Study and Design of an Encryption Algorithm for Data Transmitted Over the Network by the IDEA and RSA

Ahmed Nashaat Shakir
University of Kirkuk,
Kirkuk, Iraq

ABSTRACT
In this research, an Encryption algorithm are designed and implemented for transmitted data over a network and the program written by (c) language. Rivest-Shamir-Adleman (RSA) algorithm and International Data Encryption Algorithm (IDEA) are designed to be complex and highly confidential, where a combination of symmetric and asymmetric cryptographic algorithms is used. In addition, a special algorithm developed to generate the cryptographic key complexly and randomly, this made any data transmitted across the network, and encryption keys exchanged between network sides disobeyed for any attempt to reveal them and know their content. Thus, transmitted data will be secured across any network that use this algorithm.

Keywords
Encryption, Decryption, Symmetric Encryption, Asymmetric Encryption, RSA, IDEA

1. INTRODUCTION
Encryption is the process of converting information to secret code in which information be protected from any unauthorized access. Clear and consistent message information M will be converted to meaningless scattered symbols C and this conversation be completed by encryption algorithm E starting from a cryptographic by encryption process Ke according to the function C = E Ke (M) and the reverse of this process is called decoding where the original message M is retrieved by using decoding algorithm D Starting from the decryption key Ke according to the function M = D K_d (C) [1], [2], [3].

There are two main ways to encrypt information:

1- Symmetric encryption: It has the same encryption key in both sides and the encryption algorithm is similar to the decoding algorithm (E = D, Ke = K_d)

2- Asymmetric encryption: there in shall be (E ≠ D, $K_e \neq K_d$)

This method is called public key algorithm, where encryption key K_e is called Public key and key K_d is called Privet key.

The algorithm adopted in this study combined the two methods of coding together, a complex cryptographic algorithm is obtained with high confidentiality that ensures data protection, as well as the protection of distributed and shared encryption keys between network nodes.

2. THE MAJOR SPECIFICATION OF IMPLEMENTED ALGORITHM
The implemented cryptographic algorithm contains three partial encryption algorithms, they are:

2.1 Rivest-Shamir-Adleman(RSA) Algorithm
RSA is designed by Ron Rivest, Adi Shamir, and Leonard Adleman in 1978. It is one of the best known public key cryptosystems for key exchange or digital signatures or encryption of blocks of data. RSA uses a variable size encryption block and a variable size key. It is an asymmetric (public key) cryptosystem based on number theory, which is a block cipher system. It uses two prime numbers to generate the public and private keys. These two different keys are used for encryption and decryption purpose. Sender encrypts the message using Receiver public key and when the message gets transmit to receiver, then receiver can decrypt it using his own private key [4], [5], [6].

RSA operations can be decomposed in three broad steps, key generation, encryption and decryption. RSA have many flaws in its design therefore not preferred for the commercial use. When the small values of p & q are selected for the designing of key then the encryption process becomes too weak and one can be able to decrypt the data by using random probability theory and side channel attacks. On the other hand if large p & q lengths are selected then it consumes more time and the performance gets degraded in comparison with DES. Further, the algorithm also requires of similar lengths for p & q, practically this is very tough conditions to satisfy. Padding techniques are required in such cases increases the system's overheads by taking more processing time [7], [8].

2.1.1 Key Generation Procedure
1. Choose two distinct large random prime numbers p & q such that $p \neq q$.
2. Compute n= p × q.
3. Calculate: phi (n) = (p-1) (q-1).
4. Choose an integer e such that 1<e<phi(n)
5. Compute d to satisfy the congruence relation d × e = 1 mod phi (n); d is kept as private key exponent.
6. The public key is (n, e) and the private key is (n, d). Keep all the values d, p, q and phi secret.

2.1.2 Encryption
Plaintext: P < n
Ciphertext: C= P_e mod n.

2.1.3 Decryption
Ciphertext: C
Plaintext: P=Cd mod n.
decrypt blocks of data consisting of 64 bits by using a 64-bit key [10]. Figure 1 illustrates the sequence of events followed by RSA algorithm for the encryption of multiple blocks.

