# Hybrid Cryptographic Algorithm Using Multiprocessing for File Storage on the Cloud

**A.Vijayalakshmi**
*Vardhaman College of Engineering Hyderabad, India, vijayalakshmivardhan@gmail.com*

**N. Varun Reddy**
*Vardhaman College of Engineering Hyderabad, India, nallaboluvaruun@gmail.com*

**S. Meghamsh**
*Vardhaman College of Engineering Hyderabad, India, meghamsh25@gmail.com*

**G. Abhishek**
*Vardhaman College of Engineering Hyderabad, India, abhishek.gourishetty@gmail.com*

**Abstract**

Every aspect of our modern life is interconnected to the internet leading to the creation of massive volumes of data that we cannot store on our smartphones or other storage devices owing to their size. An effective way to store enormous amounts of is using cloud storage technology, usage of these services has been rising quickly lately. Many businesses are switching from conventional storage techniques to cloud storage as it makes it simple to access information for users whenever and anywhere needed. Different cryptographic methods are used to safeguard the data. This paper proposes a hybrid cryptographic algorithm that utilizes the AES (Advanced Encryption Standard) algorithm and the Blowfish algorithm. Our proposed cryptographic algorithm has higher speeds for executing encryption and decryption processes. This rate increase is achieved by implementing multiprocessing in their execution. Results demonstrate that our hybrid approach has faster execution times and greater throughput.

**Keywords:** *Hybrid cryptography, AES, Blowfish, Throughput.*

## I. INTRODUCTION

The steady increase in demand has made cloud computing a hot topic in recent years. Moving toward cloud-based data storage solutions has several benefits for businesses. These include remote access from practically anywhere globally with a reliable Internet connection, simpler IT infrastructure, and administration.[1]The cloud offers many services, such as infrastructure, platforms, and software, but the most significant issue is ensuring security for massive data on the cloud. In general, medical, military, and government data typically contain sensitive information that is kept in the cloud. Still, the user is unsure about the security offered by the service providers. Many resources in the cloud, such as networks, operating systems, databases, and memory management, are vulnerable to various assaults. As a result, security and privacy are critical in cloud applications[2]. Reasonable methods must be followed to transform data into string words people cannot understand to preserve the stored information. Cryptography refers to the techniques used to do this in the computer world. Cryptography is converting an original communication into an unread

message that an individual cannot comprehend. In general, cryptography is challenging.[3] Using only one algorithm is ineffective for the high-level security required for cloud computing data. Because the user's application and program are hosted by the provider, concerns regarding file security are raised. This problem may be resolved by the cloud provider by encrypting the data using an encryption technique. To effectively address the basic issue of security in the local system environment, a file security model is needed.[4]The phrase "hybrid cryptography" refers to combining two or more cryptographic techniques. by combining the two separate algorithms' speed and strength to increase the encryption's capability. This technique is used to ensure the security of file storage systems.[5]

## II. RELATED WORK

This section addresses some of the current approaches to hybrid cryptographic algorithms. In [6] Files on the device will be encrypted using the password-based AES technique in this article. The user may also download and view any encrypted files submitted to the system. AES is used for encryption as it is not susceptible to attacks except Brute Force attacks. However, even a supercomputer cannot perform a Brute Force assault. AES is also far quicker. As a result, it is an excellent solution for cloud data protection.

In[7] a survey to analyze multilevel encryption used in the cloud, various ciphers are carefully investigated and concluded that multilayer encryption enhances security compared to single encryption algorithms. In[8] different hybrid cryptographic algorithms, various models' implementations, and designs are discussed to enhance efficiency. In [9] both Vigenère and Polybius square ciphers are used; initially, the Vigenère cipher creates a ciphertext by randomly choosing a key. The following ciphertext eventually becomes a key for Polybius square cipher, and the final ciphertext is generated by applying the key to plaintext. To retrieve the message back, entire process is done in reverse order. This [10] study will demonstrate the implementation of cloud storage security by implementing a hybrid encryption method and hash functions. It implements two algorithms (AES) with a secure hashing technique.

