# ANALYSIS OF SECURITY ENHANCEMENT IN CLOUD STORAGE USING EFFECTIVE ENCRYPTION TECHNIQUE

**[1]Dr. S. Sujiya, [2]Mr. A. Srinath, [3]Mr. K. Srinath**

[1]Assistant Professor, Department of Computer Applications, Dr. SNS Rajalakshmi College of Arts & Science, Coimbatore.

[2]PG Student, II MCA, Department of Computer Applications, Dr. SNS Rajalakshmi College of Arts & Science, Coimbatore.

[3]PG Student, II MCA, Department of Computer Applications, Dr. SNS Rajalakshmi College of Arts & Science, Coimbatore.

**Abstract:**

*The world of information technology is constantly expanding. In line with the needs of community members such as security, processing, fast access and most importantly, saving money is very important. These days a solution to such problems has been introduced by the concept of cloud computing. Despite of its unique advantages, the security challenges such as data disclosure, privacy, attacks on servers, unsafe communication and resource sharing, can't be ignored. Thus, security is one of the main issues in cloud computing. Cloud computing is a fastest growing technology. It allows business organizations to use or access different applications, store information without access their personal files. While considering the power, stability and the security of cloud one can't ignore different threats to user's data on cloud storage. In cloud storage system file entrance mechanism is more challenging issue. This system in consequence produces redundant copies of similar files or involves a completely reliable cloud server. Attacks from adversary user are difficult to stop in cloud storage. The main Objective of this Paper is to analyse how the security is provided by two basic encryption techniques i.e. HMAC and SHA. The various encryption techniques used to prevent the attack from malicious users discussed with applying the enhanced technique to protect the database.*

*Keywords: - Cloud, Environment, Encryption, Data, Security, Attacks.*

## 1. INTRODUCTION

Cloud computing can be provided as a service by the service provider assessment. Users do not need to know about how services are provided (e.g. network, storage and software). Instead, the user's only concern is that services are available whenever needed. The term cloud computing is chosen for that all the details are hidden from the user's perspective. The user doesn't need to have knowledge about the infrastructure of the cloud which is used. Due to the novelty of this technology is not yet a precise definition for it. Cloud computing is a model for providing easy access to the application's user [1]. Through a changeable series of computational grid, such as nets, servers, storage space of applied program, and services." This access can be provided and released immediately with minimal need to management or direct involvement of Service Provider.

Three aspects of security are confidentiality, integrity and availability. Confidentiality is hiding information and resources. Integrity refers to the trustworthiness of data or resources, which usually prevents any incorrect or unauthorized changes. Availability refers to the ability to use the information or resource [2]. Thus, security is one of the main issues in cloud computing, as originally expected only people whose identities have been authenticated by client can access data. Given the above comments, there will be two main concerns in this area:

**External attackers:** Any person whose identity is authenticated to access the data by the client. Cloud Providers: These centres can break their pledge, to gain unauthorized access to data. Cryptographic are methods or techniques to protect data by changing its format to other formats, which is not easily understood by unauthorized users. Also, methods such as, microdots and combining words with images can be used. Encryption methods can be divided into two sets of

symmetric key algorithm and asymmetric key algorithm. Symmetric key algorithm is also known as the private or secret key.

In symmetric key algorithm sender or receiver use one key to encrypt and decrypt the data. Symmetric key algorithm is divided to the stream ciphers and block ciphers split encryption [3].

**Cryptography objectives:**

- **Confidentiality:** This service is used securely to protect the contents of the message and not allow to access information to any third person other than the person.
- **Authentication:** It is related to the identification and authentication. Parties that communicate with each setup will be able to identify each other. Information receiver system, verifies the identity of the sender to see whether the information was sent by the authorized person or another person.
- **Data integrity:** This service prevents unauthorized modification of data. This means that neither the sender nor the receiver will not be able to change the message.
- **Non repudiation:** This service allows the sender and receiver not to deny sending and receiving messages.
- **Access Control:** This service controls information access and only allows authorized persons to access data. Encryption techniques can be broadly classified into three categories as follows: symmetric key algorithms (Private Key), the asymmetric key algorithm (public key) and hybrid key algorithm. In symmetric algorithm, the transmitter and receiver both use the same key to encrypt or decrypt. This algorithm is divided into block ciphers and stream ciphers. Symmetric key algorithm is very fast due to its relatively low complexity, is easy to implement and run. Some of the most symmetric key algorithms include: DES, 3DES, AES, Blowfish, Twofish, Serpent, SEED, IDEA, RC2, RC4 and RC6.

