Computer Systems Science & Engineering
DOI:10.32604/csse.2022.020071
Article

Tech Science Press

Secured Data Storage Using Deduplication in Cloud Computing Based on Elliptic Curve Cryptography

N. Niyaz Ahamed[*] and N. Duraipandian

Department of Computer Science and Engineering, Velammal Engineering College, Chennai, 600066, India
*Corresponding Author: N. Niyaz Ahamed. Email: nyzamdmails@gmail.com

Received: 08 May 2021; Accepted: 17 June 2021

Abstract: The tremendous development of cloud computing with related technologies is an unexpected one. However, centralized cloud storage faces few challenges such as latency, storage, and packet drop in the network. Cloud storage gets more attention due to its huge data storage and ensures the security of secret information. Most of the developments in cloud storage have been positive except better cost model and effectiveness, but still data leakage in security are billion-dollar questions to consumers. Traditional data security techniques are usually based on cryptographic methods, but these approaches may not be able to withstand an attack from the cloud server's interior. So, we suggest a model called multi-layer storage (MLS) based on security using elliptical curve cryptography (ECC). The suggested model focuses on the significance of cloud storage along with data protection and removing duplicates at the initial level. Based on divide and combine methodologies, the data are divided into three parts. Here, the first two portions of data are stored in the local system and fog nodes to secure the data using the encoding and decoding technique. The other part of the encrypted data is saved in the cloud. The viability of our model has been tested by research in terms of safety measures and test evaluation, and it is truly a powerful complement to existing methods in cloud storage.

Keywords: Cloud storage; deduplication; fog computing and elliptic curve cryptography

1 Introduction

Since the twenty-first century, communication, information, computers, and related technologies have developed very rapidly in a short period of time. Cloud computing, a developing technology in this century, was suggested and defined by NIST in 2006 [1]. Cloud computing has certain advantages by enabling users to access resources, platforms, and software offered by cloud providers at a lower cost. When it was first proposed, cloud computing drew interest from a variety of industries and a greater amount of coverage from the public [2]. With the development of technology including cloud computing, the number of users is growing exponentially. According to the definition of cloud computing [3] "A large-scale computer vision model focused on economies of scale, in which a pool of abstracted, managed computing resources, dynamically scalable, virtualized, and services are delivered on-demand to external customers through the internet".

The development of cloud technology and growth is unexpected in the various factors like storage capacity, performance, stability, and cost. Considering the factors like huge storage and cost-effectiveness, most companies prefer this, as the data center is unsatisfied with the requirements of the consumers. So, people start to think and search for new technology and strategy which fulfill their needs. The greatest strength in storage capacity and cost-wise more users select the cloud storage. Cloud storage is a form of data storage and management service that has its own computing system. Inspire of cloud storage and the development, still there are a lot of questions in terms of security. Sharing the data in public leads to privacy problems, for example, Hollywood actresses preferred cloud storage to store private data and family albums, but it was hacked by hackers.

In Fig. 1, users start to upload to the cloud server direct. Here the cloud service provider (CSP) plays an important role in managing the user's data. So, the user cannot manage the data only by CSP establishes a distinction between data ownership and management [4]. As a result, the CSP has easy access to the data, while hackers can communicate with the CSP server to steal the user's information and related files. So, these are the two possibilities that lead to loss of data, data leakage where security is the question. Cloud storage and computing face these problems. However, various encryption techniques were applied, but unable to stop the internal attack. In the cloud and IoT, the main purpose of the fog concept is to improve reliability and efficiency while reducing the amount of data sent to the cloud for analysis, processing, and storage [5]. Deduplication technology is primarily used to reduce the amount of space and bandwidth needed, as well as to remove redundant data. When several users save the same data to cloud storage, deduplication is one of the most effective techniques [6].

Fog computing is a form of computing model that combines traditional cloud computing with restricted capabilities such as storing and networking services computing. It offers an excellent and better solution for latency-sensitive IoT applications [7]. The main goal of the fog concept in cloud and IoT is to increase reliability and performance while reducing the amount of data sent to the cloud for processing, examination, and storage. The rest of this work describes Section 2 summarizes the literature review of cloud security and related methodologies, Section 3 outlines the proposed multi-layer architecture technique, features, and workflow implementation, Section 4 reviews the various experiments, and Section 5 summarizes the paper's conclusion.

