Application of AES and DES Algorithms in File Management

Yan Wang1,*
1Business College, Anhui Vocational College of Industrial Economics, Hefei, 230051, Anhui, China

*Corresponding author e-mail: wyan2032@ahiec.edu.cn

Abstract. In recent years, with the rapid development of computer and network technology, major enterprises and institutions have successively popularized the use of computer for office, and electronic documents have gradually replaced paper documents. Important enterprise data, such as customer information, are stored in the computer in the form of electronic files. The use of computers in office has brought us many benefits such as high efficiency and resource saving, but also correspondingly caused the safety management of electronic documents. For example, the risk of information leakage is greatly increased, and office personnel may cause irreparable damage to electronic documents due to misoperation. This article takes the current situation of electronic document security management as the research background, and proposes a solution to securely manage electronic documents. Combine the technical theory of file management with the AES encryption algorithm, combine the theoretical principle of file management with the DES encryption algorithm, and use the characteristics of file management to make up for the defects of AES and DES algorithms to achieve better encryption effects and improve Difficulty of breaking secrets. At the same time, it takes into account file query and junk file disposal; through file remote transmission, secure communication between Internet users is realized.

Keywords: AES Encryption Algorithm, DES Encryption Algorithm, File Management, Information Encryption Technology

1. Introduction

In recent years, with the rapid development of computer and network technology, computers have been widely used in all walks of life in society, especially the rise of the Internet of Things, and electronic documents have increasingly replaced original paper materials [1]. For example, important data files of enterprises are all stored on computers in the form of electronic documents; at the same time, electronic documents and e-mails are gradually used for both internal and inter-enterprise communication and communication. The use of computer electronic office brings convenience to enterprise management and improves work efficiency. However, due to the inherent characteristics of
computers and the Internet of Things, this also increases the risk of information leakage [2]. Currently, the main threats to electronic documents: First, malicious intrusion from the outside refers to malicious third parties who use relevant computer and network technology to intrude into the internal computer of the enterprise through illegal means, and eavesdrop, steal, modify, and modify confidential data files. Destructive operations such as deletion; second, internal leakage refers to the use of convenience or corporate management loopholes to carry confidential data files out of the company using mobile devices such as U disks, causing the company to suffer huge losses [3]. According to investigations, among the numerous security incidents, the ratio of security incidents due to external intrusions and internal enterprises has reached 3:7 [4].

Encryption technology is a simple and effective way to protect user data from illegal access. Many traditional encryption technologies are currently widely used in various fields of Internet technology, providing security for data storage in the cloud, and reducing user concerns. Data security concerns [5]. Encryption algorithms are widely used in the fields of remote communication security and storage encryption. The basic concept of encryption is to prevent information from being stolen, deciphered and used illegally during transmission [6]. Since the dedicated core components are decisive and cannot be changed, if the encryption system is not flexible, it will be difficult to meet the multi-stage security performance requirements of multiple encryption users and the continuous upgrade of encryption algorithms, so there is a security risk [7]. The reconfigurable encryption chip uses the hardware resources that can be reused by the innovative encryption chip to flexibly change its own hardware structure according to different application requirements, and the internal and external structures of different encryption algorithms are consistent, and the function of the encryption chip is greatly improved. Flexibility, security and scalability [8].

With the continuous development of cryptography research and the rapid improvement of computer performance, cryptographic interpretation technology is also continuously updated. The AES algorithm is not only efficient and concise, but the most important thing is that it can resist all types of attacks well known to the AES algorithm [9]. At present, data encryption, especially data with high real-time requirements like video data, gradually replaces the DES algorithm with the AES algorithm. Not only embodies the realization of DES software, but also embodies the realization of AES's common embodying algorithm and fast algorithm respectively. In order to compare the encryption and decryption efficiency of DES algorithm and AES algorithm, and the two implementation methods of AES algorithm, the use of AES and DES algorithm to file encryption experiment [10].

2. Algorithm Establishment

2.1. AES Encryption Algorithm

To quickly implement the AES algorithm in a processor with 32 bits or more, it is best to combine multiple stages of the round transformation into a set of table queries. The implementation of this method is described in detail below.

