JOURNAL OF DYNAMICS AND CONTROL

VOLUME 8 ISSUE 9

IMPROVING IDENTITY BASED ENCRYPTION IN CLOUD COMPUTING USING SECURE CHANNEL ESTABLISHMENT ALGORITHM

Gaurav Pandey[1], Jyoti Shekhawat[2]

[1]Research Scholar (EN: 21TEC3CS004),

[2]Assistant Professor

Vivekananda Global University, Jaipur, India

IMPROVING IDENTITY BASED ENCRYPTION IN CLOUD COMPUTING USING SECURE CHANNEL ESTABLISHMENT ALGORITHM

Gaurav Pandey[1], Jyoti Shekhawat[2]

[1]Research Scholar (EN: 21TEC3CS004), *[2]Assistant Professor*
Vivekananda Global University, Jaipur, India
gauravpandey5181@gmail.com[1], jyoti.shekhawat@vgu.ac.in[2]

ABSTRACT: Cloud computing (CC) facilitates clients and businesses with different capacities so that they can store and handle their data in the data centers of third party. It depends on exchange of assets to obtain intelligibility and substantial savings, just like a utility (like the power grid) across an organization. The Fully Homomorphic Encryption (FHE) is adaptable to perform all kinds of calculation on the data that the cloud has contained. The FHE facilitates the execution of all kinds of operations on encrypted data without performing decryption. The application of FHE is essential to keep the CC infrastructure secure. The calculations are outsourced on the secret data to the cloud server with the secret key which assists in decrypting the result of calculation. This work suggests an innovative attribute-based framework for enhancing the security of cloud. The proposed model is implemented in MATLAB and results is analyzed in terms of energy consumption, time and resource consumption.

KEYWORDS: Cloud Computing, FHE, Identity Based Encryption, Diffie-Hellman.

1. Introduction

The model of Cloud computing aims at facilitating practical, on-demand network accessibility to a joint reserve of adaptive computing resources. A cloud computing framework provides multiple computing resources in form of services over the web. Storage is considered to be a major amenity offered by the cloud (for example, Simple Storage Services—Amazon S3), allowing customers to save vast volume of data on a remote cloud without having to bother with perplexing handling of storage hardware [1]. The majority of existing clouds are constructed on top of advanced data centres. It encompasses IaaS (Infrastructure as a Service), PaaS (Platform as a Service), and SaaS (Software as a Service), and delivers such amenities such as functions to enable end users to pay according to the amount used by them. Figure 1 presents a hierarchical outlook of cloud computing.

