AN ENHANCED AES-ECC MODEL FOR THE SECURITY OF MOBILE APPLICATIONS USING CLOUD COMPUTING

Maria Navin J R[*1] & Nagaraj M Lutimath[2]

[*1]Dept. of Information Science and Engineering, Sri Venkateshwara College of Engineering, Bangalore, Karnataka, India

[2]Department of Computer Science and Engineering, Dayananda Sagar Academy of Technology and Management, Bangalore, Karnataka, India

Abstract:

Intense computation on client devices with limited resources has been drawn by and made possible by the phenomenal expansion of computational clouds. Smart mobiles can deliver data and computationally heavy apps primarily by utilizing the demand service paradigm of distant data centers. Due to increased concerns over data privacy and security, it is difficult to outsource private and sensitive information to faraway data centres. Therefore, in order to meet the brand-new security challenges that have surfaced in the cloud environment, the traditional advanced encryption standard (AES) algorithm needs to be improved. Improved security and owner-data privacy are two crucial components of the framework this study proposes. AES uses the four operations Sub Bytes, Shift Rows, Mix Columns, and Add Round Key to encrypt and decrypt messages. Sub Bytes and Mix Columns are the two columns that have the most impacts on latency. The execution lag of Mix Columns is responsible for 60% of the total latency. Therefore, for these contexts, a low-cost, low-power symmetrical data encryption algorithm is essential. The goal of the study is to change the shift rows and mix column phases of AES in order to propose an effective and secure method. Several protocols and algorithms have been created employing cryptographic methods like elliptic curve cryptography (ECC) to guarantee the integrity and security of the data. The proposed system combines the ECC with modified Advanced Encryption Standard (AES) technology to guarantee data integrity and authentication. The experiments' findings demonstrate that the suggested technique is effective and produces superior outcomes to the alternatives. Cloud users can safely manage data protection and integrity according to the proposed security architecture.

Keywords: Elliptic Curve Cryptography (ECC), Cloud Security, Advanced Encryption Standard (AES), shift rows, mix columns, confidentiality, mobile device

DOI: 10.24297/j.cims.2023.4.11

1. Introduction

The most cutting-edge and practical branch of this is cloud computing, which makes use of the internet to allow users access to data storage. Researchers have paid a lot of attention to cloud computing security in recent years because of the significance of the data used and the growing use of the technology, which is now used to deliver a wide range of services in numerous industries. Data Security is thus a significant problem while keeping data in clouds. One of the most crucial ways to provide data security on the cloud is through cryptographic algorithms.

If the end user has an Internet connection, they can save money by storing and retrieving private data from remote storage in a cloud computing environment. The user can access the data at any time and from any location. However, the security of data sent over the cloud cannot always be ensured. Because the end user can only access the data with the aid of a third party, data integrity and authentication may be compromised. In a cloud computing environment, the end user can save money by storing and retrieving private data from remote storage if they have access to the Internet. The user has unrestricted access to the data at all times, from any location. However, there is no guarantee that data sent over the cloud will always be secure.

The term mobile cloud computing refers to a new paradigm where computing power is available as an on-demand service via mobile and tablet devices. Users may lose control of their data if they decide to use cloud computing. As a result, backing up data while it is in transit and in the cloud is a serious issue. Applications that rely on changing technology need to take into account all possible dangers. There are a number of problems with cloud technology, including data integrity and security. This is the principal motivation behind why individuals are wary about utilizing cloud innovation. Access control and key management are also data security factors that contribute to the full utilization of cloud technologies. The primary worries of cloud clients in portable distributed computing are information related security issues, like information honesty, privacy, accessibility, and discernibility.

Data security is crucial in the current world. particularly if the data transmission network employed is insecure. He can encrypt and decrypt data using the same key thanks to symmetric key cryptography. Despite having a straightforward architecture, brute force attacks can readily break them. The entire cryptographic security can be compromised if an attacker manages to obtain the key. Asymmetric key algorithms, on the other hand, employ key pairs (one for encryption and one for decryption) and provide greater security than symmetric key algorithms but require more time to execute.

