The Research on File Encryption Method Based on File Content Partitioning Restructuring

Hui Xiao¹, Hongbin Wang¹, and Meitong Lin² (Corresponding author: Hongbin Wang)

Faculty of Information Engineering and Automation, Kunming University of Science and Technology¹

No. 727, Jingming South Road, Chenggong New Area, Kunming, Yunnan 650504, P.R. China

School of Computer Science and Technology, Changchun University of Science and Technology²

No. 7089, Weixing Road, Changchun, Jilin 130022, P.R. China

(Email: whbin2007@126.com)

(Received June 23, 2015; revised and accepted Sept. 29 & Nov. 2, 2015)

Abstract

With the development of information technology and the application of information technology jumped into popularity, the electronic document was used as a form of information transmission and data storage, more and more electronic documents are facing the risk of being illegally acquired and viewed. In order to solve this problem, this paper proposes a novel file content protection method, which based on the file content partitioning restructuring. This method strengthens the file content protection and improves the documents security. But it also greatly increases the speed of encryption and decryption process.

Keywords: File encryption, file restructuring, files partitioning, information security

1 Introduction

In today's world, the security of information is associated with valid and reliable encryption algorithms [1]. With the popularity of the network, the individual or company's important files were potential threat (such as illegal steal, view, etc.). Therefore, we need to strengthen the Sensitive information files and important documents secrecy. However, choose what kind of encryption algorithm used to encrypt the file is a problem, and we hope that the algorithm is difficult to break, while the encryption and decryption process is fast enough, do not delay too much time. According to different situation to take different approach; they can better guarantee the important file content security.

Nowadays, there have two types of commonly used encryption algorithms at home and abroad, they are symmetric encryption algorithm and asymmetric encryption algorithm [6]. The DES algorithm is the old typical representative of symmetric encryption algorithm [14], the

security of the encrypted file depends on the length of the password. If the password is not enough length, the encrypted file is easiest to crack by exhaustive method. If we build specialized hardware to crack the DES algorithm encryption file, the less time was being required, in addition, the DES algorithm need to 16 rounds of password replacement and substitution operation, then operate the documents again, it is need to consume a lot of time [5]. Another typical symmetric encryption algorithms like AES algorithm [4], a block encryption standard adopted by the US federal government, this standard is used to replace the original DES algorithm. However, the block length of AES algorithm is fixed at 128 bits, the key length needed is 128, 192 or 256 bits, has many limitations. Moreover, the greater of the used password length, the longer of the encryption and decryption process time-consuming.

The RSA algorithm is a typical representative of the asymmetric encryption algorithm [13], its safety is high, but this algorithm has more calculation. Encryption or decryption large files consumed time is hundreds of times of the AES algorithm, and the security of RSA algorithm depends on the large number decomposition, due to its public key is known, the private key can be calculated according to the public key. If we can find a kind of efficient large number decomposition algorithm, so it is not difficult to break out of the private key [7, 8].

In addition, there are many novelty encryption methods, Hwang et al. [2] proposed a simple batch verifying multiple RSA digital signatures. Their scheme is efficient to reduce computation of verifying multiple RSA signatures. Dong [3] proposed to enhance threshold secret sharing schemes based on the Chinese remainder theorem (CRT) by incorporating the well-known RSA Cryptosystem. Chang et al. [11] proposed an improved version to make the RSA-based certificateless scheme stronger and more secure. Bao et al. [9] proposed a cryptanalysis and improvement of Hwang et al. proposed scheme. Wang [12] proposed a method with bind the executable file to encrypt files, its advantage is run directly executable file input password can view the file, its no need to repeat the encrypted file, but the program need to store the correct password to compare, namely the encrypted file contains the correct password. We also can obtain the password through other means. This method reduces the security of the encrypted file content. Wang et al. [10] proposed a policy based de-duplication architecture, using the mechanism of security proxy (SP) and random storage, which separate storage services and security services to ensure the security of user data and improve the system efficiency at the same time.

But above all, there is a password related information, encrypted file is always insecure, public key may be cracked; stored passwords also can be obtained. According to this problem, this paper present file encryption algorithm based on file content partitioning restructuring to strengthen the file content protection.

The rest of paper is organized as follows. Section 2 introduces the file content partitioning restructuring encryption method. Section 3 introduces the experiment and discussion. Finally, Section 4 concludes this paper.