Asymmetric key algorithms, also called public key algorithm. In this algorithm, both the sender and receiver used two different keys, public key and private key, to encrypt and decrypt data. Sender used receiver's public key to encrypt the plain text to cipher text. Receiver used their own private key, to descript cipher text to original text [4]. The private key which is used is always confidential. Asymmetric key algorithms are slower than symmetric key algorithms, and have higher computational burden. RSA is the most famous asymmetric key algorithm. Other asymmetric key algorithms include: Diffe_Hellman, DSA, EIGamal, XTR, and ECDSA.

In modern encryption system combination of a symmetric key asymmetric key algorithm is used for encryption and decryption process. Asymmetric key algorithm is used to distribute a symmetric key at the start of the session. When the symmetric key was known in session, encryption is able to operate very quickly. This algorithm is mainly having the problem of key distribution [5]. In each section or subsection, there are one or several paragraphs. Note that the sentences in each paragraph chain together and pursue a topic.

### 1.1. Cloud Computing Architecture

Basically Cloud computing includes following entities:-

- Customers: the entities who are using the services.
- VM:VM stand for virtual Machine, which is the medium to make interaction between the customers and the CSP.
- Cloud service Provider (CSP):CSP plays important role to upload the data of customers on the Virtual Servers.
- Virtual Server:-Virtual server are the database servers on which database is actually stored.
- Third Party: Third party server is used to reduce the overhead of the CSP.

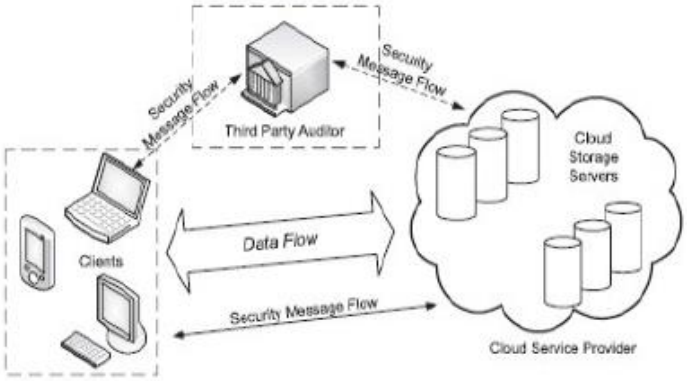


**Fig 1.1: - Architecture of Cloud Computing**

### 1.2. Deployment models in Cloud computing:-

There are four types of clouds available in the cloud computing. Private, Public, Hybrid, and Community Cloud [6]. These Deployment models describe who owns, and is responsible for the services.

1. **Private cloud:** The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g. business units). It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises.
2. **Public cloud:** The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises of the cloud provider.
3. **Community cloud:** The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises.

## 2. SECURITY MODEL IN CLOUD COMPUTING

This paper presents a data security model for cloud computing, provides encrypted data services and reliable mass data security support. The platform used in this paper is second development software platforms [7, 8]. The interface required by the data security system provided by cloud computing, through it to achieve organic integration with other applications.

### 2.1. Function of the Model

The security features of the model that proposed by this paper include encryption and decryption of data, cipher text retrieval and integrity verification [9, 10].

