The FastDES: A New Look of Data Encryption Standard

Muhammad Nawaz Khan
School of Electrical Engineering & Computer Science (SEECS), National University of Science & Technology (NUST),
Islamabad, Pakistan.

Ishtiaq Wahid, Atual Aziz Ikram
Department of Computing & Technology,
Iqra University
Islamabad, Pakistan

ABSTARCT
The FastDES algorithm is based on Data Encryption Standard (DES). In FastDES, the one round function concept of DES are used in three different customs to produce a new fast and more efficient secure algorithm. The DES operates on half of the input data, 32-bit of chunk out of 64 bits, while FastDES take block of 32-bit and work on it simultaneously. The encryption start from 32-bit of data and 32-bit of key bits, which passes from three consecutive rounds and 32-bit of cipher is produce. This algorithm is fast, so it can be used in many further securities mechanisms like Hash functions (MD5, SHA series, HMAC) and modes of the operation like CBC, CFB etc. Because of its fast operations it is also recommended to use with other security applications. The DES round function provide a best confusion-diffusion structure of Substitution-Permutation Network (SPN). The FastDES based on same design criteria of S-boxes and permutation functions of DES. Like 3-DES with two keys and 3-DES with three keys, the FastDES can use 3-FastDES with 2 different keys and 3-FastDES with 3 different keys, which not only enhanced system security but also thwart the cryptanalysis for brute force and meet in meddle attacks.

General Terms
Security Algorithm, Communication Security, Data Encryption Standard.

Keywords
FastDES, DES, permutation, Substitution, Lucifer, CBC, CFB, SHA-1, feistal, non-linear, mode of operation etc.

1. INTRODUCTION
Simple Feistel cipher is works on miniature amount of data with small size of key (2^n. 2^4=16-bits). Such ideal ciphers are easily comprehensible but this concept only provides basis for further encryption processes and these are not use in practical applications. These ideal ciphers have some inherent weaknesses like no error detection and ease of statistical analysis [1]. These problems can be removed in security algorithm by using large block size of data (2^{32}/2^{62}/2^{128}/2^{256}……). The Feistel cipher, however, provide the concept of product cipher by introducing the concept of confusion and diffusion i.e. substitution and permutation respectively. The Feistel cipher is based on some design criteria including large block (2^{64}/2^{128}), large key size (2^{64}/2^{128}), number of rounds (3/5/8/10/16/……..) and complexity of key generation algorithm. Others important features is the operation of each round function and two other most important factors are fast encryption/decryption and ease of analysis.

On the basis of the Feistel cipher the IBM developed LUCIFER in 1970 [2] working on 64-bit block of data, using the key size of 128-bit. The IBM submitted the refine form of LUCIFER to NBS in 1973 [3] and accepted as Data Encryption Standard (DES). The standard DES uses short 56-bits of key length and 64-bits of block size. For many years the DES cipher was the standard used by many national and international organizations until DES is considered to be an out-dated and easily hacked encryption option. An improvement in DES is Triple Data Encryption Standard (3-DES). The 3-DES is simply the DES with three times in progression. In 3-DES, the short key problem of 56-bits been solved by increasing the key length up to 168 bits. This longer key length thwart against a brute force attack. 3-DES is still being widely used in today financial transactions and is still considered being secured. Another variation of security algorithm is "Rivest Cipher," or RC2 encryption algorithm. RC2 uses a 64 bit block size and variable key length. 16 round RC2 algorithm is based on source-heavy Feistel network. RC2 is secure for long time but now easily broken by chosen plaint text attack. Ron Rivest improve the original RC2 into a modified form known as RC4, also known as ARC4 or ARCFOUR. Today RC4 is used in SSL and WEP wireless applications. Currently RC4 consider as a software stream cipher having good level of security. Another development is Blowfish cipher developed by Bruce Schneider. Blowfish is a symmetric key block cipher with 64-bits block size and variable key length. The key in Blowfish can vary from 32 bits to 448 bits in length. Blowfish considered being secure and fastest block cipher however it has been replaced by Twofish and Rijndael due to its small 64 bit block size. Blowfish is available in public domain. After the Blowfish, Bruce Scherier developed another algorithm named Twofish. Twofish is actually improvement in Blowfish. It is symmetric key block cipher but uses a larger block size of 128 bits and variable key sizes up to 256 bits. Twofish is faster than Blowfish yet slightly slower than Rijndael for 128 bit keys [4][5][6][7][8][12][14].

