ARTICLE IN PRESS
Future Generation Computer Systems ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

A game-theoretic model and analysis of data exchange protocols for Internet of Things in clouds

Xiuting Tao [a], Guoqiang Li [a,*], Daniel Sun [b], Hongming Cai [a]

[a] School of Software, Shanghai Jiao Tong University, Shanghai, China
[b] Data61, CSIRO, Australia

HIGHLIGHTS

- The paper proposes an extensive game based model for behavior analysis of IoT protocols. Analysis techniques are given, with aim to the rationality and fairness properties.
- The properties are proposed to verify the security of business in the cloud computing.
- To verify the properties, a tree analysis method and a linear algorithm are described. As a case study, some flaws of the ASW protocol are identified.

ARTICLE INFO

Article history:
Received 25 May 2016
Received in revised form
20 September 2016
Accepted 22 December 2016
Available online xxxx

Keywords:
Exchange protocols
Game theory
Rationality
Fairness

ABSTRACT

Big data, Internet of things (IoT), and cloud computing have been recognized a family of technologies for a connected world. Besides hailed hope for the future, there are also challenges to security due to complexity and unpredictability of the Internet, clouds, and data. One of the challenges is information and data exchange, for example, identifying untrustworthy cloud users and analyzing abnormal user behavior during information exchange. This paper addresses exchange mechanism, which is a useful theoretic basis to make secure electronic commerce and electronic business transactions possible. To ensure and verify the property of fairness, a crucial property of exchange mechanism, this paper proposes a specific model for behavior analysis based on the extensive game with imperfect information. Rationality and fairness properties are built in the corresponding game and the game tree. To verify the properties, a tree analysis method is proposed, and a linear time algorithm is given. As a case study, some flaws of the ASW protocol are found.

1. Introduction

The number of physical objects connected to the Internet is growing at an amazing rate. The Internet of Things (IoT) is a novel paradigm that a variety of things or objects are able to interact with each other and cooperate with their neighbors to reach common goals. There are a lot of domains and environments in which the IoT will play a remarkable role and improve the quality of our lives in the near future, including domotics, transportation, healthcare, and industrial automation [1]. In the IoT, Internet protocols are crucial in the communication of exchange message. For the IoT protocols, security and privacy play a significant role in all markets globally due to the sensitivity of consumers privacy [2].

As the amount of data and information increases, the big data analyzing and informs and supports decision making becomes increasingly important. Big Data analytics is one of the core technologies used by businesses today for decision making and applying game theory data science for strategic decision making, is definitely an intelligent move that will help enterprises predict likely outcomes for businesses, individuals and societies. Games theory is the study of strategic decision making, and games provide alternative means of sharing information and knowledge and participating in decision making. In [3], game theory is applied to model the mechanisms for big data analytics and decision making in the field of geosciences and remote sensing.

Cloud computing is defined as an access model to an on-demand network of shared configurable computing sources such as networks, servers, warehouses, applications, and services. With the rapid development of cloud computing, it brings people to enjoy the convenience such that more lower costs, improved operational efficiency and so on. However more severe information

* Corresponding author.
E-mail address: li.g@sjtu.edu.cn (G. Li).

http://dx.doi.org/10.1016/j.future.2016.12.030

Please cite this article in press as: X. Tao, et al., A game-theoretic model and analysis of data exchange protocols for Internet of Things in clouds, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.12.030

security challenges are faced. In the open cloud computing, attackers have a greater temptation, like opening access interface of the cloud, end-users can directly use the cloud and cloud service. For example, Amazon web service does not have any access rights to customer instances and cannot log into the guest operating system; customers may utilize certificate-based SSHv2 to access the virtual instance to share service with others [4]. For the open net work, the middle attack in which the attacker makes independent connections with the victims and relays messages between them, may bring more serious broken than the current use of the Internet to share resources [5]. The behavior of end-user is an important part in the credibility of cloud computing security. Authentication technology is relatively mature, but does not prevent malicious destruction of legal status. The analysis of cloud end-user behavior is a research focus for the cloud computing.

In the open cloud computing, there are some protocol mechanisms and resource allocation mechanisms to exchange and share the electronic resource. Game theory was considered as a formal model for protocol [6–9] and resource allocation [10–14] frequently in recent years. In [15], Chuang Ma had considered an IPv6 control protocol based on game theory to maximize the throughput. J.M. Estevez-Tapiador adopted game theory to model the information of protocols [16]. Chenming Li used a complete information dynamic game model for an automated negotiation protocol [17]. Tian Jun gave a game theory model based on carrier sense multiple access protocol in wireless network [18].

An exchange protocol [19] is fair if at the end of exchange, either each participant receives expected items or neither two receives any useful formation about the other's items. Such protocol example includes *signing of electronic contracts*, *certified e-mail delivery*, and *purchase of network delivered services* [20,21]. Due to difficulties in understanding fairness, there are some definitions given by researchers [22,23].

In [24], the notion of rational exchange is introduced by Syverson in 1998. The rationality is another property of protocol which can replace the fairness to resolve the problem. A rational exchange protocol provides incentives so that rational (self-interested) parties have more reason to follow the protocol faithfully than to deviate from it.

For the rationality and fairness properties, there were some works [25,26,11]. Furthermore, Gu applied game theory and process algebra to analyze the fair exchange protocols [27]. The basic idea of game-based model for fair exchange protocols was offered in [28]. They did not consider the unreliability of network when modeling fairness.

In this paper, an extensive game with imperfect information is adopted to model exchange protocols. The participants are taken as rational players; the communicating messages are actions of players. The rationality and fairness properties are defined on the payoffs of players. An analysis method of the corresponding game tree with its a linear time algorithm is presented to compute weights of leaves on the tree.