By combining modified Advanced Encryption Standard (AES) with ECC, this research aims to create a hybrid encryption method that safeguards multimedia data like photos, text files, audio files, and video files. It makes use of a type of hybrid cryptography that uses both symmetric and asymmetric keys. Utilizing symmetric technology, such as AES, DES, and others, to accomplish this is susceptible to attacks using brute force. As a result, symmetric/asymmetric hybrid systems

are evaluated as having security levels that are equal to or higher than those of traditional systems. A 1024-bit RSA key and a 160-bit ECC key share the same level of security. It was decided to use ECC as the provider for asymmetric keys in our research due to the fact that it occupies less space than other methods.

It was in use for a very long time before the inherent limitations of symmetric encryption algorithms like DES were exploited. DES uses a 56-bit key to encrypt 64-bit plaintext and produce 64-bit cipher text. Brute force attacks are capable of breaking even the most straightforward and straightforward encryption due to the short key length.

Triple DES (3DES), which employs two keys and triple-encrypts the plaintext with a 112-bit or 168-bit key, was used to get around these restrictions. However, compared to ordinary DES, the encryption process is substantially slower. It was suggested to use the AES cypher as an encryption standard to overcome the drawbacks of DES. Depending on how many rounds are used, a single key that is 128, 192, or 256 bits long is used (10, 12, or 14). AES often involves substitution, shifting or rows, combining columns modifications, and adding a round key to all rounds but the final one in order to obtain the associated cipher-text. While DES is quicker than AES, it does not provide the same level of security, according to a performance comparison of the techniques now in use (2).

Cryptography is categorised as either symmetric or asymmetric in the majority of texts on the topic. Rijndael, a participant in the US National Institutes of Standard and Technological Computing [1, 2], initially introduced it in 2001. It takes the place of DES. The most widely used symmetric cypher encryption method for data security is AES. Each cycle in the Secret Writing Method contains four steps [3]. Shift Rows, Sub Byte, Round Key, and Mix Column are added. The method supports successive rounds of 10, 12, and 14 and keys with a length of 128, 192, or 256 bits [4, 5].

More postponement is delivered by Sub Bytes and Blend Sections. Subsequently, the AES calculation isn't utilized for the Web of Things, remote recognition organizations, and low-power gadgets like advanced cells and PDAs. Therefore, for these contexts, a low-cost, low-power symmetrical data encryption algorithm is essential [6]. The study's main goal is to suggest an effective method for changing the shift rows and mix column stages of AES coupled with ECC. When contrasted with different methodologies, ECC can offer a similar degree of safety with a more modest key size. The creation of a system that uses the cloud to provide data security at a lower computational cost and faster encryption/decryption procedure is necessary.

To utilize the characteristics of the two strategies, we blend them in our recommended model. The study's main findings are as follows: We propose a hybrid paradigm that utilizes ECC for AES key creation and combines AES and ECC. The four sub-operations of ECC-EAES are Sub Bytes, Symmetrical Transposition, Bitwise Reverse Transposition and Add Round Key Operation. In our

proposed method ECC-EAES 2nd and 3rd stages of AES are substituted by Symmetrical Transposition and Bitwise Reverse Transposition, respectively. A 256-bit key and a block of 128-bit plaintext (data) are used as inputs by the algorithm.

Similar to symmetric/asymmetric encryption, data encryption and decryption require either a private key or a public key. Because of the tremendous key size expected by this strategy, it requires a great deal of figuring power. The ECC-EAES hybrid strategy that has been proposed deals with the issue of the key's size while using fewer computer resources for memory optimization, which has the advantage of hastening the increase in system security.

The sections that follow make up the rest of this paper. Studies that are related are provided in Section 2. In Part 3, the research technique is explained in detail. In Section 4, the results and observations of the experiment are discussed. In Area 6, the work's decision is introduced.

2. Related Work

Because all users share resources in sync, cloud storage is becoming more and more popular. Because cloud storage is always accessible, data owners choose it over other providers. Data integrity and data preservation should be examined for this reason in order to boost system security.

For enhancing system security, AES and ECC are suggested [7]. Without a trusted center, the system is distributed and managed using Shamir secret sharing. The combination strategy that has been offered does increase system security, but at a significant computational cost and time expense.

