Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2014, Article ID 138370, 10 pages
http://dx.doi.org/10.1155/2014/138370

Research Article

Formal Modeling and Analysis of Fairness Characterization of E-Commerce Protocols

Chengwei Zhang, Xiaohong Li, Jing Hu, Zhiyong Feng, and Jiaojiao Song

School of Computer Science and Technology, Tianjin University, Tianjin 300072, China

Correspondence should be addressed to Xiaohong Li; xiaohongli@tju.edu.cn

Received 24 January 2014; Revised 4 May 2014; Accepted 4 May 2014; Published 1 June 2014

Academic Editor: Xiaoyu Song

In the past, fairness verification of exchanges between the traders in E-commerce was based on a common assumption, so-called nonrepudiation property, which says that if the parties involved can deny that they have received or sent some information, then the exchanging protocol is unfair. So, the nonrepudiation property is not a sufficient condition. In this paper, we formulate a new notion of fairness verification based on the strand space model and propose a method for fairness verification, which can potentially determine whether evidences have been forged in transactions. We first present an innovative formal approach not to depend on nonrepudiation, and then establish a relative trader model and extend the strand space model in accordance with traders' behaviors of E-commerce. We present a case study to demonstrate the effectiveness of our verification method.

1. Overview

The E-commerce protocol is a special kind of security protocol aiming to coordinate the exchange of valuable information between traders. The fairness of E-commerce protocols is the essential property and has become a research hotspot in recent years. E-commerce protocols are different from traditional cryptographic protocols, and common security analysis methods are invalid to fairness validating. It is common to extend existing methods like Kailar logic [1], SVO logic [2], CSP process algebra [3], and strand space model [4–6] to analyze such protocols. These methods are based on the assumption that the nonrepudiation has been established, and then verify fairness of fair exchange protocols. However, fairness validating of E-commerce protocol needs to verify not only the exchange process of protocols but also the evidences exchanged between traders in transactions.

To verify fairness of E-commerce protocols, we present a strand space model based fairness verification method in this paper, which is independent of nonrepudiation. The concept of fairness is decomposed into fair exchange of evidences and fair evidences of exchange, and a formal definition of fairness and fairness evidences is introduced. The trader model is constructed according to trader's behaviors, which is different from Dolev-Yao penetrator model [7]. The strands, in which no evidences from its opponent are obtained by the entity, are analyzed; thus, the nontermination dilemma caused by the state space explosion problem can be avoided.

The paper is organized as follows: related work on fairness verification is discussed in Section 2. Some background knowledge about strand space theory and the Dolev-Yao penetrator model is introduced in Section 3. The core body of the paper is followed in Section 4, which consists of definition and analysis of fairness as well as the discussion of trader models and the extended theories of the strand space model. The improved strand space model is tested by using the EMH protocol in Section 5, which proposes an improved protocol. Finally, the paper's conclusion is offered in Section 6.

2. Related Works

The E-commerce protocol has two objectives: the first one enables each protocol participant to seamlessly exchange valuable information. The second one enables each protocol participant to ensure the exchangeable fairness. The fair exchange protocol is a basic protocol of various E-commerce applications. From the perspective of protocol structure, it can be divided into three categories: gradual exchange protocols, on-line TTP exchange protocols, and offline TTP exchange protocols. In the 1980s, the appearance of gradual exchange protocols gradually increases the probability of correctness over several rounds of communication, but these protocols only can do progressive fairness [8]. The third party protocol requires a trusted third party. The third party is on-line which is required to be active in every exchange transaction [9]. These protocols relying heavily on TTP could easily lead to overload of networks and susceptible attacks. Offline fair exchange protocols have two phases: message exchange phase and dispute resolution phase. TTP was used only in the dispute resolution phase. This type of protocol reduces the problem of TTP as a source of bottleneck, because TTP is used very rarely and not involved in every round of exchange. Currently, most E-commerce protocols are based on offline TTP exchange protocols.

Formal analysis of security in E-commerce protocols is carried out more frequently than the proposal of traditional security protocols. For this reason, a variety of theories have been proposed, such as Kailar logic, SVO logic, and analysis method based on the CSP process algebra. Kailar proposed the concept "accountability," and a kind of logic to analyze accountability named Kailar logic. Accountability refers that protocol participants an prove that they have done something. Kailar logic verifies accountability by analyzing whether the parties have obtained the evidence that demonstrates the occurrence of the exchange. Zhou Jianying and Dieter Gollman put forward concepts EOO (nonrepudiation evidences of origin) and EOR (nonrepudiation evidences of receipt), which are used as the evidences of accountability. A nonrepudiation protocol allows two potentially mistrusting parities to exchange an electronic message together with EOO and EOR over the Internet in a fair way, that is, each party gets the other's term(s) or neither party does. They use SVO logic to verify nonrepudiation protocols. Steve Schneider uses CSP process algebra to analyze nonrepudiation protocols. The method he proposed can be utilized to analyze accountability and fairness. However, the analysis of the exchangeable fairness is based on on-line TTP rather than on offline TTP protocols which own a branched structure.

It is a prevailing approach for researchers to extend the strand space model for the analysis of E-commerce protocols in recent years. Yang and Deng [10] extended the strand space model to analyze TLS and IKE protocols. They proposed semiregular entities to denote entities which are different from penetrator and regular entities. Wang et al. [11] employed not only nonrepudiation EOO and EOR as the fairness evidences but also a similar method to prove the fairness of the iKP protocol as well. At the same time, Liu et al. [12] used a similar method to prove the fairness of the IBS protocol. These studies have provided a detailed description and analysis of E-commerce as well as its security properties.

All these methods verifying fairness are based on the prerequisite of guaranteeing exchangeable testimony as the satisfied nonrepudiation. They use nonrepudiation evidences as the fairness evidences. Nonrepudiation means that counterparts cannot deny that they have received or sent some information [13]. The E-commerce protocol requires not only fair exchange of evidences, but also the equivalence of exchange evidences.

The methods mentioned above are limited in the scope of the analyzed protocol, first of all, not all E-commerce protocols use TTP as an arbitration and fault-tolerant process [14]; in addition, the nonrepudiation evidences are the evidences of participants having sent or received information, while the fairness evidences are the valuable, quantifiable information exchanged by traders such as electronic money, bills, and signatures in running of an E-commerce protocol. These methods must assume that traders cannot forge evidences first and then analyze the fairness of exchange processes. Therefore, the study needs to explore a formal definition of fairness which does not rely on nonrepudiation.

In addition, Fröschle [15] put forward the concept branch of the strand space model. Branch describes the different choices of entities in an E-commerce protocol, which helps to traverse all the behaviors of protocol entities. Guttman [16, 17] defined fairness evidences as a collection of valuable information and then tracked all steps of a run to prove the fairness of the exchange protocol. Strand space model uses strands to describe all possible behaviors of protocol participants, and needs more than one strand to describe an entity's behaviors, making the model complex owing to the fact that participants in E-commerce protocols are possibly dishonest. At the same time, as the analysis of the fairness evidences requires considerable understanding of a protocol, so fairness can't be verified automatically. In addition, the space of entities' behaviors may be unlimited because of dishonesty, and the traversal may not be terminated; thus, a fairness verification method not to traverse all possible behaviors of protocol's entities needs to be explored.