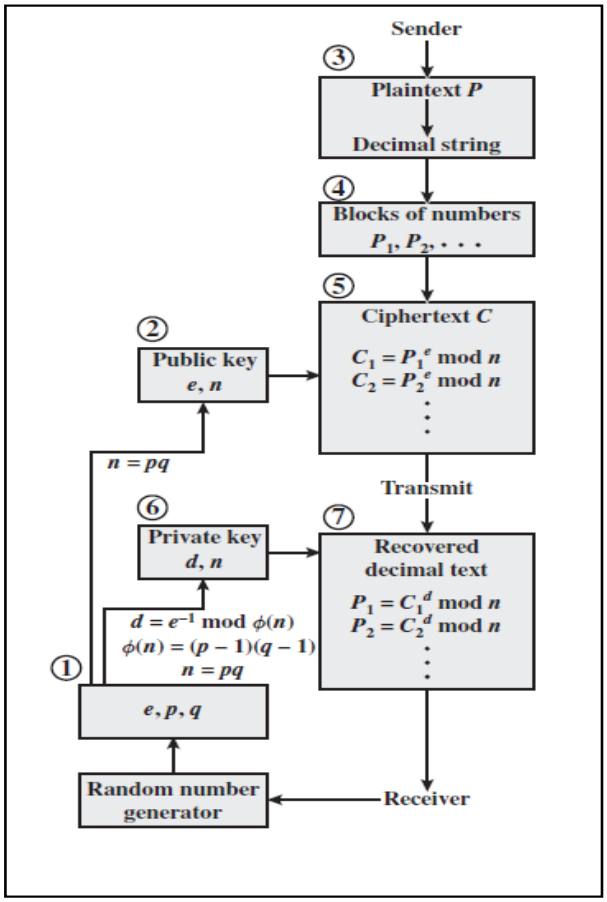

Fig.1 RSA processing of Multiple Blocks [7], [9]

For a survey on fast variants of RSA see [4], [10], [11].

It is a public key algorithm and its operation principle summarized as follows:

Firstly, the message divided into blocks in which each block has a binary value that is smaller than a number n, encryption and decryption will be in the following format:

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

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

Where M is the message or a letter from the message, or more precisely is the number assigned to the symbol or to a letter from the message. The keys are as follows:

public key Ku = {e , n} , Privet key Kr = {d , n}

And get these keys as follows:

1- An odd number e chosen and it is a part of public key.

2- Two primary numbers q,p chosen where number

(P-1)(q-1)-1 Accept division to e .

3- calculate n where n = p.q

Where n as we mentioned is the first part of both keys .

4- Calculate d where d = (p-1)(q-1)(e-1) +1 / e

It is the second part of the private key

Illustrated this algorithm with the following example:

A- The odd number e=3 chosen .

B- Two primary numbers q=11, p=5 chosen where:

(q - 1)(p - 1) – 1 = 39 acceptable division to e and calculate n: n = q . p = 55 , calculate d: d = (p-1)(q-1)(e-1) +1 / e = 27

So have Ku ={3 , 55} Kr ={27 , 55 }

Assuming M=7 then it be:

Encryption $C = M^e \bmod n = (7)^3 \bmod 55 = 13$

Decryption $M = C^d \bmod n = (13)^{27} \bmod 55 = 7$

In order to increase confidentiality and eliminate the problem of key distribution of this algorithm on each node of the network, nodes has its own key that generate it in secret and then each node has to publish the public key and keep the private key and accordingly there is not need to devise any secret method to exchange these keys. The responsibility of securing the private key is on each node aloof for the other nodes. The network contract agreed upon in a special strategy for changing keys, which may be periodic or when necessary, in order to raise the level of confidentiality of the approved encryption method.

2.2 International Data Encryption Algorithm IDEA

IDEA is one of the strongest cryptographic algorithms. Idea is a block cipher. It works on 64-bit plain text blocks. The key is longer and consists of 128 bits. IDEA is reversible of DES [4], [12].

The 64-bit plaintext block is partitioned into four 16-bit sub blocks. Four 16-bit key sub-blocks are required for the subsequent output transformation, it is generated from the 128-bit key. The key sub-blocks are used for the encryption and the decryption. IDEA was used in Pretty Good Privacy (PGP), (International Data Encryption Algorithm) IDEA is a block encryption algorithm designed by Xuejia Lai and James L and it was first described in 1991.The original algorithm went through few modifications and finally it got named as International Data Encryption Algorithm (IDEA) [12].

