Probabilistic Reasoning of Inconsistent Belief in Protocol Analysis

Qingfeng Chen[1] Chengqi Zhang[2] Yi-Ping Phoebe Chen[1]

[1]School of Engineering and Information Technology, Deakin University, VIC 3125, Australia
[2]Faculty of Information Technology, University of Technology Sydney, NSW 2017, Australia

Abstract

Security protocols have been recently found with subtle flaws due to incomplete or ambiguous specification. Although formal methods have remarkably assisted in protocol analysis, they ignores the effect of hostile/uncertain environment, which might lead to inconsistent belief that can be held by principals in delivered messages. This discrepant belief may prevent us from representing the insecurity and uncertainty in a real trading situation. Unfortunately, the current approaches lack the ability to handle the inconsistent belief. This article presents a probabilistic method, which intuitively measures the belief from different principals that can be put on the goal of the protocol. The experiments demonstrate our method is useful to enhance the protocol analysis.

Keywords: Belief, Secure message, Probability, Inconsistency, Protocol

1. Introduction

Security protocols have been widely used to achieve not only data confidentiality, integrity, and authentication, and various desired security properties. However, some protocols are recently found to be subject to subtle flaws due to ambiguous/incomplete design and malicious communication environment [1]. *Belief* is referred as a principal's view of secure messages, which can be introduced directly or inferred through perception, assumption or communication between principals rather than an individual. Although a variety of formal methods, representing the belief and/or knowledge have been successfully used to analyze the protocol, they are either concerned with measuring the trust that can be put on the goal by the legitimate communicants using beliefs of the principals, or analyzing the security of a protocol by examining the knowledge gained by an intruder in the process of the protocol [2].

They ideally assume that the principals and communication channel are secure and trustworthy. However, in a malicious environment, the belief of principals in secure messages can no longer be justified. Furthermore, the above formal methods usually fail to model insecurity. This may lead to inconsistent beliefs between principals and prevent us from correctly evaluating the performance of the protocol. Thus, it is important to have the capability of modeling the imperfect working conditions and verifying the protocol under such circumstances.

The inconsistency in knowledge base has been widely studied in the past few years. The studies focus on either merging inconsistent knowledge [8] or measuring inconsistency between knowledge bases [7], whereas they rarely talk about the handling of inconsistent beliefs in secure messages.

Fagin and Halpern [6] presented probabilistic methods to deal with uncertainty and partial belief. They have been successful in handling the uncertainty and putting emphasis upon the beliefs of principals. In addition, Campbell et al. [2] proposed a probabilistic semantics for BAN logic [1] by adding capability of modeling less than perfect working conditions and drawing conclusions in such cases. However, the required inexact reasoning is computationally expensive. Thus, it is critical to deal with the inconsistent belief and enhance the formal proof of the protocol.

This paper presents how to deal with the inconsistent belief between principals by allowing probabilistic rules. It proposes a probabilistic semantic for ENDL [3](extension of non-monotonic dynamic logic), in which the probabilities of sentence and rule are viewed as observed belief and inferred belief, respectively and can be modeled in various levels of detail. A numeric estimation to the partial belief is presented by computing the minimum trust in the authenticated goal of the protocol in terms of QC logic [7]. The results are applied to SET protocol [5].

The rest of this paper is organized as follows.

In Section 2, we present the basic concepts of this article. Section 3 shows how to deal with the inconsistent belief in secure messages. The experiments are showed in Section 4. Section 5 concludes this paper.

2. Preliminaries

2.1. Formal logic, probability and belief

Some recent work regarding probabilistic logic can be found in [2]. We will be concerned here with the probability assigned to a set of assumptions and inference rules and relate them to the probability of conclusions which can be derived from them. These probabilities are useful to express less than certain assumptions and rules in the formal proof of the correctness of protocols.

A formal logic used to verify security protocols includes a set of sentences and inference rules, such as "*knows* (*Alice*, k)". The sentence is formed in terms of the syntax of the logic. It generates meaningful statements according to its semantics. An inference rule actually indicates the relationship between a collection of sentences.

