Relation between Unification Problem and Intruder Deduction Problem

Pascal Lafourcade

Verimag, University of Grenoble, CNRS UMR 5104
Centre Equation, 2 avenue de Vignate
38610 Gières, France
pascal.lafourcade@imag.fr

Abstract

Intruder deduction problem constitutes the first step in cryptographic protocols verification for a passive intruder. In the case of an active intruder, we know that undecidability of the unification problem implies undecidability of the secrecy problem. In this paper, we analyze the link between the unification problem and the intruder deduction problem. Through examples using equational theories, we show that these two problems are not linked. We present situations where one problem is decidable and the other one is not, or the both are decidable or not. All these examples prove that the two problems are independent for a passive intruder.

Keywords: Decidability, unification, rewriting, verification, cryptographic protocols, intruder deduction problem, secrecy.

1 Introduction

There are different approaches for modeling cryptographic protocols and analyzing their security properties. One of them was introduced by Dolev and Yao in [DY83]. This approach consists in modeling the attacker capabilities by a deduction system. This method is used to analyze the security of protocols against a *passive* attacker, *i.e* an intruder which obtains some information by eavesdropping on the network the communication between honest participants and deduces some information from these messages. The question whether a passive attacker can obtain a certain secret informations from observed messages on the network is called the *intruder deduction problem*.

The Dolev-Yao model has been extended by several equational theories to analyze in a more realistic way the cryptographic protocols. These new models allows searchers to find new attacks (see [CDL06] for a survey) by developing solutions to the intruder deduction problem modulo an equational theory for in-

[1] This work was supported by ANR SeSur SCALP, SFINCS and AVOTE.

stance, for *exclusive-or*, Abelian groups [CLS03,CKRT03], a homomorphism symbol alone [CLT03], and combinations of these theories [LLT05,Del06a,CR05].

We note that small changes in equational theory can turn decidability into undecidability as it has been observed for the active case with *exclusive-or* and homomorphism. With this equational theory the secrecy problem is decidable [DLLT06], but with Abelian group and homomorphism it is undecidable [Del06b].

Recently in [ANR07], the authors investigate sufficient decidability conditions on the rewriting system modeling the intruder's abilities to ensure if there exists a *cap* solving the intruder deduction problem (a cap consists of all possible actions doable by an intruder on eavesdropped terms). In Section 6, they obtain undecidability for a cap with one hole, *i.e.* the intruder cannot reuse terms. It shows that even in a restricted case it is not obvious to solve the intruder deduction problem.

We also notice that in the active case, as we explain in [CDL06], undecidability of the unification problem implies undecidability of the secrecy problem. The idea of this result is that we can build a protocol between two agents A and B. First A sends to B a set of n messages. If B is able, using the n received messages, to produce a message composed of the encryption of the two terms $u[x_1, \dots, x_n]$ and $v[x_1, \dots, x_n]$, and if A receives a message composed of two identical term then he sends the secret on the network. Hence the secret is published only if B solved the unification problem between u and v.

One can imagine that the link between unification which has been exhibited for the active case, already exist in the intruder deduction problem. By consequence we are exploring in this paper if there is for a passive case any relation between the unification problem and the intruder deduction problem.

Contributions:

We present first an extended Dolev-Yao system modulo an equational theory. Then we analyze the link between the intruder deduction problem and the unification problem. We recall the locality method which is the main technique used in the intruder deduction problem. Through examples using equational theories, we show that these two problems are not linked. We present situations where one problem is decidable and the other one is not, or both are decidable or not. All these examples prove that the two problems are independent in the case of a passive intruder.

Outline:

In the next section, we introduce some notations used in the rest of the paper, then in Section 3 we present the Dolev-Yao intruder deduction system classically used and a natural extension for equational theory. In Section 4, we consider the case where the intruder deduction problem is decidable and show that the unification can be decidable or not using two well-known examples. In Section 5, we construct two equational theories where the intruder deduction problem is undecidable and the unification is decidable in one example and not in the another one. In last section we conclude by summarizing our results.

2 Preliminaries

We give basic notations used here, see [DJ90,BN98] for an overview of rewriting.

Terms: Let Σ be a signature. $T(\Sigma, X)$ denotes the set of terms over the signature Σ and the set of variables X, that is the smallest set such that

(i) $X \subseteq T(\Sigma, X)$;

(ii) if $t_1, \ldots, t_n \in T(\Sigma, X)$, and $f \in \Sigma$ has arity $n \geq 0$, then $f(t_1, \ldots, t_n) \in T(\Sigma, X)$.

We abbreviate $T(\Sigma, \emptyset)$ as $T(\Sigma)$; elements of $T(\Sigma)$ are called Σ-*ground terms*. The set of variables occurring in a term t is denoted by $Vars(t)$.

The *set of occurrences* of a term t is defined recursively as $\mathcal{O}(f(t_1, \ldots, t_n)) = \{\epsilon\} \cup \bigcup_{i=1\ldots n} i \cdot \mathcal{O}(t_i)$. For instance, $\mathcal{O}(f(a, g(b, x))) = \{\epsilon, 1, 2, 21, 22\}$. The *size* $|t|$ of a term t is defined as its number of occurrences, that is $|t| = \mathit{cardinality}(\mathcal{O}(t))$. If $o \in \mathcal{O}(t)$ then the *subterm of t at position o* is defined recursively by

- $t\mid_\epsilon = t$
- $f(t_1, \ldots, t_n)\mid_{j \cdot o} = t_j\mid_o$

$St(t)$ is the set of *subterms* of the term t, that is $St(t) = \{t\mid_o \mid o \in \mathcal{O}(t)\}$. We also call it syntactic subterm because there is no equational theory used.

