# Normal proofs in intruder theories

Vincent Bernat and Hubert Comon-Lundh

LSV, Ecole Normale Supérieure de Cachan
94235 Cachan cedex, France
{bernat,comon}@lsv.ens-cachan.fr

**Abstract.** Given an arbitrary intruder deduction capability, modeled as an inference system $\mathcal{S}$ and a protocol, we show how to compute an inference system $\widehat{\mathcal{S}}$ such that the security problem for an unbounded number of sessions is equivalent to the deducibility of some message in $\widehat{\mathcal{S}}$. Then, assuming that $\mathcal{S}$ has some subformula property, we lift such a property to $\widehat{S}$, thanks to a proof normalisation theorem. In general, for an unbounded number of sessions, this provides with a complete deduction strategy. In case of a bounded number of sessions, our theorem implies that the security problem is co-NP-complete. As an instance of our result we get a decision algorithm for the theory of blind-signatures, which, to our knowledge, was not known before.

## 1 Introduction

Cryptographic protocols aim at achieving some security goal, while relying on a public network. Several such protocols have been designed and used in various applications. Many of them appeared to be flawed, not because of the weakness of cryptographic primitives, but simply because of the logical structure of the protocol (see the protocols repository [20] for examples). That is why several teams started to work on the formal verification of security protocols.

Though there are not so many different protocol designs, if one wants, say, to set a shared secret between two agents using a public key infrastructure, there are many small variants, which depend on the application domain. For instance, if a server has to manage thousands of users, it is mandatory to reduce both the time and space resources on its side and let the client do the job. Also, the companies (e.g. telecommunication companies) not only want to reduce the cost, but also to get robust protocols: if a key is compromised, this should not compromise the whole infrastructure. If possible, be resistant to guessing attacks on weak passwords... That is the reason why we need to *automate* the protocol verification: we can check many small variants of a same protocol.

Formally, what we need to achieve depends on several inputs: the protocol itself, but also what are the intruder capabilities, what are the known properties of cryptographic primitives and which property we want to prove. In the present paper, we only consider reachability properties: is some message deducible by the intruder after message exchanges that follow the protocol rules ?

Fixing an a priori bound on the number of messages which are sent through the network, several decision procedures have been designed, depending on the

intruder capabilities and the algebraic properties. Let us cite our main source: in [18] the authors prove that the security problem is co-NP complete for the so-called "Dolev-Yao" intruder, assuming no equational property (the message algebra is free). The proof in this paper relies on two properties: first the intruder deduction rules are *local* (according to [17]), meaning some sub-formula property: to prove $s$ from hypotheses $T$, we only need intermediate formulas in the subterms of $s, T$[1]. The other ingredient is similar: if there is an attack, there is one which binds variables to subterms of messages already sent at this point.

We aim at viewing these two ingredients as a single subformula property. In other words, we want to design a proof system and a normalisation theorem such that all formulas occurring in the proof are subterms of the protocol rules, the hypotheses and the conclusion, and variable substitutions are also bound in a similar way. In addition, we would like to abstract from the particular intruder capabilities, relying only on the good properties of the inference systems which allow to derive the normalisation theorem.

To achieve these goals, we show first in section 3 that, given any inference system $S$ describing intruder capabilities, an equational theory $E$ and a protocol $P$, we can compute a proof system $\widehat{S}$, such that the protocol $P$ is insecure if and only if some term is deducible in $\widehat{S}$. For further strategies design, formulas $\widehat{S}$ will be constrained by equations representing possible bindings of the free variables.

Then, our main result in section 4 is a proof normalisation result, which assumes additional hypotheses on the intruder capabilities and no equational properties. Normal proofs only involve terms that can be computed from the hypotheses, the conclusion and (instances of) the protocol rules. This result does not assume that the number of sessions is bounded. Hence we can use it as a proof search strategy. Moreover, if we fix the number of sessions, our main theorem shows that protocol security is then decidable, and co-NP-complete for intruder deduction systems that are in PTIME. As an example of application, we show in section 4.4 that, for the deduction system corresponding to blind signatures, the security problem is co-NP-complete. Security of protocols with blind signatures were also studied in [10], for an unbounded number of sessions, as an application of more general completeness results. In [10] the protocols are however approximated using a clausal formalism.

We also show in section 4.1 that our additional hypotheses on $S$ are necessary: dropping them yields undecidability, even for a fixed number of sessions. We also briefly compare in section 5 our result, when restricted to a bounded number of sessions to [21] and the so-called "oracle rules" (see also [3] for oracle rules in presence of exclusive or).

Our ultimate hope is to generalize the proof normalisation result to associative and commutative symbols. This would be a major step, since it is shown in [7] that many relevant equational theories can be reduced to associativy and commutativity. The price to pay is a modification of the inference system $S$ (and the protocol $P$). However, this transformation preserves the locality of $S$. And that is what we need in our theorem.

[1] A similar property was observed in [5] in the particular case of atomic keys.

## 2 Models of security protocols

We recall here a possible model of security protocols, which we use for proving the adequacy of our proof system in the next section. This model is close to many existing ones (e.g. [11, 2, 13]).

### 2.1 Messages

Messages are terms of an algebra, generated by a finite set of function symbols $\mathcal{F}$. Typically, $\mathcal{F}$ contains (a)symetric encryption (a binary symbol used in infix notation: $\{\_\}\_$), pairing (a binary symbol used in infix notation $< \_, \_ >$) and constants. It may also contain symbols for public keys, decryption functions, exclusive or, etc... If $X$ is a set of (first-order) variable symbols, $T(\mathcal{F}, X)$ is the set of terms built over $\mathcal{F}$ and $X$. $T(\mathcal{F})$ is an abbreviation for $T(\mathcal{F}, \emptyset)$. A *substitution* $\sigma$ is a mapping from a finite subset of $X$, called its *domain*, into $T(\mathcal{F}, X)$. As usual, such mappings are confused with their (unique) extension as an endomorphism of $T(\mathcal{F}, X)$. Substitutions are used in postfix notation. If $\sigma, \tau$ are substitutions, which coincide on $Dom(\sigma) \cap Dom(\tau)$, then $\sigma \uplus \theta$ is the substitution whose domain is $Dom(\sigma) \cup Dom(\theta)$ and coincides respectively with $\sigma$ and $\theta$ on their domains. Terms can also be seen as trees, i.e. mappings $t$ from a finite prefix-closed domain $Dom(t)$ to $\mathcal{F}$ (or $\mathcal{F} \cup X$). If $p \in Dom(t)$, $t|_p$ is the subterm of $t$ at position $p$, $t[u]_p$ is the term obtained by replacing $t|_p$ with $u$ in $t$. The *depth* of $t$ is the maximal length of a position in $Dom(t)$. Finally, for any expression $e$, $Var(e)$ is the set of variables occurring in $e$.

