Original Article
ISSN (Online): 2582-7472

ShodhKosh: Journal of Visual and Performing Arts
April 2024 5(4), 861–869

DATA ENCRYPTION TECHNIQUES FOR SECURING CLOUD STORAGE AND COMMUNICATION

Ravindrakumar [1]

[1] Assistant Professor, Department of Computer Science, Government First Grade College Chitaguppa

ABSTRACT

Because of the fast use of cloud computing, effective security measures have become necessary in order to safeguard sensitive data from being accessed by unauthorised parties and from being compromised. The encryption of data is an essential component of cloud security, since it guarantees the confidentiality and integrity of data while it is being stored and sent. Within the scope of this work, a number of distinct encryption methods, including as symmetric key encryption, asymmetric key encryption, and homomorphic encryption, are investigated. The advantages, limits, and appropriateness of these encryption approaches for various cloud settings are discussed. A strong emphasis is placed on the role that encryption plays in maintaining regulatory compliance, minimising risks associated with eavesdropping, and safeguarding communications from beginning to finish. In addition, we investigate hybrid encryption strategies, which combine the advantages of symmetric and asymmetric systems in order to strike a balance between efficiency and security. Another topic that is covered is the difficulties associated with key management, performance overhead, and interaction with the cloud infrastructure that is already in place. The use of encryption frameworks to protect cloud storage and communication in areas such as healthcare, banking, and e-commerce is demonstrated through case studies that are based on real-world scenarios. In the future, there is the possibility of developing encryption that is resistant to quantum computing in order to combat the dangers that are posed by quantum computing. The purpose of this study is to provide organisations with a guidance that can assist them in improving their cloud security postures by utilising modern encryption technology.

Keywords: Data Encryption, Cloud Security, Symmetric Encryption, Asymmetric Encryption, Homomorphic Encryption, Key Management, Quantum-Resistant Cryptography

DOI
10.29121/shodhkosh.v5.i4.2024.3360

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

1. INTRODUCTION

With the broad adoption of cloud storage and communication technologies, data security has arisen as a primary concern in the constantly changing digital era. This is especially true that cloud storage has become increasingly popular. When it comes to storing, processing, and exchanging sensitive information, cloud services are being more relied upon by individuals, organisations, and governments alike. This transformation has been brought about by the cloud's unrivalled advantages, which include decreased costs, increased scalability, more accessibility, and the opportunity for increased collaboration. The migration of data to cloud settings, on the other hand, makes it more vulnerable to cyber attacks, security breaches, and unauthorised access. When it comes to protecting cloud storage and communication,

encryption methods have evolved into indispensable instruments for mitigating the dangers that are associated with them.

2. BENEFITS OF DATA ENCRYPTION IN CLOUD STORAGE

Individuals and organisations have become aware of the significance of gradually lowering the risks of data breaches as a result of the growing accumulation of vast amounts of data that are both sensitive and kept on the cloud. The susceptibility of cloud storage to assaults has been increased as a result of the growing volume of data stored in the cloud. More than seven million records are compromised every single day, according to the Breach Data Index, which indicates that the pace of data breaches is increasing at an alarming rate. When it comes to enhancing the safety of any data that users keep in cloud storage, data encryption is an absolutely necessary component. Encryption is the process of converting plaintext data into ciphertext, which renders the data unintelligible to anybody who does not possess the key to decrypt it. During the process of data transmission from one party to another, the use of encryption in cloud storage increases the level of trust and secrecy that accompany the data. A user can have peace of mind regarding the safety of their data during the processes of transmission, storage, and retrieval if they encrypt their data before releasing it into cloud storage via encryption. During such a situation, there is the prevention of unlawful interception of data, the minimisation of the danger of data breaches, and the guarantee that only those who are authorised with decryption keys are able to access the data.