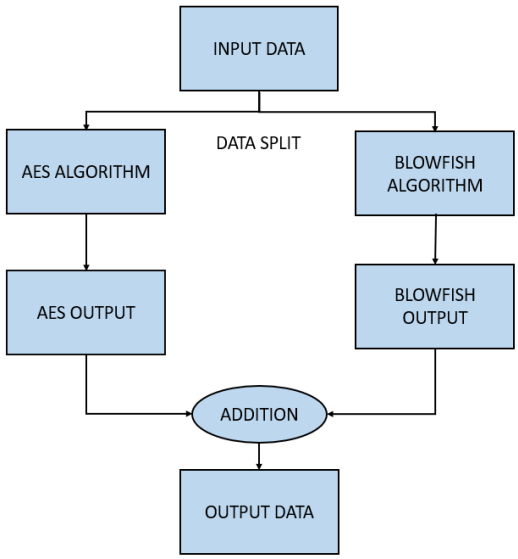
In [11] they presented a hybrid method that combines RSA and Diffie-Hellman, The Diffie-Hellman keys are used before and after transmitting to change and recover the original encryption text. The Diffie-Hellman keys are the original encryption text.

In [12] AES, RC6, Blowfish, and BRA algorithms provide block-wise security, and all these require a single key of the same size for file encoding and decoding. LSB steganography is used to conceal crucial information in the process. [13] The fundamental goal of this provided work is to separate the file into four different sections and store the individual parts on other cloud servers. The optimization of computing resources, such as execution time and solution optimization, leads to the modification of encryption algorithms utilizing a hybrid approach.

## III. PROPOSED METHOD

Initially, input data is divided into two equal parts; after the plaintext is broken down into two equal parts, two different encryption algorithms are applied to each. Here, the multiprocessing approach enables independent parallel operation of this encryption or decryption process. Due to the use of multiprocessing, the process of encryption or decryption is completed relatively faster.

**Figure1 Block diagram of the proposed model**




In this work, both encryption and decryption processes have different flows of execution, but most of the processes in their execution are similar to each other

Encryption:

The encryption process begins with a plain text file as input, which the program reads using file-handling methods in Python. The contents are stored and split into two equal parts using slicing operations. The user is prompted to enter a key, which is essential for the encryption algorithms. AES encryption is applied to the first half of the data, and Blowfish encryption is used to the second half. Due to the independence of the two halves, multiprocessing is used to execute each part's encryption concurrently. The outputs are then merged using string concatenation techniques to produce the ciphertext. The ciphertext is saved into a new file and then uploaded to the selected cloud provider.

Decryption:

The decryption begins by taking an encrypted file input from a cloud provider or the local device. The file's contents are read, and the user is asked to enter the key used during the encryption process. Once the key is entered, it is verified. An error is thrown if the entered key does not match the key used in the encryption process. The encrypted data is then broken down into two parts and decrypted using AES and Blowfish ciphers in parallel, utilizing the multiprocessing feature of Python. After completing both processes, the outputs are combined using string concatenation methods. The final result is inserted into a newly created file, whose content is the same as the original plaintext input.

Multiprocessing:

Multiprocessing is a built-in python package that allows for multiple processes to run simultaneously, with each process executing independently. Multiprocessing divides programs into smaller chunks of code that operate independently of one another, increasing the system's performance[14] [15]The primary benefit of multiprocessing is that it provides an option for parallel processing, which can decrease the execution times of a program.

This study uses two algorithms: the Advanced encryption algorithm and the blowfish algorithm. Each of these algorithms is briefly explained here.