In addition, we consider an equational theory $\mathcal{E}$ defined by a finite set of equations over the alphabet $\mathcal{F}$ and variables. Some typical examples are studied in [8, 3, 19, 4]. We assume that $\mathcal{E}$ is given by a convergent rewrite system, possibly modulo the axioms of associativity and commutativity, in which case it is supposed to be coherent as well (see e.g. [12] for definitions concerning rewrite systems). This is the case in all relevant examples. For any message $m \in T(\mathcal{F}, X)$, $m\downarrow$ will denote the (unique up to associativity communativity) normal form of $m$. Substitutions are assumed to be normalized: the image of every variable is a term in normal form. We let $\Sigma$ be the set of normalized substitutions.

### 2.2 Protocols

Protocols consist in a finite set of roles $R_1, \ldots R_m$, each of which consists in

- agent generations of the form $\lambda a$: these agent names are the parameters of the role. For each role there is a distinguished agent name: the *main actor*.
- nonce generations of the form $\nu N$
- a finite sequence of rules $u_i \Rightarrow v_i$, where $u_i, v_i$ are either empty or a term in normal form in the algebra $T(\mathcal{F}, X)$.

*Example 1.* Consider the following protocol (inspired by Denning and Sacco):

$$A \to B : < A, \{\{K_{ab}\}_{\mathsf{priv}(A)}\}_{\mathsf{pub}(B)} >$$
$$B \to A : \{s_b\}_{K_{ab}}$$

The agent playing the role $A$ sends a new key, signed by her private key and encrypted with the public key of an agent playing the role $B$. Then he replies, sending a confidential text $s_b$ encrypted with the newly generated key. (We overload symmetric and asymmetric encryption). This may be compiled into two roles:

**A role:** $\begin{cases} \lambda a, \lambda b, \nu K_{ab}. & \Rightarrow < a, \{\{K_{ab}\}_{\mathsf{priv}(a)}\}_{\mathsf{pub}(b)} > \\ & \{x\}_{K_{ab}} \Rightarrow \end{cases}$

$A$ generates a key $K_{ab}$, sends a message (no premisse) and receives a message (without reply): $u_1, v_2$ are empty.

**B role:** $\{ \lambda b, \nu s_b. \ < y, \{\{z\}_{\mathsf{priv}(y)}\}_{\mathsf{pub}(b)} > \Rightarrow \{s_b\}_z$

For this protocol, the equational theory $\mathcal{E}$ is defined by the two rewrite rules:

$$\{\{x\}_{\mathsf{pub}(y)}\}_{\mathsf{priv}(y)} \rightarrow x \quad \{\{x\}_{\mathsf{priv}(y)}\}_{\mathsf{pub}(y)} \rightarrow x$$

which express that decryption of a message encrypted with a key is the same as encryption of the message with the inverse key.

The $\lambda, \nu$ statements for parameters and nonces can be removed in the case of a finite number of sessions, duplicating and instantiating the roles as many times as necessary.

We assume that, in any protocol rule $u_i \Rightarrow v_i$, the variables of $v_i$ are contained in the parameters, the nonces, and the variables of $u_j, j \leq i$. In other words, the action of sending $v_i$ is determined by the actual values of the parameters and the messages which have been received so far.

### 2.3 Offline Intruder theories

We consider sequents $T \vdash u$ where $u \in T(\mathcal{F}, X)$ and $T \subseteq T(\mathcal{F}, X)$ is finite. The intended meaning is "from a knowledge $T$, it is possible to deduce $u$".

The *offline intruder theory* $\mathcal{S}$ is defined by a (recursive) set of inference rules

$$\frac{T \vdash u_1 \ \cdots \ T \vdash u_n}{T \vdash u} \text{ If } C$$

where $C$ is a recursive predicate on instances of $u_1, \ldots, u_n, u$, which is invariant by $\mathcal{E}$. $\mathcal{S}$ contains for instance the *composition rules*:

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

for a subset $\mathcal{C}$ of $\mathcal{F}$. This states the ability of an intruder to apply $f$ to known terms. There is no side condition here. We also consider the axiom $T, u \vdash u$ as a composition rule. Examples of composition rules are $(E), (S), (B)$ in figure 1.

We write $T \vdash_{\mathcal{S}} u$ when $T \vdash u$ is derivable in $\mathcal{S}$. We will assume in the following that the deduction system $\mathcal{S}$ has some kind of subformula property. For any set of terms $T$ let $\mathsf{St}(T)$ be the set of strict subterms of terms in $T$ (for instance $\mathsf{St}(\{f(g(a), b), g(a), f(b, g(c))\}) = \{a, b, c, g(a), g(c)\}$), then:

**Definition 1.** *Let $F$ be a computable mapping from sets of terms to sets of terms such that for every set of terms $T$, $T \cup \mathcal{St}(T) \subseteq F(T)$ and $F(F(T)) \subseteq F(T)$. We say that the inference system $\mathcal{S}$ is $F$-*local*, if, for every proof $T \vdash_{\mathcal{S}} s$,*

- *Either the last inference rule is a composition and there is a proof in which all nodes belong to $F(T \cup \{s\})$*
- *Or all nodes of the proof belong to $F(T)$*

