South American Journal of Logic
Vol. 4, n. 2, pp. 281–312, 2018
ISSN: 2446-6719

Dolev-Yao Multi-Agent Epistemic Logic

Mario R. F. Benevides, Luiz C. F. Fernandez and Anna C. C. M. de Oliveira

Abstract

In this work, we propose a new logic for reasoning about security protocols which extends multi-agent epistemic logic. This logic is inspired on the work of Dolev and Yao [8]. We introduce a new semantics based on structured propositions. Instead of building formulae from atomic propositions, they are built from expressions. The latter, are any piece of information that can appear in protocols: keys, messages, encrypted messages, agents and properties or some combination of this information in pairs. First, we propose this new semantics for our logic and provide an axiomatization for it. Second, we prove its soundness and completeness. Finally, we illustrate the use of our logic analyzing some well-known protocols.

Keywords: Dolev-Yao model, modal logic, epistemic logic, communication protocols.

1 Introduction

There are many approaches to formally verify authenticity and secrecy in communication protocols. For instance, we may have algebraic, probabilistic, logical approaches and so on. In this work we are most interested in logical approaches to verify authenticity and secrecy in communication protocols.

The Dolev and Yao work [8] was the first one to use a deductive approach to prove that a protocol could be broken by a malicious intruder. BAN logic [4] also uses a deductive approach to protocol verification but with more flavor of logic. After those works many more followed and the area has developed a lot.

Epistemic logics are logic to deal with the concepts like knowledge and believes. Many of theses logics have been tailored to be applied to computer science problems, like multi-agent epistemic logics [10]. They have a semantics based on Lamport model for distributed systems [15] and they can be used for protocol verification.

Some works extend Dolev-Yao model with some arithmetical theory in order to verify that a given protocol based on some encryption function expressed in this theory can be broken [6, 17].

There are many works in the area of access control logics. What they have in common, in general, are primitive operators like *says*, *speaks for* and some forms of delegation of authority. In [1], it is studied various possible axioms for these operators and they are formalized in higher order intuitionistic logic. Other line of research in this direction, are the ones that give a modal representation to these concepts. In [12, 13] a modal analysis of *says* and *speaks for* are provided, those approaches have the advantage of being based on decidable logics.

Other approaches use epistemic logic to reasoning about protocol specifications [5, 3, 14]. They are quite related to our proposed logic. The most important feature that differentiate our approach is the use of structured propositions, i.e., propositions have some inner structure and this is reflected in the semantics.

In this work we present a novel epistemic logic for reasoning about properties in protocols. It uses structure propositions, which is a new technique to deal with messages, keys and properties in security protocols in an uniform manner, keeping the logic propositional.

In Section 2, it is presented the necessary background concepts for the rest of the paper: multi-agent epistemic logic and Dolev-Yao model. Section 3 introduces the Dolev-Yao Epistemic Logic, illustrates its use with some examples and discuss the relationship between deductions in Dolev-Yao model and deductions in Dolev-Yao Epistemic Logic. In Section 4, we apply our approach to some well-known protocols and, in Section 5, we discuss some improvements and future works. Finally, Section 6 provides the final remarks. Appendices A and B show the proofs of soundness and completeness, respectively, for the Dolev-Yao Epistemic Logic.

2 Background

This section presents a brief overview of some topics in which our proposal is based on. First, we introduce the syntax and semantics of multi-agent epistemic logic. Then, we present the Dolev-Yao model [8].

2.1 Multi-agent epistemic logic

This section presents the multi-agent epistemic logic $\mathcal{S}5_a$. Using Kripke structure, the multi-agent approach allows us to represent knowledge and belief of

an agent or a group of agents. We can use it in many applications, such as puzzles, negotiations and protocols.

2.1.1 Language and semantics

The definitions below are based on [11, 21].

Definition 2.1 *The epistemic language consists of an enumerable set of propositional symbols Φ, a finite set of agents $\mathcal{A}$, the Boolean connectives $\neg$ and $\wedge$ and a modality K_a for each agent a. The formulae are defined as follows, represented in BNF-notation:*

$$\varphi ::= p \mid \top \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid K_a\varphi$$

where $p \in \Phi$, $a \in \mathcal{A}$.

$K_a\varphi$ is intended to mean that "agent a knows φ". We are considering the standard abbreviations and conventions: $\bot \equiv \neg\top$, $\varphi \vee \phi \equiv \neg(\neg\varphi \wedge \neg\phi)$, $\varphi \rightarrow \phi \equiv \neg(\varphi \wedge \neg\phi)$ and $B_a\varphi \equiv \neg K_a \neg\varphi$ ($B_a\varphi$ may be read as "agent a believes φ").

Definition 2.2 *A multi-agent epistemic* frame *is a tuple $\mathcal{F} = (S, R_a)$ where:*

- *S is a non-empty set of states;*
- *R_a is a binary relation over S, for each agent $a \in \mathcal{A}$.*

Definition 2.3 *A multi-agent epistemic* model *is a pair $\mathcal{M} = (\mathcal{F}, V)$, where $\mathcal{F}$ is a multi-agent epistemic frame and V is a valuation function $V : \Phi \rightarrow 2^S$. We call a rooted multi-agent epistemic model $(\mathcal{M}, s)$ an epistemic state.*

In most applications of multi-agent epistemic logic, the relations R_a are equivalence relations (reflexive, transitive and symmetric relations). In this work we only deal with the case where R_a are equivalence relations. We use the symbols $\sim_a$ for each agent a instead of R_a.

Definition 2.4 *Let $\mathcal{M} = \langle S, \sim_a, V\rangle$ be a multi-agent epistemic model. The notion of satisfaction $\mathcal{M}, s \models \varphi$ is defined as follows:*

- $\mathcal{M}, s \models p$ *iff* $s \in V(p)$
- $\mathcal{M}, s \models \neg\phi$ *iff* $\mathcal{M}, s \not\models \phi$
- $\mathcal{M}, s \models \phi \wedge \psi$ *iff* $\mathcal{M}, s \models \phi$ *and* $\mathcal{M}, s \models \psi$
- $\mathcal{M}, s \models K_a\phi$ *iff for all* $s' \in S : s \sim_a s' \Rightarrow \mathcal{M}, s' \models \phi$

2.1.2 Axiomatization

The axioms and inference rules (also called *derivation rules*) for $\mathcal{S}5$ are given below:

1. *All instantiations of propositional tautologies.*
2. $K_a(\varphi \to \psi) \to (K_a\varphi \to K_a\psi)$
3. $K_a\varphi \to \varphi$
4. $K_a\varphi \to K_aK_a\varphi$ [*positive introspection*]
5. $\neg K_a\varphi \to K_a\neg K_a\varphi$ [*negative introspection*]

Inference Rules

Modus Ponens $\varphi, \varphi \to \psi/\psi$ *Generalization* $\varphi/K_a\varphi$ *Substitution* $\varphi/\sigma\varphi$

where σ is a map uniformly substituting formulae for propositional variables.

Theorem 2.5 *$\mathcal{S}5_a$ is sound and complete with respect to its semantics.*

Proof. This proof is standard in multi-agent epistemic logics literature [11, 21]. ■

2.2 Dolev-Yao Model

The Dolev and Yao's article [8] is a seminal work for analyzing security protocols. They work with symmetric public key protocols and they considered a perfect encryption, i.e., the keys used are unbreakable.

2.2.1 Public key protocols

In this system, based on [7, 19], it is assumed that every user X in the network has an *encryption function* E_X, which generates a pair (X, E_X), inserted in a secure public directory, and a *decryption function* D_X, known only to user X. It is important to notice that the sender's public key is represented, in the message exchange, as a subscript of E. The main requirements on the functions above are:

- $D_X(E_X(M)) = M$;
- for any user Y, knowing $E_X(M)$ and the directory containing all the public pairs does not reveal anything about M.

2.2.2 Examples

To illustrate intruder's possible behaviours, let's consider the following examples.

