Proceedings of the International Conference on Artificial Intelligence and Smart Systems (ICAIS-2021)
IEEE Xplore Part Number: CFP21OAB-ART; ISBN: 978-1-7281-9537-7

Client-side Cryptography Based Security for Cloud Computing System

Md. Abu Musa and Md. Ashiq Mahmood
Institute of Information and Communication Technology
Khulna University of Engineering & Technology (KUET)
Khulna, Bangladesh
Email: amsohag007@gmail.com, ashiqmahmoodbipu@gmail.com

2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS) | 978-1-7281-9537-7/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICAIS50930.2021.9395890

***Abstract*-Cryptography indicates to techniques of securing information and communication derived from mathematical perception to convert messages in ways that are tough to interpret. Cryptography is firmly associated with the department of cryptology along with cryptanalysis. It consists of techniques such as blending words with images, microdots, and alternative ways to mask data during storage or else transit. However, in the modern era, cryptography is repeatedly related to cloud computing. But, moving data into a cloud is a huge modification and has real involvement that makes users lapse before one can sign up for the desired service which can cause unwanted instruction on sensitive information and data lost. For the security of cloud data, a symmetric algorithm had been introduced by previous research work which used simple algorithms and had performance issues. In this research paper we have introduced and enforced symmetric key encryption that would encrypt a file locally at the client-side prior to uploading to the cloud and the file would decrypt after downloading on the client-side using key generated during encryption. This algorithm also uses a different algorithm to calculate the key value. As a result, our algorithm offers better security and better performance for large files. This way we can add an extra layer of security which would restrain unwanted attacks on intimated information as well as lack of standardization.**

Keywords— Cryptography; Cloud Computing; Encryption; Decryption; Standardization; Cloud Storage.

I. INTRODUCTION

Humans always wanted to manage sensitive information securely. Trespassing into confidential information could drive into a huge dispute. Thus, human has been using cryptography from ancient times concerning to have their information secure. So simply, cryptography is all about hiding and protecting the confidential data from third parties. A sign of practicing cryptography has been found in the earliest civilizations. Historian claims that cryptography was first used in Egypt in around 1900 BC [1]. In Greek, master tattooed messages on slave's shaved head which would conceal beneath the regrown hair. Fast-forwarding around 100 BC, the Roman military used encryption to send classified information to his army generals which are also known as Caesar cipher [2]. Around the mid-1400s, a key-encryption was designed by Vigenere which was probably the early key encryption. In his technique, the key used for encryption was imitated numerous times traversing the entire message, and by summing the message character along with the key character modulo 26 ciphertexts were formed [3]. Yet it has been passed down for hundreds of years to disguise classified information, the methodical exercise of cryptology in the act of science just evoked around the last century. At the early stage of the 19th century when many things turn into electric powered, an electro-mechanical appliance was designed by Hebern which was later known as the Hebern rotor machine. It operates a single rotor, which is used to embed the secret key on a rotating disc. The key conceals on a trade table and when a key is pressed using the keyboard it produces the output of ciphertext [4]. After the Second World War, cryptography engages commercial applications post-war, with businesses vexing to secure their information from competitors. As a result, some liabilities have been introduced by the digital revolution of the last century. These liabilities repeatedly led to huge data damage and data heist. So cryptography is only a competent approach to clarify these liabilities. The technological revolution has introduced to the term "cloud computing' which was first arose in early 1996 in a Compaq private record. Cloud computing refers to the continuous sharing of different storing services employing the Internet [5]. Day by day cloud computing is getting trendy choice for developers as well as businesses enterprise for various reasons. We frequently use cloud services to store our important data so that we can get accessed every time we need them. Cloud storages provide an enormous amount of flexibility and make it accessible whenever and wherever we demand the content. Backing up sensitive data to keep them safe is the most crucial step of cloud storage services.

