

# Configuration Strategies For Collaborative IDS Using Game Theory

Mohsen Ghorbani, Hamid Reza Ghorbani, Mahmoud Reza Hashemi

School of Electrical and Computer Engineering

College of Engineering, University of Tehran

Tehran, Iran

[ghorbani.mohsen@ut.ac.ir](mailto:ghorbani.mohsen@ut.ac.ir), [ghorbani.it@ut.ac.ir](mailto:ghorbani.it@ut.ac.ir), [rhashemi@ut.ac.ir](mailto:rhashemi@ut.ac.ir)

**Abstract**— In recent years, distributed and zero-day attacks have emerged as one of the most serious security threats. The incomplete knowledge and information of a stand-alone intrusion detection system (IDS) is one of the main reasons for the success of these attacks. Collaborative IDS (CIDS) is one solution to address this problem. IDSs in this framework share their knowledge and consult with each other. Having access to a larger number of detection libraries for IDS configuration, along with the possibility of more cooperation with other participants in this collaborative system can lead to improved overall performance. However, a larger number of libraries and more collaborative activities increase resource consumption and communication overhead, which may in turn reduce system performance. There are a large number of papers in the literature that have utilized game theory to describe the optimal configuration of standalone or networked IDSs. In this paper, those works have been extended and the interactions between the attackers and IDSs in a CIDS framework have been modeled with a non-zero sum stochastic game. In this regard, the solution concept of stationary Nash equilibrium has been applied to this game to describe the optimal configuration of each IDS in a CIDS and the expected behavior of attackers.

**Keywords**- Collaborative IDS; Network Security; Stochastic Games; Stationary Nash equilibrium.

## I. INTRODUCTION

With the growing complexity of security threats, various counter measures have been proposed for intrusion detection and prevention. As a complementary defense mechanism in addition to the other protection tools, Intrusion Detection Systems (IDS) are employed to monitor network events and user activities. Emerging unknown, distributed and fast spread attacks have highlighted the deficiencies of standalone IDSs which generally suffer from local knowledge and limited information about the whole environment [1]. To overcome these problems, Collaborative Intrusion Detection Systems (CIDSs) have been proposed in the literature by researchers and security specialists [2].

A CIDS consists of a set of IDSs which are deployed strategically in different locations throughout a network, and communicate with each other using standard communication protocols such as IDMEF [3]. These IDSs can be heterogeneous (e.g. provided by different vendors) and employ different detection technologies [4]. They are trying to make better decisions and reach better performance by collaborating with other nodes in the CIDS. The strength of a CIDS depends on the strength of each standalone IDS, in terms of accuracy, detection rate, etc. [5]. Collaboration will help standalone IDSs

to improve their detection rate since they benefit from the collective knowledge and experience shared by other nodes. This access to network-wide information will increase the probability of detection of zero-day and distributed attacks [6].

If IDSs utilize a larger number of detection libraries and more collaboration with their neighboring IDSs, it has been proven that the performance will be enhanced [7]. However, the larger number of libraries will decrease system performance in terms of IDS throughput, and also more collaboration will increase communication overhead [8, 9]. Hence there is a tradeoff between system performance and security enforcement level [3]. In other words, standalone IDSs in a CIDS require a proper strategy for library configuration [5].

Policy based configuration is an approach to IDS configuration [10, 11]. In order to improve system performance, researchers have utilized game theory to tune configuration policies. Non-cooperative interactions between attackers and defenders, the existence of several trades-offs that exist in IDS problems, and the rationality of attacker and defender are some of the characteristics of this network security problem which makes game theory a promising approach [12, 13].

Considering the urge for CIDS self-configuration [14], we have utilized a non-zero sum discounting stochastic game to model the problem of collaborative IDS configuration. In particular, we have exploited game theory to describe optimal configuration strategies for collaborative IDSs in a CIDS. Specifically, the solution concept of Stationary Nash Equilibrium has been applied to the game in a descriptive way which describes the players' optimal stationary strategies. Using this game approach we will be able to design a game theory-based methodology in which configuration policies are dynamically updated according to network conditions. This paper can be considered as an extension of the work in [11] where the networked IDS configuration problem has been taken into account. Since there is no collaboration between standalone IDSs in networked IDSs, our proposed game model is completely distinct from [11] in defense scenario, defenders' utility functions, and state transition probability.

The paper proceeds as follows. In Section II, we study the CIDS frameworks and their characteristics. Section III will overview some related works on collaborative IDS configuration and application of game theory in network security. In section IV, we formulate a nonzero-sum stochastic game to model the interaction between  $M$  attackers and  $N$  collaborative IDSs. The existence of a solution for this game is

presented in section V. Finally, the paper concludes in section VI.

## II. CIDS FRAMEWORKS

A CIDS consists of a set of intrusion detection systems which are located in different locations on a network and communicate with each other using a standard protocol [15]. In the literature, several frameworks have been proposed for a CIDS that consist of common modules such as: detection unit, log analyzer (correlation unit), acquaintance management, trust measurement, feedback aggregator and resource management [16].

A detection unit utilizes different technologies in order to inspect and analyze the traffic for detection of suspicious behavior. It then produces the results of this analysis in the form of low-level intrusion alerts. Processing and analyzing these alert logs is performed by the IDS correlation unit. The results of its analysis will be reported to other IDSs as a high-level intrusion report [2].