Equations and Rewriting Systems:

A Σ-*equation* is a pair $(l, r) \in T(\Sigma, X)$, commonly written as $l = r$. The relation $=_E$ generated by a set of Σ equations E is the smallest congruence on $T(\Sigma)$ that contains all ground instances of all equations in E.

A Σ-*rewriting system* R is a finite set of so-called *rewriting rules* $l \rightarrow r$ where $l \in T(\Sigma, X)$ and $r \in T(\Sigma, Vars(l))$. A term $t \in T(\Sigma, X)$ *rewrites* to s in one step by R if there is a rewriting rule $l \rightarrow r$ in R, an occurrence o and a substitution σ such that $t\mid_o = l\sigma$ and $s = t[o \leftarrow r\sigma]$. We write $\rightarrow^*$ for the reflexive and transitive closure of $\rightarrow$. A term t is in *normal form* if there is no term s with $t \rightarrow s$. If $t \rightarrow^* s$ and s is a normal form then we say that s is a *normal form of* t, and write $s = t\downarrow$, or $t \rightarrow^! s$. A term rewriting system is called *convergent* if it is

- *terminating*, that is if there is no infinite sequence of the form $t_1 \rightarrow t_2 \rightarrow t_3 \rightarrow \cdots$.
- *locally confluent*, that is if $t \rightarrow s_1$ and $t \rightarrow s_2$ then there exists a term r with $s_1 \rightarrow^* r$ and $s_2 \rightarrow^* r$.

Every convergent rewriting system is *confluent*, that is if $t \rightarrow^* s_1$ and $t \rightarrow^* s_2$ then there exists a term r with $s_1 \rightarrow^* r$ and $s_2 \rightarrow^* r$ (see, e.g., [DJ90]). Hence in a convergent rewriting system every term has a unique normal form.

By R/S we denote the so-called *class rewriting system* composed of a set $R = \{l_i \rightarrow r_i\}$ of rewriting rules and a set $S = \{u_i = v_i\}$ of equations. Generalizing the notion of term rewriting, we say that s rewrites to t *modulo* S, denoted $s \rightarrow_{R/S} t$, if $s =_S u[l\sigma]_p$ and $u[r\sigma]_p =_S t$, for some context u, position p in u, rule $l \rightarrow r$ in R, and substitution σ.

Definition 1 *We write $A \subseteq_{fin} B$ if:*

- *$A \subseteq B$*
- *A is a finite set*

3 Extended Dolev-Yao Model and Locality Result

We first present the classic model of deduction rules introduced by Dolev and Yao [DY83] in order to model the deductive capacities of a passive intruder. In this model, an intruder may use any term he has previously observed on the network, and construct new terms by pairing, unpairing, using a free constructor, encryption and decryption, where in the last two cases the intruder also has to know the key. Notice we only consider symmetric encryption (our results can be easily transferred to the case of asymmetric encryption).

Our aim in this section is to extend this model by an equational theory. We give a first variant of the Dolev-Yao model extended by equational reasoning. The extension consists of a rule for passing from one term to a term which is equivalent in the equational theory. Then we present a more effective variant of the extended Dolev-Yao model for the case where the equational theory can be presented by a convergent term rewriting system modulo a background equational theory. In this case we can work with normal forms of terms modulo the background theory, instead of allowing for unrestricted equational reasoning modulo the equational theory. In Theorem 1, we prove that these two models are indeed equivalent. The material follows the presentation of [CLT03], but is here extended to the case of an additional background equational theory.

3.1 *The Dolev-Yao Model Extended by Equational Reasoning*

Let Σ be a finite signature which can be partitioned as $\Sigma = \{\langle\cdot,\cdot\rangle, \{\cdot\}_\cdot\} \uplus \Sigma^-$. We write $A \uplus B$ for the union of two *disjoint* sets A and B. The signature Σ consists of pairing $\langle\cdot,\cdot\rangle$, encryption $\{\cdot\}_\cdot$, and some set Σ^- of so-called free function symbols. Let E be an equational theory over the signature Σ.

We use sequents of the form $T \vdash_E w$, where $T \subseteq_{\text{fin}} \mathcal{T}(\Sigma)$ is a finite subset and $w \in \mathcal{T}(\Sigma)$ the free terms algebra described by Σ. The intended meaning of such a sequent $T \vdash_E w$ is that an intruder with a certain set of deduction capabilities can deduce the term w from his knowledge T and using the equational theory E. In the context of cryptographic protocols, T is typically a set of messages that an intruder has previously observed on a network. Different deduction capabilities can be defined by different deduction systems for these sequents.

The classic Dolev-Yao model [DY83] defines the deduction capacities of an intruder assuming perfect cryptography. This deduction system is composed of the following rules: (A) the intruder knows any term that he has previously observed, (P) the intruder can build a pair of two messages, (UL, UR) he can extract each member of a pair, (C) he can encrypt a message m with a key k, (D) if he knows a key k he can decrypt a message encrypted by the same key, (F) he can construct a new term using a free function symbol $f \in \Sigma^-$.

