A Game Theoretic Approach to Decision and Analysis in Network Intrusion Detection [1]

Tansu Alpcan and Tamer Başar
Coordinated Science Laboratory, University of Illinois
1308 West Main Street, Urbana, IL 61801 USA
(alpcan, tbasar)@control.csl.uiuc.edu

***Abstract*— We investigate the basic trade-offs, analysis and decision processes involved in information security and intrusion detection, as well as possible application of game theoretic concepts to develop a formal decision and control framework. A generic model of a distributed intrusion detection system (IDS) with a network of sensors is considered, and two schemes based on game theoretic techniques are proposed. The security warning system is simple and easy-to-implement, and it gives system administrators an intuitive overview of the security situation in the network. The security attack game, on the other hand, models and analyzes attacker and IDS behavior within a two-person, nonzero-sum, noncooperative game with dynamic information. Nash equilibrium solutions in closed form are obtained for specific subgames, and two illustrative examples are provided.**

I. Introduction

Today, communication and computer networks are an indispensable part of the modern society with the prime example being the Internet. Since they bring extensive computing and communication capabilities to individuals, businesses, and organizations over long distances, and often on a global scale, networked systems are being deployed in every level of the society at an exponential rate. Existing information structures and the way information is processed in organizations are increasingly transferred to virtual environments. These revolutionary changes in information systems also pose unique problems. Network security is among the most important of these [1], and hence, it has been extensively investigated in the research community.

The distributed nature of the contemporary networks and complexity of the underlying computing and communication environments prevent administrators and organizations from having absolute control on their networks. Furthermore, network boundaries are often vague, and administrators cannot exercise control outside their local domain [2], [3], which leaves networked systems vulnerable to distant security attacks due to global connectivity. This results in a perpetual struggle between attackers who aim to intrude the deployed systems and security administrators trying to protect them. Emerging security issues such as this cannot be fully addressed by classical approaches like policing. Although technologies like firewalls, encryption, and authentication can harden the network against attacks, they fail to address issues in the case an attack is (partially) successful.

Intrusion detection systems (IDSs) extend the information security paradigm beyond traditional protective and reactive network security. They monitor the events in the networked system and analyze them for signs of security problems [4]. Hence, they increase the controlling ability of the system administrator, and help him react to security problems. Current IDSs rely mostly on human intervention in the decision and response processes against attacks, that are often automatic and script-based. In other words, the equivalence of a strategic decision making and command-and-control in battleground management is missing [5]. Hence, today's IDSs are inefficient and delayed in responding security breaches in the network. Furthermore, due to the distributed nature of the networked system a centralized security system poses scalability and efficiency problems. Utilization of autonomous software agents (ASAs) in developing distributed IDSs has recently been proposed to address the issues of automatization and scalability [6], [7]. However, a distributed IDS architecture based on ASAs still needs a decision making mechanism which requires as little human intervention as possible.

Given the current overview of the information security and intrusion detection, there is definitely a need for a decision and control framework to address issues like attack modeling, analysis of detected threats, and decision on response actions. A rich set of tools have been developed within game theory to address problems where multiple players with different objectives compete and interact with each other on the same system, and they are successfully used in many disciplines including economics, decision theory, and control. Therefore, game theory is a strong candidate to provide the much needed mathematical framework for analysis, modeling, decision, and control processes for information security and intrusion detection. Such a mathematical abstraction is useful for generalization of problems, combining the existing ad-hoc schemes under a single umbrella, and future research. Furthermore, using game theoretic tools it is also possible to develop practical schemes which can be integrated with existing intrusion detection systems. Because of these reasons, application of game theory to network security area has recently been a topic of interest [8], [9].

In this paper, we investigate the basic decision and analysis processes involved in information security and intrusion detection, as well as possible usage of game theory for developing a

[1]Supported in part by a grant from the Boeing Company through the Center for Trustworthy Systems (CTNS) at the University of Illinois at Urbana-Champaign, and in part by NFS through Grant CCR 00-85917.

formal decision and control framework. We develop a generic model for distributed IDSs by defining a network of sensors, and propose two simple, flexible, and easy-to-implement schemes utilizing both cooperative and noncooperative game theoretic concepts [10], [11]. In Section II, common tradeoffs in information security are investigated. We introduce a game theoretic framework for distributed IDSs and two schemes making use of game theoretic concepts in Section III, which is followed by the concluding remarks of Section IV.

