Directed Acyclic Graphs, One-way Functions and Digital Signatures

(Extended Abstract)

Daniel Bleichenbacher and Ueli M. Maurer

Institute for Theoretical Computer Science
ETH Zürich
CH-8092 Zürich, Switzerland
Email: {bleichen,maurer}@inf.ethz.ch

Abstract. The goals of this paper are to formalize and investigate the general concept of a digital signature scheme based on a general one-way function without trapdoor for signing a predetermined number of messages. It generalizes and unifies previous work of Lamport, Winternitz, Merkle, Even et al. and Vaudenay. The structure of the computation yielding a public key from a secret key corresponds to a directed acyclic graph $\mathcal{G}$. A signature scheme for $\mathcal{G}$ can be defined as an antichain in the poset of minimal verifyable sets of vertices of $\mathcal{G}$ with the naturally defined computability relation as the order relation and where a set is verifyable if and only if the public key can be computed from the set.

1 Introduction

Lamport [5] proposed a so-called one-time signature scheme based on a general one-way function (OWF), i.e., a function f that is easy to compute but computationally infeasible to invert, for suitable definitions of "easy" and "infeasible". Lamport's scheme for signing a single bit is set up by choosing as the secret key two strings x_0 and x_1 at random and revealing as the public key the pair $\langle f(x_0), f(x_1)\rangle$. The signature for bit b is x_b. For signing longer messages, several instances of this scheme can be used.

Motivated by Lamport's approach, many researchers have subsequently proposed more efficient one-time signature schemes. The goals of this paper are to formalize the concept of a signature scheme based on any OWF for signing a predetermined number of messages, and to present several results on the number and size of messages that can be signed with a given scheme. In contrast to Rompel's result [9] showing that a signature scheme can be obtained from any OWF, the emphasis of this paper is on efficiency and on a unified description of the general idea rather than on rigorously proving the security of schemes with respect to a certain intractability assumption.

In addition to the general interest in a class of intriguing graph-theoretic problems, our motivations for considering the design of signature schemes based

Y.G. Desmedt (Ed.): Advances in Cryptology - CRYPTO '94, LNCS 839, pp. 75-82, 1994.

on OWFs are as follows. First, there is a severe limitation on the diversity of mathematical problems (such as factoring integers or computing discrete logarithms in certain finite groups) that can at present be used as the bases for a digital signature scheme. Therefore an alternative design approach with a much larger degree of freedom in choosing the underlying cryptographic function appears to be of interest. Second, for applications where only few messages need to be signed, schemes based on an arbitrary one-way function have the potential of being computationally more efficient than presently-used number-theoretic schemes, but their disadvantage is that each public key can only be used for signing a predetermined number of messages. Moreover, even if these schemes turn out to be of limited interest as regular digital signature schemes, they do have applications in other contexts such as on-line/off-line signatures [3] and the signature schemes of [1].

The number (i.e., diversity) of messages that can be signed by the Lamport scheme with r public-key pairs is 2^r. Using the same secret key and public key, but allowing as signatures all subsets of cardinality r of the set of $2r$ public-key components, the number of messages can be improved to $\binom{2r}{r}$, which is optimal [11]. These sets are compatible because computing a signature from a different signature requires the inversion of the OWF for at least one value.

Note that the size of the secret key of such a scheme can be reduced significantly by generating all the secret-key components in a pseudo-random fashion from a single secret key S. Similarly, the public key can be reduced to a single value P by applying a one-way hash function to the list of public-key components.

A generalization of the Lamport scheme attributed by Merkle to Winternitz [6] is to apply the OWF to two secret key components iteratively a fixed number of times, resulting in a two-component public key. Meyer and Matyas [7] proposed as a further improvement to use more than two chains of function evaluations: they observed that a one-time signature scheme for a message space of size $K!$ can be obtained from a scheme with K chains of length K each, by allowing as signatures all combinations of K nodes containing one node in each chain such that at each level there is one of these nodes. This scheme was generalized further in [3] and later in [12] to a scheme with l chains of length k where the signatures consist of one node in each chain such that the total sum of the levels of these nodes (within their chains) is constant. It can be proved that this strategy yields the maximal number of signatures for such a computation structure.

