コンピュータセキュリティ　14－3
（2001．7．25）

ストリーミング転送における効率的なメッセージ認証方式の検討

田中 俊昭　　中尾 康二　　清本 晋作

（株）ＫＤＤＩ研究所
〒356-8502　埼玉県上福岡市大原 2-1-15，TEL:0492-78-7406
e-mail: tl-tanaka@kddi.com

あらまし　ブロードバンドアクセスの実現に伴い、映像や音声などを高速にダウンロードして利用する形態が普及しつつある。このため、大容量のデータを受信端末で受信しながら、メディアの再生を行なうストリーミング転送の利用が考えられる。上記において、メッセージ認証を行うためには一般的に電子署名が用いられるが、ストリーミング転送のように実時間再生を考慮した場合、署名生成/検証処理負荷等による遅延が発生する可能性がある。また、ＵＤＰ等での信頼性の低いネットワークでの利用を想定して、耐パケットロスを考慮したメカニズムを検討する必要がある。従って、本稿では、これらの課題を解決するため、ストリーミング転送における効率的なメッセージ認証について検討する。

キーワード　メッセージ認証方式、ストリーム転送、ブロードバンド、ハッシュ連鎖、ワンタイム署名

Study on Practical Message Authentication Mechanisms for Digital Streaming Services

Toshiaki Tanaka　　Kouji Nakao　　Shinsaku Kiyomoto

KDDI R & D Laboratories Inc.
2-1-15 Ohara Kamifukuoka-shi Saitama 356-8502, Japan
TEL: +492-78-7406
e-mail: tl-tanaka@kddi.com

Abstract　Download of multimedia contents is an attractive communication service for high speed networks such as the ADSL. Under such au environment, digital streaming protocols are often used. The difficulty to authenticate the digital streaming data is as follows. First, the overhead of signing and verification processes causes a delay of transmission. Second, authentication must be possible even if some packets in the sequence are missing. To solve the above issues, we propose a practical authentication mechanism for digital streaming service. Our mechanism is based on hash chains of authenticating packets, which contains two layered chaining hierarchies. We also study a comparison with the other methods to show that our mechanism is practical and efficient.

key words　Message Authentication, Digital Streaming, Broadband, Hash chaining, One time signature

1. はじめに

ブロードバンドアクセスの実現に伴い、映像や音声などを高速にダウンロードして利用する形態が普及しつつある。このため、大容量のデータをパケット列として受信端末で受信しながら、メディアの再生を行なうストリーミング転送の利用が考えられる。付加価値の高いストリーミング転送においては、その情報の発信元の正当性や改竄の有無を確認するためのメッセージ認証技術が必要となる。メッセージ認証を行うためには一般的に電子署名が用いられるが、ストリーミング転送のように実時間再生を考慮した場合、署名生成/検証処理負荷と実時間検証性との間にトレードオフが存在する。すなわち、従来のファイル転送に添付される署名の場合、全データを受信するまで、そのデータの正当性を検証できない。また、受信端末でパケット毎に署名検証を行うことにより、実時間のメッセージ認証が可能となるが、その処理負荷が大きく、実現性に乏しい。また、ＵＤＰ等での信頼性の低いネットワークでの利用を想定して、耐パケットロスを考慮したメカニズムを検討が必要となる。上記の背景を考慮し、本稿においては、バーストロスおよびランダムロスが想定されるネットワークにおいても、ある一定の耐性をもつ方式について考察し、公開鍵連鎖、階層化されたハッシュ連鎖を用いて、ストリームデータを効率的に認証するメッセージ認証方式を提案する[1]。