2 Related Reviews

The security and effectiveness in cloud storage have taken a lot of attention from a variety of industries. A lot of research is still going on in terms of security in cloud storage on daily basics. To solve those problems, the paper [8] proposes a framework that uses CBIR around image encryption to prevent confidential data from being leaked on the cloud server. To begin with, the functional design is extracted to accurately reflect the images. After that, the pre-filter tables are designed by locality-sensitive hash to effectively increase search efficiency. Secure kNN pseudo-code is used to protect the encrypted content of image pixels that are encrypted over standard stream cipher. Also, sometimes the authorized query is the chance of illegal copying during the retrieval of images. The author proposes a protocol based on the watermark to prevent such illegal distributions. Here the author has indexed two layers of data and a water-based protocol is used for encryption during the copy of the data. Fei et al. [9] described a solution for adapting to the new environment without using data mapping tables. This code considers device weights, hashing technology, and the distribution of large amounts of data objects among different user terminals based on node service availability and capability. While the data nodes are updated, the data stays evenly distributed in the different devices, and the quantity of data transfer is like the theoretical limit; and a small portion can be computed from a data frame, improving the device optimization test outputs and resolving the issue of fast data destination.

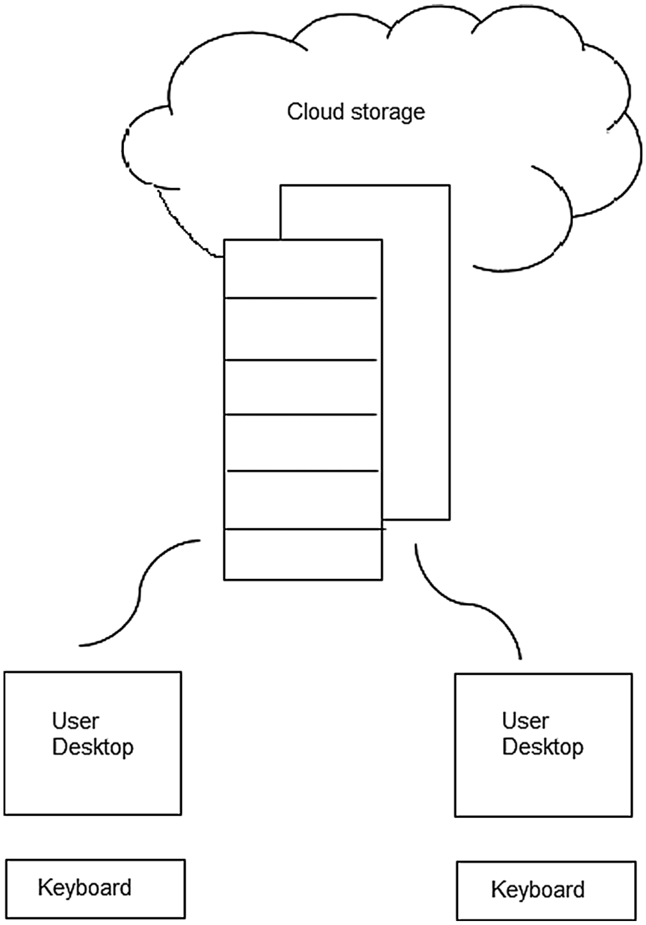

Figure 1: Traditional cloud storage structure

In the Proposal [10] when data is outsourced to the cloud, privacy is still an issue, according to Fan et al. The author presents architecture for cloud database storage. In this article, it prevents both local and cloud administrators from learning about outsourced content in the cloud. Machine-readable notations and expressions are often used to restrict database users to those who have a legitimate need to know. Administrators will not be able to modify these restrictions until a new function has been defined once the application has been introduced and moved to production. Furthermore, encrypted key information is bound to trusted states using trusted computing technologies. We mitigate the frequently cited privacy and confidentiality risks of organizational cloud storage by restricting the required confidence in both internal and external administrators and service providers. Sadeghi et al. [11] explain how to use a tamper-proof hardware token to demonstrate a technique. This tactic is used by a cloud service provider that uses a secure computing platform.

Hou et al. [10] argue that, even though a CSP becomes trustworthy and user information is stored through it, unauthorized people still access user information if they have control over the storage, they can access data. Typically, the user sends data to the cloud server along with authentication information; however, the password can be intercepted. They provided a safe virtual security scheme based on SSL from Daoli paper to make the device secure. The author [12] describes the architecture of a trusted storage system, the trusted database system (TDB) that protects a large amount of untrusted storage from hackers by using a small amount of trusted storage. Since untrusted data is encrypted and checked against a collision-resistant hash stored in trusted storage, it cannot read or modify the database. TDB combines encryption and hashing with a lower-level model to provide uniform data and metadata protection.