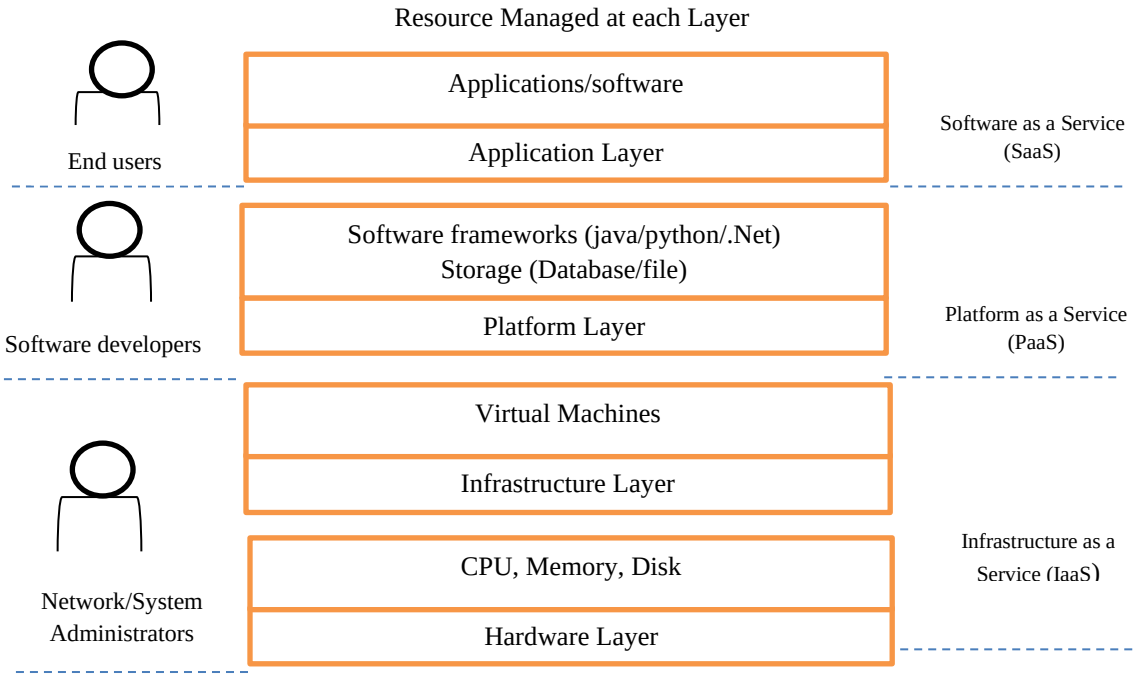

Figure 1: Hierarchical Outlook of Cloud Computing

Public Cloud implies that such computing depending upon other organization or third parties for providing effectual cloud services over Internet. This cloud framework can be easily accessed by general public or a huge industry group. The public clouds allow to use resources as a service, most often over an internet connection. Users are capable of scaling their use on demand and there is not any necessity of purchasing hardware for deploying the service. Public CPs (Cloud Providers) focus on managing the infrastructure and pool resources into the capacity that is enough for its users. Any user can deploy this cloud using an internet for offering a same range of potentials and services. Most of its users are typically residential users and they employ network of internet service provider. To illustrate, Google, Amazon and Microsoft are public cloud vendors for rendering services to the general public. The servers of third party vendor are exploited to store the data which the customer has generated and submitted. It leads to cause security issue. However, such clouds consume least cost due to its potential of sharing similar resources with various consumers. Public clouds offer reliability as restricted number of resources are obtained from varied locations.

1.1. Identity-based encryption (IBE)

Public key encryption (PKE) solves the issue of key allocation in symmetric keys by providing an outstanding solution. Authenticating the public keys of customers is a significant challenge in PKE. The conventional PKE issues certificates to authenticate the customer public keys. However, the public key system suffers a substantial load of certificate management which is also PKE's main flaw. To address this shortcoming, Shamir introduced a new concept called "Identity based public key cryptography" in 1984. This is a public key cryptosystem using the unique identity of each user like its public key. Thus, the authenticity of this public key can't be doubted any more, and therefore, certificate is unrequired. Boneh and Franklin set forth the first real-world IBE (identity-based encryption) system in 2021. Since its existence, IBE has been the center of attraction in both academic and industrial domains. A generic approach to overcome the issue of data sharing in cloud computing is to employ access controls based on cryptography schemes, for example identity-based encryption (IBE). In addition, following are three main security objectives which identity-based access control over shared data must satisfy.

- Data confidentiality: Plaintext of shared data saved in cloud servers must be out of the reach of unauthentic users. Apart from this, cloud servers, which are intended as truthful but curious, should also be stopped from accessing the plaintext of the data shared.
- Backward secrecy: To maintain backward secrecy, a user with expired authority, or compromised secret key, must be deterred from retrieving the plaintext of the data shared later that is still encrypted under his ID.
- Forward secrecy: To maintain forward secrecy, a user with expired authorization or expired secret key must be deterred from knowing the plaintext of the shared data that can be earlier retrieved by him/her.

2. Literature Survey

W. Shen, et.al (2022) designed an innovative key update approach which had compatibility with BLS signature, and utilized for inspecting whether identity-based data was reliable or not. The Third Party Auditor (TPA) was adopted for creating the update information. The user had potential to update the Private Key (PK) depending upon the PK in one earlier time period and the update information from the TPA. The results indicated the security and effectiveness of the designed approach over other methods. While verifying the proof, higher computing costs were required on the cloud side.

M. Yang, et.al (2022) constructed an effective assessment mechanism of cloud service consistency level based on D-S theory. Thereafter, a representation mechanism was put forward and its transition state was integrated with Markov chain to check whether the cloud service and its changes were reliable. At last, the results of case analysis indicated that the constructed mechanism was feasible to evaluate the reliability of cloud service and its changes and provide complete outcomes to allow users to choose the reasonable service and manage the reliability.

L. Cao, et.al (2022) presented a secure, workload-balanced, and energy-efficient Virtual Machine allocation (SWEVMA) strategy for defending against co-residence assaults. Three components, such as security risks, the energy use and the unbalanced workloads among dissimilar physical servers were assessed and diminished for displaying this issue as an optimization issue. In addition, it was assumed that the random number of VMs were shown up from dissimilar users at random timings and the optimization solution was expected for it. The experimental results uncovered that the presented technique had performed effectively. This method was not resistible against all kind of assaults which mitigated the security.