Along with the suggested technique, which makes use of comparable algorithms for secure cloud services, AES, DES, and Blowfish are also used [8]. These algorithms offer data storage efficiency and integrity to prevent conflict between large groups of users and individually secure each user's data. Additionally, the service provider manages and expedites data accessibility. The cloud computing data services also measure the size of data blocks and the avalanche effect of plain text.

Security advantages of ECC with RSA can be compared in a study [9] by Madhavi et al. utilising data that is larger than 264 bits, as 256-bit data exceeds NIST restrictions. Because to providing higher safe services over smaller data volumes and reduced storage requirements for data accessibility, the ECC approach outperforms the RSA method in this performance comparison.

Hybrid approaches for RSS and ECC are employed in the study [10]. After the data has been compressed, the elliptical curve authorities receive some pieces that need to be signed so they can message digest and sign them. The same method is used to carry out the encryption

process. The supremacy of RSS and ECC analysis provides the foundation for the development of hybrid algorithms.

Providing secure and private data security is a major challenge for cloud computing services [11, 12]. Due to privacy concerns, we are unable to keep raw data without encryption because the CSP is an untrusted third party. The proposed study examines a hybrid cryptosystem-based solution for capacity and trustworthy information transport in the cloud. By simultaneously implementing AES and ECC to enhance the framework's categorization and credibility, we may use symmetric and divergent encryption to enhance the cloud data security. As a result, the projected model manages an efficient, powerful, and safe encryption approach based on AES and ECC.

The capabilities of cryptography for providing security in distributed storage were introduced in papers [13, 14]. This was accomplished by investigating common cryptography methods like AES, ECC, and RSA. The question of identifying an efficient and secure encryption approach was, in any case, settled by this investigation with regard to the variances in the exhibition of these operations. While certain encryption techniques can guarantee security, they take a long time to encode and decrypt data. On the other hand, while various solutions may provide effective encryption, they nonetheless suffer from the negative impacts of the requirement for security.

In papers [15, 16], a strategy for improving cloud computing information security and a two-level cryptographic method were presented. By enhancing information security against intrusions by utilizing both symmetric and uneven encryption calculation (AES and ECC), prevent them from accessing the actual information, empower privacy, respectability of the information, and time required to perform cryptographic tasks, the model increases client confidence in cloud computing and accelerates the use of smaller ECC keys in cryptographic interactions.

3. Proposed Framework

In this section, we provide a comprehensive design description of the proposed scheme. We emphasize the significance of combining AES and ECC and the calculation used in this plan.

Defining ECC and AES

By employing the subsequent asymmetric key encryption, the data are protected from unauthorized access by the well-known cryptographic technique known as ECC. The ECC's security is safeguarded by using key pairs that are both public and private. As prime and binary fields, ECC makes use of two-dimensional fields. the use of enhanced operations and the creation of a connection between binary and primary fields that prevents unauthorized access, make hacking difficult with this cryptographic strategy. The small key size of ECC is a crucial feature. When maximum number of points are used for finding the right field during the cryptographic implementation can provide enhanced data security.

Before generating a massive number that can range from 0 to Z based on the input, the field's start action selects the first number. Because ECC is used exclusively to generate the key, the processes are simplified. Because to its tiny key size, ECC has a substantially larger enhancement than other cryptographic techniques. In this study, memory and space enhancement are optimized using ECC techniques [17].

One cipher text format that uses block ciphers is AES. This protects your data by encrypting and decrypting it with just one key. This includes various performance activities constrained by cloud storage, such as cloud storage retrieval and statistical analysis. Security policies for cloud storage are most often enforced by the strategic algorithm in cloud computing. This article employs the AES encryption method because it is simple to set up and compatible with data that can be retrieved from cloud storage [18, 19].

ECC and Improved AES - A Suggested Combination Hybrid Method

The most advanced and effective encryption technique for cloud storage was developed using ECC and AES. AES uses greater key size so it is slower than the hybrid (ECC-AES) method since the hybrid model permits a smaller key size and a quicker security mechanism to safeguard the data. ECC-AES hybrid model uses decreases key size for encryption and boosts performance because ECC's tiny key size is its main point of differentiation [20]. The ECC has set standards for encryption and decryption keys in order to create a safe key system and minimize key size.

