A Symbolic Intruder Model for Hash-Collision Attacks *

Yannick Chevalier and Mounira Kourjieh

IRIT Université Paul Sabatier, France
email: {ychevali,kourjieh}@irit.fr

Abstract. In the recent years, several practical methods have been published to compute collisions on some much used hash functions. Starting from two messages m_1 and m_2 these methods permit to compute m'_1 and m'_2 *similar* to the former such that they have the same image for a given hash function. In this paper we present a method to take into account, at the symbolic level, that an intruder actively attacking a protocol execution may use these collision algorithms in reasonable time during the attack. This decision procedure relies on the reduction of constraint solving for an intruder exploiting the collision properties of hash functions to constraint solving for an intruder operating on words, that is with an associative symbol of concatenation. The decidability of the latter is interesting in its own right as it is the first decidability result that we are aware of for an intruder system for which unification is infinitary, and permits to consider in other contexts an associative concatenation of messages instead of their pairing.

1 Introduction

Hash functions. Cryptographic hash functions play a fundamental role in modern cryptography. While related to conventional hash functions commonly used in non-cryptographic computer applications - in both cases, larger domains are mapped to smaller ranges - they have some additional properties. Our focus is restricted to cryptographic hash functions (hereafter, simply hash functions), and in particular to their use as cryptographic primitive for data integrity, authentication, key agreement, e-cash and many other cryptographic schemes and protocols. Hash functions take a message as input and produce an output referred to either as a *hash-code*, *hash-result*, or *hash-value*, or simply *hash*.

Collisions. A hash function is many-to-one, implying that the existence of collisions (pairs of inputs with the identical output) is unavoidable. However, only a few years ago, it was intractable to compute collisions on hash functions, so they were considered to be collision-free by cryptographers, and protocols were built upon this assumption. From the nineties on, several authors have proved the tractability of finding pseudo-collision and collision attacks over several hash

* supported by ARA-SSIA Cops and ACI JC 9005

functions. Taking this into account, we consider that cryptographic hash functions have the following properties:

- the input can be of any length, the output has a fixed length, $\mathrm{h}(x)$ is relatively easy to compute for any given x;
- pre-image resistance: for essentially all pre-specified outputs, it is computationally infeasible to find any input which hashes to that outputs, i.e., to find any x such that $y = \mathrm{h}(x)$ when given y;
- 2nd-pre-image resistance: it is computationally infeasible to find any second input which has the same output as any specified input, i.e., given x , to find x' different from x such that $\mathrm{h}(x) = \mathrm{h}(x')$;
- hash collision: it is computationally feasible to compute two distinct inputs x and x' which hash to the same output, i.e, $\mathrm{h}(x) = \mathrm{h}(x')$ provided that x and x' are created at the same time and independently one of the other.

In other words, a collision-vulnerable hash function h is one for which an intruder can find two different messages x and x' with the same hash value. To mount a collision attack, an adversary would typically begin by constructing two messages with the same hash where one message appears legitimate or innocuous while the other serves the intruder's purposes. For example, consider the following simple protocol:

$$A \rightarrow B : M, \sigma_A(M)$$

where $\sigma_A(M)$ denotes A's digital signature on message M using DAS digital signature scheme in which only the hash-value of M by a function h is considered. The following attack:

$$A \rightarrow B : M', \sigma_A(M)$$

can be launched successfully if the intruder first computes two different messages M and M' having the same hash value and then can lead Alice into executing the protocol with message M.

Collisions in practise. MD5 Hash function is one of the most widely used cryptographic hash functions nowadays. It was designed in 1992 as an improvement on MD4, and its security was widely studied since then by several authors. The first result was a pseudo-collision for MD5 [6]. When permitting to change the initialisation vector, another attack (free-start collision) has been found [8]. Recently, a real collision involving two 1024-bits messages was found with the standard value [19]. This first weakness was extended into a differential-like attack [22] and tools were developed [10, 9] for finding the collisions which work for any initialisation value and which are quicker than methods presented in [19]. Finally, other methods have been developed for finding new MD5 collisions [23, 17]. The development of collision-finding algorithms is not restricted to MD5 hash function. Several methods for MD4 research attack have been developed [20, 7]. In [20] a method to search RIPE-MD collision attacks was also developed, and in [2], a collision on SHA-0 has been presented. Finally, Wang *et al.* have developed in [21] another method to search for collisions for the SHA-1 hash function.

Goal of this paper. This development of methods at the cryptographic level to built collisions in a reasonable time have until now not been taken into account in a symbolic model of cryptographic protocols. We also note that the inherent complexity of these attacks make them not representable in any computational model that we are aware of. In this paper we propose a decision procedure to decide insecurity of cryptographic protocols when a hash function for which collisions may be found is employed. Relying on the result [3] we do not consider here other cryptographic primitives such as public key encryption, signature or symmetric key encryption, and assume that a protocol execution has already been split into the views of the different equational theories. The decidability proof presented here heavily relies on a recent result [4] that permits to reduce constraint solving problems with respect to a given intruder to constraint solving problems for a simpler one. This result relies on a new notion of *mode*. This notion aims at exhibiting a modular structure in an equational theory but has no simple intuitive meaning. In the case of an exponential operator as treated in [4] the separation was between an exponential symbol and the abelian group operations on its exponents, whereas here the separation is introduced between the application of the hash function and the functions employed by the intruder to find collisions.

Outline. We first give in Section 2 the definitions relating to terms and equational theories. We then present in Section 3 our model of an attacker against a protocol, and how we reduce the search for flaws to reachability problems with respect to an intruder theory. In Section 4 we describe in detail how we model the fact that an intruder may construct colliding messages, and how this intruder theory can be decomposed into simpler intruder theories. We give proof sketch of these reductions in Section 5 and conclude in Section 6.

2 Formal setting

2.1 Basic notions

We consider an infinite set of free constants C and an infinite set of variables $\mathcal{X}$. For any signature $\mathcal{G}$ (*i.e.* sets of function symbols not in C with arities) we denote $\mathrm{T}(\mathcal{G})$ (resp. $\mathrm{T}(\mathcal{G},\mathcal{X})$) the set of terms over $\mathcal{G}\cup\mathrm{C}$ (resp. $\mathcal{G}\cup\mathrm{C}\cup\mathcal{X}$). The former is called the set of ground terms over $\mathcal{G}$, while the latter is simply called the set of terms over $\mathcal{G}$. The arity of a function symbol f is denoted by $\mathrm{ar}(f)$. Variables are denoted by x, y, terms are denoted by s, t, u, v, and finite sets of terms are written $E, F, ...$, and decorations thereof, respectively. We abbreviate $E\cup F$ by E, F, the union $E\cup\{t\}$ by E, t and $E\setminus\{t\}$ by $E\setminus t$.

