A proposed mode for triple-DES encryption

by D. Coppersmith
D. B. Johnson
S. M. Matyas

We propose a new mode of multiple encryption—triple-DES external feedback cipher block chaining with output feedback masking. The aim is to provide increased protection against certain attacks (dictionary attacks and matching ciphertext attacks) which exploit the short message-block size of DES. The new mode obtains this protection through the introduction of secret masking values that are exclusive-ORed with the intermediate outputs of each triple-DES encryption operation. The secret mask value is derived from a fourth encryption operation per message block, in addition to the three used in previous modes. The new mode is part of a suite of encryption modes proposed in the ANSI X9.F.1 triple-DES draft standard (X9.52).

Introduction

When the Data Encryption Standard (DES) was adopted as a federal standard in 1977 [1], its expected life was ten years. Nineteen years later, in 1996, the DES is still "going strong," and it continues to have a strong base of support within the financial community. The DES is a U.S. national standard and a *de facto* international standard. In 1993, the DES was recertified by the National Institute of Standards and Technology (NIST) for another five years, although rumors persist that NIST may not recertify the DES again. During this time, no efficient attack against the DES has been reported, although several attacks based on exhaustive key search have been published [2–5], as well as attacks (differential cryptanalysis [6] and linear cryptanalysis [7]) requiring massive amounts of known plaintext and corresponding ciphertext. (Linear cryptanalysis requires knowledge of 2^{43} plaintext/ciphertext pairs, fewer than the 2^{56} trial encipherment required for key exhaustion, but more difficult to arrange, because these 2^{43} blocks must be enciphered on the target machine in possession of the secret key. Differential cryptanalysis is less efficient. Some discussion is found in [8].)

Exhaustive key search remains the fastest known attack against the DES. But improvements in technology, leading to the potential for faster key search machines, now pose a greater threat to the use of single-key DES. The use of triple encryption with multiple keys is generally accepted as the best and most practical method for increasing the strength of the DES against key search attacks. It also guards against linear and differential cryptanalysis. Much effort has gone into attempted cryptanalysis of multiple DES[1–4] [9–12]. Because the DES is still a fundamentally sound base on which to build, the American National Standards Institute (ANSI) committee X9.F.1 is working

[1] Don Coppersmith, "A Chosen-Ciphertext Attack on Triple DES CBC," revised February 8, 1995, provided to ANSI X9.F.1.
[2] Don Coppersmith, "A Chosen-Plaintext Attack on 2-Key Inner Triple DES CBC/EDE, Preliminary Version," April 19, 1995, provided to ANSI X9.F.1.
[3] Burt S. Kaliski, Jr., "On the Security and Performance of Several Triple-DES Modes (Extended Abstract)," private communication provided to ANSI X9.F.1 (January 12, 1994—second draft).
[4] James L. Massey, "Analysis of Two-Key Triple DES Encryption," April 2, 1994, provided to ANSI X9.F.1.

0018-8646/96/$5.00 © 1996 IBM

Table 1 Abbreviations and definitions.

ANSI	: American National Standards Institute
DES	: Data Encryption Standard (encryption algorithm)
CBC	: Cipher Block Chaining (mode of encryption)
CBCM	: Triple-DES External Feedback Cipher Block Chaining with Output Feedback Masking; a new encryption mode proposed here.
Double DES	: Two DES encryption/decryption operations (e.g., E-D)
$d(K, Y)$	: Decryption of input Y with key K.
ECB	: Electronic Code Book (encryption without chaining)
E-D-E	: Encrypt-Decrypt-Encrypt (with DES)
TCBC	: Triple-DES External Feedback Cipher Block Chaining (mode of encryption)
$e(K, X)$	: Encryption of input X with key K.
IV	: Initializing Vector (non-secret value XORed with *input*)
key	: A secret value that initializes the DES algorithm.
NIST	: National Institute of Standards and Technology
OFB	: Output Feedback (mode of encryption)
Triple DES	: Three DES encryption/decryption operations (e.g., E-D-E)
XOR, $\oplus$	: Exclusive-OR operation
X9.F.1	: An ANSI financial security standards committee

to standardize a suite of modes of triple-DES encryption (X9.52). (**Table 1** contains definitions for acronyms used in this paper.)

Among the issues which must be addressed when settling on such triple-DES modes, one important issue concerns block size. Ordinary DES operates on 64-bit message blocks; there are 2^{64} different 64-bit blocks; but if just 2^{32} or 4000 million randomly chosen blocks have been encrypted, it is likely that the same block has been encrypted twice. (This is due to the "birthday phenomenon": If $\sqrt{N}$ samples are produced randomly from among N possibilities, it is likely that two of them are the same.) This gives information to the attacker, since identical input blocks yield identical output; this is the "matching ciphertext attack." If we just replace the single-DES mode with the triple-DES mode, the block size remains at 64 bits, and the matching ciphertext attack remains a threat when a few billion blocks are processed. Another concern is a dictionary attack, where the attacker accumulates a dictionary of matching plaintext–ciphertext. If large enough, say a few billion blocks, the dictionary could permit some intercepted ciphertext to be decoded using a simple table lookup.

This paper describes a method for increasing the strength of the triple-DES mode against the threat of dictionary and matching ciphertext attacks, without having to change the 64-bit block size of the DES algorithm. The method uses secret masking values calculated via a parallel-running Output Feedback (OFB) mode, to destroy any relationship between matching ciphertext blocks. OFB is one of several modes of encryption that can be performed with the DES (see Meyer and Matyas [13] and the definitions by standards bodies [14–16]). The masking values are exclusive-ORed (XORed) with the intermediate outputs of each triple-DES operation. The cost for this increased security is one additional OFB encryption, along with two additional exclusive-OR operations which are interleaved between the triple-DES encryptions and decryptions (in the order Encrypt-Decrypt-Encrypt, or E-D-E).

