INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS
ISSN 2320-7345

CRITICAL REVIEW OF CRYPTOGRAPHIC TECHNIQUES

Sathyalakshmi.L[1], A.Mohanarathinam[2],V.S.Jayanthi[3]

[1]PG scholar, Department of Electronics and Communication Engineering, HICET, Coimbatore
[2]Assistant Professor, Department of Electronics and Communication Engineering, HICET, Coimbatore
[3] Professor and Head, Department of Electronics and Communication Engineering, HICET, Coimbatore

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
HINDUSTHAN COLLEGE OF ENGINEERING AND TECHNOLOGY
COIMBATORE, TAMILNADU, INDIA

Abstract

Large amounts of data are being communicated over different channels - both public and private. The volume is getting increased exponentially and continuously day by day. Though more and more ways of protecting the data are also underway, this creates a problem of Security to those concerned with the data - particularly the sender, and receiver. This problem is further aggravated with the advent of high speed computing, parallel technologies, etc. Selection of the right technique for encryption and decryption of the data communicated, with the suitable algorithm, keys etc. deserve a critical review. This paper therefore studies the most commonly used techniques of cryptographic for the purpose of right selection of the technique.

Keywords*:* Security, Sender, Receiver, Encryption, Decryption, Cryptography.

1. Introduction

Cryptography is the modern science of keeping secrecy. In the earlier days it was known as coding and decoding or encryption / decryption. A readable message is converted using some mathematical technique in to an apparently unreadable / non-sense message before sending to the receiver. The receiver gets the same and converts in to the original form and makes sense out of it. In this process, in the presence of third parties (called adversaries), both sender and receiver use some keys - namely public and private keys respectively. If the key is not known the encrypted data cannot be used. Hence the data is made secured, reliable etc in communication. More generally, it is all about constructing and analyzing protocols that overcome the influence of adversaries and that are related to various aspects such as data confidentiality, data integrity, authentication, and non-repudiation. Modern cryptography is a multidisciplinary subject that includes mathematics, computer science, and electrical engineering. Applications of cryptography include ATM cards, computer passwords, and electronic commerce.

The message to be communicated in its original form is called “Plain Text”. The transmitter of a secure system will encrypt the “Plain Text” to make it meaningless / non-sense. A reversible mathematical process will convert the original message by encrypting it. The output is called “cipher-text”. The algorithm used to encrypt the message is known as "cipher". The original message will be revealed only after the correct recipient tries to access it. The unauthenticated user can also try to access the information. The analysis is carried out to check if cipher's security is satisfactory from unauthorised access. Cryptanalysis is the science of breaking ciphers, and cryptanalysts try to protect the security of cryptographic systems. A "cipher-text" can be transmitted openly across a communications channel. Because of its encrypted nature, eavesdroppers who may have access to the "cipher-text" will ideally be unable to decipher the message. Only the intended recipient, who has the valid key, can decrypt the message to recover the "Plain Text" and interpret.

"Ciphers" are classified by some criteria as follows. According to one criterion, the ciphers are classified as symmetric key and asymmetric key. In symmetric key ciphers, the same key is used for both encryption and decryption. A major problem with such a system is that the sender and receiver must know the key prior to transmission. This requirement makes the system difficult to use in practice. The key cannot be openly transmitted as its security is very weak. It is possible that the two parties may meet and exchange the keys prior to transmitting their messages. However, this exchange becomes more difficult when many parties are involved in a communications network. An asymmetric key cipher uses different keys for encryption and decryption. Though these two keys are mathematically related, it is very difficult to obtain one from the other unless how they have been determined by some algorithm. The key used for encryption is called the public key and the key used for decryption is called the private key. The public key can be revealed without compromising the security of the system. The corresponding private key, however, must not be revealed to any party.

Now all information are electronically processed and communicated through public networks. The main objective of cryptography is, to conceal the content of messages transmitted through insecure channels so that it guarantees reliability, privacy and confidentiality in the communications to the authorized users. Since 1960, cryptography has no longer been restricted to military or governmental services. It has spurred an unprecedented development of cryptographic technique since then. The advent of digital communication technology got the benefits of cryptography. Many efficient encryption schemes were designed. Mere encryption and decryption of the message cannot ensure overall security requirements as the word *security* itself relies on confidentially, integrity (authenticity, non-repudiation) and availability.

1. Confidentiality relates to secrecy and privacy that means message should be visible only to those persons for whom it has been sent.

2. Integrity can be further classified into two terms: (1) authenticity – that means identity of the sender should be verified on delivering the message to check whether the information comes from the authentic sender, or from whom we are expecting. (2) Nonrepudiation – it means message should not be falsely modified with any kind of fake addition or deletion.

