New linking schemes for digital time-stamping

Ahto Buldas[1] and Peeter Laud[2]

[1] Küberneetika AS; Akadeemia 21, Tallinn, Estonia
[2] Küberneetika AS, Tartu Lab; Lai 36, Tartu, Estonia
{ahtbu,peeter}@cyber.ee

Abstract. Binary Linking Schemes (BLS) for digital time-stamping [3] provide (1) relative temporal authentication to be performed in logarithmic time, and (2) time-certificates of reasonable size, which together with the data published in a widely available medium enable the verifier to establish their relative temporal positions, even if the database held by the Time-Stamping Service (TSS) ceases to exist. We show that the size of a time-certificate $\tau(X)$ of a document X in the scheme presented in [3] is bounded by $4 \cdot \log_2 N$ where k is the output size of the hash function and N is the number of time-stamps issued. We then present a new BLS with $\tau(X) \approx \frac{6}{\log_2 3} \cdot k \cdot \log_2 N$ and prove that the presented scheme is optimal in that sense.

1 Introduction

Time-stamps enable an incredulous verifier to ascertain the date a digital document was created, signed or last modified. Most of the time-stamping systems proposed to date are based on trusted third parties and are, thereby, more or less vulnerable [6]. The key problem today in time-stamping is to reduce the role of trusted third parties. This is necessary for the segregation of duties and liabilities when using time-stamping for non-repudiation in legally valid digital signature schemes.

First steps in this direction were made by Haber and Stornetta who proposed a linear linking scheme [4] in which the time-certificates, issued by the Time-Stamping Service (TSS), are linked together in a one-way manner, such that the verifier, given two time-stamped documents, is able to ascertain which of the two was created earlier. The use of one-way functions significantly reduces the possibilities of the TSS to back-date documents without inverting the hash function or colluding with the clients. The idea was further refined by Pinto and Freitas [10]. According to them, the time-certificate for a document X_n is $\mathrm{sig}_{TSS}(n, X_n, L_n)$, where $L_n = H(X_n, L_{n-1})$. Although the linear linking scheme makes time-stamping more reliable, it increases the complexity of verification because the required number of hash-steps is linear on the number of time-stamps.

Tree-like linking schemes [2, 1, 5] reduce the verification cost significantly. The main idea is to use Merkle authentication trees [7–9] for storing the time-stamp requests received during fixed time-periods, referred to as rounds. The time-stamp $\mathcal{L}_r$ for round r is a cumulative hash of the time-stamp $\mathcal{L}_{r-1}$ for the $(r-1)$-th round and of all the documents submitted to the TSS during the round r, which are organized as an authentication tree. Time-certificate of a fixed document comprises the authentication path from the leaf corresponding to this document to the root. The length of this path is logarithmic in the number of documents time-stamped during the round. Thereby, the TSS has to store only the values $\mathcal{L}_r$. For temporal authentication the verifier needs some of the values $\mathcal{L}_r$ and a time-certificate. The relative temporal order of two documents submitted during the same round can be ascertained only when assuming unconditional trustworthiness of the TSS. This is not a big problem if duration of rounds is small enough. For example, it equals one second in Digital Notary $[1, 5, 11]$ system. However, if the number of time-stamp requests per round is too small the authentication trees cannot be used effectively. Another weakness of this scheme is that the verifying of one-way relationship between the time-stamps for rounds still requires linear number of hash-steps.

In Binary Linking Schemes [3] the linking item L_n is generated by applying a one-way hash function H to the concatenation comprising L_{n-1} and the value of another suitably chosen $L_{f(n)}$, with f being a fixed deterministic function, i.e.

$$L_n = H(n, X_n, L_{n-1}, L_{f(n)}),$$

where X_n is the digest of the n-th time-stamped document. These schemes are motivated by the fact that if f is chosen appropriately, the verification requires logarithmic number of hash-steps.

The structure of this work is as follows. In section 2 we outline some general requirements for time-stamping systems. In section 3 binary linking schemes (BLS) and the relevant notation is introduced. Section 4 describes antimonotone BLSs as a class of schemes that meet the requirements stated in section 2. It also introduces the notion of pass-through distance of a BLS, which is proportional to the size of time-certificates. Section 5 describes a canonical way of decomposing antimonotone BLSs. In section 6 the main result of this paper, concerning the lower bound of pass-through distance of antimonotone BLSs, is proven. Section 7 describes an antimonotone BLS which achieves this bound, and also discusses its implementation.