R. C. Patil, et.al (2022) recommended another strategy for identifying the data leakage in Cloud Computing (CC) on the basis of classifying the data with deep learning (DL) methods. This technique emphasized on get-together and processing the information data to wipe out the noise and smoothen the data. Generative Regression Kernel-Support Vector Machine (GRK-SVM) algorithm was used for classifying the data. The results demonstrated that the accuracy of recommended strategy was 97%, precision was 92%, recall was 67%, F-1 score was 66%, Root Mean Square Error (RMSE) was 62% and Signal-to-Noise Ratio (SNR) was 61%. This algorithm was not sent any notification in case of any change in information due to which the issue of security was occurred.

S. Malhotra, et.al (2023) suggested a Symmetric Searchable Encryption (SSE) in ML (Machine Learning) based approach to encrypt and retrieve the cloud data. This approach was implemented to enhance the data security and an effectual Keyword Ranking (KR) method was generated on the basis of Artificial Neural Network (ANN). A comparative analysis was conducted on the suggested approach against multiclass Support Vector Machine (SVM) and Naïve Bayes (NB) methods. Furthermore, this approach was useful to prevent the abnormalities such as cyber-attack plans and error in Cloud Storage System (CSS). The retrieval process was not properly secure.

G. Ha, et.al (2023) introduced a new threat model to resist against the side-channel assaults. Different from customary approaches, the adversary had potential to learn the approximate ratio of stored chunks to un-stored ones in outsourced files. Two defense methods were created to lessen the data privacy leakage and create the interaction protocols amid clients and the server when deduplication was verified. It led to diminish the probability that the adversary was major cause of alleviating the data privacy.

Y. Teng, et.al (2023) introduced a notion of clustering deviation for tackling the issue of internal data leakage [68]. An enhanced Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm was projected to tolerate the clustering deviation. The data deduplication technique was put forward for utilizing the users as clustering samples. The clustering results were taken in account to compute the status of the data. Diverse encryption approaches were implemented to protect the security of unpopular data.

H. Attou, et.al (2023) constructed a cloud-based Intrusion Detection System (IDS) on the basis of Random Forest (RF) and feature engineering (FE). The underlying one was embraced to perform features engineering and the last one was assisted in foreseeing and identifying the intrusions. This system was prepared after relieving how much attributes in the two models. This work emphasized on securing and combining the RF algorithm for upgrading the accuracy of the constructed system. Bot-IoT and NSL-KDD data sets were utilized for computing the constructed system. The accuracy of this system was determined 98.3% on the underlying data set and 99.99% on second one. The results portrayed that the constructed system yielded superior accuracy, precision, and recall in comparison with existing methods. The recall was found lower on NSL-KDD.

X. Ouyang, et.al (2023) suggested a Secure Retrieval and Blockchain-Assisted Verifiable (SRBAV) strategy for scrambled Remote Sensing Images (RSIs) in the cloud system. Considering the contribution of clear class features in geographical objects inside RSIs, a strategy was made arrangements for recovering RSI securely and efficiently. Besides, a verifiable strategy was carried out with Blockchain and Merkle trees to decide the honesty and accuracy of the storage and retrieval services which a Cloud Service Provider (CSP) had offered for supplanting the conventional Third Party Auditor (TPA). The experimental outcomes portrayed that the suggested technique was secure, verifiable and feasible for recovering the RSIs securely, and forestalled the malicious way of behaving of CSP. However, this technique consumed much time for encrypting the data.

3. Research Methodology

3.1. Original identity transmission and secure channel generation: - At initial stage, the user sends the original identity to the Key Identity Provider and creates a secure channel. For this, Diffie-Hellman (DH) algorithm is implemented. DH is considered as a mathematical algorithm which assists two systems in generating an identical secret whose transmission is done on both systems. However, these systems have not any communication among one another. The transmitted secret is employed for broadcasting a cryptographic encryption key in a secure manner. This work aims to encrypt the traffic amid 2 systems. This algorithm is effective for encrypting the data on web on the basis of Secure Socket Layer (SSL) or Transport Layer Security (TLS). Moreover, this algorithm makes the deployment of Secure Shell (SSH) protocol. This protocol is employed for exchanging the key securely for encrypting the data. A shared secret which is also known as Key Encryption Key (KEK) is considered for exchanging this secure transmission. Thereafter, the transmitted secret helps to encrypt the symmetric key to achieve safe transmittal. This process is initialized by generating a private key on every side of the correspondence. Every end concentrates on generating a public key that is obtained as the derivation of the private key. The next task is to exchange the public keys amid both systems. Hence, it provides own private key and the other public key of systems to each side of the correspondence. After completing the key sharing, the process is executed further. Consequently, this algorithm leads to generate a shared key as identical cryptographic key whose transmission is done via each side. Afterward, the mathematical operation is deployed against the private key and other side's public key for acquiring a value. In the end, cryptographic key assists in encrypting the traffic. The Diffie-Hellman algorithm often utilizes a shared secret for encrypting a symmetric key for one of the symmetric algorithms and transmitting it in secure manner. The inaccessible end is exploited in this algorithm for decrypting it with the shared secret. At the completion of transmitting a symmetric key securely, the data is encrypted and a communication is established securely.