Let $\mathcal{S}$ be the set of sentences in the formal logic and $\mathcal{S}^+$ be the closure of $\mathcal{S}$ which represents the set of whole sentences derived from the $\mathcal{S}$ by applying the rules $\mathcal{R}$ of the logic. A sentence ϕ can be true or false. The actual sentence, however, must be in one of true and false, but we might not know which one. Thus, we need to model this less than certain situations by using probability.

Let $\mathcal{B} = \mathcal{S} \cup \mathcal{R}$. Define the probability space $(\mathcal{W}, \mathcal{F}, \mathcal{P})$ for $\mathcal{B}$, in which the sample space $\mathcal{W}$ is a nonempty set and is known as outcomes, the events $\mathcal{F}$ denote sets of outcomes, and the probability measure $\mathcal{P}$ is a function that assigns to each event a probability between 0 and 1. Consequently, for each $w \in \mathcal{W}$, there is a truth assignment $a \mapsto w(a) = 0$ or 1 defined on $\mathcal{B}$, the probability of $a \in \mathcal{B}$ by

$$p(a) = p\ (\{w : w(a) = 1\}).$$

Let c be any conclusion that can be inferred from $\mathcal{B}$. We can extend the truth assignment to c by defining $w(c) = 1$ only if there is a proof of c from $\mathcal{B}$. In other words, we have $w(a) = 1$ for each sentences or inference rule a used in this proof. The probability of a conclusion $c \in \mathcal{A}$ can then be defined by

$$p(c) = p\ (\{w : w(c) = 1\}).$$

The set $\mathcal{W}$ actually represents the state of the world. In each state, some of the sentences in $\mathcal{S}$ and inference rules are valid. The probability p describes the degree of belief in the conclusions. The probability of a conclusion c can be defined as the probability that the proof of c from $\mathcal{B}$ is valid. It actually depends on the probabilities that the assumptions and rules are trustworthy. Therefore, it is necessary to attach probabilities to rules to model the insecure environments.

2.2. Semantic definition

A principal is a main participant in a protocol. A set of secure messages M is a finite set of sentences. We say M supports a sentence α if M implies α and trusts it (i.e., $M \models_{support} \alpha$), and M opposes α if M implies $\neg\alpha$ ($M \models_{support} \neg\alpha$) and distrusts it.

Suppose $P_1, P_2, \cdots, P_n$ ($n \geq 1$) are the principals of the protocol and ϖ is a function that assigns each of the principals a non-negative number representing the weight of the principal. The weight function ϖ is intended to capture the relative degree of importance of the principals. The smallest number that can be assigned to a principal is zero.

The following constructs defined in [3] present the basic processes of messages:

- **sends**(X, Y, m) represents the messages m was sent from principal X to principal Y.
- **knows**(X, m) represents principal X knows the message m.
- **sees**(X, m) represents principal X receives the message m.
- **fresh**(m) represents the message m is fresh.
- **believes**(X, Y, m) represents principal X believes the message m is fresh and really from Y.
- **authenticates**(X, Y, m) represents principal X believes that the message from principal Y is authentic.
- $\{m\}_K$ represents the message m is encrypted using the key K.

We also want to express facts that cannot be conveniently expressed by atomic sentences. The atomic sentences can be combined by logical operators, such as $\neg S_1$, $S_1 \wedge S_2$, $S_1 \vee S_2$.

As the symbols describe, the checking of a message authenticity could be viewed as the combination of a set of constructs. In [1], they make a realistic assumption that each principal can recognize and ignore his own messages; the originator of each

message is included for this purpose. However, this cannot exclude the possibility that different principals may have inconsistent beliefs in the message.

Sentences in the ENDL can generate various statements. The first type of sentences is the atomic sentence described above such as the *sends*() and *sees*(). It shows what knowledge are available to the principals in a transaction. In [2], these sentences are assigned a generalised truth value 1 in all possible states. However, the messages may not be always consistent in all states but can be missing or tampered in a hostile environment. It is unreasonable to continuously assign a truth value 1 to the sentences since the principals have inconsistent beliefs in them. In this article, we use the support of sentences as the truth value by extending the semantics of ENDL. The details can be seen in Section 3.

The second type of sentences include the axioms of ENDL, which represent the basic entailment relation with respect to encryption, decryption, key allocation, signature and authentication. For example, the principal P should know its own private key. The sentences below present the fundamental entailment relations of ENDL.

