International Journal of Advanced Science and Computer Applications (2024); 3(1): 31-42
DOI: 10.47679/ijasca.v3i1.46
https://ijasca.org/index.php/ijasca

RESEARCH ARTICLE

Implementation of Big Data Cloud Computing Security: Systematic Review from Different Encryption Techniques Perspective

Alsadig Mohammed Adam Abdallah[1], Amir Mohamed Talib[1], Mandour Mohamed Ibrahim[1], Elhadi Suliman AbdElgader[1]

Abstract

The volume of Big Data produced from various sources is always growing. Such vast amounts of data cannot be stored using conventional techniques. As a result, the majority of businesses have switched to using cloud storage as a substitute for storing Big Data. Cloud storage has made great progress, but there are still several problems, such security worries. This article examines Big Data, its difficulties, and many security concerns from the perspectives of various encryption approaches. Additionally, it presents various views to address these issues and suggests a new classification of Big Data security difficulties.

Keyword: Big Data, Cloud Computing and Security

Introduction

Big Data

Big data primarily refers to data sets that are too large or complex to be dealt with by traditional data-processing application software. Greater statistical power is offered by data with more entries (rows), yet data with more attributes (or columns) may have a higher false discovery rate. The definition of big data that seems to best represent it is the one linked with a massive body of information that we could not understand when utilized just in smaller amounts, despite the fact that it is sometimes used imprecisely in part due to the lack of a formal definition [1].

Cloud computing

Cloud computing is the on-demand availability of computer system resources, in particular data storage (cloud storage) and computational power, without direct active management by the user, is known as cloud computing. Large clouds commonly have uses for distributed data centers. Pay-as-you-go models are frequently used in cloud computing, which can save capital costs but also expose users to unforeseen running costs. Coherence in cloud computing depends on resource sharing. [2].

Encryption

The process of converting information into a secret code that conceals its true meaning is known as encryption. Cryptography is the study of information encryption and decryption. In computers, plaintext is another name for unencrypted data, whereas cipher text is another name for encrypted data [3].

[1] College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh 11432, Kingdom of Saudi Arabia (KSA)

*) *corresponding author*

Alsadig Mohammed Adam Abdallah

Email: sadigmo86@gmail.com

2.1 Big Data Lifecycle

Big Data Lifecycle has six cycles as illustrated in the Figure 1 below

Phase 1: Data Discovery and Formation

Every action begins with a goal in mind. You will determine your data's goal in this phase and how to accomplish it following the data analytics life cycle. This initial phase's goal is to carry out evaluations *and* assessments

Received 8 November 2023 | Received in revised form 28 December 2024 | Accepted 21 January 2024
Available online 20 March 2024

in order to create a core hypothesis for addressing any business issues and concerns [4].

Phase 2: Data Preparation and Processing

Gathering, processing, and purifying the obtained data are all part of the data preparation and processing phase. Making sure the data you need is available for processing is one of the step's most crucial components [4].

Phase 3: Design a Model

It's time to build a model that leverages the data to accomplish the goal when you've set your company goals and acquired a sizable amount of (formatted, unformatted, or semi-formatted) data. This phase of the data analytics process is known as model planning.

Phase 4: Model Building

The creation of data sets for testing, training, and production is a part of this stage of the data analytics life cycle. The model that the data analytics experts designed in the earlier stage is developed and used properly.

Phase 5: Result Communication and Publication

Remember the goal you had for your business in Phase 1. Check now to determine if the tests you run in the earlier step matched those requirements.

Phase 6: Measuring Effectiveness

The last step in your data analytics life cycle is to provide stakeholders with a comprehensive report that contains key findings, code, briefings, and technical papers/documents.

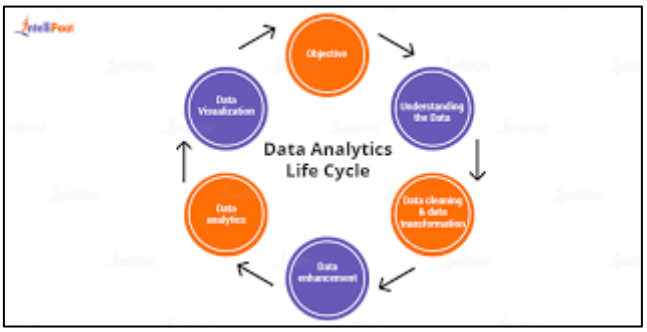

Figure 1. Big Data Analytics Lifecycle

3.1 Cloud Computing Characteristics

1. On-Demand Self-Service

You may automatically provision computer resources, such as server time and network storage, with cloud computing. There is no need for you to communicate with the service provider. Customers of cloud services can view their cloud services, track their usage, and provision and de-provision services by logging into their cloud accounts through a web self-service portal [5].

2. Broad Network Access

Broad network connectivity is another crucial aspect of cloud computing. Cloud services are accessible across a network and on portable devices including laptops, desktop computers, tablets, and cell phones. A private cloud employs a local area network, whereas a public cloud uses the internet. Due to their effects on service quality, latency and bandwidth both have a significant impact on cloud computing and widespread network access [5].