Google Drive is one of the well-known cloud services, also like all cloud services, its main purpose is to cut some payload off user hard drive. Google offers a handful of layers of encryption in order to protect user's data at rest in Google Cloud products. Before storing data is divided into chunks and every chunk is encrypted with the help of an exclusive data encryption key. An individual data chunk carries a unique identifier. This way access is prevented without authorization, aid both data security as well as privacy. Google offers data encryption prior to it being recorded to disk. Google Drive uses TLS (Transport Layer Security) standard to encrypt data afore it leaves the local devices. After receiving the data, Google unencrypted it and re-encrypted utilizing 256-bit AES (Advanced Encryption Standard) [6]. Amazon Storage Service is another cloud

storage service that aims to accomplish large-scale resource sharing smooth for users. Amazon stock users' data redundantly in their storing facilities. Amazon S3 also uses versioning to safeguard

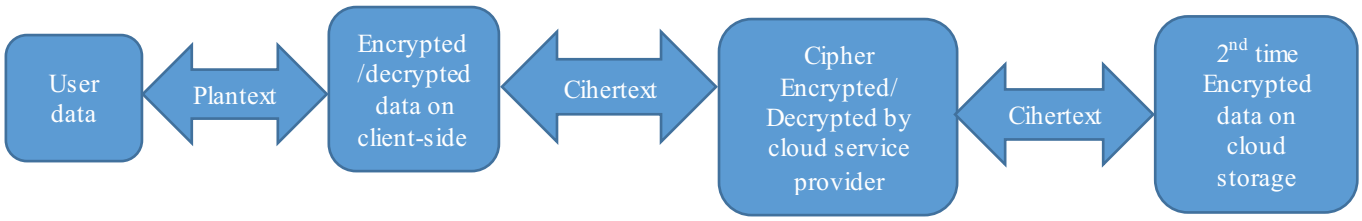


Fig 1. Client-side cryptography process

objects stocked in the Amazon S3 bucket [7]. Lastly, Amazon offers two types of encryption of data. Server-side encryption ensures the safety of data in the data center and client-side encryption ensures security during data transmission from client to amazon data center [8]. Cloud computing has caused us some complications like government intrusion, cyber-attacks, lack of standardization, and outages [9]. Hence, encrypting data locally on the client-side before uploading to cloud storage services is a productive approach to conquer these complications.

In our proposed work, data is encrypted and decrypted two times, firstly-on the client-side by user using his own key, secondly-by cloud service provider. Our main concern is to let user's encrypted data pass through cloud service provider's encryption and decryption process without any errors and safely store it. Thus user's data is safe from any kind of instruction as well as the cloud services providers.

II. RELATED WORKS

Cryptography and encryption are related to each other for the purpose of secure communication from ancient times. Symmetric-key cryptography ensures security for information by using a unique key that is used by both sender and receiver. Symmetric-key cipher discloses to block cipher which refers to a relatively new embodiment of Alberti's polyalphabetic cipher [10]. Cloud computing is evolving rapidly which is the result of technological revelation. Cloud allows on-demand access and accesses from anywhere in the world with the help of the internet. Balachandra Reddy, Ramakrisna, and Dr. Atanu Rakshit review the cloud security issues which have been included in SLA. In their paper, different level of security as well as the complexity of SLA to help the enterprises to understand security policies which are being implemented [11]. Dr. L. Arokhaim and S. Monikandan have proposed an algorithm to mark the privacy as well as the security issues to protect the cloud data. In their paper, the proposed algorithm is used to encrypt user information in the cloud [12]. Niteen, Balu, and Rahul have suggested a framework which is used for client-side AES encryption in cloud computing. In their paper, authentication is used to defend the data of the encryption algorithm which guaranty secured data in the cloud server [13]. A number of crucial ideas are being investigated in the paper of Srinivas, Venkata, and Moiz which can be composed applying distributed computing as well as find out effective ways to get advantages
from rising innovation [14]. By testing Fermats Little Theorem B.M Shereek displayed open key cryptography in the midst of key age for RSA, the time capriciousness can be decreased since sweeping prime numbers are taken. [15]. Sultan Aldossary and William Allen have reviewed the list effects related to cloud data storage and suggested a list of quick fixes to those issues in their paper which specially put a spotlight on cloud computing [16]. Rachna Arora et al represent distant computation on Blowfishes, AES, RSA in their paper and compared these algorithms based on time, memory, and CPU requirements [17]. For client-side encryption, Tania Gaura and Divya Sharma recommended an imitation by combining the AES algorithm with the Diffie-Hellman algorithm on their paper which performs really well [18]. Nesrine and Maryline introduced a client-side deduplication system in their paper for the cloud storage environment. Their proposal ensures confidentiality against unauthorized users by computing per data key to encrypt data as well as by integrating access right in the metadata file [19]. Mahdul, Zahid, and Rifat proposed a convenient way to protect data by combining AES and Hash Algorithm with Initial Vector while storing or uploading data to the cloud [20]. Prerna and Parul Agarwal claim that user can encrypt their data locally at the client-side ahead transmitting it to any cloud storage services so that an additional coating of security can be added in case of data interference or prying during data transit [21]. Dr. Subarna Shakya has introduced an efficient security framework for secure data migration in cloud computing environment by establishing secure SSL and migration ticket with minimum privilege [28]. Pandian, A. Pasumpon, and S. Smys proposed HHAR algorithm along with the cache aware filter to gather historical information's associated with the back-up system and the identify the out of order containers respectively [29]. Every service provider makes use of different encryption technique for securing data which are stored in cloud storage. Some of them offer client-side encryption which can ensure better security. Although Amazon S3, Tresorit, OwnCloud, Viivo different organizations use client-side encryption and keep the encryption keys stored on their own server. As a result, an unease about security as well as privacy goes on [22]. Admitting most of the cloud service providers uphold lofty standards of encryption when data is carried internally but while the data is dragged to and from the service providers

