ISSN 2250-3153

Analysis of comparison between Single Encryption (Advance Encryption Scheme (AES)) and Multicrypt Encryption Scheme

Vibha Verma, Mr. Avinash Dhole

Computer Science & Engineering, Raipur Institute of Technology, Raipur, Chhattisgarh, India
vbhvrm@gmail.com, avi_dhole33@rediffmail.com

***Abstract*- Advanced Encryption Standard** (**AES**) is a specification for the encryption of electronic data. It supersedes DES. The algorithm described by AES is a symmetric-key algorithm, meaning the same key is used for both encrypting and decrypting the data. **Multicrypt** is a scheme that allows for the encryption of any file or files. The scheme offers various private-key encryption algorithms, such as DES, 3DES, RC2, and AES. AES (Advanced Encryption Standard) is the strongest encryption and is used in various financial and public institutions where confidential, private information is important.

***Index Terms*-** AES, Multycrypt, Cryptography, Networking, Encryption, DES, 3DES, RC2

I. INTRODUCTION

THis article compares Single Encryption scheme and Multicrypt Encryption scheme. Security has become a significant requirement in today's multimedia communications, by using Encryption scheme we can secure our important documents like financial document, other account related document, office related document, corporate documents, new proposals etc.

In this paper for Single Encryption Scheme I have chosen AES (Advance Encryption Scheme) & for Multicrypt Encryption scheme Cryptogram, Caser Cipher, One-Time Pad and Faster RSA algorithm used one after another for encryption of the same document.

AES: *Advanced Encryption Standard* (AES) is a block cipher with block size of 128 bits, or 16 bytes. Keys for the cipher come in one of three lengths: 128, 192 or 256 bits, which are 16, 24 or 32 bytes. The algorithm is oriented toward bytes (8 bits), but there is also emphasis on what the AES specification calls *words*, which are arrays of 4 bytes.

RC2-Rivest Cipher: - In cryptography, **RC2** is a block cipher designed by Ron Rivest in 1987. "RC" stands for "Ron's Code" or "Rivest Cipher"; other ciphers designed by Rivest include RC4, RC5 and RC6.

The development of RC2 was sponsored by Lotus, who was seeking a custom cipher that, after evaluation by the NSA, could be exported as part of their Lotus Notes software. The NSA suggested a couple of changes, which Rivest incorporated. After further negotiations, the cipher was approved for export in 1989. Along with RC4, RC2 with a 40-bit key size was treated favourably under US export regulations for cryptography.
Initially, the details of the algorithm were kept secret — proprietary to RSA Security — but on 29th January, 1996, source code for RC2 was anonymously posted to the Internet on the Usenet forum, sci.crypt. Mentions of CodeView and SoftICE (popular debuggers) suggest that it had been reverse engineered. A similar disclosure had occurred earlier with RC4.
RC2 is a 64-bit block cipher with a variable size key. Its 18 rounds are arranged as a source-heavy Feistel network, with 16 rounds of one type (*MIXING*) punctuated by two rounds of another type (*MASHING*). A MIXING round consists of four applications of the MIX transformation, as shown in the diagram. RC2 is vulnerable to a related-key attack using 2^{34} chosen plaintexts.

DES-Data Encryption Standard: The **Data Encryption Standard** (**DES**) is a block cipher that uses shared secret encryption. It was selected by the National Bureau of Standards as an official Federal Information Processing Standard (FIPS) for the United States in 1976 and which has subsequently enjoyed widespread use internationally. It is based on a symmetric-key algorithm that uses a 56-bit key. The algorithm was initially controversial because of classified design elements, a relatively short key length, and suspicions about a National Security Agency (NSA) backdoor. DES consequently came under intense academic scrutiny which motivated the modern understanding of block ciphers and their cryptanalysis.

DES is now considered to be insecure for many applications. This is chiefly due to the 56-bit key size being too small; in January, 1999, distributed.net and the Electronic Frontier Foundation collaborated to publicly break a DES key in 22 hours and 15 minutes (see chronology). There are also some analytical results which demonstrate theoretical weaknesses in the cipher, although they are infeasible to mount in practice. The algorithm is believed to be practically secure in the form of Triple DES, although there are theoretical attacks. In recent years, the cipher has been superseded by the Advanced Encryption Standard (AES). Furthermore, DES has been withdrawn as a standard by the National Institute of Standards and Technology (formerly the National Bureau of Standards).

