On the Risk-Based Operation of Mobile Attacks in Wireless Ad Hoc Networks

Vasileios Karyotis∗, Symeon Papavassiliou† and Mary Grammatikou‡

Network Management and Optimal Design Lab (NETMODE)
School of Electrical and Computer Engineering
National Technical University of Athens, Zografou, 157 80, Greece
Emails:∗vassilis@netmode.ntua.gr, †papavass@mail.ntua.gr, ‡mary@netmode.ntua.gr

Abstract—In this paper we study the propagation of malicious software in wireless ad hoc networks under a probabilistic framework. We design topology control algorithms for the development of effective attack strategies by a malicious mobile node, based on the risk function metric, which indicates the network’s vulnerability. Our approach takes on the attacker’s perspective, in order to investigate the extent of its attack potentials, which in turn could be used for the effective design of network countermeasures. Our performance evaluation results demonstrate that the proposed risk-based topology control algorithms and respective attack strategies effectively balance the tradeoffs between the potential network damage and the attacker’s lifetime, and as a result significantly outperform any other flat and threshold-based approaches.

I. INTRODUCTION

Wireless ad hoc communications have been used extensively for the seamless provisioning of information exchange, where the deployment of infrastructure is difficult, if not impossible. Such cases include remote rural areas with stringent topographical profiles, disaster-recovery terrains, battlefields and popular event sites (i.e. sports stadiums, exhibition venues). In any case, the transmission of information needs to be performed efficiently and securely, so that it is received uncorrupted by the intended recipients.

Security issues concerning data and communication networks have recently become very popular, as major wired network infrastructures have received serious attacks [1]. Similar attacks have been launched against wireless devices through the bluetooth protocol [2]. In most cases the attacks were restricting the operational capabilities of the network nodes and they were using these affected nodes to increase the attack propagation rate and effectiveness.

The propagation process of malicious software in wireless ad hoc networks cannot be modeled deterministically, due to the randomness of the node locations and the variety of the channel quality. A probabilistic modeling approach is more suitable to the dynamic character of the ad hoc structure and the stochastic nature of the propagation process [3], [4].

In this paper, based on a probabilistic framework for the modeling of malicious software (malware), we study the behavior and effectiveness of a malicious node attacking an ad hoc network and the impact of topology in the propagation of malware. We propose a topology-dependent indicator of a node’s vulnerability to become infected, the risk, and use a relevant measure, the risk function, to design topology control algorithms for the development of effective attack strategies. The risk function depends solely on the availability of local information, allowing the malicious node to manage efficiently its available resources. Through analysis and simulation we evaluate the algorithms’ operation according to an offline metric that indicates the infection efficiency and characterizes the overall performance of an attack strategy.

The rest of the paper is organized as follows. In section II the assumed system model is described, while in section III the concept of risk and the risk function of a node are introduced and defined. In section IV we describe the probabilistic infection model and in section V we present the proposed topology control algorithms. Section VI contains some comparative performance evaluation results and relevant discussions, while section VII concludes the paper.

II. SYSTEM MODEL

An ad hoc network is usually represented by a network graph. Two wireless nodes are considered connected when they are able to exchange messages reliably. Thus, mobility and transmission power significantly affect the connectivity of an ad hoc network. Another major factor that contributes to the randomness of the network is the channel quality variation. Throughout our study we assume that the received signal has an inverse power dependence (path loss constant, α) of the distance from the transmitting node.

Therefore, the topology of an ad hoc network can be modeled by a random geometric graph, assuming node locations are determined by their respective coordinates in the geographical deployment region. Consequently, two nodes are connected with probability one if both of them are within transmission radius of each other, and with zero probability in all other cases. The edge set of the induced graph is completely determined by the relative positions of the nodes and their transmission radii.

In our model we assume a single malicious, mobile node with the capability of adjusting its transmission radius, r. The nodes of the underlying network are assumed to be static, having a common transmission radius R. Furthermore, it is possible that these nodes can infect their neighbors, once they become infected themselves, thus propagating potential attacks throughout the network. This behavior models the cases, where
the attacker takes control of the exploited machines and uses some of their modules at its own interest, [5].

