A Cost Effective Symmetric Key Cryptographic Algorithm for Small Amount of Data

Mohammad Zakir Hossain Sarker
Department of CSE, East West University,
43, Mohakhali, Dhaka-1212, Bangladesh
Phone: 880-2-9887989, 9882308
E-mail: zakir.publications@gmail.com

Md. Shafiul Parvez
Department of CSE, East West University,
43, Mohakhali, Dhaka-1212, Bangladesh
Phone: 880-2-9887989, 9882308
E-mail: aakaash_ewu@hotmail.com

Abstract

Once an application steps out of the bounds of a single-computer box, its external communication is immediately exposed to a multitude of outside observers with various intentions, good or bad. In order to protect sensitive data while these are en route, applications invoke different methods. In today's world, most of the means of secure data and code storage and distribution rely on using cryptographic schemes, such as certificates or encryption keys. Thus, cryptography mechanisms form a foundation upon which many important aspects of a solid security system are built. Cryptography is the science of writing in secret code and is an ancient art. Some experts argue that cryptography appeared spontaneously sometime after writing was invented, with applications ranging from diplomatic missives to war-time battle plans. It is no surprise, then, that new forms of cryptography came soon after the widespread development of computer communications. There are two basic types of cryptography: Symmetric Key and Asymmetric Key. Symmetric key algorithms are the quickest and most commonly used type of encryption. Here, a single key is used for both encryption and decryption. There are few well-known symmetric key algorithms i.e. DES, RC2, RC4, IDEA etc. This paper describes cryptography, various symmetric key algorithms in detail and then proposes a new symmetric key algorithm. Algorithms for both encryption and decryption are provided here. The advantages of this new algorithm over the others are also explained.

1. Introduction

Does increased security provide comfort to paranoid people? Or does security provide some very basic protections that we are naive to believe that we don't need? During this time when the Internet provides essential communication between tens of millions of people and is being increasingly used as a tool for commerce, security becomes a tremendously important issue to deal with. There are many aspects to security and many applications, ranging from secure commerce and payments to private communications and protecting passwords. One essential aspect for secure communications is that of Cryptography. The concept of securing messages through cryptography has a long history. Indeed, Julius Caesar is credited with creating one of the earliest cryptographic systems to send military messages to his generals. [7]

"Cryptography" derives from the Greek word *kruptos*, meaning "hidden". The key to hiding data is to devise a hiding (encryption) mechanism that is very difficult to reverse (i.e., to find the original data) without using the decryption key. [1] Usually, the harder it is to discover the key, the more secure the mechanism. In symmetric (also called "secret-key" and, unfortunately, "private key") encryption, the same key (or another key fairly easily computed from the first) is used for both encryption and decryption. In asymmetric (also called "public-key") encryption, one key is used for encryption and another for decryption. More specifically this paper deals with the Symmetric Key cryptography. A new Symmetric Key cryptographic algorithm has been propsed in this paper with its advantages and disadvantages.

2. Brief History of Cryptography

Cryptography, the science of encrypting and decrypting information, dates as far back as 1900 BC when a scribe in Egypt first used a derivation of the standard hieroglyphics of the day to communicate.[2] There are many notable personalities who participated in the evolution of Cryptography. For example, “Julius

Caesar (100-44 BC) used a simple substitution with the normal alphabet (just shifting the letters by 3 positions) in government communications", [2] and later, Sir Francis Bacon in 1623, who described a cipher is known today as a 5-bit binary encoding. He advanced it as a steganographic device by using variation in type face to carry each bit of the encoding". For all the historical personalities involved in the evolution of cryptography, it is William Frederick Friedman, founder of Riverbank Laboratories, cryptanalyst for the US government, and lead code-breaker of Japan's World War II Purple Machine, who is "honored as the father of US cryptanalysis". In 1918 Friedman authored The Index of Coincidence and Its Applications in Cryptography, which is still considered by many in this field as the premiere work on cryptography written this century.

