COMPLETENESS OF THE AUTHENTICATION TESTS

SHADDIN F. DOGHMI, JOSHUA D. GUTTMAN, AND F. JAVIER THAYER

Abstract. The *shapes* of a cryptographic protocol are its minimal, essentially different executions. In previous work, we have described a search algorithm to discover the shapes of a protocol, and implemented the algorithm in a Cryptographic Protocol Shape Analyzer cpsa.

In this paper, we show its *completeness*, i.e. that every shape can in fact be found in a finite number of steps. The steps in question are applications of two *authentication tests*, fundamental protocol patterns for analysis and heuristics for protocol design. We formulate the authentication tests in a new, stronger form, for which completeness is true.

We also introduce skeletons, as partial descriptions of executions. The information-preserving maps between skeletons are a kind of homomorphism. The completeness result shows that any homomorphism from a skeleton to a full execution may be digested into a sequence of atomic steps leading to a shape.

The executions of cryptographic protocols frequently have very few essentially different forms, which we call *shapes*. By enumerating these shapes, we may ascertain whether they all satisfy a security condition such as an authentication or confidentiality property. We may also find other anomalies, which are not necessarily counterexamples to the security goals, such as involving unexpected participants, or involving more local runs than expected. In this paper, we prove that two kinds of step suffice for finding all of the shapes of a protocol, within a pure Dolev-Yao model [4].

We use the strand space theory [7]. A *skeleton* represents regular (non-penetrator) behavior that might make up part of an execution, and a *homomorphism* is an information-preserving map between skeletons. Skeletons are partially-ordered structures, like fragments of Lamport diagrams [9] or fragments of message sequence charts [8]. A skeleton is *realized* if it is nonfragmentary, i.e. it contains exactly the regular behavior of some execution. A realized skeleton is a *shape* if it is minimal in a sense we will make precise (Definition 4.9). We *search* for shapes using the authentication tests [7] to find new strands to add when a skeleton is not large enough to be realized. In [3], we describe a search strategy that we have implemented in cpsa, a Cryptographic Protocol Shape Analyzer.

In the present paper, we focus on the main technical result underlying cpsa. This is *completeness*, in the sense that—for any protocol—the authentication tests lead to every shape for that protocol (Thm. 5.8). Given a protocol, we cannot give predict a bound on how many steps may be required [5].

The type-and-effect system for spi calculus [6] is related to the authentication tests, but differs from our work in two ways. First, we do not use the syntactically-driven form of a type system, but instead a direct analysis of behaviors. Second, type-and-effect systems aim at a sound approximation, whereas our work provides

[1]Supported by the National Security Agency and by MITRE-Sponsored Research.

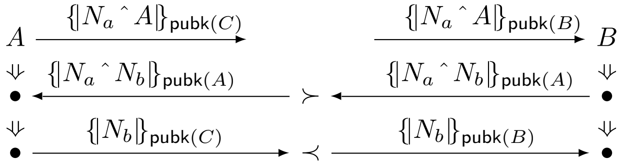

FIGURE 1. Needham-Schroeder Shape for B ($\mathsf{privk}(A)$ uncompromised, N_b fresh)

actual counterexamples when a security goal is not met. Blanchet's ProVerif [1] is also based on a sound approximation, and may thus refuse to certify a protocol even though there are no counterexamples.

CPSA's search is related to the second version of Athena [13], which adopted the authentication tests from an early version of [7]. Our work, however, is distinguished from Athena in several ways. First, it involves the regular behaviors alone; we never represent adversary activity within a shape. Second, we have improved both the search and the theory. In particular, we have introduced the notion of shape, which defines a criterion for which possible executions should be considered, among the infinitely many executions (of unbounded size) of any protocol. Third, we have now created versions of the authentication tests strong enough for completeness to be true.

The shapes describe protocol executions of all sizes; we do not follow the widely practiced *bounded* protocol analysis (e.g. [2, 11]).

1. The Idea of Shapes

In practice, protocols have remarkably few shapes. The Needham-Schoeder-Lowe [12, 10] protocol has only one. This holds whether we take the point of view of a responder B, asking what global behavior must have occurred if B has had a local run of the protocol, or whether we start from a local run of an originator A. In either case, the other party must have had a matching run. A, however, can never be sure that the last message it sends was received by B, as A is no longer expecting to receive any further messages. Uniqueness of shape is perhaps not surprising for as strong a protocol as Needham-Schroeder-Lowe.

The NS Shape. However, even a flawed protocol such as the original Needham-Schroeder protocol may have a unique shape, shown in Fig. 1. Suppose B's nonce N_b has been freshly chosen and A's private key $\mathsf{privk}(A)$ is uncompromised, and B has executed the strand shown at the right in Fig. 1. In protocols using asymmetric encryption, the private keys are used only by recipients to destructure incoming messages. Given that—on a particular occasion—B received and sent these messages, what must have occurred elsewhere in the network?

A must have had a partially matching strand, with the messages sent and received in the order indicated by the arrows of both kinds and the connecting symbols $\prec$. These symbols mean that the endpoints are ordered, but that other behavior may intervene, whether adversary strands or regular strands. A's strand is only partially matching, because the principal A meant to contact is some C which may or may not equal B. There is no alternative: Any diagram containing the responder strand of Fig. 1 must contain at least an instance of the initiator strand, with the events ordered as shown, or it cannot have happened.

Such a diagram is a *shape*. A shape consists of the regular strands of some bundle, forming a *minimal* set containing the initial regular strands (in this case, just the right-hand column). Possible bundles may freely add adversary behavior. Each shape is relative to assumptions about keys and freshness, in this case that $\mathsf{privk}(A)$ is uncompromised and N_b freshly chosen.

Although there is a single shape, there are two ways that this shape may be realized in bundles. Either (1) C's private key may be compromised, in which case we may complete this diagram with adversary activity to obtain the Lowe attack [10]; or else (2) $C = B$, leading to the intended run.

Some protocols have more than one shape, Otway-Rees, e.g., having four. In searching for shapes, one starts from some initial set of strands. Typically, the initial set is a singleton, which we refer to as the "point of view" of the analysis.

Skeletons, Homomorphisms, Shapes. Newly introduced terminology is in **boldface**. A **skeleton** $\mathbb{A}$ is (1) a finite set of regular nodes, equipped with additional information. The additional information consists of (2) a partial order $\preceq_{\mathbb{A}}$ on the nodes indicating causal precedence; (3) a set of keys $\mathsf{non}_{\mathbb{A}}$; and (4) a set of atomic values $\mathsf{unique}_{\mathbb{A}}$. Values in $\mathsf{non}_{\mathbb{A}}$ must originate nowhere in $\mathbb{A}$, whereas those in $\mathsf{unique}_{\mathbb{A}}$ originate at most once in $\mathbb{A}$.[1]

$\mathbb{A}$ is **realized** if it has precisely the regular behavior of some bundle $\mathcal{B}$. Every message received by a regular participant either should have been sent previously, or should be constructable by the adversary using messages sent previously.

Example 1.1. *Fig. 1 shows a skeleton $\mathbb{A}_{ns}$, with $\mathsf{non}_{\mathbb{A}_{ns}} = \{\mathsf{privk}(A)\}$ and $\mathsf{unique}_{\mathbb{A}_{ns}} = \{N_b\}$; indeed $\mathbb{A}_{ns}$ is a realized skeleton. The right-hand strand of Fig. 1, B's responder strand, also forms a skeleton $\mathbb{A}_b$ with the same choice of $\mathsf{non}, \mathsf{unique}$, although $\mathbb{A}_b$ is not realized. The result of replacing C by B throughout $\mathbb{A}_{ns}$—in particular, replacing $\mathsf{pubk}(C)$ by $\mathsf{pubk}(B)$—yields another skeleton $\mathbb{A}_{nsi}$, which represents the Needham-Schroeder intended run.*

A **homomorphism** is a map H from $\mathbb{A}_0$ to $\mathbb{A}_1$, written $H\colon \mathbb{A}_0 \mapsto \mathbb{A}_1$. We represent it as a pair of maps (ϕ, α), where ϕ maps the nodes of $\mathbb{A}_0$ into those of $\mathbb{A}_1$, and α is a **replacement** mapping atomic values into atomic values. We write $t \cdot \alpha$ for the result of applying a replacement α to a message t. $H = (\phi, \alpha)$ is a homomorphism iff: (1) ϕ respects strand structure, and $\mathsf{msg}(n) \cdot \alpha = \mathsf{msg}(\phi(n))$ for all $n \in \mathbb{A}_0$; (2) $m \preceq_{\mathbb{A}_0} n$ implies $\phi(m) \preceq_{\mathbb{A}_1} \phi(n)$; (3) $\mathsf{non}_{\mathbb{A}_0} \cdot \alpha \subset \mathsf{non}_{\mathbb{A}_1}$; and (4) $\mathsf{unique}_{\mathbb{A}_0} \cdot \alpha \subset \mathsf{unique}_{\mathbb{A}_1}$.

Homomorphisms are *information-preserving* transformations. Each skeleton $\mathbb{A}_0$ describes the realized skeletons reachable from $\mathbb{A}_0$ by homomorphisms. Since homomorphisms compose, if $H\colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ then any realized skeleton accessible from $\mathbb{A}_1$ is accessible from $\mathbb{A}_0$. Thus, $\mathbb{A}_1$ preserves the information in $\mathbb{A}_0$: $\mathbb{A}_1$ describes a subset of the realized skeletons described by $\mathbb{A}_0$.

A homomorphism may supplement the strands of $\mathbb{A}_0$ with additional behavior in $\mathbb{A}_1$; it may affect atomic parameter values; and it may identify different nodes together, if their strands are compatible in messages sent and positions in the partial ordering.

