2015 IEEE 54th Annual Conference on Decision and Control (CDC)
December 15-18, 2015. Osaka, Japan

A Host Takeover Game Model for Competing Malware

Phillip Lee, Andrew Clark, Basel Alomair, Linda Bushnell, and Radha Poovendran

Abstract— **Malware, or malicious software, degrades the performance of cyber-physical systems by infecting cyber systems and compromising the information exchange between cyber and physical components. Advanced malwares have the ability to modify their code over time to escape detection, while also removing competing malwares from a targeted host. In this paper, we model the interaction of multiple adaptive, competing malwares and a system owner via a resource takeover game known as FlipIt, or a game of "stealthy takeover." We characterize the unique Nash equilibrium of a generalization of FlipIt with an arbitrary number of players. We then prove that, by greedily updating their strategies at each time step using only local information, the malwares will converge to a unique Nash equilibrium. In addition, we derive the optimal mitigation strategy against competing malwares as the solution to a Stackelberg game and develop an efficient algorithm for computing the equilibrium. Our results are demonstrated via a numerical study, in which we analyze the behavior of the malwares prior to convergence to the equilibrium and compare the impact of heterogeneous and homogeneous malwares on the system owner's utility.**

I. INTRODUCTION

In a malware propagation attack, a malware process takes partial or complete control over a system, modifies the system's intended functionality, and attempts to infect other devices [1]. The potential impact of malware is especially severe in cyber-physical systems, which depend on real-time exchange of control information between cyber and physical components. As seen in the case of Stuxnet [2], malwares can cause significant physical damage by compromising the integrity and availability of this information exchange.

An increase in malware sophistication has been observed in recent years, adding to the security threats to cyber-physical systems [1], [3]. Two salient features have been observed in advanced malwares. The first feature is the malware's ability to mutate its code over time, known as polymorphic worms, enabling a malware to avoid signature-based intrusion detection while keeping its functionality intact [3]. The second feature is the competition among different malwares for control of targeted devices, in which a newly-installed malware detects and removes other malwares from the system [1]. The *persistent and adaptive nature* of malware creates a continuous, strategic interaction between the system owner and multiple competing malwares, which must be modeled and understood in order to develop effective mitigation strategies.

The FlipIt game was recently proposed in the security community to model defense against such advanced persistent threats [4]. In FlipIt, two players (attacker and defender) continuously compete for control of a host. The fraction of the time that each player controls the device, together with the resources expended to take over the device at different time instances, quantify the effectiveness of the defense strategy and provide insight into the optimal system defense.

FlipIt provides a first step in modeling the interaction between the system owner and a persistent malware. Currently, however, there is no framework in which the system owner defends against an arbitrary number of malwares competing over a single resource. Furthermore, while existing literatures have analyzed the equilibrium behavior of the attacker and defender, there has been no study of the dynamic behavior of the players, who will update their strategies based on observation of the other player's strategies. In particular, identifying efficient and realistic strategies that can be implemented by the defender and multiple malwares, and guarantee convergence to the Nash equilibrium, is an open problem for FlipIt.

In this paper, we develop a control-theoretic approach to modeling the strategies of adaptive, competing malware in FlipIt, as well as designing an optimal mitigation strategy. We consider the class of adversaries and defenders who adopt exponential takeover strategies, in which the host is compromised (by the adversary) or restored (by the defender) according to a Poisson process. The strategy is characterized by the takeover rate, and was identified as one of the key strategies in FlipIt [4] due to its ease of implementation and unpredictability to other players. Our approach is to model the takeover rate of the defender and each malware as a dynamical system, where the takeover rate of each player is a state variable that changes over time based on that player's current utility.

- We generalize the FlipIt game to include one defender and an arbitrary number of competing malwares, each of which employs an exponential strategy with a time-varying takeover rate. We then prove the existence and uniqueness of the Nash equilibrium, and present a closed form characterization of the equilibrium.
- We formulate a dynamical model for the takeover rate of each malware, in which the malware owner observes the sum of the takeover rates and updates its own takeover rate according to gradient ascent. We prove that the

P. Lee, L. Bushnell, and R. Poovendran are with the Department of Electrical Engineering, University of Washington, Seattle, WA 98195 USA. {leep3, lb2, rp3}@uw.edu

A. Clark is with the Department of Electrical and Computer Engineering, Worcester Polytechnic Institute, Worcester, MA 01609 USA. aclark@wpi.edu

B. Alomair is with the National Center for Cybersecurity Technology, King Abdulaziz City for Science and Technology, Riyadh, Saudi Arabia. alomair@kacst.edu.sa

This work was supported by ONR grant N00014-14-1-0029 and a grant from the King Abdulaziz City for Science and Technology (KACST).

proposed dynamics guarantee convergence to the Nash equilibrium via passivity based analysis.

- We investigate the optimal mitigation strategy of the system. We model the mitigation as a Stackelberg game, and prove that the optimal mitigation strategy can be obtained as the solution to a convex optimization problem.
- We evaluate the convergence of malware strategies to the Nash equilibrium via a numerical study. We verify that the proposed dynamics ensure convergence to the Nash equilibrium, and characterize the utility of the system owner under different mitigation parameters.

The paper is organized as follows. We review the related work in Section II. Section III contains our assumptions of the system and malwares, as well as background on FlipIt. Section IV presents a host takeover game formulation for multiple malwares and characterizes the unique Nash equilibrium. We describe the dynamics of each player's strategy and prove convergence to the equilibrium in Section V. In Section VI, we propose a polynomial time algorithm for computing the optimal mitigation strategy. Section VII includes our numerical results. Section VIII concludes the paper.

II. Related Work

Modeling and mitigating malwares have been significant areas of research in both control and security research communities [5], [1]. The existing literatures have focused on modeling the self-propagating aspect of a malware on a network. In these works, variations of epidemic dynamics were adapted to model the propagation of a single malware, and efficient mitigation strategies were derived from control theoretic approaches [5], [6].

The competing nature of advanced malwares has been observed in the security community as in the cases of SpamThru and Tigger [1]. However, analytically modeling the interaction between different malwares has only been investigated recently. In [7], malware interactions including coexisting and competing cases were modeled as continuous time Markov chains. This model, however, assumes that the compromise rates of malware are static, and hence does not incorporate the adaptive nature of malwares.

Game theory has emerged as an important methodology for modeling the interaction between intelligent cyber attackers and defenders, and developing efficient mitigation strategies [8], [9]. Recently, the FlipIt game was proposed to model persistent resource takeover of a single attacker and defender [4]. FlipIt was generalized to competition between two players over multiple resources in [10]. Empirical analysis of takeover strategies in FlipIt was performed in [11].