During the late 1920s and into the early 1930s, the US Federal Bureau of Investigation (FBI) established an office designed to deal with the increasing use of cryptography by criminals. At that time the criminal threat involved the importation of liquor. According to a report written in the mid-1930s by Mrs. Elizabeth Friedman, a cryptanalyst employed by the US government like her husband, William F. Friedman, the cryptography employed by bootleggers. Although cryptography was employed during World War I, two of the more notable machines were employed during World War II: the Germans' Enigma machine, developed by Arthur Scherbius, and the Japanese Purple Machine, developed using techniques first discovered by Herbert O. Yardley.

In the 1970s, Dr. Horst Feistel established the precursor to today's Data Encryption Standard (DES) with his 'family' of ciphers, the 'Feistel ciphers', while working at IBM's Watson Research Laboratory. In 1976, The National Security Agency (NSA) worked with the Feistel ciphers to establish FIPS PUB-46, known today as DES. Today, triple-DES is the security standard used by U.S. financial institututions. Also in 1976, two contemporaries of Feistel, Whitfield Diffie and Martin Hellman first introduced the idea of public key cryptography in a publication entitled "New Directions in Cryptography". Public key cryptography is what PGP, today's industry standard, uses in its software.In the September, 1977 issue of The Scientific American, Ronald L. Rivest, Adi Shamir and Leonard M. Adleman introduced to the world their RSA cipher, applicable to public key cryptography and digital signatures. The authors offered to send their full report to anyone who sent them self-addressed stamped envelopes, and the ensuing international response was so overwhelming the NSA balked at the idea of such widespread distribution of cryptography source code. In the mid-1980s ROT13 was employed by USENET groups to prevent the viewing of "objectionable material [by] innocent eyes", and soon thereafter, a 1990 discovery by Xuejia Lai and James Massey proposed a new, stronger, 128-bit key cipher designed to replace the aging DES standard named International Data Encryption Algorithm (IDEA). This algorithm was designed to work more efficiently with "general purpose" computers used by everyday households and businesses. Concerned by the proliferation of cryptography, the FBI renewed its effort to gain access to plaintext messages of US citizens. In response, Phil Zimmerman released his first version of Pretty Good Privacy (PGP) in 1991 as a freeware product, which uses the IDEA algorithm. PGP, a free program providing military-grade algorithm to the internet community, has evolved into a cryptographic standard because of such widespread use. The initial versions of PGP were geared towards the more computer literate individual, but to the individual nonetheless. Phil Zimmerman could be compared to Henry Ford in his efforts to provide PGP to every home by making it free, and therefore, affordable. Today, PGP's updated version is offered free to the public. In 1994, Professor Ron Rivest, co-developer of RSA cryptography, published a new algorithm, RC5, on the Internet. It had been claimed that RC5 is stronger than DES. [2]

3. The Purpose of Cryptography

In a typical situation where cryptography is used, two parties (X and Y) communicate over an insecure channel. X and Y want to ensure that their communication remains incomprehensible by anyone who might be listening. Furthermore, because X and Y are in remote locations, X must be sure that the information she receives from Y has not been modified by anyone during transmission. In addition, she must be sure that the information really does originate from Y and not someone impersonating Y. Cryptography is used to achieve the following goals:

- *Confidentiality*: To ensure data remains private. Confidentiality is usually achieved using encryption. Encryption algorithms (that use encryption keys) are used to convert plain text into cipher text and the equivalent decryption algorithm is used to convert the cipher text back to plain text. Symmetric encryption algorithms use the same key for encryption and decryption, while asymmetric algorithms use a public/private key pair. [10]
- *Data integrity*: To ensure data is protected from accidental or deliberate (malicious) modification. Integrity is usually provided by

message authentication codes or hashes. A hash value is a fixed length numeric value derived from a sequence of data. Hash values are used to verify the integrity of data sent through insecure channels. The hash value of received data is compared to the hash value of the data as it was sent to determine if the data was altered. [10]

- *Authentication*: To assure that data originates from a particular party. Digital certificates are used to provide authentication. Digital signatures are usually applied to hash values as these are significantly smaller than the source data that they represent. [10]

4. Types of Cryptography

Cryptography is a process which is associated with scrambling plaintext (ordinary text, or cleartext) into ciphertext (a process called encryption), then back again (known as decryption). There are several ways to classify the various algorithms. The most common types are i) Secret Key Cryptography which is also known as Symmetric Key Cryptography and ii) Public Key Cryptography which is also known as Asymmetric Key Cryptography. [1]

4.1. Secret Key Cryptography

In secret key cryptography, a single key is used for both encryption and decryption. As shown in Fig. 1, the sender uses the key (or some set of rules) to encrypt the plaintext and sends the ciphertext to the receiver. The receiver applies the same key (or ruleset) to decrypt the message and recover the plaintext. Because a single key is used for both functions, secret key cryptography is also called symmetric encryption.
With this form of cryptography, it is obvious that the key must be known to both the sender and the receiver; that, in fact, is the secret. The biggest difficulty with this approach, of course, is the distribution of the key.

