New Multiple Encryption for Making Double Encryption Secure Against Meet-in-the-Middle and Related-Key Attacks

Takashi Nishide[1], Shinichi Yoshinaga[1], Rishiraj Bhattacharyya[2], Mridul Nandi[2], Bimal Roy[2], and Kouichi Sakurai[1]

[1] Kyushu University, Japan.
[2] Indian Statistical Institute, India.
{nishide,yoshinaga,sakurai}@inf.kyushu-u.ac.jp
{rishi_r,bimal}@isical.ac.in, mridul.nandi@gmail.com

Abstract. There is no practical way to attack DES more efficiently than an exhaustive key search attack. However, the key length of DES is not sufficient now, so DES can be attacked within the realistic time and cost with the current computing power. By using the longer key, we may be able to cope with such threats. For example, a DES variant called Double DES repeats DES encryption twice with two different keys. However, double DES can also be attacked by the meet-in-the-middle attack with $O(2^n)$ memory space and $O(2^n)$ times DES operations where n is the bit length of a single DES key and an adversary uses the data which is computed between two DES operations. To prevent such a meet-in-the-middle attack, we can hide the middle data by some methods so that the adversary cannot compute the middle data. One of such methods is called DES-EXE which takes the exclusive-ORed middle data of double DES with a new third 64-bit key. However, it was indicated that DES-EXE was also vulnerable to the elaborate meet-in-the-middle attack and the other one which uses the related keys. In the meet-in-the-middle attack using related keys, the adversary uses a pair of a plaintext and ciphertext encrypted under a key K_1 and another pair of the same plaintext and the ciphertext encrypted under a different key K_2 where K_1 is related to K_2 in some mathematical way and the relationship is known to the adversary. In this work, we propose a new double DES variant (that we call DES-XEEX) that adds the exclusive-ORed data with new third and fourth keys outside two DES operations. The adversary can not apply the same elaborate meet-in-the-middle attack as in DES-EXE and can not use related keys, so the proposed scheme is more secure than DES-EXE against the meet-in-the-middle and related-key attacks. Our construction is generic and applicable to any block cipher such as AES to have a longer key effectively.

Keywords: DESX, Double DES, Meet-in-the-middle & Related-key attacks, Multiple Encryption

1 Introduction

The size of a key used in a block cipher essentially depends on limits on adversary's computational power. If the key is too short, the adversary can break the cipher by simple brute force attack. It is of immense importance to find out whether there is a general way to increase the key size of block ciphers to make exhaustive key search attacks on block ciphers infeasible. It will be useful when the key size of the basic block cipher becomes insufficient or the basic block cipher is not immune to exhaustive key search attacks anymore.

One way is to design a new block cipher algorithm (such as AES) with longer keys instead of the older algorithm (like DES). However such an action needs to be carried out through the entire industry and it is very time-consuming. The other popular way is to use multiple encryption to make it immune to exhaustive key search attacks. The idea is to reuse the basic block cipher and possibly reuse the existing hardware.

For DES encryption, several constructions (we call them DES variants) exist in the literature. Although these DES variants are designed such that they can have larger key space, the meet-in-the-middle [7] and related-key attacks [3] can be mounted on these variants.

In the meet-in-the-middle attacks, an adversary is assumed to have plaintext-ciphertext pairs, and in the related-key attacks, the adversary is assumed to have at least two plaintext-ciphertext pairs where one pair is encrypted under K_1 and the other pair is encrypted under K_2 and the relationship between K_1 and K_2 is known to the adversary though K_1 and K_2 are unknown to the adversary. Therefore, in the related-key attack model, we consider a very strong adversary.

