NHCA: Developing New Hybrid Cryptography Algorithm for Cloud Computing Environment

Ali Abdulridha Taha[1], Dr. Diaa Salama AbdElminaam[2], Prof.Dr. Khalid M Hosny[3]
Information Systems Department, Faculty of computers and informatics, Benha University, Egypt[1,2]
Information Technology Department, Faculty of computers and informatics, Zagazig University, Egypt[3]

***Abstract*—The amount of transmitted data through the internet become larger and larger every day. The need of an encryption algorithm that guarantee transmitting data speedily and in a secure manner become a must. The aim of the research is to encrypt and decrypt data efficiently and effectively protect the transmitted data. This research paper presents a model for encrypting transmitted cloud data. This model uses the following encryption algorithms RSA, Triple DES, RC4, and Krishna to generate a new encryption algorithm that encrypt and decrypt transmitted data. The algorithm will help cloud agencies and users to secure their transmitted data and prevent it from being stolen.**

Keywords—Hybrid cryptography algorithms; symmetric encryption algorithms; asymmetric encryption algorithms

I. Introduction

Nowadays the amount of currently stored data on the internet become large, cloud suffers from securing their own data when it transmitted through the internet. Users only think about how to save their own sensitive data from being stolen and when they use encryption algorithms like RSA encryption algorithm or Triple DES encryption algorithm the time become longer to encrypt or decrypt the transmitted data.

RSA is an encryption algorithm used to encrypt and decrypt data, RSA is asymmetric cryptographic algorithm and this is mean that agencies must use two different keys to encrypt the data, one of them is a public key and other is private key. RSA is a relatively slow encryption algorithm [1].

Triple DES is asymmetric block cipher algorithm created from the Data Encryption Standard (DES) by using it three times to each data block. Triple DES key sizes 168, 112 or 56 bits with three options. First one all three keys are independent, second option key one and key two are independent, and key three = key one, and finally third option all three keys are identical. But Triple DES is slow encryption algorithm [2].

Although of the advantage of high security offered by these encryption algorithms it still takes a long time to encrypt data. There are other encryption algorithms that can perform the same process with a high speed, but it suffers from ability of being easily hacked.

Krishna is an encryption algorithm which uses public random bits key merged with a secret key. This key is shared between sender and receiver to encrypt and decrypt data [3].

The Advanced Encryption Standard (AES) is the inheritor of DES as standard symmetric encryption algorithm. This algorithm was developed to solve the small size key on DES encryption algorithm. AES is faster and stronger than DES [4], [5].

Rivest Cipher 4 (RC4) is a symmetric stream cipher algorithm requiring a secure exchange of a shared key. RC4 encryption process is about 10 times faster than DES [6].

The proposed hybrid cryptography algorithm aims to build an efficient and secure encryption algorithm based on merging the encryption algorithms to make hybrid encryption algorithm that can encrypt and decrypt data efficiently and in a secure manner.

In this paper we present a hybrid cryptography algorithm that efficiently encrypts the transmitted data through the cloud. The hybrid cryptography algorithm presents a variety of different encrypting algorithms that allow the user to choose the encrypting method which is suitable with his own type of data.

The proposed hybrid cryptography algorithm studies the time and throughput for a set of encryption algorithms. More than one combination was made to implement hybrid encryption algorithms. User will have different varieties to encrypt and decrypt his own data with any of the hybrid algorithms based on the time consumed by this algorithm to encrypt the data and the level of security provided by this algorithm. Also, the proposed hybrid cryptography algorithm shows the points of strength and weakness of making hybrid encryption algorithms.

The remainder of the paper is organized into five sections: In section 2, presents an overview for the previous works related to our research. In section 3, the materials and methods of the proposed system is described. In section 4, results and discussions are produced, before drawing conclusions and future work in section 5 and 6.

II. Related Work

Jiehong et al., [7] provided evaluation of three of the common encryption algorithms: AES, Blowfish, and GOST. A comparison has been conducted for those encryption algorithms at different sizes of data blocks. They presented results of comparison among selected cryptography algorithms. The Blowfish shows better performance independently on plain-text size. They conclude that the weak part of Blowfish algorithm is a key expansion process; it could take even more time to expand the key than perform encryption or decryption if plain-text size is small. Also, they mention that GOST algorithm key expansion takes almost same time as for Blowfish, and both encryption and decryption for GOST algorithm costs almost same time. Lastly, they conclude that decryption is the longest operation for AES algorithm.

Bhandari et al., [8] mentioned that the most challenging issue today in cloud servers is to ensure data security and privacy of the users. They presented HE-RSA or hybrid encryption RSA along with Advanced Encryption Standard or AES to ensure efficiency, consistency and trustworthiness in cloud servers. They aimed to use various cryptography concepts during communication along with its application in cloud computing and to enhance the security of cipher text or encrypted data in cloud servers along with minimizing the consumption of time, cost and memory size during encryption and decryption. They observed that the difference between the running time of the original RSA and Improved Algorithm using Hybrid Encryption-RSA and AES is increasing drastically as the exponent size is increasing.