2 General requirements

A digital data item does not, by itself, comprise the seal of time. Thereby, the temporal relationship $X < Y$ between data items X and Y has to be "modeled" by another relation, either mathematical or organizational. Obviously, mathematical (one-way) relations are more reliable than, for example, the relation: "The TSS said that X is older than Y". Unfortunately, one cannot define a purely mathematical relation that fixes the temporal positions of bit-strings without doing any special-purpose computations and without interaction between the creators of the time-stamped material. Mathematics just does not depend on any physical phenomenon such as time. Thereby, using a third party (the TSS) to avoid redundant broadcast and storage [2] in a time-stamping system seems to be necessary. The key problem today is to reduce the role of trust in time-stamping systems (and also, in digital signature systems).

In an ideal time-stamping scheme each document X has a time-certificate $\tau(X)$ issued by the TSS such that the certificates $\tau(X)$ and $\tau(Y)$ together comprise information enough for establishing the one-way relationship between X and Y. In such a system the TSS is not necessary during the verification procedure. It is proven ([3], Thm.2), however, that such systems do not exist. Either the size of a certificate is unreasonably large (linear on the number of time-stamps) or the verifier has to request additional verifying data from the TSS. In real implementations a reasonable trade-off should be found.

Most of the time-stamping schemes proposed to date are vulnerable in sense that if the database held by the TSS ceases to exist, we are no more able to perform relative temporal authentication. Even if the time-stamps are regularly (say weekly) published in a newspaper, destruction of the database significantly reduces the accuracy – in Digital Notary system from one second to one week. What we really expect from the time-certificates is that:

- if X and Y are "close" enough in time (lie in the same round), their one-way relationship can be established using $\tau(X)$ and $\tau(Y)$;
- if X and Y are not "close" enough in time (lie in different rounds), their one-way relationship can be established using $\tau(X)$, $\tau(Y)$ and data published in the newspaper.

We demonstrate further that binary linking schemes provide these features. We present a new linking scheme and prove that it is optimal in the sense that it guarantees time-certificates of the smallest possible size.

3 Binary Linking Schemes. Notation

By a Binary Linking Scheme (BLS) we mean a directed graph $(G, \leftarrow)$ without cycles such that: (1) for each vertex $v \in G$ the set $\{w \mid w \leftarrow v\}$ contains no more than two vertices; (2) there is a directed path between each pair of vertices.

It is obvious that the vertices of a BLS can be indexed uniquely with consecutive positive integers $1, ..., N = |G|$ such that $v_{n-1} \leftarrow v_n$ for each $1 < n \leq N$ and there is a unique function $f\colon \{2, ..., N\} \longrightarrow \{1, ..., N\}$, further referred to as the linking function of G, such that $v_m \leftarrow v_n$ if and only if $m \in \{n-1, f(n)\}$. The vertices v_1 and v_N are called the first and the last vertex, respectively. Therefore, a binary linking scheme can be defined as a pair (G, f) of a totally ordered set and a linking function.

The set of vertices $[v_m, v_n] := \{v_k | m \leq k \leq n\}$ is called an interval between v_m and v_n. If $m \leq n$, then the minimal length of a directed path between v_m and v_n in the graph G is denoted as $d(v_m, v_n)$ and is referred to as the distance between v_m and v_n. By the diameter $\Delta(G)$ we mean the maximum of the distance function $d(\cdot, \cdot)$, i.e. $\Delta = \max_{1 \leq m \leq n \leq N} d(m, n)$. The number $d_{pt}(G) = \max_{1 \leq n \leq N} d(1, n) + d(n, N)$ is called pass-through distance of G.

Let the shortest paths between v_1 and v_n, and between v_n and v_N be unique. In this case we denote them by $\mathsf{head}(n)$ and $\mathsf{tail}(n)$, respectively. These paths are unique if the underlying scheme is antimonotone.