The rest of this paper is organized as follows: Section 2 presents some basic concepts of an extensive game in game theory. Section 3 describes how to transform an exchange protocol to an extensive game with perfect information. A formal model of a rational exchange protocol on the subgame perfect equilibrium is presented in Section 4. Section 5 analyzes the Syverson protocol in the model. The relationship with Buttyán's model is shown in Section 6. Sections 7 and 8 consider the fairness property and analyze fair exchange protocol. Section 9 gives the conclusion of this paper.

2. Extensive game

This section introduces basic definitions of extensive game theory [29,30] that will be used later.

Definition 1 (*Extensive Game*)**.** An extensive game with information is a tuple $\Gamma = \langle N, H, P, (\succeq)_{i\in N}\rangle$, where:

- N is a set of players, and i is an element of the set N;
- H is a set of action sequences history that satisfies the following three properties:
  1. the empty sequence $\emptyset$ is an element the set H,
  2. if $(a^k)_{k=1,\dots,K} \in H$ (where K may be infinite) and $L < K$, then $(a^k)_{k=1,\dots,L} \in H$, and
  3. if an infinite action sequence $(a_k)_{k=1}^{\infty}$ satisfies $(a_k)_{k=1,\dots,I} \in H$ for every positive integer I, then $(a_k)_{k=1}^{\infty} \in H$.

  Each member of H is a history, and each component of a history is an action $a \in A$, where A is the action set of players. A history $(a^k)_{k=1,\dots,K} \in H$ is terminal if it is infinite, or if there is no a^{K+1} such that $(a^k)_{k=1,\dots,K+1} \in H$. The set of terminal histories is denoted by Z.
- P is a player function that assigns to each non-terminal history (the set is denoted by $H\backslash Z$) a member of N. In other words, $P(h)_{h\in(H\backslash Z)}$ assigns the player who takes an action after the history h.
- $(\succeq)_{i\in N}$ is a preference relation for each player $i \in N$ on Z.

The definition of the subgame of the extensive game is given as following,

Definition 2 (*Subgame*)**.** A subgame of an extensive game $\Gamma = \langle N, H, P, (\succeq)_{i\in N}\rangle$ that follows the history h is an extensive game $\Gamma(h) = \langle N, H|_h, P|_h, (\succeq)_{i\in N}|_h\rangle$, where $H|_h$ is the set of sequences h' of actions for which $(h, h') \in H$. $h' \in H|_h$ for each $\succeq_i|_h$ and $h' \succeq_i |_h h''$ is defined by $(h, h') \succeq_i (h, h'')$, if and only if $h' \in H|_h$.

The extensive game is an explicit description of the sequential structure of the decision problems encountered by the players in a strategic situation. The subgame perfect equilibrium of an extensive game is given as following,

Definition 3 (*Subgame Perfect Equilibrium*)**.** A subgame perfect equilibrium of an extensive game is a strategy profile s^* such that for every player $i \in N$ and every non-terminal $h \in H\backslash Z$, for which $P(h) = i$, it has

$$O_h(s^*_{-i}|_h, s^*_i|_h) \succeq_i |_h O_h(s^*_{-i}|_h, s_i)$$

for every strategy s_i of player i in the subgame $\Gamma(h)$.

The subgame perfect equilibrium of an extensive game allows the players to find out solutions in which each player can consider his plan of action not only at the beginning of the game, but also at any point of time at which he has to make a decision.

3. Exchange protocol game

An exchange protocol is naturally represented as an *extensive game* [29], since during the execution of a given exchange protocol, messages are sent one after one by different participants, until an outcome is reached [25].

A protocol game is considered as follows,

- At each stage, only one of the participants is allowed to perform an action. If there are two or more participants take actions together, this would be modeled as an interleaving of several different stages.
- If someone requires to quit the protocol in others' stage, this requirement is just delayed to his next stage, since other participants only know his quite when he does not perform actions in his stage.

Please cite this article in press as: X. Tao, et al., A game-theoretic model and analysis of data exchange protocols for Internet of Things in clouds, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.12.030

3.1. Players

The *participants* are modeled as *players* in a game, including the *trusted third party* (TTP), an entity which facilitates interactions, and be trusted by all participants.

For simplicity, this paper only considers the *two-party exchange protocols*, and thus the set of players is denoted as $N = \{1, 2, TTP\}$. Our methodology is directly extended to analyze *multi-party exchange protocols*.

3.2. Actions

An exchange protocol is a set of transmitted messages by which the participants exchange their information, which are represented as *actions* in a game.

The available actions for participants are classified the following situations,

1. send the message correctly according to the protocols;
2. send the message incorrectly (the wrong information or the wrong destination);
3. quit the protocol without any actions.

Note that, *TTP* is assumed to performs correct actions always.

M is denoted as the set of actions for messages and M_i as the message actions set of player i, including both correct messages and incorrect ones. Furthermore, a special action q is used to represent that players quit the protocol without any actions for a long time.

Hence, the action set of players A in protocol game is defined as $A = M_i \bigcup \{q\}$, where $i \in N$. An *action sequence* (denoted by h) is a sequence of actions starting from the action of the first player. H is denoted as the set of action sequences.

3.3. Player functions

An exchange protocol game is played following this way, the first player sends a first message to initiate the protocol; each active player takes an action from his set of available actions in his turn stage, one after the other, in order; and the game is finished when all player become inactive.

The player function $\{h \in H : P(h) = i\}$ assigns an action sequence to the player who takes the action next, determined by the rule of the protocol.

Due to unreliability of the network, the transmitted message may be lost, and a function f_c is deemed to associate every message with a probability measure. Each probability measure is independent to every other such measure. For every h which $P(h) = c$,

$$f_c(h) = \begin{cases} l, & \text{the message loses} \\ r, & \text{the message reaches.} \end{cases}$$

3.4. Information set

The model of an extensive game with imperfect information allows a player, when taking an action, to have only partial information about the actions taken previously.