III. Model and Preliminaries

In this section, we present the adversary and system models, and give a formal definition of FlipIt [4].

A. Adversary Model

We consider multiple persistent malwares that compete to take control of a host. Once a malware infects the host, it removes any existing malware and installs an anti-virus software to detect attempted infections from other malwares. Each malware's signature is learned over time by both the owner of the host and other malwares. In order to avoid detection, each malware's author will change the malware's binaries to change its signature. The effort required for the author to change the binaries creates a cost associated with infecting the host for each malware.

B. System model

A system owner regularly inspects and cleans the targeted host from any potential infection from malwares. The owner inspects signatures of malwares that infected the host and patches exploited software vulnerabilities. In addition, the owner updates its anti-virus software to prevent any future infection from known malwares. We assume that during the cleaning process, the host will be taken offline, and will be unavailable for use. This creates a cost associated with cleaning the host.

C. FlipIt

FlipIt is a two-player game where attacker and defender are competing over a shared resource. Each player can make a *move* at any time. Once a player makes a move, the player owns the resource until the opponent makes a move. Each time player i makes a move, player i pays an associated cost $c_i > 0$. The number of moves made by player i up to time t is denoted as $n_i(t)$.

The utility of each player is defined as the average fraction of time the player owns the resource minus the average cost of takeover. Formally, the utility of player i is given as

$$U_i = \liminf_{t\to\infty} \frac{1}{t}\left(\int_0^t C_i(\tau)d\tau - c_i n_i(t)\right) \qquad (1)$$

where $C_i(t) = 1$ if player i owns the resource at time t, and $C_i(t) = 0$ otherwise.

IV. Game Formulation and Nash Equilibrium

In this section, we formally define the resource takeover game between multiple malwares, and prove the existence and uniqueness of the Nash equilibrium of the proposed game.

A. Game Formulation

We consider an $(n+1)$-player resource takeover game between the system owner and n competing malwares. The system owner is indexed as player 0, while the n malwares are indexed from 1 to n. Malware i has a cost per takeover denoted as c_i. Without loss of generality, we index the malwares based on the rank order of their respective costs c_i, such that $c_1 \leq c_2 \leq ... \leq c_n$. We denote $\{T_i(m)\}$ as the sequence of times that player i has taken over the resource.

In this paper, we assume that all $n+1$ players employ exponential strategies. A takeover strategy is defined as exponential if the time differences between two consecutive takeovers, i.e., $\{T_i(m) - T_i(m-1)\}$ are independent, exponential random variables. The exponential strategy was proposed in [4] due to its unpredictability compared to deterministic periodic strategies.

We consider the case when the system owner sets its takeover rate x_0 and malwares play a noncooperative game to choose the takeover rates given a fixed x_0. We denote the rate of takeover for malware i as x_i. We say that a malware *drops out* of the game if its takeover rate $x_i = 0$ and say that a malware *participates* in the game if $x_i > 0$.

Theorem 1: The utility of malware i when each malware employs an exponential strategy with rate x_i, and the owner employs an exponential strategy with rate x_0, is given as

$$U_i(\mathbf{x}) = \frac{x_i}{\sum_{j=0}^{n} x_j} - c_i x_i. \quad (2)$$

The proof when two players can be found in [4], and the proof for the general case can be found in the arxiv version of this paper.

B. Characterization of the Nash Equilibrium

We define $\mathbf{x}_{-i} := \{x_j : j \neq i\}$. The Nash equilibrium (N.E.) is defined as $\bar{\mathbf{x}} = (\bar{x}_1, \dots, \bar{x}_n) \in \mathbb{R}_+^n$ such that

$$U_i(\bar{x}_i, \bar{\mathbf{x}}_{-i}) \geq U_i(x_i, \bar{\mathbf{x}}_{-i}) \text{ for } i = 1, ..., n$$

for all $x_i \neq \bar{x}_i$, where $\bar{\mathbf{x}}_{-i} = \{\bar{x}_j : j \neq i\}$. The characterization of the Nash equilibrium is given in the following Theorem. To simplify notations, we denote $S_c(r) := \sum_{i=1}^{r} c_i$, and $f(x_0, r) := \sqrt{(r-1)^2 + 4x_0 S_c(r)}$ for $r \geq 1$ for the rest of the paper.

Theorem 2: For a given x_0, there exists a unique Nash equilibrium amongst n malwares. If c_1 satisfies the inequality $c_1 \geq \frac{1}{x_0}$, then $\bar{x}_i = 0$ for all malwares. If $c_1 < \frac{1}{x_0}$, then the unique Nash equilibrium is characterized as follows:

If there exists a malware whose cost c_{m+1} satisfies

$$c_{m+1} \geq \frac{2S_c(m)}{m - 1 + f(x_0, m)} \quad (3)$$

then $\bar{x}_j = 0$ for all $j \geq m + 1$. For malwares $i = 1, ..., m$, the N.E. is given as

$$\bar{x}_i = \frac{-x_0 c_i}{S_c(m)} + \frac{m - 1 + f(x_0, m)}{2(S_c(m))^2} \left(S_c(m) - (m-1)c_i\right). \quad (4)$$

Proof: In order to characterize the Nash equilibrium, we first compute the best response of malware i given the sum of the other players' takeover rates. It can be verified that $\frac{\partial^2 U_i}{\partial x_i^2} \leq 0$ for a given $\sum_{j\neq i} x_j$. Therefore, U_i is a concave function in x_i given $\sum_{j\neq i} x_j$, and the best response of malware i, denoted as BR_i, is given as the solution of $\frac{\partial U_i}{\partial x_i} = 0$ when $BR_i(\sum_{j\neq i} x_j) > 0$ and 0 otherwise. The best response of malware i can be explicitly written as

$$BR_i(\sum_{j\neq i} x_j) = \begin{cases} \sqrt{\frac{\sum_{j\neq i} x_j}{c_i}} - \sum_{j\neq i} x_j, & c_i < \frac{1}{\sum_{j\neq i} x_j} \\ 0, & \text{else} \end{cases} \quad (5)$$

To show that $\bar{x}_i$'s constitute a NE, it suffices to show that $BR_i(\sum_{j\neq i} \bar{x}_j) = \bar{x}_i$. Replacing for $\bar{x}_i$ from Theorem 2, we obtain

$$\bar{u}(x_0) := x_0 + \sum_{i=1}^{m} \bar{x}_i = \frac{m - 1 + f(x_0, m)}{2S_c(m)}. \quad (6)$$

