Impossibility Results for Secret Establishment

Benedikt Schmidt, Patrick Schaller, David Basin
ETH Zurich, Switzerland
Email: {benedikt.schmidt,patrick.schaller,david.basin}@inf.ethz.ch

***Abstract*—Security protocol design is a creative discipline where the solution space depends on the problem to be solved and the cryptographic operators available. In this paper, we examine the general question of when two agents can create a shared secret. Namely, given an equational theory describing the cryptographic operators available, is there a protocol that allows the agents to establish a shared secret?**

We examine this question in several settings. First, we provide necessary and sufficient conditions for secret establishment using subterm convergent theories. This directly yields a decision procedure for this problem. As a consequence, we obtain impossibility results for symmetric encryption and signature schemes. Second, we use algebraic methods to prove impossibility results for two important theories that are not subterm convergent: XOR and abelian groups. Finally, we develop a general combination result that enables modular impossibility proofs. For example, the results for symmetric encryption and XOR can be combined to obtain impossibility for the joint theory.

I. Introduction

Consider a pair (or more generally a group) of honest agents who have no shared secret, but who can communicate over a public channel in the presence of a passive adversary. Furthermore, assume that each agent can generate unguessable nonces, has access to public information, and may use different cryptographic operators. Is it possible for these agents to establish a shared secret?

There are of course many ways to answers this question positively. For example, if the cryptographic operators include a public-key cryptosystem, an agent may simply send his public key over the public channel. Any other agent could then encrypt a secret with the public key that can be decrypted only by the agent holding the corresponding private key. Similarly, if a multiplicative group is given for which the so called Diffie-Hellman problem is hard, agents can use Diffie-Hellman key exchange to establish a shared secret. There are also negative answers if the set of cryptographic operators is sufficiently restricted. In particular, there is a folk theorem that no protocol exists if only symmetric encryption can be used. However, to the best of our knowledge, no formal proof of this folk theorem has previously been given.

Establishing impossibility results and developing related proof methods are of fundamental theoretical importance as they explain what cannot be achieved using cryptographic operators, specified equationally. Practically, impossiblity results delineate the solution space in protocol design and enable a more systematic approach to protocol development by guiding the choice of cryptographic operators. This is especially relevant in resource constrained scenarios, like with smartcards or sensor networks, where operations like public-key cryptography are sometimes considered too expensive and should be avoided, where possible.

Contribution: In this paper, we present a formal framework to prove impossibility results for secret establishment for arbitrary cryptographic operators in the symbolic setting. We model messages and operations by equational theories and communication by traces of events as is standard in symbolic protocol analysis. The initial question of whether it is possible for two agents to establish a shared secret therefore reduces to the question: Is there a valid trace where two agents end up sharing a message that cannot be derived from the exchanged messages?

We start by applying our framework to the equational theory that models symmetric encryption and prove the folk theorem that secret establishment is impossible in this setting. It turns out that symmetric encryption is actually an instance of the more general case where the properties of the involved operators can be described by a subterm convergent theory. For this general class of equational theories, we present a necessary and sufficient condition for the possibility of secret establishment based on labelings of the equations. This directly yields a decision procedure that either returns a labeling that corresponds to a trace where two agents establish a shared secret or returns "impossible" if there is none. For an equational theory that models a public-key cryptosystem, the labeling returned corresponds to the message exchange previously mentioned where the secret is encrypted using the public key previously exchanged. Afterwards, we consider two important theories that implement XOR and abelian groups. These are not subterm convergent since both define associative and commutative operators. In both cases, we show that secret establishment is impossible using algebraic methods that exploit the isomorphisms between the term algebra and a standard algebraic structure: a vector space over $\mathbb{F}_2$ (for XOR) and a $\mathbb{Z}$–module (for abelian groups).

The above results are for theories in isolation. We also investigate the problem of combining theories and prove a combination result for disjoint theories: Secret establishment in the combination of the theories is possible if and only if it is already possible for one of the individual theories alone.

This allows for modular proofs where separate results are combined. For example, we prove this way that secret establishment is impossible for symmetric encryption together with XOR.

Related Work: There are few existing impossibility results for security protocols. Pereira and Quisquater [1] prove the generic insecurity for a class of authenticated group key agreement protocols. Micciancio and Panjwani [2] consider the related question of lower bounds on the communication complexity of security protocols. They establish a lower bound on the communication complexity for a class of multicast key distribution protocols.

Of course, there are many positive results for secret establishment where a protocol using given cryptographic primitives is presented. Secret establishment was one of the main goals in the development of public-key cryptography. Diffie and Hellman [3] present the first usable secret agreement protocol in the presence of a passive attacker based on multiplicative groups where the discrete logarithm problem is hard. Later, Pohlig and Hellman [4] show how the scheme can be modified to provide secret transport and Rivest, Shamir, and Adleman [5] present the first public-key cryptosystem and describe its use for secret transport. Most recent protocols are based on these primitives or variations thereof. An example of a secret establishment protocol that uses a nonstandard cryptographic operation is given by Rabi and Sherman [6] and uses an associative one-way function. But no implementation of this primitive has been proposed so far.

Several authors have investigated sufficient conditions for establishing security relations between agents based on knowledge and properties of cryptographic operators. Maurer and Schmid [7] present a channel calculus that describes how an insecure channel can be transformed into a channel providing security guarantees. The transformations rely on other channels with given properties that are used in combination with cryptographic operators. Boyd [8] presents a formal model using the language Z. His model builds on the abstract types *users* and *keys*, where communication channels are modeled as relations on the set of users and security properties of channels are predicates on the distribution of keys. Boyd then derives the same set of secure channel transformations as those presented in [7]. In terms of impossibility, both papers propose that any secure channel transformation must be based on a previously existing security relation, i.e., you cannot get security from nothing. Whereas Maurer and Schmid [7] propose this as an axiom, Boyd [8] proves that it is a property of his abstract model.

We consider protocols and protocol runs in a symbolic setting. This line of work was started by Dolev and Yao [9] and has been used successfully in many different protocol models. This modeling approach allows for efficient, fully automated protocol analysis [10, 11] and can still provide a sound abstraction of most cryptographic operators [12, 13]. A recent result here, connecting the symbolic and computational world is that of Baudet et al. [14]. They introduce the notion of soundness for computational algebras, a combination of a symbolic model given by an equational theory—the same approach we use—and a concrete computational implementation that implements the symbolic operations.

Organization: In Section II, we review background material on rewriting and deducibility. In Section III, we present our model and its properties. In Section IV, we present impossibility results for symmetric encryption and subterm convergent theories. Afterwards, we present impossibility results for XOR and abelian groups in Section V and a combination result for disjoint equational theories in Section VI.

II. Background

We start by recalling some standard notions from rewriting. More details can be found in [15] or [16].

A. Basic Definitions

A signature Σ is a set of function symbols, where each function symbol is associated with an arity. We denote the subset of n-ary function symbols by Σ^n. We assume a countably infinite set $\mathcal{V}$ of variables and a countably infinite set $\mathcal{N}$ of names that model free constants as is done in the applied pi calculus [17]. For a signature Σ and a set $\mathcal{M} \subseteq \mathcal{V} \cup \mathcal{N}$, the set $T(\Sigma, \mathcal{M})$ denotes the set of terms constructed over $\Sigma \cup \mathcal{M}$.

Positions in terms are defined as usual by integer sequences, where the root position is the empty sequence ϵ. For every term t and every position p, we denote by $t|_p$ the subterm of t at position p. If a position p_1 is a prefix of a position p_2, we say that p_1 is *above* p_2 and p_2 is *below* p_1. If p_1 is neither below nor above p_2, we say that p_1 and p_2 are *incomparable*. Furthermore, we use $t[s]_p$ to denote the term t where the subterm $t|_p$ has been replaced by s.

A context ζ is a term with holes. Holes are distinct variables x_i that occur exactly once in ζ. We use the notation $\zeta[x_1, \ldots, x_n]$ to indicate that ζ has the holes $x_1, \ldots, x_n$ and $\zeta[t_1, \ldots, t_n]$ to denote the term where the holes have been replaced by terms t_i.

A substitution σ is a function from $\mathcal{V}$ to $T(\Sigma, \mathcal{M})$ that corresponds to the identity function on all but the finite set $dom(\sigma) \subseteq \mathcal{V}$ of variables. We identify σ with its usual extension to an endomorphism on $T(\Sigma, \mathcal{M})$ and use the notation $t\sigma$ for the application of σ to the term t. Furthermore, we denote by $\{t_1/x_1, \ldots t_k/x_k\}$ the substitution σ with domain $\{x_1, \ldots, x_k\}$ where $x_i\sigma = t_i$.

An equation over a signature Σ is an unordered pair $\{s, t\}$ of terms $s, t \in T(\Sigma, \mathcal{V})$ denoted $s \simeq t$. For a set of equations E over a signature Σ, we define the equational theory $Eq(\Sigma, E)$ as the smallest congruence containing all instances of the equations of E. We say $\mathcal{H} = (\Sigma, E)$ is

an equational representation of $Eq(\Sigma', E')$ if $Eq(\Sigma', E') = Eq(\Sigma, E)$ and use the equational representation $\mathcal{H}$ and E interchangeably, if Σ is clear from the context. We then write $s =_E t$ for $(s,t) \in Eq(\Sigma, E)$. An equational theory is *consistent* if $n \neq_E n'$ for distinct names n and n'. We always assume that the equational theories under consideration are consistent since inconsistency implies that all names are equal and secret establishment is therefore impossible.

A rewrite rule is an ordered pair of terms $(l, r) \in T(\Sigma, \mathcal{V}) \times T(\Sigma, \mathcal{V})$ denoted $l \to r$. A rewrite system R is a set of rewrite rules. R defines a rewrite relation $\to_R$ with $t \to_R t'$ if there is a position p in t, a rule $l \to r$ in R, and a substitution σ such that $t|_p = l\sigma$ and $t' = t[r\sigma]_p$. We say that a rewrite system R is *convergent* if the corresponding rewrite relation $\to_R$ is confluent and terminating. A rewriting system R is *subterm convergent* if it is convergent and for every rule $l \to r$ in R, r is either a constant or a proper subterm of l. We call an equational theory E subterm convergent if the equations can be oriented to obtain a subterm convergent rewrite system.

