NET SECURITY-ACHIEVED BY CRYPTOGRAPHIC ALGORITHMS

PRESENTED BY

D.R.ESESVE,
III/IV B.Tech,
VITAM COLLEGE OF ENGINEERING.

URL: members.rediff.com/dresesve/nsaca.html

D.R.ESESVE

ABSTRACT

Dresesve@rediffmail.com

This paper is about CRYPTOGRAPHY– *the science of scrambling data to make it unintelligible to all, except the intended person, and its various methods in making the work look more perfect. Cyber crimes have become a common piece of vandalism in the present computerized world. What has changed dramatically is the ability to tamper with information. In a society where information is mostly stored and transmitted in electronic form, what has become an absolute must is a means to ensure information security that is independent of the actual physical medium responsible for generating or storing it. Today, there is an immediate need to delve into the clutter of material. There are many laws in enforcement and many new devices, which are used to prevent such crimes. Nevertheless, these all devices resemble a seismograph, which records the amount of quake but cannot prevent the same. Here comes the discussion on* cryptography, which *plays a major role in this field of security of sending messages?*

CRYPTOGRAPHY: *The science of mathematical techniques related to aspects of information security such as encryption [confidentiality] and authentication [data integrity, non-repudiation, entity authentication, and data origin authentication]. Its primary aim is to render a message incomprehensible to any unauthorized reader. The four main objectives of cryptography are…*

- Confidentiality: *To keep information secret from any unauthorized reader/s.*
- Data Integrity: *To ensure that information has not been altered by unauthorized or unknown means.*
- Non-repudiation: *To prevent the denial of previous Commitments or actions.*
- Authentication Entity Authentication: *To validate the identity of the person.*
- Data Origin (Message) Authentication: *To corroborate the source of information.*

By all these methods, cryptography helps us and thereby making us go into its details.

INTRODUCTION

News item in "The Times of India", Mumbai, on June 27, 1999: *"The Indian Institute of Technology, KANPUR has developed a new generation cipher code for the Indian navy. The breakthrough would provide a technological edge to defense communication. Christened* "TRINETRA"*, the cipher is a modern computer-based code language system which can digitize long, alphabetic messages within seconds."*

What Cryptography Can Do?

Potentially, cryptography can hide information while it is in transit or storage. In general, cryptography can:

1. Provide secrecy.
2. Authenticate that a message has not changed in transit.
3. Implicitly authenticate the sender.

Cryptography hides words: *At most, it can only hide talking about contraband or illegal actions. However, in a country with "freedom of speech," we normally expect crimes to be more than just "talk." Cryptography can kill in the sense that boots can kill; that is, as a part of some other process, but that does not make cryptography like a rifle or a tank. Cryptography is defensive, and can protect ordinary commerce and ordinary people. Cryptography may be to our private information as our home is to our private property, and our home is our "castle." Potentially, cryptography can hide secrets, either from others, or during communication. Therefore, the importance of cryptography is understood in the field of Internet security. There fore before going into the actual process its mandatory to define certain terms and give certain definitions.*

Some of the terms used on this article are explained below:

- Plaintext *is the text or data that is to be encrypted. It is one of the inputs the encryption algorithm.*
- Key: *A "key" (as it stands in the world of cryptography) is an abstraction. It is analogous to the general concept of protecting things with a "lock," thus making those things available only if one has the correct "key." Similarly, by using various values as keys, it is possible to create many different cipher texts for a message. If the value used as the "key" for a particular transformation (from "raw text" to "cipher text") is known. It is also possible to get the "raw*

text" back from the "cipher text". Broadly, which keys can be classified (based on the usage) in four categories.

1. User key: *The user remembers the value of the key.*
2. Alias key: *The key for an alias file that serves as a database of keys and their users. Such a file stores values in key-value pairs relating for e.g. user names with their keys.*
3. Message key: *A random value that differs from message to message.*
4. Running key: Normally *used in Stream Ciphers to serve as the confusion sequence. A properly chosen key should exhibit diffusion over the message. This means that changing even one bit in the value of the key should change every bit in the message with a probability of 0.5. A key with any lesser diffusion may be susceptible to attacks.*

- Ciphertext *is the output of the encryption algorithm. It can be transmitted safely as it is unintelligible without the key.*

*One of the earliest known and simplest methods of encryption is the "*Caesar cipher method*", which involves replacement of each letter in the message with some other letter or numeral. Because of the use of lesser number of keys and knowledge of encryption and decryption algorithms, any advanced computer using either parallel processing or distributed computing can crack this method. This type of cracking is termed as Brute Force Attack, in computer parlance.*

Here are certain other methods that use the same bit replacement technique but in a different way.

