Loss-Tolerant Stream Authentication via Configurable Integration of One-Time Signatures and Hash-Graphs

Alwyn Goh[1], G.S. Poh[2], and David C.L. Ngo[3]

[1] Corentix Laboratories, B-19-02 Cameron Towers, Jln 5/58B,
46000 Petaling Jaya, Malaysia
alwyn_goh@yahoo.co.uk
[2] Mimos, Technology Park Malaysia,
57000 Kuala Lumpur, Malaysia
[3] Faculty of Information Science & Technology, Multimedia University,
75450 Melaka, Malaysia

Abstract. We present a stream authentication framework featuring preemptive one-time signatures and reactive hash-graphs, thereby enabling simultaneous realisation of near-online performance and packet-loss tolerance. Stream authentication is executed on packet aggregations at three levels ie: (1) GM chaining of packets within groups, (2) WL star connectivity of GM authenticator nodes within meta-groups, and (3) signature m-chaining between meta-groups. The proposed framework leverages the most attractive functional attributes of the constituent mechanisms ie: (1) immediate verifiability of one-time signatures and WL star nodes, (2) robust loss-tolerance of WL stars, and (3) efficient loss-tolerance of GM chains; while compensating for various structural characteristics ie: (1) high overhead of one-time signatures and WL stars, and (2) loss-intolerance of the GM chain authenticators. The resultant scheme can be operated in various configurations based on: (1) ratio of GM chain to WL star occurence, (2) frequency of one-time signature affixation, and (3) redundancy and spacing of signature-chain.

1 Introduction

Lossy streaming results in the received stream being a subset of the transmitted one; which is problematic from the data authentication viewpoint, especially in comparison to the well-established authentication and verification of block-oriented data. Such signature protocols allow receiver-side establishment of: (1) *absolute* data integrity during transit, and (2) association with specified sender; thereby regarding data loss (even a single bit) in transit as equivalent to fraudulent manipulation. Block-oriented signature protocols are therefore essentially inapplicable on lossy datastreams.

1.1 Overview of Previous Research

Previous research in stream authentication [1-7] has focussed on the integrated use of signatures and hash-chaining, with the latter essentially an amortisation mechanism to compensate for the heavy overheads of the former. This basic concept is established

A . Lioy and D. Mazzocchi (Eds.): CMS 2003, LNCS 2828, pp. 239–251, 2003.

in the seminal work of Gennaro-Rohatgi (GR) [1], who also introduced the notion of *reactive* and *preemptive* mechanisms. The former is applicable when the entire datastream is available a priori to the sender, thereby enabling attachment to every packet of the hash-value of the preceeding packet. Earlier sections of the datastream are hence reactively verified by subsequently recovered packets. This contrasts with the preemptive *one-time* signatures applicable when the datastream is not entirely available a priori. In this case previously recovered one-time public-keys are used to verify subsequent one-time signatures. Note that both formulations as originally presented are *intolerant* of packet-loss.

Reactive authentication was extended by the hash-trees of Wong-Lam (WL) [2], in which the hash-values of child-nodes are aggregated and used for parent-node computation. WL hash-trees require: (1) sender-side buffering of leaf and intermediate hashes, and (2) affixation of highly redundant authenticative information to the data packets; both of which can constitute significant overheads. This formulation does nevertheless enable immediate receiver-side verification and is also maximally loss-tolerant. This tolerance against *worst-case* packet-loss contrasts with the more economical presumption of *random-bursty* loss adopted by Golle-Modadugu (GM) [3]. GM augmented chains are far more efficient to compute than WL trees, but are not loss-tolerant to the same extreme degree.

1.2 Proposed Solution

The major issue in stream authentication is the difficulty of simultaneously enabling: (1) *online* (ie immediate or with minimal delay) functionality, and (2) packet-loss robustness. The GR formulation satisfies (1) but not (2), with subsequent research [2-6] has focussed on incorporation of loss-tolerance via hash-graph topology. Over-emphasis on either of (1) or (2)—as respectively exemplified by the GR and WL/GM approaches—results in stream authentication with a narrow functional emphasis, and therefore limited usefulness.