In the protocol game, the information set $\daleth_i$ is a partition on the action sequences H of the players actions for the player i.

In some case, the player i cannot distinguish messages sent by the other players at the stage of $\{h \in H : P(h) = i\}$ in the protocol game. For example, a participant cannot distinguish the situation in which the other participant chooses q action and the situation in which he does not receive the corresponding message. So these situations are taken in the same information set. The information of these actions are in the same set I_i which is an element of the partition $\daleth_i$, on which the player i has the same action set.

3.5. Payoffs

The *payoffs* of players are the difference value of two value functions on the obtain and the lost for players on the terminal states, the functions η_i^+ and η_i^-. These are defined as follows,

$$\eta_i^+ = \begin{cases} V_i(\gamma_{-i}), & \text{received the } item_{-i} \\ 0, & \text{otherwise.} \end{cases}$$

$$\eta_i^- = \begin{cases} V_i(\gamma_i), & \text{lost his owm } item_i \\ 0, & \text{otherwise} \end{cases}$$

in which the "received" means that the player i gets the item which he wants, and the "lost" means that the player i pays out his own item. γ_i is denoted as the item of the player i, and γ_{-i} as the item of another player.

The payoffs of the terminal state ($z \in Z$) for player i can be defined as a utility function $u_i(z)$, where $u_i(z) = \eta_i^+ - \eta_i^-$. For a specific exchange protocol, if the exchange is successful, the payoff of each player is greater than zero, say, for every player, $V_i(item_{-i}) - V_i(item_i) > 0$.

3.6. The subgame of the protocol

The definition of the subgame is shown in the Section 2. This section shows how to build the subgame of the exchange protocol game. A subgame is a game that have precondition of an action sequence $h \in H \backslash Z$. let us consider the subgame of the exchange protocol game.

In an extensive game, it considers two types of subgame in the extensive game.

(1) $h = \theta$: in this subgame, all the set of players, the set of actions, player function and payoff are the same to the extensive game. The striking difference between the two games is that the subgame is a strategic game which will not consider about the sequence of actions. It just thinks about the Nash equilibrium of strategy profile.

(2) $h \in H \backslash \{Z, \theta\}$: these types of subgames also are strategic games. The set of actions are in the action sequences h' which is defined by $(h, h') \in H$, and the payoffs of the players on the Z are the same to the extensive game. It just consider the game after the action sequences h.

3.7. An example: Syverson exchange protocol

An exchange protocol proposed by Syverson in [24] is introduced in this section, as an example using the exchange protocol game. The Syverson exchange protocol illustrated as follows, is an on-line exchange protocol of a vendor V is selling Goods to a customer C.

$$Message1 \quad V \rightarrow C : [DescriptionofGoods,\ Goods_k,\ \omega(k)]_{K_V^{-1}}$$

$$Message2 \quad C \rightarrow V : [Payment,\ Message1]_{K_C^{-1}}$$

$$Message3 \quad V \rightarrow C : [K,\ Message2]_{K_V^{-1}}.$$

The detail of the Syverson exchange protocol is introduced in the Section 5.4. For this protocol, its exchange protocol game is modeled as $\Gamma^s = \langle N^s, H^s, P^s, (\succeq)^S_{i \in N^s} \rangle$. There only two players in the protocol, denoted as C and V. $N^s = \{V, C\}$. For the action, m_1, m_2 and m_3 are used for the right three messages. m_1^*, m_2^* and m_3^* are added for all the wrong there messages which are not correspond to the protocol. So, the action set $A^s = \{m_1, m_2, m_3, m_1^*, m_2^*, m_3^*, q\}$. For the player function P^s, it is obvious: the first and third message actions belong to V and the second belongs to C. The messages that sent and received are their information set. For the payoff, u_V^- and u_C^+ are denoted values of the Goods for V and C, u_V^+ and u_C^- for the Payment. The more analysis of the Syverson exchange protocol is presented in the Section 5.4.

Please cite this article in press as: X. Tao, et al., A game-theoretic model and analysis of data exchange protocols for Internet of Things in clouds, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.12.030

4. Formal model of rational exchange

The definition of the rational exchange protocol on the subgame perfect equilibrium in the extensive game is proposed in this section.

Informally, a two-party rational exchange protocol is an exchange protocol in which both main parties are motivated to behave correctly and to follow the protocol faithfully. If one of the parties deviates from the protocol, then she may bring the other, correctly behaving party in a disadvantageous situation, but she cannot gain any advantages by the misbehavior. It is very similar to the concept of a subgame perfect equilibrium of extensive game. This inspired us to give a formal definition of rational exchange in terms of a subgame perfect equilibrium of extensive game in the protocol game.

Definition 4 (*Two-Party Rational Exchange Protocol*)**.** Let us consider a two-party exchange protocol, the set of players is $N = (1, 2)$, the strategy profile (s_1^*, s_2^*) is the action sequence which the participants follow the protocol faithfully. The protocol is said to be rational

iff (s_1^*, s_2^*) is a subgame perfect equilibrium of an extensive game in the protocol game, for every player and every non-terminal history $h \in H \backslash Z$;

About a normal exchange protocol, it also is defined on the subgame perfect equilibrium. The normal exchange protocol would be considered as a N-party exchange protocol.

Definition 5. Let us consider a two-party exchange protocol, the set of players is $N = (1, 2, \ldots, n)$, the strategy profile $(s_1^*, s_2^*, \ldots, s_n^*)$ is the action sequence which the participants follow the protocol faithfully. The protocol is said to be rational

iff $(s_1^*, s_2^*, \ldots, s_n^*)$ is a subgame perfect equilibrium of an extensive game in the protocol game, for every player and every non-terminal history $h \in H \backslash Z$;

5. Game tree and analysis of the Syverson protocol

5.1. Game tree

For the extensive game, there is a very useful graphic tool called *game tree* [29], which can be used to analyze the protocol game.

