https://doi.org/10.31449/inf.v43i2.1617 Informatica **43** (2019) 161–175 **161**

On the Properties of Epistemic and Temporal Epistemic Logics of Authentication

Sharar Ahmadi and Mehran S. Fallah
Department of Computer Engineering and Information Technology
Amirkabir University of Technology (Tehran Polytechnic), Hafez Ave., Tehran, Iran
E-mail: sharar.ahmadi@aut.ac.ir, msfallah@aut.ac.ir

Massoud Pourmahdian
Department of Mathematics and Computer Science
AmirKabir University of Technology (Tehran Polytechnic), Hafez Ave., Tehran, Iran
E-mail: pourmahd@ipm.ir

Overview paper

Keywords: epistemic logic, temporal epistemic logic, formal verification, authentication protocol

Received: May 2, 2017

The authentication properties of a security protocol are specified based on the knowledge gained by the principals that exchange messages with respect to the steps of that protocol. As there are many successful attacks on authentication protocols, different formal systems, in particular epistemic and temporal epistemic logics, have been developed for analyzing such protocols. However, such logics may fail to detect some attacks. To promote the specification and verification power of these logics, researchers may try to construct them in such a way that they preserve some properties such as soundness, completeness, being omniscience-free, or expressiveness. The aim of this paper is to provide an overview of the epistemic and temporal epistemic logics which are applied in the analysis of authentication protocols to find out how far these logical properties may affect analyzing such protocols.

Povzetek: V preglednem prispevku je prestavljena epistemska in časovna epistemska logika overitvenega postopka z namenom izboljšave delovanja.

1 Introduction

The principals communicating in a network need to be assured that they are sending/receiving messages to/from the intended principals as otherwise an attacker may impersonate an authorized principal and gain access to confidential information. To prevent this, the principals use authentication protocols, which are built on cryptography, for exchanging messages [13]. Since there are many successful attacks on authentication protocols [47, 60, 49, 35, 37, 33], different formal sytems have been developed for analyzing such protocols. Many of these systems are logical and are known as logics of authentication [14, 8, 7, 36, 38].

The first formal system designated for the specification and verification of authentication protocols is an epistemic logic - called BAN [14]. Although BAN can safely verify some protocols, it does not verify some other ones successfully, e.g., it proved that the Needham-Schroeder Public Key protocol (NSPK for short) was secure but later it was shown that NSPK was vulnerable to man-in-the-middle attack [46]. To promote the verification power of BAN, some extensions of it have been developed [27, 3, 60, 62, 61, 17, 4]. Moreover, researchers have developed some other logics of authentication that are not BAN-like, but are inherited from standard logics. Many of these logics are epistemic and temporal epistemic ones that can model different runs of a protocol or can be applied to investigate the knowledge acquired by principals at different instants in protocol runs [16, 45, 50, 52, 8, 53]. For example, a principal may find out who originated a received message at specific step of a protocol run and may agree with the sender on the received information.

There are also dynamic epistemic logics that are useful for modeling knowledge protocols, which model higher-order information and uncertainties in terms of agents' knowledge about each other. However, since these logics are inconvenient in a cryptographic setting for generating equivalence relations among messages, we do not consider them in this paper [21].

Although the proposed epistemic and temporal epistemic logics have significantly improved the analysis of authentication protocols, every now and then a problem is found and we need to improve the logics to solve that problem. For example, an attack may be detected by an omniscience-free logic while it is ignored by another logic that is not omniscience-free. Similarly, an authentication protocol can be specified by a temporal epistemic logic while it cannot be specified by a logic whose modalities are only epis-

temic ones. Such issues encourage researchers to find out if logics of authentication should preserve specific logical properties. The properties that are usually discussed in this context are soundness, completeness, expressiveness, and being omniscience-free. Moreover, since a powerful attacker is traditionally modeled as the well-known Dolev-Yao message deduction system [24], it is valuable to see if these logics can model such a system. In this way, if a logic of authentication proves a security goal about an authentication protocol, one can trust that the result is indeed valid in the presence of a powerful attacker who can eavesdrop all communications, drop, manipulate and replay messages, and perform cryptographic operations using his known keys and messages.

The aim of this paper is not to compare epistemic logics of authentication to alternative security protocol analysis, such as applied pi calculus and other process calculi, strands, multiset and other forms of rewriting. The aim of this paper is to provide an overview of the epistemic and temporal epistemic logics of authentication to find out how far some of their logical properties such as soundness, completeness, being omniscience-free, and expressiveness may affect analyzing authentication protocols. To do so, we discuss not only the conditions under which these logics support the Dolev-Yao message deduction, but also the logical properties that encourage us to trust the derived judgements about the authentication protocols.

The rest of the paper is as follows: In Section 2, we provide an overview of the notions of cryptography, Kripke semantics, and epistemic logics of authentication. In Section 3, we compare epistemic and temporal epistemic logics of authentication and show how far some of their logical properties may affect them in analyzing authentication protocols. Section 4 concludes the paper.

2 Basic notions

Authentication protocols are rules built on cryptographic primitives that help principals authenticate each other while communicating in a hostile environment [13]. An authentication goal can be expressed in terms of a knowledge notion, e.g., the sender authentication can be read as "the receiver knows the sender of a received message". Consider the NSPK protocol shown in Figure 1. Every principal in this protocol has a public key and a private key such that the public key of any principal A is known to everyone but only A has the corresponding private key.

In this protocol, principal A generates a nonce n_a, pairs n_a with its name A, encrypts $n_a.A$ with principal B's public-key $pk(B)$ so that only B can decrypt it by his private key, and sends $\{n_a.A\}_{pk(B)}$ to B. By receiving this message, B decrypts it and sends n_a back along with his nonce n_b in an encrypted message so that only A can decrypt it. Then, A sends n_b back to B. The goal of the NSPK protocol is that both A and B can be assured that they are talking to each other and not to an attacker. BAN logic proved that the NSPK protocol was safe [14], whereas Lowe showed that it was vulnerable to the man-in-the-middle attack [46]. Although such a result seems confusing, it is suggested by the well-known fact that the NSPK protocol is safe assuming that no compliant initiator will ever select a non-compliant responder for a session. Needham and Schroeder assumed this fact about the principals. However, it was certainly no longer a reasonable assumption when cryptographic protocols were beginning to be used on the open internet and Lowe outlined the man-in-the-middle attack.

The man-in-the-middle attack, shown in Figure. 1, consists of two interleaved sessions of the NSPK protocol. After A initiates a protocol run with I, the intruder I extracts the message, impersonates A, and sends n_a to B. When B replies, I forwards this message to A and misuses A to obtain n_b. Then, I sends n_b back to B. Thus, B is deceived to believe that he is talking to A while he is in fact communicating with I. This attack shows that the result of analyzing the NSPK protocol using BAN logic is questionable. Since the original BAN did not have formal semantics, finding such a semantics that could model the above attack became an important topic of research.

As said earlier, the formal analysis of an authentication protocol using epistemic logics depends on the knowledge gained by the principals executing that protocol. There are two main ways to formalize such knowledge. Assume the statement: "B has sent m", where the underlying semantics of a logic of authentication interprets this statement as follows: "B is engaging in an event of a protocol sending message m". If we formalize this statement with a logical formula ϕ, A knows ϕ means: "A knows that B has sent m". This is called propositional knowledge which is implicit and does not care about the details of computation [58]. There is also algorithmic knowledge formalizing the exact models of principals' knowledge such that if a principal has some bit strings, he can apply cryptographic operators to compute more strings using some predefined algorithms [30]. In this paper, we consider both of these knowledge formalizations, but first we need to explain some primitive notions.

Assume that θ is a set of principals exchanging messages by executing an authentication protocol. We may use a logical language $\mathcal{L}$ to specify not only the steps of such a protocol, but also the intended authentication properties that we want to prove about that protocol. To do so, we need to formalize exchanged messages as message terms in $\mathcal{L}$ because protocols are a type of messages passing multi-agent systems [26]. A message may be a plain term c or a compound one constructed by encryption or pairing such that $\{m\}_k$ is the encryption of message m with the key k and $m.m'$ is the pairing of messages m and m'. There is a need for a derivation system to derive new messages from known ones using cryptographic functions. In this paper, we use the well-known Dolev-Yao message deduction system [24] as follows: $m.m'$ is a message if and only if both m and m' are messages. If $\{m\}_k$ and k are messages, then so is m.

