IMPACT: International Journal of Research in Engineering & Technology (IMPACT: IJRET)
ISSN(E): 2321-8843; ISSN(P): 2347-4599
Vol. 2, Issue 5, May 2014, 7-14

A SURVEY ON SECRET KEY ENCRYPTION TECHNIQUE

NIKITA[1] & RANJEET KAUR[2]

[1]Research Scholar, Department of Computer Science and Engineering, DAV University, Jalandhar, Punjab, India

[2]Assistant Professor, Department of Computer Science and Engineering, DAV University, Jalandhar, Punjab, India

ABSTRACT

Cryptography renders the message unintelligible to outsider by various transformations. Data Cryptography is the scrambling of the content of data like text, image, audio and video to make it unreadable or unintelligible during transmission. As the data may include some sensitive information which should not be accessed by or can only be partially exposed to the general users. The principal goal guiding the design of any encryption algorithm must be security against unauthorized attacks but performance and the cost of implementation are also important concerns. This paper provides the comparison between the three popular secret key encryption techniques, i.e., DES, AES and the Blowfish with modes of operation. The comparison has been conducted by calculating the avalanche effect of these encryption techniques and compares them on the basis of their result.

KEYWORDS: AES, Blowfish, Cryptography, DES, Encryption, IV (Initialization Vector)

INTRODUCTION

Cryptography is where security engineering meets mathematics. It provides us with the tools that underlie most modern security protocols. It is probably the key enabling technology for protecting distributed systems, yet it is surprisingly hard to do right. Cryptography [1] has often been used to protect the wrong things, or used to protect them in the wrong way. It is the art or science encompassing the principles and methods of transforming an intelligible message into one that is unintelligible and then retransforming that message back to its original form. It is not only use by spies but for phone, fax and e-mail communication, bank transactions, bank account security, PINs, passwords and credit card transactions on the web. It is also used for a variety of other information security issues including electronic signatures, which are used to prove who sent a message. In cryptography original data is transformed (encrypted) into an unreadable format, called cipher text. Only those who possess a secret key can decipher (or decrypt) the message into plaintext. Cryptography comes from Greek words meaning “hidden writing”. Cryptography converts readable data or cleartext into encoded data called ciphertext.

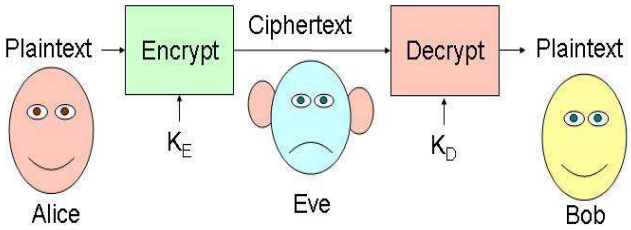

Figure 1: Cryptography

Cryptography was done to attain CIA.

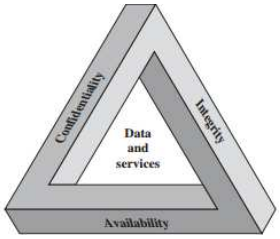

Figure 2: The Security Requirements Triad

- **Confidentiality:** Confidentiality refers to preventing the disclosure of information to unauthorized individuals or systems.
- **Integrity:** Data integrity means maintaining and assuring the accuracy and consistency of data over its entire life-cycle. This means that data cannot be modified in an unauthorized or undetected manner.
- **Availability:** The information must be available when it is needed. This means that the computing systems used to store and process the information, the security controls used to protect it, and the communication channels used to access it must be functioning correctly.

SYMMETRIC AND ASYMMETRIC ENCRYPTION

Symmetric Encryption

Symmetric encryption is the oldest and best-known technique. It is a class of algorithms for cryptography that use the same cryptographic keys for both encryption of plaintext and decryption of ciphertext. A secret key, which can be a number, a word, or just a string of random letters, is applied to the text of a message to change the content in a particular way.

