4E-4

Merkle Hash Tree と IDA を用いたストリーム認証方式*

新崎 裕隆† 上田 真太郎‡ 金子 伸一郎‡ 荻野 剛‡ 重野 寛†
慶應義塾大学理工学部† 慶應義塾大学院理工学研究科‡

1 はじめに

近年，インターネット環境の普及や，ADSL，光通信の技術向上に伴う通信速度の高速化によって，様々な情報が交わされるようになってきている．そのようなサービスの 1 つにストリーミングが挙げられる．ストリーミングとは，音声情報や動画情報などのマルチメディアデータを視聴する際に，データを受信しながら同時に再生を行う技術を指し，データ再生の際に，受信バッファを除いてデータの保存領域を必要としない．現在では IP 電話やネット会議などのシステムで利用されている．

本稿では，ストリーミング技術を利用してマルチメディアデータの送受信を行う際の汎用的なデータ認証手法を提案する．提案方式では，Merkle Hash Tree[1] を利用して生成した認証情報から署名を生成し，誤り訂正技術の 1 つである IDA[2] を用いて，その署名を各送信パケットに分散する．これにより，既存手法と比べて高い認証率を得ることを狙う．

2 ストリーミングの問題点

ストリーミングを利用した通信ではセキュリティ面から見て，技術的にデータの改ざん，成りすまし，事後否認といったことが可能である．

データの改ざんはストリーミングデータにストリーミングと関係のない任意のコードが挿入されることを指す．これを検出する機能が備わっていない場合，データ受信側でシステムに問題を引き起こす可能性がある．成りすましは，通信に無関係な第三者が通信相手を装って通信を行うことを指す．これによりデータの盗聴，改ざん，漏洩などが起こる可能性がある．また，事後否認は通信を終えた後に，送信者が通信の内容を否定する事を指す．これらはシステムには直接の被害を及ぼさないが，人と人の間での問題に発展する恐れがある．

しかしながら，データの改ざんや成りすまし，事後否認そのものを防止することは困難である．そこで，これらの攻撃が行われたとしても，それに起因する問題を防ぐことができるようにしなければならない．

また，ストリーミング転送では，リアルタイム性を確保するために，UDP などの信頼性の低いプロトコルが用いられる．そのため，データの送信にはパケットロスが生じることがある．

以上のことから，パケットロスに耐性を持ち，かつ高い認証率でストリームデータの認証を行うことができる方式が必要である．

3 提案

提案手法の処理は，認証情報の生成と受信パケットの検証に分けられる．ここに，認証情報とは各パケットを認証するために必要な全てのハッシュ値を指す．認証情報の生成処理では，Merkle Hash Tree と IDA を用いて各送信パケットに付与するデータの生成を行う．

Merkle Hash Tree は複数データの連結ハッシュ値をとるために構成された 2 分木であり，2 つ 1 組のハッシュ値を次々と連結していくことにより，複数データのハッシュ値を 1 つにまとめる手法である．Merkle Hash Tree を使うことにより，署名を行う認証情報を 1 つにまとめ，検証時に各パケットが即時検証可能なように認証情報を生成することができる．

IDA(Information Dispersal Algorithm) は符号理論における誤り訂正技術の 1 つで，元のデータを冗長度を持たせて複数のデータに分割する．分割は，分割されたデータの中で一定個数以上が集まれば元のデータが復元できるように行われる．復元に必要なデータの数は，分割の際に指定が可能である．IDA を使うことにより，各送信パケットに署名を分割して持たせ，ある程度のパケットが失われても署名を復元し，認証を行うことができるようにしている．

3.1 認証情報の生成

認証情報の生成処理の概要を図 1 に示す．提案方式では，ストリーミングデータを複数のブロックに分割し，さらに各ブロックを 2 の乗数 $(= 2^n)$ になるように，複数のデータに分割する．図 1 では，$n = 2$ の場合を示している．次に，分割したデータそれぞれに対するハッシュ値を生成し，それらをリーフとする完全 2 分木を構成する．そして Merkle Hash Tree のアルゴリズムに基づいて，1 つのハッシュ値にまとめる．このハッシュ値はルー

*Stream authentication using Merkle Hash Tree and IDA
†Yasutaka Shinzaki, Hiroshi Shigeno
†Faculty of Science and Technology, Keio University
‡Shintaro Ueda, Shinichiro Kaneko, Takeshi Ogino
‡Graduate school of Science and Technology, Keio University

