AN ORDEAL RANDOMIZED SECURE DATA ENCRYPTION SCHEME (ORSDES)

Ramveer Singh[1]
Head & Associate Professor
Deptt. of I. T.,
R. K. G. I. T., Ghaziabad, U.P., 201003, INDIA
Deo Brat Ojha[2]
Professor, Deptt. of Mathematics,
Mewar University, Rajasthan, INDIA

Abstract :
In this paper, we present a new Data encryption scheme named as Ordeal Randomized Secure Data Encryption Scheme (ORSDES). The theoretical security measures are also discussed and ORSDES advocates its competency. Through this paper, we encourage and motivate the user to use DES as ORSDES with more efficiency and security. Mainly we emphasis on secrecy of key because all knows, In cryptography key always have important role. Using a variable pseudo random number and operational function, the new generated key for each block of message make ORSDES more attractive and usable. ORSDES motivates itself the user to use it with new destiny of confidence, integrity and authentication.

Keywords: Cryptography, Data Encryption Standard, Pseudo Random Number Generator, Secret key.

Introduction

Data Encryption Standard (DES) symmetric key cryptosystem, which was the natural choice, given that this cryptosystem had been around since 1976 and adopted by the US government in 1977, is the US government's secret-key data encryption standard and is widely used around the world in a variety applications

The input message is also known as "plaintext" and the resulting output message as "ciphertext". The idea is that only recipients who know the secret key can decrypt the ciphertext to obtain the original message. DES uses a 56-bit key, so there are 256 possible keys [1].

Due to its importance, DES has received a great deal of cryptanalytic attention. However, besides using the complementation property, there were no short-cut attacks against the cipher until differential cryptanalysis was applied to the full DES in 1991 [2–4].

In [5], Chaum and Evertse presented several meet-in-the-middle attacks on reduced variants of DES.

In 1987 Davies described a known plaintext attack on DES [6]. In [7] these results were slightly improved but still could not attack the full DES faster than exhaustive key search.

In 1994 Biham and Biryukov [8] improved the attack to be applicable to the full DES. A chosen ciphertext variant of the attack is presented in [9]; it has a data complexity of 245 chosen plaintexts. The first attack on DES that is faster than exhaustive key search was presented in [10]. In [11] another attack on DES is presented, linear cryptanalysis. This attack was later improved in [12] by exploiting nonlinear relations as well. The improved attack has a data complexity of 242.6 known plaintexts. Using chosen plaintexts, Knudsen and Mathiassen reduced the data complexity in by a factor of 2.

Even after DES was theoretically broken, RSA published a plaintext and its ciphertext encrypted using DES under some unknown key, and offered a prize of several thousand US dollars for whoever finds the secret key [13]. The first exhaustive key search took about 75 days and the key was found using 14,000–80,000 computers over the Internet [14]. In 1997 the Electronic Frontier Foundation (EFF) built a special purpose machine that costs 250,000 US dollars which retrieved the key in 56 hours by means of exhaustive key search [15]. The approach of treating reduced-round DES as an algebraic equation was also suggested in [16].

Motivation behind our work: Despite the weaknesses of DES, the cipher is still widely deployed and used. In addition, DES-like ciphers are being suggested as a solution for encryption in RFID systems [17]. The results of this paper shed more light on the security of DES, leading to a better understanding on the way DES can be used [18]. The entire discussed articles describe the attack on DES, because DES has a single key to encrypt and decrypt to all blocks of message. After, study the entire journey of data encryption standard, now we are going to improve the vigour of key.

Contribution of this paper: This current article represent a new dimension in the field of cryptography with amplify the security of data encryption standard. In practical, data encryption standard is almost impossible to break. Due to the lack of security level possibility to break it, becomes easier. So, we went in search of suitable procedure to reduce the possibility of breaking security level, which certainly provides improvedness in DES. In our Ordeal Randomized Secure Data Encryption Scheme (ORSDES) suggestion, we keen about the security of symmetric key which can provide us to further secure data.

