How to Strengthen DES Using Existing Hardware

Eli Biham* Alex Biryukov**

Abstract. Differential, linear and improved Davies' attacks are capable of breaking DES faster than exhaustive search, but are usually impractical due to enormous amounts of data required. In [20] Wiener designed a million dollar special purpose computer capable of breaking DES in 3.5 hours in average by exhaustive search. In this paper we describe methods of strengthening DES against exhaustive search, differential attacks, linear attacks and improved Davies' attacks that can be applied on existing DES hardware. We use the fact that there are DES chips in the market that permit replacement of the S-boxes. We introduce the concept of key-dependent invariant S-box transformations. Differential and linear properties of the cipher are invariant under these transformations. We show how to expand the key using such transformations. Possible reorderings of S-boxes are discussed; we present orders of the original DES S-boxes which are slightly stronger than the standard order of S-boxes. Finally we suggest a concrete scheme to strengthen DES which uses the methods described above. This modified DES can be used with existing DES hardware and is much stronger than the standard DES.

1 Introduction

Since the Data Encryption Standard was introduced [17], its 56-bit key size was subject to criticism of the research community [10, 9]. It was considered to be too short to withstand exhaustive search attack on a special purpose computer. Recent results [20] show that with todays technology such computer will cost about a million US$ and will be able to find a key in 3.5 hours in average.

In parallel, many researches invested a great effort to cryptanalyze DES. Their work lead to development of two powerful methods of cryptanalysis of iterative ciphers: differential cryptanalysis [2] and linear cryptanalysis [14]. Recently Davies' attack [8] has been improved to be capable of breaking DES faster than exhaustive search [3]. Those are the only known methods of breaking DES faster than half of exhaustive search; they require huge amounts of 2^{47},2^{43} and 2^{50} plaintexts, respectively.

These attacks are very important for our understanding of the design principles of good cryptosystems. The fact that these attacks on DES are impractical (due to the enormous amounts of data required) is a result of a careful design. Still the real threat to the practical use of DES is its short secret key.

* Computer Science Department, Technion - Israel Institute of Technology, Haifa 32000, Israel.

** Applied Mathematics Department, Technion - Israel Institute of Technology, Haifa 32000, Israel.

Several suggestions were made in the last two decades in order to strengthen DES: increase of the number of rounds from 16 to 32, 64 or even more [10], multiple encryption or larger key size[9], independent subkeys (768 bits) [1], dramatic increase of the key scheduling complexity [13, 19] and others. Not considering security features of these solutions we note that most of them either require the design of new hardware or decrease encryption speed considerably.

The standard of DES[17] requires hardware implementation. During the last two decades many different DES chips were developed. All these chips will become useless if the standard is to be modified or substituted. Some of these chips were designed with this fact in mind, and they allow the user to choose his favorite S-boxes as a replacement to the standard S-boxes (for example see [18]).

In this paper we describe new methods to strengthen DES against the attacks mentioned above. These methods require no hardware replacement, whenever the existing hardware can load alternative S-boxes. Our changes to DES are based on loading new S-boxes. We introduce the concept of key-dependent invariant S-box transformation. These transformations preserve the properties of the S-boxes related to differential and linear attacks. We show how to increase the key size using these transformations, resulting in a dramatical increase of the complexity of exhaustive search. Possible reorderings of the S-boxes are discussed. In [2] it is shown that some modified orders of the S-boxes weaken DES; we present several modified orders of the S-boxes of DES that slightly increase strength against both differential and linear attacks. We give an example of an order for which linear attacks require 2^{53} known plaintexts and differential attacks require 2^{48} chosen plaintexts.

Recently Kim, Park and Lee suggested a new set of DES-like S-boxes (which they call – s^3DES). We believe that this set is more secure than the original set against differential and linear attacks (when the order of $S1$ and $S2$ is reversed). We use this set to strengthen our modified DES. We also show that random key dependent S-boxes (as in Khufu [16]) might weaken DES.

Finally we suggest a concrete scheme, which uses s^3DES S-boxes and has a longer key. This modified DES can be used with existing DES hardware and is claimed to be stronger than DES in a view of linear and differential attacks. Exhaustive search of the whole key space is infeasible in this suggested scheme, due to the longer key size.

2 Notations

Throughout this paper, the following notations are used.

n_b A binary number n is denoted with the subscript b (e.g. $110000_b = 48$).
n_x A hexadecimal number n is denoted with the subscript x (e.g. $10_x = 16$).
$E_K(P)$ The encryption of 64-bit plaintext block P under the key K.
K_d A 56-bit subkey (of our scheme) which is entered to the (original) DES key scheduling algorithm.
K_i The i-th round 48-bit subkey of K_d.
$E(\cdot)$ The expansion operation of DES.

