تحسين المفتاح لنظام DES

بحث في علوم الحاسبات
مجال التشفير

الباحث
أحمد محمد مال الله الصفار

نيسان – 2005
DES - Key Enhancement

A RESEARCH
IN
COMPUTER SCIENCE
Cryptology

BY

AHMED MHMED MAL-ALLAH AL-SAFAR

APRIL - 2005
الخلاصة

هو نظام كي و يُستخدم مفتاح (64 بت) يعمل على كتلة نص (56 بت). ونظام DES
والتحويلات ال الروسيات كتلة مصممين خصيصاً لإنتاج تطبيقات الأجهزة السريعة وتطبيقات البرامج البطيئة (8)

ما زال شيء في الكشفية لإيقاف أكثر لصوص الكمبيوتر والأفراد العشوائيين خارج امكانيه العمل، لكنه قابل للكسر بسهولة بالأجهزة

الخاصة لدى بعض المنظمات الحكومية ومجموعات الجريمة.

إذاً الضرر الواضح لهذا النظام هو؛ حيث ان مقبول على صعيد عموم البلد (أو على الصعيد العالمي)، لذا فإن احتمال وجود شخص

من يمكنه إيجاد الحل الكبير جداً. إضافة إلى أن كتلة استعماله يولد القناعة لدى المحترف بانه يمكنه، سيمكن من الدخول إلى العديد من

رسائل المستعملي. ولهذا، فيما وجد طريقة للهجوم، ستكون ذات قيمة عالية عند التطبيق مهما كانت الكلفة. بذلك، يصبح

جيداً، ولا يجب أن يستعمل في التطبيقات الجديدة.

هذا البحث يركز على مجال خوارزمية المفتاح لنظام DES فقط كما هو في الاستعمال الحالي وليس على عموم الخوارزمية. إن الغرض

الرئيسي للبحث أن ينتج تحسين على خوارزمية DES من خلال إجراء التعقيد على المفتاح المستخدم حالياً لكى يزيد التعقيد الحسابي

العام وبناءً نموذج مفتاح جديد لا يكون قابل للكسر بسهولة كما هو منشور بالنسبة إلى DES، بمنظمات الجريمة والمنتهليين ولزود

الامكانية والقابلية على وجود نموذج حقيقي جديد للتنشيط يمكن استعماله اليوم.

بالتأكيد من المستحيل الحزم بضمان أن هذه الحالة ليست قابلة للحل في الوقت المتعدد الحدود. ولذلك، من المعقول القول بان هناك دائماً

خطر حتى وإن كان صغيراً جداً. يوجد شخص ما قد يتمكن من ايجاد طريقة لكسر النظام الحالي.

Abstract
DES is a block-cipher employing a 56-bit key that operates on 64-bit blocks. DES has a complex set of rules and transformations that were designed specifically to yield fast hardware implementations and slow software implementations.(4)

DES is still strong enough to keep most random hackers and individuals out, but it is easily breakable with special hardware by government and some criminal organizations. The obvious disadvantage of standard is that; since it is accepted on nationwide (or even worldwide) basis, therefore the probability of being someone finds a solution is exceedingly large. In addition to its widely used then the cryptanalyst knows that, by breaking it, he will gain access to many users' messages. Thus, if he finds a method of attack, it will be worth his while to implement it no matter what the cost. Thereby, DES is getting too weak, and should not be used in new applications.

This research is focused on DES Key scheme as it is in current use and is not a treatise of the whole field. The major purpose of the research is to fulfill enhancement to DES algorithm by sophisticating its key procedure in order to increase overall computational complicity and to create new individual not to be breakable as easy as with published standard DES, by myriad criminal organizations and any other malicious and to provide the capability of the real example of cryptography to be use today. Absolutely it is impossible to guarantee that this particular instance is not solvable in polynomial time. Consequently, it is reasonable there is always a danger, hopefully very small, that someone may be finding a way to break the system.

1- Introduction

Within the context of any application-to-application communication, there are some specific security requirements, including: (12)

**Authentication**: The process of proving one's identity. The primary forms of host-to-host authentication on the Internet today are name-based or address-based, both of which are notoriously weak.