3. Availability means information (message, key, Certificate Verification) and medium (Certification Authority Server, online services) should be available on time when needed.

These security objectives give births to key exchange methods (Diffie, Hellman, digital signature) and the asymmetric encryption that involves third trust party and the use of two key(s). Mostly users demand secure communication especially in case organizational linkage, Governmental communication and banking transactions. Cryptographic algorithms are reliable marvels in these situations. Some cryptographic algorithms are symmetric and some are asymmetric in nature.

In Symmetric cryptography, algorithms are either block cipher or stream cipher in block cipher "plain text" is converted into blocks (64 or 128 bits) and arithmetic operations (XOR, NOT, OR and etc.) are implemented on block level in such a way that each block is encrypted separately. Each block can be encrypted either with an operation and the same operation may be repeated for any other next block but in stream cipher whole plain text is considered as single block and its every bit or byte (bit stream) relatively very small than block size is encrypted with different operation (may be repeated for any other bit or byte).

The use of symmetric algorithms require that both parties have to share and agree on same secret or private key before starting encryption procedure but in case of asymmetric algorithms public key is publically available to start encryption any time when needed before asking the other party and private key remains secret in both sides. Symmetric key length is shorter than asymmetric key length. This study does not concern with the issues of cryptographic algorithms but it mostly concerns with the cryptographic primitives to point out the latest trends, research issues and future necessities.

2. Cryptography Algorithms

2.1 Rivest Shamir And Adelmann (RSA)

The RSA algorithm may be employed to provide a simplistic form of secure key exchange. If Alice wishes to secure some large quantity of data with a fast algorithm such as DES before transmitting the data to Bob, she first chooses some random 56-bit number as the DES key and encrypts it using Bob's RSA Public Key. Only Bob will be able to decrypt this exchange using his private RSA key. The drawback with this approach is that anybody, including the cracker Trudy, can encrypt anything using Bob's Public Key. Bob therefore has no proof that it is indeed Alice with whom he is communicating. The communications channel is only secure if Alice digitally signs the DES key and encrypts both the key and her signature with Bob's Public Key. The problem with this approach is that the signature can be too big to secure in a single RSA operation[4].

2.2 Diffie-Hellman Exchange

The Diffie-Hellman key exchange was the first Public Key cryptosystem and it underpins the entire framework by that IP packets may be securely transmitted over the Internet. The participants in the exchange must first agree upon a *group* that defines the prime *p and* generator *g that* should be used. In the first part of the exchange, Alice *A* and Bob *B* each select a random private number (indicated by the lowercase initial of each party) and exponentiate to produce a corresponding public value (uppercase initial of the party). The significant property of this exchange is that the public values *A* and *B* can be exchanged over an insecure public network without reducing the security of the exchange. An eavesdropper (conventionally known as Eve) could know *g* and *p* and intercept the exchange of public values and still not be able to discover the key because one of the private values must be known to generate the shared secret [1].

The Diffie-Hellman exchange is vulnerable to a man-in-the-middle attack in that Trudy impersonates Bob to Alice and Alice to Bob. Both Alice and Bob believe they are performing a key exchange with one another, but in reality are doing so with Trudy. When Alice sends secured data to Bob, Trudy can intercept the traffic and decrypt it before passing the packets on to Bob. Neither Alice nor Bob would notice anything out of the ordinary. This type of attack may be thwarted if Alice and Bob both digitally sign their public values.

2.3 Data Encryption Standard (DES)

The Data Encryption Standard (DES) is a symmetric- key block cipher published as FIPS-46 in the Federal Register in January 1977 by the National Institute of Standards and Technology (NIST). At the encryption site, DES takes a 64-bit "Plain Text" and creates a 64-bit ciphertext, at the decryption site, it takes a 64-bit ciphertext and creates a 64-bit "Plain Text", and same 56 bit cipher key is used for both encryption and decryption[3]. The encryption process is made of two permutations (P-boxes), that we call initial and final permutation, and sixteen Feistel rounds. Each round uses a different 48-bit round key generated from the cipher key.

2.4 Advanced Encryption Standard (AES)

Advanced Encryption Standard is a symmetric- key block cipher published as FIPS-197 in the Federal Register in December 2001 by the National Institute of Standards and Technology (NIST). AES is a non-Feistel cipher. AES encrypts data with block size of 128-bits. It uses 10, 12, or fourteen rounds. Depending on the number of rounds, the key size may be 128, 192, or 256 bits. AES operates on a 4×4 column-major order matrix of bytes, known as the state [3].