Given a signature $\mathcal{G}$, a *constant* is either a free constant or a function symbol of arity 0 in $\mathcal{G}$. We define the set of atoms Atoms to be the union of $\mathcal{X}$ and the set of constants. Given a term t we denote by $\mathrm{Var}(t)$ the set of variables occurring in t and by $\mathrm{Cons}(t)$ the set of constants occurring in t. We denote by $\mathrm{Atoms}(t)$ the set $\mathrm{Var}(t)\cup\mathrm{Cons}(t)$. A substitution σ is an involutive mapping from $\mathcal{X}$ to $\mathrm{T}(\mathcal{G},\mathcal{X})$ such that $\mathrm{Supp}(\sigma)=\{x|\sigma(x)\neq x\}$, the *support* of σ, is a finite set. The

application of a substitution σ to a term t (resp. a set of terms E) is denoted $t\sigma$ (resp. $E\sigma$) and is equal to the term t (resp. E) where all variables x have been replaced by the term $\sigma(x)$. A substitution σ is *ground* w.r.t. $\mathcal{G}$ if the image of $\mathrm{Supp}(\sigma)$ is included in $\mathrm{T}(\mathcal{G})$.

An *equational presentation* $\mathcal{H} = (\mathcal{G}, A)$ is defined by a set A of equations $u = v$ with $u, v \in \mathrm{T}(\mathcal{G}, \mathcal{X})$ and u, v without free constants. For any equational presentation $\mathcal{H}$ the relation $=_{\mathcal{H}}$ denotes the equational theory generated by $(\mathcal{G}, A)$ on $\mathrm{T}(\mathcal{G}, \mathcal{X})$, that is the smallest congruence containing all instances of axioms of A. Abusively we shall not distinguish between an equational presentation $\mathcal{H}$ over a signature $\mathcal{G}$ and a set A of equations presenting it and we denote both by $\mathcal{H}$. We will also often refer to $\mathcal{H}$ as an equational theory (meaning the equational theory presented by $\mathcal{H}$). An equational theory $\mathcal{H}$ is said to be *consistent* if two free constants are not equal modulo $\mathcal{H}$ or, equivalently, if it has a model with more than one element modulo $\mathcal{H}$.

For all signature $\mathcal{G}$ that we consider, we assume that $<_{\mathcal{G}}$ is a total simplification ordering on $\mathrm{T}(\mathcal{G})$ for which the minimal element is a free constant $\mathrm{c_{min}}$. *Unfailing completion* permits, given an equational theory $\mathcal{H}$ defined by a set A of equations, to build from A a (possibly infinite) set $R(A)$ of equations $l = r$ such that the ordered rewriting relation between terms defined by $t \rightarrow_{R(A)} t'$ if:

- There exists $l = r \in R(A)$ and a ground substitution σ such that $l\sigma = s$ and $r\sigma = s'$, $t = t[s]$ and $t' = t[s \leftarrow s']$;
- We have $t' <_{\mathcal{G}} t$.

This ordered rewriting relation is convergent, that is for all terms t, all ordered rewriting sequences starting from t are finite, and they all have the same limit, called the *normal form* of t. We denote this term $(t)\downarrow_{R(A)}$, or $(t)\downarrow$ when the equational theory considered is clear from the context. In the sequel we denote $\mathrm{C_{spe}}$ the set consisting of $\mathrm{c_{min}}$ and of all symbols in $\mathcal{G}$ of arity 0.

The *syntactic subterms* of a term t are denoted $\mathrm{Sub_{syn}}(t)$ and are defined recursively as follows. If t is an atom then $\mathrm{Sub_{syn}}(t) = \{t\}$. If $t = f(t_1, \ldots, t_n)$ then $\mathrm{Sub_{syn}}(t) = \{t\} \cup \bigcup_{i=1}^{n} \mathrm{Sub_{syn}}(t_i)$. The *positions* in a term t are sequences of integers defined recursively as follows, ε being the empty sequence. The term t is at position ε in t. We also say that ε is the root position. We write $p \leq q$ to denote that the position p is a prefix of position q. If u is a syntactic subterm of t at position p and if $u = f(u_1, \ldots, u_n)$ then u_i is at position $p \cdot i$ in t for $i \in \{1, \ldots, n\}$. We write $t_{|p}$ the subterm of t at position p. We denote $t[s]$ a term t that admits s as syntactic subterm. We denote by $\mathrm{top}(\_)$ the function that associates to each term t its root symbol.

2.2 Mode in an equational theory

We recall here the notion of *mode* on a signature, which is defined in [4]. Assume $\mathcal{H}$ is an equational theory over a signature $\mathcal{G}$, and let $\mathcal{G}_0$ be a subset of $\mathcal{G}$. Assume also that the set of variables is partitioned into two sets $\mathcal{X}_0$ and $\mathcal{X}_1$. We first define

a signature function $\mathrm{Sign}(\_)$ on $\mathcal{G} \cup \mathrm{Atoms}$ in the following way:

$$\mathrm{Sign}(\_) \ : \ \mathcal{G} \cup \mathrm{Atoms} \to \{0,1,2\}$$
$$\mathrm{Sign}(f) = \begin{cases} 0 \text{ if } f \in \mathcal{G}_0 \cup \mathcal{X}_0 \\ 1 \text{ if } f \in (\mathcal{G} \setminus \mathcal{G}_0) \cup \mathcal{X}_1 \\ 2 \text{ otherwise, i.e. when } f \text{ is a free constant} \end{cases}$$

The function $\mathrm{Sign}(\_)$ is extended to terms by taking $\mathrm{Sign}(t) = \mathrm{Sign}(\mathrm{top}(t))$.

We also assume that there exists a *mode* function $\mathrm{m}(\cdot,\cdot)$ such that $\mathrm{m}(f,i)$ is defined for every symbol $f \in \mathcal{G}$ and every integer i such that $1 \leq i \leq \mathrm{ar}(f)$. For all valid f, i we have $\mathrm{m}(f,i) \in \{0,1\}$ and $\mathrm{m}(f,i) \leq \mathrm{Sign}(f)$. Thus for all $f \in \mathcal{G}_0$ and for all i we have $\mathrm{m}(f,i) = 0$.