(1) $\vdash knows(P,\ m) \rightarrow sends(P,\ Q,\ m)$. This means if the principal P knows m, then P can send m to another principal Q.

(2) $\vdash sends(P,\ Q,\ m) \rightarrow sees(Q,\ m)$. This means if P sends m to Q, then Q will see m.

We will assign the generalised truth value 1 to similar sentences since they have been proved to be true. In ENDL logic [3], *sends*, *knows*, *sees* and *fresh* are primitive operators. They can be represented as a compact form, namely rules.

$$\vdash knows(P,\ m) \times sends(P,\ Q,\ m) \times fresh(m) \times sees(Q,\ m) \Rightarrow believes(Q,\ P,\ m)$$

There are three fundamental rules in ENDL that are used for the authentication of messages. They are derived from the authentication axioms of ENDL and have been proved to be true.

(1) $\vdash knows(X,\ m) \times knows(X,\ S(<ID_Y,\ T,\ H(m)>,\ Spv(Y))) \times knows(X,\ Spb(Y)) \Rightarrow authenticates(X,\ Y,\ m)$.

(2) $\vdash knows(X,\ m) \times authenticates(X,\ Y,\ H(m)) \Rightarrow authenticates(X,\ Y,\ m)$.

(3) $\vdash knows(X,\ Spb(Y)) \times knows(X,\ S(<ID_Y,\ T,\ Spb(Y)>,\ Spv(CA))) \times knows(X,\ Spb(CA)) \Rightarrow authenticates(X,\ Y,\ Spb(Y))$.

where ID_Y is Y's identity; T is the timestamp; $H(m)$ is the hashing of message m; and $Spv(Y)$ and $Spb(Y)$ represent Y's private and public signature key, respectively. The definition and use of timestamp can be found in [4].

Inference rules of ENDL play an important role in generating new beliefs according to the existing beliefs of principals and their available messages during the transaction. The probability that a rule holds can be defined as the probability that given that the beliefs of the conditions of the rule are true, then the belief of the conclusion of the rule is also true.

Definition 1 . *Let $\models_{support}$ be a supporting relationship. For a set of secure message M, M $\models_{support}$ is defined as follows, where α is an atom in $\mathcal{A}$, and each of them virtually denotes a message.*

$$\begin{cases} M_S \models_{support} \alpha & \text{iff } \text{“}knows(S,\ \alpha)\text{”} \wedge \text{“}fresh(\alpha)\text{”} \\ M_R \models_{support} \alpha & \text{iff } \text{“}believes(R,\ S,\ \alpha)\text{”} \wedge \text{“}fresh(\alpha)\text{”} \\ M_T \models_{support} \alpha & \text{iff } \text{“}believes(T,\ S,\ \alpha)\text{”} \wedge \text{“}fresh(\alpha)\text{”} \end{cases}$$

where M_S, M_R and M_T denote the set of messages of sender, receiver and the third party, respectively. In particular, the use of timestamp guarantees the message α is not a replay and really from the expected principal.

Definition 2 . *Let $\models_{match}$ be a matching relationship. For a set of secure messages M, M $\models_{match}$ is defined as follows, where R is a set of rules used to authenticate messages.*

$$\begin{cases} M_R \models_{match} r & \text{iff “}\exists\ M_R' \subseteq M_R, \exists\ r \in R\text{” and} \\ & \text{“}M_R'\ matches\ r\text{”} \\ M_T \models_{match} r & \text{iff “}\exists\ M_T' \subseteq M_T, \exists\ r \in R\text{” and} \\ & \text{“}M_T'\ matches\ r\text{”} \end{cases}$$

where the principal intends to find required messages to match the rule. If there is an inference rule that can be satisfied by the obtained messages, it is true; otherwise false. S is the initiator of the message. Therefore he/she just verifies the message m in terms of the supporting relation defined above but does not need to match the inference rules.

Definition 3 . *Let $\models_{auth}$ be a belief relationship. For a set of secure message M, M $\models_{auth}$ m is defined as follows, where m is a message needed to be authenticated.*

$$\begin{cases} M_S \models_{auth} m & \text{iff “}M_S \models_{support} m\text{”} \\ M_R \models_{auth} m & \text{iff “}M_R \models_{support} m\text{” and} \\ & \text{“}\exists r \in R,\ M_R \models_{match} r\text{”} \\ M_T \models_{auth} m & \text{iff “}M_T \models_{support} m\text{” and} \\ & \text{“}\exists r \in R,\ M_T \models_{match} r\text{”} \end{cases}$$

The above definitions actually represent the transition of beliefs when authenticating the message. They will be operated in order during the authentication.