2 File Content Partitioning Restructuring Encryption Method

2.1 File Encode Procedure

For any file content, when the computer process these documents content, the file content was converted to "0" and "1" binary byte code. If we make the large file encrypted in binary form, the file content encryption degree equivalent to a file "0" and "1" chaos degree. In order to increase a file "0" and "1" chaos degree, we can use a variety of complex algorithms to encrypt the file content, such as exclusive or, modular operation, etc., but it does not necessarily have a high level of encryption, because in the encryption process will inevitably make some useless. For example, a third-order Rubiks cube, American scientists have proved that any disturb the Rubiks cube can be restored within twenty steps, it shows that how many times disturb the Rubiks cube, the Rubiks cube can be restored within twenty steps. The encryption file decoding is just like the revivification Rubiks cube, it through a variety of complex algorithms dealing with ten times or even dozens of times. Maybe it only need a few steps can be restored the file content.

By this we know that as long as the method is reasonable and effective, even through a simple encryption process, the level of encryption may be also complex. Therefore, this paper considers the encryption algorithm based on file content partitioning restructuring, the main idea is: according to the user input password, the passwords ASCII code as password in encoding process. We make the file content divided into the passwords length pieces (such as a passwords length is 20, then the file content is divided into 20 pieces), and each small pieces with the



Figure 1: The file content size and password

corresponding password is proportional to the size (such as user input "lThy9686", the program reads ASCII code "1088410412157545654" as a password, then according to user input passwords each character ASCII value divided file content into 108, 84, 104, 84, 57, 54, 56, 54 pieces), then according to the sorting rule make the partitioning file content integrate into a new file, the sorted new file is a restructured file. So that we can use ASCII international generality to increase the security of encryption algorithm and make the encryption algorithm has scope of application. If the user wants to encrypt files binding the users computer (that is, the encrypt files only allowed decoding on users computer), you can choose the computer's MAC address or hard disk serial number as part of the password in the encryption process, thus someone stealing encoded files to other computers cannot be decoded.

2.2 File Encode Procedure

If you have a 1.74 MB file, we want to encrypt this file based on file content partitioning restructuring (we make file in bytes as the smallest unit to avoid the numerical too big to inconvenience description, in practical cases the smallest unit is bit.). The users need to enter a password first, for example the password is 543294727, and it is converted to ASCII as 53-52-51-50-57-52-55-50-55, in order to make the block proportion has distinct distinction, we use the user enter password numbers to divide the file content. The file content size and password as shown in Figure 1.

We give the input password a sequence number and sort password from small to big order. At the same time, if the password sequence changed, the sequence number will also change. The password sorting is shown in Figure 2.

The file content is divided into nine blocks, which is proportional to the size of password; you can see different password numbers have difference size of file block. The file block sorting is shown in Figure 3.

If we want to obtain each block file content size, we only need sum of the password numbers, then the file content size divided by the sum of the password numbers and can get the unit block size (if any remainder can choose in the



Figure 2: The password sorting

No Sorting		5	4	F 3	3 2	9	4	7	2	7
Sorted	2	2	3	4	4	5	7	7		9

Figure 3: File block sorting

last block), such as Equation (1):

$$1826816 \div (5+4+3+2+9+4+7+2+7) = 42484\cdots 4.$$
 (1)

By Equation (1), the unit block size is 42484 bytes (the remainder is 4), each password corresponding block size is shown in Table 1.

We can see from Table 1, the password corresponding block is proportional to the size (the last password 7's file size is more than the first password 7's file size 4 bytes, because of containing the remainder 4 bytes), the node column denote read each file blocks end position (we assume the head node location is 0, the password 5's position is 0-212420, the password 4's position is 212420-382356, and so on).

The encrypted file is the original file content partitioning restructuring. The sorting process of the file content partitioning is stabilizing order; the same size partitioning does not change the original order. Its size of each partitioning is shown in Table 2.

We can see from Table 2, the node column denote write each file block's end position. We assume the encrypted file's header node location is 0, the first password 2's file content is copying the original file content from 509808 to 594776, and the second password 2's file content is copying the original file content from 1444456 to 1529424, and so on. Such copies of each block and writing into new file, the new file is encrypted files, the process is the original file content partitioning restructuring.

However, this is just a simple example. In the actual situation, we will judge each block. If one block is too big, then encrypt this block by using the same password, and recursion continues until all blocks are small enough.