本稿の構成としては、２章において関連する研究について、３章において提案方式について詳述し、４章において他方式との比較を述べ、５章において今後の課題を述べる。

2. 背景

ストリーミング転送に関するメッセージ認証については、最もベーシックな手法として、①一般的なデータ転送と同様にコンテンツ情報全体に署名を行う手法、②送信されるパケット毎に送信者のディジタル署名を付与し、メッセージ認証機能を実現する手法、が考えられる。しかしながら、前者の手法は、すべてのデータを受信するまで、そのコンテンツの正当性を検証できない。また、後者の場合は、計算量的に非常に負荷が大きく、とくに、多数のクライアントに対して、複数のコンテンツを提供するコンテンツサーバの処理が膨大となることが予想される。このため、効率的かつ実時間にメッセージ認証を行う手法が課題となっていた。この課題に対する解決手法としてGennaro[2]らの研究により初めてストリーミング転送の効率的なメッセージ認証に対する方式が提案された。Gennaro らは、オンライン方式とオフライン方式の２つの方式を提案している。オフライン方式とは、送信者が予め送信すべき全データを保有している場合で、初期パケットに送信者のディジタル署名を施し、以下のパケットは、前パケットとのハッシュ連鎖をとることにより、改ざんを直ちに検出できる手法である。一方、オンライン方式は、送信すべきデータが不定の場合の手法であり、衝突困難な一方向性ハッシュ関数に基づく one-time signature を用い、パケット間でその公開鍵を連鎖することで実現している。しかしながら、両方式では、パケットロスが発生した場合に検証不可能となる。また、後者のオンライン方式では、one-time signature は１ビットごとにコミットするディジタル署名が施されるため、署名情報が大きくなるという欠点をそれぞれ有する。さらに、パケットロスに耐性を有し、効率的なメッセージ認証を行う方式として、パケット間でハッシュ連鎖を用いる方法がいくつか提案されている。Wong[6]らは、Markle のハッシュツリー[7]を用いてハッシュ連鎖をとり、ルートハッシュにディジタル署名を施す手法を、Golle[5]らは、再帰的なハッシュ連鎖を行うことにより、バーストパケットロスに対応する効率的な手法を提案している。また、Perrig[4]らは、パケットに付与するメッセージ認証子 MAC に用いる鍵を、前のパケットでコミットし、次パケットでその鍵を開示することによりデータ完全性を検証する方式を提案している。ここで、上記の各方式について、以下のような課題が解決されていない。

- Wong らの方式は、グループ化するハッシュツリーが増大するにつれ、付加するハッシュ値の数が増大する。
- Golle らの方式は、任意のパケットロスに対応していない。

・Perrigらの方式は、偽造を防止するための制約として、鍵を開示するパケットが送出されるまでに、鍵をコミットするパケットを受信している必要があるなど送受信のタイミング制御が必要となる。

従って、本稿では、これら各方式の課題を踏まえ、効率的なメッセージ認証方式について検討を行う。なお、実装上のセキュリティホールなどによる脅威については、本稿の対象外とする。

3. 提案方式

本稿ではストリーミング転送を以下のように定義し、そのメッセージ認証方式について検討する。

・データは可変長パケットにより転送する。パケットにはシーケンス番号(SEQ)が含まれる。
・ある確率で受信側でパケットロスが生じる。パケットロスは最大 m（個）を想定する。
・ロスしたパケットの再送は行われない。
・送信側、受信側では、それぞれ、一定の受信バッファを有する。

2章での背景に従い、それらの課題を解決するため、上記のストリーム転送を前提とし、以下のセキュリティ要件を満足する方式を検討する。

・ストリーム転送としては、ライブ配信などを考慮し、事前に送信すべき全データを必要としないオンライン型の転送を対象とする。
・マルチキャスト配信への適用性を考慮して、送信側から受信側へ送る情報には、各受信装置の個別情報を含めないこととする。
・不正な相手からのデータ受信を防止する。
・パケットデータの改ざんや、パケット単位のなりすまし（例えば、spoofing 等の攻撃）を防ぐ。
・最大 m 個のパケットロス時にも、メッセージ認証データは影響を受けない。すなわち、パケットロス時にも、認証データは不変とする。
・さまざまなシステムへの利用を考慮し、計算量、メモリ量、通信負荷を可能な限り最小限にとどめる。

