Volume 2, Issue 12, December 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A Study of TORDES with other Symmetric Key Algorithms

Ajay Bhushan[*]
Department of I.T.
G.C.E.T. Greater Noida (U. P.)

Ajeet K. Bhartee
Department of Computer Science,
G.C.E.T. Greater Noida (U.P.)

Pawitar Dulari
Department of Physics
Govt. College, Indora (H.P.) India

Abstract— The selective application of technological and related procedural safeguard is an important responsibility of every cryptographic algorithm in providing adequate security to its electronic data systems. This paper specifies TORDES including its primary part and cryptographic engines and also provides a comparison of TORDES with other algorithms namely MODDES, DES, TDES and AES in terms of memory requirement and encryption-decryption time.

Keywords— TORDES, DES, TDES, AES, MODDES, Mirror image, Transformation

I. INTRODUCTION

For secure communication over computer network, data can be protected by the method of encryption (Stallings, 2007). Encryption converts original text data by using some encryption algorithm with its key. Authorized user accesses the key and decrypts the data. Encryption is the fundamental tool for protection of data over computer network (Tanenbaum, 2004). Most famous and widely used approach for data security is "Cryptography" which is the practice and study of hiding information. 'Cryptography' is derived from the Greek word *kruptos*, meaning "hidden". It is normally also referred as "Encryption", which is used to disguise data, making it unintelligible to unauthorized observers. Decryption is the reverse process i.e. moving from the unintelligible cipher text back to plain text. The main advantage of cryptography is that communication between both sending and receiving ends remains inconceivable by anyone who might be listening. Cryptography helps us to achieve three main security goals i.e. availability, confidentiality and integrity of the information. One example of cryptographic application is Enigma machine used by Germans during World War II to communicate safely within their defense forces and avoid eavesdropping. One way to classify cryptography is the key mechanism. Key can be defined as the rules which are Responsible for converting plain text into cipher text. Depending upon the key, cryptography can be alienated into two chief categories: Secret Key Cryptography and Public Key Cryptography. Secret Key Cryptography uses same key (single key) for encryption and decryption of data, while Public Key Cryptography uses two mathematically related keys, one for encoding data at sending end and another for decoding data at receiving end. The former approach is simpler but distribution of secret key is the chief concern. The later approach is a bit complex due to two keys: private key and public key. The sender can use public key (available to everyone) of any individual or organization to send data and that data can only be decoded using the private key which is kept confidential (with the receiving person or organization) and thus making it more secure and widely accepted across the world. Another nomenclature of cryptography takes in account the number of characters read in a single pass. First category is known as Block Cipher, which inputs a block of digits/ characters in a single pass and encodes it simultaneously. However this technique is faster but it may produce identical cipher texts for the same plaintext every time it is encrypted. Second type is known as Stream Cipher, which encrypt plain text digits/characters one at a time. This technique is a bit slower than Block Cipher but is more secure as it produces different cipher texts every time. Here we have compared the performance of a new algorithm TORDES with other algorithms like DES, TDES, AES and MODDES.

II. IMPLEMENTED ALGORITHM

A. DES

This algorithm is designed to encrypt and decrypt block of data consisting of 64 bits under control of 64 bit key (Stallings, 2007). Decryption is done by using the same key as for encryption, but with the schedule of addressing the key bits altered so that the decryption process is the reverse of the encryption process. A block to be encrypted is the matter to an initial permutation IP, then to a complex key dependent computation and finally to permutation which is inverse of the initial permutation IP, then to the complex key-dependent computation and finally to a permutation which is the inverse of the initial permutation IP^{-1}. The key-dependent computation can be simply defined in terms of a function f, called the cipher function and function K_s, called the key schedule. Implementation of DES will depend upon the application and environment.

B. TDES

Triple DES is simply another mode of DES operation. It takes three 64-bit keys for an overall key length of 192 bits. TDES is the answer of many of the shortcomings of DES. It also has the advantage of proven reliability and longer key length that eliminates many of the shortcut attacks that can be used to reduce the amount of time it takes to break DES.

Figure1(a): Encryption Operation for TDES

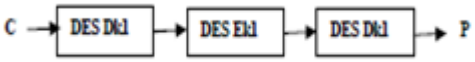


Figure1(b): Decryption Operation for TDES

C. AES

AES is a block cipher with block length of 128 bits. It allows three different key lengths of 128, 192 and 256 bits. Encryption consists of 10 rounds of processing for 128 bit keys, 12 rounds for 192 bits keys and 14 rounds for 256 bit keys. Each round of processing consist of one single byte based substitution step, followed by what is known as a row wise permutation step, followed by a column based substitution step followed by the addition of round key. So each round consists of four steps:

1) Substitution of steps
2) Shift rows
3) Mix column
4) Add round key

D. MOODES

MODDES is partial symmetric key algorithm. It uses 32 bit key with multiple binary operator and same delimiters chosen randomly from predefined stacks along with a code of sequence (Gope et al., 2009). Here with the same partial symmetric key and for same plain text the code sequences and cipher texts generate from different processes are different. This algorithm is not fully dependent on the key. As such, if the key value is known, then also it is impossible to decipher it without the knowledge of decryption algorithm as well as code sequence generated from that particular processing.