Assuming that the input of the round transform is represented by a and the output of Sub Bytes is represented by b, then:

\[ b_{ij} = S_{RD}[a_{ij}], \quad 0 \leq i < 4; \quad 0 \leq j < N_b \]  

(1)
Let the output of Shift Rows be denoted by c, and the output of Mix Columns by d:

\[
\begin{bmatrix}
c_0,0 \\
c_1,0 \\
c_2,0 \\
c_3,0 \\
\end{bmatrix} =
\begin{bmatrix}
b_{0,j} + c_0 \\
b_{1,j} + c_1 \\
b_{2,j} + c_2 \\
b_{3,j} + c_3 \\
\end{bmatrix}, 0 \leq j < N_b
\]  

(2)

\[
\begin{bmatrix}
d_0,0 \\
d_1,0 \\
d_2,0 \\
d_3,0 \\
\end{bmatrix} =
\begin{bmatrix}
02 \\
01 \\
01 \\
03 \\
\end{bmatrix} \cdot
\begin{bmatrix}
c_0,0 \\
c_1,0 \\
c_2,0 \\
c_3,0 \\
\end{bmatrix}, 0 \leq j < N_b
\]  

(3)

The above matrix multiplication operation can be changed to the following algorithm:

\[
\begin{bmatrix}
d_0,0 \\
d_1,0 \\
d_2,0 \\
d_3,0 \\
\end{bmatrix} =
\begin{bmatrix}
02 \\
01 \\
01 \\
03 \\
\end{bmatrix} S_{RD}[a_{0,j}c_0] \oplus
\begin{bmatrix}
03 \\
02 \\
01 \\
01 \\
\end{bmatrix} S_{RD}[a_{1,j}c_1] \oplus
\begin{bmatrix}
01 \\
03 \\
02 \\
01 \\
\end{bmatrix} S_{RD}[a_{2,j}c_2] \oplus
\begin{bmatrix}
01 \\
03 \\
02 \\
01 \\
\end{bmatrix} S_{RD}[a_{3,j}c_3], 0 \leq j < N_b
\]  

(4)

Define 4 T tables: T₀, T₁, T₂ to T₃:

\[
T_0[a] =
\begin{bmatrix}
02 \cdot S_{RD}[a] \\
01 \cdot S_{RD}[a] \\
01 \cdot S_{RD}[a] \\
03 \cdot S_{RD}[a] \\
\end{bmatrix} \\
T_1[a] =
\begin{bmatrix}
03 \cdot S_{RD}[a] \\
02 \cdot S_{RD}[a] \\
01 \cdot S_{RD}[a] \\
01 \cdot S_{RD}[a] \\
\end{bmatrix}
\]  

(5)

\[
T_2[a] =
\begin{bmatrix}
01 \cdot S_{RD}[a] \\
03 \cdot S_{RD}[a] \\
02 \cdot S_{RD}[a] \\
01 \cdot S_{RD}[a] \\
\end{bmatrix} \\
T_3[a] =
\begin{bmatrix}
01 \cdot S_{RD}[a] \\
01 \cdot S_{RD}[a] \\
03 \cdot S_{RD}[a] \\
02 \cdot S_{RD}[a] \\
\end{bmatrix}
\]  

(6)

Each T-list has 256 4-byte items and requires 4KB of storage space. In order to use the data in these tables more conveniently, the above formula (4) can be rewritten as the following formula (7):

\[
\begin{bmatrix}
d_0,0 \\
d_1,0 \\
d_2,0 \\
d_3,0 \\
\end{bmatrix} =
T_0[a_{0,j} + c_0] \oplus T_1[a_{1,j} + c_1] \oplus T_2[a_{2,j} + c_2] \oplus T_3[a_{3,j} + c_3], 0 \leq j < N_b
\]  

(7)

2.2. DES Encryption Algorithm

The length of the general text processed by DES is 64 bits, the length of the ciphertext group is 64 bits, and the key length used is 56 bits. In fact, this number requires 64 bits, but only weak bits are used. The remaining 8 bits can be used as block positions or completely set. The decryption process of DES is similar to encryption. Decryption uses the same algorithm as encryption, but uses the subkey in the reverse order. All DES systems are public, and the security of the system depends entirely on the confidentiality of the key.
Assume that \( B_i \) is the result of the i-th iteration, \( K_i \) is the 48-bit key of the i-th round, and \( f \) is the function that realizes each round after performing substitution, permutation, and key exclusive OR operations:

\[
\begin{align*}
L_i &= R_{i-1} \\
R_i &= L_{i-1} \oplus f(R_{i-1}, K_i)
\end{align*}
\]  

(8)

Specific steps are as follows:

1. The initial arrangement of input packets will be performed before the first round of calculation.

2. Replacing the key: In order to confirm whether the key is wrong, the 8th bit of each byte on the DES key is used as a parity check, so the actual key length of the DES is 56 bits. In each link of DES, different 48-bit subkeys are generated in 56-bit keys.