1) **Data encryption and decryption:-** The cloud users could use this model to encrypt and decrypt data. The owner of the data holds a master key, could completely access data in the files, other users could not crack cipher text content because of no keys. In order to guarantee the security of the key, users don't need to manage or save a lot of keys' information. During decryption, key file should securely store and could not repeat. The model has the ability to transmit data and decrypt data simultaneously.
2) **Cipher text retrieval:-** The model could provide searching services for encrypted data that stored in cloud platform, and could support retrieval based on file properties and file contents. Retrieval based on file properties will search data through file name, owner, file creation time, file type and other general properties and user-defined attribute. Retrieval based on file content will search data through keywords that automatically generated from the file content. This model could support retrieval based on file attributes and cipher text attribute. The access control system controls the retrieval range that comprises the user's own files and the authorization files.

3) **Massive data integrity verification:-** Integrity verification of the model is mainly responsible for the users to verify the integrity of files. This task initiated by the users and could repeatedly verify. Comparison of authentication credentials information that read from the server and the results that calculated by key and the server data, then according to the comparison results to determine the integrity of the document.

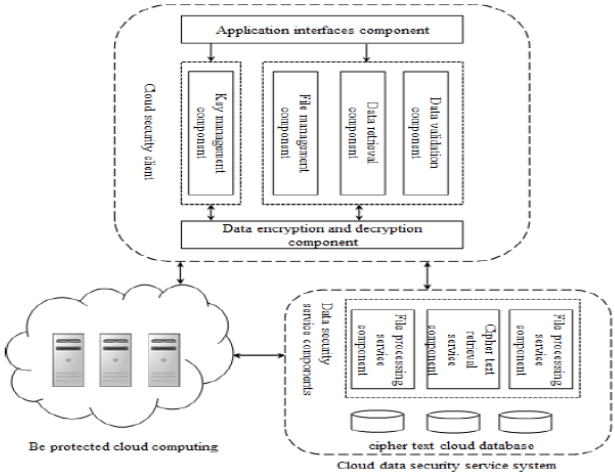


**Fig 2.1: - Structure of Cloud Model**

### 2.2. Structure of the Cloud Security Model

The cloud computing data security model that present by this paper consists of three parts (Fig 2.1. shows the specific composition):

1) Be protected cloud computing (cloud storage system). The encrypted data stored in cloud storage system, and non-authorized data transmission could not directly cause leakage.
2) Cloud data security service system. This system was responsible for the generation, storage, maintenance and management of cloud computing data attributes and other meta-data. The data security service system could provide data retrieval, data integrity verification and other services.
3) Cloud security client. The client was responsible for locally save cloud user key and data encryption and decryption, to avoid key leakage causes data security risks.

## 3. HASH MESSAGE AUTHENTICATION CODE (HMAC)

Hash based message authentication code is cryptographic hash function which is all about the concatenation of message and the key and hash them together. It is the method of calculating message authentication code with cryptographic hash function by using secret cryptographic key. The hash algorithm used to generate the authentication code is SHA. The authentication code used to verify the data integrity and authentication of the message using the security key which is necessary for producing the code. The authentication code produced by the normal hash function can be reproduced without any normal constraints [11]. The cryptographic strength of the hash function, the size of the hash output and the size and quality of the key determines the cryptographic strength of HMAC. HMAC doesn't serve the purpose of being a provider of message integrity by itself. It is one of the components in the protocol that provides message integrity. Though the HMAC is not designed to encrypt the message itself it serves as a protection shield for man in the middle attack. HMAC supports hash algorithms like MD5, SHA –1, SHA – 256 and etc.

