A Game Theoretical Method for Cost-Benefit Analysis of Malware Dissemination Prevention

Theodoros Spyridopoulos[a,*] · Konstantinos Maraslis[a] · Alexios Mylonas[b] · Theo Tryfonas[a] ·George Oikonomou[a]

[a] *Cryptography Group, Faculty of Engineering, University of Bristol,*
Merchant Venturers Building, Woodland Road, Clifton BS8 1UB, UK
{th.spyridopoulos, k.maraslis,theo.tryfonas,g.oikonomou}@bristol.ac.uk

[b] *Faculty of Computing, Engineering and Sciences, Staffordshire University*
Beaconside, Stafford, ST18 0AD, UK
alexios.mylonas@staffs.ac.uk

Abstract Literature in malware proliferation focuses on modelling and analysing its spread dynamics. Epidemiology models, which are inspired by the characteristics of biological disease spread in human populations, have been used in the past against this threat to analyse the way malware spreads in a network. This work presents a modified version of the commonly used epidemiology models SIR and SIS, which incorporates the ability to capture the relationships between nodes within a network, along with their effect on malware dissemination process. Drawing upon a model that illustrates the network's behaviour based on the attacker's and the defender's choices, we use game theory to compute optimal strategies for the defender to minimise the effect of malware spread, minimising at the same time the security cost. We consider three defence mechanisms, "patch", "removal", and "patch and removal", which correspond to the defender's strategy, used probabilistically with a certain rate. The attacker chooses the type of attack according to its effectiveness and cost. Through the interaction between the two opponents we infer the optimal strategy for both players, known as Nash Equilibrium, evaluating the related payoffs. Hence, our model provides a cost-benefit risk management framework for managing malware spread in computer networks.

Keywords Malware Proliferation · Game Theory · Epidemiology · Network Security · SIR · SIS

1 INTRODUCTION

Malicious software, known as malware, is a considerable threat to the realm of interconnected computer systems. Often built by cyber-criminals, malware aims to compromise target computers with the ultimate goal of stealing or corrupting sensitive data, gaining access to private systems or rendering cyberspace services unavailable. The impact of such malicious activities entail high financial consequences for both the targeted enterprises and their customers [5]. Different classes of malware exist, such as computer viruses, worms, trojan horses, keyloggers and many others [8]. With the ever growing importance of networked computing, malware that replicates itself in order to spread, known as a worm, has become one of the most efficient ways of wide-scale attacks.

Defence mechanisms such as firewalls and anti-viruses have been developed to defend against malicious software. Those mechanisms investigate the problem of malware at micro level by utilising experimental and heuristic findings, such as virus signatures, in order to prevent or detect and cure a computer's infection. Nevertheless, malware spread in a network of computers underlines the need for a network-level solution.

In light of these challenges several models that can describe the dynamics of malware proliferation over a computer network have been proposed [30]. Most of them base their function on epidemiological models. Such models have been used to represent and analyse the dynamics of a virus spread within a human population. Additionally, Game Theory has been introduced in a number of occasions across the fields of computer and network security (e.g. [22, 7]) in order to describe the interactions between an attacker and the defender and the ways their actions may affect each other. As malware acts based on inscribed behaviour coded by cyber criminals, approaching it as a threat agent on its own right under the premise of game theory becomes a reasonable assumption.

Our work aims to combine established epidemiology models with a game theoretic framework that captures the state of the system when both the defender and attacker use a variety of strategies to achieve their personal goal. We develop a game between the defender and malware's author, taking into account the spread dynamics, so that defenders manage to compute their optimal strategy by minimising the cost of security, on a cost-benefit basis.

The rest of the paper is structured as follows. Section 2 presents the basics of epidemiology models and game theory as it is applied in malware analysis. Special emphasis is given to the "FLIPIT" game. Section 3 introduces our model. The application of our approach to a case-study is presented in Section 4. Section 5 discusses related work. Finally, Section 6 discusses the conclusions drawn from this work and suggests ideas for further work.

2 BACKGROUND KNOWLEDGE

2.1 Mathematical Specification of Standard Epidemiology Models

This section presents the mathematical specification of the two commonly used epidemiology models (SIS and SIR) on which our model is also based. In general, such models are formulated over a fixed-size network. Nodes represent individuals and links or edges between nodes represent contacts between individuals. The infection spreads along direct links between nodes and the network is assumed to be symmetric, so that no preferential direction of the malware proliferation exists.

2.1.1 The SIR Model

In the SIR model [10, 9, 11], the total population is divided into three parts: i) susceptible nodes (denoted by S), ii) infected nodes (denoted by I) and iii) recovered nodes (denoted by R). The differential equations 1, 2 and 3 describe the rate of change of the susceptible nodes, infected nodes and recovered nodes respectively over time [4]. Here β denotes the infection rate (the rate at which an infected node infects other nodes within the network) and r denotes the recovery/immunisation rate (the rate at which we recover/patch infected nodes within the network). In our work a contact is considered as a network link between two nodes and as all nodes are connected to one another (directly or through a number of hops depending on the network's topology), they are always in contact with each other. Thus, the probability of a susceptible node to be infected does not depend on the nature of the network topology.

$$\frac{dS}{dt} = -\beta IS \quad (1)$$

$$\frac{dI}{dt} = \beta IS - rI \quad (2)$$

$$\frac{dR}{dt} = rI \quad (3)$$

2.1.2 The SIS Model

In the SIS model, the total population is divided in two parts, susceptible nodes (denoted by S) and infected nodes (denoted by I). Equations 4 and 5 model the rate of change of susceptible nodes and infected nodes respectively over time [17]. Again, β is the infection rate and this time γ is the recovery rate/disinfection. Even though the term "recovery rate" is used in both the SIR and the SIS model, it is used for different purposes. In the first case recovery rate refers to immunisation (the recovered node cannot be reinfected), while in the latter it refers to disinfection (the recovered node can be reinfected).

$$\frac{dS}{dt} = -\beta IS + \gamma I \quad (4)$$

$$\frac{dI}{dt} = \beta IS - \gamma I \quad (5)$$

2.2 Brief Introduction to Game Theory

Game theory provides us with a set of tools designed to analyse situations where two or more decision makers interact [20,2,3]. Decision makers are identified as unique players, their decisions as strategies and the formal description of the interaction between them is denoted as a game [27]. The basic assumption of game theory is that every player acts rationally, aiming at the best possible individual outcome, and take into account other players' decisions. When a player's strategy always leads to a better reward for that player compared to another strategy, no matter what the remaining players have decided, then we say that the first strategy dominates the second. Solution to a game is the description of the strategies that each player has to follow in order to achieve the best possible outcome. *Nash Equilibrium* of a game, is a state (combination of attacker's and defender's strategies) where each player has no incentive to unilaterally deviate from, because that would lead to a reduced reward. In other words, the attacker's strategy in a Nash Equilibrium is the best response to the defender's strategy that participates in the same Nash Equilibrium, and vice versa.