ECC and AES work well together to encrypt data and bar unauthorized access. Data encryption and decryption produce cypher text after selecting the key size. AES makes use of the key that ECC generates. The proposed cloud storage approach can provide a safe system because of the combined impacts of ECC and AES. In this way, the size of secure data storage can be decreased.

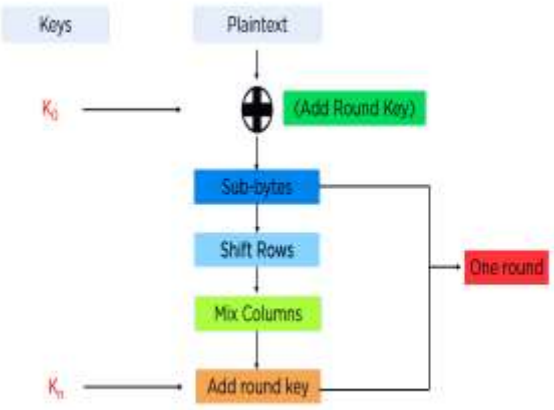

Fig. 3.1 Workflow for AES Encryption Algorithm

Key length affects the number of rounds in AES. AES employs 10 rounds for 128-bit keys. 12 rounds for a key with 192 bits. For 256-bit keys, there are 14 rounds, as seen in Fig. 3.1. Each

encryption round consists of four phases: Byte Substitution (Sub Bytes), Row Shift, Column Shuffle, and Round Key Addition. The decryption process works in exactly the opposite way. Security is not compromised as long as the system is effectively designed and uses effective key management methods.

Add Round Keys

The Add Round Key operation functions on the AES Round key similarly to how AES encoding does. The input to the round is exclusively-ored with the round key throughout this procedure.

Sub Bytes

In this study, the Add Round Key operation of ECC-EAES functions on the AES Round key similarly to how AES encoding does. Throughout this process, the round key will only be used to input data into the round.

Symmetrical Transposition

The proposed ECC-EAES symmetrical transposition part replaces the shift row of AES in this section, and all rows and columns of the cipher's internal 128-bit state are combined. Wherever there is a byte in every cell, that may be a 4x4 matrix. The positions of non-major diagonal sections that are symmetric with regard to the main diagonal components are switched using inner state bytes. The rows and columns of the matrix are filled with these bytes. Then, in reference to the non-diagonal array of elements, the primary symmetrical diagonal components are reversed.

Bitwise Reverse Transposition

The bitwise reverse transposition stage offers diffusion by mixing the input round, just like the symmetrical transposition phase of the proposed ECC-EAES. Bitwise reverse transposition, as opposed to the mix columns section of the original AES, uses optimal operations to speed up the encoding and secret writing algorithms.

The Proposed Framework's Algorithm

Utilizing Elliptic Curve Cryptography to Create Public Keys.
Step I. Consider a prime number n.
Step II. For producing the public key, we can use any of the number as n(a), where n(a) > n.
Step III. In cases when G > n, determine the curve's point as G.
Step IV. The public key is calculated using the equation P = n (a) * G.
Step V. The calculated public key P is returned.

Improved Advanced Encryption Standard encryption and decryption
Step I. First, open the input file.
Step 2: Next, add the public key produced by the ECC.
Step 3: Using the public key produced by ECC, EAES encryption is carried out on the input file.

Step 4. ECC-EAES's Round Key operation uses the AES Round key in a manner similar to how AES encoding does. Throughout this process, the round key will only be used to input data into the round.
Step 5. ECC-EAES separates the input into bytes using the Sub Bytes component. AES uses a constant S-Box for every byte. In Galois Field 2^8, the ECC-EAES S-Box also uses inverse multiplication.
Step 6. During the shift rows phase, the positions of the non-major diagonal elements that are symmetric with respect to the main diagonal ought to be swapped, and the members of the major diagonal should then be reverse
Step 7. During the mix columns phase we transfer the elements from the input's first row to the output's first column, then switching a_{21} for a_{31} and reversing the bit order in the result. Taking the elements from the second row of the input and moving them to the third column in the output, then switching a_{23} for a_{33} and flipping the bit order in the output. Taking the components from the third row of the input and moving them to the second column in the output, then swapping a_{22} and a_{32} and flipping the bit order in the output. Taking the components from the fourth row of the input and moving them to the fourth column in the output, then swapping a_{24} and a_{34} and flipping the bit order in the output.
Step 8. The file is then uploaded to the server and encrypted with EAES. After being uploaded, the original file's encryption is broken using the translation utilising the ECC's public key.