Acquaintance management is another important module in a CIDS. The responsibility of this module is to choose several IDSs for knowledge sharing and updating the list of acquaintances. There are some challenges for acquaintance management such as how to choose the trusted and expert acquaintances and also how to determine the optimum number of them. The larger the numbers of acquaintances are, the more sophisticated decisions can be made. But, this may result in more consumption of resources and more network overhead which adversely affects the system performance [17].

In order to find a more trustworthy and expert acquaintance, the trust management module is proposed. There are many methods in the literature for evaluating the trustworthiness of peers, such as test based methods, and reputation based methods [18].

Another module in a CIDS is feedback aggregator. When the IDSs in a CIDS have not enough confidence to make proper decisions about the received traffic individually, they can send consultation requests to their acquaintances to inquire their opinions. These opinions are then fed to the feedback aggregator. The output of this function is the final decision which will be applied on the traffic [19].

Processing the received traffic and collaborating with other peers in a CIDS will consume the resources of IDSs. Also due to the reciprocal altruism and incentive based design of typical CIDSs, all participants are motivated to allocate some of their resources to their acquaintances in order to answer their consultation requests. The management of these resources is the main responsibility of the resource management module [20].

In order to establish the appropriate collaboration between nodes, several architectures have been proposed, namely: Centralized, Hierarchical, and Fully-Distributed (Decentralized) [2]. Each of these aforementioned architectures has their pros and cons. The primal assumption in this paper is that the IDSs are connected to each other based on a peer-to-peer architecture (Fully-Distributed). In the Fully-Distributed approach each node possesses both the detection unit and the correlation unit, simultaneously. Hence, in this architecture, each node can communicate with its peers (e.g. in a peer-to-peer overlay). Some of the main challenges in this architecture include specifying the communication protocol between peers, and identifying appropriate mechanisms for dissemination of information [2].

## III. RELATED WORKS

Clearly the principal purpose of an IDS is to provide acceptable detection rate. In order to achieve this objective, CIDSs have been considered as an interesting topic for researchers. Appropriate configuration of IDSs in a CIDS, such that it leads to a satisfactory accuracy in terms of intrusion detection, is one of the major challenges in this domain. The term *configuration* stands for selecting a set of detection libraries which forms the core of the decision-making engine.

In [21], an IDS reconfigures its detection libraries according to an updating policy and based on information received from other distributed IDSs. Through this reconfiguration process, an IDS can acquire the information of the whole environment and would be capable of detecting new and unknown attacks. Another approach has been suggested in [22] to reconfigure an IDS in a cooperative platform. When the IDS does not have sufficient confidence to make a local decision about an event, it will consult with other IDSs and request their opinion. The IDS can then reconfigure its detection libraries using a majority mechanism (e.g. voting), after collecting its acquaintances' answers. In some other works, the standalone IDSs in a CIDS do not reconfigure themselves; instead when an IDS cannot take a decision locally, it prefers to take its acquaintances feedbacks and perform their aggregate opinion. Doing so, the IDS will increase its detection rate. None of these works have utilized a formal approach for IDS configuration in their collaborative platforms.

In the literature, there are a numerous works which utilize game theory to study IDS problems. Specifically, some of them have taken into account the problem of IDS configuration and also Networked IDS configuration. For instance, in [10] a non-cooperative game approach has been exploited to address the problem of IDS configuration. In this paper, Zhu and Basar have used a two player zero-sum stochastic game to describe the expected behavior of the attacker and the defender. The works in [11] and [23] have dealt with the problem of networked IDSs configuration with objective of detecting simultaneous attacks from different attackers, with minimum resource utilization. The problem has been formulated as a multi-player nonzero-sum stochastic game to configure the networked IDSs dynamically. The objective of these papers was to describe the optimal stationary policies (strategies) for IDS configuration using Nash equilibrium solution concept. Nonetheless, in these works the collaboration between IDSs has not been taken into account.

To summarize, some of the papers have considered the problem of IDS configuration in a collaborative platform, but to they have not used game theory to address the problem. The other works, though, they utilize game theory to address the problem of IDS and networked IDS configuration, but they did not consider the collaboration between IDSs.

To the best our knowledge, there is no work in the literature that has modeled the IDS configuration problem in a CIDS using game theory. In this paper, the configuration strategies of IDSs in a CIDS have been described through the Stationary Nash equilibrium solution concept.

## IV. NON-ZERO SUM STOCHASTIC GAME MODEL

Consider a set of defending machines (IDSs)  $\mathcal{N} = \{n_1, n_2, \dots, n_N\}$  that constitute a peer-to-peer CIDS. These machines are connected to each other by a bidirectional graph  $\mathcal{V} = (\mathcal{N}, \mathcal{E})$ . Each machine is only responsible for protecting its corresponding sub-network. The network is targeted by multiple

malicious attackers that are represented by the set  $\mathcal{M} = \{m_1, m_2, \dots, m_M\}$ .

Note that in the following subsection we utilize the game model in [11] in order to model the available actions of defenders and their configuration costs.