The set $St(t)$ of syntactic subterms of a term t is defined in the usual way. A *proper* subterm of t is a subterm different from t. For a term t, we define $vars(t) = St(t) \cap \mathcal{V}$ and $names(t) = St(t) \cap \mathcal{N}$. A term t is *ground* if $vars(t) = \emptyset$.

A replacement ρ is a function from a finite set of terms to $T(\Sigma, \mathcal{M})$ such that $dom(\rho) \cap St(range(\rho)) = \emptyset$. For an arbitrary term t, we define t^ρ as the unique term where all occurrences of subterms $s = t|_p$ of t, with $s \in dom(\rho)$ such that there is no position p' above p with $t|_{p'} \in dom(\rho)$, are replaced by $\rho(s)$. Note that an equational theory is always stable under replacements of names by terms, i.e., if $t =_E s$ and ρ is a replacement with $dom(\rho) \subseteq \mathcal{N}$, then $t^\rho =_E s^\rho$.

We use the notation $[x_1, \ldots, x_n]$ to denote finite lists and write $L \cdot x$ to denote the list L with the element x appended to the end. We use $\cup$ to denote the disjoint union of two sets. We also use $\sigma_{[L]}$ to denote the substitution $\{M_1/x_1, \ldots, M_k/x_k\}$ that corresponds to the list $L = [M_1, \ldots, M_k]$. We implicitly lift functions on terms to lists and sets of terms in the usual way.

B. Messages, Frames, and Deducibility

As is common practice in symbolic protocol analysis, we abstract away from concrete implementations where messages are encoded and manipulated as bit-strings. We define the set of messages as $\mathcal{M}_\Sigma = T(\Sigma, \mathcal{N})$, where we use the set of names $\mathcal{N}$ to model free constants that are not included in Σ^0. The function symbols in Σ model cryptographic operations on abstract messages, where the operations' semantics is given by a set of equations E that defines the equational theory $Eq(\Sigma, E)$.

We next define a notion of deducibility on a set of messages. Consider, for example, the case where an adversary has overheard communication between honest agents and has seen the messages $M_1, \ldots, M_l$. Given these messages, we are interested in the set of messages the adversary can deduce by applying cryptographic operations. In order to model that the adversary may use the set of observed messages to compose new messages, we define the notion of a *frame*, as it has been introduced in the applied pi calculus [17]. A list of messages $[M_1, \ldots, M_l]$ is organized into a frame $\phi = \nu\, \tilde{n}\,.\,\sigma$ as follows.

- $\tilde{n}$ is a finite set of *restricted* names. Intuitively this is a set of *fresh* names and models the nonces created by the honest agents. Although it might be possible for the adversary to deduce restricted names, the adversary cannot construct them directly.
- σ is the substitution $\{M_1/x_1, \ldots, M_l/x_l\}$. This allows the adversary to use the observed messages when constructing new ones.

Based on the notion of a frame, we define the deducibility relation $\vdash_E$ for an equational theory E. The corresponding rules are presented in Figure 1 and model that the adversary can take any of the following actions.

- CONST: The adversary can deduce any name, except the restricted ones in the set $\tilde{n}$.
- KNOW: The adversary can deduce all messages in the range of σ.
- APPLY: The adversary can apply functions in Σ to deducible messages.
- EQUAL: The adversary can deduce messages that are equivalent to deducible messages modulo the equational theory E.

Note that $\nu\, \tilde{n}\,.\,\sigma \vdash_E M$ if and only if there is a term $C \in T(\Sigma, dom(\sigma) \cup \mathcal{N} \setminus \tilde{n})$ such that $C\sigma =_E M$. We call such a term C a *recipe* for M.

We shall abuse notation and write $H \vdash_E s$ to denote $\nu\, names(H).\, \sigma_{[H]} \vdash_E s$ for a list of messages H.

C. Ordered Completion

We use ordered completion [16] to define the normalization of ground terms with respect to an arbitrary equational theory E. This technique has been used in similar contexts to prove the correctness of combination results for unification [18] and deducibility [19, 20].

$$\textsc{Const}\ \frac{N \in \mathcal{N} \setminus \tilde{n}}{\nu\, \tilde{n}\,.\,\sigma \vdash_E N} \qquad \textsc{Know}\ \frac{x \in dom(\sigma)}{\nu\, \tilde{n}\,.\,\sigma \vdash_E x\sigma}$$

$$\textsc{Apply}\ \frac{\phi \vdash_E M_1 \quad \ldots \quad \phi \vdash_E M_k \quad f \in \Sigma^k}{\phi \vdash_E f(M_1, \ldots, M_k)}$$

$$\textsc{Equal}\ \frac{\phi \vdash_E M \qquad M =_E N}{\phi \vdash_E N}$$

Figure 1: Inductively defined relation $\vdash_E$

Let $\succ$ be a total simplification order on ground terms, i.e., for ground terms N_1 and N_2 and a nonempty context M, we have that (i) $N_1 \succ N_2$ or $N_2 \succ N_1$, (ii) $M[N_1] \succ N_1$, and (iii) $N_1 \succ N_2$ implies $M[N_1] \succ M[N_2]$. Additionally, we assume for all $n \in \mathcal{N}$, $c \in \Sigma^0$, and $t \in \mathcal{M}_\Sigma \setminus (\mathcal{N} \cup \Sigma^0)$ that $c \succ n$ and $t \succ c$. We then use n_{min} to denote the minimum for $\succ$, which is a name. The lexicographic path ordering constructed from a total ordering on $\mathcal{N} \cup \Sigma$, where names are smaller than constants from Σ and constants are smaller than nonconstant function symbols, always has these properties (see [16]).

For a given equational theory E and a total simplification ordering on ground terms $\succ$, we define the ordered rewrite relation $\rightarrow_{(\succ,E)}$ as follows: $t \rightarrow_{(\succ,E)} t'$ if there is a position p of t, an equation $l \simeq r$ in E, and a substitution σ such that $t|_p = l\sigma$, $t' = t[r\sigma]_p$, and $t \succ t'$.

We use *ordered completion* for a given equational theory E to obtain a (possibly infinite) set of equations $\mathcal{O}_E$ such that $=_{\mathcal{O}_E}$ equals $=_E$ and $\rightarrow_{(\succ,\mathcal{O}_E)}$ is convergent on ground terms. We define $t{\downarrow_E}$ for a ground term t as t's normal form with respect to $\rightarrow_{(\succ,\mathcal{O}_E)}$. We write $t{\downarrow}$ if E is clear from the context.

III. Traces and Deducibility

We first define the set of derivation traces that models all possible agent behaviors, such as constructing messages and exchanging messages over a public channel. Afterwards, we define the notion of shared secrets for such a trace and relate derivation traces and protocols.

A. Derivation Traces

In the following, let $\mathcal{A}$ be the set of agents and let $Eq(\Sigma, E)$ describes the cryptographic operators under consideration and their relevant properties.

An event either denotes that an agent sends a message or learns a message. Events are therefore associated with the corresponding agent's identity. A learn event is additionally tagged with the rule R that describes how the agent learned the message.

$$\mathsf{Event} = \mathsf{Send}(\mathcal{A}, \mathcal{M}_\Sigma) \mid \mathsf{Learn}_R(\mathcal{A}, \mathcal{M}_\Sigma)$$

The steps taken to construct and communicate messages are modeled by traces, where a trace is a list of events. The set of valid traces TR_E is inductively defined by the rules in Figure 2. Note that we abuse notation and write $\mathsf{Learn}(A, M)$ to match $\mathsf{Learn}_R(A, M)$ events for an arbitrary R. The rules model the following actions.

- SEND: An agent A sends a previously learned message M on the public channel.
- RECV: An agent A receives a message M that has been previously sent by B.
- FRESH: An agent creates an unguessable name. Note that the minimal name n_{min} under the ordering $\succ$ used for ordered completion is distinguished in that it cannot be used here or in the PUBLIC rule.
- PUBLIC: An agent A uses a public value, where $bound(tr) = \{N \mid \exists A.\, \mathsf{Learn}_{\mathsf{Fresh}}(A, N) \in tr\}$ denotes the bound names in the trace.
- DERIVE: An agent A applies a k-ary function f to the previously learned messages $M_1, \ldots, M_k$. Here we use the fact that every message has a unique normal form modulo E with respect to $\rightarrow_{(\succ,\mathcal{O}_E)}$.

$$\textsc{Empty}\ \frac{}{[] \in \mathsf{TR}_E}$$

$$\textsc{Send}\ \frac{tr \in \mathsf{TR}_E \qquad \mathsf{Learn}(A, M) \in tr}{tr \cdot \mathsf{Send}(A, M) \in \mathsf{TR}_E}$$

$$\textsc{Recv}\ \frac{tr \in \mathsf{TR}_E \qquad \mathsf{Send}(B, M) \in tr}{tr \cdot \mathsf{Learn}_{\mathsf{Recv}}(A, M) \in \mathsf{TR}_E}$$

$$\textsc{Fresh}\ \frac{tr \in \mathsf{TR}_E \qquad N \in \mathcal{N} \setminus (names(tr) \cup \{\mathrm{n}_{min}\})}{tr \cdot \mathsf{Learn}_{\mathsf{Fresh}}(A, N) \in \mathsf{TR}_E}$$

$$\textsc{Public}\ \frac{tr \in \mathsf{TR}_E \qquad P \in \mathcal{N} \setminus (bound(tr) \cup \{\mathrm{n}_{min}\})}{tr \cdot \mathsf{Learn}_{\mathsf{Public}}(A, P) \in \mathsf{TR}_E}$$

$$\textsc{Derive}\ \frac{\begin{array}{c} tr \in \mathsf{TR}_E \qquad f \in \Sigma^k \\ \mathsf{Learn}(A, M_1) \in tr\ \ldots\ \mathsf{Learn}(A, M_k) \in tr \\ f(M_1, \ldots, M_k){\downarrow_E} = M \end{array}}{tr \cdot \mathsf{Learn}_{\mathsf{Derive}(f(M_1,\ldots,M_k))}(A, M) \in \mathsf{TR}_E}$$

Figure 2: Inductively defined set TR_E

B. Shared Secrets and Deducibility

Clearly, we must restrict the initial knowledge of agents to prove impossibility results for secret establishment. Some restrictions are necessary to prevent initial knowledge distributions that allow the creation of a shared secret, but require the previous existence of secret channels, e.g., shared secret keys distributed by a third party. We enforce this restriction by requiring that every derivation starts with the empty trace, which corresponds to the empty initial knowledge for the involved agents. However, in our model, we do not distinguish between the setup phase and the execution phase. Therefore, any prefix of a derivation trace can be interpreted as a setup phase where agents establish private and public knowledge in the presence of the attacker. This covers precisely the initial knowledge distributions that do not require secret channels and include all messages involved in establishing the knowledge.