Conceptually, the root of a tree, denoted by a small circle, represents the *initial state* Ø (the starting point of the extensive game). The edges of a tree correspond to the player actions. The path from the root node correspond the action sequences. The leaf nodes of the tree denote the terminal state Z, and the values in the side brackets express the payoffs at this terminal action sequences. Each node except root and leaves assigns a player, and the nodes in same tree layer are assigned to same player. The edges under these nodes correspond to actions of the assigned player.

5.2. Weight

Weight actions is defined to make up the payoffs of players in the protocol game tree, A weight action is an action in a protocol whose occurrence affects the payoffs of the corresponding terminal state. So in the most exchange protocols, the weight actions usually are the messages in which the exchange participants get items or pay out items.

A weight is assigned to each node in the game tree as the payoff when the game is ended at the current node, which can be calculated by our proposed algorithm. In this way, the payoffs of players would be easily fixed on every terminal state.

5.3. A linear algorithm

A linear time algorithm is introduced to calculate the weight of the nodes in protocol game tree. Firstly, the weight on the foot of tree is defined $\{0, 0\}$, then search its child nodes, if the edges between them are not in the set of weight action, the weight on a child node is the same as their parent node. If a edge is in the set, the weight on the node is equal to the sum of its parent node' weight and the weight of the edge right after this node.

Given a protocol game tree Γ with D depth, and the value action edges $E(x, y)$. For the value of all node(N) in the tree:

value$[0] = (0, 0)$

for $i : 1 \rightarrow D$

$N \in$ depth i

$value[N] = value[parent[N]] + value[E(parent[N], N)]$.

This linear algorithm can be used to compute weights of all leaves in game tree. These weights will be use to check the properties of exchange protocols.

Conceptually, an extensive game can be considered of as a tree in the [29]. The root node of the tree which be denoted by a small circle represent the initial history ∅ (the starting point of the extensive game). The edges of the tree which is among the tree layers correspond to actions of the players. These crease line segments that emanate from the root node correspond the action sequences in the game. And the leaf nodes of the tree denote the terminal state Z, and the values in the side brackets express the payoffs of players at this terminal action sequences history. All nodes except those two types assign the players in the game. All those nodes in the same tree layer denote the same player; the edges under those nodes correspond to the actions of the player.

5.4. The Syverson protocol

Rationality property is proposed as application for the weak protection of secrets in which weakness is not just acceptable but desirable. In the big data, IoT and cloud computing, weak protection is tolerated because of a number of problems associated with stronger stuff: availability, cost (both monetary and resource), and legal or policy restrictions. For example, in order to get more information, many application programs of mobile phone and other devices provide incentives so that principals operating out of enlightened self-interest have more reason to proceed with the application at each point than to abort. The Syverson exchange protocol illustrated as follows is a protocol designed through this way to exchange message.

$Message1 \quad V \rightarrow C : [DescriptionofGoods, Goods_k, \omega(k)]_{K_V^{-1}}$

$Message2 \quad C \rightarrow V : [Payment, \ Message1]_{K_C^{-1}}$

$Message3 \quad V \rightarrow C : [K, \ Message2]_{K_V^{-1}}$.

In this protocol, a vendor V is selling Goods to a customer C. In the first step of the protocol, V generates the description of goods and a random key k; encrypts *Goods* with k; computes the temporarily secret commitment $\omega(k)$; and use his private K_V^{-1} encrypts all the message, send it to C.

When C receives m_1, she uses the public key of V to decrypt the m_1 and verifies the description of goods. If C is satisfied, then she sends the encrypted by her private key message m_2 which contains *payment* and m_1 to V.

When V receives m_2, he uses the public key of C to decrypt the m_2, verifies the payment and checks if it contains m_1. If he is satisfied, then he sends the key k to C in the message m_3, which is encrypted by his private key and contains the received message m_2.

Please cite this article in press as: X. Tao, et al., A game-theoretic model and analysis of data exchange protocols for Internet of Things in clouds, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.12.030

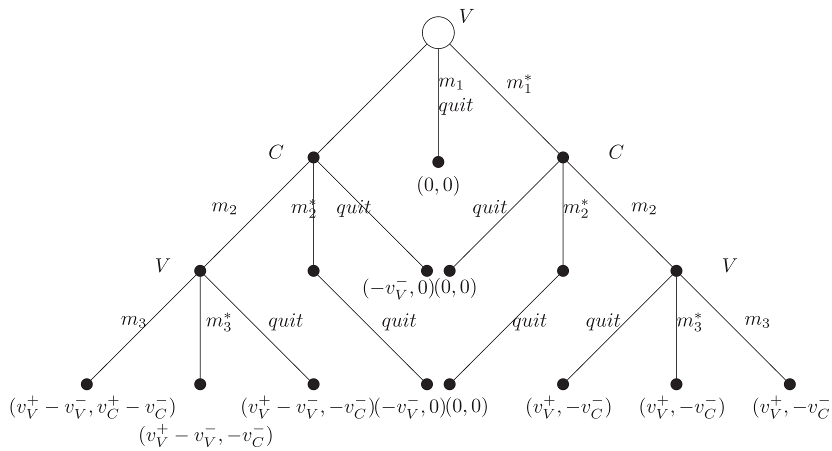

Fig. 1. Game tree of Syverson protocol.

When C receives m_3, she decrypt it, and checks if it contains m_2. Then, she decrypt the encrypted goods in m_1 which the key received in m_3. For the details on the protocol, the reader is referred to [24].

As the exchange protocol game presented in the Section 3.7, Fig. 1 is the game tree of Syverson protocol. Because of the temporarily secret commitment and the reputation, the payoffs of players are considered as u_C^- and u_V^-. For any non-terminal action sequences $h \in \{\theta, m_1, m_1m_2\}$, the strategy profile (s_V^*, s_C^*) in which $s_V^* = m_1m_3$ and $s_C^* = m_2$ is a sub-game perfect equilibrium. So the Syverson protocol is a rational exchange protocol in our model.