Feng et al. [13], highlight his points in the paper [14], database congestion and data leakage during transmission in the cloud are the possibilities. Also, the author discusses the issue and suggests a much more concise approach involving data encryption in a closed environment of the cloud. Furthermore, it is capable of layered encrypted storage with encryption. These encryptions, on the other hand, make searching in cloud storage more difficult. In today's world of cloud computing, searchable encryption is a hot subject. Different methodologies and solutions to those problems are presented in this paper. Each one achieves high levels of precision, protection, and efficiency.

In this paper [15], the author securely introduces TPA (third-party auditor). The audit function does not introduce any new vulnerability to client data privacy or increase the user's online burden. The author of this paper proposed a safe cloud storage framework that allows for public auditing with privacy protection. Our findings are expanded to allow the TPA to conduct audits for multiple users at the same time. According to comprehensive security and performance review, the proposed strategy is both provably highly secure.

Harnick et al. [16] briefs how the deduplication strategy can be used as a side-channel to expose details about the contents of other clients' files to malicious users. Based on research, the author developed a cloud storage attack scenario that employs deduplication among different users. An attacker can download all the data in the cloud storage if he obtains the hash values for the data. This is because only a portion of data or information, namely its hash value, acts as both an index of the data and a "proof" that someone who knows the hash value owns the corresponding data among many files. As a result, any user who knows the short hash value for a particular piece of data can browse all the data in the cloud storage.

3 Proposed Secure Cloud Storage Architecture

The proposed system explains cloud security with the methodologies of deduplication, divide and combine, encoding and decoding, finally encryption and decryption, when a user uses the cloud server to save the information.

3.1 Steps for Uploading the Data

1. Deduplication (dedup) Process.
2. Divide the information into three parts.
3. First data (5% data) is encoded and saved in the local machine.
4. Secondly (10% data) is encoded and saved in the fog server.
5. Third (85% data) is encrypted and saved on the cloud server.

3.2 Steps for Retrieving the Data

1. Decrypt the user information from the cloud server and restore it to the next layer.
2. Decode the data from the fog server and restore it to the next layer.
3. Decode the data from the local machine.
4. Combine the data.

3.3 Deduplication

Deduplication is an important factor in the storage environment in both physical and cloud storage. When the user has the same file but in different names, the consumption of space will be overhead. So, when the user prefers to cloud storage for his environment, the consumer needs to pay how much space is required to store the data. So, before using cloud storage, the company can have one node to remove the duplicates or unwanted blocks in the files. Deduplication (dedup) is the concept of removing duplicate blocks.

3.4 Fog Storage

Fog storage and computing is a form of distributed computing in which many devices are connected to a cloud storage system. An added advantage is that the processed data is most likely required by the same devices that produced the data, reducing latency during processing both remotely and locally. Fog storage is stated as, “It is a distributed computing architecture with a resource pool which consists of one or more ubiquitously connected heterogeneous devices and the network edge and to collaboratively provide elastic computation, storage, and communication.” Whilst, Vaquero et al. defined fog computing as; “a network through which a large number of heterogeneous, universal, and decentralized devices communicate and collaborate to perform processing tasks and to manage storage without the need for third-party intervention.” The significance of fog is to provide consistent latency and time-sensitive for various applications in IoT.

Characteristics of Fog Technology

The fog technology characteristics are.

- Location awareness and low latency.
- Geographical distribution.
- Scalability.
- Support for mobility.
- Real-time interaction and
- Heterogeneity.

3.5 Multi-Layer Secure Architecture

We propose an MLS architecture based on the fog node framework to secure customer information. The MLS framework allows users to have powerful management control and ensures that their privacy is protected. It is already noted the protection against internal attacks is difficult.

The various security models of encryption techniques are well and good for an outside attack, but the CSP has problems. Here the CSP is accessing and managing the data internally, so, encryption techniques become invalid. In the proposed model, the customer's data is divided into three parts and applies the encryption technology in cloud storage. Fig. 2 shows the model of MLS. The MLS architecture, as seen in Fig. 2, demonstrates the overall data processing capabilities along with the three divisions of user data. The model consists of one deduplication layer, a cloud entity, a fog entity with a local computer.