Finally, we give to the intruder the power to use equational reasoning modulo a given set E of equational axioms by the rule (Eq). The resulting set of deduction rules is given in Figure 1.

Definition 2 (Proof) *A* Σ, E-sequent *is an expression of the form* $T \vdash_E u$ *where* E *is an equational theory over* Σ*,* $T \subseteq_{fin} \mathcal{T}(\Sigma)$*, and* $u \in \mathcal{T}(\Sigma)$*.*

$$(A)\frac{u \in T}{T \vdash_E u} \qquad (UL)\frac{T \vdash_E \langle u, v\rangle}{T \vdash_E u}$$
$$(P)\frac{T \vdash_E u \qquad T \vdash_E v}{T \vdash_E \langle u, v\rangle} \qquad (UR)\frac{T \vdash_E \langle u, v\rangle}{T \vdash_E v}$$
$$(C)\frac{T \vdash_E u \qquad T \vdash_E v}{T \vdash_E \{u\}_v} \qquad (D)\frac{T \vdash_E \{u\}_v \qquad T \vdash_E v}{T \vdash_E u}$$
$$(Eq)\frac{T \vdash_E u \qquad u =_E v}{T \vdash_E v} \qquad (F)\frac{T \vdash_E u_1 \qquad \dots \qquad T \vdash_E u_n}{T \vdash_E f(u_1, \dots, u_n)}$$

Fig. 1. Dolev-Yao System extended equations theory E

A proof *of a Σ, E-sequent $T \vdash_E u$ is a tree whose nodes are labeled by either Σ, E-sequents or expressions of the form "$v \in T$", such that:*

- *Each leaf is labeled by an expression of the form $v \in T$, and each non-leaf node is labeled by an Σ, E-sequent.*
- *Each node labeled by a sequent $T \vdash_E v$ has n children labeled by $T \vdash_E s_1, \dots, T \vdash_E s_n$ such that there is an instance of an inference rule with conclusion $T \vdash_E v$ and hypotheses $T \vdash_E s_1, \dots, T \vdash_E s_n$.*
- *The root of the tree is labeled by $T \vdash_E u$.*

Example 3.1 From the initial knowledge $T = \{\{m\}_k, k\}$ an intruder can learn the message m with the following deduction:

$$(\mathrm{D})\ \frac{(\mathrm{A})\ \dfrac{\{m\}_k \in T}{T \vdash_E \{m\}_k} \quad (\mathrm{A})\ \dfrac{k \in T}{T \vdash_E k}}{T \vdash_E m}$$

3.2 The Dolev-Yao Model Extended by Rewriting

The above model is not appropriate for automated proof search since the Eq(E) rule allows equational reasoning at any moment of a proof. In order to define a more effective model, we split the equational theory E into a *background theory* S and a rewrite system R.

Definition 3 (Rewrite Presentation of an Equational Theory) *Let E and S be equational theories over a signature Σ, and R a Σ-term rewriting system. (R, S) is a* rewrite presentation *of E if and only if*

- *R is locally confluent modulo S*
- *R is terminating modulo S*
- *For all closed Σ-terms u, v : $u =_E v$ iff $u \downarrow_{R/S} =_S v \downarrow_{R/S}$.*

In this case we can consider a variant of the extended model of Dolev-Yao defined previously which works on the normal form modulo S of the term at each step of the proof. The idea is that equivalence modulo S is easy to decide, such that we may omit the Eq(S) rule and just work with equivalence classes modulo S. We have to verify that the rewriting systems associated to each of all equational theories are confluent and terminating (we can for instance use the rewriting tool CiME [CM96]).

To define the right notion of a normal form we need that the term rewriting system modulo the background equational theory is convergent. Notice that local confluence and termination modulo an equational theory of the term rewriting system imply its convergence. We now define normal forms for a such system.

Definition 4 (Term in Normal Form) *Let (R, S) be a rewrite presentation of some equational theory E. A term t is in* normal form *if there is no term v such that $t \rightarrow_{R/S} v$ (t reduces to v by a rule of the rewriting system of R modulo S). If $t \rightarrow^* v$ and v is in normal form then we call v the normal form of t, denoted $v = t\downarrow$.*

Note that normal forms are unique only up to S-equivalence. Normal forms have the following properties:

- $\forall u, v : u =_E v \Rightarrow u\downarrow =_S v\downarrow$
- $\forall u : u =_E u\downarrow$

Remark: if $t[.]$ is a context and u a ground term then $t[u\downarrow]\downarrow =_S t[u]\downarrow$. In particular $f(u_1, \ldots, u_n)\downarrow =_S f(u_1\downarrow, \ldots, u_n\downarrow)\downarrow$. We omit the rule (Eq(E)) and consider a new system $\vdash$ presented in Figure 2 which only works on normal forms.

$$(A)\frac{u \in T}{T \vdash u\downarrow} \qquad (UL)\frac{T \vdash r}{T \vdash u\downarrow} if \langle u, v\rangle \rightarrow^! r$$

$$(P)\frac{T \vdash u \qquad T \vdash v}{T \vdash \langle u, v\rangle\downarrow} \qquad (UR)\frac{T \vdash r}{T \vdash v\downarrow} if \langle u, v\rangle \rightarrow^! r$$

$$(C)\frac{T \vdash u \qquad T \vdash v}{T \vdash \{u\}_v\downarrow} \qquad (D)\frac{T \vdash r \qquad T \vdash v}{T \vdash u\downarrow} if \{u\}_v \rightarrow^! r$$

$$(F)\frac{T \vdash u_1 \qquad \ldots \qquad T \vdash u_n}{T \vdash f(u_1, \ldots, u_n)\downarrow}$$

Fig. 2. A Dolev-Yao proof system working on normal forms modulo a background equational theory.

