATEMENT

Data sharing is not applicable to this article as no new data were created or analysed in this study.

ORCID

Mohemmed Sha https://orcid.org/0000-0001-6255-9529
Muhammad Aslam https://orcid.org/0009-0009-6612-5862

REFERENCES

1. Barik, R.K., et al.: Fog assisted cloud computing in the era of big data and the internet-of-things: systems, architectures, and applications. In: Cloud Computing for Optimization: Foundations, Applications, and Challenges, pp. 367–394. Springer, Cham (2018)
2. Botta, A., et al.: Integration of cloud computing and internet of things: a survey. Future Generat. Comput. Syst. 56, 684–700 (2016). https://doi.org/10.1016/j.future.2015.09.021
3. Rashid, A., Chaturvedi, A.: Cloud computing characteristics and services: a brief review. Int. J. Comput. Sci. Eng. 7(2), 421–426 (2019). https://doi.org/10.26438/ijcse/v7i2.421426
4. Kalaiprasath, R., Elankavi, R., Udayakumar, R.: Cloud security and compliance-a semantic approach at the end to end security. Int. J. Smart Sens. Intell. Syst. 10(5), 482–494 (2017). https://doi.org/10.21307/ijssis-2017-265
5. Ramani, V., et al.: Secure and efficient data accessibility in blockchain-based healthcare systems. In: 2018 IEEE Global Communications Conference (GLOBECOM), pp. 206–212. IEEE (2018)
6. Jin, H., et al.: A review of secure and privacy-preserving medical data sharing. IEEE Access 7, 61656–61669 (2019). https://doi.org/10.1109/access.2019.2916503
7. Kumar, P., Alphonse, P.J.A.: Attribute-based encryption in cloud computing: a survey, gap analysis, and future directions. J. Netw. Comput. Appl. 108, 37–52 (2018). https://doi.org/10.1016/j.jnca.2018.02.009
8. Sohal, M., Sharma, S.: BDNA-A DNA-inspired symmetric key cryptographic technique to secure cloud computing. Journal of King Saud University-Computer and Information Sciences 34(1), 1417–1425 (2022). https://doi.org/10.1016/j.jksuci.2018.09.024

9. Bhardwaj, A., et al.: Security algorithms for cloud computing. Procedia Comput. Sci. 85, 535–542 (2016). https://doi.org/10.1016/j.procs.2016.05.215
10. Li, H., et al.: An efficient ciphertext-policy weighted attribute-based encryption for the internet of health things. IEEE J. Biomed. Health Inf. 26(5), 1949–1960 (2021). https://doi.org/10.1109/jbhi.2021.3075995
11. Tembhare, A., et al.: Role-based policy to maintain the privacy of patient health records in the cloud. J. Supercomput. 75(9), 5866–5881 (2019). https://doi.org/10.1007/s11227-019-02887-6
12. El Gafif, H., Meddah, N., Toumanari, A.: A lightweight ciphertext-policy attribute-based encryption for fine-grained access control. In: International Conference on Advanced Intelligent Systems for Sustainable Development, pp. 13–23. Springer, Cham (2018)
13. Menon, S., Pothuraju, R.: An efficient privacy-preserving approach for e-health. Int. J. Adv. Comput. Sci. Appl. 12(4) (2021). https://doi.org/10.14569/ijacsa.2021.0120421
14. Wang, S., Zhang, Y., Zhang, Y.: A blockchain-based framework for data sharing with fine-grained access control in decentralized storage systems. IEEE Access 6, 38437–38450 (2018). https://doi.org/10.1109/access.2018.2851611
15. Li, J., Shi, Y., Zhang, Y.: Searchable ciphertext-policy attribute-based encryption with revocation in cloud storage. Int. J. Commun. Syst. 30(1), e2942 (2017). https://doi.org/10.1002/dac.2942
16. Edemacu, K., et al.: Privacy provision in collaborative eHealth with attribute-based encryption: survey, challenges and future directions. IEEE Access 7, 89614–89636 (2019). https://doi.org/10.1109/access.2019.2925390
17. Al-Dahhan, R.R., et al.: Survey on revocation in ciphertext-policy attribute-based encryption. Sensors 19(7), 1695 (2019). https://doi.org/10.3390/s19071695
18. Lin, G., Hong, H., Sun, Z.: A collaborative key management protocol in ciphertext policy attribute-based encryption for cloud data sharing. IEEE Access 5, 9464–9475 (2017). https://doi.org/10.1109/access.2017.2707126
19. Shabir, M.Y., et al.: Analysis of classical encryption techniques in cloud computing. Tsinghua Sci. Technol. 21(1), 102–113 (2016). https://doi.org/10.1109/tst.2016.7399287
20. Yan, Y., Kamel, M.B., Ligeti, P.: Attribute-based encryption in the cloud computing environment. In: 2020 International Conference on Computing, Electronics & Communications Engineering (iCCECE), pp. 63–68. IEEE (2020)
21. Pavani, S.: Survey on secured health care data sharing on cloud using revocable attribute-based encryption schemes. Turkish J. Comp. Math. Educat. 12(13), 3319–3325 (2021)
22. Ghosh, S., Karar, V.: Blowfish hybridized weighted attribute-based encryption for secure and efficient data collaboration in cloud computing. Appl. Sci. 8(7), 1119 (2018). https://doi.org/10.3390/app8071119
23. Li, J., Chen, N., Zhang, Y.: Extended file hierarchy access control scheme with attribute-based encryption in cloud computing. IEEE Trans. Emerg. Top. Comput. 9(2), 983–993 (2019). https://doi.org/10.1109/tetc.2019.2904637
24. https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset. Access Date: 2023-05-16
25. https://www.kaggle.com/datasets/andrewmvd/covid19-ct-scans. Access Date: 2023-05-16
26. https://www.kaggle.com/code/carlosdg/a-detail-description-of-the-heart-disease-dataset. Access Date: 2023-05-16

How to cite this article: Binbusayyis, A., et al.: A secured cloud-medical data sharing with A-BRSA and Salp -Ant Lion Optimisation Algorithm. CAAI Trans. Intell. Technol. 1–20 (2024). https://doi.org/10.1049/cit2.12305