NSPK protocol	**man-in-the-middle attack**
$A \to B : \{n_a.A\}_{pk(B)}$	1. $A \to I : \{n_a, A\}_{pk(I)}$
$B \to A : \{n_a.n_b\}_{pk(A)}$	1′. $I(A) \to B : \{n_a, A\}_{pk(B)}$
$A \to B : \{n_b\}_{pk(B)}$	2′. $B \to I(A) : \{n_a, n_b\}_{pk(A)}$
	2. $I \to A : \{n_a, n_b\}_{pk(A)}$
	3. $A \to I : \{n_b\}_{pk(I)}$
	3′. $I(A) \to B : \{n_b\}_{pk(B)}$

Figure 1: NSPK protocol and the man-in-the-middle attack

Finally, if m and k are messages, then so is $\{m\}_k$. Given a set of message terms τ and a finite set of Dolev-Yao message deduction rules σ, we say that m is derivable from τ if either $m \in \tau$ or m is derivable from τ by applying the rules in σ. Assuming a set of message terms τ, there are two interpretations for knowledge.

The first interpretation says that a principal i knows a formula ϕ if he is aware of ϕ and ϕ is true in all the worlds he considers possible. In this case, a set of formulas, denoted by $\mathcal{A}_i(w)$, is associated to every possible world w such that i is aware of every formula in $\mathcal{A}_i(w)$ [31]. The intuition behind such an interpretation is that a principal needs to be aware of a formula before he can know it. For instance, a principal i may be aware of an encrypted message $\{m\}_k$ that he receives without being aware of message m. In this way, i may know that he receives $\{m\}_k$ if this message holds in all the worlds that are possible to him while he may not know that he receives m. In the context of verifying security protocols, $\mathcal{A}_i(w)$ is implemented as an algorithm that says "YES" for the formulas that agent i is aware of in his local state in w. In this way, we say that i knows ϕ explicitly using algorithmic knowledge [45].

The second interpretation says that a principal i knows a formula ϕ implicitly, shown by an epistemic formula $K_i\phi$, if i knows that ϕ is true. The set $\mathcal{F}$ of $\mathcal{L}$-formulas then, comprises not only atomic formulas about sending or receiving messages, but also compound formulas built inductively as follows: For every $\phi, \psi \in \mathcal{F}$, $i \in \theta$, and $m \in \tau$, we have $\phi \wedge \psi$, $\neg\phi$, $K_i\ \phi$, and $\mathcal{A}_i\phi$ are in $\mathcal{F}$.

The authentication protocols and goals formalized by formulas in $\mathcal{F}$ need to be interpreted in a proper formal semantics. Since an authentication protocol can be seen as a multi-agent system and it is known that an interpreted system [1] (IS for short) is a standard semantics for a multi-agent system, authentication protocols can be modeled by interpreted systems too. This can build a foundation for constructing Kripke semantics for epistemic logics of authentication as follows [26].

A Kripke model of an epistemic logic of authentication can reflect an authentication protocol. Such a model has a set of possible worlds that can be defined as $W = R \times \mathbb{N}$, where R is the set of all runs of that protocol and $\mathbb{N}$ is the set of natural numbers. Thus, a pair $\langle r, n\rangle$ - called a point - represents a run r at a time instant t. Such a point can be associated to a set of formulas that hold (are true) in that point. A Kripke model is then a tuple of the form $\mathcal{M} = \big(W, \{\sim_i\}_{i\in\mathcal{A}}, \pi\big)$ where W is the set of all possible points of the protocol. Moreover, the accessibility relation $\sim_i$ can be interpreted in different ways.

In one interpretation, for every $w_1, w_2 \in W$, $w_1 \sim_i w_2$ holds if and only if the local states of a principal i are the same in w_1 and w_2. For example, the local states of B at the end of both runs of the NSPK protocol shown in Figure. 1 are the same because B sends and receives the same messages by completing the execution of these two runs. In another interpretation, for every $w_1, w_2 \in W$, $w_1 \sim_i w_2$ holds if and only if the local states of a principal i are indistinguishable in w_1 and w_2. For example, assume that there is a protocol P such that $w_1 = \langle r_1, t_1\rangle$ and $w_2 = \langle r_2, t_2\rangle$ are two possible worlds of a Kripke model that reflects P. Principals A and B participate in two runs of P, denoted by r_1 and r_2, and formulas $A\ sends\ \{m\}_K$ and $A\ sends\ \{m'\}_{k'}$ hold in w_1 and w_2, respectively. Assume that B does not know the proper decryption keys to decrypt these messages, so he cannot distinguish formulas $A\ sends\ \{m\}_K$ and $A\ sends\ \{m'\}_{k'}$ because he sees $\{m\}_k$ and $\{m'\}_{k'}$ as two random messages. In this way, he considers both of the formulas the same. If all of the other formulas that hold in w_1 and w_2 are equal, B cannot distinguish between w_1 and w_2 even if $\{m\}_k \neq \{m'\}_{k'}$, i.e., we have: $w_1 \sim_B w_2$ [2]. In this model, $K_i\ \phi$ is true at $w \in W$ if ϕ is true at every $w' \in W$ that is accessible from w in A's view. Moreover, $\mathcal{A}_i\phi$ is true in $w \in W$ if ϕ can be computed by an awareness algorithm $\mathcal{A}_i$ in w. Such

[1] Assume that $\theta = \{i_1, \ldots, i_n, e\}$ is a set of principals such that "e" denotes a specific principal called the environment. For each $i \in \theta$, there is a finite set L_i of local states, a finite set a_i of local actions, and a local protocol $p_i : L_i \to 2^{a_i}$. The transition relation $t_i : L_i \times a_1 \times \ldots \times a_n \to L_i$ is then defined to return the next local state of i after all the principals perform their actions at the local state. Consider a set of global states $G \subseteq L_1 \times \ldots L_n \times L_e$, a set of joint actions $a = a_1 \times \ldots \times a_n \times a_e$, a joint protocol $p : (p_1, \ldots, p_n, p_e)$, and a global transition relation $t = (t_1, \ldots, t_n, t_e)$, which operates on global states by composing all local and environmental transition relations. An IS is then a tuple $\big(G, I_0, t, \{\sim_i\}_{i\in\mathcal{A}}, \pi\big)$, where G is the set of all global states accessible from any initial global state in I_0 via the transition relation t. For each $i \in \theta$, there is an accessibility relation $\sim_i \subseteq G \times G$ such that $g \sim_i g'$ if and only if $l_i(g) = l_i(g')$, where $l_i : G \to L_i(g)$ returns i's local state in the global state g, and $\pi : G \times Atom \to \{true, false\}$ is an interpretation function [26].

[2] There are also some other interpretations for the accessibility relation. We refer the interested reader to Ref. [17, 8].

an algorithm is defined specifically for every protocol and for computing intended formulas [30]. The truth of other non-atomic formulas is defined in a standard way and the atomic formulas are interpreted by the interpretation function π [15].

Assume that $\mathcal{M} = \big(W, \{\sim_i\}_{i\in\mathcal{A}}, \pi\big)$ is a Kripke model that models a protocol P, and $AuthR$ is an authentication requirement formalized by a logical formula ϕ. We say that ϕ is satisfiable with respect to $\mathcal{M}$ when there is a $w \in W$ such that ϕ is true in w, i.e., $AuthR$ holds in a run of P that is associated to w. We say that ϕ is valid with respect to $\mathcal{M}$ if ϕ is true in every $w \in W$ i.e. $AuthR$ holds in all runs of P. We discuss authentication and formalizing authentication in more detail below.

2.1 Formalizing authentication

Most of the security protocols have been designated for attaining authentication i.e. one principal should be assured of the identity of another principal. A protocol designer may assign different roles such as initiator, responder, or server to principals. Authentication protocols can be classified into two categories with respect to these roles: the protocols that try to authenticate a responder B to an initiator A, and the protocols that try to authenticate an initiator A to a responder B.

The notion of authentication does not have a clear consensus definition in the academic literature. However, the most clear and hierarchical definition for authentication has been devised by Lowe. In this definition, authentication requirements depend on the use to which the security protocol is put. These requirements can then be classified as aliveness, weak agreement, non-injective agreement, and agreement [46]. A protocol guarantees to a principal A "aliveness" of another principal B if the following condition holds: whenever the initiator A completes a run of the protocol, apparently with the responder B, then B has previously been running the protocol. Aliveness can be extended to "weak agreement" if B has previously been running the protocol with A. "Weak agreement" can be extended to non-injective agreement on a set of data items (where V is a set of free variables of the protocol) if B has previously been running the protocol with A, B was acting as responder in his run, and the two principals agreed on the values of all the variables in V. Weak agreement can be extended to "agreement" if each such a run of A corresponds to a unique run of B [46].

There are many attacks that occur due to parallel runs of a protocol [47]. The definition of weak agreement for authentication guarantees a one to one relationship between the runs of two principals as follows: a protocol authenticates a responder to an initiator, whenever a principal A starts j runs of the protocol as an initiator and l runs as a responder all in parallel; and completes $k \leq j$ runs of the protocol acting as initiator apparently with a responder B, then B has recently been running k runs acting as responder in parallel, apparently with A. Moreover, A protocol authenticates an initiator to a responder, whenever a principal B starts j runs of the protocol as a responder and l runs as an initiator, all in parallel; and completes $k \leq j$ runs of the protocol acting as responder, apparently with initiator A, then A has recently been running k runs acting as initiator in parallel, apparently with B [59]. In the following example, we explain the definition of agreement in more detail.