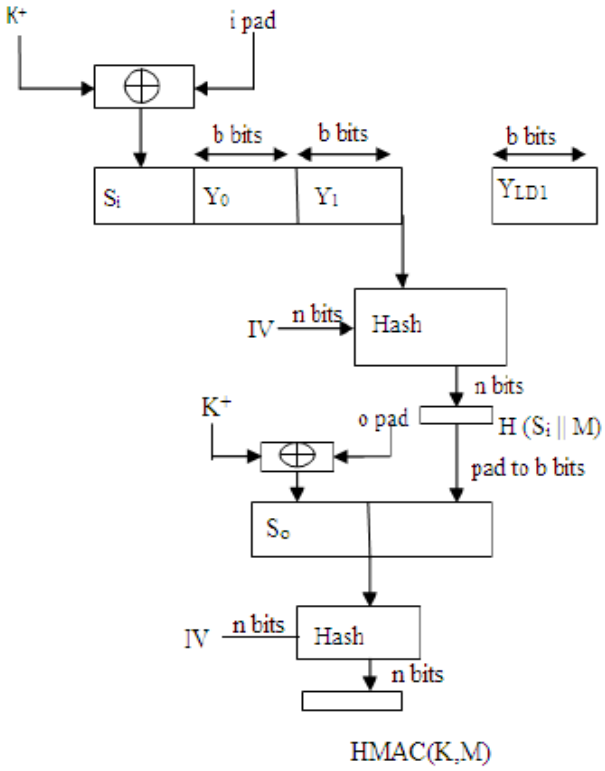


**Fig 3.1: - HMAC Algorithm Function**

K+ is K padded with zeros on the left so that the result is b bits in length ipad is a pad value of 36 hex repeated to fill block opad is a pad value of 5C hex repeated to fill block M is the message given as input to HMAC [12].

The output binary authentication code which equals in the length to that of the hash function digest.

The data integrity of the file is checked by comparing the value of hash function in both user and auditor. In HMAC, the generation of authentication code uses secret and hash based algorithm. This code ensures the usage of hash function is expanded in many places. It conserves the performance of hash function. HMAC is a symmetric process which makes use of secret and hash algorithm for the generation of authentication code. The authentication code is the core factor which ensures data integrity and authenticity because a secret key is necessary to reproduce the authentication code [13]. HMAC is used because it ensures the usage of hash functions without any modifications and these hash functions can be used in any software that is widely available. The HMAC provides the advantage of preserving the original performance of the hash function without subjecting it to any degradation.

## 4. SECURED HASH ALGORITHM (SHA)

Secure Hash Algorithm (SHA) is the most widely used Hash Function in the world. It was developed by the National Institute of Standards and Technology (NIST) in the United States and first published in 1993. The original version (SHA- 0) was found to have a serious security flaw and replaced in 1995 with SHA-1.As computing power has increased, this too is found to have weaknesses and the possibility of collisions has been identified. To further enhance security, SHA-2 was released in 2001 containing improvements in the message computation and hash output size [14]. To date, no weaknesses have been identified in SHA-2. Like all Hash Functions, SHA outputs a fixed length digest of a message with arbitrary input length. The original message is converted to blocks of a fixed size, which are sequentially reduced. This is repeated for the entire file, giving a message digest.

This is an algorithm that uses one way hashing function. It is used to generate a hash digest which is used as a fingerprint for the data. The data is given to the function which produces a value (h1), a hash value. This value (h1) is

sent to the receiver. On the other side, the receiver also passes the message to the function to generate a value (h2), a hash value. The receiver compares the values h1 and h2 as shown in Fig3. If the values, h1 and h2 are same then the data is safe otherwise the data is tampered.