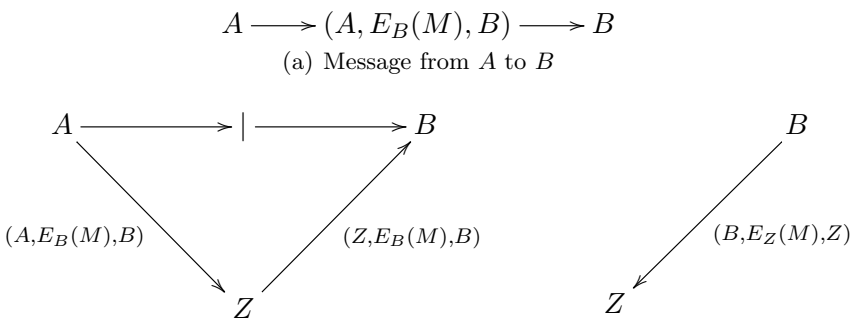
Example 2.6 *In this example, also called* Man-in-the Middle (MITM) attack, *the plaintext is encoded with an encryption function, where the receiver always replies using the sender's public key. Suppose user A wants to send a plaintext M to user B:*

a) A sends message $(A, E_B(M), B)$ *to B* [Figure 1(a)]*;*

b) Intruder Z intercepts the above message and sends message $(Z, E_B(M), B)$ *to B* [Figure 1(b)]*;*

c) B sends message $(B, E_Z(M), Z)$ *to Z* [Figure 1(c)]*;*

d) Z decodes $E_Z(M)$ *and obtains M.*



(a) Message from A to B

(b) Interception and message from Z to B

(c) Message from B to Z

Figure 1: Illustration of Example 2.6

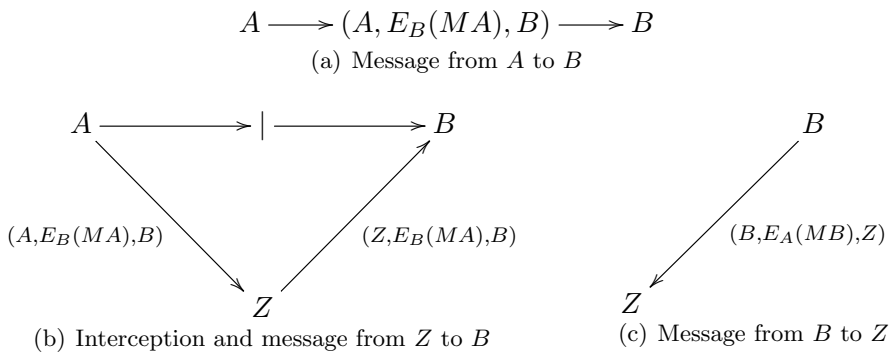
Example 2.7 *Now, the plaintext is encoded with the name of the sender, and the receiver uses the public key that corresponds to this user:*

a) A sends message $(A, E_B(MA), B)$ *to B* [Figure 2(a)]*;*

b) Z intercepts the above message and sends message $(Z, E_B(MA), B)$ *to B* [Figure 2(b)]*;*

c) B sends message $(B, E_A(MB), Z)$ *to Z* [Figure 2(c)]*;*

d) Z cannot *decode* $E_A(MB)$ *to obtain M.*



(a) Message from A to B

(b) Interception and message from Z to B

(c) Message from B to Z

Figure 2: Illustration of Example 2.7

2.3 Rules

The Dolev-Yao model can be seen as a deductive system. These rules are not presented in the original paper, but they can easily be obtained from the theory presented there. Consider a set of keys $\mathcal{K} = \{k_1, ...\}$ and an encripted message $\{M\}_k$, which represents a message M encrypted under the key k. An user can only decrypt an encrypted message $\{M\}_k$ if he knows the key k:

$$\textit{Reflexivity}\ \frac{M \in T}{T \vdash M}$$

$$\textit{Encryption}\ \frac{T \vdash M \quad T \vdash k}{T \vdash \{M\}_k} \qquad \textit{Decryption}\ \frac{T \vdash \{M\}_k \quad T \vdash k}{T \vdash M}$$

$$\textit{Pair-Composition}\ \frac{T \vdash M \quad T \vdash N}{T \vdash (M, N)}$$

$$\textit{Pair-Decomposition}\ \frac{T \vdash (M, N)}{T \vdash M} \qquad \frac{T \vdash (M, N)}{T \vdash N}$$

where T is a set containing all the messages and encrypted messages that the intruder has observed.

3 Dolev-Yao Multi-Agent Epistemic Logic

In Dolev-Yao model, the focus is on reasoning about an intruder. The idea is mapping possible actions and knowledge acquired by the agents. On the other

hand, epistemic logic can formally express what the agents know about the world.

This section presents the Dolev-Yao Multi-Agent Epistemic Logic $\mathcal{S}5_{DY}$. This logic is aimed to reasoning about knowledge in protocols, i.e, knowledge about: *keys*, *messages*, *encription/decription*, *concatenation*, *agents and groups of agents*.

3.1 Language and semantics

In the language of $\mathcal{S}5_{DY}$, formulae are built from expressions and not only from propositional symbols. Intuitively, an expression is any piece of information that can be encrypted, decrypted or concatenated in order to be communicated.

Definition 3.1 *The Dolev-Yao multi-agent epistemic language consists of an enumerable set of propositional symbols Φ, a finite set of agents $\mathcal{A}$, an enumerable set of keys $\mathcal{K} = \{k_1, \ldots\}$, the Boolean connectives $\neg$ and $\wedge$ and a modality K_a for each agent a. The expressions and formulae are defined as follows, represented in BNF-notation:*

$$E ::= p \mid k \mid (E_1, E_2) \mid \{E\}_k$$

where $k \in \mathcal{K}$ and $p \in \Phi$.

$$\varphi ::= m \mid \top \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid K_a\varphi$$

where $m \in E$ and $a \in \mathcal{A}$.

We are also considering the standard abbreviations and conventions as specified in Definition 2.1.

3.2 Semantics

Regarding the semantics, the definition of frame is the same as that in standard multi-agent epistemic logics. However, three restrictions were added to the valuation function, which we found necessary for the soundness proof developed later on.

Definition 3.2 *A Dolev-Yao multi-agent epistemic* model *is a pair $\mathcal{M} = \langle \mathcal{F}, V \rangle$, where $\mathcal{F}$ is a Dolev-Yao multi-agent epistemic frame and V is a valuation function $V : E \to 2^S$ satisfying the following conditions for all $m \in E$ and $k \in \mathcal{K}$:*

1. $V(m) \cap V(k) \subseteq V(\{m\}_k)$

2. $V(\{m\}_k) \cap V(k) \subseteq V(m)$
3. $V(m) \cap V(n) = V((m, n))$

The first condition ensures that, in any state, if we have a message m and a key k then we must be able to have the encrypted message $\{m\}_k$.

Condition 2 establishes that if we have an encrypted message $\{m\}_k$ and a key k then we must be able to decrypt it and obtain m.

Finally, the last one says that, in any state, we have messages m and n if and only if we have the pair (m, n).

The notion of satisfaction is similar to that defined for multi-agent epistemic logic in Definition 2.4. The only difference is for evaluation of expressions:

2. $\mathcal{M}, s \models m$ iff $s \in V(m)$, for $m \in E$

3.3 Axiomatization

The axiomatization for $\mathcal{S}5_{DY}$ is an extension of the one presented for multi-agent epistemic logic in Section 2.1.2, with three new axioms for encryption, decryption and pairing:

1. *All instantiations of propositional tautologies.*
2. $K_a(\varphi \to \psi) \to (K_a\varphi \to K_a\psi)$
3. $K_a\varphi \to \varphi$
4. $K_a\varphi \to K_aK_a\varphi$ [*positive introspection*]
5. $\neg K_a\varphi \to K_a\neg K_a\varphi$ [*negative introspection*]
6. $m \wedge k \to \{m\}_k$ [*encryption*]
7. $\{m\}_k \wedge k \to m$ [*decryption*]
8. $m \wedge n \leftrightarrow (m, n)$ [*pair composition & decomposition*]

Inference Rules

Modus Ponens $\varphi, \varphi \to \psi/\psi$ *Generalization* $\varphi/K_a\varphi$ *Substitution* $\varphi/\sigma\varphi$

where σ is a map uniformly substituting formulae for propositional variables.

Axioms 1, 2, 3, 4 and 5 are standard in multi-agent epistemic logics literature [11]. Axioms 6, 7 and 8 enforce the semantical properties of the valuation function (conditions 1, 2 and 3 of Definition 3.2). Axiom 6, ensures that, whenever we have a message m and a key k, then we must able to encrypt it and obtain the message $\{m\}_k$. Axiom 7, establishes that if we have an encrypted message $\{m\}_k$ and a key k, then we must be able to decrypt it and obtain m. Finally, axiom 8 says that, we have messages m and n if and only if we have the pair (m, n).

At this point, we are able to state the lemma that we will use repeatedly in our system.

Lemma 3.3 *The following formulae are theorems of* $\mathcal{S}5_{DY}$*:*

1. $K_a m \wedge K_a k \rightarrow K_a\{m\}_k$
2. $K_a\{m\}_k \wedge K_a k \rightarrow K_a m$
3. $K_a m \wedge K_a n \leftrightarrow K_a(m, n)$

Proof. This proof is straightforward from axioms 2, 6, 7, 8, inference rule *Generalization* and the fact that K_a distributes over conjunction: $\vdash K_a(\varphi \wedge \psi) \leftrightarrow (K_a\varphi \wedge K_a\psi)$. ■

Theorem 3.4 $\mathcal{S}5_{DY}$ *is sound and complete with respect to the class of* $\mathcal{S}5_{DY}$ *models.*

Proof. The soundness and completeness proof can be found in Appendix A and B. ■

3.4 Examples

Now, we revisit the examples of Section 2.2.2. The protocol actions, like *send* and *receive*, are represented in the metalanguage.