Jakimoski et al. [9] analyzed and evaluated the most important security techniques for data protection that are already accepted from the cloud computing providers. They classified them in four sections according to the security mechanisms that they provide: authentication, confidentiality, access control and authorization. They conclude that if all recommended measures are taken into account providing authentication, confidentiality, access control and authorization, then the cloud computing can be trusted in data protection. They focused on the security issues that should be taken into account in depth in order to have proper data security in the cloud. They recommended important security measures relating to data protection in the cloud that must be taken into account.

Waleed et al., [10] focused on improving the security of the cloud and user's privacy. They emphasized on the security deficiencies and the subsequent repercussions regarding the commonly ignored area of private cloud users' information privacy. They proved that UEC can be a valuable point for cloud database researchers, designers, and the cloud platform vendors. By proving that there is a balance between the duties of the unrestricted cloud administrators and providing only the important identified information based on administrator, they achieved that the UEC ensures security and privacy in cloud computing.

Alotaibi et al., [11] examined the factors contributing to the adoption of SaaS. They drew upon prior research to develop a revised model based on the UTAUT. The proposed model offers a comprehensive explanation for SaaS adoption behavior, by modeling QoS as a primary antecedent of BI, due to its role in online services. Furthermore, education was incorporated into the model as a moderating factor to fit the context of SaaS adoption in developing countries. The revised model was empirically examined using empirical data collected by means of an online questionnaire.

D S Abd Elminaam et al., [12] presents a performance evaluation for six of the most common block Symmetric encryption algorithms the selected algorithms are AES, DES, 3DES, RC6, Blowfish and RC2. The results show that Blowfish has better performance than other common encryption algorithms used, followed by RC6.

Many other researchers who are recognized in literature such as [13]-[17] reviewed a comparative analysis of encryption algorithms like AES, DES and RSA for data communication by using encryption time; memory usages output byte and battery power. They also studied the evaluation of performance of selected symmetric key algorithms. In [18]-[21] authors studied the security threats and maintain in mobile ad hoc networks. They also studied the main attack types and several security techniques that help the MANETs to protect from internal or external attacks and aspects of intrusion detection.

III. Materials and Methods

The proposed hybrid cryptography algorithm developed to secure the data and information which is transmitted through the cloud. The aim of the hybrid cryptography algorithm is to efficiently encrypt and secure the transmitted data. Fig. 1 shows the structure of the implemented hybrid cryptography algorithm.

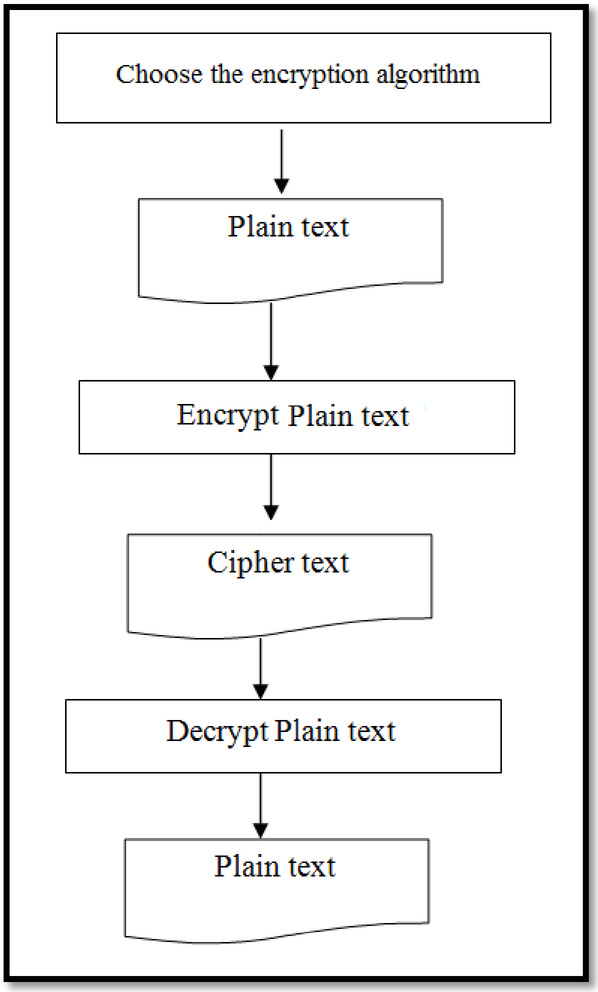


Fig. 1. System structure.

The proposed hybrid cryptography algorithm offer set of encryption algorithms:

1) Hybrid encryption algorithm using Krishna and Triple DES algorithms.
2) Hybrid encryption algorithm using Krishna and AES algorithms.
3) Hybrid encryption algorithm using Krishna and Blowfish algorithms.
4) Hybrid encryption algorithm using RSA and Triple DES algorithms.
5) Hybrid encryption algorithm Krishna and AES and Blowfish algorithms.