2.5 Triple-DES

A quite simple way of increasing, the key size of DES is to use Triple DES, to guard it against attacks without the need to design a completely new block cipher algorithm.

DES itself can be adapted and reused in a more secure scheme. Many former DES users can use Triple DES (TDES) that was described and analyzed by one of DES's patentees. It involves applying DES three times with two (2TDES) or three (3TDES) different keys. TDES is quite slow but regarded as adequately secure.

2.6 Blowfish Algorithm

Blowfish is a symmetric block cipher algorithm. It uses the same secret key to both encryption and decryption of messages. The block size for Blowfish is 64 bits; messages that aren't a multiple of 64-bits in size have to be padded. It uses a variable –length key, from 32 bits to 448 bits. It is appropriate for applications where the key is not changed frequently. It is considerably faster than most encryption algorithms when executed in 32-bit microprocessors with huge data caches.

2.7 Homomorphic Encryption

Cloud consumer encrypts its data before sending to the Cloud provider, But, each time he has to work on that will have to decrypt that data. The consumer will require giving the private key to the server to decrypt the data before to perform the calculations required that might influence the confidentiality of data stored in the Cloud. Homomorphic Encryption systems are needed to perform operations on encrypted data without decryption (without knowing the private key); only the consumer will have the secret key. When we decrypt the result of any operation, it is the same as if we had performed the calculation on the “Plain Text” (or original data). The Homomorphic encryption is distinguishing, according to the operations that are performed on raw data[2] .

- Additive Homomorphic encryption: additions of the raw data.
- Multiplicative Homomorphic encryption: products for raw data.

2.8 Cast-256

The CAST-256 is a private-key block cipher that is a generalization of the basic Feistel network. CAST-256 algorithm uses 128-bit block size and a 256-bit primary key that is used in the algorithm’s key schedule scheme to generate two sets of subkeys, each of that is used per round: a 5-bit subkey KrI is used as a rotation key for round i and a 32-bit subkey KmI is used as a masking key for round i. There are a total of 48 rounds of encryption. Three different 32-bit round functions are used in CAST-256[5].

2.9 The Serpent Algorithm

The Serpent algorithm, is a 32-round Substitution-Permutation (SP) network operating on four 32- bit words. The algorithm encrypts and decrypts 128-bit data via a key of 128, 192, or 256 bits in length. The Serpent algorithm consists of three main components.

- Initial Permutation IP
- Thirty two rounds of consisting of a Round Function that performs Key Masking, S-box
- Substitution, and (in all but the last round) data mixing via a Linear Transformation.

2.10 RC4 Algorithm

The symmetric algorithm RC4 is a simple stream cipher developed by Ron Rivest in 1987. As a stream cipher, it operates on the "Plain Text" message one byte at a time. It relies solely on confusion and requires an 8 × 8 S-Box that is generated using algorithm, as well as a 256-byte array that stores the bytes of a key repeated as many times as needed to fill the entire array K0,K1,….K255. Using the S-Box and the key array, a pseudorandom sequence of numbers is generated that is then XORed with the "Plain Text" message.RC4 is a very simple algorithm and is therefore not used for highly classified data. It is used in programs such as Oracle Secure SQL and is part of the Cellular Digital Packet Data specification.

2.11 RC6 Algorithm

RC6-w/r/b, a general version of the RC6 cipher operates on units of four w-bit words, with the encryption consisting of a nonnegative number of rounds r. The user supplies a primary key of b bytes, where 0 = b = 255 and from this key, the key schedule scheme of the RC6-w/r/b algorithm derives 2r+4 subkeys, where each subkey is a w-bit word. These 2r+4 subkeys are then stored in the array S[0, …, 2r+3]. This array of subkeys is used in both encryption and decryption.

2.12 SHA-1

SHA-1 stands for "Secure Hash Algorithm 1", and was first published for use in 1995. One of its most amazing feats is taking a message of variable-length input (up to 264 bits long), and reducing to 160-bit encrypted output. At a glance, SHA-1 requires some preparation of the "Plain Text" message followed by 80 rounds of encryption. Within each round, the functions and constants used change depending on the round number. SHA-1 cannot be reversed for a variety of reasons. Input of any length maps to an output of fixed length (160 bits). Within the algorithm, there are more operations that have multiple input combinations yielding the same output, for example, XOR. This leads to the possibility of collision attacks, in that different "Plain Text" messages hash to the same digest. SHA-1 was found to be weak against collision attacks. Although no certain collisions have been found, collisions have been found on smaller versions of the problem, e.g., in implementation that use fewer rounds. The possibility of collision attacks was enough to phase out the use of SHA-1. Since then, SHA-224, SHA-256, SHA-384, and SHA-512 have been published and implemented for general use. SHA-1 is still used often, however, on data that does not require a security clearance.

