Enhance the Security of the Cloud Using a Hybrid Optimization-Based Proxy Re-Encryption Technique Considered Blockchain

Ahmed I. Alutaibi
Department of Computer Engineering-College of Computer and Information Sciences,
Majmaah University, Majmaah 11952, Saudi Arabia

***Abstract*—Every day, a vast amount of data with incalculable value will be generated by the IoT devices that are deployed in various types of applications. It is crucial to ensure the reliability and safety of IoT data exchange in a cloud context because this data frequently contains the user's private information. This study presents a novel encrypted data storage and security system using the blockchain method in conjunction with hybrid optimization-based proxy re-encryption (HO-PREB). Dependency on outside central service providers is eliminated by the HO-PREB-based consensus process. In the blockchain system, several consensus nodes serve as proxy service nodes to restore encrypted data and merge transformed ciphertext with private data. Hybrid owl and bat optimization is employed to select the optimal key for enhancing security. This removes the limitations associated with securely storing and distributing private encrypted data via a distributed network. Moreover, the blockchain's distributed ledger ensures the permanent storage of data-sharing records and outcomes, ensuring accuracy and dependability. The simulated experiments of the designed model are evaluated with existing cryptographic techniques and gain a lower latency of 3.2 s and a lower turnaround time of 45 ms. Furthermore, the developed technique enhances cloud system security and possesses the capability to detect and mitigate attacks in the cloud environment.**

Keywords*—*Cloud security; Internet of Things; proxy re-encryption; blockchain; data sharing; hybrid optimization

I. Introduction

Cloud computing is one of the best and most difficult platforms accessible today, and businesses of all sizes are starting to employ its services. A range of cloud deployment techniques are available, and cloud services are provided according to requirements, like protecting the cloud system's internal and external safety [1]. Common technology flaws, malware, hackers, and other similar threats, unauthorized access, inadequate precautions, denial of service, unsecured interfaces, profile or business traffic hijacking, and data leaks are the primary risks to cloud computing protection [2]. Cloud control covers a broad range of topics, such as handling resources, hardware and software safety, cloud data protection, and resource consumption [3]. Despite the many benefits of cloud storage, cloud users often prioritize security and privacy issues, which discourages businesses and organizations from adopting big changes. While using the cloud benefits enterprises and institutions, privacy and confidence are the most vital challenges [4, 5]. The data of cloud customers is highly vulnerable to loss, leakage, or attack, and they are left with no way out of this untenable scenario [6].

When exporting and buying offerings from cloud providers, cloud customers may apply blockchain technology, a novel and evolving technology, to boost data privacy and confidence [7, 8]. Blockchain can provide improved security based on centralized database safety. Blockchain monitors the set of data that is encrypted, stored, and linked to the previous block using an encrypted hash function [9, 10]. A blockchain is a type of networked ledger that can prevent manipulation and store operations. Peer-to-peer networks are typically used to manage blockchains, which are meant to guard against outside interference [11, 12]. The security provided by the blockchain system can be compared to the safety of centralized data storage. From a management standpoint, attacks and damage to data storage can be prevented [13]. Furthermore, the accessible feature of the blockchain can facilitate data openness when used in an environment where sharing information is mandated. These benefits allow it to be used in a wide range of situations, including those involving the financial sector and the Internet of Things (IoT), and its applications are expected to expand [14]. Cloud computing has been integrated into many IT systems since it is effective and widely accessible. Moreover, privacy and cloud security issues have been investigated as crucial security components [15].

As IoT technology develops quickly, a vast number of IoT devices are being used in many application situations. As a result, ensuring dependability and IoT data exchange security is essential [16]. A significant volume of IoT data is generated and managed by the data owners. These enormous amounts of data must be encrypted and sent to a trusted organization for storage because IoT devices have limited storage. Before transferring the data to cloud databases and shared storage systems, they can choose to encrypt it [17, 18]. To safeguard the privacy and security of data owners (DO), an efficient access control system must be established. The users don't have a completely reliable means of communication through which to exchange the decryption key because the data is encrypted. The data owners must download, decrypt, and re-encrypt the material before they may share it directly with the recipients [19]. Unfortunately, because of their limited processing capacity, the DO is unable to pay to decrypt the data before re-encrypting it for the recipients. As a result, they can exchange IoT data by using the PRE technique. This transfers the burden of ciphertext decryption and encryption from the information owners to the procedure of creating a new encryption key, thereby redistributing the workload [20].

This work presents the development of a hybrid owl search and bat PRE method to improve cloud computing security and robustness. Additionally, the blockchain model generates blocks and secure cloud storage of user data to enhance privacy. Additionally, the blockchain concept is built to give extra scalability for safeguarding data in blocks and secret keys, and data is safely decrypted via PRE. Additionally, the HO-PREB system avoids reliance on outside central service providers. In the distributed ledger network, several consensus nodes serve as proxy service nodes to merge translated cipher text and re-encrypt data. This system can guarantee the suitability, security, and dependability of cloud computing's IoT data exchange.

The overview of the paper is arranged in sections. Sections II and III detail related works and a problem statement; Section IV explains the proposed methodology; Section V gives detailed results and discussion; and Section VI ends with a conclusion and future scope.

II. Literature Review

A. Related Works

A distributed data-controlled sharing strategy based on PRE is proposed by Ismail et al. [21]. First, a method for PRE is built using the blockchain and SM2. The information-controlled sharing method protects transaction information confidentiality while enabling data security sharing through the use of PRE. A system for adaptive user rights modification is suggested. To completely maintain the privacy of transaction data and carry out the evaluation of information access permission by monitoring PRE key settings, they are creating a PRE method with SM2.

A PRE method was created by Kwame et al. [22] for secure cloud-based data-sharing scenarios. Data owners can send their protected data to the cloud using identity-based encryption (IBE), where it may be accessed by authorized users using the PRE architecture. Moreover, it increases the quality of service by effectively supplying content that has been stored in the proxy by utilizing information-centric network functions. Moreover, blockchain is an innovative technology that permits data sharing to be decentralized. It accomplishes fine-grained control over access to data and reduces bottlenecks in central databases.

To increase safety and confidentiality in a private blockchain, Bharat et al. [23] introduced the Splitting of PRE Method (Split-PRE), which is based on the IoT. To address trust issues as well as scalability issues, this paper suggests a blockchain-based PRE method that will streamline transactions. The IoT data is saved by the system in a decentralized cloud after encryption. Through the use of an effective PRE method, the owner and the person in the smart contract can both view the data.

A blockchain-based ecosystem for IoT data exchange is presented by Ahsan et al. [24]. Additionally, transport the data from the information generator to the consumer safely and anonymously by using a PRE mechanism. Without the assistance of a reliable third party, the system establishes dynamic, temporal connections between data consumption and sensors to share the gathered IoT data. This cutting-edge website for storing, exchanging, and handling sensor data is quick, easy, and safe.

Yingwen et al. [25] combined blockchain computing with PRE to create a novel encrypted data storage and communication design. The utilization of threshold PRE as a compromise technique avoids reliance on external central service providers. In the blockchain system, several consensus nodes serve as proxy service nodes, combining translated ciphertext and re-encrypting data. The findings demonstrate that the suggested architecture can raise a reasonable time delay while satisfying the high demands for data access.