The aim of the proposed hybrid cryptography algorithm is to determine the most fast and secure encryption algorithm of the previous presented encryption algorithms. It also allows the user to choose the encryption algorithm which is more suitable for the type of his own data. Also, the hybrid cryptography algorithm implements several encryption algorithms and allow the user to use them to encrypt his own data these algorithms are:

1) AES
2) Blowfish

In Krishna algorithm, several steps have to be done to encrypt and decrypt the text. To encrypt the text the algorithm will work as follow:

1) In the Text, each letter is treated as a digit in base 26
2) A block of n letters is considered as a vector of n dimensions.
3) Multiply the vector by a n × n matrix (Key).
4) Get the modulo 26 of the resulted matrix.

In order to decrypt, we turn the cipher text back into a vector, then simply multiply by the inverse matrix of the key matrix.

Fig. 2 shows how the system encrypts the data using the Krishna encryption algorithm and Triple DES encryption algorithm.

Fig. 3 shows how the system encrypts the data using the Krishna encryption algorithm and AES encryption algorithm.

Fig. 4 shows how the system encrypts the data using the Krishna encryption algorithm and Blowfish encryption algorithm.

Fig. 5 shows how the system encrypts the data using the RSA encryption algorithm and Triple DES encryption algorithm.

Fig. 6 shows how the system encrypts the data using the Krishna encryption algorithm, AES encryption algorithm and Blowfish encryption algorithm. In this algorithm the system divides the plain text in to two parts. The first part was encrypted and decrypted using Krishna encryption algorithm and AES encryption algorithm. The second part was encrypted and decrypted using Krishna encryption algorithm and Blowfish encryption algorithm.

The proposed hybrid cryptography algorithm calculates the time and throughput for each of the pervious encryption algorithms.111

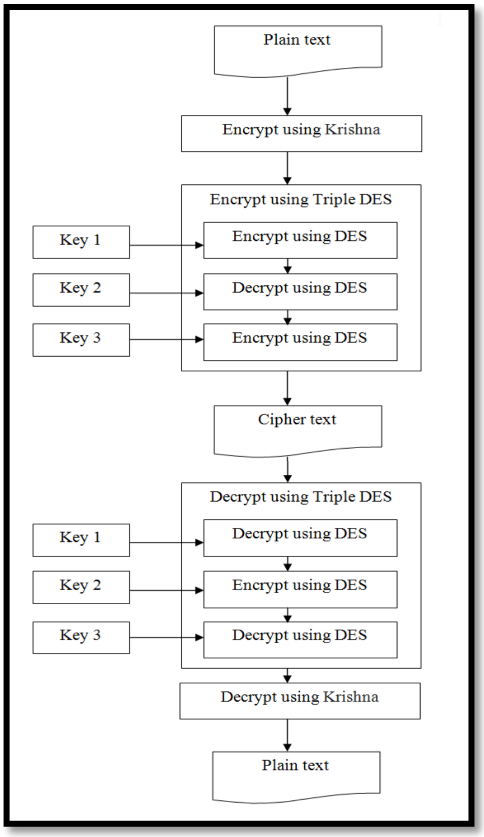


Fig. 2. Hybrid encryption algorithm using Krishna and Triple DES algorithms.

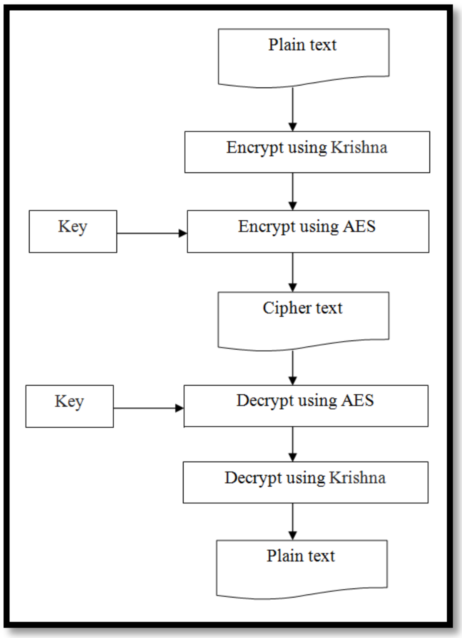


Fig. 3. Hybrid encryption algorithm using Krishna and AES algorithms.

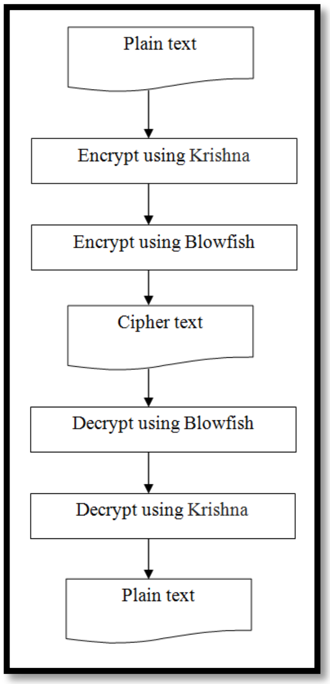


Fig. 4. Hybrid encryption algorithm using Krishna and Blowfish algorithms.