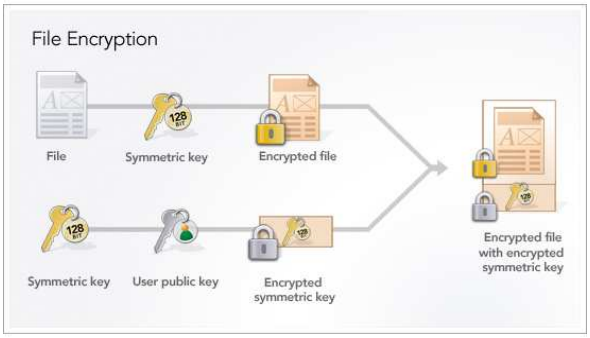

Figure 3: Symmetric Encryption

Asymmetric Encryption

The problem with secret keys is exchanging them over a large network while preventing them from falling into the wrong hands. Anyone who knows the secret key can decrypt the message. So we can use asymmetric encryption, in which there are two keys- public and private keys. A public key is made freely available to anyone who might want to send you a message. A second, private key is kept secret, so that only you know it. Any message (text, binary files, or documents) that are encrypted by using the public key can only be decrypted by applying the private key. Any message that is encrypted by using the private key can only be decrypted by using the matching public key. This means that you do not have to worry about passing public keys over the Internet.

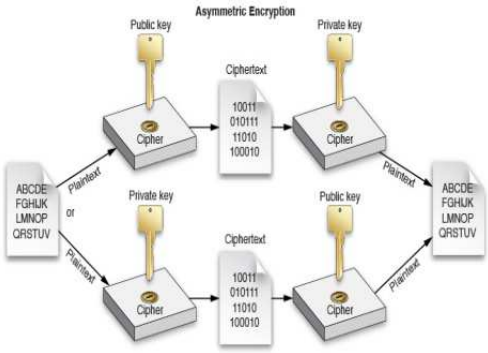

Figure 4: Asymmetric Encryption

SYMMETRIC CRYPTOGRAPHY TECHNIQUES

Data Encryption Standard (DES)

On 15 May 1973, the National Bureau of Standards, now called NIST (National Institute of Standards and Technology) published a request in the Federal Register for an encryption algorithm. In late 1974, IBM proposed "Lucifer", which was modified on 23 November 1976 to become the DES. The DES was approved by the NBS in 1978. The DES was standardized by the ANSI (American National Standard Institute) under the name of ANSI X3.92, better known as DEA (Data Encryption Algorithm). DES is a 64 bit block cipher which means that it encrypts data 64 bits at a time. This is contrasted to a stream cipher in which only one bit at a time is encrypted. DES is based on a cipher known as the Feistel block cipher. DES expects two inputs - the plaintext to be encrypted and the secret key. The manner in which the plaintext is accepted, and the key arrangement used for encryption and decryption, both determine the type of cipher it is. The key size used is 56 bits. Once a plain-text message is received to be encrypted, it is arranged into 64 bit blocks required for input. If the number of bits in the message is not evenly divisible by 64, then the last block will be padded. It consists of a number of rounds where each round contains bit-shuffling, non- linear substitutions (S-boxes) and exclusive OR operations. Decryption uses the same algorithm as encryption, except that the application of the sub keys is reversed.

Advance Encryption Standard (AES)

AES is based on the Rijndael cipher developed by Joan Daemen and Vincent Rijmen. Rijndael is a family of ciphers with different key and block sizes. It uses block size of 128 bits, but three different key lengths: 128, 192 and 256 bits. AES [2] has been adopted by the U.S. government and is now used worldwide. It replaces the Data Encryption Standard. AES was announced by the NIST on November 26, 2001 and became effective as a federal government standard on May 26, 2002. Unlike its predecessor DES, AES does not use a Feistel network. The cipher consists of rounds, where the number of rounds depends on the key length: 10 rounds for a 16-byte key, 12 rounds for a 24-byte key, and 14 rounds for a 32-byte key. The first rounds consist of four distinct transformation functions: Sub Bytes, Shift Rows, Mix Columns, and Add Round Key. The final round contains only three transformations. The third function i.e. Mix Columns is missing. Each transformation takes one or more 4x4 matrices as input and produces a 4x4 matrix as output. Once it is established that all four stages are reversible, it is easy to verify that decryption does recover the plaintext.