We are interested in impossibility results. An impossibility result for a class of adversaries implies impossibility for any larger class. As a consequence, we reason about weak

adversaries, namely passive adversaries who are restricted to eavesdropping communication on the public channel. This models an adversary who is not involved in the derivation process. For such an adversary, only the frame ϕ_{tr} that corresponds to a trace tr is important since it defines the messages m with $\phi_{tr} \vdash_E m$ that he can deduce. Let $send(tr)$ denote the list of messages $[M_1, \ldots, M_k]$ that have been sent in the trace tr. Then define the frame ϕ_{tr} as $\nu\, bound(tr).\, \sigma_{[send(tr)]}$.

Note that we combine the deduction rules for honest agents with the rules for exchanging messages in the definition of TR_E. However, the deduction rules formalize deduction capabilities that are equivalent to the standard $\vdash_E$-relation. The only difference is that the messages in derivation traces are always in normal form and we therefore do not need an EQUAL rule.

We now define what it means to share a secret.

Definition 1. *A term S is a* shared secret between A and B in a trace tr, *if $A \neq B$,* $\mathsf{Learn}(A, S) \in tr$, $\mathsf{Learn}(B, S) \in tr$, *and $\phi_{tr} \nvdash_E S$.*

To simplify subsequent proofs, we first show that we can restrict our attention to a single pair of honest agents.

Lemma 1. *There is a run that involves an arbitrary number of agents to establish a shared secret between the distinct agents A and B if and only if there is a run where only A and B participate.*

Proof: The right to left direction is trivial. The converse can be proved by translating the trace with an arbitrary number of agents to a trace with only the agents A and B. This translation maps every event executed by an agent $C \notin \{A, B\}$ to the corresponding event executed by A. This translation results in a valid trace since the premises for the rules extending a translated trace with a translated event remain valid. ∎

In the following, we therefore fix $\mathcal{A} = \{\mathbb{A}, \mathbb{B}\}$, where $\mathbb{A}$ and $\mathbb{B}$ are distinct agents. To define the notion of minimal traces, we require the following definition.

Definition 2. *We say that an agent A* constructs *a message M in tr, if A first learns M in a* Fresh, Public, *or* $\mathsf{Derive}_{M'}$ *event. We say A* freely constructs *M in tr if no reduction of the message occurs in the event where A first learns M, i.e, the event is of the form* Fresh, Public, *or* Derive_M.

We now introduce the notion of minimal trace.

Definition 3. *A trace $tr \in \mathsf{TR}_E$ is a* minimal trace *if and only if each of the following conditions hold.*

(1) *There is at most one shared secret S. This shared secret is learned by $\mathbb{B}$ in the last event, which is a* $\mathsf{Learn}_{\mathsf{Derive}}$ *event.*
(2) *There are no* $\mathsf{Learn}_{\mathsf{Public}}$ *events in the trace, i.e., all names are bound.*
(3) *There are no duplicate events and every message but S is only constructed once.*
(4) *There is at most one free construction event for each message. This also holds for S.*

We denote the subset of minimal traces by MTR_E.

Lemma 2. *If there is a trace $tr \in \mathsf{TR}_E$ that establishes a shared secret, then there is also a minimal trace $\widehat{tr} \in \mathsf{MTR}_E$ that establishes a shared secret.*

Proof: We first prove by induction that a trace $tr \in TR_E$ without a shared secret can be transformed into a minimal trace without changing $names(tr)$, the derivable messages for the adversary, and the messages known by $\mathbb{A}$ and $\mathbb{B}$. An EMPTY trace is already minimal. For the induction steps, we assume that we can transform the trace tr into a minimal trace $\widehat{tr}$ with the given properties. If tr is extended by the SEND (respectively the RECV) rule, then we extend $\widehat{tr}$ with the corresponding Send (respectively Recv) event, provided it is not a duplicate event. If tr is extended by the FRESH rule, we extend $\widehat{tr}$ with the corresponding Fresh event, which is allowed since $names(tr) = names(\widehat{tr})$. If tr is extended with an event $\mathsf{Learn}_{\mathsf{Public}}(A, N)$ by the PUBLIC rule, we extend $\widehat{tr}$ with the events $\mathsf{Learn}_{\mathsf{Fresh}}(A, N)$ and $\mathsf{Send}(A, N)$. If tr is extended by an event $ev = \mathsf{Learn}_{\mathsf{Derive}(f(M_1,\ldots,M_k))}(A, M)$ and there is no earlier event $\mathsf{Learn}(C, M)$ for some C, then we add ev to $\widehat{tr}$. If there is already such an event with $C = A$, then we do not extend $\widehat{tr}$. If $C \neq A$ for all such events, we extend $\widehat{tr}$ with the events $\mathsf{Send}(C, M)$ and $\mathsf{Learn}_{\mathsf{Recv}}(A, M)$. Since M is not a secret and shared knowledge between A and C, it must be already deducible by the adversary from $send(tr \cdot ev)$.

To transform a trace $tr \in TR_E$ with shared secrets into a minimal trace that establishes exactly one shared secret, we first take the shortest prefix $ptr \cdot ev$ of tr that establishes a secret. Since an agent cannot establish a shared secret with a Send, Recv, Public, or Fresh event, the last event ev must be a Derive event where, without loss of generality, $\mathbb{B}$ learns some secret S. We can transform ptr into a minimal trace $\widehat{ptr}$ using the previously described transformation since it does not contain a shared secret. Then $\widehat{ptr} \cdot ev \in \mathsf{MTR}_E$ since all premises for adding ev still hold for $\widehat{ptr}$ and $\widehat{ptr} \cdot ev$ has properties (1)–(4). Properties (1)–(3) are obvious. To see that (4) holds, assume that S is freely constructed by both $\mathbb{B}$ and $\mathbb{A}$ in tr. Then $S = f(M_1, \ldots, M_k)$ for some function symbol f and some messages M_i and there are events $\mathsf{Learn}_{\mathsf{Derive}(S)}(C, S)$ for $C = \mathbb{A}$ and $C = \mathbb{B}$. But then all M_i are shared knowledge since S is the only secret and the adversary can therefore deduce the M_i and $S = f(M_1, \ldots, M_k)$. ∎

Given Lemma 2, we can henceforth restrict ourselves to minimal traces in our impossibility proofs.

C. Relating Protocols and Derivation Traces

Above we have reduced the question whether it is possible to establish a shared secret using an equational theory E to the question of whether there is a derivation trace $tr \in \mathsf{TR}_E$ that establishes a shared secret. This is closely related to the question of whether there is a *protocol* that establishes a shared secret in a given symbolic protocol model.

The existence of a protocol implies that there is a successful protocol execution where the involved agents establish a shared secret. For all reasonable protocol models that use the same notion of deducibility as we do, a successful protocol execution directly yields a corresponding derivation trace that establishes a shared secret. An impossibility result for some equational theory E in our model thus directly implies the corresponding impossibility result for protocols in such a symbolic model. A concrete example of a protocol model where this relationship holds is the applied pi calculus with equational theory E where agents are only allowed to communicate over public channels.

IV. Impossibility Results for Symmetric Encryption and Subterm Convergent Theories

In this section, we prove the folk theorem that it is impossible to establish a shared secret using only symmetric encryption and public channels. We then present a necessary and sufficient condition for impossibility for the more general case of subterm convergent theories. This condition can be used to automatically decide wether it is possible to create a shared secret for a given subterm convergent theory. We have implemented a decision procedure that checks this condition and illustrate its application to the theory of symmetric encryption. Afterwards, we show how our procedure finds a derivation trace that establishes a shared secret for the theory of public-key encryption.

A. Symmetric Encryption

We use the equational theory $Eq(\Sigma_{Sym}, E_{Sym})$ to model symmetric encryption, pairing, a hash function, decryption, and projections on pairs.

$$\begin{aligned}\Sigma_{Sym} =& \{enc, \langle\rangle, h, dec, \pi_1, \pi_2\}\\ E_{Sym} =& \{dec(enc(m,k),k) \simeq m,\\ & \quad \pi_1(\langle x, y\rangle) \simeq x,\\ & \quad \pi_2(\langle x, y\rangle) \simeq y\}\end{aligned}$$

Since the rewriting system R_{Sym} obtained from E_{Sym} by orienting the equations from left to right is subterm convergent, we directly use $\rightarrow_{R_{Sym}}$ to normalize terms in the Derive rule and do not require ordered completion. The following lemma holds for all subterm convergent theories and will be used in Section IV-B as well.