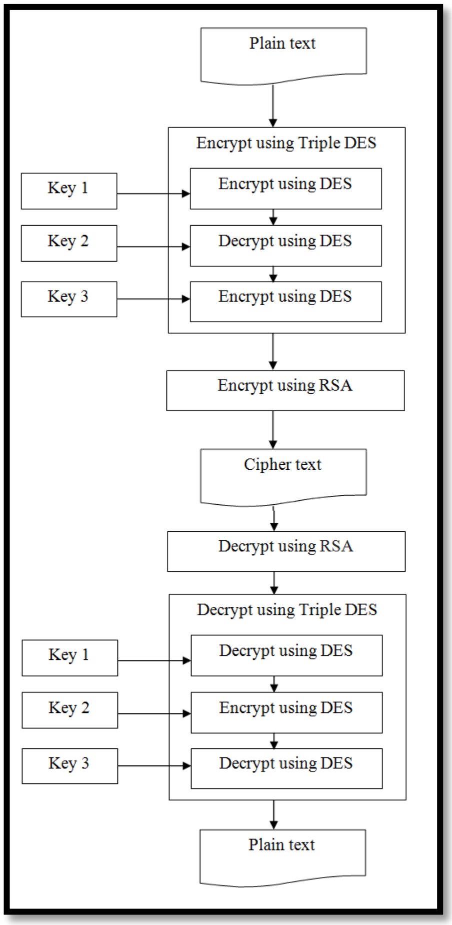


Fig. 5. Hybrid encryption algorithm using RSA and Triple DES algorithms.

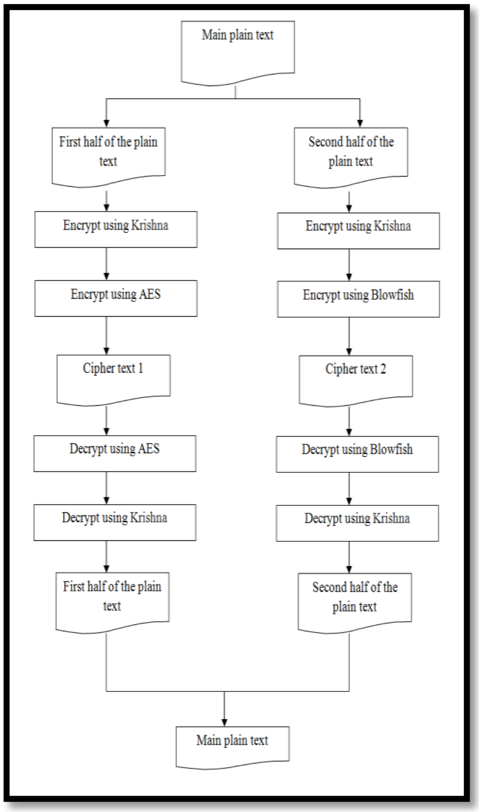


Fig. 6. Hybrid encryption algorithm Krishna, AES and Blowfish algorithms.

IV. Results and Discussion

Extensive experiments are preformed to study the efficiency of the implemented hybrid cryptography algorithm encrypt and decrypt data in a minimum time. The hybrid cryptography algorithm was tested on different file sizes. The system runs 50 times for each file size to calculate the average time for each encryption algorithm. Table 1 show the time consumed in encrypting data using the eight algorithms.

As shown in Table 1, the proposed hybrid cryptography algorithm allows the user to encrypt his data with hybrid encryption algorithms that uses two strong encryption algorithms without taking large time in encrypting data. For example, encrypting a file with 1 MB using Blowfish algorithm takes 11.99 seconds and at the same time encrypting the same file using Blowfish and Krishna takes 9,28 seconds. This is because at the second algorithm the plain text was encrypted using Krishna algorithm at the first then the cipher text was re-encrypted using Blowfish algorithm and because that Krishna minimize the size of the plain text in on word the size of the cipher text sent to Blowfish will be small and the algorithm will take minimum time to encrypt it. Table 2 and Fig. 8 illustrate the throughput of the algorithms. Throughput was calculated by dividing the size of the file in bytes over the consumed time in seconds. The throughput was calculated using the following equation:

$$\text{Throughput} = \frac{\text{File size}}{\text{Encryption time}} \qquad (1)$$