3 Possible Directions

Our aim is to develop methods that can strengthen DES (against exhaustive search, and differential and linear attacks) that require no additional investments in hardware. We also do not want to slow down the performance of encryption. Since some hardware implementations of DES permit to change S-boxes, we suggest several ways to change them with a positive effect on the security of DES. It is clear that these changes should be made very carefully since the S-boxes are a vital part of DES and the strength of the algorithm crucially depends on their choice. For example in [2] it was shown how even negligible changes to the S-boxes of DES can make differential attacks much more successful. Most previous attempts to suggest alternative S-boxes for DES actually weakened the resultant cryptosystem, since the designers of the new S-boxes were not aware of crucial design principles of DES. We suggest three approaches to reach this goal.

3.1 Key-Dependent S-Box Transformations

Our first suggestion is to create a new S-box by a transformation of an existing S-box by XORing a six-bit key material $K_a[s]$ before the S-box substitution and XORing a four-bit key material $K_b[s]$ to the output of the S-box (see Figure 1). An equivalent description of the same suggestion XORs a 48-bit subkey K_a to the expanded input of the F-function, and XORs a 32-bit subkey K_b to the result of the S-boxes, before the P permutation. Note that the same K_a and K_b are used in all the rounds during encryption. It is clear that this operation is XOR-linear and does not influence differential or linear cryptanalysis. The improved Davies' attack can become only more complicated. Constant K_a has no influence on the attack, since inputs are considered to be random. Constant K_b has no influence on even distributions and produces 16 variations of the odd distributions (permutation of rows and columns), like D_7 (which is used for the improved attack). This way one can gain up to 80 new key bits (In the following subsections we discuss the number of effective key bits that can be gained this way).

These additional XORs with constant keys induce some transformations of S-boxes, so the new set of S-boxes must be calculated from an old set of S-boxes when a new key is chosen. Consider the following transformation of some abstract S-box $\mathcal{S}$, where $K_a[s]$ is the value of the six bits of K_a which enter $\mathcal{S}$ and $K_b[s]$ is the value of the four bits of K_b which are XORed to the output of $\mathcal{S}$:

$$(T_{K_a[s]K_b[s]}\mathcal{S})(x) = \mathcal{S}(x \oplus K_a[s]) \oplus K_b[s] \qquad (1)$$

The effect of $K_a[s]$ on $\mathcal{S}$ is a transposition of rows according to the value of the two outmost bits of $K_a[s]$ and a transposition of columns according to the four inner bits of $K_a[s]$. The effect of $K_b[s]$ on $\mathcal{S}$ is an independent transposition of each row. In order to get a new set of S-boxes one must perform the operation described in equation (1). This approach does not influence differential and linear attacks. Thus it cannot make the cipher stronger or weaker

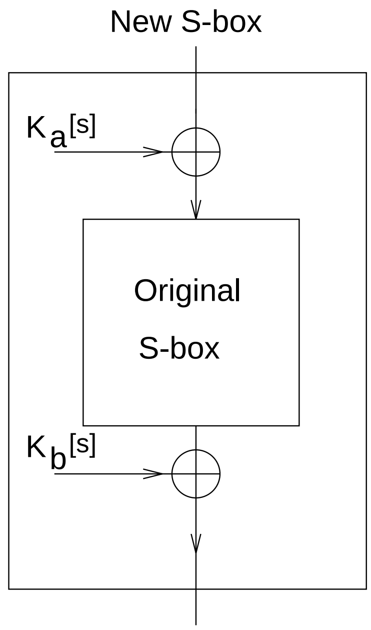

Fig. 1. New S-box.

against these attacks. The approach marginally strengthens the cipher against the improved Davies' attack. However, it increases the key size considerably in order to thwart exhaustive search. In the following subsections, we study several important properties of the resultant cipher.

Equivalent Keys

Definition 1 Two keys K_1 and K_2 are equivalent if $E_{K_1}(P) = E_{K_2}(P)$ for any P.

Assumption 1 If two keys $K = (K_a, K_b, K_d)$, $K^* = (K_a^*, K_b^*, K_d^*)$ are equivalent, then

$$K_i \oplus K_a = K_i^* \oplus K_a^*, \quad i = 1 \ldots 16 \quad \text{and} \quad K_b = K_b^*. \tag{2}$$

Clearly, if (2) holds, then the two keys are equivalent. The question is whether there are equivalent keys.

Theorem 1 If the assumption holds, any key $K = (K_a, K_b, K_d)$ has exactly three equivalent keys.

Proof If two triplets (K_a, K_b, K_d), (K_a^*, K_b^*, K_d^*) are equivalent, according to the assumption, the following relation holds for all 16 subkeys:

$$K_1 \oplus K_1^* = K_2 \oplus K_2^* = \ldots = K_{16} \oplus K_{16}^* = K_a \oplus K_a^* = Const \quad (3)$$

Due to the known regularities in the key scheduling algorithm of DES [7], equation (3) holds only when $K_d \oplus K_d^*$ is one of the four known weak keys. If $K_d \oplus K_d^* = 0$ then $K_a \oplus K_a^* = 0$, and we result with $K = K^*$. Thus, any key K has exactly three equivalent keys.
Q.E.D.