2.2.1 Game Theory in Security and Malware Analysis

Traditional network security mechanisms such as Intrusion Detection and/or Prevention Systems (IDS/IPS) analyse malware at a level of specific technical detail. They focus on collecting, dissecting and recording its structure and behaviour. This allows them to respond to attacks that are based on well-known techniques. For instance, IDS algorithms apply malware-signature identification or make use of heuristic algorithms to detect suspicious system behaviours that indicate a possible infection. Nevertheless, since they mostly rely on such experimental findings, they are insufficient against sophisticated attacks, which may utilise unknown techniques (e.g. zero-day attacks).

Traditional network security solutions lack a macro-level quantitative decision framework [22]. Various researchers have focused their work on utilising game theory in order to provide a holistic solution [15, 31, 12, 7]. The relationship between attacker and defender can be modelled as the interaction between two competing parts in a game theoretic scenario. The malware's goal is to spread widely, whereas the defender aims at protecting the network against the attack (i.e. minimising spread), whilst keeping costs as low as possible. Game theory can be used to examine and evaluate all possible scenarios given the outcomes of each player's strategy and return the best one.

The "FLIPIT" Game: To develop our game we first devised a cost-benefit model to help us compute the gain of each strategy. This cost-benefit model is based on another game theoretic model known as "The FLIPIT Game" [7]. Its authors have developed a model that describes the situation in which an attacker periodically takes over a system and is not immediately detected by the defender. The model includes two players (i.e. attacker and defender) and a shared resource. The two opponents compete to control the shared resource. The attacker tries to put the resource into a bad state (e.g. a compromise of a system or an asset, revelation of a secret), while the defender puts the resource into a good state (asset disinfection or implementation of a control). The objective of each player is to control the resource for the largest possible fraction of time and minimise at the same time their total cost. Players do not know the current situation of the game when other players make a move; they learn that only when they make a move. Making a move incurs cost and taking over control gains benefit. Each player loses some points per move and gains some points per second when he is in control.

The mathematical description of the game is provided below. Here we assume that the defender is *player 0* and the attacker is *player 1*. Player i makes $N_i(t)$ moves per second, pays k_i points per move and gains ($G_i(t)$) one point per second when the source is under their control.

The total period of time t is the time the resource is controlled by the defender plus the time controlled by the attacker as shown in Equation 6.

$$G_0(t) + G_1(t) = t \qquad (6)$$

Thus, for each player, the gain rate $\gamma_i(t)$ is equal to the fraction of time that player i has the shared resource under control, as shown in Equations 7 and 8.

$$\gamma_i = \frac{G_i(t)}{t} \qquad (7)$$

$$\gamma_0(t) + \gamma_1(t) = 1 \qquad (8)$$

Equation 9 calculates the benefit of a strategy, which is defined as the gain minus the total cost. The aim of each player is to maximise the value of benefit.

$$B_i(t) = G_i(t) - k_i \cdot N_i(t) \qquad (9)$$

The generic description of a shared resource taken under control by an attacker is suitable to describe the situation of a computer network under attack from a worm. In our work, we view the network as the shared resource, which both the attacker and defender try to take under control. However, the shared resource cannot be instantly fully taken over, since a worm spreads in a subset of the total population in each time step rather the whole population. Hence, only a fraction of the shared resource can be taken over by the attacker (§3.5 provides more details for our game).

3 PROPOSED MODEL

Worms have the ability to self-replicate and spread without human intervention in a network [23], resembling human viruses. They may utilise various proliferation mechanisms, depending on the way they scan the network to find their new targets. Our work focuses on the examination of random scanning worms, which select their target IP addresses randomly without any topological restrictions [18, 26]. For a random scanning worm the whole Internet is seen as a fully interconnected

network. Consequently, each node has the same probability to get infected. This type of malware has been widely used by cyber-criminals, as it is easy to deploy. However, such attacks have lower infection rates than other topology-oriented scanning methods, such as malware that spreads through email exchange or the social media. This holds true as, their randomly picked IP addresses might not be used by any device. It is important to note that this work does not focus on modelling the malware dissemination process itself, instead it utilises already known modelling techniques and combines them with game theory in order to propose optimal mitigation strategies for the defender.

In the human virus spread paradigm, a ``random scanning" virus would mean that individuals are always in contact with one another. As mentioned, the probability of an individual to get infected by an already infected node is the same for everyone within the population. In a network it means that an infected node can infect every other node in the network without topological restrictions, since all nodes are linked with one another either directly or indirectly.

There are three basic security mitigation practices against the random scanning worm dissemination: i) *Remove*, ii) *Patch* and iii) both *Patch and Remove*. Under the SIR and SIS models, a susceptible node can either be patched against the certain worm and become immune to it, or stay in the susceptible state. If a susceptible node is infected then it can either stay infected and consequently spread the worm, or it can use the removal tool (e.g. an antivirus) in order to remove the worm. However, the removal tool does not encompass immunisation functionality. Thus, when an infected node removes the worm it returns back to the susceptible state, where it can subsequently be reinfected. However, if an infected node uses both the remove tool and the patch against the worm then it moves to recovery state, where it is immune against the specific worm. For each of the three security strategies, we set up differential mathematical expressions as in SIR and SIS models, which describe the dynamics of the system.

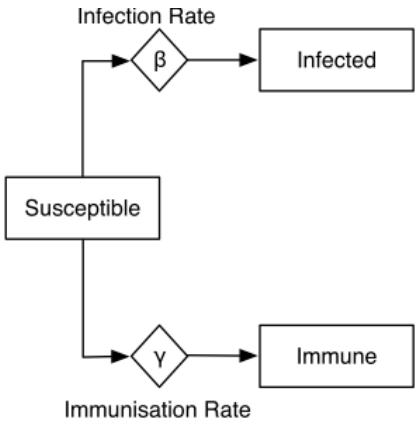

***Figure 1**: Patch Strategy Model - State transition*

3.1 Malware Proliferation with Patch Strategy

When the Patch Strategy is used, susceptible nodes become immune to the worm, but infected nodes cannot recover from the infection. In this case, the worm and the defender seem to take part in a race. If the worm spreads very fast, it will infect most computers in a short time before defenders notice it; if people in the network can patch their computers much faster than the worm's proliferation, the wide-range infection can be avoided. The model is depicted in Figure 1.

