Algebraic Intruder Deductions*

David Basin, Sebastian Mödersheim, and Luca Viganò

Information Security Group, Dep. of Computer Science, ETH Zurich, Switzerland
www.infsec.ethz.ch/~{basin,moedersheim,vigano}

Abstract. Many security protocols fundamentally depend on the algebraic properties of cryptographic operators. It is however difficult to handle these properties when formally analyzing protocols, since basic problems like the equality of terms that represent cryptographic messages are undecidable, even for relatively simple algebraic theories. We present a framework for security protocol analysis that can handle algebraic properties of cryptographic operators in a uniform and modular way. Our framework is based on two ideas: the use of modular rewriting to formalize a generalized equational deduction problem for the Dolev-Yao intruder, and the introduction of two parameters that control the complexity of the equational unification problems that arise during protocol analysis by bounding the depth of message terms and the operations that the intruder can perform when analyzing messages. We motivate the different restrictions made in our model by highlighting different ways in which undecidability arises when incorporating algebraic properties of cryptographic operators into formal protocol analysis.

1 Introduction

Motivation. Many security protocols fundamentally depend on the algebraic properties of cryptographic operators [17]. For example, protocols based on the Diffie-Hellman key-exchange, such as the Station-to-Station, IKE, and JFK protocols, exploit the property of modular exponentiation that $(g^x)^y \bmod p = (g^y)^x \bmod p$. Without this property, these protocols could not even be executed.

A number of approaches have been proposed for formally analyzing security protocols in the presence of an active intruder. Independent of which formalism is adopted, one of the core problems is the *intruder deduction problem*: given a state of the protocol execution, can the intruder derive a given message M? Derivation here is relative to the terms the intruder currently knows, i.e. relative to the closure under a set of deduction rules of his initial knowledge augmented with the messages that he has observed. The intruder deduction problem provides the basis for solving a number of practically relevant protocol analysis problems. We can, for instance, use it to determine whether the intruder is able to construct a

* This work was partially supported by the FET Open Project IST-2001-39252 and the BBW Project 02.0431, "AVISPA: Automated Validation of Internet Security Protocols and Applications", and by the Zurich Information Security Center. This work represents the views of the authors.

message of the form that some honest agent is expecting to receive, or whether he is able to obtain a message that is intended to be a secret, e.g. a key shared by two honest agents.

In this paper, we focus on the intruder deduction problem in the presence of algebraic equations that express properties of cryptographic operators. The underlying intruder model we employ is that of Dolev and Yao [19], in which the intruder observes all network traffic and can generate new messages, impersonating other agents, but cannot break cryptography. Although the Dolev-Yao intruder model is very commonly used, most analysis approaches based on this model are also based on the *free algebra assumption*. Under this assumption, two terms are equal if and only if they are syntactically equal. But, as we noted above, this is inappropriate for protocols that rely on algebraic properties.

Relaxing the free algebra assumption is however nontrivial: even for relatively simple sets of equations, the most basic problem, the *unifiability problem* (i.e. the equality of terms under substitutions for their variables), is only semi-decidable [4, 6, 23]. Moreover, even for those theories where unification is decidable, the intruder deduction problem may still be undecidable [1, 2].

Solutions for the intruder deduction problem have been given for individual algebraic theories of cryptographic operators, such as those formalizing different properties of modular exponentiation or bitwise exclusive or [12, 13, 27]. However, even though these approaches are specialized to particular algebraic properties, the algorithms and correctness proofs are quite complex and usually must be revised or completely re-designed when new properties are added. More general approaches have been recently proposed [14, 24, 26] and we compare our work with them in the concluding section §5.

Contributions. Our principal contribution in this paper is a framework for protocol analysis that is general and can handle algebraic properties of cryptographic operators in a uniform and modular way. In doing so, we pave the way for implementing analysis tools that are not specialized to particular algebraic theories and thereby allow users to declare new operators and properties as part of the protocol specifications. Of course, given the undecidability of the relevant problems, this goal cannot be achieved in full, without any restrictions. We now briefly describe the main ideas and restrictions of our proposed approach.

Our framework is based on two ideas. The first idea is to use *modular rewriting* to formalize a generalized equational deduction problem for the Dolev-Yao intruder. In doing so, we exploit the fact that we can distinguish two kinds of equational theories associated with security protocols: *cancellation theories* (where equations express that certain operations cancel each other out, such as encryption and decryption with the same symmetric key) and *finite equivalence class theories* (which are theories that induce finite equivalence classes for all terms). We show how our use of modular rewriting leads to efficient solutions to the intruder deduction problem.

The second idea is to introduce two "depth parameters" that bound the depth of message terms and the operations that the intruder can use to analyze messages (i.e. decompose messages based on his current knowledge, under perfect

cryptography). These bounds control the complexity of the equational unification problems that arise, transforming undecidable problems into decidable ones. Moreover, these bounds effectively serve as search parameters that can be used to control the search over the space of messages.

Our framework is thus parameterized by algebraic theories of the two kinds above and provides a general algorithm for the algebraic intruder deduction problem when the depth of message terms and the analysis operations of the intruder are bounded. Our framework allows us to identify several sub-problems of the intruder deduction problem (e.g. the reduction of terms to their normal forms) and provide general algorithms for them. Along the way, we also show that the problems considered become undecidable when any of the restrictions made in our framework are removed.

Two remarks are in order to help put into context our use of depth parameters. First, rather than considering specialized theories of algebraic properties of cryptographic operators, the focus of our work is to provide a general and flexible framework that supports a large class of such theories. However, in this generality, many problems are undecidable unless we introduce some restrictions. Our work shows that bounding the term depth and the message analysis by the intruder simplifies many of the problems that arise and turns undecidable problems into decidable ones. Moreover, many protocol analysis methods require bounds on messages in the first place, e.g. methods based on typed models.

Second, our algorithms are less efficient than those algorithms, when they exist, that are specialized to particular algebraic theories, e.g. [12, 13, 27], which usually work without bounds. Our framework is open to the integration of such specialized algorithms, albeit under the restriction of bounded message depth. In this way, we can benefit from research advances for specialized theories, while being able to fall back on general algorithms when specialized ones are not available.