II. Tradeoffs in Network Security

There are several common tradeoffs in designing a security system whether it is a simple mechanical door-lock protecting a house or a large-scale distributed IDS. A basic tradeoff is the one between security risk and ease of access: The more a system is protected by security mechanisms the more difficult it becomes to access it, hence less convenient, and vice versa. A simple real life example is keeping a frequently used door locked against intruders versus leaving it unlocked as a security hole. Similarly, in the context of computer networks users want to access data easily and instantly while expecting the sensitive data to be protected from unauthorized access. Obviously, achieving the latter requires sacrificing some of the convenience of the former by deploying and maintaining authentication and security mechanisms between the user and the sensitive data. A related tradeoff is the overhead caused by the IDS or by other security systems like firewalls to the networked system. Deploying intrusion detection mechanisms like packet filters, log analyzers, etc. have a cost in terms of system resources (bandwidth, memory, CPU time).

A basic performance criterion for an IDS is the false alarm rate. There exists a tradeoff between the reduction in the false alarm rate by decreasing the system sensitivity and the increase in the rate of undetected intrusions. Clearly, on either extreme the IDS becomes totally ineffective. Therefore, the IDS should satisfy some upper and lower bounds on false alarm rate and undetected intrusions according to the specifications of the deployed network. Finally, in distributed IDSs with software agents there exists the question of how autonomous the agents should be. The robustness of a completely decentralized decision process has to be weighed against the communication overhead between the agents and the processing overhead of individual agent programs.

All these tradeoffs, among others, have to be taken into account in the design and deployment of an IDS. Furthermore, the decision and analysis mechanisms of the IDS should be scalable and flexible enough to allow configuration for the specific network at hand.

III. A Game Theoretic Framework

We construct two game theoretic schemes to address some of the issues in Section II. While the foremost goal of the first scheme is simplicity and ease-of-implementation, the second one models and analyzes attacker and IDS behavior within a two-person, nonzero-sum, noncooperative game framework. We also note that both schemes are flexible and can be implemented regardless of the underlying architecture.

A. The Model

Consider a distributed IDS with a network of sensors, $\mathcal{S} := \{s_1, s_2, \ldots, s_P\}$, where a sensor is defined as an autonomous software (agent) that monitors and reports possible intrusions or anomalies occurring in a subsystem of a large network using a specific technique like signature comparison, pattern detection, statistical analysis etc. The system monitored by the IDS can be represented as a set of subsystems, $\mathcal{T} = \{t_1, t_2, \ldots, t_M\}$, which may be targeted by an attacker. We note that these subsystems can be actual computer programs or parts of the network as well as abstract (business) processes distributed over multiple hosts. Define $\mathcal{I} = \{I_1, I_2, \ldots, I_K\}$ as the set of documented threats and detectable anomalies, which may indicate a possible intrusion. The properties of an element of $\mathcal{I}$ can be further described by assigning it to one or more function classes among $\{\mathcal{F}_1, \mathcal{F}_2, \ldots\}$, where each function class, $\mathcal{F}$, represents a common property of its members. For example, $\mathcal{F}_1$ may represent web services, and if $I \in F_1$ then it means that I is an intrusion signature or anomaly related to web services.

A sensor can possibly detect more than one anomaly or possible intrusion. Let us define, using a one-to-many mapping from the set $\mathcal{S}$ to the set $\mathcal{I} \cup \{0\}$, the output vector of the network of sensors, $\mathbf{d} := [d_1, d_2, \ldots, d_N]$, where $N \geq P$. The i^{th} element of the output vector associated with the sensor $s_j \in \mathcal{S}$, $d_i(s_j)$, is equal to a single $I_k \in \mathcal{I}$ if the sensor has detected the possible intrusion or anomaly I_k. Otherwise, $d_i(s_j) = 0$. We note that each sensor can report at most one of each type of possible intrusions. Hence, $d_i(s_k) \neq d_j(s_k)\ \forall i, j$ of a given $s_k \in \mathcal{S}$, unless $d_i(s_k) = d_j(s_k) = 0$. Finally, we define the system matrix, A, describing the relationship between the sensor output vector and subsystems as

$$A_{i,j} = \begin{cases} 1, & \text{if sensor j monitors subsystem i} \\ 0, & \text{if sensor j does not monitor subsystem i} \end{cases},$$

where $i \in \mathcal{T}$ and $j \in \mathbf{d}(s)$.

