Optimal Monitor Placement for Detection of Persistent Threats

Karim Khalil*, Zhiyun Qian*, Paul Yu†, Srikanth Krishnamurthy* and Ananthram Swami†
* Department of Computer Science and Engineering
University of California, Riverside
Email: karimk@ucr.edu, {zhiyunq, krish}@cs.ucr.edu
† US Army Research Labs
Email: {paul.l.yu,ananthram.swami}.civ@mail.mil

***Abstract*—We study optimal monitor placement for network intrusion detection in networks with persistent attackers. The problem is modeled as a stochastic game in which the attacker attempts to control targets by delivering malicious packets to the targets while the defender attempts to detect such attempts. The state of the game is determined by the target end-systems in the network, each of which can be in either a healthy or a compromised state. Compromised targets are controlled by the attacker and may be used to inject malicious packets into the network to attack healthy targets. In addition, a random re-imaging process is deployed on all targets to regain control of compromised targets. We find the game value and the equilibrium strategies for both players under different assumptions on the knowledge of the state at the defender.**

I. Introduction

With the increase in the number and sophistication of attacks on modern networked systems, network security systems have become a crucial part of any modern network. One important element in network security is an Intrusion Detection System (IDS). IDS employs monitors to collect data which it can analyze to detect intrusive behavior in a network. Based on the data collected and the location of the monitors, IDSs can be classified as either host-based (operating system based) or network-based. Network-based IDS (NIDS) works by analyzing network traffic and data packets as they traverse the network.

Recently, game theory has been used in the study of many network security problems including intrusion detection [1]. Game theory is a mathematical framework to study situations of conflict between multiple agents and thus can be used to analyze attacker behaviors and to develop defense strategies that are based on a formal decision making process. In network security problems, two-player games are usually considered between an attacker and a defender. In [2], a zero-sum game is considered in which the attacker (intruder) chooses routes for the malicious packets while the defender chooses a sampling strategy to maximize the chances of detection under sampling budget constraints. This work was later generalized in [3] and [4], where the attack is split into multiple packets that can traverse different routes and detection is successful if a minimum number of the attack packets is detected. In [5], the attacker uses multiple entry points to attack multiple targets of varying importance.

The aforementioned works focused on single-shot games where the network topology is static with no uncertainties. In [6], a monitor placement problem is studied in a dynamic network setting in which routing tables may change randomly or sensors may be in faulty non-detecting states. These dynamics are captured by a Markov process that is known to the players and hence the model used is a stochastic game with complete information. In [7], a similar game is considered, however, the transition from one state to the next is also a function of the attacker's actions. In addition, the authors also studied the case where the transition probabilities may not be known to the players in advance. In both [6] and [7], value-iteration algorithms were used to find solutions to the game.

More recent works on network security study advanced persistent threats (APT). In [8], a timing game was introduced where the attacker and the defender compete for the control of a network asset (e.g., a server) for the longest possible time. In this model, called the FlipIt game, the attacker compromises the system periodically, and the attacks are stealthy, i.e., the compromise is not immediately detected. It is shown that among non-adaptive strategies, a periodic move strategy is optimal for both the defender and the attacker. In [9], the FlipIt game was extended to introduce an insider threat and a three-player game model was studied. In [10], a multiple server model is considered and a simulation based solution approach is adopted.

In this paper, we use stochastic games to study a network monitor placement problem as in [6]. However, we consider a more general scenario in which the actions of the attacker and the defender partially control the transitions of the states of the game. In addition, the rewards achieved by the players at each stage may vary according to the current state. As in [5], we consider a model in which the attacker can use multiple entry points to launch the attack (e.g., botnets). However, in our game, the set of entry points is not static, but varies dynamically according to the state. The state of the game in our model is determined by whether targets are healthy or compromised at a given stage. Whenever a target is compromised, we assume that the attacker may use it to launch attacks (send malicious packets) to healthy targets. A moving target defense process [10] is employed to regain control of compromised targets. This process, along with the players'

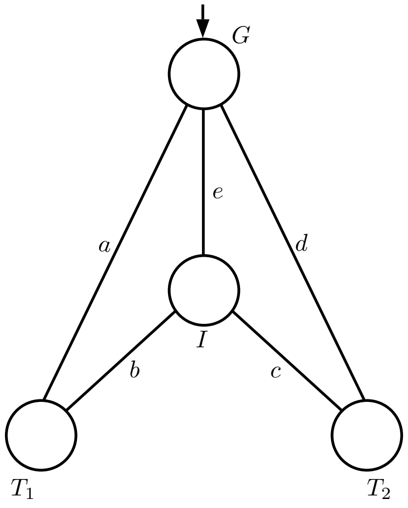

Fig. 1. Example network graph with two target nodes T_1 and T_2. Attacker chooses a target system and injects packets at G. Defender chooses an edge to deploy the monitor.

actions, specifies the stochastic process underlying the game.