AES (Advanced Encryption Algorithm):

The Rijndael algorithm, famous as the AES algorithm, is symmetric encryption, suggesting that the same key is used in encryption and decryption. This is a block cipher, meaning input is accepted in blocks, and any actions conducted are performed on these blocks. AES uses a fixed block size of 128 bits and supports key sizes of 128, 192, or 256 bits[16]. The

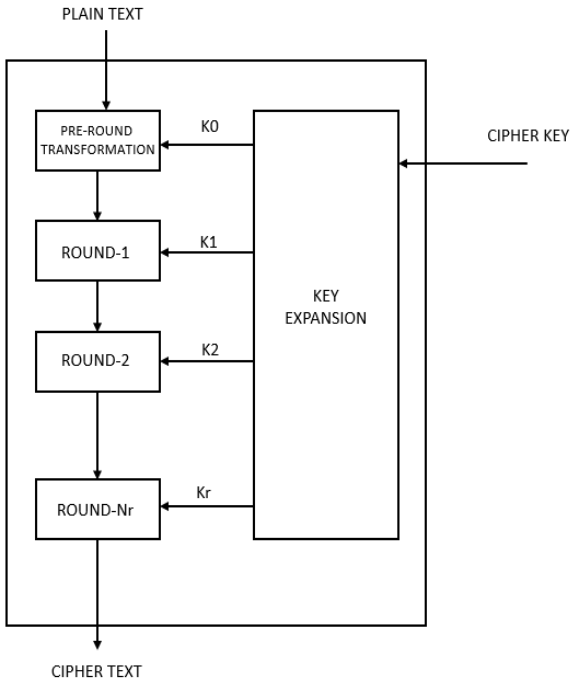
schematic of the AES structure is shown in figure 2.

It is based on a series of operations such as substitution and permutation (permutation here means exchanging bits around). Although size is calculated in bits, all actions performed on them are done in bytes. Input essential length can be 128,192,256 bits, which determines the number of rounds it goes through [17]. In the process of encryption, 128-bit plain text undergoes four transformations in each round. In figure 3 the first round process is depicted. Those transformations are:

1. Byte substitution:

The data block in the AES algorithm is 128 bits long, which implies that each data block comprises 16 bytes. In sub-byte transformation, each 8-bit in a data block is converted into a different block using an 8-bit (Byte) substitution box called the Rijndael S-box.

**Figure 2 shows the implementation of the AES algorithm**




2. Shift rows :

The matrix's four rows are rotated to the left and results in16-byte matrix.

3. Mix columns:

This is a straightforward replacement procedure. Each matrix column is modified using matrix multiplication. Following this operation, the output will be a matrix containing sixteen additional bytes.

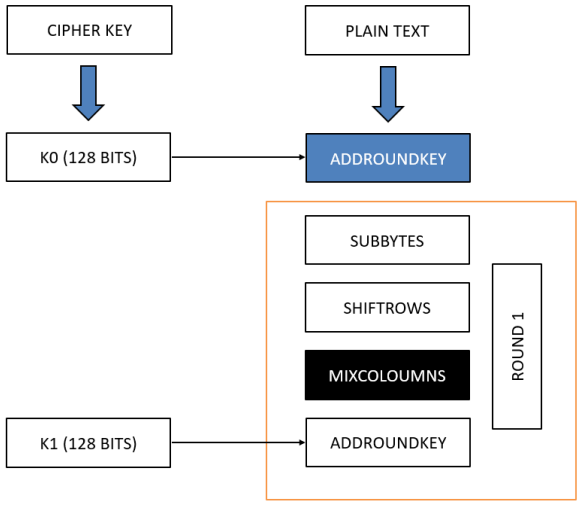
4. Add round key:

Int this stage xor is performed between output of previous stage with corresponding key .Each round repeats these four steps a total of 10, 12, or 14 times, depending on the key size, which may be 128, 192, or 256 bits. All rounds mentioned above are performed except in the final round. The Mix Columns step is omitted.

The decryption process of AES is similar to encryption but should be done in reverse order.

AES is the most secure and scalable encryption method for protecting sensitive data[18]. It is widely adopted by many governments, financial institutions, and other organizations to protect sensitive information.

**Figure 3 depicts the first round process**



Blowfish Algorithm:

Blowfish is a symmetric block encryption [19] cipher with a variable length key that ranges from 32 bits to 448 bits. It encrypts blocks of 64-bit data at a time. It is based on the [20] Feistel network, and its operation is separated into two stages.