Pseudo Code of Diffie-Hellman Algorithm

Initialization:

Choose a prime number p and a primitive root modulo p, denoted as g.

Parties (Alice and Bob) agree on these values and keep them secret.

Private Key Generation:

Both Alice and Bob independently generate their private keys.

Alice chooses a secret integer a.

Bob chooses a secret integer b.

Public Key Computation:

Both Alice and Bob compute their respective public keys using the following formula:

Alice computes her public key: A = g^a % p

Bob computes his public key: B = g^b % p

Public Key Exchange:

Alice and Bob exchange their computed public keys over an insecure channel.

Shared Secret Calculation:

Alice receives Bob's public key B, and Bob receives Alice's public key A.

Alice calculates the shared secret using Bob's public key and her private key:

Shared Secret (Alice): s = B^a % p

Bob calculates the shared secret using Alice's public key and his private key:

Shared Secret (Bob): s = A^b % p

Final Shared Secret:

Both Alice and Bob have now computed the same shared secret s.

3.2. Virtual Identity Generation and Transmission: - The key identity provider provides an identity provider and private key to the user. Moreover, it sends a public key and identity to the cloud. The fourth stage employs the HE technique so that the data is encrypted. The HE is a property of an encryption methodology which is robust to be executed in the encrypted information (ciphertexts) and to preserve the actual operation outcomes on the underlying clear text operands. This approach is consisted of a number of applications and it is deployed for carrying out computations on encrypted data on the basis of data which is concealed from it. Assume a scenario in which a client is there along with a relatively weak computing device and a server which is comprised of robust computing resources. Moreover, considers that a client focuses on outsourcing its data to the remote server so that computing results are obtained and performing computations over the data. The HE approach is suitable for this process. It helps a client in generating a ciphertext of its secret data and transmitting it to a server. Afterward, a ciphertext of the computation result is provided to the server. For this, the homomorphic evaluations are carried out on the ciphertext of client and this ciphertext is returned to the client. At last, the client is able to decrypt the evaluated ciphertext, and attains the computation result. There is not any association amid the computation cost

for a client and the size of a delegated function. HE approach is capable of handing the ciphertext for users without key, and in this process, no information related to plaintext is revealed. This attribute is effective for protecting the security of information and enhancing the efficacy to process the information. More specifically, in case an encryption function f results in satisfying

$$f(a) + f(b) = f(a + b)$$

It is considered that it has additive homomorphism, and in case it satisfies

$$f(a) * f(b) = f(a * b)$$

This implies that this function contains multiplicative homomorphism. The Homomorphic Encryption (HE) is executed for managing ciphertext and to retrieve, calculate and compute the ciphertext within the cloud directly. Hence, the output is provided to customers as ciphertext. There is not any necessity of regular encryption and decryption amid the cloud and customers in this approach. It results in mitigating the message and computational overhead. There are some properties of this approach. This procedure is executed in seven stages: original identity transmission, secure channel generation, Identity provider and private key transmission, public key and identity transmission, data encryption, data transmission and data storage in cloud.

The key sizes for Fully Homomorphic Encryption (FHE) schemes commonly used in practical applications can vary, but they are generally larger than those of traditional encryption schemes due to the additional requirements for supporting computations on encrypted data. Key sizes are influenced by factors such as the specific FHE algorithm, the desired security level. In this research work we have used 150-bit long key for the encryption and decryption. The Fully Homomorphic Encryption keys is bit longer than traditional algorithms due to which it has less chances of attacks and also less execution time.

