A Related Key Attack on the Feistel Type Block Ciphers

Ali Bagherzandi[1,2], Mahmoud Salmasizadeh[2], and Javad Mohajeri[2]
(Corresponding author: Ali Bagherzandi)

Computer Engineering Department, Sharif University of Technology[1]
Electronic Research Center, Sharif University of Technology[2]
P. O. Box 11155-8639 Azadi Avenue 14588 Tehran, Iran (Email: bagherzandi@ce.sharif.edu)
(Received Jan. 3, 2006; revised and accepted Mar. 3, 2006)

Abstract

In this paper we show that Biham's chosen key attack can be generalized to include any block cipher and we give a low complexity chosen key attack on any Feistel type cipher. Then we show that the irregularities in the shift pattern of DES key schedule algorithm is not sufficient for the cryptosystem to resist against related key attacks. We have realized our proposition by a counter example in which the E-box of DES is slightly modified whiles other components and among those, the shift pattern in key schedule algorithm is kept unchanged. We have applied a new related key attack on the resulting DES-like cryptosystem and demonstrated that the security of the system decreases drastically.

Keywords: Chosen key attack, differential related key attack, DES, feistel type cipher

1 Introduction

The major attacks on Feistel type block ciphers fall into three groups of differential analysis [1], linear analysis [10] and related-key attacks [2]. The related key attack introduced by Biham [2] is a chosen key attack assaulting the key schedule algorithm of the cipher. In this attack, attacker obtains the encryption of certain plaintexts under several keys having certain relationships with each other. The goal in this type of attack is to reveal the secret key of the cryptosystem and it can be accomplished whenever the attacker can chose the relationship between unknown keys. So, it is best practical on key-exchange protocols where key-integrity is not guaranteed thus an attacker can flip the key bits without knowing the key itself [6, 8, 9]. In differential related-key attacks [6], the relationship between the unknown keys is their difference; adversary can choose the difference between the keys and seek for the keys themselves. The plaintext ciphertext pairs can be chosen or known which categorizes the related key attack into two groups of known-related-key attacks and chosen-related-key attacks.

In order to strengthen DES, several extensions have been developed including Triple-DES, DES-X, Biham-DES, DES-X and NewDES. These extensions improved the resistance of DES against exhaustive search, linear cryptanalysis and differential cryptanalysis; but not any significant improvement has been gained against related-key attacks and several related-key cryptanalysis have been developed against these extensions [8, 9]. Even in the case of DES-X there has been introduced some novel key related attacks [9]. It is believed that the irregularities in the shift pattern of DES key schedule algorithm makes DES immune against related key attacks [2]. Recently, it has been shown that 1-bit shift in some less number of rounds in DES key schedule makes DES vulnerable to a related-key attack [7].

In this paper we show that Biham's chosen key attack can be generalized to include any block cipher. Though this attack is applicable to the ciphers having block length less than key size, it leads to a low complexity chosen key attack against any Feistel type cipher. Then we show that the resistance of DES against differential related key attack is not merely upon the irregularities in the shift pattern of its key schedule algorithm which is widely believed since the time it was mentioned in [2]. We have realized our proposition by mounting a differential related key attack similar to that of [7] against a DES-like cipher with the same key schedule algorithm as the original DES to demonstrate that the security of the system decreases drastically. Our variant of DES is different from the original DES in one index of its E-box. This makes it nonsense to build the immunity of DES against related key attacks merely upon shift pattern irregularities.

In Section 2 we formalize the chosen key attack and give a low complexity attack scenario against Feistel type ciphers. Then in Section 3 we discuss the original DES algorithm and its mixed transformation representation in brief and introduce our DES-like encryption scheme. Then in Section 4 we discuss our differential related key attack against the proposed DES-like system.

2 Generalized Biham Chosen Key Attack

The chosen key attack is based on the observation that in many block ciphers we can view the key scheduling algorithm as a set of algorithms each of which extracts one particular subkey from the subkeys of previous rounds. If all the algorithms of extracting the subkeys of the various rounds are the same then for a given a key we can shift all the subkeys one round backwards and get a new set of valid subkeys which can be derived from some other keys [12]. These keys are called related. The same argument is true for slide attacks introduced in [3, 4]. Slide attacks can be viewed as a particular case of related-key attack in which the relation is between the key and itself [1]. Extending this idea, in this section, we introduce a generalization of chosen key attack to include any block cipher and give low complexity attack scenarios against Feistel type ciphers. Several related key analysis of block ciphers can be found in [2, 6, 8, 9].