Theorem 1 *Let (R, S) be a rewrite presentation of the equational theory E, $T \subseteq_{fin} \mathcal{T}(\Sigma)$, and $T \in \mathcal{T}(\Sigma)$. If the equational theory has the following property $\langle u, v\rangle\downarrow = \langle u\downarrow, v\downarrow\rangle$ and $\{u\}_v\downarrow = \{u\downarrow\}_{v\downarrow}$, then we have that:*

$$T \vdash_E u \qquad \textit{if and only if} \qquad T \vdash u\downarrow$$

Proof. The two properties $\langle u, v\rangle\downarrow = \langle u\downarrow, v\downarrow\rangle$ and $\{u\}_v\downarrow = \{u\downarrow\}_{v\downarrow}$ assure that the pair and encryption symbols functions do not disappear using the equational theory in the rewriting system. They are used in the proof of the case of the rules (C) and (P).

Given a proof of $T \vdash u\!\downarrow$ we can easily find a proof of $T \vdash_E u$ by inserting Eq(E)-steps.

For the other direction we transform a proof in $\vdash_E$ into a proof in $\vdash$ by the transformations of Figure 3. These transformations do not change the leaves of a proof tree. We show by induction that if there is a proof of $T \vdash_E u$ then the transformation yields a proof of $T \vdash u\!\downarrow$.

$$\text{(Eq(E))}\ \frac{\text{(R)}\ \dfrac{\psi_1}{T \vdash_E u} \qquad u =_E v}{T \vdash_E v} \quad \Longrightarrow \quad \text{(R)}\ \frac{\psi_1}{T \vdash v\!\downarrow}$$

$$\text{(R)}\ \frac{\text{(R}_1\text{)}\ \dfrac{\psi_1}{T \vdash_E u_1} \quad \cdots \quad \text{(R}_n\text{)}\ \dfrac{\psi_n}{T \vdash_E u_n}}{T \vdash_E v} \quad \Longrightarrow \quad \text{(R)}\ \frac{\text{(R}_1\text{)}\ \dfrac{\psi_1}{T \vdash_E u_1} \quad \cdots \quad \text{(R}_n\text{)}\ \dfrac{\psi_n}{T \vdash_E u_n}}{T \vdash v\!\downarrow}$$

Fig. 3. Transformations of a proof of $T \vdash_E u$ into a proof of $T \vdash u\!\downarrow$.

We proceed by case distinction on the last deduction rule:

- (A): obvious.
- (Eq(E)): Since (R, S) is a rewrite presentation of E we get $u\!\downarrow = v\!\downarrow$ (modulo S) so we obtain a proof of $T \vdash u\!\downarrow$
- (P), (C) or (F): by induction hypothesis on all the hypotheses of the rule and with the fact $f(u_1, \ldots, u_n)\!\downarrow = f(u_1\!\downarrow, \ldots, u_n\!\downarrow)\!\downarrow$, where f can be encryption or pairing symbol, hence we get the result.
- (D), by induction $T \vdash \{u\}_v\!\downarrow$ and $T \vdash v\!\downarrow$, with $\{u\}_{v\downarrow} = \{u\}_{v\downarrow}\!\downarrow$ and the rule (D) we get $T \vdash u\!\downarrow$, hence a proof in $\vdash$.
- (UL) or (UR) by induction we obtain the result.

□

In the following we always work with the system $\vdash$ which uses on normal forms modulo an equational theory.

4 Intruder deduction problem is decidable

We present here two examples, where the intruder deduction problem is decidable modulo an equational theory, whereas the associated unification problem is decidable in the first example and not in the second one.

Our starting point is the locality technique introduced by McAllester [McA93], for example used in [CLS03,CKRT03]. McAllester shows that there exists an algorithm to decide the deducibility of a term w from a finite set of terms T, if the deduction system has the so-called *locality property*. A deduction system has the *locality property* if any proof can be transformed into a *local proof*, that is a proof where all nodes are in the set of syntactic subterms of $T \cup \{w\}$ denoted $St(T \cup \{w\})$. The idea of the proof is to check the existence of a local proof by a saturation algorithm which computes all subterms of $T \cup \{w\}$ that are deducible from T, if the number of rules is finite. In [LLT05], we have extended McAllester's approach to take into account some equational theories by generalizing syntactic subterm to a new notion of subterm. The difficulty resides in designing the right subterm function for a given equational theory and proving the existence of a local proof.

4.1 Unification problem is decidable

We consider the empty theory. It is know that the unification modulo this equational theory is decidable [Her30,Rob65,CB83,MM82,PW78]. Thus only Dolev-Yao intruder system has to be considered, as we know in this case the intruder deduction problem is decidable. Here, we only recall the main idea of the proof (see [Laf06] for more details). The usual syntactic subterms are enough to prove the locality result by induction for a term w and a set of hypothesis T considering minimal proof. By using the McAllester's result, we deduce the decidability of the intruder deduction problem. As the size of the subterm is computable in polynomial time in the size of the inputs w and T, and all rules of the Dolev-Yao deduction system can be applied in polynomial time, we have a polynomial time procedure to solve the intruder deduction problem without an equational theory.