Pseudo code of Homomorphic Encryption

Key Generation

Generate two large prime numbers p and q

Compute n = p * q

Compute λ = lcm (p - 1, q - 1)

Select a random integer g where 1 < g < n and gcd (g, n) = 1

Public Key: (n, g)

Private Key: λ

Encryption

Given a plaintext message m (integer)

Select a random integer r where 0 < r < n

Compute ciphertext c = (g^m * r^n) % n

Decryption

Given a ciphertext c

Compute L = (c^λ % n^2 - 1) / n

Compute plaintext message m = (L * μ) % n, where μ = mod inv (L, n)

Homomorphic Addition

Given ciphertexts c1 and c2 encrypted with the same public key

Compute ciphertext c3 = (c1 * c2) % n^2

Homomorphic Scalar Multiplication

Given ciphertext c and scalar value x

Compute ciphertext c4 = (c^x) % n^2

Homomorphic Evaluation

Perform computations on ciphertexts as if they were plaintexts

3.3. Data Security: - In the next stage, the encrypted data is transmitted to the cloud. In the last stage, the cloud attains the public key and identity from the key identity provider and private key from user. Then it focuses on decrypting both the keys and storing the data. Homomorphic Encryption (HE) is an effective method to perform data encryption and decryption in diverse phases which are described as:

3.3.1. Data Encryption: The data owner focuses on encrypting their sensitive data. For this, a homomorphic encryption method is employed. This method further has diverse kinds namely partially homomorphic encryption or fully homomorphic encryption. The next task is to transmit and store the encrypted data in a secure way for which there is not any necessity of decrypting the data.

3.3.2. Data Processing: A third party, like a service provider or an application, is capable of executing calculations on the encrypted data. HE method is composed of particular mathematical operations such as addition and multiplication, which can be executed on the encrypted data.

3.4. Data Decryption: - After completing the selected computations on the encrypted data, the results are generated in encrypted form. The data owner, who is responsible for possessing the decryption key, becomes able to decrypt the result for attaining the final output. Homomorphic encryption is consisted of diverse levels for determining the kinds of operations which are suitable to preserve the encryption. Moreover, partially homomorphic encryption methods namely Paillier encryption etc. are also available. But, these method comprises only restricted number of operations such as addition and multiplication, and fully homomorphic encryption methods, namely TFHE, BFV, CKKS can be deployed to carry out complicated computations.

4. Result and Discussion

This research work is implemented in MATLAB using the mathematical tool box. It is an interactive program which provides numerical computation and visualization of data. With the help of its programming capabilities it provides tool which is very useful for all areas of science and engineering. Image Processing Toolbox chains a various set of image and their types, together with high forceful range, embedded ICC profile, topographic and giga pixel resolution.

Table 1: Simulation Parameters

Number of cloudlets	10
Number of virtual machines	7
Operating system	Linux
Architecture	64 bit
Encryption Data	Image type

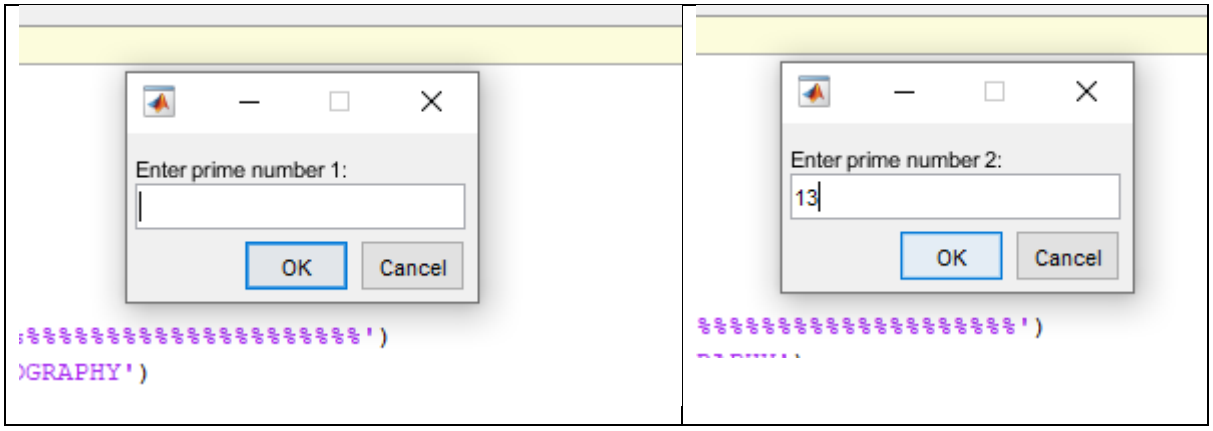

Figure 2: Enter Prime Numbers

As shown in figure 2, the Diffie-Hellman algorithm is applied for the secure channel establishment. The prime number of entered for secure key generation.