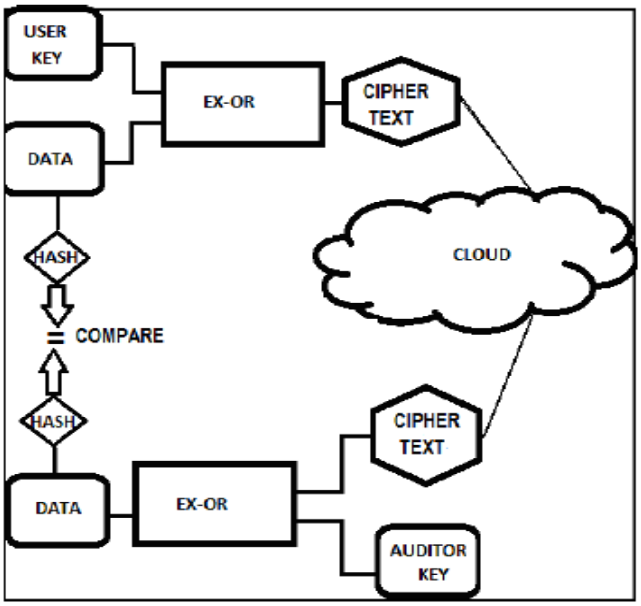


**Fig 4.1: - SHA Architecture**

First of all, a secret key is generated using the DH algorithm and the same key is known by both the user and his auditor. After that an XOR operation is done between the data and the key generated. Separately the data is passed in a hash function (using SHA1) and the hash value is obtained by the sender. The auditor on behalf of the user will check the integrity of the data stored in the cloud [15, 16]. The auditor gets the cipher text from the cloud and performs an XOR operation with the secret key generated by the DH algorithm and gets a plain text. The auditor passes this plain text to the same hash function (using SHA1) and obtains a hash value .He then compares this hash value with the hash value received from the user .If both the values are identical then the data integrity is maintained or else the data integrity is tampered.

**SHA1 Features:**

- By using SHA1 message digest for a given message or data block is obtained.
- Bit string is taken as input which can be either message or data file.
- Any number of bits can be taken in the message.
- For compactness Message is represented in hexformat if number of bits is multiple of 8.
- Message padding is done for making the total length of a padded message a multiple of 512.
- When computing the message digest SHA1processes the blocks of 512 bits sequentially.
- The SHA1 algorithm then processes padded message as 512-bit blocks.

## 5. PERFORMANCE EVALUATION

Several performance metrics are collected: encryption time, CPU process time, and CPU clock cycles and battery power. The encryption time is considered the time that an encryption algorithm takes to produce a cipher text from a plaintext. Encryption time is used to calculate the throughput of an encryption scheme. It indicates the speed of encryption. The throughput of the encryption scheme is calculated as the total plaintext in bytes encrypted divided by the encryption time [17]. The CPU process time is the time that a CPU is committed only to the particular process of calculations. It reflects the load of the CPU. The more CPU time is used in the encryption process, the higher is the load of the CPU.

The CPU clock cycles are a metric, reflecting the energy consumption of the CPU while operating on encryption operations. Each cycle of CPU will consume a small amount of energy.

The following tasks that will be performed are shown as follows:

- A comparison is conducted between the results of the selected different encryption and decryption schemes in terms of the encryption time at two different encoding bases namely; hexadecimal base encoding and in base 64 encoding.
- A study is performed on the effect of changing packet size at power consumption during throughput for each selected cryptography algorithm.
- A study is performed on the effect of changing data types -such as text or document and images- for each cryptography selected algorithm on power consumption.
- A study is performed on the effect of changing key size for cryptography selected algorithm on power consumption.

**5.1. Cryptographic algorithm metric description**

There are various cryptography and Steganography mechanisms available in related work for ensuring the safety of user data from being cracked by eavesdroppers; existing mechanisms are implemented on text or image files [18]. Each technique has some limitations due to openness and vulnerabilities in the architecture of internet and they are unable to defend data from bruit force techniques.

**1) Encryption Time**

The security of a symmetric cryptosystem is a function of the length of the key. The longer the key, the more resistant the algorithm is to a successful brute force attack. For this reason, key length was chosen as the first parameter for specifying cryptographic algorithms.