1. Preliminaries

The three most important objectives of cryptography with respect to the information security include-

1. Confidentiality.
2. Data integrity.
3. Authentication [19].

Confidentiality refers to the protection of information from unauthorized access.
Data integrity ensures that information has not been manipulated in an unauthorized way.
Authentication methods are studied in two groups: Entity authentication and message authentication.
Modern cryptographic techniques provide solutions for these three objectives.
In general, there are two types of cryptosystems:
1. Symmetric (private) key cryptosystems.
2. Asymmetric (public) key cryptosystems [21].

1.1. Symmetric key cryptosystems

All classical cryptosystems (that is cryptosystems that were developed before 1970s) are examples of symmetric key cryptosystems. In addition, most modern cryptosystems are symmetric as well. Some of the most popular examples of modern symmetric key cryptosystems include AES (Advanced Encryption Standard), DES (Data Encryption Standard), IDEA, FEAL, RC5, and many others. All symmetric key cryptosystems have a common property: they rely on a shared secret between communicating parties. This secret is used both as an encryption key and as a decryption key (thus the keyword "symmetric" in the name). This type of cryptography ensures only confidentiality and fails to provide the other objectives of cryptography. On the other hand, an advantage over public key cryptosystems is that symmetric cryptosystems require much smaller key sizes for the same level of security. Hence, the computations are much faster and the memory requirements are smaller [20].

1.2. Data Encryption Standard (DES)

DES relies upon the encryption techniques of confusion and diffusion. Confusion is accomplished through substitution. Specially chosen sections of data are substituted for corresponding sections from the original data. Diffusion is accomplished through permutation. The data is permuted by rearranging the order of the various sections. These permutations, like the substitutions, are based upon the key and the original plaintext.
There are initial and final permutations which occur before and after the sixteen rounds. These initial and final permutations exist for historical reasons dealing with implementation on hardware and do not improve the security of the algorithm. For this reason they are sometimes left out of implementations of DES. They are, however, included in this analysis as they are part of the technical definition of DES [22].

1.3. Cryptanalysis

Cryptanalysis is an art of deciphering an encrypted message in whole or in part, when the decryption key is not known. Depending on the amount of known information and the amount of control over the system by the adversary (cryptanalyst), there are several basic types of cryptanalytic attacks [23, 24, 25, 26, 27, 28, 29]. There are several known attacks on DES like:
Ciphertext-only attack, Brute force attack, Known-plaintext attack, Chosen-plaintext attack and Chosen-ciphertext attack.

1.4. Pseudo Random Number Generator

The two major requirements for a PRNG are
efficiency, as one may wish to produce a large bulk of numbers in a small amount of time, and security. Particularly, one will wish to provide a particular level of cryptographic security, where information leakage is minimized. Such a PRNG will leak a mere one bit of information after producing 2n/2 blocks of output, where n is the block size of the cipher in bits (and assuming the key length is at least as long as the block length).
One should prefer block ciphers in CTR mode to generators based on a dedicated stream cipher. CTR mode requires only that the block cipher be a pseudorandom permutation, which is widely believed to be a reasonable assumption. Dedicated stream ciphers, on the other hand, need to be strong ciphers and also need to resist related key attacks. For example, due to a related key attack, the naïve use of RC4 as a PRNG is fundamentally flawed, even disregarding biases in the cipher. Cryptographic hash functions can also be a good foundation for a PRNG. Many constructs have used MD5 or SHA1 in this capacity, but the constructions are often ad hoc. When using a hash function, we would recommend HMAC in CTR mode (i.e., one MACs counters for each successive output block). Ultimately, we prefer the use of block ciphers, as they are generally better-studied constructs. Depending on the threat model, one may wish to consider protected memory, which is difficult to ensure [30, 31, 32, 33].

2. Our Approach

In this approach, we encourage the user to use data encryption standard with more efficiency and security. Mainly we emphasis on secrecy of key because all knows key always play vital role in cryptography.