3. COMMON ENCRYPTION ALGORITHMS USED IN CLOUD STORAGE

The most prevalent types of encryption algorithms are symmetric and asymmetric. There are other types of encryption algorithms as well. Both the encrypting and decrypting processes of symmetric encryption are carried out with the same key. In general, symmetric algorithms are quick and may be utilised in a variety of contexts, such as the encryption of communication lines and financial transactions. Additionally, the Triple Data Encryption Standard, sometimes known as Triple-DES, is a symmetric technique that is widely utilised in the modern world. A more robust method known as Advanced Encryption Standard (AES) is progressively replacing Triple-DES, which is an older technique that is increasingly losing its relevance. Because it employs longer key lengths and higher encryption block sizes, Advanced Encryption Standard (AES) is more robust than Triple-DES [8]. Additional, Advanced Encryption Standard (AES) addresses various design flaws that are present in Triple-DES, which render Triple-DES vulnerable to a variety of assault methods. Within the framework of symmetrical encryption, the process of encrypting and decrypting data involves the division of the data into blocks. These blocks are then introduced into the cypher system in order to get cypher text as an output. There is a tendency for symmetric encryption algorithms to perform well since they are able to handle just tiny data chunks. The fact that symmetrical encryption algorithms are quick is one of the most major advantages they provide. This makes them an excellent choice for encrypting large amounts of data. However, the most significant disadvantage of symmetrical key encryption is the difficulty in determining the most secure method of exchanging the ciphering key with other persons who are participating in the communication channel. What this means is that the ciphertext may be converted into plain text by anybody who possesses the key. It is absolutely necessary for a user to keep the key a secret if they wish to maintain their privacy and confidentiality. The flow path of encrypting and decrypting data is depicted in Figure 1, which can be seen below. Text decryption makes use of the same symmetric key in order to successfully decrypt the data and return it to a format that can be read.

Figure 1 Symmetric Encryption

Both public and private keys are utilised in the process of encrypting and decrypting data when using asymmetric encryption. The private key, on the other hand, is kept a secret until the public key is distributed to the party that requires it. The public key, on the other hand, is accessible to any party that requires it. In order for asymmetric encryption to be effective in encrypting and delivering secrecy, authenticity, and integrity, users and systems need to be certain that the public key is genuine, that its owner is an individual or entity that has been claimed, and that it has not been tampered with by any malevolent third party. There are a variety of asymmetric encryption methods that are often utilised, such as Rivest-Shamir-Adleman (RSA) and Elliptic Curve Cryptography (ECC). Asymmetric keys are the basis of Public Key Infrastructure (PKI), which is a kind of encryption that requires two keys: one (the public key) is used to encrypt the plaintext, while the other (the private key) is used to decrypt the cyphertext. This is illustrated in figure 2 below. As a result, none of the keys are capable of doing both duties simultaneously. An individual who is in need of encrypting a certain piece of information can obtain a public key whenever they want it. Utilising the private key is what the user needs to do in order to decode the same material. Asymmetric encryption has shown to be a useful method in today's culture, giving the benefits of confidentiality, authentication, and integrity to a variety of applications.

Figure 2 Asymmetric Encryption

3.1. THE IMPORTANCE OF DATA ENCRYPTION

For the purpose of ensuring that only authorised individuals are able to access the information, the process of turning plain text into a format that cannot be read (known as ciphertext) is known as data encryption. The secrecy of sensitive data may be protected by the use of encryption, which ensures the use of complicated algorithms and cryptographic keys. Especially in light of the rising number of cyberattacks, this procedure is very necessary for preserving the validity and integrity of the material under consideration. It is possible to have secure digital interactions thanks to encryption, which not only secures personal and financial data but also acts as a protection for intellectual property and trade secrets.

3.2. CHALLENGES IN CLOUD DATA SECURITY

Despite the fact that cloud storage and communication systems have a multitude of benefits, they also present a number of issues that are associated with the protection of data. The prospect of data breaches, in which hackers take advantage of weaknesses in order to get access to sensitive information, is one of the most significant challenges. Another key problem is the shared responsibility model of cloud security, which states that both the user and the service provider are responsible for some aspects of the security of the cloud. Frequently, this results in gaps in the implementation. Further, there is a continuing worry regarding the maintenance of data integrity throughout the storage and transmission processes. Regulators like as the General Data Protection Regulation (GDPR) and the Health Insurance Portability and Accountability Act (HIPAA) require strong encryption standards. Additionally, insider threats from hostile workers provide a unique difficulty to the process of maintaining safe systems.