Definition 1. (Block cipher): *Our abstraction of block cipher is a triple $C(n, r, K)$ in which n is the data block length, r the number of rounds and K the underlying key. We also write $K \rightarrow (k_1, k_2, \ldots, k_r)$ to denote the derivation of round keys and $F(x, k_i)$ to denote the round function of $C(n, r, K)$. we also write $p \xrightarrow{K} c$ to indicate the encryption process.*

Definition 2. (Slid keys): *The keys (K, K') are called slid keys if they led to the same derivation of round keys but one round out of phase. More formally (K, K') are called slid pairs if*

$$K \rightarrow (k_1, k_2, \ldots, k_r) \Longleftrightarrow K' \rightarrow (k_2, k_3, \ldots, k_r, k_{r+1}).$$

If $k_{r+1} = k_1$ then we call (K, K') strong slid keys.

Definition 3. (Slid pair): *A pair of plaintext-ciphertext pairs $((p, c), (p', c'))$ in which p and p' are encrypted under the keys K and K' respectively, is called slid pair if:*

1) (K, K') are strong slid keys,

2) $P' = F(P, k_1)$.

Theorem 1. (Birthday Paradox): *Let $H : M \rightarrow C$ be a random function in which $|C| = 2^m$ and let X and Y be two randomly chosen subsets of M with $|X| = |Y| = n$. If $n \geq 2^{\frac{m}{2}}$ then the probability of finding one collision between sets X and Y exceeds $\frac{1}{2}$. [11]*

Proposition 1: The pair $((p, c), (p', c'))$ is slid pair if and only if it satisfies the following conditions.

1) $p' = F(p, K_1)$,

2) $c' = F(c, K_1)$.

Proof. If $((p, c), (p', c'))$ is a slid pair then by Definition 3 $p' = F(p, k_1)$ and since encryption process for p' is one round shifted so $c' = F(c, k_1)$. On the other hand suppose one can found k_1 satisfying both conditions above and consider (c, c') satisfying $c' = F(c, k_1)$ since $p' = F(p, k_1)$, k_1 is the first and last subkey of the keys K and K' respectively. Reversing the encryption process one gets the intermediate encryptions of p' one round shifted as that of p. □

Proposition 2: By $2^{\frac{n}{2}}$ random n-bit plaintext-ciphertext pair encrypted under key K and $2^{\frac{n}{2}}$ plaintext-ciphertext pair encrypted under key K' one expects to find a slid pair.

Proof. Consider plaintexts selected to encrypt under K and K' as the set X and Y respectively. According to birthday paradox the sets X and Y have a common element (so two elements with a determined relation) with a probability greater than $\frac{1}{2}$. Thus, exposing to encryption under K and K' we expect to have a slid pair. □

Proposition 3: By $2^{\frac{n}{4}}$ random n-bit plaintext-ciphertext pair of the form $P_i = L_i \parallel X$ encrypted under key K by a Feistel type cipher and $2^{\frac{n}{4}}$ plaintext-ciphertext pair of the form $P_j = X \parallel R_j$ encrypted under key K' one expects to find a slid pair.

Proof. Apply the birthday paradox to the half of the data block. □

Attack model for general case:

1) Acquire $2^{\frac{n}{2}}$ plaintext-ciphertext pair encrypted under key K and $2^{\frac{n}{2}}$ plaintext-ciphertext pair encrypted under key K'.

2) For each pair $((p, c), (p', c'))$ solve the simultaneous equation: $\begin{cases} F(p, k) = p' \\ F(c, k) = c' \end{cases}$

According to Proposition 1 if the system has solution k then $((p, c), (p', c'))$ is a slid pair. And according to Proposition 2 there is at list one slid pair. Thus in worst case performing $O(2^n)$ task, one can retrieve k_1.

Attack model for special case of Feistel ciphers:

In Feistel ciphers the round function is $F(L \parallel R) = (R \parallel L \oplus f(R))$. So the slid pair can be recognized more easily.

1) Acquire $2^{\frac{n}{2}}$ plaintext-ciphertext pair encrypted under key K and $2^{\frac{n}{2}}$ plaintext-ciphertext pair encrypted under key K'.