Ad hoc and sensor networks have very limited energy resources. A wireless node needs to consume its energy reserves according to a sophisticated discipline, so as to extend as much as possible its lifetime without sacrificing its operational characteristics. Since in this paper the emphasis is placed on the attacker’s behavior and impact, in order to better evaluate such a tradeoff, we assume the mobile attacker to have limited energy resources, while the network nodes have infinite reserves.

The resulting network topology has the form shown in Fig. 1. There exists a multihop underlying network with nodes having transmission radius $R$, where a mobile attacker moves around the deployment region, varying its transmission radius $r$ and thus adapting its attack strategy. The cardinality of the network node set is denoted by $N$ and the total area of the network by $A$. Then, assuming the network nodes are uniformly distributed over this area, the density of the network (excluding the attacker) becomes $N/A$.

III. RISK FUNCTION AND TOPOLOGICAL CHARACTERISTICS

Different attacks follow different infection processes and their results vary significantly, depending on their target groups, mechanisms, ultimate objectives and current advances in technology. General means for characterizing and evaluating the propagation of malware are necessary for proper design of efficient countermeasures. In this section, we introduce the concept of the risk of a node, while in the following, based on this concept and properly defined measures, we introduce topology control algorithms that can be used for designing effective attack strategies. Furthermore, in this section an offline metric is described, which can be used for the evaluation of the efficiency of various types of attack strategies.

As the mobile attacker changes locations, its neighborhood varies, depending on the local topological properties of the network graph and the applied mobility model. Specifically, the node degree of the attacker changes non-deterministically, due to the stochastic nature of its movement and the locations of its neighbors. So do the degrees of the network nodes that lie in the 1-hop neighborhood of the malicious node (defined by the disk of radius $r$ centered at the attacker) as this area changes. Thus, the attacker’s node degree, $K_a$, is a random variable, over the sample space of positive integers. Theoretically, this sample space is infinite, since the geometric random graph representation allows a node to have infinite neighbors over a planar region. However, practical limitations on the wireless nodes’ dimensions, force $K_a$ to assume finite only values. As the network nodes are uniformly and randomly deployed over the targeted terrain, $K_a \in [0, K_{a_{max}}]$, where $K_{a_{max}}$ is the maximum degree value, determined by the density of the network nodes.

In graph theoretic terms, the degree of a node is indicative of the topological properties of its neighborhood. In particular, higher node degree means higher node density in the locality (1-hop neighborhood) of the specific node. In terms of malware propagation in an ad hoc network, higher node density allows for greater potentials, from an attacker’s perspective, to spread successfully malicious software. Furthermore, nodes belonging in network regions of high density have greater risk to become infected, either from the attacker, or from neighbors that have already become infected.

We define the risk of a node in connection with its degree to capture the potential of infection in its region. We note that in this work, we follow the attacker’s perspective, in order to realize the potentials for causing network damage, which in turn can be taken into account in designing efficient network countermeasures.

With respect to the attacker, $K_a$ would be the risk to spread malicious software to its neighbors, while the degree, $K$, of a network node indicates its own risk. According to the previous discussion, high risk is undesirable from the network’s viewpoint, whereas it is highly preferable by the attacker.

As explained before, the attacker’s risk $K_a$ is a random variable. Thus, $K_a$ is completely defined by its probability density function or the complete set of higher order moments. The stochastic movement of the attacker (or a network node in the case of full network mobility) prohibits the a priori complete knowledge of the probability density function of its risk. Moreover, full knowledge of the higher order moments is computationally inefficient for fast decision making purposes. For these reasons, we define the Risk Function $R_F$, by using only the first two orders of moments, as follows

$$R_F = E[|K_a|] + cVar(K_a), c > 0$$  

as a measure of an attacker’s risk, where $E[\cdot]$ denotes the expectation of a random variable, $Var(\cdot)$ its variance and $c$ represents a constant. In the following, without loss of generality, we consider $c = 1$.

We have added extra sophistication to the attacker, by allowing it to adjust its transmission radius according to the risk function. As the risk function provides additional information about the topological nature of the underlying network, the malicious node can devise topology control algorithms and adapt its strategy accordingly to utilize its available resources in a more effective way. In section V we describe three topology control algorithms (two of them are based on the
risk function) that can be used by the malicious node in the process of infecting a network more effectively.