Besides, although Guttman [16, 17] and we both extend the strand space model to verify fairness, the details of verification process between us are quite different. Guttman developed a model connecting protocol execution with state and state change and defined a new notation named state synchronization events to synchronize states between protocol participants. The "fair" in "fair exchange" in their definition refers to the balanced evolution of the state. In our method, we define the trader model to restrict traders' behaviors, and analyse all the possible results of a protocol. Their fairness focus on the transaction process, while we concentrate on the transaction result.

By the way, the protocol used as an instance to test our method in Section 5 has been proved to be unfair [18]. They analyzed and improved the EMH protocol, while the method they proposed is very purposeful and can only be used in few protocols [19].

3. Basic Concepts of Strand Space Model

Strand space model was proposed by Fabrega, Herzog, and Guttma in 1998, which analyzes security protocols in a hybrid analysis method combined to theorem proving and trace. It was proposed to formally analyze authentication and confidentiality of security protocols at first. There are some basic definitions of stand space model as follows.

A denotes the set of messages that can be exchanged between principals in a protocol. Terms are the elements of A. In a protocol, principals can either send or receive terms. To strand space model, the positive sign represents sending a term, whereas the negative sign represents receiving a term according to its occurrence.

Definition 1. A signed term is a pair $\langle \sigma, a\rangle$ with $a \in A$ and $\sigma \in \{+, -\}$. We will write a signed term as $+t$ or $-t$. $(\pm A)^*$ is the set of finite sequences of signed terms. We will denote a typical element of $(\pm A)^*$ by $\langle\langle \sigma_1, a_1\rangle, \ldots, \langle \sigma_n, a_n\rangle\rangle$.

We extend the notion term to describe behaviors by traders in E-commerce protocols.

Definition 2. A strand space is a set Σ with a trace mapping $\mathrm{tr} : \Sigma \rightarrow (\pm A)^*$, where Σ is the set of strands.

Definition 3. A node is a pair $\langle s, i\rangle$, with $s \in \Sigma$ and i, an integer satisfying $1 \le i \le \mathrm{length}(\mathrm{tr}(s))$. The set of nodes is denoted by N. One will say the node $\langle s, i\rangle$ belongs to the strand s.

Definition 4. E is the set of edges. $n_1, n_2 \in N$ and $n_1 \rightarrow n_2 \in E$ means $\mathrm{term}(n_1) = +a$ and $\mathrm{term}(n_2) = -a$; $n_1 \Rightarrow n_2 \in E$ means n_1, n_2 occurs on the same strand with $\mathrm{index}(n_1) = \mathrm{index}(n_2) - 1$.

The actions available to penetrators are encoded in a set of penetrator traces that summarize the ability to discard messages, generate well-known messages, and piece messages together and apply cryptographic operations using keys that become available to him.

Definition 5. Penetrator model is defined by penetrator traces according to Dolve-Yao model assumptions:

M: text message: $\langle +t\rangle$, where $t \in T$,

F: flushing: $\langle -g\rangle$,

T: tee: $\langle -g, +g, +g\rangle$,

C: concatenation: $\langle -g, -h, +gh\rangle$,

S: separation into component: $\langle -gh, +g, +h\rangle$,

K: key: $\langle +K\rangle$, where $K \in K_P$, K_P is the set of keys initially known to the penetrator,

E: encryption: $\langle -K, -h, +\{h\}_k\rangle$,

D: decryption: $\langle -K^{-1}, -\{h\}_K, +h\rangle$.

Trace M means penetrator could send the messages they owned to the channel. While F means that they could obtain every message traveled in channel. T means repetition. C and S, respectively, represent joining and decomposition. E and D are the encryption and decryption. This set of penetrator traces ensures that the values that may be emitted by the penetrator are closed under joining, encryption, and the relevant "inverses" [5]. The trader model we proposed has the same form with penetrator model but models different abilities.

4. The Fairness Verification Using Strand Space Model

The paper focuses on E-commerce protocols containing third parties. The referred third party, here, are bank, arbiter, and trusted third party (TTP). Buyers and sellers in transactions exchange evidences, while third party guarantees fairness and effectiveness of the transactions. The model of third party is assumed to be regular and honest. This section is divided into three parts: the first part gives the formal definition of fairness and fairness evidences and then establishes the trader model and extends related concepts of the strand space model and, at last, gives a fair validation process based on those definitions.

4.1. The Formal Definition of Fairness. Fairness is one of the basic security properties that E-commerce protocols must meet; the acknowledged definition of fairness can be referred to [8–15]: an exchange is fair means that, at the end of the exchange, either each player receives the terms they expect or none receives any information about the other's terms. Fairness evidences are those terms that players expect in transaction. Fairness of E-commerce protocols includes the fair exchange of evidences and the fair evidences in exchange. The exchange in an electronic transaction is considered to be equivalent, so evidences exchanged in electronic transaction should be equivalent; otherwise, a trader may benefit from the other by forging unequal exchange evidences. To verify the fairness of the corresponding relation of evidences, the mapping relationship of defining evidences is in the following.

Definition 6. Traders B_1, $B_2 \in N$; N is the set of traders' identification; E_{B_1} and E_{B_2} are sets of fairness evidences belonging to traders B_1 and B_2, respectively. The evidence-corresponding-relation is a bijective function $f_{B_1} : E_{B_1} \rightarrow E_{B_2}$. Its inverse function is $f_{B_2} : E_{B_2} \rightarrow E_{B_1}$. $\forall e \in E_{B_1}$, $\exists! e' = f_{B_1}(e) \in E_{B_2}$, here, $f_{B_1}(e)$ means the evidences of B_2 which B_1 expects. The set of functions $f_{B_1}(e)$ is denoted by $F(E_{B_1}, E_{B_2})$. Similarly, the set from B_2 to B_1 is denoted by $F(E_{B_2}, E_{B_1})$.

Evidences exchanged in electronic transactions need to be mutually corresponded. Each trader has the ability to evaluate evidences, and the corresponding relations of traders' evidence act as an evaluation tool to measure equivalence of the evidence's value. We can verify whether traders have forged evidences from the formula $e' = f(e)$. If the formula is not established, it means that someone has forged evidences so that evidences exchanged are not equivalent. In our research, we assume that if a trader receives unequal evidence from others, they would like to ignore this message. This means that if a participant uses forged evidences in transaction, the other participant will refuse to accept the evidence.

The fair exchange of evidence, in the point of view of a trader, means that if A has not obtained the evidence from B, then B could not obtain A's evidence; else, if A obtained the evidence of B, whether B obtained evidence from A will not be considered anymore. Popular to say, the fairness means each participant of a transaction is not at a disadvantage. Combined with Definition 5, we define fairness as follows.

