Deriving the Complete Knowledge of Participants in Cryptographic Protocols

(Extended Abstract)

Marie-Jeanne Toussaint
Université de Liège
Institut Montefiore, B28, Sart Tilman
B 4000 Liège (Belgium)
toussain@montefiore.ulg.ac.be

Abstract

This paper shows how to derive a representation of the participants' knowledge in a cryptographic protocol. The modelization is based on the assumption that the underlying cryptographic system is perfect and is an extension of the "Hidden Automorphism Model" introduced by Merritt. It can be used to establish the security of the protocols.

1 Introduction

A fair amount of research is devoted to developing cryptographic systems that are as secure as possible. Unfortunately, this is not sufficient: if a cryptographic system is used in an incorrect protocol, security can be compromised even if the cryptographic system is perfect. In the literature, the analysis of the security of the protocols is often strongly linked to the particular structure of the used cryptosystem ([Moo88, Dam87, EGS86]).

We adopt the point of view of Merritt in [Mer83] who studied the problem of reasoning about cryptographic protocols, assuming the underlying cryptographic system to be perfect. In [Mer83] and [MW85], a cryptosystem is represented by an algebra (called the *crypto-algebra*) and its perfection is modeled by the fact that the crypto-algebra is isomorphic to the free algebra of the same type.

In [TW91], we introduced a new representation of the participants' knowledge. The messages and keys whose a participant knows the meaning are represented in his state of knowledge by constants of the free-algebra whereas the ones whose he does not know the meaning are represented by variables defined on the free-algebra. This representation enabled us to obtain a new method to prove the security of cryptographic protocols. But, as we have seen by studying some examples in [Tou91], it is not sufficient because it does not model all the inferences and computations that the participants (and opponents) are able to make from their states of knowledge.

J. Feigenbaum (Ed.): Advances in Cryptology - CRYPTO '91, LNCS 576, pp. 24-43, 1992.

In this paper, we refine the representation by modeling all the knowledge that a participant is able to obtain. We assume that the set of all the messages and keys that a participant knows is closed under the enciphering and deciphering operators. However, if in the computation of this closure he finds by two distinct ways the same sequence of bits, the participant will often be able to deduce from that the meaning of some messages and keys i.e. to infer from that the values of the free-algebra variables representing these messages and keys. In fact, the two ways of computing the same sequence of bits correspond to two distinct expressions in the free-algebra; the inferences that the participant is able to draw consist in unifying these two expressions. When this unification is not possible, an inconsistency is detected and an attempt at cheating is discovered. Moreover, a participant can instantiate some variables of his state of knowledge (we talk then about a 'state of belief') and try to obtain some informations by inferences. These inferences are also modeled by unifications between expressions of the free-algebra and the probability that such unifications really bring some informations is studied. In fact, other types of inferences exist and are all modeled in this paper by unifications or contra-unifications (i.e. elimination of some values from the domain of the variables because some expressions in the free or crypto-algebra which could be unified correspond to distinct elements in the other algebra) of elements of the free-algebra or the crypto-algebra.

The main originalities of our method are its generality (we can apply this method to protocols using public or private key cryptography, preserving the secret of data, to authentication protocols, signature schemes,... (see [Tou91])), the consideration of the probabilistic knowledge of the participants and the use of a model based on the assumption of a 'perfect' cryptosystem. Other papers [BAN89, Bie89, Kem89, Mea89, Var89, MCF87] mainly prove the security of keys distribution protocols where the probabilistic choice does not occur: the properties of the cryptosystem that ensure security are there modeled by axioms and the proofs often use logics of knowledge or belief.

This paper is organized as follows. In Section 2, we present our model of the representation of knowledge of the participants in a cryptographic protocol. In Section 3 (Subsection 3.1), we introduce some examples to show that this model has to be refined. Afterwards we formalize the inferences that a user is able to draw from his state of knowledge (Subsection 3.2) and we define the notion of 'inferred state of knowledge' which contains all the informations that a user can obtain by computations and inferences (Subsection 3.3).

2 Modeling States of Knowledge

2.1 Modeling the Cryptosystem

A cryptosystem is seen as an algebraic system called the *crypto-algebra* $\mathcal{C}$. For simplicity, we suppose that the plaintexts and ciphertexts belong to the same set $\mathcal{M}$ and the keys to a set denoted $\mathcal{K}$. The operators of the crypto-algebra $\mathcal{C}$ are usually the enciphering function E and the deciphering function D which are linked by some relations.

Following [MW85], we consider an idealization of the cryptosystem. It is the quotient by the relations between e and d of the free algebra of the same type as the crypto-algebra. This free-algebra represents the structure of the cryptosystem and is denoted here by $\mathcal{F}$. The operators of the free-algebra are respectively denoted by e and d corresponding tc E and D in the crypto-algebra. The 'perfection' of the cryptosystem is modeled by assuming that the crypto-algebra is isomorphic to the free-algebra.

2.2 Modeling Participants' Knowledge

As in [TW91], the state of knowledge of a participant is seen as a partial knowledge of the isomorphism (denoted here φ) between the free and crypto-algebras.

Definition 2.1 A state of knowledge *about a cryptosystem is a subset of* $\mathcal{F} \times \mathcal{C}$.

This definition is very general. However, we only consider states of knowledge which can be partitioned in three special finite sets F, V and SV: these states of knowledge are called *representable*.