B. The Security Warning System for Distributed IDS

Large number of false-alarms is a considerable problem in the world of IDS deployment [5], which can be overwhelming for system administrators. It also increases the difficulty in managing an IDS significantly. There is no immediate solution to the false-alarm problem at the sensor level. Hence, a new approach is needed for interpreting sensor data. We devise an easy-to-implement yet flexible scheme using intrusion warning levels. The security warning system enables IDS to operate in different modes at each security level, and to switch automatically between the different levels. In addition, it provides the administrator an intuitive overview of the current security situation in the network.

We make use of the model introduced in Section III-A. Define $f : \mathcal{I} \cup \{0\} \Rightarrow \mathbb{R}^+ \cup \{0\}$ as a one-to-one function assigning a positive real number to each element of $\mathcal{I}$ and

$f(0) = 0$. Hence, each documented intrusion signature and anomaly is associated with a so called *security risk value* quantified with a positive real number, $f(I)$. The function f can also be defined in such a way that it assigns security risk values to function classes $\mathcal{F}$ instead of individual elements of $\mathcal{I}$. The security warning system is based on the concept of *security level*, $\mathcal{L}$. We define L security levels, $\{l_1, \ldots, l_L\}$, with $0 < m_1 < m_2 < \ldots < m_L$ being the corresponding threshold values. Given the sensor output vector $\mathbf{d}$, the security level, $\mathcal{L}$, of the IDS is equal to l if the sum of the security risk values of detected intrusions falls in the interval $[m_l, m_{l+1}]$:

$$\mathcal{L} = \begin{cases} l_1, & \text{if } \sum_{i=1}^{N} f(d_i) < m_1 \\ l_j, & \text{if } m_{j-1} \leq \sum_{i=1}^{N} f(d_i) < m_j \\ l_L, & \text{if } \sum_{i=1}^{N} f(d_i) \geq m_L. \end{cases}$$

Thresholds and security risk values can be assigned and fine tuned according to previous experience and specifications of the network. It is also possible to use learning-based techniques to obtain these values if there is sufficient amount of prior data on the specific system.

Cooperative game theory provides a suitable framework for the design and analysis of the proposed security warning scheme. The sensor output, $\mathbf{d}$, can be modeled as an *N-person game* with $\mathcal{N} := \{1, 2, \ldots, N\}$ being the set of *players*, and each subset $\mathcal{D} \subset N$, where $d_i \neq 0 \ \forall i \in \mathcal{D}$, is called a *coalition* [11, p. 213]. Thus, each such subset of $\mathcal{N}$ (*coalition*) represents an observed threat pattern. The aggregate value of the coalition is defined as the sum of the security risk values of the detected threats, $\sum_{j \in \mathcal{D}} f(d_j)$. The security level thresholds, m, determine whether the IDS security level $\mathcal{L}$ changes or not.

In order to analyze the relative importance of each sensor output, which indicates an intrusion threat, with respect to others and the effect of the threshold values on security levels, we make use of a power index called *Shapley value* [11]. Shapley value approach has been utilized in multi-agent coalitions in earlier studies [12] albeit in different contexts. Let $f(C) := \sum_{i \in C} f(d_i), \ d_i \in \mathcal{I}, C \subset \mathcal{N}$ be the value of the *coalition* C with cardinality c. Then, Shapley value of the i^{th} element of the sensor output vector is defined by

$$\begin{aligned} &\phi(i) := \textstyle\sum_{C \subset \mathcal{N}} \dfrac{(c-1)!(N-1)!}{N!}[f(C) - f(C - \{i\})] \\ &\Rightarrow \phi(i) := \textstyle\sum_{C \subset \mathcal{N}} \dfrac{(c-1)!(N-1)!}{N!} f(i) \end{aligned} \tag{1}$$

In determining the effect of the i^{th} sensor output on the k^{th} warning level, however, the formula simplifies to

$$\phi(i) := \sum_{C' \subset \mathcal{N}} \frac{(c-1)!(N-1)!}{N!}, \tag{2}$$

where C' denotes the 'winning' coalitions with $\sum_{i \in C'} f(d_i) > m_k$. In this case, the game is said to be *simple*. Notice that the computational complexity of Shapley value is of $o(2^N)$. Therefore, it becomes impractical to calculate the exact value for large N. Instead, we make use of multilinear extensions to approximate the Shapley value as N gets large [11, p. 296]. Since this approximation is based on the law of large numbers, it becomes more accurate as $N \to \infty$.