3. Resource Pooling

Using a multi-tenant approach, resource pooling enables numerous customers to share physical resources. Based on demand, this model distributes and redistributes real and virtual resources. Customers can use the same applications or infrastructure while yet retaining security thanks to multi-tenancy. Customers may be able to designate the location of their resources at a higher level of abstraction, such as a country, state, or data center, even though they won't know the precise location of their resources. Customers can pool a variety of resources, including memory, computing power, and bandwidth.

4. Rapid Elasticity

Customers can scale swiftly based on demand thanks to the elastic provisioning and releasing capabilities of cloud services. There are essentially no limits on the capabilities that can be provisioned. Customers can use these features whenever they want and in whatever amount. Customers can scale cloud capacity, cost, and usage without incurring additional contracts or charges. You won't need to acquire computer hardware thanks to quick elasticity. employ the cloud computing resources provided by the cloud provider instead.

5. Measured Service

A metering capability in cloud systems optimizes resource utilization at an abstraction level appropriate for the type of service. For storage, processing, bandwidth, and users, for instance, you can utilize a measured service. A pay-for-what-you-use model is used to base payments on the customer's actual consumption. Consumers and service providers benefit from a transparent experience that is created by monitoring, managing, and reporting resource use.

6. Resiliency and Availability

In cloud computing, resilience refers to a service's capacity to bounce back rapidly from any interruption. The speed at which a cloud's servers, databases, and networks restart and recover after any damage is used to gauge its resilience. Cloud services make a copy of the data they save in order to prevent data loss. The copied version from the other server restores any data lost on one server for whatever cause.

Another important topic in cloud computing is availability. The advantage of using cloud resources is that there are no geographical limits because you can access them anywhere.

7. Flexibility

As a company's business expands, scaling is necessary. Customers have more flexibility to scale as they see fit in the cloud without having to restart the server. Additionally, consumers can select from a variety of payment methods to prevent splurging on resources they won't use.

8. Remote Work

Users working remotely benefit from cloud computing. Remote employees can use their devices, such as laptops and cellphones, to securely and swiftly access business data. The cloud enables effective communication between coworkers and job performance for remote workers.

3.2 Cloud Computing Service Models

Cloud Computing Service Models illustrated in the Figure 2 below [6]:

3.2.1 Infrastructure as a service (IaaS)

IaaS is a self-service paradigm for controlling the infrastructures of distant data centers. Through the use of the Internet, IaaS offers virtualized computer resources that are hosted by a third party, such as Google, Amazon Web Services, or Microsoft Azure.

Companies buy IaaS based on a consumption model rather than buying hardware. Similar to purchasing electricity. Only what you use is charged. Companies may quickly add, remove, or reorganize IT infrastructure thanks to this paradigm.

Because they are more familiar with IaaS and have years of experience with virtual environments or stringent security and regulatory requirements that can only be satisfied through IaaS, many IT businesses rely on it [6].

3.2.2 Platform as a service (PaaS)

Organizations may create, run, and manage apps without the need for IT infrastructure thanks to platform as a service (PaaS). Application development, testing, and deployment are made simpler and faster as a result [6].

Developers don't have to worry about time-consuming IT infrastructure tasks like setting up servers, storage, and backups so they can concentrate on developing code and building applications.

PaaS enhances cloud functionality. It can lower your expenses and lower your managerial overhead. You can innovate and scale your services on demand more easily with PaaS.

3.2.3 Software as a service (SaaS)

Software that is licensed on a subscription basis is used in place of conventional on-device software in software as a service (SaaS). It is hosted centrally on the cloud. Salesforce.com is a prime illustration.

The majority of SaaS applications are directly accessible via a web browser without the need for downloads or installations. Plugins are necessary for some SaaS applications, though.

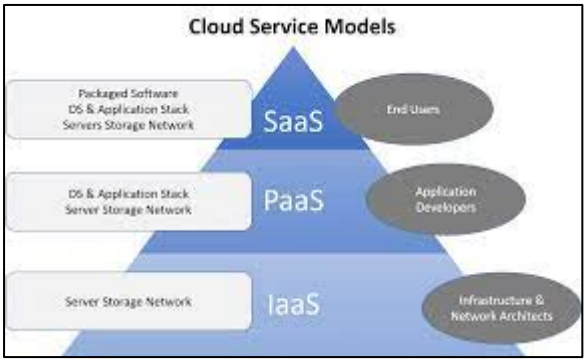

Figure 2. Cloud Computing Service Models

3.3 Cloud Computing Deployment Models

What is the cloud?

Cloud Computing is getting a lot of attention, and if you plan to use the cloud for personal use, you'll need a distinct cloud type and service.

As shown in Figure 3, there are three different cloud computing deployment models.

3.3.1 Public

Any subscriber with an internet connection and access to the cloud space can use a public cloud.

3.3.2 Private

Private Cloud - A private cloud is created for a certain group or organization and only allows members of that group to access it.