To accomplish effective data sharing and data accuracy checking, Tao Feng et al. [26] suggested a technique that utilizes IBE. Additionally, a blockchain platform is integrated to ensure secure and regulated data storage, addressing issues such as manipulation of data and insufficient supervision on external servers. Lastly, assess the computation and transmission overhead to prove the security of the planned system. The findings demonstrated that the designed method surpasses other plans in terms of efficiency and security against specific plaintext attacks with observable characteristics.

To enable user identification and access to the public key's binding properties on the Internet of Medical Things, Hongmei et al. [27] offer a safe data sharing system named PRE for safe information exchange with blockchain, thereby boosting the security of data sharing. The created approach implements the management and quick query of users by adding them to the accumulator through the application of a blockchain smart agreement. Lastly, carry out the four-stage computing performance evaluation experiments to enhance the effectiveness of the encryption, re-encryption, decryption, and re-decryption computations.

B. Problem Statement

Transaction data is kept on a blockchain in a decentralized shared global ledger. Finding the right balance while sharing data safeguarding privacy, and usefulness is difficult. Furthermore, one difficult issue is the dynamic modification of blockchain data access permissions [28]. The problem description in the realms of cloud computing and blockchain technology centers on data availability, confidentiality, and integrity protection in distributed systems. Maintaining the confidentiality of sensitive data processed and stored across enormous networks of linked servers is a difficulty with cloud computing [29]. Strong security measures are necessary to reduce risks associated with problems like unauthorized access, data breaches, and service interruptions. Similarly, it is critical to protect distributed ledgers' confidentiality and agreement in blockchain technology from risks like 51% attacks, double-spends, and vulnerabilities in smart contracts. The most frequent and difficult problems with cloud computing safety are low efficiency, trust concerns, limited scalability, difficulty in maintaining access restrictions, and leakage of sensitive data [30]. To meet these problems and maintain the confidence and integrity necessary for these revolutionary technologies, creative encryption methods, strict access controls, and robust network designs are needed.

III. Proposed Methodology

Create the best possible keys and increase the security of the proposed model by designing a hybrid owl search and bat optimization (HOSBA) algorithm. The blockchain network, storage service providers (SSP), data users (DU), and data owners (DO) make up the four primary components of the created system. Fig. 1 depicts the planned strategy process.

C. Process of the Hybrid Optimization-based PRE Model

To improve the public cloud privacy of user information, hybrid owl search and bat optimization (HOSBA) methods are used for the key generation phase. These improvements choose the most reliable and safe optimum key to improve cloud computing security.

1) System construction: To increase cloud storage security and safe data exchange, it integrates threshold PRE, hybrid optimization, and the blockchain consensus method. Elliptic Curve Digital Signature Algorithm (ECDSA) [31], PRE [32], and HOSBA are consulted in the construction of the developed mechanism. This approach can use a collection of N's blockchain consensus nodes to carry out the PRE procedure. The delegate can access the updated ciphertext by utilizing their private keys to decrypt it when necessary within the blockchain network of nodes in re-encryption. During the key generation step, the HOSBA model is employed to choose the best possible key. Below is a full explanation of the developed technique and process.

System setup

$$\left(1^{\gamma},\bar{u}\right)\rightarrow\overline{pp}$$

The security parameters 1^{γ} is selected as inputs and the output of the public parameter is $\overline{pp}$. The $\overline{pp}$ is used for creating own public and private keys. Initially, it takes security parameter γ as the results and inputs for a bilinear map m : $\hat{G}_1 * \hat{G}_1 \rightarrow \hat{G}_2$, where $\hat{G}_1$, $\hat{G}_2$ Is a prime ordermultiplicative cyclic group p. A random generator is selected $g \in \hat{G}_1$, and calculated $z = m\left(\hat{g},\hat{g}\right)$. Moreover, four hash functions are developed and are detailed in eqn. (1).

$$\hat{H}_0:\{0,1\}^{*}\rightarrow\hat{G}_1, \hat{H}_1:\{0,1\}^{*}\rightarrow\hat{G}_1, \hat{H}_2:\hat{G}_2\rightarrow\{0,1\}^{\log_2 p}, \hat{H}_3: \hat{G}_2\rightarrow\{0,1\}^{*} \quad (1)$$

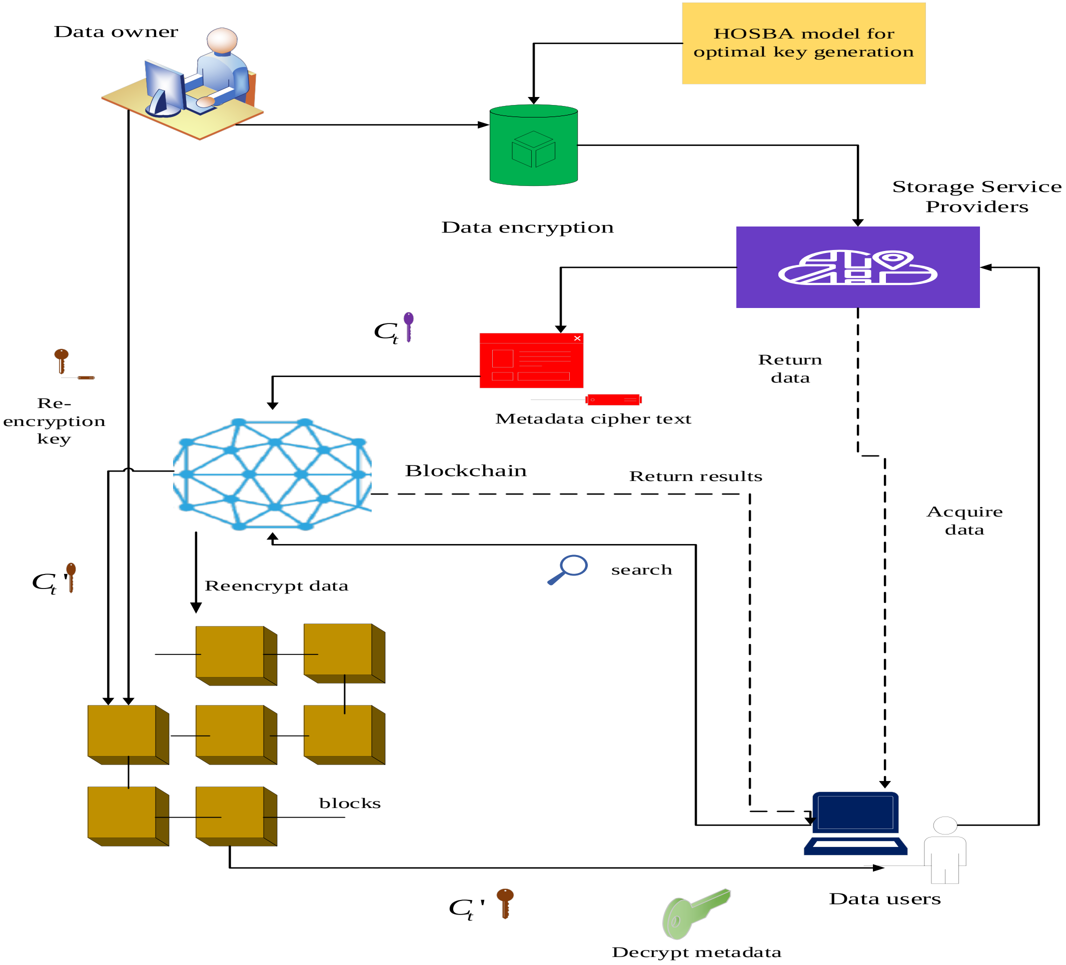

Fig. 1. Process of developed model.