The risk-based topology control algorithms provide the means to properly adjust the attacker’s transmission radius according to the measured topological characteristics (expressed by the risk function). As a result these algorithms attempt to dynamically balance the tradeoff between the number of infected nodes and the attacker’s lifetime. However, the local operation of the risk-based algorithms does not necessarily indicate how effective these algorithms are in terms of the overall network damage.

In order to evaluate the efficiency of the various topology control algorithms and corresponding attack strategies, we use a general attack-evaluation metric that captures the overall impact caused by the attacker to the network. Since unavailability of a percentage of the network for a time period causes reduced network operation, a combined metric for the number of infected nodes over a time interval is more suitable for the overall performance characterization. In this paper we use the Infection Efficiency, $I_E$, of an attacker, which is defined as the integral of the time function of the number of infected network nodes [3], to characterize the overall performance of an attack strategy. Intuitively, $I_E$ is the product of the number of infected network nodes in a time interval by the duration of the corresponding interval, and therefore provides a combined measure of the instantaneous damage (i.e. absolute value of the number of infected nodes) and the corresponding interval that the damage takes place.

IV. ATTACK MODELING

Without harming the validity of the analysis, we focus on the infection of a node itself and not on the process or type of infection. According to this, a recovering node might become infected again throughout the attacker’s lifetime. We are not interested in the particular infection mechanism (absence of immunization, new threats), but in the event that a node might become re-infected at some point of time. Within this context, we consider the malicious node to have the means to infect again already recovered nodes. This also, justifies the fact that a malicious node usually has a higher probability to infect other nodes compared to the infection capability of already infected network nodes.

The state of a node is considered to be binary, namely ‘infected’ or ‘non-infected’. Then the system state is given by a binary vector whose components correspond to the network nodes and the value of each row indicates the current node state. For each network node, there exists a link infection probability to become infected from either the attacker or an already infected neighbor, if the node’s state is ‘non-infected’. As the node has more than one neighbor, the probability of infection increases with the number of links. Thus, the number of infections that a node receives from a single link is a random variable. We assume this variable to be distributed according to a Poisson process. As various nodes follow different recovery procedures, we model the recovery process so that successive recoveries take place in exponential intervals.

The designated node model can be mapped to that of an $M/M/1$ queue, where an infection corresponds to an arrival that requires service, and the recovery to the service itself. Both arrivals and departures are exponentially distributed. Since the attacker is allowed to move throughout the network, infecting the rest of the nodes and depleting its energy, the network will recover completely in finite time after the mobile attack-node exhausts its energy.

There are two types of events taking place: infection of a node and recovery of an infected node. If $m$ denotes the total number of non-infected nodes, then in a network of $N$ nodes and one attacker, $N - m$ nodes of the network are infected (excluding the attacker). Furthermore, let $m'$ denote the number of nodes that are non-infected at a given instant and have the malicious node as a neighbor. Assuming an event has just taken place, with the above assumptions for the infection and recovery processes, the time interval for the next event is exponentially distributed with rate:

$$
\sum_{i=1}^{m'} [(k_i - 1)\lambda_i + \lambda_{mal}] + \sum_{i=m'+1}^{m} k_i\lambda_i + \sum_{i=m+1}^{N} \mu_i
$$

(2)

where $k_i$ is the number of infected neighbors of node $i$ (including the attacker if applicable), $\mu_i$ denotes its recovery rate, $\lambda_i$ denotes the link-infection rate between node $i$ and any other neighboring network node and $\lambda_{mal}$ denotes the link-infection rate between a network node and the attacker. The summation index spans the sequence of the network nodes, where without loss of generality, we assumed that the non-infected nodes that have the attacker as a neighbor are the first $m'$, the non-infected not having the attacker as neighbor follow in the sequence up to index $m$, and the rest are the infected nodes for the time instant under consideration.