Corollary 1 For any pair of weak keys of DES K_d, K_d^* and for any K_a, there exists K_a^* such that (K_a, K_b, K_d) is equivalent to (K_a^*, K_b, K_d^*), for any K_b.

Proof Equation (3) holds since the weak keys produce sets of constant subkeys.
Q.E.D.

Corollary 2 For any pair of the semi-weak keys of DES K_d, K_d^* and for any K_a, there exists K_a^* such that (K_a, K_b, K_d) is equivalent to (K_a^*, K_b, K_d^*), for any K_b.

Proof A pair of semi-weak keys suits (3) since if a key K_d generates the set of subkeys:

$$K_1 \ldots K_{16} = s, t, t, t, t, t, t, t, s, s, s, s, s, s, s, t.$$

then its counterpart K_d^* generates the set:

$$K_1^* \ldots K_{16}^* = t, s, s, s, s, s, s, s, t, t, t, t, t, t, t, s.$$

Thus $K_i \oplus K_i^* = s \oplus t = K_a \oplus K_a^* = Const \neq 0$ for any $i = 1 \ldots 16$.
Q.E.D.

The key space is divided into equivalence sets of size four: each key has three equivalents. These key equivalences cancel the effect of two key bits.

Since DES S-boxes are nonlinear, set of all transformations $T_{K_a K_b}$ contains 2^{80} different transformations, or in other words no two pairs (K_a, K_b) and (K_a^*, K_b^*) generate the same set of S-boxes. This is proved by the following corollary of Theorem 1.

Corollary 3 If the assumption holds, no two different pairs (K_a, K_b) and (K_a^*, K_b^*) are equivalent.

Proof If there are two equivalent pairs (K_a, K_b) and (K_a^*, K_b^*), then for any K_d, (K_a, K_b, K_d) is equivalent to (K_a^*, K_b^*, K_d^*), and by the proof of Theorem 1, no two equivalent keys have the same K_d.
Q.E.D.

Complementation Property In this section we show that the method of invariant S-box transformations possesses a strong complementation property, which cancels the effect of 32 out of the 48 bits of K_a.

Consider a pair of keys (K_a, K_b, K_d) and (K_a^*, K_b, K_d). K_a can be divided into two keys: 16 bit internal key K_a^i and 32 bit external key K_a^e, where $K_a = E(K_a^e) \oplus E^*(K_a^i)$, E is the expansion of DES, and E^* takes bits indexed $2j, 2j+1$ in K_a^i ($j \in [0,7]$) into bits $6j, 6j+1$ while all the other 32 bits are set to zero. Thus, we receive an equivalent description of our scheme that contains the short internal key K_a^i on each round and K_a^e is XORed to both halves of data entering the cipher after IP and leaving the cipher before IP^{-1}. This external key is ineffective since it does not take part in the encryption process. As a result of this and of the equivalences described in the previous section, (K_a, K_d, K_b) can be searched 2^{34} times faster than full exhaustive search with a chosen plaintext attack similar to the attack on the complementation property of DES (factor 2^{32} due to the complementation property and factor 2^2 due to equivalent keys).

For this shortcut we need 2^{32} chosen plaintexts representing all possible values of $P_i = P \oplus (i, i)$ encrypted into C_i under the unknown key. P is encrypted under all keys $K = (K_a, K_b, K_d)$ where $K_a = E^*(K_a^i)$ (i.e., $K_a^e = 0$), and two fixed bits of K_d (one in each half) are zero. Then we search for P_i for which $C_i = C \oplus (i, i)$ (this can be done efficiently with a lookup table). Such a P_i suggests the key $(E(i) \oplus E^*(K_a^i), K_b, K_d)$. Since the key is longer then the blocksize, we need to verify this suggested key with additional ciphertexts. This attack described above uses 32 complementation properties and two key equivalences (see section 3.1). The complementation property of DES is included in these 34 redundancies. Thus, the number of the additional effective key bits is 47 rather than 80.

Weak and Semi-Weak Keys

Definition 2 A key K is called a weak key if $E_K(E_K(P)) = P$ for any P.

Definition 3 A pair of keys K_1 and K_2 are called semi-weak keys if $E_{K_1}(E_{K_2}(P)) = E_{K_2}(E_{K_1}(P)) = P$ for any P.

All the known weak keys of DES have symmetric subkeys, st. $K_1 = K_{16}$, $K_2 = K_{15}$, etc. (actually $K_1 = K_2 = \ldots = K_{16}$). All the known pairs of semi-weak keys of DES have: $K_1 = K_{16}^*$, $K_2 = K_{15}^*$,...,$K_{16} = K_1^*$.

Lemma 1 Let K_d be a weak (or semi-weak) key of DES of the form described above. Then, for any K_a, and K_b, the key $K = (K_a, K_b, K_d)$ is a weak (semi-weak) key of our scheme.