The first block is to remove the duplicate data from the original data and the local machine have 5% of data and 10% of data uploaded to the next fog layer, here 10% of the data are encoded and saved, the rest part of the data is uploaded in the cloud server of 85%. The third layer also saves the encoded data. Elliptic curve cryptography is used in the above operations. Fig. 3 shows the overall steps in the flow diagram how the data divided, and the security concept applied.

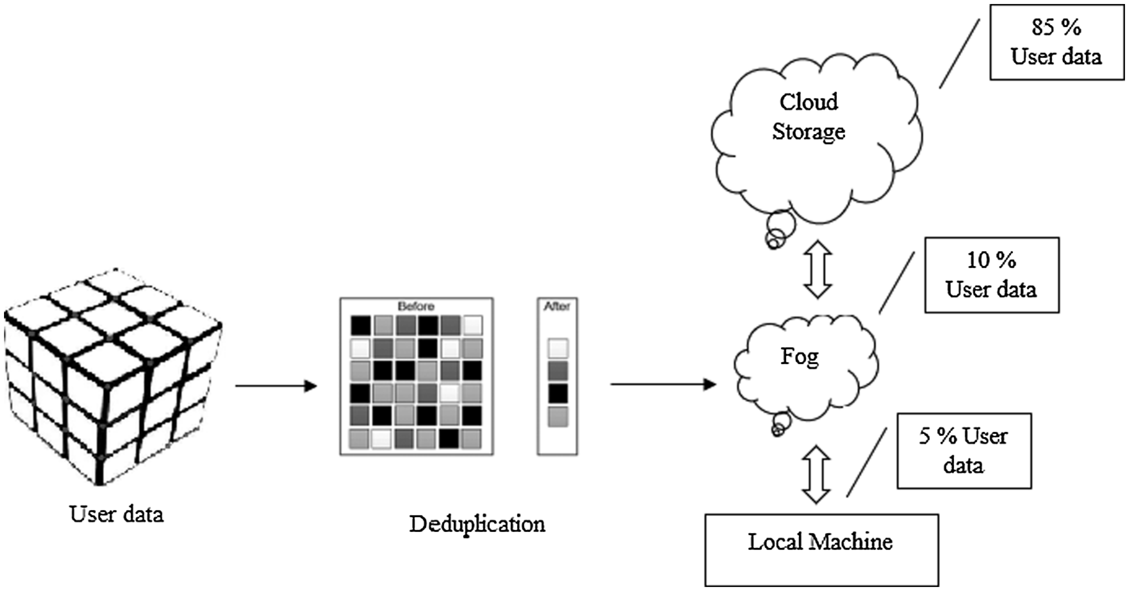

Figure 2: Overview of proposed model

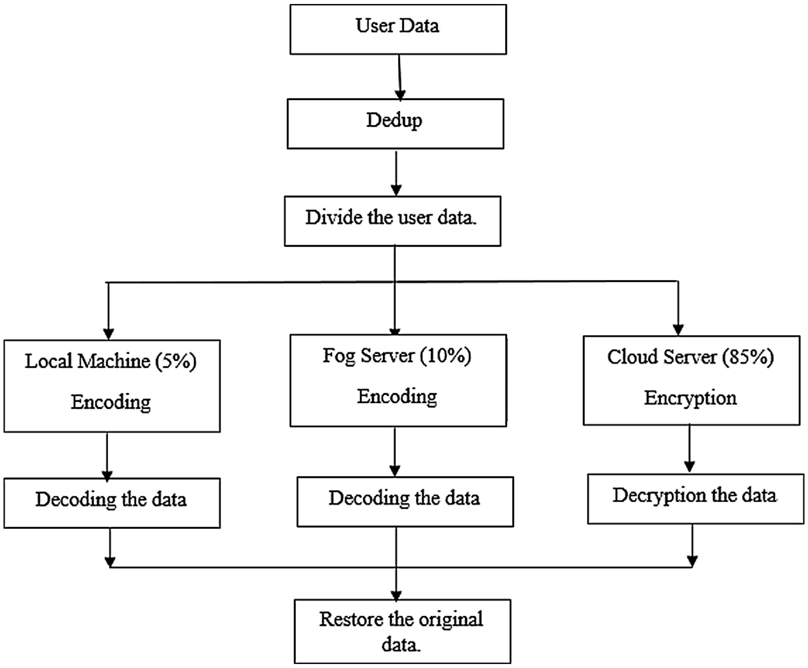

Figure 3: Flow diagram

Before dividing the user data, the deduplication concept is applied here to remove the duplicate blocks of data. Mainly, removing the duplicate blocks is for space management and bandwidth requirements. In computing, in this model, the user's data are split into three parts like user machine, fog machine, and finally cloud storage in encrypted data, with the scale of small to large data, respectively. In this strategy, while initially the duplicates are removed, and data is split into three the data is stored so the hacker may not reproduce the original form of data at any cost. The cloud storage management also gets the exact user's information, because the two layer user's machine and fog server are controlled and managed by the user.

Steps for storing the data (Local and Fog server)

As mentioned above, the first two divisions of data are saved in the local and fog machine of 5% and 10% respectively. So, the first data are encoded using the Hash-Solomon encoding algorithm. The procedure is,

1. User information is received and the received data is deduplicated to remove the duplicates.
2. Once the initial process is completed, the data is divided into 3 parts in the ratio of 5:10:85.
3. The first two parts are taken in the process and do the encoding using the Hash-Solomon encoding algorithm.
4. For encoding first, we are doing mapping transformation the file which needs to store, each which holds a number, says X it shown in Fig. 4.
5. Second, we do a hash function and on the received value X to get the matrix Y,
6. Now multiplication will generate (k) the data blocks.

Once data is received in cloud storage (X1-X4) , it calls the function elli_curv (data) in the below pseudo code. During the decoding process, the reverse action happens.