**Privacy/confidentiality**: Ensuring that no one can read the message except the intended receiver.

**Integrity**: Assuring the receiver that the received message has not been altered in any way from the original.

**Non-repudiation**: A mechanism to prove that the sender really sent this message.

The use of cryptography is no longer a privilege reserved for governments and highly skilled specialists, but is becoming available for everyone. The Internet provides essential
communication between tens of millions of people and is being increasingly used as a tool for commerce; security becomes a tremendously important issue to deal with. By encryption, we mean a process of converting information to a disguised form in order to send it across a potentially unsafe channel. The reverse process is called decryption. Using strong encryption techniques, sensitive, valuable information can be protected against organized criminals, malicious hackers, or spies from a foreign military power, for example. Indeed, cryptography used to be almost exclusively a tool for the military.

Secret key cryptography schemes are generally categorized as being either stream ciphers or block ciphers. A block cipher is so-called because the scheme encrypts one block of data at a time using the same key on each block. In general, the same plaintext block will always encrypt to the same ciphertext when using the same key in a block cipher whereas the same plaintext will encrypt to different ciphertext in a stream cipher. (3)

Block ciphers divide the plaintext into blocks, usually of fixed size, and operate on each block independently. Block ciphers are therefore transposition and simple substitution ciphers with large alphabets to prevent cryptanalysis by exhaustive search. (5)

DES is a block cipher with a 64-bit block size. It uses 56-bit keys. This makes it susceptible to exhaustive key search with modern computers and special-purpose hardware. There is another obvious disadvantage to a standard that is if any one finds a method of attack which works, it will be worth his while to implement it no matter what the cost. If, on the other hand, he were trying to break a cipher system adopted by a single user he might consider certain implementations too expensive. The adoption of a standard also focuses the attention of all cryptanalysts on the same system and must therefore increase the chances of it being broken. (3)

2- Literature survey

As it is mentioned above the Data Encryption Standard (DES) is an algorithm developed in the mid-1970s. Once DES was "officially" broken, several variants appeared. But none of them came overnight; work at hardening DES had already been underway.

In 1999, a brute-force search using a specially designed supercomputer and a worldwide network of nearly 100,000 PCs on the Internet, found a DES key in 22 hours and 15 minutes. (13)

In the early 1990s, there was a proposal to increase the security of DES by effectively increasing the key length using multiple keys with multiple passes. But for this scheme to work, it had to first be shown that the DES function is not a group, as defined in mathematics. If DES was a group, then we could show that for two DES keys, X1 and X2, applied to some plaintext (P), we can find a single equivalent key, X3, that would provide the same result; i.e.,:

$$E_{X_2}(E_{X_1}(P)) = E_{X_3}(P)$$

The specification defines use of two keys provides an effective key length of 112-bit. But the use of Triple-DES a variant of DES that employs up to three independent 56-bit keys and makes three encryption/decryption passes over the block and provides an effective key length of 168 bits; 3DES is also recommended replacement to DES. (9)
Blowfish was designed by Bruce Schneier. It is a block cipher with a 64-bit block size and variable length keys (up to 448 bits). It has gained a fair amount of acceptance in a number of applications.

Another variant of DES, called DESX, is due to Ron Rivest. Developed in 1996, DESX is a very simple algorithm that greatly increases DES's resistance to brute-force attacks without increasing its computational complexity. In DESX, the plaintext input is XORed with 64 additional key bits prior to encryption and the output is likewise XORed with the 64 key bits. (10)

IDEA (International Data Encryption Algorithm) is an algorithm developed Zurich in Switzerland by Xuejia Lai. It uses a 128-bit key, and it is generally considered to be very secure. (10)

3- DES Background

The development leading to DES started in 1973 with the initiation of a common method of encryption which could be economically employed in a variety of computer security application. (12)

The obvious advantage of using the standard is the compatibility which can be attained between various systems. Although, what have said before, about the risk that if one is broken then they all are. So one of the main advantages associated with the existence of a standard relates to the cost for the user.

DES then acts on 64-bit blocks of the plaintext, invoking 16 rounds of permutations, swaps, and substitutes, as shown in FIG (1). The standard includes tables describing all of the selection, permutation, and expansion operations these aspects of the algorithm are not secrets.

If a chip, or set of chips, is designed to implement the standard, then it can be produced in sufficient quantity that the cost to the user be low.