As shown in Section 3.1, a simple multiple encryption cannot increase the key space, but as a tradeoff, the adversary needs more memory space to mount the attack successfully. The most well-known DES variant to strengthen DES is called "triple DES." One disadvantage of triple DES is that it is slower than other constructions because of three DES operations and it may not be acceptable because of the performance requirement in some situations. DESX proposed

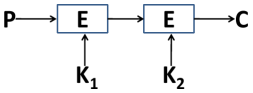

Fig. 1. Double DES

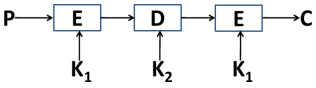

Fig. 2. Triple DES

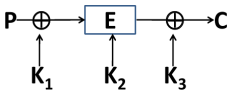

Fig. 3. DESX

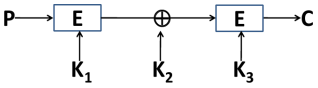

Fig. 4. DES-EXE

by Rivest uses only one DES encryption, but as shown by Phan and Shamir [12], it is vulnerable to the related-key attack.

Our Result. In this work, we propose a new construction of multiple encryption with pre and post processing. We consider to construct a multiple encryption scheme that can withstand the related-key and meet-in-the-middle attacks and that is more efficient than triple DES. We introduce the construction that we call DES-XEEX and discuss its security. We stress that our construction is generic and applicable to any block cipher.

2 Multiple Encryption: Several DES Variants

In this section, we give the descriptions of several DES variants that are designed such that we can have longer keys. These constructions use multiple encryption to realize stronger security.

2.1 Double DES

Double DES is the simplest variant. In Double DES, we use two keys K_1 and K_2 and the ciphertext C and plaintext P are computed as follows (Fig. 1):

$$C = E_{K_2}(E_{K_1}(P)), \quad P = D_{K_1}(D_{K_2}(C))$$

2.2 Two-key Triple DES

Two-key triple DES was proposed by Tuchman [14]. In two-key triple DES, we do three times DES operations with two different keys. Note that triple DES is designed such that it becomes equivalent to single DES when K_1 is equal to K_2 for compatibility.

In triple DES, we use two keys K_1 and K_2 and the ciphertext C and plaintext P are computed as follows (Fig. 2):

$$C = E_{K_1}(D_{K_2}(E_{K_1}(P))), \quad P = D_{K_1}(E_{K_2}(D_{K_1}(C)))$$

2.3 DESX

DESX was proposed by Rivest in 1984. DESX was implemented in the products of RSA Data Security, Inc., and described in the documentation for these products [13]. Kilian and Rogaway [10] analyzed the security of DESX. In DESX, we use three keys K_1, K_2, K_3 and the ciphertext C and plaintext P are computed as follows (Fig. 3):

$$C = K_3 \oplus E_{K_2}(P \oplus K_1), \quad P = K_1 \oplus D_{K_2}(C \oplus K_3)$$

2.4 DES-EXE

DES-EXE was proposed in [8]. In DES-EXE, we use thee keys K_1, K_2, K_3 and the ciphertext C and plaintext P are computed as follows (Fig. 4):

$$C = E_{K_3}(K_2 \oplus E_{K_1}(P)), \quad P = D_{K_1}(K_2 \oplus D_{K_3}(C))$$

3 Meet-in-the-Middle Attack

Although the DES variants are designed such that the exhaustive key search attack is more infeasible, some of the DES variants are vulnerable to the meet-in-the-middle attack proposed by Diffie and Hellman [7]. In this section, we explain how the meet-in-the-middle attack can reduce the key space for the exhaustive key search attack.

3.1 Application to Double DES

Intuitively, the key space for double DES seems 2^{n*2} where $n(= 56)$ is the actual bit length of a single DES key, but with the meet-in-the-middle attack, the key space is reduced to be 2^n with the tradeoff that we need the $O(2^n)$ memory space. The attack works as follows (Fig. 5):