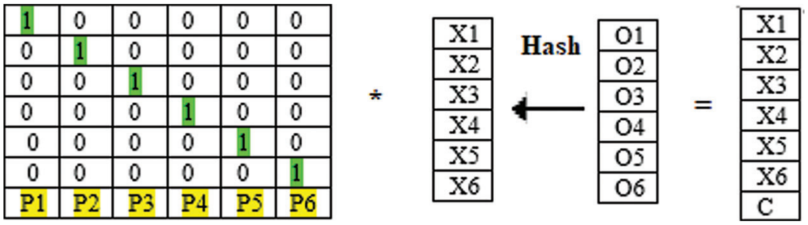

Figure 4: Hash function

3.6 Elliptic Curve Cryptography Model

It is a new form of cryptographic mechanism to secure the data in both physical and cloud storage. ECC is one of the best techniques based on the theory of elliptical curves. The property of the elliptic curve is used to generate the keys for encryption methodology instead of existing techniques which use large prime numbers. ECC uses the elliptic curve equation for key generation. In 1985, N. Kobiltz and V. Miller proposed the elliptical curve cryptography for unpredictable data to have secure data. The basic idea behind ECC is to use an elliptic curve to integrate a discrete logarithm problem. The model ECC has the most complex mathematical problems in public-key cryptography. In this case, where computing power is constantly improving, solving discrete logarithms in a mathematical sense becomes a relatively simple task. As a result, consider moving the discrete logarithm problem solution to a more critical and difficult area to implement, such as elliptic curves.

The most significant of the ECC model uses the smaller keys for security. When compared to the other model (1064-bit key) ECC takes the 164-bit key for the same level of security. This model is a public-key cryptosystem for the generation of keys centered on the mathematical complexity of solving the elliptic curve discrete logarithm problem to encrypt and decrypt data. It measures the hops needed to travel from one point on an elliptic curve to another. Elliptic curves are symmetrical over the x-axis and are binary curves.

ECC is a cryptographic scheme in which each client has two keys: a public and a private key. Encryption and signature authentication is done with the public key, while decryption and signature generation is done with the private key.

3.6.1 Key Generation

Any cryptography technique applied for security, the initial and the most significant process is the key generation for both public and private. With the help of the receiver's public key, the sender encrypts the message data, and the receiver decrypts it with its private key. Here we have the code for the elli_curv (data) function for cloud storage. The Key generation procedure is below.