IDEA is a block cipher that operates with 64 bit plain text and cipher text blocks and it is controlled by 128 bit key. This algorithm works on 64-bit plain text and cipher text block (at one time). For encryption purpose, the 64- bit plain text is divided into four 16 bits sub-blocks. In our discussion, we denote these four blocks as P1 (16 bits), P2 (16 bits), P3 (16 bits) and P4 (16 bits). Figure 2 illustrate IDEA Encryption Process.

Each of these blocks goes through 8 rounds and one output transformation phase. In each of these eight rounds, some (arithmetic and logical) operations are performed. Throughout the eight rounds, the same sequences of operations are repeated. In the last phase, output transformation phase, only arithmetic operations is performed. At the beginning of the encryption process, the 64 bit plain text is divided in four equal size blocks and ready for round1 input. The output of round1 is the input of round2. Similarly, the output of round2 is the input of round3, and so on. Finally, the output of round8 is the input for output transformation, whose output is the resultant 64 bit cipher text [assumed as C1 (16 bit), C2 (16 bits), C3 (16 bits) and C4 (16 bits)]. As the IDEA is a

symmetric key algorithm, it uses the same key for encryption and for decryption. The decryption process is the same as the encryption process except that the sub keys are derived using a different algorithm[12]. The size of the cipher key is 128 bits. In the entire encryption process used total 52 keys (round1 to round8 and output transformation phase), generated from a 128 bit cipher key. In each round (round1 to round8) we use six sub keys. Each sub-key consists of 16 bits and the output transformation uses 4 sub-keys.

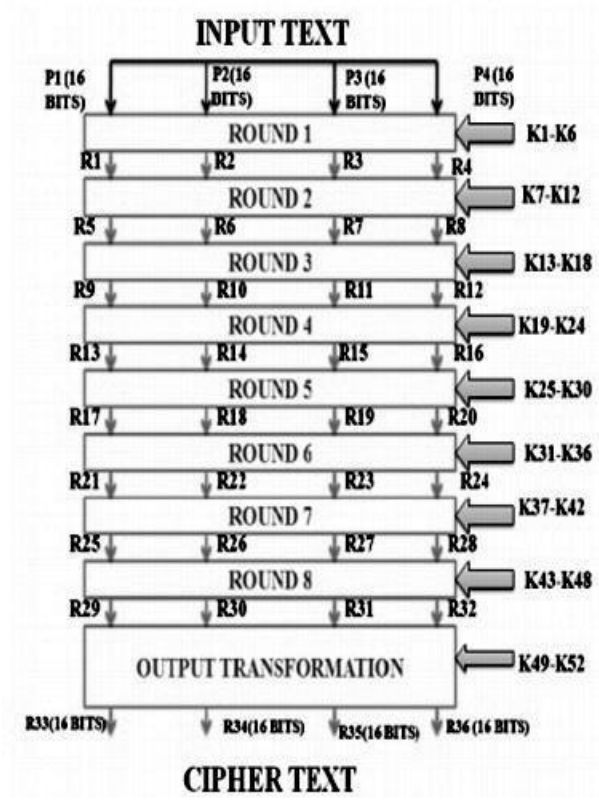

Fig.2 IDEA Encryption Process

It is a block oriented cryptographic algorithm, it encrypts blocks of messages with length 46 bit by using key with length 128 bit.

This algorithm work on principle that each bit in original text and each bit in the key effect on each bit in encoded text. This achieved by using three different operations, each operation is take two inputs from 16 bit and produce one output from 16 bit, these operations are:

1. Operation XOR bit to bit it symbolizes $\oplus$
2. Adding operation and take the rest of the total division on 2^{16} (A+B mod 2^{16}) it symbolizes $\boxplus$
3. Redemption operation and take the rest of the redemption on $2^{16}+1$ (A*B mod $2^{16}+1$) and Zero is equal to 2^{16} it symbolizes $\odot$. Figure (3) shows encryption algorithm planned IDEA notes two inputs, readable text 64 bit and the key 128 bit. This figure shows that the algorithm is consists of eight stages or repetitions followed by a final conversion. In each stage six partial keys is used and 16 bit except the last stage that uses four keys and so the sum of the partial keys will be 52 key. The right side of the figure 3 illustrate the generation of partial keys operation.