To decipher the recommended ECC-EAES encrypted cypher text, each stage of the encryption operation must be undone in the opposite order that it was utilised. It is clear that EAES and ECC successfully safeguard data when stored in the cloud. Encrypted data even ensures the safe transfer of user data to the server and the subsequent storage mechanism. Innovation can also be evaluated using costs and computation time.

4. Results and Discussions

Based on the following security assessment criteria, our suggested technique, ECC-EAES, is created and contrasted to the original AES algorithm and ECC-AES algorithm: hamming distance, avalanche effect, mathematical soundness, encryption and decryption times, and avalanche and diffusion effects. The execution makes use of an 8 GB of memory and a 2.7 GHz Intel Center i5 processor. These techniques were examined using the AWS SDK and the Amazon Simple Storage Service. A 256-bit key and a block of 128-bit plaintext (data) are used as inputs by the algorithm.

The suggested ECC-EAES algorithm includes properties that increase security by making the system more complicated and resistant to attacks. Furthermore, it is clear that the suggested h technique encrypts and decrypts data much faster than existing methods. Our computational cost decreases together with the time required for encryption, which is incredibly efficient. As can be shown in Figs. 4.1 and 4.2, our suggested method is therefore more effective than others.

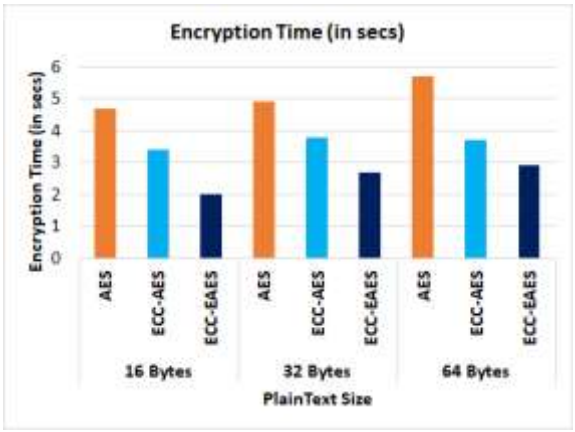

Fig. 4.1

How much an algorithm has changed can be determined using the Avalanche Effect (AE). Simply expressed, it means that the output of the text can be significantly affected by a minor change in the input. By adding up the modified bits in the cypher bits and dividing the result by the entire amount of cypher bits, we can get the AE. The equation we used to calculate the Avalanche Effect