In Traditional DES:
The key block size is 56 bit.
Data block size 64 bit,
DES follows the block cipher mode encryption and decryption.
Message (M) = {m_1, m_2, m_3,....., m_n}
Key (K) = {K}
For Encryption/Decryption:
$C_i = E_K\{m_i\}$
Cipher Text C = {C_1, C_2, ..., Cn}
And
$m_i = D_K\{C_i\}$
Plain Text M = {m_1, m_2, ..., m_n}
As per traditional DES, Encryption process follows the feistel structure (16- round) and make a new key for each round by the permutation on bits of key k. Same key k applies on each block of message m using shifting property for encryption and decryption through DES.

Orsdes Approach:

We also using the same feistel structure and same process for encryption and decryption but we add a new process for key generation. In this process, key itself generate n different keys using a function and random number generated by Pseudo Random Number Generator (PRNG) then new generated key block applies on the each block of message for all round of DES. For each block of message, the process generates a separate key. This new generated key used in encryption phase as well as the decryption phase.
The key block size is 56 bit.
Data block size 64 bit,
DES follows the block cipher mode encryption and decryption.
Message (M) = {m_1, m_2, m_3,....., m_n}
Key (K) = {$K_{new\ i}$}

Key Generation
$F\{K \text{ and } R_j\}=K_{new\ i}$
$R_j = \{K_j \| Id_{encryptor}\}$
Where K_j generated by Pseudo Random Number Generator (PRNG)
and $[1 \le K_j \le (256 = 2{,}057{,}594{,}037{,}927{,}936)]$

Function F
Step 1:
Input the bit value of initial key K (56-bit).
Step 2:
Input generated R_j with random number K_j and Id of encryptor, generated by PRNG*.
(*PRNG Property- 256 no., random number generator)
Step 3:
Convert K_j into 56- bit binary number.
Step 4:
Now, we have
Key K = {KB1, KB2, KB3, ..., KB56}
And K_j ={Rb1, Rb2, Rb3, ..., Rb56}
Where KBr is the bit of Key and Rbr is the bit of Random no. Here r =1, 2, 3.....56.
Step 5:
Apply condition on K and Rj.
IF Rbr = 1 then, Complement (convert 1 to 1 or 0 to 0) of corresponding KBr.
ANDIF Rbr = 0 then, Retain the same (1 to 1 or 0 to 0) of corresponding KBr.
Step 6:
$K_{new\ i}$ = Result of step 5.
Using this function F every time we get the result $K_{new\ i}$ for each block of message. For each block of M we generate a new no. R_j and implement function F. Finally get a new key for each block of message.

For Encryption/Decryption

In encryption phase, ORSDES take a message block m_n and a new generated key $K_{new\ i}$ implement encryption process as per traditional DES. One special thing make aur process is different- PLAUSIBLE KEYING.

PLAUSIBLE KEYING have the property to make various key for various block of message.

Now, we have a new key for every block of message. This new key $K_{new\ i}$ is apply on each block of message M.

In this process, New key is also make 16 different key for every round of DES using shifting property as per traditional DES. For every block of message M, new $K_{new\ i}$ makes a new key block for every round of DES to implement in the encryption process.

Decryption Process is the inverse step of encryption process. In decryption, we also use the same key which is used in encryption.

$C_i = E_{Knew\ i}\ \{mi\}$

and

$m_i = D_{Knew\ i}\ \{Ci\}$,

where $1 \leq i \leq n$.

Cipher Text $C = \{C_1, C_2, ..., C_n\}$ and

Plain Text $M = \{m_1, m_2, ..., m_n\}$.

3. Security Analysis of ORDES

In modern cryptography, Encryption / Decryption process based on the key. The strength of key shows that the strongness of scheme. In various cryptosystem (symmetric & asymmetric) have tedious way to find out the key.

The basic concern related to security of key are:

1. No sharing.
2. No transfer.

Benjamin Franklin says “Three people can keep a secret if two of them are dead”. This statement describes itself the major security of key is no sharing.

Cryptographic scheme strength is often described by the bit length of encryption key. The more bits in the key, the harder it is to decrypt data simply by all possible key. DES uses 56 bit, Cracking 56- bit algorithm with a single key search might take around a week on a very powerful computer.