3. Handling inconsistent beliefs in secure messages

3.1. Quality of support using minimal QC model

This section describes QC minimal model, and measures the observed belief of principals. QC [7] logic is motivated by the need to deal with belief.

Definition 4 . *Let $\mathcal{A}$ be a set of atoms. Let $\mathcal{O}$ be the set of objects defined as follows, where $+\alpha$ is a positive object, and $-\alpha$ is a negative object.*

$$\mathcal{O} = \{+\alpha \mid \alpha \in \mathcal{A}\} \cup \{-\alpha \mid \alpha \in \mathcal{A}\}$$

We call any X *in* $\wp(\mathcal{O})$ a QC model. So X can contain both $+\alpha$ and $-\alpha$ for some atom α.

For each atom $\alpha \in \mathcal{L}$, and each X *in* $\wp(\mathcal{O})$, $+\alpha \in X$ means that in X there is a reason for the belief α. Similarly, $-\alpha \in X$ means that in X there is a reason against the belief $\neg\alpha$. To measure inconsistency, the minimal QC model (MQC) is often used. A minimal QC model of M is a QC model X of M such that for every subset $Y \subset X$, $Y \notin \mathrm{QC}(X)$.

***Example* 1** . *Suppose $M_1 = \{\alpha, \beta \vee \gamma\}$ and $M_2 = \{\alpha \vee \beta, \neg\alpha \vee \gamma\}$. We have MQC($M_1$) = $\{\{+\alpha, +\beta\}, \{+\alpha, +\gamma\}\}$, MQC($M_2$) = $\{\{+\alpha, +\gamma\}, \{-\alpha, +\beta\}, \{+\beta, +\gamma\}\}$*

Definition 5 . *Let M be a set of secure messages and let X be a MQC of M. The support function from $\mathcal{A}$ to $[0, 1]$ is defined below when α is not empty, and supp(M, $\emptyset$) = 0.*

$$supp(X, \alpha) = \frac{|+\alpha|}{|+\alpha \cup -\alpha|} \times 100 \qquad (1)$$

where $|\alpha|$ is the number of occurrence of the set of α in X. If $supp(X, \alpha) = 0$, then we can say X has no opinion upon α and vice versa; if $supp(X, \alpha) = 1$, it indicates that there is no negative object $-\alpha$ in X.

Furthermore, the support between α and M is defined below.

Definition 6 . *Suppose $X_1, X_2, \cdots, X_k$ are all MQC models of M. The support between M and a sentence α is defined as the mean of all supp(X_i, α).*

$$supp(M, \alpha) = \sum_{i=1}^{k} supp(X_i, \alpha)/k \qquad (2)$$

where if $supp(M, \alpha) = 0$, it indicates α is not supported by M at all.

We must check whether α is fresh or not like Definition 1. If the α may be identified as a replay, it will be reported to the user, rather than calculating the support. The obtained support represents the observed belief. To decide the trust in the goal, we need to consider the weight of principals and the probability of inference rules together.

3.2. Evaluating inconsistent beliefs

This article aims to know the degree that in X's view the message m sent from Y is authentic. The satisfiable conditions can be seen in Definition 3.

Definition 7 . *Let m be an authenticated goal, let $Bel_P(m)$ be the belief of the principal P in m, $P \in \{S.\ R,\ T\}$ and let $p_R(r)$ be the probability of rules that can be used by P to authenticate m. The belief of principals in a statement m can be defined as follows.*

$$\begin{cases} Bel_S(m) & = supp(M_S, m) * \varpi_S \\ Bel_R(m) & = p_R(r) * \varpi_R \\ Bel_T(m) & = p_T(r) * \varpi_T \end{cases}$$

where ϖ_S, ϖ_R and ϖ_T represent the assumed belief of S, R and T, respectively; the sender only needs the assumed belief and observed belief ($supp(M_S, m)$) to authenticate m; the probabilities of $p_R(r)$ and $p_T(r)$ rely on the observed belief of R and the observed belief of T, respectively. The details can be seen in the followings.

