# リアルタイム性を持つストリーミングへの署名方式の提案

宇田 隆哉†, 江藤 秀一‡, 上田 真太郎‡, 川口 信隆‡,
伊藤 雅仁†, 市村 哲†, 田胡 和哉†, 松下 温†

本論文は、映像・音声などのストリーミングメディアを転送する際に、効率よく電子署名を付ける方式の提案である。ストリーミングメディアに対する署名の効率化に関しては、現在いくつかの研究が行われているが、それらは一定量のパケットをグループ化して扱わねばならず、IP 電話のようにリアルタイム性が要求される用途には適していない。本提案の手法では、回線速度、パケット長、端末の演算性能から適用する署名の初期値を定め、パケット転送遅延の揺らぎ、パケットのロス率の変化に対して動的に署名間隔や遅延時間を変更し対応する。本提案は、電子署名のアルゴリズムには非依存であり、一般的な VoIP で採用されている G.729 や G.723.1 を、本提案の方式の上で使用することも可能である。

# A proposal of digital signatures for real-time streaming media

Ryuya Uda†, Shuichi Eto‡, Shintaro Ueda‡, Nobutaka Kawaguchi‡,
Masahito Ito†, Satoshi Ichimura†, Kazuya Tago†, Yutaka Matsushita†

A proposal of digital signatures for real-time streaming media is described in this paper. Some efficient signature algorithms for streaming media are studied. But they are not suitable for the usage which requires real-time transfer such as IP phone because they have to manage some amount of packets as a group. In this method, distance between signatures and delay of time are dynamically changed by delay of packet transfer and by packet loss rate when parameters of signatures are initialized with network speed, packet size and calculation ability of terminals. This method is independent from signature algorithms, and G.729 or G.723.1 which are adopted in general VoIP can be used on this method.

## 1. はじめに

ブロードバンド時代に突入し、高速な有線もしくは無線による安価な常接回線が各家庭にも設置され、ストリーミング形態で映像、音楽の配信が行えるインフラが整ってきている。

ネットワークにおけるストリーミングメディア転送技術は、その特徴からリアルタイム性に関して2つに大別できる。映画や音楽の配信などに用いられるリアルタイム性をほとんど要求しない放送型の転送と、インタラクションを有するためにリアルタイム性が要求される転送である。

† 東京工科大学 Tokyo University of Technology
‡ 慶應義塾大学 Keio University

中でも、IP 電話はリアルタイム性が厳しく[8]、遅延による影響を極力抑える必要が生じる。

リアルタイムにストリーミングメディアに署名を適用するには、基本的には全てのパケットに対して署名を施せばよい。現在、電子署名は紙面の文書に対して物理的な印鑑で捺印したものと同等の信頼性を持つことが法的に認められるようになった。2001年4月1日より施行されている「電子署名及び認証業務に関する法律」[9]において、電子署名の暗号に求められる安全性は、次のような性質と定義されている。

- 電子署名を行なう符号の署名鍵が解読されないこと
- 署名鍵を誤認識せず、正しく特定できること
- 署名鍵を解読せずに署名文を偽造できないこと
- 署名文の改変が検出できること

現時点で、以上の条件を満たしているとされる電子署名方式は下記の 4 方式であると考えられている。

・鍵長が 1024bit 以上の RSA 方式で
・鍵長が 1024bit 以上の ESIGN 方式
・鍵長が 160bit 以上の ECDSA 方式
・鍵長が 1024bit 以上の DSA 方式

本提案の署名方式では、署名に使用する暗号アルゴリズムは特に限定はしないが、強度の高い公開鍵暗号演算は非常に処理が重く、計算負荷が高い。そこで、本論文では、強固な公開鍵暗号署名をリアルタイム性が必要なストリーミングメディアに対して適用しながらも、状況に応じて演算負荷を効率よく減少させることが可能な署名方式を提案する。本提案は、具体的には IP 電話への適用を想定している。IP 電話の会話内容に公開鍵署名を施すことができれば、電話での会話を通じて信頼のおける取引を行うことも可能となる。現在、認証のためのサーバを介しての音声記録公証システム[10]は開発されており、今後、重要な技術となっていくと考えられる。