The present method is related to two earlier, simpler methods. Blaze describes a mode of file encryption in which a block of data is first exclusive-ORed with a secret mask value before being encrypted [17]. Concerned that multiple encryption techniques are computationally rather expensive, especially when implemented in software, Blaze sought a way to allow access to any point within an encrypted file but still discourage structural analysis and provide greater security than regular Electronic Code Book (ECB) mode. His method consists of "crunching" a long pass-phrase into two 56-bit DES keys. The first key is used to precompute a long (half-megabyte) pseudorandom bit mask using the DES OFB mode. When a file block is to be written, it is first exclusive-ORed with the part of the mask corresponding to its byte offset in the file. The result is then encrypted with the second key using ECB mode.

Jones describes a slightly different mode in which a secret masking value, calculated via a parallel-running OFB mode, is exclusive-ORed with the intermediate output of a double-DES (E-D) operation.[5]

To contrast the methods: Blaze performs masking prior to single-DES encryption (E); Jones performs masking inside a double-DES operation (E-D); the authors (Coppersmith, Johnson, and Matyas) perform masking inside a triple-DES operation (E-D-E). However, under certain conditions, in both the Blaze and Jones methods, the effect of the exclusive-OR masking operation can be canceled out, thereby untangling the keys and exposing the method to attacks, which are detailed below. Knowledge of this method of attack suggested the present, proposed improved triple-DES mode.

Triple-DES Cipher Block Chaining with External Feedback

We present, for comparison, another triple-DES mode that is part of a suite of encryption modes in the ANSI X9.F.1 triple-DES draft standard (X9.52). One of these modes is termed Triple-DES External Feedback Cipher Block Chaining (TCBC), i.e., cipher block chaining (CBC) with external feedback. The TCBC mode uses a nonsecret

[5] Thomas C. Jones, "Cipher-Chain-Cipher Modes of Operation of Block Ciphers," draft, March 17, 1995, provided to ANSI; also, informal presentation at Crypto 95 (August 1995); not in the proceedings.

64-bit initializing vector (IV) and three secret 64-bit keys (K_1, K_2, and K_3) to encrypt a plaintext ($X_1, X_2, \cdots, X_n$) consisting of n 64-bit blocks to produce a ciphertext ($Y_1, Y_2, \cdots, Y_n$) consisting of n 64-bit blocks (**Figure 1**). At the first iteration of encryption, the IV is exclusive-ORed with the first block of plaintext, X_1. At each subsequent iteration of encryption, each block of plaintext (X_i) is exclusive-ORed with the previous block of ciphertext (Y_{i-1}). The result is then encrypted with three-key triple DES (E-D-E)—encrypted under K_1, decrypted under K_2, and encrypted under K_3—to produce the ciphertext Y_i.

Triple-DES external feedback CBC has the following advantages:

1. The input and output block size is 64 bits, the same as normal DES.
2. It is backward compatible with respect to single-key DES encryption.
   - Using one key value for all three key inputs results in the same output as a single-DES encryption.
3. It has limited error propagation.
   - If one block of ciphertext is corrupted, only two blocks of recovered plaintext will be corrupted. This is known as the self-healing or self-synchronizing property of CBC encryption.
4. It is resistant to cryptanalytic exhaustive key search attacks.
   - Using two keys, if n is the number of known plaintext blocks, the best known work factor is $2^{120}/n$ [12].
   - Using three keys, the best known work factor is 2^{112} with some known plaintext; having many known plaintext blocks does not appear to reduce this work factor.

However, any multiple-key CBC mode of operation with external feedback using a 64-bit block size has the following disadvantages:

1. It cannot be simulated easily using existing DES modes of operation.
   - This means that existing systems may not be able to simulate TCBC mode without a significant functional upgrade.
2. The mode, as defined, is not straightforward to pipeline for performance.
   - Each block must be DES encrypted, decrypted, and encrypted before the next block is processed.
   - An alternate mode could be defined by interleaving the data [that is, chaining (exclusive-ORing) the ciphertext Y_{i-3} to the plaintext X_i], but this complicates the protocol.

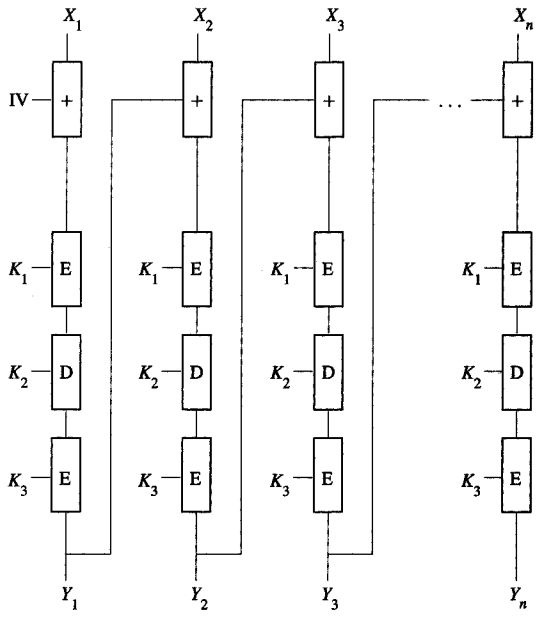

Figure 1

Triple-DES external feedback cipher block chaining (TCBC).