3.1 メッセージ認証方式の概要

本稿で提案するメッセージ認証方式の概要を以下に示す。ここで、ストリームは、有限個（e 個）のパケット列 $B_0, B_1, B_2, \ldots, B_i, \ldots B_e$ とする。

・要件のオンライン方式に対応し、かつ、メッセージ認証情報の作成、検証の負荷を軽減するため図1に示すようにメッセージ認証情報を一定パケット間隔で付与することととする。すなわち、n 個おきにメッセージ認証情報付きパケットを生成する。

B_{nj} (j:0≦j≦|e/n|)

また、e は簡単のため、n の倍数と仮定する。

・上記パケット以外のパケットに対しては、その後、引き続いてくるメッセージ認証情報付きパケットで付与されるメッセージ認証情報の対象とする。
・上記のメッセージ認証の対象となるパケットは、ハッシュ連鎖をとることによりパケットロスが生じた際にも改ざん検知を可能とする。

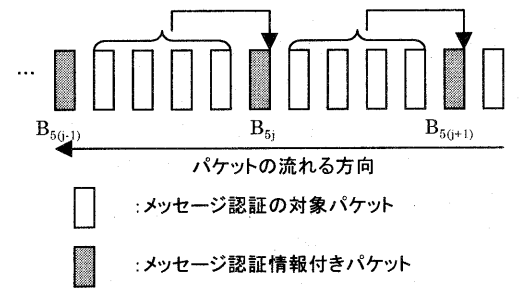

図1 メッセージ認証用パケット列のイメージ図（n=5 の場合）

3.2 メッセージ認証情報付きパケットの処理方式

メッセージ認証の手法としては、否認防止機能を提供するディジタル署名とデータ完全性を保証するメッセージ認証子（ＭＡＣ）の利用が考えられる。

（1） 否認防止パケット

ディジタル署名を一定パケット毎に付与する方式としては、高速化を目的として、one-time signature 方式をベースとし、否認防止パケットのディジタル署名に対応する公開鍵を直前の否認防止パケットのディジタル署名で認証する方法を用いる。すなわち、

$$B_{nj}=\begin{cases}<M_0, Crt_1, s(sk, H(M_0, Crt_1))> & (j:0=j),\\ <M_j, Crt_{j+1}, ots(sk_j, H(M_j, Crt_{j+1}))> & (j:0<j<|e/n|),\\ <M_j, s(sk, H(M_j))> & (j:j=|e/n|),\end{cases}$$

ここで、M_j は j 番目の否認防止パケットの情報、Crt_{j+1} は直後の否認防止パケット署名検証のための公開鍵情報、sk は送信側の秘密鍵、sk_j は j 番目の否認防止パケットに用いる署名の秘密鍵、s(x,y)は、秘密鍵 x を用いた y を対象する署名情報、ots(x,y)は秘密鍵 x を用いた y を対象とする one-time signature、H()はハッシュ関数、とする。

本方式では、最初および最終の署名パケットについては、送信側の本来保有する公開鍵暗号の秘密鍵を用いた署名を行い、その他については、one-time signature を用いる。ただし、Gennaro らの方式は、ディジタル署名のデータ量が多くなるため、一般的な公開鍵暗号を用いることとする。ここで、秘密鍵をパケット毎に選択する処理の効率性を考慮し、秘密鍵が乱数列であるアルゴリズム、例えば、ＤＳＡを用いる。

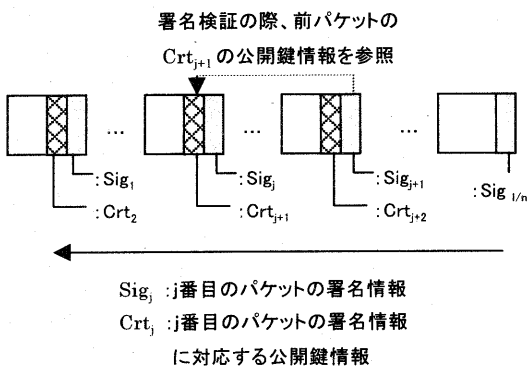

図2 否認防止パケットのイメージ図

（2）ＭＡＣ付きパケット

上述のディジタル署名の代りとして、ＭＡＣを付与することも可能である。この際、ＭＡＣ処理のための秘密鍵(MK)を送受信側で事前に秘密裡に共有していると仮定する。すなわち、