After all in 2002, it is instantly needed to replace all previous and legacy security algorithms into new and secure encryption algorithm. For this reason a new algorithm known as "Advanced Encryption Standard (AES)" is introduced. AES [9] is also known as Rijndael, because of its developers, two Belgian cryptographers Vincent Rijmen and Joan Daemen. AES is the most accepted and secure symmetric key block cipher used today. AES use block size of 128 bits with a variable key length of 128 bits to 256 bits [15].

Instead of the initial criticism on DES, the DES with 56-bit effective key length and 64-bit of data becomes as standard and widely deployed until Advance Encryption Standard (AES) are came to the screen in 2002 [9]. There are many other security algorithm used with parallel, each of them have their own application and uses. The AES [9] [10] and NESSIE [14] are famous block cipher algorithms for their

lightweight and fast key-schedule algorithms. The substitution boxes and number of rounds are the heart of the process. The logical operations are performed by xor and non-linearity added by S-Boxes. Table.1 shows the summary of the cipher algorithms.

Algorithm	Key size	Block size	Number of round	Date of creation
DES	56-bits	64-bits	16	1973
3-DES	168-bits	64-bits	48	1976
RC2	64-bits	64-bits	16	1987
RC4	Variable	Variable	Changeable	1987
Blowfish	128-bits	64-bits	16	1993
Twofish	128/192/256	128	16	1993
AES/Rijndeal	128/192/256	128	10/12/14	1998

Table .1

The variants of DES (DES, 3-DES), AES and many other algorithms are used in the world. Some organization and national organization built their own standards by using successor of DES with collaboration with other algorithms like blowfish, Twofish, MARS and CAST etc. Single or combination of these algorithms is used in iterative fashion to increase security level. But the iterative use of these algorithms increases the overhead at encryption and decryption side. These bulky calculations increase the security level and useful for isolated systems like disk encryption. But for ensuring good security, the iterative form is very useful. Mostly used iterative forms Block Cipher are, Electronic Code Book Cipher (ECB), Cipher Block Chaining Mode (CBC), Cipher Feedback Mode (CFB) and Output Feedback Mode (OFM). Although these iterations increases security but also increase overhead which not feasible for certain quick and fast applications. Therefore, a tradeoff is required at certain level between offered securities and calculation overhead. DES can be used in 3-DES form or with modes of operation (CBC/CFB) for certain quick and fast application. But 3-DES and modes of operation needed bulky calculations. Here we use the 3-FastDES with modes of operations to reduce the overhead and increase the level of security.

The inherent and main security feature of DES is S-boxes for substitution. Here in FastDES the same S-boxes are used for non-linearity to thwart the cryptanalysis attacks. The FastDES uses the concept of one round function of DES for three round of algorithm having encryption/decryption process. The DES operates on less number of bits than the enter bits. In 64-bits block of data, the operation will perform only on half of the data i.e. 32-bit of data and entered key is 64-bit but only 48-bit key are used in operations. Interesting thing about FastDES, is the use more bit operation than the enter bits. So DES use the concept of subtraction (enter data is greater than the operational data), but FastDES uses the concept of multiplication (less bits are entered and operations are performed on more bits). These operations are based on the same S-boxes for substitutions and permutation tables.

Here we proposed the FastDES, 32-bit cipher algorithm. The algorithm is explained with the reference of Data Encryption Standard. Rest of the paper is arranged as, section-2 of the paper consist on system design, section-3 have FastDES algorithm, in section-4 the system design diagrams and in section-5 the permutation/substitution tables are explained, in section 6 main features the system are discussed and conclusion remarks are added in section-7.

