コンピュータセキュリティ　18−28
(2002. 7. 19)

ブロック暗号における秘密鍵の平文ブロックのマスクについて

2-key XCBCによるMAC生成スキームの安全性

古屋　聡一†　　櫻井　幸一††

† (株) 日立製作所 システム開発研究所　〒244–0817 横浜市戸塚区吉田町 292
†† 九州大学大学院システム情報科学研究院　〒812–8581 福岡市東区箱崎 6-10-1
E-mail: †soichi@sdl.hitachi.co.jp, ††sakurai@csce.kyushu-u.ac.jp

あらまし　2-key XCBC は盛合今井が提案した CBC-MAC に基づく MAC 生成手法である。この方式は、秘密 PRP の呼出回数、鍵セットアップ回数、鍵資源の数の観点からもっとも効率的な手法であり、また呼び出す秘密 PRP の安全性に基づいて MAC としての安全性が証明されている。本稿では、2-key XCBC の本質である、秘密鍵を平文にマスクすること、の安全性への影響について議論する。PRP が特別な置換に限定される場合として、Even-Mansour 構築を使ったある 2-key XCBC の例では攻撃者が改竄可能であることを示す。また 2-key XCBC に DESX construction を使った例では、このスキームを攻撃するのに必要な計算量が、DESX が証明する安全性の下限を下回ることを示す。さらに別の観点からの安全性の評価として、2-key XCBC を AES や Camellia に用いた場合の安全性についても議論する。

キーワード　ブロック暗号, 操作モード, 証明可能安全性, XCBC, AES.

Risks for Raw-key Masking

the Security of 2-key XCBC MAC-generation scheme

Soichi FURUYA† and Kouichi SAKURAI††

† Hitachi, Systems Development Laboratory,　292 Yoshida, Totsuka, Yokohama 244–0817 Japan
†† Dept. of CSCE., Kyushu University,　6-10-1 Hakozaki, Higashi-ku, Fukuoka 812–8581 Japan
E-mail: †soichi@sdl.hitachi.co.jp, ††sakurai@csce.kyushu-u.ac.jp

Abstract　Two-key XCBC proposed by Moriai and Imai is a CBCMAC-based method to generate a message authentication code. This method is optimal from several aspects, the number of PRP invocations, key setups, and key materials. This mode is alsp proven the security as a MAC. In this paper, we discuss how masking a secret key for a plaintext block effects to its security. Concerning that the PRP is limited to the special permutations, we show the two-key XCBC is forgeable if one uses Even-Mansour constructions. In case of using DESX construction for two-key XCBC, the necessary complexity to attack the scheme is below that what is proven for DESX. We also study the security of AES and Camellia when they are used in two-key XCBC mode from another aspect, rather than the context of the provable security.

Key words　Block cipher, mode of operation, provable security, XCBC, AES.

1. はじめに

CBC-MAC はブロック暗号から MAC を生成するのに最もよく用いられる方法である。CBC-MAC が PRP といっしょに用いられる場合、メッセージ長を変化させない攻撃者に対しては MAC タグが偽造不可である [3]。一方、メッセージ長が可変であるばあい、基本的な CBC-MAC は攻撃可能であることはよく知られている。

この問題を解決するために、いくつかの改良が提案され提案者によって評価されてきた。その改良の目標は以下の通りである：

（1）　連結攻撃に対しても安全な MAC 生成

（2）　任意長メッセージ、すなわち、ブロック暗号のブロック長とは無関係にメッセージを認証できる

（3） 最小の PRP 呼出し

（4） 最小の鍵資源

Black と Rogaway は [5] で Lemma 4 を示した。これは一つの PRP から二種類のランダム置換を同時に生成する方法とその安全性に関するものである。Moriai と Imai は Lemma 4 を拡張し、少ない鍵資源のスキームを提案した (2-key XCBC) [21]。

本稿では、2-key XCBC について考察し、2-key XCBC で用いられている PRP と、既知のブロック暗号や PRP 構築法との差について議論する。結論として、現実的には (仮に用いるブロック暗号が証明可能安全な要素技術であったとしても) 2-key XCBC が常に安全性を供給するモードではないことを示す。

まず、Even-Mansour 構築 [10] を使った場合について考え、改竄が可能であるような例を示す。この EM 構築法は、公開された PRP から、秘密の PRP を構築する手法であり、この構築法が疑似ランダム置換を実現することは証明されている [10]。次に、類似の性質を持つ DESX 構築 [17] についても議論する。ここで考える攻撃アルゴリズムは、効率的ではないものの、DESX で証明された安全性よりもはるかに少ない計算量での攻撃が成立する。

また、EM 構築や DESX などの証明可能安全な PRP 構築法だけでなく、実際のブロック暗号アルゴリズムを使う場合の安全性についても議論する。ここでは特に、秘密鍵をマスクすることの安全性への影響について、AES [13] と Camellia [1] を具体的に解析する。AES については、処理段数のみの観点からは、2-key XCBC が暗号の安全性を本質的に弱めてしまう場合があることを指摘する。また、Camellia については、等価鍵の観点から検討する。そして、2-key XCBC で用いる場合の安全性の検討は、ブロック暗号単体に対する等価鍵の検討とは異なるものとなることを示す。その結果、2-key XCBC で用いる状況での等価鍵の存在可能性を解析し、事実上、現状の Camellia では問題ないことを示す。

2. 準　　備

本稿で用いる表記についてまとめる。二つの整数 a と b に対し二整数の関係 $a|b$ は、a が b の約数である、ことを示す。二つのビット文字列 a と b に対し、$a \oplus b$ は排他論理和演算である。演算 $a <<<_n b$ は n ビットレジスタ値 a に対する b ビット左巡回シフトである。