2) Sort (p_i, c_i) encrypted with key K according to right halves of p_i and c_i and sort (p'_i, c'_i) encrypted with key K' according to left halves of p'_i and c'_i.

Now in order to find a slid pair it suffices to compare the right half of p_i with left half of p'_i and right half of c_i with left half of c'_i. This requires $O(2^{\frac{n}{2}})$ task since the lists are sorted.

One may reduce the number of plaintext-ciphertext pairs required, by a kind of chosen plaintext attack. Note

that both of the above attacks are known plaintext attacks. A chosen plaintext attack may run as follows:

1) Select $2^{\frac{n}{4}}$ plaintexts of the form $p_i = L_i \parallel X$ for some constant X to submit to encrypt under the key K and another $2^{\frac{n}{4}}$ plaintexts of the form $p_j = X \parallel R_i$ to submit to encrypt under the key K'. According to Proposition 3 there is at least one slid pair.

2) Like known plaintext attack Sort (p_i, c_i) encrypted with key K according to right halves of p_i and c_i and sort (p'_i, c'_i) encrypted with key K' according to left halves of p'_i and c'_i.

Again in order to find a slid pair it suffices to compare the right half of p_i with left half of p'_i and right half of c_i with left half of c'_i. Thus by $O(2^{\frac{n}{4}})$ plaintext-ciphertext pair and $O(2^{\frac{n}{4}})$ offline work one can extract the underlying key.

3 Mixed Transformation Representation of DES

DES encryption algorithm was introduced in 1972 by the researchers of IBM. It was accepted as a federal standard the National Security Agency (NSA) in 1977. It is a 16-rounded Feistel type block cipher which uses a 56 bit key and operates on a block size of 64 bits. The encryption process consists of sixteen Feistel iterations surrounded by two permutation layers: An initial bit permutation (IP) in the input, and its inverse, IP^{-1}, in the output. The functionality of each of the Feistel iterations can be summarized as bellow:

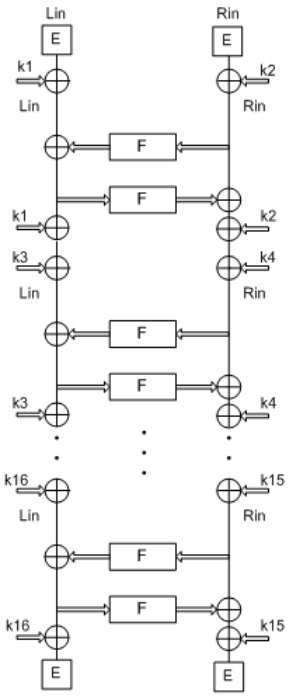
1) A 32-bit block is expanded to 48 bits through an expansion permutation function and xored with the i-th subkey.

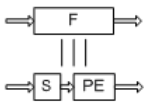
2) The output of the phase 1) is divided into eight 6-bit blocks. Each 6-bit block is then passed through one of the $S_1, S_2, \ldots, S_8$ s-boxes. Each s-box is a non-linear function which maps a 6-bit input data to 4-bit output data.

3) The output of $S_1, S_2, \ldots, S_8$ is concatenated and passed through the permutation P to obtain the final output of each round.

Davio et al. [5] proposed several equivalent representations that can be utilized for cryptanalysis or efficient implementation of DES. Here, we will use the mixed transformation representation depicted in Figure 1. The initial and the final permutation are omitted since they do not contribute to the security of the cipher. This representation is also used in [7] to attack DES.

In order to generate sixteen 48-bit sub keys from the 56-bit key, the following process is used: First, the key is loaded and manipulated by the permutation choice I (PC1) and then the key block is halved. Each half is





The P function

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

The E function

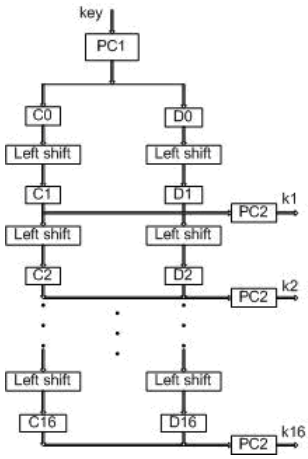
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

Figure 1: DES block diagram [5]

rotated by 2 bits in every round except in the 1st, 2nd, 9th and the last rounds. The 48 bits of the 56 bits are chosen according to permutation choice II (PC2). Figure 2 shows the key generation process for DES and the PC2 function.