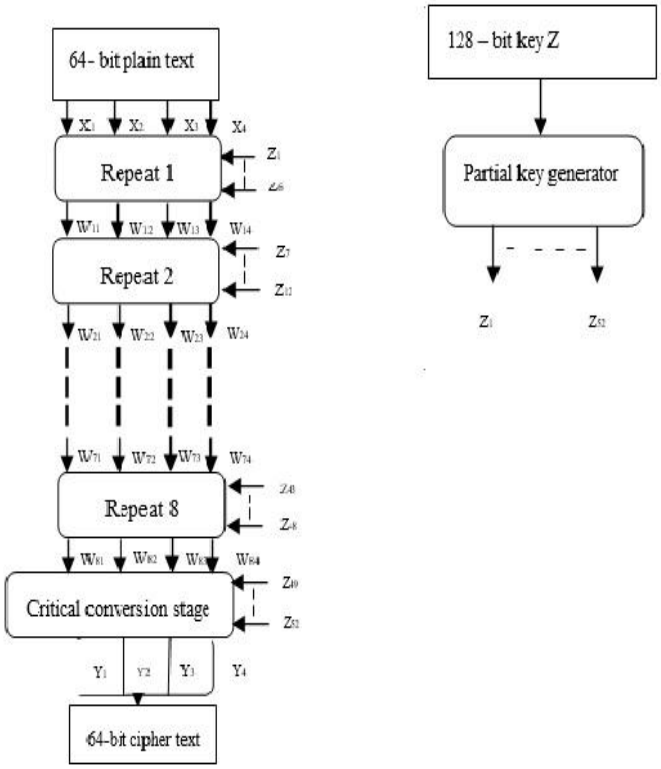

Fig.3 Generation of Partial Keys Operation

2.2.1 Details of A single Iteration

Figure (4) represents a repeated algorithm in all stages (with different input and different partial keys of course). The output for one stage is as follow:

$$W_{11} = \{z_6 \odot [[z_5 \odot [(z_1 \odot x_1) \oplus (z_3 \boxplus x_3)] \boxplus [(z_2 \boxplus x_2) \oplus (z_4 \odot x_4)]]\} \oplus \{(z_1 \boxplus x_1)\}$$

$$W_{12} = \{z_6 \odot [[z_5 \odot [(z_1 \odot x_1) \oplus (z_3 \boxplus x_3)] \boxplus [(z_2 \boxplus x_2) \oplus (z_4 \odot x_4)]]\} \oplus \{(z_1 \boxplus x_1)\}$$

$$W_{13} = \{z_6 \odot [[z_5 \odot [(z_1 \odot x_1) \oplus (z_3 \boxplus x_3)] \boxplus [(z_2 \boxplus x_2) \oplus (z_4 \odot x_4)]]] \boxplus [z_5 \odot [(z_1 \odot x_1) \oplus (z_3 \boxplus x_3)]]\} \oplus \{(z_2 \boxplus x_2)\}$$

$$W_{14} = \{z_6 \odot [[z_5 \odot [(z_1 \odot x_1) \oplus (z_3 \boxplus x_3)] \boxplus [(z_2 \boxplus x_2) \oplus (z_4 \odot x_4)]]] \boxplus [z_5 \odot [(z_1 \odot x_1) \oplus (z_3 \boxplus x_3)]]\} \oplus \{(z_4 \odot x_4)\}$$

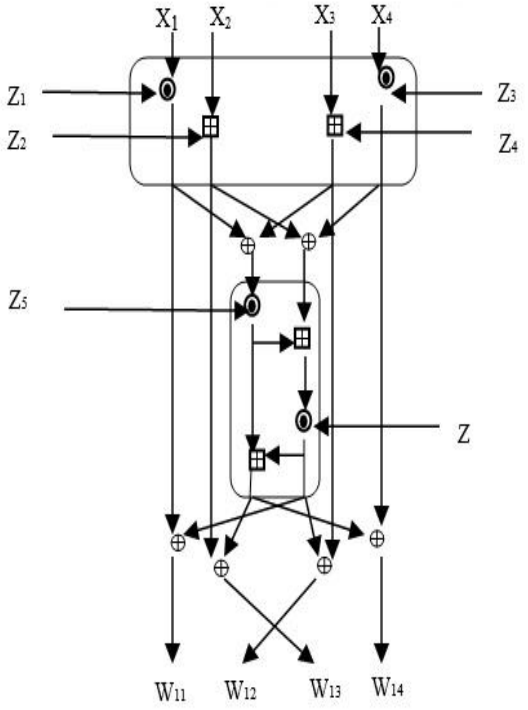

Fig.4 Represents A repeated Algorithm in All Stages

Figure (5) shows the ninth stage of the algorithm, called the final conversion stage