We have three agents, A, B and Z. In order to have a more economical notation, we use k_{AB}, k_{AZ} and k_{BZ} to denote the shared key between agents A, B and Z. Assuming that $k_{xy} = k_{yx}$ for every agent x and y, KB stands for *Knowledge Base* and *i.k.* for *initial knowledge* and *lem.* refers to Lemma 3.3.

Example 3.5 *Returning to Example 2.6, user A wants to send a message m to user B. The receiver always replies a message using the key shared with the sender:*

0. $KB_0 = \{K_A k_{AB},\ K_B k_{AB},\ K_B k_{BZ},\ K_Z k_{BZ},\ K_A m\}$ *i.k.*

$KB_0 \vdash K_A\{m\}_{k_{AB}}$ *lem. 1*

$\downarrow\ send_{AB}(\{m\}_{k_{AB}})$

$---$

$\downarrow\ Z\ intercepts$

1. $KB_1 := KB_0 \cup K_Z\{m\}_{k_{AB}}$

$\downarrow\ send_{ZB}(\{m\}_{k_{AB}})$

2. $KB_2 := KB_1 \cup K_B\{m\}_{k_{AB}}$

$KB_2 \vdash K_B m$ *lem. 2*

$KB_2 \vdash K_B\{m\}_{k_{ZB}}$ *lem. 1*

$\downarrow\ send_{BZ}(\{m\}_{k_{BZ}})$

3. $KB_3 := KB_2 \cup K_Z\{m\}_{k_{BZ}}$

$KB_3 \vdash K_Z m$ *lem. 2*

Intruder Z knows m.

Example 3.6 *Returning to Example 2.7, agent A also sends an encrypted message to agent B, but now the receiver always replies a message using the key shared with the indicated agent that is encrypted with the plaintext:*

0. $KB_0 = \{K_A k_{AB},\ K_B k_{AB},\ K_B k_{BZ},\ K_Z k_{BZ},\ K_A m\}$ *i.k.*

$KB_0 \vdash K_A(k_{AB}, m)$ *lem. 3*

$$KB_0 \vdash K_A\{(k_{AB}, m)\}_{k_{AB}} \qquad \textit{lem. 1}$$

$$\Big\downarrow {\scriptstyle send_{AB}(\{(k_{AB},m)\}_{k_{AB}})}$$

$$- - -$$

$$\Big\downarrow {\scriptstyle Z\ intercepts}$$

1. $KB_1 := KB_0 \cup K_Z\{(k_{AB}, m)\}_{k_{AB}}$

$$\Big\downarrow {\scriptstyle send_{ZB}(\{(k_{AB},m)\}_{k_{AB}})}$$

2. $KB_2 := KB_1 \cup K_B\{(k_{AB}, m)\}_{k_{AB}}$

$$KB_2 \vdash K_B(k_{AB}, m) \qquad \textit{lem. 2}$$

$$KB_2 \vdash K_B m \qquad \textit{lem. 3}$$

$$KB_2 \vdash K_B\{(k_{AB}, m)\}_{k_{AB}} \qquad \textit{lem. 1}$$

$$\Big\downarrow {\scriptstyle send_{BZ}(\{(k_{AB},m)\}_{k_{AB}})}$$

3. $KB_3 := KB_2 \cup K_Z\{(k_{AB}, m)\}_{k_{AB}}$

$$KB_3 \nvdash K_Z m$$

Intruder Z does not know m.

3.5 Relationship between $\mathcal{S}5_{DY}$ and Dolev-Yao model

In this section, we establish a relationship between deductions in Dolev-Yao model and deductions in $\mathcal{S}5_{DY}$. First, we define the notion of deduction in both systems. Second, we propose a translation from deductions in Dolev-Yao model into deductions in $\mathcal{S}5_{DY}$. Finally, we prove that for every set of expressions T and an expression m, if there exists a deduction of m from T in Dolev-Yao model, then there exists a deduction of m from T in $\mathcal{S}5_{DY}$.

3.5.1 Deduction in Dolev-Yao model and in $\mathcal{S}5_{DY}$

This section presents the definition of deduction in both systems. It is important to notice that a deduction in Dolev-Yao model is a sequence of sequents and in $\mathcal{S}5_{DY}$ is a sequence of formulae.

Definition 3.7 (Deduction in Dolev-Yao model) *Given a set of expressions $T = \{m_1, m_2, \ldots, m_n\}$ and an expression m:*

- *we call a pair $T \vdash m$ a* sequent*;*
- *the sequence of sequents $\langle S_1, \ldots, S_n \rangle$ is a* deduction *of m from T in Dolev-Yao model, $T \vdash_{DY} m$ iff $S_n = T \vdash m$ and each S_i $(1 \leq i < n)$:*
  1. *is $T \vdash m_i$ and $m_i \in T_i$; or*
  2. *is obtained by Dolev-Yao inference rules* Encryption, Decryption, Pair-Composition *and* Pair-Decomposition *applied to S_l and/or S_k and $l, k < i$.*
- *we define the* length *of the deduction $\pi = \langle S_1, \ldots, S_n \rangle$ as n, denoted as $|\pi| = n$.*

Deduction in $\mathcal{S}5_{DY}$ is defined as follows.

Definition 3.8 (Deduction in $\mathcal{S}5_{DY}$) *A formula α is said to be a* theorem *of a set of formulae Γ, $\Gamma \vdash_{\mathcal{S}5_{DY}} \alpha$ iff there exists a sequence of formulae $\langle \alpha_1, \ldots, \alpha_n \rangle$ such that $\alpha_n = \alpha$ and each α_i $(1 \leq i < n)$:*

1. *is an instance of the axioms; or*
2. *is obtained by* Modus Ponens, Generalization *or* Substitution *applied to α_l and/or α_k and $l, k < i$; or*
3. *is a member of Γ.*

The sequence of formulae $\langle \alpha_1, \ldots, \alpha_n \rangle$ is called a deduction *of α from Γ.*

3.5.2 Translation

First, we propose a map (translation) from deductions in Dolev-Yao model into deductions in $\mathcal{S}5_{DY}$.

Definition 3.9 *Let $\pi_n = \langle T \vdash m_1, \ldots, T \vdash m_n \rangle$ a deduction of m_n from T in Dolev-Yao model, $T \vdash_{DY} m_n$ of length n. We inductively define a map (translation) $(.)^t$ from deductions in Dolev-Yao model to deductions in $S5_{DY}$ as follows:*

- *$i = 1$: $\pi_1 = \langle T \vdash m_1 \rangle$, then $(\pi_1)^t = \langle m_1 \rangle$, $\quad (m_1 \in T)$;*
- *let $\pi_{i-1} = \langle T \vdash m_1, \ldots, T \vdash m_{i-1} \rangle$ and $(\pi_{i-1})^t$ its translation. Then we define the translation of π_i as follows:*

1. *if* $m_i \in T$, *then* $(\pi_i)^t = (\pi_{i-1})^t \bullet \langle m_i \rangle$, *where* $\bullet$ *is the sequence composition operator;*
2. *if* $T \vdash m_i$ *is obtained using inference rule:*
   (a) Encryption*: applied to some sequents* $T \vdash m_j$ *and* $T \vdash m_k$, *with* $1 \leq j, k < i$, *then*
   $(\pi_i)^t = (\pi_{i-1})^t \bullet \langle m_j \rightarrow (m_k \rightarrow (m_j \wedge m_k)) \bullet \langle m_k \rightarrow (m_j \wedge m_k) \rangle \bullet \langle (m_j \wedge m_k) \rangle \bullet \langle (m_j \wedge m_k) \rightarrow \{m_j\}_{m_k} \rangle \bullet \langle \{m_j\}_{m_k} \rangle$;
   (b) Decryption*: applied to some sequents* $T \vdash \{m_j\}_{m_k}$ *and* $T \vdash m_k$, *with* $1 \leq j, k < i$, *then*
   $(\pi_i)^t = (\pi_{i-1})^t \bullet \langle \{m_j\}_{m_k} \rightarrow (m_k \rightarrow (\{m_j\}_{m_k} \wedge m_k)) \bullet \langle m_k \rightarrow (\{m_j\}_{m_k} \wedge m_k) \rangle \bullet \langle (\{m_j\}_{m_k} \wedge m_k) \rangle \bullet \langle (\{m_j\}_{m_k} \wedge m_k) \rightarrow m_j \rangle \bullet \langle m_j \rangle$;
   (c) Pair-Composition*: applied to some sequents* $T \vdash m_j$ *and* $T \vdash m_k$, *with* $1 \leq j, k < i$, *then*
   $(\pi_i)^t = (\pi_{i-1})^t \bullet \langle m_j \rightarrow (m_k \rightarrow (m_j \wedge m_k)) \bullet \langle m_k \rightarrow (m_j \wedge m_k) \rangle \bullet \langle (m_j \wedge m_k) \rangle \bullet \langle (m_j \wedge m_k) \rightarrow (m_j, m_k) \rangle \bullet \langle (m_j, m_k) \rangle$;
   (d) Pair-Decomposition*: applied to some sequent* $T \vdash m_j$, *with* $1 \leq j < i$ *and* $m_j = (m_l, m_k)$, *then* $(\pi_i)^t = (\pi_{i-1})^t \bullet \langle (m_l, m_k) \rightarrow (m_l \wedge m_k) \rangle \bullet \langle (m_l \wedge m_k) \rangle \bullet \langle (m_l \wedge m_k) \rightarrow m_l \rangle \bullet \langle (m_l \wedge m_k) \rightarrow m_k \rangle \bullet \langle m_l \rangle \bullet \langle m_k \rangle$.