1. F (for Fixed) is formed by pairs (a, b) which define a one to one mapping from a subset of $\mathcal{F}$ to a subset of $\mathcal{C}$. That corresponds to elements the participant has seen or that he knew at the start of the protocol and that he can label by a fixed element of the free-algebra. The set F will be described by specifying the free-algebra component of the pairs.
2. V (for Variables) is formed by pairs (x, y) where x is a fixed generator of the free-algebra whereas y ranges over a subset of the generators of the crypto-algebra. The pairs of this type correspond to generators of the crypto-algebra that the participant has not seen but the existence of which he is aware of; they will be represented by a variable denoted by a tilded symbol, $\tilde{x}$, ranging over the image

under the isomorphism φ^{-1} of the domain of y. The variable y is then called the 'dual variable' of $\tilde{x}$. In the rest of this paper, we consider V as a set of variables of the free-algebra instead of a set of pairs.

3. SV (for *S*emi-*V*ariables) is formed by pairs (z, a) where, for each pair, a is a fixed element of the crypto-algebra and z belongs to a finite subset of

$$Cl(F \cup V) \setminus (Cl(F) \cup V)$$

where $Cl(X)$ (for any subset X of $\mathcal{F}$) denotes the closure of X under the enciphering and deciphering operators. These pairs correspond to elements that the user has seen but is unable to label. A pair (z, a) of SV is represented by a variable (denoted z^*) defined on the free-algebra with an inclusion constraint on this variable in the corresponding subset of $Cl(F \cup V) \setminus (Cl(F) \cup V)$ and occasionally inequality relations between variables.

The participants make some computations on their states of knowledge. We overvalue them by assuming that each participant is able to compute the closure $Cl(F \cup SV)$: this closure is called the ***seen fraction*** of the participant (because it corresponds to the closure of the set of elements of the crypto-algebra that he has seen) whereas $Cl(F)$ is called his ***known fraction***.

A participant can also consider some instantiations of the variables of his state of knowledge: the obtained set is then called a 'state of belief'.

Definition 2.2 *A* state of belief compatible with a knowledge state $K = F \cup V \cup SV \subset \mathcal{F} \times \mathcal{C}$ *is a maximal restriction of K to a one to one mapping ('maximal' means here that there is no other restriction of K to a one to one mapping including this state of belief).*

Even if the elements of V are represented, for simplicity, as variables in the free-algebra, their free-algebra components can be fixed whereas their crypto-algebra components vary. It is thus more natural to define an instantiation of an element of V as an element of the crypto-algebra. On the other hand, the elements of SV (or more exactly their free-algebra components) are really variables in the free-algebra and are instantiated by elements of the free-algebra. By definition, the ***free-part*** of an instantiation of the variables of a knowledge state is the restriction of this instantiation to the elements of SV and its ***crypto-part*** is its restriction to the elements of V.

For each state of belief Be compatible with a knowledge state $K = F \cup V \cup SV$, there is one and only one instantiation i of the variables of this knowledge state such that

$$Be = F \cup \{(\tilde{v}, i(\tilde{v})) : \tilde{v} \in V\} \cup \{(i(x^*), SV(x^*)) : x^* \in SV\} \tag{1}$$

where SV is considered as a function which maps the variable representing an element (i.e. a pair) of SV to its second component. The instantiation i is called the *instantiation relative to the belief state Be*.

We can divide the set of belief states compatible with a given state of knowledge in (finite) equivalence classes for the relation: "their relative instantiations have the same crypto-part". We could obtain a similar relation for the free-part of the relative instantiations of the belief states but this relation would not be used below.

3 Modeling the Possible Inferences of a User from his State of Knowledge

3.1 Problem Statement

Each participant tries to obtain some additional informations by analyzing his state of knowledge. We have overvalued the power of computation of a participant by assuming that he is able to compute his seen fraction. But that is not sufficient: we have not modeled the different inferences the participant is able to draw by analyzing his seen fraction or a finite number of his belief states.

Example 3.1 In [TW91], we studied the following example of coin-flip protocol which can use any secret key cryptosystem.

1. A chooses randomly a key k in $\mathcal{K}$ and different messages m_1, m_2 in $\{T, H\}$. A sends $em_1 = E(k, m_1)$ (the enciphering of m_1 under key k) and $em_2 = E(k, m_2)$ to B.
2. B picks u in $\{em_1, em_2\}$ and sends u to A.
3. A sends k to B and they both know the answer a of the coin flip by applying the deciphering transformation D to u.

B's knowledge at the end of the first step is modeled as in Subsection 2.2 by $K(B,1) = F(B,1) \cup V(B,1) \cup SV(B,1)$:

$$F(B,1) = \{T, H\};\ \ V(B,1) = \{\tilde{k}\};\ \ SV(B,1) = \{em_1^*, em_2^*\}$$

where

$$\begin{aligned} \tilde{k} \in \mathcal{K}\ ;\ \ em_1^*, em_2^* &\in \{e(\tilde{k}, T), e(\tilde{k}, H)\} \\ em_1^* &\neq em_2^*. \end{aligned}$$

The representation of B's knowledge has not changed at the end of the second step and becomes at the end of the third step $K(B,3) = F(B,3) \cup V(B,3) \cup SV(B,3)$:

$$F(B,3) = \{T, H, k\}; \;\; V(B,3) = \emptyset; \;\; SV(B,3) = \emptyset.$$

This model is not sufficient: it only represents the knowledge that B directly obtains but not the inferences that he is able to draw. Indeed, at the end of the first step, B tries to obtain more informations. For this, he studies some belief states. If he considers the right instantiation of the key k, he will find out that it is the right value and B's knowledge at the end of the first step can be represented by $K(B,3)$. We have to prove that the probability that happens is zero. Moreover, the transmission of k by A at the third step enables B to verify that em_1 and em_2 had the right form. Otherwise, an inconsistency is detected. This is not translated in $K(B,3)$. In fact, B computes his seen fraction which contains $e(k,T)$ and $e(k,H)$ at the end of the third step; then he compares their crypto-algebra components with the crypto-algebra components of em_1^* and em_2^* and deduces from that the values of em_1^* and em_2^* if there was no cheat at step 1 or detects an inconsistency if there was a cheat. We have to translate these inferences in our model. ∎

Let us now consider simple abstract examples in order to introduce our model.