The Shapley value of the i^{th} sensor output indicates the relative security risk value for a given threshold (of a warning level). It hence plays an important role in the analysis and fine tuning of the warning system for the specific network to be deployed by providing a guideline for choosing security risk values, $f(.)$, and thresholds of the levels, m. In the security warning system, each level can be associated with a different operation mode where agents and control mechanisms of IDS behave accordingly. The sensitivity of the agents and security measures in the network (e.g. firewall configurations, authentication mechanisms) are increased with the increasing security level. Therefore, security warning scheme addresses the problem of optimizing some of the network security tradeoffs discussed in Section II. A specific alert mechanism and a user interface can also be incorporated into this scheme to inform administrators. A simplified flow chart depicting principles of the security warning algorithm is given in Figure 1.

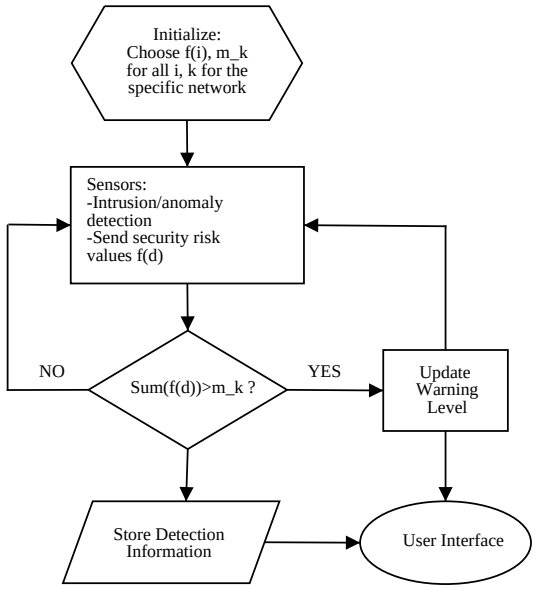

Fig. 1. A simplified algorithm for the security warning system.

An Illustrative Example: Consider a three level security warning system consisting of *green*, *yellow*, and *red* levels with thresholds $m_{yellow} = 35$ and $m_{red} = 100$. The set of known intrusions (and anomalies) $\mathcal{I}$ is chosen to be small for illustrative purposes, and the corresponding risk values are $\{10, 20, 30, 40, 90\}$. Based on the sensor configuration, the sensor output vector is $\mathbf{d} = [10, 10, 10, 20, 20, 30, 30, 30, 40, 40, 90, 90]$. In order for the system to switch to yellow (respectively, red) , sum of the values of observed intrusions within a predefined time interval has to exceed the threshold 35 (respectively, 100). Since

cardinality of $\mathbf{d}$ is small, the Shapley values can be calculated exactly using the formula in (2). As observed in Figure 2, the Shapley values of intrusions with high security risk values are larger for the red threshold than the ones for the yellow threshold. This indicates that intrusions with high values play a more significant role than others for switching to the red level.

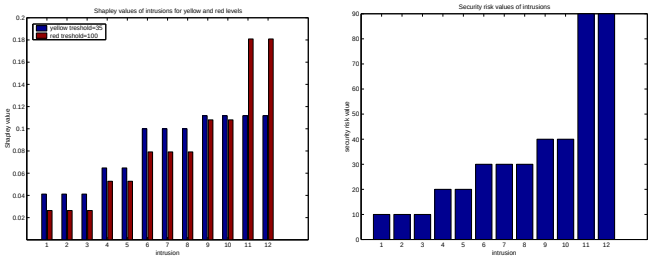

Fig. 2. Shapley values for yellow and red levels, and security risk values are shown for the sensor output vector $\mathbf{d}$ with 12 elements.

We next increase the cardinality of $\mathbf{d}$ to 60, and thresholds for yellow and red levels to 40 and 1000, respectively. In this case, Shapley values are calculated approximately as exact calculation takes excessive amount of time. In this case the results are similar to the ones in Figure 2, and hence, have not been displayed here.

C. Game Theoretic Modeling of Security Attacks

Today's intrusion detection architecture is a passive information processing paradigm [5]. However, with the security attacks becoming more frequent and sophisticated, IDSs fail to distinguish the real intent and target of attackers. In order to accurately identify the target of an attack, IDS should be able to process the attack information within a context. Deploying a network of sensors in the system, and through game theoretic analysis of the sensor output data one can model attacker's behavior, intent, and target. Furthermore, due to the flexibility of the model in Section III-A it is possible to capture not only attacks targeting specific portions of the network but also abstract targets such as processes distributed over multiple physical subsystems. In addition to modeling attacker's behavior and intent, the game theoretic framework may also be used to analyze and model IDS's response process by taking the basic security tradeoffs in Section II into account. The IDS response actions vary from setting a simple alarm to a costly system reconfiguration, which may involve shutting down some relatively less important services in the system.