3.3.3 Hybrid

A hybrid cloud is simply a fusion of at least two clouds, where the clouds combined are a mix of community, private, and public clouds [7].

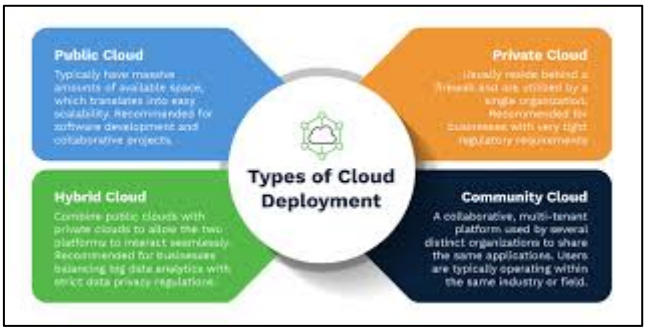

Figure 3. Cloud Computing Deployment Models

4.1 Encryption Techniques based on Big Data Cloud Computing Security Challenges

The following encryption methods are described in depth for cloud computing data security and can aid in strategic decision-making to enhance data offloading to cloud computing resources.

Blowfish with Compressed File

Users of cloud services can access pooled resources without spending much money or worrying about server upkeep. The main worry when implementing cloud computing technologies is data security. Data should be encrypted and compressed before being stored on cloud storage resources, according to Grover et al.'s [8] recommendation for Blowfish. This will save storage space. Figure 4 illustrates this process graphically. The file is first compressed, and then it is encrypted using the Blowfish technique

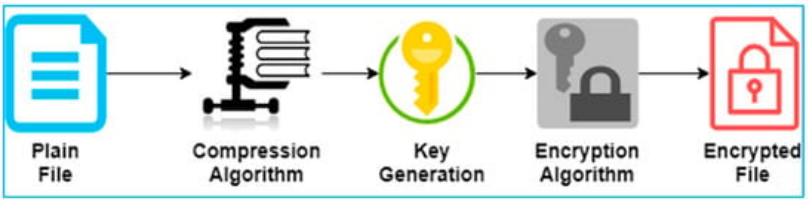

Figure 4. Blowfish encryption process [4].

By compressing the data file, the suggested solution addresses the problem of reducing encryption time and space. Three distinct file sizes are used to gauge how quickly the suggested method works. The encryption times for non-compressed and compressed files are then calculated to compare the outcomes. This method can reduce the amount of time it takes to encrypt a file and the amount of storage space needed to store it if the file is compressed before encryption. However, its performance for greater file sizes is not covered. Key management will become an increasingly difficult problem as the number of users rises.

The client-side application of the Blowfish encryption method for data security reduces server-side processing time. The drawback of this strategy is that it compromises performance because compression and encryption take longer as file sizes grow.

4.1.1 RSA with AES

The encryption methods utilized by Khanezaei et al. [9] were slightly different; they combined the Rivest Shamir Adelman (RSA) and Advanced Encryption Standard (AES) algorithms. Data sharing between users is more safe when using the double encryption approach. AES is used for quick data retrieval, whereas RSA is used to make the encrypted information more difficult. The generation of asymmetric keys, which takes time, makes using RSA for file transfers secure.

The cloud service generates a public key (PB), a private key (PK), file ID, and a big random number (RB). At first, the user requests the PB from the cloud. The cloud system provides the PB and file ID to the user. The user then sends the file encrypted using RSA to the cloud. When the user requests a specific file from the cloud service, it sends a request to the server with the public key. The cloud service finds the requested file in the Cloud Storage System (CSS). This file is then encrypted using the AES. The RB, which is the secret key of a symmetric algorithm, is encrypted using the public key. The user receives the RB and requested file from the CSS after that. Due to the key distribution issue, the symmetric technique is employed; however, the topic of key management is not covered.

The advantage of this strategy is that it employs both RSA and AES encryption techniques to provide data protection. The inclusion of RSA makes data more difficult to hack, while the use of AES speeds up file transfers between the user and cloud storage.

The main problem with the suggested method is that as file size grows, more keys are generated, which creates a problem with key management. Encryption and decryption time being a burden for huge file volumes is another flaw.

4.1.2 Advanced Encryption Standard

As cloud computing technology develops, cloud service providers must address concerns about data security and confidentiality. Data is encrypted before being sent to the cloud using the AES data security technique, as utilized by Sachdev et al. [10].

Both in software and hardware contexts, on 8-bit and 64-bit platforms, the AES performs better. AES is excellent for areas with limited space since it takes less memory. The size of the keys for multiples of 32 must be greater than 128 and AES keys are simple to set up and support any block.

Data are encrypted using the AES technique at the user end before being delivered to the cloud service provider. Since the user controls the data and key, data integrity is guaranteed. AES uses less memory and takes less time to process than other encryption methods. This method recommends setting up a separate physical server at the user end to maintain the keys, which raises the hardware cost.

The AES approach's strength is that users, not cloud service providers, handle data encryption and decryption.