It can provide faster encryption and decryption speed, but to a certain extent, the odd safety factor is reduced.

The encryption process is written as:

\[
c = E(m) = IP^{-1}oT_{16}oT_{15}o...oT_1oIP(m)
\]  

(9)

The decryption process is written as:

\[
c = D(c) = IP^{-1}oT_1oT_{12}o...oT_{16}oIP(c)
\]  

(10)

3. Modeling Method

3.1. File Encryption Model Establishment

In order to prevent errors and omissions of information and increase the complexity of model calculation, it is necessary to expand feature selection based on 30 combinable feature levels to form a relatively small optimal feature set. The \( S \) value is an information standard value that reflects the degree of influence of the file encryption feature value. The more we get, the stronger the predictive ability of the target.

\[
WOE_i = \ln \left( \frac{p_{yi}}{p_{yn}} \right)
\]  

(11)

\[
S = \sum (p_{yi} - p_{yn}) \cdot WOE_i
\]  

(12)

First, use a probability distribution \( D_m \). Learn from the training data set to get the basic security model:

\[
T_m(x): X \rightarrow \{-1, +1\}
\]  

(13)

Second, calculate the classification error rate of this processor.

\[
e_m = P(T_m(x_i) \neq y_i) = \sum_{i=1}^{N} w_i \exp(-a_m y_i T_m(x_i))
\]  

(14)

Calculate the coefficient of \( T_m(x) \):
a_m = \frac{1}{2} \log \frac{1 - e_m}{e_m} \tag{15}

Update the probability distribution of the training data set:

\[ W_{m+1,i} = \frac{w_{mi}}{z_m} \exp \left( -a_{m,i} T_m(x_i) \right), (i = 1, 2, ..., N) \tag{16} \]

After completing the above steps, construct a linear combination of classifiers:

\[ f(x) = \sum_{m=1}^{M} a_m T_m(x) \tag{17} \]

4. Evaluation Results and Research

With the advancement of science and technology, human society has entered the electronic age. Computers and the Internet play an irreplaceable role in work and life every day, bringing more meaning to human life. At the same time, a series of information security issues have also arisen, ranging from personal privacy leakage to the eavesdropping, stealing and modification of important corporate data. In order to solve the above problems, some software with scientific and effective algorithms is urgently needed to realize the management and protection of electronic documents. Just like this, the file management system is developed for the needs of the current society.

The experiment will focus on the main functional modules and implementation of the file management system, and briefly introduce and understand the functional framework of the system. The upgraded encryption algorithm is suitable for file management system. That is, AES algorithm and DES algorithm of chaos theory are applied to the system, and finally the file management system is tested.

The AES and DES encryption algorithms encrypt and decrypt files of different lengths. The processing time of the file management system is tested. The processing time of files of different lengths is different. The response time is shown in Table 1.

| Algorithm | File size (bytes) | Encryption (ms) | Decryption (ms) |
|-----------|-------------------|-----------------|-----------------|
| AES       | 996620            | 20              | 21              |
|           | 1822550           | 50              | 56              |
|           | 10378048          | 524             | 578             |
|           | 50124420          | 1245            | 1356            |
| DES       | 996620            | 16              | 16              |
|           | 1822550           | 43              | 44              |
|           | 10378048          | 434             | 453             |
|           | 50124420          | 894             | 945             |

The test results in Table 1 show that the encryption algorithm of the file management system can realize file encryption operations normally, stably and quickly, and meet the requirements of use.

When the file management system introduces the algorithm security mechanism, there will be a certain loss in system performance. This is an unavoidable problem in reality. If the performance loss, such as the loss of transmission rate, is within an acceptable range, then overall Said, still has practical significance. The same file size uses three transmission methods, namely unencrypted, AES
encryption and DES encryption. The selected file sizes are in order: 0.2GB-1.2GB, and files of this size are more universal.