6. Relationship with Buttyán's model

This section shows the relationship between the model defined in the Section 4 and Buttyán's. In the [31], Levente Buttyán gives a formal definition for rational exchange relating it to the concept of Nash equilibrium in games. This paper presents a formal definition for the rational exchange in the extensive game with perfect information relating it to the concept of the subgame perfect equilibrium of an extensive game.

The relationship between Buttyán's model with ours should be considered. Buttyán's model takes the rational protocol on the best Nash equilibrium, and our one takes the rational protocol on the subgame equilibrium.

For the relationship between the Nash equilibrium of an extensive game and the subgame perfect equilibrium of an extensive game, the subgame perfect equilibrium of an extensive game is stricter than the Nash equilibrium of an extensive game. In the game theory, the Nash equilibrium of an extensive game can imply the subgame perfect equilibrium of an extensive game. In another words, the subgame perfect equilibrium of an extensive game is contained in the Nash equilibrium of an extensive game. The subgame perfect equilibrium is a subset of Nash equilibrium.

But in Buttyán's model, he considers the best Nash equilibrium. For the rationality of players in the game, so the best Nash equilibrium also is the subgame perfect equilibrium. Because if (s_1^*, s_2^*) is the best Nash equilibrium in an two-parties exchange protocol game, $(s_1^*|_h, s_2^*|_h)$ is the best Nash equilibrium in any other subgame in which the h is a non-terminal action sequence history that the two player choose the action according to the strategy profile (s_1^*, s_2^*). Buttyán's model is contained in ours.

But Buttyán's model cannot imply ours. It would pick out and throw away some rational exchange protocol. In some protocols, there are some Nash equilibriums, but no a best Nash equilibrium, only exist the subgame perfect equilibrium. The following example can illustrate this point completely.

6.1. An example

In this section, a rational protocol illustrated as follows is considered to show the relationship between the two models of the rational exchange protocol.

$U \to S : m_1 = (Name_{srv})$
$S \to U : m_2 = (P_{srv}, tid)$
$U \to S : m_3 = (U; S; tid, P_{srv}, h(rnd), \sigma_u(U, S, tid, val, h(rnd)))$
$S \to U : m_4 = (srv)$
$U \to S : m_5 = (rnd)$

- if S received the m_3 and m_5;

  $S \to B : m_6 = (m_3, rnd, \sigma_S(m_1, rnd))$
- if S received only the m_3;

  $S \to B : m_6' = (m_3, \sigma_S(m_1))$.

The above protocol is used for transferring payment from a user U to a sell S in exchange for some service provided by S to U. In this protocol, besides the two exchange parties, there is a bank B which is a trusted third party.

In the first step of the protocol, the use U sends the name of service to the sell S to ask the price of the service. When S receives m_1, she generates a fresh transaction identifier tid, put the price and the tid in the message m_2 and send it to U.

When U receives m_2, U generates a random number rnd and computes its hash value $h(rnd)$, then, generates the digital signature $\sigma_u(U; S; tid; val; h(rnd))$ and sends the message m_3 to S.

When S receives m_3, she provides the service to U(represented by sending $m_4 = srv$). If U is satisfied, then she sends the random number rnd to S.

If S receives m_3 and m_5, then she generates the digital signature $\sigma_S(m_1, rnd)$, and sends m_6 to B. If S received only m_3, then she generates the digital signature $\sigma_S(m_1)$, and sends m_6' to B. For B, it receive the message, and use the information in the message to verify the transaction between U and S. If it received m_6, it still verifies that the hash value of rnd equals the hash value in m_3. If all these are successful, then it logs the transaction, and transfers the value val from the account of U to the account of S. Upon reception of m_6', B performs the transaction verification, and if these are successful, then it debits the account of U with the value val, but it do not credit V's account.

This protocol is a variation of a rational exchange protocol in [25], added two steps for the two exchange parties asking the

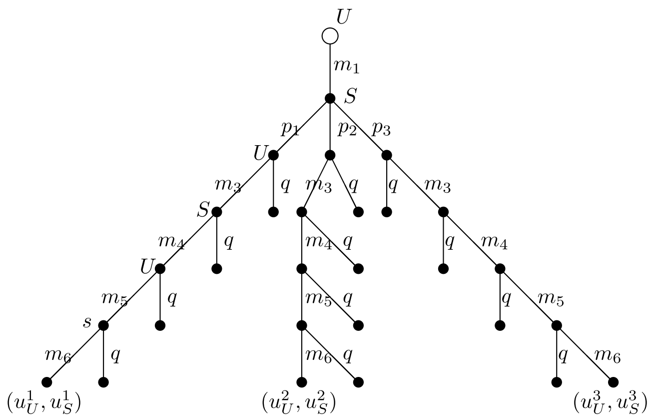

Fig. 2. Game tree of example protocol.

price of the server. This also is a rational exchange protocol. There are three prices P_1, P_2 and P_3 ($P_1 \leq P_2 \leq P_3$) sent by the player S, in which P_1 and P_2 can be accepted by the player U. Its exchange protocol game is $\Gamma^e = \langle N^e, H^e, P^e, (\succeq)^e_{i \in N^e} \rangle$, where $N^e = \{U, S, B\}$. The action set $A^e = \{m_1, p_1, p_2, p_3, m_3, m_4, m_5, m_6, m_6', q\}$. For the payoff, $\eta^-_{U_1}$ and $\eta^+_{S_1}$ are denoted values of the service for U and S at the price P_1, $\eta^+_{U_1}$ and $\eta^-_{S_1}$ for the *val*. So, $u_U^1 = \eta^+_{U_1} - \eta^-_{U_1}$ describes the payoff for U, and $u_S^1 = \eta^+_{S_1} - \eta^-_{S_1}$ for S at the success exchange in the price P_1. (u_U^2, u_S^2), (u_S^3 and u_S^3) are defined in the same way for the prices P_2 and P_3. Through the game tree of the protocol in the Fig. 2, it is easy to find that there are two non-zero Nash equilibriums in which the payoffs of players is (u_U^1, u_S^1) and (u_U^2, u_S^2). For $P_1 \leq P_2$, $u_U^1 \succ_U u_U^2$ and $u_S^2 \succ_S u_S^1$. There does not exist the best Nash equilibrium in this rational protocol. But, the game exist the subgame perfect equilibrium. For the Buttyán's model, this protocol is not a rational exchange protocol, but for our one, it is a rational protocol. So, our model strictly contains the Buttyán's.