$$B_{nj} = <M_j,\ MAC(k_j,\ M_j)>\ (j:0 \leqq j \leqq |e/n|)$$

ここで、k_j は j 番目のメッセージ認証子付きパケットで使用する秘密鍵、MAC(x,y)は x を秘密鍵とし、y を対象とした MAC 情報とする。K_j はメッセージ認証子パケット毎に変更する。従って、K_j の導出に際しては、MK および、パケットのシーケンス番号を入力としたハッシュ値、すなわち、

$$k_j = H(MK,\ SEQ)$$

とする。

3．3 メッセージ認証対象パケットのハッシュ連鎖方式

前述のように、メッセージ認証情報が付与されないメッセージ認証対象パケットについては、受信側で検証する際に、メッセージ認証対象パケット内データをすべて受信した後に、その対象となるパケット全体のメッセージ認証情報を作成・検証する方法が考えられるが、この場合最大 n 個のパケットをバッファリングする必要があり、効率的でない。従って、あるパケットのハッシュ情報をその後のパケットのメッセージに埋め込むいわゆるハッシュ連鎖方式が考えられる。ハッシュ連鎖方式を用いることによりパケットそのものではなくハッシュ値をバッファリングするだけでよく、送受信側のメモリ利用量の削減が見込まれる。

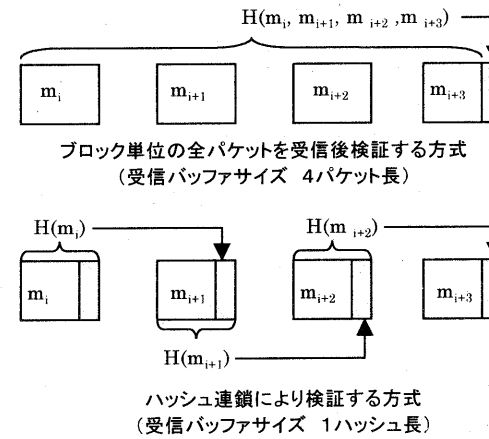

図3 メッセージ認証対象パケットの検証方式

従って、本稿においてもこのハッシュ連鎖を用いたメッセージ認証対象パケットの検証方式を採用する。

本稿では、ＵＤＰのようにパケットロスが発生することを想定しているため、ハッシュ連鎖による方式の場合、パケットロスにより、その連鎖（チェーン）が切断され、検証が不可能となる。一般的にパケットロスのカテゴリとして、

通信の瞬断により発生するバーストロスと、通信に品質に関わるランダムロスが考えられる。バーストロスならばランダムロスという関係にあるので、理想的にはランダムロスに対する耐性を有する方式が望ましいが、後述のように、処理や通信負荷とのトレードオフが存在する。従って、本稿では、パケットロスに対する耐性性能と維持しつつ、効率的なメッセージ認証を行う方式として、バーストロスとランダムロスで異なるしきい値の耐性を有する方式を検討する。ここで、G は全パケット列とし、バーストロスに耐性を有するパケット長を以下のように表す[5]。

$$m = \max\{s \text{ s.t. } \forall\ 1 \leqq s \leqq t\ G-\{B_{i+1}, \ldots, B_{i+s}\} \text{ is fully authenticated}\}$$

また、Sub $\{B_{i+1}, \ldots, B_{i+s}\}$ を要素 $B_{i+1}, \ldots, B_{i+s}$ の任意の部分集合とするとき、ランダムロスに耐性を有する限界パケット長ついては、

$$k = \max\{s \text{ s.t. } \forall\ 1 \leqq s \leqq t\ \forall\ \mathrm{Sub}\{B_{i+1}, \ldots, B_{i+s}\}\ G-\mathrm{Sub}\{B_{i+1}, \ldots, B_{i+s}\} \text{ is fully authenticated}\}$$