Those stages are:

1. key expansion

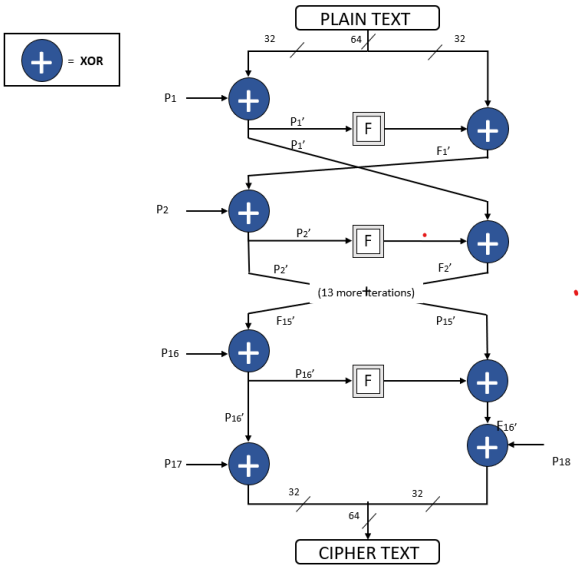
The key expansion stage converts the input key into several sub-key arrays. subkeys are named k1, k2, and Kn; where n lies between 1 and 14. A P-array is also initialized with each element of the array being 32-bit in size, digits of pi are put into these P-array. Once the p-array elements are assigned, they are xor-ed with individual subkeys, and finally modified P-array is formed having elements from P1, P2,…,and P18.

Four various s-boxes[21] are also created, each having 256 entries of 32 bits, and these initialized s-boxes are used during encryption and decryption.

2. Data encryption or decryption

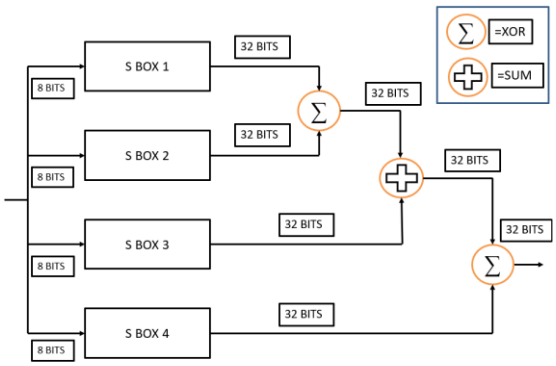
Input plain text of 64 bits is broken down into two 32-bit parts, To create the value P', [22]the "left" 32 bits of the message are XORed with the first element of a P-array. The right 32 bits of the message are then put via a transformation function called F to create a new value, F'. The "left" half of the message is replaced by F', and the "right" half is replaced by P', 15 times with following members of the P-array. To create the 64-bit ciphertext, the resulting P' and F' are then XORed with the last two elements of the P-array. Figure 4 shows a visual depiction of the blowfish algorithm.

**Figure 4 shows the working of the Blowfish algorithm**




The Transformation function [23] shown in figure 5 takes 32-bit input and gives a 32-bit output by applying bit output by using addition, XOR with S-boxes in them.

**Figure 5 shows the implementation of the Transformation function(F)**




The decryption process is alike to encryption, during decryption keys are applied in reverse order[24]. The Blowfish algorithm is a symmetric encryption algorithm that is considered one of the most robust defenses against hackers trying to penetrate the security of software developed.

## IV. IMPLEMENTATION

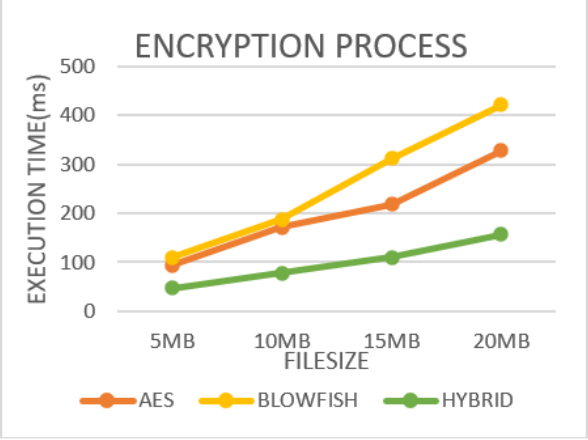
Visual studio code 1.74.3 was used for analysis on a laptop having Intel(R)Core (TM), i5-10210,8 GB RAM, Windows 11, 64-bit operating systems. visual studio code is a lightweight but powerful source code editor that runs on desktops and is available for Windows, macOS, and Linux. We used Azure as the cloud platform for project implementation. Various text file inputs of varying sizes, 5 MB,10 MB,15 MB, and 20 MB, are given as inputs, and execution times for all algorithms for various file sizes are calculated.