1. Senders public key (rand_no$_{pA}$) a random number from {1, 2,…t−1}
2. Senders public key S.public_key$_{XA}$ = rand_no$_{pA}$* Gen_pt$_{Q}$
3. Receivers public key (rand_no$_{pB}$) a random number from {1, 2,…t−1}
4. Receiver public key R.public_key$_{XB}$ = rand_no$_{pB}$* Gen_pt$_{Q}$
5. Sender security key, key = rand_no$_{pA}$* R.public_key$_{XB}$
6. Receiver security key, key = rand_no$_{pB}$*S.public_key$_{XA}$

S.public_key$_{XA}$ = rand_no$_{pA}$* Gen_pt$_{Q}$ ## sender public key
R.public_key$_{XB}$ = rand_no$_{pB}$* Gen_pt$_{Q}$## receiver public key
key == rand_no$_{pA}$* R.public_key$_{XB}$## sender security key
key == rand_no$_{pB}$* S.public_key$_{XA}$## sender security key

3.6.2 Encryption Code

From this proposal, the user data is divided into three parts. The third part of 85% of user information is encrypted with the ECC method. The plain text is sent to the cloud, before saving the data it can be encrypted into points and it can be saved as encrypted points (F1, F2). The encrypted points are stored in cloud storage. The first user data is saved in the respective storages and during the third part of data, it calls the ellp_curve function.

Now sender, send message e to the end receiver.
##e has any point Ellp_curve on elliptic curve,
##Now the sender choose a random number, r from {1, 2…t−1}
##Cipher text will be written from points (F1, F2) here,
F1 = r * Gen_pt$_{Q}$ and
F2 = Ellp_curve+(r*Gen_pt$_{Q}$)

3.6.3 Decryption Code

Once the user stores the data in encrypted points (F1, F2) during the retrieval of the data, the encrypted points are decrypted as plain text that can be user readable.

On computing the receiver side, multiplication of F1 with its private key (rand_no$_{Pb}$ * F1)
Then subtracts from the result of previous step
Ellp_curve – Original message
Ellp_curve=F2-(rand_no$_{Pb}$* F1)
Ellp_curve=F2 –(pB *F1)

4 Evaluation Results

We computed the efficiency and feasibility of the MLS system based on the ECC and the fog model through a sequence of tests that include time-consuming encoding, similarly for decoding and testing the data are in different sizes. Here, the algorithm computes the user data into three segments of data, before dividing the duplicates of data blocks are removed. Once the data segment into three parts of data, they have moved to local, fog server, and cloud server with different sizes of data. Here in the local and fog server, data is processed with an encoding scheme called the Hash-Solomon code algorithm. The simulation was used to perform all the experiments in this paper, and the environmental parameters are described in Tab. 1.

Table 1: Simulation requirements

No	Items	Parameter
1	Operating system	Windows 2008
2	Language	C
3	Processor	Intel i7 2.50 GHz
4	RAM	8 GB
5	Storage	1 TB

Let us take three different types of files which are image (42.2 MB) and video (650 MB). One more block theory is used in all the experiments in this article, which implies that there are t + 1 data volumes. By the way, the method will protect data privacy while also alleviating the storage burden on the lower servers.

Below results are from various metrics and parameters that are considered. In the figure, we take the image file for analysis of the local storage, where the x-axis consists of the number of storage blocks and the y-axis consists of data volume stored in the machine. Similarly, the figure shows the result of the video file for similar parameters. In both the graphs, we are comparing the data volumes stored on the local machine for the Reed-Solomon code and Hash-Solomon code algorithm. According to the graph, as the number of data volume increases, the amount of data stored in the user's computer decreases. So, the large volume of data shows good results during this experiment. The consumption of space is less.

Figs. 5 and 6 show the results of local storage machine consumption of image and video files of both the algorithms. In Figs. 5 and 6, the user saves the image and video file in the local storage using both the algorithms. When a user tries to save 100 data blocks, the proposed algorithm will occupy less storage compared to the existing methods. Similarly, when the user tries to save 500 data blocks, the hash code algorithm consume nearly 100 MB whereas, in the Reed-Solomon algorithm, it consumes more than 100 MB. On comparing the existing and proposed methodology, small and big data blocks have a good result in the proposed algorithms.