INTERNATIONAL DATA ENCRYPTION ALGORITHM (IDEA) and DATA ENCRYPTION STANDARD (DES) *are those two algorithms that use the bit-replacement technique explained above. These algorithms, along with compression, are used to send secure e-mail using software programs like PGP* (Pretty Good Privacy); *the present standard for encryption in the later part of the article is also used.*

The IDEA Algorithm

IDEA is best know as a component of PGP. It is a block cipher which uses a 128-bit length key to encrypt successive 64-bit blocks of plaintext which are mangled in a sequence of parameterized iterations of data manipulation to

produce 64-bit cipher text output blocks. The manipulations may consist of one or a combination of the following three operations.

- *Bit-by-Bit exclusive OR-ing.*
- *Addition of integers modulo 2^{10} .*
- *Multiplication of integers modulo 2^{16}.*

The 128 bit key size gives the key space (total number of combinations of bits in the key) of 3.4 x 10^{38} keys, which makes the "Brute force attack" *impractical. It was also designed to withstand* differential cryptanalysis. *The procedure is quite complicated using sub-keys generated from the key to carry out a series of modular arithmetic and XOR operations on segments of the 64-bit plaintext block. The encryption scheme uses a total of fifty-two, 16-bit sub-keys. These are generated from the 128-bit sub-key as follows:*

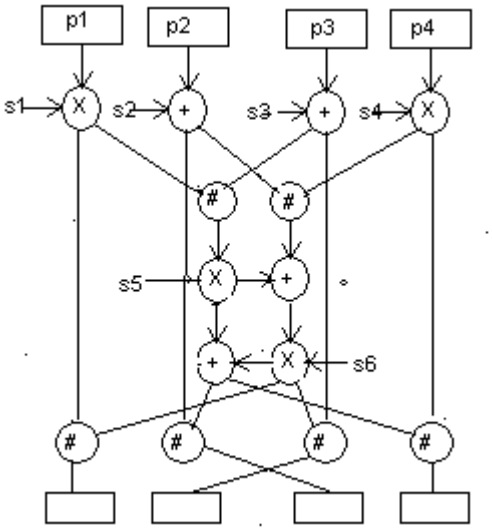
- *The 128-bit key is split into eight 16-bit keys, which are the first eight sub-keys.*
- *The digits of the 128-bit key are shifted 25 bits to the left to make a new key which is split into the next eight 16-bit sub-keys.*

The second step is repeated until the fifty-two sub-keys have been generated.

The encryption involves ***16-Bit modular multiplication*** *with a modulus of (2^{16}+1),* ***16-Bit addition*** *with a modulus of (2^{16}) and* ***16-Bit EXCLUSIVE OR*** *operations. The 64-bit plaintext block is split into four 16-bit segments, which we will call p1, p2, p3 and p4. The sub-keys are s1, s2, s3, s4, ... s52.*

The encryption consists of eight iterations with each iteration involving the following steps:
P1 x s1 --> d1
P2 + s2 --> d2
P3 + s3 --> d3
P4 x s4 --> d4
D1 XOR d3 --> d5
D2 XOR d4 --> d6
D5 x s5 --> d7
D6 + d7 --> d8
D8 x s6 --> d9
D7 + d9 --> d10
D1 XOR d9 --> d11

D3 XOR d9 --> d12
D2 XOR d10 --> d13
D4 XOR d10 --> d14



After this processes the output blocks d12, d13 are exchanged so that d11, d13, d12 and d14 are used as input to the next round (in that order) along with the next 6 sub-keys, s7 to s12. This procedure is followed for eight rounds in total giving four output blocks that we will call e1, e2, e3 and e4. Four more steps using the last four sub-keys complete the encryption:

E1 x s49 --> c1
E2 + s50 --> c2
E3 + s51 --> c3
E4 x s52 --> c4

Note: *For the purposes of the algorithm, a key of all zeros is defined as being equal to* 2^{16} *for modular multiplication steps.*