It is important to notice that, in each step of the translation, in order to obtain $(\pi_i)^t$, it is only added to $(\pi_{i-1})^t$ instance of the axioms or the conclusion of the application of *Modus Ponens*.

The following corollary asserts that every expression that appears in right hand side of any sequent in a deduction in Dolev-Yao model also occurs in the translated deduction in $\mathcal{S}5_{DY}$.

Corollary 3.10 *Let* $\pi = \langle S_1, \ldots, S_n \rangle$ *be a deduction in Dolev-Yao model and* $S_i = T \vdash m_i$ $(1 \leq i \leq n)$. *Then,* m_i *occurs in* $(\pi)^t$.

Proof. This proof follows straightforward from Definition 3.9 (translation), because in all cases $(\pi_i)^t = (\pi_{i-1})^t \bullet \ldots \bullet \langle m_i \rangle$. Thus, m_i occurs in $(\pi_i)^t$ and, consequently, m_i occurs in $(\pi)^t$.

■

We are ready to enunciate the main theorem of this section. It states that if an expression m has a deduction π from a set of expressions T in Dolev-Yao model, then there exists a deduction $(\pi)^t$ of m from T in $S5_{DY}$.

Theorem 3.11 *Let* π *be a deduction of* m *from* T *in Dolev-Yao model,* $T \vdash_{DY} m$. *Then,* $(\pi)^t$ *is a deduction of* m *from* T *in* $\mathcal{S}5_{DY}$, $T \vdash_{\mathcal{S}5_{DY}} m$.

Proof. By induction on the length of π.

Base case: $|\pi| = 1$, then $\pi = \langle T \vdash m \rangle$, where it must be the case that $m \in T$, and so $(\pi)^t = \langle m \rangle$, which is trivially a deduction of m from T in $\mathcal{S}5_{DY}$.

Induction hypothesis: suppose it holds for deductions π such that $|\pi| < i$.

Suppose we have a deduction π of m from T with length $|\pi| = i$.

Let $\pi_{i-1} = \langle T \vdash m_1, \ldots, T \vdash m_{i-1} \rangle$. By the induction hypothesis, $(\pi_{i-1})^t$ is a deduction of m_{i-1} from T in $\mathcal{S}5_{DY}$. We have to prove that $(\pi_i)^t$ is also a deduction in $\mathcal{S}5_{DY}$. We have five cases, one for each Dolev-Yao inference rule:

- $m \in T$: then $(\pi_i)^t = (\pi_{i-1})^t \bullet \langle m \rangle$ which is trivially a deduction of m from T in $\mathcal{S}5_{DY}$;
- m is obtained by the inference rule *Encryption* applied to sequents $T \vdash m_j$ and $T \vdash m_k$, with $1 \leq j, k < i$: so $m = \{m_j\}_{m_k}$.

  By Definition 3.9 (*2 (a)*),

  $(\pi_i)^t = (\pi_{i-1})^t \bullet \langle m_j \to (m_k \to (m_j \wedge m_k)) \bullet \langle m_k \to (m_j \wedge m_k) \rangle \bullet \langle (m_j \wedge m_k) \rangle \bullet \langle (m_j \wedge m_k) \to \{m_j\}_{m_k} \rangle \bullet \langle \{m_j\}_{m_k} \rangle$.

  As $(\pi_{i-1})^t$ is a deduction in $\mathcal{S}5_{DY}$, by Corollary 3.10, m_j and m_k occurs in $(\pi_{i-1})^t$. Applying *Modus Ponens* four times in the last part of $(\pi_i)^t$ we obtain $\{m_j\}_{m_k}$. Thus, as $(\pi_{i-1})^t$ is a deduction in $\mathcal{S}5_{DY}$, so is $(\pi_i)^t$.
- the cases for m obtained by the inference rules *Decryption*, *Pair-Composition* and *Pair-Decomposition* are analogous to the previous case.

■

It is worth to notice that $\mathcal{S}5_{DY}$ is more expressive than Dolev-Yao model. The former has all the booleans connectives and the modal epistemic operators. This makes possible not only expressing boolean combination of properties but also describing epistemic properties that agents knows and believes. It allows for expressing properties like "if the intruder Z knows the shared key k_{AB} and it believes that the content of message m contains some important information, then it can start the attack". For instance, suppose, agent Z has the following booleans variables:

- I_m (message m contains some important information);
- A (Z must start the attack);

- k_{AB} (shared key between agents A and B).
Then the above property could be expressed in $\mathcal{S}5_{DY}$ as

$$(K_Z k_{AB} \wedge B_Z I_m) \rightarrow A^1$$

4 Applications

This section analyses two well-known protocols, which were investigated in [4], using our proposed logic $\mathcal{S}5_{DY}$. First, we deal with Kerberos protocol [18, 16], which is used to provide a shared key between two users. We show that an intruder cannot know this shared key. Finally, we show that the Andrew Secure RPC protocol [20], which can be used when an user wants to refresh his key, is breakable by a malicious user.

It is important to notice that, in the following applications, we use agents names as propositions.

4.1 Kerberos Protocol

Based on [18], the Kerberos protocol was developed for Project Athena at MIT. It is used to provide a shared key between two users when a server is requested to do so, using timestamps.

Considering two users A and B, an authentication server S (also treated as an agent), T_x as the timestamp generated by agent x and the lifetime L, we can represent this protocol by the following steps (assuming that every user already has a shared key with the server):

1. A wants to communicate with B, so A sends a message to S stating it;
2. S replies to A, with an encrypted message containing T_S, L, k_{AB} and an encrypted message that only B can read (since it was encrypted under k_{BS}), which also contains the timestamp, the lifetime, and the shared key requested (this message is also called *ticket*);
3. A sends the ticket to B together with a timestamp encrypted under k_{AB};
4. B receives the first message sent by S and then can check T_S and L. If it has been created recently, B uses the k_{AB} to decrypt the second message sent by A. Then, B can take the communication from there, using T_A.

Supposing that an intruder Z intercepts the message sent from A to B and he already got from S what is necessary to communicate with B. Let's analyze this protocol using $\mathcal{S}5_{DY}$:

[1]Belief operator is defined as the dual of the knowledge operator $B_Z\phi \equiv \neg K_Z \neg \phi$.