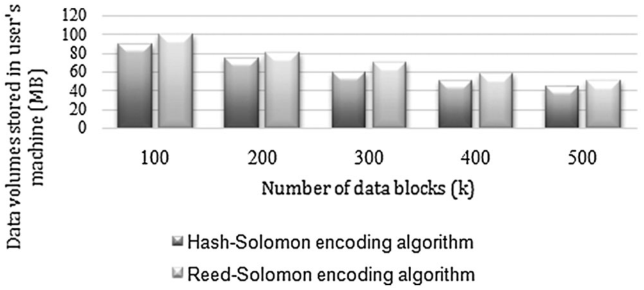

Figure 5: User machine volume of image file

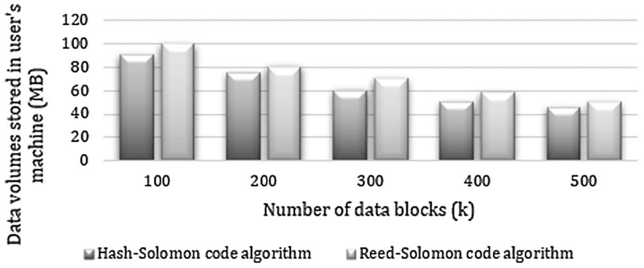

Figure 6: User machine volume of Video file

Figs. 7 and 8 are the results of the time taken for encoding and decoding the number of data blocks for both the Reed-Solomon code and Hash-Solomon code algorithm. During encoding, when the data blocks increase, the time taken for encoding increases exponentially. So, we need to monitor the local's machine performance at different time intervals. During decoding, the volume of data increases, the time taken for decoding also increases. So, the processing cost of decoding becomes more attentive.

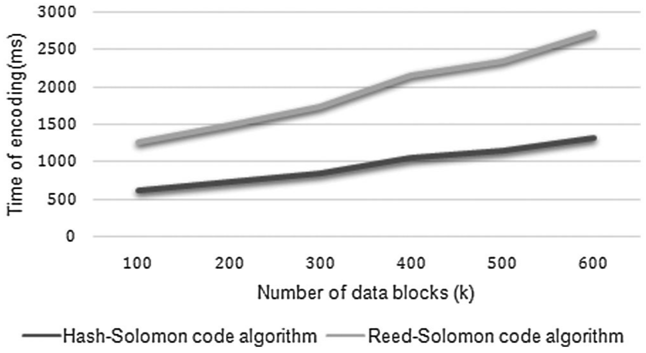

Figure 7: Encoding time

When the 100 data blocks are encoded, it takes nearly 600 ms and more than 2500 ms for the proposed and existing methods. Similarly, while decoding for 100 blocks, it consumes nearly 600 and 1200 ms respectively.

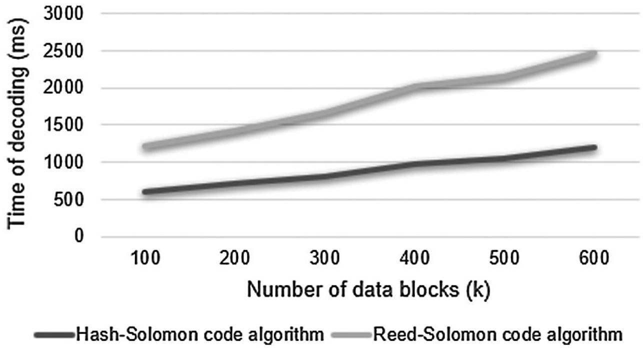

Figure 8: Decoding time

Fig. 9 is the result of the time taken to dispute the data from the cloud server. Here, it shows the execution time for both the Elliptic curve cryptography and the MRSA algorithm. Whenever the file system increases, the time also increases simultaneously. On comparing with the MRSA, the elliptic curve cryptography has a lesser execution time. Hence, the security-wise and execution speed are efficient in results.