## V. EXPERIMENTAL RESULTS AND ANALYSIS

The performance of AES and Blowfish algorithms are compared with the proposed algorithm using their execution times and their respective throughput values. Execution times and throughput values are calculated and graphically represented in Figures 6, 7, and figure8 below.

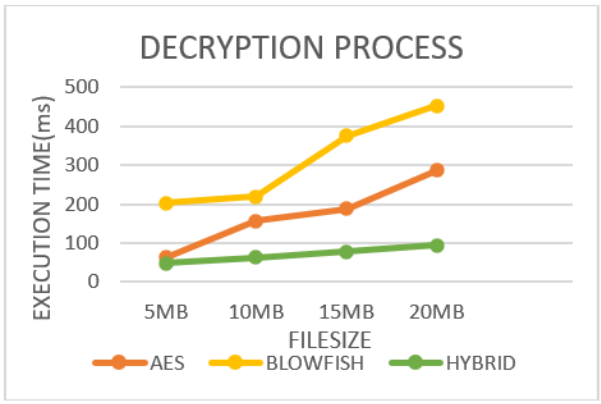
Execution time:

**Figure 6 shows execution time of the encryption process of various algorithms for given file sizes.**




The execution time-file size graph illustrates it with various input file sizes on the x-axis and their respective execution time of the encryption process on the y-axis. Figure 6 confirms that the execution time of the proposed method for all given file sizes is less execution time for both AES and Blowfish algorithms.

**Figure 7 shows execution time of various algorithms' decryption for various file sizes.**




The execution time-file size graph illustrates it with various input file sizes on the x-axis and their respective execution time of the decryption process on the y-axis. Figure 7 illustrates that for all given file sizes, the proposed method has less execution time for both AES and Blowfish algorithms, making it an ideal choice for any practical application.

Throughput:

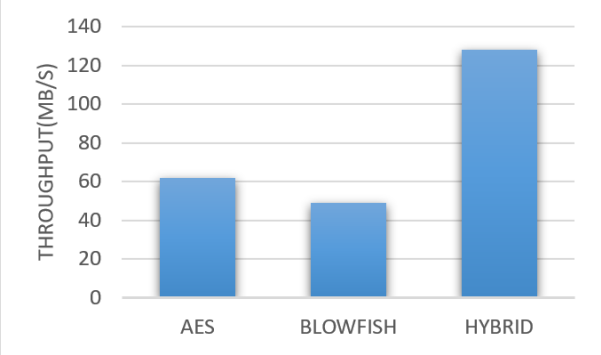
Here, the throughput of a cryptographic algorithm is calculated by taking the ratio of the size of plain text to be encrypted in megabytes (MB) to the encryption time taken for a given plain text.

$$\text{Throughput} = \frac{\text{Tp(MegaByte)}}{\text{Et(seconds)}}$$

Tp= Total amount of plain text to be encrypted.

Et= Total execution time of encryption process.

**Figure 8 shows the throughput values of various algorithms.**




The throughput graph illustrates algorithms used on the x-axis and their respective throughputs on the y-axis. Throughput indicates the speed of encryption and helps determine power consumption A cipher with higher throughput consumes less power [25] and vice versa .

Figure 8 shows the supremacy of the hybrid algorithm when compared to AES and Blowfish. The proposed algorithm has a throughput of 128 MB/S compared to 62 MB/S of AES and 49 MB/S of the Blowfish algorithm.