Although a number of methods were used to identify the weight of principals using specified criteria, it is not our emphasis to discuss the weight of principals in this article.

Until now, we have not referred closely to how the probability of the inference rules is specified. To authenticate a message m, the user has to check whether the held messages match at least one of the known rules. Sometimes, there might be more than one rule that can be satisfied.

Let $r_1, r_2, \cdots, r_n$ be a set of available inference rules. Suppose $Path_P$ represents a subset of rules that is used to authenticate the statement m.

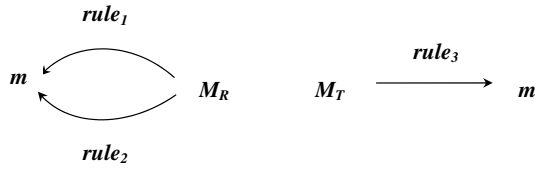

Fig. 1: Authentication path

Example 2 . Consider a process of authentication shown in *Figure 1.* R can authenticate statement m by using two paths including $rule_1$ and $rule_2$. Nevertheless, M_T verifies m by using $rule_3$ only. More generally, we have $Path_R = \{rule_1, rule_2\}$ *and* $Path_T = \{rule_3\}$.

If an authentication includes one path only, it is simple to compute the probability of the rule by using conditional probability. However, if there are multiple paths, it is necessary to use the probabilistic model to describe the probability of each way.

Suppose $A = \{A_1, \ldots, A_n\}$ denotes the conditions of $r: A_1 \times \ldots \times A_n \Rightarrow B$. As described above, the probability of a rule, namely $p(\mathrm{r})$, is a conditional probability based on the conditions. The conditions are independent with each other since they represent independent sentences. We have

$$p(r) = p(B|A_1A_2 \cdots A_n)$$

$$= p(AB)/p(A) = p(AB)/\prod_{i=1}^{n} p(A_i)$$

where $p(A_i) > 0$ and $p(B)$ can be measured by using the support defined in Definition 6. Note, $p(r) = 0$ if $\exists\ p(A_i) = 0$.

Furthermore, we have $p(A) + p(B) - 1 \leq p(AB) \leq p(B)$ according to probability theorem. Consequently, we can obtain a probability interval regarding the rule. The minimum is selected as its probability to calculate the belief, namely

$$\begin{cases} if \prod_{i=1}^{n} p(A_i) \neq 0, & (p(A) + p(B) - 1)/\prod_{i=1}^{n} p(A_i) \\ otherwise & 0 \end{cases} \quad (3)$$

Definition 8 . *Let r_i: $A_{i1} \times \ldots \times A_{in} \Rightarrow c_i$ and r_j: $A_{j1} \times \ldots \times A_{jk} \Rightarrow c_j$ be two inference rules, $i \neq j$. The r_i is said to be independent of r_i if $\{A_{i1}, \ldots, A_{in}\} \cap \{A_{j1}, \ldots, A_{jk}\} = \emptyset$; otherwise they are said to be dependent. We have*

(1) $p(r_i \cup r_j) = p(r_i) + p(r_j)$, if r_i is independent of r_j;
(2) $p(r_i \cup r_j) = p(r_i) + p(r_j) - p(r_i) * p(r_j)$, if r_i and r_j are dependent with each other.

Example 3 . *Suppose r_1: 'X knows m' × 'X knows Y's digital signature on m' × 'X knows Y's public signature key' ⇒ 'X authenticates Y, m' and r_2: 'X knows m' × 'X authenticates Y, H(m)' ⇒ 'X authenticates Y, m' are two rules. r_1 is not independent of r_2 because they have a common condition 'X knows m'.*

Suppose n rules including $r_1, \cdots, r_n$ are used in the authentication. This gives

$$P(\bigcup_{i=1}^{n} r_i) = s_1 - s_2 + s_3 + \cdots + (-1)^{n+1} s_n$$

where $s_1 = \sum_{i=1}^{n} P(r_i)$, $s_2 = \sum_{1 \leq i < j \leq n} P(r_i r_j)$, $s_3 = \sum_{1 \leq i < j < k \leq n} P(r_i r_j r_k)$, $\cdots$, $s_n = P(r_i r_j \cdots r_n)$.