V. TOPOLOGY CONTROL ALGORITHMS

Based on our previous discussion, in this section we introduce and describe in detail three different topology control algorithms. The first one is a single-threshold based topology control algorithm that uses the available energy resources to adapt the attacker’s transmission radius, while the other two are based on the risk function.

Specifically, the Threshold-based Topology Control (TTC) algorithm starts with a large transmission radius, denoted by $R_m$ and when the available energy resources drop below the threshold of 50% of the initial reserves, the algorithm switches to a smaller radius, denoted by $R_s$. We used two radii values (i.e. single-threshold) as a base case, even though the algorithm can be easily extended to a multi-threshold approach. The intuition behind TTC is that as long as the attacker has enough energy, it can use a large radius to increase its node degree as much as possible and therefore increase the number of nodes that can be directly infected. However, when its available energy reduces below a certain threshold, it is more efficient to reduce its radius to conserve energy and extend its lifetime at the cost of smaller node degree and consequently smaller instantaneous network damage.
if (RF\textsubscript{curr} > RF\textsubscript{max})
\[ RF\textsubscript{max} := RF\textsubscript{curr}; \]
else if (RF\textsubscript{curr} < RF\textsubscript{min})
\[ RF\textsubscript{min} := RF\textsubscript{curr}; \]
if (|RF\textsubscript{curr} - RF\textsubscript{min}| > |RF\textsubscript{curr} - RF\textsubscript{max}|)
\[ r := r - dr; \]
else if (|RF\textsubscript{curr} - RF\textsubscript{min}| < |RF\textsubscript{curr} - RF\textsubscript{max}|)
\[ r := r + dr; \]

Fig. 2. RFTC Algorithm

The operation of the risk-based topology control algorithms is driven by the risk function, $RF$. Ideally an attacker could utilize its maximum possible transmission radius to increase $RF$ as much as possible. However, this would lead to the quick exhaustion of its energy resources and the reduction of its lifetime. Thus, an efficient risk-based topology control algorithm should attempt to balance the demand for high risk with the need for conservative energy consumption.

The Risk Function-based Topology Control (RFTC) is a two stage algorithm that adjusts the measured value of the risk function. In a realistic operational environment, an attacker does not have prior knowledge of the $RF$ value interval. Thus, predefined thresholds on the measured $RF$ cannot be applied here for correct decision making. As a result, the first stage of RFTC is to adjust the lower and upper value bounds of $RF$, so that the attacker has up-to-date knowledge of the interval that $RF$ belongs to. The next step is to determine, based on the current value of $RF$, whether this value is closer to the upper bound and needs to be reduced or closer to the lower bound and needs to be increased. The adjustment of $RF$ can be done implicitly. That is, by increasing the transmission radius $r$ the local neighborhood of the node increases, leading to higher values of $RF$. Conversely, decreasing $r$, decreases the attacker’s node degree, leading eventually to lower values of $RF$. The operation of RFTC is shown in Fig. 2, where $RF_{\text{min}}$, $RF_{\text{max}}$, and $RF_{\text{curr}}$ is the minimum, maximum and current measured values of the risk function respectively. Parameter $dr$ denotes the step value (constant for RFTC) by which the attacker’s transmission radius is adjusted at each step as long as its minimum or maximum values are not reached.

An immediate enhancement to the operation of RFTC refers to the discipline used to increase or decrease the step value of the attacker’s transmission radius. Risk Function-based Topology Control-Enhanced (RFTC-E) algorithm takes into account the remaining energy resources when determining the level (i.e. step value) of radius adjustment (increase/decrease) required. When the available energy is more than 50% of the initial reserves, RFTC-E increases $r$ by $dr$ and decreases it slowly by $dr/2$. When the remaining energy reserves are less than 50% of the initial energy, the attacker increases slowly the radius by $dr/2$ and decreases it quickly by $dr$.

VI. Performance Evaluation

In this section we evaluate and compare the performances of the three different algorithms and respective attack strategies presented in the previous section, in terms of the expected number of infected nodes and the infection efficiency. In order to better demonstrate the improvements achieved by these strategies and reveal their corresponding tradeoffs, we also compare them against a flat radius strategy where the attacker uses a single transmission radius value throughout its lifetime. We study two variations of the flat radius strategy, the Flat Small Radius (FSR) and the Flat Large Radius (FLR). In FSR the attacker uses the minimum radius used in the other schemes, $R_s$ and in FLR it uses the maximum value, $R_m$.