$P_K(M)$ を二入力、鍵 K とメッセージブロック M を取る、疑似ランダム置換とする。b を $P_K(M)$ のブロックサイズ (ビット) とする。$len(M)$ を M のビット長とする。$b|len(M)$ であるような、ビット文字列 M に対し、添字の付いた M_i という表記により、b ビットブロックに分割したブロック列の i 番目要素を表す。二つのビット文字列 A と B について、$A||B$ は二つの文字列の連結を示す。

3. CBC-MAC ベースの MAC 生成法

これまで、いくつかの CBC-MAC [12] に基づいた MAC 生成スキームが提案されてきた。ここではこれらのスキームを復習する。

3.1 CBC-MAC とその変形

CBC-MAC は $m|b$ である null でないメッセージ M を以下のように処理して tag を生成する。

$$tag = CBCMAC_K(M)$$
$$= P_K(M_{\#M_i-1} \oplus P_K(M_{\#M_i-2} \oplus \cdots \oplus P_K(M_0))).$$

P が疑似ランダム置換である場合、秘密鍵を知らない攻撃者は、メッセージを変化させない限り偽造できない [3]。しかしながら、メッセージ長が可変な場合、一つのメッセージと有効な MAC の組み合わせ (M, tag_M) から以下のような、異なるメッセージに対する有効な MAC を生成できる。

$$MAC(M||M'||M'||\ldots||M') = tag.$$

ここで M' は M_1 を $M_1 \oplus tag_M$ に置き換えたものである。この攻撃を連結攻撃と本稿では呼ぶ。

CBC-MAC に続き、連結攻撃に対しても安全な、いくつかの MAC 生成スキームが提案されてきた [22]。ECBC の欠点 (二つの鍵セットアップ、追加の P 呼出し、M が b ビット長の倍数でなければならないという制限) を克服するために、3 つの鍵資源を使った手法が提案された。これらの技術的本質は、ケース 1($len(M) > 0$ かつ $b|len(M)$ の場合) の場合にはパディングは適用しない代わりに P とは (計算量的に) 別の疑似ランダム置換を適用することである。このアイデアを実現する一手法を示す。

（1） K_3 を使って、パディングあり/なしのメッセージを分ける。片方には K_2 を使い、もう片方に K_3 を使う。

（2） EMAC とは異なり、2 つの PRP を切替える場所を CBC モードの最終ブロックとする。

3-key ECBC [5] は上記 1 の方法を用いた方法であり、FCBC [5] は上記 1,2 両方を用いた手法である。FCBC により、どのメッセージについても最適な P の呼出し回数を実現できる。その一方で、3 つすべての鍵資源が P のセットアップに用いられてしまう(注1)。実際に FCBC を用いた場合、三つの異なる鍵資源が鍵スケジュール部に用いられることになり、例えば短いメッセージを扱う場合などには鍵セットアップにかかる計算量が無視できないほど大きくなる。

Black と Rogaway は、文献 [5] で CBC-MAC に基づく MAC 構成法としてはほぼ最適なスキームを提案している。このスキームでは、鍵セットアップはわずかに 1 回である。この改良にもっとも影響したのが、文献 [5] における補題 4 である。これは、置換ペア $(P(K \oplus \cdot), P(\cdot))$ と、同じく置換ペア $(P_1(\cdot), P_2(\cdot))$ を識別することが計算量的に困難であることを示している。これにより、FCBC で用いられる置換ペア $(P_{K_2}(\cdot), K_{K_3}(\cdot))$ を $(P_{K_1}(K_2 \oplus \cdot), P_{K_1}(K_3 \oplus \cdot))$ を置き換えても安全性のレベルが下がることはない。じゅうぶん高い安全性が達成されたまま鍵セットアップ数が最小となる。

(注1)：より厳密には、固定長の M には二つのみの鍵が常に必要である。しかし、メッセージ長に依存して第二の鍵が切り替わる。

3.2 2-key XCBC

盛合と今井[21]は、[5]の補題4をさらに拡張した。この補題では、関数のペア $(P_{K_1}(K_1 \oplus \cdot), K_{K_1}(K_2 \oplus \cdot))$ と同じく関数のペア $(P_1(\cdot), P_2(\cdot))$ を識別することは計算量的に困難である、ということを示す。この補題に基づき、$(P_{K_2}(\cdot), P_{K_3(\cdot)})$ を $(P_{K_1}(K_1 \oplus \cdot), P_{K_1(K_2 \oplus \cdot)})$ に安全に置き換えることができる。これにより、証明可能安全でありながら最小の鍵セットアップ回数(1)を保ちつつ、鍵資源1個をさらに削減することが可能である。

この補題を使ったスキーム例の一つが2-key XCBCであり、盛合と今井[21]により提案された。

2-key XCBC in [21]: 秘密鍵 $K1$、$K2$ を用意する (K_1 はランダム置換 P の鍵であり、K_2 は秘密マスクのための b ビット秘密情報である)。メッセージ M が、nullでなくかつ $len(M)|b$ の場合、$K_t = K_2$ とする。それ以外のメッセージ M には $K_t = K_1$ とし、メッセージ M にビット列 $10^{b-1-len(M) \bmod b}$ をパディングする。M を b ビットのブロックに分割したものから、最終ブロックを除いたものを M^- とする。以下をMACとして出力する。

$$tmp = CBC_{K1}(M^-)$$
$$tag = P_{K1}(tmp \oplus K_t)$$

このスキームでは、3-key XCBCの利点である、安全性、P の呼出回数 ($\lceil m/b \rceil$)、鍵セットアップ数(1)を損なわずに、鍵資源を3つから2つに改良している。3-key XCBCと同様に、P 呼出回数、鍵セットアップ数が最小である。また、文献[21]では、鍵資源が1個のスキームの存在可能性について考察し、このようなスキームは存在しないと結論付けていることから、証明可能なCBC-MAC変形の中では効率において最適であると主張されている。

4. 2-key XCBCのバリエーションと秘密鍵による平文マスク

ここでは、2-key XCBCについて詳細に議論する。文献[21]で提案された2-key XCBCの一例は既に述べた。しかし、別の変形を簡単に考えられる。まず、最も簡単で典型的な変形例2-key XCBC′を示す。