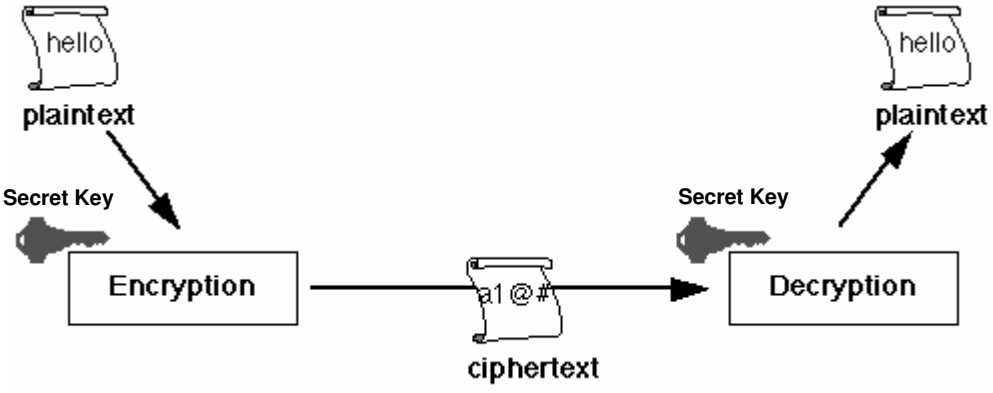

Fig. 1: Secret Key Cryptography

4.2. Public Key Cryptography

Public or asymmetric key cryptography involves the use of key pairs: one private key and one public key. Both are required to encrypt and decrypt a message or transmission. The private key, not to be confused with the key utilized in private key cryptography, is just that, private. It is not to be shared with anyone. The owner of the key is responsible for securing it in such a manner that it will not be lost or compromised. On the other hand, the public key is just that, public. Public key cryptography intends for public keys to be accessible to all users. In fact, this is what makes the system strong. If a person can access anyone public key easily, usually via some form of directory service, then the two parties can communicate securely and with little effort, i.e. without a prior key distribution arrangement. Fig. 2 describes the Public Key Cryptography.

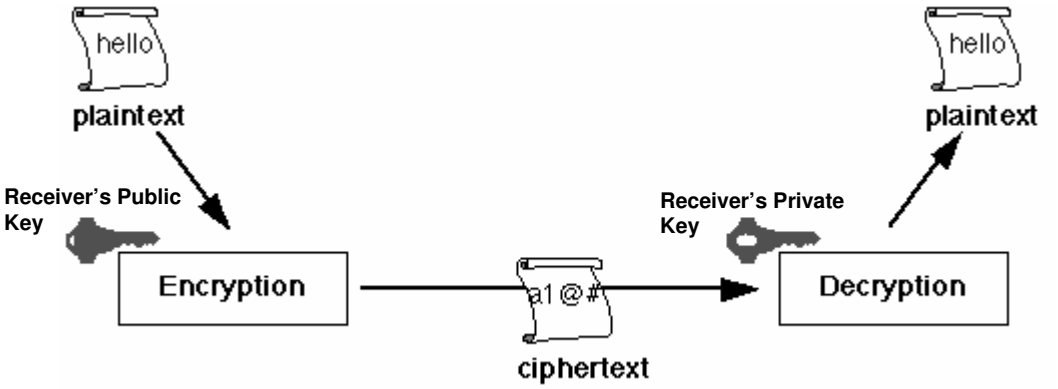

Fig. 2: Public Key Cryptography

Since this paper deals more with Secret/Symmetric Key cryptography further details of the same are discussed below.