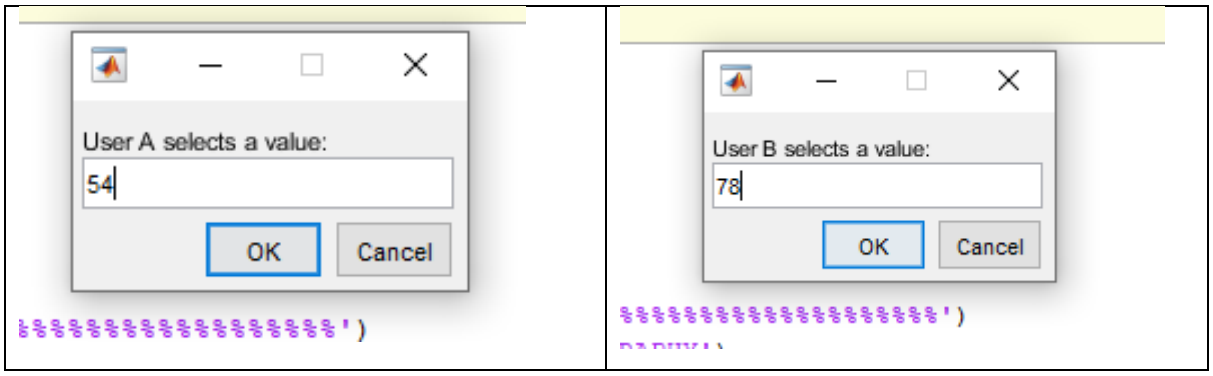

Figure 3: Enter Secret Keys

As shown in figure 3, the user A and B enter its secret keys for the secure key calculation.

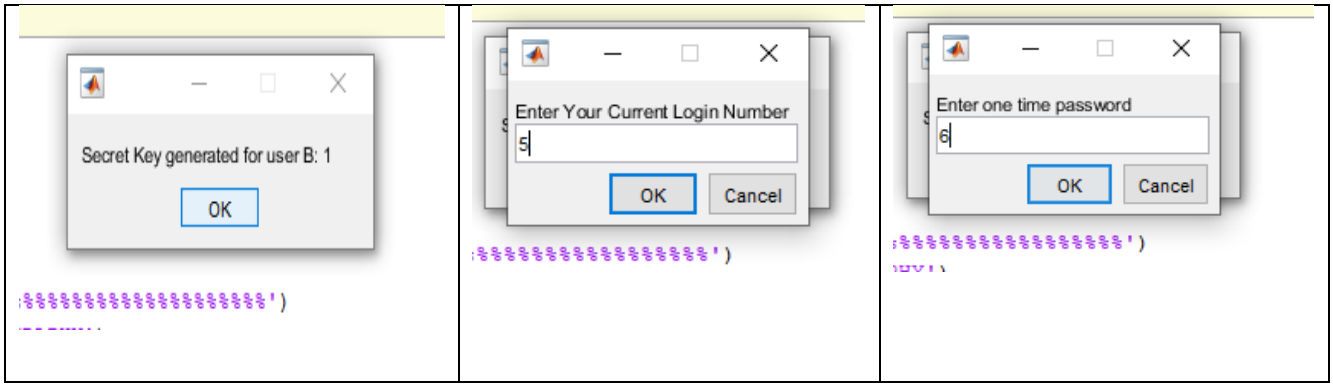

Figure 4: Generation of Keys

As shown in figure 4, the OTP is generated in this phase which is the combination of secret key and number of login times.