The final four output blocks, c1 to c4, are re-attached to form a 64-bit block of the ciphertext. The whole process is repeated for successive 64-bit blocks of plaintext until all of the plaintext has been encrypted. Decryption uses exactly the same sequence of operations of successive 64-bit blocks of the ciphertext, but with a different set of sub-keys. The decryption sub-keys are worked out from the encryption sub-keys being either multiplicative or additive inverses of them. The

decryption sub-keys (relative to the encryption sub-keys s1 to s52) are shown in the table below:

1st round	*s49* s50# s51# s52* s47 s48*
2nd round	*s43* s45# s44# s46* s41 s42*
3rd round	*s37* s39# s38# s39* s35 s36*
4th round	*s31* s33# s32# s34* s29 s30*
5th round	*s25* s27# s26# s28* s23 s24*
6th round	*s19* s21# s20# s22* s17 s18*
7th round	*s13* s15# s14# s16* s11 s12*
8th round	*s7* s9# s8# s10* s5 s6*

Final transformation... s1 s2# s3# s4**

sXX = multiplicative inverse of sXX modulus ($2^{16}+1$)*
sXX# = additive inverse of sXX modulus 2^{16}
[NOTE: A sub-key with all bits zero is its own multiplicative inverse in this algorithm]

THE DATA ENCRYPTION STANDARD (DES)

The features of DES:

- *Provides high level of security.*
- *Completely specified and easy to understand.*
- *The algorithm itself provides the security.*
- *Available to all users.*
- *Adaptable for use in diverse applications.*
- *Economical and efficient to implement in electronic devices.*

Overview of DES

DES is a combination of Substitution technique (for confusion) and Transposition technique (for diffusion). These two techniques are repeated for 16 cycles one on top of the other. Plaintext is encrypted in blocks of 64 bits. Keys are 64 bits long (only 56 are really needed). Uses only standard arithmetic and logical operations on up to 64 bit numbers.

Basically, there are four Modes of Operation

- *ECB -Electronic Code Book*
- *CBC - Cipher Block Chaining*
- *OFB - Output Feedback*
- *CFB - Cipher Feedback*

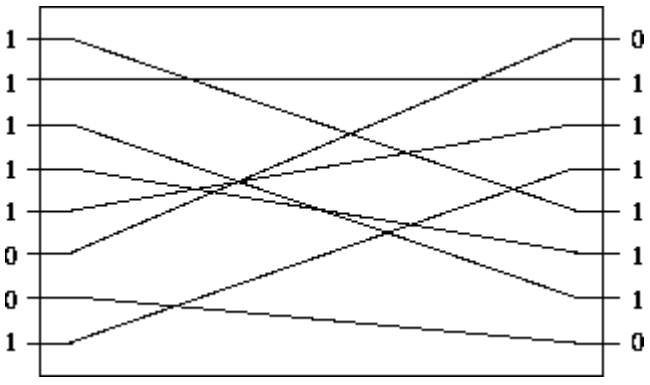
ECB Mode

The Electronic CodeBook has three modes.

- *Encrypt Mode: Data is inputted into ECB. It must have a block size of 64 bits (8 bytes) long. Key must also be 64 bits but only 56 are used.*
- *Decrypt Mode: Ciphertext block is inputted. Key is the same as used for encryption.*
- *Bit Sensitivity: If only one bit of either the input plaintext or key is changed, the output block will be completely different.*

There are two distinct algorithms in basic building blocks. They are Crypting algorithm and Key scheduler. Each is composed of subprograms i.e. Permutation Boxes (P-boxes) and Substitution Boxes (S-boxes).

Permutation Box (P-Box): *The figure below shows the permutation of the EBCDIC character "9" (binary 11111001). The P-box has transformed it into (01111110) the EBCDIC character "=".*



P-Box

The diffusion process of the P-Box has disguised the information. If we consider the P-Box to be a mini-encryption algorithm then the key is the fixed pattern of wires. The process is invertible.

The Exclusive-OR operation can be viewed also as a mini-encryption algorithm also known as "addition modulo two". The process is invertible.