Now,

At time t, the generated key is $K_{new\ x}$,

At time t + 1, the generated key is $K_{new\ y}$

And At time t + n, the generated key is $K_{new\ z}$

Here,

$K_{new\ x} \neq K_{new\ y} \neq K_{new\ z}$

It might be possible that, $K_{new\ x} \neq K_{new\ y} \neq K_{new\ z}$, are equal if and only if the generated no. R_j at time t, t + 1, t + n are same, But the probability of generate the same no. at different time is very much low because of we are using PRNG.

Now, Compare with traditional DES, we simply analyze the security level of ORSDES. If message have n blocks then the security of ORSDES is increase n times more than traditional DES. We also accept that the process might be face one problem – Increase the overhead. But due to aspect of security level, we can compromise with that problem.

Meet-in-the-Middle Attacks:

Meet-in-the-middle can minimise the number of brute force permutations required to decrypt message that has been encrypted by more than one key. This attack targets block cipher cryptographic schemes. The attackers apply brute force technique to both the plaintext and ciphertext of a block cipher. He then attempts to encrypt the plaintext according to various possible combinations of keys to achieve an intermediate ciphertext (a text that has only been encrypted by one key). Simultaneously, he attempts to decrypt the ciphertext according to various possible combinations of keys, seeking a block of intermediate ciphertext that is the same as the one achieved by encrypting the plaintext. If there is a match of intermediate ciphertext, it is highly probable that the key used to encrypt the plaintext and the key used to decrypt the ciphertext are the two encryption keys used for the block cipher.

Remark: ORSDES successfully meet with this requirement.

Linear Factors:

DES consist a plaintext, a key size, a ciphertext and a family of invertible maps indexed by the key space. We say that cryptosystem X is a factor of cryptosystem Y if there are maps (called factor maps) between the plaintext, key and ciphertext such that the enciphering and deciphering action of cryptosystem A can be recovered from those of cryptosystem B using factor of B. If the key space of A is smaller than that of B one can profitably break B by first breaking A.

Remark: ORSDES successfully meet with this requirement.

Weak Keys:
A weak key 'K' is a key for which encryption is the same function as decryption. A pair of semi-weak keys, K and K, are keys for which encryption with K is the same as decryption with K' and vice versa. DES has weak keys, if the number of weak keys is relatively small, they may not compromise the cipher when used to assure confidentiality.

Remark: ORSDES successfully meet with this requirement.

Detectable Key Classes:
One way to reduce the effective keyspace is to divide the keyspace into classes, and then find an attack that reveals to which class the key belongs. In some instances, the workload of identifying a key with a specific class is very small; these too are sometimes referred to as weak keys. IDEA has several classes of keys detectable with just two chosen-plaintext encryptions. The key schedule allows two different keys to have several round keys in common; this reduces the affective keyspace by almost a factor of four using 233chosen plaintexts. Due to the weak mixing in its key schedule, RC4 has a class of detectable keys. One out of 256 keys is detectable, and a detectable key has about a 13.8% chance of revealing 16 bits of the key in the first output byte.

Remark: ORSDES successfully meet with this requirement.

Simple Relations and Equivalent Keys:
A simple relation occurs between two different keys, manifesting itself as a relationship between the resulting plaintexts and ciphertexts. This also allows the keyspace to be reduced in a search. DES has a simple relation known as the complementation property: if K encrypts P to C, then the bitwise complement of K encrypts the bitwise complement of P to the bitwise complement of C. This reduces the effective keyspace by one bit. DES has pairs of keys for which a simple relation exists, for at least a fraction of all plaintexts.
Two keys are equivalent if they transform all plaintexts identically. This can be considered a special kind of simple relation.

Remark: ORSDES successfully meet with this requirement.

Attacks on One-Wayness:
A key schedule is one-way if, given several round subkeys, it is infeasible for an attacker to gain any new information about the master key or about other unknown round subkeys. For instance, recovering a few round subkeys allows one to recover most of the master key in the DES key schedule. Furthermore, it may be easier to find weak keys and related keys for key schedules which are not one-way.

