ホームネットワークにおけるコンテンツ保護のための暗号化方式

5D－3

相川　慎[1)]　　宝木　和夫[2)]　　古屋　聡一[2)]　　佐々本　学[1)]
(株)日立製作所 デジタルメディア開発本部[1)]　　(株)日立製作所 システム開発研究所[2)]

1. はじめに

近年、家庭内にある情報家電を、デジタル接続することで、機器間制御やデジタルデータの送受信を可能にする、ホームネットワークが注目されている。また、ホームネットワークの伝送媒体としてはIEEE1394 [1] が有力視されている。IEEE1394 は、映画や音楽データを転送するための帯域を確保してリアルタイム転送を可能にする機能を備えており、特にAV 機器同士をデジタル接続するのに最適である。しかしながら、映画などのデジタルコンテンツを、IEEE1394 経由で、そのまま機器間転送してしまうと、不正なデジタルコピーなどの著作権侵害問題が発生する恐れがある。デジタルコンテンツの機器間転送時における不正コピーを防止するためには、機器間認証及び暗号鍵共有後に、送信機器でコンテンツを暗号化し、受信機器で復号化することが考えられる。ここで、情報家電で用いる暗号化方式は、以下の項目を満足していることが重要である。

(a) 低コストで実装できること
(b) 映像データ等をリアルタイム処理できること
(c) 暗号アルゴリズムの輸出が容易であること
(d) 暗号アルゴリズムが十分に安全であること

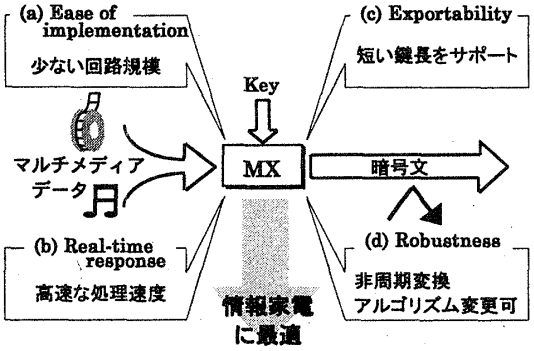


図 1 基本コンセプト

本稿では、上述の条件を満足し、ホームネットワーク上の著作権保護に適した暗号化方式である MX 暗号を提案し、本方式が、情報家電への実装が容易であると共に、十分な処理速度と安全性を実現できることを示す。

2. MX 暗号のアルゴリズム

2.1 全体ブロック図

MX 暗号は共通鍵暗号の一つであるブロック暗号を用いた暗号アルゴリズムである。また、日本でのデジタル衛星放送のスクランブル方式に採用されている、MULTI2 [2]をベースに開発された。図 2 に MX の変換ブロック図を示す。平文と暗号文のブロックの長さは 64 ビットであり、暗号変換では N 個の π 関数 $\{\pi_1,\ldots,\pi_N\}$ を用いる。また、復号変換では、N 個の π^{-1} 関数 $\{\pi_1^{-1},\ldots,\pi_N^{-1}\}$ を用いる。ここで、π_i^{-1} は、π_i の逆変換である。暗号変換では、平文 P の上位 32 ビットを L_0、下位 32 ビットを R_0 として、π_1 から π_N を順に作用することで生成されるデータ $\{L_N, R_N\}$ を結合したのもを、暗号文 C とする。同様に、復号変換では、暗号文 C に π_N^{-1} から π_1^{-1} を順に作用させていくことで元の平文 P を生成する。ここで、各 π 関数における変換をラウンドと呼ぶ。また、π関数の繰り返し数 N は可変であり、これをラウンド数と呼ぶ。