remains a concern. Although the facilities data is housed are shared but cloud data is not shared. By taking the right security measures in place this shouldn't be a problem but there is a chance that having a malicious file uploaded to the same server user's data is on could affect the users too. Sadly, there also exist limitations like lack of standardization, i.e. every service provider may not have end-to-end encryption. Currently, lack of assets/proficiency is the number one cloud challenge but security is the second most important concern when it comes to the cloud [23].

III. CLIENT-SIDE CRYPTOGRAPHY BASED SECURITY FOR CLOUD COMPUTING SYSTEM

Our research work introduced us to a symmetric key algorithm which works on the SaaS layer of cloud model and would help to encrypt sensitive information locally at client-side then transmit the encrypted information to cloud storage to stock it. Our main concern is to keep data safe by using encryption/decryption model and encrypted data such manner so that it is safe from any kind on instruction, even keep data safe from cloud service providers. Authentication process is also handled by the proposed system. In this way, we can ensure the safety of our data during transmission as well as in the data storehouse of the cloud service providers. We supervise not only the encryption process but also the encryption keys by saving the symmetric encryption key locally. To retrieve or use the information we have to download it from the cloud storage. When the download is completed, the information is decrypted by applying the same encryption keys which were stored in local storage. Securely transmitting and optimizing the performance are the major objectives. Though an encrypted link is created during the transmission through SSL (Secure Sockets Layer) but by applying our optimized algorithm we can add a special slab of security. Besides, large numbers of cloud providers don't use encryption when data is conveyed internally among their own datacenters. As a result, it can lead to the risk of intellectual property theft, data loss, government intrusions, and privacy risks [24].

A. Functional workflow

Our symmetric algorithm is developed using java with the Eclipse IDE and then implemented on Amazon S3. Amazon S3 provides constancy and flexibility that allows users to store as well as retrieve data at any moment and from anyplace using the web. At the beginning of the functional workflow, we have created an Amazon S3 bucket and a folder inside the bucket in which encrypted files would be stored. Then we run our program which was prewritten in java and choose a text file that will be encrypted. The program encrypted the text file using our proposed algorithm and produce an encrypted file and an encryption key file which are kept safely on the local storage. When encryption is done, the file which contains ciphertext is uploaded to the Amazon bucket. The uploaded encrypted file has to go through Amazon's server-side encryption once again. Then we download the file from the Amazon bucket to the local storage and decrypt the data file applying the encryption keys which was stored safely on local storage. When decryption is done, we get back our desired plain text which was secure in Amazon's S3 bucket as well as during transmission. Our encrypted data was also not corrupted by Amazon's encryption and decryption process and survive through the process.