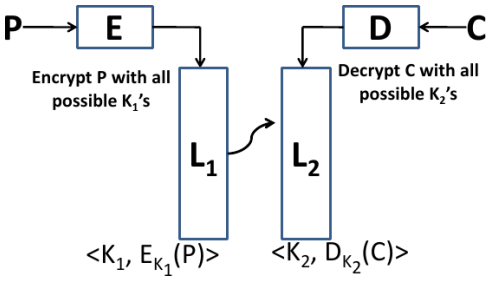

Fig. 5. Meet-in-the-Middle Attack on Double DES

1. The adversary obtains a plaintext-ciphertext pair (P, C) where $C = E_{K_2}(E_{K_1}(P))$. The keys K_1 and K_2 are unknown to the adversary.
2. The adversary encrypts P by all the possible keys K_1 and makes a list L_1 of pairs of K_1 and the ciphertexts. Similarly the adversary decrypts C by all the possible keys K_2 and makes a list L_2 of pairs of K_2 and the plaintexts.
3. The adversary sorts the lists L_1 and L_2 based on the values of the ciphertext and plaintext respectively.
4. The adversary finds the items in L_1 and L_2 where the ciphertext of the item in L_1 is the same as the plaintext of the item in L_2.
5. If a candidate key pair of (K_1, K_2) is found, the validity of the key pair can be verified by using the other pair (P', C') of the plaintext and ciphertext.

3.2 Application to DES-EXE

In order to prevent the meet-in-the-middle attack, an intuition will be to make it hard to compute the middle data. DES-EXE was designed such that the meet-in-the-middle attack is infeasible, but in [11, 6], DES-EXE was shown to be vulnerable to the meet-in-the-middle attack as follows (Fig. 7):

1. The adversary obtains three valid plaintext-ciphertext pairs $(P, C), (P', C'), (P'', C'')$. The keys K_1, K_2, K_3 are unknown to the adversary.
2. The adversary computes $S_1 = E_{K_1}(P) \oplus E_{K_1}(P')$ and $T_1 = E_{K_1}(P') \oplus E_{K_1}(P'')$ for all possible K_1 and make a list L_1 of (K_1, S_1, T_1). Similarly the adversary computes $S_2 = D_{K_3}(C) \oplus D_{K_3}(C')$ and $T_2 = D_{K_3}(C') \oplus D_{K_3}(C'')$ for all possible K_3 and make a list L_2 of (K_3, S_2, T_2).
3. The adversary finds the items (K_1, S_1, T_1) and (K_3, S_2, T_2) in L_1 and L_2 respectively where $S_1 = S_2$ and $T_1 = T_2$. Since the length of lists L_1 and L_2 is 2^n where n is 56, the time complexity to find such a pair of items is $O(2^{n*2})$. The keys K_1 and K_3 found are the correct keys with high probability.
4. The adversary determines the key K_2 by using trial encryption with K_1 and K_3 found in the previous step.

In this attack, we need three plaintext-ciphertext pairs. The dominant necessary memory space is $3 \times 2^{56+1} \times 64$ bits in Step 2. The dominant time complexity is $2^{56} \times 8$ DES encryptions in Step 2.

4 Related-key Attack on DES-EXE

The related-key attack was introduced by Biham [3] and the theoretical related definitions were given by Bellare and Kohno [2]. In a related-key attack, an adversary can obtain the ciphertexts of certain plaintexts under the unknown secret keys K and K', but the relationship between K and K' is known to and can be chosen by the adversary. The related-key attacks may be considered to be rather theoretical because the adversary is given a fairly advantageous situation.

In [6, 9, 11, 12], the related-key attacks on triple DES, DESX, and DES-EXE are shown. DESX was shown to be secure in [10], but the related-key attacks were not considered in [10]. As an example, we show the related-key attack [6] on DES-EXE (Fig. 6).

1. The adversary obtains two plaintext-ciphertext pairs (P, C) and (P'', C'') that are encrypted under the keys $K = (K_1, K_2, K_3)$ where $P \neq P''$. Also the adversary obtains a plaintext-ciphertext pair (P, C') that is encrypted under $K' = (K_1, K_2 \oplus \Delta, K_3)$ where Δ is a difference. The keys K_1, K_2, and K_3 are unknown to the adversary, but Δ is known to the adversary.