We model the interaction between the attacker and the IDS as a two-person, non-zero sum, single act, finite game with dynamic information. Given the sensor output vector $\mathbf{d}$, we obtain for each subsystem $t \in \mathcal{T}$ a threat level, y_t, using the system matrix A. Hence, we define the threat level vector as

$$\mathbf{y} := A\,\mathbf{d}.$$

The elements of $\mathcal{T}$ (the set of subsystems) are then grouped into non-overlapping information sets according to their respective threat levels in $\mathbf{y}$. To simplify the analysis, we assume that the attacker targets only a single subsystem. Hence, the actions available to the attacker in each information set is to attack a single subsystem in the set or do nothing, which indicates a false alarm in related sensors. We also limit the actions of the IDS to set an alarm for one target in the information set or do nothing. Since the IDS can distinguish between information sets but not actions within them, it is a dynamic information game. Figure 3 depicts a sample security game, where $t1, t2, t3$ denote the attacker's actions of targeting subsystems 1 to 3, $nt1, nt2$ indicate false alarms (attacker doing nothing), $a1, a2, a3$ represent the IDS's alarms for respective subsystems, and $na1, na2$ denote the IDS choosing not to set an alarm.

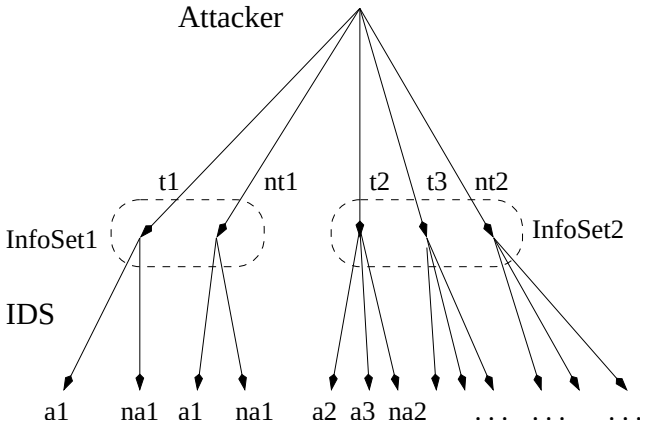

Fig. 3. A simple security game with 3 subsystems and 2 information sets.

We investigate the security game in Figure 3 recursively. Information set 1 is the simplest case, where the attacker either targets subsystem one (t1) or does nothing (nt1) equivalent to a false alarm. The set of actions of the IDS are either to set the alarm for subsystem one (a1) or do nothing (na1). This portion of the game can be represented by the following 2×2 bimatrix game

$$M_{att} := \begin{array}{|c|c|c|} \hline \text{t1} & \beta_h & -\beta_s \\ \hline \text{nt1} & 0 & 0 \\ \hline & \text{a1} & \text{na1} \\ \hline \end{array} \qquad M_{ids} := \begin{array}{|c|c|c|} \hline \text{t1} & -\alpha_h & \alpha_m \\ \hline \text{nt1} & \alpha_f & 0 \\ \hline & \text{a1} & \text{na1} \\ \hline \end{array} \tag{3}$$

where the entries of M_{ids} (M_{att}) represent the cost values, and columns (rows) correspond to the strategy spaces of the IDS and the attacker, respectively. The value $-\alpha_h$ is the gain of the IDS for detecting the target. On the other hand, α_f and α_m are the IDS's costs for false alarm and missing the attack, respectively. The cost β_h represents the detection penalty for the attacker whereas $-\beta_s$ represents the gain from an undetected intrusion. Notice that, although missing an attack is associated with a cost for the IDS, false alarms cost nothing to the attacker. The parameters α and β are always positive unless otherwise stated.

The *min-max* or *security strategy* of a player [10] guarantees a maximum cost, or so called *security level* regardless of the strategy of the opponent. Due to the detection cost of an attack, $\beta_h > 0$, the attacker's security strategy is not to attack at all

(nt), which guarantees an upper bound on the cost of zero. The IDS's security strategy, however, depends on the relative values of α_f and α_m, false alarm and missing (an attack) costs. If $\alpha_f > \alpha_m$ then the IDS chooses not to alarm at all (na), and if $\alpha_f < \alpha_m$ then the IDS always sets on the alarm (a1). We note that the security strategies are extremely conservative in this setting, and give little insight into the dynamics of the game.