ここで、例えば、偶数パケットのみロスする状況が永続的に続くようなパターンは想定しない。すなわち、ランダムロスの発生限界 k はバーストロスの発生限界 m 以下であると仮定する。

$$k \leqq m$$

従って、k パケット以下のランダムパケットロスに対して、かつ m パケット長以下のバーストロスに対して、メッセージ認証可能なメカニズムを検討する。

上記を満足するため、上位パケット群および下位パケット群の２階層のパケット群を用いる。下位パケット群の１ブロックが２つの上位パケットに挟まれるような構成とする(図４参照)。ここで、下位パケット群は上位パケットを越えて、他の上位パケットに挟まれる下位パケット群に連鎖しない制約を与え、上位パケット群の連鎖でバーストパケットロスに耐性を持たせるメカニズムを、下位パケット群の連鎖で任意のパケットロスに耐性を保証するメカニズムを実現する。

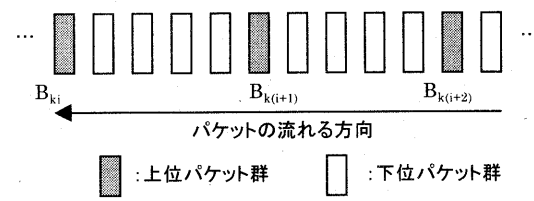

図4　上位パケット群と下位パケット群の関係図

(1)上位パケット群におけるハッシュ連鎖方式

上位パケット間には k-1 個の下位パケット群を含むものとする。上位パケット群の連鎖方式については、パケット B_{ki} のハッシュ値 $h(B_{ki})$ を次の上位パケット $B_{k(i+1)}$ および、ak 個後の上位パケット $B_{k(i+a)}$ のメッセージに埋め込む方式を用いる。すなわち、

$$\text{Hash-chain}\ (B_{ki}) = [C(B_{ki},\ B_{k(i+1)}),\ C(B_{ki}, B_{k(i+a)})]$$

ここで、C(x,y) は x のパケットのハッシュ値を y のパケットに連鎖することを意味する。この場合の最大許容バーストパケットロス長 m は、

$$m = ak-2$$

となる。a が３の場合における上位パケット列の連鎖方式例を図５に示す。

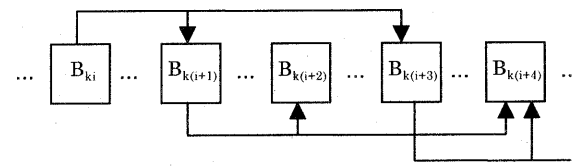

図5　上位層パケット列における連鎖のメカニズム
(a＝3の場合)

(2)下位パケット群におけるハッシュ連鎖方式

下位パケット群のハッシュ連鎖については、最大 k パケットの任意のロスに対して、メッセージ認証が可能となるよう、各下位パケットを隣接する２つの上位パケットにハッシュ連鎖する手法を用いる。すなわち、

$$\text{Hash-chain}(B_{ki+h}) = [C(B_{ki+h}, B_{ki}),\ C(B_{ki+h}, B_{k(i+1)})] \quad (h: 1 \leqq h \leqq k-1)$$

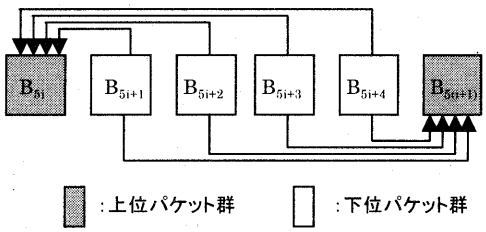

図6 下位パケット列における連鎖のメカニズム（k＝5 の場合）

4. 提案方式の評価

3章で提案した方式について、ここでは、他の方式との比較評価を行う。比較評価のためのパラメタとしては以下のものが考えられる。

Number of packets：
　比較評価を行う際の対象とする連続するパケット数
Total number of signatures：
　署名処理の合計回数
Total number of hash processes：
　ハッシュ処理の合計回数
Average com. overhead：
　1パケットあたりに付加されるメッセージ認証情報のオーバヘッドの平均（単位：バイト長）
Maximum com. overhead：
　1パケットあたりに付加されるメッセージ認証情報の最大オーバヘッド（単位：バイト長）
Permitted packet loss：
　許容パケットロス
Sender packet buffer size：
　送信側のパケットバッファ量
Verification timing：
　検証の即時性（単位：パケット数）