[1] When $n \Rightarrow^* n'$ and $n' \in \mathbb{A}$, we require $n \in \mathbb{A}$ and $n \preceq_{\mathbb{A}} n'$.

Example 1.2. *The map $H_{ns}\colon \mathbb{A}_b \mapsto \mathbb{A}_{ns}$ embedding the responder strand of Fig. 1 into $\mathbb{A}_{ns}$ is a homomorphism. Likewise if we embed the first two nodes of B's strand (rather than all of $\mathbb{A}_b$) into $\mathbb{A}_{ns}$. Another homomorphism $H_i\colon \mathbb{A}_{ns} \mapsto \mathbb{A}_{nsi}$ rewrites each occurrence of C in $\mathbb{A}_{ns}$ to B, hence each occurrence of $\mathsf{pubk}(C)$ to $\mathsf{pubk}(B)$. It exhibits the Needham-Schroeder intended run as an instance of Fig. 1.*

A homomorphism $H = (\phi, \alpha)$ is **nodewise injective** if the function ϕ on nodes is injective. The nodewise injective homomorphisms determine a useful partial order on homomorphisms: When for some nodewise injective H_1, $H_1 \circ H = H'$, we write $H \leq_n H'$. If $H \leq_n H' \leq_n H$, then H and H' are isomorphic.

A homomorphism $H\colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ is a **shape** iff (a) $\mathbb{A}_1$ is realized and (b) H is $\leq_n$-minimal among homomorphisms from $\mathbb{A}_0$ to realized skeletons. If H is a shape, and we can factor H into $\mathbb{A}_0 \overset{H_0}{\mapsto} \mathbb{A}' \overset{H_1}{\mapsto} \mathbb{A}_1$, where $\mathbb{A}'$ is realized, then $\mathbb{A}'$ cannot contain fewer nodes than $\mathbb{A}_1$, or identify fewer atomic values. $\mathbb{A}_1$ is as small and as general as possible.

We call a *skeleton* $\mathbb{A}_1$ a shape when the homomorphism H (usually an embedding) is understood. In this looser sense, Fig. 1 shows the shape $\mathbb{A}_{ns}$. Strictly, the embedding $H_{ns}\colon \mathbb{A}_b \mapsto \mathbb{A}_{ns}$ is the shape. The embedding $H_{nsi}\colon \mathbb{A}_b \mapsto \mathbb{A}_{nsi}$, with target the Needham-Schroeder intended run $\mathbb{A}_{nsi}$, is not a shape. $\mathbb{A}_{ns}$ identifies fewer atoms, and the map replacing C with B is a nodewise injective $H_i\colon \mathbb{A}_{ns} \mapsto \mathbb{A}_{nsi}$, so $H_{ns} \leq_n H_i \circ H_{ns} = H_{nsi}$.

Shapes exist below realized skeletons: If $H\colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ with $\mathbb{A}_1$ realized, then the set of shapes H_1 with $H_1 \leq_n H$ is finite and non-empty.

2. Terms, Strands, and Bundles

In this section and Section 4 we give precise definitions, which include a number of fine points which seemed an unnecessary distraction in Section 1. In this section, the definitions of replacement and protocol (Defs. 2.1, 2.5) are new versus [7].

Algebra of Terms. Terms (or messages) form a free algebra A, built from atomic terms via constructors. The atoms are partitioned into the types *principals*, *texts*, *keys*, and *nonces*. An inverse operator is defined on keys. There may be additional functions on atoms, such as an injective *public key of* function mapping principals to keys, or an injective *long term shared key of* function mapping pairs of principals to keys. These functions are not constructors, and their results are atoms. For definiteness, we include here functions $\mathsf{pubk}(a), \mathsf{ltk}(a)$ mapping principals to (respectively) their public keys and to a symmetric key shared on a long-term basis with a fixed server S. $\mathsf{pubk}(a)^{-1}$ is a's private key, where $\mathsf{pubk}(a)^{-1} \neq \mathsf{pubk}(a)$. We often write the public key pair as K_a, K_a^{-1}. By contrast, $\mathsf{ltk}(a)^{-1} = \mathsf{ltk}(a)$.

Atoms, written in italics (e.g. a, N_a, K^{-1}), serve as indeterminates (variables). We assume A contains infinitely many atoms of each type. Terms in A are freely built from atoms using *tagged concatenation* and *encryption*. The tags are chosen from a set of constants written in sans serif font (e.g. tag). The tagged concatenation using tag of t_0 and t_1 is written $\mathsf{tag}\,\hat{}\,t_0\,\hat{}\,t_1$. Tagged concatenation using the distinguished tag null of t_0 and t_1 is written $t_0\,\hat{}\,t_1$. Encryption takes a term t and an atomic key K, and yields a term as result written $\{\!|t|\!\}_K$.

Replacements have only atoms in their range:

Definition 2.1 (Replacement, Application)**.** *A* replacement *is a function α mapping atoms to atoms, such that (1) for every atom a, $\alpha(a)$ is an atom of the same* *type as* a*, and (2)* α *is a homomorphism with respect to the operations on atoms, i.e.,* $\alpha(K^{-1}) = (\alpha(K))^{-1}$ *and* $\alpha(\mathsf{pubk}(a)) = \mathsf{pubk}(\alpha(a))$*.*

The application *of* α *to* t*, written* $t \cdot \alpha$*, homomorphically extends* α*'s action on atoms. More explicitly, if* $t = a$ *is an atom, then* $a \cdot \alpha = \alpha(a)$*; and:*

$$\begin{aligned} (\mathsf{tag}\,\hat{}\,t_0\,\hat{}\,t_1)\cdot\alpha &= \mathsf{tag}\,\hat{}\,(t_0\cdot\alpha)\,\hat{}\,(t_1\cdot\alpha) \\ (\{\!|t|\!\}_K)\cdot\alpha &= \{\!|t\cdot\alpha|\!\}_{K\cdot\alpha} \end{aligned}$$

Application distributes through larger objects such as pairing and sets. Thus, $(x, y) \cdot \alpha = (x \cdot \alpha, y \cdot \alpha)$*, and* $S \cdot \alpha = \{x \cdot \alpha \colon x \in S\}$*. If* $x \notin \mathsf{A}$ *is a simple value such as an integer or a symbol, then* $x \cdot \alpha = x$*.*

Strands and Origination. Since replacements map atoms to atoms, not to compound terms, unification is very simple. Two terms are unifiable if and only if they have the same abstract syntax tree structure, with the same tags associated with corresponding concatenations, and the same type for atoms at corresponding leaves. To unify t_1, t_2 means to partition the atoms at the leaves; a most general unifier is a finest partition that maps a, b to the same c whenever a appears at the end of a path in t_1 and b appears at the end of the same path in t_2. If two terms t_1, t_2 are unifiable, then $t_1 \cdot \alpha$ and $t_2 \cdot \beta$ are still unifiable.

The direction $+$ means transmission, and the direction $-$ means reception:

Definition 2.2 (Strand Spaces)**.** *A* direction *is one of the symbols* $+, -$*. A* directed term *is a pair* (d, t) *with* $t \in \mathsf{A}$ *and* d *a direction, normally written* $+t, -t$*.* $(\pm\mathsf{A})^*$ *is the set of finite sequences of directed terms.*

A strand space *over* A *is a structure containing a set* Σ *and two mappings: a* trace mapping $\mathsf{tr} : \Sigma \to (\pm\mathsf{A})^*$ *and a replacement application operator* $(s, \alpha) \mapsto s \cdot \alpha$ *such that (1)* $\mathsf{tr}(s \cdot \alpha) = (\mathsf{tr}(s)) \cdot \alpha$*, and (2)* $s \cdot \alpha = s' \cdot \alpha$ *implies* $s = s'$*.*

By (2), Σ has infinitely many copies of each s, i.e. strands s' with $\mathsf{tr}(s') = \mathsf{tr}(s)$.

Definition 2.3. *A* penetrator strand *has trace of one of the following forms:*

M_t*:* $\langle +t \rangle$ *where* $t \in$*text, principal,nonce* K_K*:* $\langle +K \rangle$
$\mathsf{C}_{g,h}$*:* $\langle -g,\ -h,\ +g\,\hat{}\,h \rangle$ $\mathsf{S}_{g,h}$*:* $\langle -g\,\hat{}\,h,\ +g,\ +h \rangle$
$\mathsf{E}_{h,K}$*:* $\langle -K,\ -h,\ +\{\!|h|\!\}_K \rangle$ $\mathsf{D}_{h,K}$*:* $\langle -K^{-1},\ -\{\!|h|\!\}_K,\ +h \rangle$*.*

If s is a penetrator strand, then $s \cdot \alpha$ is a penetrator strand of the same kind.

The *subterm* relation, written $\sqsubseteq$, is the least reflexive, transitive relation such that (1) $t_0 \sqsubseteq \mathsf{tag}\,\hat{}\,t_0\,\hat{}\,t_1$; (2) $t_1 \sqsubseteq \mathsf{tag}\,\hat{}\,t_0\,\hat{}\,t_1$; and (3) $t \sqsubseteq \{\!|t|\!\}_K$. Notice, however, $K \not\sqsubseteq \{\!|t|\!\}_K$ unless (anomalously) $K \sqsubseteq t$. We say that a key K is *used for encryption* in a term t if for some t_0, $\{\!|t_0|\!\}_K \sqsubseteq t$.