The mathematical specification of the Patch Strategy is given in Equations 10,11 and 12, where S is the susceptible population, I is the infected population and R is the immune population. β is the infection rate and γ is the immunisation rate.

$$\frac{dS}{dt} = -\beta IS - \gamma S \qquad (10)$$

$$\frac{dI}{dt} = \beta IS \qquad (11)$$

$$\frac{dR}{dt} = \gamma S \qquad (12)$$

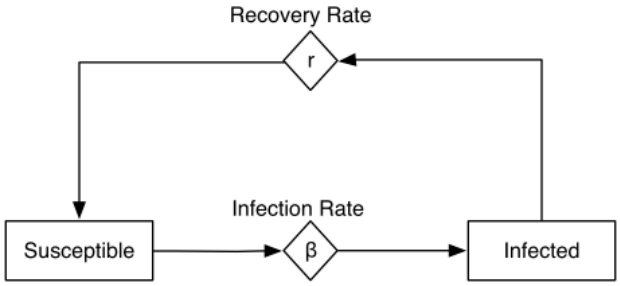

***Figure 2**: Removal Strategy Model - State transition*

3.2 Malware Proliferation with Removal Strategy

When the Removal Strategy is used, infected nodes can recover from the infection when the worm is detected and removed. However, nodes that have recovered from an infection are still susceptible to the specific worm, since no immunisation against it is included. In this case, the model is transformed into a SIS model in which the system reaches an equilibrium where the number of infected nodes and the number of susceptible nodes stay almost constant (Figure 2).

The mathematical specification of Removal Strategy is given in Equations 13 and 14. Again, S refers to the susceptible population and I refers to the infected population. β is the infection rate and r is the removal or recovery rate. As seen, no recovered population is found in the system.

$$\frac{dS}{dt} = -\beta IS + rI \qquad (13)$$

$$\frac{dI}{dt} = \beta IS - rI \qquad (14)$$

3.3 Malware Proliferation with Patch and Removal Strategy

The last strategy devised is the Patch and Removal. In this strategy both moves of patch and removal are available. A susceptible node can become immune to the worm when the patch is used. Furthermore, an infected node can recover from the infection if the worm is removed and then become immune to the worm by using the patch. This is the most efficient, yet costly, way to eliminate malware spread. Eventually, all nodes in the network will be immune against the specific worm. The strategy model is shown in Figure 3.

The differential equations that describe the dynamics of the model are shown in Equations 15, 16, 17 and 18. S refers to the susceptible population, I refers to the infected population, R is used for the recovered and immunised population and Q refers to the population that becomes immune to the malware. As before, β is the infection rate, γ refers to the immunisation rate when a susceptible node uses the specific patch and λ is the "removal and patch" rate.

$$\frac{dS}{dt} = -\beta IS - \gamma S \qquad (15)$$

$$\frac{dI}{dt} = \beta IS - \lambda I \qquad (16)$$

$$\frac{dR}{dt} = \lambda I \qquad (17)$$

$$\frac{dQ}{dt} = \gamma S + \lambda I \qquad (18)$$

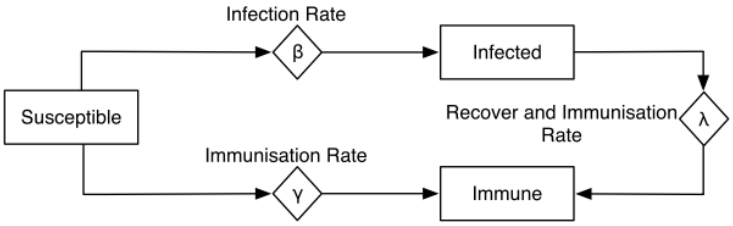

***Figure 3**: Patch and Removal Strategy Model - State transition*

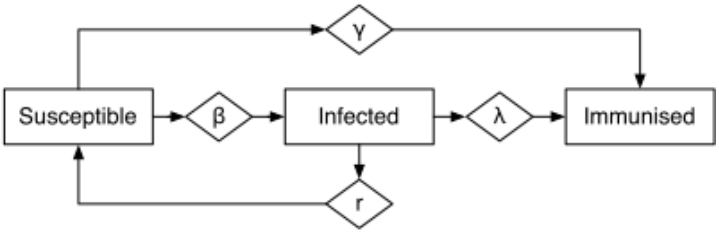

***Figure 4**: Unified Malware Dissemination Model - State transition*

3.4 Unified Malware Proliferation Model

By combining the aforementioned mitigation strategies we can construct a unified malware proliferation probabilistic model, where each of the strategies is chosen by the defender with a probability P, based on the patching and immunisation rates. In this model we have three states, the susceptible compartment, the infected population and the immunised. The state transitions of the model are depicted in Figure 4. A susceptible node can either be infected with infection rate β or immunised with immunisation rate γ. An infected node can either be disinfected and immunised with rate λ or just disinfected with rate r. Lastly, an immunised node cannot transit in any other state. The emerging dynamics are described by the differential Equations 19, 20 and 21.

By observing the model we can see that the defender can control the disinfection and immunisation rates (γ, λ, r), while the attacker controls the infection rate (β). These rates will form the strategies of the players in our game model.

$$\frac{dS}{dt} = rI - \beta IS - \gamma S \qquad (19)$$

$$\frac{dI}{dt} = \beta IS - \lambda I - rI \qquad (20)$$

$$\frac{dR}{dt} = \lambda I + \gamma S \qquad (21)$$

Game theory takes into account all the possible outcomes in order to find the optimal strategies, from which if either of the players deviates will always get less payoff. These strategies represent the Nash Equilibrium of the game. We compute all possible outcomes and find the Nash Equilibrium by the pair of strategies from which, if either players deviates will always get less payoff.

Figure 5 presents the state of the system for a specific configuration for both players (the attacker has chosen β = 1 node/hour and the defender r = 2 nodes/hour, γ = 1 node/hour, λ = 1 node/hour, N = 10^4 nodes and I_0 = 15 nodes initial infected population). It is evident that there is a very fast increase of the infected population. In fact, within less than 20 hours the malware has infected most of the population. However, after the first 20 hours a small gradual decrease of the infected population appears, followed by an increase in the immune population. This is mainly the result of the "disinfection and immunisation" strategy. Changing one of the parameters of the game can result in a whole new situation, as seen in Figure 6, where the defender has increased the "disinfection and immunisation" rate (λ) to 15 nodes/hour. In this scenario, the immunised population has increased significantly. At the same time, however, the cost for the defender increases, since higher security rate requires additional resources. As discussed earlier, the more security the defender applies the higher the cost she has to pay, however the lower becomes the impact of the attack. For instance, Figure 7 depicts the state of the system when the defender increases their immunisation rate to γ = 100

nodes/hour. The final state of the system is better than in Figure 6, since the final infected population is less. However, at the same time, this increases the defender's cost, since increasing the immunisation rate requires additional resources.