4 Differential Related Key Analysis of DES

In this section, we will analyze our attack approach against a DES like block cipher. Our variant of DES has the same components as the original DES except that its expansion permutation function has been slightly mod-



The PC2 function

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

Figure 2: Key generation block diagram of DES

The E function

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

The modified E function

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

1≤x≤16
16≤y≤32

Figure 3: The original and modified E function

ified. The modified table as well as the original one is shown in Figure 3.

By now let's consider two keys K' and K'' such that:

$$\begin{aligned} \triangle C_1 &= C' \oplus C'' \\ &= 11111111111111111111111111111111; \quad (1) \\ \triangle D_1 &= D' \oplus D'' \\ &= 00000000000000000000000000000000. \quad (2) \end{aligned}$$

Where C'_1, D'_1, C''_1, and D''_1 are the left and right halves of the keys and respectively as shown in Figure 2.

By selecting K' and K'' such that Equations (1) and (2) are satisfied, obviously we have

$$\begin{aligned} \mathrm{LSH}_1(\triangle C_i) &= \triangle C_i; \\ \mathrm{LSH}_1(\triangle D_i) &= \triangle D_i, \end{aligned}$$

where, $\mathrm{LSH}_1(x)$ indicates i-bit circular left shift of x. Thus, for $1 \leq i \leq 16$ we have:

$$\begin{aligned} \triangle K_i &= \mathrm{PC2}(C'_i \parallel D'_i) \oplus \mathrm{PC2}(C''_i \parallel D''_i) \\ &= \mathrm{PC2}(\triangle C_i \parallel \triangle D_i) = \mathrm{PC2}(1^{28} \parallel 0^{28}) \\ &= 1^{24} \parallel 0^{24}, \end{aligned}$$

where $x \parallel y$ stands for the concatenation of x and y. So, $\triangle K_i = \text{const}$.

On the other hand according to the key schedule algorithm (Figure 2), we obtain:

$$\begin{aligned} \triangle k_i &= \triangle k_{i-2} \Rightarrow k'_i \oplus k''_i = k'_{i-2} \oplus k''_{i-2} \Rightarrow k'_i \oplus k'_{i-2} \\ &= k''_i \oplus k''_{i-2}. \quad (3) \end{aligned}$$

Now consider two plaintexts x' and x'' encrypted by the keys K' and K'' respectively satisfying Conditions (4) and (5) bellow:

$$\begin{aligned} \triangle L_{in} &= L'_{in} \oplus L''_{in} \\ &= 11111111111111111111111111111111; \quad (4) \\ \triangle R_{in} &= R'_{in} \oplus R''_{in} \\ &= 00000000000000000000000000000000, \quad (5) \end{aligned}$$

where L_{in} and R_{in} are the left and right halves of plaintext x.

Let's see what happens to the plaintext during iterations of encryption algorithm.

Let $\triangle L^1_{in}$, denotes the difference between left part of plaintexts x' and x'' before the i-th round of encryption by the keys k' and k'' respectively.

In the case of $\triangle L^1_{in}$ and $\triangle R^1_{in}$ according to Figure 1 it is clear that:

$$\begin{aligned} \triangle L^1_{in} &= (K'_2 \oplus E(L'_{in})) \oplus (K''_2 \oplus E(L''_{in})) \\ &= (\triangle K_2) \oplus (E(L'_{in})) \oplus (E(L''_{in})); \quad (6) \\ \triangle R^1_{in} &= (K'_1 \oplus E(R'_{in})) \oplus (K''_1 \oplus E(R''_{in})) \\ &= (\triangle K_1) \oplus (E(R'_{in})) \oplus (E(R''_{in})), \quad (7) \end{aligned}$$

where E is the expansion permutation function at the first layer. Since E is linear we have:

$$\begin{aligned} E(L'_{in}) \oplus E(L''_{in}) = E(L'_{in} \oplus L''_{in}) = E(\triangle L_{in}); \\ E(R'_{in}) \oplus E(R''_{in}) = E(R'_{in} \oplus R''_{in}) = E(\triangle R_{in}), \end{aligned}$$

by applying expansion function E on the values of $\triangle L_{in}$ and $\triangle R_{in}$ as they are in Conditions (4) and (5) we obtain:

$$\begin{aligned} & E(\triangle L_{in}) = E(\triangle R_{in}) \\ = \; & 11111111111111111111111100000000000000000000000000, \end{aligned}$$

which is equal to $\triangle k_i$ as it is in Equation (3).