3.1.1 Inferences Obtained by Computing the Participant's Seen Fraction

Assume that the state of knowledge $K = F \cup V \cup SV$ of a participant is the following:

$$F = \{k_1, m_1\}; V = \{\tilde{k}, \tilde{m}\}; SV = \{e(\tilde{k}, \tilde{m})\}, \quad \tilde{k} \in \mathcal{K}, \tilde{m} \in \mathcal{M}.$$

The seen fraction of this participant contains the pair formed by the element $e(k_1, m_1)$ and the corresponding element of the crypto-algebra. If the images of $e(k_1, m_1)$ and $e(\tilde{k}, \tilde{m})$ under this seen fraction coincide, the participant will directly deduce that $\tilde{k} = k_1$ and $\tilde{m} = m_1$: we will say that there has been a *unification* of the elements $e(k_1, m_1)$ and $e(\tilde{k}, \tilde{m})$ of the free-algebra and $\tilde{k} = k_1$ and $\tilde{m} = m_1$ are the *constraints required by this unification*. If the images of $e(k_1, m_1)$ and $e(\tilde{k}, \tilde{m})$ are different, the user will directly deduce that $\tilde{k} \neq k_1$ or $\tilde{m} \neq m_1$: we will then talk about a *contra-unification* because it is as the opposite of a unification and $\tilde{k} \neq k_1$ or $\tilde{m} \neq m_1$ are the *constraints required by this contra-unification.*

However, the second components of the elements in the seen fraction of a user are fixed sequences of bits. If a given element of the free-algebra has as image two different elements of the crypto-algebra, there is some inconsistency in the corresponding state of knowledge and a failure is detected: it can be a failure in the communication or a cheating of another participant or of an opponent. These cases are treated in the same way and result in the rejection by the user of the current execution of the protocol.

Moreover, if the domain of the seen fraction of a user contains some variables of V, he directly deduces the value of these variables: they have thus to be removed from the set V. In order to have a uniform model for all the inferences drawn from the computation of the seen fraction, we treat this case as a unification of elements of the crypto-algebra. Indeed, the state of knowledge $K = F \cup V \cup SV$ is represented by giving the free-algebra component of the elements of F, the variables of V and SV, and the corresponding constraints defined in the free-algebra. K can thus be considered as a function which associates with the free-algebra component of an element of F its crypto-algebra component, to a variable of V its dual variable, and to a variable of SV its image under SV; K can then be written in a way rather similar to the representation (1) of a belief state

$$K = F \cup \{(\widetilde{v}, \widetilde{v}^d) : \widetilde{v} \in V\} \cup \{(x^*, SV(x^*)) : x^* \in SV\} \qquad (2)$$

where $\widetilde{v}^d$ denotes the dual variable of v. If the domain of $Cl(F \cup SV)$ contains some variables of V, each of these variables will have for image under $Cl(F \cup SV)$ a sequence of bits fixed in the crypto-algebra and under V its dual variable: it is then sufficient to unify these two images i.e. to restrict the domain of the dual variable to a singleton containing the sequence of bits in order to model the additional information obtained by the user.

3.1.2 Inferences Obtained from the Closure of Belief States

Let us assume now that $K = F \cup V \cup SV$ is the following knowledge state

$$F = \emptyset; V = \{\widetilde{\boldsymbol{k}}, \widetilde{m}\}; SV = \{e(\widetilde{\boldsymbol{k}}, \widetilde{m})\}; \widetilde{\boldsymbol{k}} \in \mathcal{K}, \widetilde{m} \in \mathcal{M}. \qquad (3)$$

The user thus knows the element of the crypto-algebra corresponding to $e(\widetilde{k}, \widetilde{m})$ but not the ones corresponding to $\widetilde{\boldsymbol{k}}$ and $\widetilde{m}$.

Let Be be a belief state compatible with K such that

$$Be = \{(\widetilde{\boldsymbol{k}}, \boldsymbol{k}^c), (\widetilde{m}, m^c)\} \cup \{e(\widetilde{\boldsymbol{k}}, \widetilde{m}), SV(e(\widetilde{\boldsymbol{k}}, \widetilde{m}))\}; \text{ for some } \boldsymbol{k}^c \in \mathcal{K}, m^c \in \mathcal{M} \qquad (4)$$

where $\mathcal{K}$ and $\mathcal{M}$ represent respectively the set of keys and the set of messages in the crypto-algebra. If, in the computation of the closure of the belief state, the encryption of m^c under the key $\boldsymbol{k}^c$ gives $SV(e(\widetilde{\boldsymbol{k}}, \widetilde{m}))$, the user can deduce from this that the value of $\widetilde{\boldsymbol{k}}$ truly is $\boldsymbol{k}^c$ and that the value of $\widetilde{m}$ is m^c. On the other hand, if the encryption of m^c under $\boldsymbol{k}^c$ does not yield $SV(e(\widetilde{\boldsymbol{k}}, \widetilde{m}))$, the user will reject this belief state.

However, when we want to model and systematize this reasoning, we have some difficulties in the case where the belief state associates the right values to $\tilde{k}$ and $\tilde{m}$ because nothing enables us to distinguish the case where the image of the free-algebra element $e(\tilde{k}, \tilde{m})$ is computed from the values k^c and m^c from the case where we simply take its image under SV. We have to distinguish these two ways of computing this image. Therefore, let us reason on the notion of belief state.