3.5 Game Theoretical Cost Benefit Analysis

In "FLIPIT" [7], two opponents compete to gain full control of a shared resource and gain is defined by the time the resource is under one's control. In our epidemiology model, the shared resource is the population of nodes in the network.

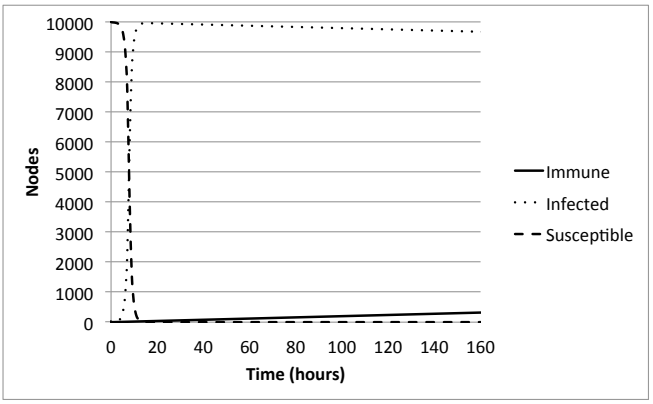

***Figure 5**: β = 1 node/hour, r = 2 nodes/hour, γ = 1 node/hour, λ = 1 node/hour, N = 10^4 nodes, I_0 = 15 nodes.*

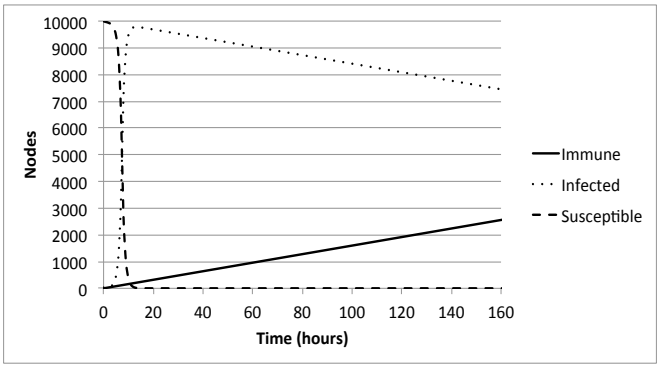

***Figure 6**: β = 1 node/hour, r = 2 nodes/hour, γ = 1 node/hour, λ = 15 nodes/hour, N = 10^4 nodes, I_0 = 15 nodes.*

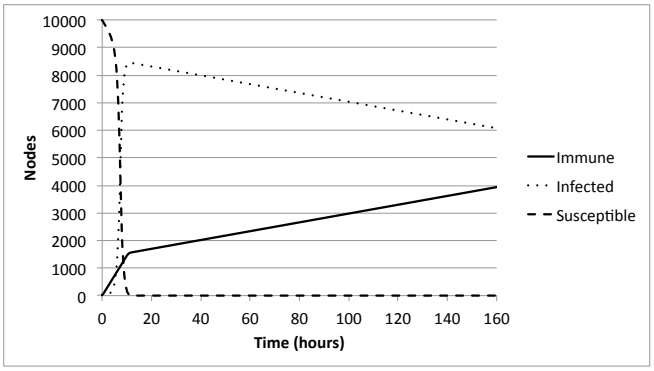

***Figure 7**: β = 1 node/hour, r = 2 nodes/hour, γ = 100 nodes/hour, λ = 15 node/hour, N = 10^4 nodes, I_0 = 15 nodes.*

Each time unit, the two opponents (attacker and defender) perform actions to take under their control a part of the population. The population under the attacker's control is denoted as the infected population and corresponds to the Infected (I) compartment in the unified malware proliferation model presented above. Therefore, the gain for the attacker when spreading malware in a network is represented by the I compartment of the model. On the other hand, if N is the initial population then $N - I$ is the

population under the defender's control. This population includes both the Immunised (R) and the Susceptible (S) states of the unified model. As the population in each compartment changes in time according to the dynamics described by the above equations, the total gain of each player is defined by the average fraction of node population under one's control. Therefore, by considering *player 0* as defender and *player 1* as attacker we define $G_i(t)$ the gain of player i and calculate it as shown in Equation 22, where $P_i(t)$ is the fraction of population under control by player i over time and t_k is the total time for which our model is running.

$$G_i(t_k)\frac{1}{t_k}\int_0^{t_k} P_i\,(t)dt \qquad (22)$$

As there are only two fractions of populations, one under the control of the defender and one under the control of the attacker, then $P_0(t) = 1-P_1(t)$. Hence: $G_0(t)+G_1(t) = 1$.

Before the game both players pick their strategies in order to optimise their outcome. The game is *non-cooperative*, *static*, *imperfect*, *complete information*, and *non-constant-sum* game. There is no cooperation between the players (network security games fall under the category of non-cooperative games, as there is no cooperation between attacker and defender [22]). As a *static* game, each player has a pre-computed move list (each move denoted as a strategy) from which the best move is chosen to maximise their personal benefit. Both players choose their strategies before the game in a one-shot fashion, not being able to change them during the game. It is an *imperfect* game as the two players choose their strategies simultaneously, without knowing the choices of the other players. However, they are aware of the opponent's available strategies and payoffs, therefore it is a *complete information game*. Finally, it is a *non-constant* game as the sum of the players' rewards is not always the same, for any combination of their strategies. In general, a pure Nash Equilibrium does not necessarily exist for this kind of games. Nevertheless, a pure Nash Equilibrium exists in our case study (see §4).

Accounting for all actions of all possible strategy combinations ($\beta, r, \gamma, \lambda$), the optimally defensive strategy can be identified, which returns the maximum possible gain under the minimum possible cost, regardless of attacker's chosen strategy.

3.5.1 Defining the Players' Strategies

In the beginning, both players choose their strategies, namely (γ, λ, r) for the defender and (β) for the attacker.