3. It has the complementarity property, which an attacker can exploit in some situations.
   - The complementarity property may be expressed as $C[e(K, x)] = e[C(K), C(x)]$, where e is DES encryption, K is any key, x is any 64-bit value, and $C(x)$ indicates bitwise complementation (bit inversion). The complementarity property allows testing for two keys for the cost of one encryption if 1) a ciphertext block happens to be the complement of another ciphertext block, or 2) a plaintext block happens to be the complement of another plaintext block. This property is therefore a possible aid to key exhaustion by reducing the work factor by about half in certain cases.
4. It has a potential text dictionary concern.
   - Having known plaintext/ciphertext pairs allows entries in a dictionary to be built. The larger the dictionary, the better the chance to find a match in the dictionary for any specific ciphertext; and the larger the amount of ciphertext, the better the chance for a matching entry to be found in a dictionary of a given size. Consider the extremes, when using TCBC mode with a 64-bit block size: If one has a dictionary of 2^{64} entries, all ciphertext is exposed; and if one has a dictionary with a single entry, it takes about 2^{63} ciphertext blocks to expect one of the ciphertext

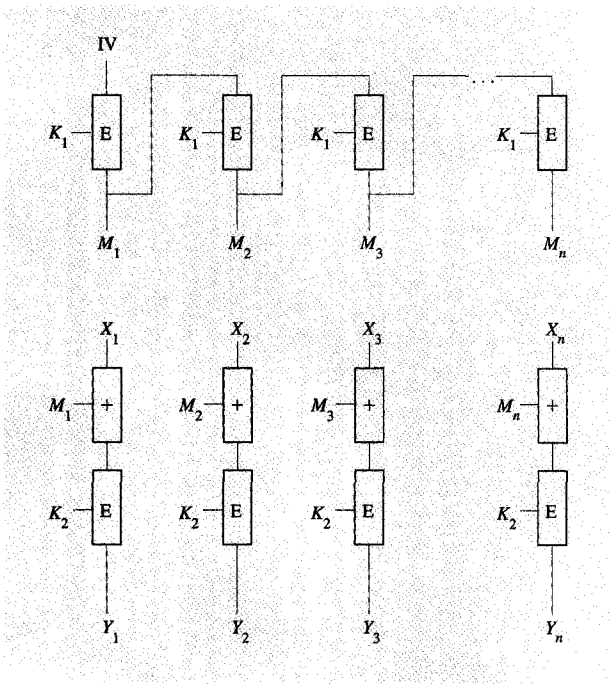

Figure 2

The Blaze method: Single-DES ECB with OFB masking mode.

Table 2 Probability of exposing a single block.

Message blocks	*Text dictionary*	*Matching ciphertext*
2^{20}	0.000000015	0.000000030
2^{25}	0.0000153	0.0000305
2^{30}	0.0155	0.0308
2^{31}	0.0606	0.118
2^{32}	0.221	0.393

blocks to be known, revealing a secret.[6] A more realistic attack occurs when about half of the encrypted text is secret and half is known. In this case, when about 2^{32} blocks of text have been encrypted, secret information should be expected to begin to leak, because of the birthday phenomenon (as mentioned above, when $2^{32} = \sqrt{2^{64}}$ random message blocks have been produced, we expect two of them to be equal); this is known as the crossover point, since exposures are expected at this point unless additional side conditions are assumed.[7]

5. It has a potential matching ciphertext concern.
   - Two ciphertext blocks can be expected to match by chance after about 2^{32} blocks have been encrypted; call them Y_i and Y_j, and call the corresponding secret plaintext X_i and X_j. If a match happens, we know that $Y_{i-1} \oplus X_i = Y_{j-1} \oplus X_j$ because the ciphertexts matched. Therefore, we know that $Y_{i-1} \oplus Y_{j-1} = X_i \oplus X_j$. If the plaintext has significant redundancy in it (for example, character data), there is a good chance that the value $X_i \oplus X_j$ will leak information. Therefore, after about 2^{32} encryptions, one should expect secrecy to begin to be lost in the general case without additional assumptions.

Remark The "text dictionary" and "matching ciphertext" differ in that the former depends on one unknown message block matching a known message block, while the latter depends on two unknown message blocks agreeing. In each case the probability of success varies as the square of the number of message blocks which have been encrypted. We show in **Table 2** the number of random message blocks enciphered and the probability, for each of the two attacks, that a single match has occurred, and so a *single* message block has been exposed. Limiting exposure of a single message block is a very conservative criterion.

Because of these two attacks, it is a good idea to limit the total amount of text encrypted under TCBC mode using a 64-bit block size to something less than 2^{32} blocks, which is 2^{35} bytes or 32 gigabytes, with the exact limits depending on the risk of leakage that one is willing to take, in view of the table. For many applications, such a limit will not raise a concern; however, with the increasing network speeds and massive databases found today and the likelihood of even faster speeds and larger databases in the future, these limits could be exceeded. For this reason, the new mode of triple-DES encryption was included in the suite of encryption modes in the ANSI X9.F.1 triple-DES draft standard (X9.52).

Attacks against earlier schemes (Blaze and Jones)

We examine here the method of Blaze [17], and an attack against it. The construction, and the attack, provide motivation for our own construction.

The Blaze method (**Figure 2**) has two secret DES keys, K_1 and K_2, which are produced from a secret "pass-phrase." Using the first key, K_1, and a standard "initializing vector" IV, one produces a long (half-megabyte) pseudorandom bit mask $M_1, M_2, \cdots, M_n$, using the DES OFB mode. The 64-bit mask value M_i is obtained

[6] However, note that chaining is much better than no chaining, as otherwise a dictionary for information that has redundancy (text, code, etc.) will be much smaller and therefore much easier to build. In effect, doing chaining with a pseudorandom value ensures that a complete dictionary must correspond to the block size, which is the best that can be achieved.

[7] If the same initialization vector is used for multiple encrypted messages, the first block of the message may have structure that can significantly reduce the size of the dictionary needed to reveal the first block. If this is a concern, a new random initialization vector should be used for each message or a new random confounder appended to the front of each message.

from its predecessor M_{i-1} by encryption under the key K_1. Each mask M_i is then exclusive-ORed, bitwise, to the corresponding plaintext X_i, and the output is encrypted under the second key, K_2, using the DES ECB mode (Electronic Code Book, or straight encryption without chaining), to yield the ciphertext Y_i:

$$M_0 = IV,$$

$$M_i = e(K_1, M_{i-1}),$$

$$Y_i = e(K_2, M_i \oplus X_i).$$

See Figure 2.

One advantage of this method is that it thwarts the dictionary attack. If an opponent sees two identical blocks of ciphertext, $Y_i = Y_j$, he can deduce that the corresponding inputs to the K_2 DES box are equal: $d(K_2, Y_i) = d(K_2, Y_j)$, so that $X_i \oplus M_i = X_j \oplus M_j$. But because the sequence M_i is unknown, the opponent gains no useful information from this.