4.2 Unification problem is undecidable

We consider the equational theory where we have an associative and commutative operator plus a homomorphic symbol h over the exclusive-or symbol. In [Nar96] P. Narendran shows by coding the tenth Hilbert problem that unification modulo this equational theory is undecidable.

In [LLT05,Del06a] the intruder deduction problem modulo this equational theory was shown decidable. In [LLT05], we first provide an exponential procedure to solve this problem using two construction rules, one for the homomorphism symbol and another one for the exclusive-or. The set of subterms is exponential in the size of the inputs. Indeed, we have to consider all possible "sums" and very special minimal proofs called *eager*, because we try to apply as eagerly as we can the exclusive or operator. Finally, in [Del06a], S. Delaune improved our result by considering an unique rule for these two operators. In this approach, intermediary sums do not appear in the proof tree and considering the minimal proof is enough to obtain a polynomial set of subterms and the locality result. The difficulty is pushed to the application of a rule, composed of combinations of the homomorphism symbol and the exclusive-or one. This problem is now solved by equation systems in ring of polynomials over $Z/2Z$, using mathematical results [KKS87,Sch86].

5 Intruder deduction problem is undecidable

In this section, we design two examples of equational theories in which intruder deduction problem is undecidable and then associated unification problem is either decidable or undecidable.

We first construct an equational theory called E where the so-called word problem is undecidable, *i.e.* knowing if two ground terms (*i.e.* without variables) are equal modulo this equational theory is undecidable. The word problem modulo an equational theory is a special case of the unification problem modulo this theory. Thus, if the word problem is undecidable then the unification problem is also undecidable.

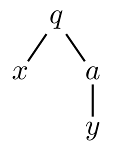
5.1 An undecidable problem

We recall first that an instance of the word problem is composed of two ground terms t_1 and t_2, and a set of equations E. The problem is to find out if using equations in E we can establish the equality between the two given terms (*i.e.* if $t_1 =_E t_2$). We construct an equational theory E such that the word problem is undecidable modulo E.

A deterministic Turing machine is defined by $M = (Q, \Sigma, \square, \delta, q_0, q_f)$ where:

- Q is a finite set of states.
- Σ is a finite alphabet for the tape.
- $\square \in \Sigma$ is the "empty" symbol (it can appear infinitely often on the tape).
- $\delta : Q \times \Sigma \to Q \times \Sigma \times \{L, R, 0\}$ is a partial function called transition function, where L corresponds to a shift of the tape on the left, R a right shift and 0 no move of the tape.
- $q_0 \in Q$ is the initial state.
- $q_f \in Q$ is the unique accepting state.

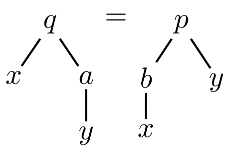
A configuration of a Turing machine is represented by a term:



where the machine is in the state $q \in Q$, the tape on the left is represented by the variable $x \in Vars$, the tape on the right side is the symbol $a \in \Sigma$ and the variable $y \in Vars$. We denote this configuration by $q(x, a(y))$. We assume that no transition starts from a final configuration, *i.e.* a configuration which contains the final state q_f. A Turing machine stops if the final configuration is reached.

Assuming that a Turing Machine is given, for each transition δ of this Turing machine we associate an equation of E. We now explain in detail the different equations of E derived from the transitions of the Turing machine. For all $a, b, \epsilon \in \Sigma$, $p, q \in Q$, and $x, y \in Vars$:

- transition $(q, a) \rightarrow (p, b, R)$ gives, with a move to the right, the following equation:



  We denote this shortly by: $q(x, a(y)) = p(b(x), y)$.
- transition $(q, a) \rightarrow (p, b, L)$ gives the equation: $\forall f \in \Sigma, q(f(x), a(y)) = p(x, f(b(y)))$
- transition $(q, a) \rightarrow (p, b, 0)$ gives equation: $q(x, a(y)) = p(x, b(y))$
- We add the following equation $q_f(x, y) = t_2$ once final state reached.

Notice that a and f can be the empty symbol. The term t_1 represents the following initial configuration $q_0 = (\epsilon, w_1(\ldots(w_n(\epsilon)))))$, where $w_i \in \Sigma$. The ground term t_2 is associated to the final configuration $q_f(x, y)$ of the Turing machine. We summarize the different equations of E in Figure 4.

$$E = \begin{cases} q(x, a(y)) &= p(b(x), y) \\ q(f(x), a(y)) &= p(x, f(b(y))) \\ q(x, a(y)) &= p(x, b(y)) \\ q_f(x, y) &= t_2 \end{cases}$$

Fig. 4. Equational theory E

We show now that the word problem $t_1 =_E t_2$ is equivalent to the halt of the Turing machine with the input associated to the configuration t_1.

Theorem 2 *Let t_1 and t_2 be two ground terms and E be the equational theory constructed above. Solving word problem $t_1 =_E t_2$ is equivalent to deciding the halt of the Turing machine with the input associated to the configuration t_1.*

*P*roof: $\Leftarrow$ By construction of E, if the Turing machine with the input associated to the term t_1 stops on an accepting state then $t_1 =_E t_2$.