In a belief state, the variables of V are instantiated by elements of the domain of their dual variables without considering the elements of SV which are expressed as functions of those variables: because of the isomorphism between $\mathcal{F}$ and $\mathcal{C}$, the variables of V appearing in the expressions representing the domain of the elements of SV cannot be replaced by their instantiations except if these instantiations correspond to their real values. We thus distinguish the variables of V from the free-algebra elements corresponding to their instantiations under the isomorphism φ between $\mathcal{F}$ and $\mathcal{C}$. We will denote these elements by overindexing them by the letter 'f' (for $f$ixed and $f$ree-algebra). In the above example, the belief state becomes

$$Be = \{(k^f, k^c), (m^f, m^c)\} \cup \{(e(\tilde{k}, \tilde{m}), SV(e(\tilde{k}, \tilde{m})))\}; k^c \in \mathcal{K}, m^c \in \mathcal{M}. \qquad (5)$$

Thus, k^f (resp. m^f) is the free-algebra element image under φ^{-1} of the crypto-algebra element k^c (resp. m^c).

We proceed then exactly in the same way as for the computation of the seen fraction of a user. If the images of $e(k^f, m^f)$ and $e(\tilde{k}, \tilde{m})$ under the closure of Be are the same, we unify these two expressions and deduce that

$$\tilde{k} = k^f \text{ and } \tilde{m} = m^f;$$

if these images are not identical, we deduce by what we have called a "contra-unification" that

$$\tilde{k} \neq k^f \text{ or } \tilde{m} \neq m^f,$$

which amounts to rejecting the belief state Be under consideration.

The example that we have just examined is extremely simple. Let us complicate slightly the knowledge state K which becomes for instance

$$F = \emptyset; V = \{\tilde{k}_1, \tilde{m}_1, \tilde{k}_2, \tilde{m}_2\}; SV = \{z^*\}; \qquad (6)$$

$$\tilde{k}_1, \tilde{k}_2 \in \mathcal{K}; \tilde{m}_1, \tilde{m}_2 \in \mathcal{M}; z^* \in \{e(\tilde{k}_1, \tilde{m}_1), e(\tilde{k}_2, \tilde{m}_2), d(\tilde{k}_1, e(\tilde{k}_2, \tilde{m}_1))\}.$$

A belief state compatible with K could then be

$$\begin{aligned} Be \quad = \quad & \{(k_1^f, k_1^c), (m_1^f, m_1^c), (k_2^f, k_2^c), (m_2^f, m_2^c)\} \quad \cup \quad \{e(\tilde{k}_1, \tilde{m}_1), SV(z^*)\}; \qquad (7) \\ & k_1^c, k_2^c \in \mathcal{K}; m_1^c, m_2^c \in \mathcal{M}. \end{aligned}$$

If the encryption of m_1^c under k_1^c is equal to $SV(z^*)$, we can not directly deduce that $\tilde{k}_1 = k_1^f$ and $\tilde{m}_1 = m_1^f$; that would only be true if the instantiation of z^* is correct (otherwise, we would have $\tilde{k}_2 = k_1^f$ and $\tilde{m}_2 = m_1^f$): note that the instantiation $i(z^*) = d(\tilde{k}_1, e(\tilde{k}_2, \tilde{m}_1))$ can directly be rejected but not the instantiation $i(z^*) = e(\tilde{k}_2, \tilde{m}_2)$. In the same way, if this encryption is not equal to $SV(z^*)$, we can deduce that $\tilde{k}_1 \neq k_1^f$ or $m_1 \neq \tilde{m}_1^f$, only if the instantiation of z^* is correct.

We see that the conclusions obtained by computing the closure of a belief state can be considered only under the auxiliary hypothesis that the instantiation of some variables relative to this belief state corresponds to the current instantiation of these variables. When can we remove this hypothesis and really obtain some information to add to the state of knowledge?

Before answering this question, let us remark that in the case of the belief state (4), we have directly been able to add the deduced information to the state of knowledge without any auxiliary hypothesis. The difference between the states of knowledge (3) and (6) is that in (3) the set SV contains an element whose domain can be considered as a singleton which only varies when the variables of V vary; whereas in (6) the domain of z^* contains three elements and thus for the same values of the variables of V, z^* can take three different values. The problem in (6) appears because two elements in the domain of z^* have the same form i.e. are unifiable. The instantiation of the element of SV in (3) i.e. the free-part of the instantiation relative to the state of belief (4) is thus always correct contrary to the one relative to the state of belief (7).

Note also that with the notation used in relations (5) and (7), the variables of V appear only in the instantiated values of the elements of SV. The different constraints are thus necessarily the result of a unification of the instantiated values of variables of SV with other computed elements and are valid only if these instantiated values correspond to the real values of the elements of SV. We thus see that the auxiliary hypothesis is concerned only with the elements of SV, i.e. with the free-part of the instantiation relative to the belief state.

A constraint will be added to the state of knowledge of a user only if it is valid for every instantiation of the elements of SV. For a given instantiation of the variables of V, we have thus to successively study the different possible instantiations of the variables of SV: a constraint will then only be considered if it appears for all the instantiations of the variables of SV. In other words, a constraint will be added to the state of knowledge of a user only if it is required by the unifications and contra-unifications made in the analysis of each of the states of belief whose the relative instantiations have the same crypto-part. At the end of Section 2, we have grouped these belief states in an equivalence class for the relation

"their relative instantiations have the same crypto-part".

Each equivalence class contains a finite number of belief states because the domain of the elements of SV expressed in terms of the variables of V is finite.

A constraint on the variables required by a unification or contra-unification in the computation of the closure of a belief state can be added to the state of knowledge only if it is deduced from the computation of the closure of each (nonrejected) belief state of a same class. We will then say that this constraint is *characteristic of the considered class*. A way of obtaining the constraints characteristic of a class consists in taking the disjunction of all the constraints required in the analysis of the belief states of this class.