The scheme becomes vulnerable if the opponent knows plaintext and corresponding ciphertext from two different messages, both encrypted using the same initializing vector IV. Denote the two plaintext sequences as X_i and X'_i, and the corresponding ciphertext as Y_i and Y'_i. The opponent compares plaintext and ciphertext from a single position i. One has that $X_i \oplus M_i = d(K_2, Y_i)$ and $X'_i \oplus M_i = d(K_2, Y'_i)$. The two mask values M_i are the same, because M_i depends only on IV, K_1, and the index i, not on X_i or Y_i. Combining the two equations, one finds $X_i \oplus X'_i = d(K_2, Y_i) \oplus d(K_2, Y'_i)$.

This suggests the following attack, which is only three times as expensive as single-key exhaustion (that is, 3×2^{56} encryptions, and no memory requirements). For each of 2^{56} trial values k_2 for the unknown key K_2, he evaluates the quantity $d(k_2, Y_i) \oplus d(k_2, Y'_i)$, and compares against the known value $X_i \oplus X'_i$. For the correct value $k_2 = K_2$, the two quantities will be equal. For an incorrect value $k_2 \neq K_2$ the two might accidentally agree, but with a slim probability 2^{-64}, so that the expected number of "false alarms" is only $2^{56} \times 2^{-64} = 1/256$. Thus, he is highly likely to find K_2 unambiguously. (In the case of ambiguity, he can try another index $j \neq i$.)

Having found K_2, he can find the mask values M_i from $M_i = X_i \oplus d(K_2, Y_i)$, and discover the key K_1 by single-key exhaustion, by seeing which trial value k_1 for K_1 satisfies the requirement $M_i = e(k_1, M_{i-1})$.

The attacker was able to separate the effects of the two keys K_1 and K_2, so that he could attack each key separately by (essentially) single-key exhaustion. A strong scheme must entangle the effects of the several keys more effectively.

[This attack depended on the IV being held constant across different encryptions, so that the values M_i would be repeated. If, instead, the IV is changing but is publicly known, we can use an attack suggested by Michael Steiner:[8] Select a ciphertext satisfying $Y_1 = Y_2$, so that $d(K_2, Y_1) = M_1 \oplus X_1 = d(K_2, Y_2) = M_2 \oplus X_2$, and exhaustively search for the key K_1 which (together with the known IV) would produce values M_i satisfying that equation. For an IV which is changing and kept secret, more expensive attacks are available; at any rate, extreme care must be devoted to the protection of the IV.]

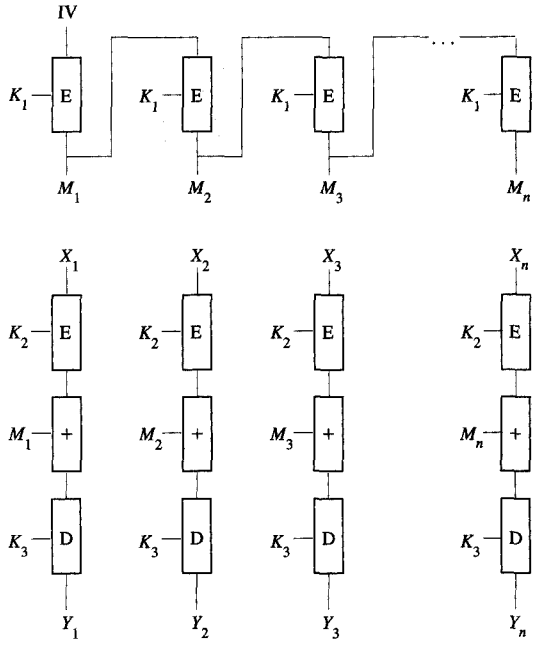

Figure 3

Double-DES with OFB masking (Jones).

Jones[5] mentions a method (ECB|OFB|ECB) taking the idea one step further. As with the Blaze method, the Jones method (**Figure 3**) produces a sequence of masking values M_i using a key K_1 and initializing vector IV with output feedback mode. It then applies M_i, by exclusive-OR, in the middle of a double-DES operation: The plaintext X_i is enciphered under a DES key K_2, the result is exclusive-ORed with M_i, and this result is deciphered under a DES key K_3 to produce the ciphertext Y_i:

$$M_0 = IV,$$

$$M_i = e(K_1, M_{i-1}),$$

$$Y_i = d[K_3, M_i \oplus e(K_2, X_i)].$$

In the attack against this scheme, the opponent again uses known plaintext X_i and X'_i and corresponding ciphertext

[8] Michael Steiner, IBM Zurich Research Laboratory, 1995; private communication.

Y_i and Y'_i from two messages enciphered with the same keys and the same initializing vector IV. The effect of the masking value M_i can be canceled, reducing the scheme to a double-DES scheme, which can then be attacked by a meet-in-the-middle operation, as detailed below. The computational cost of this attack is about the same as before—4×2^{56} DES operations—but it apparently requires massive storage as well, about 2×2^{56} 64-bit blocks, or 2^{60} bytes. Time and space requirements may be traded off, so that an attack requiring 4×2^{66} DES operations and 2^{50} bytes could be mounted instead, if that was deemed more feasible. Namely, we would repeat the attack 1024 times and use 2×2^{46} 64-bit blocks each time.

The opponent uses the equations $e(K_3, Y_i) = M_i \oplus e(K_2, X_i)$ and $e(K_3, Y'_i) = M_i \oplus e(K_2, X'_i)$, along with the fact that M_i is the same for both encryptions. Combining the two equations, he obtains $e(K_3, Y_i) \oplus e(K_3, Y'_i) = e(K_2, X_i) \oplus e(K_2, X'_i)$. The unknown values M_i have been canceled out.