4.1.3 Fully Homomorphic Encryption

Zhao et al.'s Fully Homomorphic Encryption method [11] enables users to do calculations on encrypted material without having to first decrypt it. As a result, this method increases the data's security by allowing the user to edit ciphertext without disclosing the original data to the cloud.

Analyzing the output of the fully homomorphic encryption approach only requires addition and multiplication. The ciphertext is multiplied and added to, and the results are then contrasted with the plain text. In comparison to plain text, ciphertext requires more complicated calculation. As a result, such a difficult computation will take a long time for a tiny set of data. The user logs in and then chooses the storage option based on the level of data security. The AES algorithm is utilized for encryption if the private section is chosen, but the Blowfish encryption method is used for the public area. For hybrid data storage, which offers security in either the private or public sector, two data encryption types are available. If low-level security is necessary, the Simplified Advanced Encryption Standard (S-AES), International Data Encryption Algorithm (IDEA), or Blowfish encryption approach can be chosen. IDEA is employed after Blowfish encryption has been performed for high-level security. After a file is encrypted, the Secure Hashing Algorithm (SHA-1) is employed to generate the integrity code. This code is put to the beginning of the encrypted file, and SHA-1 is once more used to create a 16-digit alphanumeric token. To maintain the integrity of the data, the SHA-1 token provided at the time of retrieval is compared with the one provided after the user has been authenticated in order to retrieve the file.

The benefit of the Fully Homomorphic Encryption technique is that it uses a variety of encryption algorithms with integrity verification schemes to guarantee security in public, private, and hybrid storage sections. Depending on the needs for security, the user chooses the cloud storage area. Using the AES algorithm, the private area offers the maximum level of protection. There isn't much protection offered in the public area. If a user requires quick processing and little encryption and decryption time, this part will work best for them.

4.1.4 Hybrid Techniques with Secure Endpoint

In order to protect the users' private information and enable ciphertext retrieval, Rani et al. [12] employed public and private key encryption algorithms. When logging into the cloud system, security is provided via the hybrid approach. This method uses various encryption techniques to encrypt the password while providing the username in plain text. To verify the user's legitimacy, the username and password are then matched to those that have already been saved.

The login and password that are saved in the cloud will be used to identify any person who has tempered with the data, which can speed up the process of determining the root cause of data tempering. The password is encrypted using Caesar cipher, while the username is displayed in plain text. The RSA technique is then used to re-encrypt the encrypted output. Then monoalphabetic substitution is used to encrypt this result. The authors recommend encrypting passwords three times to boost security, although doing so also triples the encryption time, as illustrated in Figure 6.

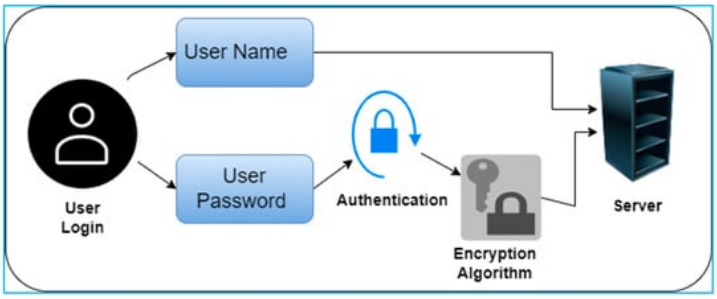

Figure 6. Flow architecture of Hybrid Techniques with Secure Endpoint [13].

The hybrid technique's advantage is that it encrypts the message using a combination of the Ceaser cipher, RSA, and monoalphabetic substitution. The main disadvantage of the suggested method is that it takes longer to encrypt passwords because it requires multiple encryption methods. When compared to the previously encrypted result, the use of several encryption algorithms increases computational complexity as well.

4.1.5 Three-Layer Security Structure

The three-layer security structure method, used by the Amazon EC2 cloud service provider, is explained in [14]. In order to offer security at various levels, various encryption techniques are used. When a short encryption time performance of the security method is desired rather than improved data security, the Blowfish or DES algorithms are used. AES is preferable, but, when very high data security is needed.

The current working paradigm is three-layered centric, with data security falling under each tier. Data security is built into the model layer by layer. As a case study, Amazon EC2 is employed. Based on encryption time and speed and the cloud infrastructure, the best technique is chosen. AES is best suited for Amazon EC2 due to its high level of security and quick encryption.

When performance and speed are necessary, the Three-layer Security Structure performs well. This method offers a data security model based on the host cloud architecture, which is a salient characteristic.

4.1.6 RSA with MD5 Hash Algorithm

The authors of [15] provided a Java platform-based data security environment. The finest feature of the suggested method is that the client environment will have to grant permission before the cloud administrator may read or alter data. The user similarly requests authorization from the cloud to read and change. While being uploaded via RSA, the data is encrypted using MD5.

In a cloud environment, the user is identified for updating by supplying a secure key, which is delivered with a tag during data upload. The cloud will be alerted and deny access to the data if it recognizes the altered tag.