トハッシュ（図 1 では h_{1-4} に相当）と呼ぶことにする．さらにルートハッシュからディジタル署名を生成し，この署名を IDA を利用して 2^n 個の IDA データとして分割する．(図 1 では $S_1 \cdots S_4$ に相当）この際，データを復元するために必要なデータ数 m が閾値として定められる．各パケットには，そのパケットのグループ内におけるシーケンス番号（$\#i$），ルートハッシュに至るまでに必要な兄弟ノードのハッシュ値（h_2，h_{3-4}など)，IDA による署名の分割データ（S_i）が付与される．ストリームとしては，先頭に署名パケットを単独で送り，続いて先に述べた認証情報を付与したパケットを送信する．

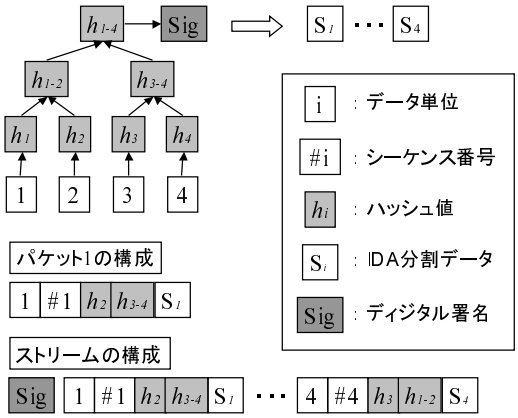

図 1: 認証情報の生成手法 (1 ブロック)

3.2 検証手法

ストリームで送られた署名パケットに含まれるルートハッシュの値と，各パケットのデータおよび付与されたハッシュ値から計算されるルートハッシュの値を比較する．これらの値が等しければ，そのパケットは改ざんが行われず，かつ成りすましも行われていないことが確認できる．署名パケットが受信できている場合は，パケットが到着次第、即時にパケットの検証が可能となる。署名パケットが受信できていない場合は，各パケットに分散させた IDA 分割データから署名を復元することを試みる．ブロックのデータのうち m 個以上のパケットが届いていれば，署名情報を復元することができ，その時点からパケットの検証が可能になる．

4 シミュレーションによる評価

提案方式について、シミュレーションにより評価を行った．評価はパケットロス率と認証率の関係を計測した．シミュレーションにおけるパケットロスモデルは，2-state Marcov Chain Loss Model を利用した．図 2 に，1 ブロックあたり 32 パケット：$n = 5$ の場合を示す．凡例中の数字は IDA のデータ復元閾値 m を表す．図より，提案方式は SAIDA[3] 方式に比べ m の値が大きいほど，認証率において優位であることがわかる．SAIDA 方式は，各パケットからハッシュ値を生成し，それらの連結ハッシュ値と，連結ハッシュ値から生成されるディジタル署名を IDA によって各パケットに分散する方式である．SAIDA 方式では署名情報は IDA 分割データにのみ含まれるのに対して，提案方式では署名情報を署名パケットとして別送し，さらに IDA 分割データとして各パケットに分散させていることが，認証率に影響していると考えられる．

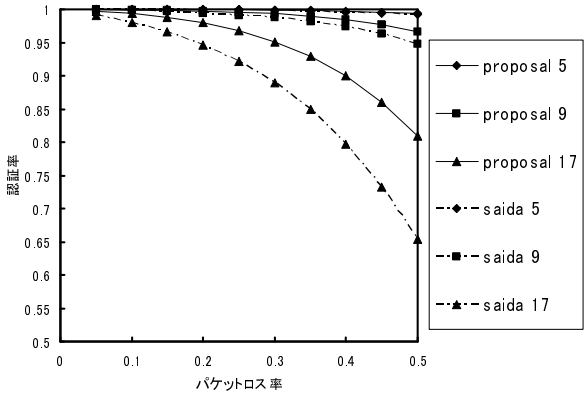

図 2: パケットロス率と認証率の関係

5 おわりに

本稿では，Merkle Hash Tree を利用して生成した認証情報から署名を生成し，その署名を誤り訂正技術の 1 つである IDA を用いて各送信パケットに分散するストリーム認証方式を提案し，提案方式の認証率について評価を行った．今後の課題として，提案方式における情報送信時および受信時の認証処理の遅延の問題を検討していく予定である．

参考文献

[1] R.C.Merkle, “A certified digital signature”, Advances in Cryptology CRYPTO’89, 1989, pp.218-238.

[2] M.Rabin, “Effiient dispersal of information for security, load balancing and fault tolerance” J.ACM 36, 1989, pp.335-348.

[3] J.M.Park, E.K.P.Chong, and H.J.Seigel, “Efficient Multicast Stream Authentication Using Erasure Codes” ACM Trans. Inf. Syst. Security, May 2003, pp.258-285.