Well-moded equational theories. A position different from ε in a term t is *well-moded* if it can be written $p \cdot i$ (where p is a position and i a nonnegative integer) such that $\mathrm{Sign}(t_{|p\cdot i}) = \mathrm{m}(\mathrm{top}(t_{|p}), i)$. In other words the position in a term is well-moded if the subterm at that position is of the expected type w.r.t. the function symbol immediately above it. A term is *well-moded* if all its *non root* positions are well-moded. Note in particular that a well-moded term does not contain free constants. If a position of t is not well-moded we say it is *ill-moded* in t. A term is *pure* if its only ill-moded subterms are atoms. An equational presentation $\mathcal{H} = (\mathcal{G}, A)$ is well-moded if for all equations $u = v$ in A the terms u and v are well-moded and $\mathrm{Sign}(u)$=$\mathrm{Sign}(v)$. One can prove that if an equational theory is well-moded then its completion is also well-moded [4].

Note that if $\mathcal{H}$ is the union of two equational theories $\mathcal{H}_0$ and $\mathcal{H}_1$ over two disjoint signatures $\mathcal{G}_0$ and $\mathcal{G}_1$, the theory $\mathcal{H}$ is well-moded when assigning mode i to each argument of each operator $g \in \mathcal{G}_i$, for $i \in \{0,1\}$.

Subterm values. The notion of mode also permits to define a new subterm relation in $\mathrm{T}(\mathcal{G}, \mathcal{X})$.

We call a *subterm value* of a term t a syntactic subterm of t that is either atomic or occurs at an ill-moded position of t[1]. We denote $\mathrm{Sub}(t)$ the set of subterm values of t. By extension, for a set of terms E, the set $\mathrm{Sub}(E)$ is defined as the union of the subterm values of the elements of E. The subset of the maximal and strict subterm values of a term t plays an important role in the sequel. We call these subterm values the *factors* of t, and denote this set $\mathrm{Factors}(t)$.

Example 1. Consider two binary symbols f and g with $\mathrm{Sign}(f) = \mathrm{Sign}(g) = \mathrm{m}(f,1) = \mathrm{m}(g,1) = 1$ and $\mathrm{m}(f,2) = \mathrm{m}(g,2) = 0$, and $t = f(f(g(a,b), f(c,c)), d)$. Its subterm values are a, b, $f(c,c)$, c, d, and its factors are a, b, $f(c,c)$ and d.

In the rest of this paper and unless otherwise indicated, *the notion of subterm will refer to subterm values.*

[1] Note that the root position of a term is *always* ill-moded.

Unification systems. We review here properties of well-moded theories with respect to unification that are addressed in [4].

Assume $\mathcal{H}$ is a well-moded equational theory over a signature $\mathcal{G}$, and let $\mathcal{H}_0$ be its projection over the signature $\mathcal{G}_0$ of symbols of signature 0. Let us first define unification systems with ordering constraints.

Definition 1. *(Unification systems) Let $\mathcal{H}$ be a set of equational axioms on $\mathrm{T}(\mathcal{G}, \mathcal{X})$. An $\mathcal{H}$-*unification system *$\mathcal{S}$ is a finite set of couples of terms in $\mathrm{T}(\mathcal{G}, \mathcal{X})$ denoted by $\{u_i \stackrel{?}{=} v_i\}_{i\in\{1,\ldots,n\}}$. It is satisfied by a ground substitution σ, and we note $\sigma \models_{\mathcal{H}} \mathcal{S}$, if for all $i \in \{1, \ldots, n\}$ we have $u_i\sigma =_{\mathcal{H}} v_i\sigma$.*

We will consider only satisfiability of unification systems with ordering constraints. That is, we consider the following decision problem:

Ordered Unifiability

Input: A $\mathcal{H}$-unification system $\mathcal{S}$ and an ordering $\prec$ on the variables X and constants C of $\mathcal{S}$.

Output: SAT iff there exists a substitution σ such that $\sigma \models_{\mathcal{H}} \mathcal{S}$ and for all $x \in X$ and $c \in C$, $x \prec c$ implies $c \notin \mathrm{Sub}_{\mathrm{syn}}(x\sigma)$

3 Analysis of reachability properties of cryptographic protocols

We recall in this section the definitions of [3] concerning our model of an intruder attacking actively a protocol, and of the simultaneous constraint satisfaction problems employed to model a finite execution of a protocol.

3.1 Intruder deduction systems

We first recall here the general definition of intruder systems, as is given in [3]. We then recall the definition of a *well-moded intruder* that we will use in this paper. In the context of a security protocol (see *e.g.* [12] for a brief overview), we model messages as ground terms and intruder deduction rules as rewrite rules on sets of messages representing the knowledge of an intruder. The intruder derives new messages from a given (finite) set of messages by applying intruder rules. Since we assume some equational axioms $\mathcal{H}$ are satisfied by the function symbols in the signature, all these derivations have to be considered *modulo* the equational congruence $=_{\mathcal{H}}$ generated by these axioms. In our setting an intruder deduction rule is specified by a term t in some signature $\mathcal{G}$. Given values for the variables of t the intruder is able to generate the corresponding instance of t.

Definition 2. *An* intruder system *$\mathcal{I}$ is given by a triple $\langle \mathcal{G}, \mathcal{S}, \mathcal{H}\rangle$ where $\mathcal{G}$ is a signature, $S \subseteq \mathrm{T}(\mathcal{G}, \mathcal{X})$ and $\mathcal{H}$ is a set of equations between terms in $\mathrm{T}(\mathcal{G}, \mathcal{X})$. To each $t \in S$ we associate a* deduction rule $\mathrm{L}^t : \mathrm{Var}(t) \to t$ *and $\mathrm{L}^{t,\mathrm{g}}$ denotes the set of ground instances of the rule L^t* modulo *$\mathcal{H}$:*

$$\mathrm{L}^{t,\mathrm{g}} = \{l \to r \mid \exists \sigma, \textit{ground substitution on } \mathcal{G},\ l = \mathrm{Var}(t)\sigma \text{ and } r =_{\mathcal{H}} t\sigma\}$$

The set of rules $\mathrm{L}_{\mathcal{I}}$ is defined as the union of the sets $\mathrm{L}^{t,\mathrm{g}}$ for all $t \in \mathcal{S}$.