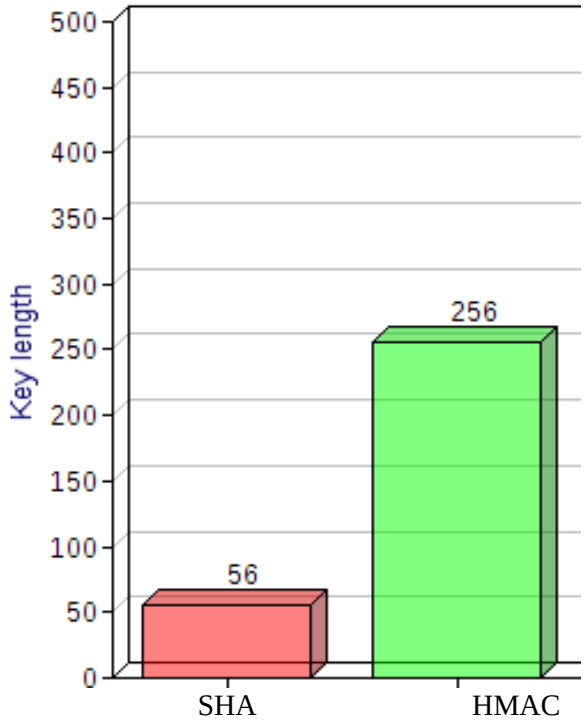


**Fig 5.1: - Encryption time**

**2) Throughput**

Attack Steps is defined as the number of steps required to perform the best known attack. The number of steps helps determine the time that might be required for a successful attack, using a particular processor, without having to actually run the attack on the algorithm, which may not be feasible.

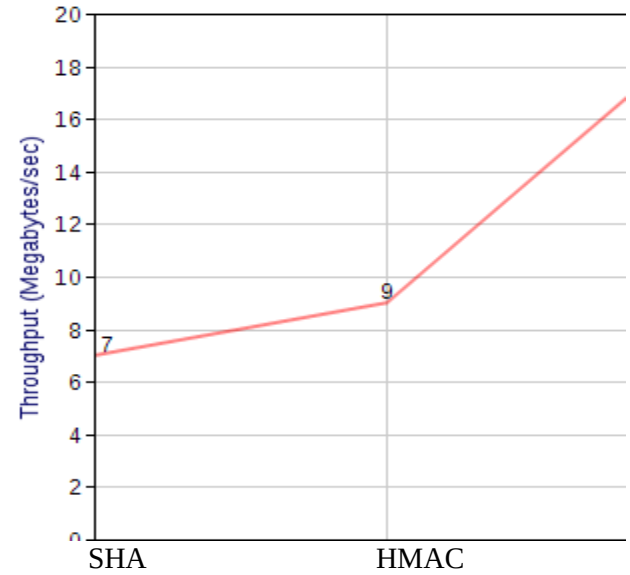


**Fig 5.2: - Throughput of Algorithm**

**3) CPU Cycle**

Rounds by themselves may not have great value in specifying meaningful thresholds. (A one-time pad effectively has 1 round and a block size of 1 bit.) However, rounds are important to the strength of some ciphers.

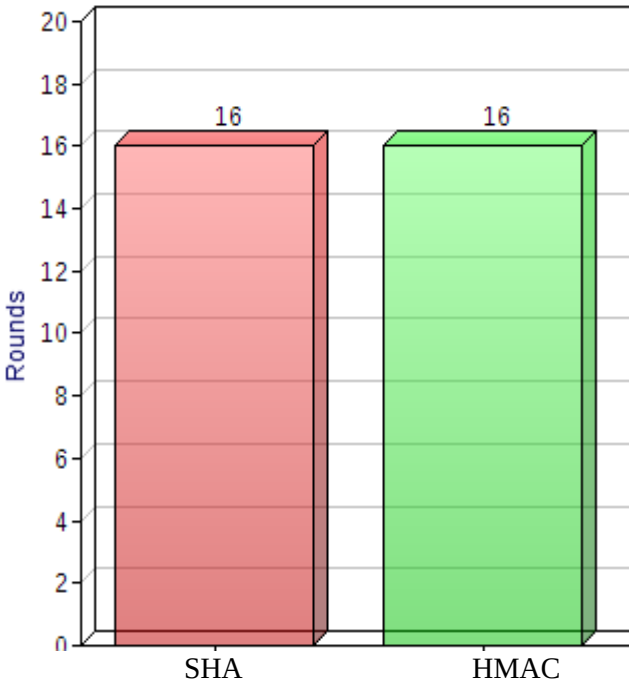

**Fig 5.3: - CPU Cycle of Algorithm**

**4) CPU Execution Time**

The performance matrices are encryption and decryption time. The encryption time is defined as the time that an encryption algorithm takes to generate a cipher text from plain text and decryption time is defined as the time that an encryption algorithm takes to generate plain text from cipher text.