Lemma 2 For any two semi-weak keys of DES (not necessarily from one pair) for which $K_d \oplus K_d^*$ is a weak key, and for any K_a, there exists K_a^* such that for any K_b the pair of keys (K_a, K_b, K_d) and (K_a^*, K_b, K_d^*) is a pair of semi-weak keys of our scheme.

Related Keys Consider a pair of new keys with difference $(\triangle K_a, \triangle K_b, 0)$, where $\triangle K_a^i = 0$. The maximal probability of $\triangle K_a \rightarrow \triangle K_b$ by the F-function of DES is 1/4. Then if the plaintext difference is $(\triangle K_a^e, \triangle K_a^e)$, the ciphertext difference will be the same with probability $\frac{1}{2^{32}}$. Though the keys used are unknown the relation between them leaks relation between possible ciphertexts. We estimate that there are no relations with probability higher than $\frac{1}{2^{32}}$. In order to use this property for an attack, the attacker must be able to choose relations between keys, making such an attack to be of theoretical interest only. There are several ways to avoid weak choices of related keys in this scheme. One of them is to hash the key with a hash function in order to create K_a and K_b. Then, an attacker will not be able to find a pair of keys with the required relation.

Equivalent Representations of Our Scheme This scheme is equivalent to the following scheme which has no K_b, but has two different K_a's used in different rounds: No XOR with the key is performed after the S-boxes. In rounds 1,4,5,8,9,12,13,16 there is a XOR with K_a^1 = original K_a before the S-boxes and in rounds 2,3,6,7,10,11,14,15 there is a XOR with $K_a^2 = E(P(\text{original } K_b)) \oplus K_a^1$ before the S-boxes.

3.2 Key-Dependent Reorderings of the S-Boxes

In [2] it was shown that many changes to the order of the S-boxes can make DES much weaker. One can ask the question whether the standard order of the S-boxes is optimal. In [6] it is stated that somewhat reduced probability p of $\psi \rightarrow 0$ in case of three active S-boxes was designed into DES (design rule S-8 in [6]).

In order to find good orders, we must solve an optimization problem, finding the maximal probability p for each possible order of the eight S-boxes ($7! = 5040$ orders, rather than $8!$, since the analysis is invariant for cyclic reorderings). The

S box	$00xy11_b$	$11xy10_b$	$10xy00_b$
1	14	6	12
2	6	8	10
3	8	8	10
4	8	16	16
5	8	4	8
6	6	8	10
7	8	16	14
8	8	8	10

Table 1. Maximal values corresponding to the entries $00xy11_b \rightarrow 0$, $11xy10_b \rightarrow 0$, $10xy00_b \rightarrow 0$ in difference distribution tables of DES S-boxes.

characteristics $\psi \rightarrow 0$ with only three adjacent active S-boxes takes place in

DES if the input differences of these S-boxes are of the forms $00xy11_b \to 0$, $11xy10_b \to 0$, $10xy00_b \to 0$ respectively. Table 1 describes the standard order of the S-boxes in terms of the maximal entries in the difference distribution tables for each pattern. Table 2 shows the maximal probabilities of the two-round iterative characteristics for the standard order of the S-boxes. We can

Active S-boxes	$Prob(\psi \to 0)(\cdot 2^{-18})$
1,2,3	1120
2,3,4	768
3,4,5	1024
4,5,6	320
5,6,7	896
6,7,8	960
7,8,1	768
8,1,2	480

Table 2. Standard order of the S-boxes, maximal probabilities of two-round iterative characteristic for pattern $\psi \to 0$.

see that p is reached when the first three S-boxes are active, with $p = \frac{1120}{64^3}$. From Table 1 we can see also that the order $S1, S7, S4, \ldots$ [2] (the order of the remaining S-boxes is irrelevant) is the worst, giving the highest probability of two round iterative characteristic ($p = \frac{14 \cdot 16 \cdot 16}{64^3} \approx \frac{1}{73}$). Clearly, any rotation of the order of the S-boxes does not change p, thus, the orders come in sets of eight orders.

A program was written to solve the described optimization problem. It found $136 \cdot 8$ orders for which the maximal probability of a two-round iterative characteristic is as in DES or smaller. Among those, $32 \cdot 8 = 256$ orders have probability lower than in DES: $p = \frac{1024}{64^3} = \frac{1}{256}$.

The improved Davies' attack suggests that $32 \cdot 8$ orders out of $136 \cdot 8$ are weaker than the original order of the S-boxes by a factor of 4–5 in terms of required known plaintexts and complexity of the attack. These are all the orders where S_8 comes after S_2 or S_4. Among the $32 \cdot 8$ *best* differential orders $18 \cdot 8$ are weaker under Davies' attack. These results can be verified easily by looking at Figure 9 in [8].

We performed extensive analysis for one of the *best* orders. Table 3 describes the order: $S2, S4, S6, S7, S3, S1, S5, S8$.