Finally, we note that our framework is not biased towards a particular protocol analysis method. It can be used as a basis for handling algebraic equations when employing different types of formalisms (such as strand spaces, process calculi, or rewriting) or techniques (such as abstractions or the symbolic *lazy intruder* technique employed in our protocol model-checker OFMC [8, 10]).

Organization. We proceed as follows. In §2 we provide background for our approach. In §3 we introduce a concrete equational theory as a running example and give an overview of our framework, presenting the central definitions and theorems. In §4 we focus on how the intruder can analyze messages. In §5 we compare with related work and draw conclusions.

Due to lack of space, discussions, examples, and proofs have been shortened or omitted; details can be found in the extended version of this paper [9].

2 Background

Messages and cryptography. As is standard, we represent protocol messages as terms built over a finite signature Σ. We write Σ^n, for $n \geq 0$, to denote function

symbols of arity n. Terms in Σ^0 are *constants* (i.e. nullary function symbols) and represent *atomic messages* like agent names or nonces. We define the *depth* of a term t as the number of nodes in the longest path from the root to a leaf in its tree representation, and the *size* of t as the number of nodes (both inner nodes and leaves). We write $\mathcal{T}(\Sigma, V)$ to denote the set of terms that can be generated using symbols of Σ and variables from a set V, and we write $\mathcal{T}(\Sigma)$ for the set of *ground* terms.

Algebraic properties of cryptographic operators. Most approaches to protocol analysis follow the *free algebra assumption*, under which two ground terms are equal iff they are syntactically equal. Many protocols, however, do actually depend on algebraic properties of cryptographic operators, in the sense that the properties are required for the agents to carry out the steps prescribed by their protocol roles. Hence, unlike the practice of abstracting from the concrete behavior of cryptography, we cannot ignore the algebraic properties on which the protocol to be analyzed is based. For example, as we noted above, protocols based on the Diffie-Hellman key-exchange, such as the Station-to-Station, IKE, and JFK protocols (see the web-page of the IETF [21]), exploit the property of modular exponentiation that $(g^x)^y \bmod p = (g^y)^x \bmod p$. As another example, note that many protocols combine two secrets into one using *associative* and *commutative (AC) operators* like *bitwise exclusive or (xor)* $\cdot \oplus \cdot$. Given such a composed secret, every agent who knows one of the two secrets can also find out the other one, but no other agent can. For instance, if an agent knows $x \oplus y$ and x, then he can exploit the properties of $\oplus$ to compute y as $(x \oplus y) \oplus x$.

Equational Theories. The formal analysis of protocols like those above requires explicitly reasoning about the relevant properties of the cryptographic operators employed. We address in this paper those properties that are formalizable by finite sets of equations of the form $t \approx s$, where $t, s \in \mathcal{T}(\Sigma, V)$. For example, the property required for the Diffie-Hellman key-exchange is that $exp(exp(g, x), y) \bmod p \approx exp(exp(g, y), x) \bmod p$.

We assume that notions like *substitution*, *matching*, *unification*, and *unifiability* are defined as standard, e.g. as in [4, 6]. *Term positions* are represented as sequences of natural numbers, which are partially ordered by the prefix ordering. We define the *equational theory* $\approx_E$ *induced by a set* E *of equations* to be the least congruence on the term algebra that is closed under substitution and contains E. We define the *equivalence class* $[t]_{\approx_E}$ of a term t as $\{s \mid t \approx_E s\}$. Given a set E of equations, we interpret terms of $\mathcal{T}(\Sigma, V)$ in the *quotient algebra* of the term algebra with the congruence on terms, written $\mathcal{T}(\Sigma, V)/_{\approx_E}$. In this algebra, two terms are equal iff they are equivalent due to $\approx_E$. The *ground word problem* for a theory E is the problem of deciding $s \approx_E t$ for arbitrary $s, t \in \mathcal{T}(\Sigma)$. Note that, for brevity, we often refer to a set E of equations as a "theory", meaning the equational theory $\approx_E$ induced by E.

We say that a substitution σ is an *instance* of a substitution θ modulo E, and write $\sigma \gtrsim_E \theta$, iff there is a substitution λ such that $x\sigma \approx_E x\theta\lambda$ for all $x \in$

domain(θ). Given a set $\mathcal{S}$ of substitutions, $\mathcal{S}_0$ is a *complete set of substitutions of* $\mathcal{S}$ *under* E iff for all $\sigma \in \mathcal{S}$ there is a $\theta \in \mathcal{S}_0$ with $\sigma \gtrsim_E \theta$.

Definition 1. *Let* vars(t) *denote the variables of a term* t*. A* rewrite rule *is an equation* $l \approx r$*, where* l *is not a variable and* vars$(l) \supseteq$ vars(r)*. In this case, we may write* $l \to r$ *instead of* $l \approx r$*. A* term-rewriting system (TRS) *is a set of rewrite rules. A TRS* C *and an equational theory* E *induce a* modular rewriting relation *on* E*-equivalence classes of terms as follows:* $[t]_{\approx_E} \to_{C/E} [s]_{\approx_E}$ *iff there are terms* t' *and* s' *such that* $t \approx_E t'$*,* $t' \to_C s'$*, and* $s' \approx_E s$*.*

Let $\to^+$ *and* $\to^*$ *denote the transitive and the transitive-reflexive closure of a binary relation* $\to$*. Given* $\to$*, we say that* t *is* reducible *(and we call* t *a* redex*) iff* $t \to s$ *for some* s*.* t_1 *and* t_2 *are* joinable*, denoted by* $t_1 \downarrow t_2$*, iff there is some* s *such that* $t_1 \to^* s$ *and* $t_2 \to^* s$*.* t *is a* normal form *iff it is not reducible, and* s *is* a normal form of t *iff* $t \to^* s$ *and* s *is a normal form. We denote the normal form of* t *by* $t{\downarrow}$*, when it is unique. We say that* $\to$ *is* confluent *iff* $t \to^* t_1$ *and* $t \to^* t_2$ *implies that* $t_1 \downarrow t_2$*. Finally,* $\to$ *is* convergent *iff it is confluent and terminating.*

Although $\to_{C/E}$ is defined on equivalence classes of terms, for notational simplicity we will also write $t \to_{C/E} s$, for terms s and t, rather than $[t]_{\approx_E} \to_{C/E} [s]_{\approx_E}$. Employing the same convention, we will also write $t{\downarrow_{C/E}}$ for $[t]_{\approx_E}{\downarrow_{C/E}}$. Note that for a convergent relation $\to$, every term has a unique normal form, and hence $t{\downarrow_{C/E}}$ is always defined.