More specifically, *player 0* (defender) can manipulate the immunisation rate of two compartments: the susceptible population by immunising susceptible nodes before the spread of the worm (represented by γ in our unified model), and the infected population by disinfecting and then immunising the infected nodes (represented by λ in our unified model). Furthermore, the defender can disinfect infected nodes with disinfection rate r. Therefore, its strategy is defined by those three parameters in the unified malware proliferation model. Choosing them efficiently can increase the player's benefit. However, each of these actions costs, what is known as security cost. In this work, the cost of immunisations is considered higher (a more resource demanding operation) than the cost of disinfection, since it implies patching the vulnerability. In some scenarios, for instance patching an organization's mission critical host, it can become prohibitively costly.

On the other hand, *player 1* (attacker) has the ability to manipulate the infection rate (denoted by β in our model) of the malware, by choosing among different random scanning worms with different infection rates. The infection rate of each scanning worm may depend on the vulnerability it exploits and the randomisation mechanism it utilises. There-fore, the higher the infection rate, the higher is the software complexity of the malware and consequently the cost of deploying the attack increases. For that reason, the attacker aims to find the infection rate that will return the optimal payoff.

***Table 1**:Cost for Code-Red worm.*

Actions	Complexity			Total
	Low:1	Medium:2	High:3	
Exploit the buffer vulnerability		2		
Generate random IP addresses	1			4
Launch 99 threads with IP addresses	1			

3.5.2 Defining the Players' Payoffs

As mentioned both players' strategies bear some cost. For *player* 0, we define cost ($C_0(t)$) as the total number of moves made by *player* 0 ($n_0(t) = n_{0,1}(t) + n_{0,2}(t) + n_{0,3}(t)$, where $n_{0,1}(t)$, $n_{0,2}(t)$ and $n_{0,3}(t)$ correspond to the number of disinfections, immunisations and 'disinfections and immunisations' respectively), multiplied by each move's cost ($k_{0,j}$) (Equation 23). The move's cost is defined as the cost of disinfecting ($k_{0,1}$), immunising ($k_{0,2}$) or disinfecting and immunising ($k_{0,3}$)) a node.

$$C_0(t) = n_{0,1}(t) \cdot k_{0,1} + n_{0,2}(t) \cdot k_{0,2} + n_{0,3}(t) \cdot k_{0,3} \quad (23)$$

We define as cost for *player 1* the perceived complexity of the algorithm that the malware implements. The complexity of the algorithm is commensurate with the infection capabilities of the malware. Therefore, the higher the infection rate of the worm is, the higher is also the cost, k_1, that attacker has to pay in order to implement the malware.

$$C_1(t) = k_1 \quad (24)$$

Each player's payoff is equal to the player's total gain minus the related cost according to Equation 25.

$$B_i(t) = G_i(t) - C_i(t) \quad (25)$$

In order to compute costs, we utilise quantitative tables of operational complexity. A strategy by either player (e.g. Patch Strategy for the defender or Code-Red worm for the attacker) may encompass several actions, with each action characterised by a complexity level. For practical reasons, we set up empirically three levels of perceived complexity, low, medium and high, and assign a score to each of them, 1, 2 and 3, respectively. Therefore, the cost of a move for player 0 or the total cost of player 1 is equal to the sum of the costs of the actions it involves. An example is given in Tables 1 and 2 where we present the actions cost for the defender and the attacker when the latter uses the Code- Red worm (in which case the attacker has already chosen its strategy).

As in "FLIPIT", the gain is defined by fraction of population under each player's control. The population under the attacker's control corresponds to the infected population, while the population under the defender's control corresponds to the susceptible plus the infected population.

In order to find the defender's strategy that will return the optimal payoff, known as the Nash Equilibrium strategy, we construct the description of the game, which is a table with all possible payoffs for both players for all the available combinations of strategies. More details on how these strategies are calculated, can be found in Section 4.

4 Application of the Model - Case Study

In this section we apply our game theoretical malware proliferation model to a real case scenario, where the attacker can choose between five hypothetical worms with different infection rates (β). For the determination of the infection rates we used as a reference the Code-Red worm. According to [28], a node infected by this worm infects other nodes with a rate 1.62 nodes per hour, thus $\beta = 1.62$ nodes/hour. Albeit old, we have chosen Code-Red because it is a random-scanning worm with no topology constraints and, thus, its characteristics fit well into the generic nature of our abstraction. Its behaviour has also been thoroughly studied in the past [19,28,24].

Table 2: The cost of each move for the defender

Actions		Complexity			Total
		Low:1	Medium:2	High:3	
Patch	Detection		2		4
	Patch		2		
Removal	Detection		2		3
	Reboot	1			
Patch and Removal	Detection		2		5
	Reboot	1			
	Patch		2		

To provide the attacker with more options, we make the assumption that she can choose among five different types of worms, whose propagation rates are equal to integral multiples of Code-Red's propagation rate. As a result, the attacker can choose from five different worms and her available strategies are described as $\beta = k \cdot 1.62$, where $k = 1, 2, .., 7$.

Moreover, we assume that the defender can determine its strategy by choosing the immunisation rate ($0 \leq \gamma \leq 100$ immunisations per hour), the disinfection rate ($0 \leq r \leq 100$ disinfections per hour) or/and the combination of both disinfection and immunisation with rate $0 \leq \lambda \leq 100$ per hour. It has to be noted that in reality these rates can potentially get higher values, depending on the capabilities of the stakeholder and the criticality of the system to which the node population belongs. However, for the purposes of our experiment we limit all three rates from 0 to 100.

To simplify the scenario we predetermined that the cost for the attacker is equal to $\beta \cdot 1000$, implying that the propagation rate of the attack affect the complexity of the algorithm that implements the attack. We also make the assumption that the cost of a disinfection is 10 and the cost of an immunisation is 100. Table 3 forms the description of the game. Each cell within the table corresponds to a pair of payoffs, one for the attacker and one for the defender, for the specific pair of strategies. For instance, $PA_{1,1}$ corresponds to the attacker's payoff when the attacker chooses the strategy $\beta = 1.62$ and the defender chooses the strategy ($\gamma = 0$, $r = 0$, $\lambda = 0$). The defender's payoff for the same pair of strategies is $P_{D1,1}$

It is worth noting that a different selection of the parameters of this case study will obviously change the outcome of the game, without however changing the principles of the proposed model. These parameters are given here not as a reference, which falls outside the scope of this work, but to demonstrate the application of our unified model against malware proliferation. The parameters depend on each malware proliferation scenario, namely depend on the skills and goals of the attacker and the risk appetite of the defender (organization, individual).