3DES -Triple DES: In cryptography, Triple DES is the common name for the Triple Data Encryption Algorithm (TDEA or Triple DEA) block cipher, which applies the Data Encryption Standard (DES) cipher algorithm three times to each data block.
The original DES cipher's key size of 56 bits was generally sufficient when that algorithm was designed, but the availability of increasing computational power made brute-force attacks feasible. Triple DES provides a relatively simple method of increasing the key size of DES to protect against such attacks, without the need to design a completely new block cipher algorithm.

II. SIMULATION SOFTWARE TOOLKIT (MULTICRYPT V1.0)

Multicrypt v1.0: Multicrypt v1.0 is a tool that allows for the encryption and decryption of the files in single or batch modes.

The Encryption algorithm used:
DES-Data Encryption Standard: DES is the archetypal block cipher — an algorithm that takes a fixed-length string of plaintext bits and transforms it through a series of complicated operations into another ciphertext bitstring of the same length. In the case of DES, the block size is 64 bits. DES also uses a key to customize the transformation, so that decryption can supposedly only be performed by those who know the particular key used to encrypt. The key ostensibly consists of 64 bits; however, only 56 of these are actually used by the algorithm. Eight bits are used solely for checking parity, and are thereafter discarded. Hence the effective key length is 56 bits, and it is never quoted as such. Every 8th bit of the selected key is discarded, that is, positions 8, 16, 24, 32, 40, 48, 56, 64 are removed from the 64 bit key leaving behind only the 56 bit key.

Like other block ciphers, DES by itself is not a secure means of encryption but must instead be used in a mode of operation. FIPS-81 specifies several modes for use with DES. Further comments on the usage of DES are contained in FIPS-74.
The algorithm's overall structure is shown in Figure 1: there are 16 identical stages of processing, termed *rounds*. There is also an initial and final permutation, termed *IP* and *FP*, which are inverses (IP "undoes" the action of FP, and vice versa). IP and FP have almost no cryptographic significance, but were apparently included in order to facilitate loading blocks in and out of mid-1970s hardware.

Before the main rounds, the block is divided into two 32-bit halves and processed alternately; this criss-crossing is known as the Feistel scheme. The Feistel structure ensures that decryption and encryption are very similar processes — the only difference is that the subkeys are applied in the reverse order when decrypting. The rest of the algorithm is identical. This greatly simplifies implementation, particularly in hardware, as there is no need for separate encryption and decryption algorithms.
The ⊕ symbol denotes the exclusive-OR (XOR) operation. The *F-function* scrambles half a block together with some of the key. The output from the F-function is then combined with the other half of the block, and the halves are swapped before the next round. After the final round, the halves are not swapped; this is a feature of the Feistel structure which makes encryption and decryption similar processes.

1. ***Expansion*** — the 32-bit half-block is expanded to 48 bits using the *expansion permutation*, denoted *E* in the diagram, by duplicating half of the bits. The output consists of eight 6-bit(8*6=48bits) pieces, each containing a copy of 4 corresponding input bits, plus a copy of the immediately adjacent bit from each of the input pieces to either side.
2. ***Key mixing*** — the result is combined with a *subkey* using an XOR operation. 16 48-bit subkeys — one for each round — are derived from the main key using the *key schedule* (described below).
3. ***Substitution*** — after mixing in the subkey, the block is divided into eight 6-bit pieces before processing by the *S-boxes*, or *substitution boxes*. Each of the eight S-boxes replaces its six input bits with four output bits according to a non-linear transformation, provided in the form of a lookup table. The S-boxes provide the core of the security of DES — without them, the cipher would be linear, and trivially breakable.
4. ***Permutation***s — finally, the 32 outputs from the S-boxes are rearranged according to a fixed permutation, the *P-box*. This is designed so that, after expansion, each S-box's output bits are spread across 6 different S boxes in the next round.

The alternation of substitution from the S-boxes, and permutation of bits from the P-box and E-expansion provides so-called "confusion and diffusion" respectively, a concept identified by Claude Shannon in the 1940s as a necessary condition for a secure yet practical cipher.

RC2-Rivest Cipher: **-** In cryptography, **RC2** is a symmetric-key block cipher. Designed by Ronald Rivest in 1987. "RC" stands for "Rivest Cipher", or alternatively, "Ron's Code".
RC2 is a 64-bit block cipher with a variable key size and using 18 rounds.
Rounds are arranged as a source-heavy feistel network, with 16 rounds of one type called *"mixing rounds"* interleaved by two rounds of another type called *"mashing rounds"*.
The 18 rounds are performed using the following interleaved sequence:

1. performe 5 mixing rounds.
2. performe 1 mashing round.
3. performe 6 mixing rounds.
4. performe 1 mashing round.
5. performe 5 mixing rounds.