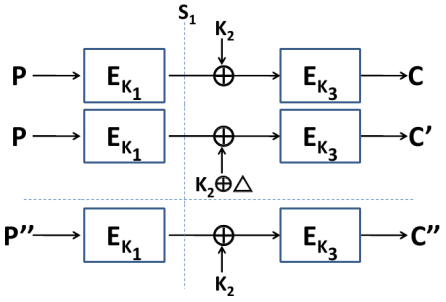

Fig. 6. Related-key Attack on DES-EXE

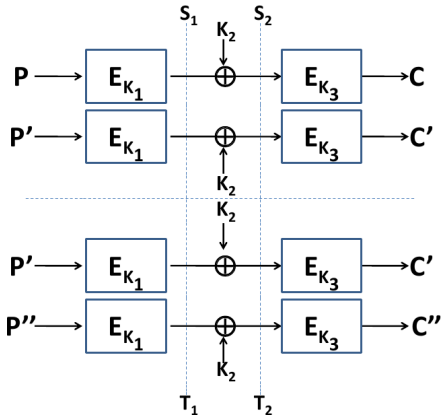

Fig. 7. Meet-in-the-Middle Attack on DES-EXE

2. The adversary guesses K_3 by checking whether $\Delta = D_{K_3}(C) \oplus D_{K_3}(C')$. If a candidate K_3 is found, the adversary computes a pair $(K_2, S_1 = D_{K_3}(C) \oplus K_2)$ for all possible K_2 and make a list L_1 of (K_2, S_1).
3. The adversary guesses K_1 by checking whether $E_{K_1}(P) = S_1$ for some (K_2, S_1) in L_1. If such (K_2, S_1) is found in L_1, (K_1, K_2, K_3) is a candidate. Furthermore the adversary checks whether $C'' = E_{K_3}(E_{K_1}(P'') \oplus K_2)$. If so, (K_1, K_2, K_3) is a correct key with high probability.

In this attack, we need two plaintext-ciphertext pairs and one related-key adaptive chosen plaintext. The dominant necessary memory space is $3 \times 2^{56} \times 64$ bits in Step 2. The time complexity is $2^{56+1} + 2^{56}$ DES encryptions in Step 2 and 2^{56} DES encryptions in Step 3.

5 Proposed Scheme

5.1 Main Construction: DES-XEEX

To avoid the related-key attack and meet-in-the-middle attack, we propose the following encryption that we call DES-XEEX (Fig. 8). DES-XEEX was first proposed by the second author of this paper [15].

$$C = E_{K_2}(E_{K_1}(P \oplus K_2)) \oplus K_1$$

5.2 DES-XEEX Variant

The construction introduced in Section 5.1 assumes that the size of a key is the same as that of a plaintext and ciphertext, but this may not be the case. Therefore, we also propose the following variant (Fig. 9).

$$C = E_{K_2}(E_{K_1}(P \oplus E_{K_2}(0))) \oplus E_{K_1}(0)$$

Note that $E_{K_1}(0)$ and $E_{K_2}(0)$ can be precomputed and stored with K_1 and K_2, so the encryption and decryption procedures can be done with two DES operations.

6 Discussion on Security of Proposed Scheme

6.1 Heuristic Discussion

We examine whether the attacks on DES-EXE can work on the proposed scheme.

First we consider the related-key attack. The adversary obtains the plaintext-ciphertext pair (P, C) encrypted by (K_1, K_2). Also the adversary obtains the plaintext-ciphertext pair encrypted under $(K_1 \oplus \Delta, K_2)$. As in the related-key attack on DES-EXE, we consider the middle data at the point S (Fig. 10).

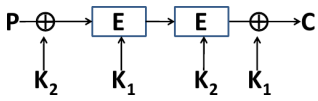

Fig. 8. DES-XEEX

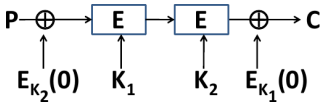

Fig. 9. DES-XEEX Variant

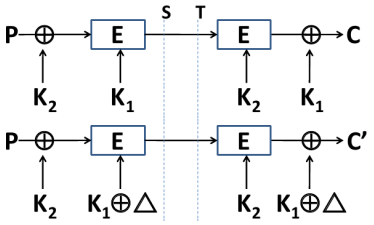

Fig. 10. Related-key Attack on DES-XEEX

At the point S in Fig. 10, we can obtain $E_{K_1}(P \oplus K_2)$ and $E_{K_1 \oplus \Delta}(P \oplus K_2)$, but it seems difficult to obtain some useful data from these data. Similarly at the point T in Fig. 10, we can obtain $D_{K_2}(C \oplus K_1)$ and $D_{K_2}(C' \oplus K_1 \oplus \Delta)$, but it seems difficult to obtain some useful data from these data.