0. $$KB_0 = \{K_A A, K_A B, K_A k_{AS}, K_A T_A, K_B k_{BS},$$

$$K_B T_B, K_S T_S, K_S L, K_S k_{AB}, K_S k_{BS}, K_Z T_Z, \qquad \textit{i.k.}$$

$$K_Z k_{ZB}, K_Z\{(T_S, L', k_{ZB}, Z)\}_{k_{BS}}\}$$

$$KB_0 \vdash K_A(A, B) \qquad \textit{lem. 3}$$

$$\downarrow\ send_{AS}((A,B))$$

1. $$KB_1 := KB_0 \cup K_S(A, B)$$

$$KB_1 \vdash K_S A \qquad \textit{lem. 3}$$

$$KB_1 \vdash K_S B \qquad \textit{lem. 3}$$

$$KB_1 \vdash K_S(T_S, L, k_{AB}, A) \qquad \textit{lem. 3}$$

$$KB_1 \vdash K_S\{(T_S, L, k_{AB}, A)\}_{k_{BS}} \qquad \textit{lem. 1}$$

$$KB_1 \vdash K_S(T_S, L, k_{AB}, B, \{(T_S, L, k_{AB}, A)\}_{k_{BS}}) \qquad \textit{lem. 3}$$

$$KB_1 \vdash K_S\{(T_S, L, k_{AB}, B, \{(T_S, L, k_{AB}, A)\}_{k_{BS}})\}_{k_{AS}} \qquad \textit{lem. 1}$$

$$\downarrow\ send_{SA}(\{(T_S,L,k_{AB},B,\{(T_S,L,k_{AB},A)\}_{k_{BS}})\}_{k_{AS}})$$

2. $$KB_2 := KB_1 \cup K_A\{(T_S, L, k_{AB}, B, \{(T_S, L, k_{AB}, A)\}_{k_{BS}})\}_{k_{AS}}$$

$$KB_2 \vdash K_A(T_S, L, k_{AB}, B, \{(T_S, L, k_{AB}, A)\}_{k_{BS}}) \qquad \textit{lem. 2}$$

$$KB_2 \vdash K_A\{(T_S, L, k_{AB}, A)\}_{k_{BS}} \qquad \textit{lem. 3}$$

$$KB_2 \vdash K_A k_{AB} \qquad \textit{lem. 3}$$

$$KB_2 \vdash K_A(A, T_A) \qquad \textit{lem. 3}$$

$$KB_2 \vdash K_A\{(A, T_A)\}_{k_{AB}} \qquad \textit{lem. 1}$$

$$KB_2 \vdash K_A(\{(T_S, L, k_{AB}, A)\}_{k_{BS}}, \{(A, T_A)\}_{k_{AB}}) \qquad \textit{lem. 3}$$

$\Big\downarrow$ $send_{AB}(\{\{(T_S,L,k_{AB},A)\}_{k_{BS}},\{(A,T_A)\}_{k_{AB}}\})$

$- - -$

$\Big\downarrow$ Z *intercepts*

3. $KB_3 := KB_2 \cup K_Z(\{(T_S, L, k_{AB}, A)\}_{k_{BS}}, \{(A, T_A)\}_{k_{AB}})$

Now Z has two possibilities. The first one is to send the same intercepted message to B:

3. $KB_3 := KB_2 \cup K_Z(\{(T_S, L, k_{AB}, A)\}_{k_{BS}}, \{(A, T_A)\}_{k_{AB}})$

$\Big\downarrow$ $send_{ZB}(\{\{(T_S,L,k_{AB},A)\}_{k_{BS}},\{(A,T_A)\}_{k_{AB}}\})$

4. $KB_4 := KB_3 \cup K_B(\{(T_S, L, k_{AB}, A)\}_{k_{BS}}, \{(A, T_A)\}_{k_{AB}})$

$$KB_4 \vdash K_B\{(T_S, L, k_{AB}, A)\}_{k_{BS}} \qquad \textit{lem. 3}$$

$$KB_4 \vdash K_B\{(A, T_A)\}_{k_{AB}} \qquad \textit{lem. 3}$$

$$KB_4 \vdash K_B(T_S, L, k_{AB}, A) \qquad \textit{lem. 2}$$

$$KB_4 \vdash K_B k_{AB} \qquad \textit{lem. 3}$$

$$KB_4 \vdash K_B(A, T_A) \qquad \textit{lem. 2}$$

$$KB_4 \vdash K_B T_A \qquad \textit{lem. 3}$$

$$KB_4 \vdash K_B\{T_A\}_{k_{AB}} \qquad \textit{lem. 1}$$

$$\downarrow\ send_{BZ}(\{T_A\}_{k_{AB}})$$

5. $KB_5 := KB_4 \cup K_Z\{T_A\}_{k_{AB}}$

$$KB_5 \nvdash K_Z T_A$$

Intruder Z does not know T_A.

Or he can send a concatenation of the ticket he previously got from S and the encrypted message:

3. $KB_3 := KB_2 \cup K_Z(\{(T_S, L, k_{AB}, A)\}_{k_{BS}}, \{(A, T_A)\}_{k_{AB}})$

$$KB_3 \vdash K_Z\{(A, T_A)\}_{k_{AB}} \qquad \textit{lem. 3}$$

$$KB_3 \vdash K_Z(\{(T_S, L', k_{ZB}, Z)\}_{k_{BS}}, \{(A, T_A)\}_{k_{AB}}) \qquad \textit{lem. 3}$$

$$\downarrow\ send_{ZB}((\{(T_S,L',k_{ZB},Z)\}_{k_{BS}},\{(Z,T_A)\}_{k_{AB}}))$$

4. $KB_4 := KB_3 \cup K_B(\{(T_S, L', k_{ZB}, Z)\}_{k_{BS}}, \{(Z, T_A)\}_{k_{AB}})$

$$KB_4 \vdash K_B\{(T_S, L', k_{ZB}, Z)\}_{k_{BS}} \qquad \textit{lem. 3}$$

$$KB_4 \vdash K_B\{(A, T_A)\}_{k_{AB}} \qquad \textit{lem. 3}$$

$$KB_4 \vdash K_B(T_S, L', k_{ZB}, Z) \qquad \textit{lem. 2}$$

$$KB_4 \nvdash K_B k_{AB}$$

$$KB_4 \nvdash K_B(T_A, A)$$

$$KB_4 \nvdash K_Z T_A$$

Since B is not able to continue the communication, Z cannot know T_A.

4.2 Andrew Secure RPC Handshake Protocol

The Andrew Secure RPC protocol [20] can be used when an user wants to refresh his key. So, in this scenario, let's consider that a handshake between user A and server S is made when a shared key k_{AS} already exists and A wants to obtain a new key k'_{AS}. We can represent this protocol by the following steps (assuming that *nonces* are "expressions invented for the purpose of being fresh" and "commonly include a timestamp or a number that is used only once" [4]):

1. A sends a nonce N_A encrypted with the key shared with S to state that he wants a new shared key;
2. S returns this nonce concatenated with N_S, also encrypted;
3. A returns only N_S to the server, encrypted under k_{AS};
4. after check the last message, S can send the new shared key k'_{AS} concatenated with N'_S, which "is an initial sequence number to be used in subsequent communication" [4], and encrypted with the first shared key.

Since there is no indication of who originated the third message, the server will reply this message using the key shared with the sender. Let's suppose that an intruder Z intercepts this message, we can also analyze this protocol:

$$0. \quad KB_0 = \{K_A k_{AS}, K_A k_{AZ}, K_A N_A, K_A A, K_S k_{AS},$$

$$K_S k_{ZS}, K_S N_S, K_S k'_{AS}, K_S N'_S, K_Z k_{AZ}, K_Z k_{ZS}\} \qquad \textit{i.k.}$$

$$KB_0 \vdash K_A(A, N_A) \qquad \textit{lem. 3}$$

$$KB_0 \vdash K_A\{(A, N_A)\}_{k_{AS}} \qquad \textit{lem. 1}$$

$$\Big\downarrow send_{AS}(\{(A,N_A)\}_{k_{AS}})$$

$$1. \quad KB_1 := KB_0 \cup K_S\{(A, N_A)\}_{k_{AS}}$$

$$KB_1 \vdash K_B(A, N_A) \qquad \textit{lem. 2}$$

$$KB_1 \vdash K_B N_A \qquad \textit{lem. 3}$$

$$KB_1 \vdash K_B(N_A, N_S) \qquad \textit{lem. 3}$$

$$KB_1 \vdash K_B\{(N_A, N_S)\}_{k_{AS}} \qquad \textit{lem. 1}$$

$$\Big\downarrow {\scriptstyle send_{SA}(\{(N_A,N_S)\}_{k_{AS}})}$$

2. $KB_2 := KB_1 \cup K_A\{(N_A, N_S)\}_{k_{AS}}$

$$KB_2 \vdash K_A(N_A, N_S) \qquad \textit{lem. 2}$$

$$KB_2 \vdash K_A N_S \qquad \textit{lem. 3}$$

$$KB_2 \vdash K_A\{N_S\}_{k_{AS}} \qquad \textit{lem. 1}$$

$$\Big\downarrow {\scriptstyle send_{AS}(\{(N_S)\}_{k_{AS}})}$$

$$---$$

$$\Big\downarrow {\scriptstyle Z\ intercepts}$$

3. $KB_3 := KB_2 \cup K_Z\{N_S\}_{k_{AS}}$

$$\Big\downarrow {\scriptstyle send_{ZB}(\{(N_S)\}_{k_{AS}})}$$

4. $KB_4 := KB_3 \cup K_S\{N_S\}_{k_{AS}}$

$$KB_4 \vdash K_S N_S \qquad \textit{lem. 2}$$

$$KB_4 \vdash K_S(k'_{AS}, N'_S) \qquad \textit{lem. 3}$$

$$KB_4 \vdash K_S\{(k'_{AS}, N'_S)\}_{k_{ZS}} \qquad \textit{lem. 1}$$

$$\Big\downarrow \; send_{SZ}(\{(k'_{AS}, N'_S)\}_{k_{ZS}})$$

5. $KB_5 := KB_4 \cup K_Z\{(k'_{AS}, N'_S)\}_{k_{ZS}}$

$$KB_5 \vdash K_Z(k'_{AS}, N'_S) \qquad \textit{lem. 2}$$

$$KB_5 \vdash K_Z k'_{AS} \qquad \textit{lem. 3}$$

$$KB_5 \vdash K_Z N'_S \qquad \textit{lem. 3}$$

$$KB_5 \vdash K_Z\{(k'_{AS}, N'_S)\}_{k_{AZ}} \qquad \textit{lem. 1}$$

$$\Big\downarrow \; send_{ZA}(\{(k'_{AS}, N'_S)\}_{k_{AZ}})$$

6. $KB_6 := KB_5 \cup K_A\{(k'_{AS}, N'_S)\}_{k_{AZ}}$

$$KB_6 \vdash K_A(k'_{AS}, N'_S) \qquad \textit{lem. 2}$$

$$KB_6 \vdash K_A k'_{AS} \qquad \textit{lem. 3}$$

$$KB_6 \vdash K_A N'_S \qquad \textit{lem. 3}$$

Now, intruder Z is able to decrypt any message eventually sent by A or S and encrypted under k'_{AS}.