4 Antimonotone schemes

In binary linking schemes [3] a time-certificate $\tau(X_n)$ for n-th document X_n of the round r comprises the authentication paths from the time-stamp $\mathcal{L}_{r-1}$ for the previous round to the linking item L_n, which is represented by the path $\mathsf{head}(n)$ in the linking graph; and from L_n to the time-stamp $\mathcal{L}_r$ for the current round, represented by $\mathsf{tail}(n)$. It is shown [3] that if the linking function f is antimonotone, i.e.

$$f(n) < m \leq n \qquad \Longrightarrow \qquad f(n) \leq f(m)$$

for arbitrary m and n, then $\mathsf{tail}(m)$ and $\mathsf{head}(n)$ have an intersection point for every L_m and L_n $(m < n)$ which belong to the same round. Therefore, antimonotone linking schemes guarantee that any two time-certificates $\tau(X_m)$ and $\tau(X_n)$ together contain information enough for establishing a one-way relationship between L_m and L_n, which is sufficient to meet the requirements stated above.

Though the size of a time-certificate is logarithmic, it may become significant if the rounds are large. If the average number of documents time-stamped during a round is N and a k-bit hash function is used, the size of a certificate $\tau(X)$ is

$$|\tau(X)| = 2k \cdot d_{pt}$$

where d_{pt} is the pass-through distance of the linking scheme used. In [3] an upper bound $2k \cdot \Delta$ for the size of $\tau(X)$ has been given. However, the pass-through distance may be smaller than the diameter. As we are going to show below, the linking scheme in [3] has $d_{pt} \leq 2 \cdot \log_2 N$, while it has been shown in [3] that $\Delta \approx 3 \cdot \log_2 N$ for this linking scheme. For example, if $N = 10^7$ and $k = 160$ bits then $|\tau(X)| \leq 1.9$ K bytes.

Another thing that has to be kept in mind is the number of hash-steps during verification. It is proportional to the diameter $\Delta = \Delta(G)$. Linking schemes that are optimal for Δ may not be optimal for d_{pt} and *vice versa*. However, the diameter and pass-through distance of an antimonotone scheme are of the same magnitude, i.e.

$$\frac{1}{2}\Delta \leq d_{pt} \leq 2\Delta.$$

The linking schemes used inside the rounds may be optimized for d_{pt} because in this case the storage is more expensive; the schemes used to link the time-stamps for rounds may be optimized for Δ because d_{pt} does not make sense here. In this article we concentrate on finding schemes with minimal pass-through distance.

5 Structure of antimonotone schemes

We define a binary operation $\otimes$, further referred to as product of binary linking schemes. For schemes (G_1, f_1) and (G_2, f_2) the product-scheme $G_1 \otimes G_2$ is a pair (G, f) where $|G| = |G_1| + |G_2|$ and

$$f(i) = \begin{cases} f_1(i), & \text{if } 1 < i \leq |G_1| \\ f_2(i - |G_1| + 1) + |G_1| - 1 & \text{if } |G_1| < i < |G| \\ |G_1| & \text{if } i = |G|. \end{cases}$$

The resulting scheme is depicted in Figure 1. The product operation is an essential tool when studying the structure of antimonotone binary linking schemes because

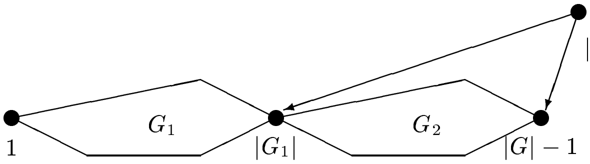

Fig. 1. The product-scheme $G_1 \otimes G_2$

$$(G, f) \cong [v_1, v_{f(|G|)}] \otimes [v_{f(|G|)}, v_{|G|-1}] \tag{1}$$

whenever G is antimonotone. Therefore, all antimonotone schemes can be generated using singleton scheme I and the $\otimes$-operation. Each of them is represented uniquely as an element of the free groupoid $(\langle I \rangle, \otimes)$ with one generator. Whereas the number of vertices in the scheme is equal to the number of I-s in the corresponding non-associative word, the number of antimonotonic schemes with n vertices is equal to the n-th Catalan number

$$C_n = \frac{1}{n}\binom{2n-2}{n-1}.$$

Here and further it has been assumed that all binary linking schemes being spoken about are antimonotone. Let G and H be binary linking schemes. We now concentrate on representing $d_{pt}(G_1 \otimes G_2)$ as a function of $d_{pt}(G_1)$ and $d_{pt}(G_2)$. It turns out that we need additional parameter $D(G) := d(1, |G|)$ for this purpose. We have

$$\begin{aligned} D(G_1 \otimes G_2) &= D(G_1) + 1 \\ d_{pt}(G_1 \otimes G_2) &= \max\{d_{pt}(G_1) + 1, d_{pt}(G_2) + D(G_1) + 1\}. \end{aligned} \tag{2}$$