The time consumption of uploading files is shown in Figure 1. The vertical axis in the Figure is the time spent uploading files, in seconds (s).

![Figure 1](image1.png)

**Figure 1.** Time consumption of uploading files in different situations

From the experimental results in the Figure, it can be seen that if the upload file is unencrypted, the transfer rate is generally faster than the encrypted upload file. When using encrypted upload files, the encryption algorithm AES is slightly faster than DES. In terms of user experience, the extended upload time due to encryption is still acceptable.

The time consumption of downloading files is shown in Figure 2. The vertical axis in the Figure is the time spent uploading files, in seconds (s). The data in the Figure shows that the trend of downloading files and uploading files is almost the same, that is, downloading files that need to be decrypted consumes more time than files that do not need to be decrypted, and downloading files with large sizes also consumes more time than files with small sizes. In general, the time consumption of downloading is smaller than that of uploading. It satisfies the needs of sharing in the system, and is suitable for a single write and multiple read process of file data in sharing, that is, the characteristics of frequent download operations.

![Figure 2](image2.png)
Figure 2. Time consumption of downloading files in different situations

It can be seen from the Figure that using DES encryption to upload files, the client's CPU occupancy is the largest, and the two methods of unencrypted uploading files and AES encrypted uploading files are generally consistent in terms of CPU usage. The increase in the size of the transferred file did not cause the CPU occupancy rate to increase all the time. When it reached a certain size, the CPU occupancy rate showed a downward trend. It can be seen from the experimental results in the Figure that the system is more suitable for large-size file transfer.

5. Conclusion

Encrypting files is a problem that needs to be solved urgently, and designing a scientific and effective encryption algorithm is also the main direction of current research. Encryption algorithms include DES, AES, etc. In order to better verify whether the data is damaged in the transmission process, the method of extracting the file data and other related operations is adopted to ensure the integrity of the data. This article introduces the AES and DES encryption algorithms, and applies the algorithms to the file management system to integrate theory with practice and apply theory to reality. Apply AES and DES algorithms to the specific file management system, so that the theory can be realized, landed and tested. The development of the file management system greatly facilitates the user's file management, has a certain social practicability, and can produce a certain social value.

References

[1] Albahar M A , Olawumi O , Haataja K , et al. Novel Hybrid Encryption Algorithm Based on Aes, RSA, and Twofish for Bluetooth Encryption[J]. Journal of Information Security, 2018, 09(2):168-176.

[2] Murugan C A , Karthigaikumar P , Priya S S . FPGA implementation of hardware architecture with AES encryptor using sub-pipelined S-box techniques for compact applications[J]. Automatika, 2020, 61(4):682-693.

[3] Agarwal A , Raj P , Katiyar S . Secured Audio Encryption using AES Algorithm[J]. International Journal of Computer Applications, 2019, 178(22):29-33.

[4] Friedhelm, Meyer, auf, et al. Paul Curzon, Peter W. McOwan: Computational Thinking; Die Welt des algorithmischen Denkens – in Spielen, Zaubertricks und R?tseln[J]. Mathematische Semesterberichte, 2019, 66(2):259–260.

[5] Kumar S S , Sherfin A . A cryptographic encryption technique byte – Spiral rotation encryption algorithm[J]. Journal of Discrete Mathematical Sciences and Cryptography, 2019, 22(6):1-6.

[6] Beenu V P . Mathematical formulation of des algorithm[J]. International Journal of Pure and Applied Mathematics, 2018, 118(10):363-376.

[7] Uka K K , Oguoma S I , Chuma-Uba U P . Analysis of Blockchain Architecture in File Sharing Management for Tertiary Institution[J]. Intelligent Information Management, 2020, 12(3):88-104.

[8] Zhu X G , Shen Y . MapReduce optimisation information query method for file management.
system[J]. International Journal of Reasoning-based Intelligent Systems, 2018, 10(2):117-121.

[9] Khlif N, Ghorbel A, Aydi W, et al. Generation of Chaotic Signal for Scrambling Matrix Content[J]. International Arab Journal of Information Technology, 2020, 17(4):548-553.

[10] Yang L, Yang B, Xiang C. Quantum public-key encryption schemes based on conjugate coding[J]. Quantum Information Processing, 2020, 19(11):1-16.