## 2. 関連研究

ストリーミングメディアの認証時に効率化を図る技術については、現在いくつかの提案がなされている。一般的に認証と呼ばれるセキュリティ技術は、詳細には以下の 3 つに区分される。

・認証(Authentication)
　ある情報が本物であるかどうかを確かめる手段
・機密性(Confidentiality)
　送信データが他人から見えないことの保証
・完全性(Integrity)
　送信データが途中で改ざんされてないことの保証

認証に関しては、Gennaro らの研究[1]が早くから行われている。Gennaro らの Chain 方式では、各パケットが 1 つ後ろのパケットのハッシュ値を持つようになっている。ただし、これでは全てのパケットが揃わない限り送信側で計算が行えないため、リアルタイム送信をする際には一定の範囲でパケットを区切って署名を行っている。このタイプの方式の欠点として、パケットロスに対する耐性がないことが挙げられる。通常、ストリーミングメディア転送はリアルタイム性を重視するために UDP を使って送信される。そして、規定の時間内に到着しなかったパケットは全てロスパケットとして扱われる。Chain 方式では、パケットロスによって署名が連続しない部分が出来てしまうと認証が続かなくなってしまう。

Wong ら[4]の方式では、署名を star 型もしくは tree 型の構造にし、ハッシュ計算を多用することで署名の効率を上げている。これらの方式は非常に効率よく署名を行うことができるが、送信時にバッファリングしなくてはならない時間が長くなってしまう。

Golle ら[3]の方式は、パケットのバーストロス耐性の効率化を図る手法であるが、ランダムロスに対応していないのが弱点といえる。

Perrig ら[2]の方式は、メッセージ認証子 MAC を用いて認証を行うが、これはパケットの完全性を保証するものであり、第三者による改竄の有無は検出できるが、受信者が送信者のメッセージの正当性を第三者に対して証明する用途には使用できない。

最新のものでは KDDI の田中ら[5]の署名方式もあるが、これらは全て、パケットを一定数バッファリングして処理することで効率化を図ることを前提としており、1 パケットに含まれる音声や映像の時間が長い場合は全体の遅延を短くするのが困難となる。

そこで、本研究では、上記をふまえて、リアルタイム性を重視し、遅延時間の許容範囲内で署名演算負荷の効率化を図れる署名方式を提案する。

## 3. リアルタイム性の維持に適した署名方式の提案

本研究では、IP 電話などの音声通話に使用するための、リアルタイム性を重視したストリーミングメディアの署名方式を提案する。

### 3.1 署名方式

本方式では、以下のようにストリーミングメディアの各パケットに対して、図 1 の方法で署名を施す。

P はパケットを表す。P(i)は i 番目のパケットであることを示す。V は、各パケットに含まれる音声データ部である。V(i)は i 番目のパケットに含まれる音声データであることを示す。V のハッシュを計算したものが Hv である。Hv(i)ならば V(i)のハッシュということになる。

$$Hv(i) = Hash(V(i))$$

Ha は、音声データ V と音声データのハッシュ値 Hv を含んだデータを合わせてハッシュ計算を行った値である。例えば、図 1 で V(i)には Hv(i+1)、Hv(i+2)、Hv(i+3)の 3 つの Hv が付随しているとする。このとき、Ha(i)は、これら 3 つの Hv を連結（concatenation）した値のハッシュ値であり、