Therefore, $BR_j(\bar{u}(x_0)) = 0$ for all malware j such that $c_j \geq \frac{1}{\bar{u}(x_0)}$. For the malwares indexed $i = 1, \dots, m$, we need to show that

$$BR_i(x_0 + \sum_{j\neq i} \bar{x}_j) = \sqrt{\frac{\bar{u}(x_0) - \bar{x}_i}{c_i}} - (\bar{u}(x_0) - \bar{x}_i) = \bar{x}_i \quad (7)$$

which is equivalent to $\bar{x}_i = \bar{u}(x_0) - \bar{u}(x_0)^2 c_i$. Note that $\bar{u}^2(x_0)$ can be expressed as

$$\begin{aligned} \bar{u}^2(x_0) &= \frac{(m-1)^2 + (m-1)f(x_0, m) + 2x_0 S_c(m)}{2S_c(m)^2} \\ &= \bar{u}(x_0)\frac{m-1}{S_c(m)} + \frac{x_0}{S_c(m)}. \end{aligned}$$

Therefore,

$$\bar{u}(x_0) - \bar{u}(x_0)^2 c_i = -\frac{x_0 c_i}{S_c(m)} + \bar{u}(x_0)\frac{S_c(m) - (m-1)c_i}{S_c(m)}$$

which shows that $\bar{u}(x_0) - \bar{u}(x_0)^2 c_i = \bar{x}_i$ as given in (4) for all $i = 1, \dots, m$.

Since any N.E. $\bar{\mathbf{x}}$ has to satisfy $\bar{u}(x_0) - \bar{u}(x_0)^2 c_i = \bar{x}_i$ for $i = 1, \dots, m$, summing over $\bar{u}(x_0) - \bar{u}(x_0)^2 c_i = \bar{x}_i$ from $i = 1$ to m results in

$$S_c(m)\bar{u}(x_0)^2 - (m-1)\bar{u}(x_0) - x_0 = 0$$

which is a quadratic equation in $\bar{u}(x_0)$. This quadratic equation admits one positive, and one negative solution with the positive solution being equation (6). Substituting $\bar{u}(x_0)$ from equation (6) to $\bar{x}_i = \bar{u}(x_0) - \bar{u}(x_0)^2 c_i$ for $i = 1, \dots m$, we obtain $\bar{x}_i$ as equation (4). This establishes the uniqueness of the Nash equilibrium.

■

Theorem 2 establishes existence and uniqueness of the Nash equilibrium, but does not guarantee that the equilibrium is reached by the malwares. In the next section, we prove convergence to the Nash equilibrium arising from simple distributed strategies of the players.

V. Convergence to the Nash Equilibrium

In this section, we propose a dynamical model of the malware strategy, in which each adversary greedily updates its takeover rate at each time step. We prove that, under these greedy dynamics, the takeover rates converge to the unique N.E. identified in Section IV.

The proof that all malwares converge to the N.E. of Theorem 2 is divided into three parts. In the first part (Section V-B), we prove that all malwares with $\bar{x}_i = 0$ will eventually drop out under the proposed dynamics. In the second part (Section V-C), we establish that there exists a finite time T_{part} such that $x_i(t) > 0$ for all malwares i

with $\bar{x}_i > 0$ and all $t > T_{part}$. In the third part (Section V-D), we prove convergence to the unique N.E. using passivity analysis, under the assumption that all malwares $1, \ldots, m$ participate and malwares $(m+1), \ldots, n$ have dropped out (i.e., $t > \max\{T_{drop}, T_{part}\}$).

A. Greedy Dynamics of Malwares

We assume that each malware can observe the sum of takeover rates $u(t) := \sum_{i=0}^{n} x_i(t)$ at time t. Each malware i can compute the total takeover rate $u(t)$ by observing the fraction of time that i controls the host.

We consider an adversary who, at each time t, updates its takeover rate $x_i(t)$ to move in the gradient direction, corresponding to an adaptive and greedy (myopic) adversary. This behavior results in dynamics

$$\dot{x}_i(t) = \left(-x_i(t) + u(t) - u^2(t)c_i\right)^+_{x_i} \tag{8}$$

where $(\cdot)^+_{x_i}$ is the positive projection defined as

$$(h(x_i,u))^+_{x_i} = \begin{cases} 0, & x_i = 0 \text{ and } h(x_i,u) < 0 \\ h(x_i,u), & \text{else} \end{cases}$$

We say that the positive projection is active in the case $x_i = 0$ and $h(x_i,u) < 0$, and inactive otherwise. The dynamics (8) capture the behavior of an adversary who attempts to maximize his utility at each time step, and can be computed by an adversary using only knowledge of the total takeover rate $u(t)$. We now show the proposed dynamics guarantee convergence to the Nash equilibrium in Theorem 2.

B. Malware Dropout

In this subsection, we show that all malwares $i \geq m+1$ will eventually drop out of the game, i.e., there exists time T_{drop} such that the positive projections will remain active for malwares $i \geq m+1$ at any time $t > T_{drop}$. We begin by proving the following lemmas.

Lemma 1: Suppose there exists a malware $m+1$ with cost c_{m+1} satsifying the inequality (3). Then, for all malwares indexed $j \geq m+1$, the following inequalities hold:

$$c_j \geq \frac{2S_c(j)}{j + f(x_0, j-1)} \tag{9}$$

$$c_j \geq \frac{-(j-2) + f(x_0, j-1)}{2x_0} \tag{10}$$

$$= \frac{2S_c(j-1)}{j-2+f(x_0,j-1)} \tag{11}$$

The proof is omitted due to space constraints.

We now develop an inductive proof that malwares $\{m+1, \ldots, n\}$ eventually drop out. We first show that, for any $j \geq (m+1)$, if malwares $\{(j+1), \ldots, n\}$ drop out, then malware j will eventually drop out as well.

Lemma 2: Consider the set $U_j = \{\frac{1}{c_{j+1}} \leq u(t) \leq \frac{1}{c_j}\}$ where $m+1 \leq j \leq n-1$. Suppose that $x_i(t) = 0$ for $i \geq j+1$, and $\bar{u}(x_0) \neq \frac{1}{c_j}$. Then, $\dot{u}(t) > 0$ for all $u(t) \in U_j$, and there exists a finite time $T_{escape}(j)$ such that $u(t) > \frac{1}{c_j}$ for all $t \geq T_{escape}(j)$. If $x_i(t) = 0$ for $i \geq j+1$, and $\frac{1}{c_j} = \bar{u}(x_0)$, then $u(t)$ converges to $\bar{u}(x_0)$ if $u(t_0) \in U_j$ for some t_0.