This paper describes stream authentication in a broad functional context, where online (or near-online) performance and packet-loss tolerance are both important. We outline a composite solution featuring both preemptive and reactive mechanisms, with authentication operations executed at three packet-aggregation layers ie: (1) packet level GM chaining thereby ensuring efficient authentication on the bulk of the datastream, (2) group level WL star connectivity to protect the functionally important chain authenticators, and (3) meta-group level one-time signature [10-12] chaining to establish data-to-sender association.

2 Basic Mechanisms

A digital stream D differs significantly from conventional block-oriented data in several important respects ie: (1) a priori undefined (ie *infinite*) length, (2) online generation in terms of finite L-packet substreams $D_k = \{d_1, \ldots, d_L\} \subset D$, (3) online consumption upon receipt of one or more substreams D'_k, and (4) probable loss of packets during transit so that $D'_k \subseteq D_k$ and $\bigcup_{\forall k} D'_k \subseteq D$. Attributes (2, 3) necessitate high-throughputs for sender-side authentication and receiver-side verification, thereby motivating the use of collision-resistant hash functions H rather than the (significantly slower) number-theoretic constructions. These hashes are used as the fundamental building blocks for the subsequently described H-graphs and sender-irrefutable signatures, the latter of which are functionally equivalent to block-oriented signatures. Such schemes are denoted $\sigma : G_{x,y}, S_x, V_y$ [13] with key-(G)eneration, (S)igning and (V)erification parameterised by key-pair (x, y) of private and public portions; as would be familiar from number-theoretic cryptography.

H-graphs and one-time signatures are respectively reactive and preemptive authentication mechanisms, with computations in the former case necessitating forward-buffering of data. Reactive authentication enables receiver-side verification to be loss-tolerant to a certain degree, but is not genuinely online ie only *nearly* so if the buffer-size is relatively small compared to characteristic stream-lengths. Proactive authentication, in contrast, enables online performance, but requires lossless recovery of the transmitted stream. The inherently dichotomous requirements of online signing/verification (2, 3) and loss-tolerance (4) is an important motivation for the featured research, and will be subsequently discussed in more detail.

2.1 H-Chains

H-graphs result from the conceptualisation of authenticated message packets as vertices and H-based computation as directed edges, the latter of which establishes one-way connectivity among the former. Multi-packet authentication via H-graphs can be (depending on the topology) highly efficient due to overhead amortisation over the multiple packets in a particular graph. This is achieved through appending a particular packet hash (immediately or otherwise) ahead or astern of its location in the packet-graph.

H-graph authentication can also be robust—to some degree, depending again on the topology—against packet-loss, usually at the expense of signing/verification delays arising from the necessity for packet buffering during graph construction. Various buffered H-graph schemes [2-7] are therefore loss-tolerant, while others [1] are genuinely online-computable but loss-intolerant. The simplest H-graph construction is a linear chain on finite-stream D ie:

$$\pi_0 = H(d_1), S(H(d_1)) \text{ and } \pi_i = d_i, H(d_i, H(d_{i+1})) \tag{1}$$

for i ∈ [1, L–1]. The number-theoretic signature on initial packet π_0 is required to establish sender-association, which *bootstraps* the authentication process. There is then the necessity for d_{i+1} prior to computation of π_i, which is characteristic of reactive schemes.

2.2 One-Time Signatures

H- based signatures [10-12] are significantly faster than the corresponding number-theoretic operations, but functionally limited in that a key-pair can only be used to sign and verify a single-message. This results in a linear key-to-data overhead, as opposed the constant overhead of number-theoretic formulations with long-term reusable key-pairs. The signatures themselves also tend to be quite large—ie in the kbit-range for Even-Goldreich-Micali (EGM) [10] formulation—thereby rendering impractical signature affixation on every stream packet.

One-time signatures do not (in contrast to H-graphs) require packet-buffering and can therefore be operated online. The basic operational concept is for the i-th signing S_i and verification V_i components (as parameterised by the i-th key-pair) to be applied on packet $d_i \in D$. Note that one-time schemes also require bootstrapping with a number-theoretic signature on the initial one-time public-key y_0. This certified public-key can subsequently be used for receiver-side verification of a subsequent packet, the logic of which extends down the one-time signature chain as follows:

$$\pi_0 = y_0, S(y_0) \text{ and } \pi_i = d_i, y_i, S_{i-1}(H(d_i, y_i)) \tag{2}$$

This specific formulation is *not* loss-tolerant, and cannot recover from dropped packets. It can, however, be extended via association of *multiple* public-keys to a particular packet π_i. Such a m-time scheme would be able to tolerate lost packets, but at an increased key-to-data overhead.