The definition of modular rewriting works directly on E-equivalence classes, rather than defining a special notion of convergence modulo E. However, while theoretically appealing, this definition is algorithmically difficult to work with. Therefore many approaches to modular rewriting employ a weaker but more tractable variant $\to_{C,E}$ of the relation $\to_{C/E}$, namely $s \to_{C,E} t$ iff $\exists (u \to v) \in C.\ \exists \sigma.\ s \approx_E u\sigma \wedge t = v\sigma$. For $\to_{C,E}$, there is a completion method [7, 22], and it is not necessary to explore the entire E-equivalence class of a term t in order to determine if t is a redex. While we consider here the relation $\to_{C/E}$, we remark that all constructions and algorithms in this paper can be adapted to $\to_{C,E}$ as well.

A standard result tells us that we can solve the ground word problem for terms in the theory $C \cup E$ by normalizing the terms under C and checking the results for equality modulo E. Formally, if $\to_{C/E}$ is convergent and t_1 and t_2 are ground terms, then $t_1 \approx_{C \cup E} t_2$ iff $[t_1]_{\approx_E}{\downarrow_{C/E}} = [t_2]_{\approx_E}{\downarrow_{C/E}}$.

The Dolev-Yao intruder. The standard Dolev-Yao model [19] formalizes the abilities of an intruder who controls the communication network. The intruder can analyze messages, decomposing them into submessages, and synthesize new messages from their subparts. In our formalization of this, we assume we are given a set of function symbols $\mathcal{O} \subset \Sigma$ that describe the ways of constructing messages (e.g. pairing or cryptographic operations like encryption or hashing). We also call the set $\mathcal{O}$ the set of *intruder-accessible operators*. For readability, we will however avoid displaying the set $\mathcal{O}$ as an explicit parameter of the intruder deduction problem.

Definition 2. *Given a finite set of ground terms IK (for "intruder knowledge") and an equational theory E, we define $\mathcal{DY}_E(IK)$ (for "Dolev-Yao") as the least set that is closed under the rules*

$$\frac{}{t \in \mathcal{DY}_E(IK)}\ \text{AX}\ (t \in IK)\,, \qquad \frac{t_1 \in \mathcal{DY}_E(IK)}{t_2 \in \mathcal{DY}_E(IK)}\ \text{EQ}\ (t_1 \approx_E t_2)\,,$$

$$\frac{t_1 \in \mathcal{DY}_E(IK) \quad \cdots \quad t_n \in \mathcal{DY}_E(IK)}{op(t_1, \ldots, t_n) \in \mathcal{DY}_E(IK)}\ \text{OP}\ (op \in \mathcal{O})\,.$$

The (Dolev-Yao) intruder deduction problem with respect to the equational theory *E is the problem of deciding whether $t \in \mathcal{DY}_E(IK)$ for ground terms t and finite sets of ground terms IK.*

Note that in this formalization we do not have analysis rules for decomposing terms. For example, the decryption rule for symmetric encryption

$$\frac{\{\!|m|\!\}_k \in \mathcal{DY}_E(IK) \quad k \in \mathcal{DY}_E(IK)}{m \in \mathcal{DY}_E(IK)}$$

is subsumed by the equation $\{\!|\{\!|m|\!\}_k|\!\}_k \approx m$: whenever the intruder has $\{\!|m|\!\}_k$ and k, he can compose them to construct $\{\!|\{\!|m|\!\}_k|\!\}_k$, which is equal under $\approx_E$ to m.

The intruder deduction problem is the core deduction problem in protocol analysis. Consider a trace of messages exchanged between honest agents and an intruder. For each message m that is sent by the intruder in this trace, the intruder must be able to derive m, i.e. $m \in \mathcal{DY}_E(IK)$, where E is the equational theory considered and IK is the intruder knowledge consisting of the initial intruder knowledge and all messages the intruder has observed so far. Note that in many state-of-the-art approaches to protocol analysis (see [15] for an overview), the term m may contain variables and the resulting symbolic trace represents the set of traces that are obtained by substituting for the variables arbitrary terms from $\mathcal{DY}_E(IK)$. The use of symbolic terms avoids the naïve enumeration of all terms that the intruder can generate from his knowledge.

3 A framework for algebraic properties

While equational reasoning is a general paradigm, our focus in this paper is on its application to security protocol analysis. Let us begin with a concrete example: an algebraic theory formalizing relevant properties used in many protocols, including those based on the Diffie-Hellman key-exchange.

Example 1. Let $\Sigma_{ex} = (\Sigma^0_{ex}, \Sigma^1_{ex}, \Sigma^2_{ex})$, where Σ^0_{ex} is a countable set of constants; $\Sigma^1_{ex} = \{inv(\cdot), \cdot^{-1}\}$, where $inv(t)$ and t^{-1} are the inverses of a message term t for asymmetric encryption and exponentiation, respectively, and the symbols in $\Sigma^2_{ex} = \{\{\cdot\}_\cdot, \{\!|\cdot|\!\}_\cdot, \langle\cdot,\cdot\rangle, exp(\cdot,\cdot), \cdot \oplus \cdot\}$ denote *asymmetric encryption* $\{t_2\}_{t_1}$ and *symmetric encryption* $\{\!|t_2|\!\}_{t_1}$ of a message t_2 with a message t_1, *concatenation*