We next investigate the existence of a Nash equilibrium (NE) in the matrix game (3). Clearly, there is no NE in pure strategies. Therefore, we extend the analysis by considering mixed strategies of players defined as probability distributions on the space of their pure strategies [10, p.23]. Let p_1 and $1-p_1$ be the probabilities for strategies (t1) and (nt) of the attacker, respectively. Also let q_1 and $1-q_1$ be the probabilities for strategies (a1) and (na) of the IDS. The pair (p^*, q^*) is said to constitute a noncooperative NE solution to the bimatrix game (M_{att}, M_{ids}) if the following inequalities are satisfied:

$$\begin{aligned} p_1^*(\beta_h q_1^* - \beta_s(1-q_1^*)) \leq p_1(\beta_h q_1^* - \beta_s(1-q_1^*)), \\ p_1^*\alpha_m + q_1^*[\alpha_f - (\alpha_f + \alpha_h + \alpha_m)p^*] \\ \leq p_1^*\alpha_m + q_1[\alpha_f - (\alpha_f + \alpha_h + \alpha_m)p^*], \end{aligned} \tag{4}$$

where $0 \leq p_1, q_1 \leq 1$. The only solution to the set of inequalities in (4) constitutes the unique NE of the game, and is given by

$$p_1^* = \frac{\alpha_f}{\alpha_f + \alpha_h + \alpha_m}, \text{and} \quad q_1^* = \frac{\beta_s}{\beta_h + \beta_s}. \tag{5}$$

Note in (5) an interesting, and rather counter-intuitive, feature of NE solution in mixed strategies. While computing his mixed NE strategy, each player pays attention only to the average cost function of his co-player, rather than optimizing his own average cost function. Hence, the nature of the optimization (i.e., minimization or maximization) becomes irrelevant in this case [10, p. 86]. The equilibrium costs of the attacker V_{att}^* and the IDS V_{ids}^* for this subgame are given by

$$V_{att}^* := [p_1^*\ (1-p_1^*)]M_{att}[q_1^*\ (1-q_1^*)]^T,$$

and

$$V_{ids}^* := [p_1^*\ (1-p_1^*)]M_{ids}[q_1^*\ (1-q_1^*)]^T,$$

where $[.]^T$ denotes the transpose of a vector.

In the context of intrusion detection, we interpret the NE (5) in the following way: The probability of attacker targeting subsystem one, y_1^*, at NE point decreases with decreasing α_f since the smaller the false alarm cost for the IDS, the more it is inclined to set an alarm and catch the attacker. One can argue that this can be achieved by setting the alarm very frequently as in the case with current IDSs. However, there are *hidden costs* with this approach like rendering the IDS practically useless under a flood of false alarms. Similarly, an increase in α_h and α_m play a deterrent role for the attacker. On the other hand, the probability of the IDS setting an alarm is affected by the gain of attacker from a successful intrusion, $-\beta_s$. If $\beta_s \gg \beta_h$, then the IDS is inclined to set the alarm more frequently. The penalty of getting detected for the attacker may vary significantly depending on the physical reachability of him. If, for example, the attacker is employed by the same organization he tries to intrude, then β_h is much larger than the marginal detection cost of a script-based attack from the other side of the globe.

The parametric analysis is repeated for information set 2 in Figure 3. In order to simplify the analysis we associate the same costs with subsystems two and three. This game can also be represented as a 2×2 bimatrix game given by

$$M_{att} := \begin{array}{|c|c|c|c|} \hline \text{t2} & \beta_h & -\beta_d & -\beta_s \\ \hline \text{t3} & -\beta_d & -\beta_h & -\beta_s \\ \hline \text{nt2} & 0 & 0 & 0 \\ \hline & \text{a2} & \text{a3} & \text{na2} \\ \hline \end{array}$$

$$M_{ids} := \begin{array}{|c|c|c|c|} \hline \text{t2} & -\alpha_h & \alpha_d & \alpha_m \\ \hline \text{t3} & \alpha_d & -\alpha_h & \alpha_m \\ \hline \text{nt2} & \alpha_f & \alpha_f & 0 \\ \hline & \text{a2} & \text{a3} & \text{na2} \\ \hline \end{array} \tag{6}$$