For the simulation of the epidemiology model we utilised the Ventana Simulation Environment (Vensim). The simulations are based on Equations 19, 20 and 21, with total population of 10000 nodes, 15 of which were initially infected. For different values of β, γ, r and λ we run the model for $t_k = 168hours = 7days$. Vensim produces the data that are needed to set up the game. More particularly, it provides the infected, disinfected and immunised populations per unit time (in our case the software is set to run the simulations per hour). Therefore, the total number of disinfections ($n_{0,1}(tk)$), immunisations ($n_{0,2}(tk)$) and 'immunisations and disinfections' ($n_{0,3}(tk)$) in those 7 days can be found. Vensim also returns the infected ($P_1(t)$) and uninfected ($P_0(t)$, susceptible plus immunised) populations per unit time.

Based on the results from Vensim, the related costs and gains for both players can now be computed. For every combination of strategies (β, γ, r and λ), Vensim returns the $P_1(t)$ and $P_0(t)$ values, which are used to calculate the gain for each player according to Equation 22. As mentioned, it also returns the values $n_{0,1}(t_k)$, $n_{0,2}(t_k)$ and $n_{0,3}(t_k)$, which, in conjunction with the assumed cost of disinfection ($k_{0,1} = 10$) and cost of immunisation ($k_{0,2} = 100$ and therefore $k_{0,3} = 110$) and based on Equation 23, return the defender's cost. Based on Equation 24 and our assumptions about the attacker's cost, the attacker's cost for the different values of β is equal to $\beta \cdot 1000$. The gain of each player, as mentioned earlier, is equal to the mean fraction of population under each player's control. Thus, for every different combination of strategies we can now compute the related payoffs for both players according to Equation 25, populating Table 3. Figure 8 and Figure 9 present the defender's and the attacker's payoffs respectively, for the different values of β and r, when the defender preselects γ=1 and λ=2. We can observe how both players' payoffs - especially the defender's payoff - change depending on both player's strategies. By following the same process for every possible combination of β, r, γ and λ, we obtain a four-dimensional matrix, each dimension of which corresponds to either β, r, γ or λ. Each cell within the matrix corresponds to the respective payoff, as shown in Table 3. The purpose of the game now is to identify the optimal strategy for the defender that returns the best possible payoff regardless the attacker's actions.

Table 3: The description of the Game.

		Defender (γ,r,λ)				
		(0,0,0)	(1,0,0)	(2,0,0)	...	(100,100,100)
Attacker (β)	1.62	$PA_{1,1}$,$PD_{1,1}$	$PA_{1,2}$,$PD_{1,2}$	$PA_{1,3}$,$PD_{1,3}$	...	$PA_{1,10^6}$,$PD_{1,10^6}$
	3.24	$PA_{2,1}$,$PD_{2,1}$	$PA_{2,2}$,$PD_{2,2}$	$PA_{2,3}$,$PD_{2,3}$	...	$PA_{2,10^6}$,$PD_{2,10^6}$
	4.86	$PA_{3,1}$,$PD_{3,1}$	$PA_{3,2}$,$PD_{3,2}$	$PA_{3,3}$,$PD_{3,3}$	...	$PA_{3,10^6}$,$PD_{3,10^6}$
	6.48	$PA_{4,1}$,$PD_{4,1}$	$PA_{4,2}$,$PD_{4,2}$	$PA_{4,3}$,$PD_{4,3}$	...	$PA_{4,10^6}$,$PD_{4,10^6}$
	8.1	$PA_{5,1}$,$PD_{5,1}$	$PA_{5,2}$,$PD_{5,2}$	$PA_{5,3}$,$PD_{5,3}$	...	$PA_{5,10^6}$,$PD_{5,10^6}$

To solve the game the Lemke-Howson algorithm was used, which returns Nash Equilibria for two-player non-zero- sum games [14, 25, 29]. The algorithm (implemented in Matlab) takes Table 3 as input and returns the Nash Equilibria of the game. The results revealed a unique pure Nash Equilibrium that corresponds to the optimal strategy for defender, represented by the values γ = 10, r = 100, λ = 10 with *payoff* = -215.0926. Our experiment suggests that even though the proactive immunisation should be preferred to the other two actions for security reasons, it does not get the maximum value. In fact, the game results in a Nash Equilibrium where the disinfection rate (r) is larger than the immunisation rates, meaning that security costs have changed the optimal solution for the defender. On the other hand, the attacker's optimal strategy is to choose β = 11.34 infections/hour, which is the maximum infection rate in the table. This happens due to the fact that in this particular experiment the attacker's gain is much higher than the cost of his strategy and, therefore, he will always get larger payoff by choosing the worm with the highest infection rate. If the cost of attack is much higher (for instance in case the attacker chooses a zero-day attack), the resulted Nash Equilibrium may differ.

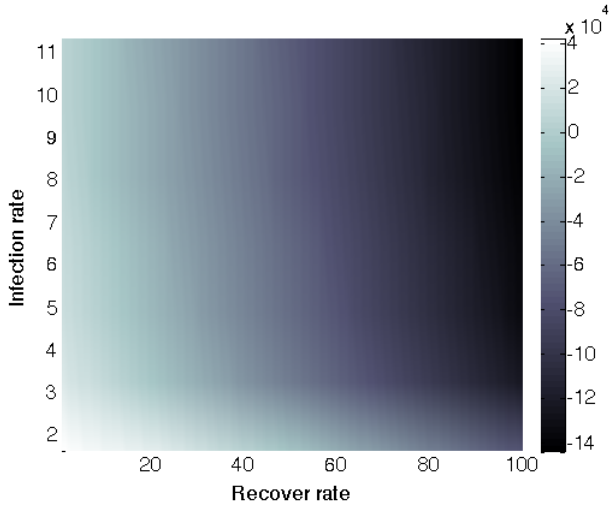

Figure 8: *Defender's payoff depending on the infection rate (β) and recover/disinfection rate (r). The defender has already chosen γ=1 and λ=2.*

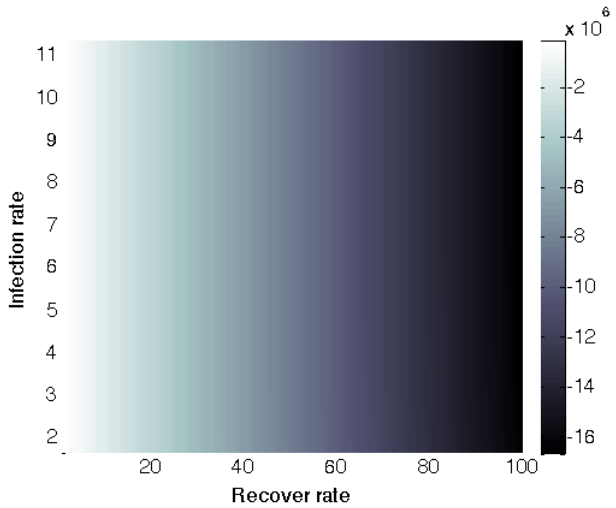

Figure 9: *Attacker's payoff depending on the infection rate (β) and recover/disinfection rate (r). The defender has already chosen γ=1 and λ=2.*

5 Related Work

The way that viruses and worms spread in a computer network shares common characteristics with the proliferation of biological diseases in human populations. Therefore, the analysis of malware can benefit from investigating the behaviour of biological diseases. Two types of models for analysing malware proliferation in epidemiology exist, namely stochastic and deterministic models. Stochastic models (e.g. [32]) are used to analyse small-scale networks, while deterministic models are mainly used to analyse large-scale networks [1]. Our work focuses on malware spread in a large computer network, thus we utilise deterministic models.