2. DESIGN OF THE SYSTEM

The FastDES starts when 32-bit of data block and 32-bit of key is entered for encryption. The 32-bit data block is passed from expansion/permutation table which expand the 32-bit block into 48-bit block for further operations. The expansion/permutation table is the same used in DES round function. This table just convert the 32-bit into 48-bit as in the DES round function (4*8 == 6*8). The 32-bit key is also expended into 48-bit by applying the same expansion/permutation key table. Then these 48-bit of key is also passed from permuted choice-1 for further permutation. These key bits are now XOR with the 48-bit block of data to produced 48-bit block of output data. Now the 48-bit block of data is passed from substitution/choice (S-box-1), which consists of eight S-box of DES, for producing more confusion. The same DES S-boxes are used here, which produce more secure and non-linear approach to our encrypted data. The 32-bit output block from S-boxes is now passed into permutation function (P), for producing more diffusion at some extent. So with this one round function is completed.

The 32-bit block is now entering into another round of FastDES, which have the same structure as the first round, but some variations. The 32-bit block is now passed from the expansion/ permutation table, which expends the 32-bit into 48-bit of data. But the 48 key bits coming from the above expansion/permutation key table-1, is passed into expansion/permutation key table-2, which have different structure from the expansion/permutation table from original DES. Each of them with some operation is mention below in the paper. The 48-bit key bits now passed into permuted choice-2, for more permutation, this time the same permutation table is used in the previous round. The 48-bit data block and 48-bit key bits are XOR, we get 48-bit block of data. This block is now passed into substitution/choice, having the DES S-box, which convert the 48-bit data into 32-bit, produced more confusion as in the first round. 32-bit block is now passed into permutation table as in the first round, results 32-bit block of data from second round.

The third round have the same structure as the previous only the expansion/permutation for key bits are changed by expansion/permutation key table-3 and then passed into permuted choice-3. The same operation is performed (XOR, S-boxes, permutation) and the 32-bit of encrypted data now ready for storing or in suitable form for network communication.

An XOR operation is between 32-bit block of data and 32-bit of key bits. After passing into s-box at the end is optional. This 32-bit block, now used in many way like it can be used as operational modes (CBC, CFB,…) and also in massage authentication codes(HMAC,…).

In FastDES, the decryption is performed in the same way as the encryption but order of sub key generation is work in reverse order. The 32-bit block of cipher is enter into FastDES with 32-bit of key. The 32-bit block is first passed into permutation table then passed into substitution/choice S-box, which results as 48-bit block of data. The 48-bit sub key coming from permuted choice-3, is XOR with the data and passes into expansion/permutation table in reverse order, which produce 32-bit of data. The same three rounds work in

the same manner. The FastDES decryption is the same as it's encryption but works as in reverse order.

3. THE FastDES ALGORITHM

This algorithm used for 32-bit input block of data. The key size is 32-bit, avoiding weak keys. The operations actually show the permutation-substitution network (SPN). The Expansion/Permutation table expends the 32-bit value into 48-bit value. The Expansion/Permutation key tables expend the 32-bits of key into 48-bits. These 48-bits of key as well as data bits are logical operate (XOR) each other. The permuted bits now passed from Substitution/choice (S-box). The S-Boxes convert these 48-bits into 32-bits which increase further substitution. These 32-bits now passed into Permutation (P) tables for further permutation and the process is recycle with other S-Boxes.

1: Cipher [Plain text (32-bit), Round (3), Round Key (48), Cipher text (32)]

2: **ITERATE** (Rounds == 1 to 3)

DO {

INPUT (32-bit input, 32-bit key)

Permute parallel **Pu** (Plain text [32], Key bits [32])

Expand Ex (32 bit plain text → 48 bit plain text)

Expand Ex (32 bit key → 48 bit key)

3: Exclusive OR, 48 bit of plain text and 48 bit of key, this operation is bit by bit.

$\sum_1^{48} Ex$ Plaintext XOR $\sum_1^{48} Ex$ key bit

REPEAT (i=1 t0 48)

DO {

Plain text **XOR** Key bits

END REPEAT

}

4: Substitute these 48 bits (Ex) into S-Box, here they convert into 32 bits, which increase non-linearity.