Lemma 3. *Let $Eq(\Sigma, E)$ be a subterm convergent theory and $tr \in \mathsf{TR}_E$ a valid trace. For every event $\mathsf{Learn}(A, M) \in tr$ and $P \in St(M) \setminus \Sigma^0$, P has been freely constructed by some agent C. More precisely, if P is a name, then $\mathsf{Learn}_{\mathsf{Fresh}}(C, P) \in tr$ or $\mathsf{Learn}_{\mathsf{Public}}(C, P) \in tr$. Otherwise there exist $M_1, \ldots, M_k$ and an $f \in \Sigma^k$, such that $P = f(M_1, \ldots, M_k)$ and $\mathsf{Learn}_{\mathsf{Derive}(P)}(C, P) \in tr$.*

Proof: The proof is straightforward using rule induction on TR_E. The rules Empty and Send are trivial since they do not add any Learn events. The Fresh and Public rules are also trivial since they only add (atomic) names. The Recv rule adds an event $\mathsf{Learn}_{\mathsf{Recv}}(A, M)$, but a corresponding $\mathsf{Learn}(C, M)$ for some C must be already in the trace since someone must have sent M. Therefore the statement holds by the induction hypothesis. The Derive rule adds an event $\mathsf{Learn}_{\mathsf{Derive}(g(N_1,\ldots,N_l))}(A, M)$, where $M = g(N_1, \ldots, N_l)\downarrow$. We must show that the lemma's statement holds for all subterms of M that are not constants. Since the equational theory is subterm convergent, we have either $M = c$, $M = g(N_1, \ldots, N_l)$, or that M is a proper subterm of $g(N_1, \ldots, N_l)$. In the first case, the statement follows trivially, since $St(c) \setminus \Sigma^0 = \emptyset$. In the second case, a subterm of M is either M itself and the statement trivially holds or a subterm of some N_i and the statement holds by the induction hypothesis since there is an event $\mathsf{Learn}(A, N_i)$ in the trace. The same reasoning applies to the final case since M and all its subterms are subterms of some N_i. ■

Theorem 1. *There is no derivation trace using $Eq(\Sigma_{Sym}, E_{Sym})$ that establishes a shared secret. Namely, if $tr \in \mathsf{TR}_{E_{Sym}}$, $\mathsf{Learn}(\mathbb{A}, S) \in tr$, and $\mathsf{Learn}(\mathbb{B}, S) \in tr$, then $\phi_{tr} \vdash_{E_{Sym}} S$.*

Proof: We prove the theorem by contradiction. Assume that there is a trace in $\mathsf{TR}_{E_{Sym}}$ that establishes a shared secret. Then there is also a trace $tr \in \mathsf{MTR}_{E_{Sym}}$ that establishes a shared secret S and the last event of tr is of the form $ev = \mathsf{Learn}_{\mathsf{Derive}(f(M_1,\ldots,M_k))}(\mathbb{B}, S)$ for $S = f(M_1, \ldots, M_k)\downarrow$. Thus $\mathsf{Learn}(\mathbb{A}, S) \in tr$ and we show that $\phi_{tr} \vdash_{E_{Sym}} S$ to obtain a contradiction. We distinguish two cases.

1. If $S = f(M_1, \ldots, M_k)$, then Lemma 3 can be applied to $\mathsf{Learn}(\mathbb{A}, S)$. Thus there are two free construction events $\mathsf{Learn}_{\mathsf{Derive}(S)}(C, S)$ in tr for $C = \mathbb{A}$ and $C = \mathbb{B}$, which contradicts minimality of tr.
2. If $f(M_1, \ldots, M_k)$ is not normalized, then we must consider two more cases.
   - (2a) Assume that $f(M_1, \ldots, M_k) = \pi_i(\langle A_1, A_2\rangle)$ for some A_i. Then $S = A_i$ and $\langle A_1, A_2\rangle$ has been freely constructed by $\mathbb{A}$ by Lemma 3. Since the pair is known to both $\mathbb{A}$ and $\mathbb{B}$ in tr and S is the only secret, the pair is deducible and the intruder can thus deduce S.
   - (2b) Assume that there are P and K such that $f(M_1, \ldots, M_k) = dec(enc(P, K), K)$. Then $S = P$ and $enc(P, K)$ has been freely con-

structed by $\mathbb{A}$ by Lemma 3. Then K and $enc(P, K)$ are known to both agents in tr and are therefore deducible. It follows that the adversary can deduce S.

■

Note that the only parts in the proof that are specific to E_{Sym} and do not hold for all subterm convergent theories are the cases (2a) and (2b) for the different rules in E_{Sym}. Here, a rule $l \to r$ is applied to the result M of the function application in the DERIVE rule and we prove impossibility by showing that there is no way for $\mathbb{A}$ and $\mathbb{B}$ to build an instance of l such that the corresponding instance of r is not deducible by the adversary. We will show in the next section that this proof method works with arbitrary subterm convergent theories.

B. Subterm Convergent Theories

We now generalize the previous proof method to a necessary and sufficient condition for deciding the possibility of secret establishment for any subterm convergent theory $Eq(\Sigma, E)$. The main idea is that if there is a derivation trace where a secret is established, then there is also a minimal trace with exactly one reduction step that establishes a secret. We can decide if such a minimal trace exists by considering all equations in E individually and enumerating all possible ways to jointly construct a reducible term. Finally, we check if this construction leads to a shared secret.

In the rest of this section, we assume that E is a subterm convergent equational theory. Note that we can assume without loss of generality that the right-hand sides of the rules in the corresponding rewrite system are normalized. Since we want to associate these rules over $T(\Sigma, \mathcal{V})$ with traces that contain messages, we define a substitution σ_{gnd} that converts between $T(\Sigma, \mathcal{V})$ and $\mathcal{M}_\Sigma$. The substitution σ_{gnd} is defined by some fixed bijection from $\mathcal{V}$ to $\mathcal{N}$. To prove our main result of this section, we require the following lemma about deducibility in subterm convergent theories.

Lemma 4. *If H is a list of terms in $T(\Sigma, \mathcal{V})$, $s \in T(\Sigma, \mathcal{V})$, and $H\,\sigma_{gnd} \vdash_E s\,\sigma_{gnd}$, then $H\sigma \vdash_E s\sigma$ for all substitutions σ such that $H\sigma$ and $s\sigma$ are ground.*

Proof: If $H\,\sigma_{gnd} \vdash_E s\,\sigma_{gnd}$, then there is a recipe C such that $names(C) \cap names(H\,\sigma_{gnd} \cup H\sigma) = \emptyset$ and $C\sigma_{[H\,\sigma_{gnd}]} =_E s\,\sigma_{gnd}$. Thus $C\sigma_{[H\sigma]} = C\sigma_{[H]}\sigma =_E s\sigma$ since E is stable under replacement of names with terms and therefore $H\sigma \vdash_E s\sigma$. ■

We also require the notion of reduction events.

Definition 4. *A* reduction event *is an event of the form $\mathsf{Learn}_{\mathsf{Derive}(f(M_1,\ldots,M_k))}(C, M)$ where $f(M_1, \ldots, M_k)$ is not in normal form, i.e., M is a subterm of some M_i or a constant.*

We now show that we can restrict ourselves to a subset of the minimal traces in the case of subterm convergent theories.

Lemma 5. *If there is a trace $tr \in \mathsf{TR}_E$ that establishes a shared secret between $\mathbb{A}$ and $\mathbb{B}$, then there is also a minimal trace $\widehat{tr} \in \mathsf{MTR}_E$ that establishes a shared secret S between $\mathbb{A}$ and $\mathbb{B}$ such that S is freely constructed by $\mathbb{A}$ and the last event is the only reduction event in $\widehat{tr}$.*

Proof: Assume that there is a trace that establishes a shared secret. Then there is a minimal trace $tr \in \mathsf{MTR}_E$ establishing the shared secret S, where the last event is of the form $\mathsf{Learn}_{\mathsf{Derive}(f(M_1,\ldots,M_k))}(\mathbb{B}, S)$. This event must be a reduction event. Otherwise, $S = f(M_1, \ldots, M_k)$ and Lemma 3 can be applied to $\mathsf{Learn}(\mathbb{A}, S)$. Then there must be two events $\mathsf{Learn}_{\mathsf{Derive}(S)}(C, S)$, for $C = \mathbb{A}$ and $C = \mathbb{B}$, which contradicts the minimality of tr. Note also that $\mathbb{A}$ must have freely constructed S by Lemma 3, since it is new to $\mathbb{B}$.

We now show that all other reduction events in tr can be replaced by non-reduction events. A reduction event has the form $\mathsf{Learn}_{\mathsf{Derive}(f(M_1,\ldots,M_k))}(C, M)$ such that M is a subterm of some M_i or a constant c in normal form. In the first case, the other agent must have freely constructed M. M is therefore shared knowledge and deducible and can be openly sent to C. In the second case, the reduction event can be replaced by $\mathsf{Learn}_{\mathsf{Derive}(c)}(C, c)$ or c can be sent by the other agent if it is already known. ■

Lemma 5 implies that we can consider equations in E separately, since we only have to consider derivation traces with a single reduction step to decide impossibility.

To enumerate all different ways of building a reducible term that establishes a shared secret, we introduce labelings of terms. A labeling of a term t is a function from $St(t)$ to $\{\mathbb{A}, \mathbb{B}\}$ and captures which agent has constructed which subterm. For such a labeling l_t of a term t, the $minsent$ function returns the minimal set of exchanged terms that corresponds to l_t. It captures that if $\mathbb{A}$ uses a term created by $\mathbb{B}$ or vice versa, it must have been sent.

Definition 5. *We define the* minimal set of sent terms of a term t with label l_t *as follows:*

$$minsent(t, l_t) = \begin{cases} (\bigcup_{i\in\{1,\ldots,k\}} minsent(t_i)) & \text{if } t = f(t_1, \ldots, t_k) \\ \cup \{t_i \mid i \in \{1, \ldots, k\} \wedge & \\ \qquad l_t(t) \neq l_t(t_i)\} & \\ \emptyset & \text{otherwise} \end{cases}$$

Using the notion of labeling and function $minsent$, we obtain the following condition to decide impossibility.