Thus according to Equations (6) and (7)

$$\begin{aligned} \triangle L^1_{in} = \triangle k_2 \oplus E(\triangle L_{in}) = \triangle k_2 \oplus \triangle k_2 = 0; \\ \triangle R^1_{in} = \triangle k_1 \oplus E(\triangle R_{in}) = \triangle k_1 \oplus \triangle k_1 = 0. \end{aligned}$$

But in the case of $\triangle L^i_{in}$ and $\triangle R^i_{in}$ for $i > 1$ there is non linear function F. However, since the difference of inputs to s-boxes will remain zero, this is not a problem. For example in the case of $\triangle L^2_{in}$ and $\triangle R^2_{in}$ we will have:

$$\begin{aligned} \triangle L^2_{in} &= L'^2_{in} \oplus L''^2_{in} \\ &= (k'_4 \oplus k'_2 \oplus L'^1_{in} \oplus F(R'^1_{in})) \\ &\quad \oplus (k''_4 \oplus k''_2 \oplus L''^1_{in} \oplus F(R''^1_{in})) \\ &= \triangle k_4 \oplus \triangle k_2 \oplus \triangle L^1_{in} \oplus F(R'^1_{in}) \oplus F(R''^1_{in}). \end{aligned}$$

Now since $\triangle L^1_{in} = 0$ and $F(x) = PE(S(x))$ and PE is a linear function, we have:

$$\triangle L^2_{in} = \triangle k_4 \oplus \triangle k_2 \oplus PE(S(R'^1_{in})) \oplus S(R''^1_{in}).$$

But according to pairs XOR distribution table of s-boxes, whenever the difference of input to an s-box equals to zero, the difference of corresponding outputs will be zero too. So, the above formula reduces to:

$$\triangle L^2_{in} = \triangle k_4 \oplus \triangle k_2.$$

The same argument can be carried out for $\triangle R^2_{in}$ and consequent data parts. By doing so, we obtain:

$$\begin{aligned} \triangle L^i_{in} &= \triangle k_{2i} \oplus \triangle k_{2i-2}; \\ \triangle R^i_{in} &= \triangle k_{2i-1} \oplus \triangle k_{2i-3}. \end{aligned}$$

And according to Equation (3) this will leads to $\triangle L^i_{in} = \triangle R^i_{in} = 0$ for $1 \leq i \leq 8$.

Now, we encrypt two plaintexts x' and x'' satisfying Conditions (1) and (2) by the keys K' and K'' satisfying

Table 1: The complexity of our attacks with respect to block size -n

Cipher and Attack Type	Plaintext-Ciphertext Pairs Required	Offline Work
General block cipher Known Plaintext Attack	$O(2^{\frac{n}{2}})$	$O(2^n)$
Feistel Type Known Plaintext Attack	$O(2^{\frac{n}{2}})$	$O(2^{\frac{n}{2}})$
Feistel Type Chosen Plaintext Attack	$O(2^{\frac{n}{4}})$	$O(2^{\frac{n}{4}})$

Conditions (4) and (5) respectively in order to get the corresponding ciphertexts y' and y''.

According to Figure 1 we have,

$$R'^{8}_{in} = E(R'_{out}) \oplus k'_{15} \oplus F(E(L'_{out}) \oplus k'_{16}); \quad (8)$$
$$R''^{8}_{in} = E(R''_{out}) \oplus k''_{15} \oplus F(E(L''_{out}) \oplus k''_{16}). \quad (9)$$

Thus, we can search for subkeys k'_{16} satisfying the Conditions (8) and (9).

This requires 8×2^6 search operation; for we don't need to search among all possible subkeys k'_{15}. Instead we group carefully the subkeys into 6-bit groups that bits in one group affect just one s-box. By knowing the subkey k'_{16} we can easily evaluate the key K.

5 Conclusions

We showed that Biham's chosen key attack can be generalized to include any block cipher regarding it as a random substitution permutation network. We also proposed a general attack scenario with low plaintext-ciphertext and offline complexity against any Feistel type cipher. The complexity of these attacks is summarized in Table 1. Then we introduced a differential related key attack to a variant of DES which has the same shift pattern as the original one. This violates the previous belief that the strength of DES against related key attacks, is due to the irregularities in the shift pattern of its key schedule. We realized this, by applying our attack on a variant of DES with a slightly modified E-box -as indicated in Figure 3- to demonstrate that the security of the system decreases drastically.