これらの比較パラメタを用いて、Wong および Golle らの方式と比較した結果を表1に示す。ここで、対象とするのは16個のパケット列、すなわち、Number of packets＝16の場合を例にとっている。また、ハッシュ長は160ビット（20バイト）を想定する。本提案(proposed)方式のパラメタとしては、k=8，a=2とする。すなわち、下位パケット列は7パケットで、上位パケットのハッシュ連鎖は、次パケットと次々パケットにリンクする場合を想定する。

以下に、各方式の概略を簡単に説明する。

WL star 方式は、Wong らが提案した1つのルートノードに全パケットのハッシュを連鎖させる方式である（図7a 参照）。WL tree(2level)方式は、2段のハッシュツリーを用いて、検証を行う方式（図7b参照）、WL tree(full binary)はハッシュの2分木を用いる方法である。いずれの手法もすべてのパケット内にルートノードに至るに必要な複数のハッシュ値とルートノードの署名情報が含まれる。一方、Golle の方式は、図8のようなハッシュ連鎖を基本として、内部の1,2のパケットに、あたらに新規1,2のパケットを挿入し、旧1,2のパケットが、A'、B'となるような再帰的に増殖していくハッシュ連鎖のメカニズムである。

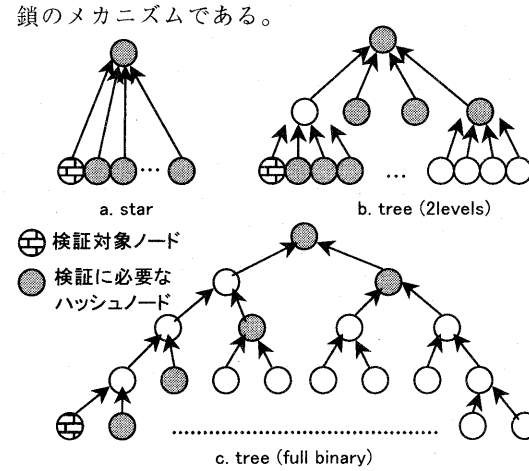

図7 Wong らの方式 イメージ図

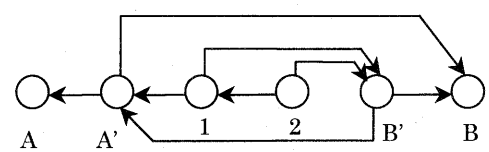

図8 Golle の方式 イメージ図

・通信オーバヘッド
　ここで、通信のオーバヘッドについて、Wong らの方式は、1パケット毎の認証情報が非常に長くなる、さらに、パケット毎に署名情報が必

要であり、署名長がハッシュ長と比較して一般的に大きい(SHA-1:１６０ビットに対して、RSA:１０２４ビット程度)ことを勘案すると、通信オーバヘッドに問題がある。オーバヘッドが小さいことは、限られた帯域のなかで、マルチメディアデータ本来の品質を劣化させずに、メッセージ認証を行う上で非常に重要である。表１の Average com. overhead より、本提案方式は、これらの方式の中で最も通信量のオーバヘッドが小さい方式である。

・処理負荷

処理負荷について、本提案方式はハッシュの処理が各方式のなかで Golle らと同様最も少ない。従って、処理負荷の観点から十分な効率化がなされているといえる。

・パケットロスに対する耐性

パケットロスに対する耐性については、Wong らの方式は、パケット列の任意のロスに対して検証可能である点(Permitted packet loss = any)で優れている。Golle の方式は、バーストパケットのみに対応しており、パケットロスが点在するような状態においては、受信側でメッセージの検証が、バーストロスとしてのみ可能となる。すなわち、点在しているパケットの認証が不可能となる。