Example 2.1. *Consider the following challenge-response protocol that aims to authenticate an initiator A to a responder B, and to authenticate a responder B to an initiator A. In this protocol, k_{ab} is a shared key between A and B. Moreover, n_a and n_b are two nonces generated by A and B, respectively.*

$$A \to B : n_a$$
$$B \to A : \{n_a\}_{k_{ab}}.n_b$$
$$A \to B : \{n_b\}_{k_{ab}}$$

There is the following reflection attack on the protocol that consists of two sessions of the protocol executed in parallel. In this attack, B has the responder role and $I(A)$ denotes an intruder who impersonates A:

$$1.\ I(A) \to B : n_a$$
$$2.\ B \to I(A) : \{n_a\}_{k_{ab}}.n_b$$
$$1'.\ I(A) \to B : n_b$$
$$2'.\ B \to I(A) : \{n_b\}_{k_{ab}}.n'_b$$
$$3.\ I(A) \to B : \{n_b\}_{k_{ab}}$$

B starts two runs of the protocol as a responder to A, but it completes only one run (lines: 1, 2, and 3) with $I(A)$ while A does not participate in these runs. So, the protocol fails to aim the agreement requirement.

In the next example, we show how we can formalize an authentication requirement.

Example 2.2. *Consider the NSPK protocol, shown in Figure 1. We want to formalize the non-injective agreement authentication requirement. To do so, we use epistemic modalities as follows:*

$$K_B\ K_A\ msg\ n_a.n_b$$

This formula can be read as follows: "B knows that A knows the message $n_a.n_b$". If this formula can be proven for the NSPK protocol, then we say that the protocol guarantees "non-injective agreement" to B, where $\{n_a.n_b\}$ appears as the set of data items that the two principals agree on their value. Since B encrypts n_b with A's public-key and sends $\{n_a.n_b\}_{pk(A)}$ to A, whenever B receives a message containing n_b, he concludes that A has previously been running the protocol with B because A is the only principal who has A's private key to decrypt $\{n_a.n_b\}_{pk(A)}$ in order to extract n_b. The man-in-the-middle attack deceives B to believe that he is talking to A while he is in fact talking to I, who is an intruder. All BAN-like logics

proved the above formula for the NSPK protocol, whereas an omniscience-free epistemic BAN-like logic, that is referred to WS5 throughout this paper, could identify this insider attack [17]. In fact, being omniscience-free enabled WS5 to model the Dolev-Yao message deduction properly. We will explain this logic in detail at next sections.

3 Logical properties

In this section, we investigate how far some properties of epistemic and temporal epistemic logics of authentication may affect the analysis of authentication protocols. The properties that we investigate are soundness, completeness, being omniscience-free, and expressiveness.

3.1 Soundness and completeness

Beside syntax and semantics, every logic may have a proof system $\mathcal{X}$ consisting of some axioms and rules where the axioms are valid with respect to the logic's semantics and the rules preserve validity i.e. if the premise of a rule is valid, the result of it is also valid. Let $\mathcal{X}$ be a proof system of a logic of authentication that is based on the Dolev-Yao deduction system. Moreover, let the following statement be an authentication property: "principals i and j know that they are talking to each other", where i and j are engaging only in one session and both peers has received certain messages as common knowledge to authenticate each other. In this case, proving ϕ in $\mathcal{X}$ means that i and j know that they are indeed talking to each other even in an environment where there are attackers who can derive messages due to the Dolev-Yao deduction system.

The proof system $\mathcal{X}$ may have some interesting properties, two of which are soundness and completeness: $\mathcal{X}$ is sound if every derivable formula ϕ in $\mathcal{X}$ is also valid. $\mathcal{X}$ is complete if every valid formula ϕ is provable in $\mathcal{X}$. This is also called "weak completeness" by some researchers [15]. Logical analysis of security protocols relies on formal models of cryptography where cryptographic operations and security properties are defined as formal expressions. Such models ignore the details of encryption and focus on an abstract high-level specification and analysis of a system [1, 14, 24, 28].

Proving the soundness and completeness of a logic of authentication gives a strong intuition that the formal semantics of that logic is defined properly and it is working as expected. So, the logic can be applied safely in analyzing authentication protocols. For formal verification of a security protocol, there is a need for a formal model to reflect that protocol appropriately i.e. there is a need for a sound formal model for that protocol. Using a logical model, the verification is then dependent on the following parameters: first, the protocol must be described in the language of the logic. This description will be a part of a trust theory which consists of correct and acceptable propositions used in deducing security requirements. Even with a bad description of a protocol and its initial assumptions, the logic should consider all possible runs of that protocol.

To discuss the second parameter for defining a sound formal model, we first provide an overview of the Dolev-Yao indistinguishability relation. The intuitive idea for defining this relation is the fact that two messages are indistinguishable if any test - based on a limited set of operations on messages - gives the same result about the configuration of those messages. The Dolev-Yao indistinguishability relation can be related to cryptographic computing models. In this case, a formal model is said to be computationally sound. This is expressed for static equivalence, which is a general form of indistinguishability, explicitly: two local states are static equivalents if they satisfy the same equivalence tests. For a given theory of equation, static equivalence is based on a computable efficient set of operations such as symmetric and asymmetric encryption and decryption. For example, consider the simplest equivalence theory satisfying an equation of the form $dec(enc(m, pk), pr) = m$, where pk, pr, and m are a public key, a private key, and a message, respectively. Moreover, enc is an asymmetric encryption operator that encrypts m with pk and dec is an asymmetric decryption operator that decrypts an encrypted message with pr [2].

There is also another parameter for defining a sound formal model. Assume that there is a specification of an authentication protocol P and some initial assumptions using a logic of authentication L. We need to show whatever is deduced in L about P should be consistent with what the principals involved in executions of P actually infer. Assume that Γ is a finite set of logical formulas including the specification of P and its initial assumptions. Moreover, assume that the desired security goal is formalized by a formula ϕ that can be proven by applying the formuals in Γ and the axioms and rules of L's proof system. If L is logically sound and $\mathcal{M}$ is its model that satisfies the formulas in Γ, $\mathcal{M}$ also satisfies ϕ. If we show that $\mathcal{M}$ considers all possible runs of P, including those that attackers may participate in, the model reflects P properly.

As said earlier, the other theorem that is usually investigated for every logic is completeness. All valid formulas of a complete logic are also provable in its proof system. This motivates researchers to provide provers for the analysis of security protocols [22, 52, 51, 29]. If such a logic is also sound, the derived statements are more trusted since completeness shows that the formal semantics work as expected. Completeness may be a result of another property. As an example, the completeness of BAN-like logics has been an open problem for many years because some of them do not have any formal semantics and some other ones have inaccurate formal semantics. So, the logics could not model some possible runs executed by a Dolev-Yao attacker. However, it has been shown that the completeness of BAN-like logics can be proven by presenting a formal semantics that avoids logical omniscience [18]. We will discuss logical omniscience in more details later.

There is also another line of research that proves com-

pleteness for monadic fragments of a first-order temporal epistemic logic with respect to their corresponding classes of quantified interpreted systems. Such systems may have the following typical properties: synchronicity, perfect recall, no learning, and unique initial state. In contrast to most of the logics of authentication, such a logic has some axioms and rules that explore the relationship between its time and knowledge modalities [5, 6].

3.2 Logical omniscience

The semantics of a logic of authentication can be defined based on the standard Kripke structure. Such a semantics may lead to the logical omniscience problem where principals know all logical truths i.e. they know all consequences of what they know [31]. In fact, the problem bypasses the limitations placed on the knowledge of a principal who receives an encrypted message but does not have the right key to decrypt that message. Assume that L is an epistemic logic of authentication, which has a formal semantics built on the standard Kripke structure, and Γ is a set of logical formulas in L. Moreover, assume that a formula ϕ can be derived from Γ in L's proof system and an agent i knows all of the formulas in Γ. Then, the formal semantics of L leads to the logical omniscience where i knows ϕ. This fact is an immediate result of the interpretation of the knowledge modality of L with respect to the underlying standard Kripke semantics. In this way, a formula $K_i\phi$ is true in a state (possible world) w if and only if ϕ is true in every state that is indistinguishable (accessible) from w in i's view.

For example, assume that the following formula is true in all possible runs of a protocol,

$$i\ sent\ msg\ \{m\}_k \rightarrow submsg(m)$$

where $submsg(m)$ is read as m is a sub-message i.e. m is a sub-message of $\{m\}_k$. Using the standard Kripke semantics and for every principal j, the following formula is also true in all runs of that protocol:

$$K_j\ i\ sent\ msg\ \{m\}_k \rightarrow K_j\ submsg(m)$$

Now, assume that the anonymity of i fails and the formula $K_j\ i\ sent\ msg\ \{m\}_k$ is valid. Therefore, $K_j\ submsg(m)$ can be deduced by applying modus ponens. But in fact, this judgment is true only if j knows the symmetric key k to decrypt $\{m\}_k$. Thus, logical omniscience should be avoided in order to restrict principals' knowledge to what they can compute from their known facts, messages, and keys.