The working procedure of the proposed technique addresses the trust issues when operating in the cloud by transferring the data via APIs, which are always encrypted, alleviating the concerns of insecure data transfer. Data loss concern is also addressed since the cloud admin cannot modify data. The main feature of the proposed technique is the implementation of double security measures with RSA and message digest, which ensures data integrity and security.

If the MD5 hash matches, the cloud administrator can decode the data but not change it. The suggested method's administrative burden is the management of public-private key pairs used to encrypt data at the cloud user level and decrypt it at the cloud administrator level. Every time there is a data breach, a new private-public key pair must be generated.

4.1.7 Security-Aware Efficient Distributed Storage Technique (SA-EDS)

The SA-EDS model [16] consists of two components; the first component is the verified Deterministic Process (DP), and the second is the Data Distributed Storage Process (D2SP).

Data is protected from immoral acts via D2SP, while the DP confirms the data's high level of security. The original data is divided into two portions before being sent to the cloud. The Alternative Data Distribution (AD2), Secure Efficient Data Distribution (SED2), and Efficient Data Conflation (EDCon) methods are used to encrypt both pieces.

Data packets are divided and then retrieved in the two steps of the suggested model's operation. The plain text is separated into smaller sections in the splitting step based on the level of information security. The sensitive portion is saved on server A after encryption, whereas the other portion is stored on server B. As seen in Figure 7, when a user needs to access data from cloud servers, they receive the ciphertext and XOR it with the associated key to create the original text.

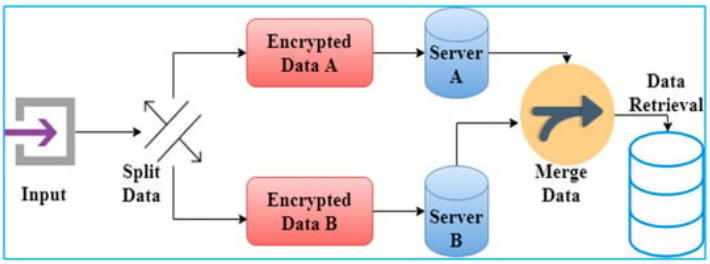

Figure 7. Flow steps of SA-EDS [14].

This model's strength is its ability to partition data into two parts and send each component to a different cloud server. The difficulty of security is raised when multiple servers are used. The model's downside is an increase in the time required to retrieve data from servers and convert it back to its original format.

4.1.8 Centralized Key Management System (CKMS) with RSA Encryption Algorithm

The functionality to generate, store, and distribute the key among two parties is covered by the encryption/decryption key management. K. V. Pradeep et al. employed the RSA encryption technique with the Centralized Key Management System (CKMS) in [17]. In this method, keys are distributed using modified Diffie-Hellman distribution algorithms, and the data is then encrypted using those same keys. The linked file owner's private key is utilized by the CKMS to encrypt the file before it is uploaded to the cloud server. The file is then encrypted after that. The Certificate Authority (CA) is used in every single CKMS operation. The server name, certificate directory, registration authority, and key generator are all verified by the CA. Figure 8 illustrates how the public key is split into two parts using CKMS; one half is sent to the user and the other portion is stored by CKMS. The file is decrypted using the entire key

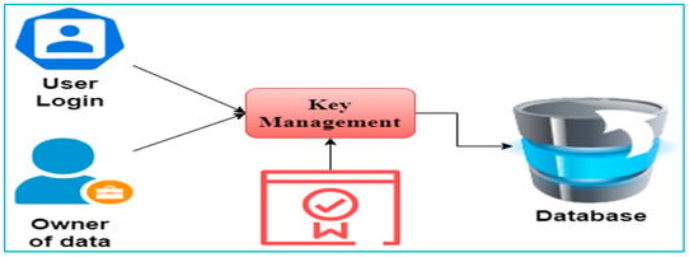

Figure 8. CKMS working model [15].

The benefit of this approach is that it offers safe cloud server data storage. The registration process makes it difficult to access.

4.1.9 Triple Level Encryption

Using triple-level encryption, Chandrika et al. suggested a solution to the security issues with cloud computing in [16]. They use many encryption algorithms to encrypt a file in order to ensure maximum security at each level. When uploading files, the DES algorithm is utilized to implement first-level encryption. The second level is then achieved using the AES method, and the third level uses RSA encryption. The ciphertext is then saved in a database, as depicted in Figure 9. The first level's ciphertext is first decrypted using RSA, while the second level's ciphertext is decrypted using AES. The final level of decryption then uses DES to restore the original plain text.

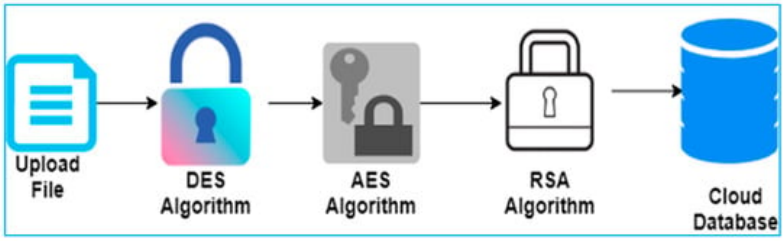

Figure 9. Flow Steps in [16].