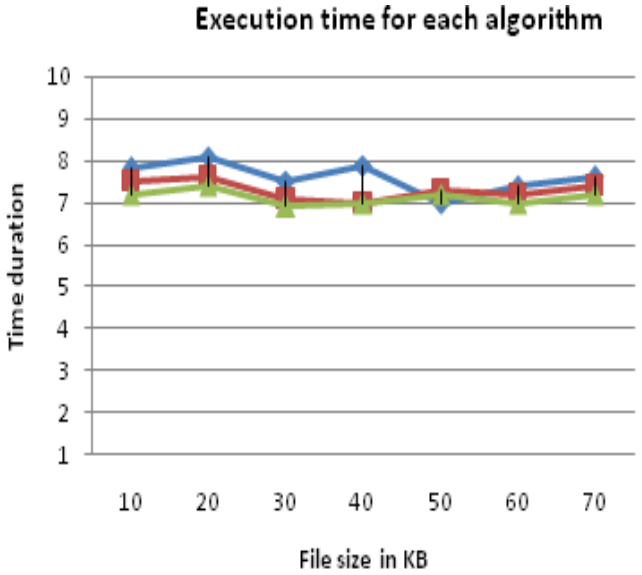


**Fig 5.4: - CPU Execution Time**

**5. Memory Usage**

Algorithm takes to produces a cipher text from a plain text. Encryption time is used to calculate the throughput of an encryption scheme, is calculated as the total plaintext in bytes encrypted divided by the encryption time. Comparisons analyses of the results of the selected different encryption scheme are performed.

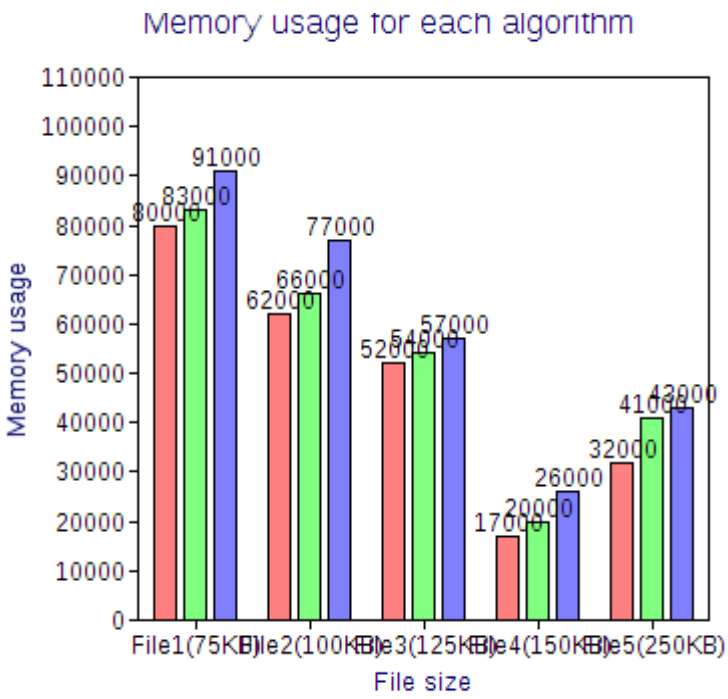


**Fig 5.5: - Memory Usage of Algorithm**

## 6. CONCLUSION

The objective of the paper is to provide a performance analysis between symmetric key cryptography algorithms: HMAC and SHA. The analysis has been conducted by running several encryption settings to process different sizes of data blocks to evaluate the algorithm's speed for encryption and decryption. Each algorithm is designed and executed in these modes. The variation is provided in data size given by the user. The data is retrieved from various text files to calculate the time consumed by each algorithm to process the retrieved data.

## REFERENCES

[1]. Pancholi, V. R., & Patel, B. P. (2016). Enhancement of cloud computing security with secure data storage using AES. International Journal for Innovative Research in Science and Technology, 2(9), 18-21.