The described schemes can only be used to sign a single message. Merkle [6] proposed the so-called tree-authentication scheme for signing several messages consecutively with a single public key P.

2 One-time signature schemes based on directed acyclic graphs

In this paper, vertices and sets of vertices of a graph are denoted by small and capital letters, respectively, and graphs, posets as well as sets of sets of vertices are denoted by calligraphic letters. We summarize some well-known definitions and results on partially ordered sets (poset). A poset is defined as a set with an antisymmetric, transitive and reflexive order relation, denoted $\leq$. Two elements x and y of a poset $\mathcal{Z} = (Z, \leq)$ are *comparable* if and only if $x \leq y$ or $y \leq x$ and they are *incomparable* otherwise. A subset $U \subseteq Z$ is a *chain* if every pair of elements of U is comparable, and it is called an *antichain* if every pair of elements of U is incomparable. A chain (antichain) is called *maximal* if it is not a subset of another chain (antichain).

Definition 1. The *width* of a poset $\mathcal{Z}$, denoted $w(\mathcal{Z})$, is the maximal cardinality of an antichain.

Definition 2. For a poset $\mathcal{Z} = (Z, \leq)$, a function $r : Z \rightarrow \mathbb{N}$ is called a *representation function* of $\mathcal{Z}$ if for all distinct $x, y \in Z$, $x \leq y$ implies $r(x) < r(y)$.

Let B be a suitable, large set (e.g., the set of 64, 96 or 128-bit strings) and let $f_1, f_2, \ldots$ with $f_i : B^i \longrightarrow B$ be a list of one-way functions, where f_i takes as input a list of i values in B and produces as output a single value in B. Consider a scenario in which a secret key S consisting of u values $s_1, \ldots, s_u \in B$ is chosen at random, and a sequence of values $s_{u+1}, s_{u+2}, \ldots, s_t$ is computed from $s_1, \ldots, s_u$ by applications of the one-way functions f_i. More precisely, for $u+1 \leq j \leq t$, s_j is the result of applying an appropriate OWF to a subset U_j (of appropriate size) of $\{s_1, \ldots, s_{j-1}\}$, where the order of the arguments is assumed to be fixed but is irrelevant for the further discussion. Some of these computed values will not be used as input to a OWF and are published as the public key P. Signatures consist of appropriately chosen subsets of $\{s_1, \ldots, s_t\}$.

In the following we need to distinguish between the structure of the described computation for setting up a digital signature scheme and the particular values resulting for a particular choice of the secret key. Consider a directed graph $\mathcal{G} = (V, E)$ with vertex set $V = \{v_1, \ldots, v_t\}$, where v_i corresponds to the value s_i, and with edge set E containing the edge (v_i, v_j) if and only if s_i is an input to the OWF resulting in s_j. Hence the value corresponding to v_j can be computed from the values corresponding to the predecessors of v_j, and it functionally depends on the value s_k (corresponding to v_k) if and only if there exists a directed path from v_k to v_j.

In such a graph the secret key set and the public key set correspond to the sets of vertices with in-degree 0 and out-degree 0, respectively. The graph $\mathcal{G}$ is assumed to be known publicly and can be used by all users, but the values corresponding to the vertices for a user's particular secret key are kept secret by the user, except those values corresponding to the public key. A signature scheme assigns a signature pattern, i.e. an appropriate subset of vertices, to

every message in the message space. A user's signature for a given message consists of the values (for that user's secret key) corresponding to the vertices in the signature pattern for that message, when the computation according to $\mathcal{G}$ is performed for that user's secret key. The set of signature patterns must satisfy certain conditions discussed below.