There are different approaches for solving the logical omniscience problem in the analysis of security protocols. In Ref. [19], the problem is solved by presenting a generalized Kripke semantics based on a permutation-based IS. Such a semantics results in a weakened necessitation rule for a logic faithful to BAN. Hence the logic becomes an omniscience-free weakened $S5$. Such a logic formalizes an implicit form of knowledge that results in abstract high-level reasoning of security protocols [17]. The logical omniscience problem can also be avoided by exact models of knowledge that a principal acquires during protocol runs. For example, such models are applied by a logic - called TDL [45]. So, a part of the logic that links epistemic modalities to awareness algorithms becomes omniscience-free. In this way, a principal knows a fact if he is aware of that fact. The idea of using awareness algorithms in formal security was originated in Ref. [30]. We will talk about the logics that use such algorithms at the end of this section.

3.3 Expressiveness

Epistemic logics of authentication usually have three different operators besides the standard ones of propositional logics. These operators are temporal, epistemic, and awareness. Hybrid systems may also have some other operators such as type operators or algebraic operators, but we do not consider hybrid systems in this paper and refer the interested reader to Ref. [39]. Temporal modalities formalize precedence of actions, time intervals, etc., such as "next" and "in a time interval $[t_1, t_2]$". Epistemic modalities formalize the knowledge of principals. Awareness operators show the algorithms that principals use to become aware of facts. The expressiveness of epistemic logics of authentication relies on their logical order and modalities. Moreover, if a logic has temporal operators, its expressiveness also relies on the method that the epistemic core is augmented by temporal modalities.

There are three approaches for adding temporal modalities to an epistemic core of a logic of authentication. The first approach, which makes the resulting temporal epistemic logic very expressive, is the fusion approach where epistemic and temporal modalities may appear in each other's scope without any restrictions. Moreover, the resulting logic may have axioms and rules that explore interactions between time and knowledge [5, 8, 23, 4]. This approach has been used in developing logics applied in analyzing a wide range of security protocols. Two examples of these protocols are classical authentication protocols such as NSPK and stream authentication protocols such as TESLA [57], which is used for sending streams of messages: videos, audios, etc.

The second approach is to use fibring technique where temporal and epistemic modalities may appear in each other's scope without any restrictions. But, the time and knowledge dimensions of a fibred logic are orthogonal. So, such a logic has no axioms and rules to explore the relationship between time and knowledge. Fibring technique does not model the knowledge which is obtained as a consequence of a particular event. Thus, a fibred logic is less intuitive for modeling security protocols and less expressive in comparison with other logics built on the fusion approach. However, theorems such as soundness and completeness may be easily proven for a fibred logic if its constituent logics are sound and complete. Moreover, a prover can be easily constructed for a fibred logic if its constituent logics have provers. Such a logic has been developed for

the analysis of the TESLA protocol [56].

Finally, the last approach for adding temporal modalities to an epistemic core of a logic of authentication is using the temporalization technique. This technique operates in a hierarchical way such that the temporal modalities can never appear in the scope of epistemic modalities i.e. the resulting logic does not have any formula of the form $K_i \bigcirc \phi$, which can be read as follows: agent i knows that at the next step ϕ holds. This approach has been used for verifying different protocols such as TESLA and WMF [53, 52].

In Figure 2, we compare some important epistemic and temporal epistemic logics of authentication against the above properties where every row of the figure is dedicated to a specific logic. The 1^{st} column of each row shows the logic name. The 2^{nd} column shows if the logic order is "propositional" or "first-order", denoted by "PR" and "FO", respectively. The 3^{rd} column shows the type of operators used in the logic where "E", "A", and "T" denote "epistemic", "awareness", and "temporal", in order. The 4^{th} column shows if the logic has a proof system, model checker, tableau .... The 5–7^{th} columns show if the logic is sound, complete, or omniscience-free, respectively. If a logic is sound, complete, or omniscience-free, we show this by a "✓" symbol. If any of these properties does not hold, we show this by a "×" symbol. If a logic does not have a proof system, it has no soundness and completeness theorems. In this case, we use a "−" symbol. Finally in the last column, "EXP" denotes that the explicit part of the logic is omniscience-free. Some of the security protocols analyzed by these logics are shown in Figure 3. The attacker models of these logics are also summarized in Figure 4. In what follows, we explain the above logical properties in more detail.

3.4 Inside the logics

In 1989 BAN logic was proposed as the first formal system for the specification and verification of authentication protocols [14]. This is a simple intuitive propositional epistemic logic named after its developers Burrows, Abadi, and Needham. The syntax of BAN consists of inference rules about principals' beliefs and their actions. This syntax enables BAN not only to specify the steps of an authentication protocol and its security goals, but also to derive the intended goals about that protocol. The first step of applying BAN is to idealize a protocol into an abstraction. In the second step, one should translate the initial assumptions and the security goals into BAN language, relate each idealized step of the protocol to a BAN formula, and then use BAN inference rules to derive intended goals.

The soundness and completeness theorems cannot be proven for BAN because it has no formal semantics. This logic has only epistemic modalities and it is not expressive enough to specify or verify highly time-dependent protocols such as stream authentication protocols [29]. This, along with BAN's propositional order, results in its low specification power. BAN has no formal attacker model but the capabilities of attackers are somehow embedded in its proof system.

For example, BAN has a rule - called the message-meaning rule - which has two premises. Assume that P and Q are two agents, m is a message, k is a key, and $\{m\}_k$ denotes that m is encrypted with k. The first premise of this rule says that P believes that k is a shared key between P and Q. This is formalized as follows: $P\ Believes\ P \leftrightarrow^k Q$. The second premise says that someone has sent a message which contains $\{m\}_k$ to P. This is shown as follows: $P\ sees\ \{m\}_k$. The conclusion of this rule is that P believes that at some time Q sent a message which contains m. This is formalized by the following formula: $P\ Believes\ Q\ said\ \{m\}_k$. As k is a shared key between P and Q, only these two agents can use k to encrypt m and no other agent can create $\{m\}_k$. Thus, when P receives $\{m\}_k$ it concludes that at some time Q has sent a message which contains $\{m\}_k$. In this way, even if an attacker has sent a message containing $\{m\}_k$ to P, P believes that this message has not been originated by Q [14]. As an example of formalizing authentication in BAN, we say that authentication is complete for the NSPK protocol if there are nonces n_a and n_b, generated by A and B, respectively, such that the following statements hold:

$$A\ Believes\ A \leftrightarrow^{n_a.n_b} B$$
$$B\ Believes\ A \leftrightarrow^{n_a.n_b} B.$$

The above formalizations can be classified as non-injective agreement. However, other weaker formalizations can be presented too.

As said earlier, BAN has some problems while verifying authentication protocols. Many extensions of this logic have been developed to resolve its problems. One of these extensions is GNY developed in 1990 for verifying a wider range of authentication protocols. GNY emphasizes separating the content and meaning of messages while it follows the same method as BAN for formalizing authentication. This logic is named after its developers Gong, Needham, and Yahalom. Although GNY can be applied in verifying a sample voting protocol successfully, the logic still suffers the same problems as its predecessor. Neither BAN nor GNY preserves the properties discussed in Subsections 3.1, 3.2, and 3.3. Thus, their derivations are not trustworthy. Moreover, these logics cannot analyze highly time-dependent protocols such as the TESLA protocol.