*Example 2.* The Dolev-Yao intruder theory is $F$-local, $F(T)$ being the set of subterms of terms in $T$. We will later consider another example: the *blind signatures*. They are used in electronic vote protocols such as [14]and have been formalised in [16]. Following [16] and using a result of [7], the intruder deduction capabilities can be described by the inference rules of figure 1. Let $F$ be the

$$\frac{T \vdash m \;\; T \vdash r}{T \vdash \{m\}_r}\,(E) \qquad \frac{T \vdash \mathsf{sign}(m, sk) \;\; T \vdash \mathsf{pub}(sk)}{T \vdash m}\,(V) \qquad \frac{T \vdash m \;\; T \vdash r}{T \vdash \mathsf{blind}(m, r)}\,(B)$$

$$\frac{T \vdash \{m\}_r \;\; T \vdash r}{T \vdash m}\,(D) \qquad \frac{T \vdash \mathsf{blind}(m, k) \;\; T \vdash k}{T \vdash m}\,(UB_1) \qquad \frac{T \vdash m \;\; T \vdash sk}{T \vdash \mathsf{sign}(m, sk)}\,(S)$$

$$\frac{T \vdash \mathsf{sign}(\mathsf{blind}(m, r), sk) \;\; T \vdash r}{T \vdash \mathsf{sign}(m, sk)}\,(UB_2)$$

**Fig. 1.** Intruder deduction rules for blind signatures

function, mapping a set of terms $T$ to the least set $S$ containing $T$, such that $\forall t \in S, \mathsf{St}(t) \subseteq S$ and

1. If $\mathsf{sign}(m, sk) \in S$ then $\mathsf{pub}(sk) \in S$
2. If $\mathsf{sign}(\mathsf{blind}(m, r), sk) \in S$ then $\mathsf{sign}(m, sk) \in S$

$F(T)$ is finite and polynomially computable, $F(F(T)) \subseteq F(T)$ and the inference system of figure 1 is $F$-local.

### 2.4 Transition systems

We shortly describe a standard semantics of security protocols, which is compatible with, e.g., [2, 9, 6]. We assume that $\mathcal{A}$ is a (infinite) set of agent names, some of which are honest and others are dishonest (or compromised). Nonces are modeled as terms $N_1(s), \ldots, N_k(s)$ where $s \in \mathbb{N}$ is a *session number*.

A *state* $q$ consists in a set $\mathcal{I}_q$ of terms (the *intruder knowledge*) and, for each agent name $a$, a *local state*. A local state is a partial mapping $M_{q,a}$ from non-negative integers (session numbers) to tuples containing at least a role, a step number and a list of bindings (for roles or protocol variables). The initial state $q_0$ consists in a set $\mathcal{I}_0$ of terms representing the initial intruder knowledge. The mapping $M_{q_0,a}$ is empty for all $a$.

There is a transition from state $q$ to a state $q'$ if $q$ and $q'$ only differ in their intruder knowledge component and in the local state of an agent $a$ and there is a role $R$ such that one of the following holds:

**session opening** : let $u \Rightarrow v$ be the first rule of $R$, $\sigma$ be a binding of the session parameters such that the main actor is bound to $a$. Either $u$ must be empty or $\mathcal{I}_q \vdash_{\mathcal{S}} u\theta \downarrow$ for some $\theta$ such that $x\sigma = x\theta$ for variables on which they are both defined. Then the local state of $a$ is changed by adding an integer $s$, which does not occur in any (sub)term of $q$, to the domain of $M_{q,a}$ and letting $M_{q',a}(s)$ be $(R, 1, \sigma \uplus \theta)$.
The intruder knowledge is increased by adding $v\theta \downarrow$ and, if $a$ is dishonest, also adding $N_1(s)$, ..., $N_k(s)$: $\mathcal{I}_{q'} = \mathcal{I}_q \cup \{v\theta \downarrow\}$ or $\mathcal{I}_{q'} = \mathcal{I}_q \cup \{v\theta \downarrow, N_1(s), \ldots, N_k(s)\}$.

**session progression** : Let $u \Rightarrow v$ be the rule $k$ of $R$ , $M_{q,a}(s) = (R, k-1, \sigma)$. There must be a substitution $\theta$ such that $\mathcal{I}_q \vdash_{\mathcal{S}} u\theta \downarrow$ and $x\sigma = x\theta$ for variables on which they are both defined. Then $M_{q',a}(s) = (R, k, \sigma \uplus \theta)$. The intruder knowledge is increased by adding $v\theta \downarrow$: $\mathcal{I}_{q'} = \mathcal{I}_q \cup \{v\theta \downarrow\}$.

We assume that the variables are renamed in such a way that two distinct role instances do not share variables. Then, for every state $q$, there is a unique substitution $\sigma_q$ such that, for every $a, s$ and every variable $x$, if $M_{q,a}(s) = (R, k, \theta)$ and $x$ is in the domain of $\theta$, then $x\theta = x\sigma_q$.

For any session number $s$, the role $R_s$ and the parameters $a_s^1, \ldots, a_s^m$ are entirely determined by $s$, by uniqueness of session numbers.

**Definition 2.** *There is an attack on the secrecy of some term $t \in T(\mathcal{F}, X)$ if there is a substitution $\sigma$ and a reachable state $q$ such that $\mathcal{I}_q \vdash_{\mathcal{S}} t\sigma \downarrow$ and all session variables occurring in $t$ are substituted with integers $s$ such that $a_s^1, \ldots, a_s^m$ are honest identities.*

*Example 3.* We continue example 1: Assume there are two honest agents $a_1, a_2$ The initial knowledge of the intruder consists of the names $a_1, a_2$ and their public keys $\mathsf{pub}(a_1), \mathsf{pub}(a_2)$. The initial state $q_0$ is defined by this knowledge $\mathcal{I}_0$.