2-key XCBC′: 2-key XCBCと同様に秘密鍵 K_1、K_2 を用意する。メッセージ M が、nullでなくかつ $len(M)|b$ の場合、$K_t = K_1$ とする。それ以外のメッセージ M には $K_t = K_2$ とし、メッセージ M に $10^{b-1-len(M) \bmod b}$ をパディングする(2-key XCBCとは K_t の定義において、K_1 と K_2 が逆になっていることに注意する)。M を b ビットのブロックに分割したものから、最終ブロックを除いたものを M^- とする。以下をMACとして出力する。

$$tmp = CBC_{K1}(M^-)$$
$$tag = P_{K_1}(tmp \oplus K_t)$$

さらに、K_t を使ったさまざまな演算についてもその変形を考えることができる。最も簡単な例として $len(K_1) \neq b$ の場合である。$len(K_1) > b$ の場合、K_1 の一部分のみを使って K_t を生成し、$len(K_1) < b$ の場合、K_1 にデータ(定数orにK_1 の一部)をパディングして K_t を生成することが考えられる。また、K_t を平文ブロックにマスクする演算(2-key XCBCや2-key XCBC′では排他論理和)に、別の演算(例えば、算術加算や減算、巡回シフト命令を組み合わせたこれら演算)も考えることができる。そして、これらバリエーションそれぞれに対して、K_t を定義する K_1 と K_2 の選択が逆であるさらなる変形も考えることができる。

秘密鍵による平文マスク:2-key XCBC、およびその変形の安全性を議論するために、我々は特別なフォームの平文のみを考えてゆく。これらの特別な平文を定義するためにまず2-key XCBCのバリエーションを、次の二つに分けて考える;一つは、2-key XCBC グループ $K_t = K_1$ for $len(M) > 0$ and $b|len(M)$、であり、もう一つは2-key XCBC′ グループ、$K_t = K_1$ for $len(M) = 0$ or $b(not)|len(M)$、である。2-key XCBC グループについては、$len(M) = b$ のメッセージのみによる集合 $\mathcal{M}$ をメッセージ入力として考える(平文空間は 2^b であり十分な大きさである)。この場合には、P の呼出が1回であり、MAC生成の式は $tag = P_{K_1}(f(M_0, K_1))$ となる。ここで f は演算回数の少ない非暗号学的な(あるいは暗号学的に弱い)関数、例えばxor、加算、減算、K_1 への巡回シフトを伴うxorなどである。

同様にXCBC′ グループの場合には、$0 \leq len(M) < b$ であるようなメッセージの集合 $\mathcal{M}$ を考える(平文空間はやはり $\sum_{i=0}^{b-1} 2^i = 2^b - 1$ とじゅうぶん大きい)。この場合、メッセージには適当なパディングが施されて、最終的には $len(M) = b$ となり、P の呼出は1回だけとなる。よって、MAC生成の式は、同様に f を使って $tag = P_{K_1}(f(M_0, K_1))$ と書ける。以降では、XCBCの安全性について、特に、$tag = P_{K_1}(f(M_0, K_1))$ の疑似乱数性について議論してゆく。

注意:本稿で扱う設定では、攻撃者が、自分で選ぶ暗号文ブロック C に対してその平文ブロック $P_{K_1}^{-1}(C)$ を返すようなオラクルは存在しないと仮定している。なぜならば、我々の設定はMACのスキームであり、このスキーム自体では $P_{K_1}^{-1}(C)$ は実装されないからである。しかし、もし、あるプロトコルで実装が不備であり、攻撃者がこのような復号オラクルを用いることができるならば、2-key XCBCに基づくすべてのスキームは安全でない。このような設定は K_1 を暗号処理(例えばCBCモード)への同時の利用などにより実現されるが、本稿では評価が自明であり、評価の対象外とし、扱わないことにする。

5. PRP構築法を使う2-key XCBCの安全性

ここでは、$tag = P_{K_1}(f(M_1, K_1))$ の安全性について議論する。特に P が証明可能安全なPRPの構築法である場合を考える。f については考えうるような関数、主に排他的論理和を扱い、そのtruncationやextensionは自然なものを扱う。

これまで述べたように、攻撃の対象となるメッセージは $M \in \mathcal{M}$ のみを扱う。$\mathcal{M}$ の定義より、上で示したすべての変形2-key XCBCに対して、MAC生成の関数が $tag = P_{K_1}(f(M_1, K_1))$ に置き換えることができる。攻撃者はメッセージと正しいMAC

のペア (M_i, tag_i) を自由に、かつ十分な数だけ得ることができる。よって、この攻撃者は関数 $tag = P_{K_1}(f(M_1, K_1))$ に対する選択平文攻撃の環境とみなすことができる。

5.1 Even-Mansour 構築法

Even-Mansour 構築法により、公開された PRP は効率的に証明可能安全な秘密 PRP へ変換が可能である [10]。この構築法は効果的であり、特に秘密鍵が頻繁に更新されるような場合はひじょうに効率的である。これに加え、通常のブロック暗号における、データ撹拌部分の効率も向上することができる。秘密 PRP は多くの暗号学的スキーム、例えば MAC 生成や秘密鍵暗号など、に用いられることから、鍵が頻繁に変わるような実装では公開の PRP(例えば、鍵 0 が使われる AES) を使った EM 構築法は利点がある。

EM 構築法の安全性は文献 [10] で証明されている。EM 構築法に必要な秘密情報の長さは $2b$ ビットである。ここで b は、公開 PRP のブロック長 (ビット) である。このとき、EM 構築法の証明可能安全性により、実効鍵長は $b - l - m$ ビットとなる。ここで l と m は、攻撃者の能力によって決められるパラメータであり、2^l、2^m はそれぞれ攻撃者が行う EM 構築法、公開 PRP のオラクルへの呼出回数の上限である。