For this particular order the complexity of differential attack becomes $(\frac{1}{256})^6 = 2^{-48}$ instead of $(\frac{1}{234})^6 = 2^{-47.2}$. This is not a major gain against differential attack; however, quite surprisingly this order of S-boxes makes Matsui's linear attack much harder since it requires 2^{53} known plaintexts, and actually it is the best order in the view of both differential cryptanalysis and linear cryptanalysis. The improved Davies' attack requires about 2^{55} known plaintexts to attack this order.

Several papers study the influence of reorderings of the S-boxes in the view of

Active S-boxes	$Prob(\psi \to 0)(\cdot 2^{-18})$
2,4,6	960
4,6,7	896
6,7,3	960
7,3,1	768
3,1,5	384
1,5,8	560
5,8,2	640
8,2,4	1024

Table 3. New order of the S-boxes $S2, S4, S6, S7, S3, S1, S5, S8$: the maximal probabilities of the two-round iterative characteristics with $\psi \to 0$.

linear cryptanalysis [4, 15]. It is stated that the current order of the S-boxes is relatively weak against linear cryptanalysis, and that most of the orders strengthen DES against this attack. We estimate that we can choose out of the pool of $104 \cdot 8$ orders that are strong as DES or better against differential and improved Davies' attacks, at least 32 orders *strong* against differential, linear and improved Davies' cryptanalysis and add 5 new bits to the key which will decide the order of S-boxes in use. This increases the complexity of exhaustive search by a factor of 32. A variant of the attack on the full 16-round DES described in [2] can be performed for this modification of DES with metastructures covering $15 \cdot 8 = 120$ different characteristics which suffice for the all orders. It requires about 2^{51} chosen plaintexts. The 32 best orders of the DES S-boxes are shown in Table 4.

24673158	64273158	12643758	82764513
73158642	76431582	12643875	12673845
16273845	87512643	73158246	76451382
82467513	84512673	15642738	16243758
16243875	75812643	86724513	73458162
75642138	76438152	76458132	26738451
38752461	46731582	62738451	13642758
86427513	15824673	15864273	67384512

Table 4. The best orders of the DES S-boxes against both differential and linear attacks

The approach described in this section strengthens marginally against linear, differential, improved Davies' and exhaustive search attacks.

3.3 Using Alternative DES-Like S-Boxes

One can compute several different sets of S-boxes according to the design principles of DES, and use additional key bits to control which set is used. The key size is increased by $\log(\mathcal{N}_{sets})$ bits. The problem with this approach is that we need to perform a thorough check for each set of S-boxes against all known attacks. The storage of the S-boxes is increased by the factor of $\mathcal{N}_{sets}$, but the scheme is strengthened by a smaller factor.

Another approach is to add new design principles to DES. This was already done in [11]. Their s^3DES S-boxes suit all the published design principles of DES plus one more — two inputs with difference $11xy10_b$ cannot have the same output. This property prohibits three adjacent active S-boxes to cause zero output difference (in DES this is possible with probability about $\frac{1}{234}$ — which was intentionally lowered by the designers [6]). Moreover, in s^3DES all the eight S-boxes should be active in order to have such effect. This lowers the probability of the two-round iterative characteristic used to attack the full 16-round DES to 2^{-96} (independent of the order of the S-boxes). Knudsen[12] found a four-round iterative characteristic of s^3DES. The iteration of this characteristic to 13 rounds has probability about $2^{-66.5}$, but there is a 16-round linear approximation of s^3DES with probability about $\frac{1}{2} \pm 2^{-26}$, for which an attack requires about 2^{43} known plaintexts. However, if we reverse the order of $S1$ and $S2$ of s^3DES, then the best known differential characteristic has probability about $2^{-63.6}$ and the best linear approximation has probability $\frac{1}{2}+2^{-33.7}$ [5]. In [3] it is shown that s^3DES is immune to the improved Davies' attack. Thus, it is predicted that this variant of the s^3DES S-boxes is much more secure than the standard set of S-boxes of DES.

The properties of the S-boxes that we used in sections 3.1 and 3.2 also hold for the S-boxes of s^3DES, and thus this suggestion can be used simultaneously with the previous ones.

3.4 Random Key-Dependent S-Boxes

S-boxes are probably the most studied and still the most mysterious parts of DES. Since most attacks start from an analysis of the S-boxes, one of the methods, that makes cryptanalysis more complex is to hide the S-boxes, and to make them key-dependent. For example, Khufu [16] is more secure than Khafre [16, 2] although they are very similar, only because of the hidden key-dependent S-boxes.

In [2] a study of DES with known random S-boxes is described. It is shown that 97% of the random sets of S-boxes are vulnerable to differential attack with only 2^{21} chosen plaintexts. Here we analyze a more general case of unknown random S-boxes in which the S-boxes are key-dependent. We will show that approximately 10% of the resultant sets of S-boxes are breakable with only 2^{26} chosen plaintexts.