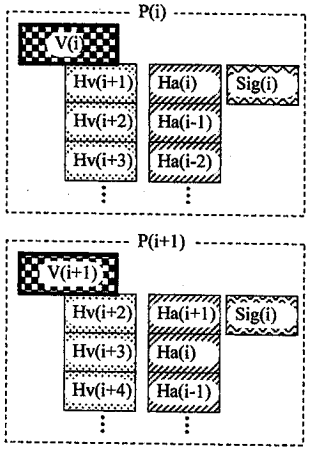


図 1 署名方式

$Ha(i) = Hash(Hv(i+1) + Hv(i+2) + Hv(i+3))$

となる。各パケットに付随する Hv の数と Ha の数は一定ではなく、署名間隔に応じて動的に変化する。署名間隔の変化については後述する。Sig は、パケットの電子署名である。本研究では安全性の観点から署名には公開鍵暗号を用いることを想定しているが、本方式においては署名アルゴリズムについて限定されることはない。Sig(i)は、具体的には i 番目のパケットに付随する Ha の署名である。例えば、図 1 で V(i)には Ha(i)、Ha(i-1)、Ha(i-2)の 3 つの Ha が付随しているとする。このとき、Sig(i)は、これら 3 つの Ha を連結（concatenation）した以下に示すハッシュ値であり、

$Hash(Ha(i) + Ha(i-1) + Ha(i-2))$

これを公開鍵暗号の秘密鍵で暗号化したものとなる。なお、図 1 において、P(i+1)中の Sig は、i+1 番目のものではなく i 番目のものとなっているが、これは P(i)のパケット中にある Sig(i)を複写したものである。全てのパケットで署名の計算を行わずに、演算の効率化を図るのが本研究の目的であり、署名間隔のアルゴリズムについては後述する。以上の V、Hv、Ha、Sig をまとめて、ひとつのパケット P の中に格納する。実際はこれに、TCP などのヘッダがついた状態のパケットがネットワーク内を流れることとなる。

### 3.2 署名間隔と遅延

本提案の方式では、図 1 の Sig で表される署名間隔は、端末の演算負荷を軽減するために動的に変化する。

署名間隔を $\delta$ パケットとおくと、本方式では Hv 署名数は $\delta$-1、Ha 署名数は $\delta$ となる。よって、署名 Sig は以下の式で表される。

$$Sig(i) = Enc(Ks, Hash(\sum_{j=i-\delta+1}^{i} Ha(j)))$$

ここで、

*Enc(KEY, DATA)*

は、KEY の鍵を使って DATA に暗号化処理を施すことを示す。Ks は、公開鍵暗号の秘密鍵である。Enc(Ks, DATA)で、いわゆる公開鍵署名を表すこととする。3.1 節でも述べたが、本提案では署名に使用する公開鍵暗号アルゴリズム、鍵長は特定せず、実装によるものとする。

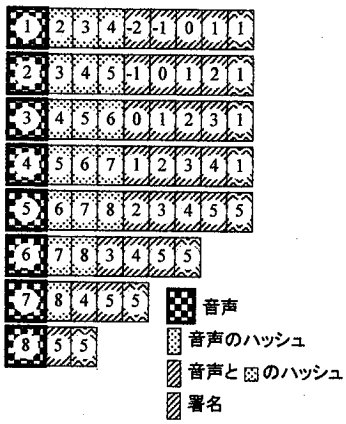


図 2 署名例

図 2 は、各音声データに具体的にハッシュ値と署名を付加し、パケット単位でまとめたものである。図をわかりやすくするため便宜的にパケットには具体的な番号を記している。図 2 では、署名間隔 $\delta$=4 の場合であり、よって各パケットに Hv は $\delta$-1 で 3 個、Ha は $\delta$ で 4 個付随している。図は 8 番目までの音声データ V が生成された送信側の状態を示しており、6 番目のパケットに Hv(9)などが存在しないのは、V(9)が未生成で Hash(V(9))がこの時点では計算できないためである。つまり、送信側では音声データ V が生成されても、それより先の V のハッシュ値 Hv が計算されるのを待たねばパケットを送出することが出来ない。これを送信側遅延 Ds とおく。Ds は付随する Hv の数だけ待つため、