A *node* is a pair $n = (s, i)$ where $i \leq \mathsf{length}(\mathsf{tr}(s))$; $\mathsf{strand}(s, i) = s$; and the *direction* and *term* of n are those of $\mathsf{tr}(s)(i)$. We prefer to write $s \downarrow i$ for the node $n = (s, i)$. A term t *originates* at node n if n is positive, $t \sqsubseteq \mathsf{msg}(n)$, and $t \not\sqsubseteq \mathsf{msg}(m)$ whenever $m \Rightarrow^+ n$. Thus, t originates on n if t is part of a message transmitted on n, and t was neither sent nor received previously on this strand. If a originates on strand s, we write $\mathcal{O}(s, a)$ to refer to the node on which it originates.

Example 2.4. N_a *originates on the first node of the Needham-Schroeder initiator strand* s_i*, so we write* $\mathcal{O}(s_i, N_a) = s_i \downarrow 1$*.* N_b *originates on the second node of the responder strand* s_r*, i.e.* $\mathcal{O}(s_r, N_b) = s_r \downarrow 2$*. More precisely,* $\mathcal{O}(s_r, N_b) = s_r \downarrow 2$ *unless* $N_b = N_a$*, because if the two nonces were the same, then* N_b *would not originate on the responder strand at all. Instead, it would have been received before* *being re-transmitted. Thus, the replacement $\beta = [N_b \mapsto N_a]$ destroys the point of origination. Even if we have $\mathcal{O}(s_r, N_b) = s_r \downarrow 2$, we have $\mathcal{O}(s_r \cdot \beta, N_b \cdot \beta)$ undefined. In this sense, applying β to s_r is a kind of degeneracy that destroys a point of origination. When we have assumed that a value such as N_b originates uniquely, we will avoid applying replacements that would destroy its point of origination. (See Def. 2.5, regular strands, and Def. 4.6, homomorphism.)*

A *listener role* is a regular strand $\mathsf{Lsn}[a]$ with trace $\langle -a \rangle$. It documents that a is available on its own to the adversary, unprotected by encryption. Since replacements respect type, atoms of different type must be overheard by different roles. We assume each protocol Π has listener roles $\mathsf{Lsn}[N]$ and $\mathsf{Lsn}[K]$ for nonces and keys respectively, with traces $\langle -N \rangle$ and $\langle -K \rangle$.

Protocols and Bundles.

Definition 2.5 (Protocols)**.** *A* candidate $\langle \Pi, \mathsf{strand\_non}, \mathsf{strand\_unique} \rangle$ *consists of: (1) a finite set* Π *of strands—containing the listener strands* $\mathsf{Lsn}[N], \mathsf{Lsn}[K]$*—called the* roles *of the protocol; (2) a function* $\mathsf{strand\_non}$ *mapping each role r to a finite set of keys* $\mathsf{strand\_non}_r$*, called the non-originating keys of r; and (3) a function* $\mathsf{strand\_unique}$ *mapping each role r to a finite set of atoms* $\mathsf{strand\_unique}_r$ *called the uniquely originating atoms of r.*

A candidate $\langle \Pi, \mathsf{strand\_non}, \mathsf{strand\_unique} \rangle$ *is a* protocol *if (1)* $K \in \mathsf{strand\_non}_r$ *implies that K does not occur in any node of r, but either K or K^{-1} is used for encryption on some term of* $\mathsf{tr}(r)$*; and (2)* $a \in \mathsf{strand\_unique}_r$ *implies that a originates on r, i.e. $\mathcal{O}(r, a)$ is well defined.*

The regular strands *of* $\langle \Pi, \mathsf{strand\_non}, \mathsf{strand\_unique} \rangle$ *form the set* $\Sigma_\Pi =$

$$\{r \cdot \alpha \colon r \in \Pi \text{ and } \forall a \in \mathsf{strand\_unique}_r, (\mathcal{O}(r,a)) \cdot \alpha = \mathcal{O}(r \cdot \alpha, a \cdot \alpha)\}.$$

The non-originating keys $\mathsf{strand\_non}_r$ and uniquely originating atoms $\mathsf{strand\_unique}_r$ are used in Defs. 4.3 and 5.1, Clauses 1c,d. The condition that constrains $r \cdot \alpha$ based on $\mathcal{O}(r, a)$ is a non-degeneracy condition. It says that replacement α determines an instance of r only if it does not cause a value a, assumed uniquely originating, to collide with another value already encountered in executing r. Since for $a \in \mathsf{strand\_unique}_r$, the left hand side of $(\mathcal{O}(r,a)) \cdot \alpha = \mathcal{O}(r \cdot \alpha, a \cdot \alpha)$ is well-defined, we interpret the equation as meaning that the right hand side is also well-defined, and has the same value.

Example 2.6 (Needham-Schroeder Protocol)**.** *The Needham-Schroeder protocol has a set Π_{ns} of roles containing the two roles shown in Fig. 1 and two listener roles, to hear nonces and keys. For each $r \in \Pi_{ns}$,* $\mathsf{strand\_non}_r = \emptyset = \mathsf{strand\_unique}_r$.

Setting $\mathsf{strand\_non}_{init} = \{\mathsf{privk}(B)\}$, $\mathsf{strand\_non}_{resp} = \{\mathsf{privk}(A)\}$ *reproduces the original Needham-Schroeder assumption that each peer chosen is uncompromised. The protocol achieves its goals relative to this assumption.*

Setting $\mathsf{strand\_unique}_{init} = \{N_a\}$ *would express the assumption that every initiator uses a strong random number generator to select nonces, so that the probability of a collision or of an adversary guessing a nonce is negligible.*

The set $\mathcal{N}$ of all nodes forms a directed graph $\mathcal{G} = \langle \mathcal{N}, (\rightarrow \cup \Rightarrow) \rangle$ with edges $n_1 \rightarrow n_2$ for communication (with the same term, directed from positive to negative node) and $n_1 \Rightarrow n_2$ for succession on the same strand.

Definition 2.7 (Bundle)**.** *A finite acyclic subgraph $\mathcal{B} = \langle \mathcal{N}_{\mathcal{B}}, (\rightarrow_{\mathcal{B}} \cup \Rightarrow_{\mathcal{B}})\rangle$ of $\mathcal{G}$ is a* bundle *if (1) if $n_2 \in \mathcal{N}_{\mathcal{B}}$ is negative, then there is a unique $n_1 \in \mathcal{N}_{\mathcal{B}}$ with $n_1 \rightarrow_{\mathcal{B}} n_2$; and (2) if $n_2 \in \mathcal{N}_{\mathcal{B}}$ and $n_1 \Rightarrow n_2$, then $n_1 \Rightarrow_{\mathcal{B}} n_2$. When $\mathcal{B}$ is a bundle, $\preceq_{\mathcal{B}}$ is the reflexive, transitive closure of $(\rightarrow_{\mathcal{B}} \cup \Rightarrow_{\mathcal{B}})$.*

A bundle $\mathcal{B}$ is over $\langle \Pi, \mathsf{strand\_non}, \mathsf{strand\_unique}\rangle$ *if for every $s \downarrow i \in \mathcal{B}$, (1) either $s \in \Sigma_\Pi$ or s is a penetrator strand; (2) if $s = r \cdot \alpha$ and $a \in \mathsf{strand\_non}_r \cdot \alpha$, then a does not occur in $\mathcal{B}$; and (3) if $s = r \cdot \alpha$ and $a \in \mathsf{strand\_unique}_r \cdot \alpha$, then a originates at most once in $\mathcal{B}$.*

Example 2.8. *Fig. 1 is a bundle if we replace C with B and then connect arrows with matching labels. Alternatively, it becomes a bundle by adding penetrator strands to unpack values encrypted with K_C and repackage them encrypted with K_B.*

We say that a strand s is *in $\mathcal{B}$* if s has at least one node in $\mathcal{B}$. Henceforth, assume fixed some arbitrary protocol $\langle \Pi, \mathsf{strand\_non}, \mathsf{strand\_unique}\rangle$.

Proposition 2.9. *Let $\mathcal{B}$ be a bundle. $\preceq_{\mathcal{B}}$ is a well-founded partial order. Every non-empty set of nodes of $\mathcal{B}$ has $\preceq_{\mathcal{B}}$-minimal members.*

Let α be a replacement. Suppose for every regular strand $s = r \cdot \beta$ in $\mathcal{B}$, for every $b \in \mathsf{strand\_unique}_r \cdot \beta$, we have $(\mathcal{O}(s, b)) \cdot \alpha = \mathcal{O}(s \cdot \alpha, b \cdot \alpha)$. Then $\mathcal{B} \cdot \alpha$ is a bundle.

3. Strengthened Authentication Tests in Bundles

To direct the process of searching for realized skeletons, we use the *authentication tests* [7] in a strengthened and simplified form.

We say that t_0 *occurs only within* S in t, where S is a set of terms, if:

(1) $t_0 \not\sqsubseteq t$; or
(2) $t \in S$; or
(3) $t \neq t_0$ and either (3a) $t = \{\!|t_1|\!\}_K$ and t_0 occurs only within S in t_1; or (3b) $t = \mathsf{tag}\,\hat{}\,t_1\,\hat{}\,t_2$ and t_0 occurs only within S in each t_i $(i = 1, 2)$.

So t_0 occurs only within S in t if in the abstract syntax tree, every path from the root t to an occurrence of t_0 as a subterm of t traverses some $t_1 \in S$ before reaching t_0. On the other hand, t_0 *occurs outside* S *in* t if t_0 does not occur only within S in t. This means that $t_0 \sqsubseteq t$ and there is a path from the root to an occurrence of t_0 as a subterm of t that traverses no $t_1 \in S$.