Remark 3.1 In the case of the above example (6), each instantiation of the variables $\widetilde{k_1}, \widetilde{k_2}, \widetilde{m_1}, \widetilde{m_2}$ leads to an equivalence class which contains three belief states corresponding to the three possible instantiations of z^*. If the encryption of m_1^c under k_1^c is equal to $SV(z^*)$, the constraint

$$(\widetilde{k_1} = k_1^f \text{ and } \widetilde{m_1} = m_1^f) \text{ or } (\widetilde{k_2} = k_1^f \text{ and } \widetilde{m_2} = m_1^f)$$

is characteristic of the class of the belief state Be given in (7) and can be added to the state of knowledge (6), but if this encryption is not equal to $SV(z^*)$, the constraint

$$(\widetilde{k_1} \neq k_1^f \text{ or } \widetilde{m_1} \neq m_1^f) \text{ or } (\widetilde{k_2} \neq k_1^f \text{ or } \widetilde{m_2} \neq m_1^f) \text{ or } (z^* = d(\widetilde{k_1}, e(\widetilde{k_2}, \widetilde{m_1})))$$

is characteristic of the class of Be. ∎

We assume that a user is able to compute the closure of any fixed finite number of belief states. The best strategy to add as much information as possible to a state of knowledge seems to consist in analyzing all the belief states of some classes (rather than to consider some belief states of some classes): the disjunction of the obtained constraints can then directly been added to the state of knowledge without any auxiliary hypothesis. Because these classes are finite, we can assume that a user is able to compute the closure of all the belief states of a fixed finite number of classes. A user who wants to obtain a maximum of informations will thus choose a finite number of classes and analyze all the belief states of these classes: the choice of a finite number of belief states without considering their belonging to the equivalence classes would be much less useful because the constraints obtained by their analysis would only be valid under the auxiliary hypothesis that the corresponding instantiation of the variables of SV is correct.

3.1.3 Summary of the Possible Inferences

The seen fraction of a user and the closure of each belief state of a finite number of chosen classes are successively examined. Several cases can appear.

- When two elements of the free-algebra (resp. of the crypto-algebra) have for image (resp. are image of) a same element of the crypto-algebra (resp. of the free-algebra), we have to be able to unify these two elements; if it is not possible, either a failure is detected if the seen fraction of the user is analyzed or, in the case of the computation of the closure of a belief state, this belief state is rejected. Note that the rejection of a belief state can be translated as a constraint on the variables: at least one of the variables is not instantiated as specified in this belief state. If all the belief states are rejected at the end of the analysis, a cheating has been detected.
- If two unifiable elements (i.e. such that we can reduce the domain of the variables so that they become equal) of the free-algebra have different images in the crypto-algebra, we introduce what we call a contra-unification, i.e. we exclude the values of variables which would unify these two elements.

In the case of the computation of the seen fraction, the constraints required by the unifications and contra-unifications are directly added to the state of knowledge; in the case of the closure of a belief state, they are added only if they are characteristic of the class of this belief state. Moreover, the domains of the variables satisfying all these constraints have to be not empty; otherwise an attempt to cheat is again detected and the current execution of the protocol is stopped and rejected.

Now, we are going to formalize all these different concepts.

3.2 Unifiable Elements

Before building the inferred state of knowledge of a user, which has to contain all the informations directly obtained or deduced by this user, we need to introduce some definitions and make some remarks.

The notion of 'domain' which is very usual in mathematics can seem a little strange here for the elements of V: as we have defined in Subsection 2.2, the domain of these elements is the free-algebra subset corresponding to the domain of their dual variable. Moreover, by extension, we say that the domain of a constant of the free-algebra or the crypto-algebra is the singleton including this constant.

As we have seen above, in a given state of knowledge $K = F \cup V \cup SV$, the variables introduced in the free-algebra are the variables of V and SV, and those introduced in the crypto-algebra are the dual variables. The free and crypto-algebras components of the elements of the seen fraction and of the closure of a belief state can be expressed in terms of the corresponding variables. The problem is to unify two elements of one of the two algebras.

Let $\mathcal{A}$ be the free-algebra or the crypto-algebra; the set of variables defined in $\mathcal{A}$ given the state of knowledge K of a user is denoted $V_{\mathcal{A}}$. As we have seen previously, the variables of $V_{\mathcal{A}}$ are introduced in the representation of the state of knowledge K and thus depend on this state of knowledge. Let us consider the elements of $\mathcal{A}$ which can be expressed in terms of the variables of $V_{\mathcal{A}}$ and of the corresponding enciphering and deciphering operators. Unifying two elements of $\mathcal{A}$ in the state of knowledge K amounts to limiting the domain of the variables of $V_{\mathcal{A}}$ so that these two elements are equal for the remaining values of the variables.

Let $K = F \cup V \cup SV$ be a (assumed to be representable) state of knowledge, the set $V_{\mathcal{A}}$ is finite and can be denoted

$$V_{\mathcal{A}} = \{v_1, \ldots, v_n\}$$

with the constraints

$$v_1 \in D_{v_1}, \ldots, v_n \in D_{v_n}$$

where $D_{v_1}, \ldots, D_{v_n}$ denote respectively the domain of $v_1, \ldots, v_n$. An element of $\mathcal{A}$ expressed in terms of $0, 1, \ldots$ or n variables of $\mathcal{A}$ and of the enciphering and deciphering operators is denoted by an expression like

$$exp(v_1, \ldots, v_n);$$

its domain is

$$\{exp(v_1, \ldots, v_n) : v_1 \in D_{v_1}, \ldots, v_n \in D_{v_n}\}$$

and is denoted here briefly

$$exp(D_{v_1}, \ldots, D_{v_n}).$$

The unification of the two elements $exp_1(v_1, \ldots, v_n)$ and $exp_2(v_1, \ldots, v_n)$ amounts to replacing their domain by the set

$$\{exp_1(v_1, \ldots, v_n) : v_1 \in D_{v_1}, \ldots, v_n \in D_{v_n} \text{ such that } exp_2(v_1, \ldots, v_n) = exp_1(v_1, \ldots, v_n)\}.$$