This method's increasing temporal complexity and lengthy file access times are its key weaknesses.

4.1.10 Encryption with Secure HTTPS Connection

The application, storage, and networking parts make up the cloud computing architecture. Each category provides various goods and services to companies all over the world. This method ensures security at both the architectural and application levels. When the user presses the upload button, the file is transmitted via an encrypted connection to the cipher cloud. Using HTTPS protects the encrypted connection. As seen in Figure 10 [18], the user-selected symmetric key algorithm is used to carry out the double encryption

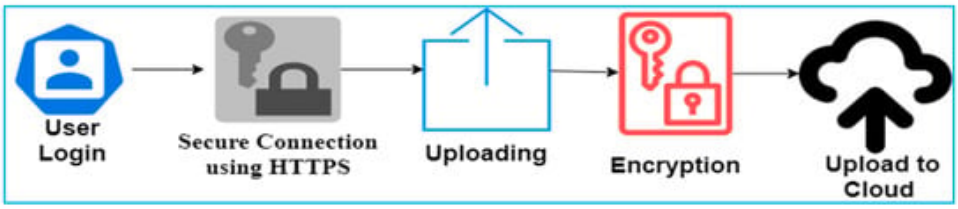

Figure 10. Flow steps of encrypting data with secure HTTPS connection [19].

The appeal of this method is that the user choose the encryption algorithm on their own in accordance with their needs.

4.1.11 AWS Logins with Homomorphic Encryption

In [20], Poteya et al. suggested a method for encrypting data using a homomorphic algorithm. To protect user data, this version is used with DynamoDB on the AWS public cloud. The user enters their credentials and logs in in accordance with the operation criteria. After validating the user's information using the key selection component kept in the database, the AWS cloud database provides services for storing and accessing user data through the login module. Through the use of the encryption and decryption components, the data is saved in an encrypted manner and then retrieved. The AWS computation component manipulates user data in accordance with their needs or the query that was issued. Because no data is ever exposed in plain text throughout any phase of the proposed technique, it offers confidentiality to the data.

Double Encryption Model

Prior to uploading to the cloud, the double encryption technique encrypts the data twice to increase data security. The RSA-1024 algorithm is used to encrypt the AES key after plain text has been encrypted using the AES technique according to [21]. Figure 11 depicts the twofold encryption scheme's operational process.

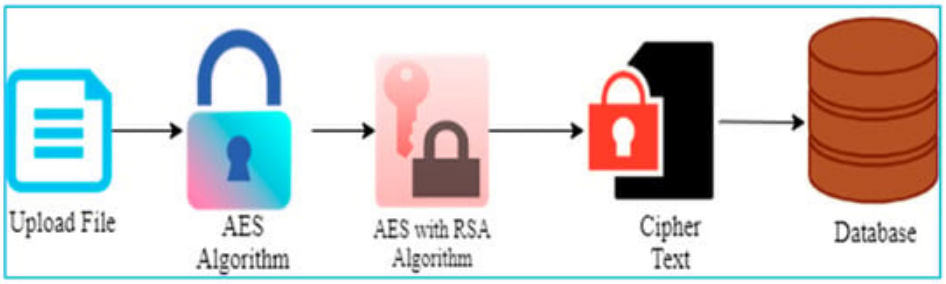

Figure 11. Double encryption model.

This method's strength is that RSA assures excellent security while AES reduces the time complexity of file exchange. Key management and encryption/decryption overhead for the huge files are this method's weaknesses.

4.1.12 Hybrid Encryption with MD5 Hash Function

The hybrid approach, which combines symmetric and asymmetric algorithms with a hashing key using the MD5 hashing function, was introduced by Salma D. et al. in [22]. The entire plain text is split into n blocks for the encryption process, and each block is then split into two parts. One portion of each block is encrypted using the AES technique, and the other with the Blowfish algorithm. The outcome is a single block of text. Figure 12 illustrates how the MD5 hashing function is used to hash the encryption key. Similar to the encryption phase, the decryption phase begins with the entire ciphertext being divided into n blocks, with each block then being further divided into two pieces. Each component uses the specific algorithm it used for the encryption phase along with a hashed key to decrypt the ciphertext. The size of the ciphertext, time required for encryption and decryption, and throughput can all be used to gauge how effective the hybrid approach is.

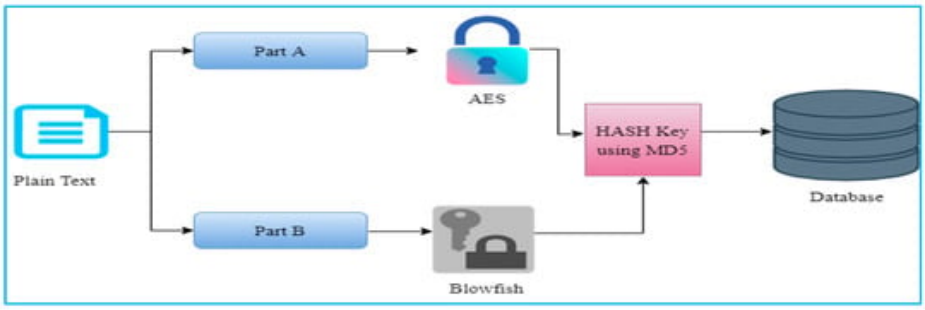

Figure 12. Hybrid encryption process with MD5 hash function [20].