Definition 3. For a given directed acyclic graph (DAG) $\mathcal{G} = (V, E)$, the *secret key pattern* $S(\mathcal{G}) \subset V$ and the *public key pattern* $P(\mathcal{G}) \subset V$ are defined as the sets of vertices with in-degree 0 and out-degree 0, respectively. Let X be a subset of V. A vertex v is defined recursively to be *computable* from X if either $v \in X$ or if v has at least one predecessor and all predecessors are computable from X. A set Y is computable from X if every element of Y is computable from X. Note that V and hence every subset of V is computable from the secret key $S(\mathcal{G})$. A set $X \in V$ is called *verifyable* (with respect to the public key) if $P(\mathcal{G})$ is computable from X. A verifyable set X is *minimal* if no subset of X is verifyable. Two minimal verifyable sets (MVS) X and Y are *compatible* if neither X is computable from Y nor Y is computable from X. A set of MVSs is compatible if they are pairwise compatible.

Remarks.
(1) Of course, the OWFs used for evaluating different vertices can be different, as long as a function together with the order of the arguments is uniquely specified for each vertex.
(2) As mentioned before, the secret key components can be generated in a pseudo-random manner from a single secret key. We can hence extend $\mathcal{G}$ by introducing an extra vertex s_0 (the real secret key) and edges form s_0 to the vertices $s_1, \ldots, s_u$ (using the convention that when two vertices in $\mathcal{G}$ have the same set of predecessors, then the two OWFs used in the corresponding computation steps are different and unrelated). Similarly, one can without much loss of generality restrict the discussion to graphs with only one public-key component because a list of public-key components could be hashed using a secure cryptographic hash function.
(3) Because messages can be hashed prior to signing, it suffices to design signature schemes for a message space corresponding to the range of a secure cryptographic hash function, for instance the set of 128-bit strings.

The computability relation on the set of MVSs of a graph is transitive, antisymmetric and reflexive, and hence the set of MVSs of a graph $\mathcal{G}$, denoted $\mathcal{W}(\mathcal{G})$, forms a poset $(\mathcal{W}(\mathcal{G}), \leq)$ with computability as the order relation, i.e., we have $X \leq Y$ for $X, Y \in \mathcal{W}(\mathcal{G})$ if and only if X is computable from Y. Note that two MVSs of $\mathcal{G}$ are compatible if and only if they are incomparable in $(\mathcal{W}(\mathcal{G}), \leq)$.

Definition 4. The *associated poset* of DAG $\mathcal{G}$ is the poset $(\mathcal{W}(\mathcal{G}), \leq)$ of minimal verifyable subsets of $\mathcal{G}$.

In order to remove a possible source of confusion it should be pointed out that a DAG in which every edge (x, y) is the only path from x to y has itself the structure of a poset and $\mathcal{W}(\mathcal{G})$ is the poset of cutsets in this poset. However, we

have avoided the term "cutset" for the signature patterns because this term has a different meaning for graphs.

Definition 5. A *one-time signature scheme* $\mathcal{A}$ for an acyclic directed graph $\mathcal{G} = (V, E)$ is an antichain of the associated poset $\mathcal{W}(\mathcal{G})$.

The important parameters of a one-time signature scheme $\mathcal{A}$ for a graph $\mathcal{G} = (V, E)$ are the number $|V|$ of vertices (which is equal to the sum of the size of the secret key and the number of function evaluations required for computing a public key from a secret key), the number $|\mathcal{A}|$ of signatures which must be at least equal to the size of the message space, and the maximal size of signatures, $\max_{X \in \mathcal{A}} |X|$.

Example: Figure 1 shows a graph for which it is especially easy to design a signature scheme. At each level of the graph, the verifier is given one of two unknown values, where a message bit determines which one is given. This scheme allows to sign 1 bit per three vertices, i.e., 1/3 bit per vertex. More efficient schemes will be discussed later.