TABLE I. THE TIME CONSUMED IN ENCRYPTING DATA FOR EACH ALGORITHM

File size / Algorithm	250 KB	500 KB	750 KB	1 MB	1.25 MB	1.5 MB	1.75 MB	2 MB	2.25 MB	2. 5 MB	2.75 MB	3 MB	3.25 MB	3. 5 MB	3.75 MB	4 MB
Triple DES & Krishna	0.7	0.8	2.2	3.1	4.3	5.7	6.4	7.1	9	10.9	12.3	13.4	14.7	15.3	16.6	18.1
AES & Krishna	.6	1.1	1.7	3.4	4.2	7.1	8.5	12.3	16.8	18.9	20.5	23.7	25.2	29.6	33.1	36.3
Blowfish & Krishna	2.3	4.9	6.4	9.2	12.3	13.6	15.5	18.6	20.1	23.2	27.8	29.6	32.9	36.7	40.7	42.5
Triple DES & RSA	2.7	4.5	6.8	8.5	11.3	15.5	19.4	20.8	23.7	25.8	27.4	31.6	33.4	34.4	38.3	42.1
AES & Blowfish & Krishna	2.8	4.0	4.3	6.9	8.9	10.4	13.1	15.9	17	19.4	22.1	24.9	27.6	30.3	32.8	34.7
AES	0.6	1.2	1.8	2.4	3.3	4.2	5.8	6.9	7.3	8.7	9.9	11.5	13.6	15.9	18.1	20.6
Blowfish	2.4	5.1	8.4	11.9	14.2	17.9	21.4	24.3	26.2	29.5	31.8	35.7	39.1	42.4	46.2	48.9

TABLE II. THE THROUGHPUT FOR EACH ALGORITHM

File size / Algorithm	250 kb	500 kb	750 kb	1MB
Triple DES & Krishna	328947.36	595238.09	337837.83	327156.54
AES & Krishna	412903.23	435744.68	432919.95	300193.52
Blowfish & Krishna	110678.77	103038.84	11864.66	112968.75
Triple DES & RSA	90252.70	109649.12	112275.44	120046.89
AES & Blowfish & Krishna	89074.46	126482.21	175824.17	150484.50

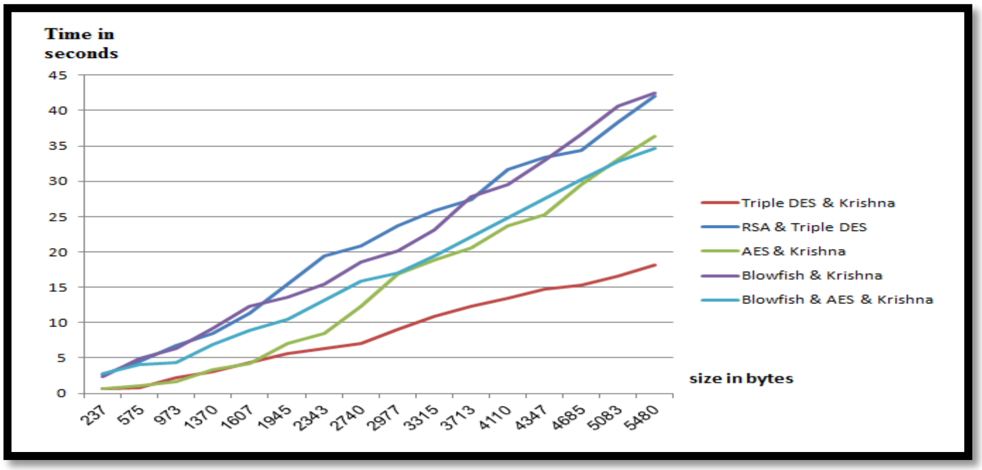


Fig. 7. The time consumed in encrypting data for each algorithm.

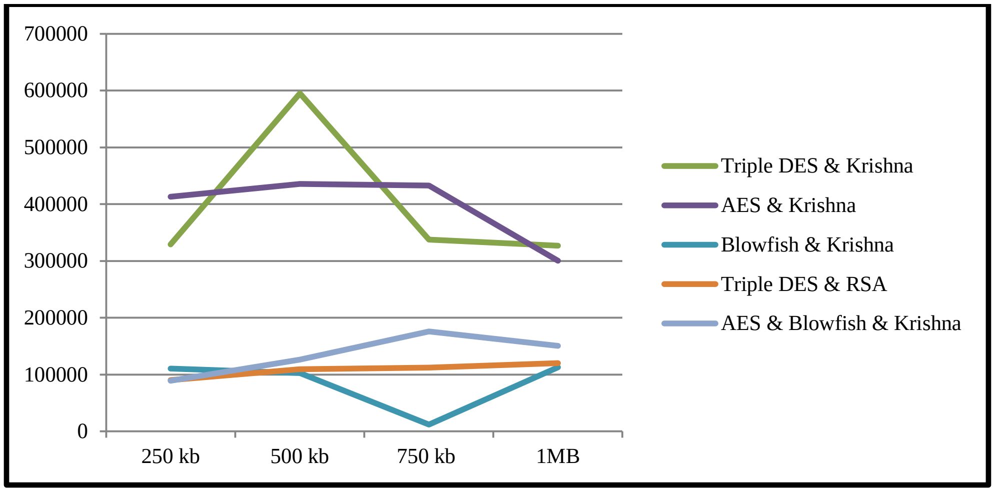


Fig. 8. The throughput for each algorithm.