As for the meet-in-the-middle attack, to obtain the middle data, we need to specify both K_1 and K_2 and $O(2^{n*2})$ memory space is necessary where n is the bit length of a single DES key.

Therefore, the related-key and meet-in-the-middle attacks on the DES-XEEX construction seem unsuccessful compared with the case of DES-EXE.

6.2 Insecurity under Related-key Attack and Theoretical Impossibility Result

Security under related key attack is a recent requirement for block ciphers. In [2], the security game is defined as follows:

DEFINITION. Let $E : \mathcal{K} \times \mathcal{D} \to \mathcal{D}$ be the block cipher. Also allow the adversary to make related-key oracle queries consisting of a related-key-deriving (RKD) function $\phi : \mathcal{K} \to \mathcal{K}$ and a point $x \in \mathcal{D}$. In world 1, a key K is chosen at random from $\mathcal{K}$ and query (ϕ, x) is answered by $E(\phi(K), x)$. In world 0, a key K is again chosen at random from $\mathcal{K}$, and a permutation $G(L, \cdot) : \mathcal{D} \to \mathcal{D}$ is also chosen at random for each key $L \in \mathcal{K}$, and the query (ϕ, x) is answered by $G(\phi(K), x)$. The advantage of the adversary is the difference between the probabilities that it returns 1 in the two worlds. For any set Φ of functions mapping $\mathcal{K}$ to $\mathcal{K}$, E is said to be secure against Φ-restricted related-key attacks if the advantage of the adversary is negligible where the adversary can use RKD functions in its oracle queries only from Φ.

Then the attack on the proposed scheme works as follows:

1. Query $E_{K_1,K_2}(M_1)$ to get C_1 where K_1 and K_2 are chosen by the challenger and unknown to the adversary. This means that the query (ϕ, x) is set to be such that ϕ is an identity function and x is M_1.
2. Query $E_{K_1,K_2}(M_2)$ to get C_2.
3. Query $E_{C_1,C_2}(M)$ to get C. This means that the query (ϕ, x) is set to be such that ϕ is a constant function that returns (C_1, C_2) and x is M.
4. The adversary returns 1 if $E_{C_2}(E_{C_1}(M \oplus C_2)) \oplus C_1 = C$, and otherwise returns 0.

Obviously if E is a DES-XEEX construction, the adversary returns 1 with probability 1. On the other hand, if E used by the oracle is a random permutation G, the corresponding probability is negligible. However, from the impossibility result (Proposition 4.1) in [2], it is easy to check that if ϕ can be a constant function and can use the cipher itself as the related key function, there exists an adversary against any pre or postprocessing scheme that can win the security game successfully in this definition and it is the theoretical limitation.

6.3 Related-key Attack on DES and AES

As mentioned in [2], it is well-known that DES itself is also insecure against related-key attacks in the definition of [2] because DES has the complementation property $\overline{\mathrm{DES}_K(P)} = \mathrm{DES}_{\overline{K}}(\overline{P})$ for all keys K and plaintexts P where $\overline{X}$ means the bitwise complement of X. This complementation property of DES can also lead to the insecurity of our DES-XEEX against related-key attacks in a theoretical sense because we have, for instance, $\mathrm{DES}_{\overline{K_2}}(\mathrm{DES}_{\overline{K_1}}(P \oplus \overline{K_2})) \oplus \overline{K_1} =$

$\mathrm{DES}_{K_2}(\mathrm{DES}_{K_1}(P \oplus K_2)) \oplus K_1$ though this property can be invalidated by using our variant because, for instance, we have $\mathrm{DES}_{\overline{K_2}}(\mathrm{DES}_{\overline{K_1}}(P \oplus \mathrm{DES}_{\overline{K_2}}(0))) \oplus \mathrm{DES}_{\overline{K_1}}(0) = \mathrm{DES}_{K_2}(\mathrm{DES}_{K_1}(P \oplus \mathrm{DES}_{K_2}(\overline{0}))) \oplus \mathrm{DES}_{K_1}(\overline{0})$.