Execute the procedure for parameter generation to produce shared parameters and is exhaustive in Eq. (2).

$$P_r = \left(\widehat{g}, z, m, p, \widehat{G}_1, \widehat{G}_2, \widehat{H}_0, \widehat{H}_1, \widehat{H}_2, \widehat{H}_3\right) \tag{2}$$

D. Key Generation

The public and private key pairs are created by the DU and the DO based on the provided public parameters. The DO selected at random $u \in z_p^*$ is the private key and calculates the public key as $\overline{pk_{do}} = \widehat{g}^u$. DU selected at random $v \in z_p^*$ is the private key and calculates the public key as $\overline{pk_{du}} = \widehat{g}^v$.

Hybrid owl search and bat algorithm:

The optimal key is a cryptographic key that is chosen or fine-tuned to maximize the security of encrypted data. This key is crucial for ensuring that data is securely encrypted and protected against unauthorized access. The Hybrid Owl and Bat optimization algorithm is used to find the best possible key for encryption. This optimization process involves selecting parameters for encryption that enhance security measures, such as resisting attacks or ensuring robustness against various threats. This algorithm explores the key space efficiently to identify the key that provides the highest level of security.

A novel hybrid optimization approach has been developed to select the best optimal key for enhancing the security of data in the cloud. It draws inspiration from the actions of the Bat Optimization (BAO) [33] and Owl Optimization Algorithm (OOA) [34] when a threat is present. The proposed hybrid optimization method selects the best optimal key according to the efforts of both parties to increase efficiency and security. It conceptually integrates BAO and OOA.

Owl begins its optimization process using a range of initial random solutions that match the starting positions of owls in a forest. The n quantity of owls in a woodland, x is the space of dimensions used for searches. Next, a random position is saved for them in a $n \times x$ matrix as given in Eq. (3).

$$\vec{o} = \begin{bmatrix} \vec{o}_{1,1} & \vec{o}_{1,2} & \cdots & \vec{o}_{1,x} \\ \vec{o}_{2,1} & \vec{o}_{2,2} & \cdots & \vec{o}_{2,x} \\ \vdots & \vdots & \vdots & \vdots \\ \vec{o}_{n,1} & \vec{o}_{n,2} & \cdots & \vec{o}_{n,x} \end{bmatrix} \tag{3}$$

The element of the matrix $\vec{o}_{i,j,}$ characterizes the j^{th} variable of an i^{th} owl. Eq. (4) is utilized to assign the beginning location of every owl in the forest based on a uniform distribution.

$$\vec{o}_i = \vec{o}_l + \vec{u}\left(0,1\right) \times \left(\vec{o}_u - \vec{o}_l\right) \tag{4}$$

Let, $\vec{o}_l$ and $\vec{o}_u$ are the lower and upper boundaries of i^{th} owl $\vec{o}_i$ in j^{th} length and $\vec{u}\left(0,1\right)$ is an arbitrary amount in the range [0,1] that is evenly distributed. An objective function is used to assess the fitness of each owl's placement inside a forest, and the results are kept in the matrix shown in Eq. (5).

$$\vec{f} = \begin{bmatrix} \vec{f}_1\left(\left[\vec{o}_{1,1}, \vec{o}_{1,2}, ...\vec{o}_{1,x}\right]\right) \\ \vec{f}_2\left(\left[\vec{o}_{2,1}, \vec{o}_{2,2}, ...\vec{o}_{2,x}\right]\right) \\ \vdots \\ \vdots \\ \vec{f}_n\left(\left[\vec{o}_{n,1}, \vec{o}_{n,2}, ...\vec{o}_{n,x}\right]\right) \end{bmatrix} \tag{5}$$

The current study assumes that each owl's positional fitness value is directly correlated with the volume of information it hears. Since the greatest owl is closer to the vole, it receives the maximum intensity. The information concerning the adjusted intensity of the i^{th} owl is calculated using Eq. (6) and is used to update the position.

$$\vec{I}_i = \frac{\vec{f}_i - \vec{w}}{\vec{b} - \vec{w}} \tag{6}$$

Equation (7) calculates the details of the separation between each owl and its meal.

$$\vec{r}_i = \left\|\vec{o}_i, \vec{v}\right\|_2 \tag{7}$$

Let, $\vec{v}$ is the position of the prey that the most agile owl finds. Additionally, it is presumable that the forest has a single vole, or worldwide optimum. Owls fly silently in the direction of their prey. As a result, they experience altered intensity by the sound intensity inverted square law. The variation in strength for i^{th} owl can be acquired using Eq. (8).

$$\vec{I}c_i = \frac{\vec{I}_i}{\vec{r}_i^{\,2}} + random\_noise \tag{8}$$

Eq. (8), $\vec{r}_i^{\,2}$ is used instead of $4\pi\vec{r}_i^{\,2}$ and thought that adding random noise to the surroundings will enhance the realism of the computational framework. Since voles are active in the actual world, their mobility compels owls to adjust their existing location softly. At this point, the optimal optima key is generated for the owl's new position phase via bat optimization, which improves the cloud computing privacy procedure. All bats utilize echolocation to detect distance, and in some mysterious way, they are also able to distinguish between background obstacles and food/prey; Bats fly arbitrarily with velocity $\widehat{v}_i^t$ at position $\widehat{x}_i^t$ with a fixed frequency $\widehat{f}_i$ to search for prey. Depending on the closeness of their target, they can automatically modify the frequency and rate at which they emit pulses within the interval [0, 1].

Thus, Eq. (9), which gives the new positions of the hybrid owl and bat model.

$$\vec{o}_i^{t+1} = \begin{cases} \vec{o}_i^{t} + \beta \times \vec{I}c_i \times \left|\alpha \vec{v} - \vec{o}_i^{t}\right| & if \vec{p}_v < 0.5 \\ \widehat{x}_i^{t-1} + \widehat{v}_i^{t} & if \vec{p}_v < 0.5 \end{cases} \quad (9)$$

Let, $\vec{p}_v$ is the likelihood of a vole moving, α is a uniformly distributed random number across the interval [0, 0.5], and β is a constant that decreases linearly from 1.9 to 0. β encourages the investigation of the search space and first makes significant alterations. Moreover, $\widehat{x}_i^{t-1}$ is denoted as the new position of the bats concerning time $t-1$, $\widehat{v}_i^{t}$ is denoted as velocity at time t, and the velocity of the bat is measured using Eq. (10).

$$\widehat{v}_i^{t} = \widehat{v}_i^{t-1} + \left(\widehat{x}_i^{t} - \widehat{x}_*\right)\widehat{f}_i \quad (10)$$

$\widehat{f}_i$ is considered as frequency, $\widehat{x}_i^{t}$ is denoted as the new position of the bats, and $\widehat{x}_*$ is denoted as the current global best position. Ultimately, the optimized model selects the best secret, private, and public key to secure the user data from unauthorized access. Algorithm 1 provides the Hybrid owl search and bat algorithm.