*Ds = $\delta$-1 (パケット個)*

となる。ここで、許容可能遅延を Da とおく。Da は、暗号化や音声の圧縮、セル化などの遅延、またネットワーク通過時の遅延を除き、パケットの送受信タイミングを送らせることのできる限

界の時間である。もちろん、Da が 0 の状態が一番リアルタイム性に優れた通話品質となる。このとき、受信側遅延を Dr とすると、

*Da >= Ds + Dr*

でなければならない。

### 3.3 署名確認とパケットロス

本提案の方式は、公開鍵暗号を用いて電子署名の確認を行うことを前提としている。これにより、セキュリティレベルの高い認証を行うことができ、公開鍵暗号署名を用いない方式と比較して改竄に対する耐性が非常に高くなる。

本提案の方式では、音声データの認証方法は 2 通り存在する。

1 つ目は Hv による認証である。図 2 で、1 番目のパケット P(1)に着目する。1 番目のパケットに付随する Sig(1)は、Ha(-2)、Ha(-1)、Ha(0)、Ha(1)を連結した値に対する公開鍵署名である。ここで、Ha(1)は V(1)、Hv(2)、Hv(3)、Hv(4)を連結した値のハッシュ値であるので、V(1)が正しいことはむろん、Hv(2)、Hv(3)、Hv(4)の元となる V(2)、V(3)、V(4)に関しては、受信時にそのハッシュ値を計算するだけで、正当性の有無が Sig(1)の公開鍵署名によって検証される。このように、送信側でパケットを遅延させることによって、受信時にハッシュ値から音声データの正当性が確認できる。

2 つ目の方法は、Ha による認証である。図 2 で、4 番目のパケット P(4)に着目する。P(4)の音声データ V(4)は、P(1)、P(2)、P(3)が持つ Hv(4)によっても認証可能であるが、これらのパケットが受信時に失われていた場合にも、P(5)が持つ Ha(4)によって正当性の確認を行うことが出来る。P(5)では Ha(4)に対して Sig(5)の公開鍵署名が付随しているため、確認時の暗号強度は公開鍵暗号のものとなる。ただし、P(5)を待って V(4)を確認するためには、受信側で 1 パケット余計に待つ必要が生じる。これが、3.2 節で述べた受信側遅延 Dr である。

以上のように、本提案では各パケット中の音声データ V は、送信側遅延 Ds を許すことによって付随された Hv、もしくは受信側遅延 Dr を許すことによって確認される Ha のいずれかによって公開鍵暗号による署名を常に確認できるものとしている。そのため、署名間隔を δ とすると、各パケットは Hv、Ha のいずれかによって署名の検証が行われなくてはならないため、

*δ <= Ds + Dr*

となる。この説明をふまえた本方式の署名検証における少し複雑な例について解説する。図 2 で 5 番目のパケット P(5)が失われたとき、V(4)は P(1)が持つ Hv(4)によって確認が取れる。V(6)は P(6)が持つ Sig(5)が Ha(2)、Ha(3)、Ha(4)、Ha(5)の署名であり、Ha(3)、Ha(4)、Ha(5)は P(6)が持ち、Ha(2)は P(4)が持っているため、Sig(5)の確認が行える。これにより、Ha(4)の正当性が確認されれば、P(4)中の Hv(6)の正当性も検証可能なため、受信側でシームレスな署名付きの再生が継続可能となる。

### 3.4 署名の動的な変化

まず、署名間隔などの初期値を決定するため、ネットワークの遅延、最大パケットサイズ、端末の演算性能から最低の署名間隔 δ min、最大の署名間隔 δ max（δ が大きくなると Hv、Ha も増加するため、パケット分割が起こらない限界値で δ max を決定する）および許容可能遅延 Da を決定する。Da は、希望する遅延時間（少ない方が高品質となる）から、ネットワークによる伝送遅延時間、音声のセル化による遅延時間、コーデックによる音声圧縮時間、署名などの演算時間を差し引き、決定する。そして Da の時間を、1 パケット辺りに格納される音声データの時間で割り、Ds と Dr の和が最大何パケットまで許容されるのかを算出する。