Remark: ORDES successfully meet with this requirement.

Related Key Attack:
A related-key attack is one where the attacker learns the encryption of certain plaintext not only under the original (unknown) key K, but also under some derived keys K'= f (K). In a chosen-related-key attack, the attacker specifies how the key is to be changed; known-related-key attacks are those where the key difference is known but cannot be chosen by the attacker. We emphasize that the attacker knows or chooses the relationship between keys, but not the actual key values.
Three-key triple-DES is a well-known method for strengthening DES with a 168-bit key; it is also vulnerable to related-key attacks. This mode can be considered a 3-round cipher with independent 56-bit round subkeys, realizing that each round is very strong. Quixotically, one might use rotational related-key cryptanalysis; however, such an approach would require many known plaintexts.

Remark: ORSDES successfully meet with this requirement.

4. Conclusion:
Our section 4 completely advocates the plausibility of ORSDES. This plausibility of key gives an authenticated encryption due to concatenation of identity of encryptor. It is successful and needful for current communication scenario.

Case I
If we take one and only one Key on the place of n keys then $K_{new\ 1}$ is K, at this condition our approach works like DES.

Case II
If $K_{new\ 1} = K_{new\ 1} = K_{new\ 1}$, than our approach also works like DES.

References

[1] Data Encryption Standard, Federal Information Processing Standards Publication (FIPS PUB) 46, National Bureau of Standards, Washington, DC (1977).

[2] Eli Biham, Adi Shamir, Differential Cryptanalysis of DES-like Cryptosystems, Journal of Cryptology, Vol. 4 No. 1, Springer, pp. 3–72, 1991.

[3] Eli Biham, Adi Shamir, Differential Cryptanalysis of the Full 16-Round DES, Advances in Cryptology, proceedings of CRYPTO '92, Lecture Notes in Computer Science 740, Springer, 1993.

[4] Eli Biham, Adi Shamir, Differential Cryptanalysis of the Data Encryption Standard, Springer, 1993.

[5] David Chaum, Jan-Hendrik Evertse, Cryptanalysis of DES with a Reduced Number of Rounds: Sequences of Linear Factors in Block Ciphers, Advances in Cryptology, proceedings of CRYPTO '85, Lecture Notes in Computer Science 218, pp. 192–211, Springer, 1986.

[6] Donald W. Davies, Investigation of a Potential Weakness in the DES Algorithm, private communications, 1987.

[7] Donald W. Davies, Sean Murphy, Pairs and Triplets of DES S-Boxes, Journal of Cryptology, Vol. 8, No. 1, pp. 1–25, Springer, 1995.

[8] Eli Biham, Alex Biryukov, An Improvement of Davies' Attack on DES, Journal of Cryptology, Vol. 10, No. 3, pp. 195–206, Springer, 1997.

[9] Sebastien Kunz-Jacques, Frederic Muller, New Improvements of Davies-Murphy Cryptanalysis, Advances in Cryptology, proceedings of ASIACRYPT 2005, Lecture Notes in Computer Science 3788, pp. 425–442, Springer, 2005.

[10] Kunz-Jacques, S., Muller, F.: New Improvements of Davies-Murphy Cryptanalysis.In: Roy, B. (ed.) ASIACRYPT 2005. LNCS, vol. 3788, pp. 425–442. Springer, Heidelberg (2005)

[11] Mitsuru Matsui, Linear Cryptanalysis Method for DES Cipher, Advances in Cryptology, proceedings of EUROCRYPT '93, Lecture Notes in Computer Science 765, pp. 386–397, Springer, 1994.

[12] Takeshi Shimoyama, Toshinobu Kaneko, Quadratic Relation of S-box and Its Application to the Linear Attack of Full Round DES, Advances in Cryptology, proceedings of CRYPTO '98, Lecture Notes in Computer Science 1462, pp. 200–211, Springer, 1998.

[13] CNET News.com, Users take crack at 56-bit crypto. Available on-line at http://news.com.com/2100-1023-278658.html?legacy=cnet, 1997.

