Implementation and Comparison of Symmetric Key Crypto System

Aung Yar Zar
University of Computer Studies (Mandalay)
aungyz82@gmail.com

Abstract

In the world of today, computer and Internet have become part of our everyday lives. Security for data and information are very important. Secure communication is an intrinsic requirement for many popular online transactions such as e-commerce, stock trading and banking. These transactions employ a combination of public-key and symmetric key cryptography to authenticate participants and guarantee the integrity and confidentiality of information in transit. Data confidentiality and authentication are normally provided using cryptographic techniques. Cryptography is either based on symmetric keys or asymmetric keys. This system intends to analyze and implement the most popular symmetric cryptographic algorithms Data Encryption Standard, Advanced Encryption Standard, International Data Encryption Algorithm and RC4. Based on implementation and study, runtime comparison between the symmetric cryptosystems has been made.

1. Introduction

Cryptography is the science of using mathematics to encrypt and decrypt data. Cryptography enables to store sensitive information or transmit it across insecure networks like the Internet so that it cannot be read by anyone except the intended recipient. While cryptography is the science of securing data, cryptanalysis is the science of analyzing and breaking secure communication Classical cryptanalysis involves an interesting combination of analytical reasoning, application of mathematical tools, pattern finding, patience, determination, and luck. Cryptanalysts are also called attackers. Cryptology embraces both cryptography and cryptanalysis.

Cryptography is used to protect data while it is being communicated between two points. Cryptography is either based on symmetric keys or asymmetric keys [1]. This paper analyzes and studies between the most popular symmetric cryptographic algorithms such as data encryption standard, advanced encryption standard, international data encryption algorithm and RC4. Based on analyze and by doing experiment, runtime comparison between the symmetric cryptosystems have been made.

This paper is organized with six sections. The first section is introduction of the system. Section two explains related work for the system. Section three explains symmetric algorithms used in this paper. Section four describes the implementation of the system. Experimental result is explained in section five and the next is conclusion of the system.

2. Related work

To understand the comparison of symmetric key crypto system, in this paper, it describes other systems presented by many researchers.

Andreas Sterbenz and Peter Lipp analyzed the five AES candidate algorithms MARS, RC6, Rijndael, Serpent, and Twofish as well as DES, Triple DES, and IDEA by examining independently developed Java implementations. It gave performance results on several platforms, list the memory requirements, and present a subjective estimate for the implementation difficulty of the algorithms. [2]

Abdel-Karim Al Tamimi described the two main characteristics that identify and differentiate one encryption algorithm from another are its ability to secure the protected data against attacks and its speed and efficiency in doing so. It provided a performance comparison between four of the most common encryption algorithms: DES, 3DES, Blowfish and AES. The comparison had been conducted by running several encryption settings to process different sizes of data blocks to evaluate the algorithm's encryption/decryption speed. [3]

3. Symmetric algorithms

The Symmetric algorithms use the same key for encryption and decryption while asymmetric

algorithms use a different key for encryption and decryption, and the decryption key cannot be derived from the encryption key. Symmetric algorithms can be divided into stream ciphers and block ciphers. Stream ciphers can encrypt a single bit of plaintext at a time, whereas block ciphers take a number of bits (typically 64 bits in modern ciphers), and encrypt them as a single unit. There are many cryptographic algorithms. This thesis analyzes and studies four of the most common symmetric algorithms DES, AES, IDEA and RC4. [5]

3.1. Data encryption standard (DES)

DES encrypts and decrypts data in 64-bit blocks, using a 64-bit key although the effective key strength is only 56 bits, as explained below. It takes a 64-bit block of plaintext as input and outputs a 64-bit block of cipher text. Since it always operates on blocks of equal size and it uses both permutations and substitutions in the algorithm, DES is both a block cipher and a product cipher. The basic building block of DES is a suitable combination of permutation and substitution on the plaintext block 16 times. Substitution is accomplished via table in S-boxes. Both encryption and decryption use the same algorithm except for processing the key schedule in the reverse order[1]