**Table1 shows algorithms and their respective values**

| ALGORITHM | THROUGHPUT |
|---|---|
| AES | 62 |
| BLOWFISH | 49 |
| HYBRID | 128 |

Table 1 shows that the throughput of the hybrid algorithm is calculated as 128 MB/S which is 51.5 % higher than the throughput of AES and 61.71% higher than the Blowfish algorithm. As discussed above higher throughput value means low power consumption of the algorithm; therefore, due to better performance and high throughput value proposed hybrid algorithm can be widely used among various applications.

## VI. CONCLUSION

Because cloud storage is among the most extensively utilized products in practically every industry, its security is one of the significant customer concerns. It is inferred through opposing experiments that the hybrid encryption algorithm enhances the efficiency of encryption, decryption, and achieves a higher throughput. AES is the most secure algorithm with no known exploit. Still, blowfish has an exploit known as a dictionary attack (all the words in the dictionary are given as passwords), a type of brute force attack. Still, usage of a combination of both AES and Blowfish keeps it safe from any form of brute force attack as AES resists any form of brute force attack. Aside from ensuring secrecy and usability, hybrid cryptosystems give a high level of security as the data is stored in the cloud as ciphertext rather than its original form. The hybrid method put forward in this study has applications in device architecture, software application, and other areas.

## Reference

[1] H. Tabrizchi and M. Kuchaki Rafsanjani, “A survey on security challenges in cloud computing: issues, threats, and solutions,” Journal of Supercomputing, vol. 76, no. 12, pp. 9493–9532, Dec. 2020, doi: 10.1007/s11227-020-03213-1.

[2] I. A. T. Hashem, I. Yaqoob, N. B. Anuar, S. Mokhtar, A. Gani, and S. Ullah Khan, “The rise of ‘big data’ on cloud computing: Review and open research issues,” Information Systems, vol. 47. Elsevier Ltd, pp. 98–115, 2015. doi: 10.1016/j.is.2014.07.006.

[3] A. D. Achmad, A. A. Dewi, M. R. Purwanto, P. T. Nguyen, and I. Sujono,

"Implementation of vigenere cipher as cryptographic algorithm in securing text data transmission," Journal of Critical Reviews, vol. 7, no. 1, pp. 76–79, 2020, doi: 10.22159/jcr.07.01.15.

[4] S. Gokulraj, P. Ananthi, R. Baby, and E. Janani, "SECURE FILE STORAGE USING HYBRID CRYPTOGRAPHY." [Online]. Available: https://ssrn.com/abstract=3802668

[5] A. K. Bermani, T. A. K. Murshedi, and Z. A. Abod, "A hybrid cryptography technique for data storage on cloud computing," Journal of Discrete Mathematical Sciences and Cryptography, vol. 24, no. 6, pp. 1613–1624, 2021, doi: 10.1080/09720529.2020.1859799.

[6] D. Mukhopadhyay, P. S. Gupta, G. Sonawane, S. Gupta, S. Bhavsar, and V. Mittal, "Enhanced Security for Cloud Storage using File Encryption Spam 2.0 View project Cellular Automata usage in Designing Search Engine View project Enhanced Security for Cloud Storage using File Encryption," 2014. [Online]. Available: https://www.researchgate.net/publication/258818843

[7] V. Ghorpade and V. Dalimbkar, "A Survey Paper on Data security in Cloud Computing," International Journal of Computer Sciences and Engineering International Journal of Computer Sciences and Engineering, 2016, [Online]. Available: www.ijcseonline.org

[8] M. K. Sinchana and R. M. Savithramma, "Survey on Cloud Computing Security," in Lecture Notes in Networks and Systems, vol. 103, Springer, 2020, pp. 1–6. doi: 10.1007/978-981-15-2043-3_1.

[9] North Eastern Hill University. Department of Biomedical Engineering, Institute of Electrical and Electronics Engineers. Kolkata Section, IEEE Industry Applications Society, and Institute of Electrical and Electronics Engineers, International Conference on Computational Performance Evaluation: ComPE 2020 online conference: 2nd-4th July 2020.