Example 3.1 (Needham-Schroeder Occurrences)**.** *N_b occurs only within the set $S_r = \{\{\!|N_a\,\hat{}\,N_b|\!\}_{\mathsf{pubk}(A)}\}$ in the term $\{\!|N_a\,\hat{}\,N_b|\!\}_{\mathsf{pubk}(A)}$. However, N_b occurs outside S_r in the term $\{\!|N_b|\!\}_{\mathsf{pubk}(B)}$.*

We say that a is *protected* in $\mathcal{B}$ iff $\mathsf{msg}(n) \neq a$ for all $n \in \mathcal{B}$. Equivalently, a is protected in $\mathcal{B}$ iff the listener strand for a is not in $\mathcal{B}'$ for any $\mathcal{B}' \sim_{\mathsf{L}} \mathcal{B}$; that is, $(\mathsf{Lsn}[a] \downarrow 1) \notin \mathcal{B}'$.

We say that a is *protected up to m* in $\mathcal{B}$ iff, for all $n \in \mathcal{B}$, if $\mathsf{msg}(n) = a$ then $m \prec_{\mathcal{B}} n$. We write $a \in \mathsf{Prot}_m(\mathcal{B})$ if a is protected up to m in $\mathcal{B}$.

By the definitions of the penetrator strands for encryption and decryption (Definition 2.3), if the adversary uses K for encryption or decryption anywhere in $\mathcal{B}$, then K is not protected in $\mathcal{B}$. Thus, the adversary cannot create any encrypted term with a protected key K. If K^{-1} is protected, it cannot decrypt any term encrypted with K. If a key is protected up to a negative node m, then the adversary cannot use that key to prepare the term received on m.

For instance, if $\mathsf{privk}(A)$ is assumed uncompromised in some bundle $\mathcal{B}$, then $\mathsf{privk}(A) \in \mathsf{Prot}_m(\mathcal{B})$.

Proposition 3.2 (Outgoing Authentication Test)**.** *Suppose that $n_0, n_1 \in \mathcal{B}$, and*

$$S \subset \{\{\!|t|\!\}_K \colon K^{-1} \in \mathsf{Prot}_{n_1}(\mathcal{B})\}.$$

Suppose that a originates uniquely in $\mathcal{B}$ at node n_0 and occurs only within S in $\mathsf{msg}(n_0)$, but a occurs outside S in $\mathsf{msg}(n_1)$.

There is an integer i and a regular strand $s \in \Sigma_\Pi$ such that $m_1 = s \downarrow i \in \mathcal{B}$ is positive, and i is the least integer k such that a occurs outside S in $\mathsf{msg}(s \downarrow k)$. Moreover, there is a node $m_0 = s \downarrow j$ with $j < i$ such that $a \sqsubseteq \mathsf{msg}(s \downarrow j)$, and $n_0 \preceq_\mathcal{B} m_0 \Rightarrow^+ m_1 \preceq_\mathcal{B} n_1$.

Proof. Apply Prop. 2.9 to $T =$

$$\{m \colon m \preceq_\mathcal{B} n_1 \text{ and } a \text{ occurs outside } S \text{ in } \mathsf{msg}(m)\}.$$

$n_1 \in T$, so T has $\preceq_\mathcal{B}$-minimal members m_1. Since keys K used in S have $K^{-1} \in \mathsf{Prot}(\mathcal{B})$, m_1 cannot lie on a decryption penetrator D-strand. By the assumptions, a does not originate on m_1, so that m_1 does not lie on a M-strand or K-strand. By the definitions of S and "occurs only within," m_1 does not lie on a S-, C-, or E-strand. Thus, m_1 lies on some $s \in \Sigma_\Pi$ at some index i. □

In the Outgoing Authentication Test, we call $m_0 \Rightarrow^+ m_1$ an *outgoing transforming edge* for a, S. It transforms the occurrence of a from lying only within S to occurring outside it. We call (n_0, n_1) an *outgoing test pair for a, S* when these nodes satisfy the condition in the first paragraph of the proposition. When we do not know the set $\mathsf{Prot}_{n_1}(\mathcal{B})$, we consider the set $\mathsf{used}(S)$ of keys used for some outermost encryption in S as an approximation, and we speak of an *outgoing test pair for a, S.*

Example 3.3. *In the Needham-Schroeder protocol, with responder role s_r, the nodes $(s_r \downarrow 2), (s_r \downarrow 3)$ form an outgoing test pair for N_b, S_r, where S_r is as given in Example 3.1. If the initiator role is s_i, then the edge $s_i \downarrow 2 \Rightarrow s_i \downarrow 3$ is a outgoing transforming edge for N_b, S_r.*

On the other hand, the nodes $(s_i \downarrow 1), (s_i \downarrow 2)$ form an outgoing test pair for N_a, S_i, where S_i is the singleton set $\{\{\!|N_a \,\hat{}\, A|\!\}_{\mathsf{pubk}(C)}\}$. Letting $s'_r = s_r \cdot [B \mapsto C]$, then $s'_r \downarrow 1 \Rightarrow s'_r \downarrow 2$ forms an outgoing transforming edge for N_a, S_i.

Proposition 3.4 (Incoming Authentication Test)**.** *Suppose that $n_1 \in \mathcal{B}$ is negative, $t = \{\!|t_0|\!\}_K \sqsubseteq \mathsf{msg}(n_1)$, and $K \in \mathsf{Prot}(\mathcal{B})$. There exists a regular $m_1 \prec n_1$ such that t originates at m_1.*

The proof applies Prop. 2.9 to the set $T = \{m \colon m \preceq_\mathcal{B} n_1 \text{ and } t \sqsubseteq \mathsf{msg}(m)\}$. We call m_1 as an *incoming transforming node*, and n_1 an *incoming test node.*

4. Preskeletons, Skeletons, and Homomorphisms

Skeletons. A preskeleton is potentially the regular (non-penetrator) part of a bundle or of some portion of a bundle.

A preskeleton consists of nodes annotated with additional information, indicating order relations among the nodes, uniquely originating atoms, and non-originating atoms. We say that an atom a *occurs* in a set nodes of nodes if for some $n \in \mathsf{nodes}$, $a \sqsubseteq \mathsf{msg}(n)$. A key K is *used* in nodes if for some $n \in \mathsf{nodes}$, $\{\!|t|\!\}_K \sqsubseteq \mathsf{msg}(n)$. We say that a key K is *mentioned in* nodes if K or K^{-1} either occurs or is used in nodes. For a non-key a, a is mentioned if it occurs.

Definition 4.1. *A four-tuple* $\mathbb{A} = (\mathsf{nodes}, \preceq, \mathsf{non}, \mathsf{unique})$ *is a* preskeleton *if:*

(1) nodes *is a finite set of regular nodes;* $n_1 \in \mathsf{nodes}$ *and* $n_0 \Rightarrow^+ n_1$ *implies* $n_0 \in \mathsf{nodes}$*;*

(2) $\preceq$ *is a partial ordering on* nodes *such that* $n_0 \Rightarrow^+ n_1$ *implies* $n_0 \preceq n_1$*;*

(3) non *is a set of keys, and for all* $K \in \mathsf{non}$*, either* K *or* K^{-1} *is used in* nodes*;*

(3′) *for all* $K \in \mathsf{non}$*,* K *does not occur in* nodes*;*

(4) unique *is a set of atoms, and for all* $a \in \mathsf{unique}$*,* a *occurs in* nodes*.*

A preskeleton $\mathbb{A}$ *is a* skeleton *if in addition:*

(4′) $a \in \mathsf{unique}$ *implies* a *originates at no more than one node in* nodes*.*

We select components of a preskeleton using subscripts, so, in $\mathbb{A} = (N, R, S, S')$, $\preceq_{\mathbb{A}}$ means R and $\mathsf{unique}_{\mathbb{A}}$ means S'. $\mathbb{A}$ need not contain all of the nodes of a strand, just some initial subsequence. We write $n \in \mathbb{A}$ to mean $n \in \mathsf{nodes}_{\mathbb{A}}$, and we say that a strand s is in $\mathbb{A}$ when at least one node of s is in $\mathbb{A}$. The $\mathbb{A}$-height of s is the largest i with $s \downarrow i \in \mathbb{A}$. By Clauses 3, 4, $\mathsf{unique}_{\mathbb{A}} \cap \mathsf{non}_{\mathbb{A}} = \emptyset$.

Example 4.2. $\mathbb{A}_{ns}$*, shown in Fig 1, is a skeleton with* $\mathsf{non} = \{\mathsf{privk}(A)\}$*,* $\mathsf{unique} = \{N_b\}$*. Its ordering is generated from the double arrows* $\Rightarrow$*, single arrows* $\rightarrow$*, and precedence signs.* $\mathbb{A}_b$*, containing only the responder strand* s_r *on the right side of Fig 1, is also a skeleton (equipped with* $\mathsf{non} = \{\mathsf{privk}(A)\}$*,* $\mathsf{unique} = \{N_b\}$*). However, if we adjoin a copy* $s'_r = s_r \cdot [B \mapsto C]$ *to* $\mathbb{A}_{ns}$*, then the result is not a skeleton, but only a preskeleton* $\mathbb{A}_{pre}$*.* N_b *originates both at* $s_r \downarrow 2$ *and at* $s'_r \downarrow 2$*. If instead we adjoin* $s''_r = s_r \cdot [B \mapsto C, N_b \mapsto N'_b]$*, we obtain a skeleton* $\mathbb{A}'_{pre}$*.*

The skeletons for a protocol $\langle \Pi, \mathsf{strand\_non}, \mathsf{strand\_unique} \rangle$ are defined like the bundles for that protocol.