Consider an instance of the $B$-role, in which $b$ is bound to $a_1$. There is a transition step from $q_0$ to the state $q_1$ such that $M_{q_1,a_1}(1) = (B, 1, \theta)$, with $\theta = \{b \mapsto a_1; y \mapsto a_1; z \mapsto a_2\}$ since

$$\mathcal{I}_0 \vdash_{\mathcal{S}} < a_1, a_2 > =< y\theta, \{\{z\theta\}_{\mathsf{priv}(y\theta)}\}_{\mathsf{pub}(y\theta)} > \downarrow$$

$\mathcal{I}_{q_1} = \mathcal{I}_{q_0} \cup \{\{s_b\}_{a_2}\}$ and $s_b$ is deducible in this state: the protocol is insecure.

## 3 Online deductions

Now, our goal is to enrich the intruder theory, so as to capture *online deduction*, i.e. deductions which use the protocol as an oracle. Moreover, as explained in the introduction, we want to keep the attack (substitution) apart from the deduction itself and that is why we use constrained sequents. The syntax of such formulas

is $T \vdash u[\![E]\!]_{\mathcal{E}}$ where $u$ is a term containing possibly variables, $T$ is a finite set of ground terms, $E$ is a conjunction of equations and $\mathcal{E}$ is a finite set of equations. The idea is to lift the offline intruder deduction rules to terms with variables, recording in $E$ the variable bindings, introduced by some inference rule.

Equations in the constraint part are interpreted modulo $\mathcal{E}$: $\sigma$ is a *$\mathcal{E}$-solution of $E$* if, for every equation $u = v \in E$, $\mathcal{E} \models u\sigma = v\sigma$, which we also write $\sigma \models_{\mathcal{E}} u = v$. More generally, given two formulas $\phi_1$, $\phi_2$, we write $\phi_1 \models_{\mathcal{E}} \phi_2$ instead of $\mathcal{E}, \phi_1 \models \phi_2$ (the usual logical consequence of first-order logic) and $\phi_1 \models\!\!\models_{\mathcal{E}} \phi_2$ if $\phi_1 \models_{\mathcal{E}} \phi_2$ and $\phi_2 \models_{\mathcal{E}} \phi_1$. We omit the subscript $\mathcal{E}$ when it is irrelevant.

Finally, we also record in the constraints the control points of the various instances of the roles; we use variables $x_{R,k,s}$ ranging over $\{0, 1\}$ and which, when raised, mean that the session $s$ of role $R$ reached stage $k$.

Given an inference system $\mathcal{S}$, we compute an inference system $\widehat{\mathcal{S}}$, which acts on constrained sequents and extends $\mathcal{S}$ with the use of protocol rules as oracles. First, every rule of $\mathcal{S}$:

$$\frac{T \vdash u_1 \dots T \vdash u_k}{T \vdash u} \text{ If } C$$

is replaced with the rule

$$\frac{T \vdash u'_1[\![E_1]\!] \dots T \vdash u'_k[\![E_k]\!]}{T \vdash u \downarrow [\![E_1 \wedge \dots \wedge E_k]\!]} \text{ If } C \wedge R$$

$u'_1, \dots, u'_k$ are the result of linearizing $u_1, \dots, u_k$. $R$ is the set co-references: $\forall i. u'_i \sigma_R = u_i$. As an example:

$$\frac{T \vdash \{x\}_y \;\; T \vdash y}{T \vdash x} \quad \text{becomes} \quad \frac{T \vdash \{x\}_y[\![E_1]\!] \quad T \vdash y'[\![E_2]\!]}{T \vdash x \downarrow [\![E_1 \wedge E_2]\!]} \text{ If } y = y'$$

In addition to the extensions of rules of $\mathcal{S}$, we add the following inference rules:

**Instantiation rule:**

$$\frac{T \vdash u[\![E]\!]}{T \vdash u\sigma \downarrow [\![\sigma]\!]} \text{ If } \sigma \models_{\mathcal{E}} E \text{ and } \mathcal{X}_c \cap Var(\sigma) \subseteq Var(E)$$

$\sigma$ is meant to be any solution of $E$: this reflects the semantics of the constraints. Moreover, $\sigma$ should not bind any control variable, which was unbounded before.

**Weakening rule:**

$$\frac{T \vdash u_1[\![E_1]\!] \quad T \vdash u_2[\![E_2]\!]}{T \vdash u_1 \downarrow [\![E_1 \wedge E_2]\!]} \text{ W}$$

Such a rule is useful when deducing $u_2$ is irrelevant: only deducing *some* term with constraint $E_2$ is used later in the proof. This happens when the intruder needs to use the second rule of a role, but not the first one. Then he must be able to force the agents to play the first rule, regardless to the result, moving on the control point to the second rule.

**The protocol rule for session progression.**

$$\frac{T \vdash u[\![E]\!]}{T \vdash w \downarrow [\![u = v \wedge E \wedge x_{R,k,s} = 1]\!]}\ \mathrm{P} \qquad \text{If} \begin{cases} 1.\ v \Rightarrow w \text{ is the rule } k > 1 \text{ of role } R, \text{ session } s \\ 2.\ E \not\models_{\mathcal{E}} x_{R,k,s} = 1 \text{ and } E \models_{\mathcal{E}} x_{R,k-1,s} = 1 \end{cases}$$

The rule expresses that, whenever some (instance of) $v$ has been deduced and the protocol as reached stage $k-1$ of session $s$ of role $R$, then the intruder may use the $k$th rule of $R$ as an oracle: he gets the corresponding instance $w$ and the control point moves to $k$ for that session. In addition, we impose $u = v$ in the constraint, meaning that the corresponding instance of $u$ matches (modulo $\mathcal{E}$) the left side of the protocol rule.

Note that the rules of $R$ are assumed to be renamed in such a way that the variables have different names in different sessions.

There is a similar rule for session opening: the only difference is that we do not require having reached the stage $k-1$ of session $s$, but we require $s$ to be a new session number instead. Moreover, parameters of the new sessions are bound and their values are recorded in the constraint for further compatibility check.