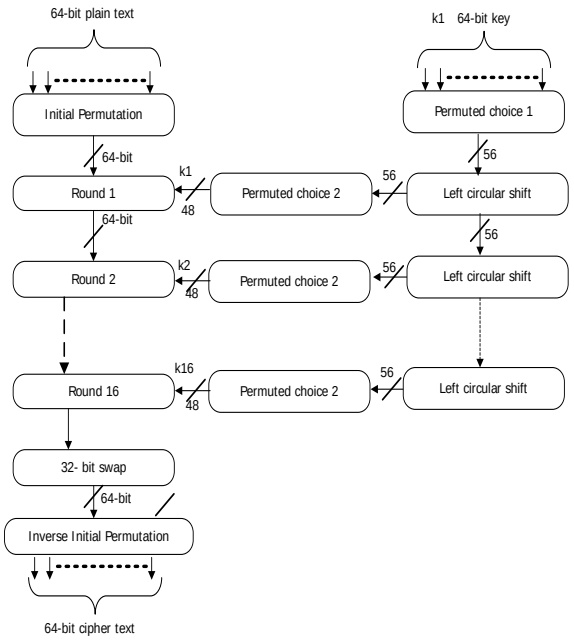

Figure 1. Flow of DES

3.2. Advanced Encryption Standard (AES)

In 1997, NIST initiated a very public, 4-1/2 year process to develop a new secure cryptosystem for U.S. government applications. The result, the Advanced Encryption Standard, became the official successor to DES in December 2001. AES uses an SKC scheme called Rijndael, a block cipher designed by Belgian cryptographers Joan Daemen and Vincent Rijmen. The algorithm can use a variable block length and key length ; the latest specification allowed any combination of keys lengths of 128,192, or 256 bits and blocks of length 128, 192, or 256 bits.

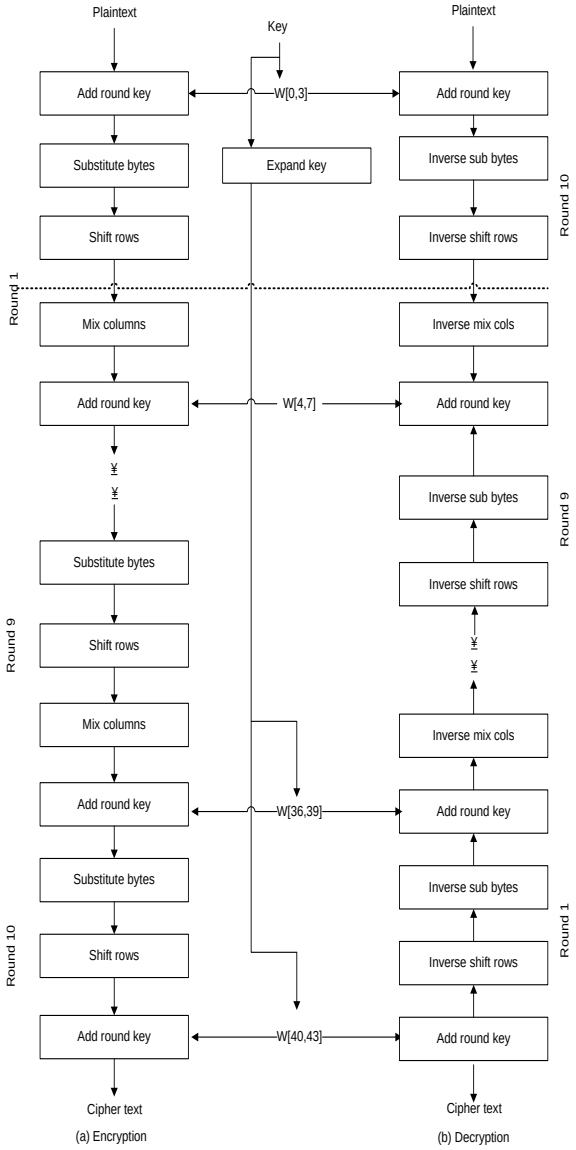

Figure 2. AES encryption and decryption

Figure 2 shows the overall structure of AES. The input to the encryption and decryption algorithms is a single 128-bit block. In FIPS PUB 197, this block is depicted as a square matrix of bytes. This block is copied into the State array, which is modified at each stage of encryption or decryption. After the final stage, State is copied to an output matrix. These operations are depicted in Figure 2a.

Similarly, the 128-bit key is depicted as a square matrix of bytes. This key is then expanded into an array of key schedule words; each word is four bytes and the total key schedule is 44 words for the 128-bit key Figure 2b. Note that the ordering of bytes within a matrix is by column. So, for example, the first four bytes of a 128-bit plaintext input to the encryption cipher occupy the first column of the in matrix, the second four bytes occupy the second column, and so on.