Proof: If $u(t) \in U_j$, then positive projections are inactive for all malwares $1, \ldots, j$ since $u - u^2 c_i \geq 0$ for all $i \leq j$. Moreover, $u - u^2 c_i \leq 0$ for all $i = (j+1), \ldots, n$. If $x_i(t) = 0$ for $i \geq j+1$, then the positive projections are active for $i \geq j+1$, and $\dot{u}(t)$ can be computed as

$$\dot{u}(t) = (j-1)u - u^2 S_c(j) + x_0 \tag{12}$$

We will now show that $\dot{u}(t) \geq 0$ for $\{\frac{1}{c_{j+1}} \leq u(t) \leq \frac{1}{c_j}\}$ if $x_i(t) = 0$ for $i \geq j+1$. We have

$$\begin{aligned} \dot{u}(t) &= u\left(j - 1 - uS_c(j)\right) + x_0 \\ &\geq \frac{j-2-f(x_0,j-1)}{2c_j} + x_0 \geq -x_0 + x_0 = 0 \end{aligned}$$

The first inequality follows from the assumption that $u(t) \geq \frac{1}{c_j}$, the lower bound (9), and the fact that $j-2-f(x_0,j-1) < 0$ for all $x_0 > 0$. The last inequality is from (10) and the negativity of $j-2-f(x_0,j-1)$.

The set of inequalities hold with equality if and only if $u(t) = \frac{1}{c_j}$, and c_j satisfies the inequalities (9)-(11) with equality. From the proof of Lemma 1, this is true if and only if $c_j = c_{m+1} = \frac{1}{\bar{u}(x_0)}$. Therefore, if $\frac{1}{c_j} \neq \bar{u}(x_0)$, then $\dot{u}(t) > 0$ for $u(t) \in U_j$.

Since $u(t)$ is monotonically increasing in U_j, there exists a finite time $T_{escape}(j)$ such that $u(t)$ will exit U_j to the set U_{j-1}, and $u(t)$ will not enter U_j for all $t \geq T_{escape}(j)$ since $\dot{u}(t) > 0$ at the boundary point $u(t) = \frac{1}{c_j}$.

If $\frac{1}{c_j} = x_0$, then $\dot{u}(t) > 0$ in U_j and $\dot{u}(t) = 0$ at $u(t) = \frac{1}{c_j}$. Hence $u(t) \rightarrow \bar{u}(x_0)$ if $u(t) \in U_j$ initially. ■

Given Lemmas 1 and 2, we now prove the following theorem, which states that malwares $i \geq m+1$ will eventually drop out of the game.

Theorem 3: If $\bar{u}(x_0) \neq \frac{1}{c_{m+1}}$, then there exists a finite time T_{drop} such that $u(t) > \frac{1}{c_{m+1}}$ and $x_i(t) = 0$ for $i \geq m+1$ and $t \geq T_{drop}$. If $\bar{u}(x_0) = \frac{1}{c_{m+1}}$, and $u(t) \leq \frac{1}{c_{m+1}}$, then $u(t)$ converges to $\bar{u}(x_0)$.

Proof: First assume that $\bar{u}(x_0) \neq \frac{1}{c_{m+1}}$. We start by considering the case $0 \leq u(t) \leq \frac{1}{c_n}$, and use induction to prove that all malwares $i \geq m+1$ will drop out of the game. For $0 \leq u(t) \leq \frac{1}{c_n}$, the positive projection is inactive for all malwares. Therefore, $\dot{u}(t)$ is written as

$$\dot{u}(t) = (n-1)u - u^2 S_c(n) + x_0 \tag{13}$$

By the same argument as we made for the set $U_j = \{\frac{1}{c_{j+1}} \leq u(t) \leq \frac{1}{c_j}\}$ in Lemma 2, $\dot{u}(t) > 0$, for $\{0 < u(t) \leq \frac{1}{c_n}\}$, and there exists $T_{escape}(n)$ such that $u(t) > \frac{1}{c_n}$ for all $t \geq T_{escape}(n)$.

Moreover, in the interval $u(t) \geq \frac{1}{c_n}$, $x_n(t)$ is monotonically decreasing since

$$\dot{x}_n(t) = \left(-x_n(t) + u(t) - u^2(t)c_n\right)^+_{x_n} < 0 \text{ for } x_n(t) > 0$$

Therefore, there exists a finite time $T_{drop}(n)$ such that $x_n(t) = 0$ for all $t \geq T_{drop}(n)$, and the positive projection will remain active for malware n.

Given that malware n's positive projection is active, from Lemma 2, we conclude that there exists a finite time

$T_{escape}(n-1)$ such that $u(t) > \frac{1}{c_{n-1}}$ for all $t > T_{escape}(n-1)$. Since $x_{n-1}(t)$ is monotonically decreasing when $u(t) > \frac{1}{c_{n-1}}$, there exists a time $T_{drop}(n-1)$ such that malware $(n-1)$ will be active for $t \geq T_{drop}(n-1)$.

Inductively, given $x_i(t) = 0$ for $i \geq j+1$, we conclude that $u(t)$ will escape the set $\{\frac{1}{c_{j+1}} < u(t) \leq \frac{1}{c_j}\}$ and will remain in $\{u(t) > \frac{1}{c_j}\}$. This proves that there exists time T_{drop} such that for all $t > T_{drop}$, $u(t) > \frac{1}{c_{m+1}}$ and $x_i(t) = 0$ for $i \geq m+1$.

On the other hand, suppose that $\bar{u}(x_0) = \frac{1}{c_{m+1}}$, and $u(t) \leq \frac{1}{c_{m+1}}$ initially. Then, for all U_j such that $c_j \neq c_{m+1}$ and $j > m+1$, $u(t)$ will eventually escape U_j in a finite time and enter $U_{c_{m+1}}$. From Lemma 2, $u(t)$ will converge to $\frac{1}{c_{m+1}}$, which completes the proof. ∎

C. Participation of Malwares

In this subsection, we prove that all malwares with positive takeover rates $\bar{x}_i > 0$ at N.E. will eventually participate in the game. In other words, we will show that all positive projections will eventually become inactive and remain inactive for malwares $i \leq m$.