$\langle t_1,t_2\rangle$ of two messages t_1 and t_2, *modular exponentiation* $exp(t_1,t_2)$ of a message t_1 with a message t_2, and *bitwise xor* $t_1 \oplus t_2$ of a message t_1 with a message t_2 (with identity element e). Our example theory E_{ex} is induced by the following equations over Σ_{ex} (where the x_i are variables from a set disjoint from Σ_{ex}):

$$x_1 \oplus x_2 \approx x_2 \oplus x_1 \quad (1)$$
$$(x_1 \oplus x_2) \oplus x_3) \approx x_1 \oplus (x_2 \oplus x_3) \quad (2)$$
$$exp(exp(x_1,x_2),x_3) \approx exp(exp(x_1,x_3),x_2) \quad (3)$$
$$exp(exp(x_1,x_2),{x_2}^{-1}) \approx x_1 \quad (4)$$
$$inv(inv(x_1)) \approx x_1 \quad (5)$$
$$({x_1}^{-1})^{-1} \approx x_1 \quad (6)$$
$$\{\{x_2\}_{x_1}\}_{inv(x_1)} \approx x_2 \quad (7)$$
$$\{\{x_2\}_{inv(x_1)}\}_{x_1} \approx x_2 \quad (8)$$
$$\{|\{|x_2|\}_{x_1}|\}_{x_1} \approx x_2 \quad (9)$$
$$x_1 \oplus x_1 \approx \mathsf{e} \quad (10)$$
$$x_1 \oplus \mathsf{e} \approx x_1 \quad (11)$$

We split E_{ex} into two subtheories: F_{ex} is induced by the equations (1)–(3), and C_{ex} is induced by the equations (4)–(11). □

Note that, as is often done, we leave implicit the modulus of exponentiation in E_{ex}: instead of $g^x \bmod p$ (i.e. $exp(g,x) \bmod p$) we write simply g^x (i.e. $exp(g,x)$), assuming that exponentiation is always performed using the same (publicly known) modulus. Note also that E_{ex} does not contain redundant equations (which are entailed by the given equations) such as $\mathsf{e} \oplus x_1 \approx x_1$.

3.1 Two kinds of theories

Our framework is based on *modular rewriting* and exploits the fact that we can distinguish two kinds of equational theories associated with security protocols: cancellation theories and modulo theories. C_{ex} is an example of a *cancellation theory*, which is a theory whose equations express that certain operations (such as encryption followed by decryption with the same key) cancel each other out. Such equations can usually be described by a convergent TRS and we can thus apply these equations to rewrite all terms into normal form. The advantage of separating out a convergent subtheory is that we can then neglect its equations during subsequent equality reasoning when all terms are normalized.

Definition 3. *A* cancellation theory *is a theory induced by* cancellation rules *of the form* $op(t_1,\ldots,t_n) \approx s$*, with* s *a constant or a subterm of one of the* t_i*.*

F_{ex} is an example of a *modulo theory*, which is a theory that comprises equations that cannot be oriented into terminating rewrite rules; the standard examples from rewriting are the equations for properties like associativity and/or commutativity. It is common for these equations to form a "background theory" used when applying other rewrite rules (such as the cancellation equations); that is, one performs rewriting modulo the equations of a modulo theory.

Here we will not restrict ourselves to a particular modulo theory, like AC, but rather work with a class of theories, namely *finite equivalence class theories*.

Definition 4. *An equational theory* E *is a* finite equivalence class (FEC) theory *if the equivalence class* $[t]_{\approx_E} = \{t' \mid t' \approx_E t\}$ *is finite for all terms* $t \in \mathcal{T}(\Sigma,V)$*.*

We can then, for example, prove that F_{ex} is an FEC theory and C_{ex} is a cancellation theory. In the following, we will use C and F to denote cancellation and FEC theories, respectively. Note also that FEC and cancellation theories are disjoint theory classes as for a cancellation theory, there are always terms with an infinite equivalence class.

As is standard, the *equational matching problem* for a theory E is the question of whether, given a ground term t and a term s with variables, there is a substitution σ such that $t \approx_E s\sigma$. From the definition of FEC theories, we have:

Theorem 1. *The equational matching problem for an FEC theory F is decidable. In particular, there is a terminating algorithm that returns a complete set of matches modulo F for a given instance of the problem.*

A special case of equational matching is the ground word problem (when s is also ground), and hence this problem is also decidable for FEC theories.

As we will see below, our framework relies on the decidability of matching for FEC theories. In contrast, the unification problem (where both terms may contain variables) for FEC theories is undecidable. Consider the theory of distributivity and associativity $D_{\star+}A_+ = \{x\star(y+z) \approx (x\star y)+(x\star z),\ x+(y+z) \approx (x+y)+z\}$. Unifiability in this theory is undecidable as shown in [28]. As equivalence classes in $D_{\star+}A_+$ are finite, we thus have that unifiability modulo an FEC theory is in general undecidable.

In §4 we will use the following important property of FEC theories, namely that they cannot contain equations that introduce new variables:

Lemma 1. *If $l \approx r$ is an equation of an FEC theory, then* $\mathrm{vars}(l) = \mathrm{vars}(r)$.

Hence, $l \in \mathcal{V}$ implies $l = r$, so that such trivial equations can be safely omitted.

We conclude this subsection by observing the relevance of these two kinds of theories to security protocol analysis. As we will see, cancellation rules are closely related to the analysis (e.g. decryption) of terms by the intruder and honest agents, and therefore have a distinguished role in deductions. We will namely define a normal form of the intruder knowledge as a state where the applications of cancellation rules do not give him any "new" terms (in a sense to be precisely defined later).

3.2 Restriction to a bounded variable depth model

As unifiability modulo an FEC theory is undecidable, we must introduce a restriction under which unification becomes decidable. We achieve this by introducing bounds on messages. There are several ways to do this, e.g. by bounding the number of operations that the intruder can perform to synthesize new messages from his knowledge, or by limiting the depth of terms that may be substituted for variables in the rules formalizing the steps of a protocol execution. We take the second approach here and bound the depth of message terms. To this end, we first define a subset of the variable symbols with an associated depth bound, and we then define which substitutions are permissible for these variables.