References

[1] E. Biham and A. Shamir, "Differential cryptanalysis of DES-like cryptosystems," *Journal of Cryptology*, vol. 1, pp. 3-72, 1991.

[2] E. Biham, "New types of cryptanalytic attacks using related keys," *Journal of Cryptology*, vol. 7, no. 4, pp. 229-246, 1994.

[3] A. Biryukov and D. Wagner, "Slide attacks," *Advances in Cryptology - Proceedings of FSE' 99*, LNCS, pp. 245-259, L.R. Knudsen, editor, Springer-Verlag, 1999.

[4] A. Biryukov and D. Wagner, "Advanced slide attacks," *Advances in Cryptology - Proceedings of Eurocrypt '00*, LNCS 1807, pp. 589-606, B. Preneel, editor, Springer-Verlag, 2000.

[5] M. Davio, Y. Desmedt, M. Fosseprez, R. Govaerts, J. Hulsbosch, P. Neutens, P. Piret, J. J. Quisquater, J. Vandewalle, and P. Wouters, "Analytic characteristics of DES," *Advances in Cryptology, Proceedings Crypto '83* D. Chaum (Ed.), Plenum Press, New York, pp. 171-202, 1984.

[6] G. Jakimoski and Y. Desmedt, "Related-key differential cryptanalysis of 192-bit key AES variants," *Tenth Annual Workshop on Selected Areas of Cryptography*, LNCS 3006, pp.208-221, Springer-Verlag, 2004.

[7] G. Jakimoski and Y. Desmedt, "On resistance of DES to related-key differential cryptanalysis," ePrint archive no. 2005/085, http://eprint.iacr.org/2005/084.pdf.

[8] J. Kelsey, B.Schneier, and D. Wagner, "Key schedule cryptanalysis of IDEA, GDES, GOST, SAFER and triple DES," *Advances in Cryptology, Proceedings Crypto '96*, LNCS 1109, pp. 237-252, 1996.

[9] J. Kelsey, B. Schneier, and D. Wagner, "Related-key cryptanalysis of 3-WAY, Biham-DES, CAST, DES-X, NewDES, RC2, and TEA," *ICICS' 97 Proceedings*, pp. 233-246, Springer-Verlag, Nov. 1997.

[10] M. Matsui, "Linear cryptanalysis method for DES cipher," *Advances in Cryptography Eurocrypt '93*, pp. 386-397, 1994.

[11] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, p. 53, CRC Press, 1996.

[12] G. Piret, M. Ciet, and J. Quisquater, "Related key and slide attacks: Analysis, connections, and improvements," *Proceedings of the 23rd Symposium on IT in Benelux*, pp. 315-325, 2002.

Ali Bagherzandi is a senior B.Sc. student in Computer Engineering department of Sharif University of Technology, Tehran, Iran, studying a dual degree program of computer science and hardware engineering. He has joined electronic research center of Sharif University of Technology as a research assistant in Jan 2005. He is the principal author of 5 refereed papers. His main research interests include theory of computation, foundations of cryptography and data and network security.

Mahmoud Salmasizadeh received the B. S. and M. S. degrees in Electrical Engineering from Sharif University of Technology in Iran, in 1972 and 1989, respectively. He also received the Ph.D. degree in Information Technology from Queensland University of Technology in Australia, in 1997. Currently he is an assistant professor in Electronic Research Center and adjunct assistant professor in

Electrical Engineering Department at Sharif University of Technology, Tehran, Iran. His research interests include cryptology and network security. He is the founding member and the head of scientific committee of Iranian Society of Cryptology.

Javad Mohajeri received his B.Sc. in Mathematics from Isfahan University, Isfahan, Iran in 1986 and his M.S. in Mathematics from Sharif University of Technology, Tehran, Iran in 1990. He is a lecturer in the electronic research center of Sharif University of Technology. Javad Mohajeri has published 31 papers in refereed journals and conferences. He is a member of founding committee of Iranian Society of Cryptology. He was the chairman of the technical program committee of the 2nd Iranian society of cryptology conference on cryptology communications and computer security. His research interests include design and analysis of cryptographic algorithms, data security, secret sharing schemes and PKI.