Suppose $\{r_{R1}, \cdots, r_{Rk}\}$ and $\{r_{T1}, \cdots, r_{Tl}\}$ are two set of rules used to validate m by R and T, respectively. Therefore, we have

$$p_R(r) = p(\bigcup_{i=1}^{k} r_{Ri}) \quad (4)$$

$$p_T(r) = p(\bigcup_{i=1}^{l} r_{Ti}) \quad (5)$$

Refer to the strict weighted majority mentioned in [8], the combined weight of the support for m should be over 50% of the total weight. Let η be the threshold of weight. We have

$$\eta = \sum_{P \in \{S,R,T\}} \varpi_P / 2 \quad (6)$$

We can work out the sum of the beliefs of S, R and T regarding m.

$$Bel(m) = Bel_S(m) + Bel_R(m) + Bel_T(m) \quad (7)$$

Then the user is able to determine whether m is secure or not according to the discriminant below.

$$\begin{cases} \text{if } Bel(m) \geq \eta, & m \text{ is believed to be secure} \\ otherwise & m \text{ is insecure} \end{cases}$$

4. Experiments

We now look at an experiment of dealing with inconsistent belief in secure messages using a simulated data from SET protocol. For simplify, it assumes that all messages are fresh and the principals are trustworthy.

$M_S = \{\alpha, \neg\alpha, \beta, \varphi, \phi, \theta, \alpha \wedge \beta \wedge \theta \rightarrow \gamma, \delta\}$, $M_R = \{\neg\alpha, \alpha, \phi, \varphi, \neg\gamma, \phi \wedge \neg\gamma \rightarrow \neg\theta, \neg\delta\}$ and $M_T = \{\alpha, \beta, \phi, \varphi, \theta, \gamma, \delta\}$. Let $\varphi \equiv$ *'the hashing*

of message m' and $\varphi^{'} \equiv$ '*A authenticates* φ'. Let $\alpha \equiv$ '*message m*', $\beta \equiv$ '*the sender's private signature key Spv(S)*', $\phi \equiv$ '*the sender's public signature key Spb(S)*', $\theta \equiv$ '*the sender's identity* ID_S', $\gamma \equiv$ '*the sender's digital signature* $S(<ID_S, T, H(m)>, Spv(S))$' and $\delta =$ '*authenticated m*'. Let $\alpha^{'} \equiv$ '*A* knows α', $\phi^{'} \equiv$ '*A* knows ϕ', $\gamma^{'} \equiv$ '*A* knows γ' be statements, which are three conditions of the inference rule r: $\alpha^{'} \times \phi^{'} \times \gamma^{'} \Rightarrow \delta$. *Let* $\varpi_S = \varpi_R =$ 0.4 and $\varpi_T = 0.9$.

Two rules can be used to authenticate α in this experiment. The user can use not only r_1: $\alpha^{'} \times \beta^{'} \times \gamma^{'} \Rightarrow$ '*A* authenticates α', and another rule r_2: $\alpha^{'} \times \varphi^{'} \Rightarrow$ '*A* authenticates α'. It is noted that there is an intersection between the conditions $\{\alpha^{'}, \beta^{'}, \gamma^{'}\}$ of r_1 and the conditions $\{\alpha^{'}, \varphi^{'}\}$ of r_2, namely $\{\alpha^{'}, \beta^{'}, \gamma^{'}\} \cap \{\alpha^{'}, \varphi^{'}\} = \{\alpha^{'}\}$.

We can first generate $MQC(M_S) \equiv \{\alpha, \neg\alpha, \varphi, \beta, \theta, \phi, \gamma, \delta\}$, $MQC(M_T) \equiv \{\alpha, \beta, \varphi, \theta, \phi, \gamma, \delta\}$ and $MQC(M_R) \equiv \{\alpha, \neg\alpha, \varphi, \phi, \neg\theta, \neg\gamma, \neg\delta\}$.