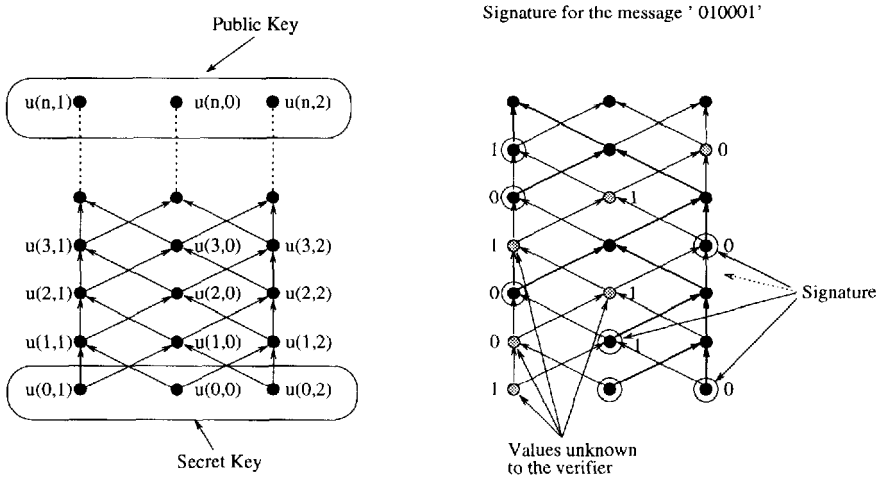

Figure 1.

The following interesting problems are now well-motivated. First, for a given graph $\mathcal{G}$ to find a large (ideally a maximal-sized) antichain in the associated poset. Note that $w(\mathcal{W}(\mathcal{G}))$ denotes the maximal size of such an antichain. Second, for a given size of the message space to find a graph with few (ideally the minimal number of) vertices allowing the construction of a one-time signature scheme. Third, both problems should be treated with a constraint on the maximal size of signatures, and also for a generalized definition of a signature scheme for signing several rather than only one message.

The maximal-sized antichain of a poset can be determined by a flow algorithm whose running time is polynomial in the number of elements of the poset. This method is only feasible for graphs of less than about 30 vertices. For larger

graphs, a very useful technique for determining lower bounds on the size of the maximal antichain is based on representation functions.

It follows from the definition of a representation function r of a poset $\mathcal{Z} = (Z, \leq)$ that for any $x \neq y$, $r(x) = r(y)$ implies that x and y are incomparable. Hence for any representation function r of the associated poset $(\mathcal{W}(\mathcal{G}), \leq)$ of a given DAG $\mathcal{G}$ and for any integer k, the set $\{U \in \mathcal{W}(\mathcal{G}) : r(U) = k\}$ is a one-time signature scheme. Let

$$\beta(\mathcal{G}, r) = \max_k(|\{U \in \mathcal{W}(\mathcal{G}) : r(U) = k\}|)$$

be the maximal cardinality of these sets.

In order to find good signature schemes for a given graph, we need to find a good representation function, that is one with a large maximal coefficient. For a given DAG $\mathcal{G}$ let $c_{\mathcal{G}} : \mathcal{W}(\mathcal{G}) \rightarrow \mathbb{N}$ be the function defined by

$$c_{\mathcal{G}}(U) := |\{v : v \notin U \text{ and } v \text{ is computable from } U\}|$$

i.e. $c_{\mathcal{G}}(U)$ is the cardinality of the set of vertices of $\mathcal{G}$ that are computable from U but are not contained in U.

A proof of the following theorem appears in the full paper.

Theorem 6. *For any DAG $\mathcal{G}$, the function $c_{\mathcal{G}}$ is a representation function of the associated poset $\mathcal{W}(\mathcal{G})$ of $\mathcal{G}$.*

This representation function can be computed easily for many graphs (e.g., for all trees) by using generating functions. Moreover $c_{\mathcal{G}}$ is an optimal representation function for many graphs $\mathcal{G}$, in the sense that $\beta(\mathcal{G}, c_{\mathcal{G}})$ is equal to the maximal number $w(\mathcal{W}(\mathcal{G}))$ of signatures patterns, but this is not true in general.

3 Optimal graphs and signature schemes

A reasonable implementation of a list of OWFs $f_1, f_2, f_3 \ldots$ with one, two, three, etc. arguments is by implementing a OWF f_2 with two arguments and implementing the function f_1 with one argument as $f_1(x) = f_2(x, x)$ and the functions f_i for $i \geq 3$ as $f_i(x_1, \ldots, x_i) = f_2(f_{i-1}(x_1, \ldots, x_{i-1}), x_i)$. The function f_2 can for instance be implemented by applying DES in an appropriate mode, but much more efficient implementations of good candidate OWFs are possible.