Definition 7. Given an E-commerce protocol Γ, here, $B_1, B_2 \in N$, $e_1 \in E_{B_1}$, $e_2 \in E_{B_2}$, E_{B_1}, and E_{B_2} are sets of evidences of B_1 and B_2. $f_1 \in F(E_{B_1}, E_{B_2})$ and $f_2 \in F(E_{B_2}, E_{B_1})$ are evidence-corresponding-relations between B_1 and B_2, respectively. Protocol Γ is fair, if and only if the following two conditions hold:

(1) at the end of an exchange, if trader B_1 has not obtained $f_1(e_1)$, then B_2 cannot get e_1;

(2) at the end of an exchange, if trader B_2 has not obtained $f_2(e_2)$, then B_1 cannot get e_2.

During the fairness validation process, we first generate a trader's strand which he has not got at the end of a transaction; the other trader acts as a penetrator to analyze whether the penetrator could obtain the evidence they want through a variety of deceptions or attacks from one of the strands above. The penetrator model here is different from the model used in the classical strand space model. Therefore, we need to model the behaviors of traders and extend related theories of the strand space model to describe E-commerce protocol and its properties.

4.2. Extension of Strand Space Model. Traders in E-commerce protocols are dishonest not like regular entities and penetrator entities in general security protocols. Apart from traders, all participants in E-commerce protocols are regular entities and perform in accordance with the agreement provisions. Because their sequence of events and entity model are constant, each of them has only limited numbers of strands. While traders may opt out of the transaction, or repeatedly use outdated evidences, like orders and electronic money, they may use a variety of behaviors to obtain benefits, thus the events sequence of these entities may not be complete in accordance with the protocol. Original strand space model uses the penetrator model to describe behaviors of attackers. A penetrator can intercept, send, forgery, tamper messages, and so forth. They can do almost everything except for resolving cryptographic algorithms, while traders are still bounded by the protocol. If using penetrator model to describe traders' behaviors, we might make a wrong judgment to a correct protocol. Thus, we need to establish a model to describe the behaviors of transactions between buyers and sellers. The capability of this model is between regular entities and penetrator entities. We call it trader model.

The study imitates the penetrator model in original strand space model to build the trader model and describe the behaviors of traders into atomic behaviors and their combinations. E-commerce protocols have little constrains for traders, similarly, traders cannot be completely separated from the protocol, so we use atomic behavior as well as some constraints on these atoms to describe the behaviors of traders. The model assumptions are showed as follows:

(1) traders can encrypt, decrypt, connect, and decompose message;

(2) traders can forge evidences and messages by using unequal evidences to get the other's interests;

(3) traders can send or receive messages belonging to this trader in protocol; only the messages are in specified format of protocol;

(4) both entities and messages in E-commerce protocols satisfy authentication and confidentiality, which being basic properties of security protocols; the protocol is insecure if these two properties are not met; fairness and nonrepudiation are based on the premise of security of the protocol; we should verify security before analysis fairness;

(5) the sequence of events in traders model needs not to be the order in accordance with the protocol; this assumption needs to describe the behaviors of traders in the form of atoms.

Considering E-commerce, protocols satisfy authentication and confidentiality; we add the identifiers of entities into the concept of the term, and terms in our model are represented as a 3-tuple. This modification eliminates the need for verification of authentication. Term is defined as follows.

Definition 8. Term is a 3-tuple $\langle \sigma, a, B\rangle$, where $\sigma \in \{+, -, \bigoplus\}$, $a \in A$, and $B \in N$. A is the message set and P is the set of entities' identifier. $\langle +, a, B\rangle$ means entity B sends message a; $\langle -, a, B\rangle$ means an entity receives message a from entity B; and $\langle \bigoplus, a, B\rangle$ means entity B owns message a. For convenience, abbreviate term $\langle \sigma, a, B\rangle$ as $\sigma a.B$; a is the unsigned portion of $\sigma a.B$.

We use symbol "$\bigoplus$" to meet the third assumption. Traders in E-commerce protocols cannot receive or send some messages, but can calculate. In the following, A_B is the set of terms that B could send or receive in protocol, and A_B^* is the set of messages B owns at the time.

Traders' atomic behaviors can be described by trace.

Definition 9. A trader trace is one of the following:

(1) M^*: text message: $\langle \bigoplus t.B, +t.B\rangle$, where $+t.B \in A_B$, $t \in A_B^*$, and $B \in N$; B is an entity's identifier; trace M^* expresses entity B sending message t;

(2) F^*: flushing: $\langle -t.C, \bigoplus t.B\rangle$, where $-t.C \in A_B$ and B, $C \in N$; trace F^* means entity B receives message t, and then B owns t;

(3) P^*: falsifying: $\langle \bigoplus e.B, \bigoplus f.B\rangle$, where $e \in E_B$ and $f \in F(E_B, E_{B'})$; E_B is the set of evidence belonging to entity B, $F(E_B, E_{B'})$ which is the set of evidence-corresponding-relations from B to B'; trace P^* expresses the situation that entity B forges evidence and its descriptions in protocol;

(4) C^*: concatenation: $\langle \bigoplus g.B, \bigoplus h.B, \bigoplus gh.B\rangle$, where $g, h \in A_B^*$ and $B \in N$; trace C^* means that if entity B owns messages g and h, then B owns gh;

(5) S^*: separation: $\langle \bigoplus gh.B, \bigoplus g.B, \bigoplus h.B\rangle$, where $gh \in A_B^*$ and $B \in N$; trace S^* means that if entity B owns message gh, then B owns g and h;

(6) K^*: key: $\langle \bigoplus K.B \rangle$, where $K \in K_B$, K_B is the set of keys B owns; trace K^* means entity B owns key K;

(7) E^*: encryption: $\langle \bigoplus K.B, \bigoplus h.B, \bigoplus \{h\}_K.B \rangle$, where $K \in K_B$, $h \in A_B^*$, and $B \in N$; trace E^*expresses the situation that entity B can encrypt h with key K;

(8) D^*: decryption: $\langle \bigoplus \{h\}_K.B, \bigoplus K^{-1}.B, \bigoplus h.B \rangle$, where $K \in K_B$, $\{h\}_K \in A_B^*$, and $B \in N$; trace D^* expresses the situation that entity B can decrypt $\{h\}_K$ into h if they own both $\{h\}_K$ and K^{-1}.

The strand of a trader is composed of nodes in traces M^* and F^*, expressing a sequence of events of the entity, and the remaining six traces are used to describe the entity's message space. Here, "+" and "−" denote messages delivered between entities, and "$\bigoplus$" stands for messages generated inside entities. The classical strand space model does not describe the concept "owned," because the number of messages sent or received by generators are unlimited. Messages which can be sent or received are owned by penetrators, while traders can only send or receive messages belonging to them. We propose the concept "owns" to describe the ability that a trader can process all the messages that they encountered. In summary, we modify the definition of terms to solve authentication and confidentiality, remove the tee trace T, add trace P^* and symbol "$\bigoplus$" to limit traders' abilities, and, finally, build the traders model.