Our contributions are summarized as follows. First, we present a novel model to study monitor placement in a network where compromised target systems can be used to launch attacks. To the best of our knowledge, this is the first work to consider a game theoretic solution for NIDS monitor placement for persistent attackers. Next, we characterize the equilibrium of the stochastic game for both when state information is (i) known or (ii) unknown at the defender. Finally, we present simulation results to compare the performance of the derived policies with other heuristic policies for the defender and the attacker.

The rest of this paper is organized as follows. In Section II, we introduce our system model, notation, and assumptions. Next, we present the analysis and equilibrium results in Section III. Then, we discuss simulation results in Section IV. Finally, we conclude the paper by discussing future directions in Section V.

II. System Model

In this section, we first introduce the network setup; then we present the details of our game model.

A. Network Setup

We consider an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $|\mathcal{V}| = N$ nodes and $|\mathcal{E}| = E$ edges. Let $\mathcal{V}_T \subset \mathcal{V}$ be the set of target nodes of size M. These nodes represent systems that the attacker wishes to compromise and control (e.g., servers, subnetworks).

The attacker injects malicious packets into the network and aims to compromise as many targets as possible. Without loss of generality, we assume that malicious packets are injected at point G. In addition, a compromised target might also be used by the attacker to inject malicious packets. A packet route between two nodes v_i, v_j is given by $f_k(v_i, v_j) \subset \mathcal{E}$, where k represents the index of the route, if multiple routes exist. Route $f_k(v_i, v_j)$ is the set of edges that connect nodes v_i and v_j on route k. For instance, in the example network in Fig. 1, we have $f_1(G, T_1) = \{a\}$ and $f_2(G, T_1) = \{e, b\}$.

An intrusion detection system monitors the links of $\mathcal{G}$ to detect and discard malicious packets. A monitor could be either a software application activated at an interface of an intermediate router (e.g., node I in Fig. 1, or a dedicated middle box that is installed on a certain link and is activated according to the monitoring policy chosen by the defender. Since packet inspection processing may cause latency, monitoring is constrained by a given budget and the system has to choose which links to monitor and how often to sample packets. In other words, monitoring resources are limited and thus not all links can be monitored at all times. The question is: Which links should be monitored such that malicious packets detection is maximized.

B. Game Model

The problem is modeled as a two-player zero-sum finite stochastic game [11], [12] between the attacker and the defender (intrusion detection system). We denote the game as Γ. Players have diametrically opposing objectives, and thus the reward for one player is the cost for the other. In addition, each player has a finite set of actions. Moreover, the game is played repeatedly in stages, where each stage is similar to a single shot game. At every stage, the game is defined by a state chosen from a finite set of states. The game moves from one state to the next based on the current state and the actions of the players. The details of the state model are described in the next subsection.

We assume that time is discrete. Both $\mathcal{G}$ and $\mathcal{V}_T$ are assumed to be common knowledge. At every stage of the game, the attacker decides the route, and hence the target, for its malicious packers from its action set $\mathcal{A}_a$. On the other hand, the defender chooses links on which to deploy monitors, where the action set is $\mathcal{A}_d \subset \mathcal{E}$. We assume the defender has only one monitor to deploy. However, our model and the results can be easily generalized when more than one monitor is available. For the given network in Fig. 1, $\mathcal{A}_a = \{a, eb, ec, d, bc, cb\}$ and $\mathcal{A}_d = \{a, b, c, d, e\}$. A strategy profile is a pair of actions, one for each player (e.g., (eb, e)). Note that, in this example, routes bc (cb) can be used by the attacker to send malicious packets when T_1 (T_2) is compromised.

C. State and Rewards Model

In our model, a given target node t is either in a healthy state ($s_t = \mathcal{H}$) or in a compromised state ($s_t = \mathcal{C}$). The collection of states of all target nodes determines the state of the game $S \in \mathcal{S}$ at a given stage.