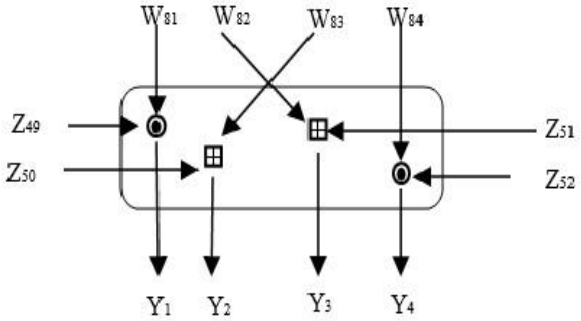

Fig.5 Ninth Stage of The Algorithm

2.2.2 Subkey Generation in IDEA

The first eight keys taken (Z_1,……,Z_8) directly from the primary key (Z_1 is 16 bit the first……. and so on), then doing rotational displacement process 25 bit for the primary key and the other eight keys taken (Z_9,……,Z_{16}) and the displacement process followed until generate all partial keys (52 keys).

2.2.3 IDEA Decryption

The decoding mechanism is similar to the encryption mechanism in essence but with different choosing for partial decryption U_1,……..,U_{52} that extracted from encryption keys. Table (1) illustrate the relationship between encryption keys and partial decryption keys:

$$Zi \odot Zi^{-1} = 1 \qquad -Zj \boxplus Zj = 0 \qquad \text{----- (1)}$$

Table 1. Relationship Between Encryption Keys and Partial Decryption Keys

	Encryption		**Decryption**	
Stage	**Partial keys number**	**The numbers of the fields corresponding to the key**	**Partial keys**	**Approval for**
Repetition1	$Z_1Z_2Z_3Z_4Z_5Z_6$	Z[1..96]	$U_1U_2U_3U_4U_5U_6$	$Z_{49}^{-1} – Z_{50} – Z_{51}Z_{52}^{-1} Z_{47}Z_{48}$
Repetition2	$Z_7Z_8Z_9Z_{10}Z_{11}Z_{12}$	Z[97..28;26..89]	$U_7U_8U_9U_{10}U_{11}U_{12}$	$Z_{43}^{-1} – Z_{45} – Z_{44}Z_{46}^{-1} Z_{41}Z_{42}$
Repetition 3	$Z_{13}Z_{14}Z_{15}Z_{16}Z_{17}Z_{18}$	Z[90..128;1..25;51..82]	$U_{13}U_{14}U_{15}U_{16}U_{17}U_{18}$	$Z_{37}^{-1} – Z_{39} – Z_{38}Z_{40}^{-1} Z_{35}Z_{36}$
Repetition 4	$Z_{19}Z_{20}Z_{21}Z_{22}Z_{23}Z_{24}$	Z[83..128;1..50]	$U_{19}U_{20}U_{21}U_{22}U_{23}U_{24}$	$Z_{31}^{-1} – Z_{33} – Z_{32}Z_{34}^{-1} Z_{29}Z_{30}$
Repetition 5	$Z_{25}Z_{26}Z_{27}Z_{28}Z_{29}Z_{30}$	Z[76..128;1..43]	$U_{25}U_{26}U_{27}U_{28}U_{29}U_{30}$	$Z_{25}^{-1} – Z_{27} – Z_{26}Z_{28}^{-1} Z_{23}Z_{24}$
Repetition 6	$Z_{31}Z_{32}Z_{33}Z_{34}Z_{35}Z_{36}$	Z[44..75;101..128;1..36]	$U_{31}U_{32}U_{33}U_{34}U_{35}U_{36}$	$Z_{19}^{-1} – Z_{21} – Z_{20}Z_{22}^{-1} Z_{17}Z_{18}$
Repetition 7	$Z_{37}Z_{38}Z_{39}Z_{40}Z_{41}Z_{42}$	Z[37..100;126..128;1..29	$U_{37}U_{38}U_{39}U_{40}U_{41}U_{42}$	$Z_{13}^{-1} – Z_{15} – Z_{14}Z_{16}^{-1} Z_{11}Z_{12}$
Repetition 8	$Z_{43}Z_{44}Z_{45}Z_{46}Z_{47}Z_{48}$	Z[30..125]	$U_{43}U_{44}U_{45}U_{46}U_{47}U_{48}$	$Z_7^{-1} – Z_9 – Z_8Z_{10}^{-1} Z_5Z_6$
Final conversion	$Z_{49}Z_{50}Z_{51}Z_{52}$	Z[23..86]	$U_{49}U_{50}U_{51}U_{52}$	Z_1^{-1}-$Z_2 – Z_3Z_4^{-1}$