Blowfish

Blowfish is a symmetric block cipher that can be effectively used for encryption and safeguarding of data. It takes a variable-length key, from 32 bits to 448 bits, making it ideal for securing data. Blowfish was designed in 1993 by Bruce

Schneier as a fast, free alternative to existing encryption algorithms. Blowfish is unpatented and license-free, and is available free for all uses. Blowfish has a 64-bit block size. It is a 16-round Feistel cipher and uses large key-dependent S-boxes. It is suitable for applications where the key does not change often, like a communications link or an automatic file encryptor. The Blowfish algorithm has many advantages. It is suitable and efficient for hardware implementation. The algorithm consists of two parts: a key-expansion part and a data- encryption part. Key expansion converts a key of at most 448 bits into several sub key arrays totaling 4168 bytes. Data encryption occurs via a 16-round Feistel network. Each round consists of a key dependent permutation, and a key- and data-dependent substitution. The elementary operators of Blowfish algorithm include table lookup, addition and XOR.

MODES OF ENCRYPTION/DECRYPTION

Electronic Code Book (ECB)

In electronic code book (ECB) [3], we just encrypt each succeeding block of plaintext with our block cipher to get ciphertext. There is no interdependency between blocks. In this mode data is divided into 64-bit blocks. This mode is deterministic as identical plaintexts are encrypted similarly. There is no chaining and error propagation. Using ECB mode to encrypt messages of more than one block length and that have an authenticity requirement—such as bank payment messages—would be foolish. A big advantage of this mode is that you can encrypt or decrypt multiple blocks in parallel but we have to make sure that all blocks will be placed in correct order. The bit errors caused by noisy channels only affect the corresponding block but not succeeding blocks. As blocks can be reordered, it is its disadvantage as reordering or repetition of blocks can change the message.

Cipher Block Chaining (CBC)

In this mode we exclusive-or the previous block of ciphertext to the current block of plaintext before encryption. The same key is used for each block. The encryption of all blocks is “chained together” ciphertext Ci depends not only on block Xi but on all previous plaintext blocks as well. CBC mode was mainly designed to overcome the security deficiency of ECB mode. In this the same plaintext block, if repeated, produces different ciphertext blocks. For decryption, each cipher block is passed through the decryption algorithm. The result is XO Red with the preceding ciphertext block to produce the plaintext block. The input IV is an initialization vector, a random number is XO Red with the first block of plaintext. IV provides the semantic security. The IV must be known to both the sender and receiver but is unpredictable by a third party. This is most commonly used mode of operation.

Cipher Feedback (CFB)

CFB mode is a kind of stream cipher. It is possible to convert block cipher into stream cipher by using stream cipher modes. In this case, rather than blocks of bits, the plaintext is divided into segments of bits. IV is used in this mode as well as an input to the encryption function. As a result of the use of an IV, the CFB encryption is also nondeterministic. In CFB mode [4], the previous ciphertext block is encrypted and the output is XO Red with the current plaintext block to create the current ciphertext block. CFB is primarily a mode to derive some characteristics of a stream cipher from a block cipher. In common with CBC mode, changing the IV to the same plaintext block results in different output. In this the message does not need to be padded to a multiple of the cipher block size. For decryption, the same scheme is used, except that the received ciphertext unit is XO Red with the output of the encryption function to produce the plaintext unit. Cipher feedback is not used much anymore. It is a specialized mode of operation for applications such as military HF radio links.

Output Feedback (OFB)

It is a stream cipher mode that can encrypt arbitrary blocks of data. OFB encrypts plaintext a full block at a time, where typically a block is 64 or 128 bits. Many stream ciphers encrypt one byte at a time. IV is the initial cipher input. Output of cipher is the key stream, and is XO Red with the plaintext to create the ciphertext. Prior key stream becomes the next IV. Key stream is in no way affected by the plaintext. In this mode each bit in the ciphertext is independent of the previous bit or bits. This avoids error propagation. The disadvantage of OFB is that it is more vulnerable to a message stream modification attack than is CFB.