The first attempt for developing a formal semantics for BAN was in 1991 when Abadi and Tuttle improved the syntax and inference rules of BAN and also presented a formal semantics - called AT - for BAN [3]. This semantics is based on the standard Kripke structure constructed from interpreted systems. In this model, a principal i knows a formula ϕ if we have: "i knows ϕ" is true at a point $\langle r, k\rangle$ if it is true in every point $\langle r', k'\rangle$ that is indistinguishable from $\langle r, k\rangle$ in i's view. This extension of BAN - called AT logic - is sound with respect to the AT semantics. Thus, the proofs in this logic are more trustworthy. However, the completeness of AT remained an open problem for years

Logic	Order	Operators	Prover, Model Checker, ...	Sound	Ccomplete	O-free
BAN [14]	PR	E	proof system	–	–	×
GNY [27]	PR	E	proof system	–	–	×
AT [3]	PR	E	proof system	✓	×	×
TBAN [60]	PR	E/T	proof system	✓	×	×
VO [62]	PR	E	proof system	–	–	×
SVO [61]	PR	E	proof system	✓	–	×
WS5 [17]	PR	E	proof system	✓	✓	✓
FWS5 [17]	FO	E	proof system	✓	✓	✓
TWS5 [4]	PR	E/T	proof system	✓	✓	✓
L_n^{KX} [30]	PR	E/A	knowledge algorithm	–	–	EXP
TDL [45]	FO	E/A/T	proof system	✓	✓	EXP
KL_n [23]	FO	E/T	resolution prover [22]	–	–	×
TBL [53]	FO	E/T	proof system	✓	✓	×
TML+ [41]	FO	E/T	tableau prover [51,52]	✓	✓	×
FL [56]	FO	E/T	KEM prover [29]	✓	✓	×
$ECKL_n$ [50]	FO	E/T	MCTK model checker	–	–	×
CTLK [8]	FO	E/T	MCMAS model checker [43,44]	–	–	×
CTLS5 [9]	FO	E/T	MCMAS model checker [43,44]	–	–	×
CTLKR [10]	FO	E/T	MCMAS-E model checker	–	–	×
ICTLK [12]	FO	E/T	MCMAS-S model checker [11]	–	–	×

Figure 2: Some Logics Applied in Analyzing Authentication Protocols

Protocol	Logics	Protocol	Logics
NSPK	WS5,KL_n,$ECKL_n$,CTLK,CTLS5,ICTLK	Mix	FWS5
WMF	BAN,TWS5,TML+,CTLK,CTLS5	Duck-Duck-Goose	WS5, L_n^{KX}
TESLA	TWS5,TDL,TBL,FL	M-TESLA	TWS5
KSL2	CTLK	ISO2PUCCF	CTLK
ISOSK2PU	CTLK	S-RPC	CTLS5
KSL	CTLS5	FOO e-voting	CTLKR

Figure 3: Some Protocols Analyzed by The Logics in Figure 2

due to the logical omniscience problem, which was an immediate consequence of its standard Kripke semantics [19]. Because of the logical omniscience problem, the logic bypasses the principals' restricted knowledge. Thus, AT is not enough for modeling different runs of a protocol. This logic follows BAN's method for formalizing security properties. However, it can also formalize such properties at specific points of protocol runs. For example, we may want to verify whether a formula such as $P \leftrightarrow^k Q$ is true at a specific point $\langle r, t\rangle$ of a protocol run or not.

The correctness of a security protocol highly depends on the evolving knowledge of principals communicating through the protocol steps while time is passing. In 1993, Syverson added temporal operators to BAN for the first time. We call this logic TBAN throughout this paper [60]. TBAN could verify a key distribution protocol that the previous BAN-like logics could not because they lacked temporal modalities and ignored a casual consistency attack on the protocol. TBAN is sound with respect to the AT semantics. Moreover, it is able to formalize temporal modalities and statements. Thus, the specification power of TBAN is more than its predecessors. This logic was a starting point in using both temporal and epistemic modalities for analyzing authentication protocols and later many other logics followed this approach [53, 4, 23, 22, 52, 56, 8]. For example, we can formalize authentication for NSPK as follows where the symbol $\Box$ is read as "always":

$$\Box\ A\ Believes\ A \leftrightarrow^{n_a.n_b} B$$
$$\Box\ B\ Believes\ A \leftrightarrow^{n_a.n_b} B.$$

Although TBAN is more expressive than its predecessors, it cannot analyze such protocols as M-TESLA, Mix, and Dual Signature protocols [4, 17] since it is not omniscience-free.

Contemporary to TBAN, van Oorschot followed another line of research and extended BAN to facilitate the verification of key agreement protocols [62]. The extended logic was named VO. Although BAN, GNY, and VO have proof

Logic	Attacker	Comments
BAN [14]	Implicit	It has no formal attacker model, but the capabilities of the attacker are somehow embedded in the proof system.
GNY [27]	Implicit	It considers the attacker similar to that one of BAN.
AT [3]	Implicit	It considers the attacker similar to that one of BAN.
TBAN [60]	Implicit	It can verify a protocol against an attack that needs temporal modalities to be formalized.
VO [62]	Implicit	It considers the attacker similar to that one of BAN.
SVO [61]	Implicit	It considers the attacker similar to that one of BAN.
WS5 [17]	Implicit	The underlying generalized Kripke semantics restricts the knowledge gained by the attacker.
FWS5 [17]	Implicit	It considers the attacker similar to that one of WS5.
TWS5 [4]	Implicit	It considers the attacker similar to that one of WS5, but it can also verify highly time-dependent protocols.
L_n^{KX} [30]	Explicit	It models the Dolev-Yao attacker by awareness algorithms.
TDL [45]	Explicit	It models the Dolev-Yao attacker by awareness algorithms.
KL_n [22]	Explicit	It specifies the attacker's capabilities by logical formulas.
TBL [53]	Implicit	It does not prove if it can model the defined attacker.
TML+ [52]	Implicit	The attacker capabilities are embedded in the proof system.
FL [56]	Implicit	It does not prove if it can model the defined attacker.
$ECKL_n$ [50]	Explicit	The attacker is defined similar to that one of Dolev-Yao.
CTLK [8]	Explicit	It models the Dolev-Yao attacker as an environment.
CTLS5 [9]	Explicit	The attacker model is similar to that one of CTLK.
CTLKR [10]	Explicit	The passive attacker links receipt provider and its vote.
ICTLK [12]	Explicit	The attacker model is similar to that one of CTLK.

Figure 4: The Attacker Model of The Logics in Figure 2

systems, we cannot analyze their soundness or completeness because they lack formal semantics. All of these BAN extensions were unified into a sound logic - called SVO [61] - whose axioms and rules were simplified. The completeness of BAN was finally proven in 2007 when a proper proof system and formal semantics were provided for this logic. GNY, AT, VO, and SVO have no formal attacker model, but the capabilities of the attacker are somehow embedded in the proof system. So, the attacker model of these logics is similar to that of BAN.

In 2005, it was shown that the AT semantics could not identify some possible attacks because of the logical omniscience problem. To solve this problem, Cohen and Dam provided a generalized Kripke semantics for BAN such that BAN's soundness, completeness, and decidability were proven [19, 18]. Using this semantics, BAN can be embedded into an S5 logic where some specific message permutations are defined over messages. In this way, a formula $K_i\phi$ is true in a possible world w if for every possible world w' of the model which is indistinguishable from w in i's view and with respect to a message permutation ρ, $\rho(\phi)$ is true in w'. The formula $\rho(\phi)$ is the one in which every message m is replaced by $\rho(m)$ [19].

Such a generalized Kripke semantics results in a weak necessitation rule for BAN. In this way, the application of a weak necessitation rule along with axiom K does not lead to logical omniscience in the derivations. So, the underlying semantics restricts the knowledge gained by an attacker. We call this logic Weakened S5 - WS5 for short - throughout the paper. WS5 can be extended by first-order quantifiers. The resulting logic is a sound and complete first-order logic [20], denoted by FOWS5 in this paper. It is shown that these logics can safely specify and verify the Mix protocol since they are omniscience-free. However, they do not have temporal modalities to analyze such protocols as stream authentication.

WS5 can also be extended by temporal modalities. The resulting logic is called TWS5. This logic successfully verifies a modified TESLA protocol - called M-TESLA - which cannot be analyzed by the previous temporal epistemic logics that are not omniscience-free [4]. However, these logics make use of message permutation functions,

which cause exponential run time [39]. Although WS5 and its extensions are sound, complete, and omniscience-free, the expressiveness of WS5 is less than its extensions since it is propositional and does not have temporal modalities.

WS5, FWS5, and TWS5 use different symbols to formalize security properties. While WS5 can use epistemic modalities, FWS5 and TWS5 can make use of quantifiers and temporal modalities along with epistemic modalities, respectively. As an example, assume that we want to formalize a message sending axiom in TESLA which says that if the sender S sends a message M to the receiver R, then R may receive this message in time interval [u, v] [4]. This can be done as follows where $next^u$ denotes u clock ticks later:

$S\ sends\ M\ \rightarrow\ (\bigcirc^u\ R\ receives\ M) \vee \ldots \vee (\bigcirc^v R\, receives\, M)$.

It is proven that the underlying generalized Kripke semantics of WS5 restricts knowledge gained by an attacker because the message permutation functions make the logic omniscience-free. In fact, it is shown that the Dolev-Yao deduction system reflects the semantics of WS5 implicitly because such a semantics considers all of the possible runs of a protocol in the formal model by applying the permutation functions on messages even those runs executed by a Dolev-Yao attacker [17, 4].

There are also many standard logics that were not originally developed for analyzing authentication protocols, but they are well adapted to this purpose. One of these logics is L_n^{KX} proposed by Halpern and Pucella in 2003 [30]. This logic uses two kinds of knowledge: implicit knowledge which is similar to that of BAN-like logics and explicit knowledge which links to some knowledge algorithms. The attacker model is defined based on the explicit knowledge in L_n^{KX}. In fact, L_n^{KX} defines attackers as the Dolev-Yao deduction system explicitly using a Dolev-Yao knowledge algorithm. We refer the interested reader to Ref. [30] to review the full algorithm.