We focus on persistent attackers, where the attacker's objective is to keep control of compromised targets as long as possible by launching stealthy attacks as in [8]. We consider the case in which targets (e.g., servers) are re-imaged (reconfigured) repeatedly. Whenever a target node is re-imaged, its control is immediately regained by the network administrator. The competition for control of network assets between the network administrator and persistent and stealthy attackers and periodic re-imaging strategies were first considered in [8]. In our work, this process is a given in the game, and it partially determines the stochastic transition process of the game from one state to the next.

In particular, we assume that at each stage, the network administrator randomly selects and re-images a target node t with probability p_t, such that $\sum_{t=1}^{T} p_t = 1$. A similar strategy was also considered in [10], where $p_t = 1/M$, i.e., one node is selected uniformly at every stage. This constraint models the scenario when the cost to re-image all nodes in every period is high, for example, when network operation is severely affected whenever servers are down for re-imaging. Our model could be readily extended to the case when multiple targets are selected for re-imaging at every stage. However, for simplicity of exposition, we consider the case when exactly one target is re-imaged during each stage.

The state of the game changes from one stage to the next according to the state transition function $\delta : \mathcal{S} \times \mathcal{A}_a \times \mathcal{A}_d \times \mathcal{S} \to [0, 1]$. For concreteness, in the rest of the paper we consider general network topologies with a constraint of having two target nodes. The model and the results can be extended to the more general case in a straightforward way.

Note that depending on the state, the actions available to the attacker can be different. Consider for example the network graph in Fig. 1. If the state of the game is such that $s_{T_1} = \mathcal{C}$ and $s_{T_2} = \mathcal{H}$, then the action bc is available to the attacker while the action cb is not, since T_2 cannot be used to launch attacks at this stage. This represents a game with varying action sets which is difficult to analyze. To simplify the analysis we consider an equivalent game model in which all possible actions are available to the attacker at every stage. However, the rewards for different strategies vary according to the state of the game. Specifically, with a slight abuse of notation, we define the reward of the attacker at a given stage of the game as $R_a(f_k, i, s_k)$. Here, f_k is the attacker's chosen route, i is the defender's chosen link to deploy the monitor, and s_k is the state of the origin node on route f_k. If the attack originates from G, then $s_G = \mathcal{C}$ by definition. The reward function for the attacker $R_a : \mathcal{A}_a \times \mathcal{A}_d \times \mathcal{S} \to \mathbb{R}$ is given as follows.

$$R_a(f_k, i, s_k) = \begin{cases} -1; & \text{if } i \in f_k, s_k = \mathcal{C} \\ +1; & \text{if } i \notin f_k, s_k = \mathcal{C} \\ -W; & \text{if } s_k = \mathcal{H}, \end{cases} \qquad (1)$$

where $W > 1$ is some large number. The first two cases in (1) correspond to malicious packet detection and miss detection cases, respectively. The third case corresponds to unavailable attack actions, which costs the attacker a value W. Note that in our game, the reward for the defender is $R_d(\cdot) = -R_a(\cdot)$. In addition, in the third case in (1), the reward for the defender has the same value for all defender actions and thus is irrelevant to the defender strategy. This model generalizes the model considered in [6] by including costs that vary with the game state[1]. Our model can also be easily generalized to consider targets with different weights.

TABLE I
STATE TRANSITIONS AND CORRESPONDING REWARDS.

Current State $s_t s_{-t}$	Strategy Profile	Prob.	Next State $s_t s_{-t}$	R_a
$\mathcal{HH}$	$f(x,y)\ \forall x \neq G$	1	$\mathcal{HH}$	$-W$
	$i \in f(G,y)$	1	$\mathcal{HH}$	-1
	$i \notin f(G,y)$	p_t	$\mathcal{HH}$	$+1$
		$\bar{p}_t$	$\mathcal{CH}$	
$\mathcal{CH}$	$i \in f(x,y), x \in \{G,t\}, y \neq t$	p_t	$\mathcal{HH}$	-1
		$\bar{p}_t$	$\mathcal{CH}$	
	$i \notin f(x,y), x \in \{G,t\}, y \neq t$	p_t	$\mathcal{HC}$	$+1$
		$\bar{p}_t$	$\mathcal{CH}$	
	$f(x,y), x \notin \{G,t\}$ or $y = t$	p_t	$\mathcal{HH}$	$-W$
		$\bar{p}_t$	$\mathcal{CH}$	