Theorem 2. *There is a derivation trace $tr \in \mathsf{TR}_E$ that establishes a shared secret if and only if there is an equation*

$t \simeq s$ in E where s is a proper subterm of t and a labeling l_t of t such that each of the following holds:

(1) $l_t(t) = \mathbb{B}$
(2) $l_t(c) = \mathbb{A}$, *for all* $c \in St(t) \cap \Sigma^0$
(3) $l_t(s) = \mathbb{A}$
(4) $minsent(t, l_t)\,\sigma_{gnd} \nvdash_E s\,\sigma_{gnd}$

Proof: ($\Rightarrow$): Assume there is a trace that establishes a shared secret. Then there is also a trace $tr \in \mathsf{MTR}_E$ where $\mathbb{B}$ learns the secret S in the last event, which is the only reduction event by Lemma 5. Consider the term $T = f(M_1, \ldots, M_k)$ where S is extracted using the rule $t \to s$ in the last step. Then s is a proper subterm of t and there is a unifier σ such that $t\sigma = T$ and $s\sigma = S$.

We can extract a labeling l_T from the trace by labeling T with $\mathbb{B}$, all constants in T with $\mathbb{A}$ and all proper subterms of T that are not constants with the agent who freely constructed the term. Note that every subterm of T that is not a constant must have been freely constructed by exactly one of the agents because of Lemma 3 and minimality of tr. The labeling l_T can be translated to a labeling l_t of t by defining $l_t(u) := l_T(u\sigma)$ for $u \in St(t)$. Then l_t obviously has properties (1)–(2). It has property (3) because $s\sigma = S$ and $\mathbb{A}$ freely constructs S in tr. We know that $minsent(t, l_t)\sigma \subseteq minsent(T, l_T) \subseteq send(tr)$ and $sent(tr) \nvdash_E S$. Thus we have $minsent(t, l_t)\sigma \nvdash_E s\sigma$. By Lemma 4, we obtain $minsent(t, l_t)\,\sigma_{gnd} \nvdash_E s\,\sigma_{gnd}$.

($\Leftarrow$): We first show that we can translate an arbitrary labeling l_t of a term $t \in T(\Sigma, \mathcal{V})$ to a trace $tr \in \mathsf{TR}_E$ where all names are bound, $sent(tr) = (minsent(t, l_t)\,\sigma_{gnd}){\downarrow}$, and for all $u \in St(t)$ the agent $l_t(u)$ learns $(u\,\sigma_{gnd}){\downarrow}$ in tr. We prove this by induction over the term t.

First, if $t = x$ for a variable x, then tr consists of the single event $\mathsf{Learn}_{\mathsf{Fresh}}(l_t(x), x\,\sigma_{gnd})$. Next, if $t = f(t_1, \ldots, t_k)$ for a function f of arity k and terms t_i, then there are traces tr_i for t_i with the expected properties by the induction hypothesis. Let $\widehat{tr}$ denote the concatenation of the tr_i, where duplicate events are removed, keeping only the first occurrence of an event. Then $\widehat{tr} \in \mathsf{TR}_E$ and we can extend $\widehat{tr}$ with events $\mathsf{Send}(l_t(t_i), (t_i\,\sigma_{gnd}){\downarrow})$ and $\mathsf{Learn}_{Recv}(l_t(t), (t_i\,\sigma_{gnd}){\downarrow})$ for all i where $l_t(t) \neq l_t(t_i)$. Then we add the event $\mathsf{Learn}_{Derive(F)}(l_t(t), M)$ with $F = f((t_1\,\sigma_{gnd}){\downarrow}, \ldots, (t_k\,\sigma_{gnd}){\downarrow})$ and $M = (f(t_1, \ldots, t_k)\,\sigma_{gnd}){\downarrow}$ to obtain the trace $tr \in \mathsf{TR}_E$ with the desired properties. If there is a corresponding equation $t \simeq s$ where s is a proper subterm of t in E and (1)–(4) hold for the given labeling, then this trace establishes the shared secret $s\,\sigma_{gnd}$ between $\mathbb{A}$ and $\mathbb{B}$. ∎

Based on this theorem, we define a decision procedure FIND-DERIVATION-TRACES that checks the theorem's conditions for a given theory. Our procedure takes a subterm convergent theory E as input and either returns IMPOSSIBLE if there is no labeling that allows secret establishment for a rule in E or a list of derivation traces if there are labelings of rules in E that satisfy the conditions of Theorem 2.

```
FIND-DERIVATION-TRACES(E)
1  traces = []
2  for (t ≃ s) in E
3      for l_t in LABELINGS(t,s)
4          if MINSENT(t, l_t) σ_gnd ⊬_E s σ_gnd and s ∈ St(t)
5              traces = traces · LABEL2TRACE(t, l_t, s)
6  if (traces = []) return IMPOSSIBLE
7  else return traces
```

The procedure uses four subroutines. The subroutine LABELINGS returns all labelings for a rule that satisfy conditions (1)–(3). MINSENT returns the minimal set of sent terms for a labeling. The subroutine for $\vdash_E$ implements the procedure described in [21] to check ground deducibility. Finally, LABEL2TRACE converts a labeling to a trace. Note that by Theorem 2 we can consider all rules individually. However, we must check deducibility for the whole equational theory E.

We have implemented our decision procedure in Haskell. Although the number of labelings grows exponentially in the size of the equations, the procedure returns the result immediately for the examples we considered. Typically, the rules are small and most labels are already predetermined by the conditions of Theorem 2. We can further optimize LABELINGS by taking into account that properties (3) and (4) imply $l_t(u) = \mathbb{A}$ for all $u \in St(t)$ that have s as immediate subterm. This is because s would otherwise be in $minsent(t, l_t)$ and therefore be trivially deducible.

We have applied our implementation of the decision procedure to the theory E_{Sym} from the previous section. We have thereby obtained an automated confirmation of the pen-and-paper proof of Theorem 1. We use E_{Sym} and a theory that models public-key encryption below to illustrate our decision procedure.

Example 1. *The procedure considers the three following reduction rules individually.*

(1) $\pi_1(\langle x, y\rangle) \to x$: *There are only two choices for the labeling. We can set* $C = \mathbb{A}$ *or* $C = \mathbb{B}$ *in* $\pi_1(\langle x^{\mathbb{A}}, y^C\rangle^{\mathbb{A}})^{\mathbb{B}}$. *For both labelings,* $\langle x, y\rangle$ *is in the minimal set of sent terms and* x *is therefore derivable.*
(2) $\pi_2(\langle x, y\rangle) \to y$: *Analogous to previous rule.*
(3) $dec(enc(m, k), k) \to m$: *The only possible labelings are* $dec(enc(m^{\mathbb{A}}, k^C)^{\mathbb{A}}, k^C)^{\mathbb{B}}$ *for* $C \in \{\mathbb{A}, \mathbb{B}\}$. *For both choices of* C, k *is used by both* $\mathbb{A}$ *and* $\mathbb{B}$ *and thus must have been sent. Similarly,* $enc(m, k)$ *must have been sent in both cases. Therefore* m *is deducible by the intruder.*

Thus $\text{MINSENT}(t, l_t)\,\sigma_{gnd} \vdash_E s\,\sigma_{gnd}$ *for all labelings and the procedure returns* IMPOSSIBLE.

Example 2. *Consider the theory* E_{PubKey} *that consists of the single rule* $pdec(penc(m, pk(k)), sk(k)) \to$

m. *For this theory, our procedure returns the labeling* $pdec(penc(m^{\mathbb{B}}, pk(k^{\mathbb{A}})^{\mathbb{A}})^{\mathbb{B}}, sk(k^{\mathbb{A}})^{\mathbb{A}})^{\mathbb{A}}$ *and the following derivation trace that establishes the secret* m.

$$
\begin{aligned}
&[\,\mathsf{Learn}_{\mathsf{Fresh}}(\mathbb{A}, k), \mathsf{Learn}_{\mathsf{Derive}(sk(k))}(\mathbb{A}, sk(k)),\\
&\;\mathsf{Learn}_{\mathsf{Derive}(pk(k))}(\mathbb{A}, pk(k)), \mathsf{Send}(\mathbb{A}, pk(k)),\\
&\;\mathsf{Learn}_{\mathsf{Recv}}(\mathbb{B}, pk(k)), \mathsf{Learn}_{\mathsf{Fresh}}(\mathbb{B}, m),\\
&\;\mathsf{Learn}_{\mathsf{Derive}(penc(m,pk(k)))}(\mathbb{B}, penc(m, pk(k))),\\
&\;\mathsf{Send}(\mathbb{B}, penc(m, pk(k))), \mathsf{Learn}_{\mathsf{Recv}}(\mathbb{A}, penc(m, pk(k))),\\
&\;\mathsf{Learn}_{\mathsf{Derive}(pdec(penc(m,pk(k)),sk(k)))}(\mathbb{A}, m)]
\end{aligned}
$$

Note that Theorem 2 uses the notion of secret establishment introduced in Definition 1. Therefore, the existence of a trace that establishes a shared secret according to Theorem 2 does not guarantee that secret establishment is possible in the presence of active adversaries.

We also use our implementation to obtain new impossibility results. For example, our implementation returns IMPOSSIBLE for the theory describing pairing, signatures, and symmetric cryptography and thereby proves the following theorem.

Theorem 3. *Secret establishment is impossible for the theory* $Eq(\Sigma_{Sym} \cup \Sigma_{Sig}, E_{Sym} \cup E_{Sig})$*, where the theory for signatures is defined as follows.*

$$
\begin{aligned}
\Sigma_{Sig} =&\{sign, extr, check, sk, pk\}\\
E_{Sig} =&\{extr(sign(m,k)) \simeq m,\\
&\quad check(sign(m, sk(k)), pk(k)) \simeq m\}
\end{aligned}
$$

We will later prove a combination result that allows us to consider the two theories E_{Sym} and E_{Sig} separately.

V. XOR and Abelian Groups

In this section, we prove impossibility for two important theories that are not subterm convergent. The theory Xor, which models an XOR operator, and the theory AG, which models an abelian group. We will use rather different proof techniques here, this time based on algebraic reasoning in different algebraic structures. The techniques are similar to earlier work [22, 23] on unification and deducibility for monoidal theories.