Throughout our study two different topologies are considered. In the first one (type I) all nodes, including the attacker, are assumed to have a common infection rate, while in the second one (type II) the attacker is assumed to have a higher infection rate than that of the rest of the nodes. In the following we first describe the simulation model used and then present the corresponding numerical results and some relevant discussion.

A. Simulation Model and Assumptions

For the purposes of our study we consider a wireless ad hoc network, where the nodes are uniformly and randomly distributed over a square area of $1500m \times 1500m$ and each network node has a fixed transmission range of $R = 200m$.

The attacker is assumed to move around the network according to the random walk model with wrapping and zero pause time [6]. The speed of the attacker is randomly chosen in the interval $(0, v_{\text{max}})$ and its direction is uniformly distributed within the interval $(0, \theta_{\text{max}})$. We allow nodes to move along all possible directions by setting $\theta_{\text{max}} = 2\pi$. In practice, this model does not differ from that with reflection, as both of them yield a uniform steady state location distribution [6].

It has been demonstrated that most of a wireless node’s energy consumption occurs during the communication processes, while the processing and idle power consumptions are relatively small compared to the transceiver’s needs, [7]. Therefore, in this paper we assume that the lifetime of the malicious node depends only on its available energy and transmission radius. Considering a specific time interval and two wireless nodes with different transmission radii, the energy consumption will be higher for the node with the higher radius, and as a result its lifetime becomes smaller. In fact, the greater the difference in the radii, the greater the consumption difference will be. In this paper, we model the energy depletion rule through the following iterative process:

\[
E_{\text{avail.}} = E_{\text{prev.}} - C(t_{\text{sp}}^\alpha)E_{\text{step}}
\]

(3)

where $E_{\text{avail.}}$ is the updated energy inventory, $E_{\text{prev.}}$ is the inventory prior to update, $E_{\text{step}}$ denotes a depletion step value, $r$ is the attacker’s transmission radius, $t_{\text{sp}}$ denotes the time interval spent since the last energy update, $\alpha$ is the path loss constant and $C$ is a normalization constant chosen such that
the depletion rate ensures sufficient simulation time for the system to reach its steady state.

The initial energy reserve was set to $E = 150$ energy units (eu), the step depletion value to $E_{\text{step}} = 0.05\text{eu}$ and the path loss constant was assumed $\alpha = 2.5$. The attacker’s radius varies within the range of a small radius $R_s = 200m$ and a large one $R_m = 300m$ with $dr = 10m$. The maximum speed for the mobile attacker was set to $v_{\text{max}} = 15m/\text{tu}$ (where $m/\text{tu}$ denotes meters per time units), and the recovery rate was chosen to be common for all network nodes, $\mu = 0.9$. For the results presented, with respect to type II topologies, the attacker’s infection rate was $\lambda_{\text{mal}} = \lambda + 0.2$, where $\lambda$ is the corresponding network node infection rate.

### B. Results and Discussion

In Fig. 3, the expected number of infected nodes as a function of the network size is presented for type I topologies, while Fig. 4 shows the corresponding results for type II topologies. It should be noted that since the area of the deployment region is fixed, the network density is proportional to the number of network nodes. These two figures present comparative results for algorithms FSR, FLR, TTC and RFTC.

Specifically, Fig. 3 shows that, as expected, FLR is an upper bound to the number of infected nodes. The attacker uses the maximum allowed $r$, which means that its neighborhood has the maximum possible cardinality throughout its lifetime. However, this is done at the expense of smaller lifetime, which also reduces the attacker’s efficiency. Similarly, FSR is a lower bound on the expected number of infected nodes, since the attacker’s neighborhood will have its smaller cardinality for the duration of its lifetime.