次に、任意の n パケット間（n=16～20 が適当）で、許容可能遅延時間内でのパケットロス状況を確認する。

パケットロスが全く無いか 1 個の場合、

*Da >= Ds + Dr*

より、Ds が最大値を取るように Ds、Dr を決定する。Ds を大きく取った方が公開鍵署名 Sig の間隔を開けることができ、演算効率が上がるためである。これにより、Hv は Ds と同じ数であるのが最大効率のため Hv が決定し、δ =Hv+1、Ha= δ で署名間隔 δ と Ha 署名数が決定する。

n パケット間でのパケットロスが 2 個以上の場合、パケットロスが連続して起こる場合と、ランダムに抜ける場合についてわけて考える。ランダムロスが起きるのは、非常に不安定なネットワークを使用している場合であり、この場合は署名間隔 δ を小さくして対応する。ロスパケット間の距離が最も短いものの値に δ を指定するが、δ min および δ max の範囲を超えない値とする。連続するパケットロスの場合は、最大何個連続してパケットが抜けたかを調べ、Dr を連続パケットロスの値まで引き上げる。このとき、Dr+Ds は一定のため、署名間隔 δ が小さくなることになる。

ここでδがδminよりも小さくなる場合は、初期値で設定した通信遅延品質ではシームレスな署名が不可能であるため、通信遅延品質を落として署名を継続させるか、署名不能として諦める。

以上の手法を繰り返すことにより、本提案の署名方式では、動的に署名に対する演算負荷を軽減することが可能となる。動的に変化させるタイミングは、受信側からの応答を待つため、最短で、生成された音声データが署名間隔δ分の受信を終えるまでの間隔となる。

本提案は、パケットロスが少ない場合には公開鍵署名を可能な限り減らすことにより端末の処理負荷を下げ、パケットロスの増加に伴い動的に公開鍵署名頻度を増加させ、シームレスな署名付きストリーミングメディアの再生を可能とする。

## 4. 評価

使用するアルゴリズムにもよるが、ハッシュ計算は公開鍵署名の演算に比べて、遙かに高速に処理できる。また、ハッシュ計算は、MD5 で 16 バイト、SHA-1 で 20 バイトと、ハッシュ値をパケットに付加した際のオーバーヘッドも小さい。

本提案の方式は、表 1 に示すように、署名数やハッシュの数は他のどの提案方式よりも多く、署名のためのオーバーヘッドも他の提案方式よりも特によいとは言えないが、送信時にパケットをバッファリングする時間を短くすることが可能なため、リアルタイム性の維持が必要不可欠なストリーミングメディアに適用するには非常に優れていると評価できる。

表 1　署名方式の評価

| Scheme | Signature | hash | Overhead average (bytes) | Overhead max (bytes) | Sender packet buffer size | Verification timing |
|---|---|---|---|---|---|---|
| Chains | 1 | 16 | 43 | 100 | 5 | 16 |
| KDDI | 1 | 16 | 39 | 280 | 7 | 16 |
| WL star | 1 | 17 | 340 | 340 | 16 | 1 |
| WL tree | 1 | 21 | 160 | 160 | 16 | 1 |
| WL tree full | 1 | 31 | 120 | 120 | 16 | 1 |
| Proposed $\delta=1$ | 16 | 16 | 60 | 60 | 0 | 1 |
| Proposed $\delta=2$ | 8 | 40 | 100 | 100 | 1 | 1 |
| Proposed $\delta=3$ | 5 | 37 | 140 | 140 | 2 | 1 |
| Proposed $\delta=4$ | 4 | 36 | 180 | 180 | 3 | 1 |
| Proposed $\delta=5$ | 3 | 35 | 220 | 220 | 4 | 1 |