Definition 4.3. $\mathbb{A}$ *is a* preskeleton for *protocol* $\langle \Pi, \mathsf{strand\_non}, \mathsf{strand\_unique} \rangle$ *iff for every* $n \in \mathsf{nodes}_{\mathbb{A}}$ *with* $n = s \downarrow i$*, (1)* $s \in \Sigma_{\Pi}$*; (2) if* $s = r \cdot \alpha$ *and* $a \in \mathsf{strand\_non}_r \cdot \alpha$*, then* a *does not occur in* $\mathbb{A}$*; and (3) if* $s = r \cdot \alpha$ *and* $a \in \mathsf{strand\_unique}_r \cdot \alpha$*, then* $a \in \mathsf{unique}_{\mathbb{A}}$*.* $\mathbb{A}$ *is a* skeleton for *a protocol if* $\mathbb{A}$ *is a skeleton, and* $\mathbb{A}$ *is a preskeleton for that protocol.*

Skeletons and Bundles. Bundles correspond to certain skeletons:

Definition 4.4. *Bundle* $\mathcal{B}$ realizes *skeleton* $\mathbb{A}$ *if:*

(1) *The nodes of* $\mathbb{A}$ *are the regular nodes* $n \in \mathcal{B}$*.*

(2) $n \preceq_{\mathbb{A}} n'$ *just in case* $n, n' \in \mathsf{nodes}_{\mathbb{A}}$ *and* $n \preceq_{\mathcal{B}} n'$*.*

(3) $K \in \mathsf{non}_{\mathbb{A}}$ *iff case* K *or* K^{-1} *is used in* $\mathsf{nodes}_{\mathbb{A}}$ *but* K *occurs nowhere in* $\mathcal{B}$*.*

(4) $a \in \mathsf{unique}_{\mathbb{A}}$ *iff* a *originates uniquely in* $\mathcal{B}$*.*

The skeleton *of* $\mathcal{B}$ *is the skeleton that it realizes. The skeleton of* $\mathcal{B}$*, written* $\mathsf{skeleton}(\mathcal{B})$*, is uniquely determined.* $\mathbb{A}$ *is* realized *if some* $\mathcal{B}$ *realizes it.*

Two bundles $\mathcal{B}, \mathcal{B}'$ *are* similar*, written* $\mathcal{B} \sim_{\mathsf{L}} \mathcal{B}'$*, if they differ only in what listener strands they contain. Two realized skeletons* $\mathbb{A}, \mathbb{A}'$ *are* similar*, written* $\mathbb{A} \sim_{\mathsf{L}} \mathbb{A}'$*, if for some* $\mathcal{B}, \mathcal{B}'$ *with* $\mathcal{B} \sim_{\mathsf{L}} \mathcal{B}'$*,* $\mathbb{A} = \mathsf{skeleton}(\mathcal{B})$ *and* $\mathbb{A}' = \mathsf{skeleton}(\mathcal{B}')$*.*

By condition (4), $\mathcal{B}$ does not realize $\mathbb{A}$ if $\mathbb{A}$ is a preskeleton but not a skeleton. Given a skeleton $\mathbb{A}$, methods derived from [7] determine whether $\mathbb{A}$ is realized. Skeleton $\mathbb{A}_{ns}$ from Example 4.2 is realized, but N_b is not.

Homomorphisms. When $\mathbb{A}$ is a preskeleton, we may apply a substitution α to it, subject to the same condition as in Prop. 2.9. Namely, suppose α is a replacement, and suppose that for each regular strand $s = r \cdot \beta$ such that s has nodes in $\mathbb{A}$, and for each atom $b \in u_r \cdot \beta$,

$$(\mathcal{O}(s, b)) \cdot \alpha = \mathcal{O}(s \cdot \alpha, b \cdot \alpha).$$

Then $\mathbb{A} \cdot \alpha$ is a well defined object. However, it is not a preskeleton when $x \cdot \alpha = y \cdot \alpha$ where $x \in \mathsf{non}_{\mathbb{A}}$ while y occurs in $\mathbb{A}$. In this case, no further identifications can restore the preskeleton property. So we are interested only in replacements with the property that $x \cdot \alpha = y \cdot \alpha$ and $x \in \mathsf{non}_{\mathbb{A}}$ implies y does not occur in $\mathbb{A}$. On this condition, $\mathbb{A} \cdot \alpha$ is a preskeleton.

However, $\mathbb{A}$ may be a skeleton, while objects built from it are preskeletons but not skeletons. In a preskeleton, we can sometimes, though, restore the skeleton unique origination property ($4'$) by a mapping ϕ that carries the two points of origination to a common node. This will be possible only if the terms on them are the same, and likewise for the other nodes in $\mathbb{A}$ on the same strands. We regard ϕ, α as an information-preserving, or more specifically information-increasing, map. It has added the information that a_1, a_2, which could have been distinct, are in fact the same, and thus the nodes n_1, n_2, which could have been distinct, must also be identified.

Example 4.5. *$\mathbb{A}'_{pre}$ is a skeleton, but the result of applying the replacement $[N'_b \mapsto N_b]$ yields the preskeleton $\mathbb{A}_{pre}$ which is not a skeleton. If the map $\phi\colon \mathsf{nodes}_{\mathbb{A}_{pre}} \mapsto \mathsf{nodes}_{\mathbb{A}_{ns}}$ maps the successive nodes of the strand s'_r to the nodes of the strand s_r, then it will identify $s'_r \downarrow 2$ with $s_r \downarrow 2$, and thus restore the unique point of origination for N_b.*

Definition 4.6. *Let $\mathbb{A}_0, \mathbb{A}_1$ be preskeletons, α a replacement, $\phi\colon \mathsf{nodes}_{\mathbb{A}_0} \to \mathsf{nodes}_{\mathbb{A}_1}$. $H = [\phi, \alpha]$ is a* homomorphism *if*

1*a*. *For all $n \in \mathbb{A}_0$, $\mathsf{msg}(\phi(n)) = \mathsf{msg}(n) \cdot \alpha$, with the same direction;*

1*b*. *For all s, i, if $s \downarrow i \in \mathbb{A}$ then there is an s' s.t. for all $j \leq i$, $\phi(s \downarrow j) = (s', j)$;*

2. *$n \preceq_{\mathbb{A}_0} m$ implies $\phi(n) \preceq_{\mathbb{A}_1} \phi(m)$;*
3. *$\mathsf{non}_{\mathbb{A}_0} \cdot \alpha \subset \mathsf{non}_{\mathbb{A}_1}$;*
4. *$\mathsf{unique}_{\mathbb{A}_0} \cdot \alpha \subset \mathsf{unique}_{\mathbb{A}_1}$; and $\phi(\mathcal{O}(s, a)) = \mathcal{O}(s', a \cdot \alpha)$ whenever $a \in \mathsf{unique}_{\mathbb{A}_0}$, $\mathcal{O}(s, a) \in \mathbb{A}_0$, and $\phi(s \downarrow j) = s' \downarrow j$.*

We write $H\colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ when H is a homomorphism from $\mathbb{A}_0$ to $\mathbb{A}_1$. When $a \cdot \alpha = a \cdot \alpha'$ for every a that occurs or is used for encryption in $\mathsf{dom}(\phi)$, then $[\phi, \alpha] = [\phi, \alpha']$; i.e., $[\phi, \alpha]$ is the equivalence class of pairs under this relation.

The condition for $[\phi, \alpha] = [\phi, \alpha']$ implies that the action of α on atoms not mentioned in the $\mathbb{A}_0$ is irrelevant. The condition on $\mathcal{O}$ in Clause 4 avoids the degeneracy in which a point of origination is destroyed for some atom $a \in \mathsf{unique}_{\mathbb{A}_0}$. We stipulate that such degenerate maps are not homomorphisms. For instance, a replacement α that sends both N_a and N_b to the same value would not furnish homomorphisms on $\mathbb{A}_{ns}$. For the responder, expecting to choose a fresh nonce, inadvertently to select the same nonce N_a he has just received, would be an event of negligible probability. Thus, there is no harm in discarding this degenerate set.

A homomorphism $I = [\phi, \alpha]\colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ is an *isomorphism* iff ϕ is a bijection and α is injective. We say that two homomorphisms H_1, H_2 are isomorphic if they differ by an isomorphism; i.e. $H_1 = I \circ H_2$ for some isomorphism I.

When transforming a preskeleton $\mathbb{A}$ into a skeleton, one identifies nodes n, n' if some $a \in \mathsf{unique}_{\mathbb{A}}$ originates on both; to do so, one may need to unify additional atoms that appear in both $\mathsf{msg}(n), \mathsf{msg}(n')$. This process could cascade. However, when success is possible, and the cascading produces no incompatible constraints, there is a canonical (universal) way to succeed:

Proposition 4.7. *Suppose $H_0 \colon \mathbb{A} \mapsto \mathbb{A}'$ with $\mathbb{A}$ a preskeleton and $\mathbb{A}'$ a skeleton.*

There exists a homomorphism $G_{\mathbb{A}}$ and a skeleton $\mathbb{A}_0$ such that $G_{\mathbb{A}} \colon \mathbb{A} \mapsto \mathbb{A}_0$ and, for every skeleton $\mathbb{A}_1$ and every homomorphism $H_1 \colon \mathbb{A} \mapsto \mathbb{A}_1$, for some H, $H_1 = H \circ G_{\mathbb{A}}$. $G_{\mathbb{A}}$ and $\mathbb{A}_0$ are unique to within isomorphism.

We call this universal map $G_{\mathbb{A}}$ (or sometimes its target $\mathbb{A}_0$) the *hull* of $\mathbb{A}$, $\mathsf{hull}(\mathbb{A})$.

