Cyber Insider Threats Situation Awareness Using Game Theory and Information Fusion-based User Behavior Predicting Algorithm

Ke Tang*, Mingyuan Zhao, Mingtian Zhou

School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China

Abstract

Cyber insider threat is a difficult problem because it is always covered by a legal identity. Researchers have proposed many methods to deal with this kind of problem which are model-based, graph-based and access control-based algorithms. However, many of these methods are dependent upon traditional IDS which are impacted by false positive rate and not suitable for insider problem any more. Some other game-based methods are dependent on assumption that insiders’ decisions are optimal and rational. Nevertheless, this kind of algorithm can not handle some irrational insider’s behavior and determine when a round of interaction starts or ends for system defender. In this paper, we proposed our algorithm for insider threat situation awareness, which is based on game theory and information fusion. We use dynamic Bayesian network (DBN) structure and exact inference to acquire and fuse different type of insider information for behavior analysis and avoid traditional IDS shortcoming, finally we obtain situation awareness or prediction trend of insider’s future actions by quantal response equilibrium (QRE) calculation. Simulation experiment results indicate that our algorithm has better convergence and precision than other same algorithm even though we should pay additional but accepted computation cost.

Keywords: Insider; Situation Awareness; Quantal Response Equilibrium; Dynamic Bayesian Network; Information Fusion; User Behavior

1 Introduction

The “cyber insider threats” or “cyber internal attack problems” have been considered one of the most serious security problems in the computational information system such as computer network and database system [1, 2]. These problems are revealed by IT infrastructure sabotage, theft or modification for financial gain, theft or modification for business advantage and miscellaneous. It is difficult and hard to deal with them, because an insider has information and capabilities not known to external attackers. Although we have not standard definition of insider threat until

*Corresponding author.
Email address: kevindoctor@gmail.com (Ke Tang).

1548–7741/ Copyright © 2011 Binary Information Press
March 2011
now, we still have a common notion about insider and his or her actions [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. Generally speaking, the “insider” is an individual currently or at one time authorized to access an organization’s information system, data, or network; such authorization implies a degree of trust in the individual. The insider threat refers to harmful acts that trusted insiders might carry out [4]. Its two primitive actions can be defined violation of a security policy using legitimate access and violation of an access control policy by obtaining unauthorized access [6].

A lot of research has been proposed on detection against insider attacks [2, 3, 5, 7]–[30]. Some of them are modeling method based such as ABGAC (Attribute-Based Group Access Control) [9], insider threat behavior modeling based on multi-entity Bayesian network [13], a behavioral theory model of the dynamics of insider-threat risks [14], a formal model of systems describing real-world scenarios and mapping to a process algebra-aklaim [15], a knowledge-based Bayesian model [16], insider threat prediction model [18], an insider model based on logged information and documents accessed accomplishing an intelligence analysis [22] and a three-layer architectural security model [24] containing the physical, logical and social levels. Others are graph-based methods which are CAG (Capability Acquisition Graph) evaluating insider threat cumulative effect and detecting possible violations [17], and a graph-based approach to discovering anomalous instances of structural patterns in data that represent entities, relationships and actions [26]. The objective of many methods is to detect deviated action from a predefined model of normal system behavior. However, it is more difficult to identify normal system behavior and insider attack action, especially at early insider threat action stage, because an insider has more capabilities and resources than outside attackers. He or she may have legitimate account and password which means that an insider has authorized access to the target system, and he or she may have more extensive knowledge of the target system than external attackers. Many traditional and generic approaches for anomaly detection are not good at detecting effectively insider attacks because of its high false positive rate. Hence, the other approach is proposed recently not to revise the existing anomaly detection techniques but to build upon them based on game-theoretic techniques [12, 29, 30]. Security monitor and control of computational information system is a human-computer interactive situation. Game theory is the discipline which studies interactive situation. Recognizing and detecting the insider threat as a significant source of security incidents, some game-theoretic approaches are proposed to deal with insider problems, which can use to accurately predict an insider’s moves and identify the optimal defense strategy. However, there is a common assumption in these approaches, which insiders are rational and the game solution is based on Nash equilibrium [12, 29]. In actual situation, users are not always at expert level, some of them (including insiders) are naïve instead and their decisions may not be optimal and rational. Nash equilibrium is not appropriate to use to solve the game. On the other hand, previous detection algorithms of game-based [12, 29, 30] against insider are built upon traditional IDS (Intrusion Detection System) which will impact on their detection accuracy. Traditional IDS lacks for detection capability against insider and can not use to monitor legitimate access account user’s action directly. We need a novel equilibrium calculation to solve the game and information fusion algorithm based on traditional IDS deployment to acquire insider’s move more accurately. In this paper, we propose game theory and information fusion-based user behavior predicting algorithm which combine QRE (Quantal Response Equilibrium) calculation to solve above problem. By using dynamic Bayesian network (DBN) as information acquiring and fusion utility, we improved uncertainty problem that many other algorithms have, which the defender does not have perfect knowledge about the outcome of its interactions with the attackers or even the existence of such interaction at all, and the defender cannot determine when a
round of interaction starts or ends [12]. By using QRE calculation, our algorithm can describe all kinds of strategies of different insider used. Finally, our proposed algorithm can give the situation awareness of insider’s behavior trend in order to support system administrator decision.

2 Game Theory and Information Fusion-based User Behavior Predicting Algorithm

Compared with an external attacker, an insider who has legitimate access account may have more knowledge about system defenses and, therefore, may be more capable of hiding his malicious activities. As a result, a traditional detection system used to external intruders can be less effective against insiders, producing more false negatives or false positives. Traditional IDS’s sensors are deployed separated and send their alarms respectively when security events occurred. In order to improve the detection accuracy, we need more information about human-computer interactive situation such as system resource consuming status and system audit log records [25] etc., in addition to IDS alerts. Acquiring more information depends upon more sensors deployment. However, deployment, management and application of sensors are constrained by costs; we must select an optimal deployment [33] and a subset of sensors that are the most decision-relevant. Information fusion algorithm design is the right way to attain our goal. In this paper, we assume that we have deployed sensors optimally and we will focus on dynamic information fusion problem.

On the other hand, game theory is our mathematical tool to model the insider problem. A computational information system is naturally represented as a state machine, and is directly extended to describe the game by considering the discrepant objectives of the players who control state transitions. Each atomic attack action of insider, which may cause a transition of the current system state, is regarded as an action in a game where an insider’s choice of action is based on a consideration of the possible consequences. The interactions between the insiders and the systems can then be modeled as a game. A joint action at one state will drive the system to the next state in a probabilistic manner, and this is the probabilistic transition between states. For simplicity, we use the same assumption in [29, 30], which all the malicious insiders are teamed up as a single player and all the administrators of system are fully coordinated, and this results in a two-person game.

2.1 Operation Attribute of User and Its Behavior Ranking

As we know, an insider may have legitimate access account and understand more knowledge about system defenses. We do not depend upon traditional IDS to detect them and need to find novel methods to solve the problem. Firstly, we must study and acquire the operation attribute of insider in order to monitor and response to it. We subdivide user’s behavior into some type of behavior attributes, such as session, resource consuming, document access and query. The definition is \((User\ behavior) = \{Session, Resource\ consuming, Document\ access, Query\}\). Then we subdivide these attributes into more small units respectively in order to acquire behavior evidences. We decompose user’s behavior into three layers like Fig. 1. This layered and subdivided method can be constructed Bayesian network (i.e. Fig. 1) naturally because there is some causality between these attributes. Hence, we can classify and rank user’s behavior trusted grade according to its evidences. For example, we divide them into \(N\) grades, let \(i \in [1, N]\), they are ranking
trusted grade from high to low such as \[1-\beta \div (N-1), 1-2 \times \beta \div (N-1), 1-3 \times \beta \div (N-1), \cdots, 1-i \times \beta \div (N-1), 1-(i-1) \times \beta \div (N-1), \cdots, 1-\beta, 1-(i-2) \times \beta \div (N-1), \cdots, 0, \alpha\], where \(\alpha + \beta = 1\), \(\alpha\) is the trusted threshold, when the evaluation value is below the \(\alpha\), the system will reject to provide service to the user and stop its access at once. After each interaction between user and system, the node’s value needs to be computed by Bayesian network. The saved data is only the times that node value falls into certain range. After each interaction of two entities-user and system, if certain node’s value falls into certain range, then the corresponding interaction times add one and the other statistical times remain immovability.

**Fig. 1: Instantiation of Bayesian Network of User Behavior and Its attributes**

**Definition 1** Let \(G\) express root node-user behavior, \(A\) express session node, \(B\) express resource consuming node, \(C\) express document access node, \(D\) express query node and \(I\) express those leaf nodes.

For example, let \(N = 5\) and \(\alpha = 0.6\), then we rank user’s behavior trusted grade five range section such as \([0.9, 1]\), \([0.8, 0.9]\), \([0.7, 0.8]\), \([0.6, 0.7]\), \([0, 0.6]\). In these scopes, the value of \(G\) falls into \([0.9, 1]\) or \([0.8, 0.9]\), we think user’s behavior is normal. When it falls into \([0.7, 0.8]\), user may make mistake operation and system will give recommendation to him. When it falls into \([0.6, 0.7]\), we think that it is suspect insider action and system will give warning to him. When it falls into \([0, 0.6]\), user’s behavior has been lost any trusted grade and system will reject to provide any service and deny user’s access. We use \(G_i, A_i, B_i, C_i, D_i (1 \leq i \leq 5)\) to denote those scopes respectively, and we can compute the prior probability of each node according to Bayesian network theory.

\[
P(G_i) = k_i \div n, \quad (1 \leq i \leq 5), \quad \text{where} \sum_{i=1}^{N} P(G_i) = 1. \tag{1}
\]

In above formula, \(k_i\) denotes the times that node \(G\)’s value falls into \(G_i, (1 \leq i \leq N)\), and \(n\) denotes the total times that system interacts with user. The calculation method can be used to those child nodes of the root as the same.
2.2 Information Fusion Based on Dynamic Bayesian Network

Information fusion is a process which can deal with the association, correlation and combination of information collected from various disparate sources into one coherent structure in order to apply inferring algorithm. The results of the process will be used by a computer system to make a better decision than from single source only. In this paper, we assume that we have deployed many sensors for acquiring user’s operation and the system state information distributed in the protected computational information system through optimal method [33]. The result of information fusion is defined a fusion function which is used to combine multiple sensors and output a single decision. The fusion function can be defined as

\[ \Phi = F(S_1, S_2, \cdots, S_n), \]

where \( S_i \) (1 ≤ i ≤ n), denotes an individual sensor that gives a detection and measurement; \( \Phi \) is the output of information fusion which is decision; and the fusion function \( F \) is determined by fusion methods. Since any user operation is a dynamic human-computer interactive process, dynamic Bayesian network (DBN) [34] which is a dynamic probabilistic network is the appropriate method for information fusion. The fusion output \( \Phi \) is the posterior probability which is calculated by inferring algorithm of DBN. Furthermore, DBN is also used in network security domain as a usual measure, i.e. hidden Markov model [37], which is a special model of DBN.

DBN is a way to extend Bayesian network (BN) to model probability distributions over semi-infinite collections of random variables, \( Z_1, Z_2, Z_3, \cdots \), and we can partition the variables into \( Z_t = (U_t, X_t, Y_t) \) to represent the input, hidden and output variables of a state-space model [34]. A DBN is defined to be a pair, \( (B_1, B_-) \), where \( B_1 \) is a BN which defines the prior \( P(Z_1) \), and \( B_- \) is a two-slice temporal Bayesian network (2TBN) which defines \( P(Z_t|Z_{t-1}) \) by means of a DAG (directed acyclic graph) as follows:

\[ P(Z_t|Z_{t-1}) = \prod_{i=1}^{N} P(Z_i^t|Pa(Z_i^t)). \] (2)

where \( Z_i^t \) is the \( i \)'th node at time \( t \), which could be a component of \( X_t, Y_t \) or \( U_t \), and \( Pa(Z_i^t) \) are the parents of \( Z_i^t \) in the graph. The nodes in the first slice of a 2TBN do not have any parameters associated with them, but each node in the second slice of the 2TBN has an associated conditional probability distribution (CPD), which defines \( P(Z_t|Pa(Z_i^t)) \) for all \( t > 1 \). When we unroll the 2TBN until we have \( T \) time slices, the resulting joint distribution probability is defined as follows:

\[ P(Z_{1:T}) = \prod_{t=1}^{T} \prod_{i=1}^{N} P(Z_i^t|Pa(Z_i^t)). \] (3)

We can build fusion structure of DBN based with the Bayesian network of Fig. 1. The root node of such a network would contain fusion result variables which is user’s behavior trusted grade level. The sensors are leaf nodes and the only observable variables in the model. Evidences from each source are collected by these sensors. There are two intermediate levels in our DBN structure except root node and leaf nodes. They are user behavior attributes and its child nodes-information variables, which are interrelated by cause and effect. The root node is causally linked to the sensor nodes which are leaf nodes by the intermediate variables. DBN structure is shown in Fig. 2. In the real situation, there are some factors which will impact on sensor detection accuracy such as sensor imprecise acquisition and sensor noise etc. These factors are called uncertainty of sensor measurement. We must find method to improve this situation. According to DBN theory foundation, conditional probabilities between information variables and sensors...
can quantify the uncertainty of sensor measurements. Consequently, the uncertainty of sensor inputs is incorporated into the fusion system to update the probability distribution over the root node variable. Evidences regarding information variables are collected by sensors and are fused by DBN inference algorithm [34] which is online joint tree algorithm.

**Definition 2** Let $\Phi$ be a set of fusion result $\phi_k$, $k = 1, 2, 3, \cdots, K$. Let the sensors $S = \{S_1, S_2, \cdots, S_n\}$ be a subset of sensors selected at time $t$, where $n \in \{1, 2, \cdots, m\}$. Definition of nodes $A, B, C$ and $D$ are the same to definition 1. A sensor $S_i$ at time $t$ is denoted as $O_t(S_i)$. Let $O_t = \{O_t(S_1), O_t(S_2), \cdots, O_t(S_n)\}$ represent the information variable at current time $t$ on which the sensor selection is based at time $t + 1$.

There are four main steps in information fusion process, which are sensor activation and selection, acquisition observation, system state computation and giving fusion result respectively. Firstly, after receiving $O_t$ with the system state, we can select an optimal subset of sensors to be activated at the next time slice $t + 1$; then we continue acquisition observation $O_{t+1}(S_i)$, $i = 1, 2, \cdots, n$; thirdly, we calculate the posterior probability $P(\Phi_{t+1}|O_{t+1})$ through using DBN online joint tree inference algorithm; finally, we conclude fusion result.

As above description, the uncertainty of sensor readings is a critical problem we must solve when we make information fusion process. Modified Dempster-Shafer evidence theory [35] can be used to represent the information uncertainty. However, information prior knowledge can not be dealt with this method and temporal dependency relationship between variables can not be reflected either. In this paper, we take another method for the problem. According to information theory, mutual information is defined and used to measure the statistical dependency between two random variables. Actually, the critical problem we must solve is acquisition of causality of fusion result $\Phi$ and sensor $S_i$. Hence, we apply calculation of mutual information to acquire dependency between them in order to reduce information uncertainty.

**Definition 3** For two random variables $X$ and $Y$ with a joint probability distribution $P(x, y)$ and marginal probability distributions $P(x)$ and $P(y)$, mutual information $I(X, Y)$ is defined as follows:

$$I(X, Y) = \sum_x \sum_y P(x, y) \log \frac{P(x, y)}{P(x)P(y)}.$$  
(4)

$$\therefore P(x, y) = P(x|y)P(y).$$  
(5)

$$\therefore I(X, Y) = \sum_x \sum_y P(x, y) \log \frac{P(x|y)}{P(x)}.$$  
(6)
According to above definition, mutual information $I(\Phi, S)$ for a sensor set $S = \{S_1, S_2, \cdots, S_n\}$ can be represented as

$$I(\Phi, S) = \sum_{\Phi} \sum_{S} \{P(\phi, S_1, S_2, \cdots, S_n) \log \frac{P(\phi|S_1, S_2, \cdots, S_n)}{P(\phi)}\}.$$  

(8)

In Bayesian network, we can apply mutual information to test if two variables are dependent and evaluate the strength of corresponding dependence. In theory, we claim $\Phi$ and $S$ are independent when $I(\Phi, S) = 0$ given the actual distributions of corresponding variables. In practice, given a data set $D$, we use empirical instead of theoretic distributions of variables when computing mutual information. Therefore, the normal practice is usually to set up a small threshold $\varepsilon$. When $I(\Phi, S) > \varepsilon$, we claim $\Phi$ and $S$ are mutually dependent and they must be causality relationship, on the contrary, their relationship is independent.

### 2.3 Game Model, Nash Equilibrium and Quantal Response Equilibrium

Game theory has influenced many fields, including economics (its initial focus), political science, biology, and many others. It is an integral part of artificial intelligence, theory, e-commerce, networking, and other areas of computer science [36]. One reason is application pull: the Internet calls for analysis and design of systems that span multiple entities, each with its own information and interests. Game theory is by far the most developed theory of such interactions [36]. Game theory is the study of the strategic interactions among players in which every player chooses a move based on counter-speculation of other player’s move. A solution of a game is determined by the point of equilibrium, which defines a fixed point of the player’s strategic interaction [29]. Applying game theory to the insider problem can predict the most probable a series of moves which an insider may take in the future action and enable the system defender to prepare for the move and take measures to stop it.

There are two kinds of form of stochastic game, strategic game and extensive game. A strategic game is a model of a situation in which each player chooses his plan of action once and for all, and all players’ decisions are made simultaneously. However, the model of an extensive game specifies the possible orders of events; each player can consider his plan of action not only at the beginning of the game but also whenever he has to make a decision. The form of extensive game is close to real scenario because of the dynamic interactions which take place between an insider and a system defender. In this paper, our research will focus on the extensive game.

The insider game model of extensive form is described by a 6-tuple $\Gamma = (W, E, H, L, Q_E, Q_H)$, where $W$ is a set of system states, $E$ is a set of actions of insider, $H$ is a set of actions of system defender, $L$ is a system state transition function, $Q_E$ and $Q_H$ are insider’s and system defender’s payoff functions respectively. Their detail definitions are described as follows:

System state $W = [w_1, w_2, \cdots, w_n]$ is composed of many security and performance relevant status information of computation system. For example, host resource occupation, network bandwidth used, host security status such as normal, vulnerable, probed, attacked and compromised
etc. A computation system moves from one state to another according to players’ actions in a probabilistic manner.

Insider’s action set $E = \left[e_1, e_2, \cdots, e_{m'}\right]$ is composed of actions he may take when the interaction between user and system happened. For example, these actions are included user login times, user’s operation intention and whether unauthorized access system etc. Generally speaking, a player who is an insider prefers an action over another because he may gain more benefits or pay less cost. It is the most desirable for an insider, to successfully attack the system without being stopped, or even caught. His next choice is to act normally without being stopped [30]. Therefore, action in set $E$ can be ranked preference level according to insider’s strategy and system defender’s response at different stage. The definition of set of system defender’s action is the same as insider, such as $H = \left[h_1, h_2, \cdots, h_{k'}\right]$, $h_i \in H$, $1 \leq i \leq k'$, $h_i$ is defender’s response action when $e_j$ ($e_j \in E, 1 \leq j \leq m'$) occurred. In game theory, $e_j$ is insider’s possible move and $h_i$ is system defender’s respondent move. For example, let $E = \left[e_1, e_2, e_3, e_4\right]$ and $H = \left[h_1, h_2, h_3, h_4\right]$ we instantiate them as follows:

\[
\begin{align*}
\{e_1 & = \text{normal}, e_2 = \text{mistake}, e_3 = \text{pre-attack}, e_4 = \text{attack}\} \\
\{h_1 & = \text{continue}, h_2 = \text{recommend}, h_3 = \text{warning}, h_4 = \text{rejection}\}
\end{align*}
\]

For an insider, move pair $e_i h_1$ is the most desirable strategy and the highest preference level. From insider’s view point, we rank these instantiation move pair from the highest level to the lowest level such as \(\{e_i h_1, e_j h_1, e_i h_2, e_j h_2, e_i h_3, e_j h_3, e_i h_4, e_j h_4, e_i h_2, e_i h_3, e_i h_4\}\). For a system defender, move pair $e_i h_1$ is the most desirable strategy and the highest preference level. From his view point, we also rank these instantiation move pair from the highest level to the lowest level such as \(\{e_1 h_1, e_4 h_1, e_3 h_4, e_2 h_4, e_3 h_3, e_4 h_2, e_4 h_3, e_2 h_3, e_3 h_2, e_1 h_2, \cdots, e_4 h_1\}\).

State transition function $L : W \times E \times H \times W \rightarrow [0, 1]$. The actions taken by the insider and defender at one state will drive the system to another state in a probabilistic manner. The probabilistic state transition is described by a state transition function $L$, which is represented as a set of discrete probability distributions over the set of states $W$.

Let insider’s payoff function be $Q_E : E \times H \rightarrow \mathbb{R}$ and system defender’s payoff function be $Q_H : H \times E \rightarrow \mathbb{R}$.

When both players’ (insider and system defender) strategies are optimal with regard to their counterparts’ strategies, their interactions are fixed in a way that none of them has the incentive to change to another strategy. Such a strategy pair is called a Nash equilibrium. That is, a Nash equilibrium is a set of strategies which has the property that each player’s choice is his best response to the choices of the other players. A Nash equilibrium offers a credible prediction of the insider’s moves because it gives the insider the best he can get given the defender’s strategy. It also identifies the defender’s best countermeasure to the insider’s strategy [29]. However, in real scenario, not all attackers are rational players as assumed in most game-theoretic studies. Instead, a number of them may be irrational or naive attackers who simply do not care about being detected [12]. So Nash equilibrium calculation for solving the game is not appropriate under this circumstance, and we need a new equilibrium calculation to solve the game. Quantal response equilibrium (QRE) [30, 31, 32] is proposed to handle the situation. Quantal response equilibrium is a solution concept in game theory and it provides an equilibrium notion with bounded rationality. It can give significantly different results than Nash equilibrium because it can be used to capture players’ bounded rationality. It is a generalization of Nash equilibrium, which has also been used to give reasons why players deviate from the equilibrium path [30]. In
quantal response equilibrium, players are assumed to make errors in choosing which pure strategy to play. The probability of any particular strategy being chosen is positively related to the payoff from that strategy, which means that very costly errors are unlikely.

There are two players in our insider game model, insider and defender. Let the set of strategy profiles \( V \) be the Cartesian product of \( E \) and \( H \), 
\[
V = E \times H.
\]
Let \( m' \) be the number of strategies in \( E \) and \( k' \) be the number of strategies in \( H \) and define \( J = m' + k' \). For simplicity, we call insider as player \( i \) and defender as player \( d \). A mixed strategy for each player is a probability distribution over the set of their strategies respectively. Let \( \Delta_i \) denote the set of such distributions for player \( i \) and \( \Delta_d \) for player \( d \). The set of mixed strategy profiles is 
\[
\Delta = \Delta_i \times \Delta_d.
\]
A typical mixed strategy profile will be denoted by \( \pi \in \Delta \). The payoff functions for the players are extended over the set of mixed strategies in the usual expected value way. According to literatures [31] and [32], we review and refine quantal response equilibrium definition of insider game model related as follows.

**Definition 4** Let \( \pi_{lj} \) represent the probability distribution of a strategy of someone player. For example, when \( l = i \), \( 1 \leq j \leq m' \), \( \pi_{ij} \) represent the probability distribution of a strategy of player \( i \); Similarly, when \( l = d \), \( 1 \leq j \leq k' \), \( \pi_{dj} \) represent the probability distribution of a strategy of player \( d \). Let \( s_{lj} \) represent one of strategy element, \( l = i \), \( s_{ij} = e_j \), \( 1 \leq j \leq m' \); \( l = d \), \( s_{dj} = h_j \), \( 1 \leq j \leq k' \).

**Definition 5** Let \((s_{lj}, \pi_{-l})\) denote the mixed strategy profile where player \( i \) or \( d \) plays strategy \( s_{lj} \) with probability one, while another player plays according to the mixed strategy \( \pi_{-l} \). Thus, we have follows:
\[
\mu_l : \Delta \to \mathbb{R}^{ij} \times \mathbb{R}^{jd}.
\]
\[
\mu_{lj}(\pi) = \mu_l(s_{lj}, \pi_{-l}). \tag{10}
\]

From above formulas, \( \mu_{lj} \) is the expected payoff to one of players from playing his \( j \)th strategy, holding fixed his opponents’ mixed strategy \( \pi_{-l} \).

The QRE [31, 32] notion is based on the introduction of payoff perturbations associated with each action of each player. So we represent it as
\[
\hat{\mu}_{lj}(\pi) = \mu_{lj}(\pi) + \varepsilon_{lj}. \tag{11}
\]
where the \( \varepsilon_{lj} \) is random variable drawn according to some joint distribution. When the \( \varepsilon_{lj} \) is chosen independently from an extreme value distribution with non-negative parameter \( \lambda \), a logistic quantal response equilibrium profile is defined
\[
\pi_{lj} = \delta_{n''} \times e^{\lambda \mu_{lj}(\pi)} \div \left( \sum_{r=1}^{j} e^{\lambda \mu_{lr}(\pi)} \right) \tag{12}
\]
Thus, the set of logit equilibria can be viewed as a correspondence mapping non-negative parameter \( \lambda \) into a set of mixed strategy profiles in \( \Delta \). Actually, \( \pi_{lj} \) is the probability distribution and the probability of player \( l \) choosing strategy \( j \) of respective strategy action sets. \( \delta_{n''} \) is a user behavior attribute ranking parameter. \( \lambda \) can be thought of as the rationality parameter or logit precision parameter. When \( \lambda \to 0 \), players become “completely irrational” and play each strategy with equal probability. When \( \lambda \to \infty \), players become “perfectly rational” and play approaches a Nash equilibrium. Therefore, QRE calculation can be used to integrate smart and naïve insider description in one algorithm.
2.4 Game Theory and Information Fusion-based User Behavior Predicting Algorithm

As we know, optimal sensor selection will need computational cost but maximize a utility for information fusion. So we define the utility function which consisted two parts, information gain $\gamma_1$ and the cost $\Omega(S)$ to activate sensors $S$. Let $\gamma_2 = \Omega(S)$, $\gamma_1$ and $\gamma_2$ are mutually independent, the utility function can be represented as

$$R(\gamma_1, \gamma_2) = (a \times \gamma_1 + 1)(b \times \gamma_2 + 1).$$

(13)

In this definition, $a$ and $b$ are the normalization parameters which have the characteristic such as $a + b = 1$.

According to above analysis, we acquire fusion information $P(\Phi|O_t)$ through mutual information $I(\Phi,S)$ and cost $\Omega(S)$. Hence, $\gamma_1 = I(\Phi,S)$, $R(\gamma_1, \gamma_2) = R(I(\Phi,S), \Omega(S))$.

$$R(I(\Phi,S), \Omega(S)) = a \times b \times I(\Phi,S) \times \Omega(S) + a \times I(\Phi,S) + b \times \Omega(S) + 1.$$  

(14)

The purpose of information fusion is to compute $P(\Phi|O_t)$ and we will obtain it by DBN inference [34]. Since inference in DBN can use inference in BN which is a subroutine, we choose junction tree algorithm as our inference algorithm. All hidden nodes of DBN are discrete with discrete parents and their observed nodes have any distribution while all their parents are discrete or observed. The network characteristic and its attribute satisfied conditions of exact inference algorithm. Hence, exact inference algorithm-junction tree is selected to be DBN inference. On the other hand, we can speed up exact discrete inference by using causal independence which is obtained by $I(\Phi,S)$. When $I(\Phi,S) < \varepsilon$, we claim $\Phi$ and $S$ are mutually independent.

Firstly, we construct our DBN through Fig. 1 and Fig. 2 according to its framework and variable’s definitions. There are four levels in the DBN which are fusion result level, user behavior attribute level, information level and observable sensor level respectively. The parameters of CPD can be computed through prior knowledge. We assumed that we have deployed limited number of sensors because if sensor’s number increased very largely and quickly, DBN’s exact inference will be a NP-hard problem. We must control and limit the sensor’s number to be an appropriate size. Secondly, we collect information from each sensor. Thirdly, we compute mutual information $I(\Phi,S)$ and optimal sensor subset by utility function $R(I(\Phi,S), \Omega(S))$. While we obtain enough sensor information, DBN inference will be carried out in order to decide whether security events occurred. If the decision is true about event happening, we construct game model of insider and defender as different players. In this game model, QRE will be computed according to each player’s payoff functions, and the results of QRE will give us the players’ next future move (behavior) trend based on probability. Computational system administrator can take measures according to the results in order to stop any early insider’s action. The algorithm is described as follows:

**Step 1** Initialization $t \leftarrow 0$

**Step 2** Build DBN structure, compute CPD parameters

**Step 3** Compute $I(\Phi,S)$ and $\Omega(S)$

**Step 4** Compute $R(I(\Phi,S), \Omega(S))$
Step 5 Acquire subset $S' = \arg \max_S R(I(\Phi, S), \Omega(S))$

Step 6 While $t < T$ and $n < N$

Step 7 Get $O_t$ from each activate sensor $S_i$, $S_i \in S$

Step 8 $P(\Phi|O_t) = \text{Junction\_tree\_dbn(dbn\_structure, CPD\_parameter})$

Step 9 Compute $I(\Phi, S)$ and $\Omega(S)$

Step 10 Compute $R(I(\Phi, S), \Omega(S))$

Step 11 if $R(I(\Phi, S), \Omega(S)) > \varepsilon$, construct insider game model

Step 12 $\lambda \leftarrow 0$, each player’s behavior probability is assigned averagely

Step 13 Compute $\pi_{ij}$ iteratively and increase $\lambda$ at the same time

Step 14 if $\lambda$ reaches end point value, stop computing $\pi_{ij}$

Step 15 else goto Step 13

Step 16 $S' = \arg \max_S R(I(\Phi, S), \Omega(S))$

Step 17 $t = t + 1$, goto Step 6

3 Simulation Experiments and Results Discussion

We experiment with a simple example for clarifying the basic notions and we assume that we have some monitor sensors for acquiring user’s operation behavior in the computational system. Actually, these sensors are monitor process which can get our source information for fusion and analysis. Fig. 1 presents a BN model for such a system. Since there are four type user behavior attributes and eleven information variables respectively, we have four type monitor processes at least to obtain these eleven source evidences. We have had external modules that can receive monitor process data and make the data available as input evidence to the DBN model. Then we select the most decision relevant sensors in order to make a timely decision and construct a game model. In our experiments, we set mutual information decision parameter $\varepsilon = 0.2$ for four type behaviors of insider and $\varepsilon = 0.3$ for five type behaviors of insider respectively, we construct two different stages game of insider and use the same tool in [30].

Table 1, Fig. 3 and Fig. 4 are the results of $\varepsilon = 0.2$, When $R(I(\Phi, S), \Omega(S)) > 0.2$, some security events happened and the insider game model will be built. By QRE computing, we can acquire insider’s action trend in the next move after security events occurred. In table 1 and Fig. 3, we find that no probability has changed very significantly up to step 8, which means that no preference of insider has been indicated. When we continue QRE calculation, the probability of each type behavior and parameter $\lambda$ are changing gradually. $\lambda$ is increasing slowly and insider’s behaviors begin to return rationality. The probability of normal has increased until step 22, and then it has declined gradually from step 23 to the end of QRE calculation. The reason is that insider wants to avoid detection by covering his illegal actions under normal behavior. However, in order to attain insider’s final goal, he wants to attack really. So the probabilities of mistake and
pre-attack are declining continually and attack probability is increasing constantly. Pre-attack is a stepping stone stage and its probability has declined gradually until end of QRE calculation. Finally, when $\lambda$ reached end value point, all probabilities are almost zero except attack is one. The insider’s attack intent is very clearly from RQE calculation process. On the other hand, defender’s

Table 1: QRE Calculations for Four Type Behaviors of Insider and System

<table>
<thead>
<tr>
<th>Step</th>
<th>$\lambda$</th>
<th>Probabilities of four type behaviors of insider</th>
<th>Probabilities of system</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Normal</td>
<td>Mistake</td>
</tr>
<tr>
<td>1</td>
<td>0.0000</td>
<td>0.2500</td>
<td>0.2500</td>
</tr>
<tr>
<td>2</td>
<td>0.0022</td>
<td>0.2519</td>
<td>0.2475</td>
</tr>
<tr>
<td>3</td>
<td>0.0046</td>
<td>0.2540</td>
<td>0.2448</td>
</tr>
<tr>
<td>4</td>
<td>0.0073</td>
<td>0.2563</td>
<td>0.2417</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>13</td>
<td>0.0471</td>
<td>0.2864</td>
<td>0.1957</td>
</tr>
<tr>
<td>14</td>
<td>0.0539</td>
<td>0.2905</td>
<td>0.1879</td>
</tr>
<tr>
<td>15</td>
<td>0.0612</td>
<td>0.2946</td>
<td>0.1794</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>46</td>
<td>0.8393</td>
<td>0.0337</td>
<td>4.08E-05</td>
</tr>
<tr>
<td>47</td>
<td>0.9084</td>
<td>0.0257</td>
<td>1.80E-05</td>
</tr>
<tr>
<td>48</td>
<td>0.9854</td>
<td>0.0190</td>
<td>7.18E-06</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>62</td>
<td>6.8173</td>
<td>1.44E-12</td>
<td>2.96E-36</td>
</tr>
<tr>
<td>63</td>
<td>7.4946</td>
<td>9.56E-14</td>
<td>8.75E-40</td>
</tr>
<tr>
<td>64</td>
<td>8.2395</td>
<td>4.86E-15</td>
<td>1.15E-43</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>164</td>
<td>835650.4852</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>165</td>
<td>919215.5292</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>166</td>
<td>1071137.078</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

Fig. 3: QRE for Four Type Behaviors of Insider
counteraction is dynamic changing from viewing Table 1 and Fig. 4. Probability of “continue” is declining the fastest in all of counter-measures, “recommend” and “warning” are declining mildly and “rejection” is increasing properly because appropriate counter-measures must be taken in time according to the game solution result.

Similarly, Table 2, Fig. 5 and Fig. 6 are the results of $\varepsilon = 0.3$, $R(I(\Phi, S), \Omega(S)) > 0.3$. The

<table>
<thead>
<tr>
<th>Step</th>
<th>$\lambda$</th>
<th>Probabilities of five type behaviors of insider</th>
<th>Probabilities of system</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Normal</td>
<td>Mistake</td>
</tr>
<tr>
<td>1</td>
<td>0.0000</td>
<td>0.2000</td>
<td>0.2000</td>
</tr>
<tr>
<td>2</td>
<td>0.0019</td>
<td>0.2018</td>
<td>0.1988</td>
</tr>
<tr>
<td>3</td>
<td>0.0040</td>
<td>0.2038</td>
<td>0.1974</td>
</tr>
<tr>
<td>4</td>
<td>0.0063</td>
<td>0.2060</td>
<td>0.1959</td>
</tr>
<tr>
<td>13</td>
<td>0.0406</td>
<td>0.2383</td>
<td>0.1715</td>
</tr>
<tr>
<td>14</td>
<td>0.0465</td>
<td>0.2435</td>
<td>0.1670</td>
</tr>
<tr>
<td>15</td>
<td>0.0529</td>
<td>0.2489</td>
<td>0.1620</td>
</tr>
<tr>
<td>50</td>
<td>0.8530</td>
<td>0.0319</td>
<td>3.47E-05</td>
</tr>
<tr>
<td>51</td>
<td>0.9243</td>
<td>0.0242</td>
<td>1.49E-05</td>
</tr>
<tr>
<td>52</td>
<td>1.0037</td>
<td>0.0177</td>
<td>5.78E-06</td>
</tr>
<tr>
<td>65</td>
<td>6.9325</td>
<td>9.06E-13</td>
<td>7.43E-37</td>
</tr>
<tr>
<td>66</td>
<td>7.6204</td>
<td>5.78E-14</td>
<td>1.93E-40</td>
</tr>
<tr>
<td>67</td>
<td>8.3770</td>
<td>2.80E-15</td>
<td>2.20E-44</td>
</tr>
<tr>
<td>68</td>
<td>9.2093</td>
<td>1.00E-16</td>
<td>1.01E-48</td>
</tr>
<tr>
<td>168</td>
<td>8.49E5</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>169</td>
<td>9.34E5</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>170</td>
<td>1.07E6</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
</tbody>
</table>
changing feature of probability is near the above experiment results because both of them have the same payoff function value. Furthermore, as stepping stone stage, pre-attack1 and pre-attack2 are declining very similarly because of their same payoff function value. Compare Fig. 4 and Fig. 6, we find that their curve feature is very similar because the insider’s attack, whether has one stepping stone or two stepping stones or even more stepping stones, has the same severe result to the computational system, which disable its service function. Hence, we must take the
same serious measures to stop them.

The influence factor of algorithm performance is mainly at DBN exact inference and QRE iterative computation. Let $M$ represents the maximum number of values each discrete hidden node of DBN can take and represents the size of the largest clique of junction tree, so the computation cost of DBN exact inference is $O(M^c)$. In our case, we have $M \leq 6$ and $c \leq 4$. Let $\eta$ represents the iterative numbers of QRE calculation, the QRE iterative computation cost is $O(\eta)$, and in our case, we also have $\eta \leq 170$. So the total computation cost is $O(\eta \times M^c)$, although our algorithm has bigger cost than the method in [30], we have improved 17% convergence and precision in [30].

4 Conclusions and Future Work

In this paper, we focus on situation awareness of insider threats by game theory and information fusion based algorithm. We build DBN model in order to make monitor information fusion and avoid shortcoming of traditional IDS. And we have used junction tree algorithm as DBN exact inference method because we can acquire better accurate detection result although we must pay additional but accepted computation cost. After obtaining information fusion result, game is built and QRE is computed that show us the probabilities of both players’ actions evolution. Our method reduces uncertainty of information acquiring before constructing game between both of players and can give system defender more accurate actions of insider in order to make correct decision after constructing game. On the other hand, QRE calculation can describe rational or irrational players of the game and it provides an equilibrium notion with bounded rationality. Hence, QRE calculation can give results that show how an insider’s future behavior which may be cyber insider threats situation awareness. From simulation experiments, we can find that DBN exact inference may have more computation cost than other algorithm [30], but we obtain improved convergence and precision. QRE calculation process indicates that it can be used to capture players’ bounded rationality and give the future behavior situation awareness of insider based on the probability.

In the next research, we will continually optimize DBN structure and inference algorithm in order to reduce its computation cost and improve execution performance. For QRE iterative calculation, we will also continuously improve its payoff function’s definition and computation for more fast convergence.

Acknowledgement

This work is supported in part by the National High-Tech Research and Development Plan of China under Grant No. 2008AA01A303 and the National Science and Technology Supporting Plan of China under Grant No. 2008BAH37B3.

References