In the described implementation based on a function f_2, the graph could be considered to consist only of vertices with fan-in 1 or 2. In the sequel we discuss the problem of maximizing the number of signature patterns for a given number n of vertices under this fan-in restriction. Let $\nu(n)$ be the maximal number of MVSs obtainable for a graph with n vertices and let $\mu(n)$ be the maximal number of compatible MVSs for a graph with n vertices, i.e., let

$$\nu(n) = \max\{|\mathcal{W}(\mathcal{G})| : \mathcal{G} = (V, E) \text{ with } |V| = n\}$$
$$\mu(n) = \max\{w(\mathcal{W}(\mathcal{G})) : \mathcal{G} = (V, E) \text{ with } |V| = n\},$$

where $\mathcal{G}$ has fan-in at most 2 and public key of size 1. We will now derive concrete and asymptotic results on $\mu(n)$.

For a DAG $\mathcal{G} = (V, E)$ we define $\mathcal{R}_{\mathcal{G},l}$ to be the graph consisting of l unconnected identical copies of $\mathcal{G}$. The poset of MVSs of $\mathcal{R}_{\mathcal{G},l}$ consists of all l-tuples $(S_1, \ldots, S_l)$ for which S_i is a MVS of the i-th copy of $\mathcal{G}$.

Let $r_{\mathcal{G}}$ be any representation function of $\mathcal{W}(\mathcal{G})$ such that there exist $S_1, S_2 \in \mathcal{W}(\mathcal{G})$ where $r_{\mathcal{G}}(S_1) - r_{\mathcal{G}}(S_2) = 1$. We define the representation function r of $\mathcal{W}(\mathcal{R}_{\mathcal{G},l})$ by $r(S) = \sum_{i=1}^{l} r_{\mathcal{G}}(S_i)$ for an MVS $S = (S_1, \ldots, S_l) \in \mathcal{W}(\mathcal{R}_{\mathcal{G},l})$.

Theorem 7. *For the representation function r defined above we have*

$$\lim_{l\to\infty} \beta(\mathcal{R}_{\mathcal{G},l}, r)\frac{\sqrt{l}}{m^l} = \frac{1}{\sigma\sqrt{2\pi}}$$

where $m = |\mathcal{W}(\mathcal{G})|$ and σ is the standard deviation of $r_{\mathcal{G}}(S)$ if S is chosen uniformly from $\mathcal{W}(\mathcal{G})$.

Proof sketch. Let Y be the random variable defined by $Y = (r_{\mathcal{G}}(S) - E[r_{\mathcal{G}}(S)])/\sigma$ where S is chosen uniformly from $\mathcal{W}(\mathcal{G})$. The distribution of Y is a lattice distribution with span $1/\sigma$, $E[Y] = 0$ and $E[Y^2] = 1$. Now we can apply Theorem 3 of [4, p.490] to complete the proof. □

It should be mentioned that $\beta(\mathcal{R}_{\mathcal{G},l}, r) = O(m^l/\sqrt{l})$ is satisfied for any choice of $r_{\mathcal{G}}$. Theorem 7 implies the following result which is proved in the full paper.

Corollary 8. $\lim_{n\to\infty} \dfrac{\log_2 \mu(n)}{n} \geq \sup_m \dfrac{\log_2 \nu(m)}{m+1}$.

We have found a DAG $\mathcal{G}$ with 26 vertices whose associated poset has 5004 vertices. Thus $\mathcal{R}_{\mathcal{G},l}$ contains $O(5004^l/\sqrt{l})$ signatures patterns. In order to combine l copies of the graph in a tree with fan-in 2 we need a tree with only $l-1$ additional vertices. Therefore, there exists a sequence of graphs with n vertices which allows to sign $\alpha n - O(\log n)$ bits, i.e., α bits per vertex, where $\alpha = \log_2(5004)/27 = 0.4551$. This result cannot be achieved by trees as the following theorem demonstrates.