Definition 5. *We call a* bounded variable *a variable for which only terms with bounded depth can be substituted. Let* $\mathcal{VB} \subseteq \mathcal{V}$ *be the* set of bounded variables *such that every variable* v *has an associated depth bound* $depth(v) \in \mathbb{N}$*. We extend the function* $depth(\cdot)$ *to arbitrary terms as follows:* $depth(v) = \infty$ *for* $v \in \mathcal{V} \backslash \mathcal{VB}$*,* $depth(c) = 0$ *for* $c \in \Sigma^0$*,* $depth(op(t_1, \ldots, t_n)) = 1 + max_{i=1}^{n}\ depth(t_i)$ *for* $op \in \Sigma^n$*, with* $n > 0$*. We say that* a substitution σ respects the depth restrictions of the variables in a term t*, and write* $respect\_depth(\sigma, t)$*, iff* $depth(v\sigma) \leq depth(v)$ *for all* $v \in \mathrm{vars}(t)$*.*

We call the *bounded variable depth model (BVDM)* the restricted protocol analysis model in which only substitutions are allowed that respect the depth of variables.

The following lemma tells us that any computable function on ground terms can be extended to a computable function on terms with bounded variables. This will allow us, in the rest of this paper, to restrict ourselves to the ground case while all results can be carried over to terms with bounded variables.

Lemma 2. *Let* f *be a computable function that takes as input* n *terms that may contain variables and* m *ground terms, and which returns a finite set of terms. Then the following function* f' *is also computable.* f' *takes as input* n *terms that may contain (arbitrary) variables and* m *terms that may contain only bounded variables, and returns a finite set of terms and substitutions such that:*

$$\begin{aligned}
&\forall s_1, \ldots, s_n \in \mathcal{T}(\Sigma, V).\ \forall t_1, \ldots, t_m \in \mathcal{T}(\Sigma, \mathcal{VB}).\ \forall \sigma.\\
&\ [ground(t_1\sigma) \wedge \ldots \wedge\ ground(t_m\sigma) \wedge\ domain(\sigma) \subseteq \mathcal{VB} \wedge\\
&\qquad respect_depth(\langle s_1, \ldots, s_n, t_1, \ldots, t_m\rangle, \sigma)] \implies\\
&\ [(r, \sigma) \in f'(s_1, \ldots, s_n, t_1, \ldots, t_m) \iff r\sigma \in f(s_1\sigma, \ldots, s_n\sigma, t_1\sigma, \ldots, t_m\sigma)]\,.
\end{aligned}$$

Lemma 2 allows us, for instance, to easily lift the matching algorithm for FEC theories F to a unification algorithm where one of the two input terms contains only bounded variables.

Note that the depth of messages is often bounded in protocol analysis. For instance, many model-checking approaches bound terms to obtain a finite-state system, e.g. [3, 25]. Moreover, when other parameters of the model are unbounded, like the number of sessions, then restricting the message depth is essential for decidability [20]. Note also that [11] presents an approach that similarly bounds the depth of message terms in order to tackle the problem of algebraic properties in intruder deductions; the approach of [11] is however specialized to a particular algebraic theory.

3.3 Matching and unification in FEC theories in the BVDM

We have shown that for every FEC theory F, we can decide the matching problem. By Lemma 2, when the variables are bounded on one side, we can reduce an F-unification problem to a finite number of F-matching problems, which we can solve by Theorem 1. The algorithms that we can obtain from the constructive

proof of Theorem 1 however have poor complexity. Moreover, there exist more efficient, specialized algorithms for some of the theories that are relevant for the analysis of security protocols, e.g. [12, 13, 27].

We give a solution to handle F-unification efficiently in the bounded case and which allows for the straightforward integration of existing unification algorithms for disjoint subtheories of F. Due to lack of space, we briefly sketch this solution here and refer to [9] for details. The basic idea is the following. In a free algebra, every term $op(t_1, \ldots, t_n)$ can be decomposed into an operator and its arguments in only one way. Modulo a theory E, however, there may be other ways to decompose a term. For instance, in our example theory E_{ex}, $exp(exp(g, x), y)$ may be decomposed into the exponentiation of $exp(g, x)$ with y or the exponentiation of $exp(g, y)$ with x as these two terms are equal modulo E_{ex}.

For FEC theories, there are only finitely many ways to decompose a ground term, since its equivalence class is finite. For the BVDM, in [9] we show that given a complete decomposition algorithm for an FEC theory F, we can construct a complete one-side-bounded F-unification algorithm. The advantage of this unification algorithm is that it does not explore the entire equivalence class of terms, but rather just what different decompositions are possible at the topmost level of the term.

Moreover, we can show that FEC-decomposition has a nice compositionality property in the BVDM.[1] Let the FEC theory F be composed from disjoint subtheories F_1 and F_2 (i.e. subtheories that have no constant or function symbols in common). Consider F-unifying the two terms $t = op(t_1, \ldots, t_n)$ and $s = op'(s_1, \ldots, s_m)$. For the unification to succeed, op and op' must belong to the same subtheory, say F_1. Then, the unification problem $t \approx_F s$ can be broken into the "smaller" unification problems $t \approx_{F_1} op'(s'_1, \ldots, s'_m)$ and $s'_j \approx_F s_j$ for $1 \leq j \leq m$. That is, $t \approx_F s$ can be reduced to an F_1-problem together with F-problems for the subterms (which may belong to different subtheories). This allows us to construct an F-unification algorithm from the F_i-unification algorithms for the disjoint subtheories F_i.

3.4 Intruder deduction modulo F

So far we have considered the problem of unification and matching modulo an FEC theory F. We now turn to the intruder deduction problem modulo F, i.e. whether $t \in \mathcal{DY}_F(IK)$ holds for a ground term t and a set of ground terms IK.