A. XOR

The XOR operation can be described by the following equational theory with equations for associativity, commutativity, a unit, and nilpotence.

$$
\begin{aligned}
\Sigma_{Xor} =&\{\oplus, 0\}\\
E_{Xor} =&\{(x \oplus y) \oplus z \simeq x \oplus (y \oplus z),\\
&\quad x \oplus y \simeq y \oplus x,\\
&\quad x \oplus 0 \simeq x,\\
&\quad x \oplus x \simeq 0\}
\end{aligned}
$$

Note that for a finite set of names N, $T(\Sigma_{Xor}, N)/E_{Xor} \simeq \mathbb{F}_2^{|N|}$. I.e., the set of messages over N is isomorphic to the $|N|$-dimensional vector space over the finite field $\mathbb{F}_2$. An isomorphism is given by the map $\mathcal{R}$ where $\mathcal{R}(n_i) = v_i$, the vector where the only nonzero entry is a 1 at position i, $\mathcal{R}(0) = (0, \ldots, 0)$, and $\mathcal{R}(t_1 \oplus t_2) = \mathcal{R}(t_1) + \mathcal{R}(t_2)$ where $+$ denotes componentwise addition modulo 2. We abuse notation and use $\mathcal{R}(t)$ to denote the corresponding vector for $t \in T(\Sigma_{Xor}, N)$ and $\mathcal{R}^{-1}(v)$ to denote the normalized term t with $\mathcal{R}(t) = v$.

Lemma 6. *For finite sets of names* N *and* $\tilde{n}$*, a message* M*, and a finite substitution* σ *where all terms in* $range(\sigma)$ *are built using only names from* N*,* $\nu\,\tilde{n}\,.\,\sigma \vdash_{E_{Xor}} M$ *if and only if* $\mathcal{R}(M) \in span(\mathcal{R}(range(\sigma) \cup N \setminus \tilde{n}))$*, where* $span(\mathcal{R}(range(\sigma) \cup N \setminus \tilde{n}))$ *denotes the subspace of* $\mathbb{F}_2^{|N|}$ *generated by the vectors in* $\mathcal{R}(range(\sigma) \cup N \setminus \tilde{n})$.

Proof: ($\Rightarrow$): Assume that $\nu\,\tilde{n}\,.\,\sigma \vdash_{E_{Xor}} M$. We prove that $\mathcal{R}(M) \in span(\mathcal{R}(range(\sigma) \cup N \setminus \tilde{n}))$ by rule induction. The rules CONST, KNOW, and EQUAL are obvious. The rule APPLY corresponds to computing the sum of two terms or computing the all zero vector, which is always in $span(\mathcal{R}(range(\sigma) \cup N \setminus \tilde{n}))$.
($\Leftarrow$): Assume that $\mathcal{R}(M) \in span(\mathcal{R}(range(\sigma) \cup N \setminus \tilde{n}))$. Then $\mathcal{R}(M) = a_1 + \ldots + a_k$ for $a_i \in \mathcal{R}(range(\sigma) \cup N \setminus \tilde{n})$. All the $\mathcal{R}^{-1}(a_i)$ can be derived using the CONST and KNOW rules. Then the APPLY rule can be used with $\oplus$ to XOR the terms and derive $\mathcal{R}^{-1}(\mathcal{R}(M))$. Finally, the EQUAL rule can be used to derive M if it is not normalized. ∎

Theorem 4. *There is no derivation trace using the equational theory* $Eq(\Sigma_{Xor}, E_{Xor})$ *that establishes a shared secret. If* $tr \in \mathsf{TR}_{E_{Xor}}$, $\mathsf{Learn}(\mathbb{A}, S) \in tr$*, and* $\mathsf{Learn}(\mathbb{B}, S) \in tr$*, then* $\phi_{tr} \vdash_{E_{Xor}} S$.

Proof: We prove this by contradiction. Assume that there is a trace tr where all names are bound with a minimal number of exchanged messages such that $\mathbb{A}$ and $\mathbb{B}$ derive a shared secret S. Then the list of exchanged messages is $H = [M_1, \ldots, M_k]$ and there are finite sets of bound names $N_{\mathbb{A}}$ and $N_{\mathbb{B}}$ created by $\mathbb{A}$ and $\mathbb{B}$ respectively such that all messages in the trace are in $T(\Sigma_{Xor}, N_{\mathbb{A}} \cup N_{\mathbb{B}})$. We assume without loss of generality that $\mathbb{A}$ sent the last message M_k. Note that $T(\Sigma_{Xor}, N_{\mathbb{A}} \cup N_{\mathbb{B}})/E_{Xor}$ is isomorphic to the direct sum of $\mathbb{F}_2^{|N_{\mathbb{A}}|}$ and $\mathbb{F}_2^{|N_{\mathbb{B}}|}$. Since S is a shared secret, we use Lemma 6 to conclude $\mathcal{R}(S) \notin span(\mathcal{R}(H))$ and $\mathcal{R}(S) \in span(\mathcal{R}(H \cup N_{\mathbb{A}})) \cap span(\mathcal{R}(H \cup N_{\mathbb{B}}))$. We can therefore write $\mathcal{R}(S)$ as $v_B + v_P$ with $v_B \in span(\mathcal{R}(N_{\mathbb{B}}))$ and $v_P \in span(\mathcal{R}(H))$. But then $\mathcal{R}^{-1}(v_B)$ must also be a shared secret since it is derivable by $\mathbb{B}$ and $\mathbb{A}$, and $\phi_{tr} \vdash_{E_{Xor}} \mathcal{R}^{-1}(v_B)$ if and only if $\phi_{tr} \vdash_{E_{Xor}} S$. But v_B is derivable by $\mathbb{B}$ before receiving M_k, which contradicts our assumption that tr is minimal in the number of exchanged messages. ∎

B. Abelian Groups

The theory of abelian groups can be presented by the following equations.

$$\begin{aligned}\Sigma_{AG} =&\{+, 0, -\}\\ E_{AG} =&\{(x+y)+z \simeq x+(y+z),\\ &\quad x+y \simeq y+x,\\ &\quad x+0 \simeq x,\\ &\quad x+(-x) \simeq 0\}\end{aligned}$$

Note that for a finite set of names N, $T(\Sigma_{AG}, N)/E_{AG} \simeq \mathbb{Z}^{|N|}$. I.e., the set of messages over N is isomorphic to the free module over $\mathbb{Z}$ generated by N. An isomorphism is given by the map $\mathcal{R}$ with $\mathcal{R}(n_i) = v_i$, the vector where the only nonzero entry is a 1 at position i, $\mathcal{R}(0) = (0, \ldots, 0)$, $\mathcal{R}(t_1 + t_2) = \mathcal{R}(t_1) + \mathcal{R}(t_2)$, and $\mathcal{R}(-t) = -\mathcal{R}(t)$.

Lemma 7. *For finite sets of names N and $\tilde{\mathrm{n}}$, a message M, and a finite substitution σ where all terms in $range(\sigma)$ are built using only names from N, $\nu\,\tilde{\mathrm{n}}\,.\,\sigma \vdash_{E_{AG}} M$ if and only if $\mathcal{R}(M) \in span(\mathcal{R}(range(\sigma) \cup N \setminus \tilde{\mathrm{n}}))$ where $span(\mathcal{R}(range(\sigma) \cup N \setminus \tilde{\mathrm{n}}))$ denotes the submodule of $\mathbb{Z}^{|N|}$ generated by the vectors in $\mathcal{R}(range(\sigma) \cup N \setminus \tilde{\mathrm{n}})$.*

Proof: We immediately obtain a proof by replacing E_{Xor} with E_{AG} and $\mathbb{F}_2^{|N|}$ with $\mathbb{Z}^{|N|}$ in the proof of Lemma 6. ∎

Theorem 5. *There is no derivation trace using the equational theory $Eq(\Sigma_{AG}, E_{AG})$ that establishes a shared secret. If $tr \in \mathsf{TR}_{E_{AG}}$, $\mathsf{Learn}(\mathbb{A}, S) \in tr$, and $\mathsf{Learn}(\mathbb{B}, S) \in tr$, then $\phi_{tr} \vdash_{E_{AG}} S$.*

Proof: The proof resembles our proof of Theorem 4. To obtain a contradiction, assume that there is a trace tr where all names are bound with a minimal number of exchanged messages such that $\mathbb{A}$ and $\mathbb{B}$ derive a shared secret S. Then the list of exchanged messages is $H = [M_1, \ldots, M_k]$ and there are finite sets of fresh names $N_{\mathbb{A}}$ and $N_{\mathbb{B}}$ created by $\mathbb{A}$ and $\mathbb{B}$ respectively such that all messages in the trace are in $T(\Sigma_{AG}, N_{\mathbb{A}} \cup N_{\mathbb{B}})$. We assume without loss of generality that $\mathbb{A}$ sent the last message M_k. Note that $T(\Sigma_{AG}, N_{\mathbb{A}} \cup N_{\mathbb{B}})/E_{AG}$ is isomorphic to the direct sum of $\mathbb{Z}_2^{|N_{\mathbb{A}}|}$ and $\mathbb{Z}_2^{|N_{\mathbb{B}}|}$. Since S is a shared secret, from Lemma 7 we conclude $\mathcal{R}(S) \notin span(\mathcal{R}(H))$ and $\mathcal{R}(S) \in span(\mathcal{R}(H \cup N_{\mathbb{A}})) \cap span(\mathcal{R}(H \cup N_{\mathbb{B}}))$. We can therefore write $\mathcal{R}(S)$ as $v_B + v_P$ with $v_B \in span(\mathcal{R}(N_{\mathbb{B}}))$ and $v_P \in span(\mathcal{R}(H))$. But then $\mathcal{R}^{-1}(v_B)$ must also be a shared secret since it is derivable by $\mathbb{B}$ and $\mathbb{A}$, and $\phi_{tr} \vdash_{E_{AG}} \mathcal{R}^{-1}(v_B)$ if and only if $\phi_{tr} \vdash_{E_{AG}} S$. But v_B is derivable by $\mathbb{B}$ before receiving M_k, which contradicts our assumption that tr is minimal in the number of exchanged messages. ∎

VI. Combination Results for Impossibility

In Section IV, we have presented an automated method for deciding impossibility for subterm convergent theories. Unfortunately, not all theories in cryptography are subterm convergent. However, many of those theories can be presented as the disjoint union of a subterm convergent theory, such as Sym or $PubKey$, and another theory that is not subterm convergent, such as Xor or AG.