The result show that the hybrid encryption algorithm (Triple DES & Krishna) takes the lowest time to encrypt the data comparing with other algorithms as shown in Table 1 and Fig. 7.

Jiehong et al., [1] provided evaluation of three of the common encryption algorithms: AES, Blowfish, and GOST. A comparison has been conducted for those encryption algorithms at different sizes of data blocks. Also, Bhandari [2] presented HE-RSA or hybrid encryption RSA along with Advanced Encryption Standard or AES to ensure efficiency, consistency and trustworthiness in cloud servers. His goal was to use various cryptography concepts during communication along with its application in cloud computing and to enhance the security of cipher text or encrypted data in cloud servers along with minimizing the consumption of time, cost and memory size during encryption and decryption. Our implemented hybrid cryptography algorithm (Triple DES & Krishna) shows a better result than the results presented in [1], [2].

To use the hybrid cryptography algorithm the user, choose the algorithm he wants to encrypt his own data with as shown in Fig. 9. The user will choose the algorithm and press “use” to begin the encryption process.

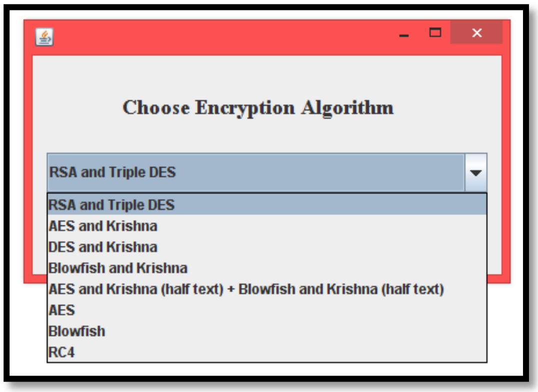
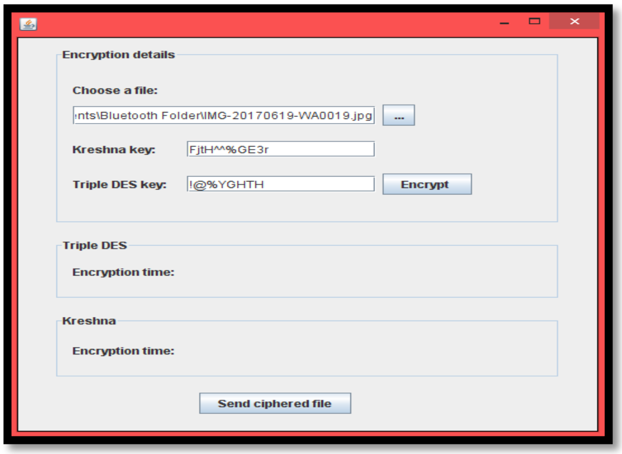


Fig. 9. Selecting the encryption algorithm.

The user determines the place of the file as shown in Fig. 10 then he will enter the keys of the algorithm and press encrypt. As shown in Fig. 11 the hybrid cryptography algorithm retrieves the time consumed in encrypting the data on the selected file. The same steps will be made in decrypting process but the selected file will be the encrypted file resulted from the encryption process; Fig. 12 and 13 illustrate the decryption process.



Encryption details

Choose a file: ·nts\Bluetooth Folder\IMG-20170619-WA0019.jpg

Kreshna key: FjtH^^%GE3r

Triple DES key: !@%YGHTH

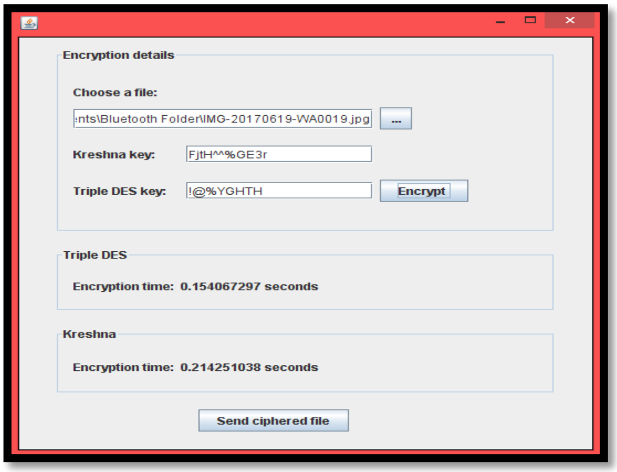
Triple DES

Encryption time:

Kreshna

Encryption time:

Fig. 10. Selecting the file to be encrypted.