$\Rightarrow$ Given an equation between two terms t_1 and t_2 using E, we orient from left to right all equations of E. We select the smallest sequence of equations applied from left to right to prove $t_1 =_E t_2$. This minimal sequence is finite and has no loops. We prove the result by induction on the size of this sequence.

- Base case: there is no equation between t_1 and t_2, *i.e.* t_1 and t_2 are syntactically equal. We obtain directly that the initial configuration is also the final configuration, hence the Turing machine stops.
- Induction: We consider a minimal sequence $t_1 =_E t' =_E \ldots =_E t^{(n-1)} =_E t_2$ of n equations between t_1 and t_2 . We first notice that the final configuration $q_f(x, y)$ represents the term t_2. Consider the first equation between t_1 and t', there are $n-1$ equations between t' and t_2. We can apply the induction hypothesis, hence the Turing machine stops with the initial configuration associated to t'. The equation between t_1 and t' has been produced from a transition, then with this

transition the Turing machine moves from the initial configuration associated to t_1, to a state from which it terminates by induction hypothesis.

5.2 Intruder deduction modulo E is undecidable.

We prove that the intruder deduction problem modulo equational theory E is undecidable. We reduce in Lemma 5.1 the word problem $t_1 =_E t_2$ to the intruder deduction problem modulo E defined in Figure 4.

Lemma 5.1 *Let t_1 and t_2 be two ground terms and E the equational theory defined in Figure 4. Solving word problem $t_1 =_E t_2$ is equivalent to solving the intruder deduction problem modulo E.*

Proof: We prove that $t_1 \vdash t_2 \Leftrightarrow t_1 =_E t_2$.

- ($\Leftarrow$) If $t_1 =_E t_2$ then we build with an axiom and the rule (E_q) a proof of $t_1 \vdash t_2$
- ($\Rightarrow$) Consider the minimal proof of $t_1 \vdash t_2$ (minimal in number of nodes). We analyze now the last rule of this proof case by case.
  - The term t_2 is not a pair neither an encryption, hence it can not have been generated by the rule (P), neither the rule (C).
  - Let the last rule be one of the rules (UR), (UL) or (D). The only term in the initial knowledge is the term t_1 which is not an encryption neither a pair. We also know that the equations in E have no pair symbol neither encryption symbol. Hence, the only possibility for having an application of one of the rules (UR), (UL), or (D) is to have an application of a constructor rule of pair or encryption on t_1. This implies that we can build a smaller proof of $t_1 \vdash t_2$ by cutting the proof between the constructor rules and the destructor rules leading and starting with the same term. Hence we have a contradiction with the minimality of the proof. We conclude that the last rule cannot be one of the following rule (UR), (UL) and (D).
  - If the proof ends by an application of the rule (E_q), we get a smaller proof ending on t_2', we can apply the induction hypothesis. Hence $t_1 =_E t_2'$. Using the equation applied in the rule (E_q), we obtain the result $t_1 =_E t_2' =_E t_2$.

5.3 Unification modulo E is undecidable.

Using Theorem 2, the word problem is undecidable for the equational theory E, by consequence the unification problem associated to this equational theory is undecidable. The equational theory E is our example which shows that the two problems can be at the same time undecidable.

Now we modify E to obtain a new equational theory E_s, where the unification is decidable and the intruder deduction problem is not.

5.4 Unification modulo E_s is decidable.

New Equational Theory

We modify the equational theory E defined previously by adding a new symbol function s, we obtain E_s described in Figure 5.

$$E_S = \begin{cases} s(q(x,a(y))) = p(b(x),y) \\ s(q(f(x),a(y))) = p(x,f(b(y))) \\ s(q(x,a(y))) = p(x,b(y)) \\ s(q_f(x,y)) = t_2 \end{cases}$$

Fig. 5. Equational theory E_s

5.5 *Intruder deduction modulo E_s is undecidable*

We use the following extended Dolev-Yao deduction system, modulo E_s, by giving the possibility to the intruder to apply the symbol s to any term:

$$(A)\ \frac{u \in T}{T \vdash_{E_s} u} \qquad (UL)\ \frac{T \vdash_{E_s} \langle u,v\rangle}{T \vdash_{E_s} u}$$

$$(P)\ \frac{T \vdash_{E_s} u \qquad T \vdash_{E_s} v}{T \vdash_{E_s} \langle u,v\rangle} \qquad (UR)\ \frac{T \vdash_{E_s} \langle u,v\rangle}{T \vdash_{E_s} v}$$

$$(C)\ \frac{T \vdash_{E_s} u \qquad T \vdash_{E_s} v}{T \vdash_{E_s} \{u\}_v} \qquad (D)\ \frac{T \vdash_{E_s} \{u\}_v \qquad T \vdash_{E_s} v}{T \vdash_{E_s} u}$$

$$(E_q)\ \frac{T \vdash_{E_s} u \qquad u =_{E_s} v}{T \vdash_{E_s} v} \qquad (S)\ \frac{T \vdash_{E_s} u}{T \vdash_{E_s} s(u)}$$

Fig. 6. Dolev-Yao deduction system extended by the equational theory (E_s).

The rewriting system R_{E_S} presented in Figure 7, obtained by orienting to the right the equations in E_s, terminates since the number of applications of s decreases and is confluent because there are no critical pairs. Using, results of Section 3, we can delete the rule (E_q) in the extended Dolev-Yao system of Figure 2 and work only using normal forms. In this new Dolev-Yao deduction system we can code an equivalent of the Turing machine used in the previous section.