Note that the signature in the i-th packet (on data and public-keys in π_i) is computed sender-side using the (i–1)-th public-key transmitted in π_{i-1}, which is characteristic of preemptive authentication formulations. This constrasts with the reactive authenticative logic of Eqn 1, where sender-side computation of H-chain node π_i presumes prior availabity of (buffered) π_{i+1}.

2.3 Wong-Lam H-Star

The H-star is the simplest case of the WL hierarchical authentication scheme [2], the basic idea of which is to bind consecutive packets into groups defined by a common signature (number-theoretic or hash-based) on the packet-group. Each packet is subsequently appended with the authenticative information (including the packet-group signature) necessary to confirm membership in the arbitrary n-sized group. This is explicitly designed to enable verification of single packets independent of any other within the same group. The result is an extremely high degree of loss-tolerance, allowing for group-level authentication even if n–1 (out of n) packets are lost during transmission. Formation of WL H-graphs, on the other hand, requires high buffering and communications overheads.

The attributes of packet-loss robustness and high authenticative overhead can be seen from the packet structure ie:

$$\pi_i = d_i, \bigcup_{\forall j \neq i} H(d_j), S\left(H\left(\bigcup_{\forall j} H(d_j)\right)\right) \tag{3}$$

with the root-node S corresponding to the common packet-group signature. Note S in this case denotes the signature-based binding together of all packets within a particular group, rather than a distinct node.

Eqn 3 facilitates packet-group verification via any packet $\pi_i \in \Pi$ within an star-authenticated sub-stream, but at the expense of O(n) hashes and one signature per packet-group. This constitutes an extremely high authenticative overhead per packet, and is also manifestly reactive ie necessitating buffering of $\forall d_i \in D$ prior to computation of any π_i. Packet-group size n is therefore indicative of both packet-loss robustness and authenticative overheads, with small (large) values appropriate for relatively low (high) loss communications environments.

2.4 Golle-Moldagu Augmented H-Chain

GM H-chaining [3] (in common with the WL constructions) is designed to facilitate verification within the context of lossy transmissions, but with a substantively different presumption of packet-loss. Note the packet-level self-similarity in Eqn 3, which renders WL stars robust against worst-case packet-loss, but at the expense of an extremely high authenticative overhead per packet. The GM approach adopts the less strenuous presumption that packets are loss in random-bursts, so that some packets in a n-sized packet-group would be successfully retrieved. There is some evidence [8, 9] that the latter presumption is more realistic, which is fortunate because mitigation of worst-case loss (as addressed by WL stars) should intuitively require heavier overheads than alternative packet-loss models. GM chains in fact enable a significant reduction in the per-packet authenticative overhead, via adoption of: (1) basic chain structure, with far less H-connections compared to the above-discussed WL configuration; and (2) non-uniform distribution of authenticative overheads over the packet-group.

Attribute (2) results in a more complex definition ie:-

$$\pi_i = \begin{cases} \psi_i, H(\psi_i) & i \in [\alpha, n-1] \\ d_n & i = n \qquad \text{with} \\ \psi_\beta, S(H(\psi_\beta)) & i = \beta \end{cases}$$

$$\psi_i = \begin{cases} d_i, H(d_{i+1}) & i = \alpha, n-1 \\ d_i, H(d_{i+1}), H(d_{i+2}) & i \in [1, n-2] \\ d_\beta, H(d_\alpha), H(d_1), H(d_2) & i = \beta \end{cases} \tag{4}$$

for i ∈ {α, 1, …, n, β}, with head node α and tail β. This is justified by the resultant constant communications overhead per packet, as opposed O(n) (ie scaling with group-size) for WL stars. The group-level communications overheads are therefore of

O(n), rather then $O\left(n^2\right)$ for WL stars. Such a reduction is of major practical significance, particularly for operational scenarios with high-volume data transmission and bandwidth-constrained environments.

GM augmented chaining allows for any packet $\pi_i \in \Pi$ to the verified so long as there is a path to the signed π_β, which must be retrieved. This position-dependant authenticative prioritisation is diametrically opposed to the node-level uniformity of WL stars, the latter of which results in loss-tolerance irrespective of position. Note also that GM chain formation at both sending and receiving endpoints requires O(n) packet-buffering, with verification predicated on recovery of the β node. This contrasts with the online verifiability of WL star-connected packets, any of which can be verified independently of all others in the packet-group.