2.3 Key Generation Algorithm

To speed up the key generation in the RSA algorithm [4] research describes a secure and fast generation of RSA Public. An algorithm that its input is a number of bytes taken optionally from the key generation file (it's a file with size 1M byte that generated randomly and sent to all network nodes in encrypted form by using RSA algorithm). From this optional number of bytes, this algorithm generates a key with length 128 bit used as the encryption key in IDEA algorithm.

The following working principle is as an example of this algorithm:

1- After taking number n bytes from generation keys file

2- a1,a2,a3,……,an increases the number of zeros where the number can be divided by 16 , this done as following:

   A- The algorithm calculates the value (n mod – 16) – 16 = i

   B- Add I zeros to the end of the series where the number of bits be divisible to 16 which (n+i) mod 16 =0

3- Divide this series of bytes to K partial series where K = (n+i) div 16 each string will made up of 16 byte .

4- Generate from these series 16 partial key with length 1 byte as follow :

The first byte of the first partial string is taken and do XOR process on it with the first byte from each partial string and the first partial key is get. The same process applies to the rest bytes so we get 16 partial keys each one of them content of 1 byte and the sum of these keys constitutes total keys that consist of 128 bit. It is possible to add some extensions to this algorithm whether in design and implementation, this make it a privacy algorithm because the security of the encryption algorithm lies in the key encryption.

3. KEY GENERATOR

Key generator that consist of 128 bit is done by :

1- Key generating algorithm.
2- Key generating file.

This done as follow:

From the key generation file, a set of bits are taken optionally according to the value of the displacement and a specified length, these will be sent in encrypted form within the header of each message where the displacement is indicated on the first byte number that will be taken from the file generating keys. The length is indicates to the number of selected bytes. Example: if the displacement = 150 & length = 225

This means that the key generating algorithm takes bytes as input starting from byte number 150 until byte number 374, thus huge possibilities of input bytes are chosen to the key generation algorithm and then get the same huge of different keys that used in the encryption algorithm IDEA .

It is worth mentioning that it must be an indicator with the message indicates to the completeness the message information and its content is not changed during the transferring operation whether the changing was intentionally or unintentionally. This indicator can called the message fingerprint. This fingerprint is calculated at the sending side and sent with the encrypted message and is counted at the receiving side to ensure that the message is received correctly. The message fingerprint calculated by one-way mathematical process that converts all encrypted message information into a short frame of information that we call the message's fingerprint. Changing one bit in the message leads to a different fingerprint for it. Message fingerprint calculating algorithm will be the one-way dependent type So that if a fingerprint identified, it is not possible to retrieve the message that generated the fingerprint. Often the length of the footprint is not less than 16 bytes.

4. SECERT LEVEL OF THE ENCRYPTION ALGORITHM

All encryption algorithms that generated mathematically are fractional theoretically. However, the coverage time distinguishes them from each other (the time required for fraction) this time that follows the computational energy that available by code analyzer but in decryption observance the following:

1- Decryption value should not be more than encryption information value.

2- The decryption time should not pass the information life time. Based upon whenever the computational energy developed for computers, the cover time reduced and its changes according to used analyses manner that may be:

   1- Random manner.

   2- Parallel processing.

Consider the same key is used for decryption all text. The attacker can use the random manner for decryption, namely experience and error. In this case, the number of possibility to try is (2^{128}) try where 128 is the key length that used in our algorithm and this is equal to $3.4 * 10^{38}$ try nearly. If the voyeur can experiment a key every one microsecond, the require time to experiment all keys is be 10^{25} year.