4. MATERIALS AND METHODS

Storage services that are scalable are made possible by cloud service providers. In this scenario, one of the user's private keys is taken into consideration as an authentication for the public auditing solution. Using the public key of each block, the integrity of the data is examined and validated. When it comes to such models, privacy considerations present a hurdle. An innovative ECRM framework is proposed in this research with the purpose of ensuring data integrity and controlling user access. As demonstrated in Algorithms 1 and 2, the strategy contributes to an increase in the effectiveness of the cloud-based data storage and retrieval processes. Figure 3 illustrates the structure that will be utilised by the suggested security model.

Figure 3 Proposed ECRM framework.

At the outset, the system determines who the group leader is and who the members are in order to save and retrieve their data (Figure 1). Access to user allocation, revocation, and count control is granted to the group leader, who is also in charge of the group. It will be informed to the group leader to join the group so they may view the data on the cloud. An authorised user receives them from the head, and they are all given keys and digital signatures. The auditing database is responsible for keeping the secret key. After acquiring and validating the digital signature, the user is authorised access to the cloud. In the cloud, the data that is accessed will be stored in the correct forms for future retrieval. Encouraging access to encrypted data is a safeguard against data corruption. The auditing group double-checks these. Cloud security is prioritised by the model through the assurance of data integrity. You may limit user access and revocation with the given digital signature. A null value is assigned to an attribute when the user revokes it, blocking further access to the cloud. By plugging in the random number Zp into Equation (1), the data owner may determine whose user Un is being encrypted. According to Equation (2), the data name (DN), extension (EX), and memory space (MS) are all involved.

$$CT_E = m^y \bmod n \quad (1)$$

$$SCT_E = (T', CT_E, DN, EX) \quad (2)$$

Text that has been encrypted and stored in the cloud. Equations (3) and (4) are used to represent the retrieval and decryption processes, respectively.

$$RCT_E = CT_E(T', Q) \quad (3)$$

$$m = CT_E^d \bmod n \quad (4)$$

Algorithm 1 Encryption

1: begin
2: Encrypt(*UK*, *M*, *n*, *e*, *DS*, *IDU*, *RID*, *Km*, *Kp*)
3: $CT_E \leftarrow UK\ (f(M, n, e))$
4: for $i \leftarrow 1$ to n do
5: if [*DSi*, *IDUi*, *RIDi*] == [1, 1, 1] then;
6: $Ks \leftarrow Km \oplus Kp$;
7: Add *Ks* user;
8: Upload CT_E;
9: else
10: Deny;
11: end for
12: Delete *Km*, *Ks*;
13: end
14: $MS \leftarrow (T', C, DN, EX)$
15: end

Algorithm 2 Decryption

1: begin
2: Decrypt (K_s, *Q Kp*, *UK*, CT_E, *n*, *d*, *DN*, T_u)
3: If K_s == 1 then;
4: for each user *i*, do
5: $Km \leftarrow K_s \oplus Kp$;
6: $M \leftarrow UK(f(CT_E, n, d))$;
7: end for
8: else
9: return;
10: end If
11: for $u \leftarrow 1$ to *T* do
12: if *DN* ∈ *Q* then
13: $CT_E(T', Q) \leftarrow 1$;
14: else
15: $C(T', Q) \leftarrow 0$;
16: end if
17: end for
18: if $CT_E(T', Q) == 1$ then;
19: *C* associated with *T'* to the user U_i in response
20: end if
21: end