Classical strand space model uses edges "$\rightarrow$" and "$\Rightarrow$" to describe the causal relationship between terms, where "$\rightarrow$" edge expresses delivering message between entities, and "$\Rightarrow$" edge describes an entity's state transition. Because traders' entities are semihonest, one may need more than one strand to describe their behaviors. Using atomic behaviors to build the trader model could describe the trader's behaviors nicely but is not conducive to express the causal relationship between nodes. For this purpose, we define owning set A_B^* and sending-receiving set A_B, where owning set A_B^* is a concept similar to ideal in the classical model and stands for messages that entity B has owned. The set is generated in a recursive way. Sending-receiving set A_B contains messages which B could send or receive and is fixed in a specific protocol. The owning set A_B^* is defined as follows.

Definition 10. A_B^* is an owning set of entity B; if $g \in A_B^*$, then

(1) $E_B \subseteq A_B^*$, $K_B \subseteq A_B^*$ and $F(E_B, E_{B'}) \subseteq A_B^*$, where E_B is the set of B's evidences, K_B is the set of B's keys, and $F(E_B, E_{B'})$ is the set of evidence-corresponding-relations from B to B';

(2) $\forall h \in A_B^*$, one has $gh \in A_B^*$;

(3) $\forall K \in K_B$, one has $\{g\}_K \in A_B^*$;

(4) If $g = g_1 g_2$, then $g_1, g_2 \in A_B^*$;

(5) If $g = \{g_1\}_K$ and $K^{-1} \in K_B$, then $g_1 \in A_B^*$.

Owning set A_B^* is used to describe all messages owned by entity B. The owning set of an entity will change with the state of the entity. When trader B receives messages S_B from their opponent, they could use those messages to obtain information from third party. If a strand of entity B includes flushing trace F^* : $\langle -h.C, \bigoplus h.B \rangle$, we have $h \in A_B^*$. We stipulate $A_B^*[h]$ as the owning set when entity B receives message h. If S_B is the set of messages entity B received from others at that time, then $A_B^*[S_B]$ will be all messages that entity B owns at the time they receive messages S_B.

Definition 11. Define owning set $A_B^*[S_B]$ as follows, where S_B is the set of messages received by entity B:

(1) $S_B \subseteq A_B^*[S_B]$, $E_B \subseteq A_B^*[S_B]$, $K_B \subseteq A_B^*[S_B]$, and $F(E_B, E_{B'}) \subseteq A_B^*[S_B]$, where E_B is the evidence set of entity B, K_B is the key set, and $F(E_B, E_{B'})$ is the set of evidence-corresponding-relations from B to B';

(2) $\forall g, h \in A_B^*[S_B]$, one has $gh \in A_B^*[S_B]$;

(3) $\forall g \in A_B^*[S_B]$, $\forall K \in K_B$, one has $\{g\}_K \in A_B^*[S_B]$;

(4) If $g = g_1 g_2 \in A_B^*[S_B]$, then $g_1, g_2 \in A_B^*[S_B]$;

(5) If $g = \{g_1\}_K \in A_B^*[S_B]$ and $K^{-1} \in K_B$, then $g_1 \in A_B^*[S_B]$;

(6) If $-a.B \in A_T \wedge +a.T \in A_B \wedge a \in A_B^*[S_B]$, in which there is a sequence $\langle -a.B, +c.\mathrm{TTP} \rangle$ in T's strand, where $-c.\mathrm{TTP} \in A_B$, then $c \in A_B^*[S_B]$.

The sixth rule expresses the situation in which trader B sends message a to third party T after they received messages S_B, and then T sends message c to B; thus entity B owns c. The third party here refers to all participators except traders in E-commerce protocol, including banks and arbitration institutions, as well as the trusted third party (TTP). Third party in E-commerce protocols is regular entity, and its strands are regular strands. For convenience, we use Definition 7 to define terms of regular strands. Edges between nodes remain unchanged, which are defined by "$\rightarrow$" and "$\Rightarrow$" in classical strands space model. In addition, we add a status node [15] at the end of each strand, which does not send or receive messages, only to be used to express the end of a sequence of events. In the description of strands, we follow the way Fröschle did in [15], using hollow circle and solid circle, respectively, to express the node of termination status and normal nodes.

4.3. Fairness Validation Process. Based on the fairness definition and the extended strand space model above, we put forward a formal method of fairness verification. The verification process is shown in Figure 1.

Build $F(E_{B_1}, E_{B_2})$: each E-commerce protocol contains some description messages. These messages express the relationship of evidences which traders exchange. Therefore, we build evidence-corresponding-relations base on them. Because traders can forge evidences and description messages, we use set $F(E_B, E_{B'})$ to express all goods descriptions traders can forge.

Build T strand: all entities except traders in E-commerce protocol are regular entities; thus, we use a regular strand to model those entities. Each T has a fixed number of strands. We need to select T strands according to the implementation of protocols. For convenience, the study defines terms in

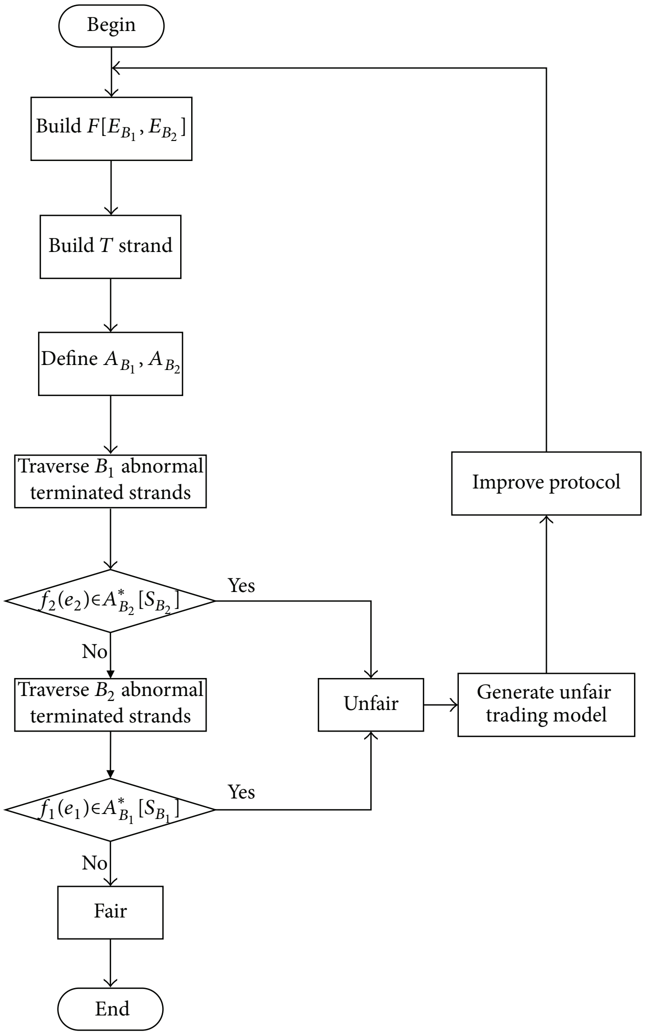
Begin
Build $F[E_{B_1}, E_{B_2}]$
Build T strand
Define A_{B_1}, A_{B_2}
Traverse B_1 abnormal terminated strands
Improve protocol
$f_2(e_2) \in A^*_{B_2}[S_{B_2}]$
Yes
No
Traverse B_2 abnormal terminated strands
Unfair
Generate unfair trading model
$f_1(e_1) \in A^*_{B_1}[S_{B_1}]$
Yes
No
Fair
End

Figure 1: Fairness verification schemes.