The following inequality holds for malwares $i = 1, \dots, m$, and can be derived by the same approach as Lemma 1.

$$c_j < \frac{2S_c(j)}{j + f(x_0, j-1)} \tag{14}$$

$$c_j < \frac{2S_c(j-1)}{j - 2 + f(x_0, j-1)} \tag{15}$$

$$= \frac{-(j-2) + f(x_0, j-1)}{2x_0} \tag{16}$$

Furthermore, $c_1 < \frac{1}{x_0}$.

The following theorem shows that malware $i \leq m$ will eventually participate in the game.

Theorem 4: There exists some finite time T_{part} such that, for all $t > T_{part}$, $u(t) < \frac{1}{c_m}$ and positive projections are inactive for malwares $i \leq m$.

Proof: Consider the set $U_1 = \{u(t) : \frac{1}{c_1} \leq u(t) < \infty\}$. In U_1, $\dot{u}(t) = 0$ if and only if $x_1, \dots, x_n = 0$ and $u(t) = x_0$ since $u - u^2 c_i \leq 0$ for all $i \geq 1$. However, this case is not possible since $x_0 < \frac{1}{c_1}$ which violates the definition of U_1. Therefore, $\dot{u}(t) < 0$ in U_1 and hence there exists a finite time $T_{escape}(1)$ such that $u(t) < \frac{1}{c_1}$ for all $t \geq T_{escape}(1)$.

Now, consider the set $U_j = \{u(t) : \frac{1}{c_j} \leq u(t) < \frac{1}{c_{j-1}}\}$ for $2 \leq j \leq m$. In U_j, positive projections are inactive for malwares $i \leq j-1$. Denote $I(t) = \{i : i \geq j, \dot{x}_i \neq 0\}$. Then $\sum_{i=1}^{j-1} \dot{x}_i(t)$ can be bounded above by

$$\begin{aligned}
\sum_{i=1}^{j-1} \dot{x}_i(t) &= -\sum_{i=1}^{j-1} x_i + u(t)(j-1) - u^2(t) S_c(j) \\
&= u(t)\left(j - 2 - u(t) S_c(j-1)\right) + x_0 + \sum_{k \in I(t)} x_k(t) \\
&< \frac{1}{c_j}\left(\frac{j-2-f(x_0, j-1)}{2}\right) + x_0 + \sum_{k \in I(t))} x_k(t) \\
&\leq -x_0 + x_0 + \sum_{k \in I(t)} x_k(t) = \sum_{k \in I(t)} x_k(t)
\end{aligned}$$

where the first inequality follows from the assumptions that $u(t) > \frac{1}{c_j}$ and (15), and the last inequality is from (16). However, in U_j, $\dot{x}_i \leq -x_i$ for all $i \in I(t)$ since $u - u^2 c_i \leq 0$ for all $i \geq j$. Therefore, $\dot{u}(t) < 0$, and hence there exists a finite time $T_{escape}(j)$ such that $u(t) < \frac{1}{c_j}$ for all $t \geq T_{escape}(j)$. Let $T_{part} = \max_j\{T_{escape}(j)\}$, then $u(t) < \frac{1}{c_m}$ after for all $t \geq T_{part}$. Moreover since $u - u^2 c_i > 0$ for all $i \leq m$, positive projection will be inactive for all $t \geq T_{part}$. ∎

The following Corollary states that all malwares $i \geq m+1$ will eventually drop out of the game and the game will be played only by the malwares $i \leq m$.

Corollary 1: There exists a finite time T such that for all $t > T$, $x_i = 0$ and $\dot{x}_i = 0$ for malwares $i \geq m+1$, and positive projections are inactive for malwares $i \leq m$. Moreover, $u(t) \in \left[\frac{1}{c_{m+1}}, \frac{1}{c_m}\right]$ for $t \geq T$.

Proof: Let $T = \max\{T_{drop}, T_{part}\}$, where T_{drop} and T_{part} were defined in Theorems 3 and 4 respectively. It is straightforward to conclude that claims of this Corollary is true for $t > T$ from the definitions of T_{drop} and T_{part}. ∎

Corollary 1 implies that all malwares with $\bar{x}_i = 0$ (e.g., $i > m$) will eventually drop out and all malwares with $\bar{x}_i > 0$ (e.g., $i \leq m$) will eventually participate. In what follows, we show that the dynamics of the participating malware takeover rates eventually converge to the N.E.

D. Passivity Approach for Convergence to the N.E.

In this section, we use a passivity analysis to prove that the proposed dynamics (8) guarantees convergence to the N.E. when only the malwares $i \leq m$ participate in the game.

Define $v_i(t) = u(t) - u^2(t) c_i$ and $\bar{v}_i = \bar{u}(t) - \bar{u}(t)^2 c_i$. Since positive projections are inactive in the interval $\left[\frac{1}{c_{m+1}}, \frac{1}{c_m}\right]$ for all $i \leq m$, $\dot{x}_i$ is given as

$$\dot{x}_i(t) = -x_i(t) + u(t) - u^2(t) c_i = -x_i(t) + v_i(t) \tag{17}$$

The main intuition of the proof is the following. If $u(t) \to \bar{u}(x_0)$, then $v_i(t) \to \bar{v}_i = \bar{x}_i$ for all i. Then the dynamics of x_i will guarantee that $x_i \to \bar{u} - \bar{u}^2 c_i$, which is the Nash equilibrium of malware i as shown in Section IV.

We will first show that $u(t)$ converges to $\bar{u}(x_0)$ using a Lyapunov method.

Lemma 3: Under the dynamics (8), $u(t)$ asymptotically converges to $\bar{u}(x_0)$.

Proof: Consider the Lyapunov function

$$V_1(u) = \gamma \int_{\bar{u}}^{u} (\sigma^2 - \bar{u}^2) d\sigma$$

where $\gamma > 0$. Then, $V_1(\bar{u}) = 0$, $\frac{dV_1}{du} = 0$ if $u = \bar{u}$, and $\frac{d^2 V_1}{du^2} = 2u \geq 0$ for $u \geq 0$. Therefore, in the region $u \geq 0$, V_1 is a convex function which achieves its global minimum zero at $u = \bar{u}$. Differentiating $V_1(u)$ with respect to time, we obtain

$$\begin{aligned}
& \dot{V}_1(u) = \gamma(u^2 - \bar{u}^2)\dot{u} \\
= \ & \gamma(u^2 - \bar{u}^2)\left((m-1)(u-\bar{u}) - S_c(m)(u^2 - \bar{u}^2)\right) \\
= \ & \gamma(u+\bar{u})(u-\bar{u})^2 \frac{m-1-f(x_0,m)}{2} \\
& -\gamma S_c(m) u(u+\bar{u})(u-\bar{u})^2 \\
= \ & \gamma(u+\bar{u})(u-\bar{u})^2 \left(\frac{m-1-f(x_0,m)}{2} - S_c(m)u\right)
\end{aligned}$$

which is bounded above by zero since $m - 1 < f(x_0, m)$ for all $x_0 > 0$. Therefore $\dot{V}_1(u) = 0$ if and only if $u = \bar{u}$ and $\dot{V}_1(u) < 0$ otherwise. ∎

We will now prove the main result of the section which shows that the greedy dynamics (8) guarantees convergence to the N.E. We use a passivity-based analysis [12] and interpret the overall dynamics as a negative feedback interconnection of two systems (Figure 1). The top block is the greedy dynamics block where each malware updates its takeover rate, and the bottom block is the computation of the gradient ascent direction of the utilities.