Encryption details

Choose a file: ·nts\Bluetooth Folder\IMG-20170619-WA0019.jpg

Kreshna key: FjtH^^%GE3r

Triple DES key: !@%YGHTH

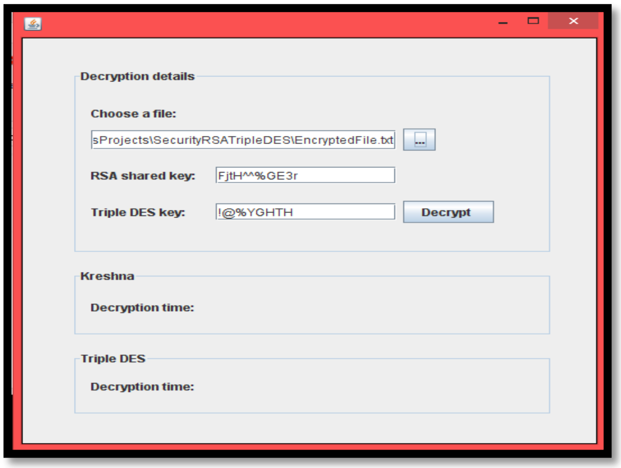
Triple DES

Encryption time: 0.154067297 seconds

Kreshna

Encryption time: 0.214251038 seconds

Fig. 11. Retrieves the time consumed in encrypting the data.



Decryption details

Choose a file:

sProjects\SecurityRSATripleDES\EncryptedFile.txt

RSA shared key: FjtH^^%GE3r

Triple DES key: !@%YGHTH

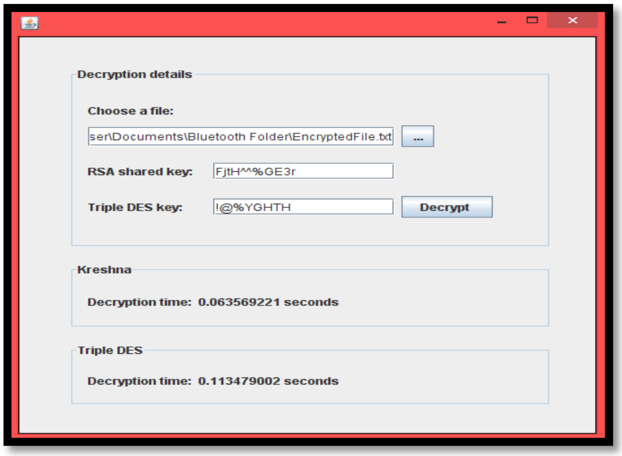
Kreshna

Decryption time:

Triple DES

Decryption time:

Fig. 12. Selecting the file to be decrypted.



Decryption details

Choose a file:

ser\Documents\Bluetooth Folder\EncryptedFile.txt

RSA shared key: FjtH^^%GE3r

Triple DES key: !@%YGHTH

Kreshna

Decryption time: 0.063569221 seconds

Triple DES

Decryption time: 0.113479002 seconds

Fig. 13. Retrieves the time consumed in decrypting the data.

V. Conclusion and Future Work

In this paper, we present a hybrid cryptography algorithm that efficiently encrypts the transmitted data through the cloud. Firstly our hybrid cryptography algorithm presents a variety of different encrypting algorithms that allow the user to choose the encrypting method which is suitable with his own type of data. Secondly, the hybrid cryptography algorithm improves the performance of the encryption algorithms since it encrypts the data in a minimum time and in a secure way. Thirdly, the proposed hybrid cryptography algorithm allows the users to send and receive data in a secure way without facing the problem of attacking data. Fourthly, the encryption times for encrypting a file with a size 1 MB using difference hybrid algorithms come in the following ascending order: using Triple DES & Krishna hybrid algorithm takes 3.13 seconds, using AES & Krishna hybrid algorithm takes 3.493 seconds, using Triple DES & RSA hybrid algorithm takes 8.53 seconds and using Blowfish and Krishna hybrid algorithm takes 9,28 seconds. Fifthly, the proposed hybrid cryptography algorithm prove that merging the three encrypting algorithms AES, Blowfish and Krishna to have AES & Blowfish & Krishna hybrid algorithm increases the security level and also saves the encryption time, so we can encrypt a file with a size 1 MB using AES & Blowfish & Krishna hybrid algorithm in 6.968 seconds. Sixthly, after calculating the throughput for all hybrid algorithms on a file with a size 1 MB, the Triple DES & Krishna hybrid algorithm shows the largest value for the throughput and on the other side the Blowfish & Krishna hybrid algorithm shows the lowest value for the throughput. Lastly, the proposed hybrid cryptography algorithm proves that using hybrid algorithms increase the level of securing the encrypted transmitted data and also minimize the time taken to encrypt it. As a future work, new hybrid algorithms will be constructed from different existence algorithms to improve the encryption process and compare it with the results of our current work.