RC2 uses ***key-expansion algorithm*** by which an expanded key consisting of 64 (16-bit words) is produces depending in a complicated way on every bit of the supplied *"variable-length"* input key. A mixing round consists of four applications of the *"mix-up"* transformation, as shown in the diagram. A round is *"mashed"* by adding to it one of the 16-bit words of the expanded key.

- **Encryption algorithm**
  1. Mix up R[i]
  2. Mixing round
  3. Mash R[i]

4. Mashing round
5. Encryption operation

- **Decryption algorithm**
  1. R-Mix up R[i]
  2. R-Mixing round
  3. R-Mash R[i]
  4. R-Mashing round
  5. Decryption operation

AES-Advanced Encryption Statndard (Rijndael):

1. KeyExpansion—round keys are derived from the cipher key using Rijndael's key schedule
   1. Initial Round-
      1. AddRoundKey—each byte of the state is combined with the round key using bitwise xor.
   2. Rounds -
      1. SubBytes—a non-linear substitution step where each byte is replaced with another according to a lookup table.
      2. ShiftRows—a transposition step where each row of the state is shifted cyclically a certain number of steps.
      3. MixColumns—a mixing operation which operates on the columns of the state, combining the four bytes in each column.
      4. AddRoundKey
   3. Final Round (no MixColumns)-
      1. SubBytes
      2. ShiftRows
      3. AddRoundKey

3DES-Triple DES: Triple DES uses a "key bundle" which comprises three DES keys, K_1, K_2 and K_3, each of 56 bits (excluding parity bits). The encryption algorithm is:
Ciphertext = E_{K3} (D_{K2} (E_{K1} (plaintext)))
I.e., DES encrypts with K_1, DES *decrypt* with K_2, then DES encrypt with K_3.

Decryption is the reverse:
Plaintext = D_{K1} (E_{K2} (D_{K3} (ciphertext)))
I.e., decrypt with K_3, *encrypt* with K_2 then decrypt with K_1.
Each triple encryption encrypts one block of 64 bits of data.
In each case the middle operation is the reverse of the first and last. This improves the strength of the algorithm when using keying option 2, and provides backward compatibility with DES with keying option 3.
Keying options:-The standards define three keying options:

- Keying option 1: All three keys are independent.
- Keying option 2: K_1 and K_2 are independent, and $K_3 = K_1$.
- Keying option 3: All three keys are identical, i.e. $K_1 = K_2 = K_3$.

Keying option 1 is the strongest, with $3 \times 56 = 168$ independent key bits.
Keying option 2 provides less security, with $2 \times 56 = 112$ key bits. This option is stronger than simply DES encrypting twice, e.g. with K_1 and K_2, because it protects against meet-in-the-middle attacks.
Keying option 3 is equivalent to DES, with only 56 key bits. This option provides backward compatibility with DES, because the first and second DES operations cancel out. It is no longer recommended by the National Institute of Standards and Technology (NIST), and is not supported by ISO/IEC 18033-3.

Multicrypt v1.0 Options:
Single File Processing: In this option we can select a file for encryption and decryption by using browsing option.
Drag & Drop File Processing: In this option we can drag a drop a file to encrypt or decrypt.
Remove Original Files After Encryption: By using this option we can remove original version of encrypted file.

Security:
Password (Key): In cryptography, a **key** is a piece of information that determines the functional output of a cryptographic algorithm or cipher. Without a key, the algorithm would produce no useful result. In encryption, a key specifies the particular transformation of plaintext into ciphertext, or vice versa during decryption. Keys are also used in other cryptographic algorithms, such as digital signature schemes and message authentication codes.

Initial Vector (IV):
In cryptography, an **initialization vector** (**IV**) is a fixed-size input to a cryptographic primitive that is typically required to be random or pseudorandom. Randomization is crucial for encryption schemes to achieve semantic security, a property whereby repeated usage of the scheme under the same key does not allow an attacker to infer relationships between segments of the encrypted message. For block ciphers, the use of an IV is described by so-called modes of operation. Randomization is also required for other primitives, such as universal hash functions and message authentication codes based thereon.
Some cryptographic primitives require the IV only to be non-repeating, and the required randomness is derived internally. In this case, the IV is commonly called a nonce (*number used once*), and the primitives are described as stateful as opposed to *randomized*. This is because the IV need not be explicitly forwarded to a recipient but may be derived from a common state updated at both sender and receiver side. (In practice, a short nonce is still transmitted along with the message to consider message loss.) An example of stateful encryption schemes is the counter mode of operation, which uses a sequence number as a nonce.