5. RESULTS

These studies were carried out utilising a Windows 10 operating system and an Intel I3 CPU that has 4 gigabytes of random access memory (RAM). The Eclipse platform and Java Pairing-Based Cryptography (JPBC) were utilised in order to carry out cryptographic operations for the purpose of determining the effectiveness of the system. Download, upload, encryption, and decryption times were the metrics that were utilised in order to assess the efficacy of the ECRM

architecture that was suggested. The length of time that would be required to save a file in the cloud is referred to as the upload time. The time required for retrieval is referred to as the download time. The needed amount of time was determined by calculating the different file sizes for each of the different file extensions. ".txt" and ".jpg" are the two forms of file extensions that are taken into consideration in this research. There is a range of 10 KB to 30 MB for the file sizes that are taken into consideration for technological feasibility. A comparison of the upload time (UT) in milliseconds for files with sizes of 10 kilobytes, 30 kilobytes, 50 kilobytes, 70 kilobytes, 90 kilobytes, 2 megabytes, 15 megabytes, and 30 megabytes is presented in Figure 2. The upload time in milliseconds are as follows: 1, 1, 2, 2, 3, 4, 5, and 6 accordingly. The amount of time required to upload a file rises in proportion to the size of the file. The performance of the proposed system is demonstrated by the fact that there is no noticeable difference in the amount of time it takes to upload a file that is 10 KB in size for 1 millisecond and a file that is 30 MB in size for 6 milliseconds.

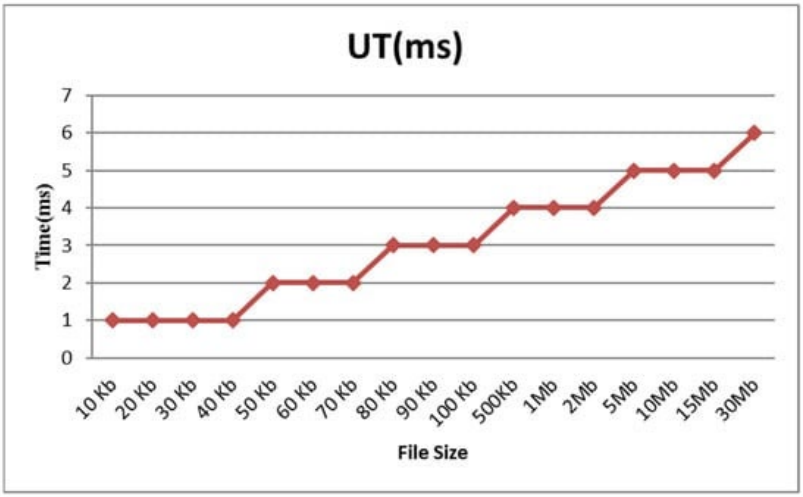

Figure 4 Upload time.

Download time (DT) analysis is depicted in milliseconds in Figure 4, which shows download times of 897 milliseconds, 1067 milliseconds, 1115 milliseconds, 1115 milliseconds, 1781 milliseconds, 1958 milliseconds, and 2045 milliseconds for file sizes of 10 kilobytes, 30 kilobytes, 50 kilobytes, 70 kilobytes, 90 kilobytes, 2 megabytes, 15 Mb, and 35 Mb, respectively. When a file is larger in size, the amount of time it takes to upload it increases proportionally. As a result of the output, it was found that both times rise as the size of the data increases. The performance of the suggested method was demonstrated by the fact that the values were found to be substantially same for files of the same size.

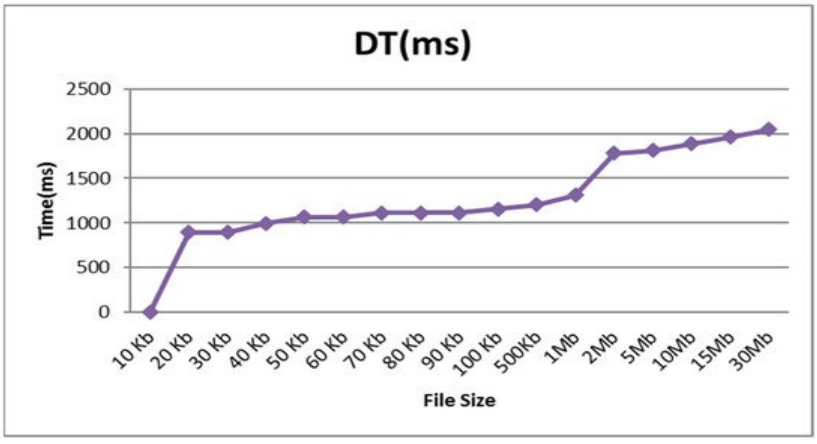

Figure 5 Download time.