4.3. More on Symmetric Key Cryptography

Secret key cryptography schemes are generally categorized as being either stream ciphers or block ciphers. Stream ciphers operate on a single bit (byte or computer word) at a time, and implement some form of feedback mechanism so that the key is constantly changing. A block cipher is so-called because the scheme encrypts one block of data at a time using the same key on each block. In general, the same plaintext block will always encrypt to the same ciphertext when using the same key in a block cipher whereas the same plaintext will encrypt to different ciphertext in a stream cipher. [7]

Stream ciphers come in several flavors but two are worth mentioning here. *Self-synchronizing stream ciphers* calculate each bit in the keystream as a function of the previous n bits in the keystream. It is termed "self-synchronizing" because the decryption process can stay synchronized with the encryption process merely by knowing how far into the n-bit keystream it is. *Synchronous stream ciphers* generate the keystream in a fashion independent of the message stream but by using the same keystream generation function at sender and receiver. While stream ciphers do not propagate transmission errors, they are, by their nature, periodic so that the keystream will eventually repeat.

Block ciphers can operate in one of several modes; the following four are the most important: Electronic Codebook (ECB), Cipher Block Chaining (CBC), Cipher Feedback (CFB) mode and Output Feedback (OFB) [6]. The most common secret-key cryptography scheme used today is the Data Encryption Standard (DES), designed by IBM in the 1970s and adopted by the National Bureau of Standards (NBS) [now the National Institute for Standards and Technology (NIST)] in 1977 for commercial and unclassified government applications. DES has been adopted as Federal Information Processing Standard 46 (FIPS 46-3) and by the American National Standards Institute as X3.92). DES is a block-cipher employing a 56-bit key that operates on 64-bit blocks. DES has a complex set of rules and transformations that were designed specifically to yield fast hardware implementations and slow software implementations, although this latter point is becoming less significant today since the speed of computer processors is several orders of magnitude faster today than twenty years ago. IBM also proposed a 112-bit key for DES, which was rejected at the time by the government; the use of 112-bit keys was considered in the 1990s, however, conversion was never seriously considered. Several variants of DES are currently in use, including Triple-DES (3DES, also described in FIPS 46-3) and DESX.

There are a number of other secret-key cryptography algorithms that are also in use today like CAST-128 (block cipher), RC2 (block cipher) RC4 (stream cipher), RC5 (block cipher),Blowfish (block cipher), Twofish (block cipher). In 1997, NIST initiated a process to develop a new secure cryptosystem for U.S. government applications. The result, the Advanced Encryption Standard (AES), became the official successor to DES in December 2001.

5. New Symmetric Key Algorithm

5.1. Encryption Algorithm

Step 1: Generate the ASCII value of the letter

Step 2: Generate the corresponding binary value of it. [Binary value should be 8 digits (no matter how much the length of it, we should represent it in 8 digits

(2^8=256). e.g. for decimal 32 binary number should be **00**100000 (underlined zeros are required)]

Step 3: Reverse the 8 digit's binary number

Step 4: Take a 4 digits divisor (>=1000) as the ***Key***

Step 5: Divide the reversed number with the divisor

Step 6: Store the remainder in first 3 digits & quotient in next 5 digits (remainder and quotient wouldn't be more than 3 digits and 5 digits long respectively. If any of these are less then 3 and 5 digits respectively we need to add required number of 0s (zeros) in the left hand side. So, this would be the cipertext i.e. encrypted text.

Now store the remainder in first 3 digits & quotient in next 5 digits.

[Since it will works character by character that is why spaces, commas, each & every character will be treated as one single character & we have to apply the above algorithm for every character.]

5.2. Case Study

Let, the character is "T". Now according to the steps we will get the following:

Step 1: ASCII of "T" is 84 in decimal.