5 Future Work

There are many possible extensions of this work. In this section, we discuss five extension there we are already working on or planning to do.

5.1 Knowledge *de dicto* and knowledge *de re*

We can refer the knowledge operator of the logic defined in Section 3 as knowledge *de dicto*. The modal operator K_a is meant to capture the standard notion

of knowledge *de dicto* that an agent a has about piece of information. For instance, the sentence $K_A K_B m$ means that agent A knows (that it is the case that) agent B knows message m. What do we mean by an agent to know the message m? Does she know the content of the message or the message itself?

We extend the language with a new knowledge operator ($\breve{K}_a$) in order to capture the notion of knowledge *de re* [14]. The intuition behind the proposition $\breve{K}_a m$ is that agent a *knows the content of* message m. For instance, the formula $K_A \breve{K}_B m$ expresses the fact that agent A knows that agent B *knows the content* of message m.

Let Φ be an enumerable set of propositional symbols, let $\mathcal{A}$ denote a finite set of agents, and let $\mathcal{K}$ be an enumerable set of keys. We use $p, q, \ldots$ as meta-variables to denote propositions. With a we denote an agent and with k a given key. We shall use k_a to refer to a's key.

5.1.1 Language

The language is an extension of Dolev-Yao Multi-Agent Epistemic Language, defined in Section 3, with this new knowledge *de re* $\breve{K}_a$.

$$E ::= p \mid k \mid (E_1, E_2) \mid \{E\}_k$$

where $k \in \mathcal{K}$ and $p \in \Phi$.

$$\varphi ::= m \mid \top \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid K_a\varphi \mid \breve{K}_a m$$

where $m \in E$ and $a \in \mathcal{A}$.

$$M ::= a \mid k \mid (M, M) \mid \{M\}_k$$

where $a \in \mathcal{A}$ and $k \in \mathcal{K}$. A message of the form (M_1, M_2) denotes the pair composition of messages M_1 and M_2, whereas $\{M\}_k$ is the encryption of message M with key k. N.B.: we allow an agent's name a in the definition of a message so that it can be appended to a message M to obtain the (signed) message (M, a), as it is used in some of the examples. With $\mathscr{M}$ we denote the set of all messages.

5.1.2 Proposal for an Axiomatization

1. *All instantiations of propositional tautologies.*
2. $\breve{K}_a a$ [*every agent knows its own name*]
3. $\breve{K}_a M \wedge \breve{K}_a M' \leftrightarrow \breve{K}_a (M, M')$ [*pair composition & decomposition*]
4. $\breve{K}_a M \wedge \breve{K}_a k \rightarrow \breve{K}_a \{M\}_k$ [*encryption*]

5. $\breve{K}_a\{M\}_k \wedge \breve{K}_a k \rightarrow \breve{K}_a M$ [*decryption*]
6. $K_a\alpha \wedge K_a(\alpha \rightarrow \beta) \rightarrow K_a\beta$
7. $K_a\alpha \rightarrow \alpha$
8. $K_a\alpha \rightarrow K_aK_a\alpha$
9. $\neg K_a\alpha \rightarrow K_a\neg K_a\alpha$
10. $\breve{K}_a M \rightarrow K_a\breve{K}_a M$ [*positive* de re *introspection*]
11. $\neg\breve{K}_a M \rightarrow K_a\neg\breve{K}_a M$ [*negative* de re *introspection*]
12. $\breve{K}_a M \rightarrow K_a M$
13. $\breve{K}_a M \rightarrow M$

These are some possible axioms. To complete this axiomatization is one of our future tasks. But we also need to propose a semantics for this new logic and prove its soundness, completeness, decidability and complexity. Moreover, we would like to investigate some applications of this new knowledge operator ($\breve{K}_a$).

5.2 Common knowledge

Much of the information in a protocol can be considered as common knowledge. In this work we intend to extend our logic with group operators like distributed knowledge and common knowledge. As far as we have done, this extension is quite standard if we do not have the knowledge *de re* operator.

5.3 Adding actions

Another possible extension would be to add actions in the sense of propositional dynamic logic. In all examples in this work, actions are performed in the meta level. It would be interesting to bring action to the object level and use the same language to reason about knowledge and actions.

5.4 Model checking

We would like to propose and implement algorithms for model checking formulae in security protocols. Also, we would like to contemplate all the extensions proposed in the sections above in this model checking project.

5.5 Adding Equational Reasoning

Some works extend Dolev-Yao model with some arithmetical theory in order to deal with situations where we do not have perfect cryptography [6, 17].

Probabilistic epistemic logic extends epistemic logic with some equations to deal with probabilities [9].

Finally, we would like to extend our axiomatic system with the possibility of make equational reasoning. By doing so, we will be able to deal with imperfect cryptography and express properties of more realistic security protocols.

6 Final remarks

In this work, we have presented a new epistemic logic for reasoning about security protocols. This logic introduces a new semantics based on structured propositions, i.e., they are any piece of information that can appear in protocols: keys, messages, agents and properties or some combination of this information in pairs, encrypted messages and so on.

We have proposed a new semantics and an axiomatization for this logic. And, we have proved its soundness and completeness.

We illustrate the use of our logic with two real protocols: Kerberos and Andrew Secure RPC Handshake.

Acknowledgements

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 and by the Brazilian Research Agency CNPq.

References

[1] Martín Abadi. Variations in access control logic. In *Proceedings of the 9th International Conference on Deontic Logic in Computer Science*, DEON '08, pages 96–109, Berlin, 2008. Springer-Verlag.

[2] Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*. Cambridge University Press, UK, 2001.

[3] Ioana Boureanu, Mika Cohen, and Alessio Lomuscio. Automatic verification of temporal-epistemic properties of cryptographic protocols. *Journal of Applied Non-Classical Logics*, 19(4):463–487, 2009.

[4] Michael Burrows, Martín Abadi, and Roger Needham. A logic of authentication. *ACM Transactions on Computer Systems*, 8(1):18–36, 1990.

[5] Mika Cohen and Mads Dam. A complete axiomatization of knowledge and cryptography. In *22nd IEEE Symposium on Logic in Computer Science (LICS 2007), 10-12 July 2007, Wroclaw, Poland, Proceedings*, pages 77–88, 2007.

[6] Véronique Cortier, Steve Kremer, and Bogdan Warinschi. A survey of symbolic methods in computational analysis of cryptographic systems. *J. Autom. Reason.*, 46(3-4):225–259, April 2011.

[7] Whitfield Diffie and Martin Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22:664–654, 1976.

[8] Danny Dolev and Andrew C. Yao. On the security of public key protocols. *Information Theory, IEEE Transactions on*, 29(2):198–208, 1983.

[9] Ronald Fagin, Ronald Fagin, Ronald Fagin, and Joseph Y. Halpern. Reasoning about knowledge and probability. *J. ACM*, 41(2):340–367, March 1994.

[10] Ronald Fagin, Joseph Y. Halpern, and Y. Moses. *Reasoning about knowledge*. MIT Press, Cambridge, Massachusetts, 1995.

[11] Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Y. Vardi. *Reasoning About Knowledge*. MIT Press, Cambridge, Massachusetts, 1995.

[12] Deepak Garg and Martín Abadi. A modal deconstruction of access control logics. In Roberto Amadio, editor, *Foundations of Software Science and Computational Structures*, pages 216–230, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[13] Valerio Genovese, Daniele Rispoli, Dov M. Gabbay, and Leendert W. N. van der Torre. Modal access control logic - axiomatization, semantics and fol theorem proving. In Thomas Ågotnes, editor, *STAIRS*, volume 222 of *Frontiers in Artificial Intelligence and Applications*, pages 114–126. IOS Press, 2010.