In contrast to the Blaze method, the attack here is made more difficult in that the opponent does not know the value of either side of the equation. The opponent can cycle through all 2^{56} trial values k_3 for K_3 and evaluate the left-hand side for each, creating a table of 2^{56} entries, each entry containing the trial key k_3 and the 64-bit quantity $e(k_3, Y_i) \oplus e(k_3, Y'_i)$. He stores this table in a convenient form, sorted and indexed by the second quantity $e(k_3, Y_i) \oplus e(k_3, Y'_i)$. For each trial value k_2 for K_2 he evaluates the right-hand side: $e(k_2, X_i) \oplus e(k_2, X'_i)$, and searches the table for a match: $e(k_3, Y_i) \oplus e(k_3, Y'_i) = e(k_2, X_i) \oplus e(k_2, X'_i)$.

The proper values (K_2, K_3) lead to a match. Each of the $2^{56} \times 2^{56} = 2^{112}$ potential pairs (k_2, k_3) has a probability 2^{-64} of creating an accidental match, so that one expects 2^{48} "false alarms." For each match, the opponent tries the pair (k_2, k_3) at a different index location j, testing whether $e(k_3, Y_j) \oplus e(k_3, Y'_j) = e(k_2, X_j) \oplus e(k_2, X'_j)$. One expects that only the correct pair (K_2, K_3) satisfies both matches.

Having found the keys K_2 and K_3, he can find K_1 by exhaustion, as before.

A more efficient implementation of meet-in-the-middle attacks is given by van Oorschot and Wiener [18].

Apparently the problem with both of these approaches (Blaze and Jones) is that the effects of the several keys are not sufficiently tangled with one another, and it is too easy to separate them out and attack the keys individually. In the present design we attempt to overcome this weakness.

Internal feedback

Having seen the weaknesses of external feedback, one might be tempted to design modes with internal feedback instead. For example, three layers of CBC (cipher block chaining) could be sandwiched, taking plaintext X_i to ciphertext Y_i via intermediate values A_i and B_i:

$$A_i = e(K_1, A_{i-1} \oplus X_i) \qquad A_0 = IV_1,$$

$$B_i = e(K_2, B_{i-1} \oplus A_i) \qquad B_0 = IV_2,$$

$$Y_i = e(K_3, Y_{i-1} \oplus B_i) \qquad Y_0 = IV_3.$$

But Biham [9, 10] gives convincing demonstrations that these modes with internal feedback are weak. The internal feedback gives the attacker the ability to probe the values of internal registers in the system. For this particular example, Biham gives an attack requiring 2^{35} blocks of chosen ciphertext. The opponent selects ciphertext with many repeated blocks: $Y_{4i-3} = Y_{4i-2} = Y_{4i-1} = Y_{4i} = c_i$ for 2^{33} different values c_i, so that $B_{4i-2} = B_{4i-1} = B_{4i}$ and $A_{4i-1} = A_{4i}$. He causes this text to be decrypted. By a birthday attack, he hopes that a coincidence has happened among the intermediate texts: $B_{4i} = B_{4j}$ but $Y_{4i} = Y_{4j}$. This will cause $X_{4i} = X_{4j}$, which he can detect. He tries all possible values k_3 for the key K_3 to find the correct one which causes $Y_{4i} \oplus d(k_3, Y_{4i}) = Y_{4j} \oplus d(k_3, Y_{4i})$, or, equivalently, $B_{4i} = B_{4j}$. Keys K_2 and K_1 are then discovered in a similar manner. The internal feedback has allowed him to probe the internal values B_{4i}. Biham shows similar techniques in [9, 10] which break other internal feedback schemes as well as this example, and which leave little hope for the security of similar schemes.

Objectives of the new mode

The design objectives for the new feedback mode are as follows:

1. The input and output block size is 64 bits, an ANSI X9.F.1 requirement.
2. Understandable design.
3. Stronger than TCBC mode with regard to text dictionary attacks.
4. Stronger than TCBC mode with regard to the matching ciphertext attack.
5. Mask patterns in the input plaintext.

Note that it was not a design objective to be backward compatible with an existing mode of operation.

Triple-DES external feedback CBC with OFB masking (CBCM) mode

The newly proposed mode uses triple-DES with external feedback CBC and OFB masking (CBCM). It is a triple-DES mode that uses a secret masking value (**Figure 4**). The secret masking value is calculated via a parallel-running OFB mode and is exclusive-ORed at each iteration with the intermediate outputs of the CBC mode. The CBCM mode is a unique design with the

characteristic that it cannot be simulated using a combination of other modes.

Strength of Triple-DES external feedback CBC with OFB masking (CBCM) mode

With regard to a matching ciphertext or dictionary attack, the adversary is forced to launch a separate attack for each IV1 and each separate iteration of CBC encryption. We assume that an adversary cannot use the keys (K_1, K_2, and K_3) in other cryptographic operations that might allow the keys to be attacked with less work than in the CBCM mode. For example, we assume that an adversary has no means to cause K_3 to be used in an ordinary OFB mode such that its outputs would be exposed, since this would allow the M_1, M_2, etc. values to be exposed for a particular IV1 and K_3.

With regard to a cryptanalytic key discovery attack, CBCM mode appears to be about as strong as TCBC mode. Of the attacks which the authors investigated, the most promising (described below) appears to require 2^{90} DES operations, as well as 2^{34} blocks of chosen ciphertext and corresponding plaintext. (The chosen ciphertext requirement is unusual, and arises because of the outer chaining.) Without large amounts of known plaintext and ciphertext (2^{34} blocks), the known attacks require 2^{112} or more operations.

In the most promising attack, the opponent creates two long ciphertexts. The first consists of 2^{33} repetitions of some constant 64-bit block Y, so $Y_i = Y$. The second contains 2^{33} repetitions of some other constant 64-bit block Y', so $Y'_i = Y'$. He then requests the decryption of both, using the same unknown keys K_1, K_2, K_3, and the same initial vector IV1 for both ciphertexts.

Because the masking values M_i depend only on K_3, IV1, and the index i, and not on the plaintext or ciphertext, one sees that same values of M_i will be used for both encryptions.