An examination of the encryption time (EnT) in milliseconds is presented in Figure 6. The file sizes of 10 kb, 30 kb, 50 kb, 70 kb, 90 kb, 2 Mb, 15 Mb, and 35 Mb are represented by 43, 74, 95, 115, 127, 145, 166, and 234 milliseconds, respectively. The length of time required to encrypt a file is contingent on both its size and its format. The amount of time required for encryption rose in proportion to the quantity of the files. Decryption time, also known as DeT, is depicted in Figure 5 in a similar fashion.

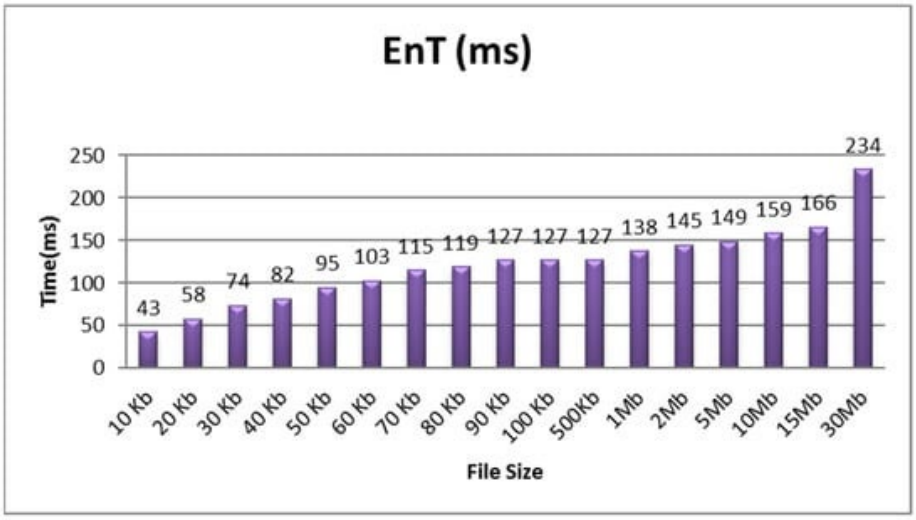

Figure 6 Encryption time.

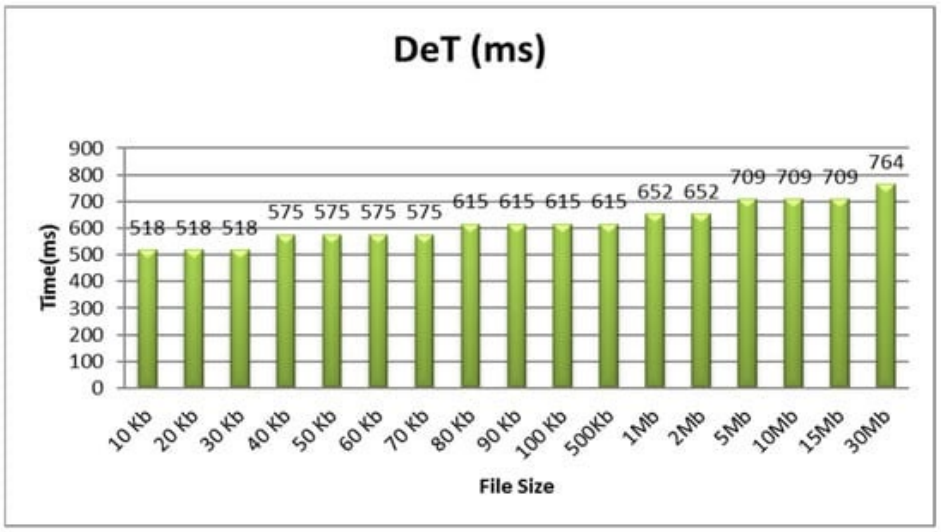

Figure 7 Decryption time.