Each rule $l \to r$ in $\mathrm{L}_{\mathcal{I}}$ defines an intruder deduction relation $\to_{l\to r}$ between finite sets of terms. Given two finite sets of terms E and F we define $E \to_{l\to r} F$ if and only if $l \subseteq E$ and $F = E \cup \{r\}$. We denote $\to_{\mathcal{I}}$ the union of the relations $\to_{l\to r}$ for all $l \to r$ in $L_{\mathcal{I}}$ and by $\to_{\mathcal{I}}^*$ the transitive closure of $\to_{\mathcal{I}}$. Note that by definition, given sets of terms E, E' ,F and F' such that $E =_{\mathcal{H}} E'$ and $F =_{\mathcal{H}} F'$ we have $E \to_{\mathcal{I}} F$ iff $E' \to_{\mathcal{I}} F'$. We simply denote by $\to$ the relation $\to_{\mathcal{I}}$ when there is no ambiguity about $\mathcal{I}$.

A *derivation* D of length n, $n \geq 0$, is a sequence of steps of the form $E_0 \to_{\mathcal{I}} E_0, t_1 \to_{\mathcal{I}} \cdots \to_{\mathcal{I}} E_n$ with finite sets of ground terms $E_0, \ldots E_n$, and ground terms $t_1, \ldots, t_n$, such that $E_i = E_{i-1} \cup \{t_i\}$ for every $i \in \{1, \ldots, n\}$. The term t_n is called the *goal* of the derivation. We define $\overline{E}^{\mathcal{I}}$ to be equal to the set $\{t \mid \exists F \text{ s.t. } E \to_{\mathcal{I}}^* F \text{ and } t \in F\}$ *i.e.* the set of terms that can be derived from E. If there is no ambiguity on the deduction system $\mathcal{I}$ we write $\overline{E}$ instead of $\overline{E}^{\mathcal{I}}$.

We now define well-moded intruder systems and their properties.

Definition 3. *Given a well-moded equational theory $\mathcal{H}$, an intruder system $\mathcal{I} = \langle \mathcal{G}, S, \mathcal{H} \rangle$ is* well-moded *if all terms in S are well-moded.*

3.2 Simultaneous constraint satisfaction problems

We introduce now the constraint systems to be solved for checking protocols. It is shown in [3] how these constraint systems permit to express the reachability of a state in a protocol execution.

Definition 4. *(Constraint systems) Let $\mathcal{I} = \langle \mathcal{G}, S, \mathcal{H} \rangle$ be an intruder system. An* $\mathcal{I}$-Constraint system *$\mathcal{C}$ is denoted: $((E_i \rhd v_i)_{i\in\{1,\ldots,n\}}, \mathcal{S})$ and it is defined by a sequence of couples $(E_i, v_i)_{i\in\{1,\ldots,n\}}$ with $v_i \in \mathcal{X}$ and $E_i \subseteq \mathrm{T}(\mathcal{G}, \mathcal{X})$ for $i \in \{1, \ldots, n\}$, and $E_{i-1} \subseteq E_i$ for $i \in \{2, \ldots, n\}$ and by an $\mathcal{H}$-unification system $\mathcal{S}$.*

An $\mathcal{I}$-Constraint system *$\mathcal{C}$ is satisfied by a ground substitution σ if for all $i \in \{1, \ldots, n\}$ we have $v_i\sigma \in \overline{E_i\sigma}$ and if $\sigma \models_{\mathcal{H}} \mathcal{S}$. If a ground substitution σ satisfies a constraint system $\mathcal{C}$ we denote it by $\sigma \models_{\mathcal{I}} \mathcal{C}$.*

Constraint systems are denoted by $\mathcal{C}$ and decorations thereof. Note that if a substitution σ is a solution of a constraint system $\mathcal{C}$, by definition of constraint and unification systems the substitution $(\sigma)\downarrow$ is also a solution of $\mathcal{C}$. In the context of cryptographic protocols the inclusion $E_{i-1} \subseteq E_i$ means that the knowledge of an intruder does not decrease as the protocol progresses: after receiving a message a honest agent will respond to it. This response can be added to the knowledge of an intruder who listens to all communications.

We are not interested in general constraint systems but only in those related to protocols. In particular we need to express that a message to be sent at some step i should be built from previously received messages recorded in the variables $v_j, j < i$, and from the initial knowledge. To this end we define:

Definition 5. *(Deterministic Constraint Systems) We say that an $\mathcal{I}$-constraint system $((E_i \rhd v_i)_{i\in\{1,\ldots,n\}}, \mathcal{S})$ is* deterministic *if for all i in $\{1, \ldots, n\}$ we have $\mathrm{Var}(E_i) \subseteq \{v_1, \ldots, v_{i-1}\}$*

In order to be able to combine solutions of constraints for the intruder theory $\mathcal{I}$ with solutions of constraint systems for intruders defined on a disjoint signature we have, as for unification, to introduce some ordering constraints to be satisfied by the solution (see [3] for details on this construction). Intuitively, these ordering constraints prevent from introducing cycle when building a global solution. This motivates us to define the *Ordered Satisfiability* problem:

Ordered Satisfiability

Input:	an $\mathcal{I}$-constraint system $\mathcal{C}$, $X = \mathrm{Var}(\mathcal{C})$, $C = \mathrm{Const}(\mathcal{C})$ and a linear ordering $\prec$ on $X \cup C$.
Output:	SAT iff there exists a substitution σ such that $\sigma \models_{\mathcal{I}} \mathcal{C}$ and for all $x \in X$ and $c \in C$, $x \prec c$ implies $c \notin \mathrm{Sub}_{\mathrm{syn}}(x\sigma)$

4 Model of a collision-aware intruder

We define in this section intruder systems to model the way an active intruder may deliberately create collisions for the application of hash functions. Note that our model doesn't take into account the time for finding collisions, which is significantly greater than the time necessary for other operations. The results that we can obtain can therefore be seen as worst-case results, and should be assessed with respect to the possible time deadline in the actual specification of a protocol under analysis. Further works will also be concerned with the fact that given a bound on intruder's deduction capabilities, a collision may be found only with a probability p, $0 \leq p \leq 1$.

We consider in this paper five different intruder models. We will reduce in two steps the most complex one to a simpler one, relying on the notion of well-moded theories and on the results in [4]. We then prove decidability of ordered reachability for this simpler intruder system.