The opponent is looking for a pair of indices (i, j) where $M_i \oplus d(K_1, Y_i) = M_j \oplus d(K_1, Y'_j)$. Call this a "coincidence." These two quantities represent the outputs of the box $d(K_2, *)$ at different instances; because they agree, the corresponding inputs agree, namely $M_i \oplus e(K_1, X_i \oplus Y_{i-1}) = M_j \oplus e(K_1, X'_j \oplus Y'_{j-1})$. Combining these two equations, he would have that $d(K_1, Y_i) \oplus e(K_1, X_i \oplus Y_{i-1}) = d(K_1, Y'_j) \oplus e(K_1, X'_j \oplus Y'_{j-1})$.

Because of the choice of ciphertext ($Y_i = Y_j$ and $Y'_i = Y'_j$), one coincidence leads to another, namely $M_j \oplus d(K_1, Y_j) = M_i \oplus d(K_1, Y'_i)$.

The attacker selects a trial value k_1 for the key K_1. He then evaluates the quantities $d(k_1, Y_i) \oplus e(k_1, X_i \oplus Y_{i-1})$ and $d(k_1, Y'_j) \oplus e(k_1, X'_j \oplus Y'_{j-1})$ for indices i, j up to 2^{33}, and compares the lists, looking for matches: $d(k_1, Y_i) \oplus e(k_1, X_i \oplus Y_{i-1}) = d(k_1, Y'_j) \oplus e(k_1, X'_j \oplus Y'_{j-1})$. In case of a match, he also checks whether the second

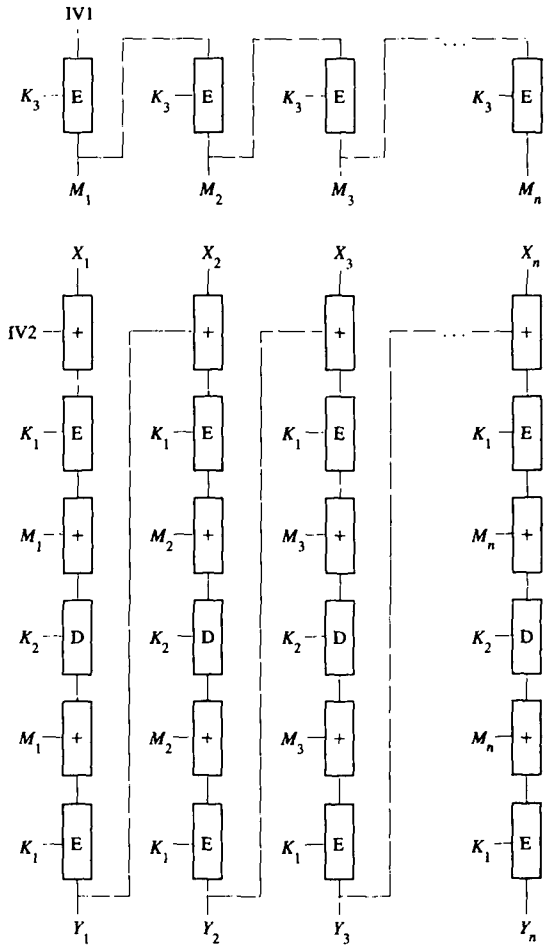

Figure 4

Triple-DES external feedback CBC with OFB masking (CBCM) mode.

match (predicted by the second coincidence) holds: namely, whether $d(k_1, Y_j) \oplus e(k_1, X_j \oplus Y_{j-1}) = d(k_1, Y'_i) \oplus e(k_1, X'_i \oplus Y'_{i-1})$. When both hold, one calls it a "double match." One expects that for the correct key $k_1 = K_1$, there will be one or two such double matches, while probably none will appear for incorrect keys, so this procedure serves to find K_1. Its cost is $2^{56} \times 2 \times 2^{33} = 2^{90}$ decipherments.

Having found K_1, it is easier to find K_3 and K_2 in the reduced scheme. (We are grateful to B. Preneel for suggesting the following improvement to our original attack.)[9] Given are $2^{16} = 65536$ ciphertext strings, each containing $2^{17} = 131072$ identical blocks $Y_{a,b} = y_a$,

[9] B. Preneel, Dept. Elektrotechniek—ESAT, Heverlee, Belgium, 1995; private communication.

$a \le 2^{16}$, $b \le 2^{17}$, and identical, known, IV1. He looks for double matches as before: $d(K_1, Y_{a,b}) \oplus e(K_1, X_{a,b} \oplus Y_{a,b-1}) = d(K_1, Y_{c,d}) \oplus e(K_1, X_{c,d} \oplus Y_{c,d-1})$ and $d(K_1, Y_{a,d}) \oplus e(K_1, X_{a,d} \oplus Y_{a,d-1}) = d(K_1, Y_{c,b}) \oplus e(K_1, X_{c,b} \oplus Y_{c,b-1})$. Find $M_b \oplus M_d = d(K_1, Y_{a,b}) \oplus d(K_1, Y_{c,d})$ with $b, d \le 2^{17}$. (He expects to find one or two such double matches, by a birthday attack.) Now for each of 2^{56} trial values k_3 for K_3, and using the known initial vector IV1, he runs through 2^{17} steps of output feedback mode to discover the values $M_1, M_2, \cdots, M_b, \cdots, M_d$, corresponding to this guessed value of k_3. If the guess is correct, the calculated values M_b and M_d will satisfy the equation, while there will probably be no false alarms from incorrect guesses. Having thus found K_3, at a cost of $2^{56} \times 2^{17} = 2^{73}$ encipherments, he can find K_2 by key exhaustion, at a cost of 2^{56} encipherments.

The requirements of 2^{34} blocks of chosen ciphertext and $2^{90} + 2^{17} + 2^{56} \simeq 2^{90}$ encryptions render this attack infeasible for many years to come.