E. TORDES

TORDES is a block cipher algorithm (Bhushan et al., 2012). It is a unique and independent approach which uses several computational steps along with string of randomized operators and delimiter selections by using some suitable mathematical logic with transformation and mirror image operation. It is specially designed to produce different cipher texts by applying same key on same plain text. It is one of the best performing partial symmetric key algorithm particularly for the text message in its class. It also safeguard against various attacks like Brute-force because it is not fully dependent on the key and code cannot be deciphered by applying all possible combinations of keys. The following information invariably used in TORDES for encryption techniques.

1) 32 bit key.
2) Code sequence string generated from a particular process (Multithread).
3) Transformation of string.
4) Mirror image of string.
5) Lookup Table
6) Randomized delimiter string

This shows that the security of text data is not only depends upon key value. This really increases the security of text file.

III PERFORMANCE AND EVALUATIONS

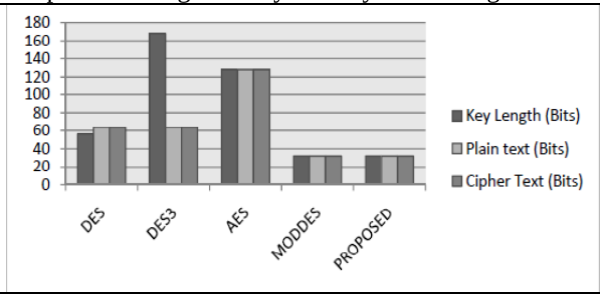
A. Memory Space

The following table shows that memory requirement of proposed algorithm is lesser as compared to considered. From table 1, it is evident that the proposed algorithm is having lesser memory requirements compared to other algorithms. Basically the encryption time increases as the key length increases.

Table1:showing memory used by various algorithm

Algorithm	Key length (bits)	Plain text (bits)	Cipher text (bits)
DES	56	64	64
DES3	168	64	64
AES	128	128	128
MODDES	32	32	32
PROPOSED	32	32	32

Graph1: showing memory used by various algorithm



B. Encryption/Decryption execution time

The proposed method has been implemented based on multi threading concept, which helps in efficient utilization of CPU. Hence encryption and decryption time is very optimum as compared to existing methods. Below table and graph shows the time (seconds) required for encryption and decryption of text file of various size .The system for this test is P(IV) 2.4GHz processor512 Mb RAM, Window XP as operating system, visual studio 2008 as platform to develop the piece of software. The following parameters have been tested

1) Encryption time
2) Decryption Time
3) Total Encryption plus Decryption Time.

Table 2: Encryption Time with various algorithm

Input (Bytes)	DES	TDES	AES	MODDES	TORDES
20527	2	7	4	12	14
36002	4	13	6	39	40
45911	5	17	8	66	67
59862	7	23	11	119	118
69642	9	26	13	167	167

Graph 2: Showing Encryption Time with various algorithm

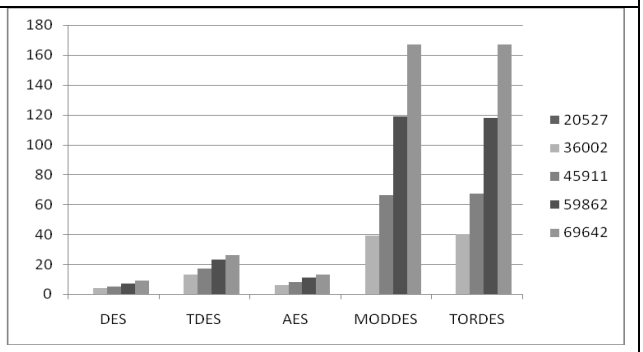


Table 2: Decryption Time with various algorithm

Input (Bytes)	DES	TDES	AES	MODDES	TORDES
20527	17	58	62	2	2
36002	30	99	94	3	3
45911	41	130	125	3	3
59862	52	181	174	3	3
69642	69	201	200	3	4