4.1.13 Homomorphic Encryption with Multi-party Computation

To protect the confidentiality and integrity of the data in the cloud environment, the authors of [23] suggested an encryption technique that combines multi-party computation and homomorphic encryption techniques. The suggested approach operates in three steps: Multi-party Computation, Homomorphic Encryption (HE), Key Generation, Encryption, and Decryption. This method stands out for integrating homomorphic encryption with multi-party computation (HE+MPC), which results in less overhead while maintaining the secrecy and integrity of the data.

4.1.15 RSA with Partitioned File

The plain text was divided into blocks of the same size by RSA using the partitioned file approach. The RSA technique is used to independently encrypt each block. Then, as illustrated in Figure 13, each encrypted block is distributed independently in the cloud server.

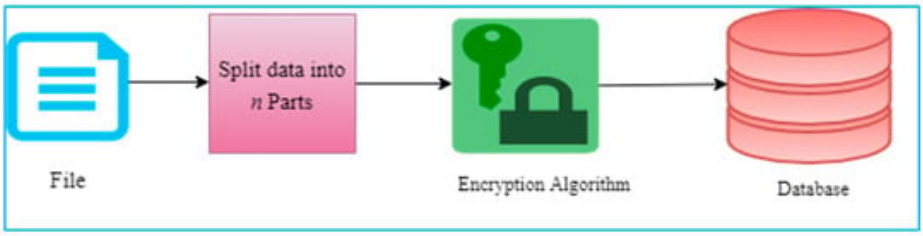

Figure 13. RSA with partitioned file technique [22].

The administration of the different blocks' keys is the only issue with this method.

4.1.16 Optimization-Based Encryption

Utilizing optimization techniques, optimization-based encryption [24] secures the data offloading to the cloud computing system. The Best-Fit Decrease Algorithm (BFD), Genetic Algorithm (GA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Parallel Particle Swarm Optimization (PPSO), and these work on a variety of intelligent methods. The aforementioned algorithms are employed to enhance outcomes and utilize the effectiveness of system resources in the cloud.

4.1.17 DNA-Based Encryption

The idea of using DNA-based encryption, which relies on the biological concept of DNA to protect massive data in the cloud, was put forth in [25]. This method produces a 1024-bit secret key based on DNA computing, user characteristics, and the user's Media Access Control (MAC) address, Supplementary and Decimal Encoding Rule, and the American Standard Code for Information Interchange (ASCII) value. The system can defend itself against numerous security threats using this technique.

4.1.18 Fully Homomorphic Encryption with Advanced Performance

In [26], a paradigm for evaluating the precision and performance of big data was presented. Fully homomorphic encryption (FHE), a potent and newly developed encryption method, has been utilized to execute analytical operations on encrypted data.

4.1.19 Dynamic Security Properties Monitoring Architecture

The idea of virtualization architecture serves as the foundation for the dynamic security strategy in [27]. This architecture's main goal is to find various risks in the instrumentation of virtualized systems. The main goal of this monitoring strategy is to combat device driver attacks and keep an eye on running applications.

Conclusion

Big Data are related with various issues, the most significant of which is security, due to the enormous expansion of data. To effectively address these issues and enhance Big Data management systems, it is necessary to have more discussions about these concerns.

Traditional approaches were insufficient for Big Data and they did not provide a comprehensive solution to address security issues. This paper presented the most important challenges associated with Big Data Cloud Computing, focusing particularly on the security challenges and the most important security requirements. In order to address these issues and benefit from various encryption techniques, researchers and Big Data stakeholders need carry out additional research and study.

References

S. Yin and O. Kaynak, "Big data for modern industry: challenges and trends [point of view]," Proceedings of the IEEE, vol. 103, pp. 143-146, 2015. http://dx.doi.org/10.1109/JPROC.2015.2388958

K. Sarkar and P. P. Deb, "A Review on Trends of Cloud Computing for Autonomous Vehicle," AIJR Abstracts, p. 20, 2022.

A. Manideep, C. N. V. Reddy, B. Srujana, and S. S. H. Raju, "Information Hiding Secretly Using Image Steganography," International Journal of Research in Engineering, Science and Management, vol. 6, pp. 17-20, 2023.

J. B. Guinee, Handbook on life cycle assessment: operational guide to the ISO standards vol. 7: Springer Science & Business Media, 2002.