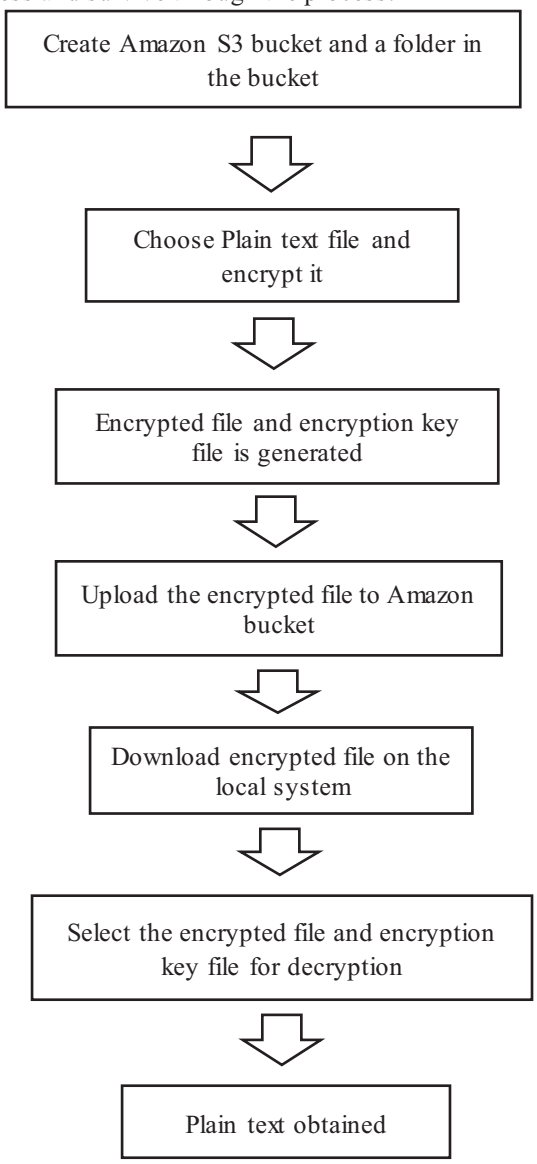


Fig. 2. Functional work flowchart

B. *Encryption algorithm*

Input: plain text
Output: cipher text, encryption key

1: **read** ASCI value for single character
2: **convert** ASCI to Binary
3: **if** (value!=8bit)
4: **then add** preceding 0's
5: **else slice** into 4bit
6: **swap** positons
7: **slice** 4bit into 2bit
8: **swap** positons
9: **then reverse** 8bit value
10: **convert** Binary to ASCI value
11: **add** keyValue
12: **return** encryption key

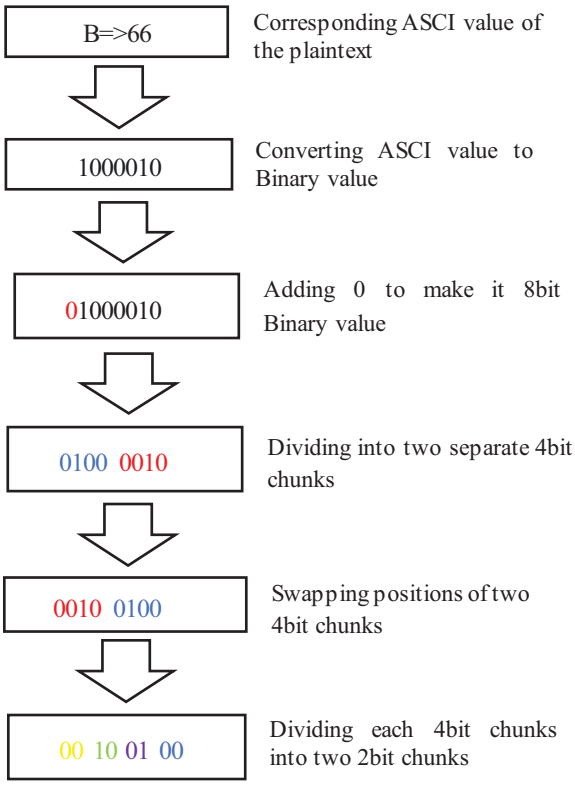
C. *Key value calculation*

Input: Amazon S3 bucketName
Output: keyValue

1: i=0, value=0;
2: c= buscketName.length;
3: **while** (i<=c)
4: **then** value+=ASCI value[i]
5: i++
6: **end while**
7: keyValue=value **mod** c
8: **return** keyValue

Encryption process example:

Let us consider a character in the text file is 'B', consequently, we will get:



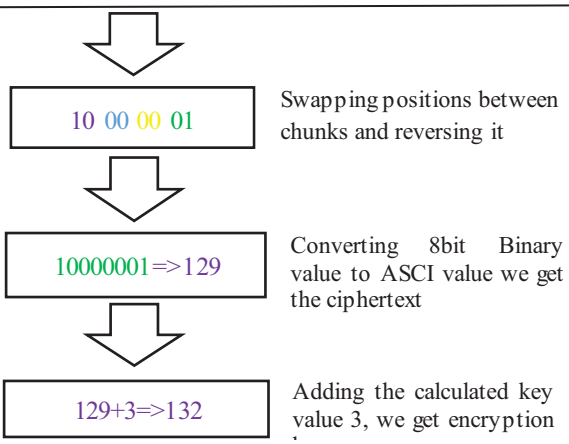


Fig. 3. Encryption process example

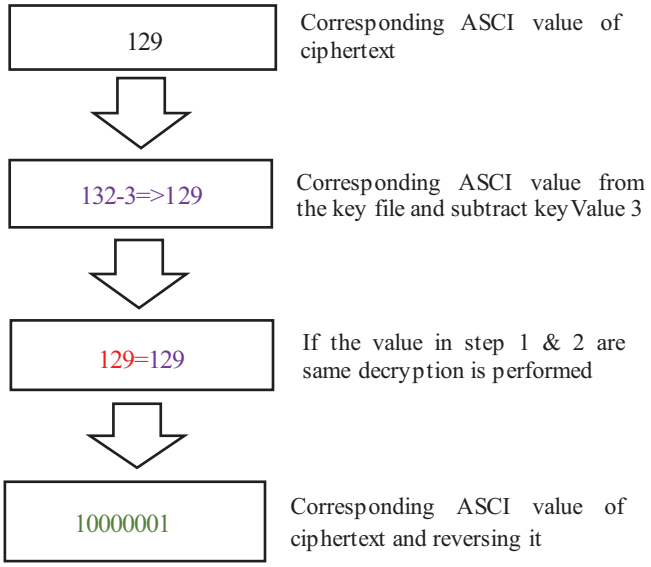
D. *Decryption algorithm*

Input: cipher text, encryption key
Output: plain text

1: c=ASCI value of cipher text
2: e= ASCI value of encryption text
3: **read** c and e
4: **subtract** keyValue from e
5: **if** (value != c)
6: **then stop** decryption
7: **else convert** ASCI to Binary
8: **reverse** binary value
9: **slice** into two 4bit chunks
10: **again slice** into two 2bit chunks
11: **swap** positions between two 2bit chunks
12: **again** swap positions between two 4bit chunks
13: **append** chunks into binary value
14: **convert** Binary into ASCI value
15: **return** plain text

Decryption process example:

Let us consider the first character of the encrypted file is '129' and the first character from the key file is '132', we will get:



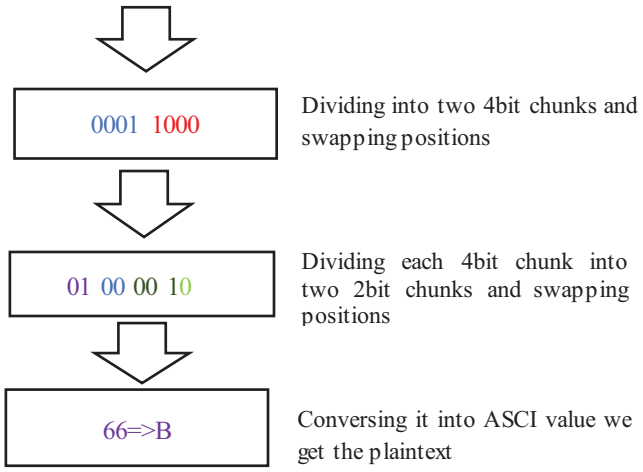


Fig. 4. Decryption process example

E. Amazon S3

In order to demonstrate we have implemented our project on Amazon S3 which provides flexibility that allows users access to the highly scalable, reliable, fast, inexpensive data storage support that Amazon employs to run its endemic global network of web sites [7]. Amazon S3 equips APIs for setting up as well as mange created buckets. By evade, users can build up to 100 buckets in an individual AWS account. If any user demands more buckets, he can increase his account bucket cap to a maximum of 1,000 buckets by agreeing to service limit increase. Users can keep any number of objects in a bucket. When any user builds a bucket, the user state a name and the AWS Region where the user wishes to create the bucket. During the demonstration, we have created an S3 bucket in EU_WEST 1(Ireland) region and upload our encrypted file to the bucket. Amazon offers multiple regions facilities to store data which is very helpful. Amazon S3 services are consist of buckets that can hold as many as objects a user wishes to store. Each bucket has a unique name which is also can be referred by a key (name) [25]. We need to use these keys to retrieve our data from Amazon S3 cloud services.

IV. EXPERIMENTAL ANALYSIS

The security of cloud storage is a crucial issue in data communication. Whenever a business maneuver to the cloud it becomes reliant on the provider's security and performance. Not only security but also performance is required for a business for better handling. An important role is performed by the encryption for better performance. If any encryption algorithm required a lot of resources and time, it can affect the satisfaction of clients. As a result, it is crucial to measure and compare performance. During the study, we have implemented the proposed algorithm in java and optimized it thoroughly for better performance as well as requires comparatively small resources. We have optimized the algorithm using the technique of swapping which doesn't require looping through data file every time. We have tested the performance of the proposed algorithm with 1000, 2000, 3000 characters respectively against the CBS [21] algorithm for which it shows better performance.

A. Encryption algorithm performance comparison

Encryption algorithms portray a vital role in security as well as the performance of the whole process. If the encryption algorithm's performance is poor, clients would not prefer using the system.

TABLE 1: PARAMETERS FOR ENCRYPTION ALGORITHM EVALUATION

Number of character	CBS [21] algorithm (execution time)	Proposed algorithm (execution time)
1000	1125ms	900ms
2000	211ms	1790ms
3000	3208ms	2900ms

TABLE 1 shows the number of characters in the text file and time required for them to encrypt into ciphertext and a key file is generated which contains the encryptions keys. Then the encrypted filed is transmitted to the cloud storage.

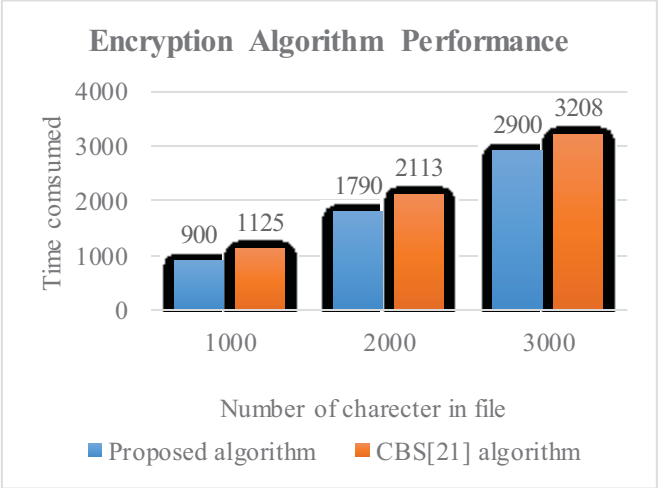


Fig. 5. Performance comparison of encryption algorithms

Fig. 5 demonstrates the time required for the encryption process of the proposed algorithm and the CBS [21] algorithm in milliseconds. It also helps to compare performance between them. In this chart, the proposed algorithm consumes less amount of time during the encryption process and provides better performance. The reason behind the better performance is the new algorithm which encrypts the data more efficiently.

B. Decryption algorithm performance comparison

Decryption algorithms ensure that the encrypted file downloaded from the cloud is not corrupted by comparing the plain text which was decrypted using encryption keys. So, optimize decryption can offer better performance.

TABLE 2: PARAMETERS FOR ENCRYPTION ALGORITHM EVALUATION

Number of character	CBS [21] algorithm (execution time)	Proposed algorithm (execution time)
1000	1250ms	1070ms
2000	2319ms	2009ms
3000	3475ms	3070ms

TABLE 2 displays the number of characters in the downloaded file from the cloud and the required time comparing the key values as well as decrypt into plain text.