[14] S. Kramer. Cryptographic Protocol Logic: Satisfaction for (timed) Dolev-Yao cryptography. *Journal of Logic and Algebraic Programming*, 77(1–2), 2008. `http://dx.doi.org/10.1016/j.jlap.2008.05.005`.

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

[16] S. P. Miller, B. C. Neuman, J. I. Schiller, and J. H. Saltzer. Kerberos authentication and authorization system. In *In Project Athena Technical Plan*, 1987.

[17] Daniele Nantes Sobrinho. *O problema da dedução do intruso para teorias AC-convergentes localmente estáveis.* PhD thesis, Universidade de Brasília, Brasília, 2013.

[18] Roger M. Needham and Michael D. Schroeder. Using encryption for authentication in large networks of computers. *Commun. ACM*, 21(12):993–999, December 1978.

[19] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21:120–126, 1978.

[20] M. Satyanarayanan. Integrating security in a large distributed system. *ACM Trans. Comput. Syst.*, 7(3):247–280, August 1989.

[21] Hans van Ditmarsch, Wiebe van der Hoek, and Barteld Kooi. *Dynamic Epistemic Logic.* Synthese Library Series, volume 337. Springer, The Netherland, 2008.

Mario R. F. Benevides
Department of Computer Science (Institute of Mathematics) and Systems Engineering and Computer cience Program (Coppe)
Federal University of Rio de Janeiro (UFRJ)
Rio de Janeiro, RJ, Brazil
E-mail: mario@cos.ufrj.br

Luiz C. F. Fernandez
Systems Engineering and Computer Science Program (Coppe)
Federal University of Rio de Janeiro (UFRJ)
Rio de Janeiro, RJ, Brazil
E-mail: lcfernandez@cos.ufrj.br

Anna C. C. M. de Oliveira
Systems Engineering and Computer Science Program (Coppe)
Federal University of Rio de Janeiro (UFRJ)
Rio de Janeiro, RJ, Brazil
E-mail: acoliveira@cos.ufrj.br

Appendices

A Soundness

We only prove the soundness of axioms 6, 7, 8. The others axioms and inference rules are standard in multi-agent epistemic logics and are well-known to be sound for the class of $\mathcal{S}5_a$ models.

Lemma A.1 *The following axioms are sound with respect to the class of $\mathcal{S}5_{DY}$ models:*

1. $m \wedge k \rightarrow \{m\}_k$ [*encryption*]
2. $\{m\}_k \wedge k \rightarrow m$ [*decryption*]
3. $m \wedge n \leftrightarrow (m, n)$ [*pair composition & decomposition*]

Proof.

1. suppose we have a model $\mathcal{M}$ and a state s such that

$$\mathcal{M}, s \Vdash m \wedge k$$

Then we have that $\mathcal{M}, s \Vdash m$ and $\mathcal{M}, s \Vdash k$.
But this is if and only if $s \in V(m)$ and $s \in V(k)$.
By condition 1 of Definition 3.2, we have that $s \in V(\{m\}_k)$ and, thus, $\mathcal{M}, s \Vdash \{m\}_k$ and $\mathcal{M}, s \Vdash m \wedge k \rightarrow \{m\}_k$.

2 & 3. analogous to the above proof, but we use conditions 2 and 3 of Definition 3.2, respectively.

■

B Completeness

Now we prove the completeness of $\mathcal{S}5_{DY}$ by *Canonical Models*, based on [2]. First, we need some definitions:

Definition B.1 (Maximal Consistent Set) *Given a system $\mathcal{S}$ and a set of formulae Γ, we say:*

1. Γ *is* $\mathcal{S}$-inconsistent *if for some subset* $\{\alpha_1, \ldots, \alpha_n\} \subseteq \Gamma$ *we have*

$$\vdash_{\mathcal{S}} \neg(\alpha_1 \wedge \ldots \wedge \alpha_n)$$

and Γ *is* $\mathcal{S}$-consistent *if it is not* $\mathcal{S}$*-inconsistent;*

2. Γ *is* maximal *if for any formula* α*, either* $\alpha \in \Gamma$ *or* $\neg\alpha \in \Gamma$*;*

3. Γ *is* maximal $\mathcal{S}$-consistent *if it is both maximal and* $\mathcal{S}$*-consistent. In this case, we say that* Γ *is a* $\mathcal{S}$-MCS*.*

Next, we list and prove the MCS properties:

Proposition B.2 (MCS Properties) *Let* Γ *be a* $\mathcal{S}$*-MCS. Then for all formulae* ϕ *and* ψ*:*

1. *either* $\phi \in \Gamma$ *or* $\neg\phi \in \Gamma$*, but not both;*

2. Γ *is closed under Modus Ponens: if* $\phi \in \Gamma$ *and* $\phi \to \psi$ *then* $\psi \in \Gamma$*;*

3. $\phi \vee \psi \in \Gamma$ *iff either* $\phi \in \Gamma$ *or* $\psi \in \Gamma$*;*

4. $\phi \wedge \psi \in \Gamma$ *iff both* $\phi \in \Gamma$ *and* $\psi \in \Gamma$*;*

In particular, if Γ *is a* $\mathcal{S}5_{DY}$*-MCS then for all messages* m *and* $\{m\}_k$*, pair* (m, n) *and key* k*:*

5. *all theorems of* $\mathcal{S}5_{DY} \subseteq \Gamma$*;*

6. *if* $m \in \Gamma$ *and* $k \in \Gamma$ *then* $\{m\}_k \in \Gamma$*;*

7. *if* $\{m\}_k \in \Gamma$ *and* $k \in \Gamma$ *then* $m \in \Gamma$*;*

8. $(m, n) \in \Gamma$ *iff both* $m \in \Gamma$ *and* $n \in \Gamma$*.*

Proof.

1. by maximality, one of them must be in Γ;

2. suppose $\psi \notin \Gamma$, then $\{\phi, \phi \to \psi, \neg\psi\} \subseteq \Gamma$, which is an absurd because $\{\phi, \phi \to \psi, \neg\psi\}$ is $\mathcal{S}$-inconsistent. Therefore $\psi \in \Gamma$;

3 & 4. analogous to property 2;

5. for all theorems $\omega \in \mathcal{S}5_{DY}$, $\vdash_{\mathcal{S}5_{DY}} \omega$. Suppose $\neg\omega \in \Gamma$, as Γ is $\mathcal{S}5_{DY}$-consistent, $\Gamma \vdash_{\mathcal{S}5_{DY}} \neg\omega$, which is a contradiction. Then $\neg\omega \notin \Gamma$. By maximality, $\omega \in \Gamma$. Therefore all theorems of $\mathcal{S}5_{DY} \subseteq \Gamma$;

6, 7 & 8. follow straightforward from properties 2 and 5.

■

Now, our aim is to state and prove *Lindenbaum's Lemma*, which shows that any consistent set of formulae can be extended to a MCS:

Lemma B.3 (Lindenbaum's Lemma) *For any $\mathcal{S}$-consistent set Σ, there is a set Σ^+ such that:*

- $\Sigma \subseteq \Sigma^+$*; and*
- Σ^+ *is a $\mathcal{S}$-MCS.*

Proof. Let $\phi_0, \phi_1, \phi_2, \ldots$ be an enumeration of formulae of our language. We define the set Σ^+ as the union of a chain of $\mathcal{S}$-consistent sets as follows:

- $\Sigma_0 = \Sigma$; $\Sigma_{i+1} = \begin{cases} \Sigma_i \cup \{\phi_{i+1}\}, & \text{if it is } \mathcal{S}\text{-consistent} \\ \Sigma_i \cup \{\neg\phi_{i+1}\}, & \text{otherwise} \end{cases}$

  Claim: Σ_j is $\mathcal{S}$-consistent for any j. We prove that by induction on j.

  Base case: $\Sigma_0 = \Sigma$ is $\mathcal{S}$-consistent by hypothesis.

  Induction hypothesis: suppose that Σ_j is $\mathcal{S}$-consistent.

  Now, we want to show that Σ_{j+1} is also consistent. By construction, we have:

  $$\Sigma_{j+1} = \begin{cases} \Sigma_j \cup \{\phi_{j+1}\}, & \text{if it is } \mathcal{S}\text{-consistent} \\ \Sigma_j \cup \{\neg\phi_{j+1}\}, & \text{otherwise} \end{cases}$$

  By the above construction we have directly that Σ_{j+1} is also $\mathcal{S}$-consistent. Thus, Σ_i is $\mathcal{S}$-consistent for any i.

- $\Sigma^+ \cup_{i \geq 0} \Sigma_i$. Now we have to prove that Σ^+ is a $\mathcal{S}$-MCS.