In this section, we consider an equational theory $Eq(\Sigma, E)$ that is the disjoint union of two equational presentations (Σ_1, E_1) and (Σ_2, E_2). I.e., $E = E_1 \uplus E_2$ and $\Sigma = \Sigma_1 \uplus \Sigma_2$, where E_i only contains equations over $T(\Sigma_i, \mathcal{V})$. We prove that secret establishment for such a theory E is possible if and only if it is possible in one of the theories E_1 or E_2. This allows us to combine our automatic method for the subterm convergent subtheory with, for example, algebraic methods for the other subtheory.

A. Factors, Interface Subterms, Normalization

We now define the notions of *sign*, *alien subterm*, *factor*, and *interface subterm* for such a theory. These definitions are adopted from earlier work [19,20] on combining equational theories. Let $t \in T(\Sigma, \mathcal{V} \cup \mathcal{N})$, then $sign(t) = i$ if $t = f(t_1, \ldots, t_k)$ for $f \in \Sigma_i$ and $sign(t) = 0$ if $t \in \mathcal{V} \cup \mathcal{N}$. A subterm u of t is *alien* if $sign(u) \neq sign(t)$.

Definition 6. *The* factors *of a term t are the maximal alien subterms $Fct(t)$. The* interface subterms *$ISt(t)$ of t are the subterms where a sign change occurs.*

$$ISt(t) = t \cup \bigcup_{s \in Fct(t)} ISt(s)$$

Note that the ordered completion $\mathcal{O}_E$ of E corresponds to the disjoint union of the ordered completions $\mathcal{O}_{E_1}$ and $\mathcal{O}_{E_2}$. See [18] for details.

We adopt the following three lemmas from [19] without providing our own proofs. These lemmas characterize the interaction between normalization, replacements, and deducibility.

Lemma 8. *If all factors of a message M are in normal form, then either $sign(M) = sign(M{\downarrow})$ and $Fct(M{\downarrow}) \subseteq Fct(M) \cup \{\mathrm{n}_{min}\}$ or $M{\downarrow} \in Fct(M) \cup \{\mathrm{n}_{min}\}$.*

The intuition behind this lemma is that if all factors of a term are normalized, then the normalization of the term does not affect its factors, i.e., either the factors of the normalized term are a subset of the original factors or the normalized term is a factor. Note that in both cases we must account for the case where free variables in the equations introduce n_{min}. The proof idea is to show the lemma's claim for a minimal reduction step and then extend the result to a minimal reduction sequence.

Lemma 9. *Let $M = \zeta[F_1, \ldots, F_k]$ be a message with normalized factors F_i. Let ρ be a bijective replacement that replaces the factors in M with fresh names. Then $(M{\downarrow})^\rho = (\zeta[F_1^\rho, \ldots, F_k^\rho]){\downarrow}$.*

This lemma states that normalization commutes with the replacement of the normalized factors by fresh names. The proof is based on Lemma 8 and the fact that we consider consistent theories where names are in normal form.

The next lemma states that deduction in the theory E_1 is not affected by replacing interface subterms whose sign is 2 by fresh names. Of course, the lemma is valid for swapped theory indices 1 and 2.

Lemma 10. *Let $\phi = \nu\,\tilde{\mathrm{n}}\,.\,\sigma$ be a frame and $M \in \mathcal{M}_\Sigma$ such that M and all the terms in $range(\sigma)$ are in normal form. Let $F_2 = \{N \mid N \in ISt(range(\sigma) \cup \{M\}) \wedge sign(N) = 2\}$, let $\tilde{\mathrm{n}}_{F_2}$ be a set of names not occuring in ϕ and M where $\tilde{\mathrm{n}}_{F_2}$ is of the same cardinality as F_2, and let $\rho_2 : F_2 \to \tilde{\mathrm{n}}_{F_2}$ be a bijective replacement. Then*

$$\phi \vdash_{E_1} M \text{ if and only if } \nu\,\tilde{\mathrm{n}}_{F_2} \cup \tilde{\mathrm{n}}\,.\,\sigma^{\rho_2} \vdash_{E_1} M^{\rho_2}.$$

To illustrate theses definitions and results, consider the equational theory $Eq(\Sigma, E)$ for $\Sigma = \Sigma_{Xor} \cup \Sigma_{Sym}$ and $E = E_{Xor} \cup E_{Sym}$.

Example 3. *The factors of $m = enc(\langle n_1, n_1\rangle \oplus n_2, n_3)$ are $\langle n_1, n_1\rangle \oplus n_2$ and n_3. The set of interface subterms $ISt(m)$ is $\{m, \langle n_1, n_1\rangle \oplus n_2, n_3, \langle n_1, n_1\rangle, n_2, n_1\}$.*

An example of the first case of Lemma 8 is $enc(dec(enc(n_1 \oplus n_2, k), k), k')\!\downarrow = enc(n_1 \oplus n_2, k')$ and an example of the second case, where the value of sign changes is $dec(enc(n_1 \oplus n_2, k), k)\!\downarrow = n_1 \oplus n_2$.

B. Combination Result

We first prove two lemmas that are required for our main result. The first lemma is similar to Lemma 3 and states that all interface subterms of learned messages not equal to n_{min} must have been learned by one of the agents.

Lemma 11. *Let $tr \in \mathsf{TR}_E$, $A \in \mathcal{A}$ and $M, N \in \mathcal{M}_\Sigma$ such that $\mathsf{Learn}(A, M) \in tr$ and $N \in ISt(M) \setminus \{\mathrm{n}_{min}\}$. Then there is a $B \in \mathcal{A}$ that constructs N in tr.*

Proof: Proof by induction over TR_E. The only nontrivial step is an extension by the DERIVE rule. Let $M = f(M_1, \ldots, M_k)$ and consider the appended event $\mathsf{Learn}_{\mathsf{Derive}(f(M_1,\ldots,M_k))}(A, M\!\downarrow)$. We distinguish two cases. If $sign(M\!\downarrow) \neq sign(M)$ then $M\!\downarrow \in Fct(f(M_1, \ldots, M_k)) \cup \{\mathrm{n}_{min}\} \subseteq ISt(\{M_1, \ldots, M_k\}) \cup \{\mathrm{n}_{min}\}$ and therefore $ISt(M\!\downarrow) \subseteq ISt(\{M_1, \ldots, M_k\}) \cup \{\mathrm{n}_{min}\}$. If $sign(M\!\downarrow) = sign(M)$, then $ISt(M\!\downarrow) \subseteq ISt(M) \cup \{\mathrm{n}_{min}, M\!\downarrow\}$. In both cases, there are Learn events for all interface subterms of $M\!\downarrow$ not equal to n_{min} by the induction hypothesis. ∎

Lemma 12. *Let $tr \in \mathsf{TR}_E$, then n_{min} is not learned in tr.*

Proof: If there is a trace tr where one of the agents learns n_{min}, then there must be a term T with $\mathrm{n}_{min} \notin names(T)$ such that $T =_E \mathrm{n}_{min}$. This term T corresponds to the tree of construction events that build n_{min} with constants and names from $\mathcal{N}\setminus\{\mathrm{n}_{min}\}$ as leaves. But since equational theories are stable under replacement of names by names, we also have $T =_E n$ for a name $n \neq \mathrm{n}_{min}$. By transitivity we obtain $\mathrm{n}_{min} =_E n$. contradicting the assumption that E is consistent. ∎

Using these lemmas, we now prove our combination result for impossibility.

Theorem 6. *Let $Eq(\Sigma, E)$ be the disjoint union of equational presentations (Σ_1, E_1) and (Σ_2, E_2). If there is a trace $tr \in \mathsf{TR}_{\Sigma,E}$ that establishes a shared secret, then there is either a trace $tr_1 \in \mathsf{TR}_{\Sigma_1,E_1}$ or a trace $tr_2 \in \mathsf{TR}_{\Sigma_2,E_2}$ that establishes a shared secret.*

Proof: If there is a trace in $\mathsf{TR}_{\Sigma,E}$ that establishes a shared secret, then there is a minimal trace $tr \in MTR_{\Sigma,E}$ that establishes a shared secret S, where the last event in tr is $\mathsf{Learn}_{\mathsf{Derive}(f(M_1,\ldots,M_k))}(\mathbb{B}, S)$ for some messages $M_1, \ldots, M_k$ and a function symbol $f \in \Sigma$. We assume without loss of generality that $f \in \Sigma_1$ and show that tr can be translated to a trace $tr_\rho \in \mathsf{TR}_{\Sigma_1,E_1}$ that establishes a translated secret S^ρ.

We first prove that for a given minimal trace $tr \in \mathsf{MTR}_E$, we can find an injective replacement ρ from $\{N \mid N \in ISt(tr) \wedge sign(N) = 2\}$ to $\mathcal{N} \setminus names(tr)$ and a trace tr_ρ in $\mathsf{TR}_{\Sigma_1,E_1}$. We define tr_ρ as the translation of tr where Derive events $\mathsf{Learn}_{\mathsf{Derive}(f(M_1,\ldots,M_k))}(A, M) \in tr$ with $sign(M) = 2$ are replaced by events $\mathsf{Learn}_{\mathsf{Fresh}}(A, M^\rho)$ and all other events $Ev(A, M) \in tr$ are replaced by events $Ev(A, M^\rho)$.