T strand by Definition 7 and adds a status node in the end of the strand.

Define A_{B_1} and A_{B_2}: sending-receiving sets A_{B_1} and A_{B_2} of traders B_1 and B_2 are fixed. Sending messages of a trader are all messages that protocol formatted, while receiving messages is not only protocol formatted but also meets the needs of their own interest.

Traverse traders' abnormal-terminated strands: abnormal terminated means a trader has not got the evidences they want at the end of a transaction. We first establish an abnormal-terminated strand of a trader with regular entities model and then detect whether the other trader could obtain the evidences they want. E-commerce protocols may be terminated abnormally in different running stages, and each trader may have more than one abnormal-terminated strand. We use regular strand to model trader's abnormal-terminated event sequence and detect whether a dishonest trader could gain evidences from an honest trader. Benefiting from assumptions, there will be a finite number of abnormal-terminated strands of each trader, which makes the detection process to be terminated possible.

Consider $f_1(e_1) \in A^*_{B_1}[S_{B_2}]$: $f_1(e_1)$ stands for the evidence B_1 expected; S_{B_2} stands for messages B_1 received from an abnormal-terminated strand of B_2; $A^*_{B_1}[S_{B_2}]$ stands for all messages B_1 owned after they receive messages S_{B_2}. $f_1(e_1) \in A^*_{B_1}[S_{B_2}]$ means that B_1 got the evidence they want, while B_2 did not; then, the protocol is unfair. The difficulty of this step is how to generate set $A^*_{B_1}[S_{B_2}]$. The recursively defined set $A^*_{B_1}[S_{B_2}]$ is an infinite set, and judgments about formula $f_1(e_1) \in A^*_{B_1}[S_{B_2}]$ need reasoning and induction. The specific process will be given in the next chapter.

Based on the strand space model, we use graphs to illustrate the process of an implementation of a protocol. If the protocol is unfair, an unfair execution process will be given by the strand space model in an intuitive way. For this reason, we modify the protocol and then verify the modified protocol again, until it is fair.

5. Case Analysis

We use the EMH protocol to test the fairness validation method which we propose. EMH protocol is an offline TTP electronic payment protocol proposed by Alaraj and Munro in 2007 [19]. The purpose of this protocol is to exchange a digital product (D) with a payment (P) between a customer (C) and a merchant (M). When we say that the protocol is fair, it means that, at the end of a transaction, either M gets P and C gets D or both of them do not get any message and vice versa. Using this protocol for the experiment can help to introduce the fairness verification process in detail and can explain the reason why we define fairness and extend the strand space model in such a way vividly.

5.1. Protocol Description. Identifier and symbol description includes the following:

C: customer,

M: merchant,

TTP: the trusted third party,

CB: the customer's bank, having the case that while the CB can also be considered as a TTP, TTP and CB are considered as third parties in our verification process and are modeled by regular entity,

D: digital product,

P: buyer's payment voucher, where D and P are the so-called fairness evidence in our model,

Desc: description of digital product, which is the link between D and P, where we build evidence corresponding relationship based on Desc, where $\text{Desc}_C(P)$ and $\text{Desc}_M(D)$ represent the evidence corresponding relationship between C and M, respectively,

$h(x)$: a strong collision-resistant one-way hash function, such as MD5,

PK_a: RSA public key of entity a,

SK_a: RSA private key of entity a,

P_cert: payment's certificate that is issued by the CB, with the contents of P_cert being d, description of payment (the amount), hp, hash value of payment, hep, hash value of encrypted payment with PK_a, and Sig_{CB}, CB's signature on P_cert,

Cert_{CT}: the certificate for the shared public key between C and TTP, which is issued by the TTP,

$\text{Enc}_{PK_a}(X) = \{X\}_{PK_a}$: an RSA encryption of X using the public key PK_a,

$\text{Dec}_{SK_a}(Y) = \{Y\}_{SK_a} = X$: an RSA decryption of Y using the private key SK_a,

$\text{Enc}_{PK_a}(X) = \{X\}_{PK_a}$: the RSA signature of party A, that is, encryption of the hash value of X using the private key SK_a,

$A \rightarrow B : X$: A which sends message X to B,

$X \| Y$: concatenation of messages X and Y.

EMH protocol is divided into three phases: the preexchange phase, the exchange phase, and the dispute settlement phase; details are given as follows.

(1) The preexchange phase includes the following:

mes1: $\text{TTP} \rightarrow C : \text{Cert}_{\text{CT}}$;

mes2: $\text{CB} \rightarrow C : P\_\text{cert}$.

(2) The exchange phase includes the following:

mes3: $C \rightarrow M : \text{Desc}\|\{P\}_{PK_{\text{CT}}}\|P\_\text{cert}\|\text{Cert}_{\text{CT}}\|\text{Sig}_C(P)$;

mes4: $M \rightarrow C : \{D\}_{PK_{\text{CT}}}\|\text{Sig}_M(D)$;

mes5: $C \rightarrow M : SK_{\text{CT}}$.

(3) The dispute settlement phase includes the following:

mes6: $M \rightarrow \text{TTP} : \text{Desc}\|\{P\}_{PK_{\text{CT}}}\|P\_\text{cert}\|\text{Cert}_{\text{CT}}\|\text{Sig}_C(P)\|\{D\}_{PK_{\text{CT}}}\|\text{Sig}_M(D)$;

mes7: $\text{TTP} \rightarrow C : \{D\}_{PK_{\text{CT}}}\|\text{Sig}_M(D)$;

mes8: $\text{TTP} \rightarrow M : SK_{\text{CT}}$.

The preexchange phase aims to award certificates from TTP and CB to C and do nothing between C and M, so we omit this phase in the verification process, considering only the last two stages.

5.2. Verification Process. To verify the fairness of EMH protocol, we need to prove that $f_C(P) \notin A_C^*[S_C]$ and $f_M(D) \notin A_M^*[S_M]$ both are true. First, we verify $f_C(P) \notin A_C^*[S_C]$; in the following proof, we could get $f_C(P) \in A_C^*[S_C]$.

5.2.1. Build the Set of Evidence Corresponding Relations. Define bijective functions $\text{Desc}_C : E_C \rightarrow E_M$ and $\text{Desc}_M : E_M \rightarrow E_C$, where $\text{Desc}_C \in F(E_C, E_M)$ and $\text{Desc}_M \in F(E_M, E_C)$. $F(E_C, E_M)$ and $F(E_M, E_C)$ are sets of evidence corresponding relations belonging to M and C, respectively. For $P \in E_C, D \in E_M$, and $\exists$ $\text{Desc} \in F(E_C, E_M)$ and its inverse $\text{Desc}^{-1} \in F(E_M, E_C)$, $D = \text{Desc}(P) \wedge P = \text{Desc}^{-1}(D)$.