3.3. International data encryption (IDEA)

The IDEA is a block oriented encryption algorithm, which operates on a 64-bit plaintext and uses a 128 bit length key. The design philosophy of this algorithm is based on the concept "of mixing operations from different algebraic groups". The substitution boxes and the associated "lookup tables" used in the rest block ciphers available to-date have been completely dispensed with the required confusion in this algorithm. It is achieved by successively using three different and "incompatible group operations on pairs of 16-bit sub-blocks and mixing them while the structure of the cipher was carefully chosen to provide the necessary diffusion requirement". These three algebraic operations are the following:

. Bit-by bit XOR

. Addition of integers modulo 216 with inputs and outputs treated as unsigned 16-bit integers.

. Multiplication of integers modulo 216 with inputs and outputs treated as unsigned 16-bit integers. All these operations operate on 16-bit sub blocks. Their use in combination provides for a complex transformation of the input making cryptanalysis much more difficult than with an algorithm, which relies solely on the XOR function. Regarding the encryption/decryption procedures, the algorithm's structure has been chosen that, with the exception of the fact that different key sub blocks are used, the encryption process consists of eight identical (encryption) steps followed by an output transformation, while also the decryption process is identical to the encryption procedure once the decryption key sub blocks have been computed from the encryption ones [2].

3.4. RC4 symmetric algorithm

RC4 is a stream cipher, symmetric key algorithm. The same algorithm is used for both encryption and decryption as the data stream is simply XORed with the generated key sequence. The key stream is completely independent of the plaintext used. It uses a variable length key from 1 to 256 bit to initialize a 256-bit state table. The state table is used for subsequent generation of pseudo-random bits and then to generate a pseudo-random stream which is XORed with the plaintext to give the cipher text.

The algorithm can be broken into two stages: initialization, and operation..In the initialization stage the 256-bit state table, S is populated, using the key, K as a seed. Once the state table is setup, it continues to be modified in a regular pattern as data is encrypted. The initialization process can be summarized

```
j = 0;
for i = 0 to 255:
S[i] = i;
for i = 0 to 255:
j = (j + S[i] + K[i]) mod 256;
swap S[i] and S[j];
```

It is important to notice here the swapping of the locations of the numbers 0 to 255 (each of which occurs only once) in the state table. The values of the state table are provided. Once the initialization process is completed, the operation process may be summarized as shown by the pseudo code below;

```
 i = j = 0;
50
for (k = 0 to N-1) {
i = (i + 1) mod 256;
j = (j + S[i]) mod 256;
swap S[i] and S[j];
pr = S[ (S[i] + S[j]) mod 256]
output M[k] XOR pr
} [2]
```

4. Implementation of the system

This paper describes the comparison of encryption time and decryption time of four popular symmetric key algorithms. Figure 3 shows the system flow that includes two main areas: analyzing area and study area.

In analyzing area, it compares encryption and decryption time of various file sizes using DES, AES, IDEA and RC4 symmetric algorithms and then shows comparison results in table and in graph.

In study area, it can learn about four symmetric key algorithms, DES, AES, IDEA and RC4.

5. Experimental result

The experimental results of encryption and decryption time on various text files using DES, AES, IDEA and RC4 algorithms are shown in Table 1 ,Table 2 and Figure 4 and 5.

Table 1 shows the results of comparing encryption on each various file sizes using four methods. Table 2 shows decryption time for each

various file size. Figure 4 and Figure 5 show results by charts.