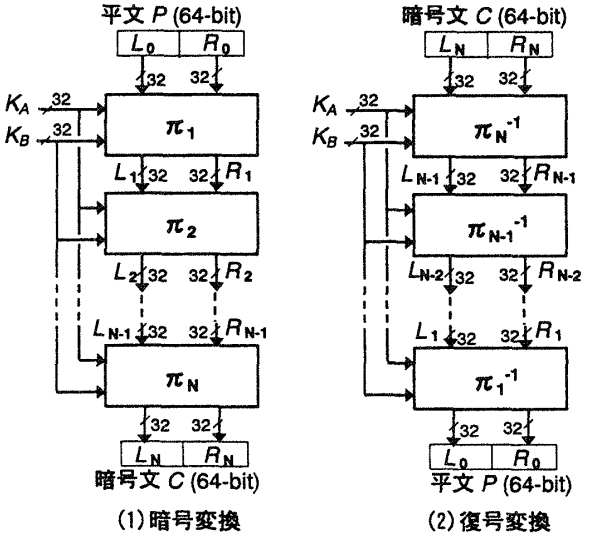


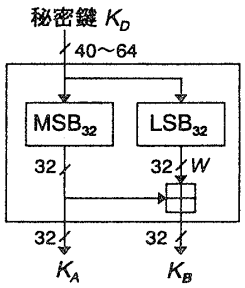
図 2 全体ブロック図

An Encryption Method for Copyright Protection on a Home Network
Makoto Aikawa[1)], Kazuo Takaragi[2)], Soichi Furuya[2)], Manabu Sasamoto[1)]
Digital Media Systems R & D Division, Hitachi, Ltd.[1)] , Systems Development Laboratory, Hitachi, Ltd.[2)]
292 Yoshida-cho, Totsuka-ku, Yokohama 244-0817, Japan

2.2 鍵スケジュール

各π関数およびπ^{-1}関数による変換では、32 ビットの鍵 K_A および K_B を用いる。K_A と K_B は副鍵と呼ばれ、図３に示す変換（鍵スケジュール）を用いて、40～64 ビットの暗号鍵 K_D より、以下の手順で生成される。

(1) K_D の上位 32 ビットを K_A とする。
(2) K_D の下位 32 ビットを W とする。
(3) $K_B = W + K_A \bmod 2^{32}$ とする。



LSB32 ：入力データの下位 32 ビットを出力
MSB32 ：入力データの上位 32 ビットを出力

図 3 鍵スケジュール

2.3 π関数

図４にπ関数およびπ^{-1}関数のブロック図を表す。

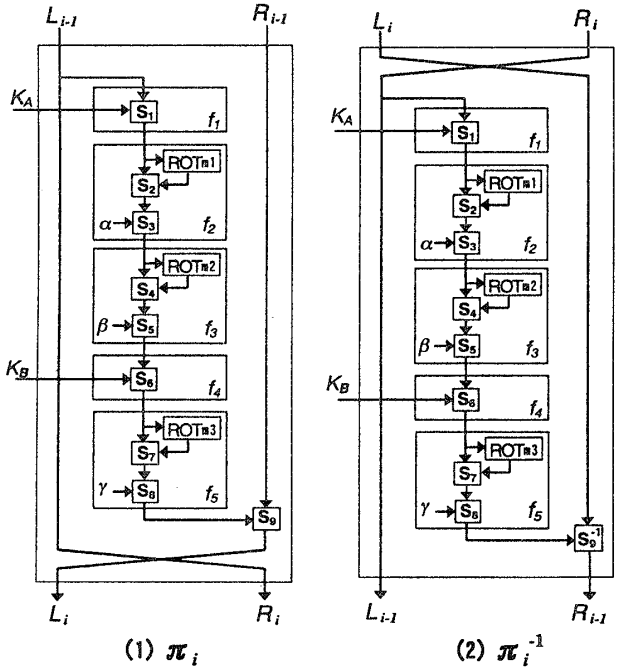


図 4 π 関数およびπ^{-1}関数のブロック図

π関数およびπ^{-1}関数内では、ラウンド関数 f_1 から f_5 を共通に用いる。それぞれのラウンド関数内で行う基本変換は図５に示すとおりである。ここで、記号⊕は排他的論理和をし、記号 $\mathrm{ROT_m}$ は m ビット左循環シフトを表す。また、変換 S_1 から S_9 はそれぞれ加算と排他的論理和のどちらかを行うが、どちらを実行するかは秘密のシステムパラメータとする。また、循環シフトパラメータ m1, m2, m3 および加算パラメータα, β, γも秘密パラメータであり、各ラウンド毎に異ならせることが可能である。これによって、π関数を繰り返し用いていながら、非周期的な変換が実現できる。

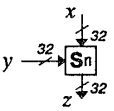
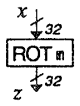
ブロック図	数式
	$z = x + y \bmod 2^{32}$ あるいは、 $z = x \oplus y$
	$z = \mathrm{ROT_m}(x)$ （x を m ビットだけ左循環シフトする）

図 5 ラウンド関数内で用いる基本変換

3. MX 暗号の特徴

3.1 実装規模と処理速度

MX 暗号はシンプルなアルゴリズムを採用しており、ハードウェアおよびソフトウェアのどちらの実装でも小規模で実現できる。これらの主な理由は以下の通りである。

(a) ２つの副鍵 K_A および K_B のみを生成する、簡素な鍵スケジュールを用いている。
(b) 変換テーブルを用いず、単純演算（加算、排他的論理和、循環シフト）のみを用いている。
(c) 各π関数は、循環シフトおよび加算パラメータが異なっているだけなので、π関数の処理回路を各ラウンドで共用することができる。