Algorithm 1: Hybrid owl search and bat algorithm

```
1. Initialize the algorithm by setting the population size, number of iterations, boundaries of the search space, and parameters for Owl and Bat Optimization.

2. Generate initial owl population by randomly generating initial positions for owls within the search space, evaluating the fitness of owl positions using the objective function, and storing the fitness values in a matrix.

3. Start the optimization process by beginning the main loop for iterations.

    Owl Optimization Phase:
    - Calculate initial positions of owls based on random distribution.
    - Update positions by calculating the intensity of information and adjusting based on proximity to prey.
    - Introduce random noise to simulate a realistic search.
    - Recalculate fitness of updated positions and store the best one found.

    Update Intensity and Distance:
    - Compute intensity for each owl based on proximity to prey.
    - Adjust positions based on intensity and random noise.
    - Calculate the distance between owls and prey and update positions.

    Bat Optimization Phase:
    - Adjust the frequency of bats based on distance to the best-known solution.
    - Update velocity and position of bats using calculated frequency.
    - Perform a local search around the best solution, adjusting positions based on probability.
    - Update positions and velocities based on proximity to prey and best-known positions.

    Combine Results from Both Algorithms:
    - Compare the best solutions from Owl and Bat Optimization phases.
    - Update the global best solution if a better position is found.

4. Conclude the optimization by finalizing the best-found solution as the optimal key after completing iterations.

5. Output the optimized key generated from the hybrid optimization process.
```

E. Data Encryption

$$\left(D_{ek}, D\right) \rightarrow C_t$$

Additionally, the IoT data is encrypted using a symmetric encryption technique. Users using symmetric key encryption must be aware of a shared secret key. The encrypted data C_t is generated using a common secret key D_{ek} are submitted for SSP to store the encrypted data using Eq. (11).

$$C_t = encrypt\left(D_{ek}, D\right) \quad (11)$$

F. Metadata Encryption

$\left(\overline{pp}, \overline{pk_{do}}, \widehat{m}\right) \rightarrow C_t\widehat{m}$: Metadata $\widehat{m}$ is encrypted by the DO by using a public key, and sends cipher text $C_t\widehat{m}$ to the blockchain. Moreover, a DO private key is desired to decrypt the data. Metadata $\widehat{m}$ contains the data summary, data store location, data decryption key D_{dk}, C_t, etc. Metadata encryption is detailed in eqn. (12).

$$C_t\widehat{m} = Enc\left(\overline{pk_{do}}, \widehat{m}\right) = \left(\widehat{g}_{uh}, \widehat{m}z_h\right) \quad (12)$$

Where, $z = m\left(\widehat{g}, \widehat{g}\right)$, h is an arbitrary coefficient.

G. Re-Encryption Key Generation

By their private key and the public key of the sender DU who wishes to obtain the data, the DO computes and produces the re-encryption key. The $\overline{sk}_{du}$ is the input secret key, the designated delegate's public key pk_{du}, proxy nodes quantity $\widehat{p}_n$, and the threshold t_h, ReKeyGen, a technique for generating re-encryption keys, calculates n components of the key used to re-encrypt data between DO and DU in Eq. (13).

$$rk_{do\to du} = \widehat{g}^{v/u} \tag{13}$$

Using the eqn. (14) that follows,

$$\overline{f}\left(x\right)=\sum_{i=1}^{h}\overline{f}\left(x_i\right)\gamma_{ij},\gamma_{ij}=\prod_{I=1,L\neq j}^{h}\frac{x-x_I}{x_i-x_I}, \tag{14}$$

each proxy node's share of the re-encryption key $rh_{o\to u}^{i}=\widehat{g}^{\overline{f}(x_i)} \bmod p$, and, $\widehat{p}_n\ i=1,2,3,.....n$. The ciphertext transformation requests will be posted to the blockchain system by the DO.

H. Re-Encryption

The ciphertext transformation procedure is started by the majority of nodes in the distributed ledger network in responding to re-encryption requests after they have received the re-encryption key made public through the DO. It re-encrypts $C_t\widehat{m}$ from the DO to the DU, according to $C_t\widehat{m}=\left(\widehat{g}_{uh},\widehat{m}z_h\right)$. Re-encryption keys allow proxy nodes to modify the original ciphertext. $rh_{o\to u}^{i}$, $i=1,2,3,.....n,$ for $\widehat{p}_n$ to the $C'_t\,\widehat{m}$ over the following Eq. (15).

$$m\left(\widehat{g}_{uh},\left(rh_{\widehat{p}_n}\right)^{\gamma_i}\right)=m\left(\widehat{g}_{uh},\widehat{g}^{\left[\gamma_i.\overline{f}\left(x_i\right)\right]}\right)=z^{uh\left[\gamma_i.\overline{f}\left(x_i\right)\right]} \tag{15}$$

Every proxy node $\widehat{p}_n$ can access the encrypted text using Eq. (16).

$$C_{v_i}=\left(z^{uh\left[\gamma_i.\overline{f}\left(x_i\right)\right]},\widehat{m}z_h\right) \tag{16}$$

When t out of n the ciphertext is appropriately completed by proxy nodes in the re-encryption computation. C_{v_i} can be integrated into the updated ciphertext. $C'_t\,\widehat{m}$, which DU can decode using their private key using Eq. (17) and (18)

$$\prod_{i=1}^{t}z^{uh\left[\gamma_i.\overline{f}\left(x_i\right)\right]}=z^{uh\sum_{i=1}^{h}\gamma_i.\overline{f}\left(x_i\right)}=z^{uh.v/a}=z^{vh} \tag{17}$$

$$C'_t\,\widehat{m}=\left(z^{vh},\widehat{m}z_h\right)\left(\bmod p\right) \tag{18}$$

Following the completion of the consensus confirmation by the proxy nodes, the new block will contain the entire interpreted ciphertext.

I. Metadata Decryption

The private key $\overline{sk}_{du}$ of the DU is suitable for decrypting the ciphertext metadata $C'_t\,\widehat{m}=\left(z^{vh},\widehat{m}z_h\right)=\left(\widehat{\alpha},\widehat{\beta}\right)$ as well as the subsequent metadata using Eq. (19).

$$\widehat{m}=\widehat{\beta}\Big/\widehat{\alpha}^{1/\overline{sk}_{du}} \tag{19}$$

J. Data Decryption

Then receiving the metadata $\widehat{m}$, using DU, they can obtain the exact location of the information storage as well as the binary decryption password for the data. Using data analysis and signature, both are contained in the metadata, we can use SSP to retrieve the plaintext of the data and confirm its accuracy and integrity. The data decryption is performed using Eq. (20).

$$Data=decrypt\left(D_{ek},C_t\right) \tag{20}$$

The designed hybrid optimization-based PRE model improves the safety of cloud data by re-encrypting user information and applying a consensus technique to increase security. Algorithm 2 illustrates the optimal threshold PRE procedure.