Table 5 describes approximate distribution of patterns of type $00xy00_b \rightarrow 0$ in random S-boxes. This table is a result of testing 100000 sets of eight random S-boxes. We see that patterns $00xy00_b \rightarrow 0$ with probability $\frac{1}{4}$ or better take place only in 1.7% of all the sets and with probability $\frac{1}{8}$ or better in 97% of all sets (as shown in [2]). Thus, in 97% of the sets, some two-round iterative characteristic with pattern $00xy00_b \rightarrow 0$ has probability higher or equal to $\frac{1}{8}$, and the corresponding 13-round characteristic has probability higher or equal to

Number of occurrences	Percentage of random S-box sets for which pattern $00xy00_b \rightarrow 0$ takes place with probability as below or better					
	6/64	8/64	10/64	12/64	14/64	16/64
0	100.0	100.0	100.0	100.0	100.0	100.0
1	100.0	97.1	68.4	27.8	7.5	1.7
2	99.9	85.7	30.9	4.0	0.3	0.0
3	99.4	65.2	10.0	0.4	0.0	0.0
4	97.5	41.7	2.5	0.0	0.0	0.0
5	92.9	22.2	0.5	0.0	0.0	0.0
6	83.6	10.0	0.1	0.0	0.0	0.0
7	69.9	3.8	0.0	0.0	0.0	0.0
8	53.0	1.3	0.0	0.0	0.0	0.0
9	36.0	0.3	0.0	0.0	0.0	0.0
10	21.7	0.1	0.0	0.0	0.0	0.0
11	11.6	0.0	0.0	0.0	0.0	0.0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
24	0.0	0.0	0.0	0.0	0.0	0.0

Table 5. Distribution of patterns $00xy00_b \rightarrow 0$ in random S-boxes.

2^{-18}. However, we do not know exactly which pattern succeeds. There are 24 different patterns of this type (three for each S-box). Therefore, we should use all the 24 characteristics that cover all the possible patterns.

Using the first round trick of the full 16-round attack on DES [2], we get a 2R-attack with 2^{13} structures, each structure contains 2^6 chosen plaintexts producing 2^9 different pairs of each of the three patterns $00xy00_b \neq 000000_b$. Each structure contains a right pair for each pattern with probability $2^{-18} \cdot 2^5 = 2^{-13}$. The plaintext difference is of the form (υ, ψ) where ψ is some input difference in two middle bits to a particular S-box and υ is some output difference of this S-box (at most four bits differ). This attack has a very high S/N since 36 bits of the ciphertext XOR must be zero (or easily predictable) so we can discard almost all wrong pairs.

The attack is as follows: In the first phase of the attack on DES with key-dependent S-boxes we find all the patterns with probability $\frac{1}{8}$ or higher using $8 \cdot 2^{13} \cdot 2^6 = 2^{22}$ chosen plaintexts. During the attack, we do not know a-priori the actual probability of the characteristics. However, after we identify the right pairs, we can find an approximation of the probability of the pattern $00xy00_b \rightarrow 0$ for each S-box by the formula:

$$P(00xy00_b \rightarrow 0) = \sqrt[6]{\frac{N_{rightpairs}}{N_{allpairs}}}. \qquad (4)$$

In the second phase, we perform a massive attack on the concrete patterns which were found in the first phase in which we find most values of the active

S-box, which we denote by S. We use the right pair(s) that we already identified and continue the 2R-attack described above in order to find more right pairs. In the first round we have $\psi \rightarrow \upsilon$. Since we do not know both the S-box S and the key, we will hide the XOR with K_1 on the first round inside the S-box S. Denote by P_L and P_R left and right halves of the plaintext. Then we obtain the following formula for the active S-box:

$$S(P_R[s]) \oplus S(P_R^*[s]) = P_L[s] \oplus P_L^*[s] = \upsilon \tag{5}$$

Only the S-box S is unknown in (5), thus we get four bits of this S-box. There can be 32 different right pairs suggesting new S-box bits. We organize these 32 pairs with their υ in a table. Thus we get $32 \cdot 4$ bits of particular S-box out of total $64 \cdot 4$. In order to find bits of the key we need to know another pattern of type $\psi \rightarrow \phi$ from the last round. Such pattern can be found in a 1R-attack. With the same data which we used to find 32 right pairs ($2 \cdot 32 \cdot 2^{18} = 2^{24}$ chosen plaintexts) we expect to find about four right pairs for a 1R-attack. Now by hiding K_{16} in S-box on the last round we can get a part of another table for the same S-box. These two tables can be used in order to find 5 key bits of $K_1 \oplus K_{16}$. In the last 16th round $\phi \rightarrow \eta$, where ϕ is the output difference of the F-function on the 15th round ($\psi \rightarrow \phi$). Since ϕ differs in inputs to at most six S-boxes, using the same 32 right pairs we can gain up to $6 \cdot 4 \cdot 32$ bits of these S-boxes.