4.1 Intruder on words

We first define our goal intruder, that is an intruder only able to concatenate messages and extract *prefixes* and *suffixes*. We denote $\mathcal{I}_{\mathrm{AU}} = \langle \mathcal{F}_{AU}, \mathcal{S}_{AU}, \mathcal{E}_{AU} \rangle$ an intruder system that operates on words, such that, if $\_ \cdot \_$ denotes the concatenation and ϵ denotes the empty word, the intruder has at its disposal all ground instances of the following deduction rules:

$$\begin{cases} x, y \rightarrow x \cdot y \\ x \cdot y \rightarrow x \\ x \cdot y \rightarrow y \\ \rightarrow \epsilon \end{cases}$$

We moreover assume that the concatenation and empty word operations satisfy the following equations:

$$\begin{cases} x \cdot (y \cdot z) = (x \cdot y) \cdot z \\ x \cdot \epsilon = x \qquad \epsilon \cdot x = x \end{cases}$$

Given these definitions, we can see terms over $\mathrm{T}(\mathcal{F}_{\mathrm{AU}}, \mathcal{X})$ as *words* over the alphabet $\mathcal{X} \cup \mathrm{C}$, and we denote letters($w$) the set of atoms (either variable or free constants) occurring in w. As usual, we extend letters(_) to set of terms in $\mathrm{T}(\mathcal{F}_{\mathrm{AU}}, \mathcal{X})$ by taking the union of letters occurring in each term.

Pitfall. Note that this intruder model does not fit into the intruder systems definition of [3, 4]. The rationale for this is that, in the notation given here, the application of the rules is non-deterministic, and thus cannot be modelled easily into our "deduction by normalisation" model. We however believe that a deterministic and still associative model of message concatenation by means of an "element" unary operator, associative operator "$\cdot$", and Head and Tail operations may be introduced. This means that we also assume that unification problems are only among words of this underlying theory, disregarding equations that may involve these extra operators. We leave the exact soundness of our model for further analysis and concentrate on the treatment of collisions discovery for hash functions.

4.2 Intruder on words with free function symbols

We extend the $\mathcal{I}_{\mathrm{AU}}$ intruder with two free function symbols $\mathtt{f}$ and $\mathtt{g}$ of arity 4, and we add to the possible deductions of the intruder the application of the following deduction rules:

$$\begin{cases} x_1, x_2, y_1, y_2 \to \mathtt{g}(x_1, x_2, y_1, y_2) \\ x_1, x_2, y_1, y_2 \to \mathtt{f}(x_1, x_2, y_1, y_2) \end{cases}$$

This leads to an intruder system that will be an intermediate in the proof of our decision procedure. We denote it $\mathcal{I}_{\mathrm{free}}$, and we have:

$$\mathcal{I}_{\mathrm{free}} = \langle \mathcal{F}_{\mathrm{AU}} \cup \{\mathtt{g}, \mathtt{f}\}, S_{\mathrm{AU}} \cup \{\mathtt{f}(x_1, x_2, y_1, y_2), \mathtt{g}(x_1, x_2, y_1, y_2)\}, \mathcal{E}_{\mathrm{AU}} \rangle \,.$$

4.3 Hash-colliding intruder

We consider a signature modelling the following different operations:

- The *concatenation* of two messages, the extraction of a suffix or a prefix of a concatenated message and the production of an empty message, as in the case of the $\mathcal{I}_{\mathrm{AU}}$ intruder system;
- The application of a hash function $\mathtt{h}$ for which it is possible to find collisions, the hash-value of a message m denoted $\mathtt{h}(m)$;
- Two function symbols $\mathtt{f}$ and $\mathtt{g}$ denoting the (complex) algorithm being used to find collisions starting from two different messages m and m'.

We assume that the algorithm employed by the intruder to find collisions starting from two messages m and m' proceeds as follows:

1. First the intruder splits both messages into two parts, thus choosing m_1, m_2, m'_1, m'_2 such that $m = m_1 \cdot m_2$ and $m' = m'_1 \cdot m'_2$;

2. Then, in order to find collisions, the intruder computes two messages $\mathrm{g}(m_1, m_2, m_1', m_2')$ and $\mathrm{f}(m_1, m_2, m_1', m_2')$ such that:

$$\text{(HC)} \quad \mathrm{h}(m_1 \cdot \mathrm{g}(m_1, m_2, m_1', m_2') \cdot m_2) = \mathrm{h}(m_1' \cdot \mathrm{f}(m_1, m_2, m_1', m_2') \cdot m_2')$$

A consequence of our model is that in order to build collisions starting from two messages m and m' the intruder must know (*i.e.* have in its knowledge set) these two messages. A side effect is that it is not possible to build three (or more) different messages with the same hash value by iterating the research for collisions. Formally, the core of the proof of this assertion is the following lemma that permits to prove that in an equivalence class of $\mathcal{E}_{\mathrm{h}}$ containing pure terms there exists only two different members modulo $\mathcal{E}_{\mathrm{AU}}$. The proof is based on the fact that occur-check analysis, and thus unification, would fail in a tentative counter-example.

Lemma 1. *Let* $t = \mathrm{h}(t_1 \cdot \phi(t_2, t_3, t_4, t_5) \cdot t_6)$ *and* $t' = \mathrm{h}(x_1 \cdot \mathrm{g}(x_1, x_2, x_3, x_4) \cdot x_2)$ *be two pure terms such that* $\phi \in \{\mathrm{f}, \mathrm{g}\}$ *and the* t_i *are pure* $\mathcal{F}_{\mathrm{AU}}$ *terms and there exists* $u_1, u_2 \in \{t_2, \ldots, t_5\}$ *such that* $t_1 =_{\mathcal{E}_{\mathrm{AU}}} u_1, t_6 =_{\mathcal{E}_{\mathrm{AU}}} u_2$. *Then for any substitution* σ, *we have* $\sigma \models t \stackrel{?}{=}_{\emptyset} t'$ *iff* $\sigma \models \left\{ t_{i+1} \stackrel{?}{=}_{\mathcal{E}_{\mathrm{AU}}} x_i \right\}_{i \in \{1,\ldots,4\}}$ *and:*

$$\begin{cases} \phi = \mathrm{g} \\ x_1\sigma = t_1\sigma \quad x_2\sigma = t_6\sigma \end{cases}$$

In a more comprehensive model we might moreover want to model that collisions cannot always be found using attacks published in the literature, but instead that given a deadline, the probability p of success of an attack is strictly below 1. This would imply that the application of this rule by the intruder would, assuming independence of collision attacks, reduce the likelihood of the symbolic attack found. In this setting our model would account for attacks with a non-negligible probability of success as is shown in [1].