The linking scheme in [3] can be defined recursively by the equations $T'_1 := I$, $T_n := I \otimes T'_n$ and $T'_{n+1} := T_n \otimes T'_n$. Thereby, it can be proven by mathematical induction that $d_{pt}(T_n) = 2n-1$ and $|T_n| = 2^n$ which gives $d_{pt}(T_n) = 2 \cdot \log_2 |T_n| - 1$. Below, we present a scheme with $d_{pt}(T_n) = \frac{3}{\log_2 3} \log_2 |T_n| + o(\log |T_n|)$ and prove that this bound cannot be tightened.

6 Structure of optimal schemes

As the Catalan numbers C_n are exponential in n, finding schemes with minimal pass-through distance by using brute force is obviously intractable. Decomposition formulas (2) slightly simplify the problem. Let

$$M(n) := \min\{d_{pt}(G) : G \text{ is a BLS with } |G| = n.\}$$
$$M(n,d) := \min\{d_{pt}(G) : G \text{ is a BLS with } |G| = n \text{ and } D(G) = d.\}.$$

Let $\mathcal{M}(n)$ denote the set of all binary linking schemes G with $|G| = n$ and $d_{pt}(G) = M(n)$. Let $\mathcal{M}(n,d)$ denote the set of binary linking schemes with $|G| = n$, $D(G) = d$ and $d_{pt}(G) = M(n,d)$. By (1) each G with $|G| = n$ can be represented as a product $G = G_1 \otimes G_2$, where $|G_1| = \ell$ and $|G_2| = n - \ell$ for suitable $1 \le \ell < n$. As the functions $d_{pt}(G)$ and $D(G)$ are monotone with respect to the arguments $d_{pt}(G_1)$, $d_{pt}(G_2)$ and $D(G_1)$, the minimum of $d_{pt}(G)$ for a fixed $|G| = n$ and $D(G) = d$ can always be obtained by choosing $G_1 \in \mathcal{M}(\ell, d-1)$ and $G_2 \in \mathcal{M}(n-\ell)$. Therefore, the values of $M(n,d)$ can be found by the recursive equations

$$\begin{aligned} M(n,d) &= \min_{d \le \ell < n} \max\{M(\ell, d-1) + 1, M(n-\ell) + d\}, \\ M(n) &= \min_{1 \le d \le n} M(n,d). \end{aligned} \tag{3}$$

n	d: 1	2	3	4	5	6	7	8	9	$M(n)$
2	1									1
3	2	2								2
4	3	3	3							3
5	4	3	4	4						3
6	4	4	4	5	5					4
7	5	4	4	5	6	6				4
8	5	5	5	5	6	7	7			5
9	6	5	5	5	6	7	8	8		5
10	6	5	5	6	6	7	8	9	9	5

Table 1. Values of $M(n,d)$ and $M(n)$ for small schemes.

These equations are valid if we assume that $M(1,0) = 0$ and $M(n,0) = \infty$ for $n > 1$. Let $X(m) := \max\{|G| : G \text{ is a BLS with } d_{pt}(G) = m.\}$. Using formulas (3) it is possible to determine, that

$$\begin{aligned} X(0) = 1, X(1) = 2, \quad X(2) = 3, \quad X(3) = 5, \\ X(4) = 7, X(5) = 11, X(6) = 16, X(7) = 23. \end{aligned} \tag{4}$$

We are going to prove that there exists no sequence of non-isomorphic binary linking schemes $G_1, G_2, ..., G_n, ...$ with $|G_{i+1}| > |G_i|$ for every i, such that

$d_{pt}(G_n) \leq c \cdot \log_2 n + c_0$ for each n, where $c < 3/\log_2 3 \approx 1.89$. We prove that

$$X(m) = 3 \cdot X(m-3) + 1, \tag{5}$$

whenever $m \geq 8$. Let $\mathcal{X}(m)$ denote the set of all binary linking schemes G with pass-through distance $d_{pt}(G) = m$ and with cardinality $|G| = X(m)$. Equation (5) implies that if $G_1, G_2, ..., G_m, ...$ is a sequence with $G_n \in \mathcal{X}(m)$ for each index m, then the corresponding sequence $d_{pd}(G_1), d_{pd}(G_2), ...$ grows approximately as $3/\log_2 3 \cdot \log_2 |G_m|$. For proving (5) we have to know more about the structure of optimal schemes $G \in \mathcal{X}(m)$.