3 Proposed Composite Solution

The above-outlined mechanisms have various attractive attributes ie: (1) structural simplicity of H-chains with single/multiple connections, (2) online performance of m-time signature chaining, (3) maximal packet-loss robustness of WL stars, and (4) efficient packet-loss robustness of GM augmented chains; as illustrated below:

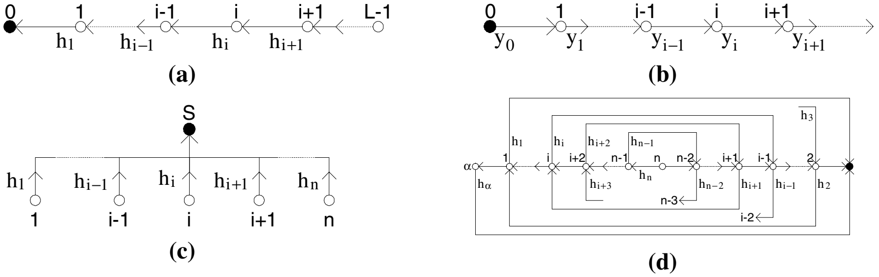

Fig 1. **(a)** Linear H-chain, **(b)** linear one-time signature chain, **(c)** WL H-star, and **(d)** GM augmented H-chain

with the arrows indicative of the authentication direction. This section describes such a framework featuring: (1) GM chain connectivity of individual packets in the datastream, (2) WL star connectivity of the GM β nodes, and (3) m-time signature affixation on the WL groups.

The basic idea is to use GM chaining—with its simultaneous realisation of loss tolerance and structural efficiency—for the bulk of the datastream. This still necessitates recovery of the β nodes, which are then protected strongly via WL star connectivity. What remains is therefore to establish an association between any given WL star-group and the sender identity, which is efficiently done via a chained sequence of H-based signatures. Note this results in a tiered authentication framework addressing (1) packet, (2) packet-group and (3) stream-level data-structures.

3.1 Packet-Level GM Chain-Connectivity

Packets within groups can be characterised as $d_i^k \in D_k$, with (k, i) the respective group and packet indices. GM chain-authenticated packets π_i^k are straightforwardly obtained via Eqn 4, with the only difference being the handling of the β nodes ie:-

$$\beta_k = \psi_\beta^k, H\left(\psi_\beta^k\right) \text{ with } \psi_\beta^k = d_\beta^k, H\left(d_\alpha^k\right), H\left(d_1^k\right), H\left(d_2^k\right) \qquad (5)$$

Note these group-wise *anchor* nodes are not signed as in Eqn 4, but rather used (as subsequently outlined) as leaf nodes within a larger-scale WL star encompassing multiple GM chains. This is denoted by i ∈ {α, 1, …, n, β} and k ∈ {1, …, N} applicable in Eqns 4 and 5. Each data packet in the GM chain then requires a communications overhead of 3 H-words—less for the i ∈ {α, n–1, n} packets—resulting in a total of 3n H-words per packet-group. This is comparable to the overhead of a *single* WL node, hence obvious attraction of GM chains. Computation of H-chain is also relatively efficient if the $H\left(d_i^k\right)$ values in Eqn 4 are buffered, resulting in a total requirement of 2n H-computations per group.

3.2 Group-Level WL Star-Connectivity

Node β_k of Eqn 5 allows verification of the k-th packet-group even if some packets d_i^k (for i ≠ β) are dropped, but can itself be loss in transit. Loss of a particular β packet must therefore be mitigated against, so that the consequences do not extend beyond the relatively small n-sized packet-group. This is addressed in our framework via inter-β WL star-connectivity, with one-time signatures on the root-nodes. The resultant structural form is modified from Eqn 3 ie:

$$B_k^\mu = \beta_k^\mu, y_k, y_{k+\sigma}, \bigcup_{\forall k' \neq k} H\left(\beta_k^\mu\right), \Sigma_{\mu-1} \text{ with}$$

$$\Sigma_{\mu-1} = S_{\mu-1}\left(H\left(H\left(\bigcup_{\forall k} H\left(\beta_k^\mu\right)\right), y_\mu, y_{\mu+\sigma}\right)\right) \qquad (6)$$

for k ∈ {1, …, N}, with μ the meta-group index and σ the inter-group spacing between the affixed one-time public-keys. These star-connected β nodes encompass N packet-groups, and are representative of a meta-group containing Nn data-packets.