However, if the parallel processing used by using many processers, every processer will process the problem immediately. The one chip that can process one key per nanosecond (if that is possible) can experiment (10^{14}) key in one day and its need about (10^{24}) days for experiments all keys. That mean (10^{24}) chip needed working parallel in this rate until the detection completed within one day. Of course, completing such this machine is not easy if not possible.

The complications that mentioned were for one key used to encrypt many messages, but in this elaborate algorithm the key is changing permanently and randomly depending on the value selected from the key generation file, where the key can changing in every massage which every massage has its own key. Therefore, even the voyeur can detect a cryptographic key for a message does not mean the encryption algorithm detected.

If someone want to detect massage content, he must:

1- Obtain the key generation file and decrypt it. Encrypted length and displacement RSA

2- He must know the tow values that put in the message header as well as breaking the key generation algorithm.

3- Brake the encryption key IDEA by an algorithm.

If it is very difficult to succeed in one of those steps, how can succeed in all steps and break all the algorithms together?

5. STAGES OF IMPLEMENTATION OF THE ENCRYPTION PROGRAM

Before start encrypting the messages by implementing the algorithm, the key generation file that generated randomly on all network branches is distributed by using RSA algorithm in order to avoid revealing this file. This file distribute ones time unless changing it from time to time then another random file is generated and send it in the same way ,this file can resizing because its size does not affect the generation of keys, as for encryption, it done as following:

1- The value of the displacement is determined and the length required to generate the encryption key.

2- The key generation algorithm based on the key generation file and the displacement and length values by generation the random key 128 bit that will used in IDEA algorithm.

3- The clear message is encrypted by the encryption algorithm IDEA that used encryption key which was previously generated.

4- Encrypting the value of the displacement and length by RSA algorithm.

5- The encrypted value of both the displacement and the length added to the encrypted message header.

6- Calculate the message fingerprint with a special algorithm and place it in the message header.

7- Eventually the message sent to the intended destination.

Figure (6) illustrated the full structure of this algorithm:

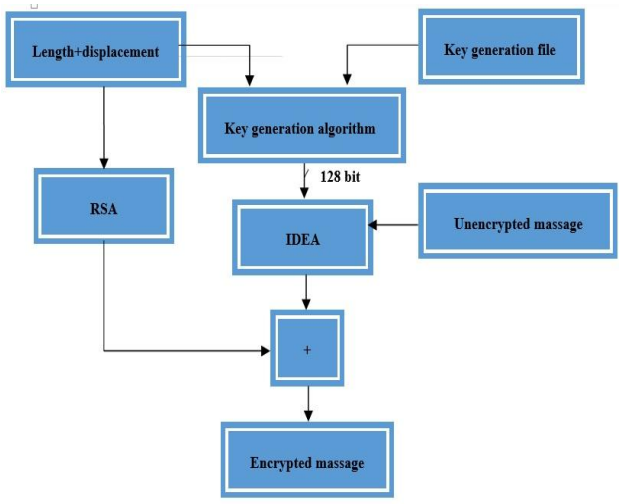

Fig.6 Structure of IDEA and RSA

6. THE DECRYPTION

1- After the message received, the decryption algorithm separates the message from the letterhead and verifies it through the message's fingerprint.

2- Encrypt the header that content displace and length by RSA algorithm.

3- The key generation algorithm generated the key 128 bit based on displacement and the length values.

4- Eventually the encrypted message is entered into IDEA algorithm that uses the generator key 128 bit to obtain the unencrypted message.

Figure (7) illustrated the schematic of the stages of the decoding algorithm.