We will call this set *the parametric intersection* of the domains of the elements

$$exp_1(v_1, \ldots, v_n) \text{ and } exp_2(v_1, \ldots, v_n)$$

because it is the union on all the variables of the intersection of the domains of these two elements i.e.

$$\bigcup_{v_1 \in D_{v_1}, \ldots, v_n \in D_{v_n} :\, exp_1(v_1, \ldots, v_n) = exp_2(v_1, \ldots, v_n)} \{exp_1(v_1, \ldots, v_n)\}.$$

We would like to reduce this intersection to the two equivalent forms

$$\{exp_1(v_1, \ldots, v_n) : v_1 \in D'_{v_1}, \ldots, v_n \in D'_{v_n}\} \text{ or } \{exp_2(v_1, \ldots, v_n) : v_1 \in D'_{v_1}, \ldots, v_n \in D'_{v_n}\}$$

where $D'_{v_1} \subseteq D_{v_1}, \ldots, D'_{v_n} \subseteq D_{v_n}$ or to a finite (disjoint) union of such sets. In our notation, the two sets above are respectively represented by

$$exp_1(D'_{v_1}, \ldots, D'_{v_n}) \text{ and } exp_2(D'_{v_1}, \ldots, D'_{v_n}).$$

When such a reduction is possible, we say that the parametric intersection of the domains of the two elements is *projectable* and we have the following definition.

Definition 3.1 *Let $K = F \cup V \cup SV$ be a given state of knowledge; $\mathcal{A}$ denotes the free or crypto-algebra and $V_{\mathcal{A}} = \{v_1, \ldots, v_n\}$ ($v_1 \in D_{v_1}, \ldots, v_n \in D_{v_n}$) is the set of variables in $\mathcal{A}$ where D_{v_i} denotes the domain of v_i ($i = 1, \ldots, n$). Let $exp_1(v_1, \ldots, v_n)$ and $exp_2(v_1, \ldots, v_n)$ be two elements of $\mathcal{A}$ expressed in terms of the variables of $V_{\mathcal{A}}$ and the operators of $\mathcal{A}$. The parametric intersection of their domains is* projectable *in the state of knowledge K if there is a finite partition of this intersection, denoted*

$$\{\{exp_1(v_1, \ldots, v_n), v_1 \in D^{(i)}_{v_1}, \ldots, v_n \in D^{(i)}_{v_n}\}; i = 1, \ldots, r\}$$

such that, for every $i = 1, \ldots, r$,

$$D^{(i)}_{v_1} \subset D_{v_1}, \ldots, D^{(i)}_{v_n} \subset D_{v_n}.$$

Intuitively, the parametric intersection of the domains of two elements of $\mathcal{A}$ is projectable in K if it can be partitioned in a finite number of sets which can be represented by any of the two expressions representing the initial domains but where the variables $V_{\mathcal{A}}$ vary on more limited domains.

The unification of the two elements $exp_1(v_1, \ldots, v_n)$ and $exp_2(v_1, \ldots, v_n)$ of Definition 3.1 will consist in replacing the domains of these elements by their parametric intersection i.e. in requiring of the variables of $V_{\mathcal{A}}$ the following additional constraint

$$\bigvee_{i=1}^{r} (v_1 \in D^{(i)}_{v_1} \wedge \ldots \wedge v_n \in D^{(i)}_{v_n});$$

this constraint is called the '*constraint required* by the unification'.

Moreover, the contra-unification of the two elements

$$exp_1(v_1, \ldots, v_n) \text{ and } exp_2(v_1, \ldots, v_n)$$

of Definition 3.1 will consist in preventing these two elements from belonging to the parametric intersection of their domains i.e. in requiring of the variables V_A the following additional constraint

$$\bigwedge_{i=1}^{r} (v_1 \notin D_{v_1}^{(i)} \vee \ldots \vee v_n \notin D_{v_n}^{(i)});$$

this constraint is called the '*constraint required* by the contra-unification'.

For simplicity, we assume that the parametric intersection of the domains of any two elements of the free or crypto-algebra is projectable in any state of knowledge of a user. Note that, in most practical cases, the partition in Definition 3.1 includes only one term.

We can now precisely define 'unification' and 'contra-unification'.

Definition 3.2

- *Two elements of the free-algebra or of the crypto-algebra are* unifiable *in a state of knowledge $K = F \cup V \cup SV$ if the parametric intersection of their domains is not empty.*
- *Let E_1 and E_2 be two unifiable elements of the free (resp. crypto)-algebra in a state of knowledge $K = F \cup V \cup SV$.*

  The unification of E_1 and E_2 *consists in limiting the domains of the variables of V and SV (resp. of the dual variables) by some constraints such that E_1 and E_2 become equal and vary in the parametric intersection of their initial domains. The constraints on the variables are called the* constraints required *by the unification.*

  The contra-unification of E_1 and E_2 *consists in limiting the domains of the variables of V and SV (resp. of the dual variables) by some constraints such that E_1 and E_2 can not become equal and thus can not belong to the parametric intersection of their initial domains. The constraints on the variables are called the* constraints required *by the contra-unification.*