EM 構築法は以下のとおり。EM 構築法にはパラメータがある；公開 b ビットブロック PRP である P と、二つの b ビット秘密鍵 K_a、K_b である。これらを用いて EM 構築法を次のように定義する：

$$EM_{P,K_a,K_b}(M) = P(M \oplus K_a) \oplus K_b.$$

EM 構築法がブロック暗号として用いられる場合、鍵セットアップに必要な計算量はほぼ無視できる。よって、もし PRP が複数の鍵資源に用いられる場合 (例えば、3-key XCBC [5] など)、現状安全とされるブロック暗号の多くに対して、鍵セットアップの効率性で利点がある。

ここで、EM 構築法による秘密 PRP を使った 2-key XCBC の安全性を議論する。攻撃者はメッセージとその正当な MAC のペア $(M^{(i)}, tag_i)$ が入手可能である。攻撃者は $(M^{(i)}, tag_i)$ の知識に基づいて、どの $M^{(i)}$ とも異なるメッセージ $M' \in \mathcal{M}$ を生成し、M' に対する正当な MACtag' を生成する。

ここでは、2-key XCBC バリエーションのうち、特別なものを考える。これは平文ブロックへの秘密情報マスク (鍵マスク) が EM 構築法で定義される K_a によるものである場合である。この例のうち典型的なもの 2 例が $2kXCBC_{EM_{P,K_a,K_b},K_a,K_3,\oplus}$ と $2kXCBC'_{EM_{P,K_a,K_b},K_3,K_a,\oplus}$ である。これらの変形はすべての可能なバリエーションの中でももっとも自然にありうるものである。なぜならば、K_a の長さは、2-key XCBC(および 2-key XCBC′) の鍵マスクに必要な長さ条件と一致し都合がよいからである。以下にこれらの安全性に関する補題を与える。この補題は 2-key XCBC のみを扱うが、同様の事実が 2-key XCBC′ にも適用可能である。

[補題 1] (Even-Mansour 構築法を使った 2-key XCBC の安全性について) もし、2-key XCBC の鍵マスクが K_a、すなわち Even-Mansour 構築の平文側の鍵資源、で定義される場合、2-key XCBC は安全でない。より厳密には、このスキームに対しては攻撃者が存在し、その攻撃者は 1 回の MAC 生成オラクルを呼出し、2 回の公開 PRP オラクルの呼出しを行う。改竄には無視できる計算量とメモリしか必要としない。

証明：補題にある 2-key XCBC で生成される MAC を偽造する攻撃者を以下に示す。簡単なため、文献 [21] で示す 2-key XCBC に対する攻撃者のみを扱うが、同様の攻撃は 2-key XCBC′ の同じバリエーションにも適用可能である。

攻撃アルゴリズム

事前計算

1. 攻撃者は、$0 \leqq |M_1| < b$ であるようなあるメッセージ M_1 を選択し、M_1 に対する正当な MAC、tag_1 を入手する。
2. 攻撃者はパディング後のメッセージとして $M_1^+ = M_1||10^i$ を生成する。ここで、$b|len(M_1^+)$ かつ $0 \leqq i < b$ である。
3. 公開されている P を使って、攻撃者は $tag' = P(M_1^+)$ を計算する。
4. $K_b' = tag_1 \oplus tag'$ とする。

改竄

5. $0 \leqq |M_2| < b$ かつ $M_2 \neq M_1$ であるような、改竄したいメッセージ M_2 を生成する。
6. パディング後のメッセージ $M_2^+ = M_2||10^i$ を計算。ここで $b|len(M_2^+)$ かつ $0 \leqq i < b$ である。
7. M_2 に対する MAC として $tag_2 = P(M_2^+) \oplus K_b'$ を出力する。

攻撃の解析

攻撃者は、$0 \leqq |M_1| < b$ となるように M_1 を選択するので M_1 は MAC 生成内部でパディングされ、M_1^+ と同じデータを扱う。また、同じ理由から 2-key XCBC は鍵マスクに K_a を使う。よって、M_1 に対する正当な MAC は数学的に以下のように記述される：

$$\begin{aligned} tag_{M_1} &= EM_{P,K_a,K_b}(M_1^+ \oplus K_t) \\ &= P(M_1^+ \oplus K_a \oplus K_a) \oplus K_b \\ &= P(M_1^+) \oplus K_b. \end{aligned}$$

攻撃者は $tag' = P(M_1^+)$ を知っているので：

$$\begin{aligned} tag' \oplus tag_{M_1} &= P(M_1^+) \oplus (P(M_1^+) \oplus K_b) \\ &= K_b. \end{aligned}$$

よって、攻撃者が計算可能な K_b' は K_b に等しい。

ここで tag_2 が M_2 に対する正当な MAC であるかどうか検証する。M_2 は $0 \leqq |M_2| < b$ となるように選択されているので、M_2 に対する正当な MAC は以下のようになる：

$$\begin{aligned} tag_{M_2} &= EM_{P,K_a,K_b}(M_2^+ \oplus K_t) \\ &= P(M_2^+ \oplus K_a \oplus K_a) \oplus K_b \\ &= P(M_2^+) \oplus K_b. \end{aligned}$$

この MAC、tag_2 は攻撃者が生成した tag_{M_2} と同じである。 □

この種の攻撃は、PRP の外部で鍵マスクをすること自体に

ある本質的弱さを用いたものである。ここでは、最も弱い (すなわち、攻撃者にとって都合のよい)f を使った。関数 f は単にデータをマスクするだけが目的の関数であるので、その他のバリエーションについても適用の可能性はある。

例えば、f 関数として、ある定数 c を使った $f(M,K) = M \oplus (K <<<_b c)$ を考えた場合、攻撃者は、$k \oplus (k <<<_b c)$ の値をゲスする追加の計算量が必要である。しかし、この場合、全体で必要な計算量は、EM 構築法の証明可能安全性と (b と c が互いに素である場合には) 等しいか、または (そうでない場合には)EM 構築法の安全性よりも低くなる。