Also in [5, 4], Biryukov et al. proposed the related-key attacks on AES though AES is still considered to be secure in practice. Because it is possible to consturct a block cipher such as Camellia [1] which is secure against the related-key attacks, it will be important to construct a multiple encryption secure against related-key attacks in a practical sense.

7 Concluding Remarks

We showed a construction that we call DES-XEEX and its variant to make double DES secure against the related-key and meet-in-the-middle attacks in practice and the construction is more secure compared with DES-EXE. Our construction is generic and applicable to any block cipher such as AES to have a longer key effectively. Considering the attacks on DES-EXE, we gave the heuristic discussion about the security of the proposed scheme, but further investigation of the security of DES-XEEX will be needed in the practical settings.

Acknowledgements

This research is (partially) supported by JAPAN SCIECE AND TECHNOLOGY AGENCY (JST), Strategic Japanese-Indian Cooperative Programme on Multidisciplinary Research Field, which combines Information and Communications Technology with Other Fields, entitled "Analysis of Cryptographic Algorithms and Evaluation on Enhancing Network Security Based on Mathematical Science."

References

1. K. Aoki, T. Ichikawa, M. Kanda, M. Matsui, S. Moriai, J. Nakajima, and T. Tokita, "Camellia: a 128-bit block cipher suitable for multiple platforms - design and analysis," Selected Areas in Cryptography, LNCS 2012, pp.39–56, Springer-Verlag, 2000.
2. M. Bellare and T. Kohno, "A theoretical treatment of related-key attacks: RKA-PRPs, RKA-PRFs, and Applications," Proc. Eurocrypt 2003, LNCS 2656, pp.491–506, Springer-Verlag, 2003.
3. E. Biham, "New types of cryptanalytic attacks using related keys," Proc. Eurocrypt 1993, LNCS 765, pp.398–409, Springer-Verlag, 1993.
4. A. Biryukov, D. Khovratovich, I. Nikolic, "Distinguisher and related-key attack on the full AES-256," Proc. Crypto 2009, LNCS 5677, pp.231–249, Springer-Verlag, 2009.
5. A. Biryukov, D. Khovratovich, "Related-key cryptanalysis of the full AES-192 and AES-256," Proc. Asiacrypt 2009, LNCS 5912, pp.1–18, Springer-Verlag, 2009.
6. J. Choi, J. Kim, J. Sung, S. Lee, and J. Lim, "Related-key and meet-in-the-middle attacks on triple-DES and DES-EXE," Proc. ICCSA 2005, LNCS 3481, pp.567–576, Springer-Verlag, 2005.
7. W. Diffie and M. Hellman, "Exhaustive cryptanalysis of the NBS data encryption standard," IEEE Computer, Vol. 10(6), pp.74–84, 1977.
8. B.S. Kaliski and M.J.B. Robshaw, "Multiple encryption: weighing security and performance," Dr. Dobb's Journal, #243, pp.123–127, January 1996.
9. J. Kelsey, B. Schneier, and D. Wagner, "Related-key cryptanalysis of 3-WAY, Biham-DES, CAST, DES-X, NewDES, RC2, and TEA," Proc. ICICS 1997, LNCS 1334, pp.233–246, Springer-Verlag, 1997.
10. J. Kilian and P. Rogaway, "How to protect DES against exhaustive key search (an analysis of DESX)," J. Cryptology, Vol. 14(1), pp.17–35, Springer-Verlag, 2001.
11. R. C.-W. Phan, "Related-key attacks on triple-DES and DESX variants," Proc. Cryptographers' Track at the RSA Conference (CT-RSA), LNCS 2964, pp.15–24, Springer-Verlag, 2004.
12. R. C.-W. Phan and A. Shamir, "Improved related-key attacks on DESX and DESX+," Cryptologia, 32:1, pp.13–22, 2008.
13. RSA Data Security, Inc., Product documentation, "Mailsafe Note #3."
14. W. Tuchman, "Hellman presents no shortcut solutions to DES," IEEE Spectrum, Vol. 16(7), pp.40–41, 1979.
15. S. Yoshinaga, "How to make double DES robust against meet-in-the-middle attack," Bachelor Thesis (in Japanese) at Kyushu University, 2010.