Step 2: The Binary value of 84 is 1010100. Since it is not an 8 bit binary number we need to make it 8 bit number as per the encryption algorithm. So it would be 01010100

0	1	0	1	0	1	0	0

Step 3: Reverse of this binary number would be 00101010

0	0	1	0	1	0	1	0

Step 4: Let 1000 as divisor i.e. ***Key***

Step 5: Divide 00101010 (dividend) by 1000(divisor)

Step 6: The remainder would be 10 and the quotient would be 101. So as per the algorithm the ciphertext would be 01000101 which is ASCII 69 in decimal i.e. "E"

0	1	0	0	0	1	0	1

5.3. Decryption Algorithm

Step 1: Multiply last 5 digits of the ciphertext by the ***Key***

Step 2: Add first 3 digits of the ciphertext with the result produced in the previous step

Step 3: If the result produced in the previous step i.e. step 2 is not an 8-bit number we need to make it an 8-bit number

Step 4: Reverse the number to get the original text i.e. the plain text

5.4. Another Case Study

After encrypting "T" we have got 01000101 as the ciphertext. Now according to decryption algorithm lets try to get back the original text i.e. "T"

Step 1: After multiplying 00101 (last 5 digits of the ciphertext) by 1000 (***Key***) the result would be 101000

		1	0	1	0	0	0

Step 2: After adding 010 (first 3 digits of the ciphertext) with 101000 the result would be 101010

		1	0	1	0	1	0

Step 3: Since 101010 is not an 8-bit number we need to make it 00101010

0	0	1	0	1	0	1	0

Step 4: After reversing the number it would be 01010100 i.e. ASCII 84 in decimal i.e. "T" as character which was the original text

0	1	0	1	0	1	0	0

6. Advantages of the New Algorithm

1. The Algorithm is very simple in nature
2. There are two reverse operations present in this algorithm which would make it more secured
3. CRC checking in receiving ends is easier
4. For a small amount of data this algorithm will work very smoothly.

7. Conclusion

Cryptography is used to achieve few goals like Confidentiality, Data integrity, Authentication etc. of the data which has sent to the receiver from the sender. Now, in order to achieve these goals various cryptographic algorithms are developed by various people. It has been found that the algorithms which are available at this moment are more or less difficult or complex in nature, and of-course it is quite obvious. Because those algorithms are used to maintain high level of security against any kind of forgeries. For a very minimal amount of data those algorithms wouldn't be cost effective since those are not designed for small amount of data. The aim of this work was to design and implement a new algorithm to address this issue so that we don't have to apply those algorithms (which are not cost-effective) to encrypt a small

amount of data. Keeping this goal in mind the proposed algorithm has been designed in a quite simple manner but of-course not sacrificing the security issues. A single is used for both encryption and decryption i.e. it is fallen under secret key cryptographic algorithm. But as public key cryptography is more secured then secret key cryptography our next task would be to develop and design a public key cryptographic algorithm in a simple manner as it is done in this paper.

8. References

[1] "Basic Cryptographic Algorithms", an article available at www.itsc.state.md.us/oldsite/info/InternetSecurity/Crypto/CryptoIntro.htm#Algorithms

[2] S. Hebert, "A Brief History of Cryptography", an article available at http://cybercrimes.net/aindex.html

[3] "Introduction to Public-Key Cryptography", an article available at developer.netscape.com/docs/manuals/security/pkin/contents.htm

[4] D. Jablon, "Strong Password Only Authenticated Key Exchange" Computer Communication Review, *ACM SIGCOMM*, Vol 26 No 5, pp 5-26, 1997

[5] S. M. Bellovin and M. Merritt, "Augmented Encrypted Key Exchange a Password Based Protocol Secure Against Dictionary Attacks and Password File Compromise" Proceedings of *the First ACM Conference on Computer and Communications Security*-1993, pp 243-250

[6] S. Goldwasser and S. Micali, "Probabilistic Encryption", *Journal of Computer and System Sciences*, Vol 28, 1994 pp 270-299

[7] S. William, *Cryptography and Network Security: Principles and Practice*, 2nd edition, Prentice-Hall, Inc., 1999 pp 23-50

[8] M. I. Jabiullah, S.M. Mizanur Rahman, M. Lutfar Rahman and M. Alamgir Hossain, "Secure Pseudoorandom Bit Generation for Cryptographic Application", Proceedings of *International Conference on Computer and Information Technology (ICCIT),* Dhaka, 2001 pp 275-277

[9] M.I. Sharif, E. Karim, A.N. Mahmood and M. A. Mottalib, "Another Tip for Secure RSA Key Selection", Proceedings of *International Conference on Computer and Information Technology (ICCIT)*, Dhaka, 2001 pp 283-285

[10] K. Gary, *"An Overview of Cryptography"*, an article available at www.garykessler.net/library/crypto.html