2.3 File Decode Procedure

The decoding process and the encoding process in the same way, it is sorting operation on file content. Most of operations and encryption process are exactly the same, just order change.

Likewise, the decoding process also requires the user to enter the password; we can refer to Figure 1. the program will rank password refer to Figure 2. The only difference is the encoding process is the inverse process of decoding, according to Table 1 and Table 2, the encryption process is the original documents according to Table 1 read step by step, according to Table 2 written to the file step by step; the encrypted file decoding process is according to the Table 2 read step by step, according to Table 1 written to the file step by step, the process of encoding and decoding depends on the sequence of Table 1 and Table 2.

3 Experiment and Discussions

We use our proposed algorithm and AES algorithm to encrypt the same file and then test the effectiveness of our proposed encryption algorithm. The test document content is "This is a file encryption test.", and encrypt

Password	5	4	3	2	9	4	7	2	7
File size	212420	169936	127452	84968	382356	169936	297388	84968	297392
Termination node	212420	382356	509808	594776	977132	1147068	1444456	1529424	1826816

Table 1: The size of original file block

Table 2: The size of encrypted file block

Password sorting	2	2	3	4	4	5	7	7	9
File size	84968	84968	127452	169936	169936	212420	297388	297392	382356
Termination node	84968	169936	297388	467324	637260	849680	1147068	1444460	1826816

Original text	our algorithm encryption	AES Open SSL encryption		
This is a file encryption test.	-€釞を? ?.d .c {q 尞 n 吰 R- V?	vZ]a觘yH燎01匾 ? f梉 by┤-:+┘		

Figure 4: File block sorting

results are shown in Figure 4.

We use 10 files to verify the performance of these encryption methods, and use the same 128 bits password as EncryptFilesTest. The parameters for computer is AMD A4 CPU 1.5 GHz and RAM 6GB, the file sizeencryption time cost and decryption time cost are shown in Figure 5.

We can be seen from Figure 5, with the increase of encrypted file contents, our method on the encryption and decode time consumption is better than the AES algorithm. The experimental results verify our algorithm efficiency. The file encryption based on the file content partitioning restructuring, which encryption process is read and write the file process. Just only read the order and position is different, namely the cloning process is the encrypt files, so that we can maximize the speed of encryption process. At the same time, the file content only needs a simple calculation before file encryption (namely, we need to compute the size of each file content block and end location), encryption process does not occupy too much time. The execution speed of these two encryption algorithms based on encrypt and decrypt the biggest file as shown in Table 3.

By encoding/decoding process, the file itself and the program will not store any information related to the password, the password is only used as a keyword, namely the user enter a password would be to encrypt or decrypt files once, the program will not compare password correctly or not, it also ensures that no one except the user know the correct password, don't leave any information may calculate the password.

Our encryption method also made the exhaustive method decrypt file changed unrealistic and doesn't like other encryption algorithm. It use the password in any

length (the length greater than 0). Due to the algorithm would recursion encrypt file until all blocks are small enough, it is possible to provide the same security for different lengths of password, but the encryption time is likely to change. In this case, you do not know the password is 128 bits or 1024 bits, it is not a fixed size, and it may be an odd number of bytes such as 17 or 23. Sometimes you have to decrypt the file completely then you could know the password is correct or not, instead of just compare password. According to the results above, a 600MB file decryption will consume nearly a minute of time, a nine digit password has a billion of possibilities, the exhaustive method to crack the code will be consumed more than 1900 years.

Thus if the user input the correct password decrypt documents will not cost too much time by our proposed method, the decryption speed always depends on the hard disk read and write speed, but the exhaustive method would increase cost observably.

4 Conclusions

In this paper, we proposed an efficient method for file content based on the file content partitioning restructuring. So we can use different sorting ways to enhance the encryption algorithm performance, the encryption algorithm is still has insufficient place, such as the large file encryption need cost slightly longer times, encryption require larger memory space. We will continue to improve our encryption method in the future work.

