# Double encryption using Private Key and Content Based Public Key

**Bharatula Lakshmi Sai Prasanna*, Palaparthi Meghana and Mr. Isunuri Bala Venkateswarlu**

*Abstract*— **Fast evolution of internet resembles the growth of files over the worldwide network, which also increases many challenges to the file security. Several predefined hacking tools became handy for hackers to target file security over networks and hence it is recommended to use multiple-encryption instead of traditional single level encryption for file servers. As the number of levels increases, time complexity and number of keys required for encryption increases. Hence, our proposed method uses only symmetric cryptography with Blowfish at first level using content-based public (CBP) key extracted from file contents and AES at second level using user's private key to encrypt the file. Here, Blowfish optimizes execution time and AES increases strength of proposed method. This method uses 16-byte keys to increase time for cracking shown in cryptanalysis. The results section proves that our method exhibits improved results than state-of-art methods in terms of execution time and cryptanalysis.**

Keywords —Double encryption, Content Based Public Key, Multiple encryption.

## I. Introduction

Client-server technologies [1] have been an important factor in the expansion of information technology across an increasing range of application business processes which focused on file sharing. From the reports of International Data Corporation (IDC) [2] and Internet World Stats (IWS) [3], it can be concluded that worldwide enterprise storage market grew 14% and reached $11.8 billion in third quarter of 2017. At the same time, worldwide internet user penetration rate was observed as 51.7%. This shows the potential need for data/file security over internet/network. Contemporary research of data security focuses on a wide variety of application including data stored in mobile phones [4] and private social network data over cloud [5]. Hence, a strong encryption system is required to secure data stored on cloud and/or server.

Now a day, there are two most common ways of encryption: symmetric cryptography and asymmetric cryptography [6]. Symmetric cryptography is also known as shared secret/private key encryption because the same key is used on both ends for both encryption and decryption. This form of encryption uses a private key to scramble the data into unintelligible gibberish. The person on the other end needs the shared private key to unlock the data, illustrated in *Table 1.*

**Notations Used:**
*PT = Plain Text*
*ET = Encrypted Text*
*$K_{pr}$ = Shared Secret/ Private Key used by both Sender and Receiver*
*$K_{rpu}$ = Receivers Public Key used by both Sender*
*$K_{rpr}$ = Receivers Private Key used by both Receiver*

| | Application | Algorithm Used | Input | Process | Output |
|---|---|---|---|---|---|
| Symmetric | Sender and Receiver uses same Private Key ($K_{pr}$) | | | | |
| | **Sender** | Encryption (EA) | PT | ET=$EA_{Kpr}$(PT) | ET |
| | **Receiver** | Decryption (DA) | ET | PT= $DA_{Kpr}$(ET) | PT |
| Asymmetric | Sender uses Receivers Public Key ($K_{rpu}$)<br>Receiver uses Receivers Private Key ($K_{rpr}$) | | | | |
| | **Sender** | Encryption (EA) | PT | ET=$EA_{Krpu}$(PT) | ET |
| | **Receiver** | Decryption (DA) | ET | PT= $DA_{Krpr}$(ET) | PT |

*Table 1: Encryption and Decryption process*

On the other hand, Asymmetric cryptography is known as public key encryption which uses encryption that splits the key into two smaller keys. One of the keys is made public and one is kept private. The sender encrypts data with the recipient's public key and the recipient can then decrypt it with their private key, shown in *Table 1*. Even though there are several symmetric and asymmetric cryptography techniques, each technique has both pros and cons [7]. In general, Attacks on cryptographic systems can be mainly classified into Inception, Modification, Fabrication and Interruption. Influence of crackers is also increasing day by day by the usage of various types of attacks which can be evidenced from history. In 1998, the DES message was cracked in 39 days for a contest at RSA conference [8] and RSA-140 was solved in nine weeks. Similarly, from [7], [8] and [9], we can say that none of these encryption algorithms is strong enough for all types of attacks. Each algorithm is vulnerable to at least one type of attack like DES to brute force, AES to side channel attack, RSA to factoring public key etc.