5.2 DESX

DESX は、ブロック暗号の (特に鍵の全数探索に対する) 安全性を、とても小さな付加処理 (と付加的秘密情報) で向上する方法である。この手法は Rivest により提案されたが、文献として残されていない (このことは文献 [19] に示されている)。DESX の安全性証明は Kilian と Rogaway によって達成されている [17]。

DESX には二つのバリエーションがある。まず一方 (3-key variation) を示し、もう片方がその特殊な場合として扱う。P_{K_a} は、秘密鍵 K_a をパラメータとしてもつ、b ビットの PRP とする。このとき 3-key DESX は、さらに二つの b ビット鍵を必要とする。これらを、K_b、K_c とする。このとき 3-key DESX は以下のように定義される：

$$DESX_{P_{K_a},K_b,K_c}(M) = P_{K_a}(K_b \oplus M) \oplus K_c.$$

2-key のものは $K_b = K_c$ であるように K_b と K_c を定義する。

DESX は容易に見てわかるとおり EM 構築法と類似している。よって同じような攻撃を DESX の特別な場合について考えることができる。しかし、EM 構築法のようには、改竄を効率的に行うことができず、ここで示す攻撃は、改竄が、DESX の証明可能安全性よりも簡単にできる、ということを示す。より厳密には、DESX を用いた 2-key XCBC は K_a のみに対する鍵の全数探索により攻撃が可能である。

同じように、補題によりこのことを示す。まず、対象とする **DESX を使った 2-key XCBC** を定義する。P_{K_a} を秘密鍵 K_a を使った PRP であるとする。ここで扱う DESX は以下のように定義される：

$$DESX_{P_{K_a},K_b,K_c}(M) = P_{K_a}(K_b \oplus M) \oplus K_c.$$

これを用いて、DESX を使った 2-key XCBC の特別な場合として以下のものを考える：

$$2kXCBC_{DESX_{P_{K_a},K_b,K_c},K_b,K_3}.$$

この MAC スキームの安全性を検討する。

[補題 2] (DESX を使った 2-key XCBC の安全性) DESX の平文側鍵マスク (上記の DESX の定義では K_b) が 2-key XCBC の鍵マスクとして使われる場合、2-key XCBC の安全性は、DESX の証明可能安全性では保証されない。より厳密には、ある攻撃者が存在して、その攻撃者に必要な計算量はほぼ、K_a への全数探索に必要な計算量と同等である。

証明：ここでは、2-key XCBC に DESX を用いた MAC スキームを攻撃するアルゴリズムを示す。EM 構築の場合と同様に、2-key XCBC は文献 [21] で示されたもののみをこの証明では扱うが、同様な攻撃は DESX を使った 2-key XCBC $'$ にも適用することができる。

攻撃アルゴリズム

事前計算

1. 攻撃者は、$0 \leqq |M_i| < b$ であるようなあるメッセージ M_1, M_2 を選択し、M_i に対する正当な MAC、tag_i をそれぞれ入手する。
2. 攻撃者はパディング後のメッセージとして $M_i^+ = M_i||10^j$ を生成する。ここで、$b|len(M_i^+)$ かつ $0 \leqq j < b$ である。
3. K_a に対する全数探索により、以下の方法で正しい (K_a, K_c) のペアを求める：
   （1） 各々の K_a 候補 $K_a^{(i)}$ に対し、$P(\cdot)$ を呼出し、$K_c^{(i)} = P_{K_a^{(i)}}(M_1) \oplus tag_1$ とする。
   （2） もし、$P_{K_a^{(i)}}(M_2) \oplus K_c^{(i)} \neq tag_2$ ならば、鍵ペアの候補 $(K_a^{(i)}, K_c^{(i)})$ を棄却し、次の K_a 候補を試す。さもなければ、$(K_a', K_c') = (K_a^{(i)}, K_c^{(i)})$ としてループを終了する。

改竄

5. $0 \leqq |M'| < b$ かつ $i = 1, 2$ について $M' \neq M_i$ であるような、改竄したいメッセージ M' を生成する。
6. パディング後のメッセージ $M'^+ = M'||10^j$ を計算。ここで $b|len(M'^+)$ かつ $0 \leqq j < b$ である。
7. M' に対する MAC として $tag' = P_{K_a'}(M'^+) \oplus K_c'$ を出力する。

攻撃の解析：この攻撃の本質は、EM 構築法の場合の攻撃と同じである。攻撃者は M_i を $0 \leqq |M_i| < b$ となるように選択する。M_i は MAC 生成の内部でパディングされ、M_1^+ と同じ値を生成する。また、同じ理由で 2-key XCBC は K_b を鍵マスクに用いる。よって、M_i に対する正当な MAC は以下のようになる：

$$\begin{aligned} tag_{Mi} &= DESX_{P_{K_a},K_b,K_c}(M_i^+ \oplus K_t) \\ &= P_{K_a}(M_i^+ \oplus K_b \oplus K_b) \oplus K_c \\ &= P_{K_a}(M_i^+) \oplus K_c. \end{aligned}$$

ここで、ある固定した M_1 について、そのメッセージと正当な MAC である (M_1^+, tag_{M_1}) を考える。DESX による 2-key XCBC のすべての鍵空間上で (M_1^+, tag_{M_1}) が正当なメッセージと MAC のペアであるような鍵はちょうど 2^k 個ある。その 2^k 個の中の一つのみが、攻撃の対象である MAC スキームに用いられている。鍵候補を一つもしくはごく少数に限定するために、別のメッセージと MAC のペアである、(M_2^+, tag_{M_2}) を用いる。ここで、M_2^+ は、(M_1^+ と同じ方法で) メッセージ M_2 にパディングしたものである。このようにして、攻撃者は (K_a, K_c) の正しい鍵ペアを求める。これらを (K_a', K_c') として、攻撃に用いる。