The WL star configuration ensures maximal robustness against loss of the group-specific β_k^μ, so that each node can be verified independently of all others. One β node out of the N is therefore sufficient to establish associativity within the larger-scale meta-group context, so long as public-key $y_{\mu-1}$ (necessary for verification of signature $\Sigma_{\mu-1}$) is previously recovered and verified. Note the inclusion of two public-keys per WL leaf in Eqn 6, thereby mitigating against discontinuities in the sequence of one-time public-keys and signatures.

Note the communications overhead of NH + mY + S per packet-group, with key/signature-lengths Y and S further expressible in terms of H-words. The featured

EGM protocol—in common with other one-time signature schemes ie Diffie-Lamport, Merkle and Merkle-Winternitz—has short Y = H key-lengths, but is particularly attractive in that the signature-lengths are both configurable and comparatively short. Typical settings then result in S in the kbit-range ie comparable to commonly encountered number-theoretic implementations. Efficient computation is facilitated—similar to the previously discussed packet-level GM chaining—by buffering of the $H\left(\beta_k^\mu\right)$ values; resulting in an overhead of NH + S per Nn-sized meta-group, with EGM signature-generation S (or verification V, both of which are equal) also configurable.

3.3 Stream-Level m-Time Signatures

The double signature-chaining of the previous section requires equivalently connected initialisation:

$$B_0 = y_0, y_\sigma, S\left(H\left(y_0, y_\sigma\right)\right) \tag{7}$$

with Eqns 6 and 7 essentially a straightforward extension of the linear chaining of [1]. Incorporation of these preemptive signatures facilitate immediate verification—upon recovery of B node as specified in Eqn 6—with multiple connectivity allowing resumption of stream verification σ meta-groups astern of any *completely* dropped meta-group. Characteristic stream-dimension σ is described in [1] as chain-*strength*, in the sense that such signature-chains would tolerate the loss of σ–1 WL stars between B_k^μ and $B_{k'}^{\mu+\sigma}$, as illustrated below:

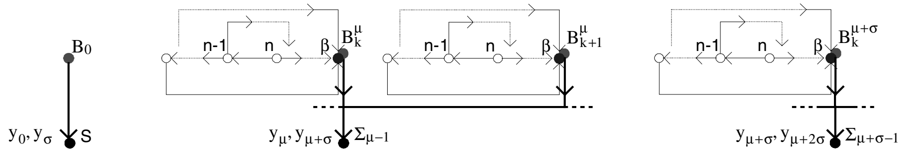

Fig 2. Layered framework featuring packet, group and meta-group mechanisms

This facilitates loss-tolerance between meta-groups; and is therefore complementary to the previously discussed WL star-aggregation of β nodes, which addresses packet-loss within specified meta-groups. The m = 2 configuration does nevertheless result in a doubled key overhead per meta-group compared to Eqn 2, hence our incorporation of the EGM formalism with short H-sized public-keys. EGM (in common with other one-time protocols) also requires sender-side generation of the one-time key-pairs—with one required for each transmitted meta-groups—which can be pre-computed for enhanced operational efficiency.

The framework as outlined features configurable parameters: (1) m public-keys per meta-group, (2) σ signature-chain meta-group spacing, (3) N groups per meta-group, and (4) n data-packets per group. Note the group/packet-level settings (N, n) facilitates more immediate verification compared to a long GM chain of length Nn, in addition to allowing more flexibility in response to different operational conditions.