where α_d (β_d) is the cost (gain) of a deception for the IDS and the attacker, respectively. One can assume that $\alpha_d > \alpha_m$ and $\beta_d > \beta_s$ as alarming a wrong subsystem is more costly for the IDS than a missed attack, and by deceiving the IDS the attacker circumvents security mechanisms of the system more successfully. Let $\bar{p}_1$, $\bar{p}_2$, and $1-\bar{p}_1-\bar{p}_2$ be the probabilities for strategies (t2), (t3), and (nt2) of the attacker. Also let $\bar{q}_1$, $\bar{q}_2$, and $1-\bar{q}_1-\bar{q}_2$ be the respective probabilities for strategies (a1), (a2), and (na2) of the IDS. The security strategy of the IDS is determined by the relative values of α_d, α_f, and α_m as in the previous case. Furthermore, there is again no NE in pure strategies. The unique NE solution in mixed strategies is obtained by solving the counterpart of the set of inequalities (5), and is given by

$$\begin{aligned} \bar{p}_1^* = \bar{p}_2^* = \frac{\alpha_f}{2\alpha_f + 2\alpha_m + \alpha_h - \alpha_d}, \text{and} \\ \bar{q}_1^* = \bar{q}_2^* = \frac{\beta_s}{2\beta_s + \beta_h - \beta_d}, \end{aligned} \tag{7}$$

if $\beta_d < \beta_h$ and $\alpha_d < 2\alpha_m + \alpha_h$. Notice that, the attack and alarm probabilities for each subsystem is the same due to the same cost structure imposed on them. In fact, it is possible to adjust the cost parameters by taking into account various factors like relative importance of a subsystem for the organization, threat levels given the output of sensors, etc. The equilibrium probabilities of the attacker and the IDS strategies have a similar interpretation as the ones in the previous analysis. The increasing cost of deception for the IDS, however, has an encouraging effect on the attacker.

Given the equilibrium solutions and costs of each bimatrix game, the IDS and the attacker determine their overall strategies. The equilibrium strategy of the IDS, γ_{ids}^*, for example,

is given by

$$\gamma^*_{ids} = \begin{cases} a1 \text{ w.p. } q^*_1 \\ na1 \text{ w.p. } 1-q^*_1 \end{cases}, \quad \text{if in InfoSet1}$$
$$\begin{cases} a2 \text{ w.p. } \bar{q}^*_1 \\ a3 \text{ w.p. } \bar{q}^*_2 \\ na1 \text{ w.p. } 1-\bar{q}^*_1-\bar{q}^*_2 \end{cases}, \quad \text{if in InfoSet2}$$

On the other hand, the equilibrium strategy of the attacker depends on the equilibrium costs of the matrix games, V^*_{att} and $\bar{V}^*_{att}$.

The analytical investigation of the security game brings valuable insight to the attacker and the IDS behavior. In addition, the simplifying assumptions we have made in order to obtain analytical results can easily be extended to capture more realistic scenarios. Thus, the sample game of Figure 3 can be made arbitrarily large. Although increasing complexity prevents derivation of a closed form solution, one can easily solve such games numerically. Thus the developed framework can easily be applied to practical cases.

A Numerical Example: We solve a numerical example using the GAMBIT game theory analysis tool [13] to demonstrate the results described above. For illustrative purposes, the action spaces of the players are again limited, and the cost parameters are chosen as

$Parameter$	α_h	α_d	α_m	α_f	β_h	β_s	β_d
$Value$	7	10	8	9	6	4	5

In this game, there is no NE solution in pure strategies. In order to investigate NE in mixed and behavioral strategies, we associate the probability vector $\mathbf{p}$ with actions [(t1) (t2) (t3) (nt1) (nt2)] of the attacker and the vector $\mathbf{q}$ with actions [(a1) (a2) (a3) (na1) (na2)] of the IDS. We obtain two Nash equilibrium points in behavioral strategies, which are in accordance with equations (5) and (7):

$$(\mathbf{p}^*_1, \mathbf{q}^*_1) = ([0\ 0.29\ 0.29\ 0\ 0.42], [0\ 0.44\ 0.44\ 0\ 0.11])$$
$$(\mathbf{p}^*_2, \mathbf{q}^*_2) = ([0.38\ 0\ 0\ 0.63\ 0], [0.40\ 0\ 0\ 0.60\ 0])$$

In both cases, the equilibrium cost for the attacker is zero, $V^*_{att} = 0$. Hence, choosing either of the equilibrium strategies is equivalent for the attacker.