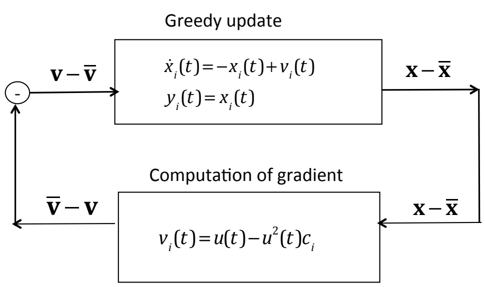


Fig. 1. Figure illustrating passivity approach for proving convergence to the N.E. The update dynamics of malwares is decomposed into two blocks, where the top block takes in the ascent direction v_i, and updates its takeover rate, and the bottom block computes the gradient from the updated x_i. Shortage of passivity of the bottom block is dissipated by the excess of passivity of the top block.

Theorem 5: The greedy dynamics (17) is output strictly passive from input $(\mathbf{v} - \bar{\mathbf{v}})$ to output $\mathbf{x} - \bar{\mathbf{x}}$.

Proof: Consider the storage function V_{top} given as

$$V_{top} = \frac{1}{2}\sum_{i=1}^{m}(x_i - \bar{x}_i)^2 + \gamma \int_{\bar{u}}^{u} (\sigma^2 - \bar{u}^2) d\sigma$$

Differentiating V_{top} with respect to time, we obtain

$$\begin{aligned}
& \dot{V}_{top} = \sum_i (x_i - \bar{x}_i)\dot{x}_i + (u^2 - \bar{u}^2)\dot{u} \\
= \ & \sum_i (v_i - \bar{v}_i)(x_i - \bar{x}_i) - \sum_i (x_i - \bar{x}_i)^2 + \gamma(u^2 - \bar{u}^2)\dot{u}
\end{aligned}$$

Moreover, from Lemma 3, we know that $\dot{V}_1(u) \leq 0$ except at $u = \bar{u}$. Therefore,

$$\dot{V}_{top} \leq (\mathbf{v} - \bar{\mathbf{v}})^T(\mathbf{x} - \bar{\mathbf{x}}) - P_{top}(\mathbf{x} - \bar{\mathbf{x}}) \tag{18}$$

where $P_{top}(\cdot) : R^m \to R_+$ is a positive definite function defined as

$$P_{top}(\mathbf{x} - \bar{\mathbf{x}}) = (\mathbf{x} - \bar{\mathbf{x}})^T(\mathbf{x} - \bar{\mathbf{x}}) - \dot{V}_1(u) \geq 0 \tag{19}$$

with $P_{top}(\mathbf{x} - \bar{\mathbf{x}}) = 0$ if and only if $\mathbf{x} = \bar{\mathbf{x}}$. ∎

Theorem 6: Computation of $v_i(\cdot)$ is input-feedforward passive from input $\mathbf{x} - \bar{\mathbf{x}}$ to output $\bar{\mathbf{v}} - \mathbf{v}$, i.e., there exists a function $P_{bot}(\mathbf{x} - \bar{\mathbf{x}})$ such that

$$(\mathbf{x} - \bar{\mathbf{x}})^T(\bar{\mathbf{v}} - \mathbf{v}) \geq P_{bot}(\mathbf{x} - \bar{\mathbf{x}}) \tag{20}$$

Proof: Define $P_{bot}(\mathbf{x} - \bar{\mathbf{x}})$ as

$$P_{bot}(\mathbf{x} - \bar{\mathbf{x}}) = -(u - \bar{u})^2 + \sum_i c_i(x_i - \bar{x}_i)(u^2 - \bar{u}^2)$$

since $u - \bar{u} = \mathbf{1}^T(\mathbf{x} - \bar{\mathbf{x}})$, P_{bot} is a function of $\mathbf{x} - \bar{\mathbf{x}}$. Expanding $(\mathbf{x} - \bar{\mathbf{x}})^T(\bar{\mathbf{v}} - \mathbf{v})$, we obtain

$$\begin{aligned}
& \sum_i (x_i - \bar{x}_i)(\bar{v}_i - v_i) = (x_i - \bar{x}_i)(\bar{u} - u + (u^2 - \bar{u}^2)c_i) \\
& = -(u - \bar{u})^2 + \sum_i c_i(x_i - \bar{x}_i)(u^2 - \bar{u}^2)
\end{aligned}$$

Therefore

$$\sum_i (x_i - \bar{x}_i)(\bar{v}_i - v_i) \geq -(u - \bar{u})^2 + \sum_i c_i(x_i - \bar{x}_i)(u^2 - \bar{u}^2)$$

thus satisfying the condition (20) with equality. ∎

The following theorem proves convergence to the N.E. The intuition is that any shortage of passivity from the computation of v_i is dissipated by the excess of passivity in the greedy dynamics.

Theorem 7: Consider the negative feedback interconnection of of the two dynamical systems corresponding to greedy update and computation of the gradient direction (Figure 1). The equilibrium of the closed loop system $\mathbf{x} = \bar{\mathbf{x}}$ is globally asymptotically stable.