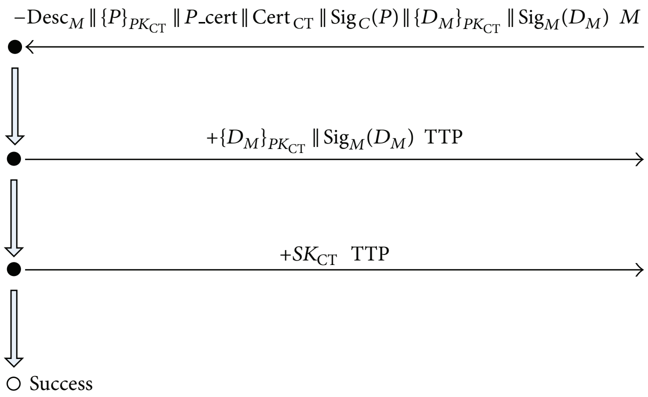
$-\text{Desc}_M\|\{P\}_{PK_{\text{CT}}}\|P\_\text{cert}\|\text{Cert}_{\text{CT}}\|\text{Sig}_C(P)\|\{D_M\}_{PK_{\text{CT}}}\|\text{Sig}_M(D_M)$ M
$+\{D_M\}_{PK_{\text{CT}}}\|\text{Sig}_M(D_M)$ TTP
$+SK_{\text{CT}}$ TTP
Success

Figure 2: Regular strand of TTP.

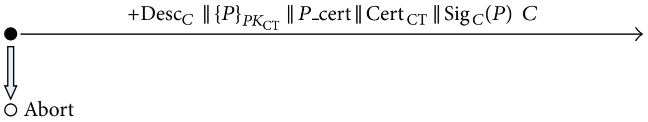
$+\text{Desc}_C\|\{P\}_{PK_{\text{CT}}}\|P\_\text{cert}\|\text{Cert}_{\text{CT}}\|\text{Sig}_C(P)$ C
Abort

Figure 3: An abnormal-terminated strand of trader C.

5.2.2. TTP Strands. TTP are regular entities, building its strands directly. Figure 2 is a regular strand of TTP.

TTP checks whether $D_M = \text{Desc}_M(P)$ is established; if so, it sends message $\{D_M\}_{PK_{\text{CT}}}\|\text{Sig}_M(D_M)$ to C and sends SK_{CT} to M.

5.2.3. Determine Sending-Receiving Set

Consider

$$\begin{aligned} A_C = \Big\{ & +\text{Desc}_C \Big\| \{P\}_{PK_{\text{CT}}} \| P\_\text{cert} \Big\| \text{Cert}_{\text{CT}} \Big\| \text{Sig}_C(P).C, \\ & - \{\text{Desc}_C(P)\}_{PK_{\text{CT}}} \Big\| \text{Sig}_M(\text{Desc}_C(P)).M, \\ & - \{\text{Desc}_C(P)\}_{PK_{\text{CT}}} \Big\| \text{Sig}_M(\text{Desc}_C(P)).\text{TTP}, \\ & +SK_{\text{CT}}.M \Big\} \\ A_M = \Big\{ & -\text{Desc}_C \Big\| \{P\}_{PK_{\text{CT}}} \| P\_\text{cert} \Big\| \text{Cert}_{\text{CT}} \Big\| \text{Sig}_C(P).C, \\ & + \{D_M\}_{PK_{\text{CT}}} \Big\| \text{Sig}_M(D_M).M, \\ & + \text{Desc}_M \Big\| \{P\}_{PK_{\text{CT}}} \| P\_\text{cert} \Big\| \text{Cert}_{\text{CT}} \Big\| \text{Sig}_C(P) \| \\ & \times \{D_M\}_{PK_{\text{CT}}} \Big\| \text{Sig}_M(D_M).M, \\ & - SK_{\text{CT}}.C, -SK_{\text{CT}}.\text{TTP} \Big\}. \end{aligned} \tag{1}$$

5.2.4. Establish Abnormal-Terminated Strands. We first analyze whether trader M could obtain the evidence P when trader C is abnormal-terminated strand (Figure 3).

5.2.5. Build Traders Model. We build M's trader model by the abnormal-terminated strand of C. Assuming that

M could obtain P; that is, there exists a flushing trace F^* : $\langle -SK_{\text{CT}}.\text{TTP}\rangle$ or $\langle -SK_{\text{CT}}.C\rangle$. Because the abnormal-terminated strand of C does not contain node $\langle +SK_{\text{CT}}.\text{TTP}\rangle$, we ignore the situation F^* : $\langle -SK_{\text{CT}}.C\rangle$.

Suppose the situation where there exists a node $\langle -SK_{\text{CT}}.\text{TTP}\rangle$ in M's strand. Because TTP strand is regular, there exists a node $\langle +\text{Desc}\|\{P\}_{PK_{\text{CT}}}\|P\_\text{cert}\|\text{Cert}_{\text{CT}}\|\text{Sig}_C(P)\|\{D_M\}_{PK_{\text{CT}}}\|\text{Sig}_M(D_M).M\rangle$ in M's strand.

And because $-\text{Desc}\|\{P\}_{PK_{\text{CT}}}\|P\_\text{cert}\|\text{Cert}_{\text{CT}}\|\text{Sig}_C(P).C \in A_M$, we have $S_M = \{\text{Desc}\|\{P\}_{PK_{\text{CT}}}\|P\_\text{cert}\|\text{Cert}_{\text{CT}}\|\text{Sig}_C(P).C\}$. Then, we analyze M's owning set $A^*_M[S_M]$.

By Definition 11, we have

$$\{P\}_{PK_{\text{CT}}} \|P\_\text{cert} \|\text{Cert}_{\text{CT}} \|\text{Sig}_C (P) \in A^*_M [S_M] \tag{2}$$

$$D_M = \text{Desc}_M (P), \text{Desc}_M, D_M \in A^*_M [S_M] \tag{3}$$

$$\{D_M\}_{PK_{\text{CT}}}, \text{Sig}_M (D_M) \in A^*_M [S_M] \tag{4}$$

$$\text{Desc}_M \left\|\{P\}_{PK_{\text{CT}}} \|P\_\text{cert} \|\text{Cert}_{\text{CT}} \|\text{Sig}_C (P) \right\|\{D_M\}_{PK_{\text{CT}}} \|\text{Sig}_M (D_M) \in A^*_M [S_M]. \tag{5}$$

Because $\text{Desc}_M\|\{P\}_{PK_{\text{CT}}}\|P\_\text{cert}\|\text{Cert}_{\text{CT}}\|\text{Sig}_C(P)\|\{D_M\}_{PK_{\text{CT}}}\|\text{Sig}_M(D_M).M \in A_M$; there exists a text message trace M^* in M's strand, where M^* is $\langle +\text{Desc}\|\{P\}_{PK_{\text{CT}}}\|P\_\text{cert}\|\text{Cert}_{\text{CT}}\|\text{Sig}_C(P)\|\{D_M\}_{PK_{\text{CT}}}\|\text{Sig}_M(D_M).M\rangle$, which means that the assumption holds; trader M can obtain the evidence P. The protocol is unfair. Figure 4 is the unfair execution process described by strand space model.