AVALANCHE EFFECT

Avalanche effect refers to a property of cryptography encryption algorithms. The small change in either the plaintext or the key should produce a significant change in the ciphertext. In the case of high-quality block ciphers, a small change in either the key or the plaintext should cause a drastic change in the ciphertext. If cryptographic functions do not exhibit the avalanche effect, then it has poor randomization and our input data is easily predicted only from the output. Thus, avalanche effect is desirable condition.

COMPARISON BASED ON AVALANCHE EFFECT

Now we compare the symmetric encryption techniques explained above i.e. DES, AES and BLOWFISH on the basis of avalanche effect with the same key and plaintexts with the difference of one alphabet.

Plaintext1: NETWORKS

Plaintext2: NETVORKS

Key: EXAMPLES

DES

IV: 10011001001101000101110011100110100000001100110100111110010010 00

ECB Mode

- **Ciphertext1:** 11011101010010111100000000011111100010001010011011011100010100 00
- **Ciphertext2:** 10100101010000000110000010000101001011100111100000011111101001 01

The change after comparing two ciphertexts by changing a single character ‘W’ to ‘V’ the change in avalanche effect was of 33-bits.

CBC Mode

- **Ciphertext1:** 01110111011010101011001111110011110110000010110101110011111000110
- **Ciphertext2:** 10100000010111010011101000001011111101001111101100011010111010 10

We compared the two ciphertexts to calculate the difference and found out that there was a change in 34-bits.

CFB Mode

- **Ciphertext1:** 10100010111111101110111011001101100001010011110111110110000000001

- **Ciphertext2:** 1010001011111110111011101100110001100001000110100000010110110101

  20 bits of difference was noted when one character was changed.

OFB Mode

- **Ciphertext1:** 1010001001010001100111110101001001011111001000001110100111000111
- **Ciphertext2:** 1010001001010001100111110101001101011111001000001110100111000111

  We compared the two ciphertexts to calculate the difference and found out that there was a change in 1-bit.

AES

IV: 10011001001101000101110011100110100000001100110100111110010010001010110011011011100110101011010000100001001011100100101111011001

ECB Mode

- **Ciphertext1:** 10110001100101111111000001010000111100110000100000110101011100100101000111101100100110100001100010011100001100101001111101010111 1
- **Ciphertext2:** 00101101111011011010111001000110111010000100000000000110000111011011100110110111011011011110111001001110011001101001000111101100

  We compared the two ciphertexts to calculate the difference and found out that there was a change in 69-bits.

CBC Mode

- **Ciphertext1:** 10110010110010101010011101100011111100001011001110010110101010010101100101011111110001000000101010011100110000010000111111011100 1
- **Ciphertext2:** 01010010100010000110111101110110101101100100011000110100010101001110110010011101101111001100100110011100000111110000100000010010

  66-bits of difference was noted when one character was changed.

CFB Mode

- **Ciphertext1:** 0101100111000011011001001011010000101011110100010111010011011010
- **Ciphertext2:** 0101100111000011011001001011010101011010011011011011110010000010

  16-bits of difference was noted when one character was changed.

OFB Mode

- **Ciphertext1:** 0101100100110101110100001001010001101100000000111101000001100001
- **Ciphertext2:** 0101100100110101110100001001010101101100000000111101000001100001

  We compared the two ciphertexts to calculate the difference and found out that there was a change in 1-bit.

Blowfish

IV: 10011001001101000101110011100110100000001100110100111110010010 00

ECB Mode

- **Ciphertext1:** 011101000101010100001100001110111111000101111010000011100101011 1
- **Ciphertext2:** 001001000010101010110101100100010011111000011100100011100110000 0

We compared the two ciphertexts to calculate the difference and found out that there was a change in 34-bits.

CBC Mode

- **Ciphertext1:** 100110110000001110101000100011111011100110001111100010011000000 0
- **Ciphertext2:** 100010111111110000011011110110111101011110010001100100011001100 1

31-bits of difference was noted when one character was changed.

CFB Mode

- **Ciphertext1:** 101000110110011001010011001000111001000111010010110100011100000 1
- **Ciphertext2:** 101000110110011001010011001000100111101001100100010010111110110 0

20-bits of difference was noted when one character was changed.