ここで、tag' がメッセージ M' に対して正当な MAC であるかどうかを検証する。M' は $0 \leq |M'| < b$ となるように選択されたので、メッセージ M' の正当な MAC は以下のように定義される：

$$\begin{aligned} tag_{M'} &= DESX_{P_{K_a},K_b,K_c}(M'^{+} \oplus K_t) \\ &= P_{K_a}(M'^{+} \oplus K_b \oplus K_b) \oplus K_c \\ &= P_{K_a}(M'^{+}) \oplus K_c. \end{aligned}$$

この tag' は、攻撃者が生成した $tag_{M'}$ と一致している。 □

この補題により、雑に言えば DESX はこの場合、DESX に期待される通常のブロック暗号に対する安全性の利点がないことが言える。この攻撃に必要なデータはほんの数ペアのメッセージと MAC である。この攻撃には、内部で用いるブロック暗号の鍵の全数探索とほぼ同じ計算量が必要であるが、スキーム全体を攻撃するのに必要な計算量は DESX で証明されている安全性で保証されるそれよりもはるかに少ない。より厳密には、c 回の $DESX_{P_{K_a},K_b.K_c}$ の呼出しと、2^k 回のオフラインな P の呼出し、そして無視できるサイズのメモリが必要である。

6. ブロック暗号を使う 2-key XCBC の安全性

ここまでは、2-key XCBC の操作モードが中で用いる暗号学的要素を本質的に弱めてしまうような特別な場合を考えた。この議論は、証明可能安全な PRP 構築法のみに適用されるものではなく、ブロック暗号のような安全性のよりどころが (証明ではなく) 発見的手法の積み重ねから経験的に得られるものにも適用できる。

その一つの例が AES 暗号である。本稿では、(暗号学的には事実上安全性の指標となる) 実効段数という観点からは、2-key XCBC が AES 暗号を弱めていることを示す。次の例として、次世代ブロック暗号として有力視される Camellia についても考察する。特に本稿では、2-key XCBC に Camellia を用いた場合の安全性について検討する。特に鍵スケジュール部の評価を通じて、この観点からの安全性評価の結果として、2-key XCBC は Camellia の暗号学的安全性を弱めないことを示す。しかしながら、我々のこの観点からの安全性評価は、Camellia の鍵スケジュール部の性質を解析したものであり、自明な結果ではない。

6.1 AES

AES [13](もともと Rijndael [9] として提案された) は鍵長 128/192/256 ビットの 128 ビットブロック暗号である。段数 r は鍵長 (128, 192, 256) に応じて 10, 12, 14 と定義される。鍵スケジュール部では、$(r+1)$ 個の 128 ビット拡大鍵を生成する。最初の段、すなわち最も平文側の段鍵は秘密鍵の上位 128 ビットに同じである。

平文を暗号化するとき、入力は最初の拡大鍵と排他論理和したあと、第一段の処理として、非線形変換、shiftrow、mixcolumn、そして鍵加算 (演算は排他的論理和) が続く。最終段 (128 ビット鍵の場合第 10 段) では、mixcolumn 演算は処理ない。

ここで、2-key XCBC を AES と一緒に用いた場合についてより詳細に検討する。あるメッセージ $M \in \mathcal{M}$ について、2-key XCBC は一般に以下のような演算により MAC を生成する：$tag_M = P_{K_1}(f(M, K_1))$. ここで関数 f を $f_{high128\cdot\oplus}$、M と K_1 の上位 b ビットによる b ビット排他論理和、とする。この鍵マスク K_1 の定義は、安全性の欠陥をわざと埋め込んだ複雑な演算とは異なり、極めて自然で仕様としてありうる演算である。

この場合、AES の初期鍵加算は、2-key XCBC モードで処理する鍵マスクとまったく同じ演算となる。よって、これら二つの演算はお互いに打ち消しあう。結果として、パディング後のメッセージブロックである M^+ そのものが、初期鍵加算後の中間値となる。AES では、第一段の最後にある鍵加算までは鍵に依存した演算がないため、攻撃者は秘密鍵の情報を知らずに、中間値を一段分の復元することができる。このことは、攻撃者にとって、2-key XCBC モードを攻撃する場合には、$(r-1)$ 段の変形 AES を選択平文攻撃により攻撃することで十分ということになる。

ブロック暗号の近年の安全性評価はその段数と綿密な関係がある。多くの暗号学的攻撃手法は、1 段あたりの性質を使い、これをなるべく多くの段数でも検出できるように解析する。このようなブロック暗号の安全性評価に関する一般的な観点からは、AES を使った 2-key XCBC モードは、中で用いているブロック暗号 AES を本質的に弱めている、と考えることができる。

6.2 Camellia

Camellia は AES と同様、鍵長 128/192/256 の 128 ビットブロック暗号である [1]。Camellia は全体に Feistel 構造を採り入れているが、これに追加の要素関数が用いられている；初期、最終鍵加算、および FL、FL^{-1} 関数である。鍵スケジュール部では、二つの中間鍵 K_L、K_A(また、鍵長 192, 256 ビットの場合はこれらに加え K_R、K_B も) が秘密鍵より生成される。各々の拡大鍵 (段鍵) は、これら中間鍵のひとつを巡回シフトしたものの、上位または下位半分 (64 ビット) である。

Camellia が 2-key XCBC で用いられた場合、暗号学的強度は特に関数 $Came_{K1}(f(M,K1))$ について注目する必要がある。Camellia では、平文はまず 128 ビットの初期鍵 (kw_1, kw_2) で排他論理和される。128 ビット秘密鍵に対する鍵スケジュール部の仕様では (kw_1, kw_2) の定義から、$kw_1 || kw_2 = K1$ となっている。また、他の鍵長の場合、2-key XCBC の鍵加算関数 f を $f = f_{high128\cdot\oplus}$ のように定義することで同じ結果が得られる。