4 Analysis of Framework

The proposed scheme can be implemented with any number-theoretic and one-time protocol. Our choice of the Rabin [14] and EGM formalisms is based primarily on performance considerations. Rabin verification (necessitating only a single modular-squaring) is, for instance, significantly more computation-efficient compared with other number-theoretic and even H-based formulations [15]. EGM, on the other hand, is significantly more communications-efficient compared with other one-time schemes, but in fact necessitates a higher computation-overhead. We therefore presume the relative preeminence of bandwidth and latency constraints over those related to endpoint computations. EGM signature generation and verification is also more computation-efficient (by up to two orders of magnitude) compared with number-theoretic protocols [15]. The constituent H-operations in EGM can be executed using block ciphers—ie the Data Encryption Standard (DES) as originally suggested—or one-way collision-resistent compressions ie Message Digest (MD) 5 or the Secure Hash Algorithm (SHA). Use of MD5 or SHA hashing results in superior computation-efficiency, and is therefore adopted in our implementation. These hashes are also used to construct the above-discussed WL stars and GM augmented chains.

Practical stream authentication must be both effective and efficient, with evaluation of both dependant on the manner in which packets are dropped in transit. Mechanisms designed to tolerate packet-loss in random-bursts (ie GM chains) can therefore be expected to be significantly more efficient that those designed for worst-case loss (ie WL stars). Our incorporation of both WL stars and GM chains addresses the fact that the β packets of the latter cannot be lost without major functional consequence. The objective is therefore to demonstrate that the featured specification of loss-tolerance effectiveness does not significantly degrade computation and communications efficiency.

Our analysis is presented as follows:

4.1 Correctness

The above-outlined layered scheme can be demonstrated to be correct by considering system-wide compromise to be equivalent to compromise of the underlying cryptography protocols ie the signatures and H-graphs. We follow the GR methodology [1], in which presumption of secure signatures—thereby establishing the initial signed packet—is subsequently extended down a linear H-chains. This established security of linear H-chaining is based on random oracles, and is therefore itself extensible to non-linear constructions (ie the proposed layer framework) via demonstration that node-level compromise is as difficult as the equivalent effort on the underlying one-time signature and H-function. The proposed framework is therefore as secure as the underlying mechanisms ie: (1) number-theoretic signature, (2) one-time signature, (3) H-graphs and (4) H-function.

4.2 Signing and Verification Delay

Delay Ω is defined as the number of packets which must be buffered prior to signing or verification operations. These operations should ideally be executed on a particular packet without delay ie Ω = 0, which denotes genuine online transmission and consumption. Recall that delay-free operations are rendered impossible by our use of H-graphs as an amortisation mechanism against the high overhead of signature operations. Our scheme results in: (1) Ω(send) = Nn from meta-group level buffering prior to signature generation on WL root-node, and (2) Ω(recv) = n from group-level buffering prior to verification with respect signed β node; the latter of which presumes recovery of the required one-time public-keys.

4.3 Communications Overhead

The communications overhead per packet:-

$$\Omega_i = \begin{cases} S' + mH & i = 0 \\ 2H & i = \alpha, n-1 \\ 3H & i \in [1, n-2] \\ 0 & i = n \\ S + (m+N+3)H & i = \beta \end{cases} \qquad (8)$$

can be surmised from Eqns 4-7, with: (1) S′ the number-theoretic signature-length, (2) S the one-time signature-length, and (3) H the hash-length. We compare the proposed scheme—in three (N, n) configurations, with m = 2 signature-chaining—against other stream authentication protocols over a 20-packet stream. Table 1 presents the delay and communication overheads associated with the various protocols:

Table 1. Buffering delays and communications overheads for 20-packet stream

Scheme	**Ω (send, recv)**	**(i) Ω_i**	**Loss**
GR	0, 0	(0) 128, (i) 10	None
GR (one-time)	0, 0	(0) 128, (i) 146	None
WL star	20, 0	(i) 318	Worst-case
GM chain	20, 20	(α, n–1) 20, (i) 30, (n) 0, **(β) 158**	Random
Proposed scheme (4, 5)	20, 5	**(0) 148**, (α, n–1) 20, (i) 30, (n) 0, **(β) 226**	Random
Proposed scheme (2, 10)	20, 10	**(0) 148**, (α, n–1) 20, (i) 30, (n) 0, **(β) 206**	Random
Proposed scheme (1, 20)	20, 20	**(0) 148**, (α, n–1) 20, (i) 30, (n) 0, **(β) 196**	Random

given 10-byte H-words, 128-byte number-theoretc signatures and 136-byte one-time signatures. These hashes are relatively short compared to those used on blocked data, but are sufficient in the context of interest ie to ensure *target* collision-resistance with respect a fixed message, rather than a more generalised *anti* collision-resistance.