さらに、MX は高速処理が可能になるように設計されている。表１に MX の実装例を示す。

表 1 MX の実装例

ハードウェア（動作クロック：25MHz）		ソフトウェア（CPU：266MHz 32-bit）	
ゲートサイズ	処理速度	コードサイズ	処理速度
6 K	32 Mbps*	1.5 K byte	200Mbps*

*ラウンド数 = 10 の場合

3.2 輸出の容易性

暗号変換で用いる鍵データのビット長は、輸出規制に大きな影響を与える。これは鍵長が長くなるほど暗号強度が増すからである。MX は 40 ビットから 64 ビットの比較的短い鍵長をサポートしており、輸出規制をクリアしやすい。

3.2 安全性について

暗号文を何らかの手段で入手して、鍵データなしに平文を復元することを解読と呼ぶが、暗号が原理的には解読できたとしても、膨大な計算量を要し、実際上解読不可能であれば、その暗号は安全であると考えられる。最もシンプルな解読方法としては、考えられる鍵データの組み合わせを全て確かめることで、平文を推定する「鍵総当たり攻撃」がある。MX に鍵総当たり攻撃を適用した場合、表 2 に示すだけの計算時間がかかると推定される。

表2　MX の解読時間

鍵長	40 ビット	56 ビット	64 ビット
解読時間	2 日	360 年	90000 年

(使用 CPU : 266MHz 32-bit)

鍵総当たり攻撃は、暗号鍵の長さが短かいほど、短時間で解読が達成できてしまうが、前述したように、鍵長は輸出規制に大きな影響を与えるため、むやみに鍵長を長くするわけにもいかない。その他の解読方法としては、暗号文の統計的な偏りから平文を推定する統計的解読法があり、例えば、差分解読法や線形解読法として知られている。統計的解読法が適用できれば、鍵総当たり攻撃より効率的に暗号を解読できる。MX の場合は、π 関数のパターンが非常に多いため、その選び方によっては統計的解読攻撃に対する安全性が大幅に異なり得るが、π 関数のアルゴリズムパラメータを適切に選ぶことで、統計的解読法に対して十分な安全性を実現できる。ここで、ラウンド数が増えれば MX の暗号強度は増加するが、その分処理速度は低下してしまうため、速度と強度のバランスを考えて、ラウンド数を決定する必要がある。

3.2 MX 暗号のバリエーション

MX 暗号は、各π 関数の加算および循環シフトパラメータを変更することで、容易にアルゴリズムを更新することができる。例えばハードウェアで実装した場合、LSI の制御レジスタの値を変更することで、アルゴリズムが変更できる構成を、容易に実現できる。またソフトウェアの場合は、それぞれのアルゴリズムに対応するソフトウェアモジュールは小規模で実装できるので、これらを置換えることで、アルゴリズム変更が可能である。

4. 適用例

この章では、MX の適用例として著作権保護システムを示し、MX が十分な処理速度と安全性を実現できることを説明していく。

4.1 著作権保護システムの概要

図 6 は著作権保護システムの概要図である。機器 A は送信機器であり、例えばデジタル放送受信機などが考えられる。機器 B は受信機器であり、例えばデジタル録画装置などが考えられる。機器 A と機器 B は IEEE1394 で接続されており、デジタル放送受信機で受信した映画などのコンテンツを、デジタル録画機器に転送して、デジタル録画するなどの用途に用いる。この時 1394 上を流れるコンテンツが不正コピーされないように、暗号処理を行う。

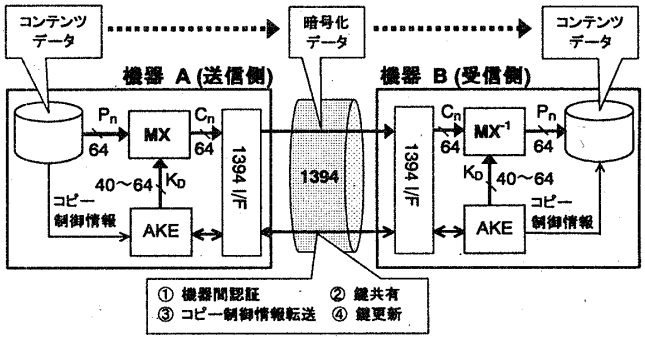


図 6　著作権保護システム

送信側では、コンテンツデータに含まれるコピー制御情報（例えば、"コピー１回可"など）に基づいて、MX による暗号化処理を実行する。受信側では、暗号化データを MX で復号化して元のコンテンツデータを得た後、転送されたコピー制御情報を更新したもの（例えば、"コピー１回可"なら、"コピー不可" に変更する）と共に保存する。この場合、事前にコンテンツデータを暗号化するための暗号鍵 K_D をお互いに共有しておく必要がある。さらに、不正な受信機器は、コンテンツデータを復号化できないようにするためのアクセス制御も必要である。