At the same time, analytical study of Zoran [9] revealed that asymmetric algorithms are superior in security while symmetric algorithms are computationally effective. Finally, with all these evidence it can be concluded that efficient

***Manuscript received Feb, 2018.***

***Bh. Lakshmi Sai Prasanna**, CSE, St. Ann's College of Engineering & Technology, Chirala, India, (e-mail: prasanna.bhvs@gamil.com).*

***P. Meghana Murali Krishna**, CSE, St. Ann's College of Engineering & Technology, Chirala, India, (e-mail: palaparthimeghana@gmail.com).*

***Mr. I. Bala Venkateswarlu**, Assistant Professor, CSE, St. Ann's College of Engineering & Technology, Chirala, India, (e-mail: baluatresearch@gmail.com).*

cryptosystems can be implemented by applying more than one algorithm to achieve high security for files which can be called as Multiple-encryption. However, the word Multiple encryption was coined by Ralph [10] in 1981 and suggested double encryption to strengthen the Federal Data Encryption Standard. Triple DES (3DES) is one of the best examples of multiple encryptions. It is a symmetric block cipher which uses three DES keys K1, K2 and K3, each of 56 bits. The process encryption and decryption can be seen from *Table 2.*

| **Application** | **Algorithm Used** | **Input** | **Process** | **Output** |
|---|---|---|---|---|
| K1, K2 and K3 are independent Private keys of 56 bit each | | | | |
| **Sender**<br>PT(64 bits) | DES Encryption (EA) | PT | $ET1=EA_{K1}(PT)$ | ET1 |
| | DES Decryption (DA) | ET1 | $ET2=DA_{K2}(ET1)$ | ET2 |
| | DES Encryption (EA) | ET2 | $ET=EA_{K3}(ET2)$ | ET |
| **Receiver**<br>ET(64 bits) | DES Decryption (DA) | ET | $ET2= DA_{K3}(ET)$ | ET2 |
| | DES Encryption (EA) | ET2 | $ET1=EA_{K2}(ET2)$ | ET1 |
| | DES Decryption (DA) | ET1 | $PT=DA_{K1}(ET1)$ | PT |

*Table 2: Steps of Triple DES*

## II. Related work

Even though 3DES uses a strong key of length 3 x 56= 168 bits, it is still vulnerable to man in the middle attack and its computational cost is very high which can be evidenced by the performance analysis of data encryption algorithms published by Ranjeet [11] and Sachin [12]. Hence, most of the researchers focused on double encryption by combining two encryption algorithms to form hybrid encryption. Diverse applications of double encryption techniques can be observed from the history. In 2013, Nigam [13] proposed a double encryption method by combining two symmetric algorithms i.e. DES and RC4 using three keys along with Huffman compression techniques as the preliminary step, which was intended for both size and security. But, this method is applicable only for text due to Huffman coding and takes large time for encryption due to DES algorithm. Xuewen [14] implemented routing and data security protocol using a symmetric and asymmetric algorithm for RTVP in MANETs.

Recently, Manjula [15] proposed a hybrid method of AES-ECC approach to enhance security for image stenography. In this method, double encrypted data is then compressed with LZW technique to reduce the residing capacity of secret data. Similarly, more secured RSA algorithm with dual modulus using double private and public keys to provide security against brute force attack was proposed by Manu [16]. On the other hand, Content-based double encryption algorithm using symmetric key cryptography was implemented by Sourabh [17] using binary addition operation, a circular bit shifting operation and folding method. Another variety of approach was proposed by Ruiguo [18], which includes a block-chain algorithm for authentication and text encryption based on the content recommendation for the social network.

From all these, it is true that almost all algorithms primarily using symmetric algorithms for the better computational cost. It is also observed that all these algorithms are mainly intended for text data with at least two or three keys. This motivated us to implement a hybrid method for binary data to achieve high security using a single private key; however, another public key will be generated from the contents of the file.

## III. Proposed double encryption method

We experimented with various combinations of symmetric algorithms for the proposed double encryption. Finally, our investigation arrived to combine Blowfish and AES to achieve feasible results in terms of execution time and cryptanalysis. Blowfish algorithm reduces the computational cost while AES increase the strength of proposed method to protect from various attacks. Our proposed method takes only one private key from the user and generates another public key from the contents of file known as Content Based Public (CBP) key and hence CBP key generation is the first step of the proposed method as this key influences the strength of the entire algorithm.