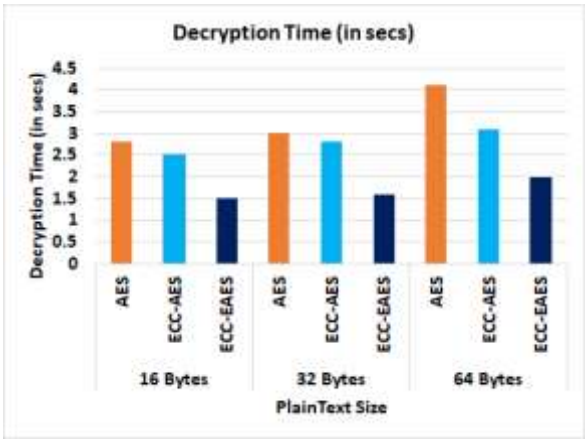

Fig. 4.2

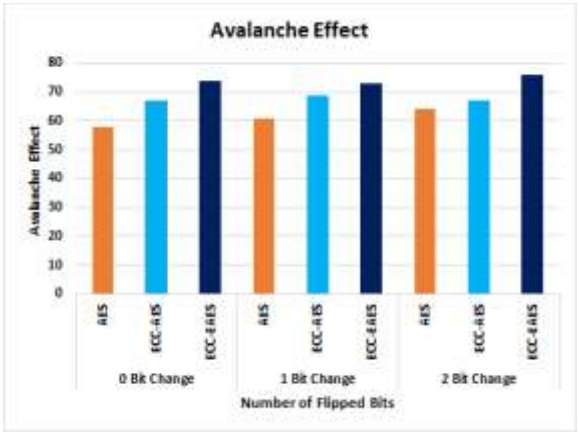

Fig. 4.3

AE = Changed bit count/total bit count

We contrasted the Avalanche effect of our proposed strategy with that of the AES algorithm and the ECC-AES algorithm, as illustrated in Fig. 4.3. If an algorithm's Avalanche impact is greater than 50%, it is regarded to have greater security strength than others. Thanks to our suggested method's ability to achieve the maximum Avalanche effect, the system is safer than others.

Diffusion, a feature of the avalanche effect in cryptography, displays the strength of a method from a cryptographic perspective. Any change to the associate degree input, no matter how modest, has a significant impact on the final result (plaintext or secret key). The avalanche effect is another name for this. We used hamming distance to calculate the Avalanche effect.

The hamming distance is a tool used in information theory to measure dissimilarity. Because it is simple to implement programmatically, as the sum of bit-by-bit xor (exclusive or) considering ASCII values, we get the hamming distance. It is recommended to have a high diffusion rate or high avalanche outcome. Avalanche findings show the performance of a cryptographic algorithm. The avalanche effect increases with the number of variations to the cypher that result from a single bit change in the key or plain text. We can observe that the harder it is to simply break the algorithm, the larger the avalanche effect is, like with our suggested solution. We can observe that our solution has improved security thanks to the Avalanche effect.

In this subsection, we compare the diffusion characteristics of the shift rows of the AES to those of the symmetrical transposition method. The diffusion attribute is calculated using the Hamming Distance (HD), which is the distance between two strings of the same length. Both the measured hamming distance between input and output (current output) and the change in the cypher value were tested during the execution of the two prior procedures.

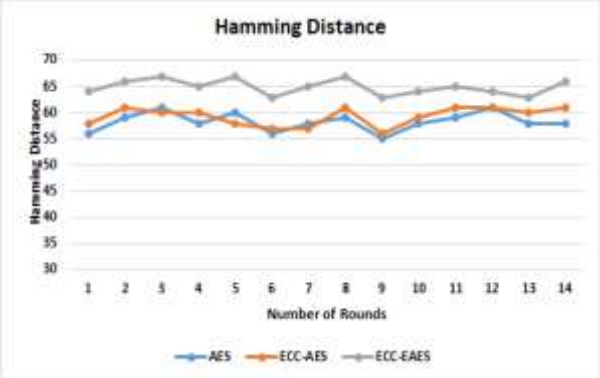

Fig. 4.4

According to the results shown in Fig. 4.4, the proposed method's hamming distance values range from 58 bits to 74 bits, whereas those of the ECC-AES and AES algorithms were, respectively, 57 bits to 71 bits and 58 bits to 78 bits. Less diffusion property is produced when a

shift rows operation exchanges positions with a sub operation. As a result, the suggested symmetrical transposition's operation sequence exhibits improved dissemination.

5. Conclusion

The infinite infrastructure for mobile applications offered by cloud computing allows for great scalability, low maintenance online data execution. It is clear that with the implementation of cryptographic techniques for safe computation, the ongoing vulnerability in the Cloud may still be managed. Data encryption protects data from unauthorized users, ensuring security and data confidentiality. Data privacy is guaranteed using a hybrid encryption scheme. The suggested model made use of the fast-symmetric scheme and less computationally demanding resilient cryptosystem methods of the AES algorithm with its key encryption utilizing ECC. As previously noted, we put forth the two proposed methods of bitwise reverse transposition and symmetrical transposition. These suggested methods were created for the ECC-EAES algorithm. The present AES shift row step was replaced by symmetrical transposition based on the suggested techniques. This took place in order to balance the trade-off between security and ESE-efficiency. AES's Additionally, bitwise reversed transposition was used in place of the AES mix columns stage. In order to balance security and encryption time, this was done in order to speed up the encryption method.