5.2.6. Unfair Analysis and Protocol Improvements. Protocol is unfair because the TTP cannot accurately determine whether M has forged evidence. Trader M can obtain SK_{CT} from TTP by forging evidence D and the description message Desc and then get C's evidence. We modify protocol; thus the TTP could compare description information of the two parties and do a fair judgment.

Here is the modified protocol.

(1) The preexchange phase includes the following:

mes1: TTP $\rightarrow$ C : Cert_{CT};

mes2: CB $\rightarrow$ C : P_cert.

(2) The exchange phase includes the following:

mes3: $C \rightarrow M$: $\text{Desc}\|\{P\|\text{Desc}\}_{PK_{\text{CT}}}\|P\_\text{cert}\|\text{Cert}_{\text{CT}}\|\text{Sig}_C(P\|\text{Desc})$;

mes4: $M \rightarrow C : \{D\}_{PK_{\text{CT}}}\|\text{Sig}_M(D)$;

mes5: $C \rightarrow M : SK_{\text{CT}}$.

(3) The dispute settlement phase includes the following:

mes6: $M \rightarrow$ TTP : $\text{Desc}\|\{P\|\text{Desc}\}_{PK_{\text{CT}}}\|P_{\text{cert}}\|\text{Cert}_{\text{CT}}\|\text{Sig}_C(P\|\text{Desc})\ \|\{D\}_{PK_{\text{CT}}}\|\text{Sig}_M(D)$;

mes7: TTP $\rightarrow$ $C : \{D\}_{PK_{\text{CT}}}\|\text{Sig}_M(D)$;

mes8: TTP $\rightarrow$ $M : SK_{\text{CT}}$.

Based on the unfair reasons above, we improve message 3 and message 6. TTP will determine $\text{Desc}_C = \text{Desc}_M \wedge D_M = \text{Desc}_M(P)$ first, when they receive a request message from M. If the formula is true, TTP sends SK_{CT} to M and D_M to C.

By verifying the fairness of the modified protocol continuously, we then establish abnormal-terminated strands of M and C, respectively, and judge each of them. The concreted analysis process of improved protocol will not be described here. Figure 5 is the model description of the modified protocol in the same situation. From it, we can know that M has to send their evidence D in order to obtain P, because the TTP has more discrimination capability.

The verification steps are the same as mentioned above, so we did not propose the detailed description here.

5.3. Analysis of Experimental Results. By using EMH protocol to test the fair authentication method proposed in this paper, we draw some conclusions: first, the method can verify the fairness of E-commerce protocols effectively and give an accurate judgment about the fair exchange of evidences and the fair evidences in exchange; second, the method generates a finite number of abnormal-terminated strands and builds the trader model explicitly by inductive reasoning, which enables the verification process to be terminated; in addition, it has a practical value in design and improvement of E-commerce protocols.

6. Conclusion

This paper proposes a formal definition of fairness as well as a new method to verify the fairness of E-commerce protocols. The trader model we build here differs from the Dolev-Yao penetrator model. Because it is established according to the E-commerce trading behaviors, it can be better to reflect the behaviors of entities in E-commerce protocols. The evidence-corresponding-relations defined by a bijective function can describe the equivalent relations of traders' evidences and give a method to determine whether someone has forged evidences in transaction. The formal definition of fairness is defined from the perspective of traders, which helps to

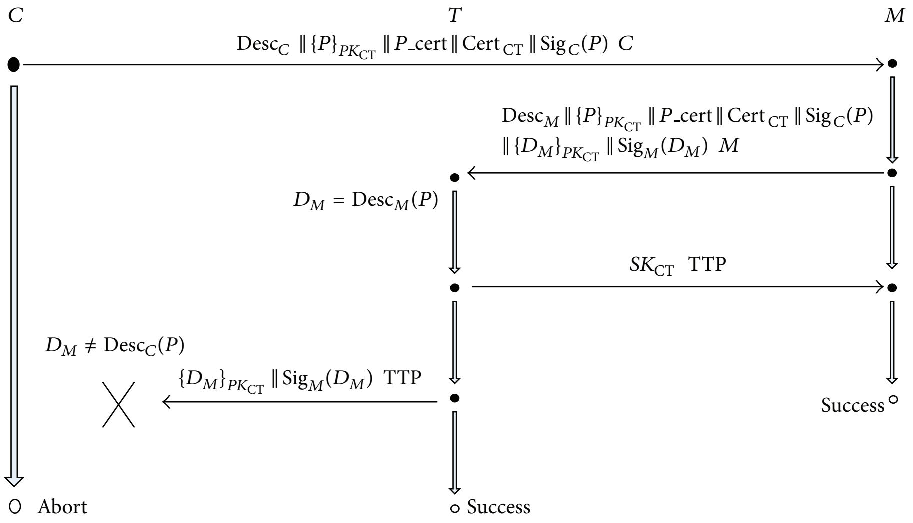
C
T
M
$\mathrm{Desc}_C \parallel \{P\}_{PK_{\mathrm{CT}}} \parallel P\_\mathrm{cert} \parallel \mathrm{Cert}_{\mathrm{CT}} \parallel \mathrm{Sig}_C(P)$ C
$\mathrm{Desc}_M \parallel \{P\}_{PK_{\mathrm{CT}}} \parallel P\_\mathrm{cert} \parallel \mathrm{Cert}_{\mathrm{CT}} \parallel \mathrm{Sig}_C(P)$
$\parallel \{D_M\}_{PK_{\mathrm{CT}}} \parallel \mathrm{Sig}_M(D_M)$ M
$D_M = \mathrm{Desc}_M(P)$
SK_{CT} TTP
$D_M \neq \mathrm{Desc}_C(P)$
$\{D_M\}_{PK_{\mathrm{CT}}} \parallel \mathrm{Sig}_M(D_M)$ TTP
Success
Abort
Success

Figure 4: An unfair case of the protocol.