**Compromised agents.**

$$\frac{T \vdash u[\![E]\!]}{T \vdash N_i(s)[\![E]\!]} \qquad \text{If} \begin{cases} 1.\ E \models_{\mathcal{E}} x_{R,1,s} = 1 \\ 2.\ \text{The main actor of session } s \text{ is compromised} \\ 3.\ N_i(s) \text{ is one of the nonces generated in session } s. \end{cases}$$

This expresses that all data generated by compromised (or dishonest) agents are available to the intruder. We only require that the session $s$ has been opened. With such a rule, we do not need to give the intruder the ability to generate new data, as soon as there is at least one compromised agent.

*Example 4.* We continue example 3, restating the attack as a deduction in $\widehat{\mathcal{S}}$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{a_1, a_2 \vdash a_2[\![]\!]} \quad \dfrac{}{a_1, a_2 \vdash a_2[\![]\!]}}{a_1, a_2 \vdash < a_2, a_2 >[\![]\!]}}{a_1, a_2 \vdash \{s_b(1)\}_z[\![y = a_2 \wedge z = a_2 \wedge b = a_1 \wedge x_{B,1,1} = 1]\!]}\ P}{a_1, a_2 \vdash \{s_b(1)\}_{a_2}[\![y = a_2 \wedge z = a_2 \wedge b = a_1 \wedge x_{B,1,1} = 1]\!]}\ \mathcal{I} \quad \dfrac{}{a_1, a_2 \vdash a_2[\![]\!]}}{a_1, a_2 \vdash s_b(1)[\![y = a_2 \wedge z = a_2 \wedge b = a_1 \wedge x_{B,1,1} = 1]\!]}$$

The next lemmas show that our inference rules are adequate with the transition system: the model in which we view the protocol rules as oracles can safely be used in place of the transition system.

**Lemma 1 (Completeness of $\widehat{\mathcal{S}}$).** *For any reachable state $q$ and any term $t$, if $\mathcal{I}_q \vdash_{\mathcal{S}} t$, then we can build a proof $\Pi$ in $\widehat{\mathcal{S}}$ of $\mathcal{I}_0 \vdash t \downarrow [\![E]\!]$ such that $\sigma_q \models_{\mathcal{E}} E$.*

Conversely, the proof system $\widehat{\mathcal{S}}$ is correct w.r.t. the trace semantics:

**Lemma 2 (Correctness of $\widehat{\mathcal{S}}$).** *Let $\Pi$ be a proof of $\mathcal{I}_0 \vdash t[\![E]\!]$. Then, for every substitution $\sigma$ such that $\sigma \models_{\mathcal{E}} E$, there is a reachable state $q$ such that $\mathcal{I}_q \vdash_{\mathcal{S}} t\sigma \downarrow$.*

Therefore, we claim that the existence of an attack can be restated in this setting:

**Theorem 1.** *There is an attack on the secrecy of $s$ iff there is a proof of some $T \vdash s'[\![E]\!]$ such that $s =_{\mathcal{E}} s'\sigma \downarrow$ and $\sigma \models_{\mathcal{E}} E$.*

## 4 A normal proof result

From now on, we assume that $\mathcal{E}$ is empty. We also assume that $T$ (in the left of sequents) always contains a constant 0 as well as terms $f(0, \ldots, 0)$ for (public) constructor symbols $f \in \mathcal{C}$. Moreover, we need some hypotheses on the inference system $\mathcal{S}$, beyond $F$-locality, as shown by lemma 3.

### 4.1 Additional hypotheses on the offline deduction system

We assume that decomposition rules (i.e. those which are not composition rules):

$$(\mathcal{D})\ \frac{t_1 \qquad \ldots \qquad t_n}{t}\ \text{IF } C$$

are such that each $t_i$ is at most of depth 2 and for each $t_i$ whose depth is 2, one of the following holds:

1. $t$ is a subterm of $t_i$ and $t$ is a variable or a term of depth 1.
2. $t_i = C[f(u_1, \ldots, u_m)]$ and $t = C[u_i]$, where $C$ is any context and $f \in \mathcal{C}$. Moreover $f$ cannot occur at depth 1 in another decomposition rule.

And the side condition $C$ is a conjunction of equations between variables such that, if $x = y \in C$ and $x$ occurs at depth 2 and $y$ occurs at depth at least one, then $x$ occurs below a unary symbol and $y$ occurs below a unary symbol.

*Example 5.* The decomposition rules of the blind signature theory satisfy these conditions, as well as classical Dolev-Yao rules or deduction rules for Cipher Block Chaining, for instance.

The conditions on $C$ might be not necessary for our main result, but the other conditions are necessary as shown by the following lemma, obtained by reduction of the Post Correspondance Problem:

**Lemma 3.** *There is a local intruder inference system (decidable in PTIME) such that every decomposition rule has a single hypothesis of depth 2 and a conclusion of depth 1 and such that the insecurity problem for one session of a protocol containing one rule is undecidable.*

### 4.2 Modifying the instanciation rule

Since constraints are now interpreted in the free algebra, following standard concepts in unification theory, we can keep a satisfiable constraint in *solved form*, i.e. a conjunction of equations $x_1 = t_1 \wedge \ldots \wedge x_n = t_n$ where $x_1, \ldots, x_n$ are distinct variables and $x_i \notin Var(t_i, \ldots, t_n)$. Such a solved form defines an *occurrence ordering* $\geq_{occ}$ by $x_i \geq_{occ} y$ for every $y \in Var(t_i)$. $E$ also defines a congruence $=_E$ on the set of terms: the least congruence containing $E$.

Our instanciation rule is currently too coarse. We want to use it more carefully, and keep the terms as small as possible. That is why we replace it with:

$$\frac{T \vdash u[x]_p[\![E \wedge x = t]\!]}{T \vdash u[t]_p[\![E \wedge x = t]\!]}\ (I)$$

in other words, we only replace one occurrence of one variable with its current binding in the equality constraint, which is assumed to be in solved form. The original instanciation rule can be simulated iterating the new one.

### 4.3 The normal proof theorem

We want to show that, if there is a proof of $T \vdash s[\![E]\!]$, then there is a proof, which only uses particular sequents, which depend on the protocol rules $P$, $T$ and $s, E$. In order to state the result, we need the notion of admissible sequent. Intuitively, the substitution defined by the final constraint $E$ should be a stack of elementary assignments, each of which to a subterm of a term in $T, s, P$.