Table 1: Probability of Attacks

RSA	BF-IBE	Out-FS	Improved Out-FS
120	100	220	32

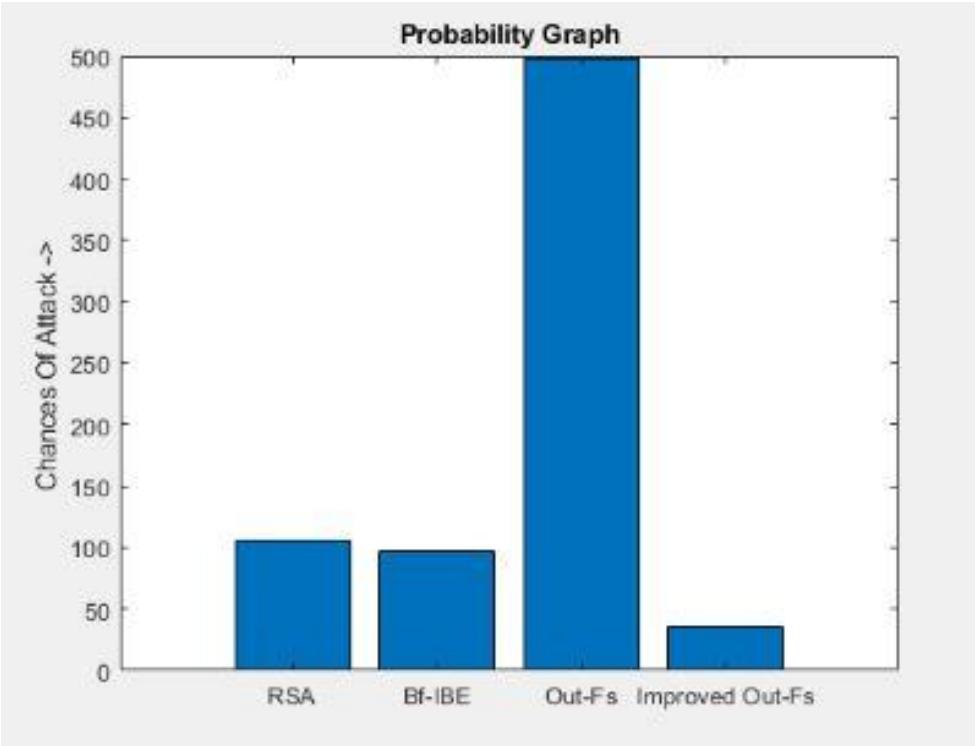

Figure 5: Attack Probability Analysis

As shown in figure 5, the attack probability of proposed technique is compared with Out-Fs, Bf-IBE and RSA. The Proposed technique has least chances of attack as compared to other techniques.

Table 1: Time Comparison

RSA	BF-IBE	Out-FS	Improved Out-FS
70	67	140	35

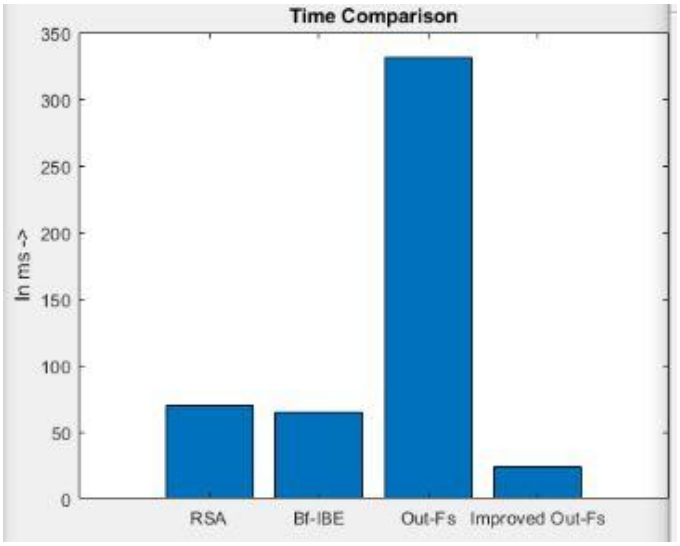

Figure 6: Time Analysis

As shown in figure 6, the Execution time of proposed technique is compared with Out-Fs, Bf-IBE and RSA. The Proposed technique has least execution time as compared to other techniques.

Table 3: Space Utilizations

RSA	BF-IBE	Out-FS	Improved Out-FS
3000	2800	2600	2500

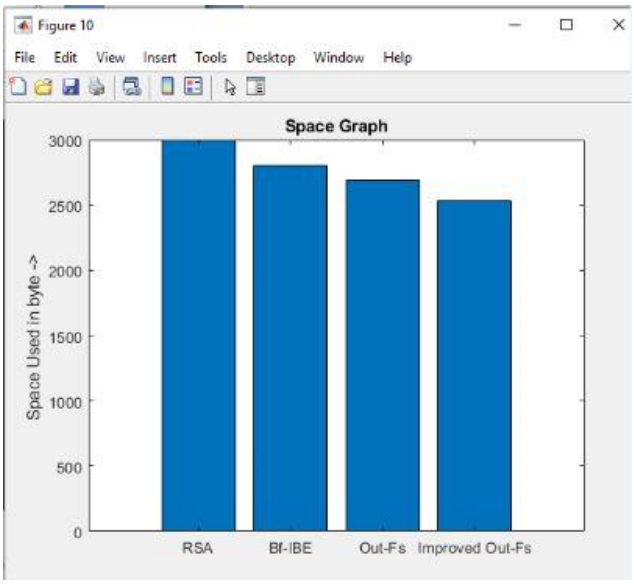

Figure 7: Space Utilization Analysis

As shown in figure 7, the space utilization of proposed technique is compared with Out-Fs, Bf-IBE and RSA. The Proposed technique has least space utilization as compared to other techniques.