Consider the previous Turing machine where we add the symbol on both sides of the transitions. This artifact does not affect the decidability of the halt of the Turing machine. We now show that with the intruder deduction problem modulo the equational theory E_s we can generate all the transitions of a deterministic Turing machine. The classical rules of Dolev-Yao deduction system, *i.e.* (P), (C), (D), (UL), and (UR), are not used to obtain from the term t_1 the term t_2. Indeed, the proof is only built with applications of the rules (S) and (Eq). Alternating the applications of these two rules we can construct all transitions of the equivalent Turing machine with symbol s on both sides. Thus, the intruder deduction problem modulo the equational theory E_s is also undecidable.

Unification modulo E_s is decidable.

We transform the equation system E_s into the rewriting system R_{E_S} presented in Figure 7.

The number of applications of the function s decreases, and consequently the rewriting system of Figure 7 terminates. As the construction of E_s is based on a

$$R_{E_s} = \begin{cases} s(q(x,a(y))) \to p(b(x),y) \\ s(q(f(x),a(y))) \to p(x,f(b(y))) \\ s(q(x,a(y))) \to p(x,b(y)) \\ s(q_f(x,y)) \to t_2 \end{cases}$$

Fig. 7. Rewriting system R_{E_S} associated to E_s.

deterministic Turing machine, there is no superposition between the terms. This implies that there are no critical pairs. We conclude that the rewriting system R_{E_S} is confluent and consequently convergent. We recall now the definition of the narrowing presented in [CK01].

Definition 5 (Narrowing) *A term t is* narrowed *into t', at the position non variable $p \in Dom(t)$, using the rewriting rule $l \to r$ and the substitution σ, where σ is the most general unifier of $t|_p$ and l, and $t' = \sigma(t[r]_p)$, denoted by $t \rightsquigarrow_{[p,l\to r,\sigma]} t'$. We always assume that there is no conflict of variables between rules and terms (this is always possible by renaming), i.e. $Vars(l,r) \cap Vars(t) = \emptyset$.*

For a given term rewriting system R, this generates a binary relation on terms called narrowing relation and denoted $\rightsquigarrow_R$.

Note that the narrowing is a natural extension of rewriting since unification is used instead of matching. Thus, the rewriting relation is always included in the narrowing one: $\to_R \subseteq \rightsquigarrow_R$, since, for terms with disjoint sets of variables, a match is always an unifier.

Example 5.2 If we consider the rule $f(f(x)) \to x$ then the term $f(y)$ narrows at the position Λ:

$$f(y) \rightsquigarrow_{[\Lambda, f(f(x))\to x, \{(x\to z),(y\to f(y))\}]} z$$

We notice on this example that narrowing may introduce new variables, due to the unification step.

According to the result by Hullot [Hul80], narrowing is complete for an equational theory represented by a convergent rewriting system (a clear presentation of this result is given in a paper of F. Baader and W. Snyder [BS01] and in the paper of A. Middeldorp [Mid94]). A consequence of this result is that if the narrowing terminates for a theory represented by a convergent rewriting system then associated unification problem is decidable.

We prove that narrowing terminates for the equational theory E_S. This implies decidability for the unification problem modulo this theory. Notice first that we have a finite number of rewriting rules in R_{E_s}. Hence, narrowing has a finite number of possibilities for possible applicable rules. Moreover, all left terms of the rules use the symbol function s and all our rules preserve the set of variables, ($Vars(l) = Vars(r)$). We consider the measure n_s that computes the number of symbol s in a term for proving the termination of the narrowing. Due to the particular form of our equalities this measure decreases: variables are preserved and the s symbol disappears. By construction of narrowing, σ is the most general unifier between $t|_p$

and l. We conclude that narrowing terminates hence the unification modulo E_s is decidable.

6 Conclusion

We have clearly established through examples that the unification problem modulo an equational theory and the intruder deduction problem modulo the same equational theory are two independent problems. We recall in Figure 8 the different equational theories exhibited in this paper for showing the independence of these two problems. [2]

		Intruder Deduction Problem	
		Decidable	Undecidable
Unification	Decidable	$\emptyset$	E_s
	Undecidable	ACh	E

Fig. 8. Summary of results and equational theories obtained for comparing unification problem and intruder deduction problem.

References

[ANR07] Siva Anantharaman, Paliath Narendran, and Michaël Rusinowitch. Intruders with caps. In Franz Baader, editor, *RTA*, volume 4533 of *Lecture Notes in Computer Science*, pages 20–35. Springer, 2007.

[BN98] Franz Baader and Tobias Nipkow. *Term Rewriting and All That*. Cambridge University Press, 1998.

[BS01] F. Baader and W. Snyder. Unification theory. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, volume I, chapter 8, pages 445–532. Elsevier Science, 2001.

[CB83] J. Corbin and M. Bidoit. A rehabilitation of robinson's unification algorithm. In *Proc. IFIP '83*, pages 909–914. North-Holland, 1983.

[CDL06] Véronique Cortier, Stéphanie Delaune, and Pascal Lafourcade. A survey of algebraic properties used in cryptographic protocols. *Journal of Computer Security*, 14(1):1–43, 2006.

[CK01] Hubert Comon and Claude Kirchner. Constraint solving on terms. *Lecture Notes in Computer Science*, 2002:47–103, 2001. Claude and Hlne Kirchner. Rewriting Solving Proving.