Note the lessened verification delay compared to unassisted GM chaining, while retaining the general efficiency of the augmented chain structure. This configurable reduction of the receiver-side delay is attained while simultaneously incorporating (random) loss-tolerance, the latter of which cannot be addressed by the GR and GR one-time formulations. The trade-off between Ω(recv) and Ω_β is also interesting, as is the significantly lessened communications overhead compared with the unassisted WL star configuration. Our scheme can therefore be said to possess functional advantages compared with previously reported formulations.

4.4 Computation Overhead

The signing overhead can also expressed in terms of the constituent operations ie: (1) S′ computations on a stream basis, (2) S computations on a meta-group basis, and (3) H computations on a group/packet basis. We presume the necessity of only a single S′ per streaming session—thereafter represented as [S′] to denote amortisation over multiple meta-groups—and also prior generation of the one-time key-pair sequence. Each meta-group then requires N(n+2) H-computations to account for all the data packets, with N(n+1) required for GM chain construction; as can be seen from Eqns 4 and 5. Meta-group formation also requires Ω(WL) = S + (N+2)H association with WL star computation from Eqn 6. Buffering of packet-level hashes as previously discussed allows for significant efficiency gains, thereby allowing analysis in terms of *incremental* overheads during GM chaining ie $\Delta\Omega_i = H$ (for i ≠ n) and $\Delta\Omega_n = 0$. This results in a total meta-group overhead of:

$$\Omega_k = [S'] + S + \big(N(n+2) + 2\big)H \tag{9}$$

associated with sender-side signature generation.

The verification overhead is likewise expressible in terms of: (1) V′ computations on a stream basis, (2) V computations on a meta-group basis, and (3) H computations on a group/packet basis; with [V′] to denote amortisation over multiple meta-groups. Presumption of packet-level hash-buffering then allows for incremental overheads $\Delta\Psi_i = H$ (for i ≠ n), $\Delta\Psi_n = 0$ and ΔΨ(WL) = V + 3H. This results in:

$$\Psi_k = [V'] + V + \big(N(n+1) + 3\big)H \tag{10}$$

associated with receiver-side signature verification. It should be emphasised that H retrieval rather than recomputation is especially significant for (N, n) configurations with relatively sparse WL star connections of long GM chains.

Our framework—once again in three (N, n) configurations—is compared with previously published protocols over multiple μ repetitions of a 20-packet meta-group, resulting in Table 2 ie:

Table 2. Signing and verification overheads for μ repetitions of 20-packet meta-group

Scheme	**Ω(sign)**	**Ψ(verify)**	**Loss**
GR	$S' + 20\mu H$	$V' + 20\mu H$	None
GR (one-time)	$S' + 20\mu S$	$V' + 20\mu V$	None
WL star	$\mu\cdot(S' + 21H)$ $S' + \mu\cdot(S + 21H)$	$\mu\cdot(V' + 40H)$ $V' + \mu\cdot(V + 40H)$	Worst-case
GM chain	$\mu\cdot(S' + 24H)$ $S' + \mu\cdot(S + 24H)$	$\mu\cdot(V' + 19H)$ $V' + \mu\cdot(V + 19H)$	Random
Proposed scheme (4, 5)	$S' + \mu\cdot(S + 30H)$	$V' + \mu\cdot(V + 27H)$	Random
Proposed scheme (2, 10)	$S' + \mu\cdot(S + 26H)$	$V' + \mu\cdot(V + 25H)$	Random
Proposed scheme (1, 20)	$S' + \mu\cdot(S + 24H)$	$V' + \mu\cdot(V + 24H)$	Random

We present the WL star and GM chain overheads in terms of: (1) originally reported two-layer (number-theoretic signatures and H-graphs) frameworks, and (2) modified two-layer framework incorporating one-time signatures; the latter of which is clearly more efficient. Note the progressively higher overheads corresponding to more frequent WL star computations. This nevertheless still results in receiver-side verification significantly more efficient than the unassisted WL star configuration. Comparison with unassisted GM chaining is obviously unfavourable, with the overhead difference interpretable as the cost of ensuring that β node verification is loss-tolerant. As with Table 1, the GR and GR one-time (both of which are loss-intolerant) formulations are the most computation-efficient. The presented framework is by contrast robustly loss-tolerant, while featuring configurable overheads not significantly greater than the optimal GM chain configuration.