The majority of the deterministic epidemiology models are continuous-time models [30], since they offer higher precision when representing the emerging dynamics compared to discrete-time models. They divide the computer population of a network, known as *node population*, in discrete compartments, such as "Susceptible" and "Infected", and model the emerging dynamics between those compartments utilising differential equations. Individuals in the epidemic population may have several states, including *susceptible*, *infected* and *recovered*. The differential equations utilised to model the transitions between those states form the mathematical description of each model.

Two models have been widely used in the field of epidemiology modelling: a) the Susceptible-Infected- Recovered (SIR), by Kermack and McKendrick [10, 9,11] and b) a modified version of it, known as the Susceptible-Infected- Susceptible (SIS) model [21]. Both models assume that all individuals within a closed population (i.e. no births and deaths) are susceptible to the malware in the initial phase and an individual may go through each state sequentially. In the SIS model, the state transitions of an individual form a circulation. The individual may recover from the infection, but there is still a chance to be reinfected. In other words, an individual node becomes again susceptible to the malware after its recovery. In the SIR model, the final state is described as the recovered state.

An infected individual can recover from the infection and become immune to the malware. An immunised individual cannot be reinfected by the same malware. However, neither SIR nor SIS can individually represent reality accurately; the SIR model lacks the option of returning an infected node into the susceptible pool, while the SIS model lacks the ability of immunisation after recovery.

The authors of [13] proposed a modified version of the SIR model by introducing the notion of "temporary immunisation". Their model consists of three compartments, Susceptible, Infected, Temporarily Recovered (SIRS). Individuals transit from the susceptible state to infected, from infected to temporary recovered and then back to susceptible. In reality this model introduces a delay in the traditional SIS model, since the infected individuals that recover return to the susceptible state after an amount of time. This amount of time is defined by the rate at which removals lose their immunisation and become susceptible again. Resusceptibility represents the situation where a computer infected by malware recovers from the infection and becomes immune, remaining however susceptible to modified versions of the same malware. Even though this model is more accurate than the traditional SIR and SIS models, it still lacks the ability to encompass situations where the individual becomes immune to the malware before getting infected. Furthermore, even though it takes into account the fact that a malware may appear in different versions, it does not clarify whether each version exploits the same vulnerability, in which case patching this vulnerability would immunise the computer against any variation of the same malware.

A similar approach is followed by Mishra and Pandey [16], who proposed a dynamic discrete compartmental model. In their work, they mathematically formulated a four-state model encompassing the population compartments of Susceptible, Exposed, Infectious and Susceptible with Vaccination (SEIS-V). This model adds one more state in the traditional SIS model, the Exposed state. By introducing this state the authors denoted that not every susceptible individual is exposed. When a susceptible individual is exposed and comes in contact with an infected node then he also gets infected. However, following the SIS paradigm an infected individual can recover and transit to the susceptible state. Another additional state is the Vaccinated state, where a susceptible node can be "vaccinated" and therefore immunised against a specific malware. Nevertheless, as in the work of [13], an immunised node can become susceptible again after a certain amount of time, and as before, the authors do not take into account the mechanism of vulnerability patching. Furthermore, the exposed state is meaningless when modelling the spread of a random scanning malware in a fully-connected network such as the Internet, where each individual within the susceptible population has the same probability of getting infected by an infectious node.

Chen et al. [6] focus on modelling the spread of topological scanning malware. This type of malware spreads based on topology information. Therefore, the connectivity of each node plays a significant role in the malware propagation within the network, directly affecting the rate of infection. Unlike the previous model, it can also be used to model random scanning malware. Nevertheless, as mentioned by the authors, this model does not take into account patching and therefore there is no transition from susceptible to immunised.

Typically, disease spreading depends on common shared characteristics of the individuals in a population. In a network of computers, malware exploits certain vulnerabilities in the system in order to infect a host [33]. Common practice of malware is to exploit vulnerabilities in software that is installed in the victim-host. Thus, in order for a host to be considered as susceptible to a certain piece of malware, it must have installed the specific software version that bears the vulnerability that the malware can exploit. Otherwise, it cannot be infected and thus cannot be considered as susceptible. In the real world, not every host in a network carries the same vulnerabilities, forming therefore a heterogeneous computer network. This heterogeneity can be considered as an additional compartment of immune nodes. Our work has also taken into account the transition to this compartment from the susceptible or recovered state through the application of patching.

6 Conclusions

In this paper, we have integrated game theory premises with virus proliferation models to develop a cost-benefit approach to evaluate defence strategies that mitigate malware proliferation. We demonstrate the application of our approach, which is based on the combination of game theory and established epidemiology models, with a case study focusing on minimising the effect of random scanning worms (such as Code-Red worm) infecting a network of susceptible hosts. In the scenario both the defender and the attacker can choose among a variety of strategies in order to achieve their goals. The results of the case study highlight that the cost of security restricts the security level of the defender, since the resulted optimal strategy does not correspond to the most secure one. An interesting extension of this work could be the introduction of a security level threshold in the game eliminating the strategies that correspond to gains that do not meet the defender's requirements. In addition, applying the model against other worms and other strategies should produce different, but interesting results, but this falls outside the scope of this paper.

Game theory, under traditional malware proliferation approaches makes those models a useful tool for the efficient and effective protection of networks. We have identified the need to incorporate topology oriented malware, such as malware that spread through social media or emails, that spread more efficiently within networks. The logic behind this will be the same, a topology oriented malware dissemination model will feed our game with the necessary parameters, while the latter will return the optimal defence strategies.

Acknowledgments

The authors would like to express their appreciation to the anonymous reviewers for their valuable comments and suggestions.