$\sum_1^{48} Ex$ (48 bits) →→→ Permute (32 bit)

5: Permute this 32 bit into another permutation table. The operation is bit by bit, which increase the non-linearity.

$\sum_1^{32} Pu$ →→ $\sum_1^{32} Pu$

REPEAT (i=1 t0 32)

DO {

Permute (32 bit)→→ permuted (32 bit)

END REPEAT

}

6: Rounds are determined for increasing security level; the FastDES recommended mode is three rounds.

REPEAT IF (Round! = 4)

DO ITERATION

END IF

}

END REPEAT

}

The algorithm consists of six steps. The iterative nature of the algorithm increases the security level.

4. FastDES DESIGN ALGORITHM

The FastDES encryption process works in a sequential way as shown in diagram.1. The diagram clearly shows each operating step. The number of bits involves in operation and operation details are also clearly mentioned. The tables involve in each operation is also explained in the paper. The diagram clearly maps the logic and methodology of the algorithm. This is a systematic approach for encryption. The decryption is also the same but opposite in working process, key algorithm, input and output.

5. MAIN TABLES USED IN FastDES

The following are the main building blocks for FastDES, also known as tables.

Table 2, is the Expansion/Permutation table for first round use for data as well as for key bits in first round of FastDES. This is the same table used in DES for expansion 0f 32 bits into 48 bit values. This table extends 32 bits (4*8 bits) into 48 bits (6*8 bits). The logic and converting mechanism is explained in [DES]. The number in table indicates the position of the bits.

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

Table. 2

The table 3, is the Expansion/Permutation table is used in FastDES only. This table is part of key algorithm for generation of sub keys for each round. We made some changes with the original above DES table. This is Expansion/Permutation key table-2.

31	1	2	3	4	6
3	5	6	7	8	10
7	9	10	11	12	14
11	13	14	15	16	18
15	17	18	19	20	22
19	21	22	23	24	26
23	25	26	27	28	30
27	29	30	31	32	2