#### A. Defending Machines

Each machine  $n_i$  can only collaborate with its adjacent nodes (peers) in the graph  $\mathcal{V}$  that we refer to as *acquaintances* throughout this paper. Let  $\mathcal{N}_i^{\mathcal{M}}$  represents the set of acquaintances of a machine  $n_i$ , i.e.,  $\mathcal{N}_i^{\mathcal{M}} = \{n_j \in \mathcal{V} | n_i n_j \in \mathcal{E}\}$ . Simplifying our model, in this paper we assumed that the number of acquaintances of a machine does not change with time. Although in real world, each machine must have the ability to change its acquaintances based on their trustworthiness, expertise, availability, etc. [24].

Miss-used based IDSs can detect the attacks by comparing the incoming traffic with their selected rules. Typically these rules are categorized into the number of detection libraries based on the class of attacks they can detect. For example Snort has 51 detection libraries with more than 9000 rules [10]. Consider  $\mathcal{L}_i = \{l_{i_1}, l_{i_2}, \dots, l_{i_{L_i}}\}$  to be the set of these detection libraries with each of them consisting of a number of rules. For example let the set  $l_k = \{r_1, r_2, \dots, r_k\}$  be a particular detection library with  $k$  rules. IDSs compare each received packet with these rules to find suspicious behavior. These rules consist of a number of parameters which are assigned proper values by an administrator. These values will be compared to the equivalent parameters' value for each packets. The received packets are compared to each rules and according to the percentage of matching, the IDS should take appropriate actions. These actions are based on predefined policies.

Note that due to the inherent heterogeneity of IDSs in a CIDS, the set of detection libraries can be different each machine. Each machine can select some of these available libraries from their rule-base and load them into its detection unit. Let  $\mathcal{L}_i^* = \sigma(\mathcal{L}_i)$  denote the set of all possible subsets of  $\mathcal{L}_i$ . Configuring its detection unit, each machine  $d_i$  chooses a set of libraries  $F_i \in \mathcal{L}_i^*$  and uses them to inspect the incoming traffic. In fact  $\mathcal{L}_i^*$  includes the available actions (configuration) of machine  $n_i$ . Also, let  $\mathcal{L}^* = \prod_{i=1}^N \mathcal{L}_i^*$  denote the joint action set and  $F = [F_i]_{i=1,2,\dots,N} \in \mathcal{L}^*$  denote the joint defender actions.

IDSs should try to decide and act automatically and independently as far as possible. Hence, a well-designed IDS should behave differently in various situations. For instance, when the network is in a normal condition and no attack has occurred, there is no need that the IDS behave too inspective. On the contrary, when the sub-network has been targeted by attackers several times, the IDS should adopt a more strict monitoring policy. Hence it can use more high-level detection libraries in its configuration. Also, in the case of CIDS, each standalone IDS should collaborate with its acquaintances according to the situation. In this regard, we adopt different states for defending machines so that they can condition their actions based on these states. It is considered that each machine can be in one of the finite number of states  $s_i \in S_N^i, n_i \in \mathcal{N}$ . The state of a machine represents the situation in which they should make a decision. In our model these states include: *High-Confidence (HC)*, *Medium-Confidence (MC)*, *Low-Confidence (LC)*. Let  $S_N^i = \{HC, MC, LC\}$  be the set of these states and  $S_N = \prod_{i=1}^N S_N^i$  denote the CIDS states. It is expected that if a defending machine is more inspective, it consumes more

resources. Furthermore, it consults more with its acquaintances when its state changes from HC to MC to LC.

The more protective an IDS is, the more resources it should consume to better configure, monitor and collaborate. These mentioned limited resources include CPU, RAM, network bandwidth, etc. These resources are used for loading detection libraries, inspecting and processing of incoming traffic, collaborating with other peers in a CIDS. Using each resource will impose a corresponding cost to the defender. In our model, resource costs include *configuration cost* and *collaboration cost*. Configuration cost  $C_{F_i}$  is equal to the sum of all independent library costs in the configuration (loaded into the detection unit). Consider  $C_{f_i}^*: \mathcal{L}_i^* \rightarrow \mathbb{R}_+$  be the mapping function which measures the cost of each configuration. Hence, the cost of a configuration is obtained from:

$$C_{F_i} = C_{f_i}^*(F_i) = \sum_{x \in F_i} C_{f_i}(x) \quad (1)$$

where  $C_{f_i}: \mathcal{L}_i \rightarrow \mathbb{R}_+$  is a mapping that measures each library cost.

The collaboration cost includes two distinct costs. First is the cost of resources that a machine should allocate to its acquaintances owing to the reciprocal altruism and incentive-based design of CIDS. Second is the cost of collaboration when a particular machine needs to consult with its acquaintances. In our model the first cost is considered to be static due to the static number of acquaintances and will be ignored in our calculation. The second cost  $C_{B_i}$  is the sum of all resources that a defender machine  $n_i$  would consume when it is obliged to collaborate with them. Let  $B > 0$  denote the static cost which the collaboration to each peer will impose to the machine  $n_i$  (The cost includes sending consultation messages, allocating bandwidth, etc.); hence the collaboration cost is obtained from  $C_{B_i} = |\mathcal{N}_i^{\mathcal{M}}| * B$ , where  $|\mathcal{N}_i^{\mathcal{M}}|$  denote the number of acquaintances.

As mention before, defending machines collaborate with their acquaintances based on their trustworthiness. In the literature there are numerous works on measuring the level of trust among IDSs in a CIDS. For instance, Fung et al. in [20] propose a Bayesian trust management model for standalone IDSs in a CIDS. In this work, the trust model evaluates the trustworthiness of each IDS based on their expertise and honesty. Using this model, the trust management component of IDSs assigns a trust value to each acquaintance. Let  $T_{ij} \in [0, 1]$  denote the trust value which defender  $n_i$  has assigned to its peer  $n_j$ . This value can be updated through time. In [20] these value can be updated by evaluating the answer of the peer to some test messages or real requests. After receiving their answers the quantitative amount of expertise and honesty are evaluated by using a satisfaction function.

#### B. Attackers

In this subsection, we review the model of attackers which is presented in [11].

Malicious attackers, depending on their intention and their motivation, utilize various attacks to compromise their targets. For example, an attacker who intends to launch a highly destructive attack would utilize more complex and creative attacks, e.g. distributed attacks, instead of simple ones. Accordingly, the attackers can be in one of the finite number of states which show the level of their aggressiveness. Let  $s_j \in$

$S_A^j, m_j \in \mathcal{M}$  denote the state of attacker  $m_j$  which can be either Aggressive (A) state or Not-Aggressive state (NA). Consequently,  $S_A = \prod_{j=1}^M S_A^j$  will be the aggregate state of attackers. For each attacker the set  $\mathcal{A}_j = \{a_j^1, a_j^2, \dots, a_j^{A_j}\}$  shows all available attacks which can be exploited to compromise the targets. Each attacker can choose a number of attacks to its targets at time instance ( $k$ ) which is denoted by  $a_j^{(k)} = [a_{ji}^{(k)}]_{i=1, \dots, N} \in \mathcal{A}_j^N$ . Consequently, the attack profile is the matrix  $a^{(k)} = [a_{ji}^{(k)}]_{j=1, \dots, M, i=1, \dots, N}$  which represents all attacks launched by all attackers to all machines at time ( $k$ ).

If the attacker  $m_j$  launches an attack  $a_{ji}^{(k)}$  to the defender  $n_i$  at time ( $k$ ), the damage caused by the attacker would be  $\tilde{d}_{ji}^{(k)}$  if not detected. The damage can be measured by the mapping  $\tilde{\mathcal{D}}_j = \mathcal{A}_j \rightarrow \mathbb{R}_+$ , i.e.,  $\tilde{d}_{ji}^{(k)} = \tilde{\mathcal{D}}_j(a_{ji}^{(k)})$ ,  $\forall a_{ji}^{(k)} \in \mathcal{A}_j$ . Also there exists the mapping  $\tilde{\mathcal{C}}_j: \mathcal{A}_j \rightarrow \mathbb{R}_+$  that measures the cost of each attack, i.e.,  $\tilde{c}_{ji} = \tilde{\mathcal{C}}_j(a_{ji})$ . It is sensible that the evaluation of attacker about the damage incurred to the defender be deferent from defender's evaluation about perceived damage. It is considered that the mapping  $\mathcal{D}_i = \mathcal{A}_j \rightarrow \mathbb{R}_+$  measures the damage perceived by the defender, i.e.,  $d_{ji}^{(k)} = \mathcal{D}_i(a_{ji}^{(k)})$ .

### C. The Attack-Defence Scenario

At the beginning of each stage of the game, the attacker chooses a vector of attacks  $a_j \in \mathcal{A}_j^N$  from its available attack types  $\mathcal{A}_j$  in order to attack its target set ( $\bar{\mathcal{N}}_j$ ). The defender, on the other side, selects a configuration of detection libraries  $F_i$  from its available configuration  $\mathcal{L}_i^*$ .

In order to detect the attacks, the defending machine  $d_i$  compares the received packets of the attack  $a_{ji}$  with each of its available rules in its selected configuration  $F_i$ . Normally, the IDSs generate an alert as a result of processing the packets and rank these alert based on some metrics. The ranking measures the level of matching of the parameters' values of the rules with the equivalent parameters' values of the received packet. For example, Snort ranks the alerts in three levels (low, medium and high) and Bro has up to 100 levels. Consider the function  $\mathcal{R}_i: \mathcal{L}_i^* * \mathcal{A}_j \rightarrow \mathbb{R}_{0 \leq \mathbb{R} \leq 1}$  maps the IDSs alert ranking onto the [0, 1] interval. Let  $R_i = \mathcal{R}_i(F_i, a_{ji})$  denote the alert ranking which is provided by the machine  $n_i$ , where  $R_i = 0$  signify a benign traffic and  $R_i = 1$  signify a highly dangerous one. A bigger value of  $R_i$  denote the more dangerous traffics.

**Definition 4.1:** If all parameter values of one of these rules match with the equivalent parameter values of a received packet of an attack, we refer to it as *matching* ( $R_i = 1$ ). If all rules were compared with the packets and none of these parameter values match, we refer to it as *non-matching* ( $R_i = 0$ ). Finally, if just some of the rules match with the packet, we refer to it as *partial-matching* ( $R_i \in (0, 1)$ ).

In our modeling of defense scenario, we assume that when an IDS receives a packet and compares it with its detection libraries it may choose one of the following actions:

1) *If matching has occurred* ( $R_i = 1$ ), IDS will block the traffic. In this situation the IDS decides that its decision is correct by a high level confidence and there is no need to check the other rules.

2) *If partial-matching has occurred* ( $R_i \in (0, 1)$ ), the IDS will log some alerts depending on the percentage of matching.

The IDS may take one of the following actions based on the level of alert. The level of alert is determined by the IDS in the generated logs: 2-1) if the level of alert is less than a predefined collaboration threshold ( $th_i^c$ ), the packet will be permitted to pass. 2-2) else, the traffic will be blocked and the IDS will collaborate with its acquaintances in order to make a final decision.

3) *If no-matching has occurred* ( $R_i = 0$ ), the packet is permitted to pass.

In order to collaborate with its acquaintances, each IDS sends the traffic as a *consultation message* to its acquaintances. Each of these peers analyzes the traffic and sends back the alert ranking as an *answer message*. The IDS receives their feedbacks and gives their answers as the input of its feedback aggregator function. In the literature, various methods have been proposed to aggregate the feedbacks. For example, Fung et al. in [20] have proposed a weighted majority method to aggregate their feedbacks. In this method the weights are based on the trust values of peers ( $T_{ji}$ ). Specifically, in this work only the feedbacks of those peers whose trust values are greater than a threshold is accepted. The output of the feedback aggregator function is denoted by  $R_i'$ , and is a value in the [0, 1] interval which is the rank of alert based on peers' opinions.

When the aggregate feedback value is obtained, the IDS will take different decisions with regard to  $|R_i - R_i'|$  which is the distance of IDS alert ranking from aggregation of its peers' alert ranking. If the distance of two alert rankings is greater than a predefined threshold ( $th_i^u$ ), the IDS will understand that its detection unit has not enough knowledge to make the decision locally and more collaboration is needed from this point on. In order to collaborate more with others, it should decrease the collaboration threshold proportional to  $|R_i - R_i'|$ . On the other hand, if the distance is lower than a predefined threshold ( $th_i^l$ ), it can decrease the collaboration threshold to avoid the excessive collaboration costs. In our modeling, these actions will be performed according to the defenders' state. Doing so, we have allocated a static state depended value for each of the aforementioned thresholds ( $th_i^c, th_i^u, th_i^l$ ). Let  ${}^{LC}th_i^c$  denote the collaboration threshold in state *LC*, and  ${}^{MC}th_i^c$  denote the collaboration threshold in state *MC*. Similar notations have been used for other thresholds for the other states. The allocation of values to each threshold should adhere to the following rules: 1)  ${}^{LC}th_i^c < {}^{MC}th_i^c < {}^{HC}th_i^c$ , 2)  ${}^{LC}th_i^u > {}^{MC}th_i^u > {}^{HC}th_i^u$ , 3)  ${}^{LC}th_i^l < {}^{MC}th_i^l < {}^{HC}th_i^l$ .

For example, consider that a partial matching has occurred and the alert ranking is greater than  $th_i^c$ . Therefore the defender has decided to collaborate with its peers. After feedback aggregation, if  $|R_i - R_i'| > th_i^u$  the defender state will transit to *LC*. Due to the lower value of predefined collaboration threshold ( $th_i^c$ ) in state *LC* proportional to other states, the defender can have more collaboration with its peers. It is noteworthy that all predefined thresholds ( $th_i^c, th_i^u, th_i^l$ ) considered for a defender  $n_i$ , only depend on its state  $s_i$  regardless of time.

### D. State Transition Probability

As stated before, the attack-defense scenario is modeled as a stochastic game. Therefore, let  $\mathcal{S} = S_N * S_A$  denote the system state, and  $s^{(k)} = [s_N^{(k)} * s_A^{(k)}] \in \mathcal{S}$  be a particular system state at each time instance ( $k$ ). The next state of the defender  $n_i$  will be obtained based on the following rules:

- 1) If  $R_i = 1$ , the next state will be *HC*.
- 2) If  $R_i \leq th_i^c$ , the next state will be *HC*.
- 3) If  $R_i \in (th_i^c, 1)$  and  $|R_i - R_i'| < th_i^l$ , the next state will be *HC*.
- 4) If  $R_i \in (th_i^c, 1)$  and  $|R_i - R_i'| > th_i^u$ , the next state will be *LC*.
- 5) If  $R_i \in (th_i^c, 1)$  and  $th_i^l < |R_i - R_i'| < th_i^u$ , the next state will be *MC*.

It is worth to note that for a particular defender, the probability of matching, non-matching, and partial-matching depend on the selected configuration  $F_i$ , attack profile  $a$  and system state  $s$ . Furthermore,  $R_i$  and  $R_i'$  depend on the defenders joint action set  $F$ , attack profile  $a$  and system state  $s$ .

The system state moves from  $s^{(k)}$  to  $s^{(k+1)}$  with a probability measure  $\mathbb{P}$ , where  $s^{(k)}, s^{(k+1)} \in \mathcal{S}$  are two system states. Owing to the independency of state transition probability of attackers and defenders,  $\mathbb{P}$  can be determined by:

$$\mathbb{P}(s^{(k+1)} | a^{(k)}, F^{(k)}, s^{(k)}) = \prod_{n_i \in \mathcal{N}} \mathbb{P}(s_{n_i}^{(k+1)} | a^{(k)}, F^{(k)}, s_{n_i}^{(k)}) \prod_{m_j \in \mathcal{M}} \mathbb{P}(s_{m_j}^{(k+1)} | a^{(k)}, F^{(k)}, s_{m_j}^{(k)}) \quad (2)$$

### E. Utilities

The utility of players is determined by the system state and the actions chosen by players. Hence, the utility function of the defending machine  $n_i$  at time  $k$  is defined by  $U_i^{(k)}: \prod_{j=1}^M \mathcal{A}_j^N * \mathcal{L}_i^* * \mathcal{S} \rightarrow \mathbb{R}_+$ ,  $n_i \in \mathcal{N}$ . In this scenario the defender should care about its security loss, the cost of its configuration and the cost of its collaboration. So the defender prefers to choose a configuration which causes less damage and less cost to itself and more cost to its attackers. Hence, the defender  $n_i$  will try to maximize its utility function  $U_i^{(k)}$  obtained as follows.

$$\begin{aligned} U_i^{(k)}(a^{(k)}, F_i^{(k)}, s^{(k)}) &= -d_i^{(k)}(a^{(k)} | F_i^{(k)}, s^{(k)}) - C_{fi}^*(F_i^k) \\ &\quad - C_{bi}^*(F_i^k, a^{(k)}) \end{aligned} \quad (3)$$

where  $d_i^{(k)}(a^{(k)} | F_i^{(k)}, s^{(k)})$  denotes the damage to machine  $n_i$ , that is caused by the attack profile  $a$  if the machine  $n_i$  chooses a configuration  $F_i$  at time instance  $k$ . The security loss  $d_i: \prod_{j=1}^M \mathcal{A}_j * \mathcal{L}_i^* \rightarrow \mathbb{R}_+$  can be expressed as:

$$d_i(a | F_i, s) = \sum_{m_j \in \mathcal{M}} d_{ji} I_{ji}, \quad (4)$$

$$I_{ji} = I(F_i, a_{ji}, s) = \begin{cases} 1 & R_i \leq th_i^c \\ 0 & \text{else} \end{cases} \quad (5)$$

To understand the above formula, consider an attack is received by the machine  $n_i$ ; if the IDS permits the traffic to pass, then the attack will be successful and the damage will be completely perceived by the defender; if the IDS blocks the traffic, then the attack will not be successful. For example, if *non-matching* has occurred ( $R_i = 0$ ) the IDS will permit the traffic to pass; and if *matching* has occurred ( $R_i = 1$ ) the IDS will block the traffic.

The configuration cost is obtained from (1), and collaboration cost is obtained from the following.

$$C_{bi}^*(F_i^k, a^{(k)}) = \begin{cases} |\mathcal{N}_i^{\mathcal{M}}| * B & R_i \in (th_i^c, 1) \\ 0 & \text{else} \end{cases} \quad (6)$$

Similarly, an attacker prefers to choose the attacks which cause more damage to defenders and less cost to itself. Hence, a rational attacker attempts to maximize its utility  $\tilde{U}_j^{(k)}$  which it obtained from:

$$\tilde{U}_j^{(k)}(a_j^{(k)}, F^{(k)}, s^{(k)}) = \tilde{d}_j(a_j^{(k)} | F^{(k)}, s^{(k)}) - \sum_{i=1}^N \tilde{c}_j(a_{ji}^{(k)}) \quad (7)$$

where  $\tilde{d}_j$  is the damage impose by the attacker  $m_j$  to its target set  $\tilde{\mathcal{N}}_j$  which can be obtained from:

$$\tilde{d}_j(a_j | F, s) = \sum_{n_i \in \tilde{\mathcal{N}}_j} \tilde{d}_{ji} I_{ji} \quad (8)$$

### F. Strategies and existence of Equilibrium

As stated before, IDSs choose their configuration from their available action set only based on the level of their confidence to their decisions. Also, the attackers' choice of attack is considered to be dependent on their aggressiveness level. Hence, the strategies of the players are stationary. Stationary strategies do not consider the entire history and only depend on the states (are independent of time).

Let  $\mathcal{h}_{(k)} = (s^{(0)}, F^{(0)}, a^{(0)}, s^{(1)}, F^{(1)}, a^{(1)}, \dots, a^{(k-1)}, s^{(k)})$  denote a history of  $t$  stages of the game, and  $\mathcal{H}_{(k)}$  denote the set of all possible histories up to this stage. Consider  $\Sigma_{B,i}(\mathcal{h}_{(k)}, F_i)$  denote the behavioral strategy of the defender  $i$  which is the probability of playing action  $F_i \in \mathcal{L}_i^*(s_i^k)$  for history  $\mathcal{h}_{(k)}$ .

**Definition 3.1 (Markov strategy, [25]):** A Markov strategy  $\Sigma_{M,i}$  is a Behavioral strategy in which  $\Sigma_{M,i}(\mathcal{h}_{(k)}, F_i) = \Sigma_{M,i}(\mathcal{h}'_{(k)}, F_i)$  if  $s_i^k = s'_i{}^k$ , where  $s_i^k$  and  $s'_i{}^k$  are the final states of  $\mathcal{h}_{(k)}$  and  $\mathcal{h}'_{(k)}$ , respectively.

**Definition 3.2 (Stationary strategy, [25]):** A Stationary strategy  $\Sigma_{X,i}$  is a Markov strategy in which  $\Sigma_{X,i}(\mathcal{h}_{(k_1)}, F_i) = \Sigma_{X,i}(\mathcal{h}'_{(k_2)}, F_i)$  if  $s_i^{k_1} = s'_i{}^{k_2}$ , where  $s_i^{k_1}$  and  $s'_i{}^{k_2}$  are the final states of  $\mathcal{h}_{(k_1)}$  and  $\mathcal{h}'_{(k_2)}$ , respectively.

The Markov strategy ( $\Sigma_{M,j}$ ) and Stationary strategy ( $\Sigma_{X,j}$ ) of the attacker can be defined similarly.

The stationary strategies of a defender will be in the form of probability distribution over its all possible configuration ( $F_i$ ) from its action set  $\mathcal{L}_i^*$  for each state. Likewise, the stationary strategy of an attacker will be in the form of probability distribution over all its possible vector of attacks  $a_j$  from its action set  $\mathcal{A}_j^N$  for each state. For example, for the defender  $n_i$ , assume  $\mathcal{L}_i = \{l_1, l_2\}$ , so its possible configuration will be  $\mathcal{L}_i^* = \{\emptyset, \{l_1\}, \{l_2\}, \{l_1, l_2\}\}$ . In regards of the above lines, an instance of its stationary strategy can be [0, 0.2, 0.4, 0.4] for state *LC*, [0, 0.3, 0.3, 0.4] for state *MC*, and [0, 0.1, 0.3, 0.6] for state *HC*. The characterization of stationary strategies of players can be found in [11].

Considering the above definitions, let  $\Sigma_i$  and  $\Sigma_j$  denote the strategy sets of a defender and an attacker, respectively. The multi-person stochastic game is well defined by the set  $\langle \mathcal{N}, \mathcal{M}, \mathcal{S}, (\tilde{U}_i)_{n_i \in \mathcal{N}}, (\tilde{U}_j)_{m_j \in \mathcal{M}}, \mathbb{P}, (\Sigma_i)_{n_i \in \mathcal{N}}, (\Sigma_j)_{m_j \in \mathcal{M}} \rangle$ .

As mentioned before, the utility of players in the stochastic game between the IDSs and attackers is discounted. Let  $\bar{\Gamma}$  denote the multi-person nonzero-sum stochastic game. Hence, the discounted utilities of defending machine  $n_i$  and attacker  $m_j$ , which is the sum of discounted payoffs over the infinite horizon can be obtained as follows [11].

$$v_i^\beta(s, f, g) = \sum_{k=0}^{\infty} \beta_i^k \mathbb{E}_{s,f,g} \tilde{U}_i^{(k)}(a, F, s), n_i \in \mathcal{N}, \quad (9)$$

$$\tilde{v}_j^\beta(s, f, g) = \sum_{k=0}^{\infty} \tilde{\beta}_j^k \mathbb{E}_{s,f,g} \tilde{U}_j^{(k)}(a_j, F, s), m_j \in \mathcal{M}. \quad (10)$$

where  $\beta_i$  and  $\tilde{\beta}_j$  are defender's and attacker's discounted factor which are indicate their level of impatience.

## V. NASH EQUILIBRIUM ANALYSIS

In order to describe the optimal stationary defense strategies of defenders and predict the expected behavior of attackers, the solution concept of Nash Equilibrium is employed to the game  $\bar{\Gamma}$ . The existence of Stationary Nash Equilibrium is ensured based on the following theorem.

**Theorem 4.1**([26, 27]): Every nonzero-sum finite discounted stochastic game possesses at least one equilibrium point in stationary strategies.

Knowing that the payoff of players in  $\bar{\Gamma}$  is bounded, an iterative method to find a stationary  $\epsilon$ -Nash equilibrium which is sub-game perfect ( $\epsilon$ -SPNE) has been proposed in [11].

## VI. CONCLUSION AND FUTURE WORKS

In this paper, the problem of IDS configuration in a Collaborative Intrusion Detection System (CIDS) has been taken into account. In this regards, a multi-player nonzero-sum stochastic game has been utilized to model the system and the interaction between attackers and IDSs. The solution concept of Stationary Nash Equilibrium has been used to describe the optimal strategies for defenders and expected behavior of attackers. The paper culminates by demonstration of the existence of a stationary Nash equilibrium for this game. Using the proposed game model, a game theoretic methodology to update configuration policies (strategies) can be developed. For future work, we intend to design a mechanism to IDSs in a CIDS by using the solution concept of Nash equilibrium in a prescriptive way.

## REFERENCES

- [1] G. White and V. Pooch, "Cooperating security managers: Distributed intrusion detection systems," *Computers & Security*, vol. 15, pp. 441-450, 1996.
- [2] C. V. Zhou, C. Leckie, and S. Karunasekera, "A survey of coordinated attacks and collaborative intrusion detection," *Computers & Security*, vol. 29, pp. 124-140, 2010.
- [3] H. Debar, H. Curry, and D. Feinstein, "The Intrusion Detection Message Exchange Format (IDMEF)," *Internet Engineering Task Force, IETF*, vol. RFC 4765 (Experimental), 2007.
- [4] K. Bartos and M. Rehak, "Self-Organized Mechanism for Distributed Setup of Multiple Heterogeneous Intrusion Detection Systems," in *Self-Adaptive and Self-Organizing Systems Workshops (SASOW), 2012 IEEE Sixth International Conference on*, 2012, pp. 31-38.
- [5] W. Yu-Sung, B. Foo, M. Yongguo, and S. Bagchi, "Collaborative intrusion detection system (CIDS): a framework for accurate and efficient IDS," in *Computer Security Applications Conference, 2003. Proceedings. 19th Annual*, 2003, pp. 234-244.
- [6] S. T. Zargar, H. Takabi, and J. B. D. Joshi, "DCDIDP: A distributed, collaborative, and data-driven intrusion detection and prevention framework for cloud computing environments," in *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2011 7th International Conference on*, 2011, pp. 332-341.
- [7] V. Vaidya, "Dynamic signature inspection-based network intrusion detection," ed: Google Patents, 2001.

- [8] K. Alsubhi, N. Bouabdallah, and R. Boutaba, "Performance analysis in Intrusion Detection and Prevention Systems," in *Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on*, 2011, pp. 369-376.
- [9] C. Fung, Z. Quanyan, R. Boutaba, and T. Basar, "SMURFEN: A system framework for rule sharing collaborative intrusion detection," in *Network and Service Management (CNSM), 2011 7th International Conference on*, 2011, pp. 1-6.
- [10] Q. Zhu and T. Basar, "Dynamic policy-based IDS configuration," in *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, 2009, pp. 8600-8605.
- [11] Q. Zhu, H. Tembine, and T. Basar, "Network security configurations: A nonzero-sum stochastic game approach," in *American Control Conference (ACC), 2010*, 2010, pp. 1059-1064.
- [12] M. H. Manshaei, Q. Zhu, T. Alpcan, T. Başçar, and J.-P. Hubaux, "Game theory meets network security and privacy," *ACM Computing Surveys (CSUR)*, vol. 45, p. 25, 2013.
- [13] M. S. Fallah, "A puzzle-based defense strategy against flooding attacks using game theory," *Dependable and Secure Computing, IEEE Transactions on*, vol. 7, pp. 5-19, 2010.
- [14] E. Vasilomanolakis, S. Karuppayah, M. Mühlhäuser, and M. Fischer, "Taxonomy and Survey of Collaborative Intrusion Detection," *ACM Computing Surveys (CSUR)*, vol. 47, p. 55, 2015.
- [15] S. Chao and X. Shengjun, "Design and implementation of distributed collaborative intrusion detection system model," in *Fuzzy Systems and Knowledge Discovery (FSKD), 2010 Seventh International Conference on*, 2010, pp. 1224-1228.
- [16] M. E. Locasto, J. J. Parekh, A. D. Keromytis, and S. J. Stolfo, "Towards collaborative security and P2P intrusion detection," in *Information Assurance Workshop, 2005. IAW'05. Proceedings from the Sixth Annual IEEE SMC*, 2005, pp. 333-339.
- [17] Y. Rebaei, V. E. Mujica-V, and D. Sisalem, "A reputation-based trust mechanism for ad hoc networks," in *Computers and Communications, 2005. ISCC 2005. Proceedings. 10th IEEE Symposium on*, 2005, pp. 37-42.
- [18] L. Wenjuan, M. Yuxin, and K. Lam-for, "Enhancing Trust Evaluation Using Intrusion Sensitivity in Collaborative Intrusion Detection Networks: Feasibility and Challenges," in *Computational Intelligence and Security (CIS), 2013 9th International Conference on*, 2013, pp. 518-522.
- [19] W. A. Jansen, "Intrusion detection with mobile agents," *Computer Communications*, vol. 25, pp. 1392-1401, 2002.
- [20] C. J. Fung, J. Zhang, I. Aib, and R. Boutaba, "Dirichlet-based trust management for effective collaborative intrusion detection networks," *Network and Service Management, IEEE Transactions on*, vol. 8, pp. 79-91, 2011.
- [21] H. R. Ghorbani and M. R. Hashemi, "An Improved Distributed Intrusion Detection Architecture for Cloud Computing," in *Computer Networks and Distributed Systems*, ed: Springer, 2014, pp. 105-116.
- [22] C.-C. Lo, C.-C. Huang, and J. Ku, "A cooperative intrusion detection system framework for cloud computing networks," in *Parallel processing workshops (ICPPW), 2010 39th international conference on*, 2010, pp. 280-284.
- [23] M. Ghorbani and M. R. Hashemi, "Networked IDS Configuration in Heterogeneous Networks- A Game Theory Approach," *23th Iranian Conference on Electrical Engineering (ICEE2015)*, vol. 23, 2015.
- [24] C. J. Fung, J. Zhang, and R. Boutaba, "Effective acquaintance management for collaborative intrusion detection networks," in *Network and Service Management (CNSM), 2010 International Conference on*, 2010, pp. 158-165.
- [25] K. Leyton-Brown and Y. Shoham, "Essentials of game theory: A concise multidisciplinary introduction," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 2, pp. 1-88, 2008.
- [26] M. Takahashi, "Equilibrium points of stochastic non-cooperative  $n$   $n$ -person games," *Journal of Science of the Hiroshima University, Series AI (Mathematics)*, vol. 28, pp. 95-99, 1964.
- [27] A. M. Fink, "Equilibrium in a stochastic  $n$   $n$ -person game," *Journal of Science of the Hiroshima University, Series AI (Mathematics)*, vol. 28, pp. 89-93, 1964.