	_	Time cost (second)						
File size	Number of bytes	our algo	orithm	AES Open SSL				
		Encryption	Decryption	Encryption	Decryption			
2.63KB	2,696	0.001	0.001	0.001	0.001			
256KB	262,656	0.016	0.015	0.031	0.031			
1.62MB	1,707,590	0.421	0.187	0.249	0.124			
6.72MB	7,057,203	1.591	0.405	0.999	0.578			
44.5MB	46,743,761	4.649	2.371	4.274	3.761			
107MB	112,895,950	8.985	5.741	10.172	9.063			
616MB	646,462,820	54.389	32.636	66.116	52.104			
1.09GB	1,174,199,531	96.341	59.751	119.045	95.417			
1.58GB	1,703,508,231	144.767	87.974	187.816	139.231			
1.97GB	2,121,837,981	204.319	119.326	219.324	174.675			





Figure 5: The encryption/decryption cost time comparison chart

Table 3: The	execution	speed
--------------	-----------	-------

Execution speed (MB/s)	our algorithm	AES Open SSL
Encryption	9.9	9.2
Decryption	16.9	11.6

Acknowledgments

This work is supported by the National Nature Science Foundation of China (61462054); the Science and Technology Plan Projects of Yunnan province (2015FB135); the Key Project of Yunnan province Education Department (2014Z021); the Nature Science Research Foundation of Kunming University of Science and Technology (KKSY201403028).

References

- M. Babaei, "A novel text and image encryption method based on chaos theory and dna computing," *Natural Computing*, vol. 12, no. 1, pp. 101–107, 2013.
- [2] C. C. Chang and M. S. Hwang, "Parallel computation of the generating keys for rsa cryptosystems," *IEE Electronics Letters*, vol. 32, no. 15, pp. 1365– 1366, 1996.
- [3] X. Dong, "A multi-secret sharing scheme based on the crt and rsa," *International Journal of Electronics* and Information Engineering, vol. 2, no. 1, pp. 47– 51, 2015.
- [4] M. X. He and H. Lin, "Implementation of the advanced encryption standard (aes)," *Application Re*search of Computers, vol. 12, no. 0, pp. 61–63, 2002.
- [5] M. Hu and R. Lu, "Analysis and research of the security of DES algorithm," *Journal of Inner Mongolia University (natural science edition)*, vol. 36, no. 6, pp. 693–697, 2005.
- [6] M. S. Hwang and I. C. Lin, Introduction to Information and Network Security (in Chinese), 5ed., McGraw-Hill, 2014.
- [7] Y. Jin, X. Cui, N. Jiang, "Design of file encryption system based on RSA algorithm," *Journal of Dalian Nationalities University*, vol. 15, no. 5, pp. 535–538, 2013.
- [8] Y. Kou, J. Tian and C. Chen, "Research and simulation of timing attacks on rsa," *Computer Technology* and *Development*, vol. 20, no. 8, pp. 150–158, 2010.
- [9] C. C. Lee, F. Bao and M. S. Hwang, "Cryptanalysis and improvement on batch verifying multiple rsa digital signatures," *Applied Mathematics and Computation*, vol. 172, no. 0, pp. 1195–1200, 2006.
- [10] Y. Lu, Z. Y. Wang and G. Z. Sun, "A policy-based de-duplication mechanism for securing cloud storage," *International Journal of Electronics and Information Engineering*, vol. 2, no. 2, pp. 70–79, 2015.
- [11] C. Y. Sun, C. C. Chang and S. C. Chang, "A strong RSA-based and certicateless-based signature scheme," *International Journal of Network Security*, vol. 18, no. 2, pp. 201–208, 2016.
- [12] X. Wang, "Based on the bundled file encryption technology and implementation," *Journal of TaiYuan* Urban Vocational College, vol. 12, pp. 135–136, 2012.
- [13] G. Zhang, H. Wang and Z. Li, "Public key cryptography RSA algorithm," *Information Technology*, vol. 0, no. 8, pp. 34–36, 2011.

[14] J. Zhang and L. J. Zhu, "The analysis and realization of arithmetic of des encryption," *Software Guide*, vol. 3, no. 36, pp. 95–97, 2007.

Hui Xiao is a college undergraduates student. His current study in Faculty of Information Engineering and Automation from Kunming University of Science and Technology. His major is computer science and technology.

Hongbin Wang received his Ph.D in computer science in 2013 from the Jilin University, Changchun, China. He is a lecturer in the Faculty of Information Engineering and Automation from Kunming University of Science and Technology. His current research interests include intelligent information system, natural language processing and network security.

Meitong Lin is a college undergraduates student. Her current study in School of Computer Science and Technology from Changchun University of Science and Technology. Her major is computer science and technology.