[CKRT03] Y. Chevalier, R. Küsters, M. Rusinowitch, and M. Turuani. An NP decision procedure for protocol insecurity with XOR. In *Proc. of 18th Annual IEEE Symposium on Logic in Computer Science (LICS'03)*, pages 261–270, Ottawa, Canada, 2003. IEEE Comp. Soc. Press.

[CLS03] Hubert Comon-Lundh and Vitaly Shmatikov. Intruder deductions, constraint solving and insecurity decision in presence of exclusive or. In *Proc. of 18th Annual IEEE Symposium on Logic in Computer Science (LICS'03)*, pages 271–280, Ottawa, Canada, 2003. IEEE Comp. Soc. Press.

[CLT03] Hubert Comon-Lundh and Ralf Treinen. Easy intruder deductions. In Nachum Dershowitz, editor, *Verification: Theory & Practice, Essays Dedicated to Zohar Manna on the Occasion of His 64th Birthday*, volume 2772 of *Lecture Notes in Computer Science*, pages 225–242. Springer-Verlag, 2003.

[CM96] Evelyne Contejean and Claude Marché. CiME: Completion Modulo E. In Harald Ganzinger, editor, *7th International Conference on Rewriting Techniques and Applications*, volume 1103 of *Lecture Notes in Computer Science*, pages 416–419, New Brunswick, NJ, USA, July 1996. Springer-Verlag.

[2] Thanks to Ralf Treinen for all helpful discussions we had.

[CR05] Yannick Chevalier and Michaël Rusinowitch. Combining intruder theories. In Luís Caires, Giuseppe F. Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung, editors, *ICALP*, volume 3580 of *Lecture Notes in Computer Science*, pages 639–651. Springer, 2005.

[Del06a] S. Delaune. Easy intruder deduction problems with homomorphisms. *Information Processing Letters*, 97(6):213–218, 2006.

[Del06b] Stéphanie Delaune. An undecidability result for AGh. Research Report LSV-06-02, Laboratoire Spécification et Vérification, ENS Cachan, France, February 2006. 9 pages.

[DJ90] Nachum Dershowitz and Jean-Pierre Jouannaud. Rewrite systems. In Jan van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B - Formal Models and Semantics, chapter 6, pages 243–320. Elsevier Science Publishers and The MIT Press, 1990.

[DLLT06] Stéphanie Delaune, Pascal Lafourcade, Denis Lugiez, and Ralf Treinen. Symbolic protocol analysis in presence of a homomorphism operator and *exclusive or*. In *Proceedings of the 33rd International Colloquium on Automata, Languages and Programming (ICALP'06)*, Lecture Notes in Computer Science, pages 132–143, Venice, Italy, jul 2006. Springer.

[DY83] D. Dolev and A.C. Yao. On the security of public-key protocols. In *Transactions on Information Theory*, volume 29, pages 198–208. IEEE Computer Society Press, March 1983.

[Her30] Jacques Herbrand. *Recherches sur la Théorie de la Démonstration*. PhD thesis, University of Paris, 1930.

[Hul80] J.-M. Hullot. Canonical forms and unification. In W. Bibel and R. Kowalski, editors, *Proceedings of the 5th Conference on Automated Deduction*, volume 87 of *Lecture Notes in Computer Science*, pages 318–334, Les Arcs, France, July 1980. springer.

[KKS87] E. Kaltofen, M. S. Krishnamoorthy, and B. D. Saunders. Fast parallel computation of hermite and smith forms of polynomial matrices. *SIAM J. Algebraic Discrete Methods*, 8(4):683–690, 1987.

[Laf06] Pascal Lafourcade. *Vérification des protocoles cryptographiques en présence de théories équationnelles*. Thèse de doctorat, Laboratoire Spécification et Vérification, ENS Cachan, France, September 2006. 209 pages.

[LLT05] Pascal Lafourcade, Denis Lugiez, and Ralf Treinen. Intruder deduction for AC-like equational theories with homomorphisms. In Jürgen Giesl, editor, *Proceedings of the 16th International Conference on Rewriting Techniques and Applications (RTA'05)*, volume 3467 of *Lecture Notes in Computer Science*, pages 308–322, Nara, Japan, April 2005. Springer-Verlag.

[McA93] David A. McAllester. Automatic recognition of tractability in inference relations. *Journal of the ACM*, 40(2):284–303, April 1993.

[Mid94] Aart Middeldorp. Completeness of combinations of conditional constructor systems. *Journal of Symbolic Computation*, 17(1):3–21, January 1994.

[MM82] A. Martelli and U. Montanari. An efficient unification algorithm. *ACM Transactions on Programming Languages and Systems*, 4(2):258–282, 1982.

[Nar96] Paliath Narendran. Solving linear equations over polynomial semirings. In *Proc. of 11th Annual Symposium on Logic in Computer Science (LICS'96)*, pages 466–472, July 1996.

[PW78] M.S. Paterson and M.N. Wegman. Linear unification. *Journal of Computer and System Sciences*, 17:348–375, 1978.

[Rob65] J. A. Robinson. A machine-oriented logic based on the resolution principle. *Journal of the ACM*, 12:23–41, 1965.

[Sch86] Alexander Schrijver. *Theory of Linear and Integer Programming*. Wiley, 1986.