As far as the topology control algorithms are concerned, the expected number of infected nodes is smaller for RFTC compared to TTC, and for higher values of the network density. However, this does not imply that RFTC’s performance is inferior to TTC’s, since the smaller expected number of infected nodes is a result of the more efficient energy conservation approach followed by RFTC. In particular, RFTC is designed to balance the attack strategy tradeoffs (number of infected nodes and energy consumption) and use the appropriate radius for every region, so that it increases the network damage without expending unnecessary resources. Thus, in areas of higher density, the attacker is expected to use moderate values of the transmission radius, and rely on the network nodes to propagate the infection with smaller energy cost for itself.

In Fig. 4, we observe that similar results hold for type II topologies. FSR is again a lower bound on the number of expected infected nodes. However, FLR is no longer a global upper bound. For higher values of the network density, RFTC produces larger number of infected nodes than all other algorithms. For lower to moderate network densities TTC has similar performance with FSR, and performs better as the network density increases.

As explained before, the risk function can be used by the attacker for topology control purposes via real-time adaptation of its transmission radius. On the other hand, the infection efficiency is a post-processing metric we use to assess the overall performance of each attack strategy. In Fig. 5 and Fig. 6 we present the comparison for the four strategies used so far, for a network with $N = 130$ nodes. Fig. 5 presents the corresponding results for type I topologies (i.e. the case of common $\lambda$ for all nodes including the attacker), whereas Fig. 6 deals with type II network topologies. $I_E$ is indicative of the network unavailability achieved by the attacker and characterizes its combined operation of infecting nodes (making them non-operational) for the duration of its lifetime.

With respect to $I_E$, for type I topologies, as observed from Fig. 5, RFTC is the most successful strategy, outperforming all the rest for all the different values of the infection rate. TTC is better than FLR and FSR for low to moderate infection rates, while the latter becomes better than FLR and TTC for higher network densities. FLR has the worst performance, which means that pumping up transmission power as much as possible increases temporarily the number of infected nodes at the expense of reduced lifetime. As a result, the combined effect yields low overall performance. On the other hand, RFTC exploits the special topological characteristics of its local neighborhood to balance the energy consumption with the potential network damage, so as to increase the attacker’s operational window without sacrificing its effectiveness. The same trend is observed for type II topologies. In that case (Fig. 6), RFTC appears more effective even for smaller values of the common infection rate, $\lambda$. This means that even in networks,
RF will not be decreased beyond a point where the attacker is towards a point where the attacker would have larger values, and thus the measured R_{F,t} will decrease R_{F,t} will decrease r in order to decrease K_{a} and as a result R_{F,t} will be reduced as well. Due to its dynamic operation, RFTC will decrease r gradually to a moderate value of R_{F,t}, where r is suitable for preserving network damage and saving energy resources at the same time. However, as network densities further increase, r will not be decreased beyond a point where the attacker will lose significant number of neighbors, because that would decrease R_{F,t} towards R_{F, min}.

In Fig. 7, we compare the performances of RFTC and RFTC-E according to I_{E}. As already mentioned, the difference in the operation of the two algorithms is that RFTC-E uses a more efficient adaptation technique, by taking into account the available energy resources in determining the appropriate transmission radius. RFTC-E adapts in a more intelligent way to the topological changes, thus achieving higher values of I_{E} by more effectively combining the potential network damage with extended lifetime.

VII. CONCLUSIONS

In this paper, we studied the problem of mobile attack propagation in wireless ad hoc networks. Our approach takes on the attacker’s perspective, in order to investigate the extent of its attack potentials, which in turn could be used for the effective design of network countermeasures.

We first introduced the concept of the risk of a node that can be used for realizing the potentials of an attack in a network region. Based on this concept we defined the risk function as a relative measure and used it to design topology controlled attack strategies. Our performance evaluation indicated that the risk function-based algorithms achieved high infection efficiency, by effectively balancing the tradeoffs between the potential network damage and the attacker’s lifetime.

In this work, we assumed a static underlying network, in order to better reveal how the mobile malicious node’s operational characteristics affect the attack propagation process. Our future work includes the study of the corresponding problem in a fully mobile network, as well as the investigation of the use of topology control by both the attacker and network nodes in a non-cooperative game.

ACKNOWLEDGMENT

This work was partially supported by Greek General Secretariat for Research and Technology of the Ministry of Development (PENED project 03ED840).

REFERENCES