References

1. Andersson, H., Britton, T.: Stochastic epidemic models and their statistical analysis, vol. 4. Springer (2000)
2. Antonopoulos, A., Verikoukis, C.: Multi-player game theoretic MAC strategies for energy efficient data dissemination. IEEE Transactions on Wireless Communications 13(2), 592-603 (2014)
3. Bousia, A., Kartsakli, E., Antonopoulos, A., Alonso, L., Verikoukis, C: Game theoretic approach for switching off base stations in multi-operator environments. In: Proceedings of 2013 IEEE International Conference on Communications (ICC), (pp. 4420-4424). IEEE (2013)
4. Capasso, V., Serio, G.: A generalization of the kermack- mckendrick deterministic epidemic model. Mathematical Bio- sciences 42(12), 43–61 (1978)
5. Cashell, B., Jackson, W.D., Jickling, M., Webel, B.: The economic impact of cyber-attacks (2004)
6. Chen, Z., Ji, C.: Spatial-temporal modeling of malware propagation in networks. IEEE Transactions on Neural Networks 16(5), 1291–1303 (2005)
7. Dijk, M.v., Juels, A., Oprea, A., Rivest, R.L.: FlipIt: the game of "Stealthy Takeover". Cryptology ePrint Archive, Report 2012/103 (2012). URL http://eprint.iacr.org/2012/103
8. Egele, M., Scholte, T., Kirda, E., Kruegel, C.: A survey on automated dynamic malware-analysis techniques and tools. ACM Computing Surveys (CSUR) 44(2), 6 (2012)
9. Kermack, W., McKendrick, A.: Contributions to the mathematical theory of epidemics. ii. the problem of endemicity. Proceedings of the Royal Society of London. Series A 138(834), 55–83 (1932)
10. Kermack, W.O., McKendrick, A.G.: A contribution to the mathematical theory of epidemics. In: Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, pp. 700–721. The Royal Society (1927)
11. Kermack, W.O., McKendrick, A.G.: Contributions to the mathematical theory of epidemics. III. further studies of the problem of endemicity. Proceedings of the Royal Society of London. Series A 141(843), 94–122 (1933). DOI 10.1098/rspa.1933.0106. URL http://rspa.royalsocietypublishing.org/content/141/843/94
12. Khouzani, M., Sarkar, S., Altman, E.: A dynamic game solution to malware attack. In: INFOCOM, 2011 Proceedings IEEE, pp. 2138–2146. IEEE (2011)
13. Kim, J., Radhakrishnan, S., Dhall, S.K.: Measurement and analysis of worm propagation on internet network topology. In: 13th International Conference on Computer Communications and Networks, 2004. ICCCN 2004. Proceedings., pp. 495–500. IEEE (2004)
14. Lemke, C.E., Howson Jr, J.T.: Equilibrium points of bimatrix games. Journal of the Society for Industrial & Applied Mathematics 12(2), 413–423 (1964)
15. Lin, J.C., Chen, J.M., Chen, C.C., Chien, Y.S.: A game theoretic approach to decision and analysis in strategies of attack and defense. In: Proceedings of the 2009 Third IEEE International Conference on Secure Software Integration and Reliability Improvement, pp. 75–81. IEEE Computer Society (2009)
16. Mishra, B.K., Pandey, S.K.: Dynamic model of worm propagation in computer network. Applied Mathematical Modelling 38(7), 2173–2179 (2014)
17. Van der Molen, H.: Math on malware. ISACA Journal 3, 40–47 (2011)
18. Moore, D., Paxson, V., Savage, S., Shannon, C., Staniford, S., Weaver, N.: Inside the slammer worm. IEEE Security & Privacy 1(4), 33–39 (2003)

19. Moore, D., Shannon, C., Brown, J.: Code-Red: a case study on the spread and victims of an Internet worm. In: Internet Measurement Workshop (IMW) 2002, pp. 273–284. ACM SIGCOMM/USENIX Internet Measurement Workshop, Marseille, France (2002)
20. Osborne, M.J., Rubinstein, A.: A course in game theory. MIT Press, Cambridge, Mass. (1996)
21. Pastor-Satorras, R., Vespignani, A.: Epidemic spreading in scale- free networks. Physical review letters 86(14), 3200–3203 (2001)
22. Roy, S., Ellis, C., Shiva, S., Dasgupta, D., Shandilya, V., Wu, Q.: A survey of game theory as applied to network security. In: 2010 43rd Hawaii International Conference on System Sciences (HICSS), pp. 1 –10 (2010). DOI 10.1109/HICSS.2010.35
23. Saudi, M., Tamil, E., Nor, S., Idris, M., Seman, K.: Edowa worm classification. In: Proceedings of the World Congress on Engineering, vol. 1 (2008)
24. Shannon, C., Moore, D.: The spread of the witty worm. Security & Privacy, IEEE 2(4), 46–50 (2004)
25. Shapley, L.S.: A note on the Lemke-Howson algorithm. Springer (1974)
26. Staniford, S., Paxson, V., Weaver, N., et al.: How to own the internet in your spare time. In: USENIX Security Symposium, pp. 149–167 (2002)
27. Turocy, T., Bernhard, v.S.: Texas a&m university. London School of Economics "Game Theory" CDAM Research Report Oct (2001)
28. Vojnovic, M., Ganesh, A.: On the race of worms, alerts, and patches. IEEE/ACM Transactions on Networking (TON) 16(5), 1066–1079 (2008)
29. Von Stengel, B.: Computing equilibria for two-person games. Handbook of game theory with economic applications 3, 1723– 1759 (2002)
30. Wang, Y., Wen, S., Xiang, Y., Zhou, W.: Modeling the propagation of worms in networks: A survey. Communications Surveys &Tutorials, IEEE 16(2), 942–960 (2014)
31. Wu, Q., Shiva, S., Roy, S., Ellis, C., Datla, V.: On modeling and simulation of game theory-based defense mechanisms against DoS and DDoS attacks. In: Proceedings of the 2010 Spring Simulation Multiconference, SpringSim '10, p. 159:1159:8. Society for Computer Simulation International, San Diego, CA, USA (2010)
32. Zonouz, S., Khurana, H., Sanders, W. H., & Yardley, T. M.: RRE: A game-theoretic intrusion response and recovery engine. IEEE Transactions on Parallel and Distributed Systems 25(2), 395-406 (2014)
33. Zou, C.C., Gong, W., Towsley, D.: Code red worm propagation modeling and analysis. In: Proceedings of the 9th ACM conference on Computer and communications security, pp. 138–147. ACM (2002)