Design and Realization of Once a Secret Key Encrypt System

Xiujuan Yan

School of Management, Xi'an University of Science and Technology, Xi'an 710054, China

Abstract

Once a secret key encrypt system in V C ++6 .0 environment is designed and realized .Aiming at the characteristic of file encrypting :there are the same or similar information about file type in the document's head and bottom , in order to prevent the infection which product those information to file encrypt ,it put forward a specifically encrypting method aim at system. moreover, the encrypt validity put up validate using the method of unlimited numerate and analyses decrypt .At the same time designed and realized secret key producing system , through the especially arithmetic , make the serial number of secret key is more closet o random serial number. User can enter secret key's serial number , the system automatically use the correspond secret key encrypt to the file ,and also can user defined secret key , updated the secret key in a period of time , insure the system possess upper security.

Keywords

Encrypt technology ;one-time one secret ;random key.

1. Introduction

"Once a secret key" is an ideal encryption scheme. Theoretically speaking, the password of " Once a secret key " management is indecipherable[1]. Each key is used only once for a single message. The sender encrypts the message, and then destroys one page or tape used in the code. The receiver has the same scrambled copy to decrypt each character of the ciphertext in turn. Part of one page or used tape will be destroyed after receiver’s decrypting the message.

The new message is encrypted with a new encryption key. The plan is so completely confidential that the stealer can't get one-time pad used to encrypt the message. The ciphertext message given must be equivalent to any possible plaintext message of the same length. The key system is the core and the key letters must be randomly generated[2]. The attack of "once a secret key" scheme is mainly aimed at the algorithm of generating key sequence. If true random source is used, then it's safe. Through a variety of random generation algorithm of security analysis, we choose Microsoft' s CryptoAPI function, tested by the federal security information processing center (FI PS 140-1) to generate a random number, which can resist attacks against key generation algorithms.

Random key sequence XOR or non random plaintext message generates completely random ciphertext message, and the larger computing power can't be done for deciphering.

2. Design of encryption system

System encryption uses each bit of a file to encrypt with each bit of XOR, the data will be written to the new file and form encrypted files. The key length is 256 bytes, namely 2048 bits.

```
/ *File encryption core program */
……
while(pw d[ ++j0] );
ch =fgetc(f p1);
/ *Encryption algorithm start */
```

```
while(!feof(fp1)){
fputc(ch pw d[ j >=j0;j=0;j ++],fp 2);/ *Exclusive or then write fp 2*/
ch =fgetc(f p1);
}
……
```

For file encryption, the same file has the same or similar information about the file type at the beginning and end of the file, which does great help to decipher the encryption. To prevent the impact of this information on file encryption, a random sequence should be added to the head and end of the encrypted file when encrypting a file, that is, the interference item. The length of the random sequence is determined by the ASC code value N of the first character I in the key serial number I, 32 < N <128 .

The decryption of the file first deletes the interference item in the file, and then encrypts it with the I key.

3. System design of key generation

The key system is the core of the system, and the key is composed of the sequence of random numbers. Because that the random number generated by computer is pseudo random number, there is a certain regularity for the generation of random numbers, which is the bottleneck of the "one time one secret" algorithm. In cryptography, the randomness of a sequence is defined in this way:

(1) It seems to be random, namely it can be checked by all the correct randomness that people can find.

(2) This sequence is unpredictable, in other words, even if the algorithm or hardware design of the sequence and all the knowledge of the previous sequence is generated, it is impossible to predict what the next bit is.

(3) This sequence cannot be repeated, even if the sequence generator operates two times with exactly the same input under the same operating conditions, two completely unrelated bit sequences will be obtained.