Explicit knowledge prevents logical omniscience. Thus, attackers infer the statements that they can compute. It has been proven that algorithmic knowledge can model the Dolev-Yao deduction system. This model is a useful abstraction because it does not consider the cryptosystem used in the protocol and it can easily capture probability for guessing appropriate keys. L_n^{KX} does not have a proof system. Thus, it does not have soundness and completeness theorems. However, a proof system may also be developed for it. This logic is flexible and modular. Moreover, it can be extended with probabilities to guess keys [25, 32] according to Lowe's model for guessing attacks [49]. L_n^{KX} cannot verify highly time-dependent protocols because it has no temporal modalities. This logic also uses implicit knowledge to model principals' beliefs about what is happening during a protocol run.

Lomuscio and Wozan followed the same approach to develop a temporal epistemic logic called TDL for the specification and verification of TESLA [45]. The logic not only has traditional knowledge modalities but also has awareness operators to represent explicit knowledge. Thus, a part of TDL that links to explicit knowledge is omniscience-free. The logic defines attackers as the Dolev-Yao deduction system explicitly. This is formalized through a derivation relation that shows how an attacker can extract a message from a set of received messages and keys using admissible operations. As an example, there is a derivation rule as follows: if an agent i derives $m.m'$ using the Dolev-Yao knowledge algorithm, denoted by an awareness operator $\mathcal{A}_i^{DY}$, i can also derive m using this algorithm. TDL has a computationally-grounded semantics. Moreover, it is intuitive, sound, complete, and decidable. TDL has a high specification power because it is a first-order modal logic that has three types of modalities: epistemic, awareness, and temporal.

In 2004, Dixon, Fernandez Gago, Fisher, and van der Hoek introduced a first-order temporal logic of knowledge - called KL_n - for the specification and verification of authentication protocols and verified NSPK as a case study [23]. This logic is useful for reasoning about the evolving knowledge of a principal over time. Especially, this is important if we want to be sure that a principal obtains certain knowledge at a time instant. KL_n is a fusion of a linear time temporal logic and a multi modal epistemic logic S5. Thus, the logic is powerful enough to specify agents' capabilities explicitly by logical formulas. As an example of using quantifiers and modalities of KL_n for formalizing security requirements, consider the NSPK protocol as shown in Figure 1. Assuming that the predicate $value - nonce(N, V)$ of the logic indicates that the value of nonce N is V, we want to show that every other principal, other than A and B, who are running the protocol can never know the value of A's nonce. This is formalized as follows:

$$\forall\, V\ \ \Box\neg K_C\ value - nonce(N_a, V)$$

where $\Box$ and K_C are two modalities of KL_n that are interpreted as "always" and "agent C knows", respectively.

To prove a security goal ϕ from protocol specification ψ, where both ϕ and ψ are formulas in KL_n, we must prove $\psi \rightarrow \phi$. To do so, a resolution method is applied which is in fact a refutation system showing that $\psi \wedge \neg\phi$ is not satisfiable. A prototype theorem prover has been developed for a single modal version of temporal logics of knowledge but the developers insisted on a need to develop a more powerful prover to deal with the multi-modal case in order to prove theorems automatically [23].

Contemporary to KL_n, Liu, Ozols, and Orgun developed the TML+ logic for analyzing authentication protocols [41]. The logic uses the temporalization technique to combine an epistemic logic - called TML - with a simple linear-time temporal logic - called SLTL. This technique allows adding SLTL to TML in a hierarchical way such that the temporal operators can never appear in the scope of epistemic ones. TML+ can be applied for the verification of time-dependent properties of security protocols. To do so, a trust theory for a protocol that is to be verified is provided. The theory consists of TML+ axioms and rules along with

a specification of the protocol and initial assumptions in the form of TML+ formulas. Then, the theory is applied to drive security goals. TML+ is sound and complete, yet it is not omniscience-free. However, the logic assumes that an attacker cannot decrypt an encrypted message if he does not have the right key for decryption.

In 2007, a tableau system was developed for TML+ [51]. This was used for verifying both static and dynamic aspects of some protocols such as WMF, Needham-Schroeder symmetric key, and Kerbros. The developers proved the soundness and completeness of the labelled tableau calculus based on the soundness and completeness results of the constituent logics of TML+. Then, they sketched a resolution-type proof procedure and proposed a model checking algorithm for TML+ [51].

In 2006, Orgun, Ji, Chuchang, and Governatori constructed the FL logic for analyzing stream authentication protocols from TML and SLTL using the fibring technique [56]. In this way, FL is sound and complete with respect to its fibered model because its constituent logics are proven to be sound and complete [40, 34]. The original idea of the fibring technique is based on the assumption that both constituent logics are endowed with point-based semantics so that a model of the fibred logic is a point-based semantics and every point of the fibred model is closely related to a point in a model of each constituent logic. Thus, a model of a fibred logic can be related to several models in each of the constituent logics. As a consequence, a fibred logic preserves the theorems proven for its original logics [42]. The FL fibred model is such that the formulas of FL may contain any number of temporal and belief operators without any restrictions. Any formula whose main operator is a temporal modality, is interpreted by referring to the SLTL semantics and any formula whose main operator is a belief modality, is interpreted by referring to the Kripke semantics of TML.

The fibring technique also allows developing proof procedures from the constituent logics since the time and knowledge dimensions of FL are orthogonal[29]. In 2008, a modal tableau was developed for FL[29] by adapting KEM, a modal tableau system, showing how combinations of multi-modal logics can provide an effective tool for verifying the TESLA protocol. It has been proven that the adapted KEM is sound and complete for SLTL and TML. As a result of the fibring technique, these theorems have also been proven for FL. KEM can be used to automatically check for formal properties of security protocols [29].

The logic FL is more expressive than a temporalised belief logic such as TML+ [52] because temporal and belief modalities may appear in each other's domains. This logic is flexible because of its modular nature and it has a high specification power since it is a first-order logic making use of temporal and knowledge modalities. However, it is not omniscience-free because of the standard Kripke semantics of its belief aspect. In 2012, Ma *et al.* constructed a temporalized belief logic - called TBL [53] - that was less expressive than FL because of using the temporalization technique. However, it was appllied in verifying the TESLA protocol successfully. TBL is sound and complete, yet it is not omniscience-free. This logic follows FL to define the attacker's capabilities where they are defined similar to the Dolev-Yao model. However, neither of these two logics prove if they can model such attackers.

In 2009, Boureanu, Cohen, and Lomuscio presented an effective fully automatic approach for analyzing security protocols in the presence of the Dolev-Yao attackers. This approach makes use of a temporal epistemic logic, called CTLK [8]. The first step is to specify a security protocol in CAPSL [54], which is a high-level specification language formally describing protocols. Then, the specifications of the security protocol and security goals in CAPSL are translated by an automatic compiler into an ISPL (Interpreted Systems Programming Language), which is a set of CTLK specifications to be checked. The result of this translation is a file that a MCMAS model checker[3] takes as input and checks whether or not the security goals are satisfied by the protocol. In most cases, this gives a countermodel if they are not satisfied by the protocol. The main contribution is a compiler from protocol descriptions given in CAPSL into ISPL, the input language for MCMAS. The translation is optimised to limit the state explosion and benefit from MCMAS's various optimisations. To do so, the authors developed PD2IS (Protocol Descriptions to Interpreted Systems), an automatic compiler from CAPSL protocol descriptions to ISPL programs. This verification method assumes a bounded number of concurrent protocol sessions instantiated to run concurrently [8].

This CTLK approach makes use of the consistent message permutations introduced in Ref. [19] through a smart translation of security goals into CTLK formulas in order to prevent bad effects of the logical omniscience. However, CTLK is not omniscience-free. The approach models the Dolev-Yao attacker explicitly as an environment which has the capabilities of honest principals, can eavesdrop all communications, compose and replay messages into any protocol session, and perform cryptographic operators using his known keys. The attacker also has an identity and public and private keys that may be used to communicate with other principals and record their send actions. To avoid the state-explosion problem while model checking, MCMAS employs a fixed limited number of interleaved sessions. Moreover, it is assumed that the verification process does not support all types of nesting while encrypting messages and the messages are of finite-length. These assumptions are applied because an attack to a protocol may be usually found in a small run of the protocol consisting of only a few sessions [48].