よって、2-key XCBC を Camellia とともに用いた場合、あるメッセージ $M \in \mathcal{M}$ に対する MAC を生成した場合、AES の場合と同様に、2-key XCBC の鍵マスクと、初期鍵加算が同じ排他論理和演算となる。よって、初期鍵加算のあとの中間値はパディングされたメッセージの値と同じである。よって、このモードでは、初期鍵加算の効果がなくなる。

この性質は AES と同じことと考えることができるが、後に続く議論は少し異なる。AES の場合とは違い Camellia は、別の鍵加算が各々の非線形段関数に対して処理される。よって、攻撃者は、非線形関数の実際の入力の値を知ることはできない。結果として、単に段数のみをブロック暗号の安全性の指標として考えた場合、初期鍵加算の効果が消えることは、安全性に関して、なんら影響を及ぼすものではない。

しかしながら、Fesitel 構造の性質から、再度等価鍵に対する安全性を検討する必要がある。Feistel 構造では、初期鍵加算、最終鍵加算はすべての拡大鍵に線形な関係で影響しあう。すなわち、これら初期/最終鍵加算は各々の段関数へ移動することができ、拡大鍵と組み合わせることができる [16]。この性質は、特に等価鍵の安全性を論じる場合には極めて重要である。例えば、LOKI89 [4] はすべての鍵に対して 15 個 (自分自身を含めると 16 個) の等価鍵を持つ [16]。しかしながら、これらの LOKI89 に対する等価鍵は. 初期鍵加算が無い場合、自明には存在しない。この LOKI89 の結果から、初期鍵加算自身の存在により、暗号全体で、暗号学的安全性が弱まる/強まるのかは自明ではない。

以下では、Camellia を用いた 2-key XCBC の安全性を議論するために、初期鍵加算のない Camellia の安全性を検討する。特に等価鍵や等価鍵に類似する性質の存在可能性について検討する。議論を簡単にするため、初期鍵加算のない Camellia に対し、別の表記による等価な暗号を考える。この表記では、最終鍵加算は各々の拡大鍵に組み込まれる。結果として得られる新しい拡大鍵を仮鍵 PK_i と呼ぶ。もし、秘密鍵ペア (K, K') が等価鍵などの、特殊な性質を持つ鍵であったとすると、鍵差分値 $\Delta_K = K \oplus K'$ は、各段の仮鍵 PK_i の差分値すべてに影響を及ぼさない、すなわち $Delta_{PK_i} = 0$ となると考えられれる。

ここで Camellia の鍵スケジュール部について、議論する。簡単なため、128 ビット鍵のみについて扱う。同じような解析は 192, 256 ビットの鍵長の鍵スケジュール部にも適用可能である。128 ビット鍵用の Camellia の鍵スケジュールは、秘密鍵 K_L から、非線形な手法で 128 ビットの K_A を生成する。各々の拡大鍵の定義については文献 [1] を参照のこと。仮鍵の定義より、重要な仮鍵のいくつかは以下のように定義される：

$$PK_{13} = (K_L <<<_{128} 94)_L \oplus kw4,$$

$$PK_{17} = (K_L <<<_{128} 111)_L \oplus kw4,$$

$$PK_{14} = (K_L <<<_{128} 94)_R \oplus kw3,$$

$$PK_{18} = (K_L <<<_{128} 111)_R \oplus kw3.$$

ここで、$kw4 = (K_A <<<_{128} 111)_R$ かつ $kw3 = (K_A <<<_{128} 111)_L$ である。

ここで、異なる二つの鍵について特徴的な性質を見付けるために、すべての仮鍵差分 Δ_{PK_i} が 0 となる場合に注目する。このための必要条件は以下のとおりである；鍵加算差分 Δ_{kw4}(または Δ_{kw3}) は、ある固定した鍵のペア (K, K') について、第 13、17 段の拡大鍵差分 (または第 14、18 段) をキャンセルしなければならない。これを式に表すと以下のようになる：

$$\Delta_{K_{13}} = \Delta_{K_{17}} = \Delta_{kw4}, \Delta_{K_{14}} = \Delta_{K_{18}} = \Delta_{kw3}.$$

これらは 1 つの 128 ビットの相関式として記述できる。

$$(\Delta_{K_L} <<<_{128} 94) = (\Delta_{K_L} <<<_{128} 111) = \Delta_{K_A} <<<_{128} 111).$$

ここで、ある固定した鍵ペア (K, K') について、中間鍵差分 Δ_{K_L} と Δ_{K_A} は固定である。二つの K_L 間の巡回シフトのシフト幅は 17 であり、レジスタサイズの 128 と互いに素である。よって、K_L と K_A の差分について唯一満足可能な差分値はビット文字列 1^{128} のみである。

K_A はデータ撹拌部分で用いられる段関数を 4 段分くり返すことで K_L から生成される。通常の差分解読法の観点からは、二つの差分値 Δ_{K_L} と Δ_{K_A} が一致する確率は極めて小さい。このことはほぼ無視できる数の秘密鍵ペアが第 13,14,17,18 段で鍵差分が一致することを示す。このことから、初期鍵加算のない Camellia は、等価鍵やそれに類似する性質をもつような鍵のクラスは存在しないことが言える。

7. まとめ

2-key XCBC モードの安全性について議論した。そして、PRP が証明可能安全な PRP 構築法であるような二つの重要な場合を例示した。そのうちの 1 つ、Even-Mansour 構築法では、効率的な改竄の手法を示した。またもう一方の DESX 構築法を使った場合、DESX の安全性が弱まっていることを示した。これらより 2-key XCBC MAC 生成スキームは、たとえ証明可能安全な暗号学的要素を用いた場合にも、いつも安全であるとは限らないことを示した。