References

1. Alexandra Durcikova Murray E. Jennex. (2017). Introduction to Confidentiality, Integrity, and Availability of Knowledge and Data Minitrack. Proceedings of the 50th Hawaii International Conference on System Sciences, 2017, Page - 4287.
2. Altatar, M. A. (2017). Modified Advanced Encryption Standard Algorithm for Reliable Real-Time Communications, International Journal of Computing and Digital Systems, Pages 303-309.
3. Awad, A. I. (2018). Introduction to information security foundations and applications. Research Gate, Retrieved from https://www.researchgate.net/Publication/325170901.
4. Ayushi Arya et al. Review Paper on Effective AES Implementation, International Journal Of Engineering And Computer Science ISSN: 2319-7242 Volume 4 Issue 12 Dec 2015,Page No. 15403-15405.
5. Avi Kak, AES: The Advanced Encryption Standard, Avinash Kak, Purdue University, January 31, 2019, page 20-11.
6. Rizky Riyaldhia, et al, (2017., October 13-14). Improvement of advanced encryption standard algorithm with shift row. Elsevier B. V., ScienceDirect, Procedia Computer Science 116 (2017), pages 401–407.
7. Shukla, D.K.; Dwivedi, V.K.; Trivedi, M.C. Encryption algorithm in cloud computing. Materials Today Proc. Volume 37, Part 2, 2021, Pages 1869-1875.
8. Yahia, H.S.; Zeebaree, S.R.M.; Sadeeq, M.A.M.; Salim, N.O.M.; Kak, S.F.; Al-Zebari, A.; Salih, A.A.; Hussein, H.A. Comprehensive survey for cloud computing based nature-inspired

algorithms optimization scheduling. Asian Journal of Research in Computer Science, May 2021, Page 1-16.

9. Qazi, R.; Khan, I.A. Data security in cloud computing using elliptic curve cryptography. International Joutnal of Computing and Communications Networks, ISSN: 2664-9519 (Online);Vol. 1, Issue1, August 2019, 46–52.
10. Manaa, M.E. Data encryption scheme for large data scale in cloud computing. Journal of Telecommunication, Electronic and Computer Engineering • September 2017. 9, 1–5.
11. Arockia, P.; Dharani, N.; Aiswarya, R.; Shailesh, P. Cloud data security using elliptic curve cryptography. International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 04 Issue: 09 | Sep -2017.
12. Li, Y.; Gai, K.; Qiu, L.; Qiu, M.; Zhao, H. Intelligent cryptography approach for secure distributed big data storage in cloud computing. Information Sciences, Vol. 387, May 2017, Pages 103-115
13. Saeed, Z.R.; Ayop, Z.; Azma, N.; Rizuan Baharon, M. Improved cloud storage security of using three layers cryptography algorithms. Journal of Computer Science IJCSIS. 2018, 16, 34–39.
14. Al-Dhuraibi, Y.; Paraiso, F.; Djarallah, N.; Merle, P. Elasticity in cloud computing: State of the Art and research challenges. IEEE Transactions on Services Computing. 2017, 11, 430–447.
15. Hodowu, D.K.M.; Korda, D.R.; Ansong, E.D. An enhancement of data security in cloud computing with an implementation of a two-level cryptographic technique, using AES and ECC algorithm. International Journal of Engineering Research & Technology (IJERT), Vol. 9 Issue 09, September-2020, 639–650.
16. Zhu, Y.; Fu, A.; Yu, S.; Yu, Y.; Li, S.; Chen, Z. New algorithm for secure outsourcing of modular exponentiation with optimal checkability based on single untrusted server, Proceedings of the 2018 IEEE International Conference on Communications (ICC), 20–24 May 2018; pp. 1–6.
17. Bhardwaj, K.; Chaudhary, S. Implementation of elliptic curve cryptography in 'C'. International Journal on Emerging Technologies 3(2): 38-51 (2012).
18. Ogiela, U. Cognitive cryptography for data security in cloud computing. Concurrency Computation Practice and Experience, 2019, 32, e5557.
19. Sood, S.K. A combined approach to ensure data security in cloud computing. Journal of Network and Computer Applications, Volume 35, Issue 6, November 2012, Pages 1831-1838.
20. Mendonca, S.N. Data security in cloud using AES. International Journal of Engineering Research & Technology (IJERT), Vol. 7 Issue 01, January-2018..