In table I, the state transitions as well as the corresponding rewards for a general network with two targets are presented. Actions categories of the defender and the attacker are listed in the second column. The last column represents the reward values for the attacker. The notation $-t$ refers to targets other than t and $\bar{p} = 1 - p$. The defender chooses some link $i \in \mathcal{A}_d$ while the attacker chooses a route $f(x, y) \in \mathcal{A}_a$. The outcome is specified by the presence (or absence) of the monitor on the attack route as well as the validity of the source and target nodes. Note that the state $S = \mathcal{CC}$ is not included since we assume one target server will be re-imaged every stage. Our model can be extended to the scenario in which monitors are imperfect with a positive miss detection probability, like in the model in [6], by adding more states to account for cases when the detector is faulty.

First consider the case when the current state is $S = \mathcal{HH}$. If $x \neq G$, then no malicious packets are sent and thus the new state is also $S = \mathcal{HH}$ with probability 1 and defender (attacker) gets a reward (cost) of W. On the other hand, if it happens that $i \in f(x, y)$ while $x = G$ (i.e., malicious packet detected), then the new state will be $S = \mathcal{HH}$ and the attacker pays a unit cost. The malicious packet is missed when the monitor is deployed such that $i \notin f(x, y)$ while $x = G$. Here, the attacker gets a reward of 1. If target t is not re-imaged at that stage, which happens with probability $\bar{p}_t$, the new state will be $S = \mathcal{CH}$.

Next, consider the case when the current state is $S = \mathcal{CH}$. When the attacker chooses $f(x, y)$ with $x \in \{G, t\}$ while the defender chooses $i \notin f(x, y)$, the transition to the next state is determined by which target is chosen to be re-imaged at the current state and the attacker's chosen target y. If the attacker chooses $y \neq t$, i.e., not the compromised target in the current state, and t is re-imaged at the current stage, then the states of both targets are flipped.

Finally, we assume that all target nodes are healthy at the first stage of the game, i.e., $s_v = \mathcal{H}, \forall v \in \mathcal{V}_T$, and we assume

[1] By modifying the attacker reward value in the third case of (1) to 0, we can give the attacker the option to not send malicious packets at a given stage of the game.

this information is common knowledge. At a given stage, players choose their actions, and then a target is randomly selected for re-imaging. Then, players receive their reward and the game moves to the next stage with a new state.

III. Results and Equilibrium Analysis

In this section, we study strategies for the attacker and the defender and characterize the equilibrium of the game $\Gamma(\mathcal{A}_d, \mathcal{A}_a, \mathcal{S}, \delta, R_a, \gamma)$, described in Section II. In our model, we consider an infinite horizon in which the game is played repeatedly in stages, and the reward at future stages is discounted by a factor $\gamma \in [0, 1)$. More concretely, the objective of each player is to solve

$$\underset{\pi_i \in \Delta(\mathcal{A}_i)}{\arg\max} \quad \mathbb{E}\left[\sum_{j=0}^{\infty} \gamma^j R_{ij}(\pi_i, \pi_{-i}, S_j)\right], \tag{2}$$

where π_i is the policy for player $i \in \{a, d\}$, which defines a mixed strategy distribution over the sets $\mathcal{A}_d$ and $\mathcal{A}_a$, R_{ij} is the reward received by player i at stage j, and π_{-i} is the strategy for the player other than player i. In (2), the expectation is with respect to the state transition function δ and the random variables $\pi_i \forall i$.

As a benchmark, we start with the scenario in which the defender has full state information at each stage of the game. Then, we study the case when the defender does not know the state of the game.

A. Full State Knowledge

Here, the defender is informed about the state of all the targets at each stage of the game. Since the attacker also knows the state of all targets, and these facts are common knowledge, the game is a stochastic game with complete information. In this case, the value of the game and the optimal policies can be derived using value-iteration algorithms [12]. Below we present the value-iteration algorithm and then apply it to the example scenario in Fig. 1.