Leaving probabilities aside, we express intruder's deductions in our setting by adding the rule $x \rightarrow \mathrm{h}(x)$ to the deduction rules of the $\mathcal{I}_{\mathrm{free}}$ intruder. As a consequence, the previous description of the $\mathcal{I}_{\mathrm{free}}$ intruder enables us to model a collision-capable intruder

$$\mathcal{I}_{\mathrm{h}} = \langle \mathcal{F}_{\mathrm{h}}, S_{\mathrm{h}}, \mathcal{E}_{\mathrm{h}} \rangle$$

$$\text{with:} \quad \begin{cases} \mathcal{F}_{\mathrm{h}} = \mathcal{F}_{\mathrm{AU}} \cup \{\mathrm{f}, \mathrm{g}, \mathrm{h}\} \\ S_{\mathrm{h}} = S_{\mathrm{AU}} \cup \{\mathrm{f}(x_1, x_2, y_1, y_2), \mathrm{g}(x_1, x_2, y_1, y_2), \mathrm{h}(x)\} \\ \mathcal{E}_{\mathrm{h}} = \mathcal{E}_{\mathrm{AU}} \cup \{(HC)\} \end{cases}$$

For the following mode and signature functions the theory $\mathcal{E}_{\mathrm{AU}} \cup \{(HC)\}$ is a well-moded theory.

$$\text{mode:} \quad \begin{cases} \mathrm{m}(.,1) = \mathrm{m}(.,2) = \mathrm{m}(\mathrm{g}\,, i) = \mathrm{m}(\mathrm{f}\,, i) = 0 \; \forall i \in \{1, \ldots, 4\} \\ \mathrm{m}(\mathrm{h}\,, 1) = 0 \end{cases}$$

$$\text{Signature:} \quad \begin{cases} \mathrm{Sign}(\cdot) = \mathrm{Sign}(\epsilon) = \mathrm{Sign}(\mathrm{f}\,) = \mathrm{Sign}(\mathrm{g}\,) = 0 \\ \mathrm{Sign}(\mathrm{h}\,) = 1 \end{cases}$$

The main result of this paper is the following decidability result.

Theorem 1 *Ordered satisfiability for the $\mathcal{I}_{\mathrm{h}}$ intruder is decidable.*

The rest of this paper is dedicated to the proof of this theorem. The technique employed consists in successive reductions to simpler problems and in finally proving that all simpler problems are decidable. These reductions are summarised in Figure 4.3. A proof sketch for the decidability of the $\mathcal{I}_{\mathrm{g}}$, $\mathcal{I}_{\mathrm{f}}$ and $\mathcal{I}_{\mathrm{AU}}$ is given in Section 5.2. Algorithm 1, that permits the first reduction, is based on the facts that the $\mathcal{I}_{\mathrm{h}}$ intruder is well-moded (as seen above) and that we can apply a reduction according to the criterion of [4] for well-moded intruder systems.

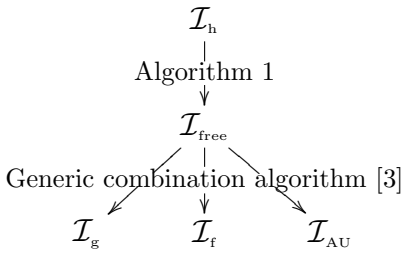

Fig. 1. Reduction strategy

HYPOTHESIS 1: If $E \to_{\mathcal{S}_1} E, r \to_{\mathcal{S}_1} E, r, t$ and $r \notin \mathrm{Sub}(E, t) \cup \mathrm{C}_{\mathrm{spe}}$ then there is a set of terms F such that $E \to^*_{\mathcal{S}_0} F \to_{\mathcal{S}_1} F, t$.

If a well-moded intruder system system satisfies this hypothesis, then the following proposition holds. It is a cornerstone for the proof of completeness of Algorithm 1.

Proposition 1. *Let $\mathcal{I}$ be a well-moded intruder that satisfies hypothesis 1, and let $\mathcal{C}$ be a deterministic $\mathcal{I}$-constraint system. If $\mathcal{C}$ is satisfiable, there exists a substitution σ such that $\sigma \models_{\mathcal{I}} \mathcal{C}$ and:*

$$|\{t \in \mathrm{Sub}((\mathrm{Sub}(\mathcal{C})\sigma)\downarrow) \,|\, \mathrm{Sign}(t) = 1\}| \leq |\{t \in \mathrm{Sub}(\mathcal{C}) \,|\, \mathrm{Sign}(t) = 1\}| + |X|$$

5 Decidability of reachability

We present here a decision procedure for *Ordered Satisfiability Problem* for $\mathcal{I}_{\mathrm{h}}$ intruder system. Our technique consists in simplifying the intruder system $\mathcal{I}_{\mathrm{h}}$ to $\mathcal{I}_{\mathrm{free}}$. We then reduce the decidability problems of ordered reachability for deterministic constraint problems for $\mathcal{I}_{\mathrm{free}}$ to the decidability problems of ordered reachability for deterministic constraint problems for $\mathcal{I}_{\mathrm{g}}$, $\mathcal{I}_{\mathrm{f}}$ and $\mathcal{I}_{\mathrm{AU}}$. We finally prove the decidability for these intruder systems.

5.1 Reduction to $\mathcal{I}_{\text{free}}$-intruder

Algorithm We present here a procedure for reducing $\mathcal{I}_{\text{h}}$ intruder system to $\mathcal{I}_{\text{free}}$ intruder system that takes as input a deterministic constraint system $\mathcal{C} = ((E_i \rhd v_i)_{i\in\{1,\ldots,n\}}, \mathcal{S})$ and a linear ordering $\prec_i$ on atoms of $\mathcal{C}$. Let $m = |\text{Sub}(\mathcal{C})|$ be the number of subterms in $\mathcal{C}$.

Algorithm 1

Step 1. Choose a number $k \leq m$ and add k equations $h_j \stackrel{?}{=} \text{h}(c_j)$ to $\mathcal{S}$ where the h_j, c_j are new variables.

Step 2. For each $t \in \text{Sub}(\mathcal{C}) \cup \{c_1, \ldots, c_k\}$ choose a *type* 0 or 1. If t is of type 1, choose $j_t \in \{1, \ldots, k\}$ and add an equation $t \stackrel{?}{=} h_{j_t}$ to $\mathcal{S}$.

Step 3. For all $t, t' \in \text{Sub}(\mathcal{C})$, if there exists $h \in \{h_1, \ldots, h_k\}$ such that $t \stackrel{?}{=} h$ and $t' \stackrel{?}{=} h$ are in $\mathcal{S}$, add to $\mathcal{S}$ an equation $t \stackrel{?}{=} t'$ to $\mathcal{S}$.