OFB Mode

- **Ciphertext1:** 101000110111001001101100101100011011000011111100101111001110001 0
- **Ciphertext2:** 101000110111001001101100101100001011000011111100101111001110001 0

We compared the two ciphertexts to calculate the difference and found out that there was a change in 1-bit.

RESULTS

The table 1 indicates the effect of avalanche effect in various techniques. So, from above result we find that AES with ECB mode has maximum deviation of bits therefore it is best of all other techniques and mode with avalanche effect. This table clearly shows the comparison between different techniques. Besides avalanche effect there are more factors which describe the performance of these techniques. According to [4], Blowfish has better performance than all other on the basis of computation time. According to [5], it is clear that the key size of blowfish algorithm is high and that of DES is lesser. Hence it can be said that security of Blowfish is far better than the other algorithms. Also DES and other algorithms are vulnerable to possible attacks but Blowfish algorithm has not been cracked till date.

Table 1

S. No	Encryption Techniques	ECB		CBC		CFB		OFB	
		Avalanche Effect (Bits)	%	Avalanche Effect (Bits)	%	Avalanche Effect (Bits)	%	Avalanche Effect (Bits)	%
1	DES	33	51.56	34	53.12	20	31.25	1	1.56
2	AES	69	53.91	66	51.56	16	25	1	1.56
3	BLOWFISH	34	53.13	31	48.44	20	31.25	1	1.56

CONCLUSIONS AND FUTURE SCOPE

In this paper the techniques of encryption including DES, AES and BLOWFISH was compared by calculating their avalanche effect with respect to each mode of operation. The two plaintexts were taken with the difference of 1 word encrypted with same key. In the end, the results were concluded which present that AES with ECB mode has maximum change in the bits of two ciphertext by changing one word in plaintext. OFB mode with all techniques showed the poor performance compared to all other modes. A proposed direction for the future work could be to measure the performance by calculating the effect of all other factors on which algorithms depend.

REFERENCES

1. W. Stallings, *cryptography and network security: principles and practices*, 5th ed., Prentice Hall.
2. Jawahar Thakur, NageshKumar, "DES, AES and Blowfish: Symmetric Key Cryptography Algorithms Simulation Based Performance Analysis" International *Journal of Emerging Technology and Advanced Engineering ISSN 2250-2459, Volume 1, Issue 2*, December 2011.
3. A. Nadeem, "A performance comparison of data encryption algorithms," *IEEE Information and Communication Technologies*, pp. 84-89, 2006.
4. Diaa Salama Abd Elminaam, Hatem Mohamed Abdual Kader and Mohiy Mohamed Hadhoud "Evaluating The Performance of Symmetric Encryption Algorithms" *International Journal of Network Security, Vol.10, No.3,* PP.216–222, May 2010.
5. Monika Agarwal, Pradeep Mishra "A Comparative Survey on Symmetric Key Encryption Techniques" *International Journal on Computer Science and Engineering (IJCSE) Vol. 4 No.* 05 May 2012.
6. Hamdan. O. Alanazi, B.B. Zaidan, A.A. Zaidan, Hamid A. Jalab, M. Shabbir and Y. Al-Nabhani "New Comparative Study Between DES, 3DES and AES within Nine Factors" *Journal Of Computing, Volume 2, Issue 3*, March 2010, ISSN 2151-9617.
7. Mohit Kumar, Reena Mishra, Rakesh Kumar Pandey and Poonam Singh "Comparing Classical Encryption With Modern Techniques" *S-JPSET, Vol. 1, Issue 1*, 2010.
8. O P Verma, Ritu Agarwal, Dhiraj Dafouti, Shobha Tyagi "Peformance Analysis of Data Encryption Algorithms" *IEEE*, 2011.
9. Singh, S Preet and Maini, Raman. "Comparison of Data Encryption Algorithms", *International Journal of Computer Science and Communication, vol. 2, No. 1*, January-June 2011, pp. 125-127.
10. Sriram Ramanujam and Marimuthu Karuppiah "Designing an algorithm with high Avalanche Effect" *IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.1*, January 2011.