No proof system has been presented for the CTLK logic yet. So, this logic does not have any proven soundness and completeness theorems [8]. However, one can develop a proof system with respect to the underlying semantics of CTLK and check if the logic is sound and complete accord-

[3]This is a BDD-based model checker for multi-agent systems supporting temporal-epistemic specifications [43]

ing to that proof system. CTLK is powerful and flexible enough to be extended. As a variant of this logic, CTLS5$_n$ has been developed to model-check detectability of attacks in multi-agent systems that are automatically generated for security protocols [9]. As another extension of CTLK, CTLKR has been developed for the analysis of an e-voting protocol - called Foo - with a passive attacker model who can link the receipt-providing principal and its vote [10]. Finally, ICTLK [12] has been developed for analyzing an unbounded number of concurrent protocol sessions. These extensions of CTLK use the same tools for verifying security protocols automatically. Contemporary to CTLK, Luo *et al.* provided a temporal epistemic logic - called ECKLn - that was applied in verifying the NSPK protocol using model checking techniques similarly. To do so, the authors implemented and automatically verified security protocols in a model checker for multiagent systems [9].

4 Conclusions

One of the main approaches for formal analysis of authentication protocols is using epistemic or temporal epistemic logics. The semantics of such logics should be able to model the protocol runs, including those executed by attackers. Then, theorem provers or model checkers are built on these logics to analyze the protocols against intended properties. Comparing the logics in Figure 2, it seems that WS5 and its extensions are the best choices for protocol analysis since they are sound, complete, and omniscience-free. However, Figures 3 and 4 show that CTLK and its extensions are very powerful in security protocol analysis, make use of automatic compilers and model checkers, and prevent bad effects of logical omniscience while modeling the Dolev-Yao message deduction.

It has been shown that it is an undecidable problem to find whether a security protocol is indeed secure or not [55]. Thus, it is practical to use trusted security protocols analyzed by different formal methods to provide fast solutions for existing problems of real life systems. At the same time, it is practical to move toward the best formal system for verifying authentication protocols. A good measure for finding such a formal system for security protocol analysis is to prove that it can model the Dolev-Yao message deduction. In this line, we investigated the epistemic and temporal epistemic logics and showed how far their properties such as soundness, completeness, being omniscience-free, and expressiveness may affect the analysis of authentication protocols. Some of these logics have no proof system, thus they have no soundness or completeness theorems. However, they may apply model checkers for analysis and use awareness algorithms for preventing the logical omniscience problem while modeling the Dolev-Yao message deduction explicitly [30, 45, 8]. Preventing the logical omniscience or avoiding its bad effects restricts a principal's knowledge to what he can compute or derive with his known keys and messages. Some other epistemic logics have proof systems, are omniscience-free, and model the attacker capabilities implicitly [19, 20, 4]. If such logics are logically sound, their derived judgments are more trustworthy. If they are complete, they may make use of automatic provers.

Comparing Figures 2 and 4, the epistemic logics modeling the Dolev-Yao message deduction are either omniscience-free or prevent the bad effects of omniscience. In fact, provers are usually built on logics that preserve properties such as soundness, completeness, and being omniscience-free so that one can trust their output. At the same time, model checkers deal with the state explosion problem by imposing some assumptions so that they can verify security protocols within an acceptable time. However, such assumptions may not stop us from using model checkers since they cover a wide range of security protocols and use existing tools. Finally, the expressiveness of such logics makes them powerful enough to both formalize attacker's capabilities by logical formulas and analyze different classes of authentication protocols. Specifically, if the logic has both temporal and epistemic modalities, it can analyze highly time-dependent protocols such as stream authentication protocols.

Although it has been shown that logics using algorithmic knowledge can model the Dolev-Yao message deductive explicitly [30], to the best of our knowledge it has not been proven if a temporal epistemic logic of authentication can model such an attacker implicitly using permutation-based generalized Kripke semantics. This can be a topic for further research. As seen in this paper, all of the logics that model the attacker capabilities implicitly are omniscience-free. Thus, this can be a starting point for this topic of research. Finally, it would be beneficial if such an overview extends to the use of epistemic and temporal epistemic logics in analyzing other security/privacy properties.

References

[1] M. Abadi and A.D. Gordon (1999) A calculus for cryptographic protocols: The spi calculus, *Information and computation*, Elsevier, 148(1): 1–70. https://doi.org/10.1006/inco.1998.2740

[2] M. Abadi and P. Rogaway (2002) Reconciling two views of cryptography (the computational soundness of formal encryption), *Journal of cryptology*, Springer, 15(2): 103–127. http://dx.doi.org/10.1007/s00145-007-0203-0

[3] M. Abadi and M.R. Tuttle (1991), A semantics for a logic of authentication, *Proceedings of the tenth annual ACM symposium on Principles of distributed computing*, ACM, 201–216. https://doi.org/10.1145/112600.112618

[4] S. Ahmadi and M.S. Fallah (2018) An Omniscience-Free Temporal Logic of Knowledge for Verify-

ing Authentication Protocols, *Bulletin of the Iranian Mathematical Society*, Springer, 44(5): 1–23. https://doi.org/10.1007/s41980-018-0087-9

[5] F. Belardinelli and A. Lomuscio (2010) Interactions between time and knowledge in a first-order logic for multi-agent systems, *Proceedings of the Twelfth International Conference on the Principles of Knowledge Representation and Reasoning*, AAAI Press, 38–48.

[6] F. Belardinelli and A. Lomuscio (2012) Interactions between time and knowledge in a first-order logic for multi-agent systems: completeness results, *Journal of Artificial Intelligence Research*, AI Access Foundation, 1–45. https://doi.org/10.1613/jair.3547

[7] B. Blanchet, B. Smyth, and V. Cheval (2015) ProVerif 1.90: Automatic Cryptographic Protocol Verifier, User Manual and Tutorial.

[8] , I. Boureanu, M. Cohen, and A. Lomuscio (2009) Automatic verification of temporal-epistemic properties of cryptographic protocols, *Journal of Applied Non-Classical Logics*, Taylor & Francis, 19(4): 463–487. https://doi.org/10.3166/jancl.19.463-487

[9] I. Boureanu, M. Cohen, and A. Lomuscio (2010) Model checking detectability of attacks in multiagent systems, *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems*, International Foundation for Autonomous Agents and Multiagent Systems, 1: 691–698.

[10] I. Boureanu, A.V. Jones, and A. Lomuscio (2012) Automatic verification of epistemic specifications under convergent equational theories, *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, International Foundation for Autonomous Agents and Multiagent Systems, 1141–1148.

[11] I. Boureanu, P. Kouvaros, and A. Lomuscio (2016) MCMAS-S- An experimental model checker or the verification of security properties in unbounded multi-agent systems. https://vas.doc.ic.ac.uk/software/mcmas-extensions/

[12] I. Boureanu, P. Kouvaros, and A. Lomuscio (2016) Verifying Security Properties in Unbounded Multiagent Systems, *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, International Foundation for Autonomous Agents and Multiagent Systems,1209–1217.

[13] C. Boyd and A. Mathuria (2013)*Protocols for authentication and key establishment*, Springer Science & Business Media, 2013. https://doi.org/10.1007/978-3-662-09527-0

[14] M. Burrows, M. Abadi, and R.M. Needham (1989) A logic of authentication, *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, The Royal Society, 426(1871): 233–271.

[15] C.C. Chang and H.J. Keisler (1990) *Model theory*, North Holland, 73.

[16] L. Chao, L. Hui, and M. Jianfeng (2009) Analysis the Properties of TLS Based on Temporal Logic of Knowledge, *Proceedings of the 5th International Conference on Information Assurance and Security*, IEEE, 2: 19–22. https://doi.org/10.1109/ias.2009.49

[17] M. Cohen (2007) Logics of Knowledge and Cryptography: Completeness and Expressiveness, PhD Thesis, KTH, Stockholm, Sweden.

[18] M. Cohen and M. Dam (2005) A completeness result for BAN logic, *Prococeedings of Methods for Modalities*, 4.

[19] M. Cohen and M. Dam (2005) Logical omniscience in the semantics of BAN logic, *Proceedings of the Foundations of Computer Security*, 121–132.

[20] M. Cohen and M. Dam (2007) A complete axiomatization of knowledge and cryptography, *Proceedings of the 22nd Annual IEEE Symposium on Logic in Computer Science*, IEEE, 77–88.

[21] F. Dechesne and Y. Wang (2010) To know or not to know: epistemic approaches to security protocol verification, *Synthese*, Springer, 177(1): 51–76. https://doi.org/10.1007/s11229-010-9765-8

[22] C. Dixon, M.C. Fernández Gago, M. Fisher, and W. van der Hoek (2007) Temporal logics of knowledge and their applications in security, *Electronic Notes in Theoretical Computer Science*, Elsevier, 186: 27–42. https://doi.org/10.1016/j.entcs.2006.11.043

[23] C. Dixon, M.C.F. Gago, M. Fisher, and W. Van Der Hoek (2004) Using temporal logics of knowledge in the formal verification of security protocols, *Proceedings of the 11th International Symposium on Temporal Representation and Reasoning*, IEEE,148–151.