Step 4. Choose a subset H of $\{c_1, \ldots, c_k\}$ and guess a total order $<_d$ on $L = H \cup \{v_1, \ldots, v_n\}$ such that $v_i <_d v_j$ iff $i < j$. Write the obtained list $w_1, \ldots, w_{n+k}$. Let $\mathcal{S}$' be the unification system obtained so far, and form: $\mathcal{C}' = ((F_i \rhd w_j)_{1\leq j\leq n+k}, \mathcal{S}')$ with:

$$\begin{cases} F_1 = E_1 & \\ F_{i+1} = F_i \cup (E_{j+1} \setminus E_j) & \text{if } w_i = v_j \\ F_{i+1} = F_i, w_i & \text{Otherwise} \end{cases}$$

Step 5. For all $t \in \text{Sub}(\mathcal{C})$ chosen of type 1, replace all occurrences of t in the F_i and all occurrence occurrences of t *as a strict subterm* in $\mathcal{S}'$ by the representant of its class h_{j_t}. Let F'_i be the set F_i once this abstraction has been applied

Step 6. Non-deterministically reduce $\mathcal{S}$' to a unification system $\mathcal{S}$" free of h symbols, and form the satisfiable $\mathcal{I}_{\text{free}}$ constraint system:

$$\mathcal{C}'' = ((F'_i \rhd w_i)_{1\leq i\leq n+k}, \mathcal{S}'')$$

Sketch of the completeness proof. Assume that the initial deterministic constraint system is satisfiable. By Proposition 1, there exists a bound substitution σ satisfying $\mathcal{C}$.

- Let the number k chosen at Step 1 be the number of subterms whose top symbol is h in $\text{Sub}((\text{Sub}(\mathcal{C})\sigma)\downarrow)$. The h_j represent the different values of the terms of signature 1. In the sequel we assume that σ is extended to the h_j such that all $h_j\sigma$ have a different value and are of signature 1.
- In Step 2, if $\text{Sign}((t\sigma)\downarrow) = 1$ we choose the j such that $(t\sigma)\downarrow = h_j\sigma$ and add the corresponding equation to $\mathcal{S}$.
- In Step 3, we had equations between terms whose normal form by σ are equals in order to simplify the reduction to $\mathcal{I}_{\text{free}}$.
- Step 4 is slightly more intricate. It relies on the fact that a rule in $\mathcal{S}_1$ may only yield a term whose normal form by σ is of signature 1.

The subset H correspond to the subterms of signature 1 of $\mathrm{Sub}((\mathrm{Sub}(\mathcal{C}\sigma))\downarrow)$ that are deduced by the intruder using a rule in $\mathcal{S}_1$. We then anticipate the construction of $h_j\sigma$ with the application of a rule in $\mathcal{S}_1$ by requiring that the corresponding $c_j\sigma$ has to be build just before. Given the bound on k, this means that all remaining deductions performed by the intruder are now instances of rules in $\mathcal{S}_0$. Since $\mathcal{C}$ is satisfied by σ there exists a choice corresponding to *quasi well-formed* derivations such that all remaining reachability constraints are satisfiable by instances of rules in $\mathcal{S}_0$.

- At Step 5 we "purify" almost all the constraint system by removing all occurrences of a symbol h but the ones that are on the top of an equality. By the choice of the equivalence classes it is clear that this purification does not loose the satisfiability by the substitution σ.
- The non-deterministic reduction is performed by guessing whether the equality of two hashes is the consequence of a collision set up by the intruder or of the equality of the hashed messages, and will produce a constraint system $\mathcal{C}$" without h symbol and also satisfiable by σ.

5.2 Decidability of reachability for the $\mathcal{I}_{\mathrm{free}}$-intruder

We first reduce the $\mathcal{I}_{\mathrm{free}}$ intruder system to simpler intruder systems using the combination result of [3]. We will consider the decidability of these subsystems in the remainder of this section.

Theorem 2 *Ordered satisfiability for the $\mathcal{I}_{\mathrm{free}}$ intruder system is decidable.*

Decidability of reachability for the $\mathcal{I}_{\mathrm{g}}$-intruder. In this subsection, we consider an $\mathcal{I}_{\mathrm{g}}$ intruder system with $\mathcal{I}_{\mathrm{g}} = \langle \mathrm{g}, \mathrm{g}(x_1, x_2, x'_1, x'_2), \emptyset \rangle$. This intruder has at its disposal all ground instances of the following deduction rule:

$$x_1, x_2, y_1, y_2 \rightarrow \mathrm{g}(x_1, x_2, y_1, y_2)$$

The proof of the following theorem consists in first proving the existence of well-formed derivations for the standard subterm relation in the spirit of [16], with the additional simplification that all rules are composition rules. One then guesses a minimal attack in non-deterministic polynomial time. Since the $\mathcal{I}_{\mathrm{g}}$ and $\mathcal{I}_{\mathrm{f}}$ intruder are isomorphic the result also applies to $\mathcal{I}_{\mathrm{f}}$ after renaming of the symbol g into f.

Theorem 3 *Ordered satisfiability for the $\mathcal{I}_{\mathrm{g}}$ intruder system is decidable.*

Decidability of reachability for the AU-intruder. We now give a proof sketch for the decidability of ordered satisfiability for the $\mathcal{I}_{\mathrm{AU}}$ intruder since the procedure is new.

Theorem 4 *Ordered satisfiability for the $\mathcal{I}_{\mathrm{AU}}$ intruder system is decidable.*

Proof. The algorithm proceeds as follows:

– Transform the deduction constraints $E \rhd v$ into an ordering constraint $<_d$;
– Check that $<=<_d \cup <_i$ is still a partial order on atoms of $\mathcal{C}$;
– Solve the unification problem with linear constant restriction $<$.

Let $\mathcal{C} = ((E_i \rhd v_i)_{0 \leq i \leq n}, \mathcal{S})$ be a deterministic constraint system for the $\mathcal{I}_{\mathrm{AU}}$ intruder, $<_i$ be a (partial) order on $\mathrm{Cons}(\mathcal{C}) \cup \mathrm{Var}(\mathcal{C})$, and let σ be a solution of the $(\mathcal{C}, <_i)$ ordered satisfiability problem.

Given a set of terms $E \subseteq \mathrm{T}(\mathcal{F}_{\mathrm{AU}}, \mathcal{X})$, let us denote $\mathrm{K}_{\mathcal{C}} = (\mathrm{Cons}(\mathcal{C}) \setminus \mathrm{letters}(E)) \setminus \mathcal{X}$. In plain words, $\mathrm{K}_{\mathcal{C}}(E)$ is the set of constants in $\mathcal{C}$ **not** occurring in E. We are now ready to define $<_d$ as a partial order on $\mathrm{Cons}(\mathcal{C}) \cup \{v_0, \ldots, v_n\}$: We set $v_i <_d c$ for all constants c in $\mathrm{K}_{\mathcal{C}}(E_i)$.