3.3 Inferred States of Knowledge and "Known Fractions" of the Participants

When can we conclude that a state of knowledge is consistent or, on the contrary, that an attempt of cheating has occurred? As we have already mentioned previously, we analyze the seen fraction of a user, then successively the closure of belief states belonging to a finite number of equivalence classes as defined in Subsections 2.2 and 3.1.2. All the free-algebra elements (resp. crypto-algebra) which have the same image in the crypto-algebra (resp. which are the image of an identical element of the free-algebra) must be unified: if they are not unifiable, the state of knowledge is said to be *inconsistent* in the case of the seen fraction of the user or the belief state is rejected in the case of the closure of a belief state. The unifiable elements remaining in the free-algebra are then contra-unified. All the required constraints have to be '*compatible*' i.e. there must be at least one value in the domain of the variables which satisfies all these constraints. More generally, we have the following definition

Definition 3.3

- *A* relation *included in $\mathcal{F} \times C$ is* consistent *if*
  - *all the elements*
    - *of the domain which have the same image on one hand or*
    - *of the codomain which are image of an identical element of the domain on the other hand*

    are unifiable and
  - *all the constraints required*
    - *by these unifications and*
    - *by the contra-unifications of the unifiable elements of the domain which have not the same image*

    are compatible.
- *The* unified function *of a consistent relation f included in $\mathcal{F} \times C$ is the one to one function obtained by*
  - *unification of the elements of the domain (resp. codomain) which have the same image in the codomain (resp. are image of an identical element of the domain) and*
  - *contra-unification of the other unifiable elements of the domain.*

  The procedure to obtain the unified function from a consistent relation is called the unification of this relation.

The unified function of a relation can be seen as this relation to which the constraints required by the possible unifications and contra-unifications are imposed.

We want to apply these definitions to the state of knowledge of a participant in an execution of a protocol in order to determine if he will accept or reject this execution. A state of knowledge is said 'consistent' if the probability that the different inferences drawn by the corresponding participant (for any adopted strategy) do not introduce any inconsistency is 1, i.e. if the probability that this participant chooses to analyze a finite number of classes of belief states such that all the deduced informations are compatible is 1. (Remark that, if the number of classes is infinite, that does not necessarily mean that all these informations have to be compatible for every choice of a finite number of classes).

In the following definition, we explicitly consider a state of knowledge and the corresponding sets F, V, SV as sets of pairs belonging to $\mathcal{F} \times \mathcal{C}$.

Definition 3.4 *A* state of knowledge $K = F \cup V \cup SV$ *is* consistent *if*

1. *$Cl(F \cup SV) \cup V$ is consistent and*
2. *the probability to choose a fixed finite number (which depends on the computational power of the corresponding user) of classes of belief states compatible with K such that all the following constraints are compatible is 1 (for any adopted strategy to choose the belief states). These constraints are*
   - *the constraints required by the unification of $Cl(F \cup SV) \cup V$,*
   - *the constraints representing the rejection of the belief states belonging to one of these classes and whose closure is not consistent,*
   - *the constraints required by the unification of the closure of the consistent belief states of these classes and characteristic of at least one of these.*

An 'inferred state of knowledge' is a state of knowledge increased by all the informations that the corresponding participant can deduce by inferences as specified in the following definition.

Definition 3.5 *Given a consistent state of knowledge $K = F \cup V \cup SV$ and a finite number of classes of belief states compatible with K, the* inferred state of knowledge *is the state of knowledge $K_i = F_i \cup V_i \cup SV_i$ obtained from K by the addition of all the constraints*

- *required by the unification of $Cl(F \cup SV) \cup V$,*

- *representing the rejection of the belief states belonging to one of the considered classes and which closure is not consistent,*
- *required by the unification of the closure of the consistent belief states of these classes and characteristic of at least one of these.*

Note that the inferred state of knowledge defined in Definition 3.5 depends on the choice of the classes of belief states: when we do not specify the chosen classes, we talk about '*an*' inferred state of knowledge.

The purpose of a malicious participant is to detect the belief states whose closure is consistent. If the number of classes of belief states is finite, the participant can analyze the consistency of the closure of each state of belief: in that case, we assume that n (the number of classes whose all the states of belief can be examined) is larger than the total number of classes of belief states. If the number of classes of belief states is infinite, an exhaustive analysis is impossible; the participant can test the consistency of the closure of belief states of only a finite number of classes that he chooses.

Remark that if its set SV is empty (it is for example the case in a simple state of knowledge), the state of knowledge K is always consistent and can not be increased: the inferred state of knowledge is identical to the initial state of knowledge K.

An inferred state of knowledge is thus a representation of all the information that the user is able to obtain. We can extend the definition of the 'known fraction of a user' from that.

Definition 3.6 *Given a state of knowledge K and an inferred state of knowledge $K_i = F_i \cup V_i \cup SV_i$, the* inferred known fraction *of the corresponding user is the closure of F_i under the operations of the free and crypto-algebras whereas the* inferred seen fraction *of this user is the closure of $(F_i \cup SV_i)$ under the same operations.*

Note that these inferred known and seen fractions are not unique but depend on the choice of the classes of belief states (because the given inferred state of knowledge K_i depends on that choice): when we do not specify the chosen classes, we talk about '*an*' inferred known fraction and '*an*' inferred seen fraction.

Example 3.2 The knowledge states of user A at the different steps of Example 3.1 are always simple: they are thus consistent. Whereas, at the end of the first step, B has an infinite number of belief states which correspond to the different possible instantiations of the variables $\tilde{k}, em_1^*$ and em_2^*: these belief states are divided in an infinite number of equivalence classes depending on the instantiation of $\tilde{k}$. When user B chooses a finite set of classes of his belief states, i.e. in fact a finite set of instantiations of the variable $\tilde{k}$, two cases are possible:

- if the real instantiation of $\tilde{k}$ is chosen, B will find out, by unifications, that it is the right value; B's inferred state of knowledge is then

$$F(B,1) = \{T, H, k\}; V(B,1) = \emptyset; SV(B,1) = \emptyset;$$

- if the real instantiation of $\tilde{k}$ is not chosen, all the belief states of the chosen classes are rejected and the chosen values of $\tilde{k}$ are removed from its domain; B's inferred state of knowledge is his initial state of knowledge where a finite number of values are removed from the domain of $\tilde{k}$.