Proof: Let γ be chosen such that

$$\gamma \geq \frac{K}{S_c(m)} \cdot \frac{m - 1 + f(x_0, m)}{-m + 1 + f(x_0, m)} \tag{21}$$

where K is given as

$$K = \frac{1}{4m}\left(m \sum_i c_i^2 - S_c(m)^2\right)$$

Using V_{top} as a candidate Lyapunov function, we obtain

$$\begin{aligned}
& \dot{V}_{top} \leq -P_{bot}(\mathbf{x} - \bar{\mathbf{x}}) - P_{top}(\mathbf{x} - \bar{\mathbf{x}}) \\
& = (u - \bar{u})^2 - \sum_i c_i(x_i - \bar{x}_i)(u^2 - \bar{u}^2) - \sum_i (x_i - \bar{x}_i)^2 \\
& + \gamma(u + \bar{u})(u - \bar{u})^2 \left(\frac{m - 1 - f(x_0, m)}{2} - S_c(m)u\right)
\end{aligned}$$

In order to obtain an upper bound on the above expression, we will first upper bound the term $(u - \bar{u})^2 - (u^2 - \bar{u}^2)\sum_i c_i(x_i - \bar{x}_i) - \sum_i (x_i - \bar{x}_i)^2$. To do so, we solve a convex optimization problem

$$\begin{array}{ll}
\min_{\mathbf{z}} & \sum_i z_i^2 + (u^2 - \bar{u}^2)\sum_i c_i(z_i) \\
\text{s.t.} & \sum_i z_i = u - \bar{u}
\end{array} \tag{22}$$

which is a quadratic program with equality constraint. Solving the optimization problem, and substituting $z_i^* = x_i - \bar{x}_i$, we obtain

$$(u-\bar{u})^2 - (u^2-\bar{u}^2)\sum_i c_i(x_i-\bar{x}_i) - \sum_i (x_i-\bar{x}_i)^2$$
$$\leq \frac{1}{4m}(m\sum_i c_i^2 - S_c^2)(u^2-\bar{u}^2)^2$$
$$+\frac{1}{m}(u-\bar{u})^2(m-1-S_c(m)(u+\bar{u})) \leq K(u^2-\bar{u}^2)^2$$

where the last inequality is due to the fact that

$$m-1-S_c(m)\bar{u} = \frac{1}{2}\left(m-1-f(x_0,m)\right) \leq 0$$

Therefore,

$$\dot{V}_{top} \leq -P_{bot}(\mathbf{x}-\bar{\mathbf{x}}) - P_{top}(\mathbf{x}-\bar{\mathbf{x}})$$
$$\leq K(u^2-\bar{u}^2)^2$$
$$+\gamma(u+\bar{u})(u-\bar{u})^2\left(\frac{m-1-f(x_0,m)}{2} - S_c(m)u\right)$$
$$= (u-\bar{u})^2(u+\bar{u})\left((K-\gamma S_c)u + K\bar{u} + \gamma\frac{m-1-f}{2}\right)$$
$$\leq (u-\bar{u})^2(u+\bar{u})\left((K-\gamma S_c)u\right) \leq 0$$

where the last two inequalities are from (21). ■

The results of this section imply that, for a given mitigation strategy modeled as the takeover rate x_0, the takeover rates of the malwares will converge to the Nash equilibrium. In the following section, we investigate selection of the optimal mitigation strategy.

VI. Optimal Mitigation Strategy

In this section, we derive an optimal mitigation strategy against competing malwares as the solution to a Stackelberg game. In the Stackelberg formulation, the system owner selects the takeover rate, x_0, and establishes this rate as a fixed policy. The malwares then select their takeover rates as the Nash equilibrium given the policy x_0, based on the dynamics of Section V. Since the sum of the takeover rates $\sum_{i=0}^n x_i$ will converge to $\bar{u}(x_0)$, a system owner will select x_0 to maximize the utility function

$$U_0(x_0) = \frac{x_0}{\bar{u}(x_0)} - c_0 x_0. \quad (23)$$

The following theorem shows that an efficient optimization algorithm can be constructed to maximize $U_0(x_0)$.

Theorem 8: Given the number of participating malwares at the N.E. m, $U_0(x_0)$ is a concave function in x_0.

Proof: It suffices to show the concavity of $\frac{x_0}{\bar{u}(x_0)}$ since $c_0 x_0$ is a linear function in x_0. From equation (6), we have

$$\frac{x_0}{\bar{u}(x_0)} = \frac{2S_c(m)x_0}{m-1+f(x_0,m)}$$

by rewriting

$$\frac{2S_c(m)}{m-1+f(x_0,m)} = \frac{-(m-1)+f(x_0,m)}{2x_0}$$

we obtain

$$\frac{x_0}{\bar{u}(x_0)} = \frac{1}{2}\left(-(m-1)+f(x_0,m)\right)$$

which is concave in x_0 since $f(x_0,m)$ is a composition of a concave function (square root) with an affine function of x_0. ■

Based on the theorem, the following optimization algorithm can be constructed by the system owner. The owner can divide the possible values of x_0 into a set of $n+1$ intervals $\{I_m(x_0)\}_{m=0}^n$ where

$$I_m(x_0) = \{x_0 : m \text{ malwares participate at N.E.}\}$$

These intervals will be disjoint since m is a monotone decreasing function in x_0. By Theorem 8, the owner can obtain $x_0^{(m)} \triangleq \arg\max\{U_0(x_0) : x_0 \in I_m\}$ for $m=1,\ldots,n$ by solving a convex optimization problem, and then select the optimal takeover rate as $x_0^* = \arg\max\{U_0(x_0^{(m)}) : m = 1,\ldots,n\}$.

We observe that, in order to solve the convex optimization problem for each interval I_m, the system owner needs to know the parameters m and $S_c(m)$. To estimate these parameters, the owner can choose an x_0 and observe the fraction of time $\frac{x_0}{\bar{u}(x_0)}$. Based on this information, the owner can construct an equation

$$\frac{x_0}{\bar{u}(x_0)} = S_c(m)\bar{u}(x_0) - (m-1)$$

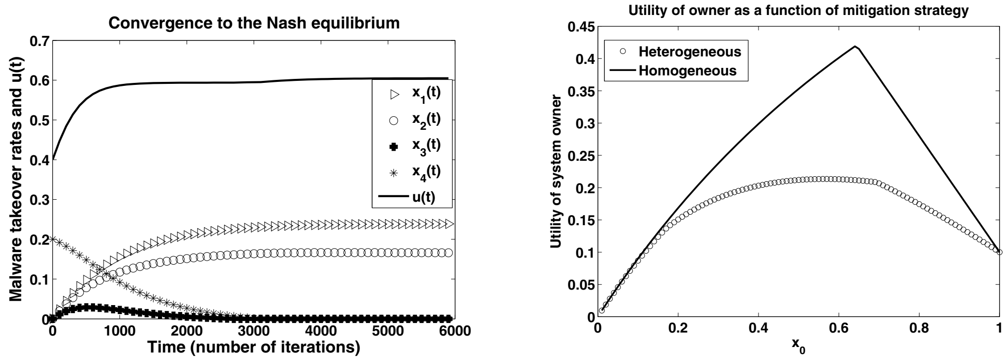
which is a linear equation in m and $S_c(m)$ given x_0 and $\bar{u}(x_0)$. Therefore, the owner can estimate the parameters by choosing two different x_0 values and constructing two linear independent equations based on the responses of the adversary.