Claim. For all σ, we have $\sigma \models (\mathcal{C}, <_i)$ if, and only if, $\sigma \models (\mathcal{S}, <_i \cup <_d)$

Proof of the claim. Let us first prove the direct implication. Let σ be a ground solution of the $(\mathcal{C}, <_i)$ ordered satisfiability problem. By definition we have that σ is a solution of $(\mathcal{S}, <_i)$ ordered unifiability problem. Since for all $0 \leq i \leq n$ we have $\sigma \models E_i \rhd v_i$, we easily see that $\mathrm{letters}((v_i\sigma)\downarrow) \subseteq \mathrm{Cons}(E_i)$, and therefore $\mathrm{letters}((v_i\sigma)\downarrow) \cap \mathrm{K}_{\mathcal{C}}(E_i) = \emptyset$. Thus σ is also a solution of $(\mathcal{S}, <_d \cup <_i)$. Conversely, assume now that σ is a ground solution of $(\mathcal{S}, <_d \cup <_i)$. By definition for all $0 \leq i \leq n$ we have $\mathrm{letters}((v_i\sigma)\downarrow) \cap \mathrm{K}_{\mathcal{C}}(E_i) = \emptyset$, and thus $\mathrm{letters}((v_i\sigma)\downarrow) \subseteq \mathrm{letters}(E_i) \setminus \mathcal{X}$. Thus we have $(v_i\sigma)\downarrow \in \overline{(E_i\sigma)\downarrow}$ for all $0 \leq i \leq n$, and thus $\sigma \models (\mathcal{C}, <_i)$ $\Diamond$

Since unifiability with linear constant restriction is decidable for the *AU* equational theory [18], this finishes the proof of the theorem. Note that the exact complexity is not known, but the problem is NP-hard and solvable in PSPACE [13, 14], and it is conjectured to be in NP [15, 11]. □

6 Conclusion

We have presented here a novel decision procedure for the search for attacks on protocols employing hash functions subject to collision attacks. Since this procedure is of practical interest for the analysis of the already normalised protocols relying on these weak functions, we plan to implement it into an already existing tool, CL-Atse. We also plan to formalise according to the model of [3] the underlying *AU* intruder system. In order to model hash functions we have introduced new symbols to denote the ability to create messages with the same hash value. This introduction amounts to the skolemisation of the equational property describing the existence of collisions We believe that this construction can be extended to model the more complex and game-based properties that appear when relating a symbolic and a concrete model of cryptographic primitives.

References

1. M. Baudet. Random polynomial-time attacks and Dolev-Yao models. In Siva Anantharaman, editor, *Proceedings of the Workshop on Security of Systems: Formalism and Tools (SASYFT'04)*, Orléans, France, June 2004.

2. E. Biham and R. Chen. Near-collisions of sha-0. In M. K. Franklin, editor, *CRYPTO*, volume 3152 of *LNCS*, pages 290–305. Springer, 2004.
3. Y. Chevalier and M. Rusinowitch. Combining intruder theories. In L. Caires, G. F. Italiano, L. Monteiro, C. Palamidessi, and M. Yung, editors, *ICALP*, volume 3580 of *LNCS*, pages 639–651. Springer, 2005.
4. Y. Chevalier and M. Rusinowitch. Hierarchical combination of intruder theories. In Frank Pfenning, editor, *RTA*, volume 4098 of *LNCS*, pages 108–122. Springer, 2006.
5. R. Cramer, editor. *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, volume 3494 of *LNCS*. Springer, 2005.
6. B. den Boer and A. Bosselaers. Collisions for the compressin function of md5. In *EUROCRYPT*, pages 293–304. Springer, 1993.
7. H. Dobbertin. Cryptanalysis of md4. In D. Gollmann, editor, *Fast Software Encryption*, volume 1039 of *LNCS*, pages 53–69. Springer, 1996.
8. H. Dobbertin. Cryptanalysis of md5 compress. Presented at the rumps session *of Eurocrypt'96*, 1996.
9. V. Klìma. Finding md5 collisions - a toy for a notebook, 2005. Cryptology ePrint Archive, Report 2005/075. http://eprint.iacr.org/.
10. V. Klìma. Finding md5 collisions on a notebook pc using multi-message modificatons, 2005. Cryptology ePrint Archive, Report 2005/102. http://eprint.iacr.org/.
11. Kim Guldstrand Larsen, Sven Skyum, and Glynn Winskel. Automata, languages and programming, 25th international colloquium, icalp'98, aalborg, denmark, july 13-17, 1998, proceedings. In *ICALP*, volume 1443 of *LNCS*. Springer, 1998.
12. C. Meadows. The NRL protocol analyzer: an overview. *Journal of Logic Programming*, 26(2):113–131, 1996.
13. W. Plandowski. Satisfiability of word equations with constants is in pspace. In *FOCS*, pages 495–500, 1999.
14. W. Plandowski. Satisfiability of word equations with constants is in pspace. *J. ACM*, pages 483–496, 2004.
15. W. Plandowski and W. Rytter. Application of lempel-ziv encodings to the solution of words equations. In *ICALP*, pages 731–742, 1998.
16. M. Rusinowitch and M. Turuani. Protocol insecurity with finite number of sessions is NP-complete. In *Proc.14th IEEE Computer Security Foundations Workshop*, Cape Breton, Nova Scotia, June 2001. IEEE Press.
17. Y. Sasaki, Y. Naito, N. Kunihiro, and K. Ohta. Wang's sufficient conditions of md5 are not sufficient, 2005. http://eprint.iacr.org/.
18. K. U. Schulz. Makanin's algorithm for word equations - two improvements and a generalization. In K. U. Schulz, editor, *IWWERT*, volume 572 of *LNCS*, pages 85–150. Springer, 1990.
19. X. Wang, D. Feng, X. Lai, and H. Yu. Collisions for hash functions md4, md5 , haval-128 and ripemd. http://eprint.iacr.org/, 2004.
20. X. Wang, X. Lai, D. Feng, H. Chen, and X. Yu. Cryptanalysis of the hash functions md4 and ripemd. In Cramer [5], pages 1–18.
21. X. Wang, Y. L. Yin, and H. Yu. Finding collisions in the full sha-1. In V. Shoup, editor, *CRYPTO*, volume 3621 of *LNCS*, pages 17–36. Springer, 2005.
22. X. Wang and H. Yu. How to break md5 and other hash functions. In Cramer [5], pages 19–35.
23. J. Yajima and T. Shimoyama. Wang's sufficient conditions of md5 are not sufficient, 2005. http://eprint.iacr.org/.