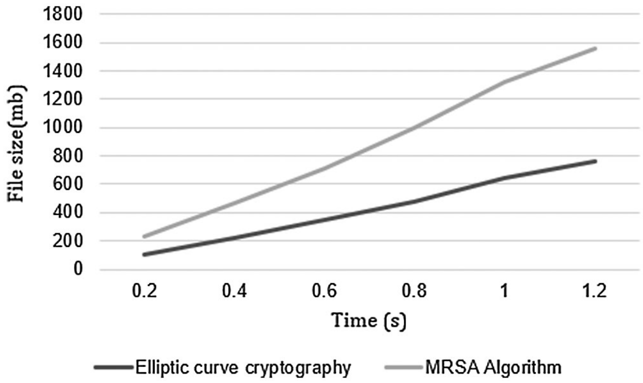

Figure 9: Execution time comparison

5 Conclusion

In the research article, a new Multi-layer secure architecture based on a fog model is proposed for protecting the information using an elliptical curve algorithm. Here, the user is divided into a ratio of 85:10:5 and all allocated into the devices and use two different techniques to make the data insecure. If hackers are trying to crack the data, two different techniques are applied. The performance of the proposed architecture is evaluated in terms of execution time, encoding time, decoding time, and storage efficiency of data. The elliptic curve cryptography algorithm performance is compared with an existing technology such as MRSA. From the experimental result analysis, the proposed technique is feasible and secure. The proposed MLS architecture achieves lower computational costs and faster. It also allows the use of secure exchange protocols, adding to the overall security. In the future, this work may be extended using advanced cryptographic techniques to improve the security of information stored in cloud computing.

Funding Statement: The authors received no specific funding for this study**.**

Conflicts of Interest: The authors declare that they have no conflicts of interest in the report regarding the present study.

References

[1] P. Mell and T. Grance, “The NIST definition of cloud computing,” *National Institute of Standards and Technology*, NIST *Special Publication* USA, vol. 53, no. 6, pp. 50–50, 2011.

[2] H. T. Dinh, C. Lee, D. Niyato and P. Wang, “A survey of mobile cloud computing: Architecture, applications, and approaches,” *Wireless Communications and Mobile Computing*, vol. 13, no. 18, pp. 1587–1611, 2013.

[3] I. Foster, Y. Zhao, I. Raicu and S. Lu, “Cloud computing and grid computing 360-degree compared,” in *Grid Computing Environments Workshop, GCE ‘08*, Chicago, IL, USA, 2009.

[4] L. Xiao, Q. Li and J. Liu, “Survey on secure cloud storage,” *Journal of Data Acquis Process*, vol. 31, no. 3, pp. 464–472, 2016.

[5] B. Qing-hail, Z. Wen, J. Peng and L. Xul, “Research on design principles of elliptic curve public key cryptography and its implementation,” in *Int. Conf. on Computer Science and Service System*, Nanjing, China, 2012.

[6] J. Hur, D. Koo, Y. Shin and K. Kang, “Secure data deduplication with dynamic ownership management in cloud storage,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 11, pp. 3113–3125, 2016.

[7] H. F. Atlam, R. J. Walters and G. B. Wills, “Fog computing and the internet of things: A review,” *Big Data Cognitive Computing*, vol. 2, no. 10, pp. 1–18, 2018.

[8] Z. Xia, L. Zhang, Z. Qin and K. Ren, “A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing,” *IEEE Transaction on Information Forensic and Security*, vol. 11, no. 11, pp. 2594–2608, 2016.

[9] M. Fei, X. Wei, S. Jiwu and Z. Weimin, “A mapping algorithm for replicated data in large-scale storage system,” *Journal of Computer Research and Development*, vol. 46, no. 3, pp. 492–497, 2009.

[10] J. Hou, C. Piao and T. Fan, “Privacy preservation cloud storage architecture research,” *Journal of Hebei Academy Sciences*, vol. 30, no. 2, pp. 45–48, 2013.

[11] A. -R. Sadeghi, T. Schneider and M. Winandy, “Token-based cloud computing,” *Trust and Trustworthy Computing*, vol. 6101, pp. 417–429, 2010.

[12] Q. Hou, Y. Wu, W. Zheng and G. Yang, “A method on protection of user data privacy in cloud storage platform,” *Journal of Computer Research Development*, vol. 48, no. 7, pp. 1146–1154, 2011.

[13] G. Feng, “A data privacy protection scheme of cloud storage,” *IEEE Transactions on Parallel Distributed Systems*, vol. 14, no. 12, pp. 174–176, 2015.

[14] Z. Fu, X. Wu, C. Guan, X. Sun and K. Ren, “Toward efficient multikey word fuzzy search over encrypted outsourced data with accuracy improvement,” *IEEE Transactions Information Forensics Security*, vol. 11, no. 12, pp. 2706–2716, 2016.

[15] C. Wang, S. M. Chow, Q. Wang, K. Ren and W. Lou, “Privacy-preserving public auditing for secure cloud storage,” *IEEE Transactions on Computers*, vol. 62, no. 2, pp. 362–375, 2013.

[16] D. Harnik, B. Pinkas and A. Shulman-Peleg, “Side channels in cloud services, the case of deduplication in cloud storage,” *IEEE Security & Privacy*, vol. 8, no. 6, pp. 40–47, 2010.