When it comes to the user of the group, the effectiveness of the strategies that have been offered differs. Furthermore, it is presumed that the number of blocks included inside each file is comparable to that of IPIC-DG. When compared to previous methods, the methodology that has been suggested demonstrates a higher level of efficiency. Every user file is cMulExpG1, which means that the suggested approach has a lower communication cost than the existing technique. This is because the new technique is more efficient. The communication cost, on the other hand, continues to be a problem since it is equal to the sum of the clq and cls values for all of the approaches, as shown in Table 1. As a result, the user whose key was input incorrectly will have their privileges terminated in the event that a file transfer attempts to decrypt using a fake key. Whenever a secure file is being utilised, the signature must be authentic, and the decryption must be identical to the signature. As a result, effective updates are essential for database verification in order to achieve efficient and safe data integrity audits for shared dynamic data in an environment with several users operating simultaneously.

Table 1. Comparison of the proposed with the existing scheme.

Parameters	Existing [5]	Proposed—ECRM	Existing [22]
Signature Efficiency	$((4+m)z+4)\times MulExp_{G_1}$ $+zMulExp_{G_T}+3Pair$	$(4mz)*zMulExp_{G_1}$ $+zMulExp_{G_T}+mPair$	$O(\lambda\beta \log 2\ n)$
Computation Cost	$(c+3)MulExp_{G_1}+cMulExp_{G_T}+2Pair$	$cMulExp_{G1}$	$O(\beta)$
	cl_q+cl_s	cl_q+cl_s	$O(\beta \log n)$
	$ml_q+l_{name}+(m+1)l_{G_1}$	$m(l_q+l_{name}+l_{G1})$	$\beta \log n + O(\lambda \log n)$

6. CONCLUSION

Cloud computing has been increasingly popular in recent years, and its incorporation into essential elements of contemporary life has brought to light the importance of implementing stringent data security protocols. A dependable method of protecting sensitive information from unauthorised access and breaches, encryption stands out as a foundational technology for safeguarding cloud storage and communication. It provides a solid approach to protect sensitive information. The findings of this study shed light on the wide variety of encryption methods that are now accessible. These methods vary from the more conventional symmetric and asymmetric encryption to more sophisticated approaches such as homomorphic encryption and hybrid models. A secure data storage and communication system may be implemented in a variety of cloud settings thanks to the fact that each solution offers its own set of advantages and tackles certain issues. In addition, the research highlights the significance of striking a balance between security, performance, and usability when it comes to the design and implementation of encryption frameworks. While there have been tremendous breakthroughs in encryption in cloud systems, there are still ongoing issues that need to be addressed. These challenges include computational overhead, complexity in key management, and resistance to upcoming threats such as quantum computing. In order to effectively address these concerns, players from academia, business, and government must engage in ongoing innovation and research, as well as develop collaborative strategies. For the purpose of protecting data stored in the cloud, encryption continues to be an indispensable and ever-evolving means of protection. It is possible for organisations to improve data protection, build user trust, and support the continuous expansion of cloud computing in a world that is becoming increasingly linked by exploiting improvements in encryption technology and incorporating them into comprehensive security plans.

CONFLICT OF INTERESTS

None.

ACKNOWLEDGMENTS

None.

REFERENCES

K. R. Dayana and P. S. Rani, "Secure cloud data storage solution with better data accessibility and time efficiency," Journal for Control, Measurement, Electronics, Computing and Communications, vol. 64, no. 4, pp. 756-763, 2023.

A. Reyana, S. Kautish, K. Mohiuddin, F. K. Karim, H. Elmannai, S. Ghorashi and Y. Hamid, "Enhanced cloud storage encryption standard for security in distributed environments," Electronics, vol. 12, no. 3, 2023.