### *A. Content Based Public (CBP) key generation and restoration*

In general, multiple encryption algorithms require two or more keys depending on levels. There are two possible cases: the first case uses the same key for all levels of encryption which lowers strength of key, the second case is considering all or some as independent keys. In the second case user needs to enter all the keys and moreover, key management is also a difficult task. Our method compromises both cases such that one key will be accepted from the user known as users Private key (Kpr) and another key is known as the Content Based Public key (Kcbp) which is generated from the content of the original/encrypted file. Even though Blowfish and AES are symmetric key algorithms, they differ in various parameters including key length, plain text length and number of rounds as follows: AES allows 128, 192 and 256 bits as key length for 128 bit plain text with 10, 12 and 14 rounds depending on key size and Blowfish accepts variable key length ranges from 32 to 448 bits for 64 bit plain text with 16 rounds. By considering this we fixed the key length of proposed method as 128 bits. However other parameters are unchanged. Either encryption or decryption of proposed method starts with Content-based public key generation process as follows:

1. Read first 128 bits or 16 bytes of original file incase of encryption / encrypted file incase of decryption
2. Shuffle the 16 byte positions randomly
3. Consider the resulting bytes as Content-based public key

Finally, ends with Content-based public key restoration process by setting first 128 bits or 16 bytes of encrypted file in case of encryption / original file in case of the decryption process. Even though attacker knows this public key, it is difficult to decrypt file as decryption starts with AES decryption which requires private key first and then public key is required.

### *B. Encryption and Decryption process*

This proposed method uses user's private key (Kpr) and Content-based public key (Kcbp) of 128 bits key each. At the sender, encryption process starts with Content-based public key generation step produces 128 bit content-based key from the original file contents. Then, Blowfish encryption algorithm is used to encrypt the original file using content-based key (Kcbp) to produce a first-level encrypted file (ET1). Now, this first-level encrypted file again encrypted with AES Encryption algorithm using user's private key (Kpr) to produce a double encrypted file (ET). Finally, Key restoration must be done to replace encrypted bytes with content-based Key.

| **Application** | **Algorithm Used** | **Input** | **Process** | **Output** |
|---|---|---|---|---|
| Kpr is users Private key and Kcbp is Content Based Public key of 128 bits key each | | | | |
| **Sender**<br>PT(64 bits)<br>ET1(128 bits) | Key Generation (KGen) | Generates 128 bit *Kcbp* from original file | | |
| | Blowfish Encryption (BlowF) | PT | $ET1=BlowF_{Kcbp}(PT)$ | ET1 |
| | AES Encryption (AES) | ET1 | $ET=AES_{Kpr}(ET1)$ | ET |
| | Key Restore (KRes) | Resets 128 bit *Kcbp* to encrypted file | | |
| **Receiver**<br>ET(128 bits)<br>ET1(64 bits) | Key Generation (KGen) | Generates 128 bit *Kcbp* from original file | | |
| | AES Decryption (AES) | ET | $ET1=AES_{Kpr}(ET)$ | ET1 |
| | Blowfish Decryption (BlowF) | ET1 | $PT=BlowF_{Kcbp}(ET1)$ | PT |
| | Key Restore (KRes) | Resets 128 bit *Kcbp* to encrypted file | | |

*Table 3: Steps for Proposed method*

At receiver, decryption procession starts with Content-based public key extraction from the encrypted file. Then, AES algorithm is used to decrypt the file using a private key to produce ET1. Later, this decrypted file ET1 will again send for decryption using Blowfish decryption algorithm using the content-based public key. Finally, the content-based public key will be restored to get original file. Even though intruders able to track content based public key it is very difficult to decrypt as decryption process need to use private key first and then public key which provides additional security for the algorithm.

## IV. RESULTS AND COMPARISONS

We used Java crypto [19] package to implement the proposed method for comparing execution time and cryptanalysis. Triple DES is the most popular multiple encryption algorithms having three level encryption and hence we have chosen this algorithm to compare results. The following section illustrates the comparisons of results based on two factors including cryptanalysis and computation time.