Block ciphers are therefore transposition and simple substitution ciphers with large alphabets to prevent cryptanalysis by exhaustive search. (11) Diffusion dissipates the redundancy of the plaintext by spreading it out over the ciphertext. A cryptanalyst looking for those redundancies will have a harder time finding them. The simplest way to cause diffusion is through transposition (also called permutation). A simple transposition cipher, like columnar transposition, simply rearranges the letters of the plaintext. Modern ciphers do this type of permutation, but they also employ other forms of diffusion that can diffuse parts of the message throughout the entire message. (8)
At simplest level of DES, the algorithm is nothing more than a combination of the two basic techniques of encryption: confusion and diffusion.

4- The Proposal

4-1 Introduction

Descriptions of many good cryptographic algorithms are widely and publicly available from any major bookstore, scientific library, patent office, or on the Internet. DES is one of them; as a block cipher with a 64-bit block size. This makes it susceptible to exhaustive key search with modern computers and special-purpose hardware. Even today, when DES is no longer considered a practical solution, it is often used to describe new crypt analytical techniques. It is remarkable that even today, there are no crypt analytical techniques that would completely break. (3,12)

Unpublished or secret algorithms should generally be regarded with suspicion. Quite often the designer is not sure of the security of the algorithm, or its security depends on the secrecy of the algorithm. Generally, no algorithm that depends on the secrecy of the algorithm is secure. So, all security must be reside with the key.

There is no excuse for a system designer to leave the system breakable. It is possible to build a system that has more complex key than DES to be broken in polynomial
computational time. This does not significantly increase system implementation effort in hardware and software; however, some care and expertise is required.

I would like to say at the outset that this research is very focused on DES as a scheme in current use and is not a treatise of the whole field. No mention is made here about pre-computerized crypto schemes, but the core of this research is how to complicate the Key algorithm of DES Scheme.

Because of the way to modify the initial key to get a subkey for each round of the algorithm certain initial keys are weak keys. (10)

As a consequence, National Institute of Standards and Technology (NIST) proposed in 2004 to withdraw the DES standard. (15)

4-2 The Current Work of DES Key Algorithm

The 64-bit permuted input of data is divided into two 32-bit blocks, called left and right, respectively. The initial values of the left and right blocks are denoted L₀ and R₀ in figure.

At any given step in the process, then, the new L block value is merely taken from the prior R block value. The new R block is calculated by taking the bit-by-bit exclusive-OR (XOR) of the prior L block with the results of applying the DES cipher function, f, to the prior R block and Kₙ. (Kₙ is a 48-bit value derived from the 64-bit DES key. Each round uses a different 48-bit according to the standard's Key Schedule algorithm).

The cipher function, f, combines the 32-bit R block value and the 48-bit subkey in the following way. First, the 32-bit in the R block are expanded to 48-bit by an expansion function (E); the extra 16 bits are found by repeating the bits in 16 predefined positions.

The 48-bit expanded R-block is then XORed with the 48-bit subkey. The result is a 48-bit value that is then divided into eight 6-bit blocks. These are fed as input into 8 selection (S) boxes, denoted S₁,...,S₈. Each 6-bit input yields a 4-bit output using a table lookup based on the 64 possible inputs; this results in a 32-bit output from the S-box. The 32-bit are then rearranged by a permutation function (P), producing the results from the cipher function.

The key length is 56-bit. (The key is usually expressed as a 64-bit number, but every eighth bit is used for parity checking ☻ and is ignored. These parity bits are the least-significant bits of the key bytes.) The key can be any 56-bit number and can be changed at any time. A handful of numbers are considered weak keys, but they can easily be avoided. (3,10,12)
Parity: is a function that is computed to provide a check on a group of binary values by forming the modulo-2 sum of the bits in the group. The generated sum, a redundant value, is called the parity bit. The parity bit is 0 if the number of 1s in the original group was even. The parity bit is 1 if the number of 1s in the original group was odd.
As in FIG (2) each round the key bits are shifted to left, and then a different 48-bit subkey is generated for each of the 16 rounds of DES selecting them from the 56-bit of the key. Because this operation permutes the order of the bits as well as selects a subset of bits, it is called a compression permutation. (10)
Permuted choice-1 and ignoring eighth bit is determined by the following table:

PC-1

<table>
<thead>
<tr>
<th>57</th>
<th>49</th>
<th>41</th>
<th>33</th>
<th>25</th>
<th>17</th>
<th>9</th>
<th>1</th>
<th>58</th>
<th>50</th>
<th>42</th>
<th>34</th>
<th>26</th>
<th>18</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>2</td>
<td>59</td>
<td>51</td>
<td>43</td>
<td>35</td>
<td>27</td>
<td>19</td>
<td>11</td>
<td>3</td>
<td>60</td>
<td>52</td>
<td>44</td>
<td>36</td>
</tr>
<tr>
<td>63</td>
<td>55</td>
<td>47</td>
<td>39</td>
<td>31</td>
<td>23</td>
<td>15</td>
<td>7</td>
<td>62</td>
<td>54</td>
<td>46</td>
<td>38</td>
<td>30</td>
<td>22</td>
</tr>
<tr>
<td>14</td>
<td>6</td>
<td>61</td>
<td>53</td>
<td>45</td>
<td>37</td>
<td>29</td>
<td>21</td>
<td>13</td>
<td>5</td>
<td>28</td>
<td>20</td>
<td>12</td>
<td>4</td>
</tr>
</tbody>
</table>

After dividing the 56-bit key into two 28-bit halves, the two halves are circularly shifted left by either one or two, as shown below, depend on the round.

The Key Bits Shifted as Rotation Schedule per Round is specified as standard:

<table>
<thead>
<tr>
<th>Round</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ks</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

The right half of the data is expanded to 48-bit via an expansion permutation, combined with 48-bit of a shifted and permuted key via an XOR, sent through 8 s-boxes producing 32 new bits, and permuted again. These four operations make up function.

The computational time complexity to the current DES algorithm is: (3)

\[ T = O(2^{56}) \]

4-3 The proposed DES Key Algorithm

The say that all security resides with the key is a fact in cryptography. Absolutely, impossible to guarantee that this particular instance (DES Key) is not solvable in polynomial time. So providing any function to increase security to the current DES Key algorithm will produce another sophisticated instance which is more robust than particular key in DES.

As I said at the outset that this research is much focused on DES as a scheme in current use and merely increasing system implementation hardware and software effort however, by some requirements; care and expertise in cryptography software is needed to enhance key DES Algorithm by increasing the complexity as explanation shown below:
The key can be any 56-bit number and can be changed at any time. After dividing the 56-bit key into two 28-bit halves, the two halves in each round are circularly shifted either to the right or to the left as shown below, depending on the result of Eq (1):

\[
\text{(The generated Sum of the value of the first seventh bits)} \mod 2 \quad \ldots \ldots \text{Eq (1)}
\]

The key is shifted right if the output is 0 and shifted left if the output is 1. That is instead of the fixed state; just shifted to the left as mentioned in current DES. In the other hand the number of bits is shifted depending on two methods. So the shifting is either two or more bits to the same side, determined by Eq (2):

\[
\text{(The generated Sum of the value of the last seventh bits)} \mod 2 \quad \ldots \ldots \text{Eq (2)}
\]

Then the key is shifted by (2 bits) if the output is 0 or shifted by (1 bit) if the output is 1. So it is clear that the computational time complexity of these two functions added (2^2 * 16 rounds):

\[
T = O(2^2 * 2^4) = O(2^6)
\]

but to avoid probably getting the same Key in two neighbor rounds; in case that the Key in the first round shifted by 1 or by 2 to the left and respectively shifted to the right by 1 or by 2 in the second round, the Key Bits also shifted to the same direction as specific Rotation Schedule per Round as specified as standard below:

<table>
<thead>
<tr>
<th>Round 1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ks</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

So hereby, the Key is shifted twice instead of the only fixed state within the table as mentioned in (4-2) as in standard in current DES. Thereby, the algorithm produces an individual Key per round perfectly.
Different 48-bit subkey is generated for each of the 16 rounds of DES selected them from the 56-bit of the key after they are divided in two halves of (28-bit) and shifted. Because this operation permutes the order of the bits as well as selects a subset of bits (48-bit), it is called a compression permutation.

But the difference of compression permutation in DES Key algorithm in this research comparing with compression permutation in current standard DES Key algorithm is that; by this proposal there are \(33,554,432\) selected tables (of the addresses of the combination of 56 position) stored in additional device (EPROM), these number of tables represent just \(2^{25}\) states as selective combination permutation table extracted from the \((56!)\) states.