VII. Numerical Study

We evaluated our approach using Matlab simulation. We conducted two numerical simulations. The first simulation verifies the convergence to the Nash equilibrium given the system owner's takeover rate x_0 and analyzes the malware dynamics prior to convergence. In the second simulation, we numerically evaluate the utility of the system owner in two cases. In the first case, multiple competing malwares have heterogeneous costs. In the second case, the competing malwares have equal costs of takeover and hence are homogeneous. The sum of the costs was equal in both cases.

Figure 2(a) shows the convergence to the N.E. of malwares given a fixed x_0. At the equilibrium, malwares 1, 2 and the system owner own the resource 0.4, 0.27 and 0.33 fraction of time respectively. Initial takeover rates of malwares were set as [0,0,0,0.2] with corresponding costs [1,1.2,1.7,1.72]. The system owner's takeover rate was fixed at $x_0 = 0.2$. Under these parameters, Theorem 2 implies that malwares 3 and 4 will drop out, and malwares 1 and 2 will converge to N.E with positive takeover rates. The numerical value of $\bar{\mathbf{x}}$ is consistent with values computed from Theorem 2.

We observe from Figure 2(a) that the proposed dynamics (8) ensures that malwares 1 and 2 participate in the game even when initial takeover rates were initialized to zero. Malware 3 initially increases its takeover rate, but eventually drops out as malwares 1 and 2 begin to converge to the N.E. It is also shown numerically that the dropouts are not



(a) (b)

Fig. 2. (a) Figure illustrating the convergence to the Nash equilibrium. Initial takeover rates of malwares were given as [0,0,0, 0.2] with corresponding costs [1,1.2, 1.7, 1.72]. System owner's takeover rate was fixed at $x_0 = 0.2$. (b) Figure illustrating the utility of the system owner when competing against heterogeneous malwares and equally powerful malwares. The cost of system owner was set to $c_0 = 0.9$. In the heterogeneous case, six malwares have corresponding costs [1,1.2,1.7,1.72, 1.8, 1.9]. In the homogeneous case, all malwares had equal cost $c = 1.553$ so that sum of the costs is equal in both cases.

necessarily sequential, i.e., malwares with higher costs do not drop out of the game first. Malware 4, given high initial takeover rate and highest cost c_4, does not drop out until after malware 3 drops out.

Figure 2(b) compares the utility of the system owner for heterogeneous and homogeneous malwares. In both cases, the simulations numerically verify that the owner's utility is a concave function in x_0, and the utility function becomes $1 - c_0 x_0$ after all malwares drop out of the takeover game. The maximum achieved utility is higher when competing against equally powerful malwares where all malwares either participate or dropout at the N.E. Letting $\hat{x}_0$ denote the minimum x_0 to make all malwares drop out in the homogeneous case, we observe that the malware with the lowest cost in the heterogeneous case will maintain a positive takeover rate when $x_0 = \hat{x}_0$. Hence, we observe that the utility of the system owner is mainly determined by the capabilities of the most powerful malware.

VIII. Conclusions

In this paper, we modeled the interaction between competing, adaptive malwares and a system owner by formulating a generalized version of the FlipIt game. We derived a closed form for the unique Nash equilibrium of the formulated game when the system owner and malwares employ exponential strategies for taking over the host with time-varying rates.

We modeled the adaptive nature of malwares as gradient ascent dynamics where each malware updates its takeover rate in order to maximize its utility. The proposed dynamics only requires the knowledge of either the fraction of time one owns the resource. Using a passivity-based approach, we proved that the greedy dynamics guarantee convergence to the Nash equilibrium. We derived an optimal mitigation strategy as a solution to a Stackelberg game, in which the system owner first commits its takeover rate x_0.

While our approach assumed that each malware can accurately observe the total takeover rate by observing the fraction of time one controls the host, in practice the observation capabilities may vary for different malwares, and may lead to incorrect estimation. Incorporating noisy observation and characterizing the deviation from the identified Nash equilibrium will be part of future work. In addition, for the case when the system owner's takeover rate $x_0 = 0$, the host takeover game between malwares reduces to a linear resource allocation game where regret minimization dynamics have been shown to guarantee convergence [13]. We will investigate whether learning dynamics can be applied to our Stackelberg-Nash game set-up.

References

[1] P. Gutmann, "The commercial malware industry," *DEFCON conference*, 2007.

[2] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," *Security & Privacy, IEEE*, vol. 9, no. 3, pp. 49–51, 2011.

[3] Z. Xu, J. Zhang, G. Gu, and Z. Lin, "Autovac: Automatically extracting system resource constraints and generating vaccines for malware immunization," IEEE, pp. 112–123, 2013.

[4] 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.

[5] M. Bloem, T. Alpcan, and T. Başar, "An optimal control approach to malware filtering," *Proceedings of the 46th IEEE Conference on Decision and Control*, pp. 6059–6064, 2007.

[6] M. H. R. Khouzani, S. Sarkar, and E. Altman, "Maximum damage malware attack in mobile wireless networks," *IEEE/ACM Transactions on Networking*, vol. 20, no. 5, pp. 1347–1360, 2012.

[7] S. Xu, W. Lu, and Z. Zhan, "A stochastic model of multivirus dynamics," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 1, pp. 30–45, 2012.

[8] T. Alpcan and T. Başar, *Network Security: A Decision and Game-theoretic Approach*. Cambridge University Press, 2010.

[9] 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.

[10] A. Laszka, G. Horvath, M. Felegyhazi, and L. Buttyán, "FlipThem: Modeling targeted attacks with FlipIt for multiple resources," in *Decision and Game Theory for Security*. Springer, 2014, pp. 175–194.

[11] D. Reitter, J. Grossklags, and A. Nochenson, "Risk-seeking in a continuous game of timing," *Proceedings of the 13th International Conference on Cognitive Modeling (ICCM)*, pp. 397–403, 2013.

[12] H. K. Khalil, *Nonlinear Systems*. Prentice Hall Upper Saddle River, 2002.

[13] E. Even-Dar, Y. Mansour, and U. Nadav, "On the convergence of regret minimization dynamics in concave games," *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pp. 523–532, 2009.