A related attack uses 2^{34} known plaintext and corresponding ciphertext (instead of chosen ciphertext), and 2^{90} encryptions, but seems to additionally require 2^{112} elementary operations such as XOR ($\oplus$). Considering a DES encipherment to be as complex as $256 = 2^8$ elementary operations, this is the equivalent of 2^{104} encipherments. The attacker is assumed to know IV. For each trial value k_1 for K_1, he evaluates the quantity $e(k_1, Y_{i-1} \oplus X_i) \oplus d(k_1, Y_i)$ for each index $i \le 2^{34}$, and looks for a few matches: $e(k_1, Y_{i-1} \oplus X_i) \oplus d(k_1, Y_i) = e(k_1, Y_{j-1} \oplus X_j) \oplus d(k_1, Y_j)$. He hopes that these matches arise because $d(k_1, Y_i) \oplus M_i = d(k_1, Y_j) \oplus M_j$. With each trial key k_1 he lists some triples $(i, j, M_i \oplus M_j)$ obtained by this process. Now for each trial value k_3 for K_3, he evaluates the various M_i from the known IV1, and sees whether any triples on the list are satisfied. He needs to do about 2^{90} encipherments altogether, but checking the lists appears to require 2^{112} elementary operations.

If known plaintext and corresponding ciphertext are not available in these relatively large amounts (2^{34} blocks), the only known attacks seem to require more than 2^{112} encipherments.

Remark on reusing the IV

This paper outlines several attacks involving the decryption of two or more messages using the same IV. This reinforces the good advice never to use the same IV twice. Even if this advice is followed, one might be concerned that an attacker with temporary access to the target machine could force the reuse of the same IV on several decrypted messages. For this reason, we have made the rather conservative assumption that the attacker has this capability, and we have evaluated the strength of our present system against such an attacker.

Some design rationale

This mode was designed to yield high security without too much computational complexity. At this point in the discussion we are able to describe the reasoning behind some of the choices which were made, recognizing that other design choices might also have satisfied the criteria:

- *Why is M_i XORed in twice?*
  If the second instance of M_i were eliminated, we could mount an attack with a few chosen ciphertexts, two instances of the same IV, and the cost of single-key exhaustion. Namely, design two ciphertexts Y_i and Y'_i with $Y_2 = Y_3$ and $Y'_2 = Y'_3$. Obtain the two decipherments X_i and X'_i. Verify that the following equation must hold: $e(K_1, X_2 \oplus Y_1) \oplus e(K_1, X_3 \oplus Y_2) = e(K_1, X'_2 \oplus Y'_1) \oplus e(K_1, X'_3 \oplus Y'_2)$. Try all possibilities k_1 for K_1 to find the correct one. Very loosely speaking, the present scheme, by sandwiching two instances of "$M_i \oplus *$" around the decryption step $d(K_2, *)$, inherits some strength from the one-way function $f(y) = y \oplus e(K, y)$.
- *Why use the outer encryptions by K_1?*
  As seen in the attack above, once K_1 is discovered, the rest of the scheme is considerably weaker.
- *Why use K_1 twice? Why not use K_4 for the second instance of K_1?*
  Using a separate K_4 instead of the second instance of K_1 did not appear to add appreciably to the strength of the scheme, while it did increase the number of secret keys that had to be transmitted and stored.
- *Why not use two different quantities M_i and M'_i in the XOR step?*
  This would again have required an additional secret key K_2 to generate the second stream. More significantly, it would have required an additional encryption for each block, for a total of five encryptions per block rather than the present four. Even without this enhancement, the present scheme seems secure enough to withstand attacks for many years to come.

Properties of the present mode

Finally, we are able to list the properties of the present mode, to enable comparison with other modes.

1. The input and output block size is 64 bits, the same as normal DES.
2. It is not backward compatible with respect to single-key DES encryption.
3. It has limited error propagation with regard to corruption, but unlimited with regard to synchronization:
   - If one block of ciphertext is corrupted, only two blocks of recovered plaintext will be corrupted.

- If synchronization is lost (a block is lost), all succeeding recovered plaintext will be in error.

4. It requires four DES encryptions or decryptions per 64-bit block. The OFB process (using K_3) can be run in parallel with the CBC process (using K_1 and K_2), but the CBC process cannot be pipelined except by interleaving the data as discussed above with regard to the TCBC mode.
5. It requires maintenance of three secret keys and two nonsecret initialization vectors (IV1 and IV2).
6. It cannot be simulated easily using existing DES modes of operation.
7. It appears to be secure against chosen plaintext and chosen ciphertext attacks: known attacks require 2^{34} blocks of chosen text and 2^{90} encryptions.
8. It has a complementarity property:
   - If IV1, K_3, and K_2 are all complemented, the resulting method is not affected; it still encrypts X_i into Y_i.
   - If IV2, K_1, and K_2 are all complemented, the resulting method encrypts X_i into $C(Y_i)$, the complement of Y_i.
9. Most significantly, the potential concerns with text dictionary and with matching ciphertext have been solved.

Concluding remarks

We have presented a new method for multiple DES encryption. Like other triple-DES modes, it uses several independent keys, achieving strength against key-exhaustion attacks. Unlike other modes, it also defends against attacks based on the small block size, namely dictionary attacks and matching ciphertext attacks. We therefore recommend that the new mode (CBCM) be adopted by ANSI.

Acknowledgments

Ron M. Smith, Jr., of Poughkeepsie discussed the threat of dictionary attacks with us. Chris Holloway, Bart Preneel, and Michael Steiner gave extensive, valuable advice improving the accuracy, content, and readability of the paper.

Addendum

The IBM proposal to ANSI has been amended to restrict IV1 to a collection of 1048576 possible values. The authors thank David Wagner for pointing out difficulties that arise when IV1 can be freely chosen.