Finally, we investigate the Nash equilibria of the security game in mixed strategies by converting it to the normal form. Solving the resulting matrix game with GAMBIT, we obtain 18 different NE points in mixed strategies. Two of these coincide with the NE solutions in behavioral strategies as expected [10, p. 103]. Thus, the NE points in behavioral strategies characterize the behavior of the players more clearly, and provide more insight to the underlying system dynamics.

IV. Conclusion

We have investigated the basic decision and analysis processes involved in information security and intrusion detection, as well as possible usage of game theory for developing a formal decision and control framework. A generic model has been developed for a distributed IDS with a network of sensors. Furthermore, two flexible, platform independent schemes based on game theoretic techniques have been proposed.

The security warning system provides system administrators an intuitive overview of the security situation in the network, and enables the IDS to operate in different modes specific to each warning level. In this simple and easy-to-implement scheme cooperative game theory, specifically Shapley values, are used for analysis and configuration. The second scheme, on the other hand, models the interaction between the attacker and the IDS as a two-person finite game with dynamic information. Nash equilibrium solutions are derived analytically and analyzed for the defined security game in two special cases.

In both schemes, using game theoretic concepts we have addressed some of the basic network security tradeoffs, and have given illustrative numerical examples. Thus, we have demonstrated the suitability of game theory for development of various decision, analysis, and control algorithms in intrusion detection. Future work on the subject includes extensions to the proposed model, development of practical algorithms, and decentralization of the decision and processes.

References

[1] E. Rabinovitch, "The neverending saga of internet security: Why? how? and what to do next?" *IEEE Communications Magazine*, pp. 56–58, May 2001.

[2] R. Ellison, D. Fisher, R. Linger, H. Lipson, T. Longstaff, and N. Mead, "Survivable network systems: An emerging discipline (cmu/sei-97-tr-013)," Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep., November 1997, citeseer.nj.nec.com/ellison97survivable.html.

[3] R. Ellison, R. Linger, H. Lipson, N. Mead, and A. Moore, "Foundations for survivable systems engineering," *The Journal of Defense Software Engineering*, pp. 10–15, July 2002.

[4] R. Bace and P. Mell, "Intrusion detection systems," NIST Special Publication on Intrusion Detection Systems, http://www.snort.org/docs/nist-ids.pdf.

[5] M. Y. Huang, R. J. Jasper, and T. M. Wicks, "A large scale distributed intrusion detection framework based on attack strategy analysis," in *Intl. Symp. on Recent Advances in Intrusion Detection (RAID)*, Louvain la Neuve, Belgium, 1998. [Online]. Available: citeseer.nj.nec.com/huang98largescale.html

[6] G. Helmer, J. Wong, V. Honavar, and L. Miller, "Intelligent agents for intrusion detection," in *In Proc. of IEEE Information Technology Conference*, Syracuse, NY, September 1998, pp. 121–124, citeseer.nj.nec.com/helmer98intelligent.html.

[7] J. S. Balasubramaniyan, J. O. Garcia-Fernandez, E. Spafford, and D. Zamboni, "An architecture for intrusion detection using autonomous agents," in *In Proc. of 14th Annual Computer Security Applications Conference (ACSAC)*, Scottsdale, AZ, December 1998, pp. 13–24.

[8] S. N. Hamilton, W. L. Miller, A. Ott, and O. S. Saydjari, "The role of game theory in information warfare," in *4th Information Survivability Workshop (ISW-2001/2002)*, Vancouver, BC, Canada, March 2002.

[9] ——, "Challenges in applying game theory to the domain of information warfare," in *4th Information Survivability Workshop (ISW-2001/2002)*, Vancouver, BC, Canada, March 2002.

[10] T. Başar and G. J. Olsder, *Dynamic Noncooperative Game Theory*, 2nd ed. Philadelphia, PA: SIAM, 1999.

[11] G. Owen, *Game Theory*, 3rd ed. New York, NY: Academic Press, 2001.

[12] J. Contreras, M. Klusch, O. Shehory, and F. Wu, "Coalition formation in a power transmission planning environment," in *In Proc. of 2nd Intl. Conference on Practical Applications of Multi-Agent Systems, PAAM*, London, U.K., April 1997, pp. 21–23. [Online]. Available: citeseer.nj.nec.com/12028.html

[13] T. G. Project, "Gambit game theory analysis software and tools," http://www.hss.caltech.edu/gambit, 2002.