Table .3

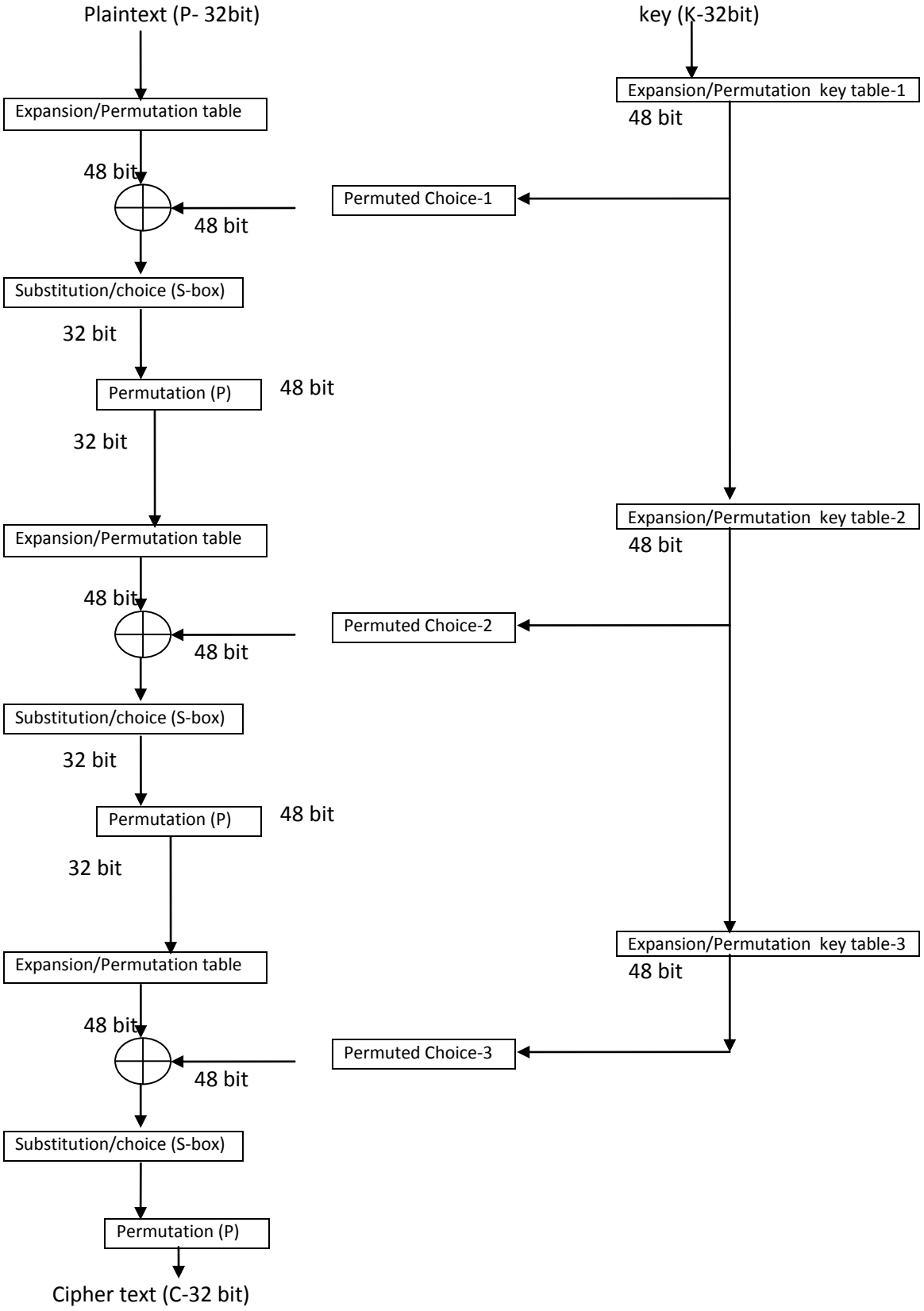

Diagram.1. The main operating model of FastDES.

The same operation applied in table 4, for producing the third expansion/permutation key table-3 for sub key generation.

30	1	2	3	4	7
2	5	6	7	8	11
6	9	10	11	12	15
10	13	14	15	16	19
14	17	18	19	20	23
18	21	22	23	24	27
22	25	26	27	28	31
26	29	30	31	32	3

Table .4

The following is the Permutation table (P). This table is responsible for further permutation of bits positions. The same permutation used as in DES.

16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

Table .5

The FastDES Substitution choice/S-boxes are the same used in DES. Eight boxes with enormous non-linear approach provide a more secure and a decent way to FastDES. Although, working on half chunks of the data block, then permutation of the data, many rounds of algorithm and a strong key management algorithm provides a secure approach to DES. But the most important thing is S-boxes of DES (3-DES), here FastDES also use the same S-boxes for its operations. All these S-boxes are universally known and therefore, not included in this paper.

6. FastDES MAIN FEATURES

A Fast Algorithm; FastDES is mainly design for high speed. It is highly recommended for further use with other security services like modes of the operations (CBC, CFB, etc.) and MAC codes. Actually 3-DES becomes little bit slowly in many practical scenarios and its uses in iterative algorithms effect system efficiency. For that reasons, 3-DES is not implemented in such real time applications. So 3-FastDES can use in those situations. 3-DES with sixteen rounds (48 rounds) structure is a complex process but here the 3-FastDES provide a non-linear and secure encryption with simplicity. It can be implemented in parallel manner in single chip for enormous speed.

Security; 2^{32} is not so much massive number for today's computer to break, but still it not so easy to break without special programmable chip or device. The 3-FastDES provide 2^{96} key length which is enough secure. Certainly it resists to brute force not like 3-DES but still secure than general block cipher algorithm (early block cipher). The FastDES can use like 3-DES, two key or three key FastDES resist to brute force attack. And also FastDES round functions provide a framework that not only thwart the cryptanalyst from frequency analysis but also from other known statistical attacks.