We prove this by induction. The statement obviously holds for the EMPTY trace. Now consider the step cases and assume that ρ and tr_ρ have the desired properties for a minimal trace tr.

- PUBLIC: A minimal trace contains no $\mathsf{Learn}_{\mathsf{Public}}$ events.
- RECV: The event $\mathsf{Learn}_{\mathsf{Recv}}(A, M)$ is added to tr, then ρ and $tr_\rho \cdot \mathsf{Learn}_{\mathsf{Recv}}(A, M^\rho)$ have the desired properties by the induction hypothesis.
- SEND: The event $\mathsf{Send}(A, M)$ is added to tr, then ρ and $tr_\rho \cdot \mathsf{Send}(A, M^\rho)$ have the desired properties by the induction hypothesis.
- FRESH: The event $\mathsf{Learn}_{\mathsf{Fresh}}(A, N)$ is added to tr. If there is an M such that $\rho(M) = N$, we can define $\rho' = \rho[M \mapsto N']$ for some $N' \notin names(tr_\rho) \cup \{N\}$, i.e., ρ' is identical to ρ except for $\rho'(M) = N'$. Otherwise, we define $\rho' = \rho$. In both cases ρ' and $tr_{\rho'} \cdot \mathsf{Learn}_{\mathsf{Fresh}}(A, N^{\rho'})$ have the desired properties.
- DERIVE: The event $\mathsf{Learn}_{\mathsf{Derive}(f(M_1,\ldots,M_k))}(A, M)$ is added to tr. We make a case distinction on the sign of f. First, if $f \in \Sigma_1$, then ρ and $tr_\rho \cdot \mathsf{Learn}_{\mathsf{Derive}(f(M_1^\rho,\ldots,M_k^\rho))}(A, M^\rho)$ have the desired properties. The trace is valid since $f(M_1^\rho, \ldots, M_k^\rho)\!\downarrow = (f(M_1, \ldots, M_k)\!\downarrow)^\rho = M^\rho$ by Lemma 9. Second, if $f \in \Sigma_2$, then we know that $sign(M) = 2$. Otherwise,

M is a factor of $f(M_1, \ldots, M_k)$ or n_{min}. The first case is impossible by Lemma 11 and the minimality of the trace since M is not a shared secret and since there cannot be two construction events for the same message in a minimal trace. The second case is also impossible because of Lemma 12. Therefore, we define $\rho' = \rho[M \mapsto N]$ for some $N \notin names(tr_\rho)$. Then ρ' and $tr_{\rho'} \cdot \mathsf{Learn}_{\mathsf{Fresh}}(A, M^{\rho'})$ have the desired properties.

This implies that for any trace $tr \in \mathsf{MTR}_{\Sigma,E}$ that establishes a secret S using $f \in \Sigma_1$ in the last event, we can find a replacement ρ such that the trace tr_ρ is in $\mathsf{TR}_{\Sigma_1,E_1}$ and establishes the secret S^ρ. Since all messages in tr_ρ are simply translations of the corresponding messages in tr by applying ρ, both $\mathbb{A}$ and $\mathbb{B}$ learn the message S^ρ in tr_ρ and $send(tr_\rho) = send(tr)^\rho$. Also note that $bound(tr_\rho) = range(\rho) \cup bound(tr)$. Using Lemma 10 and $send(tr) \not\vdash_E S$, which trivially implies $send(tr) \not\vdash_{E_1} S$, we conclude that $send(tr)^\rho \not\vdash_{E_1} S^\rho$. Hence S^ρ is a secret. ■

C. Applications

We have collected all impossibility results from this paper in Table 3. Moreover, we have augmented the table with possibility results from the literature, thereby providing an overview of existing results. Note that there are theories where, to the best of our knowledge, the problem is still open. For example, there are no such results for the theory of (nonabelian) groups, blind signatures, and homomorphic encryption.

The results presented are for minimal disjoint theories, in the sense that they cannot sensibly be further decomposed. For disjoint equational theories where secret establishment is impossible, we can apply Theorem 6 to obtain an impossibility result for the union of the two theories. For example, secret establishment using the equational theory E_{Sym} that models symmetric encryption, pairing, and a hash function combined with the theory E_{Xor} for XOR is impossible. Here, our combination result allows us to use different proof methods for the subtheories. Namely, our decision procedure for the subterm convergent theory E_{Sym} and the proof based on the isomorphism with an $\mathbb{F}_2$ vector space for XOR. Note that neither of these methods can be used for the union of the two theories.

Another application of our combination result is that we can further optimize the decision procedure from Section IV-B. Namely, we can split a subterm convergent theory into disjoint subtheories that can be checked separately. For example, as presented in the table, the theory E_{Sym} can be split into the theories E_1 for pairing, E_2 for symmetric encryption, and the empty theory E_3 for the free function symbol h. We can then call FIND-DERIVATION-TRACES(E_i) for each of the subtheories and need only check deducibility for $\vdash_{E_i}$ in the given call.

VII. Conclusion

We have initiated the systematic study of impossibility results for secret establishment protocols. We have presented three different kinds of results. First, we gave a formal model for proving impossibility results for secret establishment for cryptographic operations described by equational theories. We used this model to give the first formal impossibility proof for symmetric encryption in the symbolic setting. Afterwards, we generalized this result to necessary and sufficient conditions for the impossibility of secret establishment for any subterm convergent theory. This directly yields a decision procedure and constitutes a first step towards machine assisted analysis of impossibility. Second, we adapt algebraic methods to prove the impossibility of secret establishment for XOR and abelian groups. Finally, we proved a combination result that enables modular impossibility proofs.

As future work, we plan to investigate other equational theories where the impossibility question is still open. We would also like to investigate other security properties, such as authentication and perfect forward secrecy, as well as different adversary models. Another interesting question is whether our labeling technique and decision procedure could be used for protocol synthesis.

References

[1] O. Pereira and J. Quisquater, "On the impossibility of building secure cliques-type authenticated group key agreement protocols," *Journal of Computer Security*, vol. 14, no. 2, pp. 197–246, 2006.

[2] D. Micciancio and S. Panjwani, "Optimal communication complexity of generic multicast key distribution," *IEEE/ACM Transactions on Networking (TON)*, vol. 16, no. 4, pp. 803–813, 2008.

[3] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. IT-22, no. 6, pp. 644–654, 1976.

[4] S. Pohlig and M. Hellman, "An improved algorithm for computing logarithms over gf (p) and its cryptographic significance," *IEEE Transactions on information Theory*, Jan 1978.

[5] R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, Jan 1978.

[6] M. Rabi and A. Sherman, "Associative one-way functions: a new paradigm for secret-key agreement and digital signatures," *University of Maryland at College Park, MD, USA*, p. 13, 1993.

[7] U. Maurer and P. Schmid, "A calculus for security bootstrapping in distributed systems," *Journal of Computer Security*, vol. 4, no. 1, pp. 55–80, 1996.

[8] C. Boyd, "Security architectures using formal methods," *Selected Areas in Communications, IEEE Journal on*, vol. 11, no. 5, pp. 694–701, Jun 1993.

Theory	Possible?	Protocols/Proof technique
Free function symbols (e.g., a hash function h)	No	Subterm convergent
Pairing	No	Subterm convergent
Symmetric encryption	No	Subterm convergent
Signatures	No	Subterm convergent
Public-key encryption	Yes	Key transport in [5]
A, AC, ACU	Yes	Key agreement in [6]
ACUN (XOR)	No	Separate proof
AG	No	Separate proof
DH-Exponentiation	Yes	Key agreement [3] and Key transport in [4]

Figure 3: (Im)possibility results

[9] D. Dolev and A. Yao, "On the Security of Public Key Protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983.

[10] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Mantovani, S. Mödersheim, D. von Oheimb, M. Rusinowitch, J. Santiago *et al.*, "The AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications," *CAV '05: Proceedings of the 17th Conference on Computer Aided Verification*, 2005.

[11] B. Blanchet, "An efficient cryptographic protocol verifier based on prolog rules," in *CSFW*, 2001, pp. 82–96.

[12] M. Abadi and P. Rogaway, "Reconciling two views of cryptography (the computational soundness of formal encryption)*," *Journal of cryptology*, vol. 15, no. 2, pp. 103–127, 2002.

[13] V. Cortier and B. Warinschi, "Computationally sound, automated proofs for security protocols," *Lecture Notes in Computer Science*, vol. 3444, pp. 157–171, 2005.

[14] M. Baudet, V. Cortier, and S. Kremer, "Computationally sound implementations of equational theories against passive adversaries," *Information and Computation*, vol. 207, no. 4, pp. 496–520, 2009.

[15] F. Baader and T. Nipkow, *Term rewriting and all that*. Cambridge Univ Pr, 1999.

[16] N. Dershowitz and J. Jouannaud, "Rewrite systems, Handbook of theoretical computer science (vol. B): formal models and semantics," 1991.

[17] M. Abadi and C. Fournet, "Mobile values, new names, and secure communication," *Proceedings of the 28th ACM SIGPLAN-SIGACT symposium on principles of programming languages*, 2001.

[18] F. Baader and K. Schulz, "Unification in the union of disjoint equational theories: Combining decision procedures," *Journal of Symbolic Computation*, 1996.

[19] M. Arnaud, V. Cortier, and S. Delaune, "Combining algorithms for deciding knowledge in security protocols," in *FroCos*, 2007, pp. 103–117.

[20] Y. Chevalier and M. Rusinowitch, "Combining intruder theories," in *ICALP*. Springer, 2005, pp. 639–651.

[21] M. Abadi and V. Cortier, "Deciding knowledge in security protocols under equational theories," *Theoretical Computer Science*, vol. 367, no. 1-2, pp. 2–32, 2006.

[22] V. Cortier and S. Delaune, "Deciding knowledge in security protocols for monoidal equational theories," in *LPAR*, 2007, pp. 196–210.

[23] W. Nutt, "Unification in monoidal theories," in *Proceedings of the 10th international conference on automated deduction*, vol. 449. Springer, 1990, pp. 618–632.