J. Hassan, D. Shehzad, U. Habib, M. U. Aftab, M. Ahmad, R. Kuleev and M. Mazzara, "The Rise of Cloud Computing: Data Protection, Privacy, and Open Research Challenges—A Systematic Literature Review (SLR)," Computational Intelligence and Neuroscience, 2022.

P. Yang, N. Xiong and J. Ren, "Data security and privacy protection for cloud storage: A survey," IEEE Access, vol. 8, pp. 31723-131740, 2020.
M. A. Al-Shabi, "A survey on symmetric and asymmetric cryptography," International Journal of Scientific and Research Publications,, vol. 9, no. 3, 2019.
L. Munn, T. Hristova and L. Magee, "Clouded data: Privacy and the promise of encryption," Big Data & Society, vol. 6, no. 1, 2019.
E. Gelenbe, P. C. Czachorski, S. K. Katsikas, I. Komnios, L. Romano and D. Tzovaras, Security in computer and information sciences, Spriger Open, 2018.
B. White, D. Andre, G. Arquero, R. Bajaj, J. Cronin, A. Dames, H. Lyksborg, A. Miranda and M. Weiss, Transitioning to quantum-safe cryptography on IBM Z, International Business Machines Corporation, 2022.
B. Dufrasne, R. Fridii and A. Greenfield, Data-at-rest encryption for the IBM spectrum accelerate family, International Business Machines Corporation, 2019.
H. Kolivand, S. F. Hamood, S. Asadianfam and M. S. Rahim, "Image encryption techniques: A comprehensive review," Multimedia Tools and Applications, 2024.
Mante, R.V.; Bajad, N.R. A study of searchable and auditable attribute based encryption in cloud. In Proceedings of the 2020 5th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 10–12 June 2020; IEEE: Piscataway, NJ, USA, 2021; pp. 1411–1415.
Vennala, A.; Radha, M.; Rohini, M.; Anees Fathima, M.; Lakshmi, P.D. Efficient Privacy-Preserving Certificateless Public Auditing of Data in Cloud Storage. J. Eng. Sci. 2022, 13, 532–541.
Li, R.; Yang, H.; Wang, X.A.; Yi, Z.; Niu, K. Improved Public Auditing System of Cloud Storage Based on BLS Signature. Secur. Commun. Netw. 2022, 2022, 6800216.
He, J.; Zhang, Z.; Li, M.; Zhu, L.; Hu, J. Provable data integrity of cloud storage service with enhanced security in the internet of things. IEEE Access 2018, 7, 6226–6239.
Rathore, H.; Mohamed, A.; Guizani, M. A survey of blockchain-enabled cyber-physical systems. Sensors 2020, 20, 282.
Chen, Y.; Liu, H.; Wang, B.; Sonompil, B.; Ping, Y.; Zhang, Z. A threshold hybrid encryption method for integrity audit without trusted center. J. Cloud Comput. 2021, 10, 3.
Sajay, K.R.; Babu, S.S.; Vijayalakshmi, Y. Enhancing the security of cloud data using hybrid encryption algorithm. J. Ambient Intell. Humaniz. Comput. 2019, 1–10.
Latha, K.; Sheela, T. Block based data security and data distribution on multi cloud environment. J. Ambient Intell. Humaniz. Comput. 2019, 1–7.
Cha, J.; Singh, S.K.; Kim, T.W.; Park, J.H. Blockchain-empowered cloud architecture based on secret sharing for smart city. J. Inf. Secur. Appl. 2021, 57, 102686.
Viswanath, G.; Krishna, P.V. Hybrid encryption framework for securing big data storage in multi-cloud environment. Evol. Intell. 2021, 14, 691–698.
Xu, Y.; Ding, L.; Cui, J.; Zhong, H.; Yu, J. PP-CSA: A privacy-preserving cloud storage auditing scheme for data sharing. IEEE Syst. J. 2020, 15, 3730–3739.