CryptAcquireContext(…) and CryptGenRandom(…) in Microsoft's CryptoAPI are used to generate random numbers. Besides, CryptoAPI has passed the FIPS (F IPS 140-1) on random number security statistics test. The way CryptoAPI get randomness is acquiring the current process information, current thread information, high accuracy and high performance counters, user environment module MD4, high precision internal CPU calculator and the underlying system information, such as free space, page count etc. which is also called “entropy”. Compared with C + + TimeGetTim(E) function to initialize the random number seed[3], namely :srand((unsigned)timeGetTime()); Simple use of time as seed has higher safety, which can be assumed that the generated random number is secure . The following is the core code for system key generation[4]:

```
}
bool CCrypRandom ::get(void *lpGoop , DWORD cbGoop)
{
if(! m-rProv) return false ;
return;
CryptGenRandom(m rProv , cbGoop , reinterpret cast <LPBYTE
>(lpGoop));
}
// Generate random number
……
CCrypt Random r;
long int num ;//The number of random Numbers
```

```
int ranze ;//Random number range
int point ;//Effective decimal place
……
for(long int i =0;i <num;i ++)
 {if(r .get(&rand , sizeof rand)) if(ranze==1)
 {rand =rand(int)(pow(10, point));
t -rand =(double)rand;
t -rand =t-rand/ pow(10, point);}
else
{rand =rand %ranze;
t -rand =rand ;}
```

4. System implementation and system security analysis

4.1 system implementation

The system adopts VC ++6.0 environment, The key generation system generates 500 keys randomly at regular intervals .The key is encrypted by the predefined master key of the system.

4.2 Verification of cryptographic strength

The main purpose of information encryption is to maintain the security of plaintext or key and prevent interpreter from attacking[5] . At present, the most commonly used attack methods are exhaustive method and analytic decoding method[6]

(1) Exhaustive method: The key of the system is a sequence of random numbers with 256 characters(2048 bytes) .The length of the key is 2048 bytes. Adding the interference item, the length is 2560~4096 bits and each random number ranges from ASC to 32~128 .The number of keys is extremely huge, that is (128 -32)*(128 -32)*…(128 -32)≈10512. Obviously, using exhaustive method has no practical significance to decipher the intercepted ciphertext.

(2) Analytical decoding: It includes two categories: statistical analysis, deciphering and deterministic decoding. This method mainly uses the statistical characteristics of natural language used in ciphertext and various mathematical methods to decipher ciphertext.

(3) Statistical analysis decoding method: Chinese encoding takes up to two bytes and 16 bits, do XOR operation with 8 bit key element and completely messy code is received. Statistical analysis of ciphertext is only suitable for replacing cipher encryption system.

The interpreter generally works under the following three conditions: ciphertext only attacks known ciphertext decoding and chosen plaintext decipher. The choice of plaintext is the most favorable condition for the interpreter.

For chosen plaintext decipher, the key information can be obtained by the interpreter. Due to that each key is different, the interpreter must search in the key space every time when the three favorable conditions in the early stage are not very helpful to next decoding. This is exactly the reason why the encryption system of " Once a secret key " can not be deciphered.

4.3 Improvement

(1) Increase key space: At present, the key space is in the range of ASC code, and the next step is to use UN ICODE encoding space.

(2) Encrypting files by hashing: According to the order of file storage space address, we use hash method to store encrypted files, which can prevent the same type of file from having the same header and ending information of the file. It effectively provides help to the interpreter.

(3) Update key generation algorithm: By updating the key generation algorithm, it makes more approximate to random sequence and eliminates the possible regularity

5. Summary

In view of the current network information security situation, we design and bring about "Once a Secret Key" Encrypt System as well as secure key generation system in order to achieve secure file transmission between the two institutes. The corresponding encryption algorithm is given for file encryption, based on the theory of random number generation, the random number generation system is used in accordance with the safety statistical test standard (FIPS 140-1). It effectively enhances the security of "Once a Secret Key" encrypt system.

The system has been used in the research institute of the two files encrypted transmission, simple and effective, the encryption speed is fast with a good result in application. The next step will improve the system in terms of the improvement ideas, so as to make it more secure, effective and reliable.

References

[1] Lu Kaicheng. Computer cryptography - data secrecy and security in computer network [M]. 3rd edition. Beijing: tsinghua university press, 2003.
[2] Qin Sihan secretary of technology and computer network security [M]. Beijing: tsinghua university press, 2001.
[3] Yang changhong, xu bei.c ++ language great [M]. Beijing: electronic industry press,1994
[4] Howard, Blanc l. Writing Secure Code [M]. Cheng Yongjing. translation. (mechanical industry press, 2002.
[5] Jiao Zhanya One secret at a time study on cryptographic algorithms [J]. Journal of xi 'an university of science and technology, 2005(4): 4, 77-480.
[6] Lu Kaicheng. Computer cryptography [M]. Beijing: tsinghua university press, 1998:73-75.