Let $X(m,d) := \max\{|G| : G \text{ is a BLS with } d_{pt}(G) = m \text{ and } D(G) = d\}$ and let $\mathcal{X}(m,d)$ denote the set of all binary linking schemes G with $d_{pt}(G) = m$, $D(G) = d$ and $|G| = X(m,d)$.

Theorem 1. *Each scheme $G \in \mathcal{X}(m,d)$ can be represented as a product $G = G_1 \otimes G_2$ where $G_1 \in \mathcal{X}(m-1, d-1)$ and $G_2 \in \mathcal{X}(m-d)$.*

Proof. Let $G \in \mathcal{X}(m,d)$. By (1) we have that $G = G_1 \otimes G_2$, where $G_2 = [v_{f(|G|)}, v_{|G|-1}]$. By the definition of the operation $\otimes$, each directed path from v_1 to $v_{|G|}$ goes through the vertex $v_{f(|G|)}$. This holds also for the shortest path between these vertices. Therefore, $D(G_1) = d - 1$ and

$$m = d_{pt}(G) = \max\{d_{pt}(G_1) + 1, d_{pt}(G_2) + d\}.$$

If either $d_{pt}(G_1) + 1 < m$ or $d_{pt}(G_2) + d < m$, then G_1 and G_2 respectively could be replaced by larger schemes without changing $d_{pt}(G)$. This follows from the trivial fact that $|H \otimes I| = |H| + 1$ and $d_{pt}(H \otimes I) = d_{pt}(H) + 1$. That, however, would be a contradiction. Thereby, $d_{pt}(G_1) = m - 1$ and $d_{pt}(G_2) = m - d$. The statement of the theorem follows.

An obvious corollary of Theorem 1 is that if $G \in \mathcal{X}(m)$ and $D(G) = d$ then there exist binary linking schemes $G_1, G_2, ..., G_d \in \mathcal{X}(m-d)$ such that

$$G = (\ldots((I \otimes G_1) \otimes G_2) \otimes \ldots) \otimes G_d \tag{6}$$

Corollary 1. *For each positive integer m*

$$X(m) = \max_{1 \leq d \leq m} d \cdot X(m-d) + 1. \tag{7}$$

Proof. Let $G \in \mathcal{X}(m,d)$. By (6) we have $X(m,d) = |G| = d \cdot X(m-d) + 1$. As $X(m) = \max_{1 \leq d \leq m} X(m,d)$, equation (7) follows.

Lemma 1. *If $G \in \mathcal{X}(m)$, then $D(G) < 4$.*

Proof. Let $d = D(G) \geq 4$. By Corollary 1, we have $X(m) \geq 2 \cdot X(m-2) + 1$ and $X(m-2) \geq (d-2) \cdot X(m-d) + 1$. Therefore

$$\begin{aligned} X(m) &\geq 2 \cdot X(m-2) + 1 \geq 2(d-2) \cdot X(m-d) + 3 \geq d \cdot X(m-d) + 3 \\ &> d \cdot X(m-d) + 1, \end{aligned}$$

because $2(d-2) \geq d$.

Lemma 2. *If m is chosen such that for each $d \in \{1, 2, 3\}$*

$$X(m-d-1) \geq \frac{2}{3} X(m-d), \tag{8}$$

then $X(m-1) \geq \frac{2}{3} X(m)$.

Proof. Using Lemma 1 and Corollary 1,

$$\begin{aligned} X(m-1) &= \max_{1 \leq d \leq 3} d \cdot X(m-1-d) + 1 \geq \max_{1 \leq d \leq 3} d \cdot \frac{2}{3} X(m-d) + 1 \\ &\geq \frac{2}{3} \left(\max_{1 \leq d \leq 3} d \cdot X(m-d) + 1 \right) = \frac{2}{3} \cdot X(m). \end{aligned}$$

Lemma 3. *If $m \geq 8$, then for each $d \in \{1, 2, 3\}$*

$$X(m-d) \geq \frac{2}{3} X(m-d+1). \tag{9}$$

Proof. We prove the lemma by mathematical induction with base $m = 8$. If $m = 8$, these inequalities follow immediately from Corollary 1, Lemma 1 and Equations (4). For $m > 8$ induction hypothesis gives us that the inequalities (8) hold and by Lemma 2 we conclude that $X(m-1) \leq 2/3 \cdot X(m)$ which completes the induction step.