表 1 で、Sender packet buffer size の値だけ見ると Chains 方式や KDDI の方式も少なく見えるが、これらの方式では署名を確認するために受信側で 16 パケット分待たねばならず、結果として大きな遅延を招いている。具体的には、G.723.1 を例にとった場合 30 ミリ秒毎（秒間 33 パケット）にパケットが送出されるとき、16 パケットの遅延は 480 ミリ秒となる。IP 電話の品質クラスは、ITU-T、ETSI の TIPHONE 及び TIA における IP 電話の品質クラスおよび現行の規定を踏まえ、クラス A では 100 ミリ秒以内、クラス B では 150 ミリ秒以内、最低のクラス C でも 400 ミリ秒以内と定められている[8]ため、認証だけで 480 ミリ秒もの遅延が生じることは致命的と言える。

現在、標準化されている主な VoIP の圧縮率は以下の通りである。

G.711　：64Kbps（非圧縮）
G.726　：32Kbps に圧縮
G.728　：16K に圧縮
G.729　：8Kbps に圧縮
G.723.1　：6.3/5.3Kbps に圧縮

本提案の署名方式では、G.711 のような無圧縮の音声ストリームだけでなく、一般的な VoIP の G.723.1 や G.729 などで秒間 20～100 パケットが送出される場合に対しても、IP 電話の品質を保ちながら署名付きの音声ストリームをシームレスに再生することが可能であり、その上で署名回数を適切に減少させ端末の演算負荷を下げることができる。

また、ADSL などのパケット長が比較的長いネットワーク環境で、ハッシュ値や署名によるオーバーヘッドが増えた場合でもパケット長になお余裕があれば、受信側遅延 Dr の数だけ音声データ V を冗長的に各パケットに持たせることも可能となる。これにより、パケットが欠落しても受信側で再生するまでの遅延時間内に冗長データによって失われた音声データが回復できれば、その間の音声も途切れにくくなる。

## 5. 結論

本論文では、主に IP 電話に使用することを想定し、リアルタイム性を重視したストリーミングメディアの転送時にシームレスに検証可能な電子署名を効率よく施す手法を提案した。これにより、IP 電話を用いてお互いの会話内容を公開鍵署名が施された信頼度で認証し合うことが可能となり、電話での商品注文などの利用に役立つと思われる。将来的に IP 電話が広く普及すれば、IP 電話を行う端末は低消費電力で演算性能の低い専用端末となったり、有線ほど通信品質が安定しない無線の回線を使用したユビキタスなものとなるだろうと推測される。

本提案は、通信品質に応じて動的に署名演算の負荷を変更することが可能なため、様々な状況に対応できる署名方式であると考えられる。

**参考文献**


[1] Rosario Gennaro, Pankaj Rohatgi, "How to Sign Digital Streams", CRYPTO 1997, LNCS 1294, pp.180-197, 1997.

[2] Adrian Perrig, Ran Canetti, Dawn Song, J. D. Tygar, "Efficient and Secure Source Authentication for Multicast", NDSS '01, pp.35-46, 2001.

[3] Philippe Golle, Nagendra Modadugu, "Authenticating streamed data in the presence of random packet loss", Extended Abstract, 2001.

[4] Chung Kei Wong, Simon S. Lam, "Digital Signatures for Flows and Multicasts", IEEE/ACM Transactions on Networking, Vol.7, No.4, pp.502-513, 1999.

[5] 田中俊昭, 中尾康二, 清本晋作, "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", 第14回CSEC研究発表会No.014-003, pp.15-22, 2001.

[6] ITU-T Recommendation G.723.1, Dual Rate Speech Coder for Multimedia Communications Transmitting at 5.3 and 6.3 kbit/s, March 1996.

[7] ITU-T Recommendation G.729, Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP), March 1996.

[8] "IP ネットワーク技術に関する研究会報告書", 総務省, 2002.

[9] "電子署名及び認証業務に関する法律", 総務省情報通信政策局, 2001.

[10] "タイムスタンプ付与型音声記録公証システム", ログイット株式会社ニュースリリース, "http://www.logit.co.jp/news/release_20021129.html", 2002.