Lemma 3. *If F is an FEC theory, then the problem $t \in \mathcal{DY}_F(IK)$ is decidable for a term t and a set of terms IK.*

In the following, we will consider the generalization of the problem $t \in \mathcal{DY}_F(IK)$, where the term t may contain variables. This is an important question even for a model with only ground terms, since we will later consider intruder

[1] Note that, as we discuss in more detail in [9], standard compositionality results for disjoint theories, e.g. [5], are not applicable in the BVDM since that would give rise to unbounded unification problems.

derivations modulo $F \cup C$. In particular, given a set IK of ground terms, we must decide whether there is some ground instance $t\sigma$ of the left-hand-side t of a cancellation rule of C such that $t\sigma$ can be derived modulo F from IK (note that t is here a term with unbounded variables). As shown in [9]:

Lemma 4. *There is an FEC theory F such that it is undecidable for a term t and a set of ground terms IK, whether there exists a substitution σ such that $t\sigma$ is ground and $t\sigma \in \mathcal{DY}_F(IK)$.*

Hence, to decide the intruder deduction problem for terms with variables, we must make further restrictions. By Lemma 2, the problem is decidable if t contains only bounded variables.

4 Cancellation equations

We now turn to the cancellation equations such as $\{|\{|x_2|\}_{x_1}|\}_{x_1} \approx x_2$. Such an equation cannot be formalized as part of an FEC theory like F_{ex} since all equivalence classes are infinite. As introduced in §2, we will now consider rewriting for cancellation theories C modulo an FEC theory F. Note that every cancellation theory is a rewrite theory as every cancellation equation $l \approx r$ has the property that $\mathrm{vars}(l) \supseteq \mathrm{vars}(r)$.

The principal property that we require is that the modular rewriting relation $\rightarrow_{C/F}$ is convergent, which is the case for our example $\rightarrow_{C_{ex}/F_{ex}}$, as we show in [9]. As a direct consequence of our assumption that $\rightarrow_{C/F}$ is convergent and since we can decide matchability modulo an FEC theory F by Theorem 1, we have that the ground word problem modulo $F \cup C$ is decidable in our framework:

Theorem 2. *Let F be an FEC theory and C a cancellation theory, and let $\rightarrow_{C/F}$ be convergent. Then the ground word problem for $F \cup C$ is decidable.*

By Lemma 2, it follows that we can construct a unification algorithm modulo $F \cup C$ for terms with bounded variables. In particular, this implies that the unifiability problem modulo $F \cup C$ for terms with bounded variables is decidable.

4.1 Cancellation as analysis

The results that we have presented so far allow us to decide, for ground terms or terms with bounded variables, the equality of terms modulo an FEC theory F and a cancellation theory C, as well as the intruder deduction problem in the theory F. We now consider how to solve the intruder deduction problem in the theory $F \cup C$. In §4.2, we will see that this problem is in general undecidable, so to obtain a decidable problem we must further restrict our model: we bound the number of operations that the intruder can perform.

The idea that we put forth here to solve the intruder deduction problem with respect to $F \cup C$ is to distinguish synthesis (or composition) and analysis (or decomposition) of messages by the intruder. Observe that these two aspects of intruder deduction are not completely independent; for instance, if the intruder

knows the messages $\{\!|m|\!\}_{\langle k_1,k_2\rangle}$ and k_1 and k_2, then he can analyze the encrypted message, but only after synthesizing the key $\langle k_1,k_2\rangle$. We now define a general notion of analysis based on an arbitrary cancellation theory C.

Intuitively, we speak of *synthesis* when the intruder applies the OP rule to compose terms, excluding the case when the resulting composed term is a redex according to the cancellation theory C (as we can then reduce it to a simpler term). We speak of *analysis* when the intruder applies the OP rule to obtain a redex whose normal form cannot be composed from his current knowledge. We can then formalize the notion of the intruder knowledge being completely analyzed based on the notion of cancellation rules present in our framework: we say that the intruder has analyzed his knowledge as far as possible if, by applying the cancellation rules, the intruder can only derive messages (except redices in C) that he can also derive without cancellation rules. Formally:

Definition 6. *Let C be a cancellation theory convergent modulo an FEC theory F. We say that a finite set of ground terms IK is* analyzed with respect to C modulo F *if $t{\downarrow_{C/F}} \subseteq \mathcal{DY}_F(IK)$ for each $t \in \mathcal{DY}_F(IK)$.*

As an example, consider again F_{ex} and C_{ex}. The set $IK = \{\{\!|m|\!\}_k, k\}$ is not analyzed with respect to C_{ex} modulo F_{ex} as the intruder can generate $t = \{\!|\{\!|m|\!\}_k|\!\}_k \in \mathcal{DY}_{F_{ex}}(IK)$, and $t{\downarrow_{C_{ex}/F_{ex}}} = [m]_{\approx_{F_{ex}}}$, but $m \notin \mathcal{DY}_{F_{ex}}(IK)$. However, $IK' = IK \cup \{m\}$ is analyzed since all messages that can be obtained only by normalizing terms in $\mathcal{DY}_{F_{ex}}(IK')$ are already contained in $\mathcal{DY}_{F_{ex}}(IK')$.

We thus have a characterization of analyzed intruder knowledge as a set that contains all messages that can be derived under $\mathcal{DY}_{F\cup C}(\cdot)$ but not under $\mathcal{DY}_F(\cdot)$. The idea is that when the set of messages known by the intruder is analyzed, then there is no need to consider the cancellation theory in the derivations of the intruder. Hence we can decide the intruder deduction problem $\mathcal{DY}_{F\cup C}(\cdot)$ when the intruder knowledge is analyzed:

Theorem 3. *Let F be an FEC theory and C a cancellation theory, and let $\rightarrow_{C/F}$ be convergent. Further, let t be a ground term and IK be a finite set of ground terms analyzed with respect to C modulo F. Then it is decidable whether $t \in \mathcal{DY}_{F\cup C}(IK)$.*

By Lemma 2, it follows that the intruder deduction problem is decidable for terms with bounded variables when the intruder knowledge is analyzed.