5 Conclusions

The presented multi-layer stream authentication framework effectively leverages the desirable characteristics of the underlying mechanisms ie: (1) preemptive one-time signature-chains, (2) robustly loss-tolerant WL stars, and (3) efficiently loss-tolerant GM chains. This enables flexible configurability ie emphasising: (1) buffering or communications efficiency (ref Table 1), and (2) packet-loss tolerance or computation efficiency (ref Table 2); the relative importance of which is variably dependant on the operational scenario. Our integrated framework also compensates for the functional shortcomings of the constituent mechanisms ie: (1) high overhead of WL stars, and (2) inherent loss-sensitivity of the GM β nodes.

Note the relatively heavy authenticative overhead of the β nodes—resulting from signature and WL star related parameter affixation—compared to the bulk data packets. This authenticative non-uniformity is a consequence of the GM chain structure, and is entirely appropriate for various real-life datastreams ie as specified by the Motion Picture Experts Group (MPEG) standard. MPEG streams are, in fact, constituted from functionally distinct data constructs ie: (1) (I)ntra frames which are essentially static Joint Picture Experts Group (JPEG) images, (2) (P)redictive frames derived from previous reference data, and (3) (B)idirectional frames derived from previous/subseqeunt reference data; the first of which is by far the largest. The functional importance and large size of the I-frames allows them to *carry* the β authenticative overhead without resulting in unreasonably low data-to-(data+authenticator) ratios. We look forward to exploring the applicability of the proposed formulation in an upcoming publication.

References

1. R Gennaro & P Rohatgi, “How to Sign Digital Streams”, *Adv in Cryptology – CRYPTO ’97*, Springer-Verlag, Berlin, pp 180–197, 1997.
2. CK Wong & SS Lam, “Digital Signatures for Flows and Multicasts”, *Comp Sc Tech Rep TR-98-15*, U Texas at Austin. Also, in IEEE ICNP ’98, 1998.
3. P Golle & N Modadugu, “Authenticating Streamed Data in the Presence of Random Packet Loss”, *ISOC Network and Distributed System Security Symp*, pp 13–22, 2001.
4. P Rohatgi. “A Compact and Fast Hybrid Signature Scheme for Multicast Packet Authentication and Others Protocols”, *6th ACM Conf on Comp and Comms Security*, pp 93-100, 1999.
5. A Perrig, R Canetti, JD Tygar & D Song, “Efficient Authentication and Signing of Multicast Streams over Lossy Channels”, *IEEE Symp on Security and Privacy*, pp 56–73, 2000.
6. S Miner & J Staddon, “Graph-based Authentication of Digital Streams”, *IEEE Symp on Security and Privacy*, 2001.
7. A Perrig, “The BiBa One-Time Signature and Broadcast Authentication Protocol”, *8th ACM Conf on Comp and Comms Security*, pp 28–37, 2001.
8. V Paxson. “End-to-end Internet Packet Dynamics”, *IEEE/ACM Trans on Networking*, 7, pp 277–292, 1999.
9. M Borella, D Swider, S Uludag & G Brewster, “Internet Packet Loss: Measurement and Implications for End-to-end QoS”, *Intl Conf Parallel Processing*, 1998.
10. S Even, O Goldreich & S Micali, “On-line/Off-line Digital Signatures”, *J Cryptology*, 9(1), pp 35–67, 1996.
11. RC Merkle, “A Digital Signature based on a Conventional Encryption Function”, *Adv in Cryptology—Crypto ’8*7, LNCS **293**, pp 369–378, 1987.
12. RC Merkle, “A Certified Digital Signature”’ *Adv in Cryptology—Crypto ’8*9, LNCS **435**, pp 218–238, 1989.
13. S Goldwasser, S Micali & R Rivest, “A Digital Signature Scheme Secure Against Adaptive Chosen Message Attack”, *Siam J. Comp.*, **17**(2), 281–308, 1988.
14. MO Rabin, “Digital Signatures and Public-Key Functions as Intractable as Factorization”, *Comp Sc Tech Rep MIT/LCS/TR-212*, MIT, 1979.
15. GS Poh. “Loss-Tolerant Stream Authentication Based on One-Time Signatures and Hash-Graphs”, Comp Sc Masters Thesis, University Sains Malaysia.