Theorem 2. *If $m \geq 8$, then $X(m) = 3X(m-3) + 1$.*

Proof. Let $n \geq 8$. By Lemma 3, we have $3X(m-3) \geq 2X(m-2) \geq X(m-1)$ and thus

$$\begin{aligned} X(m) &= \max_{1 \leq d \leq 3} d \cdot X(m-d) + 1 \\ &= \max\{X(m-1), 2 \cdot X(m-2), 3 \cdot X(m-3)\} + 1 \\ &= 3 \cdot X(m-3) + 1. \end{aligned}$$

Thereby, we have an exact formula for $X(m)$:

$$X(m) = \begin{cases} 1, & \text{if } m = 0; \\ 2, & \text{if } m = 1; \\ 3, & \text{if } m = 2; \\ 5, & \text{if } m = 3; \\ 7, & \text{if } m = 4; \\ \frac{23}{2} \cdot 3^{\frac{m-5}{3}} - \frac{1}{2}, & \text{if } m \geq 5 \text{ and } m \equiv 2 \pmod 3; \\ \frac{33}{2} \cdot 3^{\frac{m-6}{3}} - \frac{1}{2}, & \text{if } m \geq 5 \text{ and } m \equiv 0 \pmod 3; \\ \frac{47}{2} \cdot 3^{\frac{m-7}{3}} - \frac{1}{2}, & \text{if } m \geq 5 \text{ and } m \equiv 1 \pmod 3. \end{cases}$$

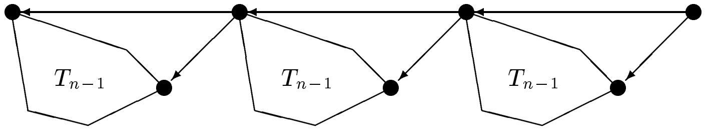

Fig. 2. Recursive construction of T_n.

7 A new linking scheme

We define a new scheme using the recursive procedure and the product operation. Let $T_1 = I$ and

$$T_n := ((I \otimes T_{n-1}) \otimes T_{n-1}) \otimes T_{n-1}.$$

The resulting scheme T_n is depicted in Figure 2. Obviously, the number of vertices in T_n is given by the recursive formula $|T_n| = 3 \cdot |T_{n-1}| + 1$ and $T_0 = 1$. Therefore

$$|T_n| = 1/2 \cdot 3^{n+1} - 1/2.$$

As for the pass-through distance we have again a recursive formula $d_{pt}(T_n) = d_{pt}(T_{n-1}) + 3$ and $d_{pt}(T_0) = 0$, it follows that $d_{pt}(T_n) = 3n$. Hence,

$$\frac{d_{pt}(T_n)}{\log_2 |T_n|} = \frac{3}{\log_2 3} \cdot \frac{1}{1 + o(1)}.$$

Although we have a scheme providing time-certificates of minimal size, it is also important for practical implementations that the values of linking function f can be found in reasonable time. We derive a formula for f for a more general case. Let $k \in \mathbf{N} \setminus \{1\}$ and let T be a BLS. Let $\mathcal{T}(k,T) = ([\mathcal{T}(k,T)]_n)_{n \in \mathbf{N} \cup \{0\}}$ be a family of BLSs defined as follows:

$$\begin{aligned} [\mathcal{T}(k,T)]_0 &:= T \\ [\mathcal{T}(k,T)]_{n+1} &:= ((\cdots (I \otimes \underbrace{[\mathcal{T}(k,T)]_n) \otimes [\mathcal{T}(k,T)]_n) \otimes \cdots) \otimes [\mathcal{T}(k,T)]_n}_{k \text{ times}}, \end{aligned}$$

where $n \in \mathbf{N} \cup \{0\}$. Let $S_n := |[\mathcal{T}(k,T)]_n|$. Thus $S_0 = |T|$ and for every $n \in \mathbf{N} \cup \{0\}$ holds $S_{n+1} = kS_n + 1$. Therefore