The size of the IV is dependent on the cryptographic primitive used; for block ciphers, it is generally the cipher's block size. Ideally, for encryption schemes, the unpredictable part of the IV has the same size as the key to compensate time-memory-data trade-off attacks. When the IV is chosen at random, the probability of collisions due to the birthday problem must be taken into account. Traditional stream ciphers such as RC4 do not support an explicit IV as input, and a custom solution for incorporating an IV into the cipher's key or internal state is needed. Some designs realized in practice are known to be insecure; the WEP protocol is a notable example, and is prone to related-IV attacks.

III. PROCESS AND IDEA

This paper compare two algorithms, namely single encryption and multicrypt (Combination of more than 1 encryption algorithm) encryption and analyze the results in form of advantages and disadvantages. Idea behind this paper is to find how multiple time encryptions are better than single time encryption and vice-versa.

Choose AES for single type encryption and RC2, DES, AES & 3DES for Multicrypt encryption scheme. In single type encryption take a document and use one encryption algorithm to encrypt the same while Multicrypt uses two or more algorithm for encryption of the same document.

Multicrypt is more secure but it is a time taking process and also we have to memorize so many password/Keys and initial vectors while single encrypt is less secure but a fast process.

IV. STUDIES AND FINDINGS

Table I: Comparison between Single & Multicrypt Encryption Scheme

S.n.	Component	Single Encryption scheme	Multicrypt Encryption scheme
1.	Computational Overhead/ Time complexity	It takes less time compares to Multicrypt	It takes more time cause of more than one encryption.
2.	Encryption Used	AES	RC2, DES, 3DES, AES
3.	Password and Initial Vector memorization complexity	Easy to memorize one password and IV	Difficult to memorize more than one passwords and IVs
4.	Security	Less Secure	More Secure
5.	Breakable	Easy to break	Difficult to break
6.	Attacks	Known attack, side channel attacks	Anyway it is breakable when we use one encryption method but when we use combination of any two or more than two then it is difficult to compromise
7.	Accessibility	Not Accessible after encryption	Not Accessible after encryption

V. CONCLUSION

After the analysis of comparison of both type of encryption scheme multicrypt is much better than single type encryption because it is very secure cause of many layers (level) of security and authentication (each encryption algorithm contain one password and initial vector for security and authentication purpose). For security of personnel document, single type encryption is enough while for corporate and business purpose multicrypt is better option to secure important files and documents.

ACKNOWLEDGMENT

The first and the foremost person who comes into my mind to express our deep sense of gratitude whole heartily is my guide **Mr. Avinash Dhole** Senior Lecturer. He was there to help me out through the thick and thin of this project. I express my indebtedness to him for the constant encouragement given throughout the project work.

I also owe a debt of gratitude to Mrs. Uzma Ansari (HOD, CSE, RITEE) for providing us with an opportunity to develop this project. Through her timely advice, constructive criticism and supervision she was a real source of inspiration for me.

I'm working in Mastek Ltd. Global co., Mumbai as a Software engineer, and I also thankful to **Mr. Dhananjay S. Sakpal** (Project Manager Mastek Ltd.), **Mr. Rahul D. Lahane** (Account manager, Mastek Ltd.) & **Mr. Sohan K. Rathor** (Business Associates, Mastek Ltd.) for support & encouragement.

At the last but not the least I am really thankful to my **parents** for always encouraging me to achieve this goal.

REFERENCES

[1] A.John Prakash and V.Rhymend Uthariaraj & Anna University Chennai, India "Multicrypt: A Provably Secure Encryption Scheme for Multicast Communication" 2009 First International Conference on Networks & Communications.

[2] Neal R. Wagner, The Laws of Cryptography with Java Code. 2003, Ch VI (114-138, 273-297)

[3] Announcing the ADVANCED ENCRYPTION STANDARD (AES), November 26, 2001, Federal Information Processing Standards Publication 197

[4] Gradeway.com - MultiCrypt (Simulation software Multicrypt v1.0)

[5] www.wekipedia.com.

AUTHORS

First Author – Vibha Verma, M.Tech., RITEE Raipur (Chhattisgarh, India), Software Engineer, Mastek Ltd. vbhvrm@gmail.com

Second Author – Mr. Avinash Dhole, Sr. Lecturer, RITEE Raipur (Chhattisgarh, India), avi_dhole33@rediffmail.com