**Definition 3.** *For any set of terms $S$ and constraints $E, G$ in solved form, a sequent $T \vdash s[\![G]\!]$ is $S, E$-*admissible *iff*

1. *for all $x = t \in G$, if $x$ is maximal (w.r.t. $\geq_{occ}$), then $T \vdash t[\![G \setminus \{x = t\}]\!]$ is $S, E$-admissible*
2. *$s \in S$ or else there is a $t$ such that $s =_E t$ and $t \in S$.*

*Example 6.* Assume $S = \{\{x_1\}_{k_1}, k_2, x_3\}$, $E = [\![x_1 = \{x_2\}_{k_2}]\!]$ The following sequents are $S, E$-admissible:

$$T \vdash \{\{x_2\}_{k_2}\}_{k_1}[\![]\!]\,,\ T \vdash x_3[\![x_3 = \{\{x_2\}_{k_2}\}_{k_1} \wedge x_2 = k_2]\!]$$

while the following are not $S, G$-admissible:

$$T \vdash \{x_3\}_{k_2}[\![]\!]\,,\ T \vdash x_3[\![x_3 = \{x_2\}_{k_2} \wedge x_2 = \{x_1\}_{k_2}]\!]$$

If $\Pi$ is a proof in $\widehat{\mathcal{S}}$ we write $V(\Pi)$ the union, for all roles $R$ and all sessions $s$ of $R$ opened in $\Pi$ of

1. The control variables $x_{R,i,s}$ ($i$ smaller than the number of rules in $R$)
2. The nonces $N_i(s)$ generated in this instance of role $R$
3. The parameter bindings for session $s$
4. The terms $\{u_i, v_i\}$ for every (renamed) protocol rule $u_i \Rightarrow v_i$ in $R$

Using for instance results in [6], the parameter bindings can actually be restricted to a finite fixed set, hence the size of each $V(\Pi)$ is $O(|R|)$.

**Theorem 2.** *Assume the hypotheses of the previous section on $\mathcal{S}$, in particular its $F$-locality. Assume that there is a proof $\Pi$ of $T \vdash s[\![E]\!]$ in $\widehat{\mathcal{S}}$, with a satisfiable $E$ then there is a proof $\Pi'$ of a sequent $T \vdash s'[\![E']\!]$ in $\widehat{\mathcal{S}}$ such that:*

1. *$s\sigma_E = s'\sigma_{E'}$*
2. *$T \vdash s'[\![E']\!]$ is $F(V(\Pi) \cup T \cup \{s\sigma_E\}), \emptyset$-admissible*
3. *every sequent $T \vdash t[\![E'']\!]$ in $\Pi'$ is $F(V(\Pi) \cup T \cup \{s\}), E'$-admissible*

The full proof of this theorem can be found in [1] (in French). We will try now to sketch it and convince the reader that it works.

The first step consists in performing several proof rewritings, such as:

**Lemma 4.** *If there is a proof $\Pi$ of $T \vdash s[\![E]\!]$, then there is a proof of the same sequent in which no weakening precedes an $\mathcal{S}$-rule.*

Slighty more complex is the control of instanciations, whose delay is necessary if we want to keep the sequents small:

**Lemma 5.** *If there is a proof $\Pi$ of $T \vdash s[\![E]\!]$ and $E$ is satisfiable, then there is a proof $\Pi'$ of $T \vdash s'[\![E]\!]$ such that $s\sigma_E = s'\sigma_E$ and any application of an instanciation rule in $\Pi'$ replaces occurrences of variables at depth at most one. Moreover, this instanciation must be followed by a decomposition rule which would not be applicable before.*

To prove this, we simply swicth instanciations with other rules when it is possible and rely on the hypotheses on the depth of the premisses of decomposition rules.

We may also rely on $F$-locality for the normalization of pure $\mathcal{S}$-parts of the proof. Then, it is not easy to perform further proof transformations: we would need contextual rewriting rules; it may be the case that a proof $\Pi$ satisfies the conditions of the theorem, while some of its subproofs do not. Also, conversely, every subproof of $\Pi$ may satisfy the theorem while $\Pi$ does not. Let us show an example of the first case, which illustrates the need of contextual rewriting (or more complicated inductive hypotheses).

*Example 7.* Consider a toy deduction system $\mathcal{S}$, with the deduction rules $\dfrac{x}{a(x)}$, $\dfrac{x}{f(x)}$, $\dfrac{b_0(x)}{c_1}$, $\dfrac{b_n(0)}{c_2}$ and $\dfrac{x \quad y}{< x, y >}$. And protocol rules $r_0 : f(x_0) \to b_0(x_0)$ and, for every $i \leq n-1$, $r_{i+1} : b_i(a(x_{i+1})) \to b_{i+1}(x_{i+1})$. The proof of $< c_1, c_2 >$ is displayed in figure 2 (we omit control variables for simplicity) The proof satisfies the hypotheses of the theorem: $a(x_i)$ is a subterm of some protocol rule, hence the final constraint is admissible. Other constraints, such as $x_0 = a^n(0)$ are also $E$-admissible since $a^n(0) =_E a(x_1)$ for instance.