[10] N. M. Abdelnapi, F. A. Omara, and N. F. Omran, "A Hybrid Hashing Security Algorithm for Data Storage on Cloud Computing." [Online]. Available: https://sites.google.com/site/ijcsis/

[11] S. Kalyan Ghosh, S. Rana, A. Pansari, J. Hazra, and S. Biswas, "Hybrid Cryptography Algorithm For Secure And Low Cost Communication," 2020.

[12] S. P. and N. 2016 C. IEEE International Conference on Wireless Communications, Institute of Electrical and Electronics Engineers, S. P. and N. 2016. 03. 23-25 C. IEEE International Conference on Wireless Communications, WiSPNET 2016.03.23-25 Chennai, and IEEE WiSPNET 2016.03.23-25 Chennai, Proceedings of the 2016 IEEE International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET) 23-25 March 2016, Chennai, India.

[13] R. Parab, A. Paul, U. Mojumdar, and R. Patil, "SECURED CLOUD STORAGE USING HYBRID CRYPTOGRAPHY," 1330. [Online]. Available: www.irjmets.com

[14] Z. A. Aziz, D. Naseradeen Abdulqader, A. B. Sallow, and H. Khalid Omer, "Python Parallel Processing and Multiprocessing: A Rivew," Academic Journal of Nawroz University, vol. 10, no. 3, pp. 345–354, Aug. 2021, doi: 10.25007/ajnu.v10n3a1145.

[15] M. Arif Sazali, M. Syahir Sarkawi, and N. Syazwani Mohd Ali, "Multiprocessing

implementation for MCNP using Python," IOP Conf Ser Mater Sci Eng, vol. 1231, no. 1, p. 012003, Feb. 2022, doi: 10.1088/1757-899x/1231/1/012003.

[16] A. Muhammad Abdullah and A. Muhamad Abdullah, "Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data Call for papers View project Application of Petri Nets in Computer Networks View project Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data," 2017. [Online]. Available: https://www.researchgate.net/publication/317615794

[17] TamilselviS, "Data Storage Security in Cloud Computing Using AES."

[18] R. Mote, A. Pawar, and A. Dani, "Review of security and privacy techniques in cloud computing environment," in Smart Innovation, Systems and Technologies, 2016, vol. 50, pp. 543–551. doi: 10.1007/978-3-319-30933-0_54.

[19] A. Narang et al., "Blowfish-Symmetric Key Cryptography Algorithms Article in," 2015. [Online]. Available: https://www.researchgate.net/publication/317719444

[20] V. Parihar and M. A. Kulshrestha, "BLOWFISH ALGORITHM: A DETAILED STUDY," 2016. [Online]. Available: www.ijtre.com

[21] A. Malhotra, A. Arora, and Dr. M. K. Bhatia, "Symmetric Cryptographic Approaches," Int J Res Appl Sci Eng Technol, vol. 10, no. 12, pp. 718–721, Dec. 2022, doi: 10.22214/ijraset.2022.47982.

[22] M. Suresh and M. Neema, "Hardware Implementation of Blowfish Algorithm for the Secure Data Transmission in Internet of Things," Procedia Technology, vol. 25, pp. 248–255, 2016, doi: 10.1016/j.protcy.2016.08.104.

[23] M. Nehakhatri -Valmik and V. K. Kshirsagar, "Blowfish Algorithm." [Online]. Available: www.iosrjournals.org

[24] K. Reddy, U. Thirupalu, R. Scholar, and E. K. Reddy, "Performance Analysis of Cryptographic Algorithms in the Information Security; Performance Analysis of Cryptographic Algorithms in the Information Security", doi: 10.13140/RG.2.2.16273.51047.

[25] M. G. Rashed, M. A. F. M. R. Hasan, R. Islam, M. Golam Rashed, and R. Yasmin, "A Comparative Analysis of Various Cryptographic Algorithms Ensuring Secrecy and Authenticity of Exchanged Information," 2021. [Online]. Available: https://www.researchgate.net/publication/349711184