### *A. Cryptanalysis*

Cryptanalysis [7] strives to break the encryption used to protect information, and to this end there are many techniques available to the modern cryptographer like brute force attack, reverse engineering, guessing, frequency analysis etc. But, Brute force attack is handy for hackers for short and ordinary passwords. For example, assume the length of the password (L) is known to the hacker like L=5 and the number of accepted characters for the password as 84. With this information, Brute force attackers can generate all possible passwords to crack and a number of possible passwords are $84^5$ = 4182119424 = 4.18 billion. If these passwords are generated by a machine having the capability to generate one password per one nanosecond, then the time taken by the hacker to crack it is 4.18 seconds. Hence we considered the length of the password as one of the measures to compare the strength of algorithms as shown in *Table 4*.

| **Parameter** | **Proposed Method** | **3DES** |
|---|---|---|
| Key1 length | 128 bit / 16 byte | 56 bit / 7 byte |
| Key2 length | 128 bit / 16 byte | 56 bit / 7 byte |
| Key3 length | - | 56 bit / 7 byte |
| No. of Possibilities | $256^{16+16}=256^{32}$ | $256^{7+7+7}=256^{21}$ |

*Table 4: Cryptanalysis of brute force attack*

Our method uses two key of length 128 bit / 16 bytes each where the first key is content-based public key used in Blowfish and the second key is a private key used in AES algorithms. This makes total key length as 16+16 = 32 bytes. As each byte has 256 ($2^8$) possible values to crack, the number of possibilities to track two passwords will be $256^{32}$. Similarly, Triple DES uses three independent keys of 56 bit / 7-byte length at each DES algorithm which has the total key length as 3 * 7= 21 bytes. From the table, last row clearly shows that our method takes the large time to crack the keys or passwords than 3DES. Moreover, the content-based key can be the variable length to increase the complexity and cracking time.

### *B. Computational Time*

Computational or execution time plays a key role in case of encryption algorithms after the strength of the password as more execution time causes inconvenience for users. We compared computational time (in milliseconds) for various types of files including PDF, Image, Audio and Video files and execution times are listed in *Table 5*. This table shows the superior performance of our method for all types of files than Triple DES.

| **File Type** | **Proposed Method (msec.)** | **3DES (msec.)** |
|---|---|---|
| PDF file | 2976 | 4345 |
| Image file | 33835 | 56915 |
| Audio file | 23602 | 35738 |
| Video file | 222546 | 336815 |

*Table 5: Computational time for various file types*

Along with we also conducted encryption over files of different sizes including 1MB, 8MB, 20MB and 60MB for comparing performance with the increase of file size and execution times (in seconds) are illustrated in the graph shown in *Figure 1*.

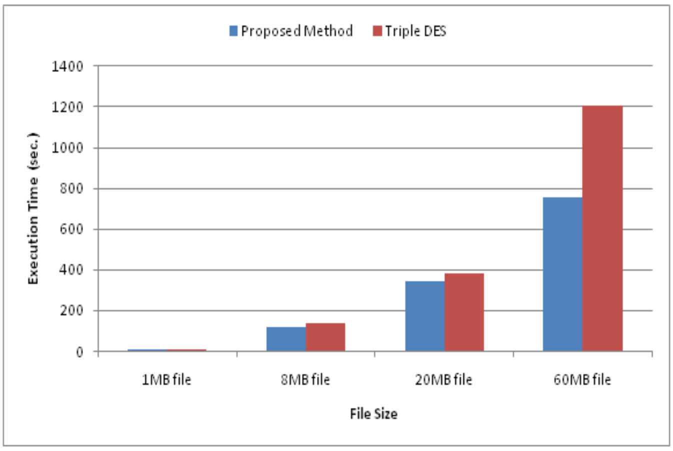

*Fig. 1: Computational time for various sizes of file*

In each case our method showing better computational time than Triple DES. From these two comparisons, it is clear that our method exhibiting considerable improvement in terms of both the strength of keys or passwords and the computational time.

## V. Conclusion

File security became the most significant challenge for the developers of network servers due to the increase in hacking techniques. Hence, single level encryption is not sufficient to protect files from intruders. This recommends using multiple-encryption like triple DES and double encryption algorithms. But, multiple encryption techniques increases computational cost of encryption. Our method reduces computational cost using two single level symmetric encryption techniques including Blowfish and AES. Content-based public key decreases complex key management problems. Moreover, Cryptanalysis results also proved that our method is stronger than other existing Triple DES method due to the usage of two 16-byte keys. This can be further strengthened by using a content-based public key of variable length.