One of these tables will be selected for each round \((i)\) by its address using Eq (3) below as a compression permutation to get 48-bit Key from 56-bit in stead of depending on the fixed table as in standard DES Key in current use. Consequently, this 48-bit Key will be XORed with \(R_i\) 32-bit after being expanded according to the algorithm to 48-bit of Text:

\[(\text{The value of the first 25 bits from either } L_i \text{ or } R_i \text{ for each Round } i) \ldots \ldots \ldots \ldots \ldots \text{ Eq (3)}\]

As it is shone in Eq (3) in each Round either \(L_i\) or \(R_i\) is determined according to the output of Eq (1); \(R_i\) if the output is 0 and \(L_i\) if the output is 1.

5- Evaluation

1- The occurrence of DES Key enhancement the fulfillment in this research to increase overall computational time complicity to DES algorithm by sophisticating its key procedure and to create new individual not to be breakable as easy as with published standard DES. Really it has awesome value to yourself and to others that the computational time complicity to the new DES algorithm is: \((1, 6, 13)\)
2- In theory, any cryptographic method with a key can be broken by trying all possible keys in sequence. DES system with 56-bit keys requires a substantial effort but using massive distributed systems requires only hours of computing. So depending on variant Key 25-bit from the individual Text in addition to DES Key 56-bit in current use will provide substantial robustness against brut-force attack as trying all possible keys in sequence. Thereby, on the other hand, if any one was trying to break a cipher system adopted by a single user he might consider certain implementations too expensive. (1, 2)

3- Correlation between the secret key and the output of the cryptosystem is the main source of information to the cryptanalyst. In the easiest case, the information about the secret key is directly leaked by the cryptosystem. More complicated cases require studying the correlation (basically, any relation that would not be expected on the basis) between the observed (or measured) information about the cryptosystem and the guessed key information. Nonlinearity, used in this proposed algorithm to select compression permutation Key 48-bit is the biggest resistance to correlation attack. (2, 13)

4- The Algorithm must provide different keys for each round otherwise security may be compromised. So multiple methods to shift the Key left or right assures prevention not to get the same Key in two neighboring rounds.

6- Future Work

- The Significance of Robustness of complexity of Key.
- Recommendation to use multiple enciphering with DES system.
- Extending DES to 128-bit data paths and 112-bit keys.
- Any mechanism that can be used to circumvent security must be made explicit, documented, and brought into the attention of the end users. (2, 4, 14)
1. Douglas Stinson,
   "Cryptography: Theory and Practice",

2. Richard Mansfield,
   "HACKER ATTACK!",
   Associate Publisher: Jordan Gold,
   Manufactured in the United States of America, Copyright © 2000 SYBEX.

3. Eli Biham, Adi Shamir,
"Differential Cryptanalysis of the Data Encryption Standard",
Springer-Verlag, 1993.

4. Xuejia Lai,
"On the Design and Security of Block Ciphers",
ETH Series in Information Processing, vol. 1,
Hartung-Gorre Verlag, Konstanz, Switzerland, 1992.

5. Meyer Carl H. & Maytas Stephen M.,
"Cryptography: A New Dimension in Computer Data Security",

6. Seberry.J. & Pieprzyk,
"Cryptography: An Introduction to Computer Security",

7. L.C. Guillou and j-j. Quisquater eds,
"Advances in Cryptology- EUROCRYPT 95, Lecturers Notes in Computer Science",

8. Manuel Blum, C.H.Lee,
Proceedings of 1999 International Workshop on:
"Cryptographic Techniques & E-Commerce",
City University of Hong Kong, 5-8 July 1999.

9. Walter Fumy (Ed.),
"Advances in Cryptology - EUROCRYPT '97"
International Conference on the Theory and Application of Cryptographic Techniques,

10. Bruce Schneier,
"Applied Cryptography",
A great overall look at all areas of cryptography. Contains basic information on about every cryptographic technique and protocol. There is also some scattered information on zero-knowledge topics. (Highly recommended)

11. Alfred J. Menezes,
"Handbook of Applied Cryptography",
CRC Press, 1996.

12. Beker H. & Piper F.,
"Cipher Systems: The Protection of Communication",

13. Gustafson H., Dawson E. and Nielsen L.,
"A Computer Package for Measuring the Strength of Encryption Algorithms",

14. Tatu Ylnen's,
"Crypto Pages",
http://www.cs.hut.fi/crypto/
This page has a lot of pointers to crypto information. It is also the home for the secure shell, program for secure remote logins etc.

15. A Bibliography of Quantum Cryptography,
http://www.itr.ch/~pheinzma/quantlit.html