Avalanche Effect; As DES exhibits strong avalanche effect, that is when a small change in either in key or in the plain text will results a significant change in the cipher text. One bit change in DES key or plain text results as 50% change in cipher text. The FastDES may also exhibit avalanche effects nearly to DES. We show an example at the end, avalanche effect is important for security perspective.

Key Scheduling Algorithm; Strong and secure key scheduling are very important in any block cipher designing. Any key algorithm must satisfy strict avalanche criteria and bit independence criteria [13]. The Feistel cipher use the key for producing sub keys for each round function, by applying an algorithm. In FastDES, an algorithm for sub key generation is implemented, which use the same concept of DES, but with different manner. Instead of this, one should avoid from using the weak keys, like 01010101, 00000000, 1F1F1F1F etc. because these weak keys also leaks out some information from plain text to cipher text. Here in FastDES, it is trying that no weak key is generate and if generates it must be wedded out.

7. CONCLUSION

The FastDES main design feature is multiplication rather than subtraction. As FastDES operates on more bits than the entered bits for encryption. While the original DES work on subtraction because DES operates half of the entered bits. The FastDES works in three rounds in a chronological fashion. The block of the plaintext (32-bit) is directly passed into three rounds of FastDES, with key (32-bit) to produced 32-bit encrypted cipher text. Each round follows the substitution and permutation steps which is a real shape of the confusion and diffusion respectively. The SPN is the basis for all block cipher algorithm. The DES S-boxes, permutation and expansion tables are used in FastDES for non-linearity, strong avalanche effects and strong key scheduling algorithm. All these features provide good security to FastDES. But the main rationale of FastDES is fastness and used in those vicinity where speed is important.

8. ACKNOWLEDGMENTS

We are very appreciative to Almighty Allah; whose elegance and blessed mercy enabled us to complete this work with full devotion and legitimacy. We are grateful to Dr. Ata ul Aziz Ikram, Associate Professor & Head of the Department, Department of Computing & Technology, Iqra University Islamabad, for their invaluable support and guidance throughout this research work.

Lastly, We also want to thank our friends (Ayub Khan and Muhammad Faisal) and family for their encouragement; without whose support we could not have lived through this dream of ours.

9. REFERENCES

[1] Feistel, H., Notz, W. and Smith j. "Some Cryptographic techniques for Machine to Machine Data Communication." IEEE 1975

[2] A. Sorkin, (1984). LUCIFER: a cryptographic algorithm. *Cryptologia*, **8**(1), 22--35, 1984

[3] Federal information processing standards publication (fips pub 46-3), u.s. department of commerce/National Institute of Standards and Technology, DES.

[4] 9. Whitman, Michael and Herbert Mattord. Principles of Information Security. Boston: Thomson Course Technology, 2005.

[5] Cryptography, http://www.ssh.com/support/cryptography /introduction/algorithms.html

[6] Katos, V.," A Randomness Test for Block Ciphers. Applied Mathematics". Elsevier Publications. (2005).

[7] Whitman, Michael and Herbert Mattord. "Principles of Information Security".Boston: Thomson Course Technology, 2005.

[8] Campbell, Paul and Ben Calvert and Steven Boswell. "Security+ Guide to Network Security Fundamentals." Boston: Thomson Course Technology, 2003.

[9] Federal Information Processing Standards Publication 197 Announcing the "ADVANCED ENCRYPTION STANDARD (AES)" November 26, 2001.

[10] Joan Daemen, Vincent Rijmen "AES Proposal: Rijndael" 3/9/1999.

[11] Portfolio of recommended cryptographic primitives NESSIE consortium? February 27, 2003.

[12] A book on "Cryptography and Network Security, Principles and practices" by William Stilling. 4th edition.

[13] "Simple and Effective key Scheduling for Symmetric Ciphers" by Adam, C. SAC 1994.

[14]. A book on "Cryptography and Network Security" by Behrouz A. Forouzan. Copyright © The McGraw-Hill Companies, Inc.

[15] "FOX : a New Family of Block Ciphers" by Pascal Junod and Serge Vaudenay. Ecole Polytechnique F ed erale de Lausanne (Switzerland)