7. Fair exchange

There are two fairness properties for exchange protocols defined in this game model.

Definition 6 (*Strict Fairness*)**.** Let Γ_2 denote a two-party exchange protocol game. The strategy (action sequences) sets of two players 1 and 2, is defined as ΣS_1 and ΣS_2. The protocol Γ_2 is strictly fair iff

$\forall S_1 \in \Sigma S_1, \qquad \forall S_2 \in \Sigma S_2,$
For $\forall z \in Z$,
$(u_1(z), u_2(z)) \in \{(V_1(\gamma_2) - V_1(\gamma_1), V_2(\gamma_1) - V_2(\gamma_1)), (0, 0)\}.$

A strictly fair two-party exchange protocol means there are only two couples of values (the two are $(V_1(\gamma_2) - V_1(\gamma_1), V_2(\gamma_1) - V_2(\gamma_1))$ and (0,0)) on the all terminal states of the protocol game for the two exchange participants. For its game tree, there also only two type weights $(V_1(\gamma_2) - V_1(\gamma_1), V_2(\gamma_1) - V_2(\gamma_1))$ and $(0, 0)$ on the every leaf.

This type fairness is very strict, which means that whatever the participants do, after completion of protocols run, either each participant receives the expected item or neither two receives any useful information about the other's item. It guarantees that any participant (no matter whether participants behave correctly or try to cheat) will be fairness at the end of protocol.

Definition 7 (*Ordinary Fairness*)**.** Let Γ_2 denote a two-party exchange protocol game. The strategy (action sequences) sets of two players 1 and 2, is defined as ΣS_1 and ΣS_2. The action strategy profile that corresponds to the faithful execution of protocol is defined by (S_1^*, S_2^*). The protocol Γ_2 is ordinarily fair iff

$\forall S_2 \in \Sigma S_2 :$
$u_1(S_1^*, S_2) > -V_1(\gamma_1)$
and $\forall S_1 \in \Sigma S_1$,
$u_2(S_1, S_2^*) > -V_2(\gamma_2).$

Ordinary fairness guarantees that a correctly behaving participant cannot be in any disadvantages (no matter whether the other participant behave correctly or try to cheat).

Effectiveness: If two parties behave correctly, they will receive the expected items without any involvement of the *TTP*.

This property can be formal as in the tree, there exist a path from the root node to a non-zero value's leaf node in which there are no the node of *TTP*.

Non-repudiation: If an item has been sent from party 1 to party 2, 1 cannot deny origin of the item and 2 cannot deny receipt of the item. About this, an action of message m_i is non-repudiation. This property can be defined as the length of its information set is one in the protocol game. $|\Im_{m_i}| = 1$.

8. Analysis of fair exchange protocol

In this section, the *ASW protocol* is adopted as a case study, to show the usage of our methodology.

8.1. ASW protocol

In the big data, IoT and cloud computing, fairness is proposed to guarantee to each participant (eventual) delivery of the agreed things, objects or information. Especially in the open market-oriented cloud computing, the fair mechanisms play an important role for the Internet business. Another thing related to fairness is nonrepudiation, what provides nonrepudiation of origin, proof of who the sender of a message is, or nonrepudiation of receipt, proof of who received the message, or both. These should bring the user more security during information exchange in the Internet. ASW protocol is an asynchronous, optimistic fair exchange protocol introduced by Asokan, Shoup and Waidner [21]. ASW protocol contains three sub-protocols: exchange, abort and resolve. In the normal case, only the exchange sub-protocol is executed. The other two sub-protocols are used only if something wrong and forcibly complete a protocol run.

The exchange sub-protocol is as follows.

1. $O \to R$: $me1 = V_o, V_R, TTP, C, H(M),$
   $sS_O(V_O, V_R, TTP, C, H(M))$
   IF R gives up THEN quit ELSE
2. $R \to O$: $me2 = H(key_R), sS_R(mes1, H(key_R))$
   IF O gives up THEN abort ELSE
3. $O \to R$: $me3 = M,\ key_O$
   IF R gives up THEN *resolve_R* ELSE
4. $R \to O$: $me4 = key_R$
   IF O gives up THEN *resolve_O* ELSE.

The abort sub-protocol is as follows.

1. $O \to TTP$: $ma1 = aborted, me1, sS_O(aborted, me1)$
   IF R has resolved THEN *resolve_O* ELSE
2. $TTP \to O$: $abort\_token = ma1, sS_{TTP}(ma1).$

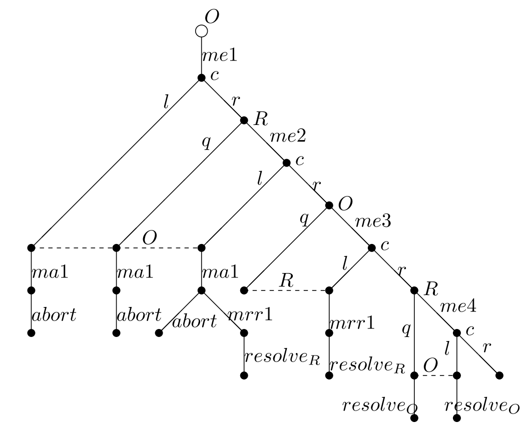

Fig. 3. Game tree of ASW protocol.