[14] RSA Data Security, Team of Universities, Companies and Individual Computer Users Linked over the Internet Crack RSA's 56-Bit DES Challenge. Available on-line at: http://www.rsasecurity.com/news/pr/970619-1.html, 1997.

[15] Electronic Frontier Foundation, Cracking DES, Secrets of Encryption Research, Wiretap Politics & Chip Design, O'reilly, 1998.

[16] Nicolas T. Courtois, Gregory V. Bard, Algebraic Cryptanalysis of the Data Encryption Standard. Available on-line at: http://eprint.iacr.org/2006/402.pdf, 2006.

[17] Axel Poschmann, Gregor Leander, Kai Schramm, Christof Paar, New Light-Weight DES Variants Suited for RFID Applications, proceedings of Fast Software Encryption 14, Lecture Notes in Computer Science, Springer (to appear), 2007.

[18] Orr Dunkelman, Gautham Sekar, and Bart Preneel: "Improved Meet-in-the-Middle Attacks on Reduced-Round DES", to appear in Indocrypt 2007.

[19] A.M. Eskicioglu, "Protecting Intellectual Property in Digital Multimedia Networks," IEEE Computer, July 2003, pp. 39-45.

[20] Whitfield Diffie and Martin E. Hellman, "New Directions in Cryptography" IEEE transactions on Information Theory, 22, 644-654

[21] P.C. van Oorschot, A.J. Menezes, and S.A. Vanstone, "Handbook of Applied Cryptography," CRC Press, Inc., 1997.

[22] D.B. Ojha, Ramveer Singh, Ajay Sharma, Awakash Mishra and Swati garg "An Innovative Approach to Enhance the Security of Data Encryption Scheme" International Journal of Computer Theory and Engineering, Vol. 2,No. 3, June, 2010,1793-8201

[23] M.Matsui: "The First Experimental Cryptanalysis of the Data Encryption Standard", Crypto'94, LNCS 839, Springer, pp. 1-11, 1994.

[24] Eli Biham and Adi Shamir: "Differential Cryptanalysis of DES-like Cryptosystems". Journal of Cryptology, vol. 4, pp. 3-72, IACR, 1991.

[25] M. Matsui: "Linear Cryptanalysis Method for DES Cipher",Eurocrypt'93", LNCS 765,Springer, pp. 386-397, 1993.

[26] Orr Dunkelman, Gautham Sekar, and Bart Preneel: "Improved Meet-in-the- Middle Attacks on Reduced-Round DES", To appear in Indocrypt 2007.

[27] Alejandro Hevia, Marcos Kiwi, "Strength of two data encryption standard implementation under timing attacks", ACM Transactions on Information and System Security (TISSEC), Volume 2, Issue 4 (November 1999) Pages: 416 – 437.

[28] M.Matsui: "The First Experimental Cryptanalysis of the Data Encryption Standard", Crypto'94, LNCS 839, Springer, pp. 1-11, 1994.

[29] Lars R. Knudsen, John E. Mathiassen, A Chosen-Plaintext Linear Attack on DES, proceedings of Fast Software Encryption 7, Lecture Notes in Computer Science 1978, pp. 262–272, Springer, 2001.

[30] M. Bellare, A. Desai, E. Jokipii and P. Rogaway, "A concrete security treatment of symmetric encryption: analysis of the DES modes of operation", In Proc. 38th Annual Symposium on Foundations of Computer Science, 1997.

[31] S. Fluhrer, I. Mantin and A. Shamir, "Weaknesses in the key scheduling algorithm of RC4", in Proc. Eighth Annual Workshop on Selected Areas in Cryptography, Aug. 2001.

[32] S. Fluhrer and D. McGrew, "Statistical analysis of the alleged RC4 stream cipher", in Proc. Fast Software Encryption Seventh International Workshop, Springer-Verlag, Mar. 2000.

[33] P. Gutmann, "The design and verification of a cryptographic security architecture," Ph.D. dissertation, Dept. Comp. Sci., Univ. of Auckland, Aug. 2000.