XOR Operation		
11111001	EBCDIC	'9' plaintext
00101101	XOR	Random bits key
11010100	EBCDIC	'M' ciphertext
11010100	EBCDIC	'M' ciphertext
00101101	XOR	Random bits key
11111001	EBCDIC	'9' plaintext

Substitution Box (S-Box): *Introduces confusion and non-linearity to DES. It interprets bits as numbers. A number is replaced by an another from a table. The table has values ranging from zero (0000) to 15 (1111) and duplications among the elements. It takes 6-bit input in which the first and last bits choose row into S-box substitution table and the middle four bits choose the column. The table returns a 4-bit number as output. They are the heart and soul of the algorithm's secrecy.*

S-Box

	Column Number															
Row No	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

Example S-box Input/Output		
INPUT	binary 101011	= decimal 43
First and Last bits	binary 11	= decimal 3
Middle four bits	binary 0101	= decimal 5
OUTPUT	binary 1001	= decimal 9

The weakness of ECB Mode are…

- *ECB Mode encrypts a 64-bit block independently of all other 64-bit blocks*

- *Given the same key, identical plaintext will encrypt the same way*
- *Data compression prior to ECB can help (as with any mode)*
- *Fixed block size of 64 bits therefore incomplete block must be padded*

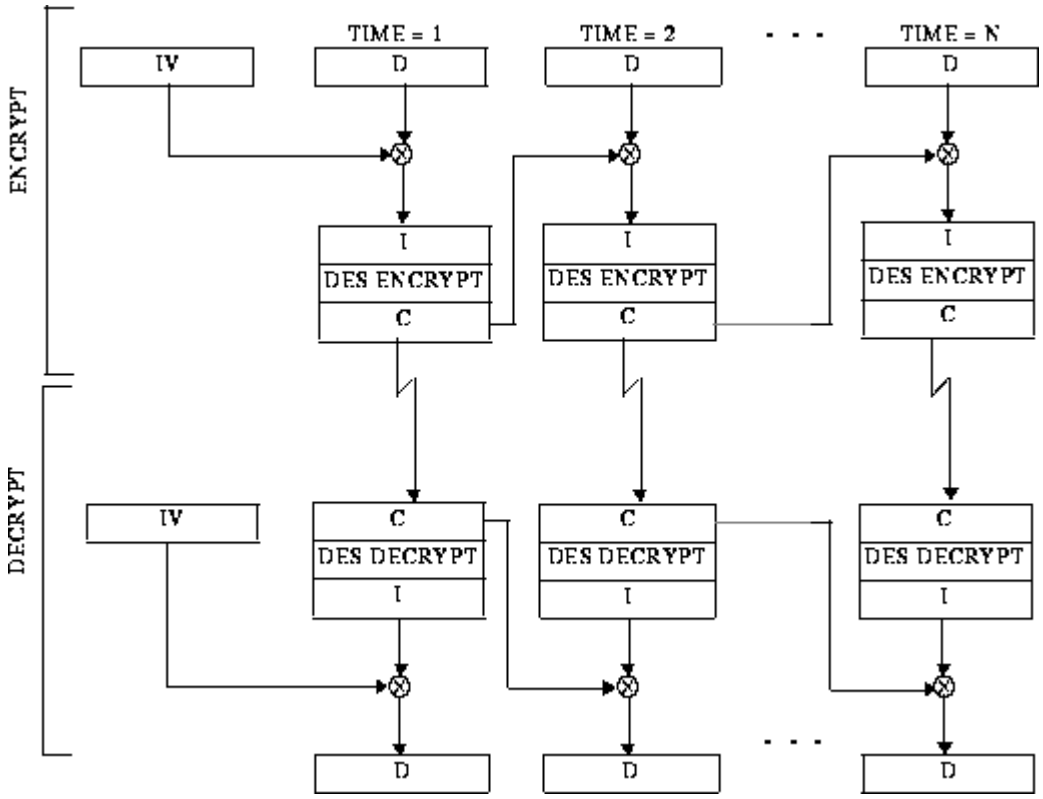
Since each block is independent of previous blocks, only those in error need be resent.