P. Mell and T. Grance, "The NIST definition of cloud computing," 2011.

F. Marozzo, "Infrastructures for High-Performance Computing: Cloud Infrastructures," 2019. http://dx.doi.org/10.1016/B978-0-12-809633-8.20374-9

A. Huth and J. Cebula, "The basics of cloud computing," United States Computer, pp. 1-4, 2011.

N. Khanezaei and Z. M. Hanapi, "A framework based on RSA and AES encryption algorithms for cloud computing services," in 2014 IEEE conference on systems, process and control (ICSPC 2014), 2014, pp. 588. http://dx.doi.org/10.1109/SPC.2014.708830

A. Sachdev and M. Bhansali, "Enhancing cloud computing security using AES algorithm," International Journal of Computer Applications, vol. 67, 2013. http://dx.doi.org/10.5120/11422-6766

F. Zhao, C. Li, and C. F. Liu, "A cloud computing security solution based on fully homomorphic encryption," in 16th international conference on advanced communication technology, 2014, pp. 485-488. http://dx.doi.org/10.1109/ICACT.2014.6779008

R. Kaur and R. P. Singh, "Enhanced cloud computing security and integrity verification via novel encryption techniques," in 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2014, pp. 1227-1233. http://dx.doi.org/10.1109/ICACCI.2014.6968328

E. M. Mohamed, H. S. Abdelkader, and S. El-Etriby, "Enhanced data security model for cloud computing," in 2012 8th International Conference on Informatics and Systems (INFOS), 2012, pp. CC-12-CC-17.

A. K. Dubey, A. K. Dubey, M. Namdev, and S. S. Shrivastava, "Cloud-user security based on RSA and MD5 algorithm for resource attestation and sharing in java environment," in 2012 CSI Sixth International Conference on Software Engineering (CONSEG), 2012, pp. 1-8.

Y. Li, K. Gai, L. Qiu, M. Qiu, and H. Zhao, "Intelligent cryptography approach for secure distributed big data storage in cloud computing," Information Sciences, vol. 387, pp. 103-115, 2017. http://dx.doi.org/10.1016/j.ins.2016.09.005

K. V. Pradeep, V. Vijayakumar, and V. Subramaniyaswamy, "An efficient framework for sharing a file in a secure manner using asymmetric key distribution management in cloud environment," Journal of Computer Networks and Communications, vol. 2019, 2019.

M. Kaur and R. Singh, "Implementing encryption algorithms to enhance data security of cloud in cloud computing," International Journal of Computer Applications, vol. 70, 2013. http://dx.doi.org/10.5120/12167-8127

M. M. P. Mr, C. A. Dhote, and D. H. S. Mr, "Homomorphic encryption for security of cloud data," Procedia Computer Science, vol. 79, pp. 175-181, 2016.

Z. Kartit, A. Azougaghe, H. Kamal Idrissi, M. El Marraki, M. Hedabou, M. Belkasmi, and A. Kartit, "Applying encryption algorithm for data security in cloud storage," in Advances in Ubiquitous Networking: Proceedings of the UNetâ€™15 1, 2016, pp. 141-154.

http://dx.doi.org/10.1007/978-981-287-990-5_12

D. S. AbdElminaam, "Improving the security of cloud computing by building new hybrid cryptography algorithms," International Journal of Electronics and Information Engineering, vol. 8, pp. 40-48, 2018.

D. Das, "Secure cloud computing algorithm using homomorphic encryption and multi-party computation," in 2018 International Conference on Information Networking (ICOIN), 2018, pp. 391-396.

http://dx.doi.org/10.1109/ICOIN.2018.8343147

D. Hyseni, A. Luma, B. Selimi, and B. Cico, "The proposed model to increase security of sensitive data in cloud computing," Int. J. Adv. Comput. Sci. Appl, vol. 9, pp. 203-210, 2018.

http://dx.doi.org/10.14569/IJACSA.2018.090229

M. B. Qureshi, M. S. Qureshi, S. Tahir, A. Anwar, S. Hussain, M. Uddin, and C.-L. Chen, "Encryption Techniques for Smart Systems Data Security Offloaded to the Cloud," Symmetry, vol. 14, p. 695, 2022.

http://dx.doi.org/10.3390/sym14040695

S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in 2010 Proceedings IEEE INFOCOM, 2010, pp. 1-9.

http://dx.doi.org/10.1109/INFCOM.2010.5462174

B. R. Purushothama and B. B. Amberker, "Efficient query processing on outsourced encrypted data in cloud with privacy preservation," in 2012 International Symposium on Cloud and Services Computing, 2012, pp. 88-95.

http://dx.doi.org/10.1109/ISCOS.2012.16

M. Tebaa, S. d. El Hajji, and A. El Ghazi, "Homomorphic encryption method applied to Cloud Computing," in 2012 National Days of Network Security and Systems, 2012, pp. 86-89.

http://dx.doi.org/10.1109/JNS2.2012.6249248

B. T. Rao, "A study on data storage security issues in cloud computing," Procedia Computer Science, vol. 92, pp. 128-135, 2016.

G. Wang, Q. Liu, and J. Wu, "Achieving fine-grained access control for secure data sharing on cloud servers," Concurrency and Computation: Practice and Experience, vol. 23, pp. 1443-1464, 2011.

http://dx.doi.org/10.1002/cpe.1698