4.2 Undecidability of analysis

The previous method for solving the intruder deduction problem is restricted to the case where the intruder knowledge is analyzed. The central question thus is how to transform an arbitrary intruder knowledge into an analyzed one. As we show in [9], based on the fact that unification modulo an FEC theory is undecidable in general, it follows that it is undecidable whether a given intruder knowledge is analyzed or not:

Theorem 4. *There is an FEC theory F and a cancellation theory C, where $\rightarrow_{C/F}$ is convergent, such that it is undecidable whether a finite set of ground terms IK is analyzed with respect to C modulo F. Moreover, the intruder deduction problem $t \in \mathcal{DY}_{F \cup C}(IK)$ is also undecidable.*

Note that [1, 2] have shown that the intruder deduction problem in a theory E can be undecidable even if unification in E is decidable. Our theorem is incomparable to this result as it does not require E to be decidable.

We thus need to make further restrictions to obtain a general procedure for analyzing the intruder knowledge. We proceed by limiting the operations that the intruder can perform when analyzing a single message (i.e. the number of steps before he obtains a new redex). We define a bounded derivation of the intruder as follows:

Definition 7. *Given a finite set IK of ground terms and an algebraic theory E, we define the* k-bounded intruder model *as the least set $\mathcal{DY}^k_E(IK)$ that is closed under the rules*

$$\frac{}{t \in \mathcal{DY}^k_E(IK)}\ \mathrm{AX}^k\ (t \in IK\,,\, k \geq 0)\,, \qquad \frac{t_1 \in \mathcal{DY}^k_E(IK)}{t_2 \in \mathcal{DY}^k_E(IK)}\ \mathrm{EQ}^k\ (t_1 \approx_E t_2)\,,$$

$$\frac{t_1 \in \mathcal{DY}^k_E(IK) \cdots\ t_n \in \mathcal{DY}^k_E(IK)}{op(t_1, \ldots, t_n) \in \mathcal{DY}^{k+1}_E(IK)}\ \mathrm{OP}^k\ (op \in \Sigma^n)\,.$$

Note that, under the EQ^k rule, the use of an equivalence from E does not count as a step, i.e. it does not increase the counter k.

Definition 8. *Let F be an FEC theory and C a cancellation theory, and let $\rightarrow_{C/F}$ be convergent. Given a constant $k \in \mathbb{N}$, we say that the intruder knowledge IK, which is a finite set of ground terms, is* k-analyzed *(with respect to C modulo F) iff $t{\downarrow_{C/F}} \subseteq \mathcal{DY}^k_F(IK)$ for each $t \in \mathcal{DY}^k_F(IK)$.*

Theorem 5. *Let F be an FEC theory and C a cancellation theory, let $\rightarrow_{C/F}$ be convergent, and let $k \in \mathbb{N}$. Then it is decidable if a finite set of ground terms IK is k-analyzed (with respect to C modulo F).*

Note, however, that given a finite set of ground terms IK, there does not always exist a finite superset IK' of ground terms that is (k-)analyzed. Consider, for example, the theories $F = \{f(x) = g(h(x))\}$ and $C = \{g(X) = X\}$. Clearly, F is a FEC theory, C is a cancellation theory, and $\rightarrow_{C/F}$ is convergent. Furthermore, let $\mathcal{O} = \{f\}$ be the set of functions that the intruder can access, and let IK be a finite set of ground terms that contains a constant c. We then, for instance, have that $h(c), h(h(c)), \ldots \in \mathcal{DY}_{F \cup C}(IK)$. Thus, there is no finite set $IK' \supseteq IK$ such that IK' is analyzed. For the bounded case, observe that $g(t) \in \mathcal{DY}^k_{F \cup C}(IK \cup t)$ for any ground term t, $k \geq 1$, and $n \in \mathbb{N}$. Thus, any k-analyzed superset of IK must also contain $g^n(c)$ for any $n \in \mathbb{N}$, so it must be infinite. Hence, to complete our framework, we must be able to check bounded derivability without first computing an analyzed intruder knowledge. The following theorem tells us that this is possible:

Theorem 6. *Let F be an FEC theory and C a cancellation theory, let $\rightarrow_{C/F}$ be convergent, and let $k \in \mathbb{N}$. Then it is decidable if a ground term t can be derived from a finite set of ground terms IK, i.e. whether $t \in \mathcal{DY}^k_{F \cup C}(IK)$.*

Together with the fact that, by Lemma 2, all problems over terms with bounded variables can be reduced to problems over ground terms, we have now the basis for protocol analysis modulo algebraic theories. Namely, we can check whether a term with bounded variables — representing the set of messages that some agent in its current state can receive as a valid protocol message — can be derived from a ground intruder knowledge under the bounds that we have introduced.

5 Related work and concluding remarks

We have presented a framework for security protocol analysis that can handle algebraic properties in a uniform and modular way. It is not specialized to any particular algebraic theory and thereby allows users to declare new operators and properties as part of the protocol specification. Our framework is based on the use of modular rewriting to formalize a generalized equational deduction problem for the Dolev-Yao intruder, and on bounding the depth of message terms and the analysis operations of the intruder to control the complexity of the equational unification problems that arise. These bounds allow us to give general algorithms for the equational unification and intruder deduction problems. Moreover, under these bounds, our framework is also open to the integration of more efficient algorithms that are specialized to particular algebraic theories (and which usually work without such bounds), e.g. [12, 13, 27].

The idea of providing a general approach for integrating equational properties into security protocol analysis has recently attracted considerable attention. [18] presents an approach based on standard rewriting that supports the specification of properties like the cancellation theories of our framework. However it does not allow for properties like AC, which are handled by our FEC theories. The approach of [14] has aims similar to ours: to provide a general framework that is open to the integration of existing algorithms. This approach, however, is based on a different idea, namely ordered rewriting, and is therefore applicable to classes of theories that are incomparable to the ones that are supported by our framework. The approaches of [2, 16, 24, 26] are the most closely related to ours as they also employ modular rewriting. They differ from our work in that they are more restrictive in terms of the kinds of modulo theories that can be considered; namely they consider a fixed modulo theory (or, similarly, assume given a unification procedure for the modulo theory), or they require that the unification problems are finitary. These restrictions, however, allow them to work without the bounds required by our approach.