CBC Mode

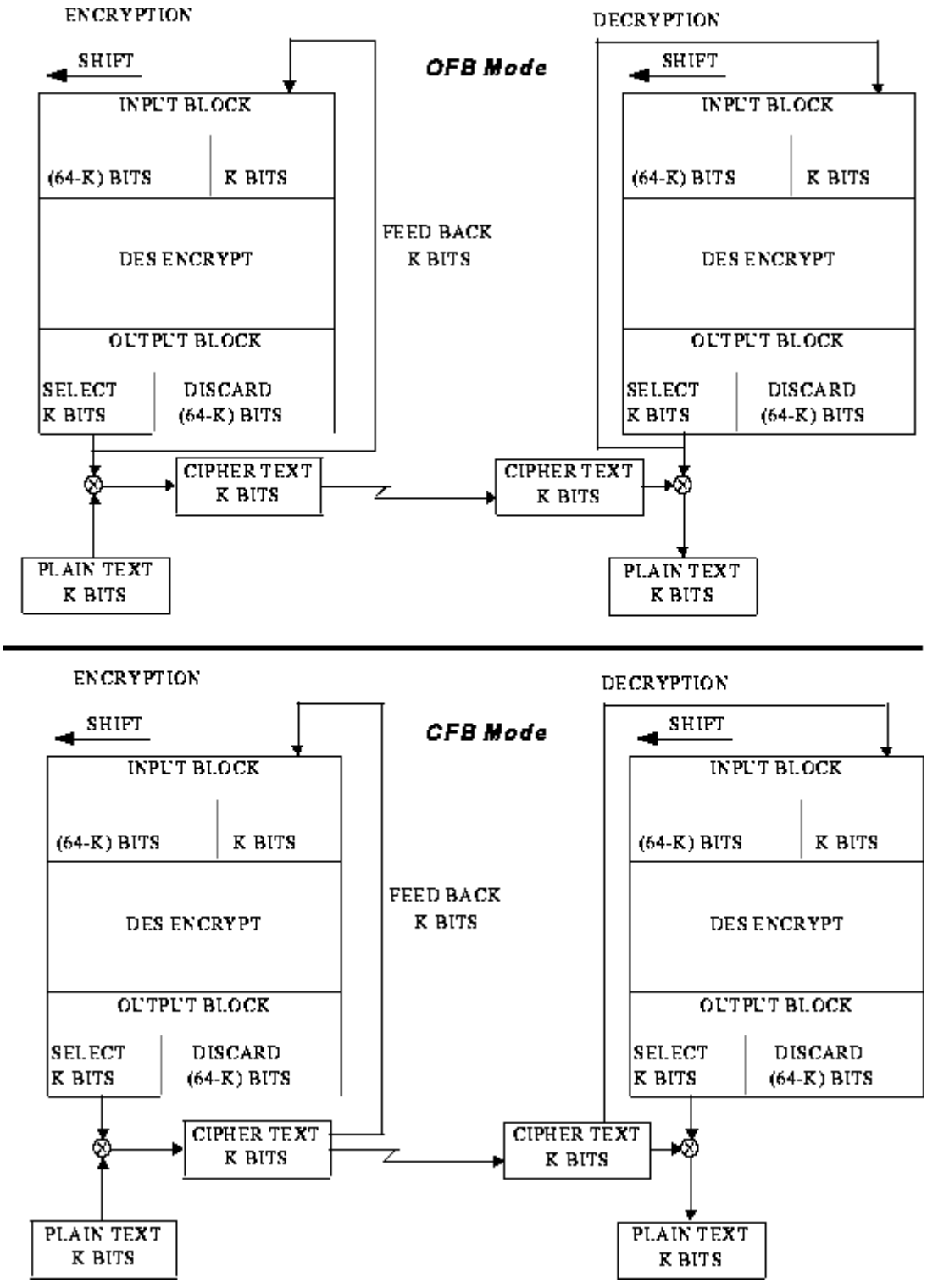
The encryption of each block depends upon the encryption of the previous block and must use an initializing vector (IV). The IV may be viewed as a second key.

- *C1 = E(B1)*
- *C2 = E(E(B1) florin B2) = E(C1 florin B2)*
- *C3 = E (E (E (B1) florin B2) florin B3) = E (C2 florin B3) etc.*

A single bit error affects two blocks.



FEEDBACK MODES



In Feedback *modes, 64-bit block now replaced by a K-bit block. Therefore, it is possible to encrypt data of any length. A 1-bit* Feedback *would require 64 cycles*

to encrypt 64 bits of data. It has the ability to encrypt on a character-by-character basis. Trade-off is Speed vs. Flexibility of block size. It is used for Message Authentication Codes (MAC) and frequently referred to as message digests.