If we have two patterns $\psi_1 \rightarrow \phi_1$ and $\psi_2 \rightarrow \phi_2$ for the same S-box then we can find $48 \cdot 4$ bits of this S-box and a sixth bit of the key $K_1 \oplus K_{16}$. In general each new pattern for S-box reduces the number of the unknown S-box bits by a factor of two. So we need about six different patterns to find all the bits of the S-box except four bits which cannot be found with this method. As seen from Table 4, the chances for several patterns with probabilities greater or equal than $\frac{1}{8}$ are very high. For example, for about 10% of the keys we can find all the bits of the S-boxes except four for each and about 30–40 bits of $K_1 \oplus K_{16}$ using 2^{26} chosen plaintexts and for about 84% it can be found using 2^{29} chosen plaintexts. Later we can complete the key using other auxiliary techniques.

Due to this weakness of the key-dependent random S-boxes of DES, we do not recommend to use this method in our proposed scheme.

4 Modified DES Scheme

In this section we present our concrete scheme of modified DES. We suggest to use s^3DES S-boxes with a reversed order of $S1$ and $S2$, instead of the standard DES S-boxes, since they are more resistant to differential cryptanalysis. It is also possible to use our scheme with the standard DES S-boxes. Our scheme uses keys of the form (K_a, K_b, K_c, K_d) where:

- K_a Consists of 16 bits. It is expanded by E^* (described in section 3.1) to 48 bits and is then XORed to the input of the S-boxes in all the rounds.
- K_b Consists of 32 bits. It is XORed to the output of the S-boxes in all the rounds.

K_c Consists of 5 bits if DES S-boxes are used and consists of 15 bits if s^3DES S-boxes are used. These bits decide the order in which the S-boxes are loaded. Each combination of bits corresponds to one of the *strong* orders of the S-boxes.

K_d Consists of 56-bits. It is loaded to the DES key scheduling algorithm.

Remark: The choice of a 16-bit K_a (rather than 48-bit) eliminates key equivalences and most complementation properties discussed in sections 3.1, 3.1. Two complementation properties still remain — one is the famous complementation property of DES and another is of the following type: Let

$$\begin{aligned} \tilde{P} &= P \oplus 7FFF80007FFF8000_x, \\ \tilde{K}_a &= K_a \oplus C00000C00000_x, \\ \tilde{K}_i &= K_i \oplus FFFFFF000000_x. \end{aligned}$$

Then

$$C = E_K(P) \Rightarrow \tilde{C} = E_{\tilde{K}}(\tilde{P}).$$

The E expansion prohibits existence of this complementation property in DES (see also [10]). In order to eliminate the two complementation properties in our scheme, we can fix one bit of each half of K_d, effectively reducing its size to 54 bits.

The key in our scheme (K_a,K_b,K_c,K_d) consist of $16+32+5+56 = 109$ bits in case of DES S-boxes and $16+32+15+56 = 119$ bits for s^3DES S-boxes. In order to change the keys during encryption, it is now required not only to change the 56-bit DES key. We should also compute the S-boxes and load them to the hardware. The following algorithm generates the S-boxes:

1. Take s^3DES S-boxes (or standard DES S-boxes) and reorder them according to K_c (using K_c as an index to a table of strong orders).
2. For each S-box S_i, $i = 1 \ldots 8$ in the received order, perform the transformation under the two bits $(2i-2, 2i-1)$ of K_a that are XORed to the input of $\mathcal{S}_i$ and the four bits of K_b XORed to the output of $\mathcal{S}_i$ according to the following formula ($K_a[s]$ is used to denote the two bits of K_a padded with four zero bits and $K_b[s]$ is used to denote the four bits of K_b which correspond to the input and the output of the S-box $\mathcal{S}$ respectively):
$$\mathcal{S}_{new}(x) = \mathcal{S}_{original}(x \oplus K_a[s]) \oplus K_b[s] \tag{6}$$
3. The resultant S-boxes are loaded into the hardware and K_d is loaded into DES keys scheduling algorithm.

Those who prefer to simplify key processing slightly (and reduce the space, required for keeping the table of orders), can eliminate the reorderings and ignore K_c, thus using keys of the form (K_a,K_b,K_d). We suggest to use in such case the s^3DES S-boxes, with the reversed order of $S1$ and $S2$(if the DES S-boxes are used, the best order is: $S2, S4, S6, S7, S3, S1, S5, S8$).

We claim that our scheme is more secure than s^3DES or DES against differential, linear and improved Davies' attacks and that an exhaustive search requires more than trillion years on Wiener's machine [20]. Its main advantage is that organizations can increase the security of their systems which use DES in hardware, with the same encryption/decryption speed. An additional advantage is the backward compatibility to the standard DES, that can be achieved by choosing $K_a = K_b = 0$ and K_c which does not modify the order of the S-boxes. Table 6 compares the security of the variants of our scheme and the standard DES.