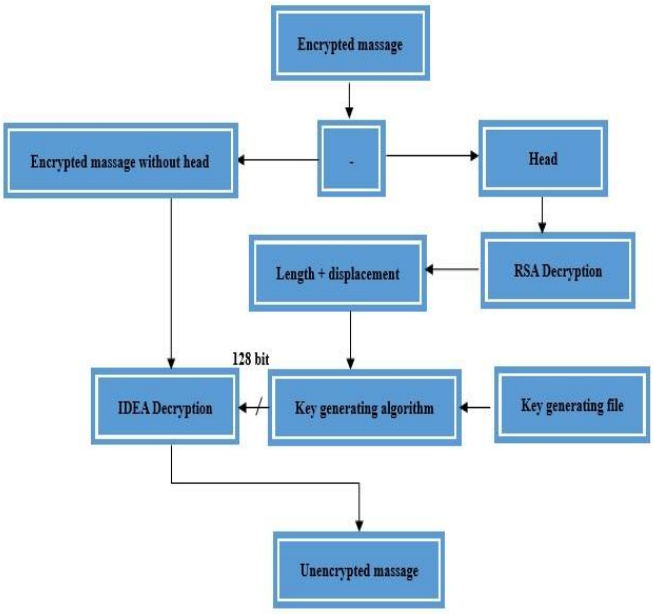

Fig.7 Stages of the Decoding Algorithm

7. CONCLUSION

The most important types and methods of encryption techniques used are discussed in the present. RSA Rivest-Shamir-Adleman (RSA), International Data Encryption Algorithm (IDEA) have been used to test the quality of security massage. Test lab showed the following results: an algorithm consisting of more than a partial algorithm has been adopted which has increased confidentiality and complexity and overcome the key distribution problem. The algorithm included some original ideas, in both design and implementation, which made it special. Because the secret lies in the encryption key , there is a generating key possibility and changing it afar of system designer for ensure Secrecy and limitation of liability. Encryption keys that used are random and hard to detect. In addition, the suggestion of using RSA algorithm along with the IDEA algorithm is provided with improved security and advanced encryption efficiency. In addition, it extended the key space without any extra running time, which can be considered to be promising in the field of information and communication technology today. In addition, the integration between the improved International Data Encryption Algorithm (IDEA) and Rivest-Shamir-Adleman (RSA) with two layers of protection, the first of which is the encryption with an improved (RSA) algorithm and the second increased confidentiality and complexity. Also in the future this method can developed by using 2 keys instead of 1 key in each algorithm and use 256 bit instead of 128 bit this will make the method more security. The complications that mentioned used one key to encrypt many messages, but in this elaborate algorithm the key is changing permanently and randomly depending on the value selected from the key generation file where the key can changing in every massage which every massage has its own key.

8. REFERENCES

[1] William, S. 2006. pearson prentice hall, 4th edition "cryptography and network security" .

[2] William, S. Ph.D. 1995. prentice-Hall .Inc , New Jersey, "Network and Internet work security principles and practice".

[3] Bruce, S., John, W. and Sons, I. 1994. "Applied cryptography –Protocols" (Algorithms, and Source code in C).

[4] Aman, K., Sudesh, J. and Mr. Sunil M. 2012. "Comparative Analysis between DES and RSA Algorithm's", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 7, pp. 386-391, July.

[5] Xin, Z. and Xiaofei T. 2011. "Research and Implementation of RSA Algorithm for Encryption and Decryption", the 6th International Forum on Strategic Technology, pp. 1118 – 1121.

[6] Yunfei, L., Qing, L. and Tong, L. 2010. Design and Implementation of an Improved RSA Algorithm, International Conference on E-Health Networking, Digital Ecosystems and Technologies.

[7] Ajay, K. and Bansal P.K. 2012. "Comparison of Various Encryption Algorithms and Techniques for Secured Data Communication in Multinode Network", International Journal of Engineering and Technology, Volume 2 No. 1, pp. 87-92, January.

[8] Popovych, R. "Cryptoanalysis of RSA system of enciphering with public key", Modern Problems of Radio Engineering, Telecommunications and Computer Science, Vol.2, pp. 301- 302.

[9] Sriram, R. and Marimuthu, K. 2011. "Designing an algorithm with high Avalanche Effect", IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.1, pp. 106-111, January.

[10] Schneier B. 1994. " Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish) Fast Software Encryption", Cambridge Security Workshop Proceedings (December 1993), Springer-Verlag, pp. 191-204.

[11] Milad, B. Mohammad, R. Omid, S. Mojtaba, A. and Mohammad**,** S. 2010. "Novel Approach witch is the Secure and Fast Generation of RSA Public and Private Keys on Smart-Card", 978-1-4244-6805-8/10/$26.00 © IEEE.

[12] Basuin, S. 2011. "International data encryption algorithm (idea) – a typical illustration", Journal of global research in computer science (JGRCS), vol. 2, no 7.