Our framework is not biased towards a particular analysis method, and thus can be used as a basis for handling algebraic equations when employing different types of formalisms or techniques for protocol analysis. As a concrete example, we have begun integrating our framework into our protocol model-checker OFMC [8, 10]. In this integration, the message and analysis bounds become parameters of

the protocol analysis problem, along with other parameters like the number of sessions. We can then use different search techniques (like iterative deepening) to effectively search the resulting multi-dimensional search space.

The equational reasoning problems that we considered in this paper are in general undecidable and hence one must introduce restrictions to regain decidability. The restrictions that we have introduced are motivated by the practical problems in security protocol analysis and we have begun investigating whether and how they can be applied to other equational reasoning problems.

References

1. M. Abadi and V. Cortier. Deciding knowledge in security protocols under equational theories. In *Proceedings of ICALP'2004*, LNCS 3142, pp. 46–58. Springer, 2004.
2. M. Abadi and V. Cortier. Deciding knowledge in security protocols under (many more) equational theories. In *Proceedings of CSFW'05*, pp. 62–76. IEEE Computer Society Press, 2005.
3. A. Armando and L. Compagna. Automatic SAT-Compilation of Protocol Insecurity Problems via Reduction to Planning. In *Proceedings of FORTE 2002*, LNCS 2529, pp. 210–225. Springer, 2002.
4. F. Baader and T. Nipkow. *Term Rewriting and All That*. Cambridge University Press, 1998.
5. F. Baader and K.U. Schulz. Unification in the union of disjoint equational theories: Combining decision procedures. *Journal of Symbolic Computation*, 21:211-243, 1996.
6. F. Baader and W. Snyder. Unification theory. In *Handbook of Automated Reasoning*, volume I, pp. 445–532. Elsevier Science, 2001.
7. L. Bachmair and N. Dershowitz. Completion for rewriting modulo a congruence. *Theoretical Computer Science*, 67:173–201, 1989.
8. D. Basin, S. Mödersheim, and L. Viganò. Constraint Differentiation: A New Reduction Technique for Constraint-Based Analysis of Security Protocols. In *Proceedings of CCS'03*, pp. 335–344. ACM Press, 2003.
9. D. Basin, S. Mödersheim, and L. Viganò. Algebraic Intruder Deductions (Extended Version). Technical Report 485, Dep. of Computer Science, ETH Zurich, 2005. Available at `www.infsec.ethz.ch`.
10. D. Basin, S. Mödersheim, and L. Viganò. OFMC: A symbolic model checker for security protocols. *International Journal of Information Security*, 4(3):181–208, 2005.
11. M. Boreale and M. G. Buscemi. A framework for the analysis of security protocols. In *Proceedings of CONCUR 2002*, LNCS 2421, pp. 483–498. Springer, 2002.
12. Y. Chevalier, R. Küsters, M. Rusinowitch, and M. Turuani. An NP Decision Procedure for Protocol Insecurity with XOR. In *Proceedings of LICS'03*, pp. 261–270. IEEE Computer Society Press, 2003.
13. Y. Chevalier, R. Küsters, M. Rusinowitch, and M. Turuani. Deciding the Security of Protocols with Diffie-Hellman Exponentiation and Products in Exponents. In *Proceedings of FST TCS'03*, LNCS 2914, pp. 124–135. Springer, 2003.
14. Y. Chevalier and M. Rusinowitch. Combining Intruder Theories. In *Proceedings of ICALP 2005*, LNCS 3580, pp. 639–651, 2005.

15. H. Comon and V. Shmatikov. Is It Possible to Decide Whether a Cryptographic Protocol Is Secure Or Not? *Journal of Telecommunications and Information Technology*, 4:5–15, 2002.
16. H. Comon-Lundh and S. Delaune. The finite variant property: How to get rid of some algebraic properties. In *Proceedings of RTA'05*, LNCS 3467, pp. 294–307. Springer, 2005.
17. V. Cortier, S. Delaune, and P. Lafourcade. A survey of algebraic properties used in cryptographic protocols. *Journal of Computer Security*, to appear.
18. S. Delaune and F. Jacquemard. A decision procedure for the verification of security protocols with explicit destructors. In *Proceedings of CCS'04*, pp. 278–287. ACM Press, 2004.
19. D. Dolev and A. Yao. On the Security of Public-Key Protocols. *IEEE Transactions on Information Theory*, 2(29), 1983.
20. N. Durgin, P. D. Lincoln, J. C. Mitchell, and A. Scedrov. Undecidability of Bounded Security Protocols. In *Proceedings of the FLOC'99 Workshop on Formal Methods and Security Protocols (FMSP'99)*, 1999.
21. IETF: The Internet Engineering Task Force. `http://www.ietf.org`.
22. J.-P. Jouannaud and H. Kirchner. Completion of a set of rules modulo a set of equations. *SIAM Journal of Computing*, 15(4):1155–1194, 1986.
23. D. Kapur, P. Narendran, and L. Wang. An E-unification algorithm for analyzing protocols that use modular exponentiation. In *Proceedings of RTA'03*, LNCS 2706, pp. 165–179. Springer, 2003.
24. P. Lafourcade, D. Lugiez, and R. Treinen. Intruder deduction for AC-like equational theories with homomorphisms. In *Proceedings of RTA'05*, LNCS 3467, pp. 308–322. Springer, 2005.
25. G. Lowe. Casper: a Compiler for the Analysis of Security Protocols. *Journal of Computer Security*, 6(1):53–84, 1998.
26. J. Meseguer and P. Thati. Symbolic reachability analysis using narrowing and its application to verification of cryptographic protocols. *Journal of Higher-Order and Symbolic Computation*, to appear.
27. J. K. Millen and V. Shmatikov. Symbolic protocol analysis with products and Diffie-Hellman exponentiation. In *Proceedings of CSFW'03*, pp. 47–61. IEEE Computer Society Press, 2003.
28. J. Siekmann and P. Szabó. The undecidability of the D_A unification problem. *Journal of Symbolic Computation*, 54(2):402–414, 1989.