5. Conclusion

The performance of cloud computing is affected due to data breaches and security issues. Therefore, to provide security measures and privacy, the service providers focus on deploying cryptography methods, authentication approaches and virtualization. The control will no longer persist in the hand of user's id. The service provider hosts the data and web applications. The encryption schemes are suggested for the secure authentication and certification in cloud computing. This work suggests a novel attribute based scheme will be proposed which will be less complex and more secure for the certificate distribution. The proposed method is the based on the Diffie-Hellman algorithm or the identity exchange. The homomorphic encryption (HE) approach is applied to encrypt the data. The proposed model is implemented on MATLAB and an analysis is conducted on results concerning energy, time and chances of attacks. It is analysed that proposed model performs well as compared to existing Identity-Based Encryption model for cloud data security.

References

[1] T. E. Trueman and P. Narayanasamy, "Ensuring Privacy and Data Freshness for Public Auditing of Shared Data in Cloud," 2015 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), Bangalore, India, 2015, pp. 22-27

[2] H. Cheng, C. Rong, K. Hwang, W. Wang and Y. Li, "Secure big data storage and sharing scheme for cloud tenants," in China Communications, vol. 12, no. 6, pp. 106-115, June 2015

[3] S. A. Ghafour, P. Ghodous and C. Bonnet, "Privacy Preserving Data Integration across Autonomous Cloud Services," 2015 IEEE 8th International Conference on Cloud Computing, New York, NY, USA, 2015, pp. 1099-1102

[4] L. Qiu, K. Gai and M. Qiu, "Optimal Big Data Sharing Approach for Tele-Health in Cloud Computing," 2016 IEEE International Conference on Smart Cloud (SmartCloud), New York, NY, USA, 2016, pp. 184-189

[5] D. Dongare and V. Kadroli, "Panda: Public auditing for shared data with efficient user revocation in the cloud," 2016 Online International Conference on Green Engineering and Technologies (IC-GET), Coimbatore, India, 2016, pp. 1-3

[6] D. Ulybyshev et al., "Privacy-Preserving Data Dissemination in Untrusted Cloud," 2017 IEEE 10th International Conference on Cloud Computing (CLOUD), Honolulu, HI, USA, 2017, pp. 770-773

[7] S. Samundiswary and N. M. Dongre, "Public auditing for shared data in cloud with safe user revocation," 2017 International conference of Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2017, pp. 53-57

[8] W. Shen, J. Yu, M. Yang and J. Hu, "Efficient Identity-Based Data Integrity Auditing with Key-Exposure Resistance for Cloud Storage," in IEEE Transactions on Dependable and Secure Computing, vol. 1, no. 13, pp. 637-645, 2022

[9] M. Yang, T. Gao, W. Xie, L. Jia and T. Zhang, "The Assessment of Cloud Service Trustworthiness State Based on D-S Theory and Markov Chain," in IEEE Access, vol. 10, pp. 68618-68632, 2022

[10] L. Cao, R. Li, X. Ruan and Y. Liu, "Defending Against Co-Residence Attack in Energy-Efficient Cloud: An Optimization Based Real-Time Secure VM Allocation Strategy," in IEEE Access, vol. 10, pp. 98549-98561, 2022

[11] R. C. Patil, A. Kumar, N. T, M. Suganthi, A. VS S. Rama Rao and Rajesh A, “Data Leakage Detection in Cloud Computing Environment Using Classification Based on Deep Learning Architectures”, International Journal of Intelligent Systems and Applications in Engineering, vol. 10, no. 2, pp. 281–285, 2022

[12] S. Malhotra and W. Singh, “An efficacy analysis of data encryption architecture for cloud platform”, Procedia Computer Science, vol. 218, pp. 989-1002, 31 January 2023

[13] G. Ha, H. Chen, C. Jia and M. Li, "Threat Model and Defense Scheme for Side-Channel Attacks in Client-Side Deduplication," in Tsinghua Science and Technology, vol. 28, no. 1, pp. 1-12, February 2023

[14] Y. Teng, H. Xian, Q. Lu and F. Guo, "A Data Deduplication Scheme Based on DBSCAN With Tolerable Clustering Deviation," in IEEE Access, vol. 11, pp. 9742-9750, 2023

[15] H. Attou, A. Guezzaz, S. Benkirane, M. Azrour and Y. Farhaoui, "Cloud-Based Intrusion Detection Approach Using Machine Learning Techniques," in Big Data Mining and Analytics, vol. 6, no. 3, pp. 311-320, September 2023

[16] X. Ouyang, Y. Xu, Y. Mao, Y. Liu, Z. Wang and Y. Yan, "Blockchain-Assisted Verifiable and Secure Remote Sensing Image Retrieval in Cloud Environment," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 16, pp. 1378-1389, 2023