これは、映像情報などの符号化がデータの欠落に耐性を有する方式の場合、本来、正常に再生される可能性がある画像情報が、バーストロスと解釈されることによりさらに品質が劣化することになり、望ましくない。一方、提案方式は、最悪の場合でも８パケット長の任意のパケットロスにおいて検証可能となる。最悪の場合とは、上位層パケットが連続して損失する場合のみであり、それ以外の場合においては、任意のパケットロスに対して検証可能となる。さらに、k の値を大きくすることにより、任意のパケットロス耐性の性能を向上させることも可能である(Max(k)=m-1)。

・検証の即時性

Wong らの方式では、１パケット毎に検証が可能である点 (Verification timing =1) で優れている。Golle の方式および本提案方式では、１６パケット毎(Verification timing =16)となる。しかしながら、ストリーム転送においては、受信側でコンテンツを再生し、利用者がコンテンツを確認するため、偽造されたコンテンツを利用者が確認する時間内に検証が可能であれば実用上問題ないといえる。例えば、１Mbps のビットレートで符号化/復号化される画像情報において、1kbyte 程度のパケット長の場合、２０パケット毎に署名を行ったとしても、検出までにかかる遅延は最大、0.2 秒程度であり実用上問題ないと考えられる。

・送信側のパケットバッファサイズ

Golle の方式が最小となっている。本提案方式は Golle の方式より若干サイズが大きくなっているが、これは、ランダムロスを許容するかどうかのトレードオフと考えられる。

・総合

これらの性能比較評価を総合的に判断して、本提案方式は、通信および処理の観点からオーバヘッドが最も少なく、しかも、ランダムなパケットロスに対してある一定以上の耐性を有する効率的な方式といえる。

表１ 各方式の性能比較 (Number of packets = 16)

scheme	Total number of signatures	Total number of hash processes	Average com. overhead	Maximum com. overhead	Permitted packet loss	Sender packet buffer size	Verification timing
WL star	1	17	340	340	16:any	16	1
WL tree (2levels)	1	21	160	160	16:any	16	1
WL tree (full binary)	1	31	120	120	16:any	16	1
Golle	1	16	43	100	16:burst	5	16
Proposed	1	16	39	280	16:burst 8:any	7	16

5. むすび

本稿では、ストリーミング転送における効率的なメッセージ認証方式について、一定周期毎に電子署名を行う手法、パケットロスに対して、ハッシュ連鎖を用いる方法、ハッシュ連鎖に対しては、バーストロスおよびランダムロスに対応するため２階層のハッシュ連鎖を用いる手法を提案した。現在、本提案方式に基づくシステムの実装を進めており、それらの評価を行うこと、受信側が途中から参加する際のメッセージ認証方式についての検討などを進める予定である。最後に、日頃ご指導頂く、㈱ＫＤＤＩ研究所、浅見所長に感謝します。

参考文献

[1] 田中 “ストリーミング転送のメッセージ認証に関する一検討” 情報処理学会第６０回全国大会 Mar. 2000.
[2] R. Gennaro, P. Rohatgi “How to sign digital streams” CRYPTO '97 LNSC1294, Springer-Verlag 1997, pp180-197.
[3] National Institute of Standards and Technology “Digital Signature Standard. NIST FIPS PUB 86, U.S. Department of Commerce, May 1994.
[4] A. Perrig, R. Canetti, D. Song and D. Tygar “Efficient and Secure Source Authentication for Multicast” NDSS2001 pp 35-46, Feb 2001.
[5] P. Golle “Authenticating streamed data in the presence of random packet loss” ECC'99 1999.
[6] C. Wong, S. Lam “Digital Signatures for Flows and Multicasts”, IEEE ICNP'98, 1998.
[7] R. C. Markle “A Certified Digital Signature” CRYPTO'89 1989.