## REFERENCES

[1] Robert Orfali, Dan Harkey and Jeri Edwards: "Client/Server Survival Guide", 3rd Edition, Wiley, ISBN: 978-0-471-31615-2, Feb 1999

[2] International Data Corporation (IDC): "https://www.idc.com/getdoc.jsp?containerId= prUS43274017 ", Nov 2017

[3] Internet World Stats(IWS): "http://www.internetworldstats.com/stats.htm", Jun 2017

[4] Abhishek Vichare, Tania Jose, Jagruti Tiwari and Uma Yadav: "Data security using authenticated encryption and decryption algorithm for Android phones", *International Conference on Computing, Communication and Automation (ICCCA)*, Dec 2017

[5] A. Praveena and S. Smys: "Ensuring data security in cloud based social networks", *International conference of Electronics, Communication and Aerospace Technology (ICECA)*, Dec 2017

[6] Tony Howlett: "Open Source Security Tools: A Practical Guide to Security Applications", Prentice Hall, ISBN: 0-321-19443-8, Jul 2004

[7] M. Arora and S. Sharma: "Synthesis of Cryptography and Security Attacks", *International Journal of Scientific Research in Network Security and Communication (IJSRNSC)*, vol. 5, Iss. 5, Oct 2017

[8] Joost Houwen: "Information Security Management Handbook", 6th Edition, pp. 1255-1269, 2007

[9] Zoran Hercigonja: "Comparative Analysis of Cryptographic Algorithms", *International Journal of Digital Technology and Economy*, vol. 1, Iss. 2, 2016

[10] Ralph C. Merkle and Martin E. Hellman: "On the Security of Multiple Encryption", *Communications of the ACM*, vol. 24, Iss. 7, pp. 465-467, July 1981

[11] Ranjeet Masram, Vivek Shahare, Jibi Abraham and Rajni Moona: "Analysis And Comparison Of Symmetric Key Cryptographic Algorithms Based On Various File Features", *International Journal of Network Security and Its Applications (IJNSA)*, vol. 6, No.4, July 2014

[12] Sachin Sharma and Jeevan Singh Bisht: "Performance Analysis of Data Encryption Algorithms", *International Journal of Scientific Research in Network Security and Communication (IJSRNSC)*, vol. 3, Iss. 1, Feb 2015

[13] Nigam Sangwan: "Combining Huffman text compression with new double encryption algorithm", *International Conference on Emerging Trends in Communication, Control, Signal Processing & Computing Applications (C2SPCA)*, Banglore, India, Oct. 2013

[14] Xuewen Wu, Xiaokai Zhu and Fei Kong: "Routing and Data Security Scheme Based on Double Encryption in Mobile Ad Hoc Networks", *Fifth International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC)*, Qinhuangdao, China, Sept. 2015

[15] Y Manjula and K. B. Shivakumar: "Enhanced secure image steganography using double encryption algorithms", *3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, India, March 2016

[16] Manu and Aarti Goel: " Encryption algorithm using dual modulus", *3rd International Conference on Computational Intelligence & Communication Technology (CICT)*, Ghaziabad, India, Feb. 2017

[17] Sourabh Chandra, Bidisha Mandal, Sk. Safikul Alam and Siddhartha Bhattacharyya: "Content Based Double Encryption Algorithm Using Symmetric Key Cryptography", *Procedia Computer Science*, vol. 57, pp. 1228-1234, 2015

[18] Ruiguo Yu, Jianrong Wang, Tianyi Xu, Jie Gao, Yongli An, Gong Zhang and Mei Yu: "Authentication With Block-Chain Algorithm and Text Encryption Protocol in Calculation of Social Network", *IEEE Access*, vol. 5, pp. 24944 – 24951, Nov 2017

[19] Java Crypto API: https://docs.oracle.com/javase/7/docs/api/javax/crypto/package-summary.html

**Bharatula Lakshmi Sai Prasanna** is currently pursuing her B.Tech. in Computer Science and Engineering from Jawaharlal Nehru Technological University, Kakinada, India. Her areas of interest include Cryptography and Network Security.

**Palaparthi Meghana Murali Krishna** is currently pursuing her B. Tech. in Computer Science and Engineering from Jawaharlal Nehru Technological University, Kakinada, India. Her areas of interest include Cryptography and Network Security.

**Isunuri Bala Venkateswarlu** received his M. Tech. in Computer Science and Engineering from Jawaharlal Nehru Technological University, Hyderabad, India. He received B. Tech. in Information Science and Technology from Acharya Nagarjuna University, Guntur, India. He is currently working as Assistant Professor, Computer Science and Engineering, St. Ann's College of Engineering and Technology, Chirala, India and has 11 years of teaching experience. His research interests include Network Security, Computer Vision and Multi-lingual Sentiment Analysis.