[24] D. Dolev and A. Yao (1983) On the security of public key protocols, *Proceedings of the IEEE Transactions on Information Theory*, IEEE, 29(2):198–208.

[25] R. Fagin and J.Y. Halpern (1994) Reasoning about knowledge and probability, *Journal of the ACM*, ACM, 41(2):340–367. https://doi.org/10.1145/273865.274429

[26] R. Fagin, Y. Moses, J.Y. Halpern, and M.Y. Vardi (2003) *Reasoning about knowledge*, The MIT Press. https://doi.org/10.7551/mitpress/5803.001.0001

[27] L. Gong, R. Needham, and R. Yahalom (1990) Reasoning about belief in cryptographic protocols, *Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy*, IEEE, 234–248. https://doi.org/10.1109/risp.1990.63854

[28] A.D. Gordon and A. Jeffrey (2003) Authenticity by typing for security protocols, *Journal of computer security*, IOS Press, 11(4): 451–519. https://doi.org/10.3233/jcs-2003-11402

[29] G. Governatori, A.M. Orgun, and C. Liu (2008) Modal tableaux for verifying stream authentication protocols, *Journal of Autonomous Agents and Multi Agent Systems*. https://doi.org/10.1007/s10458-007-9027-4

[30] J.Y. Halpern and R. Pucella (2003) Modeling adversaries in a logic for security protocol analysis, *Formal Aspects of Security*, Springer, 115–132. https://doi.org/10.1007/978-3-540-40981-6_11

[31] J.Y. Halpern and R. Pucella (2011) Dealing with logical omniscience: Expressiveness and pragmatics, *Artificial intelligence*, Elsevier, 175(1): 220–235. https://doi.org/10.1016/j.artint.2010.04.009

[32] J.Y. Halpern and M.R. Tuttle (1993) Knowledge, probability, and adversaries, *Journal of the ACM*, ACM,40(4):917–960. https://doi.org/10.1145/153724.153770

[33] J. Heather, G. Lowe, and S. Schneider (2003) How to prevent type flaw attacks on security protocols, *Journal of Computer Security*, IOS Press, 11(2): 217–244. https://doi.org/10.3233/jcs-2003-11204

[34] G.E. Hughes and M.J. Cresswell (2012) *A new introduction to modal logic*, Routledge. https://doi.org/10.4324/9780203028100

[35] G. Jakubowska, P. Dembinski, W. Penczek, and M. Szreter (2009) Simulation of Security Protocols based on Scenarios of Attacks, *Fundamenta Informaticae*, 93(1): 185-203.

[36] A. Jurcut, T. Coffey, and R. Dojen (2013) Establishing and fixing security protocols weaknesses using a logic-based verification tool, *Journal of Communication*, 8(11): 795–806. https://doi.org/10.12720/jcm.8.11.795-805

[37] A.D. Jurcut, T. Coffey, and R. Dojen (2014) Design guidelines for security protocols to prevent replay & parallel session attacks, *computers & security*, Elsevier, 45: 255–273. https://doi.org/10.1016/j.cose.2014.05.010

[38] F. Kammuller and C.W. Probst (2015) Modeling and verification of insider threats using logical analysis, *IEEE Systems Journal*, IEEE, 11(2): 534–545. https://doi.org/10.1109/jsyst.2015.2453215

[39] S. Kramer (2007) Logical concepts in cryptography, PhD Thesis, École Polytechnique Fédérale de Lausanne.

[40] C. Liu and M. Orgun (1996) Dealing with multiple granularity of time in temporal logic programming, *Journal of Symbolic Computation*, Elsevier, 22(5): 699–720. https://doi.org/10.1006/jsco.1996.0072

[41] C. Liu, M.A. Ozols, and M. Orgun (2004) A temporalised belief logic for specifying the dynamics of trust for multi-agent systems *Advances in Computer Science-ASIAN 2004. Higher-Level Decision Making*, Springer, 142–156. https://doi.org/10.1007/978-3-540-30502-6_10

[42] C. Liu, M. Ozols, and M. Orgun (2005) A fibred belief logic for multi-agent systems, *AI 2015: Advances in Artificial Intelligence*, Springer, 29–38. https://doi.org/10.1007/11589990_6

[43] A. Lomuscio, H. Qu, and F. Raimondi (2009) MCMAS: A model checker for the verification of multi-agent systems, *Computer Aided Verification*, Springer, 682–688. https://doi.org/10.1007/978-3-642-02658-4_55

[44] A. Lomuscio, H. Qu, and F. Raimondi (2015) MCMAS: an open-source model checker for the verification of multi-agent systems *International Journal on Software Tools for Technology Transfer*, Springer, 19(1): 9–30. https://doi.org/10.1007/s10009-015-0378-x

[45] A. Lomuscio and B. Woźna (2006) A complete and decidable security-specialised logic and its application to the TESLA protocol, *Proceedings of the 5th international joint conference on Autonomous agents and multiagent systems*, ACM, 145–152. https://doi.org/10.1145/1160633.1160658

[46] G. Lowe (1995) An attack on the Needham-Schroeder public-key authentication protocol, *Information processing letters*, Elsevier, 56(3): 131–133. https://doi.org/10.1016/0020-0190(95)00144-2

[47] G. Lowe (1997) A family of attacks upon authentication protocols, epartment of Mathematics and Computer Science, University of Leicester.

[48] G. Lowe (1998) Casper: A Compiler for the Analysis of Security Protocols, *Journal of Computer Security*, IOS Press, 6(1-2): 53–84. https://doi.org/10.3233/jcs-1998-61-204

[49] G. Lowe (2004) Analysing protocols subject to guessing attacks, *Journal of Computer Security*, IOS Press, 12(1): 83–97. https://doi.org/10.3233/jcs-2004-12104

[50] X. Luo, Y. Chen, M. Gu, and L. Wu (2009) Model Checking Needham-Schroeder Security Protocol Based on Temporal Logic of Knowledge, *Proceedings of the NSWCTC'09. International Conference on Networks Security, Wireless Communications and Trusted Computing*, IEEE, 2: 551–554. https://doi.org/10.1109/nswctc.2009.384

[51] J. Ma, M.A. Orgun, and K. Adi (2011) An analytic tableau calculus for a temporalised belief logic, *Journal of Applied Logic*, Elsevier, 9(4): 289–304. https://doi.org/10.1016/j.jal.2011.08.003

[52] J. Ma, M.A. Orgun, and A. Sattar (2009) Analysis of authentication protocols in agent-based systems using labeled tableaux *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, IEEE, 39(4): 889–900. https://doi.org/10.1109/tsmcb.2009.2019263

[53] J. Ma and K. Schewe (2012) A Temporalised Belief Logic for Reasoning about Authentication Protocols, *Proceedings of the IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, IEEE, 1721–1728. https://doi.org/10.1109/trustcom.2012.59

[54] J.K. Millen (1996) CAPSL: Common Authentication Protocol Specification Language, *Proceedings of the 1996 workshop on New security Paradigms (NSPW)*, ACM, 96: 132. https://doi.org/10.1145/304851.304879

[55] J. Mitchell, A. Scedrov, N. Durgin, and P. Lincoln (1999) Undecidability of bounded security protocols, *Proceedings of the Workshop on Formal Methods and Security Protocols*.

[56] M.A. Orgun, J. Ma, C. Liu, and G. Governatori (2006) Analysing stream authentication protocols in autonomous agent-based systems2nd IEEE International Symposium on, *Proceedings of the 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing*, IEEE, 325–332. https://doi.org/10.1109/dasc.2006.19

[57] A. Perrig, R. Canetti, J.D. Tygar, and D. Song (2000) Efficient authentication and signing of multicast streams over lossy channels, *Proceedings of the IEEE Symposium on Security and Privacy*, IEEE, 56–73. https://doi.org/10.1109/secpri.2000.848446

[58] R. Ramanujam and S.P. Suresh (2005) Deciding knowledge properties of security protocols, *Proceedings of the 10th conference on theoretical aspects of rationality and knowledge*, National University of Singapore, 219–235.

[59] R. Ramezanian (2015) Process Algebraic Modeling of Authentication Protocols for Analysis of Parallel Multi-Session Executions, *The ISC International Journal of Information Security*, Iranian Society of Cryptology, 1(1): 55-67. https://doi.org/10.22042/isecure.2015.1.1.6

[60] P.F. Syverson (1993) Adding time to a logic of authentication, *Proceedings of the 1st ACM conference on Computer and communications security*, ACM, 97–101. https://doi.org/10.1145/168588.168600

[61] P.F. Syverson and P.C. Van Oorschot (1994) On unifying some cryptographic protocol logics, *Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy*, IEEE, 14–28. https://doi.org/10.21236/ada465512

[62] P. van Oorschot (1993) Extending cryptographic logics of belief to key agreement protocols, *Proceedings of the 1st ACM Conference on Computer and Communications Security*, ACM, 232–243. https://doi.org/10.1145/168588.168617