References

[1]. Pereira, G. C., Alves, R. C., Silva, F. L. D., Azevedo, R. M., Albertini, B. C., & Margi, C. B. (2017). Performance Evaluation of Cryptographic Algorithms over IoT Platforms and Operating Systems. Security and Communication Networks, 2017.

[2]. Sharma, S., & Gupta, Y. (2017). Study on Cryptography and Techniques. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 2(1).

[3]. Krishna, A. V. N., and Babu A. Vinaya. "A modified hill cipher algorithm for encryption of data in data transmission." Computer Sciences and Telecommunications 3 (2007): 78-83.

[4]. Padgette, John. "Guide to bluetooth security." NIST Special Publication 800 (2017): 121.

[5]. Bogdanov, Andrey, et al. "ALE: AES-based lightweight authenticated encryption." International Workshop on Fast Software Encryption. Springer, Berlin, Heidelberg, 2013.

[6]. Kannammal, A., and S. Subha Rani. "Two level security for medical images using watermarking /encryption algorithm" International Journal of Imaging Systems and Technology 24.1 (2014): 111-120.

[7]. Wu, Jiehong, Ilia Detchenkov, and Yang Cao. "A study on the power consumption of using cryptography algorithms in mobile devices." Software Engineering and Service Science (ICSESS), 2016 7th IEEE International Conference on. IEEE, 2016.

[8]. Bhandari, Akshita, Ashutosh Gupta, and Debasis Das. "Secure algorithm for cloud computing and its applications." Cloud System and Big Data Engineering (Confluence), 2016 6th International Conference. IEEE, 2016.

[9]. Jakimoski, Kire, "Security Techniques for Data Protection in Cloud Computing." International Journal of Grid and Distributed Computing 9.1 (2016): 49-56.

[10]. Waleed, AL-Museelem, and Li Chunlin. "User Privacy and Security in Cloud Computing." International Journal of Security and Its Applications 10.2 (2016): 341-352.

[11]. Alotaibi, Mutlaq B. "Antecedents of software-as-a-service (SaaS) adoption: a structural equation model." International Journal of Advanced Computer Research 6.25 (2016): 114.

[12]. Diaa Salama Abdul.Elminaam, Hatem Mohamed Abdul kader, Mohiy Mohamed Hadhoud." Evaluating the Performance of Symmetric Encryption Algorithms “. International Journal of Network Security (IJNS), VOL.10 No.3, pp: 216- 222, May 2010.

[13]. Diaa Salama Abdul.Elminaam, Hatem Mohamed Abdul kader, Mohiy Mohamed Hadhoud." “Evaluating the Effects of Cryptography Algorithms on Power Consumption for Different Data Types “. International Journal of Network Security (IJNS), VOL.11 No.2, pp: 91-100, Sep 2010.

[14]. Diaa Salama Abdul.Elminaam, Hatem Mohamed Abdul kader, Mohie Mohamed Hadhoud," Studying the Effects of Most Common Encryption Algorithms" , International Arab Journal of e-Technology(IAJeT),VOL.2,No.1,PP:1-10, January 2011

[15]. Diaa Salama Abdul.Elminaam, Hatem Mohamed Abdul kader, Mohie Mohamed Hadhoud,M S Elsayed , “Developing and Evaluation of New Hybrid Encryption Algorithm". International Journal Of Computers & Technology (Ijct, Vol.13 No.1, PP. 4038-4052, March 2014.

[16]. Diaa Salama Abdul.Elminaam, Hatem Mohamed Abdul kader, Mohie Mohamed Hadhoud,M S Elsayed , “Performance Evaluation of New Hybrid Encryption Algorithms to be used for Mobile Cloud Computing". International Journal of Technology Enhancements and Emerging Engineering Research (IJTEEE), Volume 2 - Issue 4, PP. 63-71, April 2014

[17]. Diaa Salama Abdu.Ellminaam, Hatem Mohamed Abdul kader, Mohie Mohammed ,Performance Evaluation of Symmetric Encryption Algorithms, IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.12 pp: 280- 286, December 2008

[18]. Gurjeet Singh, "Security Threats and Maintenance in Mobile ad hoc networks", IJECT, Vol.2, Issue 3, September 2011.

[19]. Wenjia Li and Anupam Joshi, "Security Issues in Mobile Ad hoc Networks - (A Survey)", The 17 th White House Papers Graduate Research In Informatics at Sussex, (2004), pp.1-23.

[20]. Er.Deepinder Singh Wadhwa, Er.Tripatjot Singh Panag, "Performance Comparison of Single and Multipath Routing Protocols in Ad hoc Networks", Int. J. Comp. Tech. Appl., (IJCTA), Vol2, Sept-Oct 2011, PP.1486-1496.

[21]. Subash Chandra Mandhata, Dr.Surya Narayan Patro, "A Counter Measure to Black hole Attack on AODV Based Mobile Ad hoc Networks", International Journal of Computer & Communication Technology(IJCCT), Vol.2, Issue 6, 2011