Graph 3: showing Decryption Time with various algorithm

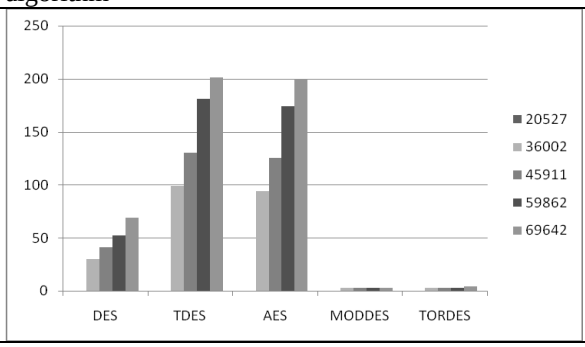
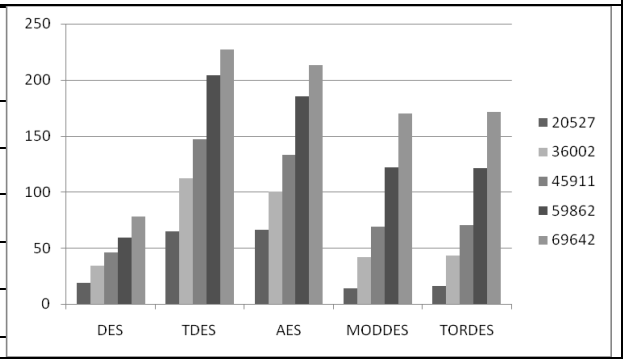


Table4:Encryption+Decryption Time with various algorithm

Input (Bytes)	DES	TDES	AES	MODDES	TORDES
20527	19	65	66	14	16
36002	34	112	100	42	43
45911	46	147	133	69	70
59862	59	204	185	122	121
69642	78	227	213	170	171

Graph 4: Showing Encryption Plus Decryption Time



IV CONCLUSION

Based on the text files used and the experimental result it was concluded that DES algorithm consumes least encryption time and TORDES algorithm has least memory usage while encryption time difference is very minor in case of AES algorithm and DES algorithm. AES consume longest decryption time. Total encryption plus decryption time of TORDES is least than DES, TDES, AES, and MODDES for various size of text files. It has been proved that performance provided by TORDES algorithm is better than other secret key algorithms considered in present study.

References

[1] Bhushan, A., 2012. "*TORDES: A New Approach To Symmetric Key Encryption*", Lambert Academic Publishing, 2012, ISBN 3659218413, 9783659218415
[2] Bhshan, A. 2012. "Transform Operator Random Generator Based Encryption Standard". M.Tech. Dissertation. Mahamaya Technical University, Noida (U.P.)-India.

[3] Bhushan, A., 2012. "*Transform Operator Random Generator Delimiter based Encryption Standard (TORDES)*". CCIT2012, Iraq, , College of Computer, University of Anbar, Ramadi, Iraq 27th-28th March 2012.
[4] Dulari, P., Bhushan, A., 2012. "*Crypto Analysis with A Symmetric Key Algorithm TORDES*", select in NCMIRA 2012, SMVD University Katra, 21 Dec-22 Dec 2012
[5] Bhushan, A., Dulari, "*TORDES-THE NEW SYMMETRIC KEY ALGORITHM*", Journal of University of Anbar for Pure Science, Vol.6:NO.2: 2012. ISSN: 1991-8941. www.iasj.net
[6] Bhushan, A., Dulari, P., 2012 "*Component of Symmetric key Algorithm TORDES with its Functionality*", published in International Journal of Computational Engineering & Management, e-ISSN 2230-7893, Sep 5, 2012
[7] W. Stallings "*Cryptography and network security principles and practice*," Fourth edition, Prentice hall, 2007
[8] Gope, P., Ghosh, D., Chelluri, A.R.K. and Chattopadhyay, P., 2009. *" Multi Operator Delimiter based Data Encryption Standard (MODDES)*" ICCNT. Chennai,India, June 27 – 29. 2009.
[9]Gope, P., Kaushik, A., Arora, K. and Kumar, N., 2010. " *XMODDES (Extended Multi Operator Delimiter Based Data Encryption Standard*", 2nd International Conference on future Networks (ICFN) 2010, China, March, 2010, pp 399-403.
[10] NIST, "Advanced Encryption Standard Call", NIST, 1997.

Ajay Bhushan is pursuing his Master's Degree in Information Technology from Galgotias College of Engineering and Technology, Greater Noida (Uttar Pradesh) – INDIA. He is working as budding researcher in the field of data security. He has more than four year experience of software industry. He is author of one book and six international papers. He has also reviewed more than 30 papers in national and international journals and conference proceedings.

Ajeet K. Bhartee is currently working as Assistant Professor, Department of Computer Science in Galgotias College of Engineering & Technology, Greater Noida (U.P.). He is having about ten year teaching experience. He completed his B.E. from Madan Mohan Malvia Engineering College, Gorakhpur (U. P.) and Masters from CDAC Noida (U.P).

Dr. Pawitar Dulari completed her M.Sc., M.Phil. and Ph.D. from Department of Physics, Institute of Basic Sciences, Dr. B. R. Ambedkar University, Khandari Campus, Agra. She has her M.Sc. specialization in Electronics and Computational Physics. Her research areas include Solid State Physics and Data security. She is currently serving as Assistant Professor, Department of Physics, Govt. P.G. College, Indora (Himachal Pradesh)-INDIA. She has published papers both in National and International journals and conferences.