Theorem 9. *For every tree $\mathcal{T}$ with n vertices,*

$$|\mathcal{W}(\mathcal{T})| \leq 2^{\gamma(n+1)} \quad \text{where} \quad \gamma = \log_2(685/216)/4 \approx 0.4162. \tag{1}$$

In other words, no tree with n vertices allows to sign more than $\gamma(n+1)$ bits. On the other hand for $n \geq 1$ there exists a tree $\mathcal{T}_n$ with n vertices with

$$w(\mathcal{W}(\mathcal{T}_n)) \geq \frac{2^{\delta n}}{3\sqrt{n} + 3/2} \text{ where } \delta = \log_2(101)/16 \approx 0.4161. \tag{2}$$

Note that the upper and lower bounds of Theorem 9 are extremely close to each other. By refined arguments, γ can be reduced and δ increased slightly so that they agree in the first 8 decimal digits, but nevertheless it remains to prove that there exists a constant which is both upper and lower bound.

4 Concluding remarks

One problem with the schemes discussed in this extended abstract is that the signatures are relatively long. An interesting problem is to devise schemes with small signature patterns. One such scheme based on a forest of chains is discussed in [12]. Our definitions can be extended in this direction and also to include schemes for signing a fixed number of messages (rather than only one). Such generalizations will be discussed in a forthcoming paper.

Acknowledgements

The authors would like to thank Martin Perewusnyk, Adi Shamir, and Roger Wattenhofer for many inspiring discussions on the topic of this paper.

References

1. J.N.E. Bos and D. Chaum, Provably unforgeable signatures, *Advances in Cryptology – CRYPTO '92* (E. Brickell, ed.), Lecture Notes in Computer Science, vol. 740 , Springer Verlag, 1993, pp. 1-14.
2. N. de Brujin, C. A. van Ebbenhorst Tengebergen, and D. R. Kruyswijk, "On the set of divisors of a number," *Nieuw Arch. Wisk*, vol. 23, pp. 191–193, 1952.
3. S. Even, O. Goldreich and S. Micali, On-line/off-line digital signatures, *Advances in Cryptology – CRYPTO '89*, Lecture Notes in Computer Science, vol. 435 (G. Brassard, ed.), Springer Verlag, 1990, pp. 263-275.
4. W. Feller, *An Introduction to Probability Theory and Its Applications*, vol. II., second corrected printing, Wiley & Sons, 1966.
5. L. Lamport, Constructing digital signatures from a one-way function, Technical Report SRI Intl. CSL 98, 1979.
6. R. Merkle, A certified digital signature, *Advances in Cryptology – CRYPTO '89*, Lecture Notes in Computer Science, vol. 435 (G. Brassard, ed.), Springer Verlag, 1990, pp. 218-238.
7. C. Meyer and S. Matyas, *Cryptography – a new dimension in computer data security*, John Wiley & Sons, Inc., 1982.
8. R.L. Rivest, A. Shamir, and L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, *Communications of the ACM*, Vol. 21, No. 2, pp. 120-126, 1978.
9. J. Rompel, One-way functions are necessary and sufficient for secure signatures, *Proc. 22nd ACM Symp. on Theory of Computing (STOC)*, 1990, pp. 387-394.
10. C.P. Schnorr, Efficient identification and signatures for smart cards, Advances in Cryptology – Crypto '89, Lecture Notes in Computer Science, vol. 435 (G. Brassard, ed.), Springer-Verlag 1990, pp. 239-252.
11. E. Sperner, Ein Satz über Untermengen einer endlichen Menge, *Mathematische Zeitschrift*, vol. 27, pp. 544–548, 1928.
12. S. Vaudenay, One-time identification with low memory, *Proc. of EUROCODE '92*, Lecture Notes in Computer Science, Springer Verlag. CISM Courses and Lectures, No. 339, International Centre for Mechanical Sciences, P. Camion, P. Charpin and S. Harari (eds.), Springer-Verlag, pp. 217–228.