However, the left part of the proof does not satisfy the theorem as, in this subproof, $a^n(0)$ is not admissible if $n \geq 1$. And, indeed, there is a much simpler

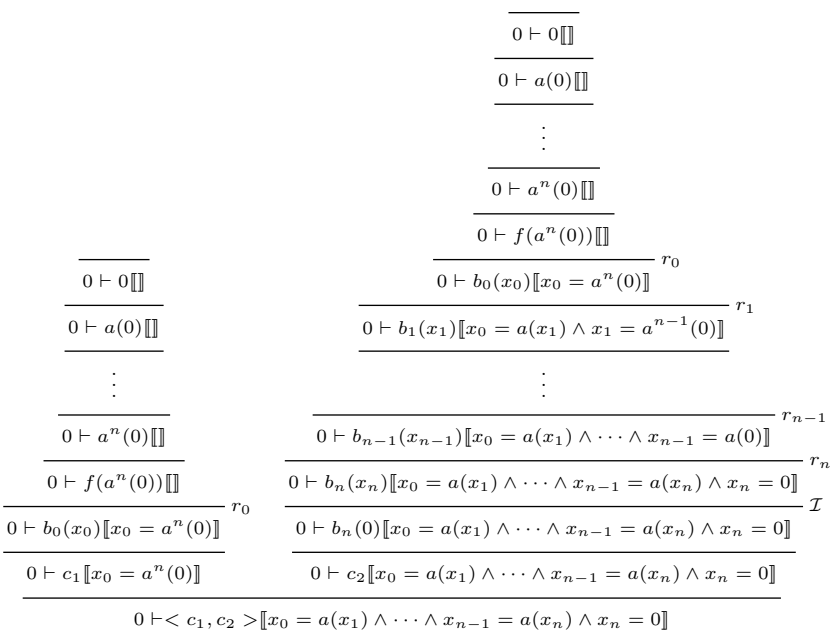


**Fig. 2.** An example of a proof

proof, with $x_0 = 0$, while there is no simpler proof of the right branch. We must bind $x_0$ to $a^n(0)$ for compatibility between the two branches. This shows that the simplification of the left part proof depends on contextual informations.

The idea is to keep in a box the terms which are superflously large (such as $a^n(0)$ in the example) and open the box only when necessary (at the last step in the above example). If a box is not opened, then it can be replaced by an arbitrary term, which can be deduced by the intruder. Now, the invariant in our proof transformations is that an expression $\Box_t$ can be replaced by an arbitrary deducible term in the current proof, but we remember that $t$ is also deducible at this stage. Let us now skecth how these boxes are introduced and opened. The core of the result is that we only need to replace $\Box_{C[t]}$ with $C[\Box_t]$ when $C$ is a piece of a protocol rule.

Assume the last rule of the proof is a protocol rule (we omit the control points here) and that (by an induction hypothesis) there is a proof of $T \vdash u[\![E]\!]$ which has the desired properties:

$$\dfrac{\dfrac{\Pi}{T \vdash u[\![E]\!]}}{T \vdash w[\![E \wedge u = v \wedge \ldots]\!]}$$

if $v \Rightarrow w$ is a protocol rule. Let also $E_1$ be a solved form of $E \wedge u = v$. Now, by definition $w \in F(V(\Pi) \cup T)$, but there might be a variable $x$ of $v$ such that $x = t \in E_1$ and $t$ is not in $F(V(\Pi) \cup T)$. We have then to transform the proof.

Let $t = f(t_1, \ldots, t_n)$. By properties of the classical matching algorithm,$t$ is a subterm of $u$. Moreover, $t \notin F(V(\Pi) \cup T)$ implies that any inference rule in $\Pi$ yielding a superterm of $t$ is a construction rule. By a simple induction, there must be in $\Pi$ a rule $\dfrac{T \vdash t_1[\![C_1]\!] \cdots T \vdash t_n[\![C_n]\!]}{T \vdash f(t_1, \ldots, t_n)[\![C_1 \wedge \ldots \wedge C_n]\!]}$. Now, we can replace $t_1, \ldots, t_n$ with arbitrary terms, which are deducible by the intruder: the proof will still be valid. Le $\Box_{t_i}$ be such a replacement. We now have a proof of $T \vdash w[\![x = f(\Box_{t_1}, \ldots, \Box t_n) \wedge ...]\!]$, which satisfies the requirements.
Now, if, later in the proof, we use an instantiation of $x$, our tranformation yields

$$\dfrac{\dfrac{\dfrac{\Pi_1}{T \vdash C[x]_p[\![x = f(\Box_{t_1}, \ldots, \Box_{t_n}) \wedge \ldots]\!]}}{T \vdash C[f(\Box_{t_1}, \ldots, \Box t_n)]_p[\![x = f(\Box_{t_1}, \ldots, \Box_{t_n}) \wedge \ldots]\!]}\,\mathcal{I} \quad T \vdash u_1[\![D_1]\!] \cdots T \vdash u_n[\![D_n]\!]}{T \vdash u[\![D]\!]}$$

which might no longer be a proof in $\widehat{\mathcal{S}}$.

By lemma 5, the last rule must be a decomposition rule and, thanks to our hypotheses on $\mathcal{S}$, $p$ has a length 0 or 1. $u$ cannot be $\Box_{t_i}$ or $f(\Box_{t_1}, \ldots, \Box_{t_n})$, otherwise we would have a much simpler proof of the same sequent using a subproof of $\Pi_1$ yielding $u$, and weakenings. Similarly, $p$ must be of length 1, otherwise we have a shorter proof. According to our conditions on $\mathcal{S}$, we are in one of the following cases:

1. $u = \Box_{t_i}$
2. $u$ is a subterm of $u_i$ at depth 1
3. $u$ is a subterm of $C[f(\Box_{t_1}, \ldots, \Box_{t_n})]_p$ at depth 1
4. $u = C[\Box_{t_i}]_p$ and $f$ cannot occur at depth 1 in another decomposition rule

The first case has already been ruled out. In the second case we still get a proof in $\widehat{\mathcal{S}}$. In the third case, either the position of $u$ is $p$ and we have seen already that there is a simpler proof, or else we get a proof in $\widehat{\mathcal{S}}$. In the last case, the idea is to apply a proof transformation in the original proof, replacing $f(t_1, \ldots, t_n)$ with $t_i$. We then get again a shorter proof.

The last problem, which we do not want to address here, are the side conditions in the decomposition rules: it might be the case that the box-replacement sketched above yields a failure of an equality test in the side conditions. Then we have to perform transformations on other branches of the proofs, and we will use the additional hypotheses on side conditions.