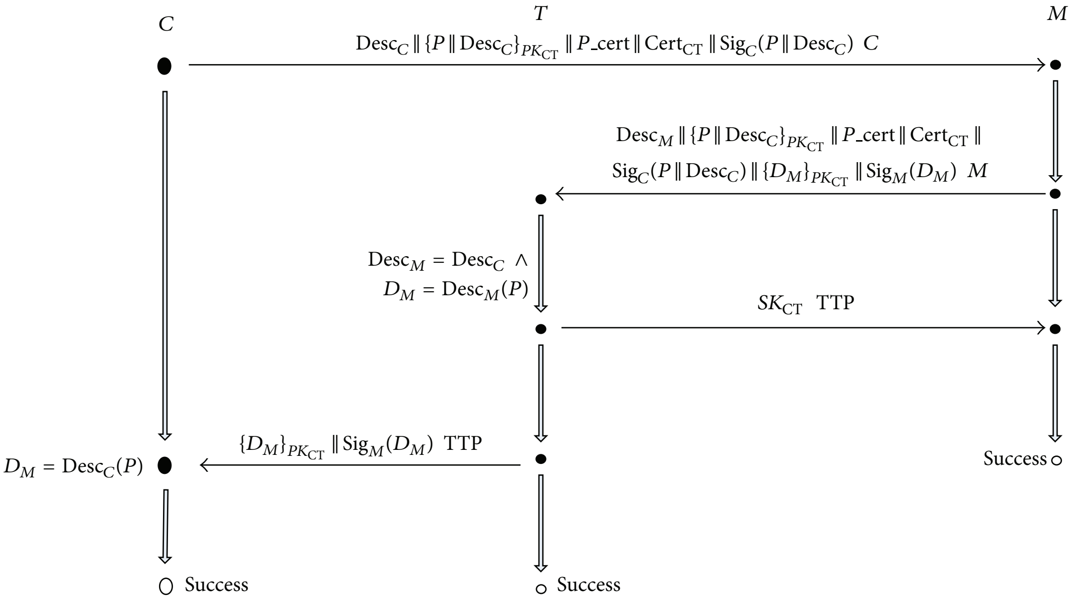
C
T
M
$\mathrm{Desc}_C \parallel \{P \parallel \mathrm{Desc}_C\}_{PK_{\mathrm{CT}}} \parallel P\_\mathrm{cert} \parallel \mathrm{Cert}_{\mathrm{CT}} \parallel \mathrm{Sig}_C(P \parallel \mathrm{Desc}_C)$ C
$\mathrm{Desc}_M \parallel \{P \parallel \mathrm{Desc}_C\}_{PK_{\mathrm{CT}}} \parallel P\_\mathrm{cert} \parallel \mathrm{Cert}_{\mathrm{CT}} \parallel$
$\mathrm{Sig}_C(P \parallel \mathrm{Desc}_C) \parallel \{D_M\}_{PK_{\mathrm{CT}}} \parallel \mathrm{Sig}_M(D_M)$ M
$\mathrm{Desc}_M = \mathrm{Desc}_C \wedge$
$D_M = \mathrm{Desc}_M(P)$
SK_{CT} TTP
$\{D_M\}_{PK_{\mathrm{CT}}} \parallel \mathrm{Sig}_M(D_M)$ TTP
$D_M = \mathrm{Desc}_C(P)$
Success
Success
Success

Figure 5: Analysis of the modified protocol.

reconcile with model assumptions of traders. We use a regular strand to model the third party and trader abnormal-terminated strands, propose the trader model to detect whether a participant can obtain regular entities' evidences, and thus complete the fairness validation. This method avoids the verification of nonrepudiation, and can verify fairness of E-commerce protocols including third-parties. Besides, it neither needs to track all statuses of protocol execution nor traverses all strands of traders. The current work is limited to manual derivation, and we will strive to the automatic verification in future.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported in part by the National Science Foundation of China (nos. 91118003, 61272106, and 61003080) and by 985 funds of Tianjin University.

References

[1] R. Kailar, "Accountability in electronic commerce protocols," *IEEE Transactions on Software Engineering*, vol. 22, no. 5, pp. 313–328, 1996.

[2] J. Zhou and D. Gollmann, "Towards verification of non-repudiation protocols," in *Proceedings of the International Refinement Workshop and Formal Methods Pacific*, pp. 370–380, Canberra, Australia, 1998.

[3] S. Schneider, "Formal analysis of a non-repudiation protocol," in *Proceedings of the 11th IEEE Computer Security Foundations Workshop (CSFW '98)*, pp. 54–65, June 1998.

[4] F. J. T. Fabrega, J. C. Herzog, and J. D. Guttman, "Honest ideals on strand spaces," in *Proceedings of the 11th IEEE Computer Security Foundations Workshop (CSFW '98)*, pp. 66–77, June 1998.

[5] F. J. T. Fabrega, J. C. Herzog, and J. D. Guttman, "Strand spaces: why is a security protocol correct?" in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 160–171, May 1998.

[6] F. J. T. Fabrega, J. C. Herzog, and J. D. Guttman, "Mixed strand spaces," in *Proceedings of the 12th IEEE Computer Security Foundations Workshop (CSFW '99)*, pp. 72–82, June 1999.

[7] R. M. Amadio and W. Charatonik, "On name generation and set-based analysis in the Dolev-Yao model," in *CONCUR 2002—Concurrency Theory*, pp. 499–514, Springer, Berlin, Germany, 2002.

[8] H. Pagnia and F. C. Gartner, On the Impossibility of Fair Exchange Without a Trusted Third Party TUD-BS-1999-02, Department of Computer Science, Darmstadt University of Technology, 1999.

[9] N. Asokan, *Fairness in electronic commerce [Ph.D. thesis]*, University of Waterloo, 1998.

[10] J. Yang and H.-F. Deng, "Security electronic commerce protocol by the third kind entities," in *Proceedings of the International Conference on Machine Learning and Cybernetics*, pp. 4438–4443, Dalian, China, August 2006.

[11] H. Wang, J. Ma, and B. Chen, "Formal analysis of fairness in E-payment protocol based on strand space," in *Web Information Systems and Mining*, pp. 469–478, Springer, Berlin, Germany, 2009.

[12] W. Liu, J. Yang, and Z. Li, "Fairness analysis of electronic commerce protocol based on strand space," in *Proceedings of the 5th International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP '09)*, pp. 714–717, Kyoto, Japan, September 2009.

[13] W. Xu, D.-Y. Wu, Y. Ma, and N. Liu, "A formal method for analyzing fair exchange protocols," in *Proceedings of the WASE International Conference on Information Engineering (ICIE '09)*, pp. 117–120, Taiyuan, China, July 2009.

[14] Q. Zhang, K. Markantonakis, and K. Mayes, "A practical fair-exchange e-payment protocol for anonymous purchase and physical delivery," in *Proceedings of the IEEE International Conference on Computer Systems and Applications (AICCSA '06)*, pp. 851–858, Sharjah, UAE, March 2006.

[15] S. Fröschle, "Adding Branching to the Strand Space Model," *Electronic Notes in Theoretical Computer Science*, vol. 242, no. 1, pp. 139–159, 2009.

[16] J. D. Guttman, "State and progress in strand spaces: proving fair exchange," *Journal of Automated Reasoning*, vol. 48, no. 2, pp. 159–195, 2012.

[17] J. D. Guttman, "Fair exchange in strand spaces," in *Proceedings 7th International Workshop on Security Issues in Concurrency (SecCo '09)*, pp. 46–60, Bologna, Italy, September 2009.

[18] S.-H. Tian, L.-J. Chen, and J.-R. Li, "Fairness analysis of electronic payment protocol based on offline TTP," *Journal of Computer Applications*, vol. 29, no. 7, pp. 1839–1843, 2009.

[19] A. Alaraj and M. Munro, "An efficient fair exchange protocol that enforces the merchant to be honest," in *Proceedings of the 3rd International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom '07)*, pp. 196–202, New York, NY, USA, November 2007.