We say that a skeleton $\mathbb{A}_0$ is *live* if for some $H, \mathbb{A}_1$, $H \colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ and $\mathbb{A}_1$ is realized. Otherwise, it is *dead*. There are two basic facts about dead skeletons:

Proposition 4.8 (Dead Skeletons)**.** *(1) If $a \in \mathsf{non}_{\mathbb{A}}$ and $(\mathsf{Lsn}[a]) \downarrow 1 \in \mathbb{A}$, then $\mathbb{A}$ is dead. (2) If $\mathbb{A}$ is dead and $H \colon \mathbb{A} \mapsto \mathbb{A}'$, then $\mathbb{A}'$ is dead.*

Shapes. Shapes are minimal realizable skeletons, or more precisely, minimal homomorphisms with realizable targets.

Definition 4.9 (Shape)**.** *$[\phi, \alpha] \colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ is* nodewise injective *if ϕ is an injective function on the nodes of $\mathbb{A}_0$.*

A homomorphism H_0 is nodewise less than or equal to *H_1, written $H_0 \leq_n H_1$, if for some nodewise injective J, $J \circ H_0 = H_1$. H_0 is* nodewise minimal *in a set S if $H_0 \in S$ and for all $H_1 \in S$, $H_1 \leq_n H_0$ implies H_1 is isomorphic to H_0.*

$H \colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ is a shape for $\mathbb{A}_0$ *if H is nodewise minimal among the set of homomorphisms $H' \colon \mathbb{A}_0 \mapsto \mathbb{A}'_1$ where $\mathbb{A}'_1$ is realized.*

The composition of two nodewise injective homomorphisms is nodewise injective, and a nodewise injective $H \colon \mathbb{A} \mapsto \mathbb{A}$ is an isomorphism. Thus, H_0, H_1 are isomorphic if each is nodewise less than or equal to the other. Hence, the relation $\leq_n$ is a partial order on homomorphisms to within isomorphism.

If we speak of a skeleton $\mathbb{A}_0$ as nodewise less than another skeleton $\mathbb{A}_1$, we mean that $H \colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ for some nodewise injective H. When we say that $\mathbb{A}_1$ is a shape, we mean that it is the target of some shape $H \colon \mathbb{A}_0 \mapsto \mathbb{A}_1$, where a particular $\mathbb{A}_0$ is understood from the context.

Proposition 4.10. *Let $H \colon \mathbb{A}_0 \mapsto \mathbb{A}_1$. The set $\mathcal{S} = \{H' \colon H' \leq_n H\}$ is finite (up to isomorphism). If $\mathbb{A}_1$ is realized, then at least one $H' \in \mathcal{S}$ is a shape for $\mathbb{A}_0$.*

Proof. Letting $H = [\phi, \alpha]$, we generate $\mathcal{S}$ by choosing, for each node $n \in (\mathbb{A}_1 \setminus \phi(\mathbb{A}_0))$, whether to omit it and all nodes later than n on the same strand.

We associate each location at which a is mentioned with an atom in $\alpha^{-1}(a)$, the inverse image of a under α. An association is permissible if locations containing the same atom in $\mathbb{A}_0$ are associated with the same atom.

The set $\mathcal{S}$ contains the homomorphisms we get given a choice of nodes to omit and a permissible association. H differs by a renaming from a member of $\mathcal{S}$, namely the one that omits no nodes and associates every occurrence of any a with a single representative from $\alpha^{-1}(a)$. Thus, if $\mathbb{A}_1$ is realized, $\mathcal{S}$ has members with a realized target. Letting $\mathcal{S}' \subseteq \mathcal{S}$ be the set of $H' \in \mathcal{S}$ such that the target of H' is realized, $\mathcal{S}'$ is non-empty and finite; hence, $\mathcal{S}'$ has $\leq_n$-minimal members. □

If $\mathbb{A}_1$ is realized and contains listener strands, and $\mathbb{A}$ results when we omit some of the listener strands, then $\mathbb{A}$ is realized and $\mathbb{A} \sim_{\mathsf{L}} \mathbb{A}_1$. In particular, $\mathbb{A}$ is nodewise less than or equal to $\mathbb{A}_1$. A minimal member of $\mathcal{A}$ will omit all of the listener strands, which is why they do not appear in Fig. 1.

Given a skeleton $\mathbb{A}_0$ as "starting point," we would like to find all the homomorphisms $H \colon \mathbb{A}_0 \mapsto \mathbb{A}$ that lead from $\mathbb{A}_0$ to a shape $\mathbb{A}$. If we find homomorphisms from $\mathbb{A}_0$ to realized skeletons $\mathbb{A}_1$, then Prop. 4.10 tells us how to obtain one or more shapes from each of these realized skeletons. We are thus most interested in homomorphisms H that do not unnecessarily identify occurrences of atoms, as we will try to distinguish the different uses of the same atom in $\mathbb{A}_1$ to find nodewise minimal members of $\mathcal{A}$.

Our search is finished when more realized skeletons cannot yield any shapes we have not yet encountered.

5. The Tests in Skeletons

To adapt the authentication tests to skeletons and homomorphisms, there are essentially two steps. First, we must "pull back" from bundles or realized skeletons to the skeletons that reach them via homomorphisms. Second, since we can no longer read off the safe atoms from $\mathsf{Prot}(\mathcal{B})$. We have only partial information about which atoms will turn out to be safe or compromised. Thus, we speculatively consider both possibilities, i.e. both the possibility that a key will turn out to be compromised, and also the possibility that the transformed nodes need to be explained. We use listener strands to mark the keys we are assuming will be compromised. If this assumption is not consistent, then the skeleton containing the listener strand will be dead, and no homomorphism leads from it to a realized skeleton.

Definition 5.1 (Augmentations, Contractions). (1) *An* augmentation *is an inclusion* $[\mathsf{id}, \mathsf{id}] \colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ *such that:*

(a) $\mathsf{nodes}_{\mathbb{A}_1} \setminus \mathsf{nodes}_{\mathbb{A}_0} = \{s \downarrow j \colon j \leq i\}$ *for some* $s = r \cdot \alpha$*;*

(b) $\preceq_{\mathbb{A}_1}$ *is the transitive closure of (i)* $\preceq_{\mathbb{A}_0}$*; (ii) the strand ordering of* s *up to* i*; and (iii) pairs* (n, m) *or* (n, m) *with* $n \in \mathsf{nodes}_{\mathbb{A}_0}$, $m = s \downarrow j$*, and* $j \leq i$*.*

(c) $\mathsf{non}_{\mathbb{A}_1} = \mathsf{non}_{\mathbb{A}_0} \cup (n_r \cdot \alpha)$*; and*

(d) $\mathsf{unique}_{\mathbb{A}_1} = \mathsf{unique}_{\mathbb{A}_0} \cup (u_r \cdot \alpha)$.

(2) *An augmentation* $H \colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ *is an* outgoing augmentation *if there exists an outgoing test edge* $n_0, n_1 \in \mathbb{A}_0$ *with no outgoing transforming edge in* $\mathbb{A}_0$*, and* $s \downarrow 1 \Rightarrow^* m_0 \Rightarrow^+ s \downarrow i$*, where* $m_0 \Rightarrow^+ s \downarrow i$ *is the earliest transforming edge for this test on* s*. The additional pairs in the ordering (clause 1b(iii)) are the pairs* (n_0, m_0) *and* $((s \downarrow i), n_1)$*.*

(3) *It is an* incoming augmentation *if it adds an incoming transforming edge for an incoming test node in* $\mathbb{A}_0$*. The pair* (m_1, n_1) *in the notation of Prop. 3.4 is the additional pair in the ordering.*

(4) *It is a* listener augmentation for a *if it adds a listener strand* $\mathsf{Lsn}[a]$*, with no pairs added to the ordering.*

(5) *A replacement* α *is a* contraction *for* $\mathbb{A}$ *if there are two distinct atoms* a, b *mentioned in* $\mathbb{A}$ *such that* $a \cdot \alpha = b \cdot \alpha$*. We write* $\mathsf{hull}_\alpha(\mathbb{A})$ *for the canonical homomorphism from* $\mathbb{A}$ *to* $\mathsf{hull}(\mathbb{A} \cdot \alpha)$*, when the latter is defined. (See Prop. 4.7.)*

We can now state the search-oriented version of Prop. 3.2. It states that when a skeleton $\mathbb{A}_0$ with an unsolved outgoing transformed pair can lead to a realized skeleton $\mathbb{A}_1$, we can get there by starting out with one of three kinds of steps: (1) an outgoing augmentation, (2) a contraction, or (3) adding a listener strand to witness for the fact that one of the relevant keys is in fact *not* properly protected by the time we reach $\mathbb{A}_1$.

Since we consider realized skeletons that differ only in their listener strands, we recall that $\mathbb{A}_1 \sim_{\mathsf{L}} \mathbb{A}_2$ if they are both realized and differ only in what listener strands they contain. We will also write $H_1 \sim_{\mathsf{L}} H_2$ if adding listener strands can equalize them; i.e., when the H_i (for $i = 1, 2$) are of the form $H_i \colon \mathbb{A} \mapsto \mathbb{A}_i$, and there are embeddings $E_i \colon \mathbb{A}_i \mapsto \mathbb{A}'$ such that $\mathbb{A}_1 \sim_{\mathsf{L}} \mathbb{A}' \sim_{\mathsf{L}} \mathbb{A}_2$ and $E_1 \circ H_1 = E_2 \circ H_2$.