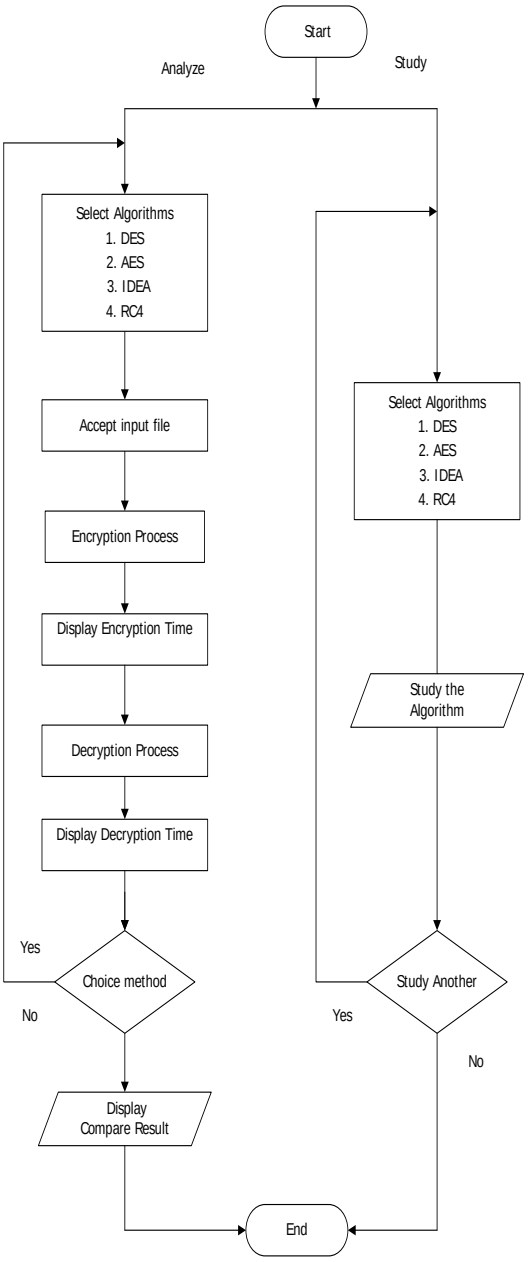

Figure 3. System flow diagram

Table 1. Encryption Time Comparison

Method	78.2kb	117.8kb	168kb	200kb	256kb
DES	5.6	11.9	25	45.4	60.3
AES	15.4	29.5	50.1	90.9	187.6
IDEA	6	12.4	26.	41.3	76.7
RC4	0.4	1	1.2	1.5	1.9

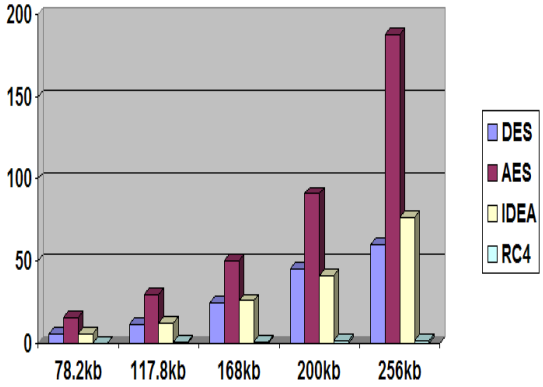

Figure 4. Encryption Time Comparison Chart

Table 2. Decryption Time Comparison

Method	78.2kb	117.8kb	168kb	200kb	256kb
DES	1	1.04	1.4	2.6	4.9
AES	1.7	2	2.2	4.1	7.9
IDEA	1.5	1.6	1.8	3.2	6.7
RC4	0.4	1	1.2	1.9.	4.2

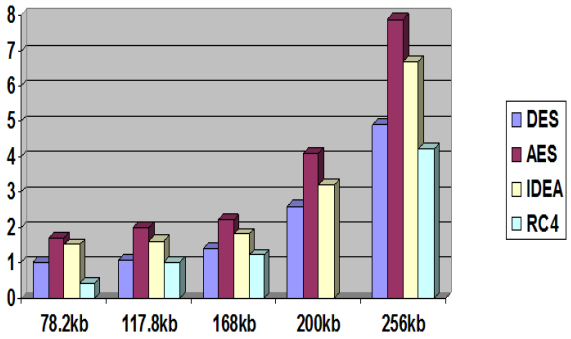

Figure 5. Decryption Time Comparison Chart

6. Conclusion

This paper has investigated the comparison among the most popular symmetric algorithms. Among several algorithms, this paper describes the comparison of encryption and decryption time of DES, AES, IDEA and RC4 methods. After making several testing and comparing four techniques, encryption time is distinctly different but decryption time is not very different.

In most of the time RC4 technique is faster than other techniques. DES can work in slower than RC4.DES and IDEA nearly equal in encryption and decryption time. Both these methods are faster than AES technique. Though AES consumes more time in encryption than other methods, decryption time is

not too different for various files with different sizes. This paper can give many advantages.

It can be used to learn cryptography and popular symmetric key algorithms, to learn comparing encryption and decryption time of these algorithms. It can be used for only text files format. So, it can be extended for audio file and image file formats and also compare for security point of view of four symmetric algorithms.

7. References

[1] William Stallings, *Cryptography and Network Security Principles and Practicle*, Fourth Edition

[2] Andreas Sterbenz, Peter Lipp, "Performance of the AES Candidate Algorithms in Java".

[3] Abdel-Karim Al Tamimi, "Performance Analysis of Data Encryption Algorithms".

[4] http://www.abo.fi/~ipetre/crypto/