OFB Mode

DES in OFB mode is being used as random number generator. IV is the seed. DES is used in encrypt mode for both encryption and decryption of text since a reverse key schedule cannot be used because of the XOR operation. OFB is often used as a random number generator. In OFB error propagation, only a single bit is affected.

CFB Mode

On encryption, the ciphertext, rather than the output from DES is fed back. It can affect two blocks, for same reasoning as ECB. Due to transparency to protocols and flexibility in block, size greater security is provided. It is used for communications and encrypting fields within a record.

CRITICISMS OF THE DES

- *Number of iterations-Is 16 enough?*
- *Key length - 256 possible keys to try. A massively parallel system could try all keys in 1 day (although it would be a very expensive proposition).*
- *Double encryption may be the answer.*

Weaknesses of the DES

- *Weak keys (e.g. all zeros or all ones).*
- *Semi-Weak keys (two separate keys can decrypt the same message).*
- *The same DES algorithm is used.*
- *The keys are used in reverse order.*
- *Key length.*

OVERVIEW OF THE RSA MATHEMATICAL ALGORITHIM:

Ron Rivest, Adi Shamir and Len Adleman at MITS developed the RSA public key scheme in 1977. The scheme has reigned supreme as the only widely accepted and implemented approach to public-key encryption.

RSA is a block cipher in which the plaintext and ciphertext are integers between 0 and n-1 for some n. Encryption and decryption are of the following form, for some plaintext block M and ciphertext block C:

$$C = M^e \bmod n$$

$$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$$

Both the sender and receiver know the value of n. The sender knows the value of e, and only the receiver knows the value of d. Thus, this is a public-key encryption algorithm with a public key of KU={e, u} and a private key of KR={d, n}. For the algorithm to be satisfactory for public-key encryption the following requirements must be met:

- *It is possible to find values of e, d and n such that* $M^{ed} = M \bmod n$ *for all* $M < n$.
- *It is relatively easy to calculate* M^e *and* C^d *for all values of* $M < n$.
- *It is infeasible to determine d given e and n.*

RSA meets these criteria for large values of e and n i.e. large key lengths. In summary, the strength of the RSA algorithm is based on the fact that factors of a large prime number cannot be easily determined from having the number alone.

ADVANTAGES/WEAKNESSES OF RSA ALGORITHM:

The RSA algorithms primary strength:

- *The 128-bit (key length) is for all intents and purposes unbreakable. The estimated time to search all the possible combinations of keys by sheer brute force would take a Pentium computer more than 5 Billion years. Although increasing both the number of computers and their processing power would reduce the time taken, it would still be unfeasible long.*

However, there are several misconceptions concerning public key algorithms:

- *Public-key encryption is more secure than from cryptanalysis than conventional encryption. The security of any encryption scheme depends on the length of the key and the computational work involved in breaking a cipher.*
- *Public-key encryption is a general-purpose technique has made conventional encryption obsolete. Because of the computational overhead of current public-key encryption schemes, there seems no foreseeable likelihood that conventional encryption will be abandoned.*

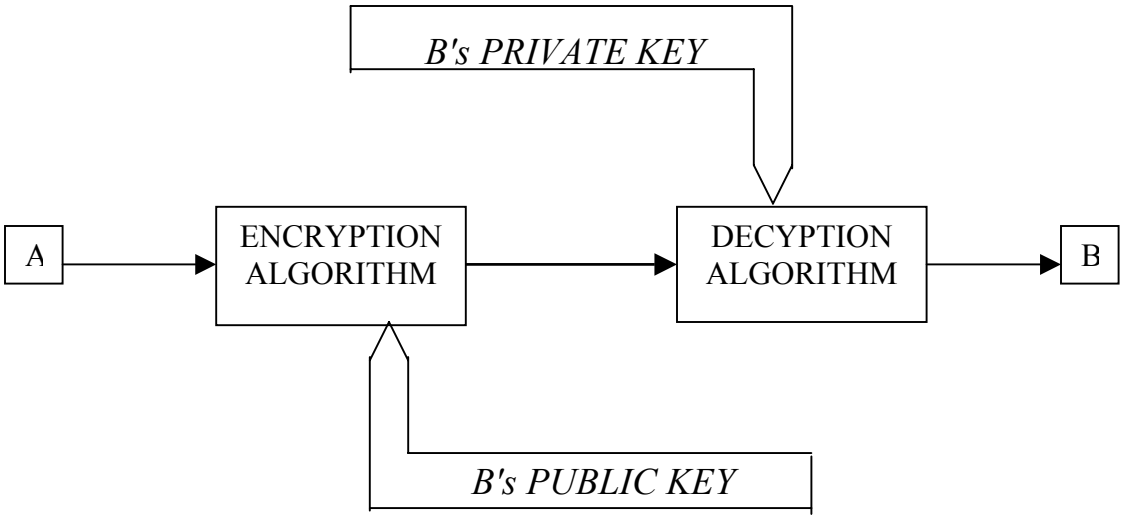
- *Public-key distribution is trivial. Some protocol is still needed. However, SSLs method of key distribution tidely caters for this through the use of digital ids and key distribution centers.*