本システムには、これらの機能を実現するために、機器間認証および鍵共有処理を行う AKE (Authentication and Key Exchange)モジュールが、それぞれの機器に内蔵されている。

4.2 鍵階層メカニズム

鍵の共有処理を実現するには、様々な方法があるが、例えば鍵階層メカニズムを用いることが考えられる。一般に鍵階層メカニズムでは以下に示す鍵データを用いる。

(a) マスタ鍵 (K_M) :最上位層の鍵
(b) 鍵暗号鍵 (K_E) :データ鍵の配送に用いる鍵
(c) データ鍵 (K_D):コンテンツを暗号化する鍵

上位層の鍵は、下位層の鍵を生成あるいは保護するために用いる。マスタ鍵 K_M は最上位層の鍵で、各機器が事前に共有しておくものとする。図7は鍵階層メカニズムの例を示している。暗号通信要求が発生すると、機器 A と機器 B は、乱数データ N とマスタ鍵 K_M からハッシュ関数 H を用いて鍵暗号鍵 K_E を生成する。その後、機器 A はデータ鍵 K_D を生成し、鍵暗号鍵 K_E で暗号化して、機器 B に転送する。機器 B は、鍵暗号鍵 K_E を用いて、データ鍵 K_D を復号する。以上の処理によって、機器 A と機器 B はデータ鍵 K_D を安全に共有することができる。

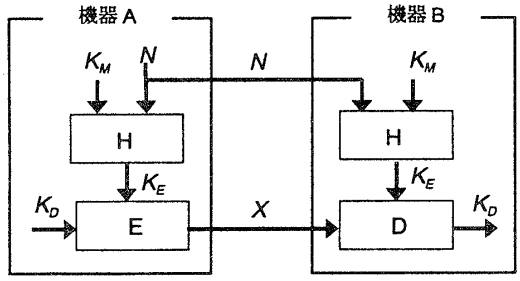


図 7 階層鍵メカニズムの例

4.2 MX 暗号の評価

まず、処理速度に関して評価する。表1で示したように、MX をハードウェアで実装した場合、25MHz のクロック数で 32Mbps のスループットを実現できる。一方で、デジタル放送で用いられている MPEG2 で圧縮された映像データのビットレートは 5Mbps から 30Mbps であるので、小規模および低コストのハードウェアでも、リアルタイム暗号処理が可能である。次に、本システムのセキュリティについて評価する。前述した鍵階層メカニズムを用いれば、送信機器 A は、データ鍵 K_D を定期的に更新して、受信機器 B に転送することが可能になる。新しいデータ鍵 K_D を共有した後に、新しいデータ鍵 K_D でコンテンツデータの暗号化処理を開始すれば、鍵の全数探索による暗号解読をより困難にすることができる。例えば、MX のデータ鍵長が 56 ビットとして、2 時間の映画コンテンツを本システムで暗号変換し、データ鍵 K_D を 2 分毎に更新していく場合を考える。現時点で、266MHz の 32-bit CPU を用いて鍵総当たり攻撃を行うものとし、将来的には CPU パワーが 5 年で 10 倍になると想定すれば、全ての映画コンテンツを解読するのに要する時間は表 3 のようになる（ここでは単一 CPU を用いる場合のみを想定しており、複数の CPU を用いた並列計算による解読攻撃は考えない)。

表3 MX の解読時間 （鍵長：56 ビット）

	現在	10 年後	20 年後
解読時間	21600 年	216 年	2.16 年

MX で用いるデータ鍵は、輸出規制を考慮して比較的短く設定されているが、本適用例のようにデータ鍵を定期的に更新することによって、個人レベルでの鍵全数探索による著作権侵害行為に対して、十分な安全性を実現できるのもと考えられる。

5. まとめ

本稿では、新しい暗号方式である MX 暗号を提案した。MX は構成が簡素であり、低コストで実装でき、高速な暗号処理が可能である。また、MX 暗号を著作権保護システムに適用した場合についても述べた。MX は著作権保護向け暗号方式として十分な安全性と処理速度を備えており、IEEE1394 などを用いたホームネットワークでの著作権保護に最適である。

参考文献

[1] IEEE Standard for a High Performance Serial Bus, IEEE Std. 1394-1995, August 1996.
[2] 宝木和夫, 佐々木良一, " マルチメディア向け高速暗号アルゴリズム Hisecurity-Multi2 の開発と利用方法", WCIS'89, 167-173, August, 1989