Fig 6 demonstrates the time required for the decryption process of the proposed algorithm as well as the CBS [21] algorithm. It also compares the time consumed during the decryption process. We observe that the decryption time consumed by the proposed algorithm is quite impressive. The proposed algorithm gives a better result than the previous one by consuming less time and decrypting the data efficiently.

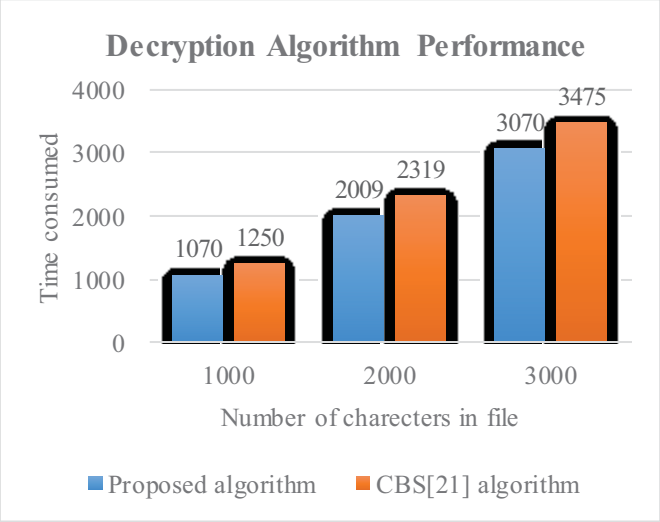


Fig. 6. Performance comparison of decryption algorithm.

V. CONCLUSION

Although an advanced era of conveying and hoarding information has been introduced by cloud storage, many companies are still hesitating to leave out a fair plan for security. Recently the "Cloud Security Spotlight Report" showed that "90 percent of organizations are very or moderately concerned about public cloud security". Encryption algorithms conduct a very important role in information security. The main aspiration of this algorithm is to add an especial coating of security, minimizes data loss or theft in transit, minimizes data intervention, and spying while data is moving internally of the service providers and also solves the dilemma of lack of standardization [26].
Currently, our implemented algorithm is reliable especially for any text files for which encryption, decryption, and storing processes execute perfectly. It cannot handle any other type of file or process it and performance is low for large file. It is symmetric key encryption but in the future, it could be implemented using asymmetric key encryption. It can be also modified thus we can encrypt our message or information from social media while using social network platforms.