  Σ^+ is $\mathcal{S}$-consistent. Because otherwise some finite subset of the set $\Sigma_i \subseteq \Sigma^+$ would be $\mathcal{S}$-inconsistent, but we just proved that any Σ_i is $\mathcal{S}$-consistent. Therefore, by item 1 of the definition of **Maximal Consistent Set** (Definition B.1), Σ^+ is $\mathcal{S}$-consistent.

  Σ^+ is maximal. Because given any formula ϕ, either $\phi \in \Sigma_j$ or $\neg\phi \in \Sigma_j$, for some j. Then $\Sigma_j \subseteq \Sigma^+$. So, Σ^+ is maximal.

  Therefore Σ^+ is a $\mathcal{S}$-MCS.

■

The Canonical Model for $\mathcal{S}$ is defined as follows:

Definition B.4 (Canonical Model) *The canonical model $\mathfrak{M}$ over $\mathcal{S}$ is the triple $\langle S^{\mathcal{S}}, \sim_a^{\mathcal{S}}, V^{\mathcal{S}}\rangle$, where:*

1. *$S^{\mathcal{S}}$ is the set of all $\mathcal{S}$-MCS;*
2. *$\sim_a^{\mathcal{S}}$ is the canonical* relation*, a binary relation on $S^{\mathcal{S}}$, for each agent $a \in \mathcal{A}$, defined by $s \sim_a^{\mathcal{S}} r$ if for all formula ψ, if $K_a\psi \in s$ then $\psi \in r$;*
3. *$V^{\mathcal{S}}$ is the canonical* valuation*, defined as $V^{\mathcal{S}}(e) = \{s \in S^{\mathcal{S}} \mid e \in s\}$, where $e \in E$.*

$\mathfrak{F} = (S^{\mathcal{S}}, \sim_a^{\mathcal{S}})$ is called the canonical frame*.*

Next, we prove the *Existence Lemma*, in order to prove later the *Truth Lemma*:

Lemma B.5 (Existence Lemma) *Let $\Gamma \in S^{\mathcal{S}}$ be a $\mathcal{S}$-MCS such that $B_a\phi \in \Gamma$. Then, there exists a $\mathcal{S}$-MCS Σ such that $\{\varphi \mid K_a\varphi \in \Gamma\} \cup \{\phi\} \subseteq \Sigma$.*

Proof. We first prove that $\Sigma^- = \{\varphi \mid K_a\varphi \in \Gamma\} \cup \{\phi\}$ is $\mathcal{S}$-consistent.

Suppose that Σ^- is $\mathcal{S}$-inconsistent. Then, there exists a finite subset $\varphi_1, \ldots, \varphi_n$ such that $\neg(\varphi_1 \wedge \cdots \wedge \varphi_n \wedge \phi)$ is a theorem:

$\vdash_{\mathcal{S}} \neg(\varphi_1 \wedge \ldots \wedge \varphi_n \wedge \phi)$

$\vdash_{\mathcal{S}} \varphi_1 \wedge \ldots \wedge \varphi_n \rightarrow \neg\phi$ [*propositional tautology*]

$\vdash_{\mathcal{S}} K_a(\varphi_1 \wedge \ldots \wedge \varphi_n \rightarrow \neg\phi)$ [*inference rule Generalization*]

$\vdash_{\mathcal{S}} K_a\varphi_1 \wedge \ldots \wedge K_a\varphi_n \rightarrow K_a\neg\phi$ [*axiom 2*]

By hypothesis, $K_a\varphi_1 \in \Gamma, \ldots, K_a\varphi_n \in \Gamma$, so, by property 2 of the **MCS Properties** (Proposition B.2), $K_a\neg\phi \in \Gamma$, and also, by duality and as Γ is $\mathcal{S}$-MCS, $\neg B_a\phi \in \Gamma$, which is a contradiction. Thus, Σ^- is $\mathcal{S}$-consistent. By **Lindenbaum's Lemma** (Lemma B.3), there exists a $\mathcal{S}$-MCS extension Σ that extends Σ^-. ■

Lemma B.6 (Truth Lemma) *For any formula ϕ, $\mathfrak{M}^{\mathcal{S}}, s \Vdash \phi$ iff $\phi \in s$.*

Proof. By induction on the length of ϕ.

Base case:

$$\mathfrak{M}^{\mathcal{S}}, s \Vdash e \text{ iff } s \in V^{\mathcal{S}}(e) \text{ iff } e \in s$$

Induction hypothesis: it holds for $|\phi| < i$:

$$\mathfrak{M}^{\mathcal{S}}, s \Vdash \phi \text{ iff } \phi \in s$$

Booleans: follows from the property 1 of the **MCS Properties** (Proposition B.2).

Knowledge operator:

$\Rightarrow$ Suppose

$$\mathfrak{M}^{\mathcal{S}}, s \Vdash K_a\phi \text{ (i)}$$

and $K_a\phi \notin s$. Thus, by maximality, we have that $B_a\neg\phi \in s$. So, by **Existence Lemma** (Lemma B.5) there exists a r such that

$$\{\varphi \mid K_a\varphi \in s\} \cup \neg\phi \subseteq r \text{ (ii)}$$

By definition of **Canonical Model** (Definition B.4) $s \sim_a^{\mathcal{S}} r$. From (i), for all s', if $s \sim_a^{\mathcal{S}} s'$ then

$$\mathfrak{M}^{\mathcal{S}}, s' \Vdash \phi$$

By the induction hypothesis, $\phi \in s'$ for all s' and in particular $\phi \in r$, which is a contradiction with (ii). Thus, $K_a\phi \in s$

$\Leftarrow$ Suppose $K_a\phi \in s$ and

$$\mathfrak{M}^{\mathcal{S}}, s \nVdash K_a\phi$$

then there exists a r such that $s \sim_a^{\mathcal{S}} r$ and

$$\mathfrak{M}^{\mathcal{S}}, r \Vdash \neg\phi$$

But by induction hypothesis, $\neg\phi \in r$. By **Canonical Model** (Definition B.4) if $s \sim_a^{\mathcal{S}} r$, for all formula ψ, if $K_a\psi \in s$ then $\psi \in r$. So, $\phi \in r$, which is a contradiction. Thus,

$$\mathfrak{M}^{\mathcal{S}}, s \Vdash K_a\phi$$

■

Lemma B.7 *The canonical model relations $\sim_a^{\mathcal{S}}$ are reflexive, transitive and symmetric.*

Proof. This follows from the definition of $\sim_a^{\mathcal{S}}$ and this proof can be found in epistemic and modal logics literature [2, 11, 21]. ■

Theorem B.8 *The canonical model $\mathfrak{M}^{\mathcal{S}5_{DY}}$ is a $\mathcal{S}5_{DY}$ model.*

Proof.

First we prove that $\mathfrak{M}^{\mathcal{S}5_{DY}}$ satisfies conditions 1, 2 and 3 of Definition 3.2:

- suppose we have $s \in V(m) \cap V(k)$ for a generic state $s \in S^{\mathcal{S}5_{DY}}$. So, we have that $s \in V(m)$ and $s \in V(k)$. Also,

$$\mathfrak{M}^{\mathcal{S}5_{DY}}, s \Vdash m$$

and

$$\mathfrak{M}^{\mathcal{S}5_{DY}}, s \Vdash k$$

which entails

$$\mathfrak{M}^{\mathcal{S}5_{DY}}, s \Vdash m \wedge k$$

As $S^{\mathcal{S}5_{DY}}$ is a $\mathcal{S}5_{DY}$-MCS, all the axioms of $\mathcal{S}5_{DY}$ are valid in s. Using axiom 6 and inference rule *Modus Ponens*, we have

$$\mathfrak{M}^{\mathcal{S}5_{DY}}, s \Vdash \{m\}_k$$

Therefore, by the **Truth Lemma** (Lemma B.6), we have that $\{m\}_k \in s$, that is, $s \in V(\{m\}_k)$. Thus, $V(m) \cap V(k) \subseteq V(\{m\}_k)$ (condition 1 of Definition 3.2).

- the proofs of conditions 2 and 3 of Definition 3.2 are analogous to the above proof, but we use axioms 7 and 8, respectively.

Together with Lemma B.7, we are done.

■

Theorem B.9 *Let Σ be a $\mathcal{S}5_{DY}$-consistent set of formulae. Then, Σ is satisfiable.*

Proof.

By **Existence Lemma** (Lemma B.5), there exists a $\mathcal{S}5_{DY}$-MCS Σ^+ such that $\Sigma \subseteq \Sigma^+$ and, by **Truth Lemma** (Lemma B.6), $\mathfrak{M}^{\mathcal{S}5_{DY}}, \Sigma^+ \models \Sigma$.

■