And weaknesses…

- *Forty bit and fifty-six bit keys have been broken a number of times.*

Key Management:

With conventional encryption, a fundamental requirement for two parties to communicate securely is that they share a secret key. How to distribute secret keys securely is the most difficult problem for conventional encryption. This problem is wiped away with public-key encryption by the simple fact that the private key is never distributed. The Private key is kept secure with the owner at all times and the Public key is broadcast to all and sundry.



In the RSA/SLL implementation, 3rd party organizations handle the public access to these keys thereby ensuring the authenticity of a key.

WHAT CRYPTOGRAPHY CAN NOT DO?

Cryptography can only hide information after it is encrypted and while it remains encrypted. However, secret information generally does not start out encrypted, so there is normally an original period during which the secret is not protected. Moreover, secret information generally is not used in encrypted form, so it is again outside the cryptographic envelope every time the secret is used. Secrets are often related to public information, and subsequent activities based on the secret can indicate what that secret is.

In addition, while cryptography can hide words, it cannot hide:

1. *Physical contraband.*
2. *Cash.*
3. *Physical meetings and training.*
4. *Movement to and from a central location.*

An extravagant lifestyle with no visible means of support, or Actions. Moreover, cryptography simply cannot protect against:

1. *Informants.*
2. *Undercover spying.*
3. *Bugs.*
4. *Photographic evidence, or Testimony.*

It is a joke to imagine that cryptography alone could protect most information against Government investigation. Cryptography is only a small part of the protection needed for "absolute" secrecy.

CONCLUSION

The net, like the remaining utilities, has all the effects both positive and negative. This will only have a good effect when it is used for the good and that can only be achieved only when all the knowledged people have good intentions. Good ideas are obtained only when people see good, but most of the children, adults watch only hazardous information on the net. The net is dumped with porn, bomb making techniques, hacking techniques. In this environment how a person can learn good?

An U.S. government expert speaking at a National Information Systems Security in Baltimore on October 17 2000 described Internet security as "a tremendous catch game"; one has to keep a step ahead of the hacker. Will this be possible always? Can we predict future? If we can then we may think in such a way, but it is sure that a human cannot look into future. Thinking abilities have been changing from generation to generation, so its not easy to manufacture security systems for future, keeping in mind the changes that are going to occur in future. There by, we can see that most of the anti virus software companies start including viruses only after they occur at some or the other place. It is the situation in everything of life. It is only possible to design a medicine for a disease unless and until it has occurred somewhere or the other.

Anyway, it is not meant that there cannot be any security, but these securities are to be found before we destroy ourselves. Because, as the way, Metnick and others acted, shows that there is an eminent threat to the greatest of greatest securities in the world. If in the same way if codes can be intercepted and decoded then there is no security for anybody, keeping individuals apart the entire networks of any country are at threat. Thereby the whole world is at the forefront of a big virtual war.

To our opinion, it seems web has lesser secure solution and users that are more destructive. However, we have scientists who have great brains bombarded with intelligent ideas, so therefore one or the other day these crime stories may have a full stop. The solutions are expected to come soon.

Hope we have a secure, child insensitive, peace provoking, fun filled, and knowledge filled world of web.

REFERENCES

- ✓ Computer Networks by Andrew S Tenanbaum.
- ✓ Most of the references were made from the temple of knowledge- World Wide Web.
- ✓ In addition, some more research work went successfully with the aid of certain best magazines, available.