$$S_n = k^n |T| + \frac{k^n - 1}{k - 1}.$$

Let $f_n : \{2, \ldots, S_n\} \longrightarrow \{1, \ldots, S_n\}$ be the antimonotone function defining the BLS $[\mathcal{T}(k,T)]_n$. The function f_0 is given by T. For f_n, $n \in \mathbf{N}$, holds (see Fig. 2)

$$f_n(x) = \begin{cases} x - S_{n-1}, & \text{if } x = lS_{n-1} + 1 \\ f_{n-1}(x - (l-1)S_{n-1}) & \\ \qquad + (l-1)S_{n-1}, & \text{if } (l-1)S_{n-1} + 2 \le x \le lS_{n-1}, \end{cases}$$

where $1 \leq l \leq k$. For every $n, n' \in \mathbf{N} \cup \{0\}$ and $x \in \mathbf{N} \backslash \{1\}$ where $n \leq n'$ and $x \leq S_n$ holds $f_n(x) = f_{n'}(x)$ because $[\mathcal{T}(k,T)]_{n'}$ contains $[\mathcal{T}(k,T)]_n$ as initial segment. The function $f : \mathbf{N}\backslash\{1\} \longrightarrow \mathbf{N}$ defined by

$$f(x) := f_n(x), \text{ if } x \leq S_n$$

thus defines the infinite BLS containing each $[\mathcal{T}(k,T)]_n$ as initial segment. The function f can be expressed as follows:

$$f(x) = \begin{cases} x - S_n, & \text{if } x = lS_n + 1, 1 \leq l \leq k \\ f(x - lS_n) + lS_n, & \text{if } lS_n + 2 \leq x \leq (l+1)S_n, 1 \leq l \leq k. \end{cases}$$

Assuming that arithmetic operations take constant time, the complexity of finding $f(m)$ is $O(\log m \cdot \log\log m)$.

8 Conclusions

We presented a new linking scheme for digital time-stamping that does not assume the trustworthiness of the TSS and remains usable even if the database held by the TSS is lost. Time-certificates in this scheme enable to ascertain the relative temporal positions of the documents time-stamped during the same round. Their size is $6/\log_2(3) \cdot k \cdot \log_2 N$, where k is the output length of the used hash function and N is the number of documents time-stamped during a round. We proved that using antimonotone linking schemes it is not possible to achieve smaller certificates.

References

1. D.Bayer, S.Haber, W.S.Stornetta, “Improving the efficiency and reliability of digital time-stamping”, *Methods in Communication, Security, and Computer Science – Sequences’91*, 329–334, 1992.
2. J.Benaloh, M. de Mare, “Efficient broadcast time-stamping”, Technical Report 1, Clarkson University Department of Mathematics and Computer Science, August 1991.
3. A.Buldas, P.Laud, H.Lipmaa, J.Villemson, “Time-stamping with binary linking schemes”, *Advances in Cryptology – CRYPTO’98 (LNCS 1462)*, 486–501, 1998.
4. S.Haber, W.S.Stornetta, “How to time-stamp a digital document”, *Journal of Cryptology,* **3**(2), 99–111, 1991.
5. S.Haber, W.S.Stornetta, “Secure names for bit-strings”, *Proc. 4th ACM Conference on Computer and Communications Security,* 28–35, April 1997.
6. M.Just, “Some time-stamping protocol failures”, *Internet Society Symposium on Network and Distributed System Security,* 1998.
7. R.C.Merkle, *Secrecy, Authentication, and Public Key Systems*, UMI Research Press, Ann Arbor, Michigan, 1979.
8. R.C.Merkle, “Protocols for public key cryptosystems”, *Proceedings of the 1980 IEEE Symposium on Security and Privacy*, 122–134, 1980.
9. R.C.Merkle, “A certified digital signature”, *Advances in Cryptology – CRYPTO’89 (LNCS 435)*, 218–238, 1990.

10. F.Pinto, V.Freitas, "Digital time-stamping to support non repudiation in electronic communications", *Proc. SECURICOM'96 – 14th worldwide Congress on Computer and Communications Security and Protection*, CNIT, Paris, 397–406, June 5-6, 1996.
11. Digital Notary service, `http://www.surety.com`.