[2] Marwan, M., Kartit, A., & Ouahmane, H. (2018). Security enhancement in healthcare cloud using machine learning. Procedia Computer Science, 127, 388-397.

[3] Balasaraswathi, V. R., & Manikandan, S. (2014, May). Enhanced security for multi-cloud storage using cryptographic data splitting with dynamic approach. In 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies (pp. 1190-1194). IEEE.

[4] Liu, J. K., Liang, K., Susilo, W., Liu, J., & Xiang, Y. (2015). Two-factor data security protection mechanism for cloud storage system. IEEE Transactions on Computers, 65(6), 1992-2004.

[5] Sajay, K. R., Babu, S. S., & Vijayalakshmi, Y. (2019). Enhancing the security of cloud data using hybrid encryption algorithm. Journal of Ambient Intelligence and Humanized Computing, 1-10.

[6] Hodowu, D. K. M., Korda, D. R., & Ansong, E. D. (2020). An enhancement of data security in cloud computing with an implementation of a two-level cryptographic technique, using AES and ECC algorithm. Int. J. Eng. Res. Technol, 9, 639-650.

[7] Sana, M. U., Li, Z., Javaid, F., Liaqat, H. B., & Ali, M. U. (2021). Enhanced security in cloud computing using neural network and encryption. IEEE Access, 9, 145785-145799.

[8] Chhabra, S., & Kumar Singh, A. (2020). Security enhancement in cloud environment using secure secret key sharing. Journal of Communications Software and Systems, 16(4), 296-307.

[9] Abbas, M. S., Mahdi, S. S., & Hussien, S. A. (2020, April). Security improvement of cloud data using hybrid cryptography and steganography. In 2020 international conference on computer science and software engineering (CSASE) (pp. 123-127). IEEE.

[10] Mary, B. F., & Amalarethinam, D. G. (2017, February). Data security enhancement in public cloud storage using data obfuscation and steganography. In 2017 world congress on computing and communication technologies (WCCCT) (pp. 181-184). IEEE.

[11] Kumar, P., & Kumar Bhatt, A. (2020). Enhancing multi-tenancy security in the cloud computing using hybrid ECC-based data encryption approach. IET Communications, 14(18), 3212-3222.

[12] Orobosade, A., Favour-Bethy, T. A., Kayode, A. B., & Gabriel, A. J. (2020). Cloud application security using hybrid encryption. Communications on Applied Electronics, 7(33), 25-31.

[13] Thabit, F., Alhomdy, S., Al-Ahdal, A. H., & Jagtap, S. (2021). A new lightweight cryptographic algorithm for enhancing data security in cloud computing. Global Transitions Proceedings, 2(1), 91-99.

[14] Ramachandra, M. N., Srinivasa Rao, M., Lai, W. C., Parameshachari, B. D., Ananda Babu, J., & Hemalatha, K. L. (2022). An efficient and secure big data storage in cloud environment by using triple data encryption standard. Big Data and Cognitive Computing, 6(4), 101.

[15] Shabbir, M., Shabbir, A., Iwendi, C., Javed, A. R., Rizwan, M., Herencsar, N., & Lin, J. C. W. (2021). Enhancing security of health information using modular encryption standard in mobile cloud computing. IEEE Access, 9, 8820-8834.

[16] Gupta, I., Singh, A. K., Lee, C. N., & Buyya, R. (2022). Secure data storage and sharing techniques for data protection in cloud environments: A systematic review, analysis, and future directions. IEEE Access, 10, 71247-71277.

[17] Sumathi, M., & Sangeetha, S. (2018, December). Enhanced elliptic curve cryptographic technique for protecting sensitive attributes in cloud storage. In 2018 IEEE international conference on computational intelligence and computing research (ICCIC) (pp. 1-5). IEEE.

[18] Rani, P. K., Sathiya, S., Sureshkumar, S., & Kumar, B. A. (2022, May). Enhancing Cloud Security with Hybrid Encryption. In 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC) (pp. 1445-1450). IEEE.