According to the formula (1), we have $supp(M_S, \alpha) = supp(M_R, \alpha) = 0.5$, $supp(M_T, \alpha) = 1$, $supp(M_S, \beta) = supp(M_T, \beta) = 1$, $supp(M_R, \beta) = 0$, $supp(M_S, \gamma) = supp(M_T, \gamma) = 1$, $supp(M_R, \gamma) = 0$, $supp(M_S, \delta) = supp(M_T, \delta) = 1$, $supp(M_R, \delta) = 0$, $supp(M_S, \varphi) = supp(M_T, \varphi) = supp(M_R, \varphi) = 1$.

Then $Bel_S(m) = supp(M_S, \alpha) * \varpi_S = 0.2$. According to the formula (3), (4) and (5), we have $p_R(\mathrm{r}_1) = (supp(M_R, \alpha) * supp(M_R, \beta) * supp(M_R, \gamma) + supp(M_R, \delta) - 1) / supp(M_R, \alpha) * supp(M_R, \beta) * supp(M_R, \gamma) = 0$; $p_T(\mathrm{r}_1) = (supp(M_T, \alpha) * supp(M_T, \beta) * supp(M_T, \gamma) + supp(M_T, \delta) - 1) / supp(M_T, \alpha) * supp(M_T, \beta) * supp(M_T, \gamma) = 1$; $p_R(\mathrm{r}_2) = (supp(M_R, \alpha) * supp(M_R, \varphi) + supp(M_R, \delta) - 1) / supp(M_R, \alpha) * supp(M_R, \varphi) = 1$; $p_T(\mathrm{r}_2) = (supp(M_T, \alpha) * supp(M_T, \varphi) + supp(M_T, \delta) - 1) / supp(M_T, \alpha) * supp(M_T, \varphi) = 1$.

Thus, $p_R(\mathrm{r}) = p_R(r_1) + p_R(r_2) - p_R(r_1) * p_R(r_2) = 0 + 1 - 0 = 1$ and $p_T(\mathrm{r}) = p_T(r_1) + p_T(r_2) - p_T(r_1) * p_T(r_2) = 1 + 1 - 1 = 1$. We have $Bel_R(m) = p_R(\mathrm{r}) * \varpi_R = 0.4$ and $Bel_T(m) = p_T(\mathrm{r}) * \varpi_T = 0.9$. As a result, m is believed to be secure in this transaction using the protocol due to $Bel(m) = 1.5 > \eta = (0.4 + 0.4 + 0.9)/2 = 0.85$.

5. Conclusions

Formal methods have played an central role in analyzing security protocols. However, they do not consider the modelling of insecurity, which may lead to inconsistent beliefs between principals. With the increasingly varied and complex protocols, it is necessary to handle the inconsistent belief and correctly measure the trust in the goal of the protocol. This article proposed a probabilistic method to intuitively measure the inconsistent beliefs between principals, and intended to merge the inconsistent belief by a weighted majority criterion. In particular, the features of secure messages are taken into account. The presented experiments demonstrate that our method is able to complement and enhance the formal proof of correctness of the protocol.

References

[1] M. Burrows, M. Abadi and R. Needham, A logic for Authentication. *ACM Transactions on Computer Systems*, 8(1): 18-36, 1990.

[2] E.A. Campbell, R. Safavi-Naini and P.A. Pleasants, Partial Belief and Probabilistic Reasoning in the Analysis of Secure Protocols. *Proceedings. Computer Security Foundations Workshop V*, pp. 84-91, IEEE Comput. Soc. Press, 1992.

[3] Q.F. Chen C.Q Zhang and S.C Zhang, An extension of NDL for verifying secure transaction protocols. *Knowledge and Information Systems*, 7(1): 84-109, 2005.

[4] D. Denning and G. Sacco, Timestamp in Key Distribution Protocols. *Communications of ACM*, 24(8): 533-536, 1981.

[5] SET (1997) SET Secure Electronic Transaction Specification, Book 3: Formal Protocol Definition, Version 1.0, 1997.

[6] J.Y. Halpern and R. Fagin, Two views of belief: belief as generalized probability and belief as evidence. *Artificial Intelligence*, 54: 275-317, 1992.

[7] A. Hunter, Measuring Inconsistency in Knowledge via Quasi-classical Models. *Proceedings of AAAI02*, pp. 68¨C73, 2002.

[8] J. Lin, Integration of Weighted Knowledge Bases. *Artificial Intelligence*, 83(2): 363-378, 1996.