The *resolve_R* sub-protocol is as follow.

1. $R \rightarrow TTP: \ mrr1 = V_R, me1, me2, \ key_R$
   IF aborted THEN
2. $TTP \rightarrow R: \ mrr2 = abort\_token$
   ELSE
3. $TTP \rightarrow R: \ mrr3 = M, \ key_O.$

The *resolve_O* sub-protocol is as follows.

1. $O \rightarrow TTP: \ mro1 = V_O, me1, me2, M, \ key_O$
   IF aborted THEN
2. $TTP \rightarrow O: \ mro2 = abort\_token$
   ELSE
3. $TTP \rightarrow O: \ affidavit\_token = affidavit, mro1,$
   $sS_{TTP}(affidavit, mro1).$

In the exchange sub-protocol, if R decides to give up before sending $me2$, it can simply terminate the protocol run without losing fairness. If O decides to give up after sending $me1$ (Usually because O does not receive $me2$ within a reasonable time), it invokes the *TTP* by running the abort sub-protocol. If R decides to give up after sending me2 (typically because R does not receive me3 in time), it invokes the *TTP* by running the *resolve_R* sub-protocol. If O decides to give up after sending me3 (typically because O does not receive me4 in time), it invokes the *TTP* by running the *resolve_O* sub-protocol.

8.2. Tree analysis method

The exchange protocol game of a ASW protocol is an extensive game $\Gamma_{ASW} = \langle N, H, P, f_c, \beth_i, (\succeq)_{i \in N} \rangle$. The player set $N_{ASW} = \{O, R, TTP\}$. In the ASW protocol, the messages with wrong information can easily uncovered by recipient. So, it is not need to care about those actions. $M_O = \{me1, me3, ma1, mro1\}$, $M_R = \{me2, me4, mrr1\}$. The actions set $A = M_O \bigcup M_R \bigcup \{q\}$.

By the game tree in the Fig. 3, it clearly finds that O cannot distinguish the three situations:

1. $me1$'s lost;
2. R's q action at the second step;
3. $me2$'s lost.

In the first situation, the protocol will be aborted at the end because R cannot receive $me1$. For the last one, the protocol will abort or use resolve sub-protocol. In these two, the protocol runs without losing fairness. But in the second situation, if R sends $mrr1$ to start the *resolve_R* sub-protocol when he does not send $me2$, O cannot abort the protocol or use the *resolve_O* sub-protocol. Because, R has taken the *resolve_R* sub-protocol and O has not got the $me2$ to start the *resolve_O* sub-protocol. So, in such way, R can get $me1$, key_O, but O cannot get $me2$, key_R at the end of the protocol.

It also finds that ASW protocol has considered the unreliability of channels between two exchange participates, but the channels between the two parties and *TTP* are not considered.

9. Conclusion

In order to address the security challenge of information exchange for big data, IoT, and cloud computing, a specific extensive-game model for behavior analysis of exchange mechanism based on the extensive game with imperfect information in this paper. Rationality property is built on the subgame perfect equilibrium of the corresponding game, and fairness property in the corresponding game tree. It is compared with the Buttyán's model of the rational exchange protocol which is defined in terms of an Nash equilibrium to shown that this definition is more powerful. To verify the properties, a tree analysis method is proposed for exchange mechanism, and a linear time algorithm is given for the game tree. The ASW protocol as a case study is used for this method. Some flaws are detected by this method.

Acknowledgment

This project is supported by the National Natural Science Foundation of China (No. 61472240, 61672340, 61572268).

References

[1] L. Atzori, A. Iera, G. Morabito, The Internet of things: A survey, Comput. Netw. 54 (15) (2010) 2787–2805.
[2] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, M. Ayyash, Internet of things: A survey on enabling technologies, protocols, and applications, IEEE Commun. Surv. Tutor. 17 (4) (2015) 2347–2376.
[3] L.M. Bruce, Game theory applied to big data analytics in geosciences and remote sensing, in: Geoscience and Remote Sensing Symposium (IGARSS), 2013 IEEE International, IEEE, 2013, pp. 4094–4097.
[4] L. Ertaul, S. Singhal, G. Saldamli, Security challenges in cloud computing, in: Security and Management, 2010, pp. 36–42.
[5] W. Stallings, Network Security Essentials: Applications and Standards, Pearson Education India, 2007.
[6] J. Hirshleifer, Protocol, payoff, and equilibrium: Game theory and social modelling, UCLA Economics Department, Working Paper 366, 1985, 201–233.
[7] L. Buttyán, J. Hubaux, S. Capkun, A formal analysis of Syverson's rational exchange protocol, in: Proceedings of the 15th IEEE Computer Security Foundations Workshop, 2002, pp. 181–193.
[8] Z. Hao, S. Zhong, N. Yu, A privacy-preserving remote data integrity checking protocol with data dynamics and public verifiability, IEEE Trans. Knowl. Data Eng. 23 (9) (2011) 1432–1437.
[9] Y. Yu, J. Ni, M.H. Au, H. Liu, H. Wang, C. Xu, Improved security of a dynamic remote data possession checking protocol for cloud storage, Expert Syst. Appl. 41 (17) (2014) 7789–7796.
[10] F. Teng, F. Magoulès, A new game theoretical resource allocation algorithm for cloud computing, in: Advances in Grid and Pervasive Computing, Springer, 2010, pp. 321–330.
[11] G. Wei, A.V. Vasilakos, Y. Zheng, N. Xiong, A game-theoretic method of fair resource allocation for cloud computing services, J. Supercomput. 54 (2) (2010) 252–269.
[12] N.S. Rao, S.W. Poole, F. He, J. Zhuang, C.Y. Ma, D.K. Yau, Cloud computing infrastructure robustness: A game theory approach, in: Proceedings of the International Conference on Computing, Networking and Communications, (ICNC), IEEE, 2012, pp. 34–38.
[13] X. Xu, H. Yu, A game theory approach to fair and efficient resource allocation in cloud computing, Math. Probl. Eng. 2014 (2014) 1–14.
[14] E. Furuncu, I. Sogukpinar, Scalable risk assessment method for cloud computing using game theory (ccram), Comput. Stand. Interfaces 38 (2015) 44–50.
[15] C. Ma, J.-P. Sheu, C.-X. Hsu, A game theory based congestion control protocol for wireless personal area networks, J. Sensors 2016 (2015) 1–14.
[16] J. Estevez-Tapiador, A. Alcaide, J. Hernandez-Castro, A. Ribagorda, Bayesian rational exchange, Int. J. Inf. Secur. 7 (1) (2008) 85–100.
[17] C. Li, W. Huang, Z. Ma, F. Wang, X. Wei, L. Xu, An automated negotiation protocol based on game theory, Comput. Water Energy Environ. Eng. 3 (3) (2014).