VI. REFERENCES

[1] Gary C. Kessler, "An Overview of Cryptography (Updated Version, 3 March 2016)", Embry-Riddle Aeronautical University - Daytona Beach, March 2016.
[2] William August kotas," A Brief History of Cryptography", University of Tennesse- Knoxvile, Spring 5-2000.
[3] Nicholas G. McDonald," Past, Present, and Future Methods of Cryptography and Data Encryption", a research review, University of Utah, May 2020.
[4] K Pommerening, Fachbereich Physik, Mathematik," Rotor Machines", Informatik der Johannes-Gutenberg-University, December 3, 1999—English version November 9, 2013—last change August 25, 2014.
[5] M.D. Dikaiakos et al., "Cloud Computing: Distributed Internet Computing for IT and Scientific Research," IEEE Internet Computing, vol. 13, no. 5, pp. 10–13, 2009.
[6] Darren Quick, Kim-Kwang Raymond Choo,"Google Drive: Forensic analysis of data Remnants", Journal of Network and Computer Applications, Volume 40, Pages 179-193, April 2014.
[7] Mayuer R. Palankar, Adrina lamnitchi, Matei Ripeanu and Simson Garfinkel ," Amazon S3 for Science Grids: A Viable Solution" DADC '08: Proceedings of the 2008 international workshop on Data-aware distributed computing, Pages 55–64, June 2008.
[8] J Yao, S chen and D levy," Truststore: Makng Amazon S3 Trustworthy with Service Composition", *2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*, Melbourne, VIC, pp. 600-605, 2010.
[9] Manish potey, C A Dhote and Deepak H. Sharma," Cloud Computing – Understand Risk, Threat, Vulnerability and Controls: A Survey", International Journal of Computer Applications (0975 – 8887) Volume 67– No.3, April 2013.
[10] Dharitri Talukdar, "Study on Symmetric Key Encryption: An overview", International Journal of Applied Research; 1(10): 543-546, 2015.
[11] Balachandra Reddy, Ramakrisna and Dr. Atanu Rakshit, "Cloud Security Issues," *2009 IEEE International Conference on Services Computing,* Bangalore, pp. 517-520, 2009.
[12] Dr. L. Arockiam, S. Monikandan," Data Security and Privacy in Cloud Storage using Hybrid Symmetric Encryption Algorithm", International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 8, August 2013.
[13] Mr. Niteen, Mr. Balu and Mrs Jayshree Katti, Pimpri Chinchwad, "Framework for Client Side AES Encryption Technique in Cloud Computing", *2015 IEEE International Advance Computing Conference (IACC)*, Banglore, pp. 525-528 , 2015.
[14] K. R. a. A. Q. J. Srinivas, "Cloud Computing Basics, Build. Infrastruct. Cloud Security," vol. 1, pp. 3-22, 2014.
[15] B. Shereek, "Improve Cloud Computing Security Using RSA Encryption," IOSR Journal of Engineering, vol. 4, no. 2, pp. 1-8, 2014.
[16] W. A. S. Aldossary, "Data Security, Privacy, Availability and Integrity in Cloud Computing: Issues and Current Solutions," (IJACSA) International Journal of Advanced Computer Science and Applications, vol. 7, no. 4, pp. 485-498, 2016.
[17] Rachana Arora and Anshu Parashar," Secure User Data in Cloud Computing Using Encryption Algorithms", International Journal of Engineering Research and Applications (IJERA), Vol. 3, Issue 4, pp.1922-1926, Jul-Aug 2013.
[18] D. s. Tania Gaura, "A Secure and Efficient Client-Side Encryption Scheme in Cloud Computing," I.J. Wireless and Microwave Technologies, vol. 1, pp. 23-33, 2016.
[19] Nesrine Kaaniche, Maryline Laurent, "A Secure Client Side Deduplication Scheme in Cloud Storage Environments", *6th International Conference on New Technologies, Mobility and Security (NTMS),* Dubai, pp. 1-7, 2014.
[20] M. M. Islam, M. Z. Hasan and R. A. Shaon, "A Novel Approach for Client Side Encryption in Cloud Computing", *International Conference on Electrical, Computer and Communication Engineering (ECCE)*, Cox'sBazar, Bangladesh, pp. 1-6, 2019.
[21] Prerna, Parul Agarwal*, "Cryptography Based Security for Cloud Computing System". International Journal of Advanced Research in Computer Science; Udaipur Vol. 8, Iss. 5, 2193-2197, (May 2017).
[22] B. M. a. K.-K. R. Choo, "Cloud storage forensics: own Cloud as a case study," Digital Investigation, pp. 287-299, 2013.
[23] S. Zargari and D. Benford, "Cloud Forensics: Concepts, Issues, and Challenges," *2012 Third International Conference on Emerging Intelligent Data and Web Technologies,* Bucharest, pp. 236-243, 2012.

[24] Keiko Hashizume, David G. Rosado, Eduardo Fernandez-Medina, Eduardo B Fernandez," An analysis of security issues for cloud computing", Journal of Internet Services and Applications, 4,5 (2013).
[25] What is SSL, TLS and HTTPS? ; an article available at https://www.websecurity.digicert.com/security-topics/what-is-ssl-tls-https, June 2020.
[26] D Boland," Securing Amazon Web Services (AWS) and Simple Storage Service (Amazon S3) Security", an article regarding Amazon Simple Storage Service security, available at infosecwriters.com, June 2020.
[27] D. R. Bharadwaj, A. Bhattacharya and M. Chakkaravarthy, "Cloud Threat Defense – A Threat Protection and Security Compliance Solution," *2018 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*, Bangalore, India, pp. 95-99, 2018.
[28] Shakya, Subarana. "An efficient security framework for data migration in a cloud computing environment." Journal of Artificial Intelligence 1, no. 01 (2019): 45-53.
[29] Pandian, A. Pasumpon, and S. Smys. "Effective Fragmentation Minimization by Cloud Enabled Back Up Storage." Journal of Ubiquitous Computing and Communication Technologies (UCCT) 2, no. 01 (2020): 1-9.