3. Performance Analysis

In simulation experiments, Blowfish has shown better performance than other commonly used encryption algorithms. AES showed poor performance compared to other algorithms, since it requires more processing power. The first set of experiments was conducted in ECB Mode. Blowfish algorithm showed its superiority over other algorithms in terms of processing time. It shows also that AES consumes more resources when data block size is relatively big. In addition, the experiments proved that 3DES requires always more time than DES because of its triple phase encryption characteristic. Blowfish, that has a long key (448 bit), outperformed other encryption algorithms. DES and 3DES are known to have worm holes in their security mechanism, Blowfish and AES do not have any worm hole so far. As expected, CBC requires more processing time than ECB because of its key-chaining nature. The result indicates also that the extra time added is not significant for many applications, knowing that CBC is much better than ECB in terms of protection[1].

As Blowfish has not shown any known security weak points so far, it is an excellent candidate to be considered as a standard encryption algorithm. AES showed poor performance results compared to other algorithms since it requires more processing power. Using CBC mode has added extra processing time, but overall it was relatively negligible especially for certain application that requires more secure encryption to a relatively large data blocks. The results showed that Blowfish has a very good performance compared to other algorithms. Also it showed that AES has a better performance than 3DES and DES[3]. Amazingly it shows also that 3DES has almost 1/3 throughput of DES, or in other words it needs 3 times than DES to process the same amount of data.

4. Conclusion

From the above discussion, it is clear that Randomness is more important in both key and data blocking to optimize encryption security. Lengthy key simply means additional processing time, wastages of extra memory and electric power. So selection of extended bits key is not advisable to achieve optimal performance. Actual need of the system therefore requires revision of the cryptographic algorithms structure and creation of large number of random probabilities under optimal key length (not too long, nor too small). Symmetric scheme proves to be a good decision with at least 112 bits key if it is combined with public key cryptography. In such a case, encrypting algorithm should be symmetric in nature. To get complete security objectives, the key (symmetric secret key) should be exchanged under public key infrastructure (PKI) having a trusted third party.

Hybrid encryption scheme (symmetric + asymmetric) can therefore ensure more security, against the hypothetical feelings of misuse or spy attempts of any third party. In order to get the satisfaction of security, one needs to exchange symmetric secret key with the involvement of a reliable third party by using PKI. But the encrypted data should be exchanged separately among sender and receivers only as compared to Identity based Public Key Cryptography (ID-PKC). This will make sure that the third party cannot view data. The confidentiality of communicated information will therefore remain 100%. On the other hand if IDPKC only is used to encrypt and exchange small video clip then it does not ensure the privacy as the third party can view the personal video clip. This means privacy will be nothing in this case. As new encryption techniques are evolving every day, fast and secure conventional encryption techniques will always work out to be better with high rate of security.

REFERENCES

[1] Ayushi, 2010,A Symmetric Key Cryptographic Algorithm, International Journal of Computer Applications (0975 - 8887) Volume 1. No. 15, 2010

[2] Fontaine. C. and Galand. F.2007 ,A Survey of Homomorphic Encryption for Nonspecialists, EURASIP Journal on Information Security Volume 2007, Article ID 13801, 10 pages, doi:10.1155/2007/13801, Hindawi Publishing Corporation.

[3] NIST Advanced Encryption Standard (AES) Development Effort web site http://csrs.nist.gov/encryption/aes/aes-home.htm".

[4] Rivest, R. L., Shamir, A., Adelmann, L.: "A method for obtaining digital signature and public –key cryptosystems", *Commun. ACM*, 1978, VOL. 21, pp. 120-126

[5] Feistel, Horst, "Cryptography and Computer Privacy," *Scientific American*, Vol. 228, No. 5, May 1973, pp. 15-23.

[6] Purdy, George B., "A High Security Log-in Procedure," *Communications of the ACM*, Vol. 17, No. 8, August 1974, pp. 442-445.

[7] A. Freier, P. Karlton, and P. Kocker: "The SSL Protocol, Version 3.0", Netscape Communications Corporation, Mountain View, CA, March 1996.

[8] The XILINX Data Book, 1999

[9] W. Diffie and M.E. Hellman. Exhaustive cryptanalysis of the NBS Data Encryption Standard. *Computer* , 10: 74-84, 1977.

[10] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall and N. Ferguson: "Performance Comparison of the AES Submissions", version 2.0, February 1, 1999.

[11] E. Biham, "Design Tradeoffs of the AES Candidates", invited talk presented at ASIACRYPT '98, Beijing, 1998