### 4.4 Consequences of the normalisation theorem

**Corollary 1.** *If the number of sessions is fixed and if $F$ can be computed in PTIME, then the insecurity problem is in NP.*

Indeed, in this case, $P = V(\Pi)$ is fixed and $E$ is empty. We can first guess $s', E'$; $E'$ must bind each variable to a term in $F(P \cup T \cup \{s\})$. Then note that there are

only a polynomial number of admissible sequents, up to $E'$: at the possible price of initial weakenings, all constraints are identical to $E'$, except for their control part. Then any sequent $T \vdash t[\![E'']\!]$ in the normal proof is such that $t = u\theta$ with $u \in F(P, T, s)$ and $E' \models \theta$. It is sufficient then to guess a subset $S$ of $T(P, T, s)$ (the deducible terms) and an ordering on $S$ (the ordering in which the terms are deduced), assign non-deterministically a control point to each of these terms and check that every term in $S$ (or one of its instances by $\sigma_{E'}$) can be deduced in one step from smaller ones and their instances by $\sigma_{E'}$.

This is actually similar to NP membership proofs in the papers by Y. Chevalier, M. Rusinowitch and M. Turuani, such as [18].

**Corollary 2.** *For the theory of blind signatures, the insecurity problem in a fixed number of sessions is in NP.*

## 5 Discussion and comparison with related work

Our starting point was the PhD thesis of M. Turuani [21]: we tried to formulate the results in terms of proof normalisations. We hoped first to better understand the reasons why we can get decidability results and, of course, derive more general results. We also planned to extend theorem 2 to equational theories, but it turns out to be quite technical and not very illuminating so far.

We partly succeeded to achieve our goals. The statement of our main theorem is satisfactory because the conditions on the inference system are straightforward to check and the normal proof results provides with a general (complete) proof strategy. Moreover, we cover some intruder theories, which are not in the scope of the "oracle rules" of [21]. We have seen blind signatures and there are other examples such as Dolev-Yao plus the rule $\dfrac{T \vdash g(x) \quad T \vdash f(g(x))}{T \vdash f(x)}$...

This is only a partial success since first there are restrictions, which are probably not necessary. Second there are also intruder theories, which can be handled by oracle rules and do not satisfy our hypotheses. For instance, we cannot handle "shortcuts", which are rules obtained by composing other rules of the system.

The main remaining work, besides tuning our prototype implementation, is to extend the results to the associative-commutative case, which looks quite challenging.

## References


1. V. Bernat. *Théories de l'intrus pour la vérification de protocoles cryptographiques.* PhD thesis, École Normale Supérieure de Cachan, 2006.
2. I. Cervesato, N. Durgin, P. Lincoln, J. Mitchell, and A. Scedrov. A meta-notation for protocol analysis. In P. Syverson, editor, *12-th IEEE Computer Security Foundations Workshop*, 1999.
3. Y. Chevalier, R. Kuester, M. Rusinowitch, and M. Turuani. An NP decision procedure for protocol insecurity with xor. In Kolaitis [15].

4. Y. Chevalier, R. Küsters, M. Rusinowitch, and M. Turuani. Deciding the security of protocols with Diffie-Hellman exponentiation and products in exponents. In *Proc. FST/TCS, Mumbai*, volume 2914 of *Lecture Notes in Computer Science*, 2003.
5. E. Clarke, S. Jha, and W. Marrero. Using state space exploration and a natural deduction style message derivation engine to verify security protocols. In *Proceedings of the IFIP Working Conference on Programming Concepts and Methods (PROCOMET)*, 1998.
6. H. Comon-Lundh and V. Cortier. Security properties: two agents are sufficient. *Science of Computer Programming*, 50(1–3):51–71, 2004.
7. H. Comon-Lundh and S. Delaune. The finite variant property: How to get rid of some algebraic properties. In *Proc. of the 16th International Conference on Rewriting Techniques and Applications (RTA'05)*, volume 3467 of *Lecture Notes in Computer Science*, 2005.
8. H. Comon-Lundh and V. Shmatikov. Intruder deductions, constraint solving and insecurity decision in preence of exclusive or. In Kolaitis [15].
9. V. Cortier, J. Millen, and H. Rueß. Proving secrecy is easy enough. In *14th IEEE Computer Security Foundations Workshop*, pages 97–108. IEEE Computer Society, 2001.
10. V. Cortier, M. Rusinowitch, and E. Zalinescu. A resolution strategy for verifying cryptographic protocols with cbc encryption and blind signatures. In *Proc. 7th ACM-SIGPLAN Int. Conf. on Principles and Practice of Declarative Programming (PPDP'05)*, pages 12–22, 2005.
11. G. Denker, J. Millen, and H. Rueß. The CAPSL integrated protocol environment. Technical report, SRI International, Oct. 2000.
12. N. Dershowitz and J.-P. Jouannaud. Rewrite systems. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, pages 243–309. North Holland, 1990.
13. F. T. Fabrega, J. Herzog, and J. Guttman. Strand spaces: Proving security protocol correct. *Journal of Computer Security*, 7:191–230, 1999.
14. A. Fujioka, T. Okamoto, and K. Ohta. A practical secret voting scheme for large scale elections. In *Proc. ASIACRYPT'92*, Lecture Notes in Computer Science, 1993.
15. P. Kolaitis, editor. *Eighteenth Annual IEEE Symposium on Logic in Computer Science*, Ottawa, Canada, June 2003. IEEE Computer Society.
16. S. Kremer and M. Ryan. Analysis of an electronic voting protocol in the applied pi-calculus. In M. Sagiv, editor, *Proc. 14th. European Symposium on Programming (ESOP'05)*, volume 3444 of *Lecture Notes in Computer Science*. Springer-Verlag, 2005.
17. D. McAllester. Automatic recognition of tractability in inference relations. *J. ACM*, 40(2):284–303, 1993.
18. 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.
19. V. Shmatikov. Decidable analysis of cryptographic protocols with products and modular exponentiation. In *Proc. European Symposium on Programming (ESOP'04)*, volume 2986 of *Lecture Notes in Computer Science*. Springer-Verlag, 2004.
20. Security protocols open repository. http://www.lsv.ens-cachan.fr/spore/.
21. M. Turuani. *Sécurité des protocoles cryptographiques: décidabilité et complexité*. PhD thesis, Université Henri Poincaré- Nancy 1, 2003.