これらの証明可能安全性の矛盾は、秘密鍵という (PRP などの理想的関数を定義できるほどの秘密情報量に対して) 比較的少ない秘密情報による PRP の構築を考えていること、またその限られた秘密情報を PRP の外部で再度用いられていること、の二点に基因するものである。しかし、実用的に用いられる PRP のほとんどはブロック暗号であり秘密鍵の長さは高々数百ビット程度である。よって、最悪の場合、現状のブロック暗号、ならびに PRP の構築法について、2-key XCBC の証明で用いた PRP のモデルとのギャップが無視できない場合がある。

特に秘密鍵による鍵マスクを実際のブロック暗号に対しておこなった場合、その安全性の検証は、上記 PRP 構築法の安全性の検証に比較して、難しくなる。Camellia の場合と同様にいくつかの攻撃手法やブロック暗号の性質については再度の評価が必要な場合がある。秘密鍵による平文のマスクを行う場合、スキームと暗号学的要素の両者の間での詳細な評価が必要である。

文　献

[1] K. Aoki, T. Ichikawa, M. Kanda, M. Matsui, S. Moriai, J. Nakajima, T. Tokita, "Camellia: A 128-Bit Block Cipher Suitable for Multiple Platforms —Design and Analysis," In Selected Areas in Cryptography, 7th Annual International Workshop, SAC 2000, Proceedings, LNCS 2012, Springer-Verlag, 2001.

[2] M. Bellare, A. Desai, E. Jokipii, P. Rogaway, "A Concrete Security Treatment of Symmetric Encryption: Analysis of the DES Modes of Operation," *Proceedings of the 38th Symposium on Foundations of Computer Science, IEEE*, 1997, full paper is available at `http://www-cse.ucsd.edu/users/mihir/`.

[3] M. Bellare, J. Kilian, P. Rogaway, "The Security of Cipher Block Chaining," *Advances in Cryptology, -CRYPTO'94, LNCS Vol. 839, Springer-Verlag*, 1994.

[4] L. Brown, J. Pieprzyk, J. Seberry, "LOKI - A Cryptographic

Primitive for Authentication and Secrecy Applications," *Advances in Cryptology - AUSCRYPT '90*, Springer-Verlag, Lecture Notes in Computer Science Vol. 453, 1990.

[5] J. Black, P. Rogaway, "CBC MACs for arbitrary-length messages: The three-key constructions," *Advances in Cryptology, -CRYPTO2000, LNCS 1880, Springer-Verlag*, 2000.

[6] J. Black, P. Rogaway, "A Block-Cipher Mode of Operation for Parallelizable Message Authentication," *Advances in Cryptology, -EUROCRYPT 2002, LNCS Vol. 2332, Springer- Verlag*, 2002.

[7] A. Biryukov, D. Wagner, "Advanced Slide attacks," *Advances in Cryptology, -EUROCRYPT 2000, LNCS Vol. 1807, Springer-Verlag*, 2000.

[8] J. Daemen, "Limitations of the Even-Mansour construction," *Advances in Cryptology - ASIACRYPT'91, LNCS, Vol. 739,* Springer-Verlag, 1993.

[9] J. Daemen, V. Rijmen, *AES Proposal: Rijndael*, AES Algorithm Submission, September 3, 1999, available at `http://www.nist.gov/CryptoToolkit`.

[10] S. Even, Y. Mansour, "A construction of a cipher from a single pseudorandom permutation," *J of Cryptology, 10(3) 151-161*, Summer 1997.

[11] National Institute of Standards and Technology, Federal Information Processing Standards Publication 46-3, Data Encryption Standard (DES), Reaffirmed 25/10/1999.

[12] National Institute of Standards and Technology, Federal Information Processing Standards Publication 81, DES Modes of Operation (DES), 1980.

[13] National Institute of Standards and Technology, Federal Information Processing Standards Publication 197, Advanced Encryption Standard (AES).

[14] O. Goldreich and S. Goldwasser and S. Micali, "How to Construct Random Functions," *Journal of the ACM*, 33(4), 1986, 792-807.

[15] E. Jaulmes, A. Joux, F. Valette, "On the security of randomized CBC-MAC beyond the birthday paradox limit: a new construction," in the Preproceedings of the Fast Software Encryption 2002, Leuven, Belgium, 2002.

[16] L.R. Knudsen, "Cryptanalysis of LOKI," *Advances in Cryptology– ASIACRYPT '91*, Springer-Verlag, 1993, pp. 22–35.

[17] J. Kilian, P. Rogaway, "How to protect DES against exhaustive search (an analysis of DESX)," *Advances in Cryptology - CRYPTO'96, Lecture Notes in Computer Science, Vol. 1190,* Springer-Verlag, 1996.

[18] M. Luby, C. Rackoff, "How to Construct Pseudorandom Permutations from Pseudorandom Functions," *SIAM J. Comput.*, Vol. 17, No. 2, April 1988.

[19] M. Liskov, R. L. Rivest, "Tweakable Block Ciphers," available at the ePrint archive of iacr web site, `http://eprint.iacr.org/`.

[20] C.J. Mitchell, "The security of two-key DESX," COSIC Seminar, Katholieke Universiteit Leuven, 15th March 2002, Leuven, Belgium.

[21] S. Moriai, H. Imai, "2-Key XCBC: The CBC MAC for Arbitrary-Length Messages by the Two-Key Construction," In the Proc. of SCIS2002, The 2002 Symposium on Cryptography and Information Security, The Institute of Electronics, Information and Communication Engineers, 2002 (in Japanese).

[22] A. Berendschot, B. den Boer, J. Boly, A. Bosselaers, J. Brandt, D. Chaum, I. Damgaard, M. Dichtl, W. Fumy, M. van der Ham, C. Jansen, P. Landrock, B. Preneel, G. Roelofsen, P. de Rooij, J. Vandewalle, *Final Report of Race Integrity Primitives, Lecture Notes in Computer Science, Vol. 1007*, Springer-Verlag, 1995.