Scheme	Number of Key Bits	Complexity of Attack: Exhaustive	Differential	Linear	Im.Davies'
DES	56	2^{55}	2^{47}	2^{43}	2^{50}
Our scheme with:					
DES S-boxes	109	2^{107}	2^{51}	2^{51}	2^{52}
DES w/o reorders*	109	2^{107}	2^{48}	2^{53}	2^{56}
s^3DES S-boxes	119	2^{117}	2^{66}	2^{64}	∞
s^3DES w/o reorders	104	2^{102}	2^{66}	2^{64}	∞

*using the best order of S-boxes $S2, S4, S6, S7, S3, S1, S5, S8$.

Table 6. Comparison of our scheme with DES.

5 Summary

In this paper we described methods of strengthening DES against exhaustive search, differential, and linear attacks that can use existing hardware, without slowing encryption speed. We used the fact that there are DES chips on the market that allow the user to load S-boxes of his choice (for example [18]).

The concept of key-dependent invariant S-box transformation was introduced. We showed several ways to expand the key of DES. Possible reorderings of S-boxes were discussed, and orders that are better than the standard order of the S-boxes were shown. We gave an example of better orders for which a linear attack needs 2^{53} plaintexts. We discussed s^3DES S-boxes. We claimed that this set of S-boxes is far more secure than the standard set. We have also shown that the replacement of the standard S-boxes by random key dependent S-boxes might weaken DES.

Finally we suggested a concrete scheme which uses s^3DES S-boxes and has a longer key. This scheme can be used with existing DES hardware and is claimed to be stronger than DES in view of differential, linear and improved Davies' attacks, and especially in view of exhaustive search.

References

1. Thomas A. Berson, *Long key variants of DES*, Advances in Cryptology, Proceedings of CRYPTO'82, pp. 311 – 313, 1982.

2. Eli Biham, Adi Shamir, *Differential Cryptanalysis of the Data Encryption Standard*, Springer-Verlag, 1993.
3. Eli Biham, Alex Biryukov, *An Improvement of Davies' Attack on DES*, Proceedings of EUROCRYPT'94, to appear.
4. Eli Biham, Alex Biryukov, Uwe Blöcher, Markus Dichtl, *Modifications of DES and their Effect on Differential and Linear Cryptanalysis*, unpublished paper, 1994.
5. Ishai Ben-Aroya, Eli Biham, *A Systematic Method to Find Characteristics*, unpublished paper, 1993.
6. Don Coppersmith, *The Data Encryption Standard (DES) and its Strength Against Attacks*, IBM Journal of Research and Development, Vol. 38, No. 3, pp. 243–250, May 1994.
7. D.W. Davies, *Some Regular Properties of the 'Data Encryption Standard' Algorithm*, Advances in Cryptology, Proceedings of CRYPTO'82, pp. 89 – 96, 1982.
8. D.W. Davies, *Investigation of a Potential Weakness in the DES Algorithm*, private communications, 1987.
9. Whitfield Diffie, Martin Hellman, *Exhaustive Cryptanalysis of the NBS Data Encryption Standard*, IEEE Computer, Vol. 10, No. 6, pp. 74 – 84, June 1977.
10. M. Hellman, R. Merkle, R. Schroeppel, L. Washington, W. Diffie, S. Pohlig, P. Schweitzer, *Results of an Initial Attempt to Cryptanalyze the NBS Data Encryption Standard*, Information Systems Laboratory Report, Stanford University, November 1976.
11. Kwangjo Kim, Sangjun Park, Sangjin Lee, *Reconstruction of s^2DES S-boxes and their Immunity to Differential Cryptanalysis*, Proceedings of JW-ISC93 – Korea-Japan Joint Workshop on Information Security and Cryptology, Seoul, Korea, October 24–26, 1993.
12. Lars Knudsen, *An Analysis of Kim, Park and Lee's DES-like S-boxes*, private communication, June 1993.
13. Lars Knudsen, *On the Design of Secure Block Ciphers*, Fast Software Encryption, Proceedings of Cambridge security workshop, pp. 9 – 11, December 1993.
14. Mitsuru Matsui, *Linear Cryptanalysis Method for DES Cipher*, Proceedings of EUROCRYPT'93, pp. 386 – 397, 1993.
15. Mitsuru Matsui, *On Correlation Between the Order of S-boxes and the Strength of DES*, Proceedings of EUROCRYPT'94, to appear.
16. Ralph C. Merkle, *Fast Software Encryption Functions*, Lecture Notes in Computer Science, Advances in Cryptology, Proceedings of CRYPTO'90, pp. 476 – 501, 1990.
17. National Bureau of Standards, *Data Encryption Standard*, Federal Information Processing Standards Publication 46, January 1977.
18. SuperCrypt, *High Speed Cryptographic Data Security Element*, Preliminary Data Sheet.
19. J.-J. Quisquater, Y. Desmedt, M. Davio, *The Importance of 'Good' Key Scheduling Schemes*, Proceedings of CRYPTO'85, pp. 537 – 542, 1985.
20. M. J. Wiener, *Efficient DES Key Search*, technical report TR-244, School of Computer Science, Carleton University, Ottawa, Canada, May 1994. Presented at the Rump session of CRYPTO'93, August 1993.