References

1. Federal Information Processing Standards Publication (FIPS PUB) 46-2, *Data Encryption Standard (DES)*, National Institute of Standards and Technology, Washington, DC, 1993.
2. 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.
3. F. Hendessi and M. R. Aref, "A Successful Attack Against the DES," *Information Theory and Application, Third Canadian Workshop Proceedings*, 1994, pp. 78–90.
4. Frank Rubin, "Foiling an Exhaustive Key-Search Attack," *CRYPTOLOGIA* **11,** No. 2, 102–107 (April 1987).
5. M. Hellman, "A Cryptanalytic Time-Memory Trade-Off," *IEEE Trans. Info. Theory* **IT-26,** No. 4, 401–406 (1980).
6. E. Biham and A. Shamir, *Differential Cryptanalysis of the Data Encryption Standard*, Springer-Verlag, New York, 1993.
7. M. Matsui, "The First Experimental Cryptanalysis of the Data Encryption Standard," *Lecture Notes in Computer Science* **839,** *Advances in Cryptology—Crypto '94 Proceedings*, Springer-Verlag, New York, 1994.
8. D. Coppersmith, "The Data Encryption Standard (DES) and Its Strength Against Attacks," *IBM J. Res. Develop.* **38,** No. 3, 243–250 (1994).
9. Eli Biham, "On Modes of Operation," *Lecture Notes in Computer Science* **809,** *Fast Software Encryption, Cambridge Security Workshop Proceedings*, Cambridge, U.K., Springer-Verlag, Berlin, 1994, pp. 116–120.
10. Eli Biham, "Cryptanalysis of Multiple Modes of Operation," *Lecture Notes in Computer Science* **917,** *Advances in Cryptology—ASIACRYPT '94 Proceedings*, Springer-Verlag, Berlin, 1994, pp. 278–292. Early draft provided to ANSI X9.F.1.
11. B. S. Kaliski, Jr. and M. J. B. Robshaw, "Multiple Encryption: Weighing Security and Performance," *Dr. Dobb's Journal* **21,** No. 1, 123–127 (1996).
12. P. C. van Oorschot and M. J. Wiener, "A Known-Plaintext Attack on Two-Key Triple Encryption," *Lecture Notes in Computer Science* **473,** *Advances in Cryptology—Eurocrypt '90 Proceedings*, Springer-Verlag, Berlin, 1991, pp. 318–325.
13. Carl H. Meyer and Stephen M. Matyas, *Cryptography—A New Dimension in Computer Data Security*, John Wiley & Sons, Inc., New York, 1982.
14. ANSI X3.106, "Data Encryption Algorithm—Modes of Operation," American National Standards Institute, May 16, 1983.
15. ISO/IEC 8372, "Information Technology—Data Cryptographic Techniques—Modes of Operation for a 64-Bit Block Cipher Algorithm," ISO/IEC 8372, 1987.
16. ISO/IEC 10116, "Information Technology—Security Techniques—Modes of Operation of an n-Bit Block Cipher Algorithm," IS 10116, 1991.
17. Matt Blaze, "A Cryptographic File System for Unix," *Proceedings of the First ACM Conference on Computer and Communication Security*, Fairfax, VA, November 1993, pp. 9–16.
18. P. C. van Oorschot and M. J. Wiener, "Parallel Collision Search with Cryptanalytic Applications," *J. Cryptol.*, to appear, 1996.

Received December 11, 1995; accepted for publication January 17, 1996

Don Coppersmith *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (COPPER at YKTVMV, copper@watson.ibm.com).*
Dr. Coppersmith received his B.S. in mathematics from the Massachusetts Institute of Technology in 1972 and his M.S. and Ph.D. in mathematics from Harvard University in 1975 and 1977. Since that time he has been a research staff member at the IBM Thomas J. Watson Research Center. His current research interests include cryptography and computational complexity. Dr. Coppersmith is a fellow of the Institute of Electrical and Electronics Engineers.

Donald B. Johnson *IBM Internet Division, 522 South Road, Poughkeepsie, New York 12601 (dbj@vnet.ibm.com).* Mr. Johnson received a B.A. in mathematics from Oakland University, Rochester, Michigan, and an M.S. in computer science from Union College, Schenectady, New York. Before joining the IBM Cryptography Center of Competence at Poughkeepsie in 1987, Mr. Johnson worked as an MVS PSR, on the DPPX C/T and with APL. He is the IBM representative to ANSI X9.F.1 and X9.F.3, IEEE P1363, and the X/Open Crypto API workgroup. He is an architect of IBM's crypto solutions and is one of the innovators of Control Vectors, the CDMF algorithm, and the RACF™ passticket algorithm. Mr. Johnson received an IBM Outstanding Innovation Award for his contributions to the IBM Common Cryptographic Architecture Application Programming Interface, and an FSC President's Patent Award for being one of the inventors of two "one-rated" patents which were deemed to be of strategic importance to IBM. He has achieved the eighth plateau in the IBM Invention Achievement Award program and holds 24 patents. Currently, he is a senior programmer.

Stephen M. Matyas *IBM Internet Division, 522 South Road, Poughkeepsie, New York 12601 (smatyas@vnet.ibm.com).* Dr. Matyas is a former member of the Cryptography Competency Center at the IBM Kingston Development Laboratory and a former manager of the Secure Products and Systems Department at the IBM FSC Manassas facility. He is currently a member of the Information Technology Security Programs Department at the IBM Poughkeepsie facility. He has participated in the design and development of all major IBM cryptographic products, he played a lead role in the design of IBM's Common Cryptographic Architecture (CCA), and he is the originator of the control vector concept incorporated in the CCA. Dr. Matyas is a senior technical staff member. He holds 52 patents and has published numerous papers covering aspects of cryptographic system design. He is the co-author of an award-winning book entitled *Cryptography—A New Dimension in Computer Data Security*, and a contributing author to the *Encyclopedia of Science and Technology* and *Telecommunications in the U.S.—Trends and Policies*. Dr. Matyas received a B.S. degree in mathematics from Western Michigan University and a Ph.D. degree in computer science from the University of Iowa. He has received an IBM Outstanding Innovation Award and an IBM FSC President's Patent Award; he has achieved a twenty-first plateau in the IBM Invention Achievement Award program.