Note that, in any case, there still are values of the variables which satisfy all the required constraints: the states of knowledge are thus consistent.

The security of our coin-flip example is preserved only if B is not able to find the value of the variable $\tilde{k}$. Intuitively, choosing the real value of $\tilde{k}$ has a probability equal to zero and in the other case, rejecting a finite number of belief states does not matter. In the full paper, we will prove this formally by applying the probabilistic measure introduced in [Tou91]. Moreover, when B analyzes his seen fraction at the end of the third step, this fraction contains $e(k,T)$, $e(k,H)$, em_1^*, em_2^*. If the crypto-algebra components of em_1^* and $e(k,T)$ or $e(k,H)$ are identical, B deduces the values of em_1^* and em_2^*. Otherwise, an inconsistency is detected. ∎

4 Conclusions

We have proposed a model of all the knowledge that a participant in a cryptographic protocol can obtain by inferences and computations. Among other things, our model enables us to represent the probabilistic knowledge of the participants in a cryptographic protocol and to prove some probabilistic properties of these protocols. This representation is necessary to enable us to find out by the method described in [TW91] and [Tou91] all the possible attacks of the participants and of the intruders against cryptographic protocols.

Acknowledgements

A large part of this work was developed in collaboration with Prof. P.Wolper; I would like to thank him very much. I wish to thank also Dr. M.Merritt and Prof. J.-J.Quisquater for several helpful discussions about this work. I thank also Philippe Lejoly for useful commentaries.

References

[BAN89] M. Burrows, M. Abadi, and R. Needham. A Logic of Authentication. Technical Report 39, Digital — Systems Research Center (SRC), 1989.

[Bie89] P. Bieber. *Aspects Epistémiques des Protocoles Cryptographiques.* PhD thesis, Université Paul-Sabatier de Toulouse (Sciences), October 1989.

[BM84] M. Blum and S. Micali. How to Generate Cryptographically Strong Sequences of Pseudo-Random Bits. *SIAM Journal on Computing*, 13(4):850–864, 1984.

[Dam87] I. B. Damgard. *The Application of Claw Free Functions in Cryptography; Unconditional Protection in Cryptographic Protocols.* PhD thesis, Mathematical Institute, Aarhus University (Denmark), 1987.

[EGS86] S. Even, O. Goldreich, and A. Shamir. On the Security of Ping-Pong Protocols using the RSA. In H. C. Williams, editor, *Lecture Notes in Computer Science. Advances in Cryptology — CRYPTO'85, #218*, pages 58–72. Springer-Verlag, 1986.

[GMR89] S. Goldwasser, S. Micali, and C. Rackoff. The Knowledge Complexity of Interactive Proof-Systems. *SIAM Journal on Computing*, 18(1):186–208, 1989.

[GNY90] L. Gong, R. Needham, and R. Yahalom. Reasoning about Belief in Cryptographic Protocols. In *Proceedings of the 1990 IEEE Computer Society Symposium on Research in Security and Privacy*, pages 234–248. IEEE Computer Society Press, 1990.

[Kem89] R. A. Kemmerer. Analyzing Encryption Protocols Using Formal Verification Techniques. *IEEE Journal on Selected Areas in Communications*, 7(4):448–457, 1989.

[MCF87] J. K. Millen, S. C. Clark, and S. B. Freedman. The Interrogator: Protocol Security Analysis. *IEEE Transactions on Software Engineering*, 13(2):274–288, 1987.

[Mea89] C. Meadows. Using Narrowing in the Analysis of Key Management Protocols. In *Proceedings of the 1989 IEEE Symposium on Research in Security and Privacy*, pages 138–147. IEEE Computer Society Press, 1989.

[Mea90] C. Meadows. Representing Partial Knowledge in an Algebraic Security Model. In *Proceedings of the Computer Security Foundations Workshop III*, pages 23–31. IEEE Computer Society Press, 1990.

[Mer83] M. J. Merritt. *Cryptographic Protocols.* PhD thesis, Georgia Institute of Technology, 1983.

[Moo88] J. H. Moore. Protocol Failures in Cryptosystems. *Proceedings of the IEEE*, 76(5):594–602, May 1988.

[MW85] M. Merritt and P. Wolper. States of Knowledge in Cryptographic Protocols (extended abstract). Unpublished Manuscript, 1985.

[Syv91] P. Syverson. The Use of Logic in the Analysis of Cryptographic Protocols. In *Proceedings of the 1991 IEEE Symposium on Research in Security and Privacy*, pages 156–170. IEEE Computer Society Press, 1991.

[Tou91a] M-J. Toussaint. Formal Verification of Probabilistic Properties in Cryptographic Protocols (Extended Abstract). to appear in the proceedings of ASIACRYPT'91, 1991.

[Tou91b] M.-J. Toussaint. *Verification of Cryptographic Protocols.* PhD thesis, Université de Liège (Belgium), 1991.

[TW91] M-J. Toussaint and P. Wolper. Reasoning about Cryptographic Protocols (Extended Abstract). In Joan Feigenbaum and Michael Merritt, editors, *Distributed Computing and Cryptography (October 1989)*, pages 245–262. DIMACS - Series in Discrete Mathematics and Theoretical Computer Science (AMS - ACM), 1991. Volume 2.

[Var89] V. Varadharajan. Verification of Network Security Protocols. *Computers & Security*, 8(8):693–708, 1989.