Algorithm: 2 data encryption and decryption using optimized threshold PRE

```
Start
{
Initialization
{
$\left(1^{\gamma},\overline{u}\right)\to\overline{pp}$        //$1^{\gamma}$ - security parameters
                                //$\overline{pp}$ - public parameter
bilinear map, $m$
four hash functions { $\widehat{H}_0$, $\widehat{H}_1$, $\widehat{H}_2$, $\widehat{H}_3$ }
}
Key generation
{
$\overline{pp}\to\left(\overline{sk}_{do},\overline{pk}_{do},\overline{sk}_{du},\overline{pk}_{du}\right)$
Update HOSBA                     // generate the best optimal key
Initialize the population of owls and bat
Update new position using eqn.9
Generate optimal keys            // public, private, and secrete keys of DO and DU
$\overline{sk}_{do},\overline{pk}_{do},\overline{sk}_{du},\overline{pk}_{du}$
}
Data encryption
{
        Use a secrete key, $D_{ek}$
        Convert plain text into cipher      //AES
        $C_t = encrypt\left(D_{ek}, D\right)$
}
Metadata Encryption
{
Encrypt $\widehat{m}$ using a public key
$\left(\overline{pp}, \overline{pk_{do}}, \widehat{m}\right) \to C_t\widehat{m}$
}
Re-encryption Key Generation
$rk_{do\to du} = \widehat{g}^{v/u}$
        If ( $rh^i_{o\to u} = \widehat{g}^{\bar{f}(x_i)} \bmod p$ )
        {
        Generate re-encryption key
        }
        End if
Re-encryption
{
Update proxy nodes $\widehat{p}_n$
consensus confirmation using proxy nodes
Re-encrypt the data using eqn. (18)
}
Metadata Decryption
{
            Using private key $\overline{sk}_{du}$
            decrypting the ciphertext
            metadata
}
Data Decryption:
{
        Decrypt using metadata $\widehat{m}$, symmetric decryption key
        $D_{ek}$
        $Data = decrypt\left(D_{ek}, C_t\right)$
}
Secure the data
Enhance performance
}
End
```

K. Consensus Mechanism based on Threshold PRE

The process of consensus is an essential element of a developed system. In contrast to the conventional PRE method, the developed approach makes use of a decentralized network to do away with the need for outside central service providers. Re-encrypted ciphertext transformation is the basis for the consensus process that is collaboratively carried out by consensus nodes in blockchain-based systems. The re-encryption key is divided and distributed across all consensus nodes within the blockchain system; the threshold PRE may be seamlessly integrated with the consensus method. The data owners concurrently communicate their demands for ciphertext transformation to the blockchain network and distribute the newly created re-encryption keys among other consensus nodes. A significant amount of processing power is needed for the ciphertext conversion operation. Since not every node in the network is capable of re-encryption, certain nodes with particular processing capabilities respond to this demand by performing re-encryption calculations. These nodes function in the distributed ledger system as consensus nodes. Fig. 2 displays a description of the consensus method based on the established model.

Process of consensus nodes: The consensus node locates the original ciphertext that matches. $C_t\widehat{m}$ in the blockchain ledger, via the re-encryption key after fulfilling the re-encryption conditions $rh^i_{o\to u}$, $i = 1, 2, 3, ...n$ to complete the conversion process. The consensus nodes confirm the ciphertext after it has been re-encrypted, finish the re-encryption calculation within the allotted time C_{v_i} , $i = 1, 2, 3, ...n$ and send it to the blockchain system; Each consensus node compiles and verifies the encrypted text that has been re-encrypted C_{v_i} . A vote for selecting the leader of the network of blockchain servers is started by the consensus node that leads the charge in gathering t authenticated re-encrypted ciphertexts.

Fig. 2 is overview of consensus model with developed technique.

After receiving this request, further consensus nodes verify that the re-encrypted ciphertext is accurate. These re-encrypted ciphertexts are combined by the leader node C_{v_i} to the novel ciphertext $C'_t\,\widehat{m}$. This leader node will also add the newly generated block, the re-encrypted data, and the modified ciphertext to the blockchain database and broadcast it to the whole network. Additional nodes modify the ledger after completing consensus confirmation. After that, the HO-PREB is finished, and the ciphertext for the metadata is changed. $C'_t\,\widehat{m}$ is updated in the block. The blockchain ledger provides users with the altered metadata ciphertext, allowing them to complete the decryption process.

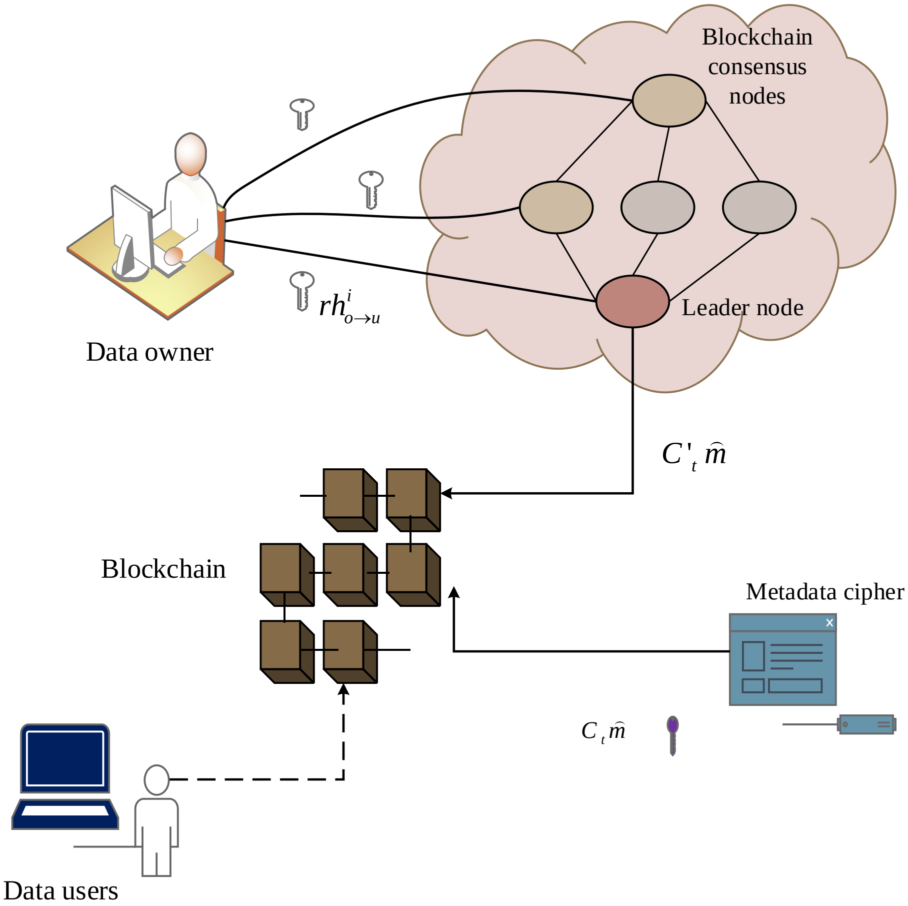

Fig. 2. Overview of consensus model with developed technique.

IV. Result and Discussion

The enhanced efficiency of the created method is verified by comparing it to current methods concerning encryption time, decryption time, re-encryption time, latency, restoration effectiveness, and turnaround time. The created model's robustness and safety are verified using accepted cryptographic approaches after it is implemented using a Python tool. The hybrid owl and bat optimization is employed to improve the user's data's security by generating the optimal key. Finally, it is securely stored in the cloud using blockchain by generating blocks.

A. *Performance Analysis*

To prove the efficiency and reliability of the developed system, the results are validated with existing classifiers such as Advanced Encryption Standard (AES) [36], Data Encryption Standard (DES) [35], Homomorphic Encryption (HE) [37], Rivest-Shamir-Adleman (RSA) [38], and IBE [39]. The performance metrics used for the validation are decryption time, encryption time, latency, re-encryption time, restoration efficiency, and turnaround time.

B. *Encryption Time*

The efficiency of any encryption operation is determined by dividing all the encrypted plaintext by the encryption time (milliseconds).

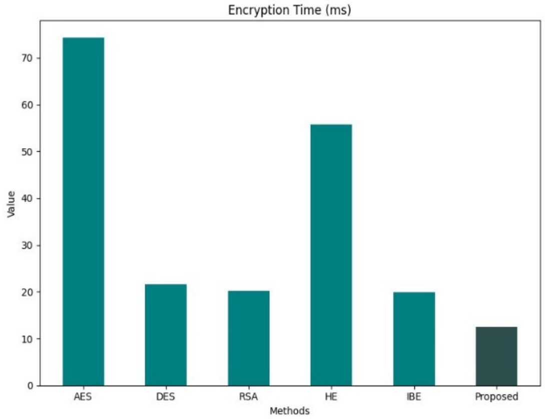

Fig. 3. Encryption time comparison.

Fig. 3 shows the milliseconds (ms) that each cryptographic algorithm takes to encrypt data. The duration of encryption operations is used to evaluate each scheme. AES, DES, RSA, HE, and IBE algorithms are among those examined in the analysis, along with a proposed method. Of the algorithms that have been studied, AES has the longest encryption time—74.25 ms—on record. DES has an encryption time of 21.66 ms, which is shorter than AES's. At 20.28 ms, RSA has a comparatively short encryption time. While IBE displays a comparatively short encryption time of 19.97 ms, HE requires 55.74 ms. With a value of 12.5 ms, the proposed algorithm finally shows the lowest encryption time out of all the algorithms examined.

C. Re-Encryption Time

With PRE, a proxy can change the encryption of a ciphertext from one key to another, encrypting the identical message.

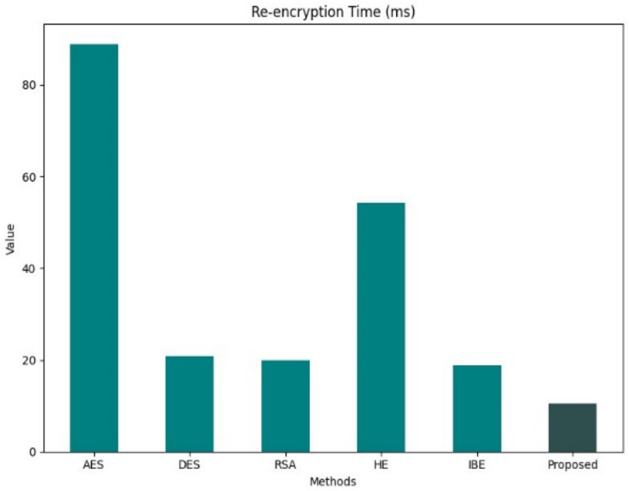

Fig. 4. Re-encryption time comparison.

Performance analyses of the data re-encryption times for the different encryption techniques are shown in Fig. 4. The time it takes to re-encrypt data is indicated by each algorithm's re-encryption time. This is an important factor in cryptographic operations, especially when situations call for regular updates or changes to encryption keys. The highest re-encryption time for the AES algorithm is 88.8 ms, but the DES algorithm displays a duration of 20.83 ms. Furthermore, the RSA and IBE re-encryption times are 19.98 ms and 18.86 ms, respectively. The developed technique has the lowest re-encryption time of all the records, at 10.4 ms, whereas the HE shows a re-encryption time of 54.32 ms. Comparing the suggested algorithm to the other stated algorithms, it may be more efficient in re-encryption activities.

D. Decryption Time

The process of restoring encrypted material to its original form is known as decryption. Usually, the encryption procedure is done in reverse. Because decryption necessitates a secret key, it examines the encrypted data to ensure only an authorized user may decrypt it.

A performance analysis of decryption times using different encryption techniques is shown in Fig. 5. While the suggested algorithm is a novel or modified encryption technique, AES, DES, RSA, HE, and IBE are well-known encryption techniques. DES and RSA are the two classic encryption methods with the fastest decryption times, respectively, at 24.81 and 23.12 milliseconds. Adopted as an encryption standard, AES is relatively slower than DES and RSA, taking 55.58 ms to decrypt. IBE shows a comparatively faster decryption time of 20.99 ms, while HE takes 43.65 ms. With a decryption time of only 8.5 ms, the suggested technique stands out as remarkable. This implies that, in comparison to the current encryption methods, the suggested approach may provide considerable gains in decryption efficiency.

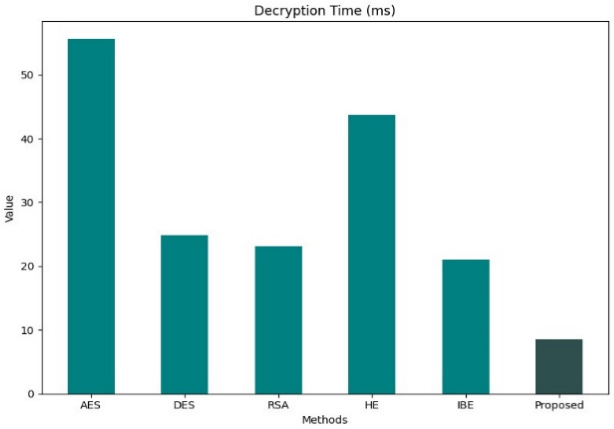

Fig. 5. Decryption time comparison.

E. Latency

The amount of time required for a specific design to finish a specified (computational) task is known as latency. Latency is the amount of time it takes for data to travel across the internet from one location to another. The term "latency" describes how long an algorithm takes to complete a task; shorter latency times correspond to quicker algorithm performance.

Performance analysis of the delay of different encryptions is provided in Fig. 6. With a latency of 47.54 seconds, AES has the highest of all the encryption methods listed, followed by DES, which has a latency of 23.45 seconds. Furthermore, the delay for RSA is 20.67 seconds, while the next highest latency is 32.6 seconds for HE. The suggested encryption technique exhibits the lowest latency of 3.2 seconds, while IBE displays a latency of 14.5 seconds. This suggests that this algorithm performs better than the others.

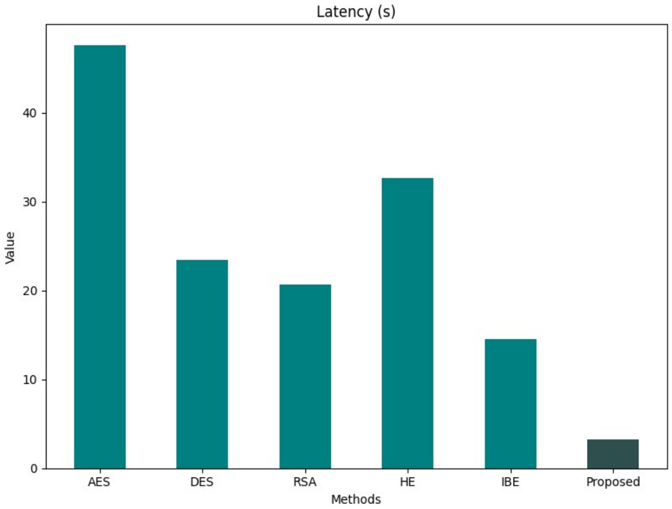

Fig. 6. Latency comparison.

F. *Restoration Efficiency*

The efficacy and speed at which a service or system may be brought back up following a malfunction, outage, or data loss is referred to as restoration efficiency.

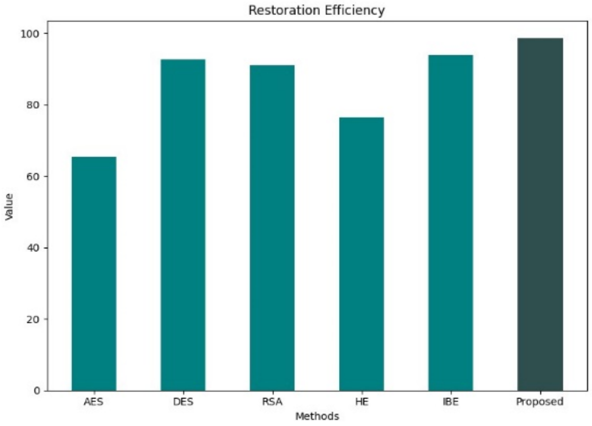

Fig. 7. Restoration efficiency comparison.

Performance analysis scores of restoration efficiency for different cryptographic methods are shown in Fig. 7. Additionally, DES obtained a better restoration efficiency score of 92.56 than AES, which got 65.4. Furthermore, in terms of restoration efficiency, RSA received a score of 91, HE earned 76.43, and IBE received an exceptionally high score of 93.9. With a restoration efficiency score of 98.5, the suggested method excels and demonstrates remarkable efficacy in quickly recovering systems or data protected by the technique.

G. *Turnaround Time*

The term "turnaround time" describes the length of time required to complete a process or task from beginning to end. Turnaround time, as used in the table, particularly refers to the milliseconds (ms) that each algorithm for cryptography requires to complete a given operation or computation.

The turnaround times, expressed in milliseconds (ms), for several cryptographic algorithms—AES, DES, RSA, HE, IBE, and a suggested algorithm—are shown in Fig. 8. With a turnaround time of 300 milliseconds, AES has the longest algorithmic delay of all the listed algorithms. With a turnaround time of 112 milliseconds, DES comes next. The turnaround time of the popular public-key encryption technique RSA is 95.3 milliseconds. The turnaround time for HE is 195 milliseconds, whereas the turnaround time for IBE is 92.54 milliseconds. It's interesting to note that, at 45 milliseconds, the suggested approach has the quickest turnaround time out of all those mentioned. This suggests a possible breakthrough in the efficiency of cryptography, providing much faster processing while upholding the essential security requirements.

H. *Discussion*

Existing cloud-based Key Management Services (e.g., AWS KMS, Azure Key Vault, and Google Cloud KMS) can be integrated with the HO-PREB system to manage encryption keys securely. The hybrid optimization algorithms (Owl and Bat optimization) used for key selection can interact with these services to dynamically generate and manage encryption keys. Encrypted data can be stored in cloud-based storage services such as Amazon S3, Azure Blob Storage, or Google Cloud Storage. The proxy nodes will manage access to the encrypted data, ensuring that only authorized entities can decrypt and access the data. However, issues such as computational overhead, integration with existing infrastructure, and compatibility with various cloud service models could be addressed to provide a more holistic view. While the blockchain model adds scalability, further details on how the system handles large-scale data and high-frequency IoT transactions would enhance understanding. This includes the impact on network bandwidth and storage requirements as the number of transactions and data size increase.

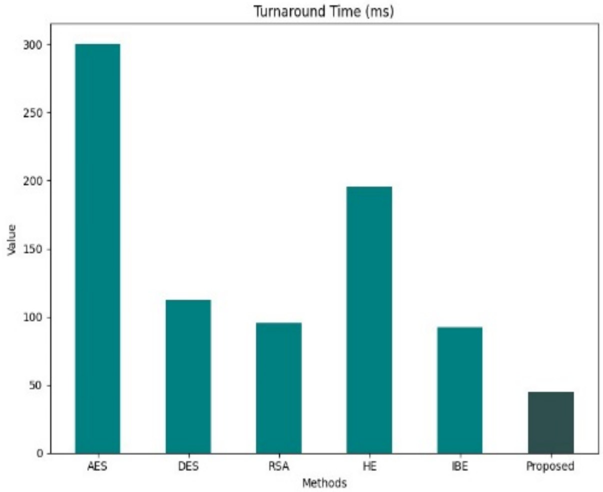

Fig. 8. Turnaround time comparison.

V. CONCLUSION

The hybrid optimization and threshold-based PRE system with consensus system, which is an innovative design that offers real-world applications for the safe sharing of IoT data, is proposed in this paper. It enables DO to effectively safeguard their encrypted data via SSP and share it with authorized users. The proposed solution can eliminate dependency on external central proxy servers by utilizing a distributed blockchain network and combining an optimized threshold PRE system with a blockchain consensus algorithm. The re-encryption of encrypted data can be computed during the consensus confirmation process of the distributed ledger network. Simulation studies showed that optimized threshold PRE can be successfully paired with the blockchain's consensus process to enhance cloud computing security and facilitate data exchange. In addition, when compared to other methods, the performance and scalability of the suggested system are adequate. The developed model attains encryption, re-encryption, and decryption times of 12.5 ms, 10.4 ms, and 8.5 ms. Also, gained less latency and a high restoration efficiency of 3.2s and 98.5%. The developed technique attains better performance to improve the cloud system's security, and it can detect and neglect attacks based on the proxy nodes in the consensus system. Also, there is potential to explore the integration of state-of-the-art cryptographic and machine learning methods to further enhance the security and efficacy of data exchange protocols.

ACKNOWLEDGMENT

The author extends the appreciation to the Deanship of Postgraduate Studies and Scientific Research at Majmaah University for funding this research work through the project number (R-2024-1241)

REFERENCES

[1] Awadallah, R. and Samsudin, A., 2021. Using blockchain in cloud computing to enhance relational database security. IEEE Access, 9, pp.137353-137366.

[2] Awadallah, R., Samsudin, A., Teh, J.S. and Almazrooie, M., 2021. An integrated architecture for maintaining security in cloud computing based on blockchain. IEEE Access, 9, pp.69513-69526.

[3] Shrivastava, P., Alam, B. and Alam, M., 2024. A hybrid lightweight blockchain based encryption scheme for security enhancement in cloud computing. Multimedia Tools and Applications, 83(1), pp.2683-2702.

[4] Velmurugadass, P., Dhanasekaran, S., Anand, S.S. and Vasudevan, V., 2021. Enhancing Blockchain security in cloud computing with IoT environment using ECIES and cryptography hash algorithm. Materials Today: Proceedings, 37, pp.2653-2659.

[5] Murthy, C.V.B., Shri, M.L., Kadry, S. and Lim, S., 2020. Blockchain based cloud computing: Architecture and research challenges. IEEE access, 8, pp.205190-205205.

[6] Zhang, H., Zang, Z. and Muthu, B., 2022. Knowledge-based systems for blockchain-based cognitive cloud computing model for security purposes. International Journal of Modeling, Simulation, and Scientific Computing, 13(04), p.2241002.

[7] Benil, T. and Jasper, J.J.C.N., 2020. Cloud based security on outsourcing using blockchain in E-health systems. Computer Networks, 178, p.107344.

[8] Gong, J. and Navimipour, N.J., 2022. An in-depth and systematic literature review on the blockchain-based approaches for cloud computing. Cluster Computing, 25(1), pp.383-400.

[9] Zou, J., He, D., Zeadally, S., Kumar, N., Wang, H. and Choo, K.R., 2021. Integrated blockchain and cloud computing systems: A systematic survey, solutions, and challenges. ACM Computing Surveys (CSUR), 54(8), pp.1-36.

[10] Wilczyński, A. and Kołodziej, J., 2020. Modelling and simulation of security-aware task scheduling in cloud computing based on Blockchain technology. Simulation Modelling Practice and Theory, 99, p.102038.

[11] Rahman, A., Islam, M.J., Band, S.S., Muhammad, G., Hasan, K. and Tiwari, P., 2023. Towards a blockchain-SDN-based secure architecture for cloud computing in smart industrial IoT. Digital Communications and Networks, 9(2), pp.411-421.

[12] Uddin, M., Khalique, A., Jumani, A.K., Ullah, S.S. and Hussain, S., 2021. Next-generation blockchain-enabled virtualized cloud security solutions: review and open challenges. Electronics, 10(20), p.2493.

[13] Bonnah, E. and Shiguang, J., 2020. DecChain: A decentralized security approach in Edge Computing based on Blockchain. Future Generation Computer Systems, 113, pp.363-379.

[14] Ali, A., Khan, A., Ahmed, M. and Jeon, G., 2022. BCALS: Blockchain-based secure log management system for cloud computing. Transactions on Emerging Telecommunications Technologies, 33(4), p.e4272.

[15] Wei, P., Wang, D., Zhao, Y., Tyagi, S.K.S. and Kumar, N., 2020. Blockchain data-based cloud data integrity protection mechanism. Future Generation Computer Systems, 102, pp.902-911.

[16] Sowmiya, B., Poovammal, E., Ramana, K., Singh, S. and Yoon, B., 2021. Linear elliptical curve digital signature (LECDS) with blockchain approach for enhanced security on cloud server. IEEE Access, 9, pp.138245-138253.

[17] Xie, G., Liu, Y., Xin, G. and Yang, Q., 2021. Blockchain-based cloud data integrity verification scheme with high efficiency. Security and Communication Networks, 2021, pp.1-15.

[18] Abirami, P. and Bhanu, S.V., 2020. Enhancing cloud security using crypto-deep neural network for privacy preservation in trusted environment. Soft Computing, 24(24), pp.18927-18936.

[19] Egala, B.S., Pradhan, A.K., Badarla, V. and Mohanty, S.P., 2021. Fortified-chain: a blockchain-based framework for security and privacy-assured internet of medical things with effective access control. IEEE Internet of Things Journal, 8(14), pp.11717-11731.

[20] Zhang, G., Yang, Z., Xie, H. and Liu, W., 2021. A secure authorized deduplication scheme for cloud data based on blockchain. Information Processing & Management, 58(3), p.102510.

[21] Keshta, I., Aoudni, Y., Sandhu, M., Singh, A., Xalikovich, P.A., Rizwan, A., Soni, M. and Lalar, S., 2023. Blockchain aware proxy re-encryption algorithm-based data sharing scheme. Physical Communication, 58, p.102048.

[22] Agyekum, K.O.B.O., Xia, Q., Sifah, E.B., Cobblah, C.N.A., Xia, H. and Gao, J., 2021. A proxy re-encryption approach to secure data sharing in the internet of things based on blockchain. IEEE Systems Journal, 16(1), pp.1685-1696.

[23] Rawal, B.S., Manogaran, G. and Hamdi, M., 2021. Multi-tier stack of block chain with proxy re-encryption method scheme on the internet of things platform. ACM Transactions on Internet Technology (TOIT), 22(2), pp.1-20.

[24] Manzoor, A., Braeken, A., Kanhere, S.S., Ylianttila, M. and Liyanage, M., 2021. Proxy re-encryption enabled secure and anonymous IoT data sharing platform based on blockchain. Journal of Network and Computer Applications, 176, p.102917.

[25] Chen, Y., Hu, B., Yu, H., Duan, Z. and Huang, J., 2021. A threshold proxy re-encryption scheme for secure IoT data sharing based on blockchain. Electronics, 10(19), p.2359.

[26] Feng, T., Wang, D. and Gong, R., 2023. A blockchain-based efficient and verifiable attribute-based proxy re-encryption cloud sharing scheme. Information, 14(5), p.281.

[27] Pei, H., Yang, P., Li, W., Du, M. and Hu, Z., 2024. Proxy re-encryption for secure data sharing with blockchain in Internet of Medical Things. Computer Networks, p.110373.

[28] Simaiya, S., Lilhore, U.K., Sharma, S.K., Gupta, K. and Baggan, V., 2020. Blockchain: A new technology to enhance data security and privacy in Internet of things. Journal of Computational and Theoretical Nanoscience, 17(6), pp.2552-2556.

[29] Poongodi, J., Kavitha, K. and Sathish, S., 2022. Healthcare Internet of Things (HIoT) data security enhancement using blockchain technology. Journal of Intelligent & Fuzzy Systems, 43(4), pp.5063-5073.

[30] Kollu, P.K., 2021. Blockchain techniques for secure storage of data in cloud environment. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(11), pp.1515-1522.

[31] Liu, S.G., Chen, W.Q. and Liu, J.L., 2021. An efficient double parameter elliptic curve digital signature algorithm for blockchain. IEEE Access, 9, pp.77058-77066.

[32] Khashan, O.A., 2020. Hybrid lightweight proxy re-encryption scheme for secure fog-to-things environment. IEEE Access, 8, pp.66878-66887.

[33] Yang, X.S. and Hossein Gandomi, A., 2012. Bat algorithm: a novel approach for global engineering optimization. Engineering computations, 29(5), pp.464-483.

[34] Jain, M., Maurya, S., Rani, A. and Singh, V., 2018. Owl search algorithm: a novel nature-inspired heuristic paradigm for global optimization. Journal of Intelligent & Fuzzy Systems, 34(3), pp.1573-1582.

[35] Vuppala, A., Roshan, R.S., Nawaz, S. and Ravindra, J.V.R., 2020. An efficient optimization and secured triple data encryption standard using enhanced key scheduling algorithm. Procedia Computer Science, 171, pp.1054-1063.

[36] Altigani, A., Hasan, S., Barry, B., Naserelden, S., Elsadig, M.A. and Elshoush, H.T., 2021. A polymorphic advanced encryption standard–a novel approach. IEEE Access, 9, pp.20191-20207.

[37] Bozduman, H.Ç. and Afacan, E., 2020. Simulation of a homomorphic encryption system. Applied Mathematics and Nonlinear Sciences, 5(1), pp.479-484.

[38] Almuzaini, K.K., Dubey, R., Gandhi, C., Taram, M., Soni, A., Sharma, S., Sánchez-Chero, M. and Carrión-Barco, G., 2023. Secured wireless sensor networks using hybrid Rivest Shamir Adleman with ant lion optimization algorithm. Wireless Networks, pp.1-19.

[39] Deng, H., Qin, Z., Wu, Q., Guan, Z., Deng, R.H., Wang, Y. and Zhou, Y., 2020. Identity-based encryption transformation for flexible sharing of encrypted data in public cloud. IEEE Transactions on Information Forensics and Security, 15, pp.3168-3180.