The value-iteration algorithm proceeds by solving the following pair of equations iteratively until convergence.

$$V_d(S) = \max_{\pi_d \in \Delta(\mathcal{A}_d)} \min_{a \in \mathcal{A}_a} \sum_{d \in \mathcal{A}_d} Q_d(a, d, S)\pi_d, \tag{3}$$

$$Q_d(a, d, S) = R_d(a, d, S) + \gamma \sum_{S'} \delta(S, a, d, S')V_d(S'), \tag{4}$$

where $V_d(S)$ is the value of the game for the defender, i.e., the total expected discounted reward starting at state S and π_d is the mixed strategy of the defender over $\mathcal{A}_d$. $V_d(S)$ and $Q_d(a, d, S)$ can be arbitrarily initialized and the convergence to a unique equilibrium can be shown [1].

When the algorithm converges, the optimal policy for the defender will be the solution of (3). For a given state S, the problem in (3) can be reformulated as a linear program as follows.

$$\max_{V_d, \pi_d} \quad V_d(S) \tag{5}$$

$$\text{s.t.} \quad \sum_{d \in \mathcal{A}_d} Q_d(a, d, S)\pi_d \geq V_d(S), \ \forall a \in \mathcal{A}_a, \tag{6}$$

$$\sum_{d \in \mathcal{A}_d} \pi_d = 1, \ \pi_d \geq 0 \ \forall d \in \mathcal{A}_d. \tag{7}$$

Next, we apply the value iteration algorithm to the example network in Fig. 1 to find the equilibrium of the game Γ. We assume that the re-imaging process selects targets uniformly at random. Thus, we have $p_1 = p_2 = 0.5$. We also consider a discount factor $\gamma = 0.5$. Now, the state transitions and rewards matrices are completely specified. In this game, $|\mathcal{A}_d| = 5$, $|\mathcal{A}_a| = 6$ and $|\mathcal{S}| = 3$. By solving (3) and (4), the (rewards) value of the game for the defender is $(-0.5263, -0.1052, -0.1052)$ for the states $(\mathcal{HH}, \mathcal{CH}, \mathcal{HC})$. Moreover, the optimal monitor placement strategy for the defender and the optimal attack strategy are given, respectively, as follows.

$$\begin{array}{c} \\ \mathcal{HH} \\ \mathcal{CH} \\ \mathcal{HC} \end{array} \begin{array}{c} \begin{array}{ccccc} a & b & c & d & e \end{array} \\ \begin{pmatrix} \frac{1}{3} & 0 & 0 & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \end{pmatrix} \end{array}$$

$$\begin{array}{c} \\ \mathcal{HH} \\ \mathcal{CH} \\ \mathcal{HC} \end{array} \begin{array}{c} \begin{array}{cccccc} a & eb & ec & d & bc & cb \end{array} \\ \begin{pmatrix} \frac{1}{3} & \frac{1}{6} & \frac{1}{6} & \frac{1}{3} & 0 & 0 \\ 0 & 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 \\ \frac{1}{2} & \frac{1}{4} & 0 & 0 & 0 & \frac{1}{4} \end{pmatrix} \end{array}$$

Each element in the matrices above represents the probability the defender (attacker) will choose a given link (attack path), specified by the column, at a given state, specified by the row. Multiple observations on this result are noteworthy. First, since the topology and the rewards are symmetric about the two targets, the value of the game (i.e., the cost for the defender) is symmetric for the two states where one of the targets is compromised. In addition, we note that the cost to the defender is lower in either of the compromised states compared to the healthy state. This is due to the fact that the detection task in this specific topology is easier when one of the targets is compromised since the IDS has to cover fewer links with the same monitoring resources. Note also that the allocation is adaptive to the state. When both targets are healthy, routes eb, ec can be both monitored on link e, hence the uniform allocation on a, e, d is optimal. However, when one of the targets is compromised, monitoring links b or c becomes critical for detection of malicious packets that can originate from either G or the compromised target.

B. No State Knowledge

A more realistic scenario for the network intrusion detection game Γ is when the defender does not have full information about the state of all the target nodes at every stage of the game. In particular, in stealthy and persistent attacks, the compromised systems might not be immediately detected [8].

In this case, the game can be modeled as a stochastic game with incomplete information (also called partially observable stochastic game POSG), in which players have limited or no knowledge about the state of the game. While some works have presented solutions to special cases of POSG (e.g., [13]), the characterization of equilibrium and the development of algorithms to compute optimal strategies in general settings remain open problems, and an area of active research [14].