Please cite this article in press as: X. Tao, et al., A game-theoretic model and analysis of data exchange protocols for Internet of Things in clouds, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.12.030

[18] J. Tian, Game-theory model based on carrier sense multiple access protocol in wireless network, J. Netw. 9 (6) (2014) 1603–1609.
[19] J. Zhou, R. Deng, F. Bao, Some remarks on a fair exchange protocol, in: Public Key Cryptography, Springer, 2004, pp. 46–57.
[20] S. Kremer, Formal analysis of optimistic fair exchange protocols, Ph. D. thesis, Universit'e Libre de Bruxelles Facult'e des Sciences (2003-2004).
[21] N. Asokan, V. Shoup, M. Waidner, Asynchronous protocols for optimistic fair exchange, in: Procceedings of the Conference on Security and Privacy, IEEE, 1998, pp. 86–99.
[22] N. Asokan, Fairness in electronic commerce (Ph. D. thesis), University of Waterloo, 1998.
[23] H. Pagnia, H. Vogt, F. Gartner, Fair exchange, Comput. J. 46 (1) (2003) 55–75.
[24] P. Syverson, Weakly secret bit commitment: Applications to lotteries and fair exchange, in: Proceedings of the 1998 IEEE Computer Security Foundations Workshop, CSFW11, 1998, pp. 2–13.
[25] L. Buttyán, H. Jean-Pierre, Rational exchange-a formal model based on game theory, Electron. Commer. (2001) 114–126.
[26] G. Kol, M. Naor, Cryptography and game theory: Designing protocols for exchanging information, Theory Cryptogr. (2008) 320–339.
[27] Y. Gu, Z. Shen, D. Xue, A game-theoretic model for analyzing fair exchange protocols, in: 2009 Second International Symposium on Electronic Commerce and Security, IEEE, 2009, pp. 509–513.
[28] L. Buttyán, J. Hubaux, Toward a formal model of fair exchange-a game theoretic approach, Tech. rep., NO. LCL-REPORT, 1999.
[29] M. Osborne, A. Rubinstein, A Course in Game Theory, The MIT press, 1994.
[30] M. Osborne, An Introduction to Game Theory, Oxford University Press, New York, NY, 2004.
[31] L. Buttyán, J. Hubaux, S. Čapkun, A formal model of rational exchange and its application to the analysis of Syverson's protocol, J. Comput. Secur. 12 (3, 4) (2004) 551–587.

Xiuting Tao, Ph.D. candidate at school of software, Shanghai Jiao Tong University, received the B.S., and M.S. degrees from Xidian University, and Zhejiang Normal University in 2007, and 2011 respectively. His research interests focus on formal verification and equivalence checking.

Guoqiang Li received the B.S., M.S., and Ph.D. degrees from Taiyuan University of Technology, Shanghai Jiao Tong University, and Japan Advanced Institute of Science and Technology in 2001, 2005, and 2008, respectively. He worked as a postdoctoral research fellow in the graduate school of information science, Nagoya University, Japan, during 2008®C2009, as an assistant professor in the school of software, Shanghai Jiao Tong University, during 2009®C2013, and as an academic visitor in the department of computer science, University of Oxford during 2015–2016. He is now an associate professor in school of software, Shanghai Jiao Tong University. His research interests include formal verification, programming language theory and computational learning theory. He published more than 30 researches papers in the international journals and mainstream conferences, including IJFCS, Science China Information Sciences, FORMATS, ATVA, etc.

Daniel Sun received his Ph.D. in Information Science from Japan Advanced Institute of Science and Technology (JAIST) in 2008. From 2008 to 2012, he was an assistant research manager in NEC central laboratories in Japan. From 2013, he has been working for National ICT Australia as a researcher. He is also a conjoint lecturer in School of Computer Science and Engineering, the University of New South Wales, Australia. His current research interests include big data, cloud computing, cybersecurity, system reliability, and data mining.

Hongming Cai received the B.S., M.S., and Ph.D. degrees from Northwestern Polytechnical University, Xi'an, China, in 1996, 1999, and 2002, respectively. During 2002–2004, he served as a Postdoctoral Research Fellow with the Department of Computer Science and Technology, Shanghai Jiao Tong University, Shanghai, China. During 2008–2009, he served as Visiting Professor with the Business Information Technology Institute, University of Mannheim, Mannheim, Germany. The visiting scholarship was sponsored by Alfried Krupp von Bohlen und Halbach Foundation, Germany. He now is an Associate Professor with the School of Software, Shanghai Jiao Tong University.

Please cite this article in press as: X. Tao, et al., A game-theoretic model and analysis of data exchange protocols for Internet of Things in clouds, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2016.12.030