Theorem 5.2 (Outgoing Augmentation)**.** *Let $H \colon \mathbb{A}_0 \mapsto \mathbb{A}_1$, where $\mathbb{A}_1$ is realized. Let $n_0, n_1 \in \mathbb{A}_0$ be an outgoing test pair for a, S, for which $\mathbb{A}_0$ contains no transforming edge. At least one of the following holds:*

1. *$H = H'' \circ \mathsf{hull}_\alpha(\mathbb{A}_0)$ for some contraction α;*
2. *$H = H'' \circ H'$, where H' is some outgoing augmentation for a, S;*
3. *There is a listener augmentation $H' \colon \mathbb{A}_0 \mapsto \mathbb{A}_0'$ for some $K \in \mathsf{used}(S)$, and a homomorphism $H'' \colon \mathbb{A}_0' \mapsto \mathbb{A}_1'$ such that $H \sim_{\mathsf{L}} H'' \circ H'$.*

Proof. Assuming $H = [\phi, \alpha] \colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ with $\mathbb{A}_1$ realized, say with $\mathsf{skeleton}(\mathcal{B}) = \mathbb{A}_1$, we have the following possibilities. If α contracts any atoms, then we may factor H into a contraction followed by some remainder H'' (clause 1).

If α does not contract any atoms, then $(\phi(n_0), \phi(n_1))$ is an outgoing test pair for $a \cdot \alpha, S \cdot \alpha, X \cdot \alpha$. There are now two cases. First, suppose $X \cdot \alpha \subseteq \mathsf{Prot}_{\phi(n_1)}(\mathcal{B})$. Then we may apply Prop. 3.2 to infer that $\mathcal{B}$ and thus also $\mathbb{A}_1$ contains an outgoing transforming edge $m_0 \Rightarrow^+ m_1$ for $a \cdot \alpha, S \cdot \alpha$. Since α is injective on atoms mentioned in $\mathbb{A}_0$, we may augment $\mathbb{A}_0$ with $(m_0 \cdot \alpha^{-1}) \Rightarrow^+ (m_1 \cdot \alpha^{-1})$.

Second, if there is some $a \in X$ such that $a \cdot \alpha \notin \mathsf{Prot}_{\phi(n_1)} \mathcal{B}$, then there is $\mathbb{A}_1' \sim_{\mathsf{L}} \mathbb{A}_1$ such that $\mathbb{A}_1'$ contains $\mathsf{Lsn}[a \cdot \alpha]$, and $\phi(n_1) \not\preceq (\mathsf{Lsn}[a \cdot \alpha]) \downarrow 1$. Hence, clause 3 is satisfied. □

In applying Theorem 5.2, we prefer to apply Clauses 2, 3 if possible; unnecessary contractions must simply be un-contracted using Prop. 4.10. In particular, we use a contraction α only if either (1) $n_0 \cdot \alpha, n_1 \cdot \alpha$ is no longer an outgoing transformed pair, or else (2) for some candidate outgoing augmentation, $n_0 \cdot \alpha, n_1 \cdot \alpha$ is the most general version of the test that it solves. The latter may occur when the protocol role mentions the same atom at several locations where different atoms are mentioned in n_0, n_1; α must then identify these atoms.

Incoming augmentations are similar to outgoing ones, except that the relevant keys are only those used for encryption in the test node:

Theorem 5.3 (Incoming Augmentation)**.** *Let $H \colon \mathbb{A}_0 \mapsto \mathbb{A}_1$, where $\mathbb{A}_1$ is realized. Let $n_1 \in \mathbb{A}_0$ be a negative node and $\{\!|t_0|\!\}_K \sqsubseteq \mathsf{msg}(n_1)$. If $\{\!|t_0|\!\}_K$ originates nowhere in $\mathbb{A}_0$, then either:*

1. *$H = H'' \circ \mathsf{hull}_\alpha(\mathbb{A}_0)$ for some contraction α;*
2. *$H = H'' \circ H'$, where H' is an incoming augmentation originating $\{\!|t_0|\!\}_K$; or*
3. *There is a listener augmentation $H' \colon \mathbb{A}_0 \mapsto \mathbb{A}_0'$ for K, and a homomorphism $H'' \colon \mathbb{A}_0' \mapsto \mathbb{A}_1'$ such that: (a) $\mathbb{A}_1'$ is realized, (b) $\mathbb{A}_1' \sim_{\mathsf{L}} \mathbb{A}_1$, and (c) $H'' \circ H' = I \circ H$, where I is an inclusion homomorphism.*

Here we use a contraction α only when α is needed to make an incoming augmentation apply. A contraction never eliminates an incoming test node.

When $a \sqsubseteq \mathsf{msg}(m)$, where $a \in \mathsf{unique}_{\mathbb{A}_0}$ and $m \in \mathbb{A}_0$, and a originates at $n \in \mathbb{A}_0$, then n will precede m in any bundle accessible from $\mathbb{A}_0$. That is, if $H\colon \mathbb{A}_0 \mapsto \mathbb{A}_1$ where the latter is realized, then H factors through H' which maps $\mathbb{A}_0$ to the order enrichment $\mathbb{A}_0'$, where $\preceq_{\mathbb{A}_0'}$ is the transitive closure of $(\preceq_{\mathbb{A}_0} \cup (n, m))$. We will rely on this implicitly in what follows. When we need to be explicit about this, to say that a skeleton needs no further enrichment of this kind, we will say that its *order reflects origination.*

Completeness of the Authentication Tests. If a skeleton $\mathbb{A}$ is not realized, does it necessarily contain an outgoing transformed edge or an incoming transformed node? Yes, it does, although to make this precise we must be careful about which atoms are protected, as this is not explicit in an unrealized skeleton.

Definition 5.4 (Penetrator web)**.** *Let $G = \langle \mathcal{N}_G, (\to_G \cup \Rightarrow_G)\rangle$ be a finite acyclic subgraph of $\langle \mathcal{N}, (\to \cup \Rightarrow)\rangle$ such that $\mathcal{N}_G$ consists entirely of penetrator nodes. G is a* penetrator web *with support S and result R if S and R are sets of terms and moreover:*

(1) *If $n_2 \in \mathcal{N}_G$ is negative, then either $\mathsf{msg}(n_2) \in S$ or there is a unique n_1 such that $n_1 \to_G n_2$.*
(2) *If $n_2 \in \mathcal{N}_G$ and $n_1 \Rightarrow n_2$ then $n_1 \Rightarrow_G n_2$.*
(3) *For each $t \in R$, either $t \in S$ or for some positive $n \in \mathcal{N}_G$, $\mathsf{msg}(n) = t$.*

If $n \in \mathcal{B}$ is a negative node, then $\mathcal{B}$ includes a penetrator web G with result $R_G = \{\mathsf{msg}(n)\}$. Its support $S_G = \{\mathsf{msg}(m) \colon m$ is positive regular and $m \prec_{\mathcal{B}} n\}$. We write the set of positive regular nodes preceding a node n as $\mathsf{support}(n)$.

Definition 5.5. *A term t is* penetrator-derivable before n *in $\mathbb{A}$ if there is a penetrator web G with $t \in R_G$ such that:*

(1) *$S_G \subset \mathsf{support}(n)$;*
(2) *If $K \in \mathsf{non}_{\mathbb{A}}$, K does not originate in G_n; and*
(3) *If $a \in \mathsf{unique}_{\mathbb{A}}$ and a originates in $\mathbb{A}$, then a does not originate in G_n.*

Proposition 5.6. *A skeleton $\mathbb{A}$ is realized iff, for every negative $n \in \mathbb{A}$, $\mathsf{msg}(n)$ is penetrator-derivable before n in $\mathbb{A}$.*

Proposition 5.7. *Suppose that $\preceq_{\mathbb{A}}$ reflects origination. If $\mathsf{msg}(n)$ is not penetrator-derivable before n in $\mathbb{A}$, then either:*

(1) *n is an incoming transformed node, i.e., for some $\{\!|t|\!\}_K \sqsubseteq \mathsf{msg}(n)$, $K \in \mathsf{non}_{\mathbb{A}} \cup \mathsf{unique}_{\mathbb{A}}$ and K is not penetrator-derivable before n in $\mathbb{A}$; or else*
(2) *(m, n) is an outgoing transformed pair with respect to a, S for (i) some $m \preceq_{\mathbb{A}} n$; (ii) some $a \in \mathsf{unique}_{\mathbb{A}}$ originating at m; (iii) some set S of encrypted terms such that a occurs only within S in $\mathsf{support}(n)$; and (iv) for each $K \in \mathsf{used}(S)$, K^{-1} is not penetrator-derivable before n in $\mathbb{A}$.*

Proof. Similar to[7, Prop. 7]. □

Recall that shapes (being minimal) do not contain listener strands, so Clause 3 of Theorems 5.2, 5.3 need not appear in the following:

Theorem 5.8 (Authentication Tests Completeness)**.** *Let $J = [\phi, \alpha] \colon \mathbb{A} \mapsto \mathbb{A}_s$ be a shape. J is isomorphic to $H_i \circ \ldots \circ H_0$ for some sequence of homomorphisms $\{H_j\}_{0 \leq j \leq i}$, where*

(1) *$H_0 \colon \mathbb{A} \mapsto \mathbb{A}_0$ is surjective and $\mathbb{A}_0$ is a substructure of $\mathbb{A}$, or a contraction of a substructure of $\mathbb{A}$; and*

(2) *For each j with $1 \leq j \leq i$, $H_j \colon \mathbb{A}_{j-1} \mapsto \mathbb{A}_j$ is a contraction or an augmentation as in Theorem 5.2 or Theorem 5.3, Clauses 1, 2.*

Proof. We define two sequences of homomorphisms, namely $\{H_j\}_{0 \leq j \leq i}$ and $\{L_j\}_{0 \leq j \leq i}$, such that (1), (2) hold, and moreover, (3) $J = L_j \circ H_j \circ \ldots \circ H_0$, and (4) each L_j is nodewise injective and L_i is an isomorphism. (3) and (4) imply that J is isomorphic to the composition of the H_j.

By the definition of shape, if any composition $H_j \circ \ldots \circ H_0$ is realized, then we may take $j = i$ and stop. The nodewise injective L_j must be an isomorphism.

First, we define H_0 to prune unnecessary strands in $\mathbb{A}$, so that L_0 will be node injective. Partition the strands in $\mathbb{A}$ by their image under ϕ; i.e. $e_s = \{s' \colon \phi(s' \downarrow 1) = \phi(s \downarrow 1)\}$. For each partition element e_s, choose a representative $r(e_s)$ of maximal height. We know that α unifies all the terms on the strands in any partition element, so there is a most general contraction β compatible with these identifications. Enrich the ordering to reflect origination. Let $H_0 = [(\lambda s \,.\, r(e_s)), \beta]$.

Next, suppose that $H_0 \ldots H_j$ and $L_0 \ldots L_j$ have been defined, with $H_j \colon \mathbb{A}_{j-1} \mapsto \mathbb{A}_j$, and $\mathbb{A}_j$ is not realized. Let $L_j = [\phi_j, \beta_j] \colon \mathbb{A}_j \mapsto \mathbb{A}_s$. Let $n_1 \in \mathbb{A}_j$ be a negative node with $\mathsf{msg}(n_1)$ not penetrator derivable before n_1 in $\mathbb{A}_j$ (Prop. 5.6).

By Prop. 5.7, n_1 is either an unsolved incoming transformed node for some $\{|t|\}_K$ or else half of an unsolved outgoing transformed pair (n_0, n_1). In the latter case, we choose n_0 to be the point of origination of some $a \in \mathsf{unique}_{\mathbb{A}_j}$ such that $a \sqsubseteq \mathsf{msg}(n_1)$, and (n_0, n_1) is an outgoing transformed edge for a, S for some S. There are now the following possibilities.

(1) $\phi_j(n_1)$ is still unsolved, meaning that $K \cdot \beta_j$ (or some $K = K_1 \cdot \beta_j$ for some $K_1^{-1} \in \mathsf{used}(S)$) is compromised. Since n_1 is not penetrator-derivable in $\mathbb{A}_j$, some such K is not derivable.

    If for some S_1, $\mathbb{A}_s \setminus J_j(\mathbb{A}_j)$ contains an outgoing transforming edge for $K \cdot \beta_j, S_1 \cdot \beta_j$, then we augment $\mathbb{A}_j$ with a most general preimage of this edge. If required, first apply a contraction H_{j+1} to $\mathbb{A}_j$. Then the augmentation is H_{j+2}, and J_{j+2} is J_{j+1} together with this addition. Otherwise, the augmentation is H_{j+1}.

    If $\mathbb{A}_s$ contains no additional outgoing transforming edge for $K \cdot \beta_j$, then this K is already derivable in $\mathbb{A}_j$, contradicting the choice of K.

(2) Outgoing transformed edge $\phi_j(n_0), \phi_j(n_1)$:

    (a) $\phi_j(n_0), \phi_j(n_1)$ is no longer a transformed edge with respect to $a \cdot \beta_j, S \cdot \beta_j$. In this case, let H_{j+1} be a most general contraction with this property.

    (b) $\phi_j(n_0), \phi_j(n_1)$ is solved in $\mathbb{A}_s$. Select a transforming edge contained in $\mathbb{A}_s$. Let $m_0 \Rightarrow^+ m_1$ be a most general preimage of the outgoing transforming edge. If $m_0 \Rightarrow^+ m_1$ is not a transforming edge for (n_0, n_1) and a, S, then the reason is that $m_0 \Rightarrow^+ m_1$ is less general than (n_0, n_1). In this case, first contract $\mathbb{A}_j$. After contracting, one will at the next step augment with $m_0 \Rightarrow^+ m_1$. If contraction is not needed, augment $\mathbb{A}_j$ with $m_0 \Rightarrow^+ m_1$.

(3) $\phi_j(n_1)$ is a solved incoming test node in $\mathbb{A}_s$. Select a transforming node contained in $\mathbb{A}_s$. Let m_1 be a most general preimage of the incoming transforming node. If m_1 is not a transforming node for n_1, then the reason is that the transforming node m_1 is not as general as n_1. In this case, first contract $\mathbb{A}_j$. After contracting, one will at the next step augment with m_1. If contraction is not needed, augment $\mathbb{A}_j$ with m_1.

Thus, if $\mathbb{A}_j$ is realized, it is isomorphic to $\mathbb{A}_s$; otherwise, we extend $\{H_j\}, \{L_j\}$. □

A Pruning Condition. Some augmentations make progress toward realized skeletons, and other augmentations make no progress, because although they introduce a strand, that new strand is a redundant copy of an existing strand. We can prune away these augmentations, and ignore them when searching for shapes.

We say $\mathbb{A}'_0$ *augments* $\mathbb{A}_0$ *with a copy of* s if $\mathbb{A}'_0$ results from $\mathbb{A}_0$ by an augmentation with a strand s' such that: (1) $\mathsf{nodes}_{\mathbb{A}'_0} \setminus \mathsf{nodes}_{\mathbb{A}_0} = \{s' \downarrow j : j \leq i\}$ for some i; (2) there is an idempotent $I_0 = [\psi_0, \beta_0] : \mathbb{A}'_0 \mapsto \mathbb{A}_0$ with $\psi_0(s' \downarrow j) = s \downarrow j$.

Proposition 5.9. *Suppose* $\mathbb{A}'$ *augments* $\mathbb{A}$ *with a copy of* s*, namely* s'*. Let* $J' = [\phi', \alpha'] : \mathbb{A}' \mapsto \mathbb{A}'_s$ *with* $\mathbb{A}'_s$ *a shape. Then* $\phi'(s' \downarrow j) = \phi'(s \downarrow j)$*.*

Proof. Let $H'_i \circ \ldots \circ H'_0$ be the decomposition of J'. We may now construct a corresponding sequence $H_i \circ \ldots \circ H_0$ starting from $\mathbb{A}$, but using the identity in place of any steps H'_j such that the non-derivable node is not present in $\mathbb{A}_j$. For any node whose derivation uses positive nodes from s', we use the corresponding positive nodes in s. Thus, the target $\mathbb{A}_s$ of $H_i \circ \ldots \circ H_0$ is realized, and a substructure of $\mathbb{A}'_s$. By the definition, $H_i \circ \ldots \circ H_0 \circ I_0$ is a homomorphism from $\mathbb{A}'$ to $\mathbb{A}_s$. Since $\mathbb{A}_s$ is embedded in $\mathbb{A}'_s$, there's a node injective map from $\mathbb{A}_s$ to $\mathbb{A}'_s$. If this is not the identity, it contradicts $\mathbb{A}'_s$ being a shape. □

6. Conclusion

In this paper, we have developed the theory of skeletons and homomorphisms. We used it together with the strong form of the authentication tests (Props. 3.2–3.4) to establish search-oriented versions of the tests (Thms 5.2–5.3). Finally, we showed that these tests have a form of *completeness* (Thm 5.8). In [3] we describe how to use these ideas to mechanize protocol analysis. In future work we intend to study whether the search ideas themselves are still applicable, even when more interesting message algebras are substituted in place of the free algebra of the pure Dolev-Yao model (Section 2). These may include algebras with probabilistic structure.

References

[1] Martín Abadi and Bruno Blanchet. Analyzing security protocols with secrecy types and logic programs. *Journal of the ACM*, 52(1):102–146, January 2005.

[2] Roberto M. Amadio and Denis Lugiez. On the reachability problem in cryptographic protocols. In *Concur*, number 1877 in LNCS, pages 380–394, 2000.

[3] Shaddin F. Doghmi, Joshua D. Guttman, and F. Javier Thayer. Searching for shapes in cryptographic protocols. In *Tools and Algorithms for Construction and Analysis of Systems (TACAS)*, LNCS. Springer, March 2007. Extended version at URL:`http://eprint.iacr.org/2006/435`.

[4] Daniel Dolev and Andrew Yao. On the security of public-key protocols. *IEEE Transactions on Information Theory*, 29:198–208, 1983.

[5] Nancy Durgin, Patrick Lincoln, John Mitchell, and Andre Scedrov. Multiset rewriting and the complexity of bounded security protocols. *Journal of Computer Security*, 12(2):247–311, 2004. Initial version appeared in *Workshop on Formal Methods and Security Protocols*, 1999.

[6] Andrew D. Gordon and Alan Jeffrey. Types and effects for asymmetric cryptographic protocols. *Journal of Computer Security*, 12(3/4):435–484, 2003.

[7] Joshua D. Guttman and F. Javier Thayer. Authentication tests and the structure of bundles. *Theoretical Computer Science*, 283(2):333–380, June 2002.

[8] ITU. Message sequence chart (MSC). Recommendation Z.120, 1999.

[9] Leslie Lamport. Time, clocks and the ordering of events in a distributed system. *CACM*, 21(7):558–565, 1978.

[10] Gavin Lowe. Breaking and fixing the Needham-Schroeder public-key protocol using FDR. In *Proceeedings of* TACAS, volume 1055 of *Lecture Notes in Computer Science*, pages 147–166. Springer Verlag, 1996.

[11] Jonathan K. Millen and Vitaly Shmatikov. Constraint solving for bounded-process cryptographic protocol analysis. In *8th ACM Conference on Computer and Communications Security (CCS '01)*, pages 166–175. ACM, 2001.

[12] Roger Needham and Michael Schroeder. Using encryption for authentication in large networks of computers. *Communications of the ACM*, 21(12), 1978.

[13] Adrian Perrig and Dawn Xiaodong Song. Looking for diamonds in the desert: Extending automatic protocol generation to three-party authentication and key agreement protocols. In *Proceedings of the 13th IEEE Computer Security Foundations Workshop*. IEEE Computer Society Press, July 2000.