In this subsection, we assume that the defender has no knowledge of the state of the game at each stage. We will further assume that the defender will not use observations about the attacker's actions, such as detected packets in previous stages, to improve its strategy. On the other hand, we assume the attacker has full state knowledge at each stage of the game. In particular, the attacker knows when a target gets compromised and thus may use it to inject malicious packets. In addition, the attacker knows when it loses control of a target whenever it is re-imaged. We call this version of the game Γ_n.

Since the defender cannot adapt to the changes in network states, it is easy to see that the optimal monitor allocation strategy is stationary and independent of the state transitions. In this case, similar to a static zero-sum game, the defender can compute a maxmin strategy that guarantees a certain payoff regardless of the strategy of the opponent. However, in this stochastic setting, the attacker can adapt its strategy to the changing state of the network. Thus, in the rest of this section, we fix the defender's policy to a maxmin strategy over all possible attacker strategies and across different network states. In particular, the strategy for the defender is the solution of the following optimization problem.

$$\max_{\pi_d \in \Delta(\mathcal{A}_d)} \min_{a \in \mathcal{A}_a, S \in \mathcal{S}} \sum_{d \in \mathcal{A}_d} R_d(a,d,S)\pi_d, \tag{8}$$

which is equivalent to the following linear program.

$$\max_{V_d, \pi_d} \quad V_d \tag{9}$$

$$\text{s.t.} \quad \sum_{d \in \mathcal{A}_d} R_d(a,d,S)\pi_d \geq V_d, \ \forall a \in \mathcal{A}_a, S \in \mathcal{S}, \tag{10}$$

$$\sum_{d \in \mathcal{A}_d} \pi_d = 1, \ \pi_d \geq 0 \ \forall d \in \mathcal{A}_d. \tag{11}$$

Contrary to (6), the number of constraints in (10) is $|\mathcal{A}_a| \times |\mathcal{S}|$. Let the solution to (9)-(11) be given by π_d^*. Note that π_d^* always exists. The defender's policy is stationary and is independent of state $S \in \mathcal{S}$. When the defender strategy is common knowledge, the attacker can compute the defender's mixed strategy π_d^* . Given this strategy for the defender, we can define a new state transition function $\bar{\delta} : \mathcal{S} \times \mathcal{A}_a \times \mathcal{S} \to \Delta(\mathcal{S})$, such that $\bar{\delta}$ is the expectation of δ with respect to defender's mixed strategy π_d^* over the defender's action set $\mathcal{A}_d$. In addition, the expected reward matrices for the attacker $\bar{R}_a : \mathcal{A}_a \times \mathcal{S} \to \mathbb{R}$ can also be computed. It is then evident that the optimal attacker strategy would be the solution of the Markov Decision Process (MDP) defined by $\mathcal{A}_a, \mathcal{S}, \bar{\delta}, \bar{R}_a$ and γ. Suppose the optimal solution to the attacker's MDP is $a^*(S)$ for $S \in \mathcal{S}$. Then, we have the following result.

Proposition 1: For the game Γ_n with the defender policy π_d^*, the optimal attacker policy is $a^*(S)$. Moreover, the pair $(a^*(S), \pi_d^*)$ form an equilibrium of Γ_n.

Proof: Given the fixed strategy π_d^*, the best response of the attacker is the solution to the MDP$(\mathcal{A}_a, \mathcal{S}, \bar{\delta}, \bar{R}_a, \gamma)$. Since the defender is unaware of the game state and the attacker's moves, the maxmin strategy is optimal [12], i.e., is the best response. The result follows. ■

Solution methods to MDPs also include value-iteration as in (3), (4). However, one important difference compared to stochastic games is that MDPs have stationary and deterministic solutions [15]. In other words, at a given state S, the optimal policy is a pure strategy, as opposed to mixed strategies in the case of stochastic games.

Finally, we solve Γ_n for the scenario considered in Section III-A. By solving (9), we find $\pi_d^* = (\frac{2}{7}, \frac{1}{7}, \frac{1}{7}, \frac{2}{7}, \frac{1}{7})$ for actions (a,b,c,d,e) while $a^*(S) = (ec, bc, cb)$ for the states $(\mathcal{HH}, \mathcal{CH}, \mathcal{HC})$. For the attacker, the (reward) value for the game is 0.8571 for all states. The value for the single-shot zero-sum game for the defender is $r_d = -0.42855$. The total expected reward for the attacker is $\sum_{j=0}^{\infty} \gamma^j r_d = \frac{r_d}{1-\gamma} = -0.8571$. We note that the value of the game is the same for all states, due the strategy of the defender. In addition, compared to the result in Section III-A, the cost for the defender is higher due to the limited knowledge of the state. In the next section, we compare the performance of the derived strategies to other heuristic strategies through simulations.

Before concluding this section, we discuss the complexity of the value-iteration algorithm used to solve our problem. In [16], complexity analysis of the value-iteration algorithm is presented for solving MDPs. It was shown that running time for each iteration is $\mathcal{O}(|\mathcal{S}|^2|\mathcal{A}_a|^2)$ per iteration. The number of iterations is also shown to be polynomial in $\gamma/(1-\gamma)$. The algorithm in Section III-A is more complex where the size of the state space is $|\mathcal{A}_a \times \mathcal{A}_d|$ and a linear program is solved in each iteration. However, since the solution of the game considered is stationary, each player can compute the optimal strategy offline before the game starts.

IV. Simulation Results

In this section, we compare the performance of the NIDS monitor allocation strategies derived in Section III to other heuristic allocation strategies using simulations. Specifically, we measure the detection rate of malicious packets for the different strategies. We consider the network topology in Fig. 1 and consider the model parameter values in the examples presented in Section III.

For the defender strategies, we consider the optimal strategies when the state is known (KOPT) and unknown (NOPT), and a uniform allocation strategy (UNIF) in which the monitor allocation is chosen uniformly over all links. On the other hand, it is assumed that the attacker always knows the state of the targets. The attacker will select attack routes according to one of the following strategies: optimal attack strategy when

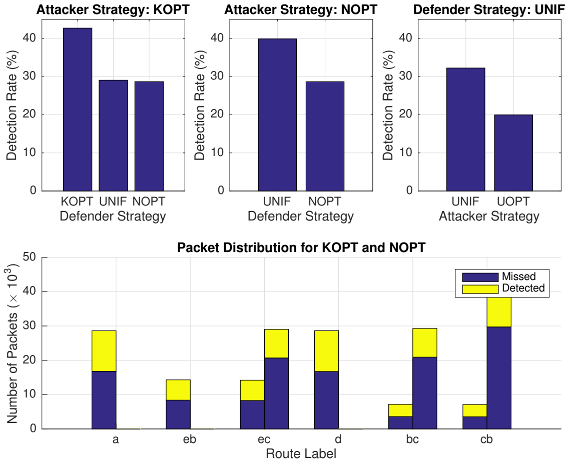

Fig. 2. The top row shows detection rates for the topology in Fig. 1 for different attack and defense strategies. The bottom row shows packet distribution over routes for KOPT (left bars) and NOPT (right bars).

the defender has state knowledge (KOPT), optimal strategy when the defender has no state knowledge (NOPT), optimal strategy when the defender is employing uniform monitor allocation (UOPT) and a uniform strategy that selects available attack routes (based on state) uniformly at random (UNIF).

In the top row of Fig. 2, the detection rates are shown for different strategy pairs. First, consider the scenario when the defender has knowledge of the state, and this fact is common knowledge. On the left graph, the attacker's strategy is fixed to KOPT. The upper bound on performance is achieved when the defender employs KOPT, achieving detection rate of 42.85%. When the attacker employs KOPT while the defender has no state knowledge and adopts NOPT or UNIF, the performance drops by at least 32%. Next, consider the case when the defender has no state knowledge and this fact is common knowledge. Here, the optimal detection rate is achieved when the defender plays NOPT. If the defender employs a UNIF strategy, however, the performance varies largely according to the attacker's strategy. Specifically, for a naive attacker, playing NOPT or UNIF improves the detection rate for the defender. However, a rational attacker optimizes its strategy given a uniform defense strategy, leading to the least possible detection rate on the right graph.

Finally, the distribution of the malicious packet injection on each link and the fraction of detected attacks are shown in bottom row of Fig. 2 for both optimal strategy pairs. The left bars represent KOPT while the right bars represent NOPT. For KOPT, it can be seen that the majority of injected packets still originate from the source, and that detection performance on the different paths is similar. However, in NOPT, most of the attacks originate from the compromised targets (paths bc, cb), while some paths are not used, consistent with our derivation in Section III. Due to the attack policy that selects the path ec when $S = \mathcal{HH}$, T_2 is compromised more often and hence more malicious packets are sent on cb compared to other paths.

V. Conclusion

We studied a monitor placement problem where attackers may control targets and use them to initiate attacks. Our model comprised a moving target defense system that randomly re-images nodes to ensure they are in a healthy state. We modeled the problem as a zero-sum stochastic game and we studied equilibrium strategies for the attacker and defender under different assumptions on state knowledge.

Based on the model and results in this paper, one can envision multiple avenues for future research. First, it will be interesting to study whether detected packets can be used to infer more information about the state of the game at the defender. Moreover, the defender can use knowledge of the identity of the re-imaged target to reduce the unknown state space. Second, it is interesting to study non-zero sum games where the cost for false alarms, for example, is considered. Finally, we will investigate a defender system in which joint strategy of monitor allocation and server re-imaging is used.

References

[1] M. H. Manshaei, Q. Zhu, T. Alpcan, T. Başar, and J.-P. Hubaux, "Game theory meets network security and privacy," *ACM Computing Surveys (CSUR)*, vol. 45, no. 3, p. 25, 2013.

[2] M. Kodialam and T. V. Lakshman, "Detecting network intrusions via sampling: a game theoretic approach," in *IEEE INFOCOM 2003*, vol. 3, pp. 1880–1889.

[3] M. Mehrandish, C. M. Assi, and M. Debbabi, "A game theoretic model to handle network intrusions over multiple packets," in *IEEE Int. Conf. on Commun. (IEEE ICC) 2006*, vol. 5, pp. 2189–2194.

[4] H. Otrok, M. Mehrandish, C. Assi, M. Debbabi, and P. Bhattacharya, "Game theoretic models for detecting network intrusions," *Computer Communications*, vol. 31, no. 10, pp. 1934–1944, 2008.

[5] O. Vaněk, Z. Yin, M. Jain, B. Bošanskỳ, M. Tambe, and M. Pěchouček, "Game-theoretic resource allocation for malicious packet detection in computer networks," in *Proc. of the 11th Int. Conf. on Autonomous Agents and Multiagent Systems 2012*, vol. 2, pp. 905–912.

[6] S. Schmidt, T. Alpcan, Ş. Albayrak, T. Başar, and A. Mueller, "A malware detector placement game for intrusion detection," in *Critical Information Infrastructures Security*. Springer, 2007, pp. 311–326.

[7] T. Alpcan and T. Başar, "An intrusion detection game with limited observations," in *Proc. of the 12th Int. Symp. on Dynamic Games and Applications*, 2006.

[8] M. Van Dijk, A. Juels, A. Oprea, and R. L. Rivest, "Flipit: The game of stealthy takeover," *Journal of Cryptology*, vol. 26, no. 4, pp. 655–713, 2013.

[9] X. Feng, Z. Zheng, P. Hu, D. Cansever, and P. Mohapatra, "Stealthy attacks meets insider threats: a three-player game model," in *IEEE MILCOM 2015*, pp. 25–30.

[10] M. P. Wellman and A. Prakash, "Empirical game-theoretic analysis of an adaptive cyber-defense scenario (preliminary report)," in *Decision and Game Theory for Security*. Springer, 2014, pp. 43–58.

[11] L. S. Shapley, "Stochastic games," *Proceedings of the National Academy of Sciences*, vol. 39, no. 10, pp. 1095–1100, 1953.

[12] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Proc. of the Eleventh Int. Conf. on Machine Learning*, vol. 157, 1994, pp. 157–163.

[13] E. A. Hansen, D. S. Bernstein, and S. Zilberstein, "Dynamic programming for partially observable stochastic games," in *Proc. of the 19th National Conf. on Artifical Intelligence*, 2004, pp. 709–715.

[14] E. Solan and N. Vieille, "Stochastic games," *Proceedings of the National Academy of Sciences*, vol. 112, no. 45, pp. 13 743–13 746, 2015.

[15] D. P. Bertsekas, *Dynamic programming: deterministic and stochastic models*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1987.

[16] M. L. Littman, T. L. Dean, and L. P. Kaelbling, "On the complexity of solving markov decision problems," in *Proc. of the 11th Conf. on Uncertainty in Artificial Intelligence*, 1995, pp. 394–402.