International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2022/214502 Volume 9, Issue 4, July – August (2022)

RESEARCH ARTICLE

Design a hybrid Optimization and Homomorphic Encryption for Securing Data in a Cloud Environment

Mercy Joseph
Department of Computer Science, Chikkanna Government Arts College, Tirupur, Tamil Nadu, India
elsinchakkala@gmail.com

Gobi Mohan
Department of Computer Science, Chikkanna Government Arts College, Tirupur, Tamil Nadu, India
mgobimail@yahoo.com

Received: 27 May 2022 / Revised: 26 June 2022 / Accepted: 01 July 2022 / Published: 30 August 2022

Abstract – Cloud Computing (CC) is denoted as web-based computing that offers devices or users a shared pool of information, resources, or software. It permits small companies and end-users for making the use of different computational resources such as software, storage, and processing ability offered via other companies. But the main problem in CC is data security because of malware and attacks. So this paper developed a novel Hybrid Bat and Cuckoo-based Pallier Homomorphic Encryption (HBC-PHE) scheme for enhancing the data security of the cloud from malware and attacks. Initially, collected datasets are stored in the cloud using the python tool, and collected datasets are transferred into the developed HBC-PHE framework. At first, generate the key for each dataset and separate the private key for all datasets. Moreover, convert the plain text into ciphertext using the bat and cuckoo fitness function in PHE. Finally, cloud-stored data are encrypted successfully and the attained performance outcomes of the developed framework are associated with other existing techniques in terms of confidential rate, decryption time, encryption time, efficiency, and throughput. Additionally, the developed model gained a throughput of 654Kbps, decryption time of 0.05ms, encryption time of 0.08ms, and efficiency of 98.34% for 500kb. As well, the designed model gained a confidential rate of 98.7% and a computation time of 0.03s for using a 500 kb.

Index Terms – Homomorphic Encryption, Secrete Key, Cloud Computing, Data Security, Attacks, Malware, Plain Text, Ciphertext.

1. INTRODUCTION

In recent years, CC becomes the most important technology because of the rapid growth of big data methodologies and the internet [1]. Moreover, CC provides more opportunities related to on-demand and trading solutions that are useful for consuming services through the cloud [2]. The cloud infrastructure achieves various activities such as business development, service organization, and data maintenance [3]. As well as it deals with different kinds of cloud services to each other. Many researchers have developed encryption techniques to secure the cloud data from attackers and third parties [4, 5]. Generally, encryption is the method of transforming original format plain text into an unreadable format like ciphertext which is transferred and stored in the cloud [6]. Consequently, encryption is the most real component which addresses significant security issues such as compliance, unauthorized access, and so on [7]. Consequently, the basic process of privacy and cloud security is exposed in Figure 1.

Furthermore, encryption leverage is the advanced algorithm for encoding the data which makes it meaningless to any users who do not have the key [8]. Also, authorized user leverage is the key to decoding the data which transfers concealed data back into the readable format. Thus the generated keys are transferred and shared only with the trusted parties and data owners [9, 10]. The encryption algorithm contains two types such as symmetric encryption and asymmetric encryption [11].

Consequently, symmetric encryption contains the same encryption and decryption key but asymmetric encryption contains two keys such as a private and public key to authenticate the user data [12]. As well, the advantage of using cloud encryption is integrity, compliance, security, and reduced risk. But the most challenging task of data encryption in the cloud is key management, data loss, time, and cost [13]. The basic ingredient to encrypting the data is security which is the common security measure when comparing password systems [14]. The cloud server gathers the different data and certainly gathers a massive amount of sensitive information about the users that sensitive information harms the security and privacy of the users [15]. So, the protection of private information is essential in a cloud environment. Since the privacy of the data is receiving more attention from foreign and domestic experts [16, 17]. The service of the cloud database is the most attractive solution to handle the large quantity of the data of their users [18]. Consequently, privacy concern is more essential before uploading the private data to an untrusted server [19].

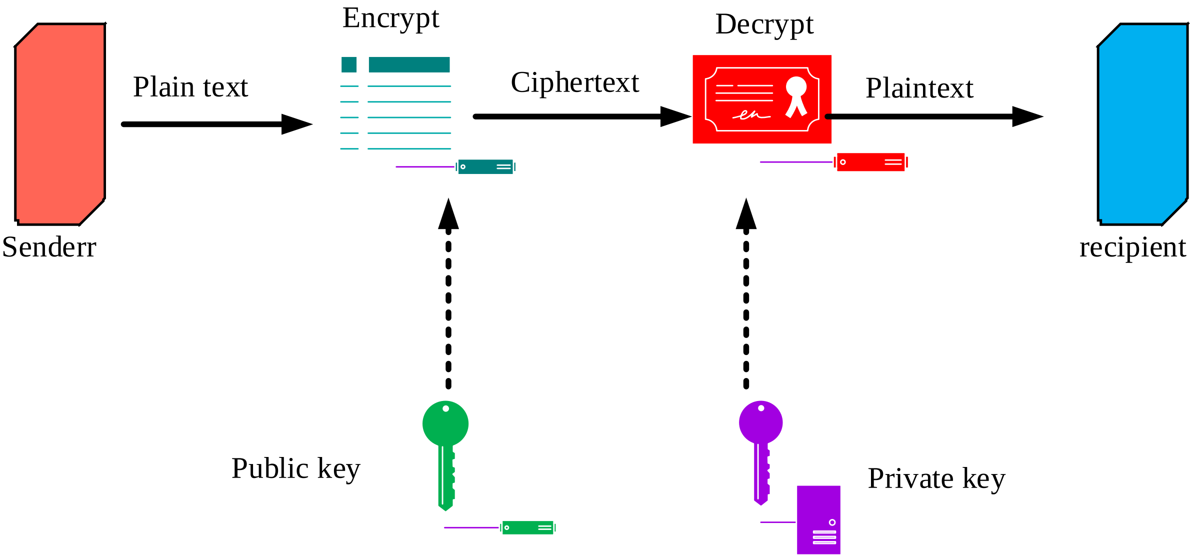

Figure 1 Cloud Privacy and Security

The main problem in CC is a lack of security and privacy which tends to hack the user information [20]. So designing a secure cloud environment is essential to secure our data from attackers and third parties. There are many researchers are designed to attain better security such as new video encryption model [21], privacy-based associate rule mining technique [22], authenticated key management technique [24], etc but still have the issues of large encryption and decryption time, error, noise, and attack. The key contribution of this research work is detailed as follows

- Initially, the cloud user dataset was collected and trained in the system using the Python tool.
- Hereafter, a new HBC-PHE is planned with appropriate parameters for securing the cloud data from attacks and third parties.
- Moreover, the decryption and decryption process is processed using a secure key.
- Subsequently, to check the strength of the designed model, a harmful attack had been launched in a cloud environment.
- At last, performance metrics of the developed model were measured and compared with prevailing models for efficiency, throughput, decryption time, confidentiality, and encryption time.

The organization of the research paper is summarized as follows: related works of cloud security are discussed in sector 2 and sector 3 details the system model and problem definition. Similarly, sector 4 elaborated on the process of the planned methodology, and sector 5 described results and discussions. Finally, the finishing summary of the designed model is discussed in sector 6.

2. RELATED WORKS

Some literature surveys based on cloud data security are detailed below,

Geetha et al [21] proposed a new video encryption model for enhancing the security of the transmission video. Moreover, the homomorphic encryption technique is developed to increase the speed of the encryption process. Thus the designed model is useful for reducing computational complexity and cloud user communication. Moreover, the developed framework is secure and highly efficient but the error rate is high when comparing other techniques.

Day by day, the development of data analytics, CC, and big data plays a significant role in producing huge market values. Hongping and Baocang [22] developed a privacy-based associate rule mining technique to encrypt the data using homomorphic encryption. The designed technique supports multiple cloud users through altered public keys. However, it has the problem of lack of data security.

Byung et al [23] proposed ring learning by an error-based communication protocol scheme to authenticate and manage the message in a cloud environment. Moreover, the designed communication protocol is used for security and safety by conducting a performance analysis of the IoT environment. Furthermore, the developed technique offers strong security and an equivalent level of efficiency. The designed model offers safe communication from user authentication to transferring the data to the users.

In the big data environment, data security is the main issue, also the distribution of keys, management, and transfer of server users are the most critical problem. Algaradi et al [24] proposed authenticated key management technique for enhancing the two-level securities of cloud users. The first process is user communication with the server and the second process is data encryption. The gained performance is compared with other existing techniques that also protect user data but the delay is high.

Nir Drucker and Shay Gueron [25] developed an encryption-based trusted execution technique for guaranteeing the correctness and integrity of database code. Moreover, homomorphic encryption is designed to encrypt the data securely. Also, explain the combined technique for the easy use of multi-party computations. Finally, construct the voting system which influences the capability of the designed model but it lacks data security. The summary of the literature survey is detailed in table 1.

Table 1 Summary of Recent Literature Survey

Author	Method	Advantage	Disadvantage
Geetha et al [21]	Novel video encryption model	• Enhance the security of the transmission video • Increase the speed of the encryption process	• The error rate is high • Less efficiency
Hongping and Baocang [22]	Privacy-based associate rule mining technique	• Reduce computational complexity • Encrypt the data using a public key	• Lack of data security • Low robustness
Byung et al [23]	Ring learning by error-based communication protocol scheme	• Authenticate and manage the message • Provide strong security and efficiency	• High complexity • High encryption time
Algaradi et al [24]	Authenticated key management technique	• Protect user data • High security	• The delay rate is high • Less efficiency
Nir Drucker and Shay Gueron [25]	Encryption-based trusted execution technique	• Guaranty the correctness and integrity • Easy use of multi-party computations	• Lack of data security • Less confidential rate

3. SYSTEM MODEL AND PROBLEM DEFINITION

Security issues are the biggest issues for developing CC; encryption is the central technology for ensuring CC data security. Moreover, security infrastructure is essential to safeguard cloud services and the web. The basic system model is shown in Figure 2. It contains privacy protection, data processing, and ciphertext retrieval. Furthermore, privacy protection transmits the user data and is stored in the cloud through encryption. While CC offers and handles easy information of plaintext. For the encryption used Fully Harmonic Encryption (FHE) which enables the users or third party for converting ciphertext. Finally, cipher data are retrieved by direct searching of ciphertext which improves efficiency and privacy. Then the retrieval data was successfully added with the corresponding plain text.

However, the main problem behind the CC techniques is security threats because of errors, system flaws, internal controls, and problems in encryption. That leads to cause the digital records and attacks vulnerability. Hence, the threat is something that can recognize the vulnerability and interrupt the sources against the authenticated users. Furthermore, less accuracy, less encryption time, attacks, and error are the most common problem in CC. To overcome these types of issues, this research has designed a security model in the cloud environment.

4. PROPOSED METHODOLOGY

Design a novel Hybrid Bat and Cuckoo-based Pallier Homomorphic Encryption (HBC-PHE) Scheme for enhancing the data security of the cloud from malware and attacks. The various user dataset is collected from the net source and stored in the cloud. Then the collected dataset is updated to the developed HBC-PHE framework for securing the data from third parties ana attacks. Furthermore, generate the key for each dataset and they are separated into private keys to enhance the security. Moreover, convert the plain text into ciphertext using the bat and cuckoo fitness function in PHE.

In encryption, plain text is converted into cipher text, and in decryption, the ciphertext is converted into plain text. The architecture of the proposed technique is shown in Figure 3.

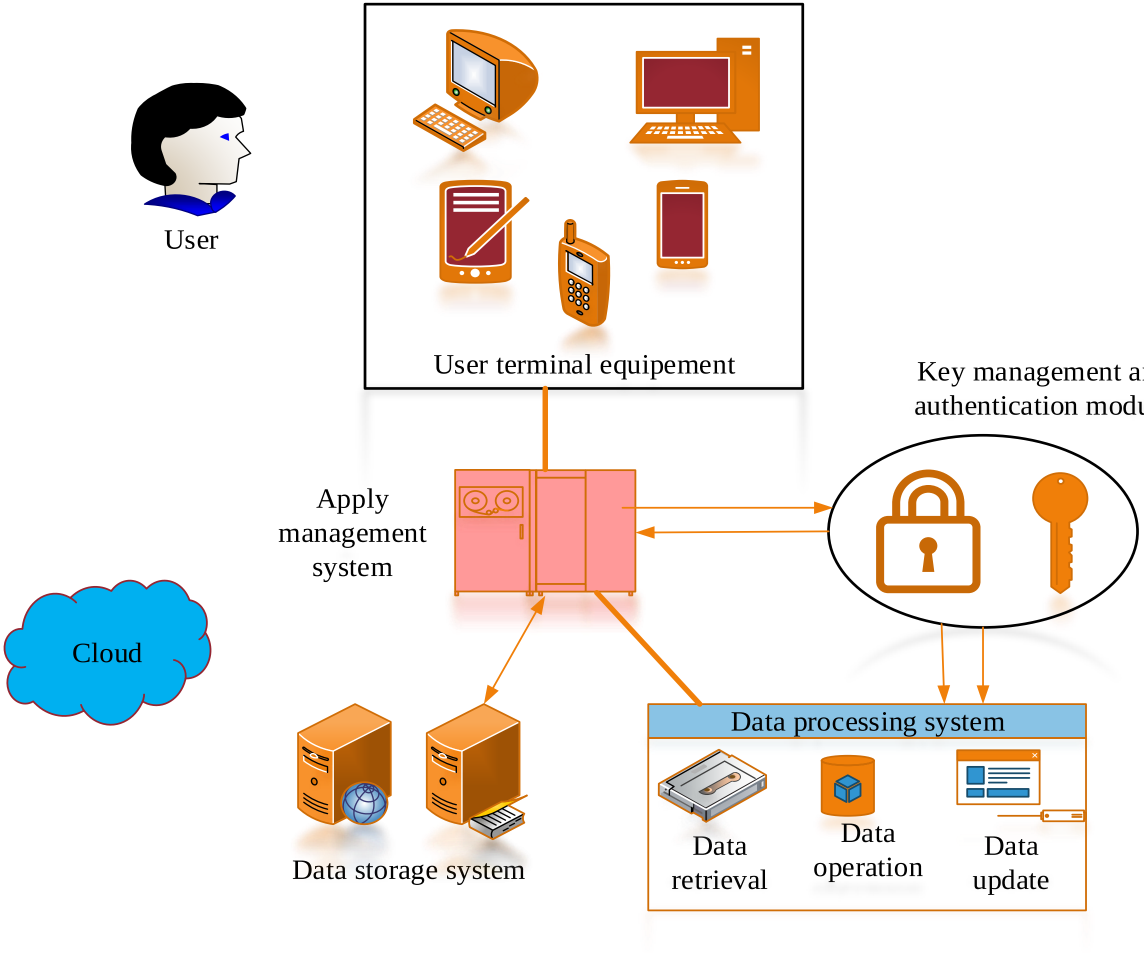

Figure 2 System Model and Problem Definition

Generally, the developed technique contains two entities such as data owner (Dos) and users. Moreover, the data owner is represented as one who stores the normal data or confidential files on the database of the cloud server. Furthermore, users are the parties or authorized entities that need the stored information of file content or any service from the cloud. Furthermore, IoT-based datasets are collected and trained in the system. The total dataset used for the developed framework is 916. From that training used 733 datasets and testing used 183 datasets. More than 70% of data are used for training and 30% of data are used for training. Thus the developed framework is trained and tested using homomorphic encryption which encrypts the data to improve the security in a cloud environment.

4.1. Processes of HBC-PHE

Initially, collected datasets are stored in the cloud they are updated to the designed model for converting the plain text into ciphertext. It secures the data from third parties and attacks. Afterward, the key is generated based on the user details and then convert the plain text into ciphertext using bat optimization which is the heuristic algorithm for identifying the prey using their echolocation and guiding the microbats based on their foraging behaviour. It senses the difference between food or prey and background barriers by echolocation. The purpose of using bat fitness is accurately converted the plain text into ciphertext which is stored in the secrete key. For the decryption process, cuckoo optimization is utilized for the random decryption which covert the ciphertext into plain text. The purpose of cuckoo optimization is to identify the good habitat by laying eggs that are used to identify the user and attacks present in the network.

4.1.1. Pallier Homomorphic Cryptosystem (PHC)

The PHC technique is executed with the support of addition. Where $h = as$ is represented as modulus through two equal sizes of the prime factors a and s. Also, $W \in Z$ and $F_1, F_2 \in Z_n^*$. Then the basic function of pallier encryption and decryption is expressed in Eqn. (1) and Eqn. (2).

$$D_e\left(e_n\left(F_1\right), e_n\left(F_2\right)\right)....\left(m_o h^2\right) = F_1 + F_1....\left(m_o\left(h\right)\right) \quad (1)$$

$$D_e\left(e_n\left(F_1\right)^g\right)....\left(m_o h^2\right) = F_1.g.....\left(m_o\left(h\right)\right) \quad (2)$$

Moreover, encryption and pubic key are represented as $e_n\left(F_1\right), e_n\left(F_2\right)$ the message F_1 and F_2 respectively. Furthermore, $e_n\left(F_1 + F_2\right)$ is represented as the possible computation with $e_n\left(g.F_1\right)$ of constant g.

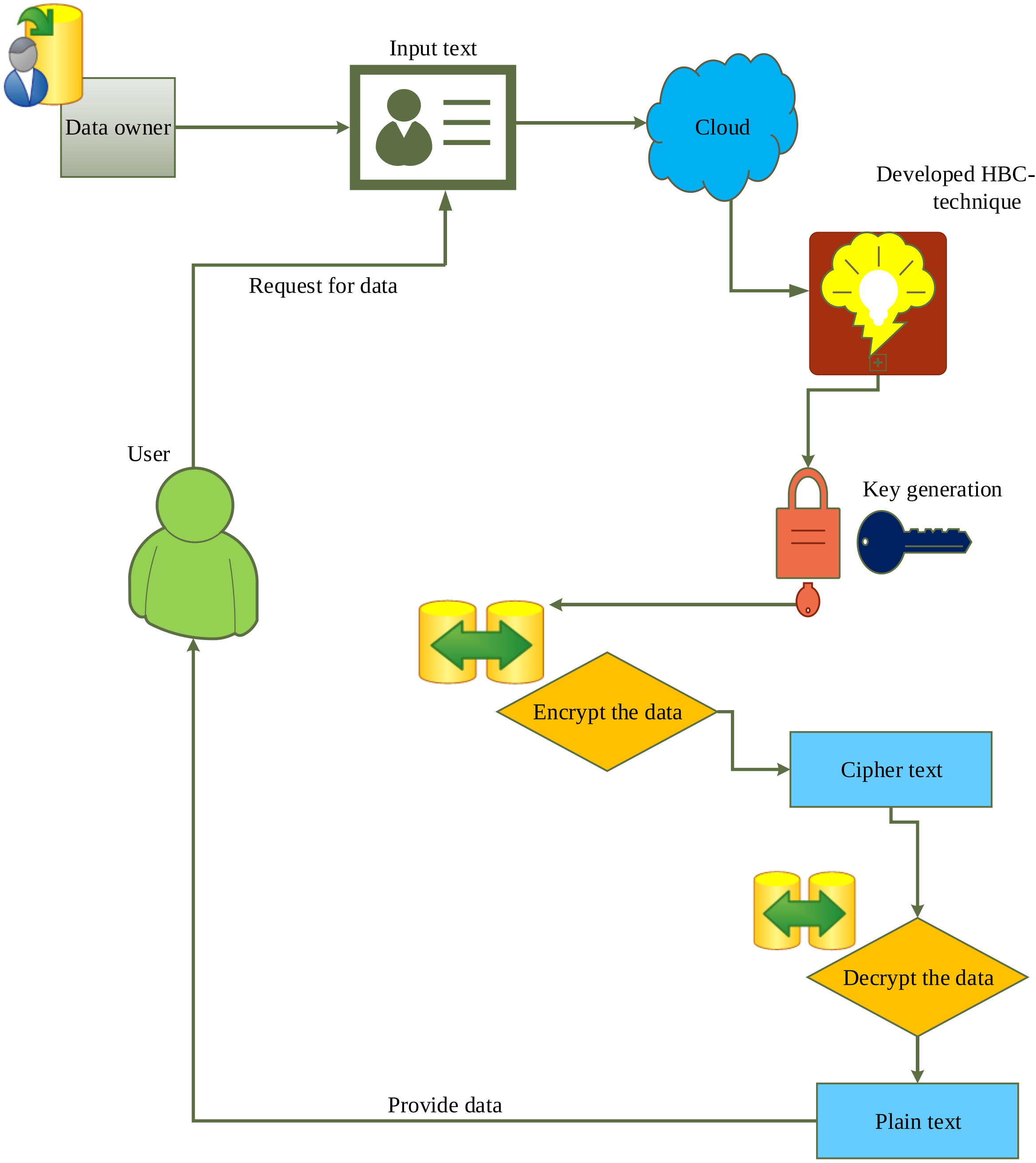

Figure 3 Proposed Methodology

4.1.2. Key Generation

In this stage, Dos verify the received data of user details which gathers the corresponding secret details such as an address, user name, ID, MAC address, and so on. Also, it asks the end-user to generate a suitable password for data. Furthermore, Dos establish the collected information in a sequential manner which converts the original information into binary numbers. The key generation is obtained using Eqn. (3)

$$\varphi(n) = (k-1)(s-1) \qquad (3)$$

Let, $\varphi(n)$ is denoted as the key generation, k is represented as user details key and s is denoted as password. Moreover, generated key is tested using Eqn. (4)

$$\delta(n) = p_m K(s)(k-1, s-1) \qquad (4)$$

Where, p_m is denoted as the pallier function and $K(s)$ is represented as plain text.

4.1.3. Data Encryption

The plaintext of the data is divided into several blocks with 256 bits each also PHC maintains the property of additive homomorphism. It includes the product of two ciphertexts which are selected by random numbers, the conversion of ciphertext is processed using Eqn. (5)

$$D_e = j_s P_q * m_o(h) \frac{p_m(A)K(s)}{R|n| \times b_s(t)} \qquad (5)$$

Let, j_s is denoted as public keys, P_q is considered as input data block, $m_o(h)$ is represented as ciphertext, and $b_s(t)$ is denoted as bat fitness function. Consequently, user information is secured and converted into cipher text which is stored in the cloud. The process of the designed model is elaborated in algorithm 1.

```
Start
{
Designed HBC-PHE
Initialize input data, key $k$, and password $s$
// $P_q$ is called as input data block
//256 bits
Key generation
// Generate a suitable password for data
For all $\varphi(n)$= 1
{
Generate key and password
}
End for
Send to designed model
// for securing the data from hackers
Data encryption          // update bat fitness
// public key to encrypt the data
Step 1
{
plain text
$\{010,110,001,101,.....\}$
// input
}
Step 2
{
Plain text is separated into blocks
$\{01\},\{10\},\{11\},\{01\}$
}
Step 3
{
Converted into ciphertext
$\{10\},\{01\},\{00\},\{11\}$
// encrypted output
}
Ciphertext
Data decryption          //update cuckoo fitness
//retrieve the input data using secrete key
{
Convert cipher text into plain text
$\{10\},\{01\},\{00\},\{11\}$-> $\{010,110,001,101,.....\}$
Recovered plain text
}
Output
}
End
```

Algorithm 1 HBC-PHE Scheme for Securing Data in the Cloud

4.1.4. Data Decryption

During the decryption update the cuckoo fitness for retrieving the data using a correct secret key. The ciphertext of the data is converted into plain text by maintaining the property of multiplicative homomorphism. Moreover, decryption is obtained using Eqn. (6)

$$D_{de} = S_k(t)P_q * K(s)\frac{p_m(M)}{C_u(t)} \qquad (6)$$

Let, $C_u(t)$ is denoted as fitness cuckoo fitness. In the decryption, ciphertext $m_o(h)$ with a secret key $S_k(t)$ is used to retrieve the information and covert the ciphertext into plain text.

5. RESULTS AND DISCUSSIONS

In this section, for securing data proposed an HBC-PHE model and it is implemented in the python tool. Initially, more than 256 bits of data are transferred into the cloud then the bits are separated into a block by 64 bits. Thus the developed framework classifies the encrypted message, decrypted message, and user data. Finally, convert the plain text into ciphertext using a generated secret key. Frequently, the developed HBC-PHE mechanism secures the data from malicious activity and unauthorized access by encrypted data also the gained performance metrics are compared with other existing techniques for checking the efficiency of the developed technique. For the simulation used a cloud simulator that is useful for several cloud applications through virtual machines, creating data centers, and other utilities that are activated based on the cloud research. The developed model simulation parameters and their types are detailed in table 2.

Table 2 Simulation Parameters

Simulation parameter	Type
Simulation area	100m x 100m
Encryption and Decryption Algorithm	PHE
Cloud type	Cloud storage
Memory usage (RAM)	2.00GB
Cloud storage	Google drive
Simulation time	200s
Data packet size	256 Bits
Encryption key size	64 Bits

5.1. Performance Metrics

The performance metrics of the planned HBC-PHE technique were calculated by calculating the key metrics of existing methods. Therefore, Efficient FHE to Secure Video (EFHE-SV) [21], Privacy Association Rule Mining (PARM) [22], Secure Communication Protocol (SCP) [23], DNA-Based Encryption in CC environment (DNAE) [26], DNA Computing (DNAC) [27], and Integrating Encryption (IE) techniques were associated with the developed technique in terms of encryption time, decryption time, delay, and throughput, etc.

5.1.1. Encryption Time (ET)

ET is well-defined by the overall time reserved for converting the Plain Text (PT) into Cipher Text (CT). Therefore, the gained ET of the developed technique is associated with other models like EFHE-SV, PARM, DNAE, SCP, DNAC, and IE are exposed in table 3. At this point, the designed HBC-PHE model has reached 0.08 ms for 500kb. Accordingly, PARM and SCP methods have attained 1.19 ms, and 3.2 ms respectively, EFHE-SV has obtained 1.8 ms, then the DNAC technique achieved 4.5 ms for 500kb. Finally, DNAE and IE techniques achieved 3 ms and 0.35 ms for 500 kb as established in Figure 4.

Table 3 ET Comparison

File size (kb)	Encryption time (ms)						
	EFHE-SV	PARM	DNAE	SCP	DNAC	IE	Proposed
500	1.8	1.19	3	3.2	4.5	0.35	0.08
1000	1.15	1.29	3.9	3.11	4.12	0.41	0.10
1500	1.24	1.37	3.15	3.23	4.17	0.45	0.12
2000	1.33	1.42	3.21	3.31	4.21	0.51	0.14
2500	1.40	1.48	3.25	3.39	4.28	0.57	0.16

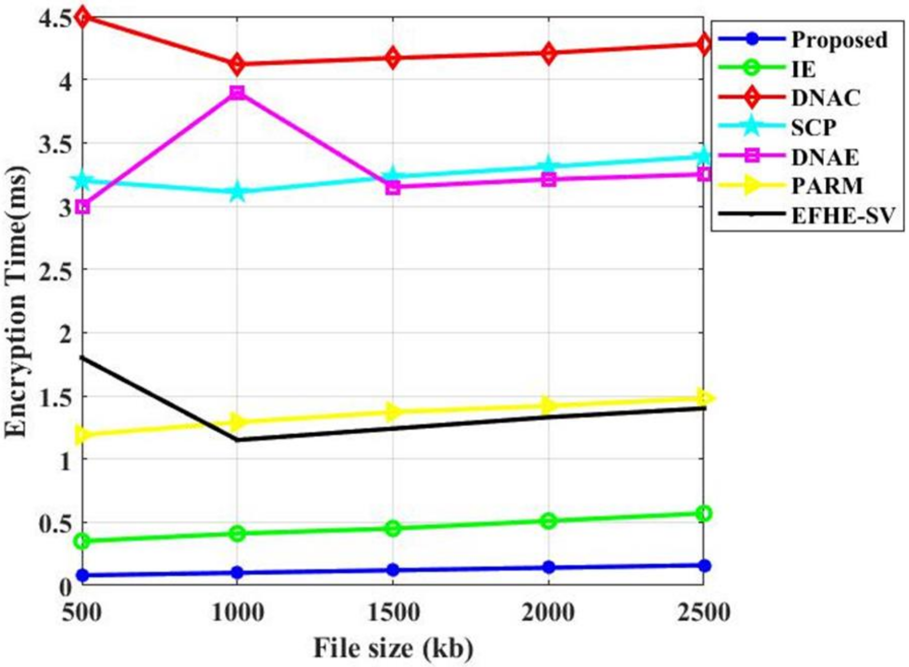

Figure 4 Comparison of ET

5.1.2. Decryption Time (DT)

DT is well-defined as the overall time engaged for converting the CT into PT. It is the opposite performance of the encryption process which is called data DT. Furthermore, the DT of the developed HBC-PHE model was associated with prevailing techniques like EFHE-SV, PARM, DNAE, SCP, DNAC, and IE. Additionally, the gained DT of the developed model graph is demonstrated in Figure 5.

The EFHE-SV technique proceeds 0.05 ms, the PARM model attained 1.76 ms, the DNAE technique attained 5 ms, the SCP model gained 4.1 ms, the IE replica has taken 0.59 ms then the DNAC model gained 3 ms for 500 kb. Nonetheless, the planned HBC-PHE technique gained 0.05 ms and the comparison of DT indicates the developed model's finest result which obtained less DT while comparing other models that are described in table 4.

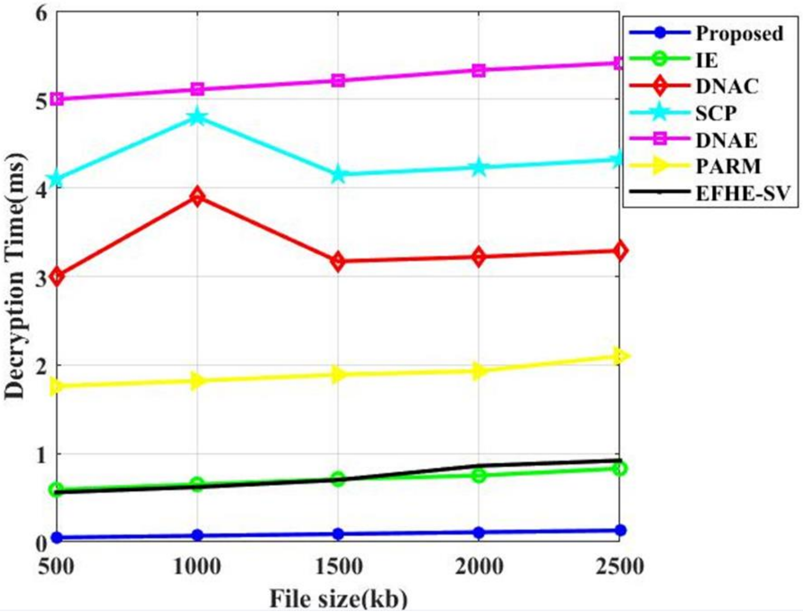

Figure 5 Comparison of Decryption Time

Table 4 DT Comparison

File size (kb)	Decryption time (ms)						
	EFHE-SV	PARM	DNAE	SCP	DNAC	IE	Proposed
500	0.56	1.76	5	4.1	3	0.59	0.05
1000	0.62	1.82	5.11	4.8	3.9	0.65	0.07
1500	0.70	1.89	5.21	4.15	3.17	0.71	0.09
2000	0.86	1.93	5.33	4.23	3.22	0.75	0.11
2500	0.92	2.1	5.41`	4.32	3.29	0.83	0.13

5.1.3. Throughput

Throughput is the most required parameter for evaluating secure data communication over a cloud storage system. Thus the throughput may change depending on the computation time of the designed technique. Here, the throughput of the designed method is associated with prevailing models which are detailed in table 5.

The achieved throughput of the developed HBC-PHE model gained 654 Kbps for 500kb which is compared with the EFHE-SV (123Kbps), PARM (356 Kbps), DNAE (88Kbps), SCP (74 Kbps), DNAC (218 Kbps), IE (76 Kbps) has achieved for 500kb file sizes which are illustrated in Figure 6. The overall validation of the throughput indicates the performance of the developed model which gained high throughput while comparing other techniques.

Table 5 Throughput

File size (kb)	Throughput (Kbps)						
	EFHE-SV	PARM	DNAE	SCP	DNAC	IE	Proposed
500	123	356	88	74	218	76	654
1000	113	344	81	68	203	69	643
1500	102	332	75	60	184	62	630
2000	87	310	69	56	173	58	623
2500	64	295	63	49	166	52	618

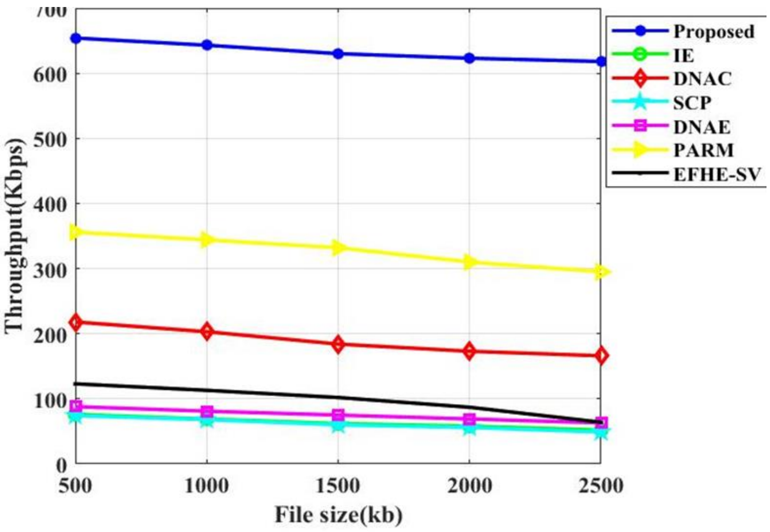

Figure 6 Throughput

5.1.4. Computation Time (CT)

CT is defined as the count of data encrypted in the cloud multiplied by CP and encrypted data execution time. Computation time is also called running time, and it is the process of time required to complete the program. Moreover, CPU time is measured by the computer for obtaining a final solution. The comparison of computational time is discussed in Table 6.

The computation time of the EFHE-SV replica achieved 0.03s and the PARM technique computation time is 0.33s. Moreover, the DNAE model attained 0.18s in computation time, and SCP accomplish a computation time of 0.14sec. Furthermore, the DNAC model attained 0.38s in computation time, and IE accomplishes computation time as 0.11sec. The comparison of computation time is elaborated in Figure 7. Moreover, the HBC-PHE technique achieved CT is 0.3s which is less while comparing other techniques.

Table 6 Comparison of computation time

Technique	Computation Time (s)
EFHE-SV	0.23
PARM	0.33
DNAE	0.18
SCP	0.14
DNAC	0.38
IE	0.11
Proposed	0.03

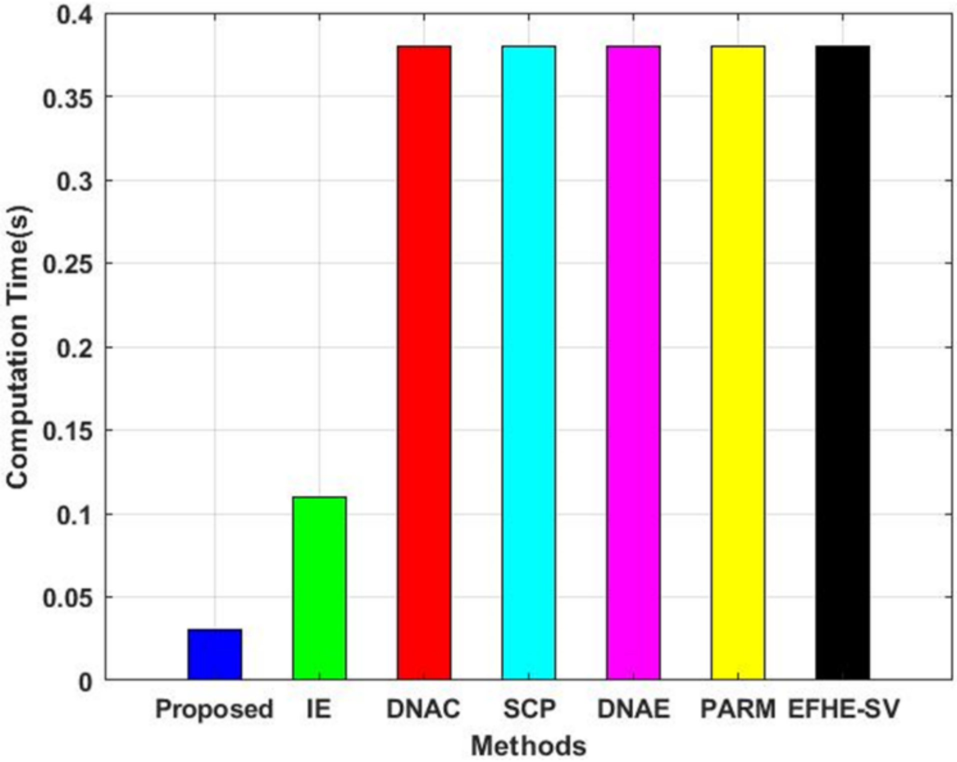

Figure 7 Comparison of Computation Time

5.1.5. Confidential Rate (CR)

CR is denoted as the modification among original data and received data through the performance of data propagation. Moreover, the CR of the designed technique is validated with prevailing models like EFHE-SV, PARM, DNAE, SCP, DNAC, and IE are shown in table 7. Likewise, the CR of the developed HBC-PHE technique gained 98.7% for 500 kb, which is associated with the other conventional methods like EFHE-SV, PARM, DNAE, SCP, DNAC, and IE. Consequently, the EFHE-SV method has attained an 88% of CR, PARM gained a 77 % CR, and then DNAE achieved 64% CR for 500 kb. Accordingly, the SCP method gained 90% of CR, DNAC accomplish 68 % CR, and IE achieved 92% CR for 500 kb. Nonetheless, the planned HBC-PHE model gained 98.7% of CR for 2500 kb, and the comparison of CR is demonstrated in Figure.8. Moreover, the overall comparison indicates the developed HBC-PHE technique is more CR than the other prevailing techniques.

Table 7 Confidential Rate comparison

File size (kb)	CR (%)						
	EFHE-SV	PARM	DNAE	SCP	DNAC	IE	Proposed
500	88	77	64	90	68	92	98.7
1000	83	75	62	88	66	89	98.2
1500	78	69	58	86	64	86	97.9
2000	73	66	56	81	62	84	97.5
2500	68	64	54	78	59	82	97

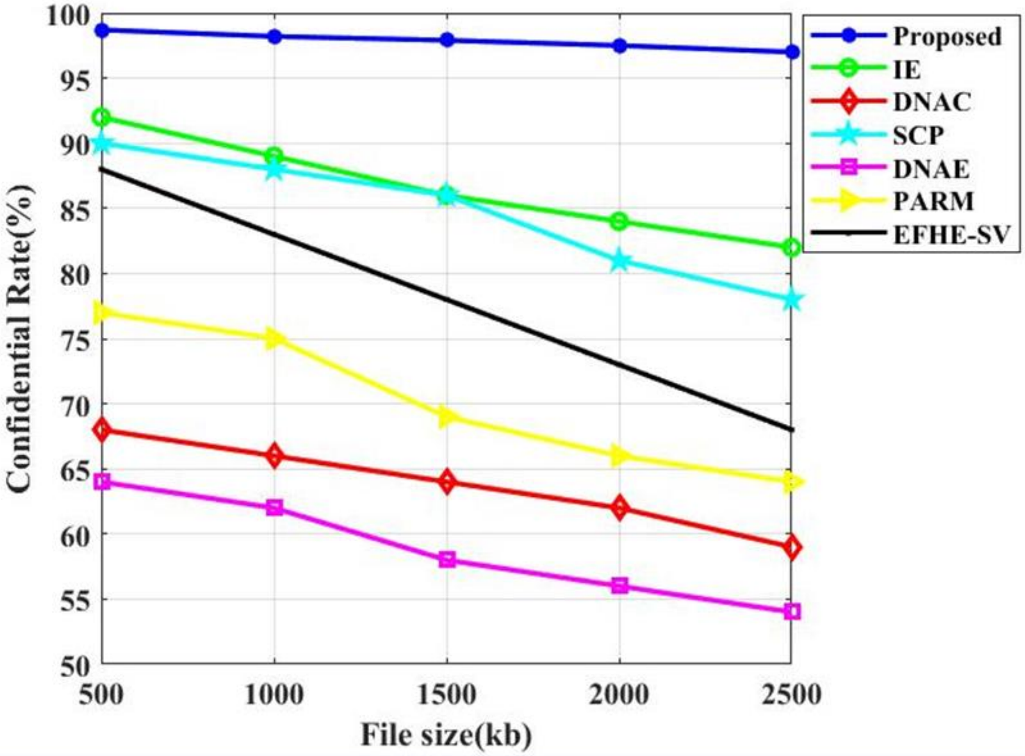

Figure 8 Comparison of Confidential Rate

5.1.6. Efficiency

One of the important parameters in a cloud environment is efficiency which proves the performance such as communication overheads and measurement. Additionally, the efficiency of the developed HBC-PHE technique is validated with other conventional models such as EFHE-SV, PARM, DNAE, SCP, DNAC, and IE as shown in table 8.

Moreover, the efficiency of the proposed HBC-PHE **(**98.34%) is compared with the conventional EFHE-SV (56%), PARM (75%), DNAE (64%), SCP (86%), DNAC (73%), and IE (90%) has achieved. Consequently, the efficiency of the developed model with other techniques was illustrated in Figure 9.

Table 8 Comparison of Efficiency

Technique	Efficiency (%)
EFHE-SV	56
PARM	75
DNAE	64
SCP	86
DNAC	73
IE	90
Proposed	98.34

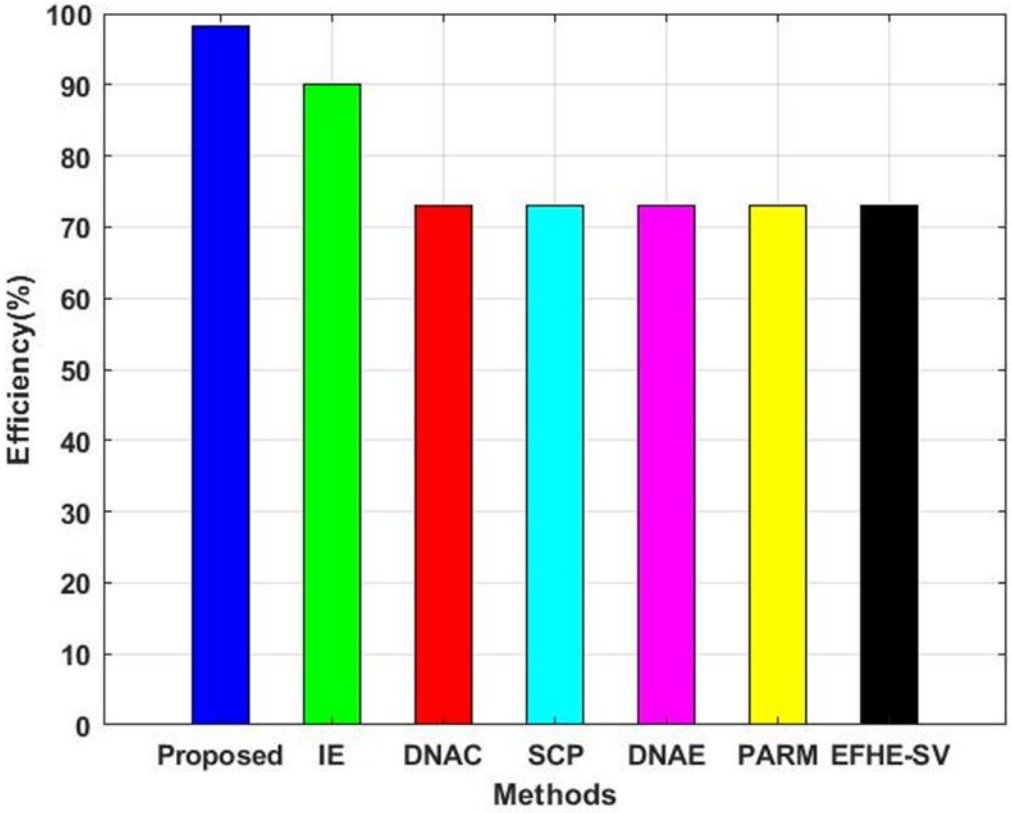

Figure 9 Comparison of Efficiency

5.2. Discussions

The proposed model of HBC-PHE has shown good performance by attaining the best results in encryption time, efficiency, confidential rate, decryption time, computational time, and throughput. Thus, the developed scheme generates the key for securing the data in the public and private keys. Next, data encryption is processed to convert the plain text into cipher text. Thus the developed HBC-PHE technique enhances the performance of security in the cloud through secure encryption.

The outstanding metrics comparisons are tabulated in table 9, in all parameter validation, the proposed HBC-PHE has gained the finest results. Moreover, the developed framework gained less encryption time of 0.08ms, less decryption time of 0.05%, high throughput of 654Kbps, and less computation time of 0.03s. Moreover, high efficiency and confidentiality rates are 98.34% and 98.7%. Hence, the robustness of the proposed HBC-PHE is verified and it can secure the data from malicious activity and unauthorized access by encrypted data. Here, the proposed technique has achieved better performance to secure the data in a cloud environment.

Table 9 Overall Performance Metrics

Methods	Performance Assessment with Key Metrics					
	Encryption Time (ms)	Decryption Time (ms)	Throughput (Kbps)	Computation Time (s)	Confidential Rate (%)	Efficiency (%)
EFHE-SV	1.8	0.56	123	0.23	88	56
PARM	1.19	1.76	356	0.33	77	75
DNAE	3	5	88	0.18	64	64
SCP	3.2	4.1	74	0.14	90	86
DNAC	4.5	3	218	0.38	68	73
IE	0.35	0.59	76	0.11	92	90
Proposed	0.08	0.05	654	0.03	98.7	98.34

6. CONCLUSIONS

Design a novel Hybrid HBC-PHE Scheme for enhancing the data security of the cloud from malware and attacks. Initially, collected datasets are stored in the cloud and they are transferred into the developed HBC-PHE framework. Initially, keys are generated and they are separated into the blocks using a private key. The designed model converts the plain text into ciphertext using the bat and cuckoo fitness function in PHE. Finally, cloud-stored data are encrypted successfully and the developed technique was validated with other prevailing techniques and has attained the finest result through attaining the correctness score of securing data encryption time as 0.08ms, decryption time as 0.05ms, and throughput as 654 Kbps.

REFERENCES

[1] Alashhab, Ziyad R., et al. "Impact of coronavirus pandemic crisis on technologies and cloud computing applications." Journal of Electronic Science and Technology 19.1 (2021): 100059.

[2] Zahoor, Saman, et al. "Cloud–fog–based smart grid model for efficient resource management." Sustainability 10.6 (2018): 2079.

[3] Sunyaev, Ali. "Cloud computing." Internet computing. Springer, Cham, 2020. 195-236.

[4] Tabrizchi, Hamed, and Marjan Kuchaki Rafsanjani. "A survey on security challenges in cloud computing: issues, threats, and solutions." The journal of supercomputing 76.12 (2020): 9493-9532.

[5] Abdulsalam, Yunusa Simpa, and Mustapha Hedabou. "Security and Privacy in Cloud Computing: Technical Review." Future Internet 14.1 (2021): 11.

[6] Samanta, Debabrata, et al. "Cipher block chaining support vector machine for secured decentralized cloud enabled intelligent IoT architecture." IEEE Access 9 (2021): 98013-98025.

[7] Al-Issa, Yazan, Mohammad Ashraf Ottom, and Ahmed Tamrawi. "eHealth cloud security challenges: a survey." Journal of healthcare engineering 2019 (2019).

[8] Yang, Tengfei, et al. "PLCOM: Privacy-preserving outsourcing computation of Legendre circularly orthogonal moment over encrypted image data." Information Sciences 505 (2019): 198-214.

[9] Athanere, Smita, and Ramesh Thakur. "Blockchain based hierarchical semi-decentralized approach using IPFS for secure and efficient data sharing." Journal of King Saud University-Computer and Information Sciences (2022).

[10] Adee, Rose, and Haralambos Mouratidis. "A Dynamic Four-Step Data Security Model for Data in Cloud Computing Based on Cryptography and Steganography." Sensors 22.3 (2022): 1109.

[11] Anwar, Md Navid Bin, et al. "Comparative Study of Cryptography Algorithms and Its' Applications." International Journal of Computer Networks and Communications Security 7.5 (2019): 96-103.

[12] Kumar, Randhir, and Rakesh Tripathi. "Secure healthcare framework using blockchain and public key cryptography." Blockchain Cybersecurity, Trust and Privacy. Springer, Cham, 2020. 185-202.

[13] Subramanian, Nalini, and Andrews Jeyaraj. "Recent security challenges in cloud computing." Computers & Electrical Engineering 71 (2018): 28-42.

[14] Abiodun, Esther Omolara, et al. "Reinforcing the security of instant messaging systems using an enhanced honey encryption scheme: the case of WhatsApp." Wireless Personal Communications 112.4 (2020): 2533-2556.

[15] Hwang, Yong-Woon, et al. "Current Status and Security Trend of OSINT." Wireless Communications and Mobile Computing 2022 (2022).

[16] Singh, Saurabh, et al. "SH-BlockCC: A secure and efficient Internet of things smart home architecture based on cloud computing and blockchain technology." International Journal of Distributed Sensor Networks 15.4 (2019): 1550147719844159.

[17] Kaur, Harleen, et al. "A proposed solution and future direction for blockchain-based heterogeneous medicare data in cloud environment." Journal of medical systems 42.8 (2018): 1-11.

[18] Mansouri, Najme, and Mohammad Masoud Javidi. "Cost-based job scheduling strategy in cloud computing environments." Distributed and Parallel Databases 38.2 (2020): 365-400.

[19] Ma, Chuan, et al. "On safeguarding privacy and security in the framework of federated learning." IEEE network 34.4 (2020): 242-248.

[20] Bordonaba-Juste, Mª, Laura Lucia-Palacios, and Raúl Pérez-López. "Generational differences in valuing usefulness, privacy and security negative experiences for paying for cloud services." Information Systems and e-Business Management 18.1 (2020): 35-60.

[21] Geetha, N., and K. Mahesh. "An Efficient Enhanced Full Homomorphic Encryption for Securing Video in Cloud Environment." Wireless Personal Communications 123.2 (2022): 1553-1571.

[22] Pang, Hongping, and Baocang Wang. "Privacy-preserving association rule mining using homomorphic encryption in a multikey environment." IEEE Systems Journal 15.2 (2020): 3131-3141.

[23] Jin, Byung-Wook, Jung-Oh Park, and Hyung-Jin Mun. "A design of secure communication protocol using RLWE-based homomorphic encryption in IoT convergence cloud environment." Wireless Personal Communications 105.2 (2019): 599-618.

[24] Algaradi, Thoyazan Sultan, and Boddireddy Rama. "An authenticated key management scheme for securing big data environment." International Journal of Electrical & Computer Engineering (2088-8708) 12.3 (2022).

[25] Drucker, Nir, and Shay Gueron. "Achieving trustworthy Homomorphic Encryption by combining it with a Trusted Execution Environment." J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl. 9.1 (2018): 86-99.

[26] Zhao, Feng, Chao Li, and Chun Feng Liu. "A cloud computing security solution based on fully homomorphic encryption." 16th international conference on advanced communication technology. IEEE, 2014.

[27] Mercy Joseph, Gobi Mohan, "A Novel Algorithm for Secured Data Sharing in Cloud using GWOA-DNA Cryptography", International Journal of Computer Networks and Applications (IJCNA), 9(1), PP: 114-124, 2022, DOI: 10.22247/ijcna/2022/211630.

Authors

Mercy Joseph – Received MSc Computer Science Degree from Mahatma Gandhi University Kottayam, Kerala and received ME (CSE) from satyabama university Chennai. Currently, she is a research scholar at the Department of Computer Science, Chikkanna Government Arts College Tirupur, India. Her research interests include Cryptography and network security, Data Security in cloud computing and machine learning.

Dr. Gobi Mohan – Associate Professor in Department of Computer Science in Chikkanna Government Arts College, Tirupur, India. He has published original articles and the finest journals in the area of cryptography and security. His research interests include (but are not limited to) Cryptography, Java, Software Engineering and Information Systems Security.

How to cite this article:

Mercy Joseph, Gobi Mohan, "Design a hybrid Optimization and Homomorphic Encryption for Securing Data in a Cloud Environment", International Journal of Computer